Aus dem Lehrstuhl für Bioinformatik

der Bayerischen Julius-Maximilians-Universität Würzburg

Direktor: Professor Dr. med. Thomas Dandekar

Die SVM-gestützte Prädiktabilität der Bindungsspezifität von SH3-Domänen anhand ihrer Aminosäuresequenz

Inaugural-Dissertation

zur Erlangung der Doktorwürde der

Medizinischen Fakultät

der

Julius-Maximilians-Universität Würzburg

vorgelegt von

Franz Axmacher

aus Nürnberg

Würzburg, im Juli 2014

Referent:	Prof. Dr. med. Thomas Dandekar
Korreferent:	Prof. Dr. Dr. Dipl. Phys. Wolfgang Bauer
Dekan:	Prof. Dr. med. Matthias Frosch
Tag der mündlichen Prüfung:	21.04.2015

Der Promovend ist Arzt

Für meine Eltern

Inhaltsverzeichnis

1 Einleitung

Protein-Proteininteraktionen sind zentrales Element vieler biologischer Prozesse [103]. Sie finden sich in allen strukturgebenden Elementen einer Zelle, wie Aktinfilamenten oder Mikrotubuli, sowie in vielen enzymatischen Prozessen [142], beispielsweise der Transduktion von Signalen innerhalb von Zellen. Die Interaktionen werden dabei von Untereinheiten der Proteine, den Domänen, reguliert, welche evolutionär konservierte, strukturelle sowie funktionelle Einheiten eines Proteins im Mittel aus ca. 100 Aminosäuren darstellen [152].

Eine wichtige Rolle hierbei spielen die sogenannten Src homology domains 2 und 3 (SH2 bzw. SH3), von denen sich in der Forschung die SH3-Domäne zu einer Art Vorzeigedomäne im Bereich der Proteininteraktion entwickelt hat, da sie nicht nur eine der ersten identifizierten Interaktionsdomänen von modular aufgebauten Proteinen war, sondern auch eine der häufigsten überhaupt darstellt [93]. Sie findet sich sowohl in Einzell- wie auch höher entwickelten Eukaryoten innerhalb verschiedenster Proteine, selbst in solchen, die sonst keine weiteren Homologien aufweisen [104].
Bereits im Jahr 2000 wurde in einer Arbeit von Rubin et al. zum Vergleich der Genome von Eukaryoten beschrieben, dass 63 Proteine von Drosophila, 55 von Caenorhabditis elegans und 25 von Saccharomyces cerevisiae mindestens eine Variante der Domäne enthalten [121]. Heute sind in Proteindatenbanken wie PFAM [8] bereits über 12´000 verschiedene SH3-Domänen beschrieben.

Somit scheint es kaum verwunderlich, dass sie immer wieder als Forschungsgrundlage für neue Methoden mit dem Ziel Bindungspartner zu identifizieren oder neue Liganden chemisch zu erzeugen diente [93]. Teil dieser Forschungsarbeiten war es die verschiedenen SH3-Domänen nach ihren Liganden zu klassifizieren, um hierdurch zum einen Domänen mit ähnlicher Funktion identifizieren zu können [33] und zum anderen Voraussagen über die Bindungsspezifität von neu gefundenen bzw. noch unklassifizierten SH3-Domänen machen zu können. Dabei wurde nach bestimmten sequenziellen Gemeinsamkeiten der identifizierten Liganden gesucht, nach welchen sie sich in verschiedene Klassen und Unterklassen zusammenfassen ließen. In weiteren Schritten wurden nun die zu einer Klasse gehörenden SH3-Domänen auf strukturelle sowie sequenzielle Ähnlichkeiten hin analysiert, um so mögliche Ansätze entwickeln zu können, welche Aussagen über die Klassenzugehörigkeit noch unklassifizierter SH3-Domänen erlauben [17]. Andere Ansätze befassten sich mit der Identifikation und Annotation von Proteindomänen, sowie der Analyse des Aufbaus von Proteinen bzw. ihren Domänen und können so auch bezüglich der SH3-Domäne zur Entdeckung weiterer Domänen bzw. Bindungspartner von SH3-Domänen beitragen. Beispiele hierfür sind unter anderem die Research Tools SMART [82][128], ARB [87], SEALS [147] oder REP [4]. Des Weiteren wurden beispielsweise positionsspezifische Interaktionsprofile von SH3-Domänen einer Klasse mit ihren Liganden erstellt, welche dann in weiteren Analysen Methoden des maschinellen Lernens, wie z.B. künstlichen neuronalen Netzen (ANN) oder Support-Vector-Machines (SVM), als Grundlage dienten, um auf diese Weise bislang noch unbekannte Bindungspartner von SH3-Domänen bestimmter Klassen identifizieren zu können [14][41][57][58][165]. In wieder anderen Studien wurden statt der reinen kontaktbasierten Interaktionsprofile z.B. „Molecular Interaction Energy Components“ (MIECs) genutzt, um das Interaktionsmuster zwischen den Domänen und ihren Liganden zu charakterisieren, welche dann ebenso für Untersuchungen anhand Methoden des maschinellen Lernens herangezogen wurden [57][58][59].
In dieser Arbeit sollte nun ein Ansatz geprüft werden, welcher ohne komplexere Struktur- bzw. Interaktionsdaten auskommt und ausschließlich die Aminosäuresequenzen der SH3-Domänen nutzt, um anhand von *Support-Vector-Machines* (Kapitel 2.2) Aussagen bzw. verbesserte Vorhersagen über ihre Klassenzugehörigkeit machen zu können.

1.1 Etymologie und Geschichte des Begriffs „*SH3*“

1966 erhielt Francis Peyton Rous den Nobelpreis für Physiologie oder Medizin für seine Entdeckung der interindividuellen Übertragbarkeit von bestimmten Tumoren durch ein filtrierbares Agens. Er transferierte zellfreies Extrakt aus Fibrosarkomzellen von Hühnern in gesunde Hühner, welche hierauf ebenso ein Fibrosarkom entwickelten [118]. Das hierbei kanzerogen wirkende Agens stellte sich später als Retrovirus heraus, welcher fortan als Rous Sarcoma Virus (RSV) bezeichnet wurde. 1970 konnte G. Steven Martin das Gen v-src (virales Sarkom) des Virus als ursächlich hierfür identifizieren [135].

Im Jahre 1976 entdeckten J. Michael Bishop und Harold E. Varmus in gesunden Hühnern ein strukturell dem v-src stark ähnelndes Gen, welches Teil des normalen Erbguts dieser war und eine zentrale Rolle in Zellwachstum und Zellteilung einnahm. Es wurde daher geschlussfolgert, dass das in RSV gefundene Gen v-src kein eigentlich virales, sondern ein während der Evolution zufällig in den Virus mit aufgenommenes Wirt-Gen sei, welches im Laufe der Zeit onkogen mutiert war und bei einer RSV-Infektion nun ursächlich für die Krebsentstehung sei. Das zuerst in diesen Hühnern gefundene physiologische Homolog von v-src wurde c-src (zelluläres src) genannt. Die Erkenntnisse führten zur Annahme von sogenannten Protoonkogenen als natürlich vorkommende Gene im Erbgut, welche bei Mutation kanzerogene Potenz entwickelten [99]. Bis heute konnten zahlreiche solcher Protoonkogene identifiziert werden [25].

Das Genprodukt von c-src ist die protoonkogene, ubiquitär vorkommende Tyrosin-Proteinkinase Src (pp60c-src), ein zytoplasmatisches Enzym, welches nach Myristoylierung/Palmitoylierung am N-terminalen Ende mit der Zellmembran assoziiert vorliegt. Es besteht neben der katalytischen Untereinheit (Src Homologie 1 Domäne, kurz SH1), sowie einer N-terminalen variablen Region und einem flexiblen, Tyrosin beinhaltenden C-terminalen Abschnitt aus zwei weiteren konservierten Domänen, welche Src Homologie 2 (SH2) und Src Homologie 3 (SH3) genannt werden (Abb. 1.1).
Man nimmt an, dass die SH2- und SH3-Domäne an der Regulation von c-Src beteiligt sind. Dabei spielen reversible Phosphorylierungsprozesse eine zentrale Rolle. Bei Inaktivierung des humanen Proteins kommt es nach Phosphorylierung eines im C-terminalen Ende von c-Src enthaltenen Tyrosinrests (Tyr 527) durch Bindung dieses Rests an die SH2-Domäne zur Konformationsänderung des Enzyms. Stabilisierend bindet hierbei die SH3-Domäne an einen prolinreichen Sequenzabschnitt zwischen der SH2-Domäne und der Kinase. Durch Dephosphorylierung von Tyr 527 kehrt das Enzym wieder in seinen aktiven Zustand zurück [19]. Die virale Variante der Tyrosin-Proteinkinase Src (pp60v-src) unterscheidet sich von seinem natürlichen Homolog unter anderem durch das Fehlen des Tyrosinrests an Position 527 [135]. Hierdurch erklärt sich auch deren permanent aktiver Zustand, der eine pathologische Aktivierung diverser Proteine (unter anderem STAT3 [109], Cyclin D1 [113] und HIF-1α [132]) nach sich zieht, welche letztlich auch für die tumor- bzw. angioproliferativen Prozesse bei RSV-Infektion mitverantwortlich sein dürfte.

Strukturell und sequenziell der Src Homologie 3 Domäne von c-src sehr ähnliche Proteindomänen, ließen sich im Folgenden in unzähligen weiteren Signalproteinen nachweisen. Einige der ersten Proteine, in denen diese Region ähnlicher Sequenz auffiel, waren das virale Adapterprotein v-Crk, sowie die Phospholipase C-γ. [93][142]. Die Bezeichnung SH3-Domäne (kurz für Src Homologie 3) leitet sich jedoch vom Gennamen v-Src der Tyrosinkinase des Rous-Sarkoma-Retrovirus ab [33].

1.2 Vorkommen und Funktion der SH3-Domäne

SH3-Domänen sind in der Natur nahezu ubiquitär innerhalb unterschiedlichster eukaryoter Organismen verbreite, hoch konservierte Proteininteraktionsmodule innerhalb unterschiedlichster Proteine, selbst solcher, welche ansonsten keinerlei Gemeinsamkeiten aufweisen [104]. Ihr Vorkommensspektrum reicht vom Einzeller, wie dem Hefepilz, bis zu komplexen Organismen, wie dem Menschen. Dies indiziert eine weit zurückreichende evolutionäre Geschichte und suggeriert eine zentrale Rolle der Domäne bei vitalen zellulären Prozessen [106]. Die Hauptaufgabe von SH3-Domänen liegt dabei in der Interaktion von Proteinen, so spielen sie vor allem bei der Aggregation von Proteinen und bei regulativen zellulären Prozessen eine wichtige Rolle [164]. Sie sind beispielsweise an der Organisation des Zytoskeletts beteiligt und finden sich hierbei in zahlreichen Proteinen, wie den Aktin bindenden Proteinen Spektrin, Abp1 von S. cerevisiae oder Cortactin. Auch bei der Zellpolarität scheinen SH3-Domänen über Interaktion mit kleinen GTP-bindenden Proteinen der Ras-Familie eine wichtige Rolle einzunehmen. In höher entwickelten Eukaryoten sind sie außerdem an der Regulation und Aggregation von Proteinen innerhalb vieler Signaltransduktionswege und somit an zentralen Prozessen wie Zellproliferation und -teilung, Immunregulation oder Endozytose beteiligt [164][80]. Sie können hierbei als Untereinheiten von Enzymen, wie im Falle von Tyrosinkinasen der src-Familie oder im Falle von PLC-γ auftreten oder als Teil von Adapterproteinen wie z.B. Grb2 oder Nck fungieren, welche interessanterweise zum Teil ausschließlich aus einer Kombination von SH2- und SH3-Domänen bestehen können [106].

Exemplarisch für die Beteiligung von SH3-Domänen an der Aggregation von Proteinen soll hier ihre Funktion innerhalb des Proteins Grb2 bei der p21 Ras-Protein abhängigen *Epidermal Growth Factor* (EGF) Signalkaskade erläutert werden. Grb2 ist ein Adapterprotein, welches aus einer SH2-Domäne flankiert von zwei SH3-Domänen besteht [50]. Nach Aktivierung und Autophosphorylierung des EGF-Rezeptors (EGFR) sowie Phosphorylierung von anderen membranassoziierten Proteinen kann Grb2 mit seiner SH2-Domäne, welche Bindungsaffinität für Phosphotyrosin-Motive besitzt, an den Rezeptor binden. Die beiden SH3-Domänen in Grb2 binden an spezifische, prolinreiche Motive des zytosolischen Proteins Sos-1, welches der *Guanine Nucleotide Exchange Factor* des Ras-Proteins ist. Hierdurch kommt es zur Anlagerung von Sos-1 an die Zellmembran, wo es den Wechsel der GDP- zur GTP-gebundenen Form des zellmembranständigen Ras-Proteins und somit dessen Aktivierung katalysiert. Die aktive Form des Ras-Proteins induziert nun ihrerseits einen MAP-Kinase-Weg (MAP = mitogen-activated protein), der letztendlich eine Vielzahl verschiedener regulativer Moleküle aktiviert, um Prozesse wie Zellproliferation, -differenzierung und -entwicklung zu initiieren, aber auch zu koordinieren, wie beispielsweise die Aktivierung von Thrombozyten (Abb. 1.1) [45][94][131][162][164].

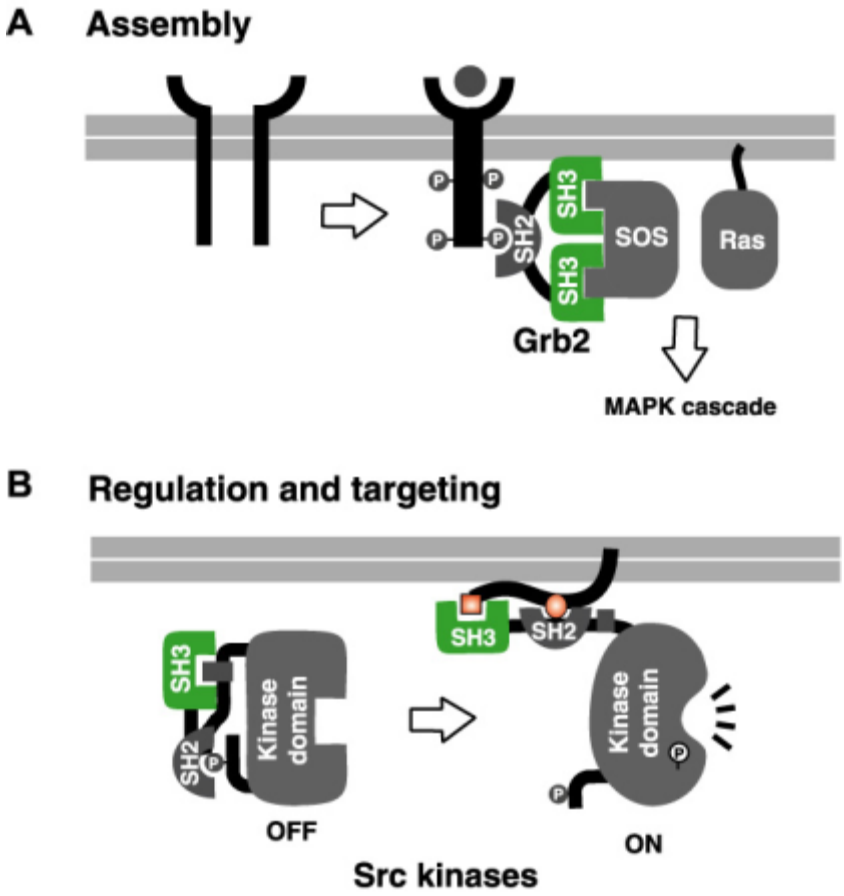


Abb. 1.1 Unterschiedliche Aufgaben der SH3-Domäne. (A) Schematische Darstellung der Beteiligung von SH3-Domänen an der Aggregation von Proteinen am Beispiel ihrer Funktion innerhalb des Proteins Grb2 bei der p21 Ras-Protein abhängigen EGF-Signalkaskade. Nach Aktivierung und Autophosphorylierung des EGF-Rezeptors (EGFR) bindet Grb2 mit seiner SH2-Domäne an den Rezeptor, während die beiden SH3-Domänen von Grb2 jeweils an spezifische, prolinreiche Motive des zytosolischen Proteins Sos-1 binden. Hierdurch kommt es zur Anlagerung von Sos-1 an die Zellmembran, wo es den Wechsel der GDP- zur GTP-gebundenen Form des zellmembranständigen Ras-Proteins und somit dessen Aktivierung katalysiert. (B) Schematische Darstellung der Beteiligung von SH3-Domänen bei regulativen Prozessen am Beispiel von Tyrosinkinasen der src-Familie, welche unter anderem etwa in Thrombozyten eine wichtige Rolle spielen [32]. Während der inaktiven Konformation der Kinase ist der phosphorylierte Tyr 527-Rest an die SH2-Domäne und die SH3-Domäne an einen prolinreichen Sequenzabschnitt zwischen der SH2-Domäne und der Kinase gebunden. Durch Dephosphorylierung von Tyr 527 (nicht dargestellt) bzw. über eine Bindung externer Liganden an die SH2- bzw. SH3-Domäne wird das Enzym aktiviert.
(Aus Zarrinpar A., et al., The Structure and Function of Proline Recognition Domains [164])

Ein gutes Beispiel für die Rolle von SH3-Domänen bei regulativen Prozessen ist ihre bereits erwähnte Funktion in der Aktivitätssteuerung von Tyrosinkinasen der src-Familie. Während der inaktiven Konformation dieser Tyrosinkinasen, welche neben der Kinase-Domäne aus einer SH2- und SH3-Domäne bestehen, sind diese an einem im C-terminalen Ende des Proteins enthaltenen Tyrosinrest (Tyr 527) phosphoryliert. Die hierdurch ermöglichte Bindung dieses Restes an die Phosphotyrosin-Bindungsstelle der SH2-Domäne bewirkt eine Konformationsänderung des Enzyms, welche durch die Bindung der SH3-Domäne an einen prolinreichen Sequenzabschnitt zwischen der SH2-Domäne und der Kinase stabilisiert wird. Die Aktivierung des Enzyms ist nun sowohl über die Dephosphorylierung von

Tyr 527 [19], als auch über eine Bindung externer Liganden an die SH2- bzw. SH3-Domäne möglich (Abb. 1.1) [164].

Neben ihrem Vorkommen nahezu ubiquitär in verschiedensten Lebensformen, erklärt sich das hohe wissenschaftliche und medizinische Interesse an der Erforschung von SH3-Domänen also auch durch ihre vielfältige Beteiligung an verschiedensten zentralen zellulären Prozessen.

1.3 Strukturelle Eigenschaften von SH3-Domänen

SH3-Domänen sind ca. 30 Å große Proteinuntereinheiten, welche typischerweise aus einer Sequenz von ca. 50 bis 75 Aminosäuren bestehen. Trotz begrenzter sequentieller Homologie der verschiedenen SH3-Domänen, ist ihr dreidimensionaler Aufbau (Tertiärstruktur) zumeist sehr ähnlich [106][114]. In strukturellen Analysen von Noble et al., (1993) zeigte sich die root mean square (rms) Deviation zwischen den α-Kohlenstoffen der SH3-Domänen von Spectrin und Fyn bei nur 1.1 Å [110]; analog hierzu wurde in einer Arbeit von Koyama et al., 1993, die rms Deviation der Hauptkette in der konservierten Region der SH3-Domänen von Src und PI-3 mit nur 0.87 Å beschrieben [76].

Die Struktur von SH3-Domänen ist normalerweise gekennzeichnet durch ein aus fünf bis zehn antiparallelen Strängen bestehendes β-Faltblatt, welches sich in zwei etwa orthogonal zueinander stehende Sheets unterteilt und hierdurch eine fassartige Struktur formt. Die N- und C-terminalen Enden der Domäne liegen dabei dicht beieinander, wodurch der Einbau der Domäne in seine einzelnen ´Host`-Proteine mit nur geringen sterischen Änderungen des Gesamtproteins verbunden ist [103].
Die einzelnen β-Stränge werden (nach Noble et al., 1993 [110]) grundsätzlich mit ´Strang a` bis ´Strang e` bezeichnet, wobei Strang b in einen Strang b_1 und b_2 unterteilt wird. Strang a, b_1 und e bilden das eine und Strang b_2, c und d das andere Sheet des β-Faltblatts. Die Stränge sind untereinander durch mehrere, hochvariable, teils aus α-Helices, teils aus 3_{10}-Helices und teils aus gecoilten Sequenzabschnitten bestehende loops miteinander verbunden, von denen nach Noble et al., 1993 [110] der zwischen Strang a und b_1 als RT-loop, der zwischen Strang b_2 und c als n-Src-loop und der zwischen Strang c und d als ´distal loop` bezeichnet werden (Abb. 1.2) [106][114]. Der n-Src- und RT-loop bilden dabei die äußere Rahmenstruktur der Liganden-Bindungsstelle von SH3-Domänen und tragen durch ihre hohe Sequenz- und Längenvariabilität, mit den daraus resultierenden variablen strukturellen und chemischen Eigenschaften dieser Strukturen, signifikant zur Bindungsspezifität der Domänen bei [39].
Bei der Liganden-Bindungsstelle selbst handelt es sich um einen relativ ebenen Bereich an der Oberfläche der Domäne, welcher von drei kleinen hydrophoben Taschen gebildet wird (Abb. 1.3). Diese hydrophoben Taschen bestehen typischerweise aus hochkonservierten aromatischen Aminosäureresten, welche die Bindungsaffinität von SH3-Domänen zu prolinreichen Liganden gut erklären [93].

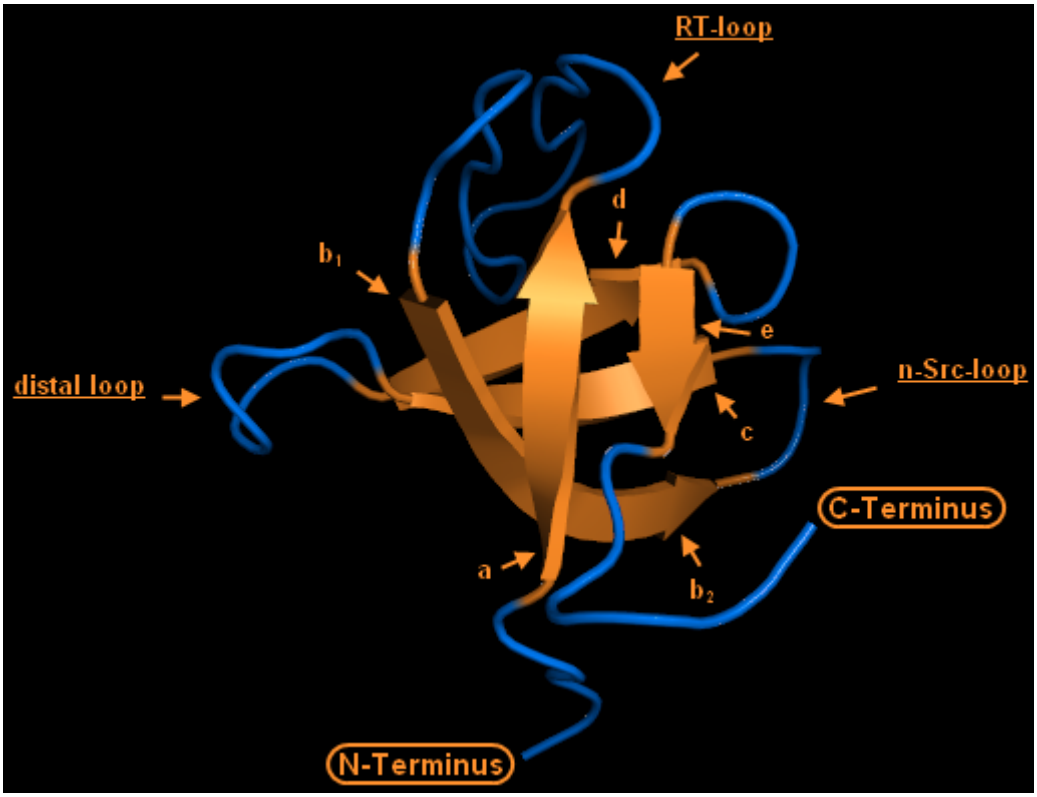


Abb. 1.2 Schematische Darstellung einer SH3-Domäne mit der typischen Anordnung der β-Faltblätter (orange Abschnitte a bis e) und den in blau dargestellten, hochvariablen α-Helices, genannt RT-loop, n-Src-loop und distal loop.
(Darstellung mithilfe von PyMOL [28][139] auf Basis der PDB-Datei „2LCS.pdb“ [10][75])

1.4 Bindungseigenschaften der SH3-Domäne

Trotz ihrer hohen strukturellen Homologie und der ihnen fast allen gemeinen Affinität zu polyprolinhaltigen Liganden, weisen SH3-Domänen doch eine gewisse Spezifität für ihre Bindungspartner auf [17]. Um die genauen Vorgänge bei der Bindung von SH3-Domänen an ihre Liganden besser verstehen zu können und konsekutiv möglicherweise sogar potentielle Bindungspartner vorhersehen zu können, wurden hierzu in den letzten Jahren zahlreiche Studien durchgeführt [17][39][71][93]. So ist heute bekannt, dass das Kernmotiv der meisten SH3-Liganden aus zwei xP-Dipeptiden besteht (x ≙ zumeist hydrophobe Aminosäure, P ≙ Prolin), welche durch eine weitere Aminosäure (häufig ebenso Prolin) voneinander getrennt sind. Zwei der hydrophoben Taschen in der Bindungsstelle von SH3-Domänen werden durch diese Dipeptide des Liganden besetzt, während die dritte Tasche, welche in vielen Fällen von negativ geladenen Aminosäureresten umgeben ist, meist von einer positiv geladenen Seitenkette (häufig R ≙ Arginin) proximal oder distal des Kernmotivs ´xPxxP` besetzt wird (Abb. 1.3) [17]. Der Ligand nimmt dabei eine typische links-helikale Konformation an. Diese sogenannte ´Polyprolin Typ II`-Helix (PPII) besteht aus drei Aminosäuren pro Drehung und bildet so im Querschnitt eine dreiecksartige Struktur, welche mit ihrer Basis auf der

Bindungsstelle sitzt, sodass die beiden Prolin-Dipeptide und die positiv geladene Seitenkette – jeweils an der Basis des Dreiecks lokalisiert – direkten Kontakt zu den Taschen der Bindungsstelle haben [93]. Die Lokalisation der positiven Seitenkette, das heißt C- oder N-terminal des Kernmotivs, bestimmt hierbei Orientierung des Liganden bei der Bindung. Liganden mit der Konsenssequenz +xxPxxP binden in Typ-I-Orientierung, während Liganden mit der Konsenssequenz PxxPx+ in Typ-II-Orientierung binden [17]. Dabei kommen die beiden Prolinreste, abhängig vom Typ des Liganden, jeweils an verschiedenen Stellen der Bindungstaschen zum Liegen (Abb. 1.3). Das bedeutet, dass die Positionierung, die die Prolinreste in der Bindungstasche einnehmen, nicht von wesentlicher Bedeutung sein dürfte. Diese Beobachtung lässt sich möglicherweise dadurch erklären, dass der entscheidende Grund für die Affinität von SH3-Domänen für das Kernmotiv des Liganden, bestehend aus den beiden Dipeptiden xP, an einer der strukturellen Besonderheit von Prolin liegen dürfte. Für eine effektive Bindung des Liganden an die hydrophoben Taschen scheint es in den meisten Fällen nötig, dass dieser in der Backbone-Sequenz seiner Aminosäuren im Kernmotiv ein alkyliertes Kohlenstoff- und Stickstoffatom enthält, welche nur durch ein Backbone-Kohlenstoffatom voneinander getrennt sind. Dies könnte daran liegen, dass sich hierdurch eine relativ kontinuierliche Kante bildet, welche sich gut in die Bindungstaschen einpassen kann.

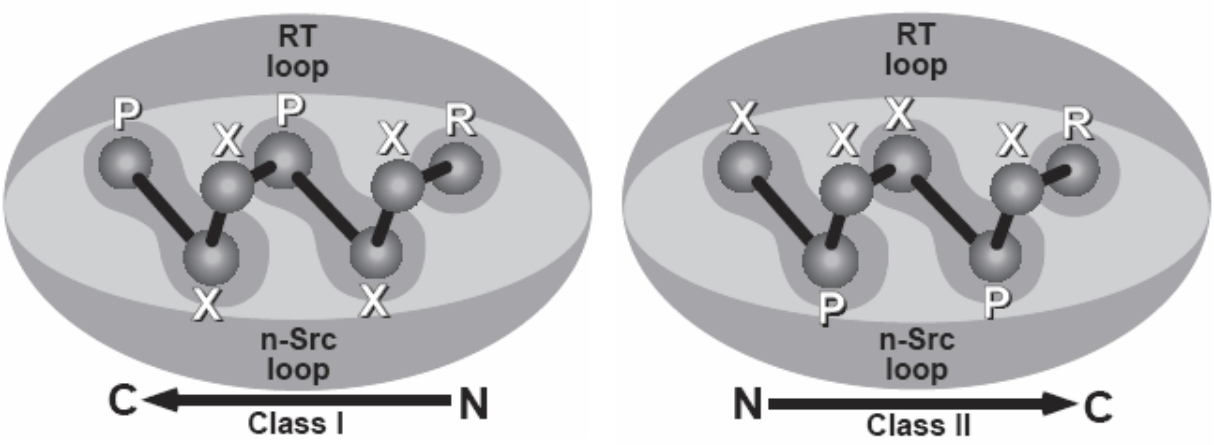


Abb. 1.3 Schematische Darstellung der Bindungsstelle von SH3-Domänen mit gebundenem Liganden in Typ-I-Orientierung (links) und Typ-II-Orientierung (rechts). Gut zu erkennen ist die dreiecksartige Struktur der PPII-Helix, die mit den beiden Prolin-Dipeptiden und der positiv geladene Seitenkette direkten Kontakt zu den Taschen der Bindungsstelle hat.
(Aus Mayer B.J., SH3 domains: complexity in moderation [93])

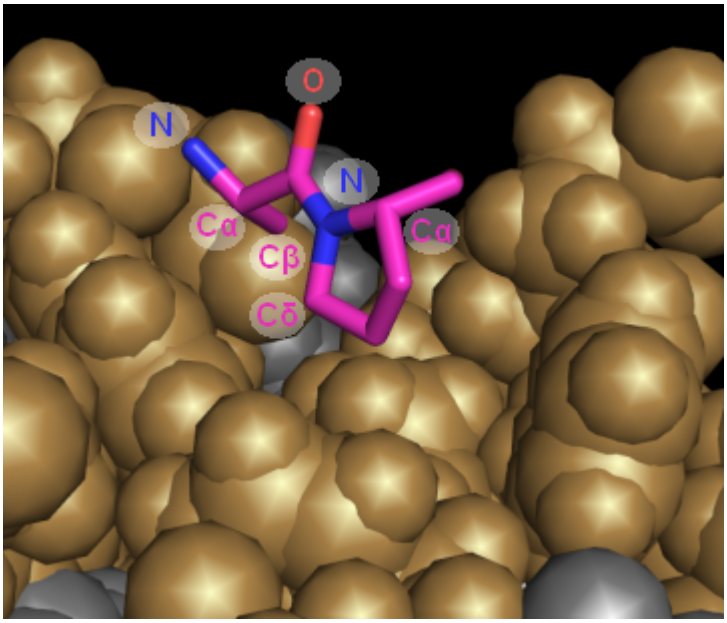


Abb. 1.4 Darstellung der mittleren Bindungstasche der SH3-Domäne NBP2 (golden in vertikaler Richtung verlaufend) mit gebundenem Dipeptid xP (pink mit blauen Aminogruppen und roter Carboxylatgruppe) in Typ I-Orientierung. Durch die alkylierten Kohlen- und Stickstoffatome in der Backbone-Sequenz des Dipeptids (N mit C_{δ}, C_{α} mit C_{β}), welche nur durch ein Kohlenstoffatom voneinander getrennt sind wird die Bindungstasche nahezu optimal ausgefüllt.
(Darstellung mithilfe von PyMOL [28][139] auf Basis der PDB-Datei „2LCS.pdb“ [10][75])

Dieses Kriterium lässt sich nur durch die Kombination einer am C_{α}-Atom alkylierten Aminosäure mit Prolin erfüllen, da Prolin die einzig natürlich vorkommende Aminosäure ist, die ein solches alkyliertes Stickstoffatom besitzt (Abb. 1.4). Diese Hypothese konnte in Versuchen von Nguyen et al., 1998 und 2000, durch den Ersatz von Prolin an den entsprechenden Stellen des Kernmotivs mit künstlichen, an gleichem Stickstoff alkylierten Aminosäuren belegt werden, wodurch sich sowohl die Affinität wie auch die Spezifität der einzelnen Domänen für diese Liganden signifikant modifizieren und zum Teil sogar steigern ließen [93][107][108].

Um jedoch die Spezifität der einzelnen SH3-Domänen für ihre jeweiligen Liganden in vivo erklären zu können, bedarf es neben der limitierten Variationsmöglichkeiten des Kernmotivs sowie der bereits erwähnten unterschiedlichen Bindungsorientierungen des Liganden noch weiterer Variablen, die bei der Selektion des Liganden eine Rolle spielen. Hierzu kommen mehrere Möglichkeiten in Frage, so existieren beispielsweise SH3-Domänen, wie die von Abl oder Crk, welche im Bereich der dritten Bindungstasche statt Arginin andere Aminosäurereste präferieren [159][119][151][93].
Einen entscheidenden Beitrag zur Selektivität scheinen auch die hochvariablen n-Src- und RT-loops zu leisten, die aufgrund ihrer unmittelbaren Nähe zur Bindungsstelle direkt mit den Liganden interagieren und so durch sterische bzw. chemische Unterschiede die Selektion des Liganden beeinflussen können. Die Interaktion mit dem Liganden findet hierbei außerhalb und zum Teil sogar weit entfernt von den eigentlichen Typ-I bzw. -II Bindungsmotiven statt. Ein Beispiel hierfür ist die Bindung des HIV Nef-

Proteins an die SH3-Domäne von Hck, da die hohe Bindungsaffinität hierbei nicht durch das Kernmotiv PxxP, sondern eine einzige Seitenkette im RT-Loop von Hck-SH3 bewirkt wird [81][93]. Durch Randomisierung der Aminosäurereste an sechs verschiedenen Positionen innerhalb des RT-loops der Hck SH3-Domäne ließen sich mutierte Domänen kreieren, die eine bis zu 40 mal höhere Affinität für das Nef-Protein aufwiesen als ihre natürliche Form. Außerdem gelang es mutierte Hck SH3-Domänen mit sehr hoher Affinität und Selektivität für eine Variante des Nef-Proteins zu kreieren, die eine Punktmutation in der Region aufwies, von der angenommen wird bei der Bindung an Hck in Berührung mit dessen RT-loop zu kommen [56][93][81].
Zudem scheinen SH3-Domänen mit hoher Affinität für Liganden zu existieren, welche statt des bekannten Kernmotivs ´xPxxP` ein anderes Motiv an ihrer Bindungsstelle tragen, so beispielsweise die SH3-Domänen der Eps8-Familie mit dem Konsensmotiv PxxDY ihrer Liganden [102] oder die SH3-Domäne von Boi2, die an Liganden mit dem Konsensmotiv PxRNPxR bindet [141].

Generell jedoch ist die Affinität und Spezifität von SH3-Domänen zu ihren natürlichen Liganden relativ niedrig, sodass SH3-tragende Proteine weiterer Mechanismen zur Steigerung ihrer Selektivität für potentielle Bindungspartner bedürfen, um spezifische biologische Effekte bewirken zu können. Eine Möglichkeit besteht im additiven Effekt multipler verschiedener Interaktionen, wie z.B. bei der Aktivitätssteuerung der Src-Kinase, oder in der Bildung von Multiproteinkomplexen. So besitzt Grb2 beispielsweise zwei SH3-Domänen, die an der Bindung von Sos-1 beteiligt sind [134][93]. Eine relativ simple Strategie die Selektivität zu beeinflussen besteht darin, das Vorkommen bestimmter SH3-Domänen bzw. potentieller Bindungspartner örtlich durch subzelluläre Kompartmentbildung zu restringieren [93].

Allerdings ist es sogar möglich, dass eine hohe Selektivität von SH3-Domänen bei der Auswahl ihrer Liganden in vivo gar nicht zwingend nötig ist, sondern eine gewisse Kreuzreaktivität sogar erwünscht ist. Ein Repertoire an Kombinationen probabilistisch möglicher Interaktionen (z.B. A bindet an U, V, W, X, Y, Z; während X, abgesehen von A, an B, C, D, E, F bindet, usw.) könnte unter Umständen sogar wesentlich mehr spezifische Information erzeugen als auf ganz spezielle Bindungspartner restringierte Interaktionen, da die Kombinationsmöglichkeiten hierbei wesentlich größer sind als die dazu benötigte Anzahl verschiedener Interaktionspartner [93].

1.5 Klassifizierung der SH-Domäne nach ihren Liganden

Um die Mechanismen der Selektivität von SH3-Domänen für ihre jeweiligen Liganden besser zu verstehen und somit die Anzahl potentieller Liganden für bestimmte SH3-Domänen zukünftig besser eingrenzen zu können, wurden in einer Arbeit von Cesareni, et al. aus dem Jahre 2002 [17] sämtliche bekannte SH3-Domänen der Spezies Saccharomyces cerevisiae, stellvertretend für SH3-Domänen allgemein, auf ihre Bindungseigenschaften hin untersucht und den sequenziellen Eigenschaften ihrer Liganden nach klassifiziert. In der Annahme, dass die Spezifität von SH3-Domänen für ihre Liganden in vivo in den meisten Fällen durch Variationen des bekannten polyprolinhaltigen Kernmotivs bestimmt wird, beschränkte man sich in dieser Arbeit auf Untersuchungen der Bindung an relativ kleine Peptide. Hierzu wurden den einzelnen SH3-Domänen randomisierte Nonapeptide aus einer Peptid-Datenbank zur Bindung präsentiert. Jeweils zehn bis zwanzig, die jeweilige SH3-Domäne bindende Peptide wurden sequenziert und ihr Konsensmotiv anhand der Häufigkeit einzelner Aminosäuren an bestimmten Positionen des Peptids ermittelt.

In einem zweiten Schritt wurden die einzelnen SH3-Domänen von Saccharomyces cerevisiae bezüglich der Konsensmotive ihrer Liganden sowohl untereinander als auch mit weiteren bereits analysierten SH3-Domänen anderer Spezies verglichen und in ein System aus acht verschiedenen Klassen eingeteilt (Tabelle 1.1). Die Konsensmotive der meisten SH3-Domänen ließen sich dabei problemlos, nach den bereits weiter oben erwähnten Kriterien, in Liganden des Typ-I (+xxPxxP) bzw. Typ-II (PxxPx+) einteilen. Der größte Anteil der Domänen präferierte Arginin an der Position der positiv geladenen Seitenkette (P-3), sodass diese Domänen den Klassen ***1R*** (Rx#PxxP) bzw. ***2R*** (Px#PxR) zugeordnet wurden (# = normalerweise hydrophobe Seitenkette). Domänen, welche an Liganden mit dem Konsensmotiv KxxPxxP bzw. PxxPxK banden, bildeten entsprechend Klasse ***1K*** bzw. ***2K***. Liganden der Klasse ***1@*** entsprechen denen vom Typ-I, tragen aber statt der positiv geladenen zumeist eine aromatische (manchmal aliphatische) Seitenkette an Position P-3. Klasse ***2D*** wurde durch die SH3-Domänen der Eps8-Familie definiert, welche Liganden mit der Konsenssequenz PxxDY binden. Die Bezeichnung *2D* ist jedoch in gewisser Weise irreführend, da letztlich nicht geklärt ist, ob Liganden dieser Klasse eine PPII-Helix bilden und in Typ-II-Orientierung binden. Einige SH3-Domänen banden Peptide mit Konsensmotiven, die keiner der beschriebenen Klassen zugeordnet werden konnten bzw. zusätzlich sehr spezifische Merkmale aufwiesen. Daher wurden diese Domänen in einer separaten Klasse zusammengefasst, die man ***X*** bzw. ***ORS*** (*Odd Recognition Specifity*) bezeichnete. Mehrere Domänen zeigten sich zudem affin für Liganden beider Bindungsorientierungen (Typ-I und Typ-II) bzw. Liganden unterschiedlicher Klassen und wurden daher mehr als einer Klasse zugeordnet. SH3-Domänen, für die keine Bindungsaffinität nachgewiesen werden konnte, wurden schließlich der Klasse ***Y*** zugeordnet [17].

Es muss jedoch erwähnt werden, dass die Klassenzugehörigkeit eines Liganden bzw. einer Domäne allein noch keine Rückschlüsse auf seine bzw. ihre physiologischen Funktionen erlaubt, da das beschriebene Klassenmodell ja nicht funktionsorientiert ist, sodass die biologischen Funktionen der verschiedenen Domänen einer Klasse durchaus äußerst unterschiedlich sein können. Exemplarisch hierfür seien die SH3-Domänen der Proteine Rvs167 und Fyn angeführt, welche beide der Klasse *1R* zugehörig sind [17]. Während Rvs167 hauptsächlich an der Organisation des Zytoskeletts beteiligt ist, wobei ihre SH3-Domäne unter anderem Bindungen mit dem „Aktin-bindenden Protein" Abp1 und dem Protein Las17p/Bee1p – einem Homolog des menschlichen „Wiskott-Aldrich Syndrome Proteins" (WASP) – eingeht [21], ist die Tyrosinkinase Fyn bzw. ihre SH3-Domäne eher als regulatives Element innerhalb zahlreicher Signalkaskaden zu verstehen. Neben ihrer autoregulativen Funktion auf Fyn selbst [15] wirkt sie unter anderem auch bei der Signaltransduktion des T-Zellrezeptors (z.B. durch Bindung an das Adapterprotein SAP) [83], der Koordination von Myelinisierungsprozessen in Oligodendrozyten (z.B. durch Bindung an das Tau-Protein) [73] oder bei der Aktivitätsregulation von Thrombozyten (durch Bindung an Glycoprotein VI) [137] mit.

Selbst in Klassen, die sich nur aus wenigen Domänen zusammensetzen – wie z.B. Klasse *1@*, findet sich ein ähnliches Bild. Auch hier existieren wieder solche Domänen, die in Konjugation mit ihren Trägerproteinen als Organisationselemente des Zytoskeletts ihre Aufgabe erfüllen, und solche, deren Hauptaufgabe eher einem regulativen Moment innerhalb von Signalkaskaden gleichkommt. So lassen sich für erstere Gruppe hinsichtlich Klasse *1@* beispielsweise die SH3-Domänen der beiden Myosin-I Proteine Myo3 und Myo5 anführen [17]. Diese binden unter anderem, wie auch Rvs167, an Las17p/Bee1p, das bereits erwähnte Homolog des menschlichen „Wiskott-Aldrich Syndrome Proteins" (WASP), aber z.B. auch an Vrp1p, das Homolog des menschlichen „WASP-interacting Proteins" (WIP), was letztendlich die Arp2/3-Komplex vermittelte Aktin-Polymerisation induziert [100]. Zudem wird der Bindung der SH3-Domäne von Myo5 an Vrp1p auch eine Rolle bei der ATP-unabhängigen Interaktion der Tailregion von Myo5 mit Aktinfilamenten zugesprochen [144][48]. Im Gegensatz hierzu ist etwa die SH3-Domäne der Tyrosinkinase Abl1, welche ebenso der Klasse *1@* zugehörig ist [17], eher wieder als regulatives Element innerhalb von Signalkaskaden zu verstehen [20]. Zwar leistet diese auch einen Beitrag zur der Organisation des Zytoskeletts – z.B. durch Bindung an das „Wiskott-Aldrich Syndrome-related Protein" WAVE2, welches nach anschließender Phosphorylierung und Aktivierung durch die Kinasedomäne von Abl1 (ebenso wie Las17p/Bee1p) die Arp2/3-Komplex abhängige Aktin-Polymerisation induziert [136], jedoch kommt Abl1 bzw. ihrer SH3-Domäne beispielsweise auch eine entscheidende Rolle bei der Regulierung von DNA-Reparaturprozessen oder der Apoptose zu [20]. So bindet – aktiviert durch DNA-Doppelstrangbrüche – beispielsweise die Serinkinase ATM an die SH3-Domäne von Abl1, was die Phosphorylierung und Aktivierung von Abl1 durch ATM zur Folge hat. Dies wiederum induziert komplexe Signalkaskaden, welche letztlich entweder zur Reparatur des DNA-Schadens oder zur Apoptose der Zelle führen [138][96].

Class	Class *consensus*	SH3 domain *consensus1*	
1R	Rx#PxxP	Rvs167	Rx#PxpP
		Nbp2	PxRPaPxxP
		Pex13	RxlPx#P
		Yhl002	yRp#PxxP
		Sla1-3,	hRxpPxpP
		Yes	RPLPxLP
		PI3Kp85	RPLPPLP
		Src	RPLPx#P
		Hck	RxLPx#P
		Lyn	RPLPPLP
		Fyn	RPLPP#P
2R	PxxPxR	Yfr024	PpLPxRP
		Ysc84	PxLPxR
		Ygr136	Px#PxRp
		Ypr154	Pp#PxRp
		PLCg	PPVPPRP
		CAP	PxPPxRxSSL
		p53BP2	RPx#P#R+
		Grb2-C	PxxPxR
1K	+xxPxxP	Sho1	s+xLPxxP
		Bzz1-1	K+xPPpxp
		Bzz2-2	++pPPPp#P
		Itk/Tsk,	YxKxPPPIP
2K	PxxPx+	Crk N	P#LP#K
		Cortactin	+PP#PxKPxWL
		Abp1	+xxPxxPx+PxW#
1@	Px@xxPxxP	Abl	PPx@xPPP#P
		Myo3	Px@pPPxxP
		Myo5	Px@pPPxxP
		Spectrin	@xPPx#P
2D	PxxDY	Eps8 and rel	PxxDY
X		Ygr136	Rx+%xlP
		Ypr154	@+RPp%%%P
		Bbc1	P+#PxRP
			R+xPxpP
ORS		Boi1	pPRxPrR#
			PxRxPxR
		Boi2	pPRnPxR#
			PxRNPxR
		Amph	PxRPxR
		End	P+RPPxP
		Fus1	RxxR(ST)(ST)(ST)L
		Bem1_1	PPxVPY
Y	No peptide selected.	Cdc25, Hof1, Yd1117w, Yar014c	

Tabelle 1.1 Klassifizierung der SH3-Domäne entsprechend des Konsensmotivs ihrer Liganden nach dem von Cesareni, et al. (2002) vorgeschlagenen Modell.
(Frei nach: Cesareni, G., et al., Can we infer peptide recognition specificity mediated by SH3 domains? FEBS Lett. 2002 Feb 20;513(1):40.)

Die einzige Klasse, die ein relativ einheitliches Bild hinsichtlich der Funktion ihrer SH3-Domänen vorweisen dürfte, ist Klasse *2D*, da sämtliche Trägerproteine der Domänen dieser Klasse derselben Proteinfamilie (Eps8) entstammen [17]. Mit Ausnahme der von Eps8R3 konnte für alle Domänen dieser Klasse entsprechend auch eine Beteiligung an der Regulation der Signalweitergabe nach EGF-Rezeptorstimulation nachgewiesen werden [112]. Dies geschieht unter anderem durch Bindung an Abi-1, was wiederum die Bildung des sogenannten Eps8-Abi-1-Sos-1 Komplexes ermöglicht und konsekutiv zur Aktivierung von Rac führt [84][112]. Obwohl das Protein Eps8R3 als einziges nicht an der Aktivierung von Rac beteiligt zu sein scheint, so konnte jedoch zumindest für seine SH3-Domäne nachgewiesen werden, dass auch diese Abi-1 bindet [112]. Dennoch dürfte sich – angesichts der ansonsten eher hohen Funktionsheterogenität innerhalb der anderen Klassen – auch das Funktionsspektrum von Klasse *2D* mit Erforschung weiterer Domänen dieser Klasse noch verbreitern, insbesondere wenn deren Trägerproteine nicht der Familie Eps8 entstammen.
Dennoch erscheint die Klassifikation von SH3-Domänen nach dem vorgestellten Modell [17] sinnvoll, da durch die Kenntnis der Klasse bzw. Bindungseigenschaften einer Domäne wesentlich gezielter nach physiologischen Bindungspartnern gesucht werden kann, welche dann wiederum durchaus Rückschlüsse auf die Funktion der entsprechenden Domäne ermöglichen können. Zudem erlaubt ein detailliertes bindungsphysiologisch orientiertes Klassenmodell präzise Einblicke in die klassenspezifischen Vorgänge während der Bindung an den jeweiligen Ligandentyp, was wiederum bei der Entwicklung medikamentöser Interventionsstrategien hilfreich sein dürfte.

1.6 Projektübersicht

Ziel dieser Arbeit war es die SH3-Domäne entsprechend dem von Cesareni, et al. (2002) [17] postulierten Modell mithilfe von Strategien des maschinellen Lernens besser zu charakterisieren. Dabei sollte vor allem untersucht werden, ob die Primärstruktur von SH3-Domänen (also die Aminosäuresequenz) als einzige Informationsquelle für Support-Vector-Machines (Kapitel 2.2) ausreichend ist, um präzise Vorhersagen über die Bindungsspezifität bzw. Klassenzugehörigkeit von SH3-Domänen nach dem Modell von Cesareni, et al. (2002) [17] machen zu können. Es musste also ein SVM-basiertes Klassifikatormodell erstellt werden, welches ausschließlich die Aminosäuresequenzen der Domänen nutzt, um Vorhersagen über ihre Klassenzugehörigkeit zu machen. Die Definition der Klassen sollte sich dabei streng an dem Modell von Cesareni, et al. (2002) [17] orientieren und wurde daher unverändert übernommen. Als Ausgangsdatensatz dienten SH3-Domänen aus den Arbeiten von Cesareni, et al. (2002) [17] und Tong, et al. (2002) [141], die bereits nach ihren Liganden klassifiziert wurden bzw. deren Bindungsspezifität bekannt war. Auf das Funktionsprinzip von Support-Vector-Machines, sowie die ihnen zugrunde liegenden mathematischen Formeln, wird in Kapitel 2.2 genauer eingegangen. Im ersten Teil der Arbeit wird die Erstellung und Validierung des Klassifikators beschrieben und gezeigt, dass valide Vorhersagen weit über der Zufallswahrscheinlichkeit möglich sind.

Im zweiten Teil werden die für den Klassifikator zur Differenzierung bedeutsamsten Positionen innerhalb der Aminosäuresequenzen anhand von *Feature Selections* (Kapitel 2.3) ermittelt und mit den Ergebnissen einer Studie von Friedrich, et al. (2006) [46] verglichen. Im dritten Teil werden die ermittelten Positionen schließlich innerhalb exemplarischer SH3-Domänen mithilfe von PyMOL [28][139] strukturell visualisiert und analysiert. Zudem werden sämtliche genutzte Aminosäuresequenzen anhand von WebLogo [26] auf klassenspezifische, sequenzielle Gemeinsamkeiten bzw. Differenzen zu den jeweils anderen Klassen hin untersucht und die Ergebnisse hieraus mit denen der Feature Selections bzw. der strukturellen Analysen verglichen.

1.6.1 Generieren eines Klassifikators und Kreuzvalidierung der Prädiktabilität

Da ein auf mathematischen Grundlagen basierender Klassifikator nicht in der Lage ist mit derart konkreten Dingen wie Aminosäuresequenzen umzugehen, war zunächst deren Umwandlung in abstrahierbare numerische Größen nötig, wofür Fisher-Scores (Kapitel 2.1) genutzt wurden. Anhand dieser mathematisch vergleichbaren Vektoren ließen sich nun verschiedene SVM-basierte Klassifikatoren generieren. Aufgrund teilweise nur geringer Anzahl an Domänen innerhalb der einzelnen Klassen, wurden zur Verminderung eines hieraus resultierenden „overfitting“ (Kapitel 2.3) des Klassifikators zudem Klassifikatoren generiert, welche neben den natürlichen noch mit künstlich emittierten Aminosäuresequenzen trainiert wurden. Zur Beurteilung der prognostischen Präzision der Klassifikatoren kamen verschiedene Methoden der Kreuzvalidierung zum Einsatz. Durch Vergleich der einzelnen Klassifikatoren bzw. Subklassifikatoren konnte letztendlich der Klassifikator mit der höchsten prognostischen Präzision ermittelt werden. Zur weiteren Analyse der Güte dieses Klassifikators wurde er schließlich mit bislang noch nicht verwendeten Sequenzen aus einer Studie von Friedrich, et al. (2006) [46] validiert.

1.6.2 Identifikation der zur rechnergestützten Vorhersage signifikantesten Aminosäurepositionen

Dieser Teil der Arbeit befasst sich mit der Identifikation der Aminosäurepositionen innerhalb der Sequenzen, welche für den zuvor erstellten Klassifikator mit der höchstmöglichen prognostischen Präzision zur Differenzierung von besonderer Bedeutung waren. Hintergedanke hierbei war die Frage, ob zur Differenzierung der einzelnen Klassen wichtige Positionen möglicherweise auch biologisch Einfluss auf die Bindungsspezifität der Domänen haben. Hierzu war es nötig den Klassifikator einer Reihe von Feature Selections zuzuführen. Da dieser aus mehreren, arbiträr aufeinander folgenden Subklassifikatoren bestand, mussten die Feature Selections an allen Subklassifikatoren einzeln erfolgen. Nach ausreichend häufiger Repetition der Feature Selections kristallisierten sich schließlich hinsichtlich jedes Klassifikationsschritts die Positionen heraus, welche zur Differenzierung bezüglich des

betreffenden Schritts am signifikantesten sind. Die Ergebnisse hieraus wurden nun in einem weiteren Teilschritt graphisch visualisiert und mit denen einer Studie von Friedrich, et al. (2006) [46] verglichen, in welcher unter anderem die Aminosäurepositionen verschiedener SH3-Domänen hinsichtlich ihres Kontakts mit dem Liganden analysiert wurden.

1.6.3 Analyse der zur Klassifikation signifikantesten Aminosäurepositionen im Hinblick auf ihre Rolle bei der Bindungsselektivität

Die im zweiten Teil der Arbeit identifizierten Positionen wurden nun zur strukturellen Analyse graphisch mithilfe von PyMOL [28][139] innerhalb ausgewählter Beispieldomänen visualisiert und hinsichtlich ihrer ihre Rolle bei der Bindungsselektivität der Klassen untersucht. Zusätzliche Information lieferten hierbei klassenspezifische Sequenzanalysen in Form komparativer Sequenzlogos anhand von WebLogo [26]. Es ließ sich zeigen, dass Positionen mit Einfluss auf die Ligandenpräferenz der Domänen auch hinsichtlich der SVM-basierten Klassifikatoren von besonders großer Signifikanz zu sein scheinen.

2 Bioinformatische Grundlagen

Bioinformatik umfasst die Erforschung, Entwicklung oder Anwendung von rechnergestützten Werkzeugen, um den Nutzen von biologischen, medizinischen, verhaltenswissenschaftlichen oder gesundheitsbezogenen Daten zu vergrößern, eingeschlossen solcher zur Erfassung, Speicherung, Organisierung, Archivierung, Analyse oder Visualisierung dieser Daten [13]. Größere Schwerpunkte in der Bioinformatik liegen in der rechnergestützten Genomik und Proteomik, im Medikamentendesign, Biodatenbanken und im Data Mining, der molekularen Phylogenetik, Microarray Informatics und der Systembiologie [2]. Die vorliegende Arbeit ist dem Bereich der Proteomik, speziell dem Teilbereich der Sequenzanalyse zuzuordnen. Die wichtigsten hierbei zur Anwendung gekommenen Methoden werden in den folgenden Kapiteln kurz zusammengefasst.

2.1 Fisher-Scores

Die Anwendung von Fisher-Scores diente in den folgenden Kapiteln als zentrales Element, um die zu untersuchenden Aminosäuresequenzen in numerischen, miteinander vergleichbaren Größen ausdrücken zu können. Daher sei an dieser Stelle zum besseren Verständnis eine kurze Herleitung dieser Funktion angeführt.

Wenn es in der in der Statistik gilt innerhalb großer Kollektive, die aufgrund ihrer Größe und dem damit verbundenen enormen Rechenaufwand nicht in ihrer Gesamtheit untersucht werden können, Aussagen über bestimmte Kennwerte dieser Kollektive zu treffen, bedient man sich häufig repräsentativer Stichproben, anhand derer man mithilfe parametrischer Schätzverfahren versucht Rückschlüsse auf das Gesamtkollektiv zu ziehen. Ein solches Verfahren stellt die von R. A. Fisher entwickelte Maximum-Likelihood-Methode dar [43]. Sie wird in Situationen benutzt, in denen eine solche Stichprobe des Gesamtkollektivs als Realisierung eines Zufallsexperiments interpretiert werden kann, das von einem unbekannten Parameter **θ** abhängt, bis auf diesen aber eindeutig bestimmt und bekannt ist. Entsprechend hängen die interessanten Kennwerte ausschließlich von diesem unbekannten Parameter ab, lassen sich also als Funktion von ihm darstellen [156].

Es bezeichne nun **X** eine Zufallsvariable (≙ Stichprobe) mit ihrer zugehörigen Dichte- bzw. Wahrscheinlichkeitsfunktion ***f (X ; θ)***. Hierbei ist **θ** ein (möglicherweise mehrdimensionaler) unbekannter Parameter. Weiterhin seien $\boldsymbol{x_1, x_2, \dots, x_n}$ verschiedene Realisationen dieser Zufallsvariablen. Die Likelihood-Funktion dieser Stichprobe ist definiert als die Funktion, die jedem Parameterwert **θ** den Wert

$$L(\theta|x_1, x_2, \dots, x_n) = f(x_1, x_2, \dots, x_n; \theta) = \prod_{i=1}^{n} f_{x_i}(x_i; \theta)\,, \qquad (2.1.1)$$

also die gemeinsame Dichte- bzw. Wahrscheinlichkeitsfunktion, zuordnet [77].

Als Maximum-Likelihood-Schätzer wird nun derjenige Parameter **θ** bezeichnet, der die Wahrscheinlichkeit, die Stichprobe **X** zu erhalten, maximiert. Es handelt sich also um den Wert von **θ**, bei dem die Stichprobenwerte $\boldsymbol{x_1, x_2, \dots, x_n}$ die größte Dichte- bzw. Wahrscheinlichkeitsfunktion haben. Die Maximierung dieser Funktion erfolgt, indem man die erste Ableitung der Likelihood-Funktion nach **θ** bildet und gleich Null setzt. Da dies bei Dichtefunktionen mit komplizierten Exponentenausdrücken sehr aufwändig werden kann, wird häufig die logarithmierte Likelihood-Funktion (≙ Log-Likelihood-Funktion) verwendet, da sie auf Grund der Monotonie des Logarithmus ihr Maximum an derselben Stelle wie die nicht logarithmierte Dichtefunktion besitzt, jedoch einfacher zu berechnen ist:

$$\ell(\theta) = \log L(\theta; X) = \log\left(\prod_{i=1}^{n} f_{x_i}(x_i; \theta)\right) = \sum_{i=1}^{n} \log f_{x_i}(x_i; \theta) \tag{2.1.2}$$

Da die erste Ableitung nach **θ** die Steigung der Likelihood-Funktion beschreibt und damit angibt, wie different sich verschiedene Werte von **θ** auf die Funktion auswirken, ist diese auch ein Maß für die Sensitivität der Funktion [156][158]. Sind sowohl die Stichprobenwerte $\boldsymbol{x_1, x_2, \dots, x_n}$ als auch der Parameter **θ** der Likelihood-Funktion bekannt, lässt sich anhand dieser Ableitungsfunktion somit auch der Wert errechnen, mit dem die beobachteten Werte $\boldsymbol{x_1, x_2, \dots, x_n}$ unter dem gegebenen Parameter **θ** zu erwarten waren. Handelt es sich um ein mehrdimensionales **θ**, lassen sich die partiellen Ableitungsfunktionen für jedes Element von **θ** in Form eines Vektors formulieren, also in Form des Gradienten **∇** der (Log-)Likelihood-Funktion nach ihrem Parameter **θ**, genannt Fisher-Score:

$$U_X = \nabla_\theta \log L(\theta; X) = \frac{\partial}{\partial \theta_i} \log L(\theta; X) \tag{2.1.3}$$

Fisher-Scores dienen also als numerische Vergleichswerte, die sich in Abhängigkeit eines Parameters **θ** aus dem Vergleich einer Nullhypothese mit einem Vergleichsobjekt errechnen und damit Angaben über die Differenz des Vergleichsobjekts zur Nullhypothese bezüglich dieses Parameters machen [65][64][158].

2.2 Support-Vector-Machines

(Die Kapitel 2.2, 2.2.1, 2.2.2 und 2.2.3 basieren – sofern nicht anders vermerkt – frei auf der Arbeit von Fletcher T.: Support Vector Machines Explained (2009) [44])

Als Support-Vector-Machine (SVM) wird in der Statistik und Informatik ein Konzept des maschinellen Lernens bezeichnet, das in der Lage ist Daten nach bestimmten Mustern zu analysieren und zu klassifizieren. Eine SVM ist als nicht-probabilistischer, binärer Klassifikator zu verstehen, welcher auf der Basis eines Trainingsdatensatzes versucht ein Modell zu schaffen, das in der Lage ist die Zugehörigkeit unbekannter Daten zu einer von zwei Klassen zu berechnen. Das SVM-Modell repräsentiert die einzelnen Daten des Trainingsdatensatzes nach ihren jeweiligen Merkmalen als Objekte in einem vordefinierten Raum und versucht diese anhand ihrer Klassenzugehörigkeit so voneinander zu trennen, dass um die Klassengrenzen herum ein möglichst breiter Bereich frei von Objekten bleibt [155]. Dies wird anhand einer Hyperebene realisiert, welche die Objekte beider Klassen voneinander trennt (Abb. 2.1). Die Objekte, die der Hyperebene dabei am nächsten liegen, werden Support-Vektoren genannt und sind die Namensgeber des Verfahrens. Ziel ist es die Hyperebene so auszurichten, dass ihr Abstand zu den Support-Vektoren beider Klassen, genannt Margin, maximal wird. Neue Daten können dann, abhängig davon wo sie sich in diesem Raum abbilden, der einen oder anderen Klasse zugeordnet werden.

Der Algorithmus für SVMs wurde in seiner derzeitig gebräuchlichen Form erstmals durch Vladimir Vapnik und Corinna Cortes 1995 beschrieben [22].

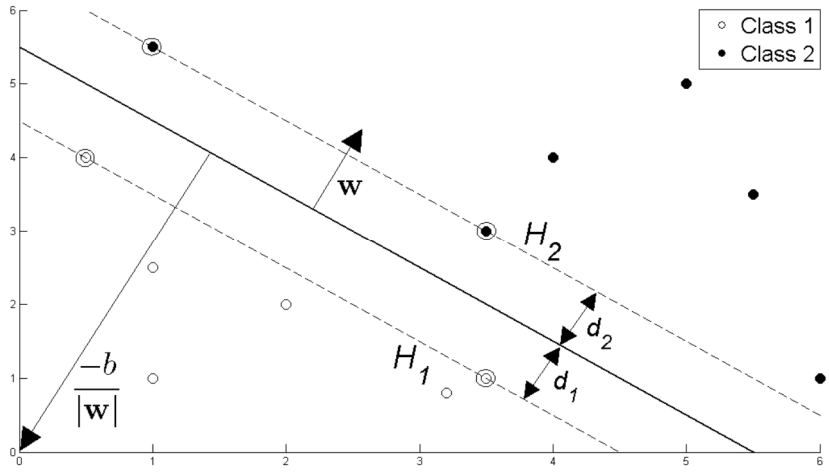


Abb. 2.1 Hyperebene durch zwei linear separierbare Klassen. Die eingekreisten Punkte entsprechen den Support-Vektoren, auf denen die Ebenen H_1 und H_2 liegen (gestrichelte Linien). Die Äquidistanz d_2 und d_2 der Hyperebene (durchgezogene Linie) zu den Ebenen H_1 und H_2 definiert den Margin der SVM.
(Aus Fletcher T.: Support Vector Machines Explained (2009) [44])

2.2.1 Mathematische Herleitung im Falle von linear separierbaren Daten (Hard-Margin)

Sei L eine Menge von Trainingsobjekten, bei der jedes Objekt x_i aus D Attributen bzw. Dimensionen besteht und einer von zwei Klassen y_i = -1 oder +1 zugeordnet ist, so entsprechen die Trainingsdaten T:

$$T = \{(x_i, y_i) \mid x_i \in \mathbb{R}^D, y_i \in \{-1, 1\}\}_{i=1}^{L} \tag{2.2.1}$$

Unter der Voraussetzung linear separierbarer Daten kann bei $D = 2$ eine Trennlinie auf dem Graphen von x_1 vs. x_2, bei $D > 2$ eine Hyperebene auf den Graphen von x_1, x_2,,… x_D gebildet werden, die die beiden Klassen voneinander trennen. Diese Hyperebene lässt sich formulieren als:

$$w \cdot x + b = 0\,, \tag{2.2.2}$$

wobei w dem Normalenvektor der Hyperebene und $b / \|w\|$ der lotrechten Distanz der Hyperebene zum Nullpunkt des Vektorraums entspricht. Die Ebenen H_1 und H_2, auf denen die Support-Vektoren liegen (Abb. 2.1), lassen sich mit folgenden Formeln beschreiben:

$$x_i \cdot w + b = +1 \quad \text{für } H_1 \tag{2.2.3}$$

$$x_i \cdot w + b = -1 \quad \text{für } H_2 \tag{2.2.4}$$

Für die Trainingsdaten gilt somit:

$$x_i \cdot w + b \geq +1 \quad \text{für } y_i = +1 \tag{2.2.5}$$

$$x_i \cdot w + b \leq -1 \quad \text{für } y_i = -1 \tag{2.2.6}$$

Kombiniert man die Gleichungen (2.2.5) und (2.2.6) entsteht:

$$y_i(x_i \cdot w + b) - 1 \geq 0 \;\; \forall_i \tag{2.2.7}$$

Als Margin der SVM wird die Äquidistanz von H_1 und H_2 zur Hyperebene (also $d_1 = d_2$ in Abb. 2.1) bezeichnet, das heißt:

$$\frac{1}{2} \cdot (H_1 - H_2) = \frac{1}{2} \cdot \frac{2}{\|w\|} = \frac{1}{\|w\|} \tag{2.2.8}$$

Um die Hyperebene nun beiderseits soweit als möglich entfernt von den Support-Vektoren platzieren zu können, muss dieser Margin unter der Nebenbedingung (2.2.7) maximiert werden. Es gilt daher:

$$\min\|w\| \quad \text{unter der Bedingung} \quad y_i(x_i \cdot w + b) - 1 \geq 0 \;\; \forall_i \tag{2.2.9}$$

Damit den Nebenbedingungen dieser Minimalisierung Rechnung getragen werden kann, wird das Problem in einer Lagrange-Funktion formuliert, in der für ihre Multiplikatoren $\boldsymbol{\alpha}$ gilt: $\boldsymbol{\alpha_i \geq 0 \ \forall_i}$. Da die Minimierung von $\|\boldsymbol{w}\|$ gleichbedeutend der Minimierung von $\frac{1}{2}\|\boldsymbol{w}\|^2$ ist, lautet das Primärproblem:

$$L_P \equiv \frac{1}{2}\|w\|^2 - \alpha[y_i(x_i \cdot w + b) - 1 \ \forall_i] \tag{2.2.10}$$

$$\equiv \frac{1}{2}\|w\|^2 - \sum_{i=1}^{L} \alpha_i[y_i(x_i \cdot w + b) - 1] \tag{2.2.11}$$

$$\equiv \frac{1}{2}\|w\|^2 - \sum_{i=1}^{L} \alpha_i y_i(x_i \cdot w + b) + \sum_{i=1}^{L} \alpha_i \tag{2.2.12}$$

$\boldsymbol{L_p}$ aus (2.2.12) muss nun bezüglich $\boldsymbol{w}$ und $\boldsymbol{b}$ minimiert und bezüglich $\boldsymbol{\alpha}$ (für das weiter gelten soll: $\boldsymbol{\alpha_i \geq 0 \ \forall_i}$) maximiert werden; die Minimierung von $\boldsymbol{w}$ und $\boldsymbol{b}$ lässt sich durch Ableiten der Funktion nach $\boldsymbol{w}$ und $\boldsymbol{b}$, sowie anschließendem Gleichsetzen der Ableitungen mit null erreichen:

$$\frac{\partial L_P}{\partial w} = 0 \Rightarrow w = \sum_{i=1}^{L} \alpha_i y_i x_i \tag{2.2.13}$$

$$\frac{\partial L_P}{\partial b} = 0 \Rightarrow \sum_{i=1}^{L} \alpha_i y_i = 0 \tag{2.2.14}$$

Durch Einsetzen der Gleichungen (2.2.13) und (2.2.14) in Gleichung (2.2.12) erhält man das duale Problem, welches jetzt abhängig von $\boldsymbol{\alpha}$ maximiert werden muss:

$$L_D \equiv \sum_{i=1}^{L} \alpha_i - \frac{1}{2}\sum_{i,j} \alpha_i \alpha_j y_i y_j x_i \cdot x_j \ \text{ s.t. } \ \alpha_i \geq 0 \ \forall_i, \sum_{i=1}^{L} \alpha_i y_i = 0 \tag{2.2.15}$$

$$\equiv \sum_{i=1}^{L} \alpha_i - \frac{1}{2}\sum_{i,j} \alpha_i H_{ij} \alpha_j \quad \text{mit: } H_{ij} \equiv y_i y_j x_i \cdot x_j \tag{2.2.16}$$

$$\equiv \sum_{i=1}^{L} \alpha_i - \frac{1}{2}\alpha^T H \alpha \ \text{ s.t. } \ \alpha_i \geq 0 \ \forall_i, \sum_{i=1}^{L} \alpha_i y_i = 0 \tag{2.2.17}$$

Die hieraus resultierende Formel zur Maximierung des dualen Problems

$$\max_{\alpha}\left[\sum_{i=1}^{L}\alpha_i - \frac{1}{2}\alpha^T H\alpha\right] \qquad \text{s.t. } \alpha_i \geq 0 \ \forall_i, \sum_{i=1}^{L}\alpha_i y_i = 0 \tag{2.2.18}$$

– und damit auch die Lösung für $\boldsymbol{\alpha}$ – lässt sich mithilfe von Methoden der quadratischen Optimierung lösen, wodurch nun auch $\boldsymbol{w}$ anhand von (2.2.13) bestimmt werden kann. Nun bleibt lediglich die Bestimmung von $\boldsymbol{b}$. Nach den Karush-Kuhn-Tucker-Bedingungen [70][79] gilt die Lösung als optimal, wenn:

$$\alpha_i[y_i(x_i \cdot w + b) - 1] = 0 \tag{2.2.19}$$

Das bedeutet, nur für Punkte $\boldsymbol{x_i}$ mit funktionalem Rand = 1, die also auf dem Rand des Margins liegen, sind die entsprechenden $\boldsymbol{\alpha_i} > 0$ (alle anderen $\boldsymbol{\alpha_i} = 0$). Diese entsprechen den Support-Vektoren $\boldsymbol{x_s}$. Die Lagrange-Multiplikatoren $\boldsymbol{\alpha_i}$ geben also Auskunft über die Wichtigkeit eines Trainingspunktes $\boldsymbol{x_i}$ bei der Formulierung des finalen Modells [123]. Dies zeigt sich auch in (2.2.14), da nur Punkte $\boldsymbol{x_i}$ mit $\boldsymbol{\alpha_i} > 0$ diese Gleichung erfüllen und somit auf dem Rand des Margins liegen. Damit lässt sich jeder Support-Vektor $\boldsymbol{x_s}$ nun wie folgt beschreiben:

$$y_s(x_s \cdot w + b) = 1 \tag{2.2.20}$$

Setzt man diese Gleichung in (2.2.13) ein erhält man:

$$y_s\left(\sum_{m\in S}\alpha_m y_m x_m \cdot x_s + b\right) = 1 \tag{2.2.21}$$

wobei $\boldsymbol{S}$ den Satz an Indizes $\boldsymbol{i}$ bezeichnet, der Support-Vektoren entspricht. Wie bereits beschrieben, definieren sich diese durch die Indizes $\boldsymbol{i}$, an denen $\boldsymbol{\alpha_i} > 0$. Durchmultiplizieren der Gleichung (2.2.21) mit $\boldsymbol{y_s}$ liefert:

$$y_s^2\left(\sum_{m\in S}\alpha_m y_m x_m \cdot x_s + b\right) = y_s \tag{2.2.22}$$

Aus den Gleichungen (2.2.5) und (2.2.6) folgt $\boldsymbol{y_s^2 = 1}$, daher gilt für $\boldsymbol{b}$:

$$b = y_s - \sum_{m\in S}\alpha_m y_m x_m \cdot x_s \tag{2.2.23}$$

Es empfiehlt es sich jedoch, statt eines beliebigen Support-Vektors x_s, das arithmetische Mittel aller Support-Vektoren in S zu nutzen:

$$b = \frac{1}{N_s} \sum_{s \in S} \left(y_s - \sum_{m \in S} \alpha_m y_m x_m \cdot x_s \right) \tag{2.2.24}$$

Somit sind nun alle Variablen bestimmt, die nötig sind, um die Orientierung der idealen Hyperebene und damit die Support-Vector-Machine zu definieren.

Da w normiert ist, bezeichnet das Skalarprodukt $\langle x,w \rangle$ gerade die Länge der Projektion von x in Richtung w. Durch Addieren von b erhält man den Abstand von x zur Hyperebene. Somit kann die Entscheidungsfunktion zur Klassifikation unbekannter Daten durch Anwendung der Signum-Funktion auf $\langle x,w \rangle + b$ gebildet werden:

$$f(x) = \operatorname{sgn}(w \cdot x + b) = \operatorname{sgn}\left(\sum_{i=1}^{L} \alpha_i y_i x \cdot x_i + b \right) \tag{2.2.25}$$

[42]

2.2.2 Klassifikation nicht vollständig linear separierbarer Daten (Soft-Margin)

Da sich Daten in der Praxis in aller Regel nicht streng linear voneinander trennen lassen, sondern an ein oder anderer Stelle überlappen, wird der Algorithmus der SVM so erweitert, dass Verletzungen der Nebenbedingungen möglich sind, jedoch minimal gehalten werden sollen.

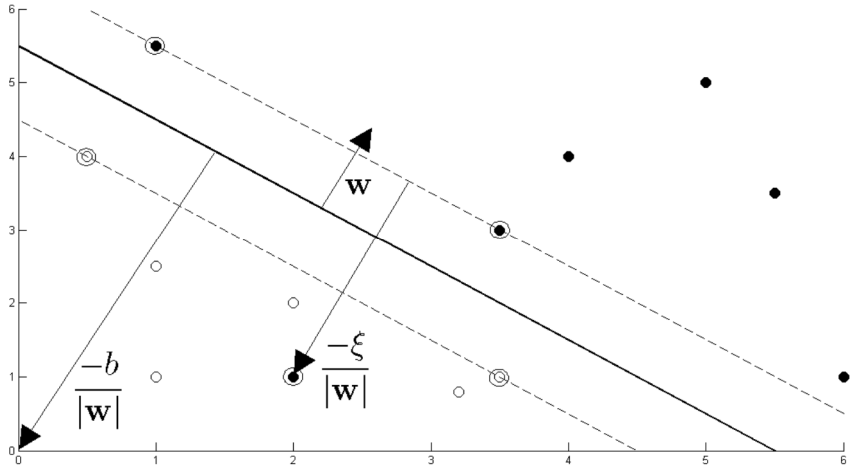


Abb. 2.2 Hyperebene durch zwei nicht vollständig linear separierbare Klassen, mit graphischer Erläuterung der Schlupfvariablen ξ.
(Aus Fletcher T.: Support Vector Machines Explained (2009) [44])

Zu diesem Zweck wird eine positive Schlupfvariable ξ_i ***(i = 1,….L)*** eingeführt, deren Wert umso mehr zunimmt, je weiter entfernt der Punkt x_i von der korrekten Seite der Hyperebene liegt. Zudem wird eine positive Konstante C eingeführt, die die Abstimmung der Fehler-Minimierung, das heißt der Minimierung von ξ_i, mit der Größe des Margins regelt.

Die Gleichungen (2.2.5) und (2.2.6) werden nun also folgendermaßen ausgedrückt:

$$x_i \cdot w + b \geq +1 - \xi_i \qquad \text{für } y_i = +1 \tag{2.2.26}$$

$$x_i \cdot w + b \leq -1 + \xi_i \qquad \text{für } y_i = -1 \tag{2.2.27}$$

$$\xi_i \geq 0 \ \forall_i \tag{2.2.28}$$

Nach Kombination von (2.2.26), (2.2.27) und (2.2.28) erhält man:

$$y_i(x_i \cdot w + b) - 1 + \xi_i \geq 0 \qquad \text{mit: } \xi_i \geq 0 \ \forall_i \tag{2.2.29}$$

Nach Erweiterung der Gleichung (2.2.9) entsteht die neue Forderung:

$$\min \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{L}\xi_i \quad \text{s.t.} \quad y_i(x_i \cdot w + b) - 1 + \xi_i \geq 0 \ \forall_i \tag{2.2.30}$$

Dieses Problem lässt sich nun in analoger Weise zu Kapitel 2.2.1 durch Formulierung in einer Lagrange-Funktion mit anschließender Minimalisierung dieser bezüglich w, b und ξ_i sowie Maximierung bezüglich α lösen.

2.2.3 Nicht lineare Support-Vector-Machines

Ist eine lineare Separation der Daten durch eine Hyperebene mit den oben beschriebenen Methoden nicht ohne Weiteres möglich, da der überlappende Bereich der Klassen zu groß ist oder das Klassifikationsproblem nicht linearer Art ist, nützt man das Theorem von Cover [23] zur Lösung des Problems. Dieses besagt, dass ein komplexes Klassifikationsproblem, welches nicht-linear in einem hochdimensionalen Raum abgebildet wird, mit höherer Wahrscheinlichkeit linear lösbar wird als in einem niedrigdimensionalen Raum, vorausgesetzt, der Raum ist nicht dicht besiedelt.

Dazu müssen die Daten mit der Funktion Φ verarbeitet werden:

$$\Phi\colon \mathbb{R}^{d_1} \to \mathbb{R}^{d_2}, x \longmapsto \Phi(x) \quad \text{wobei: } d_1 < d_2 \tag{2.2.31}$$

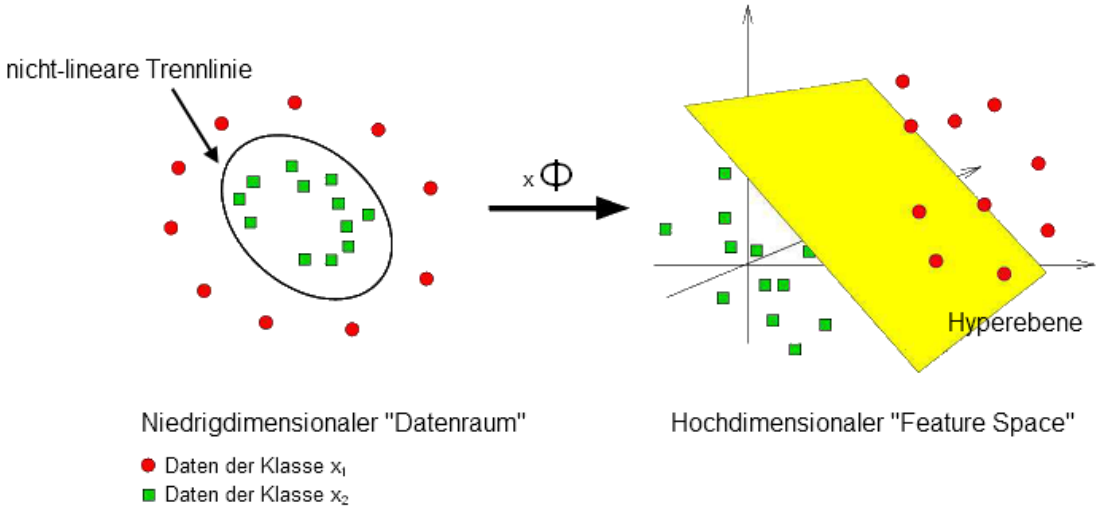


Abb. 2.3 Ein nicht-lineares Problem wird durch Abbildung im hochdimensionalen Feature-Space linear separierbar.
(Frei nach: Markowetz, F., Klassifikation mit SVM. Genomische Datenanalyse (2003) [92])

Der erweiterte Merkmalsraum $\mathbb{R}^{d_2}$ wird *Feature-Space* genannt und ist ein hochdimensionaler Raum in dem ein Skalarprodukt definiert ist. Das Skalarprodukt $\langle x_i, x_j \rangle$ aus Gleichung (2.2.16) zur Berechnung der Datenmatrix $\boldsymbol{H}$, und damit zur Lösung des Dualen Problems, muss also nun durch das Skalarprodukt $\langle \Phi x_i, \Phi x_j \rangle$ ersetzt werden. Da die Berechnung von $\langle \Phi x_i, \Phi x_j \rangle$ im hochdimensionalen Feature-Space jedoch sehr aufwendig bis nahezu unmöglich werden kann, bedient man sich hierbei des sogenannten „Kernel-Tricks“ [52]: Das Skalarprodukt $\langle x_i, x_j \rangle$ kann auch durch die Gleichung

$$y_i y_j k(x_i, x_j) = x_i \cdot x_j \tag{2.2.32}$$

beschrieben werden, wobei $k(x_i, x_j)$ eine sogenannte *Kernel-Funktion* beschreibt. Diese Funktionen basieren alle auf dem Errechnen eines Skalarprodukts zweier Vektoren. Wird in den oben genannten Funktionen das Skalarprodukt $\langle \Phi x_i, \Phi x_j \rangle$ durch eine solche möglicherweise nicht-lineare Kernel-Funktion ausgedrückt, so ist es möglich diese in einem hochdimensionalen Raum abzubilden, ohne Φ dabei explizit berechnen zu müssen, da die Skalarprodukte der Objekte nur im niedrigdimensionalen Merkmalsraum $\mathbb{R}^{d_1}$ berechnet werden müssen. Häufig benutze Kernel-Funktionen sind:

- Linearer Kernel: $K(x_i, x_j) = x_i^T x_j$
- Polynomieller Kernel: $K(x_i, x_j) = (\gamma x_i^T x_j + r)^d, \gamma > 0$
- Radial Basis Funktion (RBF): $K(x_i, x_j) = \exp\left(-\gamma \|x_i - x_j\|^2\right), \gamma > 0$
- Sigmoider Kernel: $K(x_i, x_j) = \tanh(\gamma x_i^T x_j + r)$ [60]

Zur genauen Funktionsweise von Kernel-Funktionen sei auf entsprechende Fachliteratur verwiesen, beispielsweise [24] oder [126].

2.2.4 Multiclass SVM

Aufgrund der Schwierigkeiten und des enormen Aufwands einen einzigen Algorithmus zur direkten Klassifikation von Problemen mit mehr als zwei verschiedenen Klassen zu entwickeln, zieht man vor Multiclass-Probleme auf mehrere binäre Probleme aufzuteilen, um diese im Anschluss mithilfe einer Kombination aus binären SVMs zu lösen [34]. Die zu diesem Zweck am häufigsten eingesetzten Methoden sind:

- die *„One-Against-All"*-Methode (OvA), die auf einer *„Winner-Takes-All"*-Strategie basiert [34],
- die „One-Against-One"-Methode (OvO), die auf einem *„Max-Wins-Voting"*-Prinzip basiert [34],
- DAGSVM [115] und
- Error-Correcting Codes [30].

Da das in dieser Arbeit zur Erstellung von SVMs angewandte Softwarepaket e1071 [31][97] die OvO-Methode zur Lösung von Multiclass-Problemen nutzt, soll sich im Folgenden auf eine (kurze) Beschreibung dieser beschränkt werden. Bei dieser Methode wird für alle möglichen, paarweisen Kombinationen der einzelnen Klassen $y_{1,\ldots,n}$ eines Multiclass-Problems ein separater binärer Subklassifikator konstruiert – im Falle von insgesamt n Klassen also $n(n-1)/2$ Subklassifikatoren. Wenn nun ein zu klassifizierendes Objekt x vom Subklassifikator C_{ij} der Klasse y_i zugeordnet wird, erhält diese Klasse eine Stimme. Nachdem alle $n(n-1)/2$ Subklassifikatoren ihre Stimmen verteilt haben, wird das Objekt x der Klasse zugeteilt, die in allen Subklassifikationen die meisten Stimmen erhalten hat [34].

2.3 Feature Selection

Ein bekanntes Problem bei der Klassifikation von Daten mit hoher Dimensionalität liegt in der Gefahr des sogenannten "*overfitting*" des Klassifikators. Hierunter versteht man eine Überanpassung des Klassifikationsmodells an die gegebenen Trainingsdaten, sodass unbekannte Daten, die naturgemäß leicht von den Trainingsdaten abweichen, nicht mehr suffizient klassifiziert werden können. Overfitting tritt vor allem dann auf, wenn die einzelnen Datenobjekte eine sehr hohe Dimensionalität (das heißt eine große Anzahl von zu beobachtenden Merkmalen) aufweisen, die Menge an zur Verfügung stehenden Trainingsdaten selbst im Verhältnis aber gering ist [54].

Um dieses Verhältnis zugunsten der Datenmenge anzuheben, muss also entweder die zur Verfügung stehende Datenmenge erhöht werden, was sich beispielsweise durch Regularisierung der Daten (das heißt Generieren künstlicher Daten) erreichen lässt, oder die Dimensionalität des Datenraumes verringert werden. Ansätze mit dem Ziel einer derartigen Dimensionsreduktion des Datenraumes werden im Bereich des maschinellen Lernens als *Feature Selection* bezeichnet. Hierbei wird nach Wegen gesucht, anhand derer sich aus den gegebenen Daten die relevanten Merkmale bzw. Dimensionen des Datenraums zur Erstellung eines Klassifikators „herausfiltern" lassen, um so redundante oder irrelevante Merkmale bzw. Features als Störfaktoren zu vermeiden. Ein positiver Nebeneffekt dieser Methoden – neben der offensichtlichen Verkürzung von Lernprozessen des Klassifikators – liegt in der hierdurch zusätzlich verbesserten Interpretierbarkeit der Daten und des Klassifikationsmodells selbst [54].

Als Form der Feature Selection, basierend auf Support-Vector-Machines, wurde von Guyon et al. die sogenannte *Recursive-Feature-Elimination* (RFE) vorgeschlagen. Bei dieser Methode handelt es sich um eine schrittweise, rückwärtige Merkmals- bzw. Feature-Elimination. Die einzelnen Merkmale werden zunächst ihrer Signifikanz nach für die Klassifikation bewertet. Dies geschieht durch eine Sensitivitätsanalyse der Merkmale auf den Normalenvektor der Hyperebene $\boldsymbol{w}$, das heißt je höher der Einfluss eines Merkmals auf $\boldsymbol{w}$, umso höher ist seine Wertung. Im zweiten Schritt wird das bzw. werden die am niedrigsten bewertete(n) Merkmal(e) bei der Klassifikation weggelassen. Beide Schritte werden solange wiederholt bis nur noch ein Merkmal übrig ist. Hiernach kann die Teilmenge an Merkmalen bestimmt werden, mit der die beste Klassifikation erreicht wurde [161].

3 Material und Methoden

3.1 Verwendete Datenbanken, Hard- und Software

3.1.1 Datenbanken/Online Ressourcen

Name	**Beschreibung / Referenz**
Bioconductor	Gemeinschaftsprojekt zur Erstellung von erweiterbarer Software für bioinformatische Zwecke [49]. (http://www.bioconductor.org/)
CRAN	Das *Comprehensive R Archive Network* stellt weltweit identische, aktuelle Versionen der Programmiersprache *R* sowie zahlreiche zusätzliche Software-Pakete hierfür zur Verfügung [157]. (http://www.r-project.org/)
NCBI Entrez	*Entrez* ist ein System zur Bereitstellung von Informationen über biologische Daten durch Verlinkung verschiedener Datenbanken miteinander, unter anderem *GenBank* [9], *SwissProt* [111], *Protein Information Resource* [7], *Protein Data Bank* [10][75] und *RefSeq* [116][153][127]. (http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=Protein)
PDB	Die *Protein Data Bank* ist die weltweit einzige Online-Ressource, die Informationen über den strukturellen Aufbau von biologischen Makromolekülen bietet [10][75]. (http://www.rcsb.org/pdb/)
SMART	Das *Simple Modular Architecture Research Tool* dient der Identifikation und Annotation von Proteindomänen, sowie der Analyse des Aufbaus von Proteinen bzw. ihren Domänen. Die aktuelle Version enthält Modelle für 1009 Proteindomänen sowie vollständig sequenzierte Genome von 1133 Spezies [82][128]. (http://smart.embl-heidelberg.de)

Zbio.net	*Zbio.net* ist eine Internetplattform für Biologen und Molekularbiologen, auf der unter anderem Werkzeuge, wie Sequence Tools zur Analyse von Aminosäureketten bzw. Nukleinsäuren zur Verfügung gestellt werden. Die Plattform wird durch Aleksey Soldatov und Tanja Borodina unterhalten (Max-Planck-Institut für Molekulare Genetik Berlin, Abteilung Prof. Dr. H. Lehrach). (http://zbio.net)

3.1.2 Bioinformatische Software

Name	**Beschreibung / Referenz**
Biobase (v1.10)	Bioconductor-Softwarepaket mit mathematischen bzw. bioinformatischen Basisfunktionen für *R* [49]. (http://bioconductor.case.edu/bioconductor/2.5/bioc/html/Biobase.html)
convert.pl	Kleines Programm von Madera M. und Gough J. zur Konvertierung von SAM-Modellen in HMMER2.0-Modelle und umgekehrt. (http://www.mrc-lmb.cam.ac.uk/genomes/julian/convert/convert.html)
e1071 (v1.5-8)	CRAN-Softwarepaket zur Implementation von SVM-Algorithmen in *R* [31][97]. (http://cran.r-project.org/web/packages/e1071/index.html)
HMMer (v2.3.2)	Erstellung und Alignment von Hidden Markov-Modellen (HMM) [55][6][78] und Multiple-Sequence-Alignments sowie unter anderem noch Generieren künstlicher Sequenzen [36]. (http://hmmer.janelia.org)
MASS (v7.2-2)	CRAN-Softwarepaket mit *R*-Funktionen und -Datensätzen zur Unterstützung von Venables und Ripley, 'Modern Applied Statistics with S' (4. Edition, 2002) [145]. (http://cran.r-project.org/web/packages/MASS/index.html)

multtest (v1.5.2)	Bioconductor-Softwarepaket mit *R*-Funktionen zum Testen multipler Hypothesen [35][133]. (http://www.bioconductor.org/packages/release/bioc/html/multtest.html)
pamr (v1.14.2 - v1.44.0)	CRAN-Softwarepaket mit *R*-Funktionen zur Sample-Klassifikation in Microarrays [53][140]. In der vorliegenden Arbeit wurde aus v1.14.2 die Funktion *balanced.folds* zur Durchführung einer Kreuzvalidierung genutzt. (http://cran.r-project.org/web/packages/pamr/index.html)
PyMOL (v0.99)	Programm zur Visualisierung, Animation und zum Rendern von dreidimensionalen molekularen Strukturen [28][139]. (http://www.pymol.org/)
RFE (v0.2)	Softwarepaket mit Funktionen zur Recursive-Feature-Elimination von SVMs auf der Basis von *R*; Autoren: Christophe Ambroise (ambroise@utc.fr) und Geoff McLachlan (gjm@maths.uq.edu.au) [54][161]. (http://www.hds.utc.fr/~ambroise/softwares/RFE/)
SAM (v3.4)	Das *Sequence Alignment and Modeling System* ist eine Sammlung verschiedener Software-Tools zum Erstellen und Bearbeiten von linearen Hidden Markov Modellen für biologische Sequenzanalysen [61][62]. (http://compbio.soe.ucsc.edu/sam.html)
WebLogo (v2.8.2)	Applikation zum Erstellen von komparativen Sequenzlogos aus Alignments [26] (http://weblogo.berkeley.edu/)
xtable (v1.2-4 und v1.2-5)	CRAN-Softwarepaket zum Einbau von Daten in LaTeX und HTML-Tabellen auf der Basis von *R* [27]. (http://cran.r-project.org/web/packages/xtable/index.html)

3.1.3 Hardware und Allgemeine Software

Als Hardware kamen handelsübliche Desktop- und Notebook-Rechner mit Single-Core *Intel*- bzw. *AMD*-Prozessoren zum Einsatz, wobei die Desktop-Rechner vom Lehrstuhl für Bioinformatik der Universität Würzburg zur Verfügung gestellt wurden. Auf den Desktop-Rechnern diente *SuSE Linux 8.2* mit der Linux-Kernel Version 2.4.20 und auf dem Notebook-Rechner *SUSE Linux 9.2* mit der Linux-Kernel-Version 2.6.8 als Betriebssystem. Als Programmiersprache wurde *R* Version 1.9.1 verwendet. *R* ist eine frei erhältliche Programmiersprache und Software-Umgebung für statistisches Rechnen, zur Daten-Analyse und zur graphischen Darstellung von Daten. Es wurde 1992 von Ross Ihaka und Robert Gentleman an der Universität von Auckland, Neuseeland, entwickelt und ist in Anlehnung an die Programmiersprache *S* entstanden. Es ist Teil des *GNU-Projekts* und sein Quell-Code ist unter der *GNU General Public License* frei erhältlich [157]. Die Funktionen zur Erstellung und Validierung der SVM basierten Klassifikatoren, sowie zur Datenanalyse mittels Recursive-Feature-Elimination und zur Aufbereitung der Daten für die graphische Darstellung in PyMOL [28][139] wurden sämtlich auf der Basis von *R* erstellt. Graphische Darstellungen wurden mithilfe der frei erhältlichen Software *Paint.NET* [154] bearbeitet.

3.2 Datenakquisition und Methoden

Der in der vorliegenden Arbeit verwendete Ansatz zur Einteilung der SH3-Domäne nach ihren Liganden in acht verschiede Klassen (Kapitel 1.5) basiert auf dem von Cesareni, et al. in der Arbeit: *Can we infer peptide recognition specificity mediated by SH3 domains?* vorgeschlagenen Modell aus dem Jahre 2002 [17]. Insgesamt 47 bereits klassifizierte SH3-Domänen dieser Arbeit wurden für den Ausgangsdatensatz zur Erstellung des Klassifikators ausgewählt. Weitere vier Domänen (Yjl020c, Cyk3, Yll017w, Ymr032w) mit bekannter Bindungsspezifität entstammen der Arbeit von Tong, et al. (2002) [141]. Da für diese noch keine Klassenzuordnung existierte, wurden sie nach den Kriterien des von Cesareni, et al. (2002) [17] vorgeschlagenen Modells anhand ihrer Bindungsspezifität ihren entsprechenden Klassen zugeordnet. Von den so insgesamt 51 Domänen entsprachen 14 der Klasse *1R*, sieben der Klasse *2R*, vier der Klasse *1K*, drei der Klasse *2K*, vier der Klasse *1@*, fünf der Klasse *2D*, sieben der Klasse *X_ORS* und weitere sieben der Klasse *Y*. Die Gesamtzahl der bereits nach diesem Schema klassifizierten Domänen war zwar geringfügig größer, jedoch sollten Domänen mit Affinität zu Liganden mehrerer Klassen hier möglichst nicht berücksichtigt werden. Es stellte sich allerdings erst im Nachhinein heraus, dass das Protein Yjl020c aus der Arbeit von Tong, et al. (2002) [141], für dessen SH3-Domäne dort eine Bindungsspezifität für Liganden der Klasse *1R* angegeben wurde, identisch ist mit dem Protein Bbc1 der Arbeit von Cesareni, et al. (2002) [17], in welcher die Domäne der Klasse *X_ORS* zugeteilt wurde. Ferner stellte sich erst nachträglich heraus, dass die Proteine Cyk3 und Ymr032w der Arbeit von Tong, et al. (2002) [141] identisch sind mit den Proteinen Ydl117w respektive Hof1 der Arbeit von Cesareni, et al. (2002) [17], wobei die Klassenzuordnung hier in beiden Fällen

identisch war. Bezüglich des Proteins SLA1 wurde die Arbeit von Cesareni, et al. (2002) [17] so interpretiert, dass alle drei seiner SH3-Domänen der Klasse *1R* angehören. Da dies jedoch nicht eindeutig aus der Arbeit hervorgeht, kann nicht mit letzter Sicherheit beurteilt werden, ob diese Einschätzung zutreffend ist.

Anhand der Gen- bzw. Protein-Identifikatoren der ausgewählten Domänen wurde in der *NCBI Entrez*-Datenbank [153] nach der Aminosäuresequenz des die jeweilige Domäne tragenden Proteins gesucht. Hinsichtlich dieser Proteine gelang es allerdings nicht in allen Fällen die Aminosäuresequenz der Spezies zu ermitteln, die in der in der Arbeit von Cesareni, et al. (2002) [17] bzw. Tong, et al. (2002) [141] angegeben wurde. Daher wurden in solchen Fällen auch Proteine anderer Spezies verwendet, die ein Äquivalent der gesuchten Domäne mit möglichst minimaler Sequenzvarianz enthielten. Um die Möglichkeit der Zuordnung dieser Sequenzen zu den entsprechenden der Arbeiten von Cesareni, et al. (2002) [17] und Tong, et al. (2002) [141] zu erhalten, wurden die Sequenz-Identifikatoren jedoch möglichst nicht geändert. Lediglich bezüglich der Domänen Eps8R1, Eps8R2, Abp1 und Domänen, die aus Proteinen mit mehr als einer SH3-Domäne stammen, wurden die Identifikatoren leicht modifiziert. Bei Eps8R1 und Eps8R2 erfolgte die Abkürzung von Homo sapiens nicht mit ´H` sondern ´HS` und Abp1 wurde als ABP-1 bezeichnet. Hinsichtlich Domänen aus Proteinen mit mehreren SH3-Domänen wurde nach dem Gen- bzw. Protein-Identifikator und vor der Domänennummer noch die Bezeichnung ´SH3` eingefügt, also beispielsweise wurde Bzz1-1 als Bzz1_SH3_1 bezeichnet. Da die im Netz erhältliche Sequenz der Domäne Lyn_H nicht mit der der Arbeit von Cesareni, et al. (2002) [17] übereinstimmte, in welcher die Sequenz dieser Domäne der der Domäne Fyn_M entsprach, wurde auch hier statt Lyn eine Variante der Domäne Fyn verwendet. Bezüglich einer klassenspezifischen Auflistung der Domänen incl. der Spezies, der sie entstammen sei auf die beigefügte DVD und die Hinweise hierzu im Anhang verwiesen[*]. Anschließend wurden innerhalb der annotierten Sequenzen mithilfe von *SMART* [82][128] diejenigen Sequenzabschnitte ermittelt, die der jeweils gesuchten SH3-Domäne entsprachen. Diese wurden unaligniert im *FASTA*-Format gespeichert und dienten im Folgenden als Ausgangsdatensatz des Klassifikators[†]. Wie sich erst im Nachhinein herausstellte, gingen jedoch während des Speicherns aus ungeklärter Ursache bei mehreren Sequenzen einige Aminosäuren jeweils vom N- bzw. C-Terminus verloren. Eine Datei, in welcher sämtliche Sequenzen des Ausgangsdatensatzes hinsichtlich ihrer hier verwendeten mit ihren unmodifizierten Originalvarianten verglichen werden, findet sich ebenso auf beigefügter DVD[‡]. Um die Sequenzen des Ausgangsdatensatzes (im Folgenden *Basissequenzen* genannt) so gut wie möglich

[*] „Ausgangsdatensatz.txt“, im Anhang auf DVD unter: „~\Sequenzen\Trainingssaetze“

[†] „R0“, im Anhang auf DVD unter: „~\Sequenzen\Trainingssaetze\R0“

[‡] „Verlorene Sequenzabschnitte.txt“, im Anhang auf DVD unter: „~\Sequenzen\Trainingssaetze“

alignieren zu können, wurde zunächst anhand der Funktion *hmmbuild* aus dem Softwarepaket *HMMer* [36] ein *Hidden Markov Modell* (HMM) des in der SMART-Datenbank [82][128] erhältlichen Familien-Alignments der SH3-Domäne (Home Page → search SMART: SH3 "Family Alignment" → SH3 : Src homology 3 domains → Button: „Family Alignment in“ und Drop down list: CLUSTALW-Format[*]) erstellt[†]. Dieses Familien-HMM diente nun der Funktion *hmmalign* – ebenso aus dem Softwarepaket HMMer [36] – als Modell beim Alignment der Basissequenzen. Auf diese Weise erstellte Alignments der Basissequenzen wurden im *CLUSTAL*- und *A2M*-Format gespeichert[‡]. Nach Konvertierung des SH3-Familien-HMMs in ein mit *SAM* [61][62] kompatibles HMM anhand der Funktion *convert.pl*[§], konnten mit *get_fisher*_scores[**], einer Unterfunktion von SAM [61][62], die Fisher-Scores (Kapitel 2.1 und 4.1.1) der Basissequenzen berechnet werden. Die hierdurch in numerischen Vektoren ausgedrückten Basissequenzen konnten nun in *R* implementiert und in Form verschiedener Matrices aneinandergefügt werden, welche folgend zur Erstellung der SVM-Modelle anhand der Funktionen des Softwarepakets *e1071* [31][97] dienten.

Sowohl zur Erstellung der Modelle als auch zu ihrer späteren Validierung stand jedoch ausschließlich dieser Datensatz an Basissequenzen zur Verfügung. Sequenzen, die zum Training eines Modells genutzt werden, sollten jedoch nicht mehr zu dessen Validierung eingesetzt werden. Daher musste ein Weg gefunden werden, der es zulässt die Daten so aufzuteilen, dass einerseits das bestmögliche Modell auf der Basis so vieler Daten wie möglich erstellt werden kann, andererseits aber noch eine ausreichende Menge an Daten zur Validierung desselben zur Verfügung steht. Ein Lösungsansatz ist es, die Modelle einer Kreuzvalidierung zuzuführen, wofür grundsätzlich mehrere Verfahren zur Verfügung stehen. Im Rahmen dieser Arbeit kamen die Methoden der *k-fachen* Kreuzvalidierung in Form der zehnfachen und der *Leave-One-Out* Kreuzvalidierung zum Einsatz. Bei einer k-fachen Kreuzvalidierung wird die Menge aller Daten $\boldsymbol{D}$, aus $\boldsymbol{n}$ einzelnen Datenelementen ($\boldsymbol{d_1, d_2 \ldots d_n}$), in $\boldsymbol{k}$ möglichst gleich große Teilmengen $\boldsymbol{x_1, x_2 \ldots x_k}$ unterteilt, wobei $\boldsymbol{k \leq D}$ ist. Im zweiten Schritt werden nun $\boldsymbol{k}$ Modelle $\boldsymbol{M_1, M_2, \ldots M_k}$ generiert und getestet, für die jeweils eine Teilmenge $\boldsymbol{x_i}$ als Testmenge bestimmt und die verbleibenden Teilmengen $\boldsymbol{x_1 \ldots x_{k-i}}$ zur Trainingsmenge des jeweiligen Modells $\boldsymbol{M_i}$ addiert werden. Die Gesamtfehlerquote entspricht somit dem Durchschnitt der Einzelfehlerquoten der $\boldsymbol{k}$ Modelle. Die Leave-One-Out

---

[*] „SMART.aln“, im Anhang auf DVD unter: „~\Sequenzen\SH3-Familienalignment_SMART“

[†] Befehl: „hmmbuild -n <(filename).hmm> <hmmfile> <alignfile>“, gespeichert als „SMART.hmm“, im Anhang auf DVD unter: „~\Sequenzen\SH3-Familienalignment_SMART“

[‡] Befehl: hmmalign -o <(filename).aln> --outformat Clustal <hmmfile> <sequencefile>
bzw. hmmalign -o <(filename).a2m> --outformat A2M <hmmfile> <sequencefile>

[§] Befehl: convert.pl <hmmfile>

[**] Befehl: get_fisher_scores unused -fisher_feature match -i <SAM-hmmfile> -write_dist 0 -db <a2m-alignfile> -sw 0 > <(filename).fsv>

Kreuzvalidierung ist insofern speziell, als dass bei dieser – wie der Name bereits andeutet – jede Teilmenge $\boldsymbol{x_i}$ aus genau einem Datenelement $\boldsymbol{d_i}$ besteht, somit also $\boldsymbol{n}$ Modelle erstellt und getestet werden müssen [74]. Grundsätzlich gilt die zehnfache Kreuzvalidierung als das bessere Verfahren [74], ist jedoch aus in Kapitel 4.1 dargelegten Gründen nicht auf jede der hier vorliegenden Fragestellungen anwendbar gewesen.

Mit dem Ziel einer etwaigen Verbesserung der prognostischen Präzision der Modelle wurde die Menge an zur Verfügung stehenden Sequenzen (Kapitel 4.1.2.1) in einigen Testreihen um künstlich erzeugte Sequenzen erweitert. Dies geschah durch die Funktion *hmmemit*[*] des Softwarepakets HMMer [36]. Für eine genauere Beschreibung der Algorithmen, nach denen die künstlichen Sequenzen generiert wurden, sei auf Kapitel 4.1.2.1, zur Funktionsweise von SVMs auf Kapitel 2.2 und zur detaillierten Beschreibung der erstellten Klassifikatoren auf Kapitel 4.1.2 und 4.1.3 verwiesen.

Feature Selections, welche ebenso mit dem Ziel einer Verbesserung der prognostischen Präzision der Modelle (Kapitel 4.1.2.3) wie auch zur Identifikation der signifikantesten Aminosäurepositionen hinsichtlich des im Rahmen dieser Arbeit erstellten Klassifikators mit der höchsten prognostischen Präzision (Kapitel 4.2) eingesetzt wurden, erfolgten mithilfe des Softwarepakets *RFE (Recursive Feature Elimination)* [54][161]. Da die Unterfunktion *rfe.predict* dieses Softwarepakets versucht mit einem (bezüglich der verwendeten Version von e1071 [31][97]) inkompatiblen Befehl auf e1071 [31][97] zuzugreifen (statt *predict* wird *predict.svm* aufgerufen), wurde sie diesbezüglich manuell geändert und bei Arbeiten mit RFE [54][161] statt der Originalversion geladen[†]. Zur besseren Anpassung der Feature Selections an die hier vorliegenden Bedürfnisse, wurden zudem die (Unter-)Funktionen *orderFeatures*, *rfe.fit* und *rfe.plotcv* leicht modifiziert:

Da die generierten Klassifikatoren teilweise mit linearem und teilweise mit radialen Kernel arbeiteten, musste dies in den (Unter-)Funktionen *orderFeatures* und *rfe.fit* berücksichtigt werden. Daher wurde in der Unterfunktion *orderFeatures* der Abschnitt
model <- svm(x[, FeaturesToTest], y) durch
model <- svm(x[, FeaturesToTest], y, kernel="linear") respektive
model <- svm(x[, FeaturesToTest], y, kernel="radial") ersetzt[‡].

[*] Befehl: hmmemit -n <Anzahl künstl. Sequenzen> -o <filename> <hmmfile>

[†] „rfe.txt", im Anhang auf DVD beispielsweise unter: „~\Klassifikatoren\Kreuzvalidierung_Klassifikatoren"

[‡] „orderFeatures_linear_04-2012" bzw. „orderFeatures_radial_04-2012", im Anhang auf DVD beispielsweise unter: „~\Klassifikatoren\Kreuzvalidierung_Klassifikatoren"

In der Funktion *rfe.fit* wurde der Abschnitt
modelFeatures = (1:nbFeatures) um
modelFeatures = (1:nbFeatures),kernel ="linear" respektive
modelFeatures = (1:nbFeatures),kernel ="radial" erweitert*.

Die Funktion *rfe.plotcv*, die zur graphischen Visualisierung der Errorwerte bei der Feature Selection dient, wurde bezüglich mehrerer Aspekte modifiziert:

Da es sich bei dem vorliegenden Klassifikationsproblem nicht um Gene, sondern Fisher-Sites handelte, wurden bei allen Anwendungen der Funktion die Elemente *xlab* von
xlab = "log2(Number of Genes)" in
xlab = "log2(Fisher Sites)" abgewandelt†.

Mussten in der durch die Funktion erstellten unteren Teilgrafik viele Elemente dargestellt werden, war der Platz in der hierfür vorgesehenen Legende zum Teil nicht ausreichend. Daher wurde die Zeile
legend(0, 0.9, rownames(fit$error.ind), col = (2:(nc + 1)),lty = 1) bei Bedarf um
ncol=2 zu
legend(0, 0.9, rownames(fit$error.ind), col = (2:(nc + 1)),lty = 1,ncol=2) erweitert‡.

Ferner werden durch die Funktion in der erstellten Graphik grundsätzlich nur die niedrigsten Errorwerte markiert, in der vorliegenden Arbeit waren jedoch zum Teil auch die zweitniedrigsten Errorwerte von Bedeutung. Daher wurde für diese Fälle eine modifizierte Variante dieser Funktion erstellt, welche *rfe.plotcvZweitBestes* genannt wurde. Diese Variante enthielt ebenso die abgewandelten Elemente *xlab*. Die beschriebene Modifikation der Legende war hinsichtlich der Einsatzgebiete dieser Funktion jedoch nicht nötig und wurde daher hier nicht angewandt. Zur Markierung der zweitniedrigsten Errorwerte wurden in dieser Variante unter der Zeile
o <- fit$error.cv == min(fit$error.cv) die Zeilen
Error <-fit$error.cv
Error[o] <- 1
o2 <- Error == min(Error)

---

* „rfe2.txt“ bzw. „rfe3.txt“, im Anhang auf DVD beispielsweise unter: „~\Klassifikatoren\Kreuzvalidierung_Klassifikatoren“

† „plofit“, im Anhang auf DVD beispielsweise unter: „~\Klassifikatoren\Kreuzvalidierung_Klassifikatoren“

‡ „plofit2“, im Anhang auf DVD beispielsweise unter: „~\Klassifikatoren\Kreuzvalidierung_Klassifikatoren“

sowie unter der Zeile
points(log2(fit$nbFeatures[o]), fit$error.cv[o], pch = "x") die Zeile
points(log2(fit$nbFeatures[o2]), fit$error.cv[o2], pch = 23) eingefügt*.

Zur Anwendung des im Rahmen dieser Arbeit erstellten Klassifikators mit der höchsten prognostischen Präzision auf fremde Sequenzen (Kapitel 4.1.3), sowie zur Analyse der Daten aus Feature Selections (Kapitel 2.3) mit RFE [54][161] hinsichtlich der Identifikation der für letzteren Klassifikator signifikantesten Aminosäurepositionen (Kapitel 4.2), wurde sich weiterer 29 Aminosäuresequenzen von SH3-Domänen bedient, von denen im Rahmen einer Arbeit von Friedrich, et al. (2006) [46] unter anderem ein positionsspezifisches Interaktionsprofil mit ihren Liganden erstellt wurde† (da weder die 29 Sequenzen, noch ihr Interaktionsprofil innerhalb des Papers [46] detailliert aufgeführt sind, wurde hierfür mit dem Autor persönlich Kontakt aufgenommen). Allerdings war die Klassenzuordnung dieser Sequenzen nicht bekannt, daher sollten mithilfe der von Friedrich, et al. (2006) [46] für diese Sequenzen angegebenen *PDB*-Identifikatoren ihre entsprechenden PDB-Dateien in der *Protein Data Bank* [10][75] gesucht werden, um die Sequenzen anhand ihres in der jeweiligen PDB-Datei dargestellten Liganden – falls diese Angaben über den Liganden enthielten – nach dem Modell von Cesareni, et al. (2002) [17] zu klassifizieren (Search: PDB ID → Download Files → PDB-Files (Text))‡. Die strukturelle Darstellung der PDB-Dateien sowie die Analyse der Liganden geschahen mithilfe von *PyMOL* [28][139]. Das Alignment sowie die Errechnung der Fisher-Scores dieser Sequenzen geschahen in analoger Weise zu den Basissequenzen.

Plot-Grafiken zur Analyse der wichtigsten Aminosäurepositionen wurden mit der R-Standardfunktion *plot* erstellt. Strukturelle Analysen der durch die Feature Selections gewonnenen Daten über die für den im Rahmen dieser Arbeit erstellten Klassifikator mit der höchsten prognostischen Präzision signifikantesten Aminosäurepositionen geschahen anhand von PDB-Dateien geeigneter Beispieldomänen mit *PyMOL* [28][139]. Hierzu wurde aus den Basissequenzen exemplarisch jeweils mindestens eine Sequenz bezüglich jeder Klasse gewählt, für die eine PDB-Datei mit zusätzlicher Darstellung des klassenspezifischen Liganden gebunden an die Domäne existierte (sofern es sich um eine bindende Sequenz handelte). Die Suche nach diesen PDB-Dateien geschah anhand der Gen- bzw. Protein-Identifikatoren der Domänen in der *NCBI Entrez*-Datenbank [153] (Search Gene/Protein: Identifier → Öffnen des entsprechenden Ergebnisses → Links → 3D-structure → PDB ID). Die

---

* „PlotRFE_Markierung_auch_zweit_kleinster_Error“, im Anhang auf DVD beispielsweise unter: „~\Klassifikatoren\Kreuzvalidierung_Klassifikatoren“

† „Testsatz&Interaktionsprofil-Friedrich“ bzw. „Testsatz&Interaktionsprofil-Friedrich_angepasst“, im Anhang auf DVD unter: „~\Sequenzen\Testsatz-Friedrich“ (Kapitel 7.1.1.4)

‡ Die PDB-Dateien der 29 Sequenzen der Arbeit von Friedrich, et al. (2006) [46] finden sich im Anhang auf DVD unter: „~\PDB-Dateien\Testsatz-Friedrich“

entsprechende PDB-Datei konnte dann anhand des PDB-Identifikators in PDB [10][75] gefunden werden (Search: PDB ID → Download Files → PDB-Files (Text)). Zwei der PDB-Dateien (die Domänen YAR014C und YDL117W) entstammen jedoch der SAM-Website http://compbio.soe.ucsc.edu/sam.html (SAM-T02 HMM WWW Servers → Yeast protein predictions) [61][62][69]. Für Klasse *X_ORS* konnte keine Domäne mit geeigneter PDB-Datei gefunden werden*.

Komparative Sequenzlogos aus den Alignments der einzelnen Klassen wurden mit WebLogo [26] erstellt. Zur Konversion der Buchstabenkodes von Aminosäuren wurde das „Three-/ one-letter Amino Acid Codes“ Programm der Internetplattform Zbio.net (http://zbio.net/eng/scripts/01_17.html) genutzt.

Die Softwarepakete *Biobase* [49], *MASS* [145], *multtest* [35][133] und *xtable* [27] wurden zur Erweiterung des Reservoirs an statistischen Funktionen bei der Arbeit mit *R* genutzt.

* Die gewählten klassenspezifischen Beispieldomänen und ihre entsprechenden PDB-Dateien: „2LCS.pdb“ = NBP2, „1SSH.pdb“ = Yfr024c, „2VKN.pdb“ = SHO1, „2RPN.pdb“ = ABP-1, „1BBZ.pdb“ = Abl1_H, „2ROL.pdb“ = Eps8R1_HS, „YAR014C.t2k.undertaker-align.pdb“ = Yar014c, „YDL117W.t2k.undertaker-align.pdb“ = Ydl117w, im Anhang auf DVD unter: „~\PDB-Dateien\Beispieldomaenen“

4 Ergebnisse

4.1 Erstellen des bestmöglichen Klassifikators und Validierung der Prädiktabilität

Hintergedanke der Arbeit war die Fragestellung, inwieweit sich Support-Vector-Machines, denen ausschließlich die Aminosäuresequenzen der SH3-Domäne als Informationsquelle zur Verfügung stehen, zu Vorhersagen über die Bindungsspezifität bzw. Klassenzugehörigkeit von SH3-Domänen nach dem Modell von Cesareni, et al. (2002) [17] eignen. Dieses Modell beschreibt acht verschiedene Klassen von SH3-Domänen, welche sich jeweils durch bestimmte sequenzielle Eigenschaften ihrer Liganden definieren, und diente im Folgenden als Grundlage der Analysen (Kapitel 1.5). Es galt also einen SVM-basierten Klassifikator zu generieren, der in der Lage sein sollte, Domänen dieser Familie rein anhand ihrer Aminosäuresequenz den beschriebenen acht Klassen möglichst valide zuzuordnen. Informationen über die Struktur oder Sequenz der Liganden wurden dem Klassifikator hierbei ebenso wie strukturelle Informationen über die Domänen selbst bewusst vorenthalten. Die Klassifikation sollte ausschließlich auf dem Boden einer Primärstruktur-Analyse der Domänen, also ihrer Aminosäuresequenz, erfolgen. Insgesamt 51 bereits nach ihren Liganden klassifizierte Sequenzen von SH3-Domänen dienten hierbei als Ausgangsdatensatz des zu erstellenden Klassifikators (Kapitel 3.2).

4.1.1 Konversion der Sequenzen in numerische Vektoren

Zur Erstellung des Klassifikators war es nötig, die Basissequenzen in einer Form miteinander vergleichbar zu machen, bei der dem Klassifikator möglichst viele Informationen über ihre evolutionäre bzw. sequentielle Verwandtschaft zur Verfügung stehen. Das in der SMART-Datenbank [82][128] zur Verfügung gestellte Familien-Alignment der SH3-Domäne bot sich hierfür aus zweierlei Gründen als Vergleichsgrundlage an: zum einen, da in multiplen Sequenzalignments [18] gerade solche Informationen konserviert sind und zum anderen, da dieses Alignment auf der Basis aller bekannten SH3-Domänen erstellt wurde, es somit die maximal erhältliche Informationsmenge über SH3-Domänen diesbezüglich enthält.

Um die Basissequenzen an das Familien-Alignment zu alignieren, wurde von diesem zunächst mithilfe der Funktion *hmmbuild* (Kapitel 3.2) aus dem Softwarepaket HMMer [36] ein HMM errechnet. Dieses diente im Folgenden als Ausgangsmodell beim Alignment weiterer Sequenzen mit der Funktion *hmmalign* (Kapitel 3.2) des Softwarepakets HMMer [36]. Die Alignments wurden sowohl im A2M- wie auch im CLUSTAL-Format erstellt und ließen sich nun bezüglich ihrer einzelnen Aminosäurepositionen untereinander sowie mit dem Familien-Alignment bzw. dem hieraus erstellten Hidden Markov Model präzise in Relation stellen (Kapitel 3.2). Die alignierten Sequenzen bestanden jeweils aus 58 Positionen, etwaige Gaps mit eingeschlossen.

Um die Informationen der Sequenzen für den auf mathematischer Basis konstruierten Klassifikator verwertbar zu machen, mussten diese in numerischen Werten ausgedrückt werden. Zu diesem Zweck wurden die Sequenzen anhand der Funktion *get_fisher_scores* aus dem Softwarepaket SAM [61][62] (Kapitel 3.2) nach der von Jaakkola T., et. al. (1999) [65] beschriebene Methode mit dem erstellten Familien-HMM der SH3-Domäne verglichen und hieraus ihre Fisher-Score-Vektoren (Kapitel 2.1) berechnet. Wie bereits in Kapitel 2.1 erläutert, beschreiben Fisher-Scores $\boldsymbol{U_x}$ den Gradienten $\boldsymbol{\nabla}$ einer Log-Likelihood-Funktion ***logL(θ,X)*** mit der Zufallsvariablen $\boldsymbol{X}$ nach einem Parameter $\boldsymbol{\theta}$:

$$U_X = \nabla_\theta \log L(\theta; X) \tag{4.1.1}$$

Auf den vorliegenden Fall übertragen ist dies der Gradient der Log-Likelihood (≙ „a posteriori" Wahrscheinlichkeitsfunktion) eines HMMs für jede Aminosäure $\boldsymbol{x_i}$ an Position $\boldsymbol{p_i}$ der zu untersuchenden Sequenz $\boldsymbol{X}$ bezüglich der gewählten Parameter $\boldsymbol{\theta}$ des HMMs, also z.B. der Emissions-, Insertions-, oder Deletions-Wahrscheinlichkeiten. Der Fisher-Score hinsichtlich eines bestimmten HMM-Parameters $\boldsymbol{\theta_i}$ beschreibt demnach die Steigung der Log-Likelihood-Funktion des HMMs bezüglich dieses Parameters für eine gegebene Aminosäure $\boldsymbol{x_i}$ an Position $\boldsymbol{p_i}$ der Sequenz $\boldsymbol{X}$. Fisher-Scores dienen im vorliegenden Fall sozusagen als ein Maß der Verwunderung über das Vorliegen einer Aminosäure $\boldsymbol{x_i}$ an Position $\boldsymbol{p_i}$ in der Sequenz $\boldsymbol{X}$ bei Vergleich der Sequenz $\boldsymbol{X}$ mit dem gegebenen HMM [129]. Da die Steigung im Maximum einer Funktion gleich null ist, ist also eine Aminosäure an einer bestimmten Position mit dem jeweiligen HMM umso konformer, je mehr sich die Werte ihres Fisher-Score-Vektors null nähern. Fisher-Scores sind daher ein ideales System zur positionsspezifischen, numerischen Codierung von Aminosäuresequenzen (aber auch anderen Sequenzen wie z.B. DNA/RNA), das jede Aminosäureposition (bzw. jedes Element) einer Sequenz abhängig der gewählten Parameter $\boldsymbol{\theta}$ durch einen numerischen Vektor mit gleicher Anzahl an Dimensionen ausdrückt, indem es ihre (bzw. seine) Eigenschaften in Relation zum genutzten HMM stellt. Da in dieser Arbeit ausschließlich die Wahrscheinlichkeiten der *Match-Emissionen* zur Berechnung der Fisher-Scores dienten, wurde jede Aminosäureposition demnach durch einen Vektor von 20 Dimensionen codiert. Bei einer Sequenzlänge von 58 Aminosäuren entsprach dies also einem Vektor mit 1160 Dimensionen pro Sequenz. Die Fisher-Score-Vektoren der einzelnen Basissequenzen konnten nun in *R* implementiert und ihren Klassen nach geordnet zu einer Matrix aneinandergefügt werden, bei der jede Zeile/Basissequenz durch einen Fisher-Score-Vektor mit 1160 Spalten/Dimensionen ausgedrückt wurde. Die Matrix der Basissequenzen bestand also aus 51 Zeilen zu 1160 Spalten.

```
HMMER2.0  [2.3.2]
NAME  SMART.hmm
LENG  58
ALPH  Amino
RF    no
CS    no
MAP   yes
COM   hmmbuild -n SMART.hmm SMART.hmm SMART.aln
NSEQ  290
DATE  Mon Sep 27 18:30:05 2004
CKSUM 4784
XT      -8455   -4  -1000  -1000  -8455   -4  -8455   -4
NULI     -4  -8455
NULE    595  -1558    85    338   -294   453  -1158   197   249   902  -1085  -142   -21  -313    45   531   201   384  -1998  -644
HMM       A       C       D       E       F       G       H       I       K       L       M       N       P       Q       R       S       T       V       W       Y
        m->m    m->i    m->d    i->m    i->i    d->m    d->d    b->m    m->e
          -6       *   -8017
  1   -3031     844    -648     975     429     148     -29   -1116     763   -1048      64    -809    1326     540     490   -1240    -342      40   -6792   -145     3
      -  -149    -500     233      43    -381     399     106    -626     210    -466    -720     275     394      45      96     359     117    -369    -294    -249
      -     0  -13179  -14221    -894   -1115    -701   -1378      -6       *
  2   -1638    -501   -2010      38    1076   -6220     176    -127     776    -610     162   -1323    -897    1326     569   -1071     639     172    -340    1678     4
      -  -149    -500     233      43    -381     399     106    -626     210    -466    -720     275     394      45      96     359     117    -369    -294    -249
      -     0  -13185  -14227    -894   -1115    -701   -1378       *
  3    1601    1948   -7803   -7167     308    -555     -51   -1130   -6762   -1476     748   -3493   -7055   -3278   -6561   -3772   -2573    2355    -660    1910     5
      -  -149    -500     233      43    -381     399     106    -626     210    -466    -720     275     394      45      96     359     117    -369    -294    -249
      -     0  -13185  -14227    -894   -1115    -701   -1378       *
  4   -2929   -6349   -5183    -175   -6554   -6222   -2219    1132    1584   -2556   -1263   -2972   -3331     780    2070   -3892    -115    1780   -6607   -802     6
      -  -149    -500     233      43    -381     399     106    -626     210    -466    -720     275     394      45      96     359     117    -369    -294    -249
      -     0  -13185  -14227    -894   -1115    -701   -1378       *
  5    3199    -668   -7929   -7314    -5397   -1768   -6031   -2553   -4201   -3245   -2311   -6763   -7146   -6532   -6709   -1633    -130    1360   -5905   -5564     7
      -  -149    -500     233      43    -381     399     106    -626     210    -466    -720     275     394      45      96     359     117    -369    -294    -249
      -     0  -13185  -14227    -894   -1115    -701   -1378       *
  6   -2222    -101   -2672   -1626   -2413   -6629    -345    1248     369    2248     256   -1302   -6701     200    -829   -2930    -805     482   -6038   -5618     8
      -  -149    -500     233      43    -381     399     106    -626     210    -466    -720     275     394      45      96     359     117    -369    -294    -249
      -     0  -13185  -14227    -894   -1115    -701   -1378       *
  7   -1600     294   -1907   -1698    1970   -2957     825   -6318   -1432   -2742   -2322   -3027   -6291     -85    -730   -2078   -2619   -1792     764    4048     9
      -  -149    -500     233      43    -381     399     106    -626     210    -466    -720     275     394      45      96     359     117    -369    -294    -249
      -     0  -13185  -14227    -894   -1115    -701   -1378       *
  8     141   -6629    3278    -749   -6950   -1829   -1040   -6701   -1728   -6645   -5718     565     983   -1161   -2231     106    -777   -2917   -6813   -6130    10
      -  -149    -500     233      43    -381     399     106    -626     210    -466    -720     275     394      45      96     359     117    -369    -294    -249
      -     0  -13185  -14227    -894   -1115    -701   -1378       *
  9   -9327     897  -10070   -10292    3236   -9867     106   -3612   -9853   -3138   -7802   -2230   -9749   -8695   -1631   -9089   -9203   -2880     296    3947    11
      -  -149    -500     233      43    -381     399     106    -626     210    -466    -720     275     394      45      96     359     117    -369    -294    -249
      -     0  -13185  -14227    -894   -1115    -701   -1378       *
 10    -532   -1533    1315    1279   -2440   -3143     -89   -1229     459   -1870     -33     114    -444    1407   -1727    -359    1244    -188   -6808   -1974    12
      -  -149    -500     233      43    -381     399     106    -626     210    -466    -720     275     394      45      96     359     117    -369    -294    -249
      -     0  -13185  -14227    -894   -1115    -701   -1378       *
```

Abb. 4.1 Ausschnitt aus dem errechneten Familien-HMM der SH3-Domäne auf Basis des Alignments der SMART-Datenbank [82][128] mit Darstellung der Sequenzpositionen 1-10. Die einzelnen Positionen des Alignments werden unter Beachtung der phylogenetischen Eigenschaften und der Wahrscheinlichkeiten für das Auftreten einzelner Aminosäuren probabilistisch in Zahlenvektoren zusammengefasst. Zur Berechnung der Fisher-Scores dienten ausschließlich die Wahrscheinlichkeiten der Match-Emissionen (mit roten Pfeilen markierte Zeilen).
(„SMART.hmm“, im Anhang auf DVD unter: „~\Sequenzen\SH3-Familienalignment_SMART“)

4.1.2 Klassifikation mit SVMs und Kreuzvalidierung der Klassifikatoren

Um zu prüfen, inwieweit sich rein anhand der Aminosäuresequenzen der Domänen mit Support-Vector-Machines überhaupt verwertbare Vorhersagen bezüglich des Klassenmodells nach Cesareni, et al. (2002) [17] treffen lassen, sollten zunächst zwei Ausgangsklassifikatoren (einer trainiert mit linearer und der andere mit radialer Kernel-Funktion) erstellt und getestet werden, denen ausschließlich die Informationen des Ausgangsdatensatzes, also die in Fisher-Score-Vektoren ausgedrückten Basissequenzen, zur Verfügung stünden. Zur Validierung der Klassifikatoren sollte eine zehnfache Kreuzvalidierung zur Anwendung kommen. Die zur Verfügung stehenden Daten wurden also zunächst nach den Methoden einer zehnfachen Kreuzvalidierung (Kapitel 3.2) aufgeteilt und im Anschluss in den einzelnen Klassifikationsmodellen sukzessive als Trainings- bzw. Testsatz verwendet. Die Modelle/SVMs wurden hierbei als Multiclass-SVMs (wie bei e1071 [31][97] vorgegeben mit OvO-Klassifikationsprinzip) generiert und entsprechend der jeweiligen Klassifikatorvariante mit linearer bzw. radialer Kernel-Funktion trainiert. Bezüglich der übrigen, hierbei modifizierbaren Parameter wurde, wie auch bei allen weiteren im Folgenden generierten Modellen, jeweils der Standard gewählt (Testreihe 1)[*].

Es konnte gezeigt werden, dass sich zumindest anhand der linearen Klassifikatorvariante sehr valide Ergebnisse erzielen ließen. So war ihre Gesamttrefferquote von 50,98% der Zufallstrefferquote von 1/8 = 12,50% mehr als vierfach überlegen. Zwar lag auch die Gesamttrefferquote des radialen Klassifikators oberhalb der Zufallstrefferquote, dennoch sind dessen Resultate nicht als reliabel anzusehen, da hier sämtliche Testsequenzen – unabhängig ihrer wahren Klassenzugehörigkeit – Klasse *1R* zugeteilt wurden. Die hier dem Zufall überlegene Trefferquote ist also lediglich darauf zurückzuführen, dass die meisten Testsequenzen Klasse *1R* entstammten (Tabelle 4.1). Diese Beobachtungen spiegeln sich auch in den durchschnittlichen F-Maßen wider. So beträgt es unter der linearen Variante immerhin 49,57%, während es unter der radialen Variante gerade mal bei 5,38% liegt.

[*] Programm: „Test-AlleKlassen-10CV-LinearRadial_0“, im Anhang auf DVD unter: „~\Klassifikatoren\Kreuzvalidierung_Klassifikatoren“ (Kapitel 7.1.3.1 und 7.2).

```
$MapLinear
       pred1
true1   1R 2R 1K 2K 1@ 2D X_ORS Y
  1R     9  2  0  0  0  0     3 0
  2R     5  2  0  0  0  0     0 0
  1K     2  0  1  0  0  0     1 0
  2K     2  1  0  0  0  0     0 0
  1@     2  0  0  0  2  0     0 0
  2D     0  0  0  0  0  5     0 0
  X_ORS  4  1  0  0  0  0     2 0
  Y      2  0  0  0  0  0     0 5
```

```
$MapRadial
       pred2
true1   1R 2R 1K 2K 1@ 2D X_ORS Y
  1R    14  0  0  0  0  0     0 0
  2R     7  0  0  0  0  0     0 0
  1K     4  0  0  0  0  0     0 0
  2K     3  0  0  0  0  0     0 0
  1@     4  0  0  0  0  0     0 0
  2D     5  0  0  0  0  0     0 0
  X_ORS  7  0  0  0  0  0     0 0
  Y      7  0  0  0  0  0     0 0
```

```
[[5]]$tot.accuracy_linear
[1] 51

[[5]]$single.accuracies_linear
 [1] 20 67 60 50 57 33 50 50 80 43


$Total_SV_linear
 [1] 45 47 45 47 43 47 45 44 45 43
```

```
[[5]]$tot.accuracy_radial
[1] 28.45238

[[5]]$single.accuracies_radial
 [1] 40 33 20 25 29 33 33 17 40 14


$Total_SV_radial
 [1] 46 48 46 47 44 48 45 45 46 44
```

Tabelle 4.1 Testreihe 1: Resultate der anhand einer zehnfachen Kreuzvalidierung getesteten Ausgangsklassifikatoren trainiert mit linearer (links) und radialer (rechts) Kernel-Funktion zur Erstellung der Hyperebene. Unter Training mit linearer Kernel-Funktion war der Klassifikator dem Zufall mit einer Gesamttrefferquote von 50,98% deutlich überlegen.
pred1 = Klassenzuordnung durch Klassifikator, true1 = wahre Klassenzugehörigkeit, tot.accuracy_linear/radial = durchschnittliche Gesamttrefferquote der zehnfachen Kreuzvalidierung, single.accuracies_linear/radial = Gesamttrefferquoten der einzelnen Testläufe, Total_SV_linear/radial = Anzahl an Supportvektoren während der einzelnen Testläufe.
(„Erg-AlleKlassen-10CV-Linear_0“ bzw. „Erg-AlleKlassen-10CV-Radial_0“, im Anhang auf DVD unter: ~\ Ergebnisse_Klassifikatoren\Kreuzvalidierung_Klassifikatoren“; Kapitel 7.1.4 und 7.3)

4.1.2.1 Emission künstlicher Sequenzen

In den Ergebnissen der Kreuzvalidierung des linearen Ausgangsklassifikators lässt sich hinsichtlich der Abgrenzung von Sequenzen der Klassen *2D* und *Y* zu Sequenzen der übrigen Klassen eine besonders hohe Präzision mit einer Sensitivität von 100% respektive 71,43%, sowie einer Relevanz von jeweils 100% feststellen (Confusion-Map linear in Tabelle 4.1). Die Klassifizierung von Sequenzen anderer Klassen mit eher wenigen Trainingsdaten gelang hingegen deutlich schlechter, so lag die Sensitivität für Sequenzen der Klasse *2K* beispielsweise sogar bei 0,00%. Zudem drängte sich der Verdacht auf, dass die Klassifikatoren Testsequenzen umso wahrscheinlicher einer Klasse zuteilten, je größer die Menge ihrer Trainingsdaten ist. Dies zeigt sich besonders deutlich am Beispiel der Klasse *1R*, für die mit Abstand die meisten Trainingsdaten zur Verfügung standen. Hier lag die Relevanz, also die Wahrscheinlichkeit, dass eine Sequenz, die dieser Klasse zugeteilt wurde, auch tatsächlich dieser Klasse angehört, im Falle des linearen Ausgangsklassifikators bei nur 34,62%. Im Falle des radialen Ausgangsklassifikators wurden sogar sämtliche Sequenzen – unabhängig ihrer wahren Klassenzugehörigkeit – dieser Klasse zugeordnet. Diese Beobachtung dürfte sich auf ein overfitting (Kapitel 2.3) der Klassifikatoren bezüglich der kleineren Klassen zurückführen lassen, denn bei Klassen

mit nur wenigen Trainingsdaten können die Klassengrenzen entsprechend eng gezogen werden, wodurch zu klassifizierende Sequenzen solcher Klassen, die sich schon gering von den Trainingsdaten ihrer entsprechenden Klasse unterscheiden, bereits nicht mehr diesen Klassen zugeordnet werden können und dementsprechend größeren Klassen mit weiteren Grenzen zugeteilt werden. Dem Problem eines overfitting kann, wie bereits in Kapitel 2.3 ausgeführt, entweder durch Reduktion der Datendimensionen (≙ Feature Selection) oder aber durch Vergrößerung der Menge an Trainingsdaten begegnet werden. Unter Anwendung einer Feature Selection ändert sich aber nicht das Verhältnis der Klassengrößen zueinander, weshalb auch das relative overfitting hinsichtlich kleinerer Klassen hierunter nur bedingt abnehmen dürfte*. Daher sollte der Problematik durch Vergrößerung der Menge an Trainingsdaten begegnet werden, da hiermit eine Nivellierung der Klassengrößen möglich ist. Zudem lässt sich hiermit auch das Verhältnis zwischen Datendimensionalität zu Daten-Menge hinsichtlich aller Klassen (im vorliegenden Fall 1160 Dimensionen zu nur 51 Sequenzen) – und damit auch ein etwaiges generelles overfitting vermindern. Weitere, bereits klassifizierte Sequenzen standen jedoch nicht zur Verfügung, sodass zusätzliche Sequenzen künstlich emittiert werden mussten. Diese Sequenzen sollten jeweils zwei Kriterien erfüllen: einerseits sollten sie zwar die speziellen Charakteristika jener Klasse besitzen, für welche sie emittiert wurden, andererseits aber doch so different von den Basissequenzen der jeweiligen Klasse sein, dass hierdurch eine ausreichende Erweiterung der Klassengrenzen bewirkt würde. Die Funktion *hmmemit* des Softwarepakets HMMer [36] (Kapitel 3.2) setzt genau hier an, indem sie die generative Eigenschaft eines HMMs nutzt, um Sequenzen mit jeweils ähnlichen Eigenschaften derer zu kreieren, die zur Errechnung des betreffenden HMMs dienten, aber dennoch so different voneinander sind, dass eine Varianz, ähnlich der des natürlichen Vorbilds gewahrt bleibt. Auf diese Weise klassenspezifisch emittierte künstliche Sequenzen besitzen jedoch naturgemäß große sequenzielle Ähnlichkeit untereinander wie auch mit den zur Erstellung des jeweiligen HMMs herangezogenen natürlichen Sequenzen der entsprechenden Klasse. Daher galt es zur Vermeidung von Ergebnisverfälschungen bei späteren Kreuzvalidierungen einerseits darauf zu achten nur solche Basissequenzen als Testsequenzen zu verwenden, die nicht als Vorbild bei der Emission der künstlichen Sequenzen dienten, sowie andererseits auf den Einsatz künstlich generierter Sequenzen als Testsequenzen generell zu verzichten.

Anhand der Basissequenzen der einzelnen Klassen mussten also zunächst mithilfe der Funktion *hmmbuild* jeweils klassenspezifische HMMs errechnet werden, welche folgend als Grundlage bei der Emission klassenspezifischer, künstlicher Sequenzen dienen sollten. Um alle Basissequenzen einer

* Kleinere Klassen besitzen engere Klassengrenzen, da hier – durch die geringere Anzahl an Sequenzen – die Varianz der einzelnen Features insgesamt geringer ist als bei größeren Klassen. Mit anderen Worten, eine kleinere Klasse erlaubt einen im Verhältnis kleineren Spielraum an zulässigen Werten pro Feature als eine größere Klasse. Je mehr Features während einer Feature Selection also entfernt werden, umso mehr gleichen sich die Klassengrenzen einander an.

Klasse später auch als Testsequenzen einsetzen zu können, musste zudem jede von ihnen sukzessive einmal aus der Berechnung der klassenspezifischen HMMs der betreffenden Klasse ausgeschlossen werden, da so bezüglich jeder Basissequenz für ihren Einsatz als Testsequenz ein separater Trainingssatz erstellt werden konnte, in welchem die betreffende Basissequenz nicht als Vorlage bei der Emission der künstlichen Sequenzen fungierte. Insgesamt mussten also pro Klasse neben einem HMM, welches nach dem Vorbild aller Basissequenzen der betreffenden Klasse erstellt wurde, entsprechend der Anzahl an Basissequenzen ***n*** einer Klasse, zudem noch ***n*** HMMs berechnet werden, bei welchen sukzessive jeweils eine Basissequenz dieser Klasse aus der Berechnung ausgeschlossen wurde. Da die Funktion *hmmbuild* ein Alignment als Vorlage benötigt, wurden die zur Erstellung der HMMs jeweils genutzten Basissequenzen zuvor mit der Funktion *hmmalign* an das SH3-Familien-HMM aligniert und im CLUSTAL-Format gesichert. Anhand der berechneten HMMs konnten dann mithilfe der Funktion *hmmemit* beliebig viele künstliche Sequenzen emittiert werden. Nach heuristischen Methoden wurde sich für zwei Arten der Regularisierung entschieden, um hieraus resultierende Effekte möglichst valide abschätzen zu können. Primäres Ziel der der ersten Variante (im Folgenden **R20** genannt) war es die negativen Auswirkungen ungleicher Klassengrößen [148] durch Nivellierung derselben zu verringern. Daher wurden die Klassen bei dieser Variante mit einer eher kleinen Menge an zusätzlichen künstlichen Sequenzen auf jeweils insgesamt 20 Sequenzen pro Klasse regularisiert. Theoretisch ließen sich die negativen Effekte ungleicher Klassengrößen auch durch Gewichtung der einzelnen Klassen mithilfe des Arguments *class.weights* der Funktion *svm* aus dem Softwarepaket e1071 [31][97] verringern, da dies innerhalb des Softwarepakets RFE [54][161], welches unter anderem in Kapitel 4.2 zur weiteren Analyse der Klassifikatoren verwendet wurde, jedoch keine wählbare Zusatzoption ist, wurde auf die Anwendung dieser Methode generell verzichtet. Ziel der zweiten Variante (im Folgenden **R100** genannt) war es überdies noch einem etwaigen generellen overfitting zu begegnen, sodass hier möglichst viele künstliche Sequenzen pro Klasse emittiert werden sollten. Um aber die Erschaffung eines nicht mehr zu bewältigenden Rechenproblems zu vermeiden, wurde die Anzahl der zu emittierenden Sequenzen auf 100 pro Klasse festgelegt. Da die Unterschiede in den Klassengrößen hinsichtlich der Basissequenzen mit zunehmender Regularisierung immer unbedeutender werden, wurde bei dieser Variante auf eine präzise Nivellierung der Klassengrößen verzichtet (Kapitel 7.1.1.2).

Bezüglich beider Regularisierungsvarianten sollte nun ebenso je ein linearer und ein radialer OvO-Multiclass-Klassifikator erstellt und validiert werden und im Anschluss mit den Resultaten der Ausgangsklassifikatoren verglichen werden. Wie bereits beschrieben, durften zur Validierung von Modellen mit künstlichen Sequenzen nur Basissequenzen als Testsequenzen herangezogen werden, die zudem in ihrer Funktion als Testsequenz je einen eigenen Trainingssatz erforderten. Daher wäre eine zehnfache Kreuzvalidierung hier nur mit enormem Aufwand möglich gewesen, sodass auf das Prinzip einer Leave-One-Out Kreuzvalidierung (Kapitel 3.2) zurückgegriffen wurde. Entsprechend der Anzahl an Basissequenzen wurden also insgesamt je 51 Trainingssätze für jede der beiden

Regularisierungsvarianten erstellt. Diese bestanden jeweils aus den Basissequenzen – die jeweils zu testende wurde jedoch nicht zum Training genutzt – sowie den jeweiligen künstlichen Sequenzen der einzelnen Klassen, wobei bezüglich der Klasse, aus der die jeweils zu testende Basissequenz stammte, jeweils der Satz an künstlichen Sequenzen Verwendung fand, bei welchem die jeweils zu testende Basissequenz nicht als Vorlage bei der Emission diente (Kapitel 7.1.1.3)[*]. Anschließend wurden die 51 Trainingssätze pro Regularisierungsvariante inklusive ihrer zugehörigen Testsequenzen nach der in Kapitel 4.1.1 beschriebenen Methode erneut in numerische Vektoren umgewandelt und in *R* implementiert. Pro Klassifikator mussten nun anhand der 51 Trainingssätze der jeweiligen Regularisierungsstufe folglich 51 Klassifikationsmodelle in Form von OvO-Multiclass-SVMs generiert und entsprechend der jeweiligen Klassifikatorvariante mit linearer bzw. radialer Kernel-Funktion trainiert werden. Hiernach konnten die Modelle jeweils anhand der entsprechenden Basissequenzen im Sinne einer Leave-One-Out Kreuzvalidierung getestet werden[†] (Tabelle 4.2). Um die Ergebnisse hieraus mit denen der Klassifikatoren ohne zusätzliche künstliche Sequenzen (im Folgenden Regularisierungsvariante **R0** genannt) besser vergleichen zu können, wurden auch diese Klassifikatoren einer Leave-One-Out Kreuzvalidierung unterzogen (Testreihe 2)[‡].

Die Ergebnisse dieser Testreihe zeigen, dass hinsichtlich des vorliegenden Klassifikationsproblems sowohl der lineare wie auch der radiale Klassifikator von der Anwendung künstlicher Sequenzen profitierten und unter zunehmender Regularisierung zuverlässigere Ergebnisse lieferten. Dies ließ sich insbesondere auf eine Zunahme der Klassifikationsgüte von Sequenzen kleinerer Klassen[§] hierunter zurückführen. So konnte beispielsweise bei der Klassifikation mit linearem Kernel im Vergleich von **R0** zu **R100** ein Anstieg der Sensitivität für Sequenzen der Klassen *2R*, *1K*, *X_ORS* und *Y* um 14,29% respektive 25,00%, 14,29% und 28,57% beobachtet werden. Noch deutlicher wird dies bei Betrachtung der Ergebnisse aus der Klassifikation mit radialer Kernel-Funktion. Ohne zusätzliche künstlich generierte Sequenzen war der Klassifikator in diesem Fall gar nicht in der Lage die Testsequenzen korrekt zu klassifizieren und teilte alle der Klasse mit den meisten Trainingsdaten (*1R*) zu. Unter Verwendung künstlicher Sequenzen zeigte sich jedoch sogar hier mit zunehmender Regularisierung eine stetige Zunahme Trefferquote sowie der Sensitivität insbesondere für Sequenzen kleinerer Klassen

---

[*] Zusätzlich wurde im Hinblick auf spätere Anwendungen für jede der beiden Regularisierungsstufen noch ein weiterer Trainingssatz erstellt, dessen künstliche Sequenzen bezüglich der einzelnen Klassen jeweils nach dem Vorbild aller Basissequenzen der entsprechenden Klasse emittiert wurden (Kapitel 7.1.1.3).

[†] Programme: „Test-AlleKlassen-LinearRadial_20“ sowie „Test-AlleKlassen-LinearRadial_100“, im Anhang auf DVD unter: „~\Klassifikatoren\Kreuzvalidierung_Klassifikatoren“ (Kapitel 7.1.3.1 und 7.2).

[‡] Programm: „Test-AlleKlassen-LinearRadial_0“, im Anhang auf DVD unter: „~\Klassifikatoren\Kreuzvalidierung_Klassifikatoren“ (Kapitel 7.1.3.1 und 7.2).

[§] Bei unter Regularisierung (annähernd) gleich großen Klassen bezieht sich die Bezeichnung „kleine“ bzw. „große“ Klasse stets auf die relative Größe der jeweiligen Klasse ohne künstliche Sequenzen.

(Tabelle 4.2). Die Beobachtung einer erhöhten Sensitivität für Sequenzen der Klasse *1R* unter Verwendung der nativen Klassifikatoren, das heißt ohne Nutzung künstlicher Sequenzen, dürfte sich damit erklären lassen, dass diese Klasse ohne Regularisierung die größte Menge an Trainingsdaten bietet und daher im Vergleich mit den anderen Klassen auch die weitesten Klassengrenzen haben müsste. Dementsprechend war es bei der Klassifikation auch wahrscheinlicher, dass Sequenzen (unabhängig ihrer wahren Klassenzuordnung) eher den Klassifikationskriterien dieser Klasse entsprachen als denen einer Klasse mit engeren Grenzen. Der beobachtete Abfall der Sensitivität für Sequenzen dieser Klasse unter Verwendung künstlicher Sequenzen dürfte also eher nicht als Verschlechterung der Klassifikationsleistung sondern vielmehr als Folge einer (gewollten) Egalisierung der Klassengrenzen zu interpretieren sein.

Bei Betrachtung der Klassifikationsgenauigkeiten war unter Verwendung beider Klassifikationsvarianten (linear und radial) tendenziell zu beobachten, dass die Relevanzwerte größerer Klassen bei steigender Regularisierung zunächst zunahmen, dann aber wieder fielen, während sie sich bezüglich kleinerer Klassen genau entgegengesetzt verhielten. So stieg die Relevanz der großen Klasse *1R* beispielsweise unter Verwendung eines linearen Kernels von **R0** auf **R20** um 21,22% (von 33,33% auf 54,55%), betrug bei **R100** jedoch nur noch 35,00%; die Relevanz der kleinen Klasse *2D* fiel im gleichen Beispiel zunächst von 100% auf 57,14%, betrug bei **R100** aber wieder 100% (Tabelle 4.2). Möglicherweise lässt sich dieses Phänomen dadurch erklären, dass größere Klassen im Vergleich zu kleineren unter der Regularisierungsvariante **R20** nur eine verhältnismäßig kleine Erweiterung ihrer Klassengrenzen erfuhren, sodass während der Klassifikation nun auch verhältnismäßig häufiger Sequenzen fälschlicherweise kleineren Klassen zugeteilt werden. Im Falle der Variante **R100** würde sich dieses Phänomen durch Regularisierung aller Klassen mit gleich vielen künstlichen Sequenzen entsprechend wieder relativieren. Zudem könnten bei dieser Variante die Klassengrenzen größerer Klassen im Verhältnis sogar stärker erweitert worden sein, da durch die höhere Anzahl an Basissequenzen größerer Klassen auch die Varianz künstlicher Sequenzen solcher Klassen größer sein dürfte.

Wie bereits erwähnt, konnte auch bezogen auf die die Gesamtklassifikation mit zunehmender Regularisierung bei beiden Kernelvarianten (linear und radial) eine Steigerung der Trefferquote erzielt werden. Unter Verwendung einer linearen Kernel-Funktion stieg die Quote im Vergleich von **R0** bis **R100** um 5,88% (von 49,02% auf 54,90%), unter Verwendung einer radialen Kernel-Funktion sogar um 25,49% (von 27,45% auf 52,94%). Dennoch zeigt der Vergleich der Ergebnisse bezüglich der verwendeten Kernel-Funktionen eine stete Überlegenheit der linearen Klassifikatoren (Tabelle 4.2).

Die Berechnung des durchschnittlichen *F-Maß* [122] der Klassifikatoren ergibt, dass dies bei Anwendung einer linearen Kernel-Funktion mit zunehmender Regularisierung zunächst fällt, dann allerdings bei **R100** sogar deutlich über dem Wert des mit Variante **R0** trainierten Klassifikators liegt.

Ohne Regularisierung

Linear

	pred1							
true1	1R	2R	1K	2K	1@	2D	X_ORS	Y
1R	9	2	0	0	0	0	3	0
2R	5	2	0	0	0	0	0	0
1K	2	0	1	0	0	0	1	0
2K	2	1	0	0	0	0	0	0
1@	2	0	0	0	2	0	0	0
2D	0	0	0	0	0	5	0	0
X_ORS	4	1	0	0	0	0	2	0
Y	3	0	0	0	0	0	0	4

Radial

	pred2							
true1	1R	2R	1K	2K	1@	2D	X_ORS	Y
1R	14	0	0	0	0	0	0	0
2R	7	0	0	0	0	0	0	0
1K	4	0	0	0	0	0	0	0
2K	3	0	0	0	0	0	0	0
1@	4	0	0	0	0	0	0	0
2D	5	0	0	0	0	0	0	0
X_ORS	7	0	0	0	0	0	0	0
Y	7	0	0	0	0	0	0	0

Regularisierung auf 20 Sequenzen pro Klasse

Linear

	pred1							
true1	1R	2R	1K	2K	1@	2D	X_ORS	Y
1R	6	1	1	0	0	1	1	4
2R	0	3	0	0	0	0	3	1
1K	1	0	1	1	1	0	0	0
2K	1	1	0	0	0	0	1	0
1@	1	0	1	0	2	0	0	0
2D	0	1	0	0	0	4	0	0
X_ORS	2	0	0	0	0	1	3	1
Y	0	0	0	0	0	1	0	6

Radial

	pred2							
true1	1R	2R	1K	2K	1@	2D	X_ORS	Y
1R	5	0	0	0	0	0	2	7
2R	0	2	0	0	0	0	2	3
1K	1	0	0	1	0	0	0	2
2K	0	1	1	0	0	0	0	1
1@	0	0	0	0	2	0	1	1
2D	0	0	0	0	0	1	0	4
X_ORS	2	0	0	0	0	0	2	3
Y	0	0	0	0	0	0	0	7

Regularisierung mit 100 zusätzlichen Sequenzen pro Klasse

Linear

	pred1							
true1	1R	2R	1K	2K	1@	2D	X_ORS	Y
1R	7	3	0	0	1	0	3	0
2R	3	3	0	0	1	0	0	0
1K	2	0	2	0	0	0	0	0
2K	2	1	0	0	0	0	0	0
1@	2	0	0	0	2	0	0	0
2D	0	0	0	0	0	5	0	0
X_ORS	3	0	0	0	0	0	3	1
Y	1	0	0	0	0	0	0	6

Radial

	pred2							
true1	1R	2R	1K	2K	1@	2D	X_ORS	Y
1R	7	2	0	0	1	0	3	1
2R	1	3	0	0	1	0	0	2
1K	2	0	2	0	0	0	0	0
2K	2	1	0	0	0	0	0	0
1@	2	0	0	0	2	0	0	0
2D	0	0	0	0	0	4	0	1
X_ORS	3	0	0	0	1	0	3	0
Y	0	0	0	0	1	0	0	6

Tabelle 4.2 Testreihe 2: Gegenüberstellung der Resultate der sowohl mit linearer (links) wie auch radialer (rechts) Kernel-Funktion trainierten und anhand einer Leave-One-Out Kreuzvalidierung getesteten Klassifikatoren nach Anwendung der drei Regularisierungsvarianten **R0**, **R20** und **R100**. In den Confusion-Maps beider Kernelvarianten lässt sich der Anstieg der Gesamttrefferquote sowie der Sensitivität vor allem hinsichtlich kleinerer Klassen unter zunehmender Regularisierung gut beobachten.
pred1 = Klassenzuordnung durch Klassifikator, true1 = wahre Klassenzugehörigkeit
(„Erg-AlleKlassen-Linear_0“, „Erg-AlleKlassen-Radial_0“, „Erg-AlleKlassen-Linear_20“, „Erg-AlleKlassen-Radial_20“, „Erg-AlleKlassen-Linear_100“ und „Erg-AlleKlassen-Radial_100“, im Anhang auf DVD unter: „~\Ergebnisse_Klassifikatoren\Kreuzvalidierung_Klassifikatoren“; Kapitel 7.1.4 und 7.3)

Mit anderen Worten, das F-Maß fällt von **R0** auf **R20** zunächst um 4,39% (von 48,10% auf 43,71%), beträgt bei **R100** jedoch sogar 54,07% (Tabelle 4.2). Die Beobachtung dieses initialen F-Maß-Abfalls lässt sich hier allerdings lediglich auf den relativ großen Einfluss der abnehmenden Relevanzwerte kleinerer Klassen bei geringer Regularisierungsstufe (**R20**) zurückführen. Im Gegensatz hierzu stieg das durchschnittliche F-Maß bei Klassifikation mit radialer Kernel-Funktion unter zunehmender Regularisierung stetig an (sogar schon bei geringer Regularisierungsstufe), was sich mit der starken Erhöhung der Sensitivität unter zunehmender Regularisierung in diesem Fall erklären lässt. Betrug es bei **R0** noch 5,38%, so lag es bei **R20** schon bei 31,75% und war bei **R100** mit 50,45% fast schon so hoch wie bei entsprechender Klassifikation mit linearer Kernel-Funktion.
Zusammenfassend lässt sich festhalten, dass sich mit zunehmender Regularisierung sowohl das relative overfitting hinsichtlich kleinerer Klassen verringern wie auch die Güte des Klassifikators insgesamt verbessern ließ. Da sich zudem in dieser Testreihe stets die SVM-Modelle mit linearer Kernel-Funktion denen mit radialer Kernel-Funktion überlegen zeigten, war der lineare Klassifikator mit Regularisierungsstufe **R100** demnach mit einer Gesamttrefferquote von 54,90% und einem durchschnittlichen F-Maß von 54,07% die präziseste der bis dahin erstellten Klassifikatorvarianten.

4.1.2.2 Arbiträre Klassifikation

Obwohl die Klassifikationsgüte unter Regularisierung mit künstlichen Sequenzen (Kapitel 4.1.2.1) insgesamt zunahm, war sie dennoch bezüglich einzelner Klassen noch immer eher unbefriedigend. Dies dürfte nicht zuletzt auch auf das von e1071 [31][97] genutzte Konzept zur Lösung von Multiclass-Problemen zurückzuführen sein, da unter Klassifikation mit OvO-Prinzip Informationen über die Taxonomie eines Klassensystems keine Berücksichtigung finden. Die Integration solcher Informationen in den Klassifikationsprozess dürfte jedoch von entscheidender Bedeutung sein, da diese die verwandtschaftlichen Zusammenhänge zwischen den Klassen herausstellen [12]. Um dies im vorliegenden Fall zu erreichen, musste zunächst die Taxonomie des Klassensystems definiert werden. Anschließend galt es das Multiclass-Problem entsprechend dieser Taxonomie auf mehrere, möglichst binäre Subklassifikationsschritte aufzuteilen, für welche jeweils ein separater Subklassifikator zu erstellen war. Diese Subklassifikatoren galt es dann, der hierarchischen Reihenfolge der Taxonomie folgend, in einem arbiträr aufgebauten Gesamtklassifikator miteinander zu kombinieren. Zur Definition der Taxonomie wurde sich an der Bindungsspezifität der einzelnen Klassen bzw. dem Verwandtschaftsgrad ihrer Konsensmotive orientiert. Dabei lässt sich zunächst Klasse *Y* von allen anderen abgrenzen, welche im Folgenden in Klasse *Nicht-Y* zusammengefasst wurden, da die Domänen der Klasse *Y* als einzige keine Bindungsaffinität zu den gebotenen Peptiden aufwiesen [17]. Klasse *Nicht-Y* lässt sich hierauf in die (Sub-)Klassen *X_ORS* mit atypischen und *Nicht-X_ORS* mit typischen Bindungseigenschaften ihrer Domänen aufteilen. Innerhalb der Klasse *Nicht-X_ORS* kann wiederum binär zwischen den Klassen mit Typ-I- (zusammengefasst als Klasse *1*) und denen mit Typ-II-

Orientierung ihrer Liganden (zusammengefasst als Klasse *2*) unterschieden werden. Zum Schluss gliedert sich noch Klasse *1* in die Subklassen *1R*, *1K* und *1@* bzw. Klasse *2* in die Subklassen *2R*, *2K* und *2D*. Da diese Stufe der Taxonomie allerdings durch je drei Subklassen definiert wird, musste hier jeweils wieder auf OvO-Multiclass-Analysen zurückgegriffen werden. Nachdem sich von einem Zusammenschluss jeweils zweier der drei Subklassen von Klasse *1* bzw. *2* mit anschließender binärer Differenzierung dieser von der jeweils verbleibenden Subklasse bessere Relevanzwerte hinsichtlich dieser verbleibenden Subklasse im Vergleich mit der entsprechenden OvO-Multiclass-Subklassifikation erhofft wurden, sollte evaluiert werden, inwiefern sich hiermit die Ergebnisse der OvO-Multiclass-Subklassifikationen möglicherweise präzisieren ließen. Ziel war es Testsequenzen nach den beiden OvO-Multiclass-Subklassifikationen in einem weiteren Schritt anhand solcher binärer Subklassifikationen noch hinsichtlich ihrer tatsächlichen Zugehörigkeit zu den ihnen in den OvO-Multiclass-Subklassifikationen jeweils zugeschriebenen Subklassen zu überprüfen (Abb. 4.2). Dies erschien allerdings nur dann sinnvoll, wenn die binäre Subklassifikation einerseits tatsächlich bessere Relevanzwerte und andererseits auch ein höheres F-Maß hinsichtlich dieser Subklasse liefert.

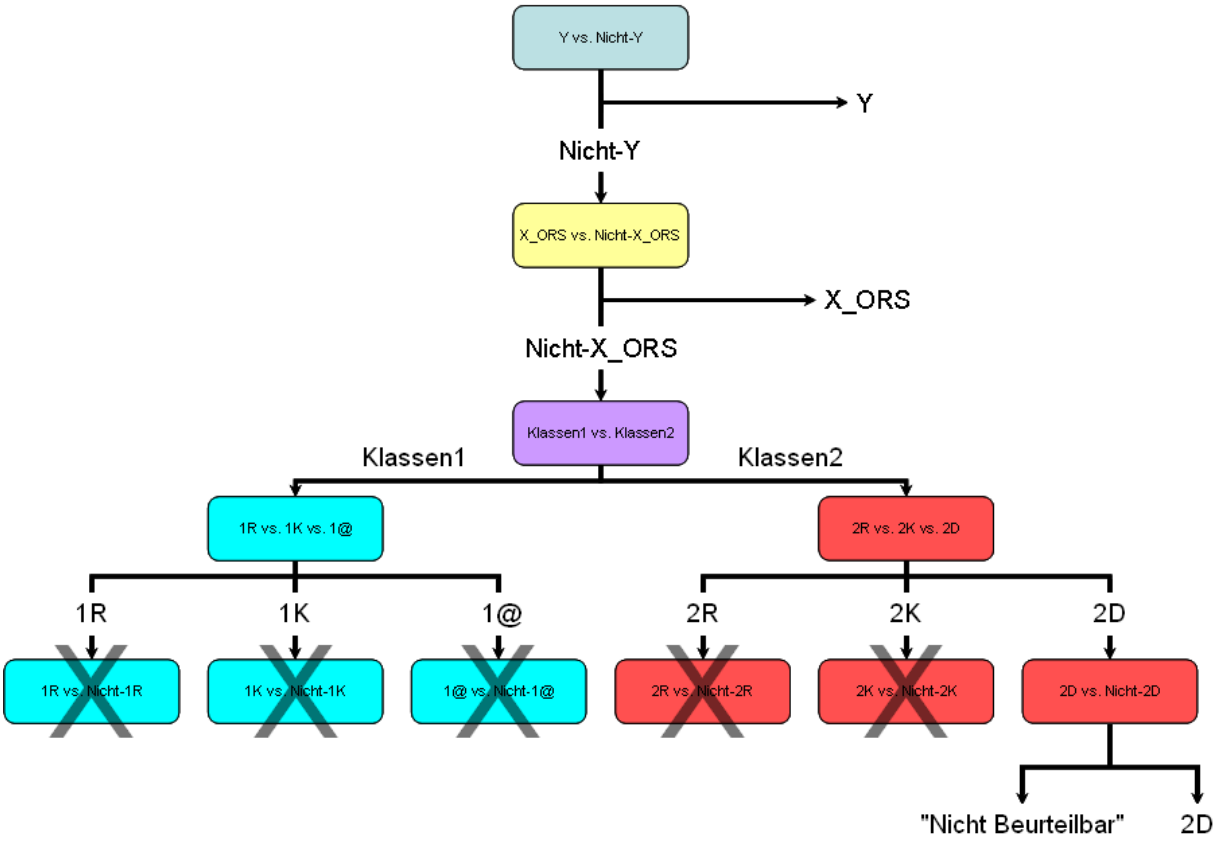


Abb. 4.2 Arbiträre Aufteilung des Multiclass-Problems entsprechend der Taxonomie der Klassen hinsichtlich ihrer Bindungsspezifität auf mehrere, möglichst binäre Subklassifikationsschritte. Die grauen Kreuze markieren die Subklassifikationsschritte, deren Integration in den arbiträr aufgebauten Gesamtklassifikator sich als nicht sinnvoll erwies. Eine Testsequenz, welche im Klassifikationsschritt *„2D vs. Nicht-2D"* der Klasse *Nicht-2D* zugeordnet wurde, wurde letztendlich mit „Nicht Beurteilbar" bzw. „Not Detectable" betitelt.

Zur Erstellung der Subklassifikatoren sollte zunächst ermittelt werden, welche der drei Regularisierungsvarianten (**R0**, **R20**, **R100**) bzw. welche der beiden Kernel-Funktionen (linear bzw. radial) sich hinsichtlich der einzelnen Subklassifikationsschritte jeweils am besten eignet. Daher wurde anhand der entsprechenden Trainingssätze der Regularisierungsstufen **R0**, **R20** bzw. **R100** bezüglich jedes Subklassifikationsschritts für jede Regularisierungsvariante je ein linearer und ein radialer Subklassifikator erstellt. Von den Sequenzen der Trainingssätze kamen dabei stets nur die derjenigen Klassen zum Einsatz, welche auch im jeweiligen Subklassifikationsschritt zu differenzieren waren. Anschließend wurde jede Subklassifikatorvariante anhand der Basissequenzen der im jeweiligen Subklassifikationsschritt jeweils zu differenzieren Klassen im Sinne einer Leave-One-Out Kreuzvalidierung getestet. Im Falle von Subklassifikatorvarianten mit künstlichen Sequenzen wurde hierbei (wie auch im Folgenden) stets nach den in Kapitel 4.1.2.1 beschriebenen Prinzipien verfahren. Obwohl sich in den binären Subklassifikationsschritten die jeweils zu differenzierenden Klassen hinsichtlich ihrer Größe zumeist deutlich voneinander unterschieden, musste dennoch von einer Nivellierung derselben nicht nur bei Regularisierungsvariante **R0**, sondern in den meisten Fällen – außer im Falle des Subklassifikationsschritts *„Klassen 1 vs. Klassen 2“* – auch bei Variante **R20** bzw. **R100** abgesehen werden. Dies begründet sich damit, dass eine Modifikation der Klassengrößen auch eine Veränderung der Informationen innerhalb der Klassen mit sich gebracht hätte, wodurch hierunter gewonnene Daten bzw. Resultate nicht mehr mit denen der reinen OvO-Multiclass-Klassifikatoren aus Testreihe 1 und 2 (Kapitel 4.1.2 und 4.1.2.1) vergleichbar gewesen wären. Die Anwendung künstlicher Sequenzen per se erschien aber dennoch sinnvoll, da sich unter ihrer Anwendung im Falle eines generellen overfitting trotz allem bessere Ergebnisse erhofft wurden[*].

Anhand der Ergebnisse dieser Untersuchungen galt es nun jeweils die für ihren Subklassifikationsschritt geeignetste Subklassifikatorvariante zu bestimmen. Dies war jedoch nicht immer eindeutig möglich: Hinsichtlich des Differenzierungsschritts „*Y vs. Nicht-Y*“ wurde sich – obwohl sie formal nur die zweitbesten Ergebnisse lieferte – für die Variante linear mit Regularisierungsstufe **R20** entschieden, da hier da hier die Sensitivität für Sequenzen der Klasse *Y* auffallend höher lag. Bezüglich des Subklassifikationsschritts *„X_ORS vs. Nicht-X_ORS“* wurde die präziseste Klassifikation von mehreren Varianten mit äquivalenten Ergebnissen erzielt (linear mit Regularisierungsstufe **R0** bzw. **R100** und radial mit Regularisierungsstufe **R100**). Daher wurde sich hier heuristisch für die Variante linear mit Regularisierungsstufe **R100** entschieden. Gleiches konnte auch im Falle des Subklassifikationsschritts *„2D vs. Nicht-2D“* beobachtet werden, bei welchem dies auf die Varianten linear mit Regularisierungsstufe **R0** und **R20** zutraf. Daher fiel hier die Wahl (ebenso heuristisch) auf den linearen

[*] Die hierfür verwendeten Programme tragen jeweils die Bezeichnung „Test-“ kombiniert mit dem Namen des Subklassifikationsschritts sowie der verwendeten Kernel-Funktion und Höhe der Regularisierungsstufe (also 0, 20 bzw. 100), wobei die Subklassifikationsschritte *„1R vs. 1K vs. 1@“* mit „Klassen1“ und *„2R vs. 2K vs. 2D“* mit „Klassen2“ bezeichnet sind. Die Programme finden sich im Anhang auf DVD unter: „~\Klassifikatoren\Kreuzvalidierung_Klassifikatoren“ (Kapitel 7.1.3.1 und 7.2).

Subklassifikator mit Regularisierungsstufe **R20**. Hinsichtlich der restlichen Subklassifikationsschritte – abgesehen *„1@ vs. Nicht-1@“* – waren die Ergebnisse eindeutig (Tabelle 4.3). Da allerdings bereits in der OvO-Multiclass-Analyse *„1R vs. 1K vs. 1@“* hinsichtlich der Klasse *1@* eine Relevanz von 100% erzielt werden konnte, war die Implementation des Subklassifikationsschritts *„1@ vs. Nicht-1@“* in den Gesamtklassifikator ohnehin nicht sinnvoll. Gleiches traf auch auf die Relevanzwerte der Klassen *1K* bzw. *2K* hinsichtlich der OvO-Multiclass-Analysen *„1R vs. 1K vs. 1@“* bzw. *„2R vs. 2K vs. 2D“* zu, sodass auch auf eine Anwendung der Schritte *„1K vs. Nicht-1K“* und *„2K vs. Nicht-2K“* verzichtet werden konnte. Letztlich erschien hinsichtlich dieser Stufe der Klassifikation nur die Integration des Subklassifikationsschritts *„2D vs. Nicht-2D“* sinnvoll zu rechtfertigen, da nur dort Ergebnisse erzielt werden konnten, die die beschriebenen Kriterien erfüllten (Tabelle 4.4)[*].

Durch Kombination der entsprechenden Subklassifikatorvarianten nach der in Abb. 4.2 dargestellten Reihenfolge konnte schließlich der arbiträr aufgebaute Gesamtklassifikator erstellt werden. Die Validierung des Gesamtklassifikators erfolgte anhand sämtlicher Basissequenzen, wiederum im Sinne einer Leave-One-Out Kreuzvalidierung. Dabei wurden Testsequenzen, die in der Subklassifikation *„2D vs. Nicht-2D“* der Klasse *Nicht-2D* zugeordnet wurden, mit „Nicht Beurteilbar“ bzw. „Not Detectable“ betitelt. Zwar hätten diese Sequenzen auch als zugehörig zu Klasse *2R* bzw. *2K* bezeichnet werden können, da sie jedoch in der OvO-Multiclass-Subklassifikation *„2R vs. 2K vs. 2D“* als eben nicht zu diesen gehörig identifiziert wurden, erschien die Bezeichnung „Nicht Beurteilbar“ sinnvoller (Testreihe 3)[†].

Der Vergleich der Klassifikationsergebnisse des arbiträren Gesamtklassifikators mit denen des bislang präzisesten Klassifikators (lineare OvO-Multiclass-Differenzierung aller Klassen mit Regularisierungsstufe **R100** aus Testreihe 2; Kapitel 4.1.2.1) zeigte, dass die Anwendung des arbiträren Klassifikationsansatzes eine weitere, erhebliche Verbesserung der Klassifikationspräzision mit sich brachte (Tabelle 4.5). So ließ sich die Gesamttrefferquote hierunter um weitere 9,81% (von 54,90% auf 64,71%) steigern. Dies ist insbesondere auf die hier deutlich höhere Sensitivität bezüglich Klasse *1R* und *2R* zurückzuführen, welche hinsichtlich beider Klassen jeweils um 28,57% (von 50,00% auf 78,57% bzw. von 42,86% auf 71,43%) anstieg.

---

[*] Die Dateien mit den Confusion-Maps der Validierung der einzelnen Subklassifikatorvarianten tragen jeweils die Bezeichnung „Erg-“ kombiniert mit dem Namen des Subklassifikationsschritts sowie der verwendeten Kernel-Funktion und Höhe der Regularisierungsstufe (also 0, 20 bzw. 100), wobei die Subklassifikationsschritte *„1R vs. 1K vs. 1@“* mit „Klassen1“ und *„2R vs. 2K vs. 2D“* mit „Klassen2“ bezeichnet sind. Sie finden sich im Anhang auf DVD unter: „~\Ergebnisse_Klassifikatoren\Kreuzvalidierung_Klassifikatoren“ (Kapitel 7.1.4 und 7.3).

[†] Programm: „Test-Arbitraer“, im Anhang auf DVD unter: „~\Klassifikatoren\Kreuzvalidierung_Klassifikatoren“ (Kapitel 7.1.3.1 und 7.2).

Subklassifikationsschritt	Variante	Gesamttreffer-quote	durchschnittliches F-Maß	Gesamt-klassifikator
„1R vs. Nicht-1R"	linear **R0**	**77,27%**	72,70%	--
	linear **R20**	**77,27%**	**76,84%**	
	linear **R100**	63,64%	62,39%	
	radial **R0**	63,64%	38,89%	
	radial **R20**	36,36%	26,67%	
	radial **R100**	68,18%	68,12%	
„1K vs. Nicht-1K"	linear **R0**	**86,36%**	66,15%	--
	linear **R20**	**86,36%**	**74,52%**	
	linear **R100**	81,82%	61,40%	
	radial **R0**	81,82%	45,00%	
	radial **R20**	81,82%	45,00%	
	radial **R100**	**86,36%**	66,15%	
„1@ vs. Nicht-1@"	linear **R0**	**90,91%**	**80,70%**	--
	linear **R20**	86,36%	74,52%	
	linear **R100**	86,36%	74,52%	
	radial **R0**	81,82%	45,00%	
	radial **R20**	**90,91%**	**80,70%**	
	radial **R100**	86,36%	79,05%	
„2R vs. Nicht-2R"	linear **R0**	66,67%	66,67%	--
	linear **R20**	**80,00%**	**79,64%**	
	linear **R100**	53,33%	49,76%	
	radial **R0**	33,33%	25,00%	
	radial **R20**	53,33%	34,78%	
	radial **R100**	60,00%	48,86%	
„2K vs. Nicht-2K"	linear **R0**	80,00%	44,44%	--
	linear **R20**	80,00%	64,00%	
	linear **R100**	73,33%	42,31%	
	radial **R0**	80,00%	44,44%	
	radial **R20**	**86,67%**	**71,15%**	
	radial **R100**	66,67%	53,42%	
„2D vs. Nicht-2D"	linear **R0**	**100%**	**100%**	linear **R20****
	linear **R20**	**100%**	**100%**	
	linear **R100**	86,67%	86,11%	
	radial **R0**	66,67%	40,00%	
	radial **R20**	66,67%	40,00%	
	radial **R100**	93,33%	92,06%	
„Klassen 1" *(„1R vs. 1K vs. 1@")*	linear **R0**	**77,27%**	**63,84%**	linear **R0**
	linear **R20**	68,18%	53,57%	
	linear **R100**	68,18%	61,00%	
	radial **R0**	63,64%	25,93%	
	radial **R20**	13,64%	18,74%	
	radial **R100**	59,09%	51,79%	
„Klassen 2" *(„2R vs. 2K vs. 2D")*	linear **R0**	80,00%	60,78%	linear **R20**
	linear **R20**	**86,67%**	**85,54%**	
	linear **R100**	53,33%	43,16%	
	radial **R0**	46,67%	21,21%	
	radial **R20**	6,67%	7,41%	
	radial **R100**	46,67%	48,42%	
„Klassen 1 vs. Klassen 2"	linear **R0**	72,97%	71,97%	radial **R100**
	linear **R20**	75,68%	75,02%	
	linear **R100**	72,97%	71,27%	
	radial **R0**	59,46%	37,29%	
	radial **R20**	10,81%	10,55%	
	radial **R100**	**86,49%**	**86,12%**	

Subklassifikationsschritt	Variante	Gesamttreffer-quote	durchschnittliches F-Maß	Gesamt-klassifikator
„X_ORS vs. Nicht-X_ORS"	linear **R0**	**86,36%**	**66,15%**	linear **R100****
	linear **R20**	68,18%	51,11%	
	linear **R100**	**86,36%**	**66,15%**	
	radial **R0**	84,09%	45,68%	
	radial **R20**	81,82%	45,00%	
	radial **R100**	**86,36%**	**66,15%**	
„Y vs. Nicht-Y"	linear **R0**	**94,12%**	**84,72%**	linear **R20****
	linear **R20**	90,20%	82,35%	
	linear **R100**	**94,12%**	**84,72%**	
	radial **R0**	86,27%	46,32%	
	radial **R20**	**94,12%**	**84,72%**	
	radial **R100**	**94,12%**	**84,72%**	

Tabelle 4.3 Gesamttrefferquoten und durchschnittliche F-Maße der einzelnen Subklassifikatorvarianten mit Hervorhebung der jeweils höchsten Werte bezüglich eines Subklassifikationsschritts. Die jeweils in den arbiträren Gesamtklassifikator integrierte Variante ist in der rechten Spalte aufgeführt. Bezüglich der Gründe für die Implementation der mit ** markierten Varianten sei auf die genaueren Ausführungen im Text verwiesen.

Die Sensitivitätswerte der weiteren Klassen zeigten sich allerdings in den meisten Fällen mit der OvO-Multiclass-Differenzierung vergleichbar.

Auch die Klassifikationsgenauigkeit profitierte von der Anwendung des arbiträren Ansatzes. So konnte bei vier der insgesamt acht Klassen eine Steigerung der Relevanz beobachtet werden: Klasse *1R*: von 35,00% auf 52,38%, Klasse *2R*: von 42,86% auf 83,33%, Klasse *1@*: von 50,00% auf 100% und Klasse *X_ORS*: von 50,00% auf 66,67%. Diesbezüglich erwies sich auch die Implementation des Subklassifikationsschritts *„2D vs. Nicht-2D"* von Vorteil, da zwei zuvor fälschlicherweise der Klasse *2D* zugeordnete Sequenzen hierunter nun korrekt als nicht zu dieser gehörig identifiziert werden konnten (Tabelle 4.6). Nur im Falle der Klasse *Y* waren die Relevanzwerte schlechter als unter der reinen OvO-Multiclass-Differenzierung (von 85,71% auf 60,00%), was möglicherweise auf die Problematik der ungleichen Klassengrößen zurückzuführen sein könnte. Vergleicht man nämlich die Ergebnisse sämtlicher Subklassifikatorvarianten des Differenzierungsschritts „*Y vs. Nicht-Y*" mit denen der präzisesten OvO-Multiclass-Differenzierung, so fällt auf, dass die meisten von ihnen deutlich schlechtere Sensitivitätswerte bezüglich Klasse *Y* lieferten. Nur im Falle der genutzten Variante (linear mit Regularisierungsstufe **R20**) konnte eine vergleichbare Sensitivität hinsichtlich der Klasse *Y* erzielt werden, allerdings unter Inkaufnahme einer schlechteren Relevanz bezüglich dieser.

Die Berechnung des durchschnittlichen F-Maßes unterstreicht die bereits beschriebenen Ergebnisse nochmals; so stieg auch dies unter Anwendung des arbiträren Klassifikators um weitere 9,31% (von 54,07% auf nun 63,38%). Die Ergebnisse der arbiträren Klassifikation bestätigen also, dass eine Integration taxonomischer Informationen in den Klassifikationsprozess von Vorteil sein dürfte. Zwar konnte hierunter nicht bezüglich aller Klassen eine Verbesserung der Klassifikationsleistung erzielt werden, jedoch war die Klassifikationspräzision insgesamt wesentlich höher als bei der reinen OvO-Multiclass-Differenzierung.

Subklassifikationsschritt	Variante	Klasse	Sensitivität	Relevanz	F-Maß
„Klassen 1" (*„1R vs. 1K vs. 1@"*)	linear **R0**	*1R:*	100%	73,68%	84,85%
		1K:	25,00%	100%	40,00%
		1@:	50,00%	100%	66,67%
„Klassen 2" (*„2R vs. 2K vs. 2D"*)	linear **R20**	*2R:*	85,71%	85,71%	85,71%
		2K:	66,67%	100%	80,00%
		2D:	100%	83,33%	90,91%

Subklassifikationsschritt	Variante	Klasse	Sensitivität	Relevanz	F-Maß
„1R vs. Nicht-1R"	linear **R0**	*1R:*	92,86%	76,47%	83,87%
	linear **R20**		71,43%	90,91%	80,00%
	linear **R100**		64,29%	75,00%	69,23%
	radial **R0**		100%	63,64%	77,78%
	radial **R20**		0,00%	--	--
	radial **R100**		50,00%	100%	66,67%
„1K vs. Nicht-1K"	linear **R0**	*1K:*	25,00%	100%	40,00%
	linear **R20**		50,00%	66,67%	57,14%
	linear **R100**		25,00%	50,00%	33,33%
	radial **R0**		0,00%	--	--
	radial **R20**		0,00%	--	--
	radial **R100**		25,00%	100%	40,00%
„1@ vs. Nicht-1@"	linear **R0**	*1@:*	50,00%	100%	66,67%
	linear **R20**		50,00%	66,67%	57,14%
	linear **R100**		50,00%	66,67%	57,14%
	radial **R0**		0,00%	--	--
	radial **R20**		50,00%	100%	66,67%
	radial **R100**		75,00%	60,00%	66,67%
„2R vs. Nicht-2R"	linear **R0**	*2R:*	71,43%	62,50%	66,67%
	linear **R20**		71,43%	83,33%	76,92%
	linear **R100**		28,57%	50,00%	36,36%
	radial **R0**		0,00%	0,00%	--
	radial **R20**		0,00%	--	--
	radial **R100**		14,29%	100%	25,01%
„2K vs. Nicht-2K"	linear **R0**	*2K:*	0,00%	--	--
	linear **R20**		33,33%	50,00%	40,00%
	linear **R100**		0,00%	0,00%	--
	radial **R0**		0,00%	--	--
	radial **R20**		33,33%	100%	50,00%
	radial **R100**		33,33%	25,00%	28,57%
„2D vs. Nicht-2D"	linear **R0**	*2D:*	100%	100%	100%
	linear **R20**		100%	100%	100%
	linear **R100**		100%	71,43%	83,33%
	radial **R0**		0,00%	--	--
	radial **R20**		0,00%	--	--
	radial **R100**		80,00%	100%	88,89%

Tabelle 4.4 Vergleich der beiden für den arbiträren Gesamtklassifikator ausgewählten OvO-Multiclass-Subklassifikatoren *„1R vs. 1K vs. 1@"* bzw. *„2R vs. 2K vs. 2D"* (obere Tabelle) mit den binären Subklassifikatorvarianten der ihnen jeweils nachgeschalteten Subklassifikationsschritte (untere Tabelle). Aufgeführt sind jeweils die Sensitivitäten, Relevanzen und F-Maße der Subklassen von Klasse *1* bzw. *2*. Es zeigt sich, dass lediglich im Falle der Klasse *2D* die Implementation des entsprechenden binären Subklassifikationsschritts (*„2D vs. Nicht-2D"*) in den arbiträren Gesamtklassifikator sinnvoll war, da nur dort bessere Relevanzwerte kombiniert mit höheren F-Maßen bezüglich der betreffenden Subklasse erzielt werden konnten.

Lineare OvO-Multiclass-Differenzierung aller Klassen (R100)

	pred1							
true1	1R	2R	1K	2K	1@	2D	X_ORS	Y
1R	7	3	0	0	1	0	3	0
2R	3	3	0	0	1	0	0	0
1K	2	0	2	0	0	0	0	0
2K	2	1	0	0	0	0	0	0
1@	2	0	0	0	2	0	0	0
2D	0	0	0	0	0	5	0	0
X_ORS	3	0	0	0	0	0	3	1
Y	1	0	0	0	0	0	0	6

Arbiträres Klassifikationsmodell

	pred1								
true1	1R	2R	1K	2K	1@	2D	X_ORS	Y	Not Detectable
1R	11	1	0	0	0	0	1	1	0
2R	0	5	0	0	0	0	0	1	1
1K	3	0	1	0	0	0	0	0	0
2K	2	0	0	1	0	0	0	0	0
1@	2	0	0	0	2	0	0	0	0
2D	0	0	0	0	0	5	0	0	0
X_ORS	3	0	0	0	0	0	2	2	0
Y	0	0	0	0	0	0	0	6	1

Tabelle 4.5 Vergleich der Resultate der linearen OvO-Multiclass-Differenzierung aller Klassen unter Regularisierungsstufe **R100** von Testreihe 2 (oben) mit denen des arbiträren Klassifikationsansatzes von Testreihe 3 (unten).
pred1 = Klassenzuordnung durch Klassifikator, true1 = wahre Klassenzugehörigkeit
(„Erg-AlleKlassen-Linear_100“ bzw. „Erg-Arbitraer“, im Anhang auf DVD unter: „~\Ergebnisse_Klassifikatoren\Kreuzvalidierung_Klassifikatoren“; Kapitel 7.1.4 und 7.3)

„2D vs. Nicht-2D“
(nach OvO-Multiclass-Analyse *„2R vs. 2K vs. 2D“*)

	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]
Pred_Test_Klasse2_bzgl_2D	"2D"	"2D"	"2D"	"2D"	"2D"	"2D"	"2D"
Pred_Test_2D	"Nicht_2D"	"2D"	"2D"	"2D"	"2D"	"2D"	"Nicht_2D"
Sequenz_Nr_bzgl_2D	"17"	"33"	"34"	"35"	"36"	"37"	"45"
True_Class_bzgl_2D	"2R"	"2D"	"2D"	"2D"	"2D"	"2D"	"Y"

Tabelle 4.6 Subklassifikationsergebnisse des arbiträren Gesamtklassifikators bezüglich des Differenzierungsschritts *„2D vs. Nicht-2D“*. Sequenzen, welche in der OvO-Multiclass-Subklassifikation *„2R vs. 2K vs. 2D“* der Klasse *2D* zugeordnet wurden, wurden hiermit noch auf ihre tatsächliche Zugehörigkeit zu dieser Klasse überprüft. Pred_Test_Klasse2_bzgl_2D = Klassenzuordnung der betreffenden Sequenz seitens der OvO-Multiclass-Subklassifikation *„2R vs. 2K vs. 2D“*; Pred_Test_2D = Klassenzuordnung der betreffenden Sequenz seitens des Subklassifikators *„2D vs. Nicht-2D“*; Sequenz_Nr_bzgl_2D = Sequenznummer der betreffenden Sequenz im Alignment der Basissequenzen; True_Class_bzgl_2D = wahre Klassenzugehörigkeit der betreffenden Sequenz.
(„Erg-Arbitraer“, im Anhang auf DVD unter: „~\Ergebnisse_Klassifikatoren\Kreuzvalidierung_Klassifikatoren“; Kapitel 7.1.4 und 7.3)

4.1.2.3 Klassifikation nach Feature Selection

Neben Erhöhung der Datenmenge durch Einbeziehung künstlicher Sequenzen sollte der Problematik eines overfitting nun auch durch Reduktion der Datendimensionalität begegnet werden. Dies sollte zunächst anhand einer relativ simplen Variante erfolgen, bei der die Datendimensionen auf ausschließlich die Spalten der Fisher-Score-Matrices mit den 500 größten Varianzwerten reduziert werden. Da die Auswahl der betreffenden Fisher-Score-Spalten nicht durch Informationen von zu testenden Sequenzen beeinflusst werden durfte, sollte sie stets ausschließlich anhand des jeweiligen Trainingssatzes erfolgen. Die Anwendung dieser Variante einer Feature Selection auf die reinen OvO-Multiclass-Klassifikatoren mit Regularisierungsstufe **R0** von Testreihe 2 (Testreihe 4)* dokumentierte jedoch keine wegweisende Verbesserung, sodass auf eine zusätzliche, mit deutlichem Mehraufwand verbundene Anwendung auf weitere Modelle verzichtet wurde. Bei Vergleich der Ergebnisse – insbesondere der radialen Variante – mit denen der entsprechenden nativen Variante fallen aber dennoch einige Veränderungen auf. Teilte der native Klassifikator mit radialer Kernel-Funktion noch alle Sequenzen der größten Klasse (*1R*) zu, so wurden unter reduzierten Dimensionen bzw. Features einige Sequenzen nun auch anderen Klassen zugeteilt, was möglicherweise doch eine Reduktion des overfitting indizieren könnte. Zudem zeigt sich ein signifikantes Signal bezüglich der Abgrenzungsfähigkeit von Sequenzen der Klasse *Y*, da unter reduzierten Features nun immerhin vier der sieben Testsequenzen von Klasse *Y* korrekt klassifiziert werden konnten.
Allerdings dürfte zwischen der Varianz einer Fisher-Score-Spalte bzw. eines Features und seiner Signifikanz für den Klassifikator kein proportionaler Zusammenhang bestehen, was auch erklären könnte, warum nach Anwendung dieser Variante einer Feature Selection insgesamt dennoch keine wesentliche Verbesserung der Klassifikation zu erzielen war (Tabelle 4.7).

Daher sollte nun eine Feature Selection anhand der wesentlich avancierteren Methoden des Softwarepakets RFE [54][161] erfolgen, um so möglicherweise doch eine weitere Verbesserung der bisherigen Klassifikationsmodelle erzielen zu können. Zum besseren Verständnis soll an dieser Stelle kurz auf die im Rahmen der vorliegenden Arbeit hierzu hauptsächlich zum Einsatz gekommenen Funktionen dieses Softwarepakets eingegangen werden:
Mit der Funktion *rfe.fit* werden die Spalten einer Matrix, welche jeweils den Dimensionen/Features der durch die Zeilen der Matrix definierten Vektoren entsprechen, hinsichtlich einer Differenzierung dieser Vektoren in vordefinierte Klassen, nach ihrer Signifikanz zur Klassifikation beurteilt. Auf den vorliegenden Fall übertragen bedeutet dies, dass sich damit die Spalten der Fisher-Scores eines Sequenzsatzes entsprechend ihrer Signifikanz für den jeweiligen Klassifikator beurteilen lassen.

* Programm: „Test-AlleKlassen-Varianz-LinearRadial_0“, im Anhang auf DVD unter: „~\Klassifikatoren\Kreuzvalidierung_Klassifikatoren“ (Kapitel 7.1.3.1 und 7.2).

OvO-Multiclass-Differenzierung aller Klassen (R0)

Linear

	pred1							
true1	1R	2R	1K	2K	1@	2D	X_ORS	Y
1R	9	2	0	0	0	0	3	0
2R	5	2	0	0	0	0	0	0
1K	2	0	1	0	0	0	1	0
2K	2	1	0	0	0	0	0	0
1@	2	0	0	0	2	0	0	0
2D	0	0	0	0	0	5	0	0
X_ORS	4	1	0	0	0	0	2	0
Y	3	0	0	0	0	0	0	4

Radial

	pred2							
true1	1R	2R	1K	2K	1@	2D	X_ORS	Y
1R	14	0	0	0	0	0	0	0
2R	7	0	0	0	0	0	0	0
1K	4	0	0	0	0	0	0	0
2K	3	0	0	0	0	0	0	0
1@	4	0	0	0	0	0	0	0
2D	5	0	0	0	0	0	0	0
X_ORS	7	0	0	0	0	0	0	0
Y	7	0	0	0	0	0	0	0

OvO-Multiclass-Differenzierung aller Klassen (R0)
(500 Fisher-Score-Spalten mit größten Varianzwerten)

Linear

	pred1							
true1	1R	2R	1K	2K	1@	2D	X_ORS	Y
1R	11	1	0	0	0	0	2	0
2R	5	2	0	0	0	0	0	0
1K	2	0	1	0	0	0	1	0
2K	2	0	1	0	0	0	0	0
1@	2	0	0	0	2	0	0	0
2D	2	0	0	0	0	3	0	0
X_ORS	5	0	0	0	0	0	2	0
Y	2	0	0	0	0	0	1	4

Radial

	pred2							
true1	1R	2R	1K	2K	1@	2D	X_ORS	Y
1R	13	0	0	0	0	0	1	0
2R	7	0	0	0	0	0	0	0
1K	4	0	0	0	0	0	0	0
2K	3	0	0	0	0	0	0	0
1@	4	0	0	0	0	0	0	0
2D	5	0	0	0	0	0	0	0
X_ORS	7	0	0	0	0	0	0	0
Y	3	0	0	0	0	0	0	4

Tabelle 4.7 Gegenüberstellung der Resultate der linearen und radialen OvO-Multiclass-Differenzierung aller Klassen mit Regularisierungsstufe **R0** unter Einbeziehung aller Fisher-Score-Spalten / Features (aus Testreihe 2; oben) bzw. jeweils nur der 500 Features mit den größten Varianzwerten (Testreihe 4; unten). Insgesamt konnte mithilfe dieser Variante einer Feature Selection keine wegweisende Verbesserung der Klassifikation erzielt werden, sodass auf eine Ausweitung dieses Ansatzes verzichtet wurde: Gesamttrefferquote unter Einbeziehung aller Features linear: 49,02% bzw. radial: 27,45%, unter Einbeziehung jeweils nur der 500 Features mit den größten Varianzwerten linear: 49,02% bzw. radial 33,33%. Durchschnittliches F-Maß unter Einbeziehung aller Features linear: 48,10% bzw. radial: 5,38%, unter Einbeziehung jeweils nur der 500 Features mit den größten Varianzwerten linear: 45,92% bzw. radial 14,51%. pred1 = Klassenzuordnung durch Klassifikator, true1 = wahre Klassenzugehörigkeit („Erg-AlleKlassen-Linear_0“, „Erg-AlleKlassen-Radial_0“, „Erg-AlleKlassen-Varianz-Linear_0“, „Erg-AlleKlassen-Varianz-Radial_0“, im Anhang auf DVD unter: „~\Ergebnisse_Klassifikatoren\Kreuzvalidierung_Klassifikatoren“; Kapitel 7.1.4 und 7.3)

Hierfür wird zunächst ein SVM-Modell anhand der unveränderten Matrix trainiert und die Features diesem Modell entsprechend ihrer Signifikanz nach beurteilt. Je nach gewählter Variante *(speed = „high“* bzw. *„low“*) werden nun ***n*** der als am unwichtigsten erachteten Features aus der Matrix entfernt und anhand der modifizierten Matrix erneut ein SVM-Modell trainiert. Die Features dieser Matrix werden sodann wiederum ihrer Signifikanz nach beurteilt und so weiter. Diese Schritte werden solange wiederholt bis nur noch ein einziges Features übrig ist. Die Varianten *speed = „high“* bzw. *„low“*

definieren eine logarithmische bzw. Leave-One-Out Strategie bei der Elimination der als am unwichtigsten erachteten Features.
Mit der Funktion *rfe.ae* werden die durch *rfe.fit* erstellten Modelle anhand sämtlicher Vektoren der (modifizierten) Matrix validiert. Hieraus lässt sich ersehen, mit welcher Feature-Menge die beste Klassifikation möglich bzw. welche Feature-Menge für eine valide Klassifikation mindestens erforderlich ist. Bezüglich dieser Funktion gilt es jedoch zu bedenken, dass die Modelle mit den ebengleichen Vektoren bzw. Sequenzen getestet werden, anhand welcher sie auch trainiert wurden, das heißt eine strenge Teilung zwischen Trainings- und Testdaten findet nicht statt.
Im Gegensatz hierzu findet dies bei der Funktion *rfe.cv* Berücksichtigung, da hier die Feature Selection dem Prinzip einer Kreuzvalidierung folgt. Die Vektoren der Matrix werden also zunächst auf die gewünschte Anzahl an Teilmengen verteilt. Im Anschluss werden sukzessive bezüglich jeder Trainingsmenge die Features nach den Methoden der Funktion *rfe.fit* beurteilt und die hierbei (anhand der jeweiligen Trainingsmenge) erstellten Modelle mithilfe der entsprechenden Testmenge validiert. Zum Schluss werden für jede der getesteten Feature-Mengen aus den Werten der einzelnen Testläufe der Kreuzvalidierung der durchschnittliche Gesamt-Error, der durchschnittliche Individual-Error bezüglich jeder Klasse und der durchschnittliche Standardfehler der durchschnittlichen Gesamt-Errors errechnet. Diese Funktion unterscheidet also nicht nur bei der Validierung der Modelle sondern auch bei der Bestimmung der wichtigsten Features streng zwischen Trainings- und Testdaten. Daraus folgt jedoch auch, dass die hierunter als am signifikantesten ermittelten Features nicht unbedingt mit denen übereinstimmen müssen, welche hinsichtlich aller Vektoren errechnet würden. Das bedeutet, dass mit *rfe.cv* zwar eine präzisere Validierung der Feature Selection als mit *rfe.ae* gelingt, allerdings sind die für jede Trainingsmenge angegebenen Features nicht ohne Weiteres auf das Gesamtmodell übertragbar.

Um den Effekt einer Feature Selection mit RFE [54][161] auf die vorliegende Problematik überhaupt abschätzen zu können, sollte diese mithilfe von *rfe.cv* in erster Instanz nur an den beiden Ausgangsklassifikatoren (lineare und radiale OvO-Multiclass-Differenzierung aller Klassen mit Regularisierungsstufe **R0** aus Testreihe 1) erfolgen. Hierzu war es zunächst nötig die einzelnen (Unter-)Funktionen von RFE [54][161] nach der in Kapitel 3.2 beschriebenen Weise entsprechend anzupassen. Bezüglich der Validierung wurde sich den Ausgangsklassifikatoren bzw. dem Standard des Programms entsprechend für eine zehnfache Kreuzvalidierung entschieden. Da die anfallende Datenmenge unter einer Leave-One-Out Strategie zur Reduktion der Features (*speed = „low“*) die Kapazität des benutzten Rechnersystems überschritt, mit dieser jedoch präzisere Ergebnisse als unter logarithmischer Reduktion der Features (*speed = „high“*) zu erwarten sind, wurde sich eines Kompromisses bedient, der beide Varianten miteinander kombiniert. So wurde eine Funktion erstellt, die zunächst einen ersten Testlauf von *rfe.cv* unter logarithmischer Reduktionsvariante durchführt. Im nächsten Schritt werden entsprechend der kleinsten Menge an Features, mit der zuvor die durchschnittlich niedrigste

Fehlklassifikation erreicht wurde, die hinsichtlich der einzelnen Trainingsmengen jeweils signifikantesten Features gewählt und ihre Matrices entsprechend modifiziert, das heißt alle anderen Features jeweils entfernt. Anhand dieser Matrices wird nun erneut eine Feature Selection mit *rfe.cv* durchgeführt, diesmal jedoch unter Leave-One-Out Reduktionsvariante. Dieser Schritt erfolgt allerdings nur, falls die betreffende Feature-Menge mehr als ein und weniger als 512 Features umfasst, andernfalls wird sich auf die logarithmische Variante beschränkt. Aus den insgesamt zehn Durchläufen von *rfe.cv* im zweiten Schritt wird dann der durchschnittliche Gesamt-Error bezüglich jeder Menge an Features errechnet. Von der kleinsten Feature-Menge mit dem durchschnittlich niedrigsten Gesamt-Error werden zudem der durchschnittliche Individual-Error bezüglich jeder Klasse und der durchschnittliche Standardfehler des betreffenden durchschnittlichen Gesamt-Errors berechnet (Testreihe 5)[*].

Aus den Ergebnissen dieser Testreihe[†] lässt sich ersehen, dass der Einsatz von RFE [54][161] hier hinsichtlich beider Klassifikatoren zu einer erheblichen Verbesserung der Klassifikationspräzision führte. So lag der durchschnittlich niedrigste Gesamt-Error bei Anwendung auf den linearen Klassifikator beispielsweise bei nur 37,90% und wurde unter einer Menge von lediglich acht Features erzielt. Im Vergleich mit den Ergebnissen aus der zehnfachen Kreuzvalidierung der nativen Variante des linearen Klassifikators (Tabelle 4.1), bei der der Gesamt-Error noch 49,02% betrug, entspricht dies einer Verbesserung von 11,12%. Dies spiegelt sich auch im Individual-Error der einzelnen Klassen wider, der – außer im Falle der Klasse *2D* – hinsichtlich aller Klassen geringer war als bei der nativen Variante. Aber auch der radiale Klassifikator profitierte deutlich von der Feature Selection mit RFE [54][161], wenngleich hier 47 Features für den durchschnittlich niedrigsten Gesamt-Error von 49,07% nötig waren. Vergleicht man hier nämlich den Individual-Error jeder Klasse mit dem entsprechenden aus der zehnfachen Kreuzvalidierung der nativen Variante des radialen Klassifikators (Tabelle 4.1), so lässt sich feststellen, dass im Gegensatz zu dieser nun doch hinsichtlich einiger Klassen relativ reliable Ergebnisse erzielt werden konnten.
Der Gesamt-Error des linearen Klassifikators war nach Feature Selection sogar um 7,20% niedriger als der des bisher zuverlässigsten OvO-Multiclass-Klassifikators zur Differenzierung aller Klassen (linearer OvO-Multiclass-Klassifikator mit Regularisierungsstufe **R100**, Kapitel 4.1.2.1) und um nur 2,61% höher als der des insgesamt präzisesten Klassifikators (arbiträrer Klassifikator aus Testreihe 3, Kapitel 4.1.2.2). Allerdings gilt es zu beachten, dass die Validierung dieser Klassifikatoren jeweils

---

[*] Programme: „Test-AlleKlassen-RFE-10CV-Linear_0“ und „Test-AlleKlassen-RFE-10CV-Radial_0“, im Anhang auf DVD unter: „~\Klassifikatoren\Kreuzvalidierung_Klassifikatoren“ (Kapitel 7.1.3.1 und 7.2)

[†] Die Ergebnisse von „Test-AlleKlassen-RFE-10CV-Linear_0“ und „Test-AlleKlassen-RFE-10CV-Radial_0“ finden sich im Anhang auf DVD unter: „~\Ergebnisse_Klassifikatoren\Kreuzvalidierung_Klassifikatoren\Erg-AlleKlassen-RFE-10CV-Linear_0“ bzw. „~\Ergebnisse_Klassifikatoren\Kreuzvalidierung_Klassifikatoren\Erg-AlleKlassen-RFE-10CV-Radial_0“ (Kapitel 7.1.4 und 7.3).

anhand einer Leave-One-Out Kreuzvalidierung erfolgte, sodass die Ergebnisse hieraus nur bedingt mit denen der Testreihe 5 vergleichbar sind.

Um also eine bessere Vergleichbarkeit schaffen, sollte RFE [54][161] nun unter einer Leave-One-Out Kreuzvalidierung durchgeführt werden. Zwar wird durch den Autor des Softwarepakets RFE [54][161] in den Hilfedateien die Option zur Leave-One-Out Kreuzvalidierung beschrieben, jedoch war eine Anwendung dieser trotz multipler Versuche nicht möglich. Eine weitere Problematik dieses Softwarepakets bestand darin, dass bei seiner Anwendung auf Modelle mit künstlichen Sequenzen – selbst wenn die Leave-One-Out Kreuzvalidierung möglich gewesen wäre – keine exakte Validierung möglich schien (Kapitel 5.2.2). Trotz dieser Restriktionen sollte aufgrund der vielversprechenden Ergebnisse von Testreihe 5 evaluiert werden, wie sich eine Anwendung von RFE [54][161] auf das arbiträre Klassifikationsmodell (ohne Einbeziehung künstlicher Sequenzen) auswirkt. Hierzu wurde sich folgendes Konzept überlegt: Zunächst müssten innerhalb des arbiträren Gesamtklassifikators sämtliche Subklassifikatoren mit künstlichen Sequenzen mit der jeweils geeignetsten Variante ohne künstliche Sequenzen (entsprechend der Ergebnisse aus Kapitel 4.1.2.2) ersetzt werden. Da dieses Modell auf der Basis einer Leave-One-Out Kreuzvalidierung arbeitet, müsste dann jeder Subklassifikator für sich ohne die jeweils zu testende Sequenz einer Feature Selection mit RFE[54][161] zugeführt werden, in der die Bestimmung der optimalen Feature-Menge mithilfe von *rfe.cv* und die Beurteilung der Features mithilfe von *rfe.ae* erfolgt. Nach entsprechender Anpassung jedes Subklassifikators, könnte schließlich die Validierung anhand der betreffenden Testsequenz erfolgen. Da dies umzusetzen jedoch sehr aufwendig ist, sollte zunächst geprüft werden, ob mit einem auf diese Weise anhand von RFE [54][161] modifizierten Klassifikator überhaupt bessere Ergebnisse als mit seiner native Variante zu erwarten sind. Als Grundlage dieses Versuchs wurde die lineare OvO-Multiclass-Differenzierung aller Klassen mit Regularisierungsstufe **R0** aus Testreihe 2 (Kapitel 4.1.2.1) gewählt. Aus dem **R0**-Trainingssatz mit seinen 51 Basissequenzen wurden hierfür 51 Trainingsmatrices à 50 Sequenzen erstellt, denen jeweils eine unterschiedliche Basissequenz fehlte, da diese später als Testsequenz der jeweiligen Trainingsmatrix fungieren sollte. Nach entsprechender Modifikation der (Unter-)Funktionen von RFE [54][161] (Kapitel 3.2) erfolgte dann zunächst hinsichtlich jeder Trainingsmatrix für sich die Beurteilung der Features mithilfe von *rfe.ae*, wobei hier nur die Reihenfolge der Features ihrer Signifikanz nach interessierte. Im nächsten Schritt wurden die Trainingsmatrices mit *rfe.cv* in der bereits beschriebenen Weise analysiert, um für jede die minimal notwendige Feature-Menge zur durchschnittlich präzisesten Klassifikation zu bestimmen. Der jeweiligen Menge und Beurteilung von *rfe.ae* hinsichtlich der betreffenden Trainingsmatrix entsprechend wurden dann die jeweils signifikantesten Features gewählt und die einzelnen Trainingsmatrices jeweils auf diese Features

reduziert. Für jede der so modifizierten Trainingsmatrices wurde nun ein SVM-Modell mit linearer Kernel-Funktion trainiert und anhand der entsprechenden Testsequenz validiert (Testreihe 6)*.

Lineare OvO-Multiclass-Differenzierung aller Klassen (R0)
(Testreihe 2)

	pred1							
true1	1R	2R	1K	2K	1@	2D	X_ORS	Y
1R	9	2	0	0	0	0	3	0
2R	5	2	0	0	0	0	0	0
1K	2	0	1	0	0	0	1	0
2K	2	1	0	0	0	0	0	0
1@	2	0	0	0	2	0	0	0
2D	0	0	0	0	0	5	0	0
X_ORS	4	1	0	0	0	0	2	0
Y	3	0	0	0	0	0	0	4

Lineare OvO-Multiclass-Differenzierung aller Klassen (R0)
nach Modifikation mit RFE
(Testreihe 6)

	pred1							
true1	1R	2R	1K	2K	1@	2D	X_ORS	Y
1R	8	2	0	1	0	0	2	1
2R	3	3	0	0	0	0	1	0
1K	1	0	0	1	1	0	1	0
2K	3	0	0	0	0	0	0	0
1@	2	0	0	0	1	0	1	0
2D	0	1	0	0	0	4	0	0
X_ORS	1	0	1	1	0	0	4	0
Y	3	0	0	0	0	0	0	4

$AverageFeatureMenge_linear
[1] 21.88235

	pred1							
true1	1R	2R	1K	2K	1@	2D	X_ORS	Y
1R	9	1	0	2	0	0	1	1
2R	4	2	0	0	0	0	1	0
1K	0	0	1	1	1	0	1	0
2K	2	1	0	0	0	0	0	0
1@	2	0	0	0	1	0	1	0
2D	0	1	0	0	0	4	0	0
X_ORS	2	0	1	1	0	0	3	0
Y	3	0	0	0	0	0	0	4

$AverageFeatureMenge_linear
[1] 33.13725

Tabelle 4.8 Vergleich der Resultate der linearen OvO-Multiclass-Differenzierung aller Klassen unter Regularisierungsstufe **R0** von Testreihe 2 (oben) mit denen zweier Beispieltestläufe der Testreihe 6 (siehe Text; unten). Auch nach mehreren Testläufen konnten anhand von Testreihe 6 keine der nativen Variante von Testreihe 2 überlegenen Resultate erzielt werden.
pred1 = Klassenzuordnung durch Klassifikator, true1 = wahre Klassenzugehörigkeit, AverageFeatureMenge_linear = Durchschnitt der Featuremengen, die von RFE [54][161] jeweils als optimal für die Klassifikation (unter linearer Kernel-Funktion) berechnet wurden..
(„Erg-AlleKlassen-Linear_0“ und „Erg-AlleKlassen-RFE-Linear_0“, im Anhang auf DVD unter: „~\Ergebnisse_Klassifikatoren\Kreuzvalidierung_Klassifikatoren“; Kapitel 7.1.4 und 7.3)

Auch nach mehreren Testläufen lieferte diese Testreihe keine solch überzeugenden Ergebnisse (Tabelle 4.8) wie Testreihe 5. So lag der durchschnittliche Gesamt-Error (nach Berechnung anhand der Ergebnisse der beiden in Tabelle 4.8 exemplarisch dargestellten Testläufe) hier mit 52,94% sogar 1,96% höher als bei der nativen Variante (50,98%). Auch im Vergleich der durchschnittlichen

* Programm: „Test-AlleKlassen-RFE-Linear_0“, im Anhang auf DVD unter: „~\Klassifikatoren\Kreuzvalidierung_Klassifikatoren“ (Kapitel 7.1.3.1 und 7.2).

klassenspezifischen Sensitivitäten und der durchschnittlichen F-Maße war der Klassifikator nach der beschriebenen Modifikation mit RFE [54][161] (durchschnittliche klassenspezifische Sensitivität bzw. durchschnittliches F-Maß aus beiden in Tabelle 4.8 dargestellten Testläufen: 40,13% respektive 42,45%) der nativen Variante (durchschnittliche klassenspezifische Sensitivität: 44,20%, durchschnittliches F-Maß: 48,10%) unterlegen. Interessant ist auch die Beobachtung, dass in dieser Testreihe verglichen mit der linearen Variante aus Testreihe 5 durchschnittlich höhere Feature-Mengen als optimal errechnet wurden, im ersten der beiden Testläufe durchschnittlich 21,88 und im zweiten 33,14 Features. Da die Ergebnisse von Testreihe 6 also suggerierten, dass unter dieser Art der Anwendung von RFE [54][161] keine besseren Ergebnisse als unter der entsprechenden nativen Klassifikatorvariante zu erwarten waren, wurde letztlich auch auf eine Implementation von RFE [54][161] in das arbiträre Klassifikationsmodell verzichtet.

4.1.3 Anwendung des Klassifikators mit der höchsten prognostischen Präzision auf fremde Sequenzen

Da nun das Klassifikationsmodell feststand, welches unter den Bedingungen der Validierung an sich selbst und den Voraussetzungen dieser Arbeit als das Modell mit der höchsten prognostischen Präzision anzusehen war (arbiträrer Klassifikator aus Testreihe 3, Kapitel 4.1.2.2), sollte im nächsten Schritt geprüft werden, inwieweit sich dieses auch zur Klassifikation fremder SH3-Domänen eignet. Zu diesem Zweck wurde ein Satz aus 29 Aminosäuresequenzen von SH3-Domänen gewählt, von denen im Rahmen einer Arbeit von Friedrich, et al. (2006) [46] unter anderem ein positionsspezifisches Interaktionsprofil mit ihren Liganden erstellt wurde. Allerdings war die Klassenzuordnung dieser Sequenzen nicht bekannt, daher sollten sie – soweit möglich entsprechend ihres in der betreffenden PDB-Datei dargestellten Liganden – nach dem Modell von Cesareni, et al. (2002) [17] klassifiziert werden. Hierfür wurden die PDB-Dateien der Sequenzen in PyMOL [28][139] geladen und – falls der Ligand mit dargestellt war – sein Konsensmotiv aus den jeweiligen Aminosäureresten im Bereich der Bindungsstelle abgeleitet. Auf diese Weise ließen sich 22 der insgesamt 29 Domänen einer Klasse zuordnen. Bezüglich solcher Sequenzen, für welche anhand ihres Liganden keine eindeutige Klassenzuordnung möglich war bzw. für welche keine Angaben über ihre jeweiligen Liganden existierten, wurde nach identischen bzw. ähnlichen Sequenzen innerhalb der Basissequenzen gesucht, sodass auch diese klassifiziert werden konnten. Dies betraf die Sequenzen 1IO6, 1QKW, 1I0C, 1JO8, 1JQQ, 1NEG und 1OOT. Dabei entsprachen 1I0C: Eps8_M, 1JO8: ABP-1, 1JQQ: Pex13 und 1OOT: Yfr024c aus den Basissequenzen. 1QKW und 1NEG wiesen sehr hohe sequenzielle Ähnlichkeit mit Spe_C der Basissequenzen auf, sodass diese der Klasse *1@* zugeteilt wurden. Aufgrund des Umstands, dass sich der Ligand der Sequenz 1IO6 nicht mit letzter Sicherheit einer bestimmten Klasse zuordnen ließ, wurde innerhalb der Basissequenzen nach einer ähnlichen bzw. identischen Sequenz gesucht. Da

sie dort einer Sequenz der Klasse *2R* entspricht (GRB2_C_H), erfolgte die Zuordnung zu dieser Klasse (Tabelle 4.9).
Die auf diese Weise klassifizierten Sequenzen konnten nun als Testsequenzen des nach dem Vorbild von Testreihe 3 erstellten, arbiträren Klassifikators eingesetzt wurden. Im Unterschied zu Testreihe 3, konnte hier jedoch bei den Subklassifikatoren mit Regularisierungsstufe **R20** bzw. **R100** hinsichtlich sämtlicher jeweils zu differenzierender Klassen stets der Satz an künstlichen Sequenzen zum Training genutzt werden, der auf der Basis aller Basissequenzen der jeweiligen Klasse emittiert wurde. Ebenso konnten jetzt (bezüglich aller Regularisierungsvarianten) stets alle Basissequenzen der jeweils zu differenzierenden Klassen zum Training eingesetzt werden*.

Die Ergebnisse der Klassifikation dieses Sequenzsatzes anhand des so erstellten Klassifikators (Tabelle 4.9) dokumentieren eine Gesamttrefferquote von 55,17% bei einem durchschnittlichen F-Maß von 59,96%, womit der Klassifikator dem Zufall mehr als viermal überlegen war. Auffällig präzise, sogar fehlerfrei gelang hierbei die Abgrenzung der Klassen *1@*, *1K*, 2D und *Y*. Die Abgrenzung der Klassen *1R*, *2R*, *2K* und *X_ORS* hingegen schien dem Klassifikator deutlich größere Schwierigkeiten zu bereiten, so lag die Relevanz der Klasse *1R* beispielsweise bei nur 40,00% und die Sensitivität für Sequenzen der Klasse *X_ORS* sogar bei 0,00%.
Allerdings ist die Interpretation der Ergebnisse dieser Testklassifikation aus mehreren Gründen kritisch zu betrachten. Vergleicht man nämlich die Sequenzen des genutzten Testsatzes miteinander, so wird ersichtlich, dass sich ein Großteil einander entspricht, das heißt derselben Domäne entstammt. Die Klassen, in die diese Sequenzen entsprechend ihrer dargestellten Liganden eingeordnet wurden, waren jedoch häufig nicht die gleichen. So entsprechen beispielsweise die Sequenzen 1PRM, 1QWF, 1NLP, 1PRL, 1QWE, 1NLO, 1RLQ, 1RLP und 1JEG sämtlich der C-terminalen SH3-Domäne von SRC der Spezies Gallus gallus bzw. des Avian sarcoma virus, werden aber in den entsprechenden PDB-Dateien, die ja zur Klassifizierung der Sequenzen herangezogen wurden, mit unterschiedlichen Liganden, nämlich sowohl der Klasse *1R* als auch der Klassen *2R* und *X_ORS* dargestellt. Dasselbe Problem findet sich bei den Sequenzen 1GBR, 1GBQ und 3GBQ, den Sequenzen 1N5Z und 1JQQ, sowie den Sequenzen 1H3H und 1UTI. 1GBR, 1GBQ und 3GBQ entsprechen alle der N-terminalen SH3-Domäne von GRB2 der Spezies Mus musculus, die PDB-Dateien zeigen jedoch unterschiedliche Liganden, nämlich der Klassen *2K* sowie *2R*. 1N5Z und 1JQQ entstammen beide der SH3-Domäne von Pex13 der Spezies Saccharomyces cerevisiae. Da 1JQQ in seiner PDB-Datei keine Darstellung eines Liganden besitzt, wurde diese Sequenz, entsprechend der Klassifizierung von Pex13 durch Cesareni, et al. (2002) [17], der Klasse *1R* zugeteilt. In der PDB-Datei von 1N5Z jedoch ist ein Ligand der Klasse *2R* dargestellt. 1H3H und 1UTI entsprechen beide der SH3-Domäne des GRB2-RELATED ADAPTOR

* Programm: „Test-Testsatz-Friedrich“, im Anhang auf DVD unter: „~\Klassifikatoren\Validierung_Testsatz-Friedrich“ (Kapitel 7.1.3.2 und 7.2).

PROTEIN 2 der Spezies Mus musculus. Während die PDB-Datei von 1H3H einen Liganden der Klasse *X-ORS* zeigt, zeigt die von 1UTI jedoch einen der Klasse *2R*. Es lässt sich also folgern, dass aufgrund von Kreuzreaktivität vieler Sequenzen anhand der PDB-Dateien keine validen Rückschlüsse auf die Klassenzugehörigkeit einer Sequenz möglich waren. Vielmehr müsste die Klassifizierung der Sequenzen nach der Klasse erfolgen, für die die größte Affinität besteht. Da dies im vorliegenden Fall in den meisten Fällen jedoch nicht möglich war, ist natürlich auch die Interpretation der Ergebnisse der Testklassifikation nur eingeschränkt möglich.

Sequenzen des Testsatzes mit ihrer jeweiligen Klassenzuordnung vs. Zuordnung durch Klassifikator

	Sequenz	Klasse	SVM
1.	1GBR	2K	2K
2.	1IO6	2R	2R
3.	1GBQ	2R	2K
4.	1PRM	2R	1R
5.	1QKW	1@	1@
6.	1QWF	1R	1R
7.	1BBZ	1@	1@
8.	1AZE	2R	2R
9.	1BBZ2	1@	1@
10.	3GBQ	2R	2K
11.	1NLP	X_ORS	1R
12.	1PRL	2R	1R
13.	1QWE	2R	1R
14.	1NLO	X_ORS	1R
15.	1RLQ	1R	1R

	Sequenz	Klasse	SVM
16.	1RLP	1R	1R
17.	1I0C	2D	2D
18.	1JEG	2R	2R
19.	1JO8	2K	2K
20.	1N5Z	2R	1R
21.	1JQQ	1R	1R
22.	1NEG	1@	1@
23.	1H3H	X_ORS	2R
24.	1OOT	2R	2R
25.	1UTI	2R	2R
26.	1T0J	X_ORS	2K
27.	1VYT	X_ORS	2R
28.	1VYU	X_ORS	2R
29.	1T0H	X_ORS	2K

Confusion-Map

	pred1								
true1	1R	2R	1K	2K	1@	2D	X_ORS	Y	Not Detectable
1R	4	0	0	0	0	0	0	0	0
2R	4	5	0	2	0	0	0	0	0
2K	0	0	0	2	0	0	0	0	0
1@	0	0	0	0	4	0	0	0	0
2D	0	0	0	0	0	1	0	0	0
X_ORS	2	3	0	2	0	0	0	0	0

Tabelle 4.9 Resultate der Validierung des arbiträren Klassifikationsmodells aus Testreihe 3 anhand der 29 Testsequenzen aus der Arbeit von Friedrich, et al. (2006) [46]. Oben: tabellarische Auflistung der 29 Testsequenzen – jeweils benannt nach ihrem PDB-Identifikator (Spalte: „Sequenz“) – und Gegenüberstellung ihrer jeweils vermuteten Klassenzugehörigkeit (Spalte: „Klasse“) mit der anhand des Klassifikators jeweils ermittelten (Spalte: „SVM“). Unten: Darstellung der Ergebnisse in Form einer Confusion-Map.
pred1 = Klassenzuordnung durch Klassifikator, true1 = wahre Klassenzugehörigkeit
(„Erg-Testsatz-Friedrich“, im Anhang auf DVD unter: „~\Ergebnisse_Klassifikatoren\Validierung_ Testsatz-Friedrich“; Kapitel 7.1.4 und 7.3).

Eine weitere Limitation bei der Interpretation der Ergebnisse ergibt sich aus der Tatsache, dass einige der Testsequenzen mit Sequenzen aus dem Basissatz (also dem Trainingssatz) identisch waren. Dies ist vor allem dann kritisch zu bewerten, wenn deren Klassifizierung anhand des Äquivalents aus dem Basissatz erfolgte (Sequenzen, die nicht oder nur unzureichend nach ihren Liganden klassifiziert werden konnten). Solche Sequenzen müssten eigentlich aus dem Test ausgeschlossen werden, will man nur die Klassifikation wirklich fremder Sequenzen beurteilen.
Entfernt man aus dem Test alle Sequenzen, die ihren Liganden nach mehreren Klassen zugeteilt werden können bzw. die nach der Vorlage einer ähnlichen bzw. identischen Basissequenz klassifiziert wurden so bleiben nur noch die sechs Sequenzen 1AZE, 1BBZ-2, 1T0H, 1T0J, 1VYT und 1VYU. Von diesen konnten allerdings lediglich die Sequenzen 1AZE und 1BBZ-2 korrekt klassifiziert werden. Da aber die Sequenzen 1VYT und 1VYU derselben Domäne entstammen, kann die Fehlklassifikation dieser einfach gewertet werden, sodass sich eine Trefferquote von immerhin 40,00% errechnet. Inwieweit jedoch die in den jeweiligen PDB-Dateien dargestellten Liganden dieser Sequenzen auch den Klassen entsprechen, für die die betreffenden Sequenzen die tatsächlich höchste Affinität besitzen, kann an dieser Stelle nicht beurteilt werden.

4.2 Identifikation der zur Klassifikation signifikantesten Aminosäurepositionen

In diesem Abschnitt der Arbeit sollten nun die Features innerhalb der Fisher-Score-Vektoren ermittelt werden, welche für den in Kapitel 4.1 erstellten Klassifikator mit der höchsten prognostischen Präzision (arbiträrer Klassifikator aus Testreihe 3, Kapitel 4.1.2.2) zur Differenzierung von besonderer Bedeutung sind. Ziel dieser Untersuchungen war es zu analysieren, inwieweit die Aminosäurepositionen, die durch diese Features codiert werden, biologischen Schlüsselpositionen bei der Bindung der Liganden entsprechen könnten. Diese Fragestellung erscheint aus zweierlei Gesichtspunkten von Bedeutung. Zum einen dürfte ein Klassifikator, welcher solche Informationen nutzt, die auch tatsächlich physiologisch von Bedeutung sind, reliabler sein als einer, dessen Ergebnisse sich ausschließlich auf zufällige Differenzen der Fisher-Scores zwischen den einzelnen Klassen zurückführen lassen. Zum anderen – sollten informatisch und biologisch relevante Informationen bzw. Positionen tatsächlich miteinander korrelieren – könnte das vorgestellte Klassifikatormodell zukünftig dann auch zur Identifikation bislang noch unbekannter biologischer Schlüsselpositionen beitragen, was wiederum letztlich nicht nur zu einem umfassenderen Verständnis der klassenspezifischen Bindungsvorgänge führen dürfte, sondern möglicherweise auch neue Ansatzpunkte für medizinisch therapeutische Interventionsmöglichkeiten eröffnen könnte.
Um die zur Differenzierung besonders signifikanten Features herausfiltern zu können, musste der Klassifikator einer Feature Selection unterzogen werden. Diese erfolgte separat an jedem der Subklassifikatoren des Klassifikators anhand der Funktion *rfe.cv* des Softwarepakets RFE [54][161]. Bei Subklassifikatoren mit künstlichen Sequenzen wurde hierbei hinsichtlich sämtlicher jeweils zu

differenzierender Klassen stets der Satz an künstlichen Sequenzen verwendet, welcher auf der Basis aller Basissequenzen der jeweiligen Klasse emittiert wurde. Bezüglich der detaillierten Gründe, weshalb eine Implementation von RFE [54][161] hinsichtlich dieser Fragestellung auch in Modelle mit künstlichen Sequenzen möglich schien, sei auf Kapitel 5.3 der Diskussion verwiesen. Nach Feature Selection einer OvO-Multiclass-Differenzierung ist es jedoch schwer zu unterscheiden, welche der ermittelten Features für die Differenzierung welcher Klassen von Bedeutung sind. Daher sollten die Feature Selections beider im Klassifikator enthaltenen OvO-Multiclass-Subklassifikatoren *„1R vs. 1K vs. 1@“* bzw. *„2R vs. 2K vs. 2D“* – zusätzlich zur ohnehin durchgeführten Feature Selection des Subklassifikators *„2D vs. Nicht-2D“* – noch durch Feature Selections an den Subklassifikatoren *„1R vs. Nicht-1R“*, *„1K vs. Nicht-1K“*, *„1@ vs. Nicht-1@“*, *„2R vs. Nicht-2R“* und *„2K vs. Nicht-2K“* ergänzt werden. Bezüglich dieser Subklassifikatoren sollte dabei jeweils die hinsichtlich ihres Differenzierungsschritts präziseste Variante zum Einsatz kommen (Kapitel 4.1.2.2). Nur im Falle des Differenzierungsschritts *„1@ vs. Nicht-1@“* wurde sich für die formal zweitbeste Variante entschieden (radial mit Regularisierungsstufe **R100**), da diese eine bessere Sensitivität für Sequenzen der Klasse *1@* bei nur minimal schlechterer Gesamtperformance aufwies (Tabelle 4.3 bzw. entsprechende Confusion-Maps[*]).

Da bei Kreuzvalidierung der Feature Selection die Beurteilung der Features jeweils nur anhand der jeweiligen Trainingsmenge und die Validierung nur anhand der jeweiligen Testmenge erfolgt – also immer nur an einem Teil der Daten – mussten die Feature Selections entsprechend häufig wiederholt werden, um auch den Großteil der für das Gesamtmodell wichtigen Features erfassen und hervorheben zu können (Kapitel 5.3). Zudem war dies – aus ebenso in Kapitel 5.3 dargelegten Gründen – insbesondere hinsichtlich der Subklassifikatoren nötig, welche künstliche Sequenzen nutzten, um unwichtigere Features herausfiltern zu können. Zur Anwendung RFE [54][161] mussten zunächst wiederum die (Unter-)Funktionen dieses Softwarepakets nach der in Kapitel 3.2 beschriebenen Weise entsprechend angepasst werden. Bezüglich der Kreuzvalidierungen innerhalb der Feature Selections kam wiederum – es war ja nicht anders möglich (Kapitel 4.1.2.3) – die zehnfache Kreuzvalidierung zum Einsatz.

Da die anfallenden Datenmengen unter einer reinen Leave-One-Out Strategie zur Reduktion der Features (*speed = „low“*) die Kapazität des benutzten Rechnersystems auch bezüglich dieser Fragestellung überschritten, wurde sich erneut des in Kapitel 4.1.2.3 bereits beschriebenen Kompromisses bedient, der die logarithmische Reduktion der Features (*speed = „high“*) mit der Leave-One-Out Reduktionsvariante in einem System miteinander kombiniert. Hierbei erfolgt zunächst eine grobe Feature Selection nach dem Prinzip der logarithmischen Reduktion. Im nächsten Schritt werden entsprechend der kleinsten Menge an Features, mit der zuvor die durchschnittlich niedrigste

[*] „Erg-1@vsNicht1@-*“, im Anhang auf DVD unter: „~\Ergebnisse_Klassifikatoren\Kreuzvalidierung_Klassifikatoren“, wobei „*“ der jeweiligen Kernel-Funktion und Höhe der Regularisierungsstufe (also 0, 20 bzw. 100) entspricht (Kapitel 7.1.4 und 7.3).

Fehlklassifikation erreicht wurde, die hinsichtlich der einzelnen Trainingsmengen jeweils signifikantesten Features gewählt und ihre Matrices entsprechend modifiziert, das heißt alle anderen Features jeweils entfernt. Anhand dieser Matrices wird nun erneut eine Feature Selection mit *rfe.cv* durchgeführt, diesmal jedoch unter Leave-One-Out Reduktionsvariante. Dieser Schritt erfolgt allerdings nur, falls die betreffende Feature-Menge mehr als ein und weniger als 512 Features umfasst, andernfalls wird sich auf die logarithmische Variante beschränkt.
Da *rfe.cv* mithilfe einer zehnfachen Kreuzvalidierung durchgeführt wurde, worunter jede der hierbei gebildeten zehn Trainingsmengen einer separaten Feature Selection unterzogen wurde, erfolgten also bereits im Rahmen der logarithmischen Reduktionsvariante insgesamt zehn separate Feature Selections. Lag die hierbei ermittelte ideale Feature-Menge bei mehr als einer und weniger als 512 Features, wurde jede Trainingsmenge mit entsprechend modifizierter Matrix, das heißt nur mit den bezüglich der jeweiligen Trainingsmenge im ersten Schritt als am signifikantesten gewerteten Features, im Rahmen der Leave-One-Out Variante einer weiteren Feature Selection anhand zehnfacher Kreuzvalidierung unterzogen. So wurden die Sequenzen jeder der zehn Teilmengen (mit jeweils entsprechend modifizierter Matrix) erneut zufällig auf jeweils zehn Unter-Teilmengen verteilt, aus welchen wiederum jeweils zehn Trainingsmengen gebildet wurden, die dann jeweils einer Feature Selection mit Leave-One-Out Strategie unterzogen wurden. In diesem Fall erfolgten also pro Feature Selection an einer Teilmenge der logarithmischen Reduktionsvariante jeweils zehn weitere Feature Selections, sodass mit jeder Anwendung dieses Systems bis zu 100 Feature Selections durchgeführt wurden. Allerdings gilt es zu beachten, dass die Feature Selections im zweiten Schritt lediglich eine Verfeinerung der Resultate des ersten Schritts darstellten. Zudem erfolgte dieser Schritt nur, wenn die im ersten Schritt ermittelte ideale Feature-Menge bei mehr als einer und weniger als 512 Features lag. Dies bedenkend, wurde sich anhand heuristischer Methoden entschieden das System selbst 20-mal zu wiederholen, sodass insgesamt bis zu 2´000 Feature Selections pro Subklassifikator erfolgten.

Es zeigte sich allerdings, dass die minimalen Errorwerte des Gesamterrors vieler Subklassifikatoren während der logarithmischen Reduktionsvariante häufig bei Feature-Mengen von über 256 lagen. Daher wurde sich zweier zusätzlicher Kompromissvarianten bedient, mit denen im Falle solcher Subklassifikatoren auch dann eine weitere Reduktion der Features auf die wesentlichsten möglich wurde, wenn die minimalen Error-Werte nur unter mehr als 256 Features erzielt werden konnten. Hinsichtlich einiger Subklassifikatoren wurden die minimalen Errorwerte häufig nur unter Erhalt sämtlicher Features oder unter Erhalt von mindestens 1024 der 1160 Features erzielt. Eine weitere Reduktion der Feature-Menge führte in diesen Fällen zunächst zu einem starken Anstieg des Errors, welcher im weiteren Verlauf (unter weniger als 512 Features) jedoch wieder auf Werte ähnlich denen des minimalen Errors fiel. Aus diesem Grund wurde in diesen Fällen die zur Leave-One-Out Variante genutzte Feature-Menge nicht nach dem niedrigsten, sondern nach dem zweitniedrigsten Errorwert

bestimmt (Abb. 4.3). Diese Variante kam bei den Subklassifikatoren *„1R vs. 1K vs. 1@“*, *„2R vs. 2K vs. 2D“*, *„1R vs. Nicht-1R“*, *„1K vs. Nicht-1K“*, *„2R vs. Nicht-2R“* und *„2D vs. Nicht-2D“* zum Einsatz, wobei es nur im Falle des Subklassifikators *„1R vs. 1K vs. 1@“* nötig war auch die Errorwerte unter 1024 Features zu ignorieren, welche stets identisch mit denen unter 1160 Features waren. Die zweite Variante ergab sich aus der Tatsache, dass die Error-Kurve des Subklassifikators *„Klassen 1 vs. Klassen 2“* unter logarithmischer Reduktion von 1160 bis 64 Features stets eine Art Plateau niedriger Werte bildete und sich erst nach weiterer Reduktion der Features auf unter 64 ein starker Anstieg der Kurve dokumentierte (Abb. 4.4).

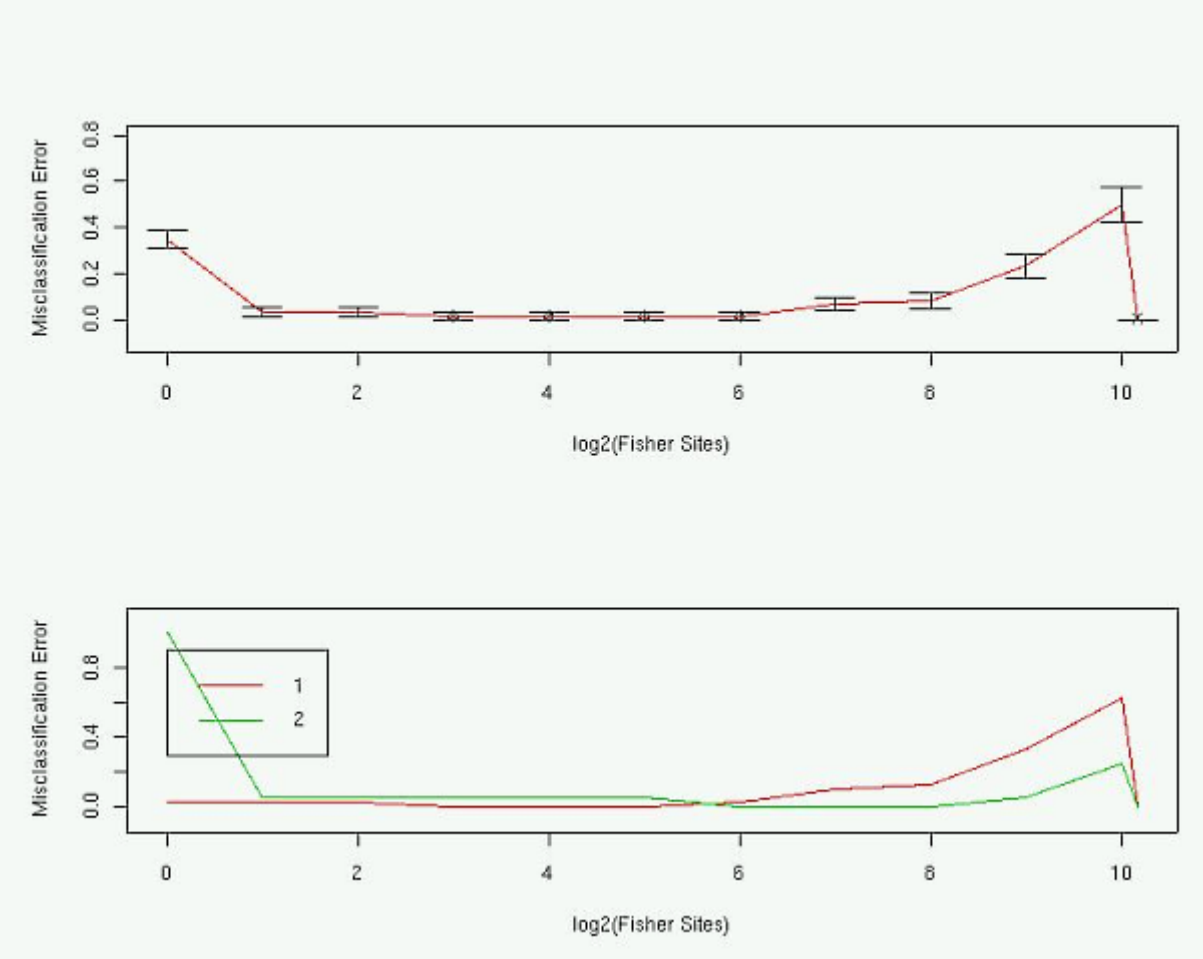


Abb. 4.3 Darstellung des Error-Verlaufs während einer Feature Selection unter logarithmischer Reduktionsvariante am Beispiel des Subklassifikators *„2D vs. Nicht-2D“*. Oben: Gesamt-Error; das Kreuz markiert den niedrigsten Errorwert (1160 Features), die Rauten die zweitniedrigsten Errorwerte (8, 16, 32 bzw. 64 Features). Liegt der minimale Errorwert bei 1160 (und/oder 1024) Features, ohne dass niedrigere Feature-Mengen mit äquivalentem Errorwert vorliegen, wurde sich nach dem zweitniedrigsten Errorwert zur Bestimmung der Feature-Menge für die Leave-One-Out Variante gerichtet. Unten: Individual-Error der zu differenzierenden Klassen (rot = *Nicht-2D*, grün = *2D*).

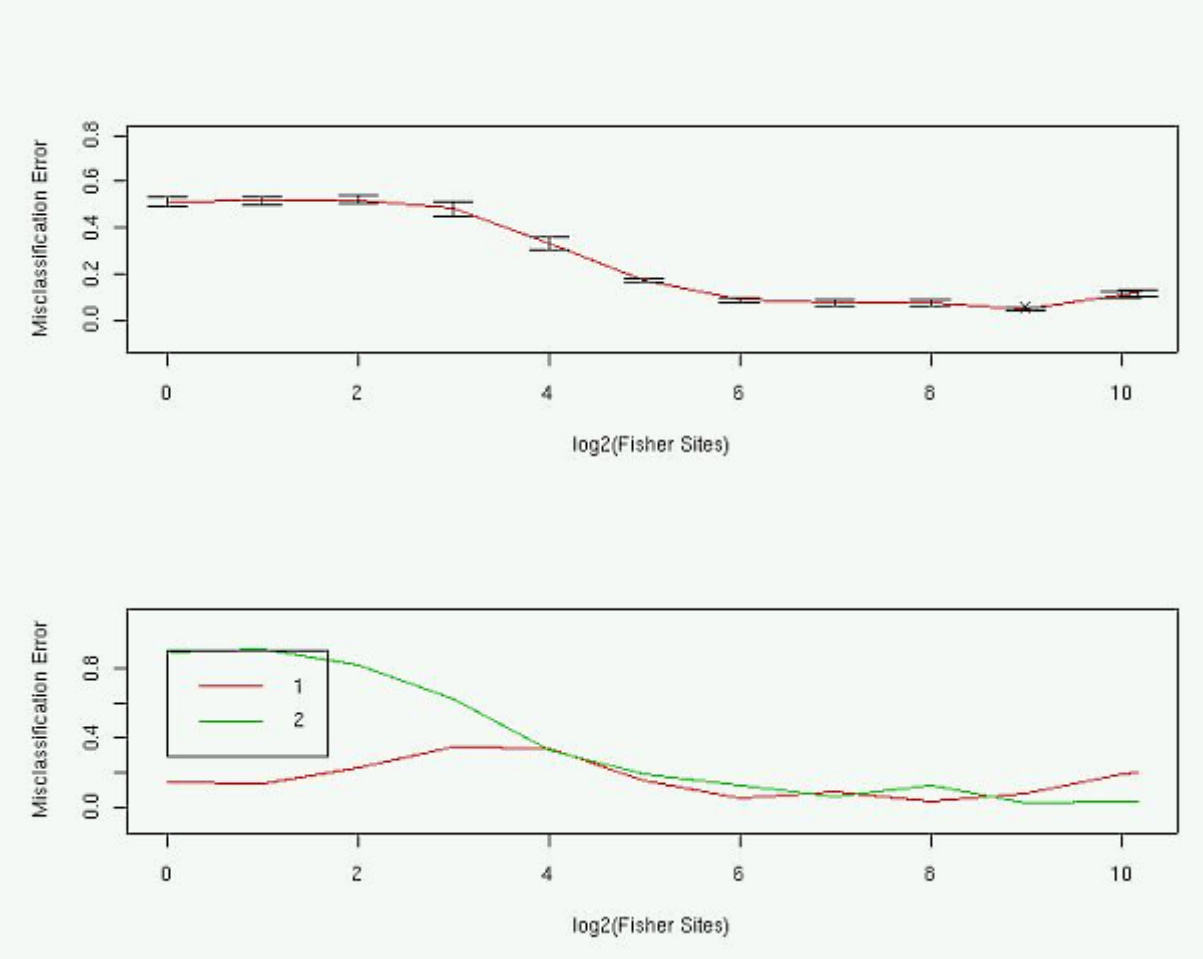


Abb. 4.4 Typischer Verlauf der Error-Kurven während einer Feature Selection unter logarithmischer Reduktionsvariante des Subklassifikators *„Klassen 1 vs. Klassen 2“*. Oben: Gesamt-Error; das Kreuz markiert den niedrigsten Errorwert (512 Features). Deutlich zu sehen ist die Plateaubildung der Error-Kurve aus niedrigen Werten zwischen 1160 und 64 Features, sowie der signifikante Anstieg der Kurve nach weiterer Reduktion der Features auf unter 64. Unten: Individual-Error der zu differenzierenden Klassen (rot = Klasse *1*, grün = Klasse *2*)

Daher wurde sich bezüglich dieses Subklassifikators entschieden, die zur Leave-One-Out Variante genutzte Feature-Menge auf 64 zu fixieren[*].

Die nach jeder Feature Selection isolierten Features sollten nun zur weiteren Analyse graphisch in Form eines Säulendiagramms mit den Ergebnissen der Arbeit von Friedrich, et al. (2006) [46] verglichen werden. Das in dieser Arbeit erstellte positionsbezogene Interaktionsprofil der dort untersuchten SH3-Domänen mit ihren Liganden wurde hierbei genutzt, um die Häufigkeit der sterischen Beteiligung jeder Aminosäureposition an der Bindung des Liganden zu bestimmen. Da jede Aminosäureposition

[*] Die einzelnen Error-Graphiken der Feature Selections unter logarithmischer Reduktionsvariante sämtlicher Subklassifikatoren können unter: „~\Ergebnisse_Bestimmung_Signifikanter_Positionen\…\Error-Plots“ im Anhang auf DVD eingesehen werden, wobei „…“ der Bezeichnung des jeweiligen Subklassifikators entspricht (*„1R vs. 1K vs. 1@“* = „Klassen1“ und *„2R vs. 2K vs. 2D“* = „Klassen2“); die Error-Graphiken sind hierbei jeweils mit Nummern von „1000001“ bis „1000020“ benannt (Kapitel 7.1.6 und 7.3).

innerhalb der Fisher-Score-Vektoren anhand von jeweils 20 Features codiert wird, konnten die den isolierten Features entsprechenden Aminosäurepositionen mithilfe der Gleichung

$$\text{Aminosäureposition} = \left\lfloor \frac{\text{Isoliertes Feature}}{20} + 1 \right\rfloor \tag{4.2.1}$$

berechnet werden.

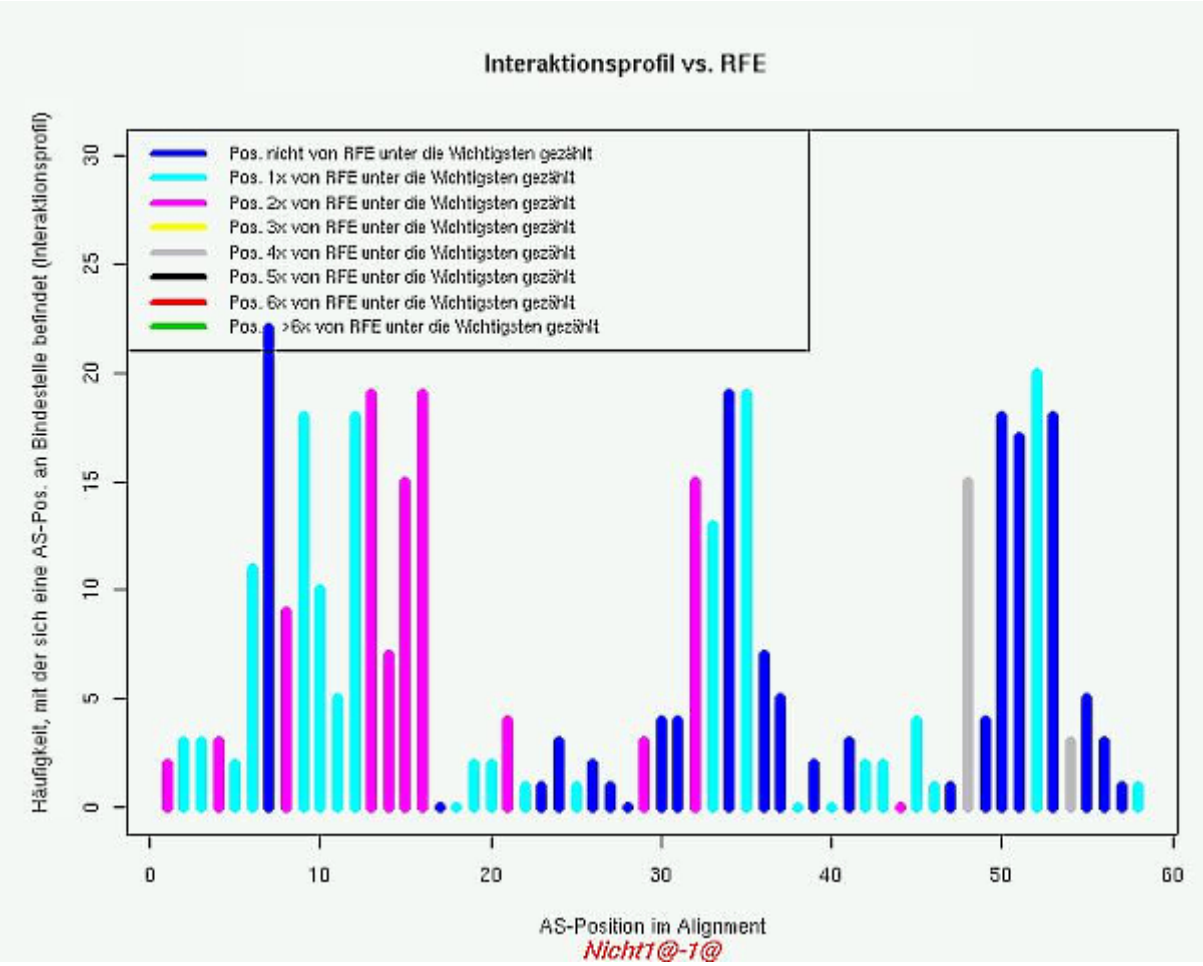


Abb. 4.5 Visualisierung der Ergebnisse einer einzelnen Feature Selection in Form eines Säulendiagramms am Beispiel des Subklassifikators *„1@ vs. Nicht-1@“*. Die Anzahl an Säulen (58) gibt die Menge der Aminosäurepositionen im Alignment der SH3-Domänen wieder. Die Höhe der Säulen beschreibt die Frequenz, mit der eine Position im Interaktionsprofil der Arbeit von Friedrich, et al. (2006) [46] als an der Bindung des Liganden (zumindest sterisch) beteiligt beschrieben wird. Die Farbe der einzelnen Säulen gibt die Häufigkeit an, mit welcher die jeweilige Position durch die während der Feature Selection isolierten Features codiert wurde (Legende oben links).

Durch Auftragen dieser Positionen in Relation zur Häufigkeit, mit der sie durch die isolierten Features codiert wurden, gegen die anhand des Interaktionsprofils errechneten Beteiligungsfrequenzen sämtlicher Positionen an der Bindung ließen sich die Ergebnisse jeder Feature Selection gut mit den Daten des Interaktionsprofils vergleichen (Abb. 4.5)[*].

Wie aus Abb. 4.5 schön zu entnehmen ist, scheinen innerhalb der meisten Sequenzen drei Abschnitte (≈ Positionen 6-16, Positionen 30-37 und Positionen 48-55) zu existieren, welche besonders häufig an der Interaktion mit dem Liganden beteiligt sind. Auffällig in diesem Beispiel ist die Häufigkeit, mit der Positionen innerhalb des ersten dieser Abschnitte anhand der Feature Selection isoliert wurden. Auch die große Anzahl an isolierten Features der Positionen 48 und 54, welche sich im dritten dieser Sequenzabschnitte befinden, könnte eine Korrelation zwischen zur Klassifikation signifikanten und biologisch hinsichtlich der Affinität zu einem Liganden relevanten Positionen indizieren.
Da die graphische Interpretation der Ergebnisse einer einzelnen Feature Selection jedoch wenig aussagekräftig ist und die manuelle Interpretation der Graphiken sämtlicher Feature Selections der einzelnen Subklassifikatoren äußerst aufwendig wäre, sollten die Ergebnisse aller Feature Selections eines Subklassifikators jeweils in einem weiteren Säulendiagramm zusammengefasst werden. Hierzu wurde zunächst für jede Position des Alignments die Häufigkeit bestimmt, mit der sie während der Feature Selections des jeweiligen Subklassifikators isoliert wurde. Dies konnte bezüglich jeder Position durch Addition der Frequenzen berechnet werden, mit denen ihre einzelnen Features dabei isoliert wurden. Hinsichtlich des jeweiligen Subklassifikators wurden nun die Positionen mit der maximalen und minimalen Häufigkeit bestimmt und die Differenz ihrer Isolationshäufigkeiten in fünf nahezu gleich große Bereiche (von Teilbereich A mit der größten bis Teilbereich E mit der geringsten Häufigkeit) unterteilt, wobei Teilbereich A aus rechnerischen Gründen stets etwas größer war. Diesen Bereichen konnten sämtliche Positionen schließlich anhand ihrer jeweiligen Isolationshäufigkeit hinsichtlich des betreffenden Subklassifikators zugeteilt werden. Auf diese Weise ließen sich sämtliche Positionen entsprechend ihrer jeweiligen Isolationshäufigkeit hinsichtlich des betreffenden Subklassifikators in Relation zueinander stellen und kategorisieren. Für Positionen, welche jeweils den Bereichen A und B – also den Teilbereichen mit den größten Isolationshäufigkeiten – zugeordnet wurden, wurden zudem mithilfe der Gleichung

$$\text{Aminosäure} = \text{Isoliertes Feature} - \left\lfloor \frac{\text{Isoliertes Feature}}{20} \right\rfloor \cdot 20 \qquad (4.2.2)$$

[*] Die einzelnen Säulendiagramme jeder Feature Selection eines Subklassifikators finden sich im Anhang auf DVD unter: „~\Ergebnisse_Bestimmung_Signifikanter_Positionen\…\Saeulendiagramme“, wobei „…“ der Bezeichnung des jeweiligen Subklassifikators entspricht (*„1R vs. 1K vs. 1@“* = „Klassen1“ und *„2R vs. 2K vs. 2D“* = „Klassen2“); die Säulendiagramme sind hierbei jeweils mit Nummern von „1“ bis maximal „2000“ benannt; in den Fällen, in denen sich auf die logarithmische Reduktion der Features beschränkt wurde, erfolgte die Nummerierung der Diagramme in 10-er Schritten (Kapitel 7.1.6 und 7.3).

die Aminosäuren bestimmt, auf welche sich die Werte ihrer isolierten Features bezogen. Da mit dieser Formel Tyrosin allerdings nicht – der Reihenfolge von HMMs entsprechend – durch die Zahl 20 sondern die Zahl 0 beschrieben wird, musste das Ergebnis in diesen Fällen entsprechend angepasst werden.
Die ihrer Isolationshäufigkeit hinsichtlich des betreffenden Subklassifikators nach kategorisierten Positionen konnten schließlich für jeden Subklassifikator separat wieder gegen die anhand des Interaktionsprofils errechneten Beteiligungsfrequenzen graphisch aufgetragen und analysiert werden (Abb. 4.6)*.

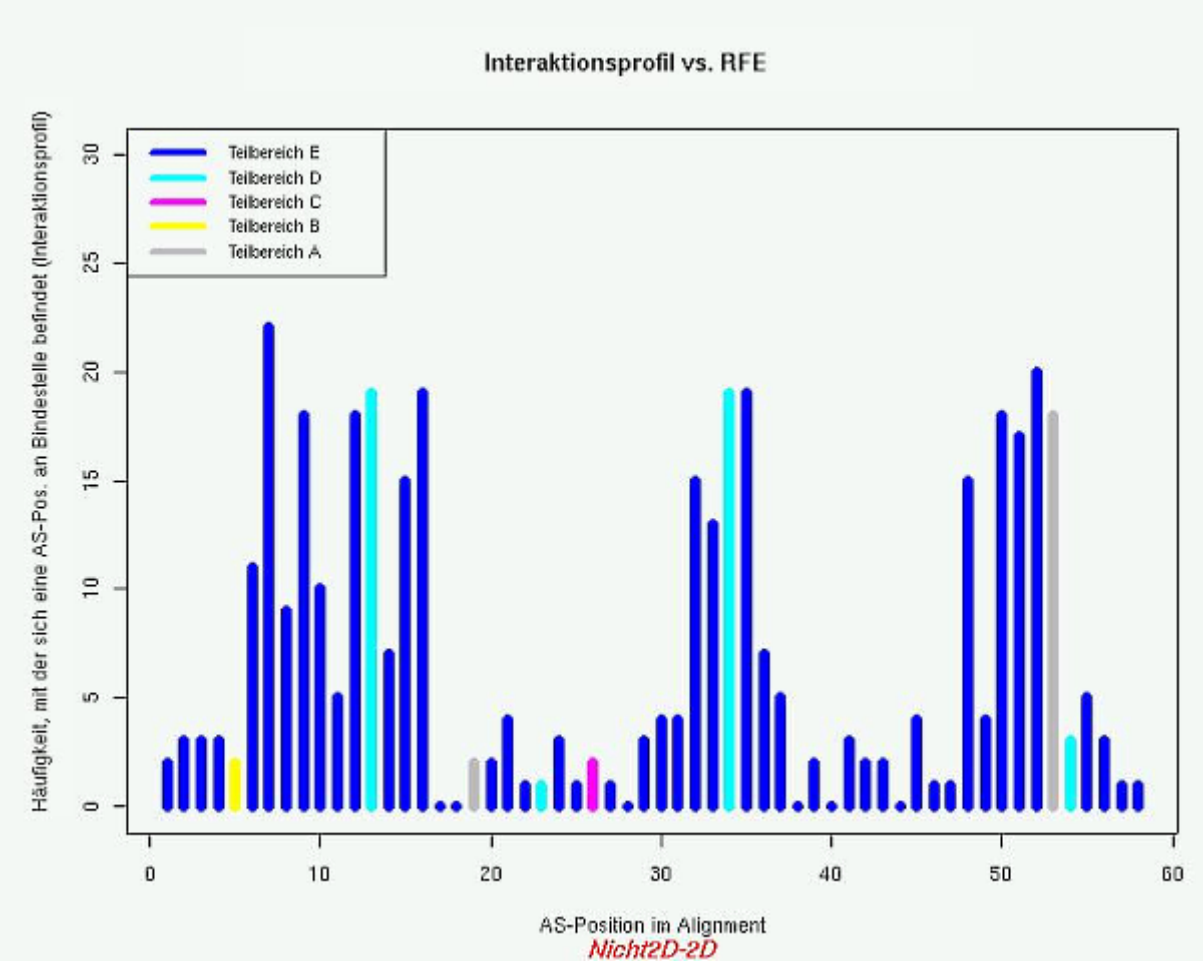


Abb. 4.6 Zusammenfassendes Säulendiagramm am Beispiel des Subklassifikators *„2D vs. Nicht-2D“* mit Gegenüberstellung der zusammengefassten Ergebnisse aller Feature Selections dieses Subklassifikators gegen die anhand des Interaktionsprofils der Arbeit von Friedrich, et al. (2006) [46] errechneten Beteiligungsfrequenzen sämtlicher Positionen an der Bindung. Die Anzahl an Säulen (58) gibt die Menge der Aminosäurepositionen im Alignment wieder. Die Höhe der Säulen beschreibt die Beteiligungsfrequenz einer Position im Interaktionsprofil. Die Farbe der Säulen spiegelt die Isolationsfrequenz einer Position wider (Legende oben links).

* Diese zusammenfassenden Säulendiagramme mit Darstellung der nach sämtlichen Feature Selections eines Subklassifikators ermittelten Positionen finden sich im Anhang auf DVD unter: „~\Ergebnisse_ Bestimmung_Signifikanter_Positionen\…\Saeulendiagramme“, wobei „…“ der Bezeichnung des jeweiligen Subklassifikators entspricht (*„1R vs. 1K vs. 1@“* = „Klassen1“ und *„2R vs. 2K vs. 2D“* = „Klassen2“), die Diagramme tragen jeweils den Namen „RFE_Summary.jpeg“ (Kapitel 7.1.6 und 7.3).

Um die Veränderung der Prädiktabilität der durch die einzelnen Feature Selections modifizierten Subklassifikatoren bezogen auf fremde Sequenzen in etwa abschätzen zu können, erfolgte ferner vor und nach jeder Feature Selection eine Testklassifikation anhand des in Kapitel 4.1.3 beschriebenen Sequenzsatzes aus der Arbeit von Friedrich, et al. (2006) [46]. Getestet wurden dabei stets alle Sequenzen dieses Satzes, welche den hinsichtlich des betreffenden Subklassifikators zu differenzierenden Klassen entsprachen. Die Testergebnisse nach den Features Selections wurden für jeden Subklassifikator anschließend in einer Gesamt-Confusion-Map zusammengefasst. Zudem wurde für jeden Subklassifikator die während der Feature Selections durchschnittlich als optimal angegebene Menge an Features berechnet. Durch Anwendung dieses Schemas auf sämtliche Subklassifikatoren konnten schließlich bezüglich jedes Subklassifikators die am häufigsten isolierten Positionen ermittelt werden, welche jeweils auch zu den signifikantesten zählen dürften[*†].

Die zusammenfassenden Säulendiagramme der Resultate aus den Feature Selections der einzelnen Subklassifikatoren zeigen, dass sich anhand dieses Schemas die Zahl der am häufigsten isolierten Positionen (Teilbereiche A und B) in der Regel auf eine sehr übersichtliche Menge von im Durchschnitt 6,55 Positionen reduzieren ließ. Im Falle des Subklassifikators *„1K vs. Nicht-1K“* umfasste sie sogar nur eine einzige Position (Position 21).
Die weitere Analyse der Positionen dieser Teilbereiche erbrachte, dass die Menge unterschiedlicher isolierter Features an den einzelnen Positionen sowohl im Vergleich der Positionen eines

---

[*] Die Programme hierzu finden sich im Anhang auf DVD unter: „~\Bestimmung_Signifikanter_Positionen“. Die Namen der einzelnen Programme setzen sich jeweils aus dem Terminus „RFE-“, der Bezeichnung des Subklassifikators (*„1R vs. 1K vs. 1@“* = „Klassen1“ und *„2R vs. 2K vs. 2D“* = „Klassen2“), der verwendeten Kernel-Funktion und der Höhe der Regularisierungsstufe (also 0, 20 bzw. 100) zusammen (Kapitel 7.1.5 und 7.2).

Die Dateien mit den Ergebnissen hieraus finden sich im Anhang auf DVD unter: „~\Ergebnisse_Bestimmung_Signifikanter_Positionen\…“, wobei „…“der Bezeichnung des jeweiligen Subklassifikators entspricht (*„1R vs. 1K vs. 1@“* = „Klassen1“ und *„2R vs. 2K vs. 2D“* = „Klassen2“). Sie tragen jeweils die Bezeichnung „Erg-RFE-“, gefolgt von der Bezeichnung des Subklassifikators (*„1R vs. 1K vs. 1@“* = „Klassen1“ und *„2R vs. 2K vs. 2D“* = „Klassen2“), der verwendeten Kernel-Funktion und der Höhe seiner Regularisierungsstufe (also 0, 20 bzw. 100) (Kapitel 7.1.6 und 7.3).

[†] Die Testklassifikationen vor Feature Selection erfolgten jeweils anhand separater Programme. Sie finden sich ebenso im Anhang auf DVD unter: „~\Bestimmung_Signifikanter_Positionen“. Die Titel der einzelnen Programme bestehen jeweils aus dem Terminus „Testklassifikation-vor-RFE-“, gefolgt von der Bezeichnung des Subklassifikators (*„1R vs. 1K vs. 1@“* = „Klassen1“ und *„2R vs. 2K vs. 2D“* = „Klassen2“), der verwendeten Kernel-Funktion und der Höhe der Regularisierungsstufe (also 0, 20 bzw. 100) (Kapitel 7.1.5 und 7.2).

Die Dateien mit den Ergebnissen hieraus finden sich im Anhang auf DVD unter: „~\Ergebnisse_Bestimmung_Signifikanter_Positionen\…“, wobei „…“ wieder der Bezeichnung des jeweiligen Subklassifikators entspricht (*„1R vs. 1K vs. 1@“* = „Klassen1“ und *„2R vs. 2K vs. 2D“* = „Klassen2“). Sie tragen jeweils die Bezeichnung „Erg-Testklassifikation-vor-RFE-“, gefolgt von der Bezeichnung des Subklassifikators (*„1R vs. 1K vs. 1@“* = „Klassen1“ und *„2R vs. 2K vs. 2D“* = „Klassen2“), der verwendeten Kernel-Funktion und der Höhe seiner Regularisierungsstufe (also 0, 20 bzw. 100) (Kapitel 7.1.6 und 7.3).

Subklassifikators untereinander wie auch im Vergleich mit denen aller anderen Subklassifikatoren äußerst variabel war. So zählte bezüglich des Subklassifikators *„1R vs. Nicht-1R“* Position 49 nur aufgrund drei verschiedener Features zu den häufigsten, während bei Position 29 insgesamt zehn verschiedene Features dazu beitrugen. Ähnliches findet sich bei Betrachtung des Subklassifikators „*Y vs. Nicht-Y“*: hier wurde die Häufigkeit von Position 41 ebenso durch lediglich drei besonders häufig isolierte Features definiert, während es sich bei Position 11 um acht Features handelte.

Der Vergleich des Interaktionsprofils der Arbeit von Friedrich, et al. (2006) [46] mit sämtlichen Positionen aller Teilbereiche A und B offenbarte, dass die Mehrzahl dieser Positionen (23 von insgesamt 42 unterschiedlichen Positionen = 54,76%) in den drei Sequenzabschnitten (Positionen 6-16, 30-37 und 48-55) lagen, die auch zumindest sterisch hinsichtlich der meisten für das Interaktionsprofil herangezogenen Domänen an der Bindung der Liganden beteiligt sind. Dies erscheint umso eindrucksvoller, bedenkt man, dass diese Sequenzabschnitte zusammen nur 46,55% aller Aminosäurepositionen ausmachen. Bezogen auf die einzelnen Subklassifikatoren für sich traf dies allerdings nur in fünf Fällen zu. So befand sich beispielsweise hinsichtlich des Subklassifikators *„1K vs. Nicht-1K“* gar keine und hinsichtlich des Subklassifikators *„2R vs. Nicht-2R“* nur eine der dort jeweils am häufigsten isolierten Positionen (Teilbereiche A und B) in diesen Abschnitten, wohingegen es beispielsweise bezüglich des Subklassifikators *„Klassen 1 vs. Klassen 2“* 85,71% und bezüglich des Subklassifikators *„1@ vs. Nicht-1@“* sogar alle waren. Eine Korrelation zwischen der Klassifikationsgüte eines Subklassifikators vor Feature Selection und der Anzahl von Positionen seiner Teilbereiche A und B innerhalb solcher Sequenzabschnitte konnte nicht festgestellt werden.

Die Auswertung der Ergebnisse hinsichtlich der durchschnittlich als optimal angesehenen Feature-Mengen legt nahe, dass diese mit zunehmender Regularisierung der Subklassifikatoren anstiegen. So lag sie bezüglich des einzigen anhand der Variante **R0** regularisierten Subklassifikators *„1R vs. 1K vs. 1@“* noch durchschnittlich bei 13,80 Features, bezüglich der mit Stufe **R20** regularisierten Subklassifikatoren schon bei durchschnittlich 31,02 Features und bezüglich der mit Stufe **R100** regularisierten Subklassifikatoren sogar bei durchschnittlich 102,14 Features. Jedoch lässt sich diese Beobachtung nicht pauschalisieren, da sie beispielsweise bezüglich des mit Variante **R20** regularisierten Subklassifikators *„2D vs. Nicht-2D“* durchschnittlich nur 4,75 Features betrug. Auch die ähnlich hohen durchschnittlichen Feature-Mengen des mit Stufe **R20** regularisierten Subklassifikators *„2R vs. Nicht-2R“* (∅ 53,69 Features) und des anhand der Variante **R100** regularisierten Subklassifikators *„1@ vs. Nicht-1@“* (∅ 53,63 Features) zeigen das dies nicht in jedem Fall zuzutreffen scheint. Zudem gilt es bei Interpretationen dieser Ergebnisse zu beachten, dass zum einen die seitens RFE [54][161] als optimal gewertete Feature-Menge bezüglich mehrerer Subklassifikatoren teils manipuliert werden musste, um die wichtigsten Features überhaupt isolieren zu können, und andererseits unter Anwendung von *rfe.cv* in Kombination mit künstlichen Sequenzen ohnehin größere Feature-Mengen zu erwarten sind. Allerdings wäre die Beobachtung möglicherweise dadurch erklärbar, dass mit größerer Anzahl an

Sequenzen auch die Menge sich klassenspezifisch ähnelnder Features größer sein dürfte, sodass sich während der Kreuzvalidierungen konsekutiv auch bessere Ergebnisse unter Erhalt dieser erzielen lassen sollten.

Unter Voraussetzung, dass diese Hypothese zutreffend ist, könnte man bei Betrachtung der Ergebnisse zudem eine Korrelation zwischen der Präzision eines Subklassifikators vor Feature Selection und der für ihn anhand der Feature Selections ermittelten optimalen Feature-Menge vermuten. So wiesen beispielsweise die beiden mit Variante **R100** regularisierten Subklassifikatoren ähnlicher Präzision *„1@ vs. Nicht-1@“* und *„Klassen 1 vs. Klassen 2“* auch eine vergleichbare durchschnittliche Menge an Features auf (∅ 53,63 Features bzw. ∅ 61,37 Features), während sie hinsichtlich des anhand derselben Variante regularisierte Subklassifikators mit deutlich geringerer Präzision *„X_ORS vs. Nicht-X_ORS“* durchschnittlich deutlich höher lag (∅ 191,44 Features). Betrachtet man die mit Stufe **R20** regularisierten Subklassifikatoren, findet sich im Vergleich von *„2D vs. Nicht-2D“*, dem präzisesten dieser Subklassifikatoren, mit den restlichen dieser Regularisierungsstufe ein ähnliches Bild, da dieser durchschnittlich auch die wenigsten Features (∅ 4,75) aufwies. Allerdings lässt sich diese Vermutung bei Vergleich der weiteren Subklassifikatoren dieser Regularisierungsstufe untereinander (ohne den Subklassifikator *„2D vs. Nicht-2D“*) nicht bestätigen, da diese bei ähnlichen Präzisionen durchschnittlich doch recht unterschiedliche Mengen an Features boten (zwischen ∅ 15,58 und 53,69 Features). Zudem ist auch diese Vermutung nur unter den beiden genannten Vorbehalten zu betrachten. Dennoch erscheint sie sinnvoll, da eine Differenzierung von Klassen mit per se unspezifischeren Merkmalen bzw. Features unter Zuhilfenahme einer entsprechend größeren Menge an Merkmalen präziser werden sollte.

Die Resultate der Testklassifikationen dokumentieren nahezu bezüglich aller Subklassifikatoren eine geringfügige Verschlechterung der Klassifikationspräzision nach Feature Selection (Tabelle 4.10). Dies ist jedoch durchaus nicht verwunderlich, da zum einen die Kreuzvalidierungen der Feature Selections in den meisten Fällen ja unter Anwendung künstlicher Sequenzen erfolgten (Kapitel 5.2.2), zum anderen die Klassifikationen nicht mit den anhand aller Feature Selections des jeweiligen Subklassifikators durchschnittlich als am signifikantesten gewerteten Features erfolgten, sondern stets nur mit dem Satz an Features, welcher hinsichtlich einer einzigen Trainingsmenge als optimal gewertet wurde. Zudem war es bezüglich mehrerer Subklassifikatoren zur Analyse der wichtigsten Features – wie bereits weiter oben ausgeführt – häufig nötig die seitens RFE [54][161] als optimal gewerteten Feature-Mengen zu manipulieren, sodass auch die Testklassifikationen in diesen Fällen nicht anhand der tatsächlich optimalen Feature-Mengen erfolgen konnten. Ferner sind die hierbei gewonnenen Daten über Veränderungen der Präzision der einzelnen Subklassifikatoren auch nur unter Vorbehalt verwertbar, da die aus Kapitel 4.1.3 übernommene Klassenzuordnung der Sequenzen des für die Testklassifikationen genutzten Testsatzes in vielen Fällen eher fraglich war. Dies dürfte insbesondere auch erklären, weshalb die Testergebnisse nach Feature Selection im Falle der Subklassifikatoren *„2R vs. 2K vs. 2D“* und vor

allem *„2R vs. Nicht-2R“* vergleichsweise noch etwas schlechter waren, denn hinsichtlich der Testsequenzen von Klasse *2R* war die Klassenzuordnung von Kapitel 4.1.3 besonders fraglich. Die Ergebnisse der Testklassifikationen sollten hier jedoch auch nicht als Maß zur exakten Beurteilung von Veränderungen der Klassifikationspräzision fungieren, sondern lediglich als Anhaltspunkt für die Reliabilität der isolierten Features dienen, welche angesichts der allgemein doch eher geringen Verschlechterung der Klassifikationspräzision nach den Feature Selections durchaus hoch sein dürfte.

Subklassifikationsschritt	Variante	RFE	Gesamttreffer-quote	durchschnittliches F-Maß
„1R vs. Nicht-1R“	linear **R20**	vor:	100%	100%
		nach:	99.61%	99.61%
„1K vs. Nicht-1K“	linear **R20**	vor:	100%	--
		nach:	99.78%	--
„1@ vs. Nicht-1@“	radial **R100**	vor:	100%	100%
		nach:	100%	100%
„2R vs. Nicht-2R“	linear **R20**	vor:	71.43%	68.89%
		nach:	48.99%	48.80%
„2K vs. Nicht-2K“	radial **R20**	vor:	85.71%	46.15%
		nach:	80.78%	71.85%
„2D vs. Nicht-2D“	linear **R20**	vor:	100%	100%
		nach:	99.79%	99.23%
„Klassen 1“ („1R vs. 1K vs. 1@“)	linear **R0**	vor:	100%	100%
		nach:	94.16%	94.37%
„Klassen 2“ („2R vs. 2K vs. 2D“)	linear **R20**	vor:	85.71%	85.56%
		nach:	68.42%	73.43%
„Klassen 1 vs. Klassen 2“	radial **R100**	vor:	81.82%	81.67%
		nach:	79.15%	79.03%
„X_ORS vs. Nicht-X_ORS“	linear **R100**	vor:	75.86%	43.14%
		nach:	75.86%	43.14%
„Y vs. Nicht-Y“	linear **R20**	vor:	100%	--
		nach:	98.11%	--

Tabelle 4.10 Resultate der Testklassifikationen vor und nach Feature Selection mit RFE [54][161] in Form der Gesamttrefferquote und des durchschnittlichen F-Maßes jedes Subklassifikators vor und nach den Feature Selections.

4.3 Analyse der zur Klassifikation signifikantesten Aminosäurepositionen im Hinblick auf ihre Rolle bei der Bindungsselektivität

Um die am häufigsten isolierten Positionen aus Kapitel 4.2 noch präziser im Hinblick auf etwaige Übereinstimmungen mit biologisch zur Bindung der Liganden funktionellen Positionen untersuchen zu können, sollte nun die tertiär- bzw. quartärstrukturelle Darstellung dieser Positionen innerhalb ausgewählter Beispieldomänen zur Veranschaulichung ihrer räumlichen Lage bzw. ihrer Beziehung zu den Liganden erfolgen. Als Grundlage hierfür fungierte die Softwareplattform PyMOL, welche unter anderem durch die Anwendungsmöglichkeit von in verschiedenen Datenbanken zur Verfügung gestellten molekularen Strukturinformationen dreidimensionale Darstellungen von Proteinen erlaubt [28][139]. Anschließend wurden für die Trainingssätze jedes Subklassifikators klassenspezifisch komparative Sequenzlogos mithilfe von WebLogo [26] erstellt, welche folgend zur Ermittlung positionsspezifischer Aminosäuredifferenzen – speziell in Bezug auf die am häufigsten isolierten Positionen – zwischen den einzelnen Klassen herangezogen werden konnten.

4.3.1 Identifikation möglicher biologischer Schlüsselpositionen anhand von PyMOL [28][139]

Im vorliegenden Fall definieren sich biologische Schlüsselpositionen durch ihren Einfluss auf die Ligandenpräferenz der Domänen. Es sind also die Positionen, die mit ihren klassenspezifischen Aminosäurevariationen in ihrer Gesamtheit die Bindungsspezifität einer Klasse biologisch bedingen. Eine sichere Identifikation dieser, um so etwa die Klassenzugehörigkeit einer Domäne bereits anhand ihrer Aminosäuresequenz festmachen zu können, ist jedoch üblicherweise äußerst aufwendig. Daher sollte nun geprüft werden, inwieweit die anhand der Feature Selections am häufigsten isolierten Positionen möglicherweise mit potentiellen biologischen Schlüsselpositionen korrelieren, um solche Analysen zukünftig gegebenenfalls erleichtern zu können. Hierzu sollte unter Zuhilfenahme dreidimensionaler graphischer Darstellungen zunächst bestimmt werden, welche der am häufigsten isolierten Positionen Kontakt zu Aminosäureresten innerhalb des Konsensmotivs der Liganden besitzen, da sich hinsichtlich solcher Positionen auch am ehesten ein Einfluss auf die Bindungsspezifität der Klassen vermuten lassen dürfte. Im nächsten Schritt sollte dann untersucht werden, ob sich in den graphischen Darstellungen klassenspezifische Unterschiede hinsichtlich dieser Positionen abzeichnen, die einen Einfluss dieser wahrscheinlich machen.

Dabei ermöglichte der arbiträre Aufbau des für die Feature Selections herangezogenen Klassifikators noch eine Präzisierung der Untersuchungen insofern, als dass die hierbei isolierten Positionen jeweils spezifisch für die einzelnen Differenzierungsschritte in der Taxonomie des Klassensystems waren. Somit konnte auch eine Korrelation dieser mit für jeden dieser Differenzierungsschritte spezifischen, potentiellen Schlüsselpositionen untersucht werden. Zudem wurden durch seine Zusammensetzung aus vielfach binären Subklassifikatoren auch die Analysen erleichtert, denn in diesen Fällen musste jeweils

nur bezüglich der Differenzierung zweier Klassen beurteilt werden, inwiefern die hinsichtlich des betreffenden Subklassifikators für signifikant befundenen Positionen möglicherweise biologischen Schlüsselpositionen entsprechen könnten. Da sich diese Untersuchungen bei Differenzierung von mehr als zwei Klassen deutlich schwieriger gestalten, wurden bezüglich der OvO-Multiclass-Differenzierungen *„1R vs. 1K vs. 1@“* bzw. *„2R vs. 2K vs. 2D“* zudem noch die ihnen im arbiträren Aufbau des Gesamtklassifikators folgenden binären Subdifferenzierungen ergänzend analysiert. Anhand dieses Schemas konnten also etwaige Korrelationen mit potentiellen Schlüsselpositionen bezüglich jedes Differenzierungsschritts separat untersucht werden, angefangen bei Korrelationen mit möglichen Schlüsselpositionen, die eine Bindung überhaupt erst ermöglichen bzw. verhindern bis hin zu solchen, die die Zugehörigkeit einer Sequenz zu einer bestimmten Subklasse definieren.

Die strukturellen Darstellungen sollten hierfür jeweils anhand mindestens einer Beispieldomäne für jede der in den einzelnen Differenzierungsschritten voneinander zu unterscheidenden Klassen erfolgen. Da die einzelnen Klassen zum Teil jedoch äußerst heterogen aufgebaut waren – speziell diejenigen, welche sich aus mehreren Subklassen zusammensetzten (z.B. Klasse *Nicht-Y*), sollte bezüglich solcher Klassen jeweils eine Beispieldomäne für jede ihrer Subklassen gewählt werden. Die Suche nach geeigneten Beispieldomänen erfolgte nach zwei Kriterien, zum einen musste ihre Klassenzuordnung bekannt sein und zum anderen eine PDB-Datei von dieser mit strukturellen Informationen über die Domäne selbst sowie – sollte es sich um eine bindende Domäne handeln – über ihren Liganden gebunden an die jeweilige Domäne existieren. Eine detaillierte Auflistung der entsprechenden Suchwege findet sich in Kapitel 3.2. Mit Ausnahme der Klasse *X_ORS*, für welche keine den Suchkriterien entsprechende Domäne ausgemacht werden konnte, fand sich auf diese Weise für jede der acht Klassen eine geeignete PDB-Datei. Bezüglich der Klasse *Y* wurden zwei Beispieldomänen gewählt, denn aufgrund fehlender Bindung dieser an einen Liganden war ihre graphische Interpretation entsprechend erschwert. An dieser Stelle muss erwähnt werden, dass der in PyMOL [28][139] anhand der PDB-Datei „2RPN“ der Beispieldomäne „ABP-1“ (Klasse *2K*) dargestellte Ligand scheinbar nicht Klasse *2K* entspricht. Dies liegt jedoch daran, dass seine Sequenz nicht wie üblich von N- nach C-terminal, sondern umgekehrt aufgeführt ist.

Da die Aminosäuresequenzen der Beispieldomänen in den PDB-Dateien nicht aligniert waren, die Nummerierung der Positionen also nicht mit der in dieser Arbeit verwendeten (dem SH3-Familien-HMM folgend) übereinstimmte, mussten die Nummerierungen zunächst an die hier verwendete angepasst werden. Hierzu wurden die unalignierten Sequenzen der Beispieldomänen jeweils aus den Abschnitten „*SEQRES*“ der entsprechenden PDB-Datei gewonnen und anhand des „Three-/ one-letter Amino Acid Codes“ Programm der Internetplattform Zbio.net (http://zbio.net/eng/scripts/01_17.html) in eine Einbuchstabencode-Darstellung umgewandelt. Die PDB-Dateien der Klasse *Y* wiesen keinen Abschnitt „*SEQRES*“ auf, weshalb ihre Sequenzen manuell von der Sequenzdarstellung aus PyMOL [28][139] übernommen werden mussten. So ließen sich die in der jeweiligen PDB-Datei

verwendeten Sequenzen mit ihren für die Feature Selections verwendeten alignierten Formen vergleichen und entsprechende Umrechnungsalgorithmen erarbeiten. Entsprachen für signifikant befundene Positionen einem Gap in der alignierten Sequenz einer Beispieldomäne, konnte natürlich keine Umrechnung erfolgen und konsekutiv auch keine Markierung derselben in der betreffenden Beispieldomäne*. Um die Resultate aus den Feature Selections des Differenzierungsschritts *„Y vs. Nicht-Y“* besser zu veranschaulichen, sollten in den strukturellen Darstellungen bezüglich dieses Schritts auch die Positionen markiert werden, welche sich nach dem Interaktionsprofil der Arbeit von Friedrich, et al. (2006) [46] am häufigsten in sterischer Beziehung zu den Liganden befinden. Da auch die Nummerierung der Positionen dieses Interaktionsprofils dem SH3-Familien-HMM angepasst wurde, wurden für diesen Differenzierungsschritt entsprechend erweiterte Umrechnungsalgorithmen verwendet†.

Die Befehle zur Integration der Ergebnisse in die strukturellen Darstellungen der Beispieldomänen wurden PyMOL-kompatibel in Form separater Dateien für jede Beispieldomäne des jeweiligen Differenzierungsschritts gespeichert‡. In den Darstellungen wurden die Beispieldomänen (jeweils benannt nach ihrer, innerhalb der jeweiligen PDB-Datei mit einem Buchstaben betitelten Kette) grau, in Form von Sphären und ihre jeweiligen Liganden (ebenso benannt mit dem Buchstaben ihrer Kette) orange, in Form eines Ribbon dargestellt. Um die Rolle der klassenspezifischen Konsensmotive (in den Darstellungen jeweils als „Konsenssequenz“ bezeichnet) der Liganden besser zu veranschaulichen, wurden von diesen zudem noch die Seitenketten in Form von Sticks dargestellt. In den Darstellungen eines Differenzierungsschritts wurden von den bezüglich dieses Differenzierungsschritts isolierten Positionen jeweils nur die Positionen der Teilbereiche A und B markiert (Kapitel 4.2); hinsichtlich der Gründe hierfür sei auf Kapitel 5.3 der Diskussion verwiesen. Diese Positionen wurden schokoladenfarben hervorgehoben und mit „RFE“ betitelt. Einzige Ausnahme hiervon bildete der Differenzierungsschritt *„1K vs. Nicht-1K“*. Da bei diesem die Teilbereiche A und B zusammen nur eine

---

* Eine Gegenüberstellung der Sequenzen aus den PDB-Dateien den Beispieldomänen jeweils mit ihrer alignierten Form findet sich im Anhang auf DVD unter: „~\Pymol“ mit der Bezeichnung „Positionsnummerierung.txt“. Die in R kompatibler Form geschriebenen Umrechnungsalgorithmen für die einzelnen Beispieldomänen finden sich im Anhang auf DVD beispielsweise unter: „~\Pymol\X_ORS_nicht-X_ORS“ (mit Ausnahme derer für die Beispieldomänen der Klasse *Y*). Sie sind jeweils mit der Klasse der Beispieldomäne, ihrem PDB-Identifikator und der Bezeichnung „-R-Code“ benannt (Kapitel 7.1.7, 7.2 und 7.3).

† Die speziell für den Differenzierungsschritt *„Y vs. Nicht-Y“* angepassten Umrechnungsalgorithmen befinden sich im Anhang auf DVD unter: „~\Pymol\YvsNichtY“ (hier finden sich auch die Umrechnungsalgorithmen für die Beispieldomänen der Klasse *Y*). Sie sind ebenso jeweils mit der Klasse der Beispieldomäne, ihrem PDB-Identifikator und der Bezeichnung „-R-Code“ benannt, tragen aber zusätzlich noch die Bezeichnung „-Y“ am Ende (Kapitel 7.1.7, 7.2 und 7.3).

‡ Die Dateien mit den PyMOL-Befehlen finden sich im Anhang auf DVD unter: „~\Pymol\…“, wobei „…“ der Bezeichnung des jeweiligen Subklassifikationsschritts entspricht (*„1R vs. 1K vs. 1@“* = „Klassen1“ und *„2R vs. 2K vs. 2D“* = „Klassen2“). Die Namen der einzelnen Dateien setzen sich jeweils aus der Klasse der Beispieldomäne, ihrem PDB-Identifikator und der Bezeichnung „-PymolCode“ bzw. „-PymolCode-Y“ zusammen (Kapitel 7.1.7 und 7.3).

einzige Position umfassten, wurden hier noch die Positionen des Teilbereichs C markiert*. Wie bereits erwähnt sollten in den strukturellen Darstellungen bezüglich des Differenzierungsschritts „*Y vs. Nicht-Y*“ auch die Positionen markiert werden, welche sich nach dem Interaktionsprofil der Arbeit von Friedrich, et al. (2006) [46] am häufigsten in sterischer Beziehung zu den Liganden befanden. „Häufig“ wurde dabei heuristisch damit definiert, wenn dies bezüglich einer Position an mindestens zehn der Domänen des Interaktionsprofils beobachtet werden konnte, da dies mehr als einem Drittel aller Domänen des Interaktionsprofils entsprach und so zudem eine sehr übersichtliche Darstellung der üblichen Bindungsstelle gelang.

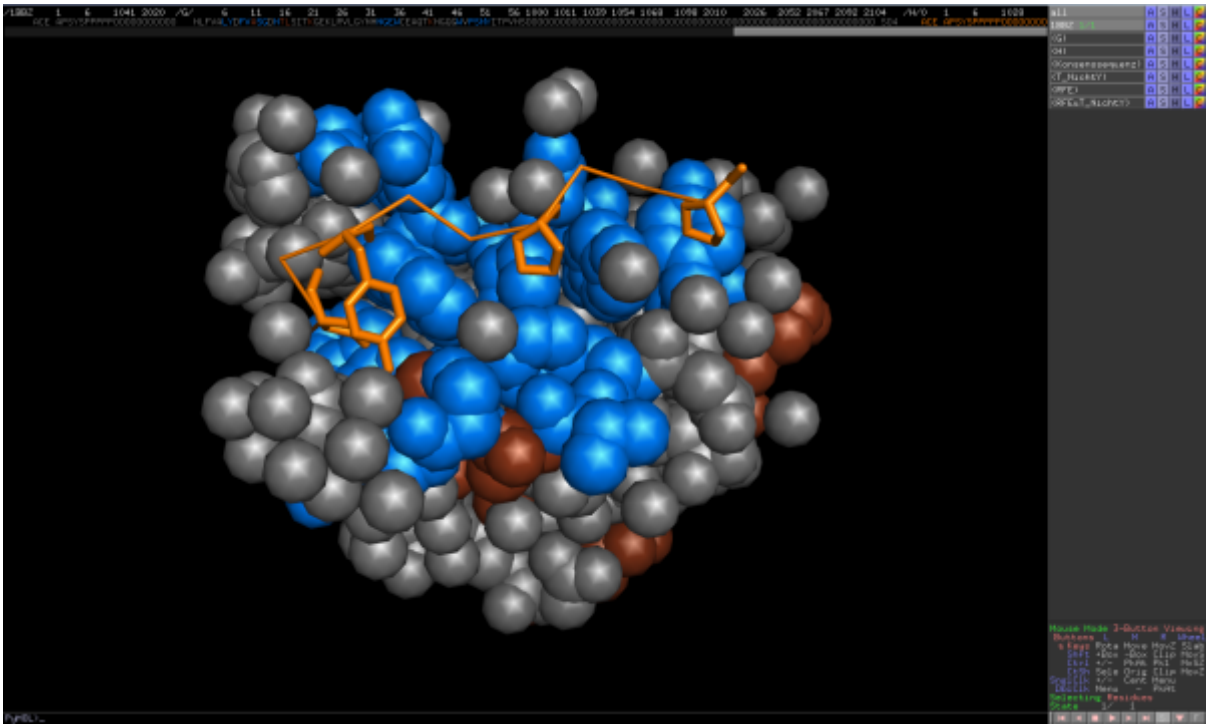


Abb. 4.7 Die Benutzeroberfläche von PyMOL [28][139] mit struktureller Darstellung der Beispieldomäne für Klasse *1@* (Abl1_H), in welcher exemplarisch die Ergebnisse des Differenzierungsschritts *„Y vs. Nicht-Y“* markiert sind. Die Domäne selbst ist grau und in Sphärenform abgebildet, während der Ligand in Form eines orangefarbenen Ribbon gezeigt wird. Die klassenspezifische Konsenssequenz ist durch die Darstellung seiner Seitenketten hervorgehoben. Die schokoladenfarbenen Bereiche entsprechen den Positionen, die in den Feature Selections dieses Differenzierungsschritts am häufigsten isoliert wurden, während die marinefarbenen Bereiche die Positionen darstellen, welche sich nach dem Interaktionsprofil der Arbeit von Friedrich, et al. (2006) 143[46] am häufigsten in sterischer Beziehung zu den Liganden befanden. Gut zu erkennen sind die durch die Konsenssequenz besetzen Bindungstaschen der Domäne und die sich in direktem Kontakt zur Konsenssequenz (dargestellter Tyrosinrest des Liganden) befindliche Position 16 aus der „RFE“-Gruppe. Die marinefarben markierten Bereiche demonstrieren schön, wie sehr die Lokalisation der Bindungsstelle dieser Beispieldomäne mit der sonst für diese Domänen typischen korreliert. Oben: Darstellung der Aminosäuresequenz von Domäne und Ligand mit der der strukturellen Darstellung entsprechenden Farbgebung. Oben rechts: namentliche Auflistung der in einzelne Gruppen zusammengefassten Objekte/Positionen.

* Mit dem Ausdruck „RFE“-Gruppe ist im Folgenden stets Bezug auf die Positionen genommen, welche in den strukturellen Grafiken unter dem Begriff „RFE“ zusammengefasst sind.

Diese Gruppe von Positionen wurde mit „T_NichtY“ gekennzeichnet und marinefarben hervorgehoben. Da sich diese Positionen mit denen der „RFE“-Gruppe jedoch teilweise überschnitten, wurden solche sich überschneidenden Positionen zusätzlich gesondert unter dem Namen „RFEuT_NichtY“ zusammengefasst und können bei Bedarf separat markiert werden[*†] (Abb. 4.7).
Für die Definition von Kontakt einer Position zu Aminosäureresten innerhalb des klassenspezifischen Konsensmotivs des Liganden wurde sich an den Arbeiten von Brannetti, et al.(2000) [14] und Ferraro, et al. (2006) [41] orientiert, in denen bezüglich SH3-Domänen Aminosäurereste von Domäne und Ligand als in Kontakt angesehen wurden, wenn die kürzeste Distanz zwischen ihren Atomen kleiner war als die Summe ihrer van der Waals Radien plus einer Toleranz von 3 Å. Entsprechend galt eine Position hinsichtlich eines Differenzierungsschritts als in Kontakt, wenn an ihr dieses Kriterium in mindestens in einer der strukturellen Darstellungen des betreffenden Differenzierungsschritts erfüllt war.

Die tertiär- bzw. quartärstrukturellen Darstellungen bestätigen den sich schon in den Ergebnissen von Kapitel 4.2 aufdrängenden Verdacht, dass durchaus nicht alle der am häufigsten isolierten Positionen zwangsläufig auch an der Bindungsstelle lagen, geschweige denn eine zentrale Rolle bei der Bindung zu spielen schienen. Dennoch befanden sich aber hinsichtlich der einzelnen Differenzierungsschritte durchschnittlich 46,95% der Positionen ihrer jeweiligen „RFE“-Gruppe in Kontakt zu Aminosäureresten innerhalb des Konsensmotivs (Tabelle 4.11). Bedenkt man zudem, dass der gesamte Bereich einer Domäne, an dem die Aminosäurereste innerhalb des Konsensmotivs des Liganden mit ihr in Kontakt stehen, jeweils relativ klein ist, erscheint diese Zahl doch recht eindrucksvoll. Hinsichtlich des Subklassifikators *„1K vs. Nicht-1K“* muss aber bedacht werden, dass Position 8 hier Teilbereich C entstammte. Die einzige Position, welche sich bezüglich dieses Subklassifikators in Teilbereich A/B befand (Position 21), bot in keiner der Darstellungen dieses Differenzierungsschritts Kontakt mit Aminosäureresten innerhalb des Konsensmotivs. Auffällig ist, dass die Menge an Positionen innerhalb der „RFE“-Gruppe eines Subklassifikators, welche sich in Kontakt zu Aminosäureresten innerhalb des Konsensmotivs zeigten, scheinbar nicht von der Präzision des betreffenden Subklassifikators abhing. So waren es beispielsweise bezüglich des eher unpräzisen Subklassifikators *„X_ORS vs. Nicht-X_ORS“* 50,00% der Positionen, während dies bezogen auf den deutlich präziseren Subklassifikator *„2R vs. Nicht-2R“* nur auf 16,67% der Positionen zutraf. Umgekehrt verhält es sich etwa bei den

---

[*] Das Programm zur Berechnung der „T_NichtY“- und „RFEuT_NichtY“-Gruppen von Positionen trägt den Namen „Berechnung_Pos_mind_10x_IP“ und findet sich im Anhang auf DVD unter: „~\Pymol\YvsNichtY“ (Kapitel 7.1.7, 7.2 und 7.3).

[†] Die gespeicherten Zwischenstände der auf diese Weise aufbereiteten graphischen Darstellungen finden sich im Anhang auf DVD unter: „~\Pymol\....“, wobei „....“ der Bezeichnung des jeweiligen Subklassifikationsschritts entspricht (*„1R vs. 1K vs. 1@“* = „Klassen1“ und *„2R vs. 2K vs. 2D“* = „Klassen2“). Die Namen dieser Dateien setzen sich jeweils aus der Klasse der Beispieldomäne, ihrem PDB-Identifikator sowie der Bezeichnung „-PyMOL.pse“ zusammen (Kapitel 7.1.7 und 7.3).

Subklassifikatoren *„Klassen 1 vs. Klassen 2“* und *„2K vs. Nicht-2K“*: hier besaßen bezüglich des Subklassifikators *„Klassen 1 vs. Klassen 2“* (dem präziseren der beiden) 71,43% der Positionen seiner „RFE“-Gruppe Kontakt, während dies hinsichtlich des Subklassifikators *„2K vs. Nicht-2K“* nur bei 33,33% der Positionen seiner „RFE“-Gruppe beobachtet werden konnte (Tabelle 4.11).

Differenzierungsschritt	Position mit Kontakt zu Aminosäureresten innerhalb des Konsensmotivs	Verhältnis zu sämtlichen Positionen der jeweiligen „RFE"-Gruppe
„1R vs. Nicht-1R“	13, 15, 16, 32, 48, 49 und 52	7/15 (46,67%)
„1K vs. Nicht-1K“	8	1/2 (50,00%)
„1@ vs. Nicht-1@“	15, 32 und 48	3/4 (75,00%)
„2R vs. Nicht-2R“	12	1/6 (16,67%)
„2K vs. Nicht-2K“	12, 50 und 52	3/9 (33,33%)
„2D vs. Nicht-2D“	53	1/3 (33,33%)
„Klassen 1“ („1R vs. 1K vs. 1@“)	13, 32 und 48	3/6 (50,00%)
„Klassen 2“ („2R vs. 2K vs. 2D“)	12 und 53	2/4 (50,00%)
„Klassen 1 vs. Klassen 2“	11, 13, 33, 35 und 52	5/7 (71,43%)
„X_ORS vs. Nicht-X_ORS“	7, 8, 13, 48, 51 und 52	6/12 (50,00%)
„Y vs. Nicht-Y“	11 und 16	2/5 (40,00%)

Tabelle 4.11 Auflistung der Positionen der einzelnen „RFE“-Gruppen mit Kontakt zu Aminosäureresten innerhalb des Konsensmotivs, sowie ihrem Mengenverhältnis zu sämtlichen Positionen der jeweiligen „RFE“-Gruppe. Durchschnittliches Verhältnis innerhalb einer „RFE“-Gruppe von Positionen mit Kontakt zu Aminosäureresten innerhalb des Konsensmotivs zu sämtlichen Positionen der Gruppe: 46,95% mit einer Standardabweichung von 15,84%.

Die genauere Untersuchung der Positionen mit Kontakt aus den einzelnen „RFE“-Gruppen hinsichtlich ihrer jeweiligen Differenzierungsschritte lässt vermuten, dass die Mehrzahl dieser Positionen auch Einfluss auf die Präferenz einer Domäne für den Ligandentyp der einen oder anderen der zu jeweils zu differenzierenden Klassen hat. So befanden sich hinsichtlich der Differenzierungsschritte *„1R vs. Nicht-1R“*, *„1K vs. Nicht-1K“*, *„1@ vs. Nicht-1@“* und *„1R vs. 1K vs. 1@“* die meisten dieser Positionen mit den Aminosäureresten in Kontakt, die in den Konsensmotiven der Liganden zwischen den jeweils zu untersuchenden Klassen auch different sind und zur Definition ihrer Klasse beitrugen. Gleiches konnte auch bei Position 12 hinsichtlich der Differenzierungsschritte *„2R vs. Nicht-2R“*, *„2K vs. Nicht-2K“* und *„2R vs. 2K vs. 2D“* beobachtet werden, wobei diese Position zum Teil auch mit dem Kernmotiv in Kontakt stand. Bezüglich Position 53, welche sich in den „RFE“-Gruppen der Differenzierungsschritte *„2D vs. Nicht-2D“* und *„2R vs. 2K vs. 2D“* befand, konnte eine weitere interessante Beobachtung gemacht werden. Die Darstellungen dieser Differenzierungsschritte dokumentieren an dieser Position

zwischen Klasse *2D* und den jeweils von ihr zu differenzierenden Klassen bei Klasse *2D* eine Strukturveränderung der Bindungsstelle, welche möglicherweise die Positionierung des Liganden in der Bindungsstelle beeinflusst. Hinsichtlich der (Sub-)Klassen *2R* und *2K* bildet diese Position jeweils einen Teil der Scheidewand zwischen den beiden Bindungstaschen für das Kernmotiv ´xPxxP` und steht in Kontakt zum ersten Prolinrest des PxxPx+-Konsensmotivs der Typ-II Liganden dieser Klassen. In der Beispieldomäne für Klasse *2D* hingegen zeigt sich, dass diese Scheidewand hier aufgrund eines anderen Aminosäurerests an Position 53 nur rudimentär vorhanden ist, wodurch die beiden Bindungstaschen kaum voneinander getrennt sind. Möglicherweise ist dies auch einer der Gründe, warum das bezüglich dieser Klasse spezifische Konsensmotiv PxxDY des Liganden mit seinem ersten Prolinrest erst in der zweiten Bindungstasche zum Liegen kommt. Hinsichtlich der „RFE"-Gruppe des Differenzierungsschritts *„Klassen 1 vs. Klassen 2"* befanden sich vier Positionen (Positionen 11, 33, 35 und 52) jeweils mit Aminosäureresten des Kernmotivs in Kontakt. Auch dies erscheint biologisch nachvollziehbar, da das Kernmotiv ja aufgrund der zwischen diesen Klassen entgegengesetzten Orientierung der Liganden klassenspezifisch jeweils unterschiedlich in den Bindungstaschen positioniert ist [59]. Position 13 hingegen – aber auch Position 11, 33 und 35 – befanden sich mit Aminosäureresten in Kontakt, hinsichtlich derer sich die Konsensmotive der Liganden einzelner Subklassen voneinander unterscheiden. Es wird also ersichtlich, dass die Positionen nicht nur mit Aminosäureresten des Konsensmotivs Kontakt hatten, die jeweils Merkmale charakterisierten, welche den Liganden aller Subklassen jeweils einer von anderen zu unterscheidenden Klasse gemein waren, sondern auch mit Aminosäureresten, die jeweils die speziellen Merkmale der Liganden einzelner Subklassen charakterisierten. Aus dieser Beobachtung lässt sich ableiten, dass für die Klassifikatoren wohl auch die speziellen sequenziellen Charakteristika der Domänen einzelner Subklassen einer zu differenzierenden Klasse eine Rolle bei der Klassifikation zu spielen scheinen. Ähnliches konnte auch bei Position 50 und 52 hinsichtlich des Differenzierungsschritts *„2K vs. Nicht-2K"* beobachtet werden. Diese befanden sich hier jeweils in Kontakt mit dem Kernmotiv der Liganden, dessen Aufbau ja in Klasse *Nicht-2K* zumindest im Falle der Subklasse *2D* different ist. So besitzt das Konsensmotiv dieser Subklasse nur einen Prolinrest, der zudem erst im Bereich der zweiten Bindungstasche zum Liegen kommt. Da Position 50 und 52 jeweils im Bereich der ersten beiden Bindungstaschen lagen, könnten im Falle von Domänen der Klasse *2D* an diesen Positionen also Aminosäurereste vorliegen, die zwar eine Bindung des Kernmotivs der Subklasse *2D* ermöglichen, die des Kernmotivs der anderen Klassen jedoch nicht. Somit könnten diese Positionen hinsichtlich des Differenzierungsschritts *„2K vs. Nicht-2K"* zumindest Einfluss darauf haben, ob eine Bindung von Liganden der Klasse *2K* überhaupt möglich ist, auch wenn passende Aminosäurereste hier die Bindung von Liganden der jeweils anderen Klassen nicht ausschließen.

Aufgrund fehlender Beispieldomäne für Klasse *X_ORS* war die Analyse der Positionen mit Kontakt hinsichtlich des Differenzierungsschritts *„X_ORS vs. Nicht-X_ORS"* nur eingeschränkt möglich. Soweit

beurteilbar, schien hier aber keine dieser Positionen aus biologischer Sicht gänzlich abwegig, da sie sich in den Beispieldomänen der Klasse *Nicht-X_ORS* entweder mit den Prolinresten des Kernmotivs oder mit Aminosäureresten in Kontakt befanden, hinsichtlich derer sich die Konsensmotive der Liganden der einzelnen Subklassen der Klasse *Nicht-X_ORS* voneinander unterscheiden. Somit markierten die Positionen mit Kontakt hier jeweils spezifische Merkmale der Liganden der einzelnen Subklassen der Klasse *Nicht-X_ORS*.

Die einzigen Fälle, in denen ein Einfluss der betreffenden Positionen auf die Präferenz für den jeweiligen Ligandentyp hinsichtlich der zu differenzierenden Klassen anhand der entsprechenden strukturellen Darstellungen eher fraglich erschien, waren Position 8 hinsichtlich des Differenzierungsschritts *„1K vs. Nicht-1K“*, Position 49 und 52 hinsichtlich des Differenzierungsschritts *„1R vs. Nicht-1R“* und die Positionen mit Kontakt hinsichtlich des Differenzierungsschritts *„Y vs. Nicht-Y“* (Position 11 und 16). Die ersten drei der genannten waren in Darstellungen ihrer jeweiligen Differenzierungsschritte jeweils in Kontakt mit dem Kernmotiv der Liganden. Da dieses aber in den betreffenden Differenzierungsschritten hinsichtlich sämtlicher Klassen sowohl die gleiche Orientierung wie auch den gleichen Aufbau hat, erscheint ein Einfluss dieser Positionen hier eher unwahrscheinlich. Allerdings ist es vorstellbar, dass aufgrund des in den Konsensmotiven zwischen den betreffenden Klassen jeweils unterschiedlichen Aminosäurerests das Kernmotiv klassenspezifisch auch anders in den Bindungstaschen zum Liegen kommt, wodurch ein Einfluss dieser Positionen wieder wahrscheinlicher würde.

Position 11 und 16 der „RFE“-Gruppe des Differenzierungsschritts *„Y vs. Nicht-Y“*, hatten bezüglich der Subklassen von Klasse *Nicht-Y* zwar in vielen Fällen Kontakt mit dem Konsensmotiv, jedoch nicht in der Darstellung für Klasse *2D*, sodass ihr Einfluss hier zumindest eher fragwürdig erscheint. Zudem bot die Beispieldomäne der Subklasse *2R* an diesen Positionen die gleichen Seitenketten wie die Beispieldomäne Ydl117w der Klasse *Y*, sodass sich die Bindungsfähigkeit einer Domäne wohl nicht alleine an diesen Positionen entscheiden dürfte.

Eine Feststellung ganz anderer Art erbrachte der Vergleich der strukturellen Darstellungen der OvO-Multiclass-Subklassifikatoren *„1R vs. 1K vs. 1@“* und *„2R vs. 2K vs. 2D“* mit denen der ihnen im arbiträren Aufbau des Gesamtklassifikators jeweils folgenden binären Subklassifikatoren. Hierbei zeigte sich, dass sämtliche Positionen mit Kontakt aus der „RFE“-Gruppe des betreffenden OvO-Multiclass-Subklassifikators auch in den „RFE“-Gruppen der jeweiligen binären Subklassifikatoren vorlagen, wobei die „RFE“-Gruppen der einzelnen binären Subklassifikatoren für sich nicht alle dieser Positionen vorweisen mussten und zudem zumeist noch weitere Positionen mit Kontakt beinhalteten. Diese Beobachtung bestätigt die Annahme, dass sich die aus den Feature Selections der OvO-Multiclass-Subklassifikatoren gewonnenen Informationen anhand der Feature Selections dieser binären Subklassifikatoren präzisieren lassen, da sich die gemeinsamen Positionen so besser einem bestimmten Differenzierungsschritt zuordnen ließen. Zudem befanden sich in den „RFE“-Gruppen einiger der

binären Subdifferenzierungen noch weitere Positionen mit Kontakt, die möglicherweise ebenso Einfluss auf die Ligandenpräferenz hinsichtlich des jeweiligen Differenzierungsschritts haben.

4.3.2 Klassenspezifische Sequenzanalyse mit WebLogo [26]

Um noch weiterführende Aussagen darüber treffen zu können, inwieweit die am häufigsten isolierten Positionen mit Kontakt zum Liganden möglicherweise biologischen Schlüsselpositionen bei der Bindung entsprechen könnten, sollten die an diesen Positionen klassenspezifisch am häufigsten vorkommenden Aminosäurereste bestimmt werden. Dies beruht auf der Überlegung, dass Positionen, an welchen klassenspezifisch einzelne bzw. bestimmte Aminosäuren besonders häufig beobachtet werden können, wohl auch eher als physiologische Schlüsselpositionen bei der Ligandenpräferenz in Frage kommen dürften als solche, an denen sich keine derartigen Muster finden. So konnte bei Untersuchungen zur Affinität von „Klasse *2D*"-Domänen für ihr Konsensmotiv PxxDY beispielsweise bereits festgestellt werden, dass hierbei Position 33 eine entscheidende Rolle zukommt, da sämtliche Domänen der Klasse *2D* an dieser Position positiv geladene Aminosäurereste (Arginin bzw. Lysin) tragen und Punktmutation dieser Aminosäurereste zum Verlust der Bindungsaffinität für das Konsensmotiv PxxDY führten [17]. Da das Aminosäurespektrum dieser Position innerhalb der übrigen Klassen zudem deutlich breiter ist und sich auch eher aus anderen Aminosäureresten zusammensetzt, dürfte diese Position sogar speziell für die Affinität einer Domäne zu Liganden der Klasse *2D* mitverantwortlich sein. Dass die Erforschung solcher Schlüsselpositionen von enormer Bedeutung sein dürfte, zeigt sich nicht zuletzt in der klinischen Relevanz eines Vertreters der eben erwähnten Klasse *2D*. So konnte das Enzym Eps8, dessen SH3-Domäne ja ebendieser Klasse zugehörig ist, mit der Genese zahlreicher solider wie hämatologischer Neoplasien in Verbindung gebracht werden, wie beispielsweise der Entstehung von Schilddrüsen-, Zervix-, Ovarial-, Kolon- oder Pankreaskarzinomen oder auch der Mixed-Lineage Leukämie, und wird aufgrund einer häufig beschriebenen Korrelation seiner Expression innerhalb neoplastischer Zellen mit der (in diesem Falle eher schlechten) Prognose des Patienten inzwischen auch als prognostischer Tumormarker diskutiert [84]. Es handelt sich hierbei um ein ubiquitär exprimiertes Enzym, das sich physiologischerweise innerhalb diverser Signalkaskaden als regulatives Element wiederfindet, etwa innerhalb der Weiterleitung von EGF-Rezeptor vermittelten Signalen oder der Koordination des „Aktin-Capping" [29][84]. Verschiedene Arbeiten beschreiben inzwischen, dass eine Überexpression von Eps8 zur Verstärkung EGF-Rezeptor vermittelter mitogener Signale führt und maligne Zelltransformationen (unter EGF-Stimulation) fördert [84][89][88][16][68]. Zudem scheint dieses Enzym auch eine wichtige Rolle hinsichtlich der Migration von Tumoren zu spielen [84][47]. Da bei vielen der von Eps8 vermittelten Prozesse auch deren SH3-Domäne eine zentrale Rolle spielt [84] – beispielsweise durch Bindung an Abi-1 bei der Formation des Eps8–Abi-1–Sos-1-Komplexes, welcher wiederum für die Aktivierung des G-Proteins Rac von Bedeutung ist [84][29] – könnte somit über eine funktionelle Inhibition dieser Domäne innerhalb neoplastischer

Zellen, etwa durch einen kompetitiven Antagonisten im Sinne eines artifiziellen, physiologisch ansonsten funktionslosen Bindungspartners, eine signifikante Drosselung dieser Prozesse erreicht werden. Informationen über klassenspezifisch bindungsrelevante Schlüsselpositionen der Domäne, wie z.B. die weiter oben beschriebene Position 33, könnten hierbei dazu beitragen diese Bindungspartner möglicherweise sogar so gestalten zu können, dass sie nicht nur eine höhere Affinität als ihr natürliches Vorbild für die entsprechende Domäne besitzen – was ja den antagonistischen Effekt verstärken dürfte, sondern auch eine geringere Kreuzreaktivität mit Domänen anderer Klassen vorweisen.

Zur Bestimmung der eingangs des Kapitels beschriebenen Aminosäurehäufigkeiten wurde sich der im Internet frei verfügbaren Applikation WebLogo [26] bedient, mit der das Erstellen komparativer Sequenzlogos für die einzelnen Klassen möglich war. Hierzu mussten die Sequenzen zunächst, für jeden Subklassifikator mit seinem speziellen Trainingssatz separat, ihren Klassen nach in einzelnen Dateien zusammengefasst werden, wobei sich die einzelnen Klassen jeweils aus den Subklassen zusammensetzten, welche im jeweiligen Differenzierungsschritt zu differenzieren waren. Dabei wurde hinsichtlich der Subklassifikatoren, welche mit regularisierten Trainingssätzen arbeiteten, jeweils der Trainingssatz verwendet, bei dem der Satz an künstlichen Sequenzen jeder Subklasse stets nach dem Vorbild sämtlicher Basissequenzen der entsprechenden Subklasse emittiert wurde. Beispielsweise wurden also bezüglich des Subklassifikators „*Y vs. Nicht-Y*“, welcher ja mit Regularisierungsstufe **R20** arbeitete, sämtliche Sequenzen der Klasse *Y* vom entsprechenden Trainingssatz der Stufe **R20** in einer und sämtliche Sequenzen der restlichen Klassen dieses Trainingssatzes, welche zusammen die Klasse *Nicht-Y* bildeten, in einer anderen Datei zusammengefasst. Um Auswirkungen der Anwendung künstlicher Sequenzen besser beurteilen zu können, sollten bezüglich der Subklassifikatoren, welche mit regularisierten Trainingssätzen arbeiteten, zudem klassenspezifische Sequenzlogos unter ausschließlicher Anwendung der Basissequenzen generiert werden. Daher mussten für diese Fälle die anhand der regularisierten Trainingssätze erstellten, klassenspezifischen Sequenzdateien noch um weitere, anhand des **R0**-Trainingssatzes erstellte ergänzt werden[*]. Anschließend wurden die Sequenzen jeder Datei mithilfe der Funktion *hmmalign* des Softwarepakets HMMer [36] an das SH3-Familien-HMM aligniert und die Alignments jeweils im *A2M*-Format gesichert.

[*] Die klassenspezifischen Sequenzdateien finden sich im Anhang auf DVD unter: „~\Weblogo\Sequenzen\....“, wobei „....“ der Bezeichnung des jeweiligen Subklassifikationsschritts entspricht („*1R vs. 1K vs. 1@*“ = „Klassen1“ und „*2R vs. 2K vs. 2D*“ = „Klassen2“). Die Benennung der Dateien entspricht jeweils dem Klassennamen der enthaltenen Sequenzen, gefolgt von der Höhe der Regularisierungsstufe (also 0, 20 bzw. 100); so enthält beispielsweise die Datei „Klassen2_100“ die Sequenzen der Subklassen *2R*, *2K* und *2D* regularisiert mit Stufe **R100** (Kapitel 7.1.8).

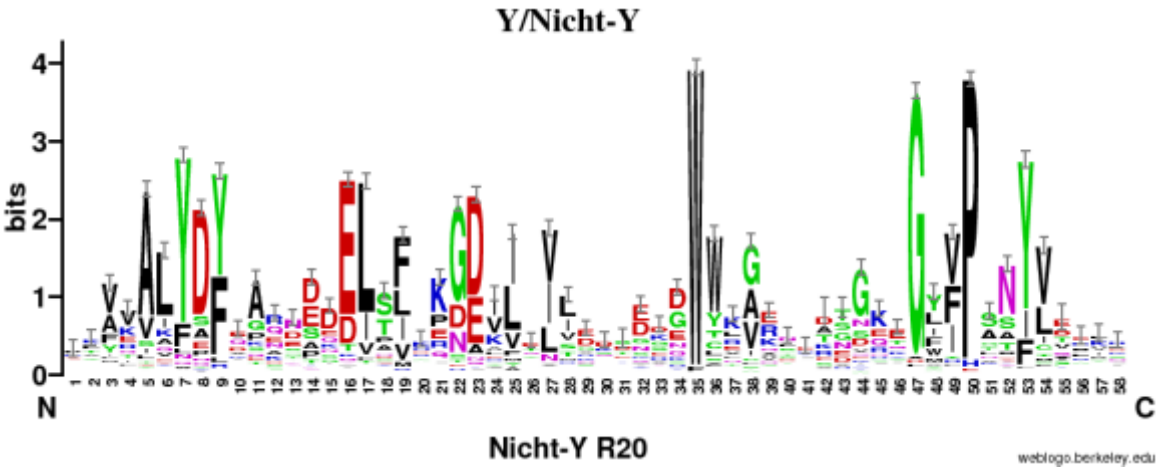


Abb. 4.8 Exemplarische Darstellung des Sequenzlogos der Klasse *Nicht-Y* bezüglich des Differenzierungsschritts *„Y vs. Nicht-Y“*. Die Nummern auf der X-Achse entsprechen den jeweiligen Positionsnummern der Alignments. Das Logo gibt die Häufigkeit einer Aminosäure an einer bestimmten Position in Form der relativen Höhe ihres Buchstaben auf der Y-Achse wieder. Die Gesamthöhe der Buchstabensäulen an den Positionen entspricht dem Grad der Sequenzkonservierung der einzelnen Positionen, welche auf der Y-Achse in Form von Bits wiedergegeben wird. Da die Höchstmenge an Bits bezüglich jeder Position maximal 4,3 beträgt (mit 20 mögliche Aminosäuren an jeder Position, $\log_2 20 = 4{,}3$ Bits pro Aminosäure), ist die Skala der Y-Achse auf etwas über 4 Bits beschränkt [125]. Die Gesamthöhe der Error Bars spiegelt die doppelte Höhe der „Small Sample Correction“ wider. Die Farbgebung der einzelnen Aminosäuren entspricht dem von der Applikation vorgegebenen Standard: polare Aminosäuren sind grün, basische blau, saure rot und hydrophobe schwarz gefärbt. In der Abbildung lässt sich gut erkennen, dass sich mit dieser Darstellungsform die unterschiedlichen Konservierungsgrade der einzelnen Positionen besonders schön herausstellen lassen, während das Auslesen der einzelnen Aminosäurereste, besonders an gering konservierten Positionen, eher schwer fällt.

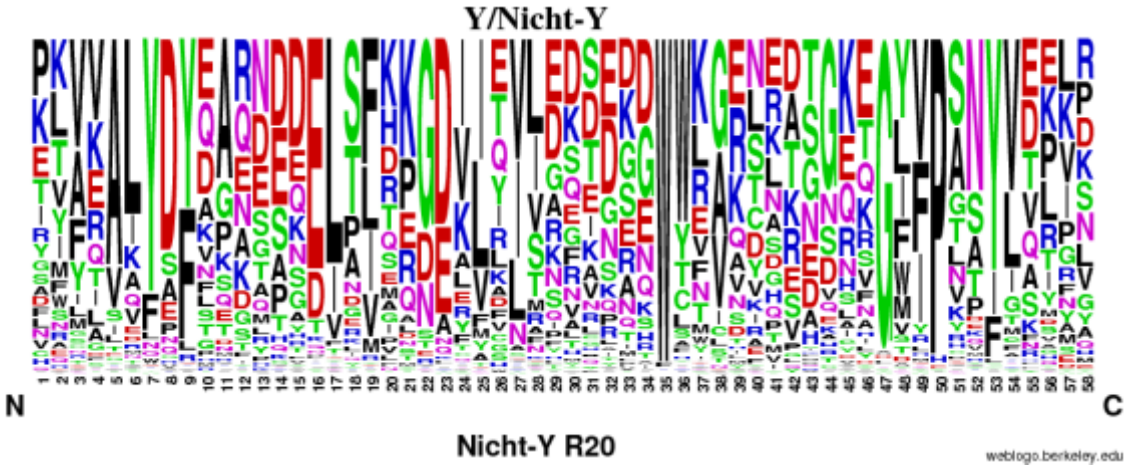


Abb. 4.9 Darstellung des Frequency Plots, welcher mit dem Sequenzlogo aus Abb. 4.8 korreliert. Die Nummerierung der X-Achse und die Farbgebung der Aminosäuren entsprechen denen aus Abb. 4.8. Im Unterschied zu den Sequenzlogos wird auf der Y-Achse der Frequency Plots nur die Häufigkeit einer Aminosäure an einer bestimmten Position in Form der relativen Höhe ihres Buchstaben wiedergegeben. Dadurch sind die Buchstabensäulen insgesamt gleich hoch, was den Vergleich hinsichtlich der Häufigkeit einzelner Aminosäuren an einer Position zwischen den Sequenzlogos von zu differenzierenden Klassen bei unterschiedlichen Konservierungsgraden besser ermöglicht.

Durch Entfernung sämtlicher Spalten mit Sternchen bzw. Punkten wurde die Nummerierung der Alignments der in dieser Arbeit üblichen, bestehend aus 58 Aminosäurepositionen angepasst*. Die so erstellten Alignments konnten nun in WebLogo [26] verwendet werden, um die einzelnen Sequenzlogos zu generieren. Bezüglich der Parameter des Bildformates wurden hierbei – mit Ausnahme von „Logo Size per Line“ und „Bitmap Resolution“, welche auf 18 x 7 cm respektive 60 pixels/cm festgelegt wurden – jeweils die Standardoptionen gewählt. Als Titel des Sequenzlogos wurde der Name des jeweiligen Differenzierungsschritts gewählt, dem das Logo zugeordnet war. Die Bezeichnung der X-Achse bestand jeweils aus dem Klassennamen der abgebildeten Sequenzen gefolgt von der verwendeten Regularisierungsstufe. Nachdem zur Erstellung der Sequenzlogos die durch die Betreiber der Seite generell empfohlene Option „Small Sample Correction“ standardgemäß verwendet wurde, wurde zur Dokumentation der Auswirkungen dieser zudem die Option „Show Error Bars“ aktiviert. Bezüglich der restlichen Parameter wurden stets die Standardoptionen gewählt (Abb. 4.8). Da sich in einigen der so erstellten Sequenzlogos die verschiedenen Aminosäuren an den einzelnen Positionen zum Teil nur sehr schlecht auslesen und bei unterschiedlichen Konservierungsgraden einer Position zwischen zu differenzierenden Klassen auch nicht vergleichen ließen, wurde zudem für jedes Alignment ein separater „Frequency Plot“ erstellt. Bei diesen wurde konsequenterweise auf die Option „Small Sample Correction“ und dementsprechend auch auf die Option „Show Error Bars“ verzichtet† (Abb. 4.9).

Der Vergleich der Sequenzlogos alleine eines Differenzierungsschritts macht bereits deutlich, wie aufwendig eine rein manuelle Analyse sämtlicher Aminosäurepositionen im Bereich der Bindungsstelle im Hinblick auf etwaige Schlüsselpositionen wäre. So müssten all diese Positionen auf signifikante sequenzielle Differenzen zwischen den verschiedenen Klassen hin untersucht werden, die zudem noch entweder signifikante chemische oder strukturelle Differenzen mit sich bringen müssten, sodass eine Affinitätsänderung zu den verschiedenen Liganden bewirkt würde. Erschwerend käme hinzu, dass die Bindungsstellen der einzelnen Domänen zum Teil ja durch unterschiedliche Positionen definiert sind. Da jedoch hauptsächlich die anhand der Feature Selections am häufigsten isolierten Positionen mit Kontakt zum Liganden untersucht werden sollten, konnten die Analysen hier auf eine überschaubare Menge reduziert werden. Bei Betrachtung der Sequenzlogos bzw. Frequency Plots regularisierter Klassen mit ihren nicht regularisierten Äquivalenten fiel auf, dass durch die Anwendung künstlicher

---

* Die Alignments der Sequenzdateien finden sich im Anhang auf DVD unter: „~\Weblogo\Alignments\....“, wobei „....“ der Bezeichnung des jeweiligen Subklassifikationsschritts entspricht (*„1R vs. 1K vs. 1@“* = „Klassen1“ und *„2R vs. 2K vs. 2D“* = „Klassen2“). Die Benennung der Alignmentdateien entspricht der der Sequenzdateien, nur dass diese die Erweiterung „.a2m“ tragen (Kapitel 7.1.8).

† Die klassenspezifischen komparativen Sequenzlogos sowie ihre Frequency Plots finden sich im Anhang auf DVD unter: „~\Weblogo\Logos\....“, wobei „....“ der Bezeichnung des jeweiligen Subklassifikationsschritts entspricht (*„1R vs. 1K vs. 1@“* = „Klassen1“ und *„2R vs. 2K vs. 2D“* = „Klassen2“). Die Benennung der Logos entspricht der der Sequenzdateien, nur dass diese die Erweiterung „.png“ tragen. Die Frequency Plots sind auf dieselbe Weise benannt, nur dass die Dateien dieser vor der Erweiterung „.png“ noch durch die Bezeichnung „-Frequency“ gekennzeichnet sind (Kapitel 7.1.8).

Sequenzen die Varianz der Aminosäuren an den Positionen teilweise nicht nur zwischen Positionen mit unterschiedlichem sondern sogar mit gleichem Konservierungsgrad unterschiedlich zunahm. Zudem zeigte sich auch zum Teil an Positionen eine Zunahme der Varianz, welche in allen Basissequenzen bezüglich einer/mehrerer teilweise auch allen Klassen stets dieselbe Aminosäure aufwiesen. Insgesamt waren die durch die Anwendung künstlicher Sequenzen hervorgerufenen Änderungen der Konservierungsgrade jedoch in den meisten Fällen in etwa proportional zum Konservierungsgrad der Basissequenzen an den jeweiligen Positionen.

Die Analyse der bezüglich der einzelnen Subklassifikatoren jeweils am häufigsten isolierten Positionen mithilfe der entsprechenden Sequenzlogos bzw. Frequency Plots offenbarte, dass an diesen Positionen zwischen den zu differenzierenden Klassen zumeist auch größere Unterschiede hinsichtlich der Sequenzkonservierung und/oder Häufigkeit einzelner Aminosäuren existierten. Mit anderen Worten, die Signifikanz der Positionen zur Klassifikation ließ sich zumeist auch auf Sequenzebene nachvollziehen. Allerdings traf dies nicht auf jeden Fall zu, so erschien beispielsweise besonders hinsichtlich des Subklassifikators *„2K vs. Nicht-2K“* die Signifikanz von Position 50 und hinsichtlich des Subklassifikators *„Klassen 1 vs. Klassen 2“* die von Position 35 auf Sequenzebene fraglich, da an diesen keine größeren Unterschiede diesbezüglich zwischen den zu differenzierenden Klassen festzustellen waren. Unklar blieb auch, weshalb seitens der Feature Selections eine Position für signifikant und eine andere mit ähnlichen Eigenschaften für eher unwichtig erachtet wurde.

Auffallend war zudem die hohe Konkordanz zwischen den Aminosäuren, auf welche sich die hinsichtlich der einzelnen Positionen aus Teilbereich A und B eines Subklassifikators jeweils isolierten Features bezogen (Kapitel 7.1.6 und Ergebnishinweise Kapitel 7.3), mit den an diesen Positionen im betreffenden Differenzierungsschritt zwischen den zu differenzierenden Klassen tatsächlich mit größerer Unterschiedlichkeit vorkommenden Aminosäuren. Bezüglich der Subklassifikatoren *„2R vs. 2K vs. 2D“*, *„1K vs. Nicht-1K“* und *„2D vs. Nicht-2D“* traf dies sogar auf sämtliche Aminosäuren zu, welche an den jeweiligen Positionen durch die jeweils isolierten Features beschrieben wurden. Besonders deutlich wird diese Beobachtung am Beispiel hochkonservierter Positionen, an denen fast ausschließlich bezüglich sämtlicher zu differenzierender Klassen nur eine einzige Aminosäure vorkommt. In diesen Fällen wurden häufig eher die Features selektiert, welche die wenigen Aminosäuren beschreiben, die neben der hochkonservierten außerdem noch an dieser Position vorkamen und bezüglich welcher (wenn auch kleinere) Differenzen zwischen den zu untersuchenden Klassen existierten. Einzig hervorstechende Ausnahme hiervon waren die Features der im Rahmen des Subklassifikators *„2K vs. Nicht-2K“* am häufigsten isolierten Positionen. Die sie beschreibenden Aminosäuren waren in den meisten Fällen weder in der einen noch der anderen Klasse an den betreffenden Positionen mit besonderer Häufigkeit vertreten. So beschrieben hier durchschnittlich nur etwa 37% der isolierten Features einer Position aus Teilbereich A und B Aminosäuren, die an der jeweiligen Position auch tatsächlich in größerer Unterschiedlichkeit zwischen den zu differenzierenden

Klassen vorkamen. Auch der Subklassifikator *„X_ORS vs. Nicht-X_ORS“* nimmt eine gewisse Sonderrolle ein, da hier jeweils nahezu alle Features bezüglich der am häufigsten isolierten Positionen selektiert wurden, sodass solche Analysen hier nicht sinnvoll möglich waren. Letztere Beobachtung dürfte sich damit erklären lassen, dass ja bezüglich der Positionen aus den Teilbereichen A und B eines Subklassifikators stets alle Features annotiert wurden, die während sämtlicher Feature Selections dieses Subklassifikators bereits einmal selektiert wurden. Es genügte also bereits einmal eine entsprechend hohe, für optimal gewertete Feature-Menge, damit hinsichtlich dieser Positionen derart viele (zum Teil auch unwichtigere) Features annotiert wurden. Dies könnte auch einer der Gründe dafür sein, weshalb bezüglich der jeweils am häufigsten isolierten Positionen der anderen Subklassifikatoren teils nicht alle der annotierten Features Aminosäuren beschrieben, die zwischen den zu differenzierenden Klassen an den betreffenden Positionen tatsächlich auch mit größerer Unterschiedlichkeit vorlagen. Ein anderer Grund hierfür könnte darin liegen, dass für die Signifikanz eines Features hauptsächlich eine möglichst unterschiedliche Konservierung seiner einzelnen Werte zwischen den zu differenzierenden Klassen von Bedeutung zu sein scheint, die Zahlenspanne der Werte selbst hingegen eher weniger.

Stellt man diese Konkordanzvergleiche hinsichtlich der regularisierten Subklassifikatoren mit den Sequenzlogos ihrer nicht regularisierten Trainingssätze an, ist festzustellen, dass die geschilderten Übereinstimmungen auch hier in den meisten Fällen vorlagen, wenn auch etwas weniger häufig als im Vergleich mit den Sequenzlogos ihrer regularisierten Trainingssätze. Dennoch lag die Konkordanz auch hier zumeist bei weit über 70%.

Der Vergleich der OvO-Multiclass-Subklassifikatoren *„1R vs. 1K vs. 1@“* bzw. *„2R vs. 2K vs. 2D“* mit den ihnen im arbiträren Aufbau des Gesamtklassifikators jeweils folgenden binären Subklassifikatoren ließ noch weitere interessante Beobachtungen zu: Zum einen zählten viele der hinsichtlich der binären Subklassifikatoren am häufigsten isolierten Positionen jeweils auch hinsichtlich des betreffenden OvO-Multiclass-Subklassifikators zu den häufigsten. Zum anderen wurden nahezu sämtliche Features, welche bezüglich dieser sich überschneidenden Positionen in den Feature Selections des betreffenden OvO-Multiclass-Subklassifikators isoliert wurden, auch in den Feature Selections der jeweiligen binären Subklassifikatoren isoliert, wobei die einzelnen binären Subklassifikatoren für sich nicht alle der Positionen bzw. Features vorweisen mussten. Auch variierte die Gesamtzahl an isolierten Features bezüglich dieser Positionen zum Teil erheblich, mit anderen Worten: in den Resultaten der einzelnen binären Klassifikatoren für sich lagen an den gemeinsamen Positionen häufig nicht alle dieser Features und meist zudem noch weitere vor. Bei genauerer Betrachtung dieser Positionen mithilfe der jeweiligen Sequenzlogos der betreffenden Differenzierungsschritte ließ sich feststellen, dass die Features, welche an diesen Positionen jeweils dem OvO-Multiclass- und dem betreffenden binären Subklassifikator gemein waren, in den betreffenden Differenzierungsschritten Aminosäuren beschrieben, die sich aufgrund ihres stärker unterschiedlichen Vorkommens an diesen Positionen zwischen den untersuchten Klassen auch besonders gut zur Differenzierung eignen sollten. Die weiteren an diesen Positionen

bezüglich der einzelnen Subklassifikatoren noch speziell isolierten Features schienen jeweils hinsichtlich des betreffenden Differenzierungsschritts die speziellen Aminosäuredifferenzen an diesen Positionen zwischen den untersuchten Klassen noch präziser hervorzuheben. Diese Beobachtungen stützen erneut die Annahme, dass sich mithilfe der Feature Selections dieser binären Subklassifikatoren präzisiere Informationen gewinnen lassen als mit den Feature Selections der OvO-Multiclass-Subklassifikatoren, da sich zum einen mit ihrer Hilfe die bezüglich der OvO-Multiclass-Subklassifikatoren isolierten Features bzw. Positionen zumeist der Abgrenzung einer bestimmten Klasse zuordnen ließen und zum anderen die jeweils bezüglich der binären Subklassifikatoren noch speziell isolierten Features bzw. Positionen die Differenzen einzelner Klassen noch präziser herausstellten.

4.3.3 Integrative Analyse der Daten aus Kapitel 4.3.1 und 4.3.2

Anhand der strukturellen Darstellungen konnte bereits gezeigt werden, dass die Positionen mit Kontakt der „RFE“-Gruppe eines Differenzierungsschritts zumeist mit Bereichen innerhalb der Konsensmotive der Liganden interagierten, hinsichtlich derer auch Differenzen zwischen den jeweils zu untersuchenden Klassen zu beobachten waren. Inwieweit diese Positionen bezüglich der einzelnen Differenzierungsschritte allerdings jeweils auch tatsächlich Einfluss auf die Präferenz einer Domäne speziell für ihre Ligandenklasse haben – inwieweit also tatsächlich von biologischen Schlüsselpositionen gesprochen werden kann – lässt sich aber sicher nicht alleine hieran festmachen, denn die beschriebenen Beobachtungen basierten ja jeweils lediglich auf den Analysen einzelner Beispieldomänen für die jeweils zu differenzierenden (Sub-)Klassen. Um sie generalisieren zu können, müssen die Beobachtungen also auch entsprechend an den anderen Domänen – zumindest der überwiegenden Mehrzahl – der jeweiligen Klassen reproduzierbar sein. Überdies kann allein die Lage dieser Positionen nicht als einziges Beurteilungskriterium fungieren, vielmehr müssen auch die klassenspezifischen Aminosäureeigenschaften an diesen Positionen zur Affinität bzw. Präferenz der jeweiligen Klasse für ihren spezifischen Ligandentyp beitragen. Daher soll im Folgenden am Beispiel der Differenzierungsschritte *„2D vs. Nicht-2D“* und *„Klassen 1 vs. Klassen 2“* anhand ihrer entsprechenden strukturellen Darstellungen und Sequenzlogos exemplarisch diskutiert werden, inwieweit die Positionen mit Kontakt ihrer jeweiligen „RFE“-Gruppe hier diesen Ansprüchen genügen. Da für diese Untersuchungen an den betreffenden Positionen ja ausschließlich die klassenspezifischen Aminosäureeigenschaften natürlicher Sequenzen/Domänen von Interesse sind, wird stets auf die Sequenzlogos Bezug genommen, welche auf der Basis der nicht regularisierten Trainingssätze erstellt wurden.

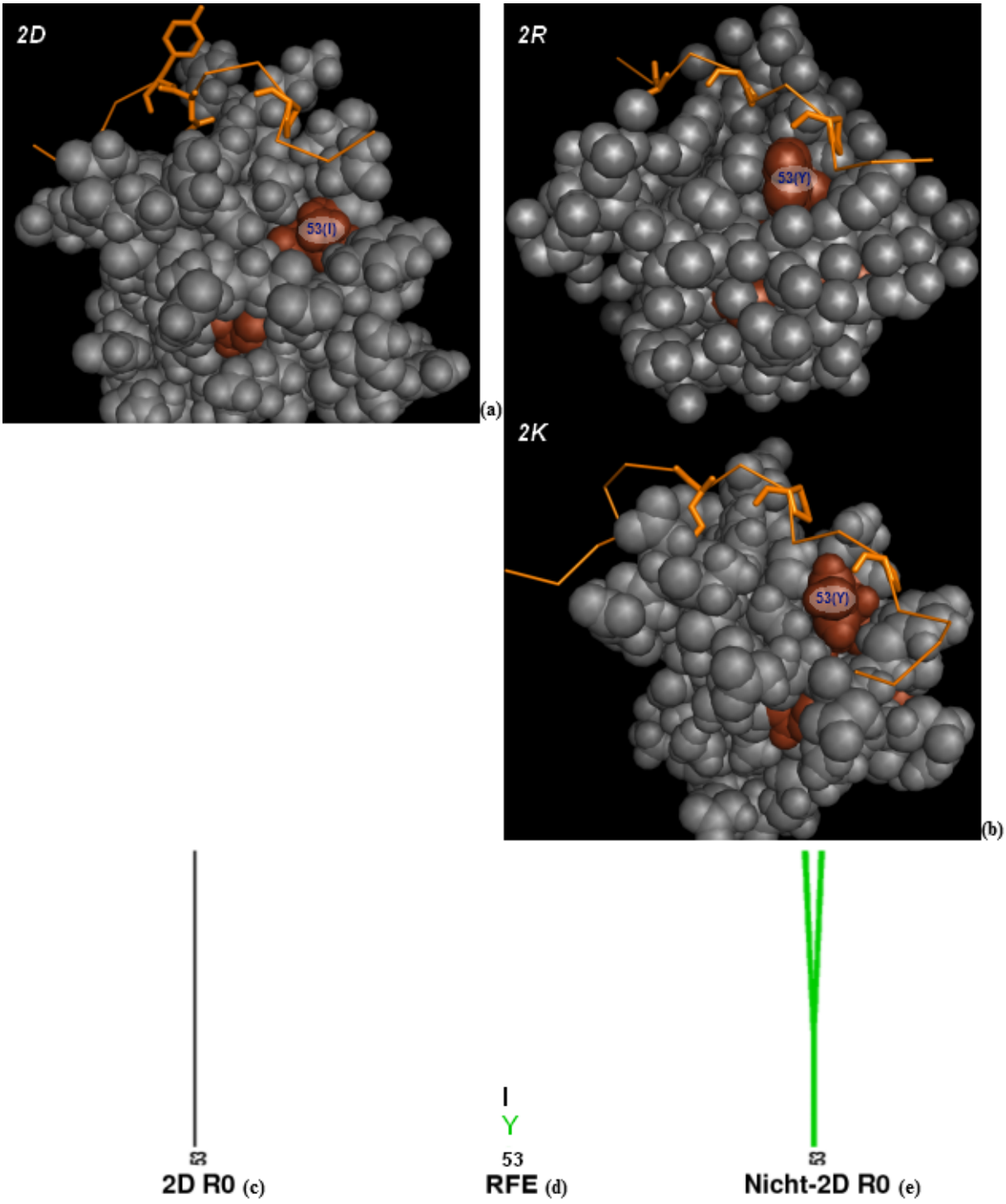


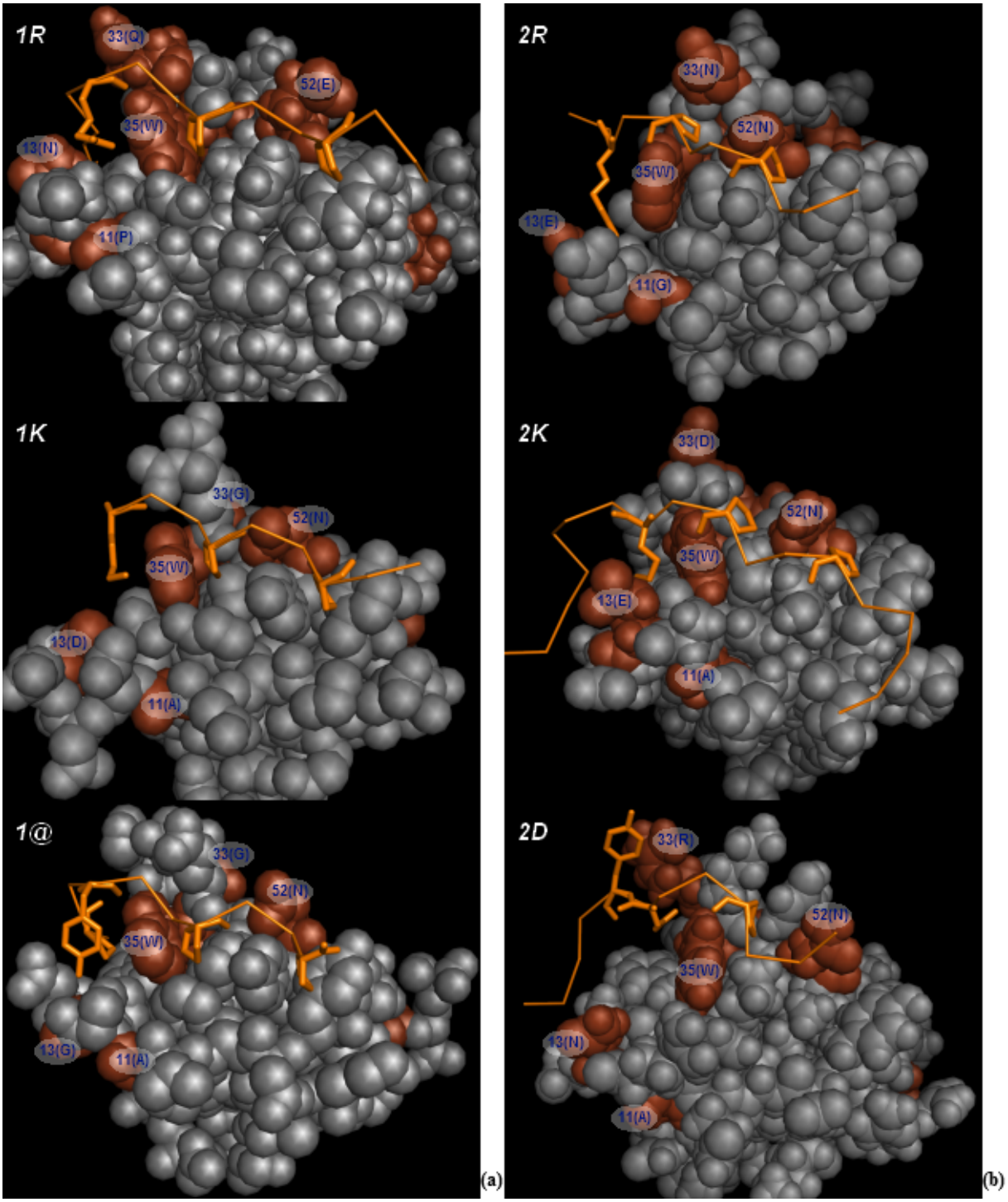
Abb. 4.10 Vergleich der PyMOL-Darstellungen des Differenzierungsschritts „*2D-Nicht2D*" – hinsichtlich Position 53 – zum einen mit den anhand der **R0**-Trainingssätze erstellten Sequenzlogos dieses Differenzierungsschritts sowie zum anderen mit den entsprechenden Ergebnissen von RFE [54][161]. (a) zeigt die Beispieldomäne für Klasse *2D* (Eps8R1_HS), während (b) die Beispieldomänen für Klasse *Nicht-2D* (Yfr024c für Subklasse *2R* und ABP-1 für Subklasse *2K*) darstellt. Der in den PyMOL-Darstellungen in Klammern gesetzte Buchstabe innerhalb der Beschriftung von Position 53 entspricht dem in der jeweiligen Beispieldomäne an dieser Position vorliegenden Aminosäurerest. Die in (c) und (e) dargestellten Ausschnitte der Sequenzlogos zeigen jeweils nur Position 53, wobei (c) dem Sequenzlogo der Klasse *2D* und (e) dem der Klasse *Nicht-2D* entnommen ist. Die in den Feature Selections dieses Differenzierungsschritts bezüglich Position 53 isolierten Features sind in Form der ihnen jeweils entsprechenden Aminosäuren in (d) aufgeführt und farbig den Sequenzlogos angepasst.

Wie bereits in Kapitel 4.3.1 erläutert, kann in den strukturellen Darstellungen hinsichtlich des Differenzierungsschritts *„2D vs. Nicht-2D“* zwischen Klasse *2D* und *Nicht-2D* (bestehend aus den Subklassen *2K* und *2R*) an Position 53, welche zur „RFE“-Gruppe dieses Differenzierungsschritts gehört, ein unterschiedlicher struktureller Aufbau der Bindungsstelle beobachtet werden. In den Darstellungen der Subklassen *2R* und *2K* liegt an dieser Position jeweils ein Tyrosinrest vor, der jeweils einen Teil der Scheidewand zwischen den beiden Bindungstaschen für das Kernmotiv ´xPxxP` bildet und in Kontakt zum ersten Prolinrest des PxxPx+-Konsensmotivs ihrer Liganden steht. Die Beispieldomäne von Klasse *2D* hingegen zeigt, dass diese Scheidewand hier aufgrund eines Isoleucinrests an Position 53 nur rudimentär vorhanden ist, wodurch die beiden Bindungstaschen kaum voneinander getrennt sind (Abb. 4.10 (a) und (b)). Hierdurch wird dem ersten Prolinrest innerhalb des für die Subklassen *2R* und *2K* typischen Konsensmotivs PxxPx+ keine Bindungsmöglichkeit geboten, was ein Grund dafür sein könnte, weshalb das Konsensmotiv der Klasse *2D* nicht das typische Kernmotiv ´xPxxP` vorweist. Umso eindrucksvoller erscheint nun der Vergleich der Sequenzlogos dieses Differenzierungsschritts hinsichtlich Position 53 (Abb. 4.10 (c) und (e)), denn er offenbart, dass sämtliche Domänen der Klasse *Nicht-2D* an dieser Position den beschriebenen Tyrosinrest besitzen, während die Domänen der Klasse *2D* stets den Isoleucinrest vorweisen. Da diese Position somit allen beschriebenen Anforderungen genügt, erscheint es also entsprechend wahrscheinlich, dass es sich hierbei tatsächlich um eine Schlüsselposition hinsichtlich dieses Differenzierungsschritts handelt. Ebenso eindrucksvoll erscheint die Analyse der bezüglich dieser Position isolierten Features, denn sie beziehen sich exakt auf die beiden beschriebenen Aminosäuren (Abb. 4.10 (d)). Allerdings zweifeln Cesareni, et al. (2002) [17] sowie Mongioví, et al. (1999) [102] die Schlüsselfunktion von Position 53 ein wenig an, da der an ihr in Domänen der Klasse *2D* stets vorliegende Isoleucinrest nicht für die Affinität von „Klasse *2D*“-Domänen zu ihrem Konsensmotiv PxxDY verantwortlich zu sein scheint. So konnte gezeigt werden, dass die Bindungsaffinität von I53Y mutierten „Klasse *2D*“-Domänen zu PxxDY-tragenden Peptiden nicht nur nicht abnahm, sondern im Falle des PxxDY-tragenden in vivo „Klasse *2D*“-Liganden Abi-1 sogar deutlich zunahm [102]. Wie jedoch der Einfluss dieser Position auf die Bindungsaffinität von Domänen der Klasse *Nicht-2D* zu ihrem jeweiligen Konsensmotiv ist, wurde dort nicht untersucht. Diesbezüglich wird ihr nämlich in einer Arbeit von Aitio, et al. (2008) [3], welche sich mit der strukturellen Analyse der SH3-Domäne von Eps8R1 und ihren Bindungseigenschaften befasst, durchaus eine Signifikanz zugesprochen. Dort geht man wie hier ebenso davon aus, dass der in Domänen der Klasse *Nicht-2D* an Position 53 vorliegende den Tyrosin- bzw. in einigen Fällen wohl auch Phenylalaninrest für den Kontakt zum ersten Prolinrest innerhalb des PxxPx+-Konsensmotivs dieser Klasse entscheidend ist. Ein Isoleucinrest an dieser Position wie in „Klasse *2D*“-Domänen würde den Aufbau der Bindungsstelle derart verändern, dass die Bindung dieses Prolinrests nicht mehr möglich wäre. Zudem liefert diese Arbeit auch eine schlüssige Erklärung für die von Mongioví, et al. (1999) [102] gemachte Beobachtung, dass I53Y mutierte „Klasse *2D*“-Domänen eine größere Affinität

zu Abi-1 besaßen als ihr Wildtyp. Betrachtet man nämlich auch die Sequenzabschnitte innerhalb von Abi-1 (PPP**PPVDY**EDEE), die das Konsensmotiv PxxDY unmittelbar flankierenden, so ist festzustellen, dass zwei Positionen N-terminal des Konsensmotivs ebenso ein Prolinrest vorliegt, welcher folglich im Bereich der ersten Bindungstasche zum Liegen kommen müsste. Da diese ja bei Wildtyp-„Klasse *2D*"-Domänen im Vergleich zu ihren I53Y mutierten Varianten nur rudimentär ausgebildet ist, dürfte dies nicht nur die höhere Affinität der I53Y mutierten Varianten zu Abi-1 erklären, sondern auch die Hypothese untermauern, dass Position 53 durchaus einen Einfluss auf die Ligandenpräferenz hinsichtlich der Differenzierung *„2D vs. Nicht-2D"* haben dürfte, und zwar speziell auf die Affinität zu Liganden der Klasse *Nicht-2D*.

In den strukturellen Darstellungen des Differenzierungsschritts *„Klassen 1 vs. Klassen 2"* zeigen vier Positionen (Positionen 11, 33, 35 und 52) jeweils mit Aminosäureresten des Kernmotivs Kontakt, während Position 13 – aber auch Position 11, 33 und 35 – mit Aminosäureresten der Liganden interagieren, hinsichtlich derer sich die Konsensmotive einzelner Subklassen voneinander unterscheiden (Abb. 4.11 (a) und (b)). Betrachtet man die Positionen aber genauer, lässt sich anhand ihrer klassenspezifischen Aminosäureeigenschaften lediglich an drei der fünf Positionen auch ein Beitrag dieser zur Affinität bzw. Präferenz der jeweiligen Klassen für ihren Ligandentyp vermuten.
So zeigt Position 11 nur in den strukturellen Darstellungen von Klasse *2R* und *2K* Kontakt zum Liganden. Hinsichtlich Klasse *2R* interagiert sie mit dem hydrophilen, basischen Teil des Argininrests an Position P-3, während sie in der Beispieldomäne von Klasse *2K* Kontakt mit dem eher hydrophoben Kernmotiv hat. Die klassenspezifischen Aminosäureeigenschaften an dieser Position weisen in den Sequenzlogos der beiden Subklassen jedoch keine größeren Unterschiede auf. Auch der Vergleich des Sequenzlogos aller „Klasse *1*"-Domänen mit dem der „Klasse *2*"-Domänen erbringt keinen weiteren Aufschluss, weshalb diese Position Einfluss auf die Ligandenpräferenz einer Domäne hinsichtlich dieses Differenzierungsschritts haben sollte. Zwar weist sie im Sequenzlogo der „Klasse *2*"-Domänen nur unpolare und von Seiten ihres Säure-/Basencharakters neutrale Aminosäurereste auf, während sich an ihr im Sequenzlogo der „Klasse *1*"-Domänen auch polare und von Seiten ihres Säure-/Basencharakters teils auch saure bzw. basische Aminosäurereste finden. Da die klassenspezifischen Aminosäureeigenschaften dieser Position jedoch selbst in Subklassen, in denen sie Kontakt mit dem Konsensmotiv der Liganden zeigt, eher unspezifisch sind, dürften diese Differenzen rein phylogenetischer bzw. zufälliger Natur sein. Der Einfluss von Position 35 auf die Präferenz des Ligandentyps erscheint aus sequenzanalytischer Sicht hingegen fraglich, da an ihr sowohl in sämtlichen „Klasse *1*"- wie auch „Klasse *2*"-Domänen stets derselbe Aminosäurerest (Tryptophan) vorliegt. Erstaunlicherweise wurde dieser Position in einer Arbeit von Fernandez-Ballester, et al. (2004) [40] aber dennoch ein sogar wesentlicher Einfluss auf die Affinität einer Domäne für Liganden der Klasse *1* bzw. Klasse *2* zugeschrieben. So könne der an dieser Position hochkonservierte Tryptophanrest zwei

Orientierungen annehmen, von welchen die eine die Bindung an „Klasse *1*“- während die andere die an „Klasse *2*“-Liganden ermöglicht. Die Orientierung des Tryptophanrests hinge dabei von den Aminosäurecharakteristika einer weiteren Position ab (Position 48). Lägen an dieser aromatische Aminosäurereste vor, so könne der Tryptophanrest frei zwischen beiden Orientierungen wechseln und sich somit der Klasse des Liganden anpassen.



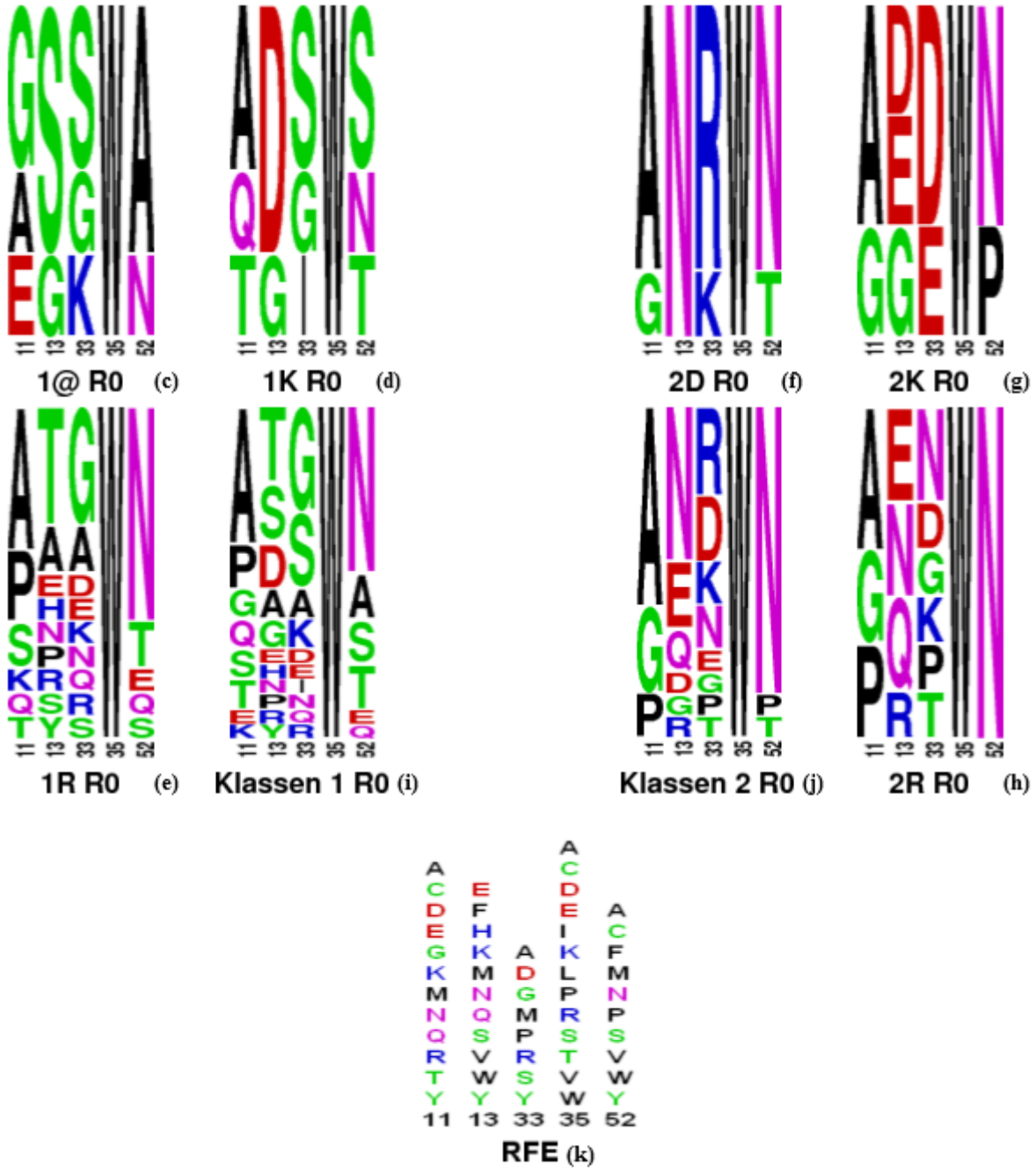


Abb. 4.11 Vergleich der PyMOL-Darstellungen des Differenzierungsschritts *„Klassen 1 vs. Klassen 2“* – hinsichtlich der Positionen mit Kontakt innerhalb der „RFE“-Gruppe dieses Differenzierungsschritts – zum einen mit den Sequenzlogos dieses Differenzierungsschritts sowie denen der Subklassen dieses Differenzierungsschritts und zum anderen mit den entsprechenden Ergebnissen von RFE [54][161]. (a) zeigt die Beispieldomänen für Klasse *1* (NBP2 für Subklasse *1R*, SHO1 für Subklasse *1K* und Abl1_H für Subklasse *1@*), während (b) die Beispieldomänen für Klasse *2* (Yfr024c für Subklasse *2R*, ABP-1 für Subklasse *2K* und Eps8R1_HS für Subklasse *2D*) darstellt. Der in den PyMOL-Darstellungen in Klammern gesetzte Buchstabe innerhalb der Beschriftung der Positionen mit Kontakt entspricht dem in der jeweiligen Beispieldomäne an dieser Position vorliegenden Aminosäurerest. Die in (c) bis (h) dargestellten Ausschnitte der Sequenzlogos zeigen jeweils nur die Positionen mit Kontakt. Die Beschreibungen unterhalb dieser Ausschnitte geben an, welchem Sequenzlogo der betreffende Ausschnitt entstammt. Die in den Feature Selections dieses Differenzierungsschritts bezüglich der Positionen mit Kontakt isolierten Features sind in Form der ihnen jeweils entsprechenden Aminosäuren in (k) aufgeführt und farbig den Sequenzlogos angepasst.

Fänden sich an Position 48 jedoch β-verzweigte oder langkettige aliphatische Aminosäuren, würde dies den Tryptophanrest an Position 35 in der Orientierung blockieren, welche die Bindung an „Klasse *2*"-Liganden ermögliche [40]. Da der Tryptophanrest an Position 35 aber ja innerhalb sämtlicher Basissequenzen konserviert ist, dürfte die Isolation dieser Position hier während der Feature Selections eher artifizieller Natur sein und sich auf den Einsatz künstlicher Sequenzen zurückführen lassen (Kapitel 5.3). Auch die von Fernandez-Ballester, et al. (2004) [40] postulierte Signifikanz von Position 48 lässt sich anhand der Methoden der vorliegenden Arbeit – zumindest hinsichtlich des Differenzierungsschritts *„Klassen 1 vs. Klassen 2"* – nicht nachvollziehen. Für Position 13 jedoch lässt sich auch aus sequenzanalytischer Sicht ein Einfluss auf die Ligandenpräferenz vermuten, obwohl sich ihre Aminosäureeigenschaften zwischen Klasse *1* und Klasse *2* im Wesentlichen nur durch der Größe der Aminosäurereste voneinander unterscheiden. Ihr potentieller Einfluss wird jedoch ersichtlich, betrachtet man die einzelnen Subklassen von Klasse *1* und Klasse *2* separat. In der Beispieldomäne für Subklasse *1R* befindet sich Position 13 sowohl mit hydrophoben wie hydrophilen Bereichen des Argininrests an Position P-3 in Kontakt, während sie in Falle der Beispieldomäne für Subklasse *2R* eher mit dem hydrophilen Teil des Argininrests Kontakt hat. Auch in den Darstellungen für die Subklassen *1K* und *2K* interagiert sie eher mit den hydrophilen, basischen Bereichen des für die Liganden dieser Subklassen typischen Lysinrests an Position P-3. Ebenso im Falle der Beispieldomäne von Subklasse *1@* befand sich Position 13 mit Position P-3 des Liganden in Kontakt, der sich allerdings ja gerade dadurch auszeichnet, an dieser Position statt einer positiv geladenen eher aromatische Aminosäure zu besitzen. Einzige Ausnahme bildete die Beispieldomäne der Subklasse *2D*, in der Position 13 keinen Kontakt zum Liganden zeigte. Bei Analyse der jeweiligen subklassenspezifischen Aminosäureeigenschaften von Position 13 fällt auf, dass diese – abgesehen der Subklasse *2D* natürlich – subklassenspezifisch jeweils auch passend zu den Eigenschaften ihrer Kontaktbereiche innerhalb des Liganden sind. Mit anderen Worten, sie erklären subklassenspezifisch (zumindest partiell) jeweils auch die Affinität dieser Position zu ihrem Kontaktbereich. So sind die Aminosäureeigenschaften von Position 13 in Subklasse *1R* eher variabel, während in Subklasse *2R* an ihr ausschließlich polare Aminosäurereste vorkommen. Passend zum überwiegend an hydrophilen, basischen Bereichen des Lysinrests stattfindenden Kontakt dieser Position in den Beispieldomänen der Subklassen *1K* und *2K*, finden sich an ihr in diesen Subklassen eher polare und saure Aminosäurereste. Domänen der Subklasse *1@* hingegen besitzen an Position 13 in der Regel zwar auch polare, jedoch hinsichtlich ihrer Größe und ihres Säure-/Basencharakters ausschließlich kleinere respektive neutrale Aminosäurereste. Ähnliche Beobachtungen können auch an Position 33 gemacht werden, welche in den Darstellungen von Subklasse *1R* und *2R* jeweils mit der Backbone-Sequenz des jeweiligen Konsensmotivs Kontakt hat und in der Beispieldomäne von Subklasse *2D* mit dem Aspartat- sowie dem relativ am Rand der Bindungsstelle zum Liegen kommenden Tyrosinrest des für diese Subklasse typischen Konsensmotivs PxxDY interagiert. Passend hierzu finden sich in Domänen der Subklasse *2D* an Position 33

ausschließlich große, polare und basische Aminosäuren, während in Subklasse *1R* bzw. *2R* hier auch unpolare und vor allem eher neutrale Aminosäurereste vorkommen. Auch Position 52 könnte hinsichtlich dieses Differenzierungsschritts für der Präferenz des Ligandentypen von Bedeutung zu sein, denn bei Analyse der strukturellen Darstellungen hinsichtlich der Positionierung von „Klasse *1*"- bzw. „Klasse *2*"-Liganden an der Bindungsstelle fällt auf, dass die variablen Aminosäurereste „x" des Kernmotivs ´xPxxP` bei „Klasse *1*"-Liganden deutlich mehr in Richtung dieser Position geneigt sind. Entsprechend nachvollziehbar erscheint dann auch die Beobachtung, dass „Klasse *1*"-Domänen verglichen mit denen der Klasse *2* an dieser Position häufig kleinere Aminosäurereste vorweisen.
Konkretisiert man die beschriebenen Beobachtungen jedoch auf den Einzelfall, wird ersichtlich dass die gerade beschriebenen und hinsichtlich der Differenzierung *„Klassen 1 vs. Klassen 2"* für signifikant erachteten Positionen alleine nicht ausreichen dürften, um die Affinität einer Domäne zu Liganden der Klasse *1* bzw. Klasse *2* zu determinieren. Dies soll im Folgenden anhand je einer Domäne der Klasse *1* bzw. Klasse *2* exemplarisch verdeutlicht werden. So zeigt die SH3-Domäne der bereits in Kapitel 1.5 hinsichtlich ihrer Funktion besprochenen Tyrosinkinase Fyn (Subklasse *1R* [17]) an diesen Positionen einen Threonin- (Position 13), Glycin- (Position 33) und Asparaginrest (Position 52). Diese Kombination aus Aminosäuren dürfte zwar eine Bindung an Liganden der Subklassen *1K*, *2K* und *2D* eher unwahrscheinlich machen, erklärt aber letztlich nicht die Präferenz der Domäne für Liganden der Klasse *1*, da sie keinen Anhalt dafür bietet, dass zu Liganden der Subklasse *2R* eine geringere Affinität besteht als zu Liganden der Subklassen *1R* bzw. *1@*. Ähnlich verhält es sich bei der C-terminalen SH3-Domäne des Adapterproteins Grb2 (Subklasse *2R* [17]), welches wie Fyn [73][83][137] ebenso an diversen Signaltransduktionswegen, unter anderem der Aktivierung des G-Proteins Ras durch Bindung an Sos-1 (Kapitel 1.2), beteiligt ist [50]. So finden sich hier an den besagten Positionen ein Glutaminsäure- (Position 13), Prolin- (Position 33) und Asparaginrest (Position 52). Auch diese Aminosäurekombination dürfte zwar die Bindung an Liganden bestimmter Subklassen eher unwahrscheinlich machen (Subklassen *1@* und *2D*), kann aber ebenso wenig erklären, weshalb diese Domäne gerade Liganden der Klasse *2* präferiert, da die Analyse dieser Kombination keinen Hinweis auf unterschiedliche Affinitäten der Domäne zu Liganden der Subklassen *1R* bzw. *1K* verglichen mit Liganden der Subklassen *2R* bzw. *2K* liefert.
Es lässt sich also festhalten, dass von den insgesamt sieben hinsichtlich dieses Differenzierungsschritts isolierten Positionen drei (Positionen 13, 33 und 52) zumindest partiell Einfluss auf die Präferenz einer Domäne bezüglich „Klasse *1*"- bzw. „Klasse *2*"-Liganden haben dürften, allerdings wohl nicht (auch in Kombination) das entscheidende Kriterium hierfür sind. Da auch der von Fernandez-Ballester, et al. (2004) vorgestellte „Tryptophan Switch" [40] die Präferenz determinierenden Mechanismen nicht ganz vollständig erklären kann – wenngleich ihm bei dieser Fragestellung sicher eine zentrale Rolle zukommen dürfte, gilt es also auch in Zukunft weiter nach den entscheidenden Faktoren hierfür zu suchen. So wäre doch gerade hinsichtlich dieses Differenzierungsschritts ein exaktes Verständnis dieser

Faktoren wünschenswert, da die Orientierung des Liganden während der Bindung wohl zu den fundamentalsten Unterscheidungsmerkmalen zwischen den einzelnen Klassen zählen und folglich auch einen entsprechend hohen Beitrag zur Bindungsspezifität der Domänen leisten dürfte. Dass diese wiederum für eine präzise Koordination der unterschiedlichen, von SH3-Domänen abhängigen Vorgänge innerhalb der Zelle unabdingbar sein dürfte, wird schnell ersichtlich, wenn man bedenkt, dass sich viele SH3-Domänen tragende Proteine in Kompartimenten mit multiplen potentiellen SH3-Bindungspartnern befinden oder an Prozessen teilnehmen, an denen auch andere (funktionell unterschiedliche) SH3-Domänen tragende Proteine beteiligt sind [84][20][130]. So nehmen beispielsweise sowohl Fyn wie auch Grb2 an der Signalweiterleitung nach T-Zell-Rezeptor Stimulation teil und finden sich dabei beide in der Umgebung des Rezeptors [1][51]. Da die SH3-Domäne von Fyn aber einer „Klasse *1*"- und die von Grb2 „Klasse *2*"-Domänen [134] entsprechen, ist gewährleistet, dass beide Proteine auch tatsächlich eher ihre Soll-Liganden und nicht die des jeweils anderen binden. Ein anderes Beispiel hierfür wären die Proteine Eps8 (Klasse *2D* [17]) und Grb2, von denen hinsichtlich beider bekannt ist, dass es sich um zytoplasmatische Proteine handelt, die mit Sos-1 interagieren [84][91]. Während Grb2 jedoch mit beiden seiner SH3-Domänen, welche beide der Klasse *2R* zugehörig sein dürften [134], direkt an Sos-1 bindet [95], so bedarf es im Falle von Eps8 noch des „Adapters" Abi-1 im Sinne einer Komplexbildung [130]. Dabei bindet Abi-1 über die seinige SH3-Domäne an Sos-1 [63], während an Eps8 über dessen SH3-Domäne [130]. Die unterschiedliche Bindungsspezifität der SH3-Domänen von Eps8 und Grb2 gewährleistet also eine unterschiedliche Koordination der Bindung dieser Proteine an Sos-1 und damit konsekutiv auch der hierdurch ausgelösten Effekte.

Bezüglich der Evaluation, inwieweit auch die isolierten Positionen mit Kontakt aus den „RFE"-Gruppen der übrigen Differenzierungsschritte Einfluss auf die Ligandenpräferenz hinsichtlich ihres betreffenden Differenzierungsschritts haben könnten, sei an dieser Stelle auf Kapitel 5.4 der Diskussion verwiesen.

5 Diskussion

5.1 Grundidee

Ziel dieser Arbeit war es zu erforschen, inwieweit mit Support-Vector-Machines, denen ausschließlich Primärstrukturdaten der SH3-Domäne als Informationsquelle zur Verfügung stehen, valide Vorhersagen über die Bindungsspezifität bzw. Klassenzugehörigkeit dieser Domänen nach dem Modell von Cesareni, et al. (2002) [17] möglich sind und die eigenen Resultate im Lichte der gegenwärtigen Literatur (z.B. MIEC-SVMs [57][58][59]; Kapitel 5.4) einzuordnen. Mit anderen Worten, es sollte untersucht werden, ob sich SH3-Domänen mit dieser Methode zuverlässig ihrer jeweiligen Ligandenklasse zuordnen lassen, um so die Menge potentieller Bindungspartner einer SH3-Domäne besser eingrenzen zu können, was unter anderem bei der Entschlüsselung des Interaktoms ihres jeweiligen Trägerproteins von Nutzen sein und im besten Falle sogar Rückschlüsse auf seine Funktion in vivo erlauben könnte. Zum Zeitpunkt der Erstellung des Modells von Cesareni, et al. (2002) [17] stand jedoch nur eine eher kleine Menge an Daten bezüglich der Ligandenpräferenz von SH3-Domänen zur Verfügung. Daher ist es durchaus vorstellbar, dass die innerhalb des Modells verwendete Differenzierung der Domänen nach der Sequenz ihrer Liganden in acht verschiedene Klassen nicht endgültig ist, sondern in Zukunft stets noch modifiziert und erweitert werden muss, um sie jeweils aktuellen Erkenntnissen anpassen zu können. Dies dürfte insbesondere die Klasse *X_ORS* betreffen, in welcher die Domänen zusammengefasst wurden, deren Liganden keiner der konventionellen Klassen zugeordnet werden konnten bzw. zusätzlich sehr spezifische Merkmale aufwiesen. Bei Vorliegen größerer Datenmengen wäre es speziell hinsichtlich dieser Klasse vorstellbar, dass sich bezüglich einiger Liganden dieser Klasse weitere Domänen finden lassen, deren Liganden ähnliche Eigenschaften aufweisen, sodass sie in neuen, eigenständigen Klassen zusammengefasst werden müssten. Eine weitere Relativierung der Präzision des Modells könnte die Methodik darstellen, mit welcher die Ligandenpräferenz der einzelnen Domänen erforscht und definiert wurde. Die Untersuchungen hierzu fanden in den meisten Fällen in vitro durch Konfrontation der Domänen mit kurzen, an Bakteriophagen dargestellten Peptiden statt. Anhand der Konsenssequenzen der gebundenen Peptide erfolgte dann die Definition der jeweiligen Klassenzugehörigkeiten [17]. Da diese Experimente jedoch zumeist nur in vitro und mithilfe kurzer Peptide statt gesamter Proteine natürlicher Bindungspartner erfolgten, sind Aussagen über das tatsächliche Bindungsverhalten der Domänen in ihrem natürlichen Umfeld (in vivo) nur eingeschränkt möglich. Beispielsweise ist es durchaus vorstellbar, dass die Wahl eines Liganden in vivo aufgrund in diesen Experimenten unbeachteten, strukturellen Eigenschaften der Bindungspartner bzw. anderer, bei der Bindung eine Rolle spielender, dort unbeachteter Faktoren gänzlich anders ausfiele.

Unabhängig davon, wie ausgereift das zugrunde liegende Klassenmodell auch sein mag, konnte aber dennoch gezeigt werden, dass zumindest seinen Vorgaben entsprechend mithilfe von Support-Vector-Machines valide Vorhersagen über die Klassenzugehörigkeit bzw. Bindungsspezifität von SH3-

Domänen unter alleiniger Berücksichtigung ihrer Aminosäuresequenzen möglich sind. Daher ist es durchaus vorstellbar, dass dies auch hinsichtlich anderer Proteininteraktionsdomänen möglich sein könnte. Allerdings gilt es zu bedenken, dass die Angaben von Cesareni, et al. (2002) [17] hinsichtlich mancher Domänen (z.B. hinsichtlich der des Proteins Lyn_H) nicht ganz eindeutig schienen und dass zudem in einigen Fällen ein und dieselbe Domäne – aufgrund zwischen den Arbeiten von Cesareni, et al. (2002) [17] und Tong, et al. (2002) [141] hinsichtlich identischer Proteine teilweise voneinander abweichender Gen- bzw. Protein-Identifikatoren, wie beispielsweise im Falle des Proteins Yjl020c bzw. Bbc1 – verschiedenen Klassen bzw. einer Klasse mehrfach zugeteilt wurden. Da jedoch nur wenige Domänen hiervon betroffen waren, dürfte der Einfluss dieser Problematiken auf das Gesamtresultat eher gering sein. Ein anderer Faktor mit potentiellem Einfluss auf die Klassifikationsergebnisse liegt darin begründet, dass während des Speicherns der Basissequenzen bei einem Teil aus ungeklärter Ursache einige Aminosäuren jeweils vom N- bzw. C-Terminus verloren gingen. Hierdurch wurden die betreffenden Sequenzen möglicherweise zum Teil in anderer Form aligniert, wodurch einzelne Positionen an anderer Stelle im Alignment zum Liegen kamen. Dies könnte im ungünstigsten Fall wiederum dazu geführt haben, dass die Klassifikatoren fälschliche Gemeinsamkeiten/Differenzen zwischen den Sequenzen bezüglich solcher Positionen als Differenzierungskriterium heranzogen, wodurch sowohl die Klassifikationsergebnisse als auch die Identifikation signifikanter Positionen verfälscht würden. Allerdings fehlten in den betreffenden Sequenzen zuallermeist nur sehr wenige Aminosäuren und zudem nur vom (hauptsächlich C-terminalen) Ende der jeweiligen Sequenzen, sodass die Alignments größtenteils korrekt waren und der resultierende Fehler eher gering sein dürfte.
Eine Problematik ganz anderer Art lag in der zumindest hier nicht lösbar erscheinenden Schwierigkeit Kreuzreaktivität in den Algorithmen der Klassifikatoren zu berücksichtigen, da Support-Vector-Machines nicht in der Lage sind ein und dieselbe Sequenz mehreren Klassen zuzuteilen. In der Natur ist Kreuzreaktivität jedoch ein häufig zu beobachtendes Phänomen (Kapitel 4.1.3), sodass auf den Einsatz laborexperimenteller Methoden, auch bei noch so guten Ergebnissen anhand des vorgestellten Klassifikators, wohl nie ganz verzichtet werden kann.

5.2 Aussagekraft der erstellten Klassifikatoren

5.2.1 Anzahl der benutzten Sequenzen, emittierte Sequenzen und Overfitting

Ein zentrales Problem, das bei dieser Arbeit stets im Vordergrund stand, war das ungünstige Verhältnis zwischen der sehr hoher Dimensionalität der Daten (1160 Features pro Sequenz) auf der einen und der zur Verfügung stehenden Menge an Daten (51 Sequenzen) auf der anderen Seite. Es galt also das hieraus resultierende overfitting entweder durch Reduktion der Datendimensionalität bzw. durch Erhöhung der Datenmenge zu vermindern. Eine Feature Selection, anhand derer sich die Dimensionalität der Daten auf ihre jeweils wichtigsten Features reduzieren ließe, scheint hierfür zunächst ideal, jedoch war auch

die Menge verfügbarer Daten hinsichtlich der einzelnen Klassen unterschiedlich groß, sodass nicht nur das Problem des overfitting generell gelöst werden musste, sondern hierbei auch noch das verhältnismäßig größere overfitting hinsichtlich kleinerer Klassen zu beachten war. Dieses würde sich zwar unter Verwendung einer Feature Selection auch etwas vermindern, denn die engeren Klassengrenzen kleinerer Klassen beruhen ja auf der hier im Vergleich mit größeren Klassen insgesamt kleineren Varianz der einzelnen Features zueinander. Je mehr Features also während einer Feature Selection entfernt werden, umso mehr gleichen sich die Klassengrenzen einander an. Der in der vorliegenden Arbeit favorisierte Ansatz dem Problem durch Anhebung der Datenmenge mithilfe künstlich emittierter Sequenzen zu begegnen birgt jedoch den Vorteil, dass sich hierdurch nicht nur das generelle overfitting ebenso effektiv vermindern lassen sollte, sondern auch das verhältnismäßig größere overfitting hinsichtlich kleinerer Klassen durch entsprechend unterschiedliche Augmentation der einzelnen Klassen mit zusätzlichen Daten direkt angegangen werden konnte. Zudem sollten sich durch Erhöhung der Datenmenge mit künstlichen Sequenzen die tatsächlich signifikanten Unterschiede zwischen den einzelnen Klassen besser hervorheben lassen, wodurch auch eine etwaig angeschlossene Feature Selection effektiver werden dürfte. Allerdings ist es vorstellbar, dass auch ein Angleichen der Datenmengen der einzelnen Klassen mit künstlichen Sequenzen nicht unbedingt eine vollständige Nivellierung Klassengrenzen zur Folge hat, denn die Varianz der Basissequenzen einer kleinen Klasse bezüglich ihrer Features ist deutlich geringer als die größerer Klassen, sodass auch die Varianz zwischen den Features künstlicher Sequenzen – die ja jeweils nach dem Vorbild der Basissequenzen einer Klasse emittiert wurden – bei kleineren Klassen geringer sein dürfte. Eine weitere Problematik, die bei der Emission klassenspezifischer künstlicher Sequenzen zu beachten ist, liegt in der Natur dieser Sequenzen selbst begründet: da sie klassenspezifisch eine Varianz der Aminosäuren ähnlich der des natürlichen Vorbilds an ihren Positionen widerspiegeln sollen, nimmt unter ihrer Anwendung natürlich auch der Konservierungsgrad der Positionen in den einzelnen Klassen entsprechend ab. Hierdurch traten zum Teil jedoch auch an Positionen Aminosäurevariationen bzw. einzelnen Features Variationen ihrer Werte auf, welche bezüglich der Basissequenzen der zu differenzierenden Klassen hochkonserviert waren. Zwar sollten künstliche Variationen mit dem biologischen Vorbild ähnlichen Aminosäuren bzw. Featurewerten erfolgt sein und sich dementsprechend an diesen Positionen bzw. Features bezüglich der betreffenden Klassen ähneln. Aufgrund der jeweils klassenspezifischen Emission der künstlichen Sequenzen fanden sich hier aber teilweise sogar klassenspezifische Differenzen zwischen den zu differenzierenden Klassen, wodurch sie für die Klassifikatoren zur Differenzierung teils sogar signifikant wurden (Kapitel 5.3). Da in den Basissequenzen an diesen Stellen aber keine Variationen auftraten, dürfte die klassenspezifische Verteilung dieser Variationen jedoch eher zufällig erfolgt und ihre Spezifität für eine bestimmte Klasse damit auch nicht repräsentativ sein. Insgesamt fanden sich allerdings nur wenige Fälle, in denen Positionen bzw. Features auf diese Weise Signifikanz zur

Differenzierung erlangten und so – hinsichtlich des natürlichen Vorbilds eher ungerechtfertigt – zu einer Verbesserung der Klassifikationsgüte beitrugen.
Daher dürfte die doch in vielen Fällen beobachtete, deutliche Steigerung der Klassifikationsgüte unter Anwendung künstlicher Sequenzen auch die Hypothese stützen, dass sich durch ihren Einsatz das overfitting erheblich reduzieren ließ. Dass sich auch das verhältnismäßig größere overfitting hinsichtlich kleinerer Klassen durch ihre Anwendung vielfach effektiv vermindern ließ, zeigt unter anderem die hierbei häufig zu beobachtende deutliche Zunahme der Relevanz größerer Klassen. Trotzdem muss bedacht werden, dass die zur Verfügung gestandene Ausgangsdatenmenge teils nur sehr klein war (im extremsten Fall nur drei Domänen pro Klasse), weshalb die hier erzielten Resultate in Zukunft sicher noch mit einer größeren Zahl an klassifizierten Domänen zu validieren sind. Einerseits wäre hierbei zu evaluieren, inwieweit die genutzten Domänen für ihre jeweiligen Klassen tatsächlich ausreichend repräsentativ waren, und andererseits, ob durch die verwendeten künstlichen Sequenzen die jeweiligen biologischen Klassengrenzen realistisch nachempfunden wurden. Da allerdings bereits unter dieser limitierten Menge an Daten sehr ermutigende Ergebnisse erzielt werden konnten, scheint es vorstellbar, dass sich mit weiter zunehmender Menge an zur Verfügung stehenden klassifizierten Domänen auch die Klassifikationsergebnisse weiter präzisieren lassen könnten.

5.2.2 Aufbau der Klassifikatoren

Um die Klassifikationspräzision weiter zu verbessern, spielte neben dem Einsatz künstlicher Sequenzen auch die Einführung eines arbiträren Klassifikationsmodells eine Rolle, da dies die Möglichkeit eröffnete Informationen über eine vordefinierte Taxonomie des Klassensystems in den Klassifikationsprozess mit einzubringen. Dass dies von Vorteil sein dürfte, konnte bereits in einer Arbeit von Binder, et al. an SVMs zur Klassifikation von Bildern aus dem Jahr 2009 demonstriert werden, deren Klassifikationsmodell hiervon deutlich profitierte und einer reinen Multiclass-Klassifikation – welche taxonomische Informationen nicht berücksichtigt – überlegen war [12]. Ein weiterer Vorteil des arbiträren Klassifikationsmodells lag darin, dass hierbei hinsichtlich jedes Klassifikationsschritts eine separate Anpassung des jeweiligen Subklassifikators bezüglich seiner Regularisierungsstufe und Kernel-Funktion erfolgen konnte, was im Falle der ***n(n-1)/2*** Subklassifikatoren eines reinen OvO-Multiclass-Klassifikators nicht möglich ist. Allerdings bereitete die Definition der Taxonomie, welche sich im vorliegenden Fall am Verwandtschaftsgrad der Klassen hinsichtlich der Konsensmotive ihrer Liganden orientieren sollte, einige Schwierigkeiten. So ist die Aufteilung der Klasse *Nicht-Y* in die beiden Taxa *X_ORS* und *Nicht-X_ORS* sicher nicht ganz unproblematisch, da einige Domänen der Klasse *X_ORS* Liganden banden, deren Konsensmotive prinzipiell schon einer der Subklassen von Klasse *Nicht-X_ORS* zugeordnet werden könnten, wenn man von ihren jeweils zusätzlichen Merkmalen absieht. Allerdings boten die Liganden der Klasse *X_ORS* vielfach auch völlig atypische Konsensmotive, sodass der beschriebene Differenzierungsschritt gerechtfertigt schien. Ein weiteres

Problem bereitete die Einordnung von Klasse *2D* innerhalb der Taxonomie, denn auch deren Konsenssequenz PxxDY ist vergleichsweise atypisch. Zudem war letztlich nicht bekannt, ob Liganden dieser Klasse tatsächlich eine PPII-Helix bilden und in Typ-II-Orientierung binden. Da diese Klasse von Cesareni, et al. (2002) [17] – an dessen Klassenmodell sich die vorliegende Arbeit ja orientierte – aber dennoch der übergeordneten Klasse *2* zugeteilt wurde, wurde entschieden, dies auch hier zu übernehmen.

Abgesehen dieser Schwierigkeiten barg der arbiträre Klassifikationsansatz aber auch noch ein ganz anderes Problem: bei Arbeiten mit SVM-basierten Klassifikatoren sollten die Klassengrößen im Idealfall jeweils nivelliert sein, da die Größe einer Klasse Einfluss auf die Weite ihre Klassengrenzen hat. Mit anderen Worten, je größer eine Klasse ist, umso weiter sind auch ihre Grenzen. Im Falle ungleicher Klassengrößen wäre der Klassifikator also stets geneigt die zu klassifizierenden Objekte eher der größten Klasse zuzuordnen [148]. Innerhalb des arbiträren Klassifikationsmodells setzten sich die Klassen eines Subklassifikationsschritts jedoch häufig aus unterschiedlich vielen Subklassen zusammen, was entsprechend häufig auch unterschiedliche Klassengrößen zur Folge hatte. Dennoch musste von einer Nivellierung derselben nicht nur bei den Subklassifikatoren mit Regularisierungsvariante **R0**, sondern in den meisten Fällen auch bei denen mit Variante **R20** bzw. **R100** abgesehen werden*. Dies lässt sich damit begründen, dass eine Modifikation der Klassengrößen auch eine Veränderung des Informationsgehalts der Klassen mit sich gebracht hätte, wodurch hierunter gewonnene Daten bzw. Resultate nicht mehr mit denen der reinen OvO-Multiclass-Klassifikatoren aus Testreihe 1 und 2 vergleichbar gewesen wären. Damit dürfte sich auch erklären, weshalb beispielsweise sämtliche Subklassifikatorvarianten des Differenzierungsschritts „*Y vs. Nicht-Y*“ der reinen OvO-Multiclass-Differenzierung hinsichtlich der Abgrenzung von Klasse *Y* unterlegen waren. Aber auch die Präzision der den beiden OvO-Multiclass-Subklassifikationen *„1R vs. 1K vs. 1@“* bzw. *„2R vs. 2K vs. 2D“* nachgeschalteten, binären Subklassifikatoren könnte unter dieser Problematik gelitten haben. Zwar sollte hier die jeweilige Klasse *„Nicht-…“* sogar eine höhere Heterogenität ihrer Features – und damit letztlich auch weitere Klassengrenzen – als die jeweils von ihr zu differenzierende Subklasse besitzen. Allerdings sollte diese höhere Heterogenität möglichst nur auf den unterschiedlichen Eigenschaften der beiden jeweils zu Klasse *„Nicht-…“* zusammengeschlossenen Subklassen beruhen und nicht auf einer größeren Menge an Trainingsobjekten. Möglicherweise erklärt dies zumindest teilweise, weshalb bei diesen Subklassifikatoren im Vergleich mit dem entsprechenden OvO-Multiclass-Subklassifikator neben (erhofften) Relevanzverbesserungen vielfach auch erhebliche Sensitivitätsverluste hinsichtlich der jeweils von Klasse *„Nicht-…“* abzugrenzenden Subklasse beobachtet werden konnten, sodass letztlich nur einer dieser Subklassifikatoren für eine Integration seiner selbst in den Gesamtklassifikator in Frage kam. Inwieweit auch die Präzision der Subklassifikation *„X_ORS vs. Nicht-X_ORS“* von dieser

* Auf einen Einsatz der Zusatzoption *class.weights* der Funktion *svm* aus dem Softwarepaket e1071 [31][97] als Alternativlösung des Problems wurde abgesehen, da dies innerhalb des Softwarepakets RFE, welches unter anderem zur Analyse der Klassifikatoren verwendet wurde, keine wählbare Zusatzoption ist.

Problematik betroffen war, lässt sich aufgrund fehlender Vergleichsmöglichkeit nur schwer beurteilen. Ein anderes Phänomen, das sich zumindest partiell auf die Problematik ungleicher Klassengrößen zurückführen lassen könnte, liegt in den häufig beobachteten besseren Resultaten der Subklassifikatoren unter eher niedriger Regularisierung, da das Verhältnis der Klassengrößen hier vielfach ausgewogener war.

Ungeachtet dessen konnte aber dennoch gezeigt werden, dass sich trotz der geschilderten Probleme mithilfe des vorgestellten arbiträren Klassifikationsmodells erheblich bessere Ergebnisse erzielen ließen als unter reiner OvO-Multiclass-Klassifikation, wobei letztlich nicht sicher beurteilbar ist, welcher der beiden beschriebenen Vorteile dieses Modells den größeren Einfluss auf seine deutliche Überlegenheit hatte. Dies bedenkend, indizieren diese Resultate umso mehr das hohe Potential, das hier noch auszuschöpfen ist, sodass in zukünftigen Arbeiten die Anwendung eines arbiträren Klassifikationsmodells mit nivellierten Klassengrößen weiterverfolgt werden sollte. Hierbei wäre auch eine weitere Verzweigung der Taxonomie vorstellbar, da einerseits den Subklassen *1R* und *1K* bzw. *2R* und *2K* hinsichtlich ihrer Konsensmotive jeweils eine noch größere Verwandtschaft miteinander attestiert werden könnte als mit Subklasse *1@* respektive *2D* und andererseits auch die binären Subklassifikatoren *„1@ vs. Nicht-1@“* und *„2D vs. Nicht-2D“* – trotz unterschiedlicher Klassengrößen – zumindest gleich gute Resultate wie die entsprechenden OvO-Multiclass-Subklassifikatoren lieferten.

Aber auch eine gänzlich andere Art von Taxonomie scheint hinsichtlich des vorliegenden Klassifikationsproblems reizvoll. So wurde in einer Arbeit von Madzarov G., et al. (2009) ein Ansatz (SVM-BDT) vorgestellt, der statt einer vordefinierten Taxonomie einen binären Entscheidungsbaum nutzt, welcher anhand eines Clustering-Algorithmus mit Distanzmessungen zwischen den Klassen im Kernel-Space erstellt wird. Die Studie zeigte, dass hiermit nicht nur teils bessere Resultate als mit anderen Prinzipien zur Lösung eines Multiclass-Problems erzielt werden konnten, sondern vor allem auch schnellere Trainings- und Testzeiten [90].

Ein anderer, vielversprechender Ansatz zur Verbesserung der Klassifikationspräzision war die Anwendung von Feature Selections anhand des Softwarepakets RFE [54][161]. Bereits die Anwendung auf die reinen OvO-Multiclass-Klassifikatoren mit Regularisierungsstufe **R0** erbrachte bezüglich beider eine enorme Verbesserung und hinsichtlich der linearen Variante sogar eine Gesamttrefferquote, welche der des präzisesten Klassifikators nur in geringem Maße nachstand – und das unter Nutzung von lediglich acht der 1160 Features (es gilt jedoch zu beachten, dass die hier verglichenen Ergebnisse unter verschiedenen Kreuzvalidierungsvarianten erzielt wurden). Diese Ergebnisse wirken noch eindrucksvoller, führt man sich die Tatsache vor Augen, dass die Reduktion der Features in beiden Fällen partiell logarithmisch erfolgte, sodass die hierbei genutzten Feature-Mengen möglicherweise gar nicht den tatsächlich optimalen entsprachen. Eine reine Leave-One-Out Strategie zur Reduktion der Features hätte also vielleicht sogar noch bessere Ergebnisse erbracht. Allerdings erschien die

Implementation von RFE [54][161] mit dem Ziel einer Klassifikationsverbesserung in andere hier erstellte Klassifikationsmodelle aus mehreren Gründen nicht sinnvoll: Die Anwendung auf Modelle mit künstlichen Sequenzen gestaltete sich in diesem Rahmen nicht als sinnvoll, da es hier für eine exakte Validierung nötig wäre, hinsichtlich der Funktion *rfe.cv* während der zehnfachen Kreuzvalidierungen streng darauf zu achten, in den einzelnen Trainingsmengen keine künstlichen Sequenzen zu verwenden, die auch am Vorbild einer Sequenz der entsprechenden Testmenge emittiert wurden. Das bedeutet, dass entsprechend der jeweiligen Sequenzen jeder Testmenge ein anderer Trainingssatz an künstlichen Sequenzen verwendet werden müsste; zudem dürften künstliche Sequenzen auch nicht als Testsequenzen fungieren. Diese Voraussetzungen waren jedoch aufgrund der limitierten Möglichkeiten des Softwarepakets RFE [54][161] nicht zu schaffen. Die Integration in das arbiträre Klassifikationsmodell (auch ohne Einbeziehung künstlicher Sequenzen) schien hingegen aufgrund der schlechten Resultate von Testreihe 6 (Kapitel 4.1.2.3) nicht sinnvoll zu sein. Diese schlechten Resultate lassen sich möglicherweise zum Teil dadurch erklären, dass durch die Kombination einer Leave-One-Out nach mit einer zehnfachen Kreuzvalidierung während der Feature Selection nicht mehr genügend Sequenzen in den einzelnen Teilmengen während der Feature Selection zur Verfügung standen, um valide Ergebnisse zu erzielen. So wurden die Basissequenzen während der zehnfachen Kreuzvalidierung ja auf mehrere Teilmengen aufgeteilt, sodass jedes hierbei erstellte Modell auch nur an einem Teil der ohnehin schon geringen Menge an Basissequenzen trainiert und getestet werden konnte. Für die Leave-One-Out Kreuzvalidierung wurde nun zusätzlich im Vorfeld jeweils noch eine Sequenz entfernt. Dementsprechend standen für die zehnfache Kreuzvalidierung während der Feature Selection jetzt noch weniger Sequenzen zur Verfügung, speziell bezüglich der Klasse, der die zuvor entfernte Sequenz entstammte. Unter diesen Umständen dürfte natürlich auch eine valide Feature Selection, vor allem in Bezug auf die Differenzierbarkeit dieser Klasse nun schwerer möglich sein. Zudem dürfte die Auswirkung dieser Problematik dadurch noch verstärkt werden, dass ja die während der folgenden Leave-One-Out Kreuzvalidierung getestete Sequenz jeweils ebendieser Klasse entstammte. Dies könnte sich auch in der deutlich höheren Anzahl an durchschnittlich benötigten Features widerspiegeln. Wurden zur optimalen Klassifikation während der Feature Selection von Testreihe 5 (Kapitel 4.1.2.3) nur acht Features benötigt, so waren es hier durchschnittlich 21,88 bzw. 33,14 Features. Eine andere Ursache für die schlechten Resultate könnte darin liegen, dass sich eine unter *rfe.cv* bezüglich der Modelle der einzelnen Trainingsmengen ermittelte optimale Feature-Menge möglicherweise nicht ohne Weiteres auf die unter *rfe.ae* für das Gesamtmodell erstellte Beurteilung der Features ihrer Signifikanz nach übertragen lässt, weshalb von solchen Versuchen an anderer Stelle auch Abstand genommen wurde. Hinsichtlich zukünftiger Analysen wäre es also von besonderem Interesse ausgefeilte Techniken zur Modifikation dieses Softwarepakets zu entwickeln, welche seine Implementation in sowohl arbiträre Klassifikationsmodelle wie auch Modelle mit künstlichen Sequenzen besser ermöglichen. Zudem wäre auch die Überarbeitung der Option zur Wahl der Kreuzvalidierungsvariante wünschenswert.

Andere Ausgangspunkte zur weiteren Verbesserung der Klassifikationspräzision lägen hinsichtlich zukünftiger Arbeiten unter anderem auch in einer präzisen Optimierung der verschiedenen Zusatzparameter der Funktion *svm*, der Analyse einer Einbindung anderer Kernel-Funktionen sowie in einer weiteren Verfeinerung der Prinzipien zur Lösung des Multiclass-Problems.

5.2.3 Beurteilung der Klassifikatoren

Die Beurteilung der Klassifikationsmodelle hinsichtlich ihrer Reliabilität war in vielen Fällen nicht ganz leicht, da zur Interpretation ihrer Ergebnisse stets auch ihr Aufbau bzw. ihre jeweilige Klassifikationsmethodik bedacht werden mussten. So wäre beispielsweise die Beurteilung der Klassifikatoren mit nicht nivellierten Klassengrößen rein anhand ihrer während der Kreuzvalidierung erzielten Gesamttrefferquote sicher nicht ausreichend gewesen. Dies liegt in zwei hierbei zusammenspielenden Problemen begründet: Zum einen entspricht bei Kreuzvalidierungen die Menge an Testobjekten jeder Klasse exakt ihrer jeweiligen Klassengröße. Je größer eine Klasse also ist, umso größer ist auch ihre Menge an Testobjekten. Zum anderen ist ein SVM-basierter Klassifikator stets geneigt Objekte der größten Klasse zuzuordnen [148], weshalb Objekte dieser Klassen auch entsprechend zuverlässiger erkannt werden dürften. Das bedeutet, im ungünstigsten Falle würde so bei ausschließlicher Beachtung der Gesamttrefferquote nicht nur die Klassifikationsgüte hinsichtlich der größten Klassen überschätzt, sondern auch die des Klassifikators insgesamt, da die große Menge an Testobjekten großer Klassen entsprechend stärker ins Gewicht fallen. Einen Ausweg hierfür böte die Nutzung der durchschnittlichen klassenspezifischen Sensitivität, also eine gewichtete Gesamttrefferquote, bei welcher die Menge an Testobjekten jeder Klasse Berücksichtigung findet. Allerdings ist auch dieser Ansatz nicht gänzlich unproblematisch, da hierbei den geringen Mengen an Testobjekten kleinerer Klassen proportional ein enormes Gewicht zukommt. Nicht nur, dass Ausreißern kleinerer Klassen dadurch ein proportional zu großer Einfluss auf das Gesamtergebnis zukäme, auch würde der Klassifikator hierdurch wahrscheinlich ungerechtfertigt abgewertet, da die durchschnittliche klassenspezifische Sensitivität den Einfluss der Klassengröße auf die Differenzierbarkeit einer Klasse nicht berücksichtigt. Mit anderen Worten, da die Klassifikationsgüte kleinerer Klassen aufgrund geringerer Anzahl an Trainingsobjekten naturgemäß schlechter sein dürfte als die größerer Klassen, sollten für eine valide Beurteilung der Reliabilität eines gegebenen Klassifikators möglichst auch die Klassen stärker ins Gewicht fallen, für die anhand einer suffizienten Menge an Objekten auch ein ausreichendes Training möglich war. Bei Anwendung der durchschnittlichen klassenspezifischen Sensitivität wird dies jedoch gezielt unterdrückt. Nachteilig bezüglich beider Methoden ist, dass sich unter ihrer Anwendung zwar die Sensitivität eines Klassifikators auf unterschiedliche Weisen beurteilen lässt, die Relevanzwerte der einzelnen Klassen allerdings keine Berücksichtigung finden. Gerade diese sind aber zur Einschätzung, wie reliabel die Zuordnung eines Objekts zu einer bestimmten Klasse ist, besonders wichtig, da sie den Anteil korrekt einer Klasse zugeordneter Objekte an der Gesamtheit aller

dieser Klasse zugeordneten Objekte wiedergeben. Hier bietet die Berechnung des durchschnittlichen F-Maßes aller Klassen eine mögliche Abhilfe, welches hinsichtlich jeder Klasse Sensitivität und Relevanz in Form ihres harmonischen Mittels miteinander kombiniert [122]. Da jedoch auch dieses dem Durchschnitt aller Klassen entspricht, gelten für seine Interpretation dieselben Restriktionen wie für die durchschnittliche klassenspezifische Sensitivität. Es wird ersichtlich, dass zu einer validen Beurteilung der im Rahmen dieser Arbeit erstellten Klassifikatoren wohl keine der gängigen Interpretationsmethoden alleine – im Sinne eines Goldstandards – suffizient gewesen sein dürfte. Sie war daher stets Resultat einer kombinierten Evaluation der Ergebnisse anhand verschiedener dieser Methoden. Daraus folgt natürlich, dass die hier aus der Beurteilung der Klassifikatoren gezogenen Schlüsse im Einzelfall sicher strittig betrachtet werden können und daher Grund zu weiterer Diskussion in zukünftigen Projekten bieten dürften. Insbesondere kritisch dürfte aber die durch RFE [54][161] verwendete Methode zur Beurteilung von Klassifikatoren zu bewerten sein, welche ausschließlich anhand der Gesamterrorquoten (≙ 1- Gesamttrefferquote) erfolgt, da dies im Falle asymmetrischer Klassengrößen zu einer fehlerhaften Einschätzung der optimalen Feature-Menge führen kann: RFE [54][161] führt nach jedem Reduktionsschritt eine Testklassifikation des Klassifikators durch und beurteilt dessen Klassifikationsgüte. Liegen hierbei asymmetrische Klassengrößen vor, werden unter Umständen statt der Feature-Mengen, unter denen eine besonders reliable Differenzierung der Klassen gelingt, diejenigen als ideal gewertet, unter denen besonders viele Objekte den größten Klassen zugeteilt werden. Das erklärt sich damit, dass ja auch die meisten Testobjekte diesen Klassen entstammen und die Gesamterrorquote dementsprechend besser werden dürfte, wenn gerade besonders viele dieser Objekte korrekt klassifiziert werden. So kann es sein, dass die Anwendung von RFE [54][161] in einigen Fällen sogar zu einer tatsächlichen Verschlechterung der Klassifikationsgüte führt, obwohl die Gesamterrorquote zunächst das Gegenteil suggeriert.
Hinsichtlich der Anwendung von RFE [54][161] in Testreihe 5 (Kapitel 4.1.2.3) schien dies jedoch nicht der Fall gewesen zu sein, da nach seiner Anwendung nur die Differenzierung einer Klasse schlechter gelang, während sich die meisten Klassen – vor allem kleinere – sogar besser differenzieren ließen. Daher ist anzunehmen, dass die geschilderte Problematik auch nicht im Falle der Testreihe 6 (Kapitel 4.1.2.3) Ursache der beobachteten Verschlechterung der Klassifikationspräzision war. Dennoch wäre – neben den in Kapitel 5.2.2 bereits angeführten Ansatzpunkten zur Verbesserung dieses Softwarepakets – auch eine Implementation von elaborierteren Methoden zur Beurteilung der Klassifikationspräzision in dieses Softwarepaket wünschenswert.

Neben den verschiedenen Methoden zur Beurteilung der Resultate eines Klassifikators muss an dieser Stelle auch noch auf die unterschiedlichen Techniken zur Validierung desselben eingegangen werden. Generell gilt in der Literatur die zehnfache Kreuzvalidierung als das beste Verfahren hierfür [74], welche allerdings nicht auf jede der hier vorliegenden Fragestellungen anwendbar war. Daher musste in

solchen Fällen auf eine Leave-One-Out Kreuzvalidierung zurückgegriffen werden. Neben den hinlänglich bekannten Nachteilen dieser Variante, könnte sie im vorliegenden Fall jedoch zumindest teilweise auch von Vorteil gewesen sein. Unter zehnfacher Kreuzvalidierung werden ja sämtliche Objekte auf zehn Teilmengen verteilt, von denen jeweils eine als Test- und die restlichen als Trainingsmengen genutzt werden. Die einzelnen Klassen des hiesigen Klassifikationsproblems setzten sich teilweise jedoch nur aus einer sehr geringen Menge an Objekten zusammen (zum Teil nur drei Sequenzen pro Klasse). Werden nun mehrere Objekte solcher Klassen derselben Teilmenge zugeteilt, so dürfte eine reliable Validierung bezüglich dieser Klassen nicht mehr möglich sein. Dies lässt sich wie folgt erklären: Fungiert diese Teilmenge als Trainingsmenge, so ist zwar ein verhältnismäßig gutes Training möglich, jedoch existieren ja nur wenige weitere Objekte dieser Klasse in anderen Teilmengen, sodass das Training mit nur wenigen Objekten dieser Klasse validiert werden kann. Fungiert die Teilmenge als Testmenge, wird das Training hinsichtlich dieser Klasse insuffizient, jedoch wird dieses hinsichtlich dieser Klasse insuffiziente Training nun anhand einer größeren Menge an Objekten dieser Klasse validiert. Die Ergebnisse bezüglich kleinerer Klassen würden also fälschlich schlechte Werte suggerieren. Bei einer Leave-One-Out Kreuzvalidierung wird hingegen stets nur ein Objekt getestet, während die restlichen zu Trainingszwecken herangezogen werden. Daher befindet sich hier – außer im Falle von Klassen mit weniger als drei Objekten – stets die Mehrzahl der Objekte einer Klasse in der Trainingsmenge, wodurch der zuvor beschriebene Fall hier nur bei Klassen mit weniger als drei Objekten auftritt.

5.3 Identifikation der zur Klassifikation signifikantesten Aminosäurepositionen

Prinzipiell wäre es sicher einfacher gewesen, die Identifikation der entscheidenden Features anhand von RFE [54][161] ausschließlich an nicht regularisierten Klassifikatoren durchzuführen. Da aber in den regularisierten Trainingssätzen von Klassifikatoren, welche mit künstlichen Sequenzen bessere Ergebnisse lieferten, die signifikanten Differenzen zwischen den zu differenzierenden Klassen für den jeweiligen Klassifikator am besten herausgestellt sein dürften, war anzunehmen, dass die Klassifikatoren in diesen Fällen auch auf ebendiese Differenzen achten und somit sinnvollere Features nutzen. Zudem schien, im Gegensatz zu der in Kapitel 5.2.2 beschriebenen Restriktion der Anwendung von RFE [54][161] auf ausschließlich solche Klassifikatoren, welche nicht auf künstliche Sequenzen zurückgreifen, diese bei der hiesigen Fragestellung nur bedingt zu gelten, da die Feature Selection hier nicht zum Ziel hatte eine potentielle Verbesserung der Klassifikationsgüte zu erreichen, sondern ausschließlich der Detektion der wichtigsten Features dienen sollte. Dies wird verständlicher, führt man sich die Problematik vor Augen, die bei der Anwendung von RFE [54][161] auf Klassifikatoren entsteht, welche zusätzlich künstliche Sequenzen nutzen. In diesem Fall wäre es für eine valide Feature Selection mit dem Ziel einer größtmöglichen potentiellen Verbesserung des Klassifikators nötig, während der Kreuzvalidierungen der Feature Selection nur Basissequenzen als Testsequenzen zu verwenden, welche

zudem nicht als Vorlage der zum Training verwendeten künstlichen Sequenzen dienten, da dem Klassifikator andernfalls beim Training Informationen über die später zu testenden Sequenzen zugespielt würden. Mit anderen Worten: die Menge sich klassenspezifisch ähnelnder Features zwischen den Test- und Trainingsmengen wäre fälschlich groß. So würde das in den jeweiligen Trainingsmengen zwischen den zu differenzierenden Klassen bestehende Verhältnis der Konservierungsgrade der einzelnen Werte von zu vielen Features die Klassenzugehörigkeit der später zu testenden Sequenzen widerspiegeln, wodurch sich während der Kreuzvalidierungen auch entsprechend bessere Ergebnisse unter Erhalt dieser erzielen ließen. Konsekutiv würden hierdurch allerdings die Klassengrenzen enger, sodass der Klassifikator nach erfolgter Feature Selection fremde Testsequenzen, welche bezüglich der insignifikanteren der konservierten Features von den Trainingssequenzen ihrer Klasse abweichen, entsprechend schlechter klassifizieren könnte. Die erhoffte Verbesserung der Klassifikationsgüte bliebe also aus. Da diese Voraussetzungen mit den limitierten Möglichkeiten des Softwarepakets RFE [54][161] jedoch nicht zu erreichen waren, war seine Anwendung mit dem Ziel einer etwaigen Verbesserung der Klassifikationsgüte ausschließlich auf die Klassifikatoren möglich, welche keine künstlichen Sequenzen nutzten. Interessiert jedoch lediglich die Detektion der zur Klassifikation entscheidenden Features, dürfte dieser Effekt weniger bedeutsam sein. Zwar werden auch hier zu viele Features isoliert, allerdings ist deren Signifikanz unterschiedlich groß. So sollten Features umso signifikanter sein, je unterschiedlicher der Konservierungsgrad ihrer einzelnen Werte zwischen den zu differenzierenden Klassen ist, und dementsprechend während der Kreuzvalidierungen auch häufiger Erwähnung finden. Durch wiederholte Repetition der Feature Selection müssten sich diese also zunehmend herauskristallisieren lassen. Diese Hypothesen fanden sich in den Ergebnissen an zahlreichen Beispielen bestätigt; so beschrieben die bezüglich der jeweiligen Positionen der einzelnen Teilbereiche A und B isolierten Features mit überwiegender Mehrheit Aminosäuren, deren Häufigkeit an der jeweiligen Position zwischen den zu differenzierenden Klassen auch stärker variierte.
Auch die Beobachtung, dass die während der Feature Selections des nicht regularisierten OvO-Multiclass-Subklassifikators *„1R vs. 1K vs. 1@“* isolierten Positionen bzw. Features vielfach mit denen der ihm im arbiträren Aufbau des Gesamtklassifikators jeweils folgenden binären Subklassifikatoren identisch waren, dürfte diese Annahmen stützen.

Allerdings erlangten unter Verwendung künstlicher Sequenzen zum Teil Positionen Signifikanz zur Differenzierung, die hinsichtlich natürlicher Sequenzen hierfür eher fraglich von Bedeutung sind, da durch ihren Einsatz zum Teil selbst an solchen Positionen klassenspezifische Aminosäurevariationen von unterschiedlicher Qualität und Quantität auftraten, welche bezüglich der Basissequenzen der zu differenzierenden Klassen hochkonserviert waren, mit anderen Worten stets dieselbe Aminosäure aufwiesen (Kapitel 5.2.1). Natürlich dürften auch solche Variationen dem biologischen Vorbild nachempfunden sein, jedoch ist die Spezifität dieser für eine bestimmte Klasse und damit auch ihre

tatsächliche Signifikanz eher fraglich. Da ja in den Basissequenzen an diesen Stellen keine Variationen auftraten, dürfte die klassenspezifische Verteilung dieser Variationen eher zufällig erfolgt und ihre Spezifität für eine bestimmte Klasse damit auch nicht repräsentativ sein. Bezüglich des Subklassifikators *„2K vs. Nicht-2K“* wurde sogar eine Position für signifikant erachtet, obwohl alle Sequenzen beider Klassen – mit Ausnahme einer künstlichen Sequenz, auf der sich die Signifikanz dieser Position aber nicht begründete – an ihr stets dieselbe Aminosäure aufwiesen. Hier zeigte sich, dass sich sämtliche künstliche Sequenzen einer der beiden zu differenzierenden Klassen – obwohl sie die hochkonservierte Aminosäure vorwiesen – hinsichtlich ihrer Werte an drei Features dieser Position von allen anderen Sequenzen unterschieden. Hierdurch entstand natürlich an diesen Features ein enormes Ungleichgewicht der Konservierungsgrade ihrer Werte zwischen den betreffenden Klassen, welches auch entsprechend ihre (hinsichtlich natürlicher Sequenzen eher fragliche) Signifikanz erklären dürfte. Alles in allem waren solche Beobachtungen jedoch eher die Seltenheit (insgesamt nur zwei aller am häufigsten isolierten Positionen), sodass der Einsatz künstlicher Sequenzen doch gerechtfertigt sein sollte. Dies zeigt sich auch in den Ergebnissen der Feature Selections, bei denen künstliche Sequenzen Verwendung fanden, da sich die Klassifikationssignifikanz der hierunter – hinsichtlich der am häufigsten isolierten Positionen – selektierten Features auch in hohem Maße an den Sequenzlogos der nicht regularisierten Trainingssätze nachvollziehen ließ. Zudem wird die Annahme ebenso durch die bereits beschriebene Beobachtung gestützt, dass die während der Feature Selections des nicht regularisierten OvO-Multiclass-Subklassifikators *„1R vs. 1K vs. 1@“* isolierten Positionen bzw. Features vielfach mit denen der ihm im arbiträren Aufbau des Gesamtklassifikators jeweils folgenden binären Subklassifikatoren identisch waren, denn sie zeigt, dass die Anwendung künstlicher Sequenzen das biologische Vorbild realistisch imitiert.

Ein bei der Anwendung der Funktion *rfe.cv* generell zu bedenkendes Problem lag darin, dass diese – im Gegensatz zu *rfe.ae* – die Signifikanz der Features jeweils nur bezüglich der einzelnen, im Rahmen der Kreuzvalidierung erstellten Trainingsmengen beurteilt. Jedoch waren mit dieser Funktion wesentlich verlässlichere Aussagen darüber möglich, wie viele der jeweils signifikantesten Features tatsächlich auch zur Klassifikation benötigt werden. Natürlich wäre es prinzipiell denkbar gewesen, die jeweils anhand von *rfe.cv* bestimmte Feature-Menge auf die jeweilige Beurteilung der Features durch *rfe.ae* zu übertragen. Da unter dieser Variante jedoch im Rahmen der Suche nach dem bestmöglichen Klassifikator keine reliablen Klassifikationsergebnisse erzielt wurden, wurde von ihr in diesem Rahmen abgesehen. So wurden die Feature Selections an den einzelnen Subklassifikatoren hier multipel repetiert und die Signifikanz eines Features für das Gesamtmodell nach der Häufigkeit bewertet, mit der es bezüglich der einzelnen Trainingsmengen während der Kreuzvalidierungen all dieser Feature Selections isoliert wurde. Dies ließ sich damit begründen, dass auf diese Weise sehr viele unterschiedliche Varianten der Sequenzaufteilung in die einzelnen Teilmengen während der Kreuzvalidierungen getestet

werden konnten. Daher dürften die hierbei am häufigsten isolierten Features bzw. Positionen auch für das Gesamtmodell repräsentativ sein. Allerdings gilt es zu bedenken, dass die Reduktion der Features partiell bzw. in manchen Fällen sogar gänzlich logarithmisch erfolgen musste, sodass möglicherweise nicht immer die optimalen Feature-Mengen ermittelt werden konnten. Hierdurch ist es prinzipiell möglich, dass zum Teil signifikante Features verloren gingen bzw. unwichtigere konserviert wurden. Zudem war es in einigen Fällen notwendig, die nach logarithmischer Reduktion der Features für optimal erachtete Feature-Menge zu modifizieren, um die zur Differenzierung signifikanten Features überhaupt identifizieren zu können. Dies ließ sich aber sinnvoll rechtfertigen, da hiermit eine enorme Reduktion der Feature-Mengen bei nur gering schlechteren Klassifikationsergebnissen unter diesen möglich wurde. Ferner spielte bei der Ermittlung der zur Differenzierung entscheidenden Features auch die Problematik ungleich großer Klassengrößen wieder eine Rolle (Kapitel 5.2.2 und 5.2.3). Zwar dürfte die Beurteilung der Features ihrer Signifikanz nach hiervon unbetroffen sein, da diese von RFE [54][161] jeweils im Rahmen des Trainings durchgeführt wird. Jedoch wird die optimale Menge an Features mithilfe der Gesamterrorquoten bestimmt, was – wie bereits in Kapitel 5.2.3 dargelegt – ebenso zu einer fehlerhaften Einschätzung der optimalen Feature-Mengen führen kann. Die Error-Graphiken der Feature Selections unter logarithmischer Reduktionsvariante zeigen aber, dass doch insgesamt zumeist auch die Feature-Mengen für optimal eingeschätzt wurden, welche sich auch am besten zur Differenzierung zwischen den verschiedenen Klassen eignen. Einzig bezüglich der Subklassifikatoren *„1R vs. 1K vs. 1@“* und *„2K vs. Nicht-2K“* konnte die geschilderte Problematik gehäuft beobachtet werden, sodass die Aussagekraft der bezüglich dieser Subklassifikatoren für signifikant erachteten Features sicher etwas kritischer zu beurteilen ist. Dies gilt natürlich auch für Features, welche bezüglich Subklassifikatoren mit ohnehin eher schlechten Klassifikationseigenschaften isoliert wurden.
Auch die Ergebnisse der nach jeder Feature Selection durchgeführten Testklassifikationen anhand Sequenzsatzes aus der Arbeit von Friedrich, et al. (2006) [46] indizieren eine größtenteils reliable Selektion der zur Differenzierung tatsächlich entscheidenden Features, da sich die Präzision der Klassifikatoren nach den Feature Selections jeweils nur geringfügig änderte.

Die Wahl des Systems zur Kategorisierung der isolierten Positionen erfolgte nach heuristischen Methoden auf der Grundlage mehrerer Überlegungen. Zum einen mussten möglichst jeweils die Positionen ausgefiltert werden, deren Features nur für die anhand der jeweiligen Trainingsmengen während der Kreuzvalidierungen erstellten Modelle signifikant waren, nicht aber für das jeweilige Gesamtmodell. Zum anderen musste auch dem Effekt zu großer Mengen konservierter Features unter Anwendung künstlicher Sequenzen in Kombination mit RFE [54][161] Rechnung getragen werden. Daher sollte die Menge der als „häufigste“ isolierten Positionen in Relation zur Gesamtmenge aller isolierten Positionen jeweils eher klein definiert werden, um möglichst nur Positionen zu erfassen, deren Features auch tatsächlich signifikant waren. Auch im Hinblick auf die Analysen zur Beurteilung einer

etwaigen Korrelation zwischen Signifikanz einer Position zur Klassifikation und ihrer biologischen Relevanz bei der Wahl des Liganden war es sinnvoll, die Menge der als „häufigste“ isolierten Positionen jeweils eher klein zu definieren. Je kleiner diese Menge nämlich gewählt wird, umso signifikanter dürften auch ihre Positionen zur Klassifikation sein, wodurch entsprechend auch die Nachvollziehbarkeit ihrer biologischen Relevanz repräsentativer sein dürfte. Allerdings sollte sie jeweils dennoch ausreichend groß sein, um auch das Ausmaß der Korrelation in etwa abschätzen zu können. Da anhand des gewählten Systems die Menge der am häufigsten isolierten Positionen (Teilbereiche A und B) mit durchschnittlich 6,55 Positionen in der Regel sehr übersichtlich wurde und zudem stets weit weniger als 50% der jeweiligen Gesamtmenge aller isolierten Positionen ausmachte, dürfte es den Ansprüchen also weitestgehend genügt haben.

Dennoch beruht das System auf einer heuristischen Einschätzung, sodass in Zukunft hier sicher weitere Untersuchungen nötig wären, insbesondere um genauer beurteilen zu können, wie viele der isolierten Positionen jeweils tatsächlich ausgefiltert werden müssen, um den beschriebenen Limitationen der Feature Selections ausreichend Rechnung zu tragen. In diesem Zusammenhang erscheint retrospektiv auch ein ganz anderer Ansatz denkbar, der möglicherweise sogar noch sinnvoller sein könnte. Zur Bestimmung der signifikanten Positionen wurden hier zunächst für alle isolierten Features die ihnen entsprechenden Positionen bestimmt und im nächsten Schritt die Positionen ermittelt, deren Features am häufigsten isoliert wurden. Einer Position wurde also umso mehr Signifikanz zugesprochen, je höher die Summe der Isolationsfrequenzen all ihrer Features war. Dies kann jedoch unter Umständen dazu führen, dass auch Positionen für signifikant erachtet werden, bezüglich derer zwar diese Summe sehr hoch ist, die Isolationsfrequenzen der einzelnen Features für sich aber niedrig sind. Dieser Punkt musste also bei der Interpretation der Ergebnisse stets beachtet werden, da besonders in solchen Fällen etwaige Korrelationen mit biologisch relevanten Positionen vielfach eher zufälliger Natur sein dürften. Daher wäre es wahrscheinlich hinsichtlich zukünftiger Arbeiten besser, zunächst die am häufigsten isolierten Features zu bestimmen und im Anschluss nur die Positionen dieser zu errechnen. Die Frage, wie viele der am häufigsten isolierten Features dabei jeweils für signifikant erachtet werden sollten, könnte sich im Falle nicht regularisierter Klassifikatoren beispielsweise an der jeweils durchschnittlich optimalen Feature-Menge orientieren. Im Falle regularisierter Klassifikatoren müsste allerdings auch hier genauer untersucht werden, wie viele der isolierten Features jeweils auf die beschriebene Problematik zurückzuführen sind, welche aus der Anwendung künstlicher Sequenzen in Kombination mit RFE [54][161] resultiert.

5.4 Analyse der zur Klassifikation signifikantesten Aminosäurepositionen im Hinblick auf ihre Rolle bei der Bindungsselektivität

Da der Einfluss der isolierten Positionen mit Kontakt aus den „RFE“-Gruppen der Differenzierungsschritte *„2D vs. Nicht-2D“* und *„Klassen 1 vs. Klassen 2“* auf die Ligandenpräferenz hinsichtlich ihres betreffenden Differenzierungsschritts ja bereits ausführlich in Kapitel 4.3.3 diskutiert wurde, sei an dieser Stelle nur noch kurz auf die übrigen Differenzierungsschritte eingegangen. Bezüglich dieser ließ sich nach den beschriebenen Kriterien zusammenfassend für folgende Positionen ein Einfluss auf die Ligandenpräferenz hinsichtlich des jeweiligen Differenzierungsschritts vermuten:

- „*1@ vs. Nicht-1@*“: Position 15, 32 und 48
- *„1R vs. Nicht-1R“*: Position 13, 15, 16, 32 und 48
- *„2K vs. Nicht-2K“*: Position 12
- *„2R vs. Nicht-2R“*: Position 12
- *„1R vs. 1K vs. 1@“*: Position 13, 32 und 48
- *„2R vs. 2K vs. 2D“*: Position 12 und 53
- „*Y vs. Nicht-Y*“: Position 16

Die Annahme, dass Position 16 doch einen gewissen Einfluss auf die Bindungsfähigkeit einer Domäne haben dürfte – obwohl dies allein anhand der strukturellen Darstellungen zunächst eher fraglich erschien, lässt sich damit begründen, dass sie hinsichtlich aller Liganden bindenden Subklassen (außer Subklasse *2D*) Kontakt mit Position P-3 des Liganden hat und ihre jeweiligen subklassenspezifischen Aminosäureeigenschaften – abgesehen der Subklasse *2D* – subklassenspezifisch jeweils wieder passend zu den jeweiligen Aminosäureeigenschaften an Position P-3 sind. So finden sich an ihr in den Sequenzlogos der Subklassen *1R* und *2R* sowie *1K* und *2K* – passend zu den hier sowohl polaren wie basischen Aminosäureresten an Position P-3 (Arginin respektive Lysin) – auch ausschließlich polare und saure Aminosäurereste. Passend zu den eher aromatischen Aminosäureresten an Position P-3 in Liganden der Subklasse *1@*, finden sich im Sequenzlogo dieser Subklasse an Position 16 neben sauren auch neutrale Aminosäurereste. Zudem zeigt sich Position 16 in Klasse *Nicht-Y* bezüglich ihrer Aminosäureeigenschaften mit nahezu ausschließlich polaren und sauren Aminosäureresten insgesamt hochkonserviert, während diese in Klasse *Y* deutlich variabler sind. Auch dies könnte – wiederum abgesehen von Subklasse *2D*, bei der diese Konservierung wohl eher phylogenetischer Natur sein dürfte – darauf hinweisen, dass diese Position einen Beitrag zur Bindungsfähigkeit einer Domäne leistet. Diese Hypothese wird durch die Arbeit von Cesareni, et al. (2002) [17] gestützt, in der dieser Position nach sequenziellen Gesichtspunkten ebenso eine gewisse Bedeutung hinsichtlich dieses Differenzierungsschritts beigemessen wurde.

Die Analyse, inwiefern auch die Positionen der „RFE“-Gruppe des Differenzierungsschritts *„X_ORS vs. Nicht-X_ORS“* hinsichtlich dieses Differenzierungsschritts Einfluss auf die Ligandenpräferenz

haben, war aufgrund fehlender Beispieldomäne für Klasse *X_ORS* nur eingeschränkt möglich. Dennoch scheint der Einfluss von drei Positionen dieser „RFE“-Gruppe (Position 7, 13 und 48) aber nicht unwahrscheinlich, da sie sich in den strukturellen Darstellungen von Klasse *Nicht-X_ORS* jeweils mit spezifischen Merkmalen innerhalb des Konsensmotivs in Kontakt befinden und ihre subklassenspezifischen Aminosäureeigenschaften auch zu den jeweiligen subklassenspezifischen Eigenschaften dieser Merkmale passen. Allerdings kann im Fall von Position 48 keine besondere Differenz der Aminosäureeigenschaften zwischen Klasse *Nicht-X_ORS* und *X_ORS* ausgemacht werden. An Position 7 hingegen finden sich in sämtlichen Subklassen von Klasse *Nicht-X_ORS* ausschließlich große, aromatische und von Seiten ihrer Säure-/Baseneigenschaften neutrale Aminosäurereste, während in Klasse *X_ORS* an dieser Position auch kleinere, aliphatische und basische Aminosäurereste vorliegen. An Position 13 unterscheidet sich Klasse *X_ORS* von *Nicht-X_ORS* hinsichtlich der Polarität ihrer Aminosäurereste; während sie in den Subklassen der Klasse *Nicht-X_ORS* eher polar sind, lässt sich diese Tendenz in Klasse *X_ORS* nicht feststellen. Vergleicht man die Positionen der „RFE“-Gruppe dieses Differenzierungsschritts mit den Resultaten der Arbeit von Hou T., et al. (2012) [59], so entsprechen fünf der insgesamt zwölf Positionen dieser „RFE“-Gruppe dort Positionen, welchen Signifikanz zur Bindung hinsichtlich „Klasse *1*“- bzw. „Klasse *2*“-Domänen zugesprochen wurde (Position 7, 8, 13, 48 und 52), was bei insgesamt zehn Positionen dort und 58 möglichen Aminosäurepositionen einer Zufallswahrscheinlichkeit von nur 2,39% entspricht. Es muss aber bedacht werden, dass die Signifikanz einer Position dort nicht nach ihrem Einfluss auf die Präferenz für den Ligandentypen sondern ausschließlich auf die Bindung generell bemessen wurde. Daher sind die dort für signifikant erachteten Positionen 35 und 50 eigentlich aus diesem Vergleich auszuschließen, da diese hinsichtlich sämtlicher Klassen stets denselben Aminosäurerest vorweisen. Hierdurch reduziert sich die Wahrscheinlichkeit, dass diese Übereinstimmung zufälliger Natur ist, sogar auf 0,68%. Diese Beobachtung erscheint umso erstaunlicher, hält man sich die eigentlich eher dürftige Klassifikationsgüte des Subklassifikators für diesen Differenzierungsschritt vor Augen. Überdies finden sich auch drei Positionen dieser „RFE“-Gruppe (Position 7, 8 und 52) unter denen, welchen in der Arbeit von Cesareni, et al. (2002) [17] nach sequenziellen Gesichtspunkten eine Bedeutung hinsichtlich dieses Differenzierungsschritts beigemessen wurden.

Natürlich könnte argumentiert werden, dass Positionen, welche bezüglich einer Klasse in den strukturellen Beispieldarstellungen Kontakt zum Liganden zeigten, möglicherweise in den anderen, nicht dargestellten Domänen der betreffenden Klasse keinen Kontakt mit dem Liganden haben. Betrachtet man hierzu aber das Interaktionsprofil der Arbeit von Friedrich, et al. (2006) [46], wird ersichtlich, dass die bei weitem überwiegende Mehrheit der Positionen, hinsichtlich derer ein Einfluss auf die Ligandenpräferenz vermutet wurde, jeweils auch in den meisten Domänen dieses Interaktionsprofils Kontakt zum Liganden besitzen.

Position 7 und 8 aus der „RFE“-Gruppe des Differenzierungsschritts *„X_ORS vs. Nicht-X_ORS“* miteinbezogen – da beide in den entsprechenden strukturellen Darstellungen Kontakt zeigten, für beide in der Arbeit von Hou T., et al. (2012) [59] Signifikanz zur Bindung hinsichtlich „Klasse *2*“-Domänen nachgewiesen werden konnte und zumindest von Position 7 auch hier ein Einfluss auf die Ligandenpräferenz hinsichtlich dieses Differenzierungsschritts wahrscheinlich schien – entsprachen dennoch lediglich elf von insgesamt 42 unterschiedlichen Positionen in den „RFE“-Gruppen der einzelnen Differenzierungsschritte den genannten Kriterien, die den Beitrag einer Position auf die Ligandenpräferenz wahrscheinlich machen. Dies erscheint zunächst eher wenig (26,19%), sodass vermutet werden könnte, die Korrelation anhand der Feature Selections isolierter mit tatsächlich biologisch zur Ligandenpräferenz signifikanten Positionen sei mehr zufälliger Natur. Zudem ergeht aus den Analysen der Arbeit von Hou T., et al. (2012) [59], dass vier dieser elf Positionen möglicherweise sogar eher geringeren Einfluss auf die Bindung des Liganden haben (Position 12, 16, 32, 33). Dies wird jedoch strittig diskutiert, da beispielsweise bezüglich Position 16 bereits in den Arbeiten von Cesareni, et al. (2002) [17], Musacchio A., et al. (1994) [105], Yu H., et al. (1994) [163] und Zucconi A., et al. (2000) [166] ermittelt werden konnte, dass ihr jeweiliger Aminosäurerest eine entscheidende Rolle bei der Präferenz des entsprechenden Aminosäurerests an Position P-3 der Liganden spielt. Zum anderen wird von Cesareni, et al. (2002) [17] vermutet, dass sie möglicherweise auch Einfluss auf die generelle Bindungsfähigkeit einer Domäne hat. Auch der Einfluss von Position 33 konnte bereits in den Arbeiten von Cesareni, et al. (2002) [17] und Zucconi A., et al. (2000) [166] belegt werden, da sich zeigte, dass ihr hinsichtlich Domänen der Klasse *2D* eine zentrale Rolle bei der Bindung zukommt. Zudem finden sich von den in der Arbeit von Hou T., et al. (2012) [59] insgesamt zehn für signifikant befundenen Positionen neun auch in den „RFE“-Gruppen wieder (Positionen 7, 8, 13, 15, 35, 48, 50, 52 und 53), was bei insgesamt 42 verschiedenen Positionen in den „RFE“-Gruppen und 58 möglichen Aminosäurepositionen einer Zufallswahrscheinlichkeit von nur 16,49% entspricht. Bedenkt man allerdings, dass die dort für signifikant erachteten Positionen 35 und 50 eigentlich aus diesem Vergleich auszuschließen sind, da sie hinsichtlich sämtlicher Klassen stets denselben Aminosäurerest vorweisen, so liegt die Wahrscheinlichkeit einer zufälligen Übereinstimmung etwas höher, jedoch immer noch bei unter 30%. Übertragen auf die „RFE“-Gruppen einzelner Differenzierungsschritte finden sich sogar Beispiele – wie im Falle des Differenzierungsschritts *„X_ORS vs. Nicht-X_ORS“* bzw. *„1R vs. 1K vs. 1@“*, in denen die Wahrscheinlichkeit für eine zufällige Übereinstimmung bei weit unter 10% (in den genannten Beispielen: 0.68% respektive 7.83%) liegt, sodass alles in allem eine rein zufällige Korrelation anhand der Feature Selections isolierter mit biologisch zur Ligandenpräferenz signifikanten Positionen eher unwahrscheinlich ist.

Ein ganz anderer Aspekt, der in diesem Rahmen noch zu diskutieren ist, ist die Tatsache, dass bereits in verschiedenen Arbeiten auch Positionen isoliert werden konnten, welche außerhalb der Bindungsstelle

liegen bzw. keinen Kontakt zum Liganden haben und offensichtlich dennoch Einfluss auf die Ligandenpräferenz nehmen [37][38]. Da der Einfluss von Positionen außerhalb der Bindungsstelle jedoch nur laborchemisch, experimentell beurteilt werden kann, waren diesbezügliche Aussagen hier nicht möglich. Daraus folgt jedoch, dass möglicherweise noch deutlich mehr der anhand der Feature Selections isolierten Positionen mit zur Ligandenpräferenz signifikanten Positionen korrelieren.

Zusammenfassend lässt sich festhalten, dass durch die Verschlüsselung der Aminosäuresequenzen in Form von Fisher-Scores eine verlässliche Konservierung der jeweiligen klassenspezifischen, sequenziellen Eigenschaften der Domänen gelang, welche jeweils auch in hohem Maße durch den erstellten, SVM-basierten Klassifikator als wichtiges Unterscheidungsmerkmal zwischen den zu differenzierenden Klassen genutzt wurden. Zwar haben sicher nicht alle klassenspezifischen Differenzen zwangsläufig auch Einfluss auf Präferenz des Ligandentyps und sind zum Teil lediglich phylogenetischer bzw. zufälliger Natur. Dennoch zeigen die Ergebnisse der Feature Selections, dass Positionen, welche Einfluss auf die Ligandenpräferenz nehmen, auch hinsichtlich der Klassifikatoren von besonders großer Signifikanz zu sein scheinen und daher entsprechend häufig isoliert wurden. Da jedoch jeweils auch Positionen isoliert wurden, deren biologische Relevanz bei der Wahl des Liganden eher fraglich schien, lässt sich folgern, dass allein die Signifikanz einer Position zur Klassifikation ohne Zuhilfenahme weiterer Analysen keine validen Rückschlüsse auf ihre biologische Relevanz erlaubt. Allerdings lässt sich mithilfe der vorgestellten Methode die Suche nach biologisch relevanten Positionen (zumindest im Fall von SH3-Domänen) deutlich erleichtern, da ja jeweils nur eine kleine Menge an Positionen weiter analysiert werden muss. Sicher scheinen andere Methoden, wie beispielsweise SPOT [14] oder MIEC-SVMs [57][58][59], hinsichtlich der Erforschung zur Bindung signifikanter Positionen zuverlässiger, jedoch liegen Vorteile dieses Systems vor allem darin, dass hiermit zum einen gerade Aussagen über die Positionen möglich sind, welche die differenten Bindungspräferenzen der einzelnen Klassen erklären (möglicherweise sogar außerhalb der Bindungsstelle) und zum anderen dass dieses System ohne komplexere strukturelle bzw. biochemische Informationen über die Bindung der Liganden auskommt und ausschließlich der Aminosäuresequenz der Domänen bedarf. Hierdurch ist es auch entsprechend flexibel, wodurch es beispielsweise ohne größeren Aufwand an neuere wissenschaftliche Erkenntnisse angepasst werden kann. Gerade dieser Flexibilität wegen böte sich auch sein Einsatz an anderen Proteininteraktionsdomänen an, weshalb die weitere Erforschung der Generalisierbarkeit dieses Systems in Zukunft umso wünschenswerter wäre.

5.5 Klinische Relevanz

Die medizinische und damit auch wissenschaftliche Signifikanz von SH3-Domänen wird bereits mit ihrer in der Natur nahezu ubiquitären Verbreitung innerhalb unterschiedlichster eukaryoter Organismen sowie ihrer Beteiligung an zahlreichen physiologischen wie pathophysiologischen Prozessen

deutlich [50][84][104][106][120][143]. Da SH3-Domänen innerhalb ihrer Trägerproteine als Proteininteraktionsdomänen agieren, richtet sich der wissenschaftliche Fokus hinsichtlich dieser Domänen entsprechend auch vor allem auf ihr Bindeverhalten, mit dem Ziel die Menge der physiologischen Bindungspartner einer SH3-Domäne möglichst präzise eingrenzen zu können und damit konsekutiv auch zur Erstellung des Interaktoms ihres Trägerproteins bzw. einem besseren Verständnis dessen physiologischer Funktionen beizutragen [17][59][93].
So wurde in diesem Rahmen von Cesareni, et al. (2002) [17] ein Modell postuliert, welches die Klassifikation von SH3-Domänen in ein System aus insgesamt acht unterschiedlichen Klassen vorschlägt, die sich durch jeweils unterschiedliche, gehäuft auftretende aminosäuresequenzielle bzw. strukturelle Charakteristika der Liganden definieren. Da sich auf diese Weise zwar die Zahl potentieller Bindungspartner einer Domäne deutlich eingrenzen lassen dürfte, dies jedoch nur unter Kenntnis der Klassenzugehörigkeit der betreffenden Domäne, wurde in selbigem Projekt zudem versucht klassenspezifische Gemeinsamkeiten innerhalb des Sequenzalignments der untersuchten SH3-Domänen zu finden, anhand welcher eine Klassenzuordnung unklassifizierter Domänen auch ohne aufwendige laborchemische Experimente möglich würde. Aufbauend auf der Arbeit und dem Klassenmodell von Cesareni, et al. (2002) [17] sollte hier nun ein in silico Klassifikatormodell erstellt werden, welches solche Gemeinsamkeiten mit noch deutlich größerer Präzision erfasst und somit auch entsprechend zuverlässig die Klassenzugehörigkeit einer Domäne zu einer der acht Klassen erkennt. Zudem sollte untersucht werden, inwieweit die Elemente, welche für das Klassifikatormodell zur Differenzierung von besonderer Bedeutung sind, sich auf Positionen im Alignment beziehen, hinsichtlich derer ein Beitrag zur Determination der Ligandenpräferenz vermutet werden kann. Da die Ergebnisse beider Fragestellungen recht vielversprechend ausfielen, könnte die in dieser Arbeit vorgestellte Methodik also nicht nur bei der Erstellung von Interaktomen noch unklassifizierter SH3-Domänen (bzw. ihrer Trägerproteine) helfen, sondern sogar zur weiteren Erforschung der Faktoren beitragen, welche die Affinität einer Domäne zu ihren Liganden definieren. Dies wiederum könnte zukünftig Ansatzpunkte für neue pharmakologische Interventionsstrategien liefern [146].

Der medizinische Wert solcher Erkenntnisse lässt sich konkret beispielsweise an der Rolle der Tyrosinkinasen Lyn und Fyn bei der Aktivierung von Thrombozyten veranschaulichen. Diese Kinasen der src-Familie binden innerhalb von Thrombozyten an Glycoprotein-VI (GPVI), einen Bestandteil des zellmembranständigen GPVI/FcR-γ-Ketten-Komplex (Glycoprotein-VI/Fc-Rezeptor-γ-Ketten-Komplex) [124][137]. Während Lyn allerdings konstitutiv in aktivierter Form (also auch in ruhenden Thrombozyten) an GPVI gebunden ist, bindet Fyn erst nach Stimulation des GPVI/FcR-γ-Ketten-Komplex an GPVI [124]. Die Bindung an GPVI erfolgt im Falle beider Kinasen jeweils über ihre SH3-Domäne, die mit einer prolinreichen, zytosolischen Domäne von GPVI (Aminosäuresequenz: RPLPPLPPLP) interagiert [137]. Kommt es nach Gefäßverletzung zu Kontakt von Kollagen mit

dimerisiertem GPVI/FcR-γ-Ketten-Komplex an der Zellmembran von Thrombozyten, bewirkt dies eine durch Lyn bzw. Fyn katalysierte Phosphorylierung des sogenannten *Immunoreceptor Tyrosine-based Activation Motif* (ITAM) in den FcR-γ-Ketten des Komplexes [150][86][137]. Dies wiederum führt zur Aktivierung der Tyrosinkinase Syk über Bindung dieser an die FcR-γ-Ketten und damit letztlich zur Initiation des Signalnetzwerks, welches die Aktivierung des Thrombozyten zur Folge hat [124][150][101][137]. Da bekannt ist, dass sowohl Lyn wie auch Fyn SH3-Domänen der Klasse *1R* entsprechen [17] – was sich unter anderem auch in der prolinreichen Domäne von GPVI widerspiegelt, deren Aminosäuresequenz das Konsensmotiv der Klasse *1R* (**R**pl**P**pl**P**plp) vorweist – wäre es naheliegend zukünftig weitere thrombozytäre Proteine mit prolinreicher Domäne der Klasse *1R* dahingehend zu untersuchen, ob diese physiologischerweise mit Lyn bzw. Fyn interagieren. Dies könnte nicht nur zur Erweiterung der thrombozytären Interaktome von Lyn und Fyn und damit letztlich auch des Interaktoms von Thrombozyten [32] selbst beitragen, sondern vor allem auch wichtige Hinweise auf weitere Funktionen dieser Proteine innerhalb von Thrombozyten liefern, wie beispielsweise die kürzlich identifizierte (neben ihrer aktivierenden) zusätzlich inhibitorische Funktion von Lyn bei der Thrombozytenaktivierung [117][98]. In umgekehrter Weise wäre dieser Ansatz auch hinsichtlich GPVI denkbar, wobei hierbei nun sämtliche thrombozytären Proteine mit SH3-Domäne der Klasse *1R* auf ihr physiologisches Interaktionsverhalten mit GPVI untersucht werden müssten. Im Falle von Proteinen mit SH3-Domänen noch unbekannter Klassenzugehörigkeit könnte dabei das hier vorgestellte Klassifikatormodell dazu beitragen diejenigen mit SH3-Domäne der Klasse *1R* herauszufiltern. Die im Rahmen der vorliegenden Arbeit bzw. anderer Arbeiten gewonnenen Informationen darüber, welche Aminosäurepositionen innerhalb von SH3-Domänen klassenspezifisch zur Bindung des Liganden von besonderer physiologischer Bedeutung sein dürften, könnten in diesem Kontext hingegen pharmakologisch genutzt werden. So wäre beispielsweise ein weiterer Ansatz zur Thrombozytenaggregationshemmung denkbar, welcher versucht auf Basis dieser Informationen einen artifiziellen Liganden zu erzeugen, der hochspezifisch die SH3-Domäne von Lyn bzw. Fyn oder aber die prolinreichen Domäne von GPVI bindet und damit im Sinne eines kompetitiven Antagonisten die Bindung von Lyn bzw. Fyn an GPVI inhibiert. Um eine etwaige Kreuzreaktivität mit anderen SH3-tragenden Proteinen bzw. SH3-bindenden Proteinen zu minimieren, könnten zur Erhöhung der Spezifität zudem noch weitere strukturelle Merkmale der betreffenden Proteine bei der Erzeugung des künstlichen Liganden Berücksichtigung finden.

Ein weiteres Beispiel für die erhebliche klinische Bedeutung der Erforschung von SH3-Domänen ist die zentrale Rolle des SH3-tragenden Proteins Eps8 bei onkologischen Prozessen [84]. Dieses ubiquitär exprimierte Enzym ist physiologischer Partizipant bei diversen Signalkaskaden, wie etwa bei der Weiterleitung von EGF-Rezeptor vermittelten Signalen oder bei der Koordination des „Aktin-Capping“ [29][84]. Hinsichtlich vieler der von Eps8 hierbei vermittelten Prozesse spielt auch deren

SH3-Domäne eine wichtige Rolle, so z.B. durch Bindung an Abi-1 bei der Formation des Eps8–Abi-1–Sos-1-Komplexes, der die Aktivierung des G-Proteins Rac zur Folge hat [84][29]. Jüngste Forschungsergebnisse zeigen, dass Eps8 in den meisten Malignomen, unter anderem dem Mamma-, Colon- und Zervixkarzinom wie auch hämatologischen Malignomentitäten, überexprimiert wird, was zu einer Verstärkung von EGF-Rezeptor vermittelten mitogenen Signalen führt und (unter EGF-Stimulation) maligne Zelltransformationen fördert [84][89][88][16][68]. Überdies scheint diesem Enzym auch eine signifikante Rolle bei der Migration von Tumoren zuzukommen [84][47]. Da die SH3-Domäne von Eps8 ein Mitglied der Klasse *2D* [17] ist, welche sich zum einen durch ihr besonderes Konsensmotiv PxxDY und ihren speziellen Bindungsmechanismus von den übrigen Klassen abhebt (Kapitel 4.3.3 und 5.4) und zum anderen dadurch auszeichnet bislang ausschließlich Domänen der Proteinfamilie von Eps8 zu umfassen [17], könnten gerade hier neben interaktomorientierten Forschungsbemühungen vor allem auch pharmakologische Interventionsansätze besonders vielversprechend sein. Bezüglich dieser dürften auch hier wieder Informationen über klassenspezifisch besonders bindungsrelevante Aminosäurepositionen innerhalb von SH3-Domänen (wie unter anderem auch im Rahmen der vorliegenden Arbeit gewonnen) von enormem Nutzen sein.

Weitere Proteine, deren SH3-Domänen in kausalen Zusammenhang mit onkologischen Erkrankungen gebracht werden bzw. potentielle Therapieansatzpunkte dieser darstellen, sind beispielsweise die p85 Untereinheit der PI3-Kinase (z.B. bestimmte Mammakarzinomentitäten) [143], das Adapterprotein Grb2 (z.B. CML) [50] oder die Tyrosinkinase Lyn (z.B. Glioblastom) [85]. Aber auch hinsichtlich vieler nicht-onkologischer Erkrankungen spielen SH3-Domänen häufig eine wichtige Rolle, so beispielsweise bei HIV [120][5] oder auch bei Alzheimer [11]. Es wird also ersichtlich, dass die Erforschung von SH3-Domänen ein sehr breites Spektrum der Medizin betrifft und dort nicht nur zu einem besseren Verständnis zahlloser Erkrankungen sondern auch zur Erweiterung der Therapiemöglichkeiten dieser beitragen dürfte.

6 Zusammenfassung

Die Identifikation der Bindungsspezifität von Proteininteraktionsdomänen bzw. die Fähigkeit Vorhersagen über ihre jeweiligen potentiellen Bindungspartner machen zu können, stellt einen wichtigen Schritt hinsichtlich des Verständnisses ihrer biologischen Funktionen dar. In dieser Arbeit wurde am Beispiel der SH3-Domäne untersucht, inwiefern solche Vorhersagen rein anhand ihrer Aminosäuresequenz mithilfe von Support-Vector-Machines möglich sind. Grundlage hierfür war das von Cesareni, et al. (2002) [17] erarbeitete Modell zur Einteilung der SH3-Domänenfamilie in acht verschiedene Klassen. Um die Aminosäuresequenzen der genutzten Domänen in abstrahierbaren numerischen Größen auszudrücken, wurden jeweils ihre Fisher-Score-Vektoren berechnet, anhand derer im Folgenden die SVM-basierten Klassifikatoren trainiert werden konnten. Mithilfe von Kreuzvalidierungsverfahren ließ sich zeigen, dass Vorhersagen über die Klassenzugehörigkeit einer Domäne anhand dieses Systems weit über der Zufallswahrscheinlichkeit möglich waren. Aufgrund teilweise nur geringer Anzahl an Domänen innerhalb der einzelnen Klassen, wurden zur Verminderung eines hieraus resultierenden overfitting des Klassifikators zusätzlich für jede Klasse künstliche Sequenzen auf der Basis klassenspezifischer HMMs emittiert, was in vielen Fällen zu einer weiteren, erheblichen Verbesserung der Klassifikationsgüte führte. Diesbezüglich indizierte auch die Anwendung von Feature Selections anhand des Softwarepakets RFE [54][161] großes Potential, jedoch gelang es aufgrund der Limitationen dieses Softwarepakets nicht sein Potential im vorliegenden Fall auch optimal auszuschöpfen. Zudem konnte gezeigt werden, dass die Anwendung eines arbiträren Klassifikationsansatzes zur Integration taxonomischer Informationen des Klassensystems in den Klassifikationsprozess dem innerhalb des Softwarepakets e1071 [31][97] bei Multiclass-Problemen angewendeten OvO-Prinzip deutlich überlegen war. Anhand der beschriebenen Methoden konnte schließlich ein Klassifikationsmodell erarbeitet werden, welches unter Kreuzvalidierungsbedingungen eine Gesamttrefferquote von insgesamt 64,71% (und ein durchschnittliches F-Maß von 63,38%) erreichte, womit es dem Zufall (bei acht verschiedenen Klassen: 12,50%) mehr als fünffach überlegen ist.

Im zweiten Teil der Arbeit wurden die Positionen innerhalb des Familien-Alignments der SH3-Domäne ermittelt, die für dieses Klassifikationsmodell zur Differenzierung der einzelnen Klassen am bedeutsamsten sind. Hierzu wurden seine einzelnen Subklassifikatoren zunächst einer Reihe von Feature Selections unterzogen und subklassifikatorspezifisch für jede Position des Alignments die Häufigkeit bestimmt, mit der all ihre Features zusammen hinsichtlich des betreffenden Subklassifikators selektiert wurden. Schließlich wurden hinsichtlich jedes Subklassifikators jeweils die Positionen mit der größten Häufigkeit und die durch ihre selektierten Features beschriebenen Aminosäuren annotiert.

Anhand klassenspezifischer Sequenzlogos der jeweils zu differenzierenden Klassen konnte im dritten Teil der Arbeit gezeigt werden, dass für die einzelnen Subklassifikatoren (zumindest hinsichtlich der Positionen mit der größten Häufigkeit) jeweils Aminosäuren von besonderer Bedeutung waren, die an

den einzelnen Positionen in den Domänen der jeweils zu differenzierenden Klassen tatsächlich auch mit größerer Unterschiedlichkeit vorkamen. Dies demonstriert einerseits, dass die angewendeten künstlichen Sequenzen die jeweiligen klassenspezifischen Eigenschaften der Domänen realitätsnah widerspiegelten, und andererseits diese Eigenschaften auch innerhalb der Fisher-Score-Vektoren verlässlich konserviert wurden. Zudem suggerierte die strukturelle Analyse der jeweils am häufigsten isolierten Positionen innerhalb für die einzelnen Differenzierungsschritte ausgewählten Beispieldomänen in Kombination mit der Untersuchung ihrer jeweiligen klassenspezifischen Aminosäureeigenschaften, dass diesen Positionen überdurchschnittlich häufig auch ein Einfluss auf die Ligandenpräferenz der Domänen zuzuschreiben ist. Mit anderen Worten, Positionen mit Einfluss auf die Ligandenpräferenz der Domänen scheinen auch hinsichtlich der Klassifikatoren von besonders großer Signifikanz zu sein.

Zusammenfassend konnte gezeigt werden, dass innerhalb der Aminosäuresequenzen von SH3-Domänen durchaus ausreichend Informationen enthalten zu sein scheinen, um diese als alleinige Informationsquelle für Methoden des maschinellen Lernens – wie Support-Vector-Machines – zu Vorhersagen über das Bindeverhalten der Domänen nutzen zu können. Natürlich kann das vorgestellte Modell laborexperimentelle Methoden nicht ersetzen, da hiermit beispielsweise alleine schon dem Phänomen der Kreuzreaktivität nicht ausreichend Rechnung getragen werden kann, jedoch könnte es sicher als verlässliches Supplement der Laborexperimente sowohl bei der Klassifizierung einer Domäne wie auch bei der Detektion biologisch, hinsichtlich der Ligandenpräferenz relevanter Positionen dienen. Hierdurch trüge es nicht nur zu einem besseren Verständnis der Interaktome SH3-tragender Proteine bei, sondern könnte möglicherweise sogar bei der Identifikation neuer Ansatzpunkte pharmakologischer Interventionsstrategien helfen. Dass dies aktuell von enormer wissenschaftlicher wie auch klinischer Relevanz sein dürfte, zeigen nicht zuletzt die vielen unlängst zum Thema SH3- (bzw. Proteininteraktionsdomänen generell) erschienenen medizinischen Publikationen [66][67][72][149][160], so beispielsweise eine Arbeit von Watkins, et al. aus dem Jahre 2013 [149], welche sich unter anderem mit der Rolle von SH3-Domänen bei der Pathogenese von AIDS auseinandersetzt. Allerdings gilt es zu bedenken, dass die im Rahmen der hier vorliegenden Arbeit zur Verfügung gestandene Datenmenge teils sehr begrenzt war. Daher müssen die hier erzielten Resultate in Zukunft sicher noch mit größeren Zahlen an klassifizierten Domänen validiert werden, um die Generalisierbarkeit des vorgeschlagenen Modells präziserer abschätzen zu können. Zudem böte es sich an, seine Reliabilität auch an anderen Proteininteraktionsdomänen zu evaluieren, um sein Potential nicht alleine auf die SH3-Domäne zu beschränken.

7 Anhang

7.1 Ordner und Dateien auf beigefügter DVD

Die beigefügte Daten-DVD gliedert sich in folgende Verzeichnisse, deren Inhalt in den folgenden Kapiteln jeweils aufgelistet und erläutert wird:

- „Sequenzen“
- „PDB-Dateien“
- „Klassifikatoren“
- „Ergebnisse_Klassifikatoren“
- „Bestimmung_Signifikanter_Positionen“
- „Ergebnisse_Bestimmung_Signifikanter_Positionen“
- „Pymol“
- „Weblogo“

7.1.1 Ordner „Sequenzen“

Dieses Verzeichnis gliedert sich in vier Unterverzeichnisse:

- „SH3-Familienalignment_SMART“
- „Künstliche_Sequenzen“
- „Trainingssaetze“
- „Testsatz-Friedrich“

7.1.1.1 Unterordner „SH3-Familienalignment_SMART“

Datei	Beschreibung
SMART.aln	Familien-Alignment der SH3-Domäne aus der SMART-Datenbank [82][128] im CLUSTAL-Format
SMART.hmm	HMM, errechnet aus dem Alignment der Datei „SMART.aln“

7.1.1.2 Unterordner „Künstliche_Sequenzen“

Dieses Verzeichnis besteht aus den Unterverzeichnissen „R20“ und „R100“, die sich wiederum jeweils in acht Unterordner – benannt nach den acht unterschiedlichen Klassen der SH3-Domäne – gliedern. Diese beinhalten folgende Dateien:

Datei	Beschreibung
Eich*	Dateien mit den Basissequenzen der Klasse, in deren entsprechendem Unterordner sich die Datei befindet und deren Namen hinter der Bezeichnung „Eich“ angegeben ist.
1, 2, 3….etc.	Dateien mit den Basissequenzen der Klasse, in deren entsprechendem Unterordner sich die Datei befindet, wobei jeweils eine Basissequenz der entsprechenden Klasse fehlt. Die Zahl, mit der die Datei bezeichnet ist, gibt an, welche der Basissequenzen fehlt und orientiert sich an der Reihenfolge der Basissequenzen in der Datei „R0“ (Kapitel 7.1.1.3).
*.aln	Alignments der Sequenzen aus den Dateien „Eich*“ und „1“, „2“, „3“….etc. im CLUSTAL-Format.
*.hmm	HMMs, errechnet aus den Alignments der Dateien „*.aln“
*emit	Künstliche Sequenzen, jeweils emittiert nach dem Vorbild der HMMs der Dateien „*.hmm“. Der jeweilige Unterordner „R20“ bzw. „R100“, in dem sich die Datei befindet, gibt die Regularisierungsstufe wieder.

7.1.1.3 Unterordner „Trainingssaetze“

Datei	Beschreibung
Ausgangsdatensatz.txt	Basissequenzen mit Angabe der Spezies und – falls nicht eindeutig aus dem Sequenz-Identifikator ableitbar – auch des Proteins bzw. der Domänennummer
Verlorene Sequenzabschnitte.txt	Auflistung der Basissequenzen in ihrer hier jeweils verwendeten gegenüber ihrer tatsächlichen Form zur Herausstellung der beim Speichern verloren gegangenen Sequenzabschnitte

In den Unterverzeichnissen „R0“, „R20“ und „R100“ dieses Unterordners befinden sich die einzelnen Trainingssätze, geordnet nach ihrer Regularisierungsstufe:

Datei	Beschreibung
R0	Trainingssatz der Regularisierungsstufe **R0**, bestehend aus allen Basissequenzen
R20, R100	Trainingssätze der Regularisierungsstufe **R20** / **R100** mit allen Basissequenzen und jeweiligem Satz an künstlichen Sequenzen, bei dessen Emission sämtliche Basissequenzen Berücksichtigung fanden
*_20	Trainingssätze der Regularisierungsstufe **R20**, jeweils bestehend aus sämtlichen Basissequenzen sowie einem der Regularisierung **R20** entsprechenden Satz an künstlichen Sequenzen, bei dem jeweils eine andere Basissequenz nicht als Vorlage bei der Emission der künstlichen Sequenzen fungierte. Die Nummer vor dem Unterstrich gibt an, welche der Basissequenzen während Emission des betreffenden Satzes künstlicher Sequenzen nicht als Vorlage diente. Sie orientiert sich an der Reihenfolge der Basissequenzen in der Datei „R0“. Die Nummer hinter dem Unterstrich entspricht der Regularisierungsstufe des jeweiligen Trainingssatzes.

*_100	Trainingssätze der Regularisierungsstufe **R100**, jeweils bestehend aus sämtlichen Basissequenzen sowie einem der Regularisierung **R100** entsprechenden Satz an künstlichen Sequenzen, bei dem jeweils eine andere Basissequenz nicht als Vorlage bei der Emission der künstlichen Sequenzen fungierte. Die Nummer vor dem Unterstrich gibt an, welche der Basissequenzen während Emission des betreffenden Satzes künstlicher Sequenzen nicht als Vorlage diente. Sie orientiert sich an der Reihenfolge der Basissequenzen in der Datei „R0“. Die Nummer hinter dem Unterstrich entspricht der Regularisierungsstufe des jeweiligen Trainingssatzes.

7.1.1.4 Unterordner „Testsatz-Friedrich“

Datei	Beschreibung
Testsatz&Interaktionsprofil-Friedrich	Testsequenzen und Interaktionsprofil der Arbeit von Friedrich, et al. (2006) [46]
Testsatz&Interaktionsprofil-Friedrich_angepasst	Testsequenzen und Interaktionsprofil der Arbeit von Friedrich, et al. (2006) [46] angepasst an das Familien-Alignment der SH3-Domäne aus der SMART-Datenbank [82][128]

7.1.2 Ordner „PDB-Dateien“

Hier befinden sich die im Rahmen der Arbeit verwendeten PDB-Dateien, nach ihrer Zugehörigkeit gegliedert in die Unterordner:

- Beispieldomaenen
- Testsatz-Friedrich

7.1.3 Ordner „Klassifikatoren“

Dieser Ordner enthält die in Kapitel 4.1 beschriebenen Programme. Sie sind nach ihrem jeweiligen Zweck in folgende beiden Unterordner gegliedert

- „Kreuzvalidierung_Klassifikatoren“
- „Validierung_Testsatz-Friedrich“

Die weiteren Dateien dieser Unterordner werden von den Programmen in unterschiedlicher Anzahl jeweils als Input verwendet. Obwohl nicht alle der modifizierten (Unter-)Funktionen des Softwarepakets RFE [54][161] bei den Programmen des Unterordners „Kreuzvalidierung_Klassifikatoren“ Verwendung fanden, wurden sie dennoch der Übersichtlichkeit wegen dort gänzlich aufgeführt.

7.1.3.1 Unterordner „Kreuzvalidierung_Klassifikatoren“

Datei	Beschreibung
Test-*	Programme zur Kreuzvalidierung der einzelnen (Sub-)Klassifikatoren
01_20, 02_20, 01_100…etc.	Trainingssätze der Regularisierungsstufe **R20** bzw. **R100**, jeweils bestehend aus sämtlichen Basissequenzen sowie einem der jeweiligen Regularisierung entsprechenden Satz an künstlichen Sequenzen, bei dem jeweils eine andere Basissequenz nicht als Vorlage bei der Emission der künstlichen Sequenzen fungierte. Die Nummer vor dem Unterstrich gibt an, welche der Basissequenzen während Emission des betreffenden Satzes künstlicher Sequenzen nicht als Vorlage diente. Sie orientiert sich an der Reihenfolge der Basissequenzen in der Datei „R0“. Die Nummer hinter dem Unterstrich entspricht der Regularisierungsstufe des jeweiligen Trainingssatzes.
R0	Trainingssatz der Regularisierungsstufe **R0**, bestehend aus allen Basissequenzen
SMART.hmm	Anhand des Familien-Alignments der SH3-Domäne aus der SMART-Datenbank [82][128] erstelltes HMM

Fisher_Scores	Unterfunktion des Programms „Test-Arbitraer“ zur Errechnung der Fisher-Scores der Trainingssätze mit künstlichen Sequenzen (01_20, 02_20, 01_100…etc.)
rfe.seqZ_linear	Unterfunktion des Programms „Test-AlleKlassen-RFE-Linear_0“ zur Durchführung einer Feature Selection an der jeweils gegebenen Trainingsmatrix
orderFeatures_*_04-2012, plofit, plofit2, PlotRFE_Markierung_auch_zweit_kleinster_Error, rfe*.txt	Modifizierte (Unter-)Funktionen des Softwarepakets RFE [54][161] (Kapitel 3.2)

7.1.3.2 Unterordner „Validierung_Testsatz-Friedrich“

Datei	Beschreibung
Test-Testsatz-Friedrich	Programm zur Anwendung des arbiträren Gesamtklassifikators auf die Testsequenzen der Arbeit von Friedrich, et al. (2006) [46]
R0	Trainingssatz der Regularisierungsstufe **R0**, bestehend aus allen Basissequenzen
R20, R100	Trainingssätze der Regularisierungsstufe **R20** / **R100** mit allen Basissequenzen und jeweiligem Satz an künstlichen Sequenzen, bei dessen Emission sämtliche Basissequenzen Berücksichtigung fanden
Torben	Testsatz der Arbeit von Friedrich, et al. (2006) [46]
SMART.hmm	Anhand des Familien-Alignments der SH3-Domäne aus der SMART-Datenbank [82][128] erstelltes HMM

7.1.4 Ordner „Ergebnisse_Klassifikatoren“

Hier finden sich die Dateien mit den Ergebnissen aus den in Kapitel 4.1 beschriebenen Programmen, gegliedert in folgende Unterordner:

- „Kreuzvalidierung_Klassifikatoren“
- „Validierung_Testsatz-Friedrich“

Datei	Beschreibung
Erg-*	Kreuzvalidierungsergebnisse des betreffenden (Sub-)Klassifikators (Unterordner „Kreuzvalidierung_Klassifikatoren“) bzw. Ergebnisse der Anwendung des arbiträren Gesamtklassifikators auf die Testsequenzen der Arbeit von Friedrich, et al. (2006) [46] (Unterordner „Validierung_Testsatz-Friedrich“)

7.1.5 Ordner „Bestimmung_Signifikanter_Positionen“

Hier befinden sich die in Kapitel 4.2 beschriebenen Programme zur Bestimmung der für die einzelnen Subklassifikatoren signifikantesten Features bzw. Positionen sowie zur Testklassifikation vor und nach Feature Selection. Die weiteren Dateien dieses Ordners werden von diesen Programmen in unterschiedlicher Anzahl jeweils als Input verwendet. Obwohl nicht alle der modifizierten (Unter-)Funktionen des Softwarepakets RFE [54][161] bei den hiesigen Programmen Verwendung fanden, wurden sie dennoch der Übersichtlichkeit wegen gänzlich aufgeführt.

Datei	Beschreibung
RFE-*	Programme zur Bestimmung der für die einzelnen Subklassifikatoren signifikantesten Features bzw. Positionen inklusive der Testklassifikationen des jeweiligen Subklassifikators nach jeder Feature Selection anhand der jeweiligen Testsequenzen der Arbeit von Friedrich, et al. (2006) [46]
Testklassifikation-vor-RFE-*	Programme zur Testklassifikation der Subklassifikatoren vor Feature Selection anhand der jeweiligen Testsequenzen der Arbeit von Friedrich, et al. (2006) [46]

R0	Trainingssatz der Regularisierungsstufe **R0**, bestehend aus allen Basissequenzen
R20, R100	Trainingssätze der Regularisierungsstufe **R20** / **R100** mit allen Basissequenzen und jeweiligem Satz an künstlichen Sequenzen, bei dessen Emission sämtliche Basissequenzen Berücksichtigung fanden
Torben	Testsatz der Arbeit von Friedrich, et al. (2006) [46]
SMART.hmm	Anhand des Familien-Alignments der SH3-Domäne aus der SMART-Datenbank [82][128] erstelltes HMM
orderFeatures_*_04-2012, plofit, plofit2, PlotRFE_Markierung_auch_zweit_kleinster_Error, rfe*.txt	Modifizierte (Unter-)Funktionen des Softwarepakets RFE [54][161] (Kapitel 3.2)
InteractionTorben_ReadTable	R-kompatible Variante des Interaktionsprofils der Arbeit von Friedrich, et al. (2006) [46]

7.1.6 Ordner „Ergebnisse_Bestimmung_Signifikanter_Positionen“

Dieser Ordner gliedert sich in elf Unterordner, benannt nach den insgesamt elf Subklassifikationsschritten, welche in Kapitel 4.2 untersucht wurden (*„1R vs. 1K vs. 1@“* = „Klassen1“ und *„2R vs. 2K vs. 2D“* = „Klassen2“). Diese enthalten jeweils die folgenden Dateien:

Datei	Beschreibung
Erg-RFE-*	Ergebnisse der Bestimmung der für den betreffenden Subklassifikator signifikantesten Features bzw. Positionen inklusive der Ergebnisse aus den Testklassifikationen des betreffenden Subklassifikators nach jeder Feature Selection anhand der jeweiligen Testsequenzen der Arbeit von Friedrich, et al. (2006) [46]
Erg-Testklassifikation-vor-RFE-*	Ergebnisse der Testklassifikation des betreffenden Subklassifikators vor Feature Selection anhand der jeweiligen Testsequenzen der Arbeit von Friedrich, et al. (2006) [46]

1,2,3…etc.	Säulendiagramme jeder einzelnen der bis zu 2000 Feature Selections eines Subklassifikators (jeweils in einem separaten Unterordner „Säulendiagramme“)
Summary.jpg	Zusammenfassendes Säulendiagramm aus allen bis zu 2000 Feature Selections eines Subklassifikators (jeweils in einem separaten Unterordner „Säulendiagramme“)
1000001, 1000002, 1000003…etc.	Error-Graphiken der Feature Selections unter logarithmischer Feature-Reduktion (jeweils in einem separaten Unterordner „Error-Plots“)

7.1.7 Ordner „Pymol“

Datei	Beschreibung
Positionsnummerierung.txt	Vergleich der alignierten und unalignierten (PDB-Dateien) Form der Aminosäuresequenzen der Beispieldomänen

Die weiteren Dateien dieses Ordners gliedern sich ebenso in elf Unterordner, benannt nach den elf Subklassifikationsschritten, welche in Kapitel 4.2 untersucht wurden (*„1R vs. 1K vs. 1@“* = „Klassen1“ und *„2R vs. 2K vs. 2D“* = „Klassen2“). Mit Ausnahme des Unterordners „YvsNichtY“ enthalten diese jeweils die Dateien:

Datei	Beschreibung
Positionen	Positionen der jeweiligen „RFE“-Gruppe des Subklassifikators, in dessen entsprechendem Unterordner sich die Datei befindet. Die Positionen sind in Form ihrer Positionsnummern im Alignment wiedergegeben.
*-R-Code	Programm mit Umrechnungsalgorithmen zur Anpassung der Positionsnummerierung des Alignments an die jeweilige Sequenz der entsprechenden PDB-Datei. Als Input wird die jeweilige Datei „Positionen“ genutzt.

*-PymolCode	PyMOL-kompatibler Code zur Markierung der vom entsprechenden Programm „*-R-Code“ errechneten Positionen innerhalb der PyMOL-Darstellung der betreffenden Beispieldomäne sowie zur individuellen Anpassung der Darstellungen
*-Pymol.pse	Gespeicherte Zwischenstände der auf der Basis der jeweiligen Datei „*-PymolCode“ aufbereiteten PyMOL-Darstellungen der Beispieldomänen. Sie können in PyMOL jeweils über das Menü (File → Open) geöffnet werden.

Innerhalb des Unterordners „YvsNichtY“ finden sich folgende Dateien:

Datei	Beschreibung
InteractionTorben_ReadTable	R-kompatible Variante des Interaktionsprofils der Arbeit von Friedrich, et al. (2006) [46]
Berechnung_Pos_mind_10x_IP	Programm zur Berechnung der Positionen im Alignment, welche innerhalb des Interaktionsprofils der Arbeit von Friedrich, et al. (2006) [46] mindestens zehnmal als in sterischer Beziehung zu den Liganden beschrieben werden (Output „T_NichtY“), sowie zur Berechnung derjenigen dieser Positionen, welche sich mit denen der „RFE“-Gruppe des Differenzierungsschritts *„Y vs. Nicht-Y“* überschneidenden (Output „RFEuT_NichtY“). Der Output „InterquantSave“ gibt für jede Position des Alignments die Häufigkeit wieder, mit der sie innerhalb des Interaktionsprofils als in sterischer Beziehung zu den Liganden beschrieben wird. Der Output „SortedInterquant“ entspricht den sortierten Häufigkeiten in aufsteigender Form. Der Output „RFE“ entspricht den Positionen der „RFE“-Gruppe des Differenzierungsschritts *„Y vs. Nicht-Y“* in Form ihrer Positionsnummern im Alignment. Als Input werden die Dateien „Positionen“ und „InteractionTorben_ReadTable“ genutzt.

RFE_Pos	Positionen der „RFE“-Gruppe des Differenzierungsschritts *„Y vs. Nicht-Y“* in Form ihrer Positionsnummern im Alignment
Positionen	Positionen der „RFE“-Gruppe des Differenzierungsschritts *„Y vs. Nicht-Y“* sowie der „T_NichtY“- und „RFEuT_NichtY“-Gruppen (errechnet durch das Programm „Berechnung_Pos_mind_10x_IP“) in Form ihrer Positionsnummern im Alignment
*-R-Code-Y	Programm mit Umrechnungsalgorithmen zur Anpassung der Positionsnummerierung des Alignments an die jeweilige Sequenz der entsprechenden PDB-Datei. Als Input wird die Datei „Positionen“ genutzt
*-PymolCode-Y	PyMOL-kompatibler Code zur Markierung der vom entsprechenden Programm „*-R-Code-Y“ errechneten Positionen innerhalb der PyMOL-Darstellung der betreffenden Beispieldomäne sowie zur individuellen Anpassung der Darstellungen
*-Pymol.pse	Gespeicherte Zwischenstände der auf der Basis der Datei „*-PymolCode-Y“ aufbereiteten PyMOL-Darstellungen der Beispieldomänen. Sie können in PyMOL jeweils über das Menü (File → Open) geöffnet werden.

7.1.8 Ordner „Weblogo“

Dieses Verzeichnis besteht aus folgenden drei Unterverzeichnissen:

- Sequenzen
- Alignments
- Logos

Jedes dieser Unterverzeichnisse gliedert sich wiederum in jeweils elf Unterordner, die ebenso nach den elf Subklassifikationsschritten benannt sind, welche in Kapitel 4.2 untersucht wurden (*„1R vs. 1K vs. 1@“* = „Klassen1“ und *„2R vs. 2K vs. 2D“* = „Klassen2“). Die einzelnen Dateien sind hierbei jeweils subklassifikatorspezifisch ihren entsprechenden Unterordnern zugeordnet:

Datei	Beschreibung
1@_0, 1K_0, 1R_20, NichtY_20...etc.	Dateien, jeweils mit den klassenspezifischen Sequenzen einer der Klassen, welche in den einzelnen Subklassifikationsschritten des arbiträren Gesamtklassifikators voneinander differenziert werden. Die Nummer nach dem Unterstrich gibt die Regularisierungsstufe wieder. (Unterordner „Sequenzen“)
*.a2m	Alignments der Sequenzen aus den Dateien „1@_0“, „1K_0“, „1R_20“, „NichtY_20“...etc. im A2M-Format. Sämtliche Spalten mit Sternchen bzw. Punkten wurden entfernt (Unterordner „Alignments“)
*0.png	Komparative Sequenzlogos, erstellt anhand der Dateien „*.a2m“ (Unterordner „Logos“)
*0-Frequency.png	Frequency Plots, erstellt anhand der Dateien „*.a2m“ (Unterordner „Logos“)

7.2 Hinweise zu den Programmen auf beigefügter DVD

Die im Rahmen dieser Arbeit erstellten Programme wurden nicht auf optimale Performance im Sinne von Ablaufgeschwindigkeit, sondern ausschließlich auf ihre Funktionalität hin programmiert. Sie enthalten daher auch einige, in den Programmen selbst nicht verwendete bzw. unnütze, jedoch nicht kontraproduktive Abschnitte. Diese dienten während des Programmierens als selbsterstellte Programmiererleichterungen und wurden zur Einreichung dieser Arbeit nicht entfernt, um potentielle Programmierfehler hierbei zu vermeiden, welche die Funktionalität der Programme hätten gefährden können. Die Programme können nach Aufruf in R (Funktion *source*) jeweils mit dem Befehl *ST()* gestartet werden.

7.3 Hinweise zu den Ergebnissen auf beigefügter DVD

7.3.1 Begriffsglossar

Im Output der Programme bzw. in den Ergebnisdateien auf beigefügter DVD wurden kapitelspezifisch folgende Termini verwendet:

Kapitel 4.1:

Terminus	Beschreibung
Map1, MapLinear, MapRadial	Unterschiedliche Bezeichnungen der verschiedenen Confusion-Maps
pred1, pred2	Unterschiedliche Bezeichnungen für die Klassenzuordnungen durch die Klassifikatoren. In den Dateien „Test-Testsatz-Friedrich“ bzw. „Erg-Testsatz-Friedrich“ (Kapitel 7.1.3.2 und 7.1.4) ist „pred1“ auch außerhalb einer Confusion-Map aufgeführt, dabei richtet sich die Reihenfolge der vorhergesagten Klassen nach der Reihenfolge der Sequenzen im Alignment der Testsequenzen von Friedrich, et al. (2006) [46]
true1	Wahre Klassenzugehörigkeit
tot.accuracy_linear/radial	Durchschnittliche Gesamttrefferquote unter zehnfacher Kreuzvalidierung
single.accuracies_linear/radial	Gesamttrefferquoten der einzelnen Testläufe unter zehnfacher Kreuzvalidierung
Total_SV_linear/radial	Anzahl an Supportvektoren während der einzelnen Testläufe unter zehnfacher Kreuzvalidierung
Test_2D	Testergebnisse des Subklassifikationsschritts *„2D vs. Nicht-2D“*
Pred_Test_Klasse2_bzgl_2D	Klassenzuordnung der betreffenden Sequenz seitens der OvO-Multiclass-Subklassifikation *„2R vs. 2K vs. 2D“*
Pred_Test_2D	Klassenzuordnung der betreffenden Sequenz seitens des Subklassifikators *„2D vs. Nicht-2D“*
Sequenz_Nr_bzgl_2D	Sequenznummer der betreffenden Sequenz im Alignment der Basissequenzen
True_Class_bzgl_2D	Wahre Klassenzugehörigkeit der betreffenden Sequenz
NbFeatures.MinError.cv	Kleinste Feature-Menge mit dem durchschnittlich niedrigsten Gesamt-Error
MinError.cv	Von sämtlichen durchschnittlichen Gesamt-Errors bezüglich der einzelnen Mengen an Features der niedrigste Wert

Error.ind_MinError.cv	Durchschnittlicher Individual-Error bezüglich jeder Klasse hinsichtlich der kleinsten Feature-Menge mit dem durchschnittlich niedrigsten Gesamt-Error
Error.se_MinError.cv	Durchschnittlicher Standardfehler des niedrigsten durchschnittlichen Gesamt-Errors
AverageFeatureMenge_linear	Durchschnitt der Featuremengen, die von RFE [54][161] jeweils als optimal für die Klassifikation (unter linearer Kernel-Funktion) berechnet wurden.

Kapitel 4.2:

Terminus	Beschreibung
MetaCopy	Häufigkeit, mit der jede einzelne Aminosäureposition während der Feature Selections des betreffenden Subklassifikators isoliert wurde (Reihenfolge entspricht Positionsnummerierung im Alignment)
MetaMaAs	Positionen der jeweiligen Teilbereiche A und B sowie die ihren Features entsprechenden Aminosäuren in Form von Nummern entsprechend der Reihenfolge der Aminosäuren innerhalb von HMMs
MetaAlignPosWichtig	Positionsnummern der jeweiligen Teilbereiche A und B
MetaAS_HMMWichtig	Die den Features der Positionen der jeweiligen Teilbereiche A und B entsprechenden Aminosäuren in Form von Nummern entsprechend der Reihenfolge der Aminosäuren innerhalb von HMMs
AverageFeatureMenge	Durchschnitt der Featuremengen, die von RFE [54][161] während der Feature Selections des betreffenden Subklassifikators jeweils als optimal für die Klassifikation berechnet wurden.
Prediction	Klassenzuordnung durch Klassifikator
True	Wahre Klassenzugehörigkeit

Map1	Gesamt-Confusion-Map aus den einzelnen Testklassifikationen des betreffenden Subklassifikators nach jeder Feature Selection bzw. Confusion-Map der Testklassifikation des betreffenden Subklassifikators vor Feature Selection

Kapitel 4.3:

Terminus	Beschreibung
T_NichtY	Positionen, welche innerhalb des Interaktionsprofils der Arbeit von Friedrich, et al. (2006) [46] mindestens zehnmal als in sterischer Beziehung zu den Liganden beschrieben werden. Output „Berechnung_Pos_mind_10x_IP“ bzw. Datei „Positionen“ (Kapitel 7.1.7): Nummerierung der Positionen = Positionsnummerierung im Alignment Output „*-R-Code-Y“ bzw. Datei „*-PymolCode-Y“ (Kapitel 7.1.7): Nummerierung der Positionen = Positionsnummerierung in Sequenz der betreffenden PDB-Datei
RFE	Positionen der „RFE“-Gruppe des betreffenden Differenzierungsschritts Output „Berechnung_Pos_mind_10x_IP“ bzw. Datei „Positionen“ und „RFE_Pos“ (Kapitel 7.1.7): Nummerierung der Positionen = Positionsnummerierung im Alignment Output „*-R-Code*“ bzw. Datei „*-PymolCode*“ (Kapitel 7.1.7): Nummerierung der Positionen = Positionsnummerierung in Sequenz der betreffenden PDB-Datei

RFEuT_NichtY	Sich überschneidende Positionen aus „T_NichtY“ und der „RFE“-Gruppe des Differenzierungsschritts „*Y vs. Nicht-Y*“ Output „Berechnung_Pos_mind_10x_IP“ bzw. Datei „Positionen“ (Kapitel 7.1.7): Nummerierung der Positionen = Positionsnummerierung im Alignment Output „*-R-Code-Y“ bzw. Datei „*-PymolCode-Y“ (Kapitel 7.1.7): Nummerierung der Positionen = Positionsnummerierung in Sequenz der betreffenden PDB-Datei
InterquantSave	Häufigkeit, mit der jede einzelne Position des Alignments innerhalb des Interaktionsprofils als in sterischer Beziehung zu den Liganden beschrieben wird.
SortedInterquant	Häufigkeiten von „InterquantSave“, sortiert in aufsteigender Form

7.3.2 Confusion-Maps und Graphiken

In den erstellten Confusion-Maps (Kapitel 7.1.4 und 7.1.6) ist die Reihenfolge der Klassen zwischen Zeilen und Spalten nicht immer identisch. Daher wurden die in Kapitel 4.1 aufgeführten Confusion-Maps teilweise manuell nachbearbeitet, um die Interpretation zu erleichtern.

Da es bei Arbeiten mit dem Softwarepaket RFE [54][161] notwendig war die einzelnen Klassen jeweils aszendierend numerisch zu benennen, sind diese innerhalb der erstellten Error-Graphiken (Kapitel 7.1.6) in der Legende des Individual-Errors nicht mit ihrem tatsächlichen Namen, sondern in Form einer Nummer aufgeführt. Die Klassen, auf welche sich die jeweiligen Nummern beziehen, sind jeweils den einzelnen Programmen zu entnehmen. Bezüglich der Feature Selections solcher Subklassifikatoren, bei denen sich die Wahl der Feature-Menge für die Leave-One-Out Variante zum Teil nach dem zweitniedrigsten Error-Wert richtet, wurde in den Error-Graphiken, in denen sich die minimalen Error-Werte (mit Kreuz markierte Werte) unter anderem bei 1160 (und/oder 1024) Features darstellen, stets auch der zweit kleinste Error-Wert markiert (mit Raute markierte Werte), unabhängig davon, ob weitere Feature-Mengen niedriger als 1160 (und/oder 1024) ebenso den minimalen Error-Wert aufwiesen. Die Wahl der Menge an Features für die Leave-One-Out Variante richtete sich jedoch nur dann nach der kleinsten Feature-Menge mit zweitniedrigstem Error-Wert, falls keine weiteren Feature-Mengen niedriger als 1160 (und/oder 1024) ebenso den minimalen Error-Wert aufwiesen.

In den Säulendiagrammen, welche die Ergebnisse sämtlicher Feature Selections eines Subklassifikators zusammenfassen (Kapitel 7.1.6), sind innerhalb der jeweiligen Legende Teilbereich E mit "Position eher unwichtig für RFE", Teilbereich D mit "Position eher gering wichtig für RFE", Teilbereich C mit "Position eher wichtig für RFE", Teilbereich B mit "Position wichtig für RFE" und Teilbereich A mit "Position sehr wichtig für RFE" bezeichnet. Innerhalb des in Kapitel 4.2 dargestellten, zusammenfassenden Säulendiagramms wurden diese Bezeichnungen für eine erleichterte Interpretation manuell durch den jeweils entsprechenden Teilbereich ersetzt. Ferner gilt es zu erwähnen, dass sämtliche Säulendiagramme der Feature Selections (Kapitel 7.1.6) eher unglücklich mit "Mikroskop. Bindeverhalten vs. RFE" betitelt wurden. Treffender wäre hier sicher die Bezeichnung "Interaktionsprofil vs. RFE" gewesen. Auch die Bezeichnung der Y-Achse dieser Diagramme sollte statt "Häufigkeit, mit der sich eine AS-Pos. an Bindestelle befindet (Mikroskopie)" entsprechend besser "Häufigkeit, mit der sich eine AS-Pos. an Bindestelle befindet (Interaktionsprofil)" lauten. Die in Kapitel 4.2 dargestellten Säulendiagramme wurden diesbezüglich jeweils manuell modifiziert.

8 Literaturverzeichnis

[1] Abraham R. T., Weiss A., Jurkat T cells and development of the T-cell receptor signalling paradigm. Nat Rev Immunol. 2004 Apr;4(4):301-8.

[2] Achuthsankar S. Nair, Computational Biology & Bioinformatics: A Gentle Overview. Communications of the Computer Society of India, January 2007.

[3] Aitio O., Hellman M., Kesti T., Kleino I., Samuilova O., Pääkkönen K., Tossavainen H., Saksela K., Permi P., Structural basis of PxxDY motif recognition in SH3 binding. J Mol Biol. 2008 Sep 26;382(1):167-78. doi: 10.1016/j.jmb.2008.07.008. Epub 2008 Jul 11.

[4] Andrade M. A., Ponting C. P., Gibson T. J., Bork P., Homology-based method for identification of protein repeats using statistical significance estimates. J Mol Biol. 2000 May 5;298(3):521-37.

[5] Backert S., Feller S. M., Wessler S., Emerging roles of Abl family tyrosine kinases in microbial pathogenesis. Trends Biochem Sci. 2008 Feb;33(2):80-90. doi: 10.1016/j.tibs.2007.10.006. Epub 2008 Jan 7.

[6] Baldi P., Chauvin Y., Hunkapiller T., McClure M. A., Hidden Markov models of biological primary sequence information. Proc. Natl. Acad. Sci. USA 1994, 91:1059-1063.

[7] Barker W. C., Garavelli J. S., Hou Z., Huang H., Ledley R. S., McGarvey P. B., Mewes H. W., Orcutt B. C., Pfeiffer F., Tsugita A., et al., Protein information resource: a community resource for expert annotation of protein data. 2001, Nucleic Acid Res. 2929–32

[8] Bateman A., et al., The Pfam Protein Families Database. Nucleic Acids Research, 2000, Vol. 28, No. 1 263-266.

[9] Benson D. A., Karsch-Mizrachi I., Lipman D. J., Ostell J., Wheeler D. L., GenBank. Nucleic Acids Res. 2008 Jan;36(Database issue):D25-30. Epub 2007 Dec 11.

[10] Berman H. M., Westbrook J., Feng Z., Gilliland G., Bhat T. N., Weissig H., Shindyalov I. N., Bourne P. E., The Protein Data Bank. Nucleic Acids Res. 2000;28:235-242.

[11] Bhaskar K., Yen S. H., Lee G., Disease-related modifications in tau affect the interaction between Fyn and Tau. J Biol Chem. 2005 Oct 21;280(42):35119-25. Epub 2005 Aug 22.

[12] Binder A., Kawanabe M., Brefeld U., Efficient Classification of Images with Taxonomies. Computer Vision – ACCV 2009, Lecture Notes in Computer Science Volume 5996, 2010, pp 351-362

[13] BISTIC Definition Committee, NIH working definition of Bioinformatics and Computational Biology (http://www.bisti.nih.gov/CompuBioDef.pdf). 2000.

[14] Brannetti B., Via A., Cestra G., Cesareni G., Citterich M. H., SH3-SPOT: An algorithm to predict preferred ligands to different members of the SH3 gene family. J. Mol. Biol. 2000, 298 (2), 313−328.

[15] Brignatz C., Paronetto M. P., Opi S., Cappellari M., Audebert S., Feuillet V., Bismuth G., Roche S., Arold S. T., Sette C., Collette Y., Alternative splicing modulates autoinhibition and SH3 accessibility in the Src kinase Fyn. Mol Cell Biol. 2009 Dec;29(24):6438-48. doi: 10.1128/MCB.00398-09. Epub 2009 Oct 5.

[16] Castagnino P., Biesova Z., Wong W. T., Fazioli F., Gill G. N., Di Fiore P. P., Direct binding of eps8 to the juxtamembrane domain of EGFR is phosphotyrosine- and SH2-independent. Oncogene. 1995 Feb 16;10(4):723-9.

[17] Cesareni G., et al., Can we infer peptide recognition specificity mediated by SH3 domains? FEBS Lett. 2002 Feb 20;513(1):38-44.

[18] Chenna R., Sugawara H., Koike T., Lopez R., Gibson T. J., Higgins D. G., Thompson J. D., Multiple sequence alignment with the Clustal series of programs. Nucleic Acids Res. 2003 Jul 1;31(13):3497-500.

[19] Cole P. A., et al., Protein tyrosine kinases Src and Csk: a tail's tale. Current Opinion in Chemical Biology 2003, 7:580–585

[20] Colicelli J., ABL tyrosine kinases: evolution of function, regulation, and specificity. Sci Signal. 2010 Sep 14;3(139):re6. doi: 10.1126/scisignal.3139re6.

[21] Colwill K., Field D., Moore L., Friesen J., Andrews B., In vivo analysis of the domains of yeast Rvs167p suggests Rvs167p function is mediated through multiple protein interactions. Genetics. 1999 Jul;152(3):881-93.

[22] Cortes C., Vapnik V., Support-Vector Networks, Machine Learning, 20, 1995. http://www.springerlink.com/content/k238jx04hm87j80g/

[23] Cover T. M., Geometrical and Statistical properties of systems of linear inequalities with applications in pattern recognition. IEEE Transactions on Electronic Computers EC-14, 1965: 326–334.

[24] Cristianini N., Shawe-Taylor J., Kernel Methods for Pattern Analysis, Cambridge University Press, Cambridge, 2004, ISBN 0-521-81397-2

[25] Croce C. M., Oncogenes and Cancer. N Engl J Med 2008; 358:502-511January 31, 2008

[26] Crooks G. E., et al., WebLogo: a sequence logo generator. Genome Res. 2004 Jun;14(6):1188-90.

[27] Dahl D. B., Package 'xtable'. Version 1.6-0, October 12, 2011. xtable.pdf at http://cran.r-project.org/web/packages/xtable/index.html

[28] DeLano W. L., The PyMOL Molecular Graphics System. San Carlos, CA: DeLano Scientific; 2002.

[29] Di Fiore P. P., Scita G., Eps8 in the midst of GTPases. Int J Biochem Cell Biol. 2002 Oct;34(10):1178-83.

[30] Dietterich T., Bakiri G., Solving multiclass problem via error-correcting output code. Journal of Artificial Intelligence Research, Vol. 2 (1995) 263–286

[31] Dimitriadou E., Hornik K., Leisch F., Meyer D., Weingessel A., e1071: Misc Functions of the Department of Statistics (e1071). TU Wien, 2006. R package version 1.5-16.

[32] Dittrich M., Birschmann I., Mietner S., Sickmann A., Walter U., Dandekar T., Platelet protein interactions: map, signaling components, and phosphorylation groundstate. Arterioscler Thromb Vasc Biol. 2008 Jul;28(7):1326-31. doi: 10.1161/ATVBAHA.107.161000. Epub 2008 May 1.

[33] Dröseler C., Untersuchung zur Selektivität versus Promiskuität ausgewählter SH3-Domänen. Humboldt-Universität zu Berlin, Medizinische Fakultät - Universitätsklinikum Charité, 21.11.2005

[34] Duan K.-B., Keerthi S. S., Which Is the Best Multiclass SVM Method? An Empirical Study. N.C. Oza et al. (Eds.): MCS 2005, LNCS 3541, pp. 278–285, 2005, Springer-Verlag Berlin Heidelberg 2005, http://research.yahoo.com/files/multiclass_mcs_kaibo_05.pdf

[35] Dudoit S., Shaffer J. P., Boldrick J. C., Multiple hypothesis testing in microarray experiments. Statistical Science, to appear, preprint available at UC Berkeley, Division Biostatistics working paper series: 2002-110, http://www.bepress.com/ucbbiostat/paper110, 2002.

[36] Eddy S. R., Multiple alignment using hidden Markov models. Proc Int Conf Intell Syst Mol Biol, 1995. 3: p. 114-20.

[37] Espanel X., Sudol M., Yes-associated protein and p53-binding protein-2 interact through their WW and SH3 domains. J Biol Chem. 2001 Apr 27;276(17):14514-23. Epub 2001 Jan 31.

[38] Fazi B., Cope M. J., Douangamath A., Ferracuti S., Schirwitz K., Zucconi A., Drubin D. G., Wilmanns M., Cesareni G., Castagnoli L., Unusual binding properties of the SH3 domain of the yeast actin-binding protein Abp1: structural and functional analysis. J Biol Chem. 2002 Feb 15;277(7):5290-8. Epub 2001 Oct 19.

[39] Feng S., et al., Specific interactions outside the proline-rich core of two classes of Src homology 3 ligands. Proc Natl Acad Sci USA. 1995 Dec 19;92(26):12408-15.

[40] Fernandez-Ballester G., Blanes-Mira C., Serrano L., The tryptophan switch: changing ligand-binding specificity from type I to type II in SH3 domains. J Mol Biol. 2004 Jan 9;335(2):619-29.

[41] Ferraro E., Via A., Ausiello G., Helmer-Citterich M., A novel structure-based encoding for machine-learning applied to the inference of SH3 domain specificity. Bioinformatics 2006, 22 (19), 2333−2339.

[42] Fischer J., Support Vector Machines (SVM), Seminar ”Statistische Lerntheorie und ihre Anwendungen“, 12. Juni 2007, Seite 8

[43] Fisher R. A., An absolute criterion for fitting frequency curves. In: Messenger of Math. Nr. 41, S. 155, 1912.

[44] Fletcher T., Support Vector Machines Explained, UCL, March 1, 2009.

[45] Flevaris P., Li Z., Zhang G., Zheng Y., Liu J., Du X., Two distinct roles of mitogen-activated protein kinases in platelets and a novel Rac1-MAPK-dependent integrin outside-in retractile signaling pathway. Blood. 2009 Jan 22;113(4):893-901. doi: 10.1182/blood-2008-05-155978. Epub 2008 Oct 28.

[46] Friedrich T., et al., Modelling interaction sites in protein domains with interaction profile hidden Markov models. Bioinformatics. 2006 Dec 1;22(23):2851-7. Epub 2006 Sep 25. Supplementary information:http://domains.bioapps.biozentrum.uni-wuerzburg.de/

[47] Funato Y., Terabayashi T., Suenaga N., Seiki M., Takenawa T., Miki H., IRSp53/Eps8 complex is important for positive regulation of Rac and cancer cell motility/invasiveness. Cancer Res. 2004 Aug 1;64(15):5237-44.

[48] Geli M. I., Lombardi R., Schmelzl B., Riezman H., An intact SH3 domain is required for myosin I-induced actin polymerization. EMBO J. 2000 Aug 15;19(16):4281-91.

[49] Gentleman R. C., et al., Bioconductor: open software development for computational biology and bioinformatics. Genome Biol. 2004;5:R80.

[50] Giubellino A., Burke T. R. Jr., Bottaro D. P., Grb2 signaling in cell motility and cancer. Expert Opin Ther Targets. 2008 Aug;12(8):1021-33. doi: 10.1517/14728222.12.8.1021.

[51] Gringhuis S. I., Papendrecht-van der Voort E. A., Leow A., Nivine Levarht E. W., Breedveld F. C., Verweij C. L., Effect of redox balance alterations on cellular localization of LAT and downstream T-cell receptor signaling pathways. Mol Cell Biol. 2002 Jan;22(2):400-11.

[52] Großekathöfer U., Lingner T., Neue Ansätze zum maschinellen Lernen von Alignments. Technische Fakultät der Universität Bielefeld, AG Neuroinformatik, 28.11.2005

[53] Guo Y., Hastie T., Tibshirani R., Regularized linear discriminant analysis and its application in microarrays. Biostatistics. 2007 Jan;8(1):86-100. Epub 2006 Apr 7.

[54] Guyon I., Weston J., Barnhill S., Vapnik V., Gene selection for cancer classification using support vector machines. Machine Learning 46, 2002, pp: 389–422

[55] Haussler D., Krogh A., Mian I. S., Sjölander K., Protein modeling using hidden Markov models: analysis of globins. In Mudge T. N., Milutinovic V., Hunter L., editors, Proceedings of the Twenty-Sixth Annual Hawaii International Conference on System Sciences, volume 1, pages 792-802, Los Alamitos, California, 1993. IEEE Computer Society Press.

[56] Hiipakka M., et al., SH3 domains with high affinity and engineered ligand specificities targeted to HIV-1 Nef. J Mol Biol. 1999 Nov 12;293(5):1097-106.

[57] Hou T. J., Xu Z., Zhang W., McLaughlin W. A., Case D. A., Xu Y., Wang W., Characterization of domain-peptide interaction interface: a generic structure-based model to decipher the binding specificity of SH3 domains. Mol. Cell. Proteomics 2009, 8 (4), 639−649.

[58] Hou T. J., Zhang W., Case D. A., Wang W., Characterization of domain-peptide interaction interface: A case study on the amphiphysin-1 SH3 domain. J. Mol. Biol. 2008, 376 (4), 1201−1214.

[59] Hou T., Li N., Li Y., Wang W., Characterization of domain-peptide interaction interface: prediction of SH3 domain-mediated protein-protein interaction network in yeast by generic structure-based models. J Proteome Res. 2012 May 4;11(5):2982-95. Epub 2012 Apr 9.

[60] Hsu C. W., Chang C. C., Lin C. J., A Practical Guide to Support Vector Classification, Department of Computer Science, National Taiwan University, Taipei 106, Taiwan. Initial version: 2003 Last updated: April 3, 2010; Seite 2

[61] Hughey R., Karplus K., Krogh A., SAM - Sequence Alignment and Modeling Software System. SAM documentation file, Baskin Center for Computer Engineering and Science University of California Santa Cruz, Updated for SAM Version 3.4 July 31, 2003. http://compbio.soe.ucsc.edu/sam.html

[62] Hughey R., Krogh A., Hidden Markov models for sequence analysis: extension and analysis of the basic method. Bioinformatics 1996;12:95.

[63] Innocenti M., Tenca P., Frittoli E., Faretta M., Tocchetti A., Di Fiore P. P., Scita G., Mechanisms through which Sos-1 coordinates the activation of Ras and Rac. J Cell Biol. 2002 Jan 7;156(1):125-36. Epub 2002 Jan 3.

[64] Jaakkola T. S., Haussler D., Exploiting generative models in discriminative classifiers. Adv. Neural Inf. Process. Syst. 1998;11:487–493.

[65] Jaakkola T., Diekhans M., Haussler D., Using the Fisher kernel method to detect remote protein homologies. Proc Int Conf Intell Syst Mol Biol. 1999:149-58.

[66] Johnson M. E., Hummer G., Interface-resolved network of protein-protein interactions. PLoS Comput Biol. 2013;9(5):e1003065. doi: 10.1371/journal.pcbi.1003065. Epub 2013 May 16.

[67] Kaneko T., Huang H., Cao X., Li X., Li C., Voss C., Sidhu S. S., Li S. S., Superbinder SH2 domains act as antagonists of cell signaling. Sci Signal. 2012 Sep 25;5(243):ra68.

[68] Kang H., et al., Gene expression profiles predictive of outcome and age in infant acute lymphoblastic leukemia: a Children's Oncology Group study. Blood. 2012 Feb 23;119(8):1872-81. doi: 10.1182/blood-2011-10-382861. Epub 2011 Dec 30.

[69] Karplus K., Karchin R., Barrett C., Tu S., Cline M., Diekhans M., Grate L., Casper J., Hughey R., What is the value added by human intervention in protein structure prediction? Proteins: Structure Function and Genetics 45(S5):86-91,2001

[70] Karush W., Minima of Functions of Several Variables with Inequalities as Side Constraints. M.Sc. Dissertation. Dept. of Mathematics, Univ. of Chicago, Chicago, Illinois, 1939.

[71] Kay B. K., et al., The importance of being proline: the interaction of proline-rich motifs in signaling proteins with their cognate domains. FASEB J. 2000 Feb;14(2):231-41.

[72] Kishore M., Krishnamoorthy G., Udgaonkar J. B., Critical evaluation of the two-state model describing the equilibrium unfolding of the PI3K SH3 domain by time-resolved fluorescence resonance energy transfer. Biochemistry. 2013 Dec 31;52(52):9482-96. doi: 10.1021/bi401337k. Epub 2013 Dec 19.

[73] Klein C., Kramer E. M., Cardine A. M., Schraven B., Brandt R., Trotter J., Process outgrowth of oligodendrocytes is promoted by interaction of fyn kinase with the cytoskeletal protein tau. J Neurosci. 2002 Feb 1;22(3):698-707.

[74] Kohavi R., A study of cross-validation and bootstrap for accuracy estimation and model selection. Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence 2 (12) 1995: 1137–1143. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.48.529. (Morgan Kaufmann, San Mateo)

[75] Kouranov A., Xie L., de la Cruz J., Chen L., Westbrook J., Bourne P. E., Berman H. M., The RCSB PDB information portal for structural genomics. Nucleic Acids Res. 2006;34:D302-D305.

[76] Koyama S., Yu H., Dalgarno D. C., Shin T. B., Zydowsky L. D., Schreiber S. L., Structure of the PI3K SH3 domain and analysis of the SH3 family. Cell. 1993 Mar 26;72(6):945-52.

[77] Krengel U., Einführung in die Wahrscheinlichkeitstheorie und Statistik. Verlag Friedrich Vieweg & Sohn, Braunschweig/Wiesbaden 1988, S. 157. ISBN 3-528-07259-8.

[78] Krogh A., Brown M., Mian I. S., Sjölander K., Haussler D., Hidden Markov models in computational biology. Applications to protein modeling. J Mol Biol. 1994 Feb 4;235(5):1501-31.

[79] Kuhn H. W., Tucker, A. W., Nonlinear programming. Proceedings of 2nd Berkeley Symposium. Berkeley: University of California Press, 1951: pp. 481–492. http://projecteuclid.org/euclid.bsmsp/1200500249. MR47303

[80] Kuribayashi F., et al., The adaptor protein $p40^{phox}$ as a positive regulator of the superoxide-producing phagocyte oxidase. The EMBO Journal (2002) 21, 6312 – 6320, doi:10.1093/emboj/cdf642

[81] Lee C. H., et al., A single amino acid in the SH3 domain of Hck determines its high affinity and specificity in binding to HIV-1 Nef protein. EMBO J. 1995 Oct 16;14(20):5006-15.

[82] Letunic I., Doerks T., Bork P., SMART 7: recent updates to the protein domain annotation resource. Nucleic Acids Res 2012; doi:10.1093/nar/gkr931

[83] Li C., Schibli D., Li S. S., The XLP syndrome protein SAP interacts with SH3 proteins to regulate T cell signaling and proliferation. Cell Signal. 2009 Jan;21(1):111-9. doi: 10.1016/j.cellsig.2008.09.014. Epub 2008 Sep 30.

[84] Li Y. H., Xue T. Y., He Y. Z., Du J. W., Novel oncoprotein EPS8: a new target for anticancer therapy. Future Oncol. 2013 Oct;9(10):1587-94. doi: 10.2217/fon.13.104.

[85] Liu W. M., Huang P., Kar N., Burgett M., Muller-Greven G., Nowacki A. S., Distelhorst C. W., Lathia J. D., Rich J. N., Kappes J. C., Gladson C. L., Lyn facilitates glioblastoma cell survival under conditions of nutrient deprivation by promoting autophagy. PLoS One. 2013 Aug 2;8(8):e70804. doi: 10.1371/journal.pone.0070804. Print 2013.

[86] Loyau S., Dumont B., Ollivier V., Boulaftali Y., Feldman L., Ajzenberg N., Jandrot-Perrus M., Platelet glycoprotein VI dimerization, an active process inducing receptor competence, is an indicator of platelet reactivity. Arterioscler Thromb Vasc Biol. 2012 Mar;32(3):778-85. doi: 10.1161/ATVBAHA.111.241067. Epub 2011 Dec 8.

[87] Ludwig W., Strunk O., Westram R., Richter L., Meier H., Yadhukumar, Buchner A., Lai T., Steppi S., Jobb G., Förster W., Brettske I., Gerber S., Ginhart A. W., Gross O., Grumann S., Hermann S., Jost R., König A., Liss T., Lüssmann R., May M., Nonhoff B., Reichel B., Strehlow R., Stamatakis A., Stuckmann N., Vilbig A., Lenke M., Ludwig T., Bode A., Schleifer K. H., ARB: a software environment for sequence data. Nucleic Acids Res. 2004 Feb 25;32(4):1363-71. Print 2004.

[88] Maa M. C., Hsieh C. Y., Leu T. H., Overexpression of p97Eps8 leads to cellular transformation: implication of pleckstrin homology domain in p97Eps8-mediated ERK activation. Oncogene. 2001 Jan 4;20(1):106-12.

[89] Maa M. C., Lee J. C., Chen Y. J., Chen Y. J., Lee Y. C., Wang S. T., Huang C. C., Chow N. H., Leu T. H., Eps8 facilitates cellular growth and motility of colon cancer cells by increasing the expression and activity of focal adhesion kinase. J Biol Chem. 2007 Jul 6;282(27):19399-409. Epub 2007 May 12.

[90] Madzarov G., Gjorgjevikj D., Chorbev I., A Multi-class SVM Classifier Utilizing Binary Decision Tree. Informatica 33, 233–241 (2009)

[91] Margolis B., Skolnik E. Y., Activation of Ras by receptor tyrosine kinases. J Am Soc Nephrol. 1994 Dec;5(6):1288-99.

[92] Markowetz, F., Klassifikation mit SVM. Genomische Datenanalyse, Max-Planck-Institut für Molekulare Genetik Berlin Center for Genome Based Bioinformatics, 2003, S. 24

[93] Mayer B. J., SH3 domains: complexity in moderation. J Cell Sci. 2001 Apr;114(Pt 7):1253-63.

[94] Mazharian A., Roger S., Maurice P., Berrou E., Popoff M. R., Hoylaerts M. F., Fauvel-Lafeve F., Bonnefoy A., Bryckaert M., Differential Involvement of ERK2 and p38 in Platelet Adhesion to Collagen. J Biol Chem 2005;280:26002-26010.

[95] McDonald C. B., Seldeen K. L., Deegan B. J., Farooq A., Structural basis of the differential binding of the SH3 domains of Grb2 adaptor to the guanine nucleotide exchange factor Sos1. Arch Biochem Biophys. 2008 Nov 1;479(1):52-62. doi: 10.1016/j.abb.2008.08.012. Epub 2008 Aug 26.

[96] Meltser V., Ben-Yehoyada M., Reuven N., Shaul Y., c-Abl downregulates the slow phase of double-strand break repair. Cell Death Dis. 2010;1:e20. doi: 10.1038/cddis.2009.21.

[97] Meyer D., Support Vector Machines - The Interface to libsvm in package e1071. R-News, Vol.1/3, 9.2001; pp: 23-26.

[98] Ming Z., Hu Y., Xiang J., Polewski P., Newman P. J., Newman D. K., Lyn and PECAM-1 function as interdependent inhibitors of platelet aggregation. Blood. 2011 Apr 7;117(14):3903-6. doi: 10.1182/blood-2010-09-304816. Epub 2011 Feb 4.

[99] MLA style: "Nobelprize.org". Nobelprize.org. 1 Dec 2010, http://nobelprize.org/nobel_prizes/medicine/laureates/1989/press.html

[100] Mochida J., Yamamoto T., Fujimura-Kamada K., Tanaka K., The novel adaptor protein, Mti1p, and Vrp1p, a homolog of Wiskott-Aldrich syndrome protein-interacting protein (WIP), may antagonistically regulate type I myosins in Saccharomyces cerevisiae. Genetics. 2002 Mar;160(3):923-34.

[101] Mócsai A., Ruland J., Tybulewicz V. L., The SYK tyrosine kinase: a crucial player in diverse biological functions. Nat Rev Immunol. 2010 Jun;10(6):387-402. doi: 10.1038/nri2765.

[102] Mongioví A. M., et al., A novel peptide-SH3 interaction. EMBO J. 1999 Oct 1;18(19):5300-9.

[103] Morton C. J., Campbell I. D., SH3 domains. Molecular 'Velcro'. Curr Biol. 1994 Jul 1;4(7):615-7.

[104] Musacchio A., Gibson T., Lehto V. P., Saraste M., SH3--an abundant protein domain in search of a function. FEBS Lett. 1992 Jul 27;307(1):55-61.

[105] Musacchio A., Saraste M., Wilmanns M., High-resolution crystal structures of tyrosine kinase SH3 domains complexed with proline-rich peptides. Nat Struct Biol. 1994 Aug;1(8):546-51.

[106] Musacchio A., Wilmanns M., Saraste M., Structure and function of the SH3 domain. Prog Biophys Mol Biol. 1994;61(3):283-97.

[107] Nguyen J. T., et al., Exploiting the basis of proline recognition by SH3 and WW domains: design of N-substituted inhibitors. Science. 1998 Dec 11;282(5396):2088-92.

[108] Nguyen J. T., et al., Improving SH3 domain ligand selectivity using a non-natural scaffold. Chem Biol. 2000 Jul;7(7):463-73.

[109] Niu G., Bowman T., Huang M., Shivers S., Reintgen D., Daud A., Chang A., Kraker A., Jove R., Yu H., Roles of activated Src and Stat3 signaling in melanoma tumor cell growth. Oncogene. 2002 Oct 10;21(46):7001-10.

[110] Noble M. E., Musacchio A., Saraste M., Courtneidge S. A., Wierenga R. K., Crystal structure of the SH3 domain in human Fyn; comparison of the three-dimensional structures of SH3 domains in tyrosine kinases and spectrin. EMBO J. 1993 Jul;12(7):2617-24.

[111] O'Donovan C., Martin M. J., Gattiker A., Gasteiger E., Bairoch A., Apweiler R., High-quality protein knowledge resource: SWISS-PROT and TrEMBL. Brief Bioinform 2002;3:275-284.

[112] Offenhäuser N., Borgonovo A., Disanza A., Romano P., Ponzanelli I., Iannolo G., Di Fiore P. P., Scita G., The eps8 family of proteins links growth factor stimulation to actin reorganization generating functional redundancy in the Ras/Rac pathway. Mol Biol Cell. 2004 Jan;15(1):91-8. Epub 2003 Oct 17.

[113] Oneyama C., Hikita T., Nada S., Okada M., Functional dissection of transformation by c-Src and v-Src. Genes Cells. 2008 Jan;13(1):1-12. doi: 10.1111/j.1365-2443.2007.01145.x.

[114] Pawson T., Schlessingert J., SH2 and SH3 domains. Curr Biol. 1993 Jul 1;3(7):434-42.

[115] Platt J., Cristanini N., Shawe-Taylor J., Large margin DAGs for multiclass classification. Advances in Neural Information Processing Systems 12. MIT Press (2000) 543–557

[116] Pruitt K. D., Tatusova T., Maglott D. R., NCBI Reference Sequence (RefSeq): a curated non-redundant sequence database of genomes, transcripts and proteins. Nucleic Acids Res 2005;33:D501-D504.

[117] Quek L. S., Pasquet J. M., Hers I., Cornall R., Knight G., Barnes M., Hibbs M. L., Dunn A. R., Lowell C. A., Watson S. P., Fyn and Lyn phosphorylate the Fc receptor gamma chain downstream of glycoprotein VI in murine platelets, and Lyn regulates a novel feedback pathway. Blood. 2000 Dec 15;96(13):4246-53.

[118] Raju T. N., The Nobel chronicles. 1966: Francis Peyton Rous (1879-1970) and Charles Brenton Huggins (1901-97). Lancet. 1999 Aug 7;354(9177):520. doi: 10.1016/S0140-6736(05)75563-X. PMID 10465213.

[119] Ren R., et al., Identification of a ten-amino acid proline-rich SH3 binding site. Science. 1993 Feb 19;259(5098):1157-61.

[120] Rom S., Pacifici M., Passiatore G., Aprea S., Waligorska A., Del Valle L., Peruzzi F., HIV-1 Tat binds to SH3 domains: cellular and viral outcome of Tat/Grb2 interaction. Biochim

Biophys Acta. 2011 Oct;1813(10):1836-44. doi: 10.1016/j.bbamcr.2011.06.012. Epub 2011 Jul 1.

[121] Rubin G. M., et al., Comparative genomics of the eukaryotes. Science. 2000 March 24; 287(5461): 2204–2215.

[122] Sasaki Y., et al., The truth of the F-measure. MIB -School of Computer Science, University of Manchester, pp. 1-5, Version: 26th October, 2007.

[123] Schenk G., TRENNUNG VON DNS-SEQUENZEN MIT SUPPORT VEKTOR MASCHINEN - PROOF OF PRINCIPLE, Bachelorarbeit, Institut für Numerische und Angewandte Mathematik des Zentrums für Informatik an der Georg-August-Universität Göttingen, September 2003, S. 14., ISSN 1612-6793, Nummer ZFI-BM-2003-01

[124] Schmaier A. A., Zou Z., Kazlauskas A., Emert-Sedlak L., Fong K. P., Neeves K. B., Maloney S. F., Diamond S. L., Kunapuli S. P., Ware J., Brass L. F., Smithgall T. E., Saksela K., Kahn M. L., Molecular priming of Lyn by GPVI enables an immune receptor to adopt a hemostatic role. Proc Natl Acad Sci U S A. 2009 Dec 15;106(50):21167-72. doi: 10.1073/pnas.0906436106. Epub 2009 Nov 25.

[125] Schneider T., Lewis K., A Glossary for Biological Information Theory and the Delila System. Schneider Lab. origin: 1999 April 15, updated: version = 3.36 of glossary.html 2012 Jun 06. http://www.ccrnp.ncifcrf.gov/~toms/glossary.html

[126] Schölkopf B., Smola A. J., Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond (Adaptive Computation and Machine Learning). MIT Press, Cambridge, MA, 2002, ISBN 0-262-19475-9.

[127] Schuler G. D., Epstein J. A., Ohkawa H., Kans J. A., Entrez: molecular biology database and retrieval system. Methods Enzymol. 1996;266:141-62.

[128] Schultz J., Milpetz F., Bork P., Ponting C. P., SMART, a simple modular architecture research tool: Identification of signaling domains. PNAS 1998; 95: 5857-5864

[129] Schwarz R., Seibel P. N., Rahmann S., Schoen C., Huenerberg M., Müller-Reible C., Dandekar T., Karchin R., Schultz J., Müller T., Detecting species-site dependencies in large multiple sequence alignments. Nucleic Acids Res. 2009 Oct;37(18):5959-68. Epub 2009 Aug 6.

[130] Scita G., Nordstrom J., Carbone R., Tenca P., Giardina G., Gutkind S., Bjarnegård M., Betsholtz C., Di Fiore P. P., EPS8 and E3B1 transduce signals from Ras to Rac. Nature. 1999 Sep 16;401(6750):290-3.

[131] Seger R., Krebs E. G., The MAPK signaling cascade. FASEB J. 1995 Jun;9(9):726-35.

[132] Semenza G. L., Targeting HIF-1 for cancer therapy. Nat Rev Cancer. 2003 Oct;3(10):721-32.

[133] Shaffer J. P., Multiple hypothesis testing. Annu. Rev. Psychol., 46:561–584, 1995.

[134] Simon J. A., et al., Grb2 SH3 binding to peptides from Sos: evaluation of a general model for SH3-ligand interactions. Chem Biol. 1995 Jan;2(1):53-60.

[135] Steven Martin G., The hunting of the Src. Nature Reviews Molecular Cell Biology 2, 467-475, June 2001. doi:10.1038/35073094

[136] Stuart J. R., Gonzalez F. H., Kawai H., Yuan Z. M., c-Abl interacts with the WAVE2 signaling complex to induce membrane ruffling and cell spreading. J Biol Chem. 2006 Oct 20;281(42):31290-7. Epub 2006 Aug 9.

[137] Suzuki-Inoue K., Tulasne D., Shen Y., Bori-Sanz T., Inoue O., Jung S. M., Moroi M., Andrews R. K., Berndt M. C., Watson S. P., Association of Fyn and Lyn with the proline-rich domain of glycoprotein VI regulates intracellular signaling. J Biol Chem. 2002 Jun 14;277(24):21561-6. Epub 2002 Apr 9.

[138] Tang J., Wang J. Y., Parker L. L., Detection of early Abl kinase activation after ionizing radiation by using a peptide biosensor. Chembiochem. 2012 Mar 19;13(5):665-73. doi: 10.1002/cbic.201100763. Epub 2012 Feb 14.

[139] The PyMOL Molecular Graphics System, Version 0.99, Schrödinger, LLC. http://www.pymol.org/

[140] Tibshirani R., Hastie T., Narasimhan B., Chu. G., Diagnosis of multiple cancer types by shrunken centroids of gene expression. PNAS 2002 99:6567-6572 (May 14).

[141] Tong A. H., et al., A combined experimental and computational strategy to define protein interaction networks for peptide recognition modules. Science. 2002 Jan 11;295(5553):321-4. Epub 2001 Dec 13.

[142] Uetz P., Pohl E., Protein-Protein- und Protein-DNA-Interaktionen. Wink, M. (ed.) Molekulare Biotechnologie, Wiley-VCH 2004, pp. 385-407.

[143] Vadlamudi R. K., Manavathi B., Balasenthil S., Nair S. S., Yang Z., Sahin A. A., Kumar R., Functional implications of altered subcellular localization of PELP1 in breast cancer cells. Cancer Res. 2005 Sep 1;65(17):7724-32.

[144] Vaduva G., Martin N. C., Hopper A. K., Actin-binding verprolin is a polarity development protein required for the morphogenesis and function of the yeast actin cytoskeleton. J Cell Biol. 1997 Dec 29;139(7):1821-33.

[145] Venables W. N., Ripley B. D., Modern Applied Statistics with S. (2002) Fourth edition. Springer. ISBN 0-387-95457-0

[146] Vidal M., Gigoux V., Garbay C., SH2 and SH3 domains as targets for anti-proliferative agents. Crit Rev Oncol Hematol. 2001 Nov;40(2):175-86.

[147] Walker D. R., Koonin E. V., SEALS: a system for easy analysis of lots of sequences. Proc Int Conf Intell Syst Mol Biol. 1997;5:333-9.

[148] Wang M., Yang J., Liu G. P., Xu Z. J., Chou K. C., Weighted-support vector machines for predicting membrane protein types based on pseudo-amino acid composition. Protein Eng Des Sel. 2004 Jun;17(6):509-16. Epub 2004 Aug 16.

[149] Watkins R. L., Zou W., Denton P. W., Krisko J. F., Foster J. L., Garcia J. V., In vivo analysis of highly conserved Nef activities in HIV-1 replication and pathogenesis. Retrovirology. 2013 Oct 30;10:125. doi: 10.1186/1742-4690-10-125.

[150] Watson S. P., Auger J. M., McCarty O. J., Pearce A. C., GPVI and integrin alphaIIb beta3 signaling in platelets. J Thromb Haemost. 2005 Aug;3(8):1752-62.

[151] Weng Z., et al., Structure-function analysis of SH3 domains: SH3 binding specificity altered by single amino acid substitutions. Mol Cell Biol. 1995 Oct;15(10):5627-34.

[152] Wheelan S. J., et al., Domain size distributions can predict domain boundaries. Bioinformatics Vol. 16 no. 7 2000, Pages 613-618.

[153] Wheeler D. L., Barrett T., Benson D. A., et al., Database resources of the National Center for Biotechnology Information. Nucleic Acids Research, vol. 35, no. 1, pp. D5–D12, 2007. doi:10.1093/nar/gkl1031

[154] Wikipedia contributors. Paint.NET. Wikipedia, The Free Encyclopedia. November 12, 2012, 19:35 UTC. Available at: http://en.wikipedia.org/w/index.php?title=Paint.NET&oldid=522684120. Accessed December 12, 2012.

[155] Wikipedia contributors. Support vector machine. Wikipedia, The Free Encyclopedia. December 5, 2012, 14:03 UTC. Available at: http://en.wikipedia.org/w/index.php?title=Support_vector_machine&oldid=526516732. Accessed December 19, 2012.

[156] Wikipedia, Die freie Enzyklopädie: Maximum-Likelihood-Methode. Wikipedia, Die freie Enzyklopädie. 20. September 2011, 11:10 UTC. URL: http://de.wikipedia.org/w/index.php?title=Maximum-Likelihood-Methode&oldid=93855869 (Abgerufen: 15. November 2011, 16:31 UTC)

[157] Wikipedia, The Free Encyclopedia: R (programming language). Wikipedia, The Free Encyclopedia. November 23, 2011, 17:37 UTC. Available at: http://en.wikipedia.org/w/index.php?title=R_(programming_language)&oldid=462130529. Accessed November 24, 2011.

[158] Wikipedia, The Free Encyclopedia: Score (statistics). Wikipedia, The Free Encyclopedia. July 2, 2011, 00:28 UTC. URL: http://en.wikipedia.org/w/index.php?title=Score_(statistics)&oldid=437316965. Accessed November 15, 2011.

[159] Wu X., et al., Structural basis for the specific interaction of lysine-containing proline-rich peptides with the N-terminal SH3 domain of c-Crk. Structure. 1995 Feb 15;3(2):215-26.

[160] Xin X., Gfeller D., Cheng J., Tonikian R., Sun L., Guo A., Lopez L., Pavlenco A., Akintobi A., Zhang Y., Rual J. F., Currell B., Seshagiri S., Hao T., Yang X., Shen Y. A., Salehi-Ashtiani K., Li J., Cheng A. T., Bouamalay D., Lugari A., Hill D. E., Grimes M. L., Drubin D. G., Grant B. D., Vidal M., Boone C., Sidhu S. S., Bader G. D., SH3 interactome conserves general function over specific form. Mol Syst Biol. 2013;9:652. doi: 10.1038/msb.2013.9.

[161] Yang K., Yoon H., Shahabi C., A supervised feature subset selection technique for multivariate time series. Proceedings of the Workshop on Feature Selection for Data Mining: Interfacing Machine Learning and Statistics. In conjunction with the 2005 SIAM International Conference on Data Mining, Newport Beach, CA, April 23, 2005, pp: 72-79 and 92-101.

[162] Yen T. L., Lu W. J., Lien L. M., Thomas P. A., Lee T. Y., Chiu H. C., Sheu J. R., Lin K. H., Amarogentin, a secoiridoid glycoside, abrogates platelet activation through PLC γ 2-PKC and MAPK pathways. Biomed Res Int. 2014;2014:728019. doi: 10.1155/2014/728019. Epub 2014 Apr 29.

[163] Yu H., Chen J. K., Feng S., Dalgarno D. C., Brauer A. W., Schreiber S. L., Structural basis for the binding of proline-rich peptides to SH3 domains. Cell. 1994 Mar 11;76(5):933-45.

[164] Zarrinpar A., et al., The Structure and Function of Proline Recognition Domains. Sci. STKE, 22 April 2003, Vol. 2003, Issue 179, p. re8, DOI: 10.1126/stke.2003.179.re8

[165] Zhang L., Shao C., Zheng D. X., Gao Y. H., An integrated machine learning system to computationally screen protein databases for protein binding peptide ligands. Mol. Cell. Proteomics 2006, 5 (7), 1224−1232.

[166] Zucconi A., Panni S., Paoluzi S., Castagnoli L., Dente L., Cesareni G., Domain repertoires as a tool to derive protein recognition rules. FEBS Lett. 2000 Aug 25;480(1):49-54.

Danksagung

Großer Dank gebührt in erster Linie meinen beiden Betreuern, Herrn Dr. rer. nat. Tobias Müller und Prof. Dr. rer. nat. Jörg Schultz, die die Idee für dieses Projekt hatten und deren Scharfsinn, Ideenreichtum und konstruktive Kritik maßgebliche Bereicherung der Arbeit waren. Ebenso danke ich auch Herrn Prof. Dr. med. Thomas Dandekar, dem Referenten der Arbeit, dessen stets große Hilfsbereitschaft (unter anderem auch im Hinblick auf die Klärung der Kor-/Referentenfrage) und prompte Korrektur der Arbeit die Fertigstellung derselben entscheidend vorantrieben. Besonderer Dank gilt auch meinem ehemaligen Chef, Herrn Prof. Dr. med. R.-I. Ernestus, der diesem Projekt neben meiner ärztlichen Tätigkeit permanent höchste Priorität einräumte und mir beinahe väterlich mit Rat und Tat zur Seite stand. Vor allem aber möchte ich an dieser Stelle meine Eltern erwähnen. Ihrer unermüdlichen Geduld, steten Motivation und außergewöhnlichen Großzügigkeit ist es letztlich zu verdanken, dass diese Arbeit überhaupt zustande kam. Ihnen gilt mein größter Dank. Schließlich seien auch noch alle Freunde und Bekannten angeführt, welche mir während dieses Projekts mit nahezu grenzenlosem Verständnis und steter Hilfsbereitschaft begegnet sind. Auch euch danke ich von Herzen!