Evaluierung von konservierten Signalen bei der Translations-Initiation in prokaryotischen Genomen

Diplomarbeit

von

Ralf Penno

aus Essen

angefertigt

am Institut für Mikrobiologie und Genetik

der Georg-August-Universität zu Göttingen

2005

Referent: Prof. Dr. A. von Haeseler

Korreferent: Prof. Dr. G. Steger

Tag der Abgabe der Diplomarbeit: 17.02.2005

Letzter Tag der mündlichen Prüfung: 23.02.2004

Ich erkläre hiermit, dass ich die vorliegende Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.

Göttingen, den 15.02.2005

Inhaltsverzeichnis

Kapitel 1

Einleitung

Die Kenntnis von interessanten Stellen in einem Genom ist in vielerlei Hinsicht eine Voraussetzung für die molekularbiologische Arbeit. Durch die Sequenzierung erhält man die Nukleotidsequenz als eine lange Kette von Basen, dargestellt durch die Buchstaben A (für Adenin), T (für Thymin), G (für Guanin), C (für Cystein). Die Frage, wo ein Gen anfängt, bzw. wo die entsprechenden Promotorregionen sind, und wo es aufhört, ist für Molekularbiologen von Interesse.

Die Lokalisation der exakten Start- und Endpunkte eines Gens ist sehr wichtig, da sich daraus die Aminosäuresequenz ergibt. Ist die Aminosäuresequenz bekannt, lassen sich hieraus physiologische und biochemische Eigenschaften des Proteins ableiten. So können zum Beispiel durch das Auffinden von Bereichen, die viele hydrophobe Aminosäuren enthalten, transmembrane Helices membrangebundener Proteine lokalisiert werden. Die exakte Vorhersage der Startposition ist unerlässlich, da hier häufig wichtige funktionelle Eigenschaften codiert sind. Signalpeptide können zeigen, wo ein Protein in der Zelle lokalisiert ist (Nielsen *et al.*, 1994), eine Ribosombindestelle, die vor dem Startcodon liegt, kann Aufschluss über die Stärke der Initiation der Translation geben (Barrik *et al.*, 1994) und der N-Terminus des Gens kann Informationen über die Lebensdauer des Proteins enthalten (Varshavsky *et al.*,

1996). Die genaue Bestimmung des Translations Startpunktes (Translation Initiation Site, TIS) ist bei Klonierungen und anderen molekularbiologischen Arbeiten wichtig, weil sonst unter Umständen ein Teil des Proteins verloren gehen kann oder gar ein Fremdteil mit kloniert wird. Dies hätte Auswirkungen auf die Struktur und die Funktion des Proteins.

Drei Nukleotide bilden ein „Triplett“ bzw. „Codon“ aus. Ein Codon codiert für eine spezielle Aminosäure. Während der Proteinbiosynthese werden die Codons abgelesen, in Aminosäuren übersetzt und diese zu einem Protein verknüpft. Durch die Codierung können 64 verschiedene Codons ($4^3 = 64$) mit den vier Nukleotiden gebildet werden. Davon werden drei als Stoppcodons benutzt, die restlichen 61 codieren die insgesamt 20, in Proteinen zu findenden, Aminosäuren. Hierbei gibt es verschiedene Codierungen für die gleiche Aminosäure. Man spricht von der „Degeneration des genetischen Codes“.

Die Codierung als Triplett ist trotzdem notwendig, da bei einer zweier-Codierung nur 16 ($4^2 = 16$) Codons gebildet werden können, folglich bietet sie nicht genügend Möglichkeiten.

Ein weiterer Punkt ist der Leserahmen. Es gibt sechs verschiedene Leserahmen, drei in „vorwärts Richtung“ und drei in „rückwärts Richtung“. Das Ribosom darf beim Ablesen, wie man sich leicht vorstellen kann, nicht aus dem richtigen „Takt“ kommen, da hierdurch die Aminosäuresequenz vollständig verändert werden würde, welches sich auf Struktur und Funktion des Proteins auswirken würde. In den meisten Fällen ist damit ein Protein nicht mehr funktionsfähig. Zusätzlich könnte bei einer Leserastermutation[1] bzw. Verschiebung ein vorzeitiges Stoppcodon übersetzt werden, wodurch das entstehende Protein nicht komplett wäre. In der folgenden DNA-Sequenz sind die drei vorwärts gerichteten Lesemöglichkeiten dargestellt (siehe Tabelle 1.1, S. 9).

Desweiteren ist noch zu erwähnen das je länger der offene Leserahmen (Open

[1] Leserastermutation (= Verschiebung des Leserahmens)

5’ 3’

atgcccaagctgaatagcgtagaggggttttcatcatttgaggacgatgta**taa**

1	atg	ccc	aag	ctg	aat	agc	gta	gag	ggg	ttt	tca	tca	ttt	gag	gac	gat	gta	taa
	M	P	K	L	N	S	V	E	G	F	S	S	F	E	D	D	V	*
2	tgc	cca	agc	tga	ata	gcg	tag	agg	ggt	ttt	cat	cat	ttg	agg	acg	atg	tat	
	C	P	S	*	I	A	*	R	G	F	H	H	L	R	T	M	Y	
3	gcc	caa	gct	gaa	tag	cgt	aga	ggg	gtt	ttc	atc	att	tga	gga	cga	tgt	ata	
	A	Q	A	E	*	R	R	G	V	F	I	I	*	G	R	C	I	

Tabelle 1.1: Hier sind drei verschiedene Leserahmen dargestellt. Der erste Leserahmen beginnt an Position 1 mit einen *atg* das für Methionin codiert. Der zweite Leserahmen beginnt mit einen *tgc* für Cystein und der dritte mit einem *gcc* für Alanin. Die Stoppcodons sind mit einem $*$ in der Proteinsequenz dargestellt.

Reading Frame, ORF) ist, desto wahrscheinlicher ist auch, dass es sich um eine codierende Region handelt.
Eine TIS beginnt normalerweise mit einem Startcodon aus der Menge {ATG, GTG, TTG,} und endet mit einem Stoppcodon aus {TAG, TAA, TGA}. Das Startcodon ATG codiert für die Aminosäure Methionin (M), was bedeutet, dass die meisten Proteine mit der Aminosäure Methionin beginnen. Dennoch tragen nicht alle fertigen Proteine ein endständiges Methionin, da es in einigen Fällen zu einer Abtrennung dieser Aminosäure kommt. Dies geschieht an der noch unfertigen, wachsenden Polypeptidkette und zwar durch das Enzym Methionin-Aminopeptidase.
Ein Stoppcodon codiert in der Regel für keine Aminosäure, sondern es terminiert die Translation. Eine Ausnahme gibt es zum Beispiel in der mitochondrialen DNA, wo das Codon TGA für die Aminosäure Trypthophan codiert. Es ist nun vorstellbar, dass man das Ende eines Gens leichter vorhersagen

kann als den Start, da es nur ein mögliches Stoppcodon im entsprechenden Leserahmen gibt und dieses auch das Genende markiert. Im Gegensatz dazu ist es schwieriger eine TIS zu bestimmen. Ein Methionin-codierendes Codon kann auch innerhalb von Genen vorkommen und dort nur für die Aminosäure Methionin codieren, aber nicht für ein Genstart. Desweiteren ist es möglich, dass ein solches Codon auch „vor“ dem Genstart lokalisiert ist. Eine solche Position würde man als *upstream* in Bezug auf die Startposition bezeichnen. Im Gegensatz dazu würde man eine Position „hinter“ dem Startcodon, also innerhalb des Gens, als *downstream* bezeichnen. Mit downstream wird immer die 3’ Richtung bezeichnet.
Die Zelle ist in der Lage den richtigen Translationsstart zu bestimmen. Das heißt sie kann unterscheiden, ob es sich um einen Translationsstart handelt oder ob das Codon nicht den Start eines Gens beschreibt.

1.1 Ribonukleinsäure

1.1.1 Struktur

Der Aufbau von RNA gleicht im Prinzip dem der DNA. Beide Moleküle sind aus einer Kette von Nukleotiden, bestehend aus Base, Zucker und Phosphat, aufgebaut.
Jedoch unterscheiden sich RNA und DNA im Zuckermolekül und in einer der Basen. In der RNA ist das Zuckermolekül eine Ribose, in der DNA eine 2’ - Desoxiribose. Der Unterschied im Zuckermolekül führt dazu, dass die RNA andere Helixkonfirmationen ausbilden kann als die DNA. Der zweite Unterschied ist, dass man in der DNA Thymin und in der RNA Uracil als Base findet. Nukleotide werden deshalb unterschiedlich abgekürzt. Zum einen A, T, C, G für ein DNA-Molekül und zum anderen A, U, G, C für ein RNA-Molekül.
Thymin und Uracil unterscheiden sich durch eine zusätzliche Methylgruppe

des Thymins, was verantwortlich für eine größere thermodynamische Stabilität der RNA gegenüber der DNA ist.

In den meisten Fällen liegt RNA einsträngig, DNA jedoch doppelsträngig

Symbol	Bedeutung	Nukleotid
G	Guanin	
U	Uracil	
C	Cytosin	
A	Adenin	
R	pu**R**in	A oder G
Y	p**Y**rimidin	C oder U
S	**S**tark	G oder C **(3 Wasserstoffbrückenbindungen)**
W	sch**W**ach	A oder U **(2 Wasserstoffbrückenbindungen)**
K	**K**eto	G oder U
M	a**M**ino	A oder C
B	nicht A	(C oder G oder U)
D	nicht C	(A oder G oder U)
H	nicht G	(A oder C oder U)
V	nicht U	(A oder C oder G)
N	beliebige Basen	A,C,G oder U

Tabelle 1.2: In dieser Tabelle sind die gängigsten Abkürzungen für Nukleotide nach **IUPAC** (International Union of Pure and Applied Chemistry) dargestellt. Hier ist zu beachten, dass sich die Base U auf RNA bezieht.

vor. In der einzelsträngigen RNA lagern sich komplementäre Bereiche anein-

ander.
Die Strukturen, die RNA Moleküle ausbilden können, kann man in Primärstrukturen, Sekundärstrukturen und Tertiärstrukturen einteilen. Die einfachste Struktur ist die Primärstruktur, welche der Abfolge der Nukleotide entspricht. Dabei sind die einzelnen Nukleotide über Esterbindungen am Phosphat miteinander verknüpft. Darauf aufbauend bilden sich räumlich Sekundär- und Tertiärstruktur.

1.2 Sekundärstruktur

Die Fähigkeit Basenpaare zu bilden ist eine Voraussetzung für die Entstehung von RNA-Sekundärstrukturen. In der Regel paaren sich die Watson & Crick Basenpaare also A:U (A:T in DNA) und G:C (Watson, J.D. und Crick, F.H.C., 1953). Desweiteren existiert das Wobble-Basenpaar G:U, welches relativ häufig im RNA-Molekül zu finden ist. Diese drei Basenpaartypen können regelmäßige Helices beliebiger Länge ausbilden.

Die Basen in der RNA-Sekundärstruktur sind über Wasserstoffbrückenbindungen miteinander verbunden. Bei der Paarung von G und C werden drei Wasserstoffbrückenbindungen ausgebildet. Ein A:T Basenpaar hat zwei Wasserstoffbrückenbindungen und das Wobble-Basenpaar hat ebenfalls zwei Wasserstoffbrückenbindungen. Je mehr Wasserstoffbrückenbindungen ausgebildet werden, desto stabiler ist die Verbindung. Dies wirkt sich aber nur geringfügig auf die Stabilität der Struktur aus.
Die RNA-Sekundärstruktur wird hauptsächlich durch die Stapelwechselwirkung („Stacking-Effekt“ siehe Abbildung 1.2, S. 14) stabilisiert. Der Stacking-Effekt beruht darauf, dass die Basenpaare beim Stapeln der aromatischen Ringsysteme miteinander wechselwirken (siehe Abbildung 1.2, S. 14), wodurch sich die durch Basenpaarung gebildete Heilx stabilisiert.

Die nicht komplementären Bereiche einer Helix bilden Schleifen (Loops),

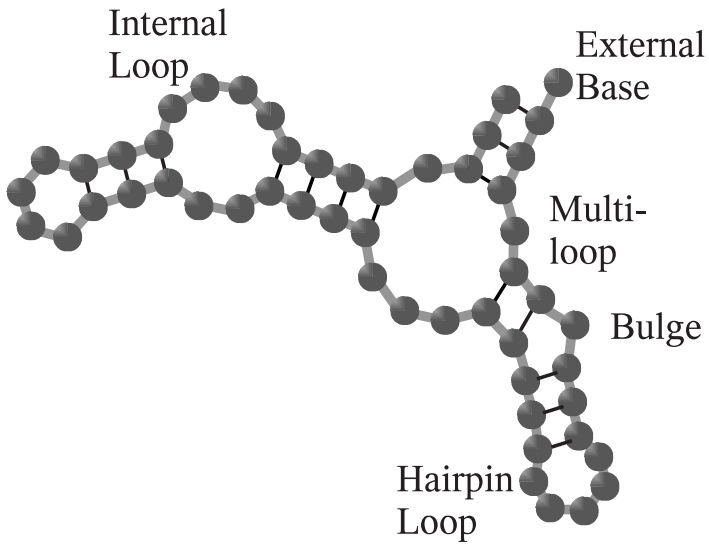


Abbildung 1.1: Dargestellt werden verschiedene Schleifen Typen (Loop-Typen). Sie können alle als Strukturelemente in RNA-Sekundärstrukturen vorkommen.

welche unterschiedliche Eigenschaften haben können. Hairpinloops sind beispielsweise dadurch ausgezeichnet, dass sie eine Helix abschließen. Meist haben sie eine Größe von fünf Nukleotiden. Ein Hairpinloop muss aber mindestens aus drei Nukleotiden bestehen, da er sonst eine Helix nicht überbrücken könnte. Es sei noch der Internal-, Bulge- und Multiloop zu erwähnen. Diese Looparten verbinden zwei oder mehrere Helices (siehe Abbildung 1.1).
Jeder Loop, egal von welchem Typ, destabilisiert die RNA Struktur, denn je mehr Basenpaare eine Struktur ausbilden kann, desto stabiler ist sie. Eine Struktur, die viele Basenpaare besitzt, kann einen größeren Stacking-Effekt

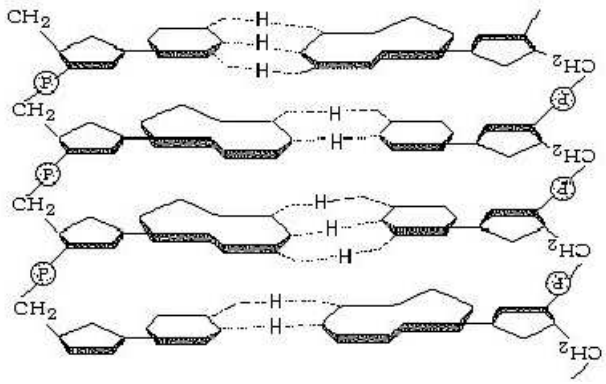


Abbildung 1.2: In der Abbildung ist der Ausschnitt einer DNA bzw. RNA-Doppelhelix zusehen. Dabei fällt auf, dass die gepaarten Basen eine Ebene bilden, also gestapelt übereinander liegen. Dadurch entsteht der so genannte „Stacking-Effekt“. Dieser verleiht der Doppelhelix eine höhere Festigkeit. Weiterhin wird die Doppelhelix durch Wasserstoffbrückenbindungen zwischen Basenpaaren stabilisiert.

und mehr Wasserstoffbrückenbindungen ausbilden als eine Struktur, die weniger Basenpaare besitzt.

1.2.1 Aufgaben von mRNAs

Die mRNA spielt eine zentrale Rolle im Informationsfluss innerhalb der Zelle. Durch Interaktionen mit Proteinen oder anderen RNA-Molekülen werden Reifung, Transport, Prozessierung, intrazelluläre Lokalisation und Translation der mRNAs reguliert. mRNAs besitzen eine komplexe Sekundärstruktur, die als spezifische Erkennungs- und Bindungsstelle dient. Ein Beispiel hierfür ist die Codierung der Aminosäure Selenocystein.
Die Aminosäure Selenocystein wird durch das Triplett UGA codiert. Die Differenzierung zwischen einem UGA-Stoppcodon und einem Selenocystein-Codon erfolgt in Prokaryoten durch eine downstream lokalisierte mRNA-Sekundärstruktur, die vom Selenocystein spezifischen Elongationsfaktor *SelB* erkannt wird (Gursinsky *et al.*, 2000).

RNA-Sekundärstrukturen können hoch konserviert sein. Es existieren gleiche RNA-Sekundärstrukturen, die durch unterschiedliche Sequenzen ausgebildet werden. Es ist nicht immer die Sequenz alleine konserviert, sondern es ist durchaus möglich, dass es eine Mutation in der Sequenz gibt, die aber keine Auswirkung auf die ausgebildete RNA-Sekundärstruktur hat.

1.3 Freie Enthalpie (Gibb'sche Energie)

Die Stabilität einer RNA-Sekundärstruktur hängt in erster Linie vom Stacking-Effekt ab. Dieser ist um so stärker, je mehr Basenpaare gestapelt sind. Die Struktur wird durch Basenpaare stabilisiert und durch Loops destabilisiert. Um eine RNA-Sekundärstruktur zu denaturieren, benötigt man Energie. Je mehr Loops eine Struktur beinhaltet, um so weniger Energie benötigt man. Wie stark der Einfluss der Loops ist, hängt von mehreren Faktoren ab. Zum einen von der Größe des Loops, da größere Loops mehr destabilisieren als kleinere.
Zum anderen spielt die Loopart und die Loopsequenz eine Rolle. Ein Hairpinloop, der auf eine Helix folgt und als letztes regulär gepaartes Basenpaar ein (G:C) besitzt, ist stabiler als ein Hairpinloop mit letztem regulär gepaarten Basenpaar (A:U). Außerdem spielt die Loopsequenz eine Rolle.
Ob sich eine Struktur spontan ausbildet oder nicht hängt von der „Freien Reaktions Enthalpie“ ab. Diese ist wie folgt definiert:

chemisches Gleichgewicht: $\text{Edukt}(E) \rightleftharpoons \text{Produkt}(P)$

Gleichgewichtskonstante: $K = \frac{[P]}{[E]}$

Energiebilanz: $\Delta G^0 = -RT \ln K$

ΔG^0: Gibb'sche Standard-Reaktionsenthalpie „standard free energie“ in kJ/mol oder $kcal/mol$; 1 $kcal/mol$ entspricht 4,18 kJ/mol.

R: steht für die Gaskonstante

T: steht für die Temperatur in Kelvin

K: steht für die Gleichgewichtskonstante

Bei endergonischen Prozessen, das heißt bei $\Delta G^0 > 0$ benötigt die Struktur eine Aktivierungsenergie, um sich auszubilden. Bei exergonischen Prozessen, das heißt bei $\Delta G^0 < 0$, benötigt die Struktur keine Aktivierungsenergie, sondern bildet sich „freiwillig“ aus. Die Struktur, die den niedrigsten ΔG^0 Wert hat, wird auch oft als optimale thermodynamische Struktur (mfe-Struktur „***m**inimum of **f**ree **e**nergie*“) bezeichnet.

Ein RNA-Molekül kann verschiedene Strukturen ausbilden. Haben die verschiedenen Strukturen eine identische oder ähnliche freie Energie, liegen sie in Lösung zu gleichen Anteilen vor. Es kann nicht davon ausgegangen werden, dass sich nur *eine* spezielle RNA-Struktur ausbildet, es handelt sich vielmehr um eine thermodynamische RNA-Struktur-Verteilung. In dieser Verteilung existieren optimale und suboptimale RNA-Strukturen gleichzeitig und es lagern sich ständig Strukturen ineinander um.

Suboptimale Strukturen können auch eine biologische Funktion haben. Bei einigen Viroiden werden suboptimale Strukturen vom Wirt erkannt und weiter prozessiert, während optimale Strukturen nicht erkannt werden und somit die Viroidreplikation verhindert würde.

1.4 Tertiärstruktur

Aus einer RNA-Sekundärstruktur kann sich die RNA-Tertiärstruktur ausbilden. Diese wird genauso durch Wasserstoffbrückenbildung und Stacking stabilisiert, wie die RNA-Sekundärstruktur. Es existieren jedoch zusätzliche Strukturelemente wie beispielsweise ein Pseudoknoten. Ein Pseudoknoten kann entstehen, wenn ein freies Ende mit einem Loop wechselwirkt (Pleij *et al.*, 1985).
Weiterhin können sich Tertiärstrukturen durch Loop-Loop Interaktion ausbilden, welche zwischen allen Looptypen auftreten kann, was häufiger als Pseudoknotenbildung vorkommt (Cate *et al.*, 1996).
Eine weitere Tertiärstrukturart ist der „Trippelstrang“. In die Trippelstrangbildung sind Basen verwickelt, die mit mehr als nur mit *einer* Base über Wasserstoffbrückenbindungen wechselwirken.
Die Tertiärstruktur spielt bei der biologischen Funktionalität eine wichtige Rolle, denn erst die Tertiärstruktur verleiht in vielen Fällen der RNA ihre biologische Funktionalität. Für diese Arbeit wurde die RNA-Tertiärstruktur vernachlässigt, da die verwendeten RNA-Strukturvorhersage Programme sie nicht berechnen können. Da RNA-Tertiärstrukturen noch komplexer aufgebaut sind als RNA-Sekundärstrukturen ist ihre Vorhersage mit noch größerer Ungenaugkeit behaftet.

1.5 Ziel dieser Arbeit

Ziel der vorliegenden Arbeit war es, RNA-Sekundärstrukturvorhersagen auf mögliche Informationen zu untersuchen, die zur Verbesserung der Vorhersage der Startposition eingesetzt werden könnten. Diese Informationen könnten beispielsweise mögliche konservierte Kontaktpunkte sein.
Die Sequenzen stammen aus einem verifizierten Datensatz und wurden in positive und negative Beispiele eingeteilt. Positive Beispiele besitzen eine

zentrierte TIS, bei den negativen Beispielen ist die TIS in einem Rahmen von 60 Nukleotiden up- und 60 Nukleotiden downstream lokalisiert (siehe Abschnitt 2.1.2, S. 20).
Mit RNA-Sekundärstrukturvorhersage Programme wurden Strukturen für alle Beispiele (positiv und negativ) vorhergesagt. Die vorhergesagten RNA-Sekundärstrukturen wurden in Kontaktpunkte übersetzt und in speziellen 3d-Dichteplots visualisiert. Die erstellten Dichteplots dienten als Voruntersuchung, ob sich im Bereich einer „wahren“ TIS Kontaktpunkte häufen. Da die TIS bei den positiv Beispielen immer an derselben Stelle und bei den negativen Beispielen an unterschiedlichen Stellen lokalisiert ist, sollten sich die Plots der positiven von den der negativen Beispiele unterscheiden. Aus den Dichteplots ließ sich entnehmen, dass sich positive und negative Beispiele in Bezug auf den Informationsgehalt unterscheiden.
Im Anschluss daran wurde eine mittelwertsbasierte Klassifikation der Kontaktpunkte durchgeführt. Dazu wurde mit Methoden des Maschinellen Lernens ein Klassifikator trainiert.

Kapitel 2

Methoden und Material

Es folgt eine Beschreibung der Programme, die zur Berechnung der RNA-Sekundärstruktur bzw. der RNA-Sekundärstrukturverteilungen benutzt wurden. Bei dieser Beschreibung wird auf die Anwendung und auf die Algorithmen der Programme eingegangen. Die benötigten Sequenzen stammen aus dem EcoGene Datensatz, welcher die Grundlage für weitere Experimente und Berechnungen ist.

2.1 Verwendete Daten

2.1.1 EcoGene-Datensatz

Der Datensatz umfasst 722 experimentell verifizierte Gene des Organismus *Escherichia coli K-12*. *E. coli* ist ein fakultativ anaerobes, Gram-negatives Stäbchenbakterium. Dieses Bakterium gehört zu den am besten untersuchten Mikroorganismen und wurde 1997 komplett sequenziert. Viele der ca. 4300 Gene sind auf ihre Funktion hin untersucht worden.
Bei dem EcoGene-Datensatz (Kenneth E. Rudd *et al.*, 2000) handelt es sich um eine Sammlung von Nukleotid- bzw. Proteinsequenzen, bei denen unter anderem die genaue Position der TIS experimentell ermittelt wurde.

Der Datensatz ist frei verfügbar und kann aus dem Internet herunter geladen[1] werden.
Es wurden aus dem Datensatz 722 Gene verwendet, die mit der Markierung „verified“ versehen sind.

2.1.2 Positive und negative Beispiele

Als positive Beispiele wurden die Startregionen der 722 Gene aus dem oben beschriebenen Datensatz verwendet. Ein positives Beispiel sieht folgendermaßen aus. Um eine „wahre“ TIS wurde ein Rahmen von 3000 Nukleotiden ausgeschnitten, so dass die TIS an Position 1500 liegt.

Abbildung 2.1: Hier ist ein positives Beispiel dargestellt. Eine verifizierte „wahre TIS“ ist flankiert von 1500 Nukleotiden.

Negativbeispiele wurden folgendermaßen generiert: In einem Rahmen von 60 Nukleotiden upstream und 60 Nukleotiden downstream einer „wahren“ TIS wurden alle „falschen“ Startcodons gesucht. Sie sind im gleichen Leserahmen wie das „wahre“ und es darf kein Stoppcodon vor dem „wahren“ Start auftreten. Alle „falschen“ Startcodons haben, wie die „wahren“, auch die

[1] http://www.bmb.med.miami.edu/EcoGene/EcoWeb

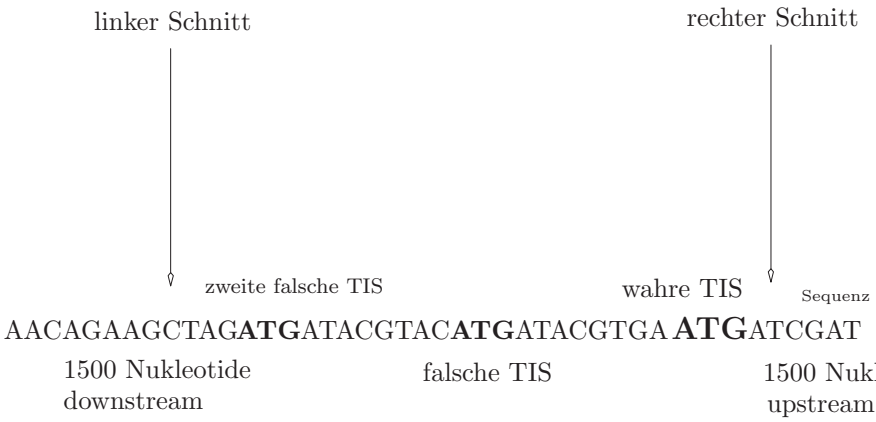


Abbildung 2.2: Hier ist ein negatives Beispiel dargestellt. Eine „falsche TIS“ ist flankiert von 1500 Nukleotiden. Die „wahre TIS“ ist unter Umständen auch in dem negativen Beispiel enthalten. Allerdings ist sie nicht zentriert wie bei den positiven Beispielen.

Triplett-Folge ATG, GTG oder TTG. Damit ist ein Negativbeispiel ein Start-Triplett in einem Rahmen von 60 Nukleotiden um den „wahren“ Genstart. Die „falsche“ TIS beschreibt keinen Genstart. Um alle „falschen“ Startcodons wird ein Rahmen von 3000 Nukleotiden ausgeschnitten. In jedem Rahmen hat die „falsche“ TIS die Position 1500. Es wurden 854 negativ Beispiele erstellt.

2.2 Vienna RNA Package

In dem Vienna RNA Package[2] (Hofacker *et al.*, 1994) stehen drei Algorithmen im Vordergrund. Diese drei Algorithmen wurden für die RNA-Sekundärstrukturvorhersage entwickelt. Zum einen, um die Struktur mit minimaler freier Energie vorherzusagen, zum zweiten um eine Basenpaarwahrscheinlichkeit zwischen einzelnen Basenpaaren zu bestimmen und zum dritten, um alle möglichen suboptimalen Strukturen einer gegebenen Sequenz zu bestimmen.

[2] http://www.tbi.univie.ac.at/∼ivo/RNA/

Der Algorithmus zur Berechnung der Struktur mit der minimalen freien Energie „the minimum free energy “ (Zuker und Stiegler, 1981) liefert nur die thermodynamisch optimale Struktur zurück.
Desweiteren gibt es den „Partition function Algorithmus“ (McCaskill *et al.*,1990). Er berechnet aus einer thermodynamischen Strukturverteilung einzelne Basenpaarwahrscheinlichkeiten. Sie können in einem Dot-Plot (siehe Abbildung 2.5, S. 25) ausgegeben werden. Diese Algorithmen sind in dem Programm `RNAfold` realisiert.
Als drittes existiert der „suboptimal folding Algorithmus“ (Wuchty *et al.*, 1999). Er berechnet alle suboptimalen Strukturen für eine eingegebene Sequenz. Die suboptimalen Strukturen werden in einer wählbaren Energiedifferenz zur mfe-Struktur berechnet. Dieser Algorithmus ist in dem Programm `RNAsubopt` implementiert worden.
Desweiteren wurde das Vienna Package um das Programm `RNALfold` erweitert. Diese Erweiterung wurde gemacht, um vollständige Genome nach lokal stabilen Strukturen zu durchsuchen, wie z.B. miRNAs (mini RNAs). Diese spielen eine Rolle bei der Expression von Genen. Die kurzen miRNA Stücke (meist 18-25 Nukleotide lang) können dann wie ein Repressor auf eine andere mRNA wirken, womit die Translation eines Protein herauf- bzw. herunterreguliert wird. Die kurzen RNA Stücke sind von der Pflanze bis zum Menschen hoch konserviert.
Im Weiteren wird auf die Programme im Detail eingegangen.

2.3 RNAfold

Das Programm `RNAfold` bekommt wahlweise eine oder mehrere Sequenzen übergeben. Für diese berechnet es die mfe-Struktur, die in der üblichen Punkt-Klammer-Notation ausgegeben wird. Hierbei steht eine öffnende Klammer „(“ für eine Base, die downstream gepaart vorliegt. Eine schließen-

de Klammer „)“ repräsentiert eine Base, die upstream gepaart vorliegt. Ein Punkt steht für eine Base, die ungepaart ist. Jedes Nukleotid wird also durch ein Zeichen repräsentiert (siehe Abbildung 2.3).
In der ersten Zeile der Ausgabe ist die Länge der Sequenz gegeben. Danach folgt die Sequenz selbst und darunter ist sie in Punkt-Klammer-Notation angegeben. Unter der Struktur ist die freie Energie ΔG^0 in $kcal/mol$ notiert.
Die Punkt-Klammer-Notation kann in andere Formate überführt werden, da sie alle dazu nötigen Informationen enthält. `RNAfold` wandelt die Punkt Klammer Notation in eine bildliche Darstellung der Struktur um. Dies kann man sich zusätzlich ausgeben lassen (siehe Abbildung 2.4, S. 24).
Beim Aufrufen des Programms können mehrere Parameter übergeben wer-

length = 31
`AUCCGUAGCUCGUCAGCUAUCGAGUCAGAUG`
`.((.((((((....)))))).)).....`
minimum free energy = -6.70 kcal/mol

Abbildung 2.3: Die Abbildung zeigt ein Ausgabebeispiel aus dem Programm `RNAfold`, mit einer zufällig generierten Sequenz.

den, wie beispielsweise die Temperatur, für welche die mfe-Struktur berechnet werden soll. Der Standard (default) ist auf 37°C eingestellt. Weitere Parameter findet man in der Beschreibung des Programms.[3]
Um die mfe-Struktur oder eine suboptimale Struktur zu berechnen, muss ΔG^0 bestimmt werden. Dazu werden Energieparameter für die unterschiedlichen Loop-Typen und für die Stapelwechselwirkungen benötigt. Die Energieparameter für die Stapelwechselwirkungen wurden experimentell gemessen (Turner und Freier, 1990) und deren aktuellen Werte können in einer entsprechenden Tabelle[4] nachgelesen werden (siehe auch Abbildung 2.8, S. 32).

[3] http://www.tbi.univie.ac.at/∼ivo/RNA/RNAfold.html
[4] http://ntdb.chem.cuhk.edu.hk/index.htm

Die Energieparameter für kleine Loops wurden ebenfalls experimentell ge-

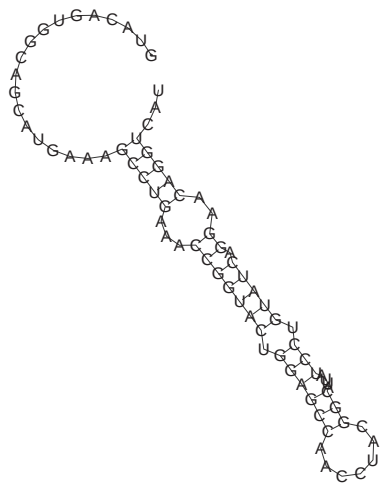


Abbildung 2.4: Dargestellt ist die Ausgabe einer RNA-Sekundärstruktur des Programms `RNAfold`.

messen und anschließend auf größere Loops extrapoliert. Für einige Loops gibt es keine seriösen Angaben, was eine Bewertung der Struktur erschwert, denn nur mit allen thermodynamischen Parametern wäre eine exakte mfe-Struktur-Vorhersage möglich.

2.3.1 Partition function

Beim Aufruf von `RNAfold` wird der Parameter `-p` übergeben. Mit dieser Option wird die „Partition function“ berechnet. Der „Partition function“ Algorithmus berechnet keine RNA-Sekundärstruktur. Vielmehr wird eine Basenpaar Wahrscheinlichkeits-Matrix berechnet. Daraus kann ablesen werden, wie wahrscheinlich es ist, dass ein bestimmtes Basenpaar in der entsprechenden Strukturverteilung gepaart bzw. ungepaart vorliegt. Die Matrix wird in Form eines Dot-Plot (siehe Abbildung 2.5, S. 25) dargestellt. Auf den beiden Achsen wird jeweils die Sequenz aufgetragen und in die Matrix werden die Basenpaare eingetragen. Die „dicke“ eines eingetragenen Punktes ist propor-

tional zur Basenpaarungswahrscheinlichkeit. Je größer ein Punkt ist desto wahrscheinlicher ist es, dass die Base (x) und die Base (y) in der Strukturverteilung gepaart vorliegen.

Desweiteren werden die vorhergesagten Strukturen in Pseudo-Bracket-Notation

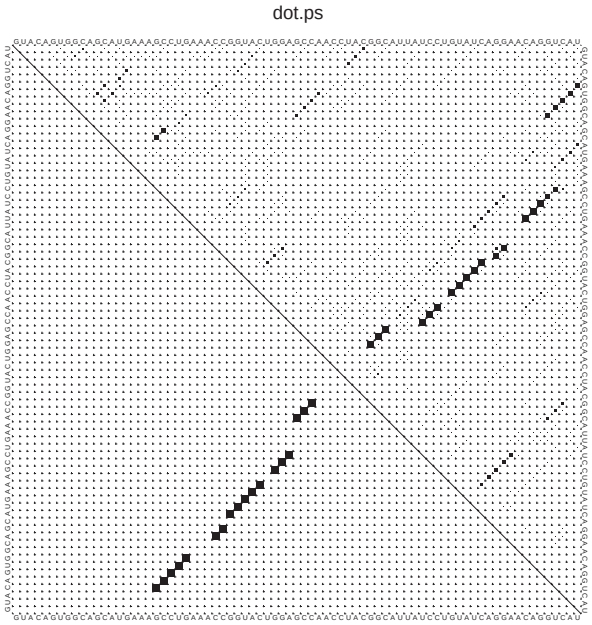


Abbildung 2.5: Dieser Dot-Plot stammt von `RNAfold`. Die Partition function wurde mitberechnet. In der unteren Hälfte wurden die Basenpaare der mfe-Struktur eingezeichnet. In der oberen Hälfte wurde die RNA-Sekundärstruktur Verteilung dargestellt. Die „dicke“ der eingetragenen Punkte ist proportional zur Basenpaarwahrscheinlichkeit.

ausgegeben. Die Pseudo-Bracket-Notation (siehe Abbildung 2.6, S. 26) ist eine Erweiterung der Punkt-Klammer-Notation um einige Symbole. Ein „,“ steht für eine schwach gepaarte Base ohne Paarungsorientierung. Ein „|“ steht für eine Base, die sehr wahrscheinlich gepaart vorliegt aber ebenfalls keine Orientierung besitzt. Die beiden geschweiften Klammern „{“ „}“ repräsentieren schwach gepaarte Basen jedoch mit entsprechender Paarungsorientierung.

In einer weiteren Zeile wird der mfe-Strukturanteil in der gesamten Ver-

```
.((.((((((....,)))))},)).........
```
free energy of ensemble = -7.44 kcal/mol
frequency of mfe structure in ensemble 0.301566

Abbildung 2.6: Ein Ausgabebeispiel des Programms `RNAfold` mit der Option `-p`. Die Struktur wird in der Pseudo-Bracket-Notation ausgegeben. Man erhält zusätzliche Informationen über die Strukturverteilung und deren Energie. Desweiteren wird der Anteil der mfe-Struktur in dieser Verteilung angegeben.

teilung angegeben. Dieser Wert beschreibt die Häufigkeit der mfe-Struktur in der Verteilung. Die Verteilung wird in Form eines Plots dargestellt (siehe Abbildung 2.5, S. 25).

2.4 RNALfold

Das Vienna Package wurde erweitert, um lokal stabile Strukturen in einem Genom zu finden. Es handelt sich dabei um die oben schon erwähnten miRNAs. Das Programm `RNALfold` wurde hinzugefügt. Mit dessen Parameter `-L` wird die Größe eines Fensters angegeben und mit diesem Fenster die maximale Anzahl an Nukleotiden in der lokal stabilen Struktur bestimmt.
Das Fenster wird dann Nukleotid für Nukleotid die Sequenz entlang geschoben (siehe Abbildung 2.7, S. 27), um bei jedem weiterschieben des Fensters die optimale Struktur zu berechnen. Um einen Energiewert (ΔG^0) für diese Struktur zu erhalten, müssen Energieparameter mit in die Berechnung einfließen. `RNALfold` wie `RNAfold` verwenden die selben Energieparameter (siehe Abbildung 2.8, S. 32 und Tabelle 2.2, S. 33).
Für jedes Fenster wird die optimale Struktur und der Energiewert berechnet. Die Energiewerte werden miteinander verglichen, wobei der niedrigste ΔG^0 Wert gespeichert und mit weiteren verglichen wird. Als Ergebnis erhält man eine Auswahl an lokal stabilen Strukturen mit möglichst geringen freien

Energiewerten.
Aus den lokal stabilen Strukturen könnte man die komplette mfe-Struktur der gesamten Sequenz rekonstruieren (Hofacker *et al.*, 2004)

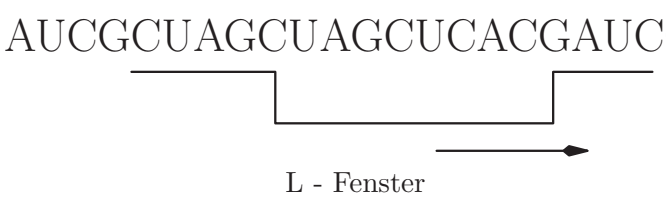


Abbildung 2.7: Hier ist ein Sequenz-Ausschnitt aus einem Genom zu sehen. Zusätzlich ist noch das `RNALfold`-Fenster zu sehen, dieses wird Nukleotid für Nukleotid durch das gesamte Genom geschoben. Bei jedem Schritt wird die mfe-Struktur für das aktuelle Fenster berechnet. So erhält man eine Auswahl an lokal stabilen Strukturen.

2.4.1 Maximierung der Basenpaaranzahl einer Sequenz

Für die Berechnung der optimalen Struktur im Lauffenster wurde der **MCMP** „maximum circular matching problem“ (Nussinov *et al.*, 1978) Algorithmus verwendet. Dieser Algorithmus aus der dynamischen Programmierung beruht auf dem Prinzip der Basenpaarmaximierung, das heißt der Algorithmus beruht nicht auf der Minimierung der freien Energie, sondern auf der Maximierung der Anzahl der Basenpaare. Bei der Realisierung macht man sich die Zerlegbarkeit des Problems in Teilprobleme zu nutze. Die Zerlegung in Teilprobleme hat den Vorteil, dass man nun in der Lage ist, große Genome in relativ kurzer Zeit nach ihren lokal stabilen Strukturen zu durchsuchen.

2.4.2 Graphendefinition der RNA-Sekundärstruktur

RNA-Sekundärstrukturen lassen sich als Graphen darstellen. Auf diesem Wege lassen sich Bedingungen für Basenpaarungen definieren, denn in Ter-

tiärstrukturen sind verschiedene Basenpaare erlaubt, die in Sekundärstrukturen verboten sind. Formale Bedingungen für Basenpaare:
Die Basen (i, j) sind, sowie (i', j') gepaart, dann gilt:

1.) $i < j < i' < j'$
2.) $i' < j' < i < j$
3.) $i < i' < j' < j$
4.) $i' < i < j < j'$

Diese vier Bedingungen bedeuten zusammengefasst, dass alle Loop-Typen (siehe Abbildung 1.1, S. 13) erlaubt sind, Pseudoknoten jedoch nicht zugelassen sind.

2.4.3 Struktur mit maximaler Zahl Basenpaare

Mit folgender Rekursion kann die maximale Anzahl an Basenpaaren (i, j) in einer Sequenz bestimmt werden, wobei x eine Sequenz mit der Länge L ist. Diese wird durch die Symbole $x_1, ..., x_L$ beschrieben.

$$\delta(i,j) = \begin{cases} 1, & \text{wenn } x_i \text{ und } x_j \text{ ein reguläres Basenpaar bilden können} \\ 0, & \text{sonst} \end{cases}$$

Reguläre Basenpaare sind Watson & Crick Basenpaare und das (A:U)-Wobble-Basenpaar. Mit der Rekursion werden „Scores“ $\gamma(i, j)$ berechnet. Dieser Score γ repräsentiert die maximale Anzahl an Basenpaaren innerhalb einer Subsequenz $x_i, ..., x_j$, die eine Länge von mindestens zwei Nukleotiden aufweisen muss. Desweiteren sind z.B. Hairpinloops einer Länge kleiner oder gleich fünf Nukleotiden energetisch ungünstig und deswegen sehr unwahrscheinlich. Ein Hairpinloop benötigt eine Mindestanzahl von drei Nukleotiden, um eine Helix überbrücken zu können. In der Regel haben Hairpinloops eine Größe von

fünf oder mehr Nukleotiden, eine Ausnahme bildet jedoch der extrastabilisierte Tetrahairpinloop mit vier Loop-Nukleotiden (Conn L., Draper E. 1998).

Initialisierung:

$$\begin{array}{lcll} \gamma(i,i-1) & = & 0 & \text{Für } i = 2 \text{ bis } L; \\ \gamma(i,i) & = & 0 & \text{Für } i = 1 \text{ bis } L. \end{array} \tag{2.1}$$

Rekursion:

$$\gamma(i,j) = max \begin{cases} \gamma(i+1,j), \\ \gamma(i,j-1), \\ \gamma(i+1,j-1) + \delta(i,j), \\ max_{i<k<j}[\gamma(i,k) + \gamma(k+1,j)]. \end{cases} \tag{2.2}$$

Die Tabelle 2.1, S. 30 beinhaltet eine maximale Summenmatrix. Die Matrix wurde nach der Rekursion 2.2 aufgefüllt. Die betrachtete Sequenz wird auf die x-Achse und die y-Achse aufgetragen.
Der Wert $\gamma(1, L)$ entspricht der maximalen Anzahl an Basenpaaren, in der betrachteten Struktur. Dieser Wert steht in der Matrix in der rechten oberen Ecke. Treten mehrere alternative Strukturen mit der gleichen Anzahl an Basenpaaren auf, so wird eine beliebige aus dieser Menge ausgewählt. Um eine der Strukturen mit maximaler Basenpaarung zu finden, wird ein optimaler Pfad durch die Summenmatrix gesucht. Der Pfad wird mit Hilfe eines „Traceback-Algorithmus“ gefunden.
Der Traceback-Algorithmus beginnt mit $\gamma(1, L)$, dem maximalen Wert. Wobei i für die x-Richtung und j für die y-Richtung steht. Es werden die Werte in unmittelbarer Nachbarschaft verglichen. Da in der rechten oberen Ecke

	G	G	G	A	A	A	U	C	C
G	0	0	0	0	0	0	**1**	**2**	**3**
G	0	0	0	0	0	0	**1**	**2**	**3**
G		0	0	0	0	0	**1**	**2**	**2**
A			0	0	0	0	**1**	**1**	**1**
A				0	0	0	**1**	**1**	**1**
A					0	0	**1**	**1**	**1**
U						**0**	0	0	0
C							0	0	0
C								0	0

Tabelle 2.1: Hier ist eine maximale Summenmatrix zu sehen. Die Matrix wurde nach der Rekursion 2.2 aufgefüllt. Der Wert „oben rechts“ gibt die maximale Anzahl der möglichen Basenpaare an. Der „Traceback“ ist in rot dargestellt und gibt den optimalen Pfad durch die Matrix an.

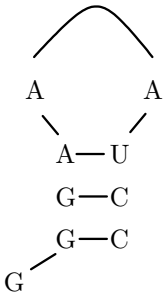
begonnen wird, gibt es drei mögliche Positionen, die mit einem Schritt erreicht werden können. $(i-1, j)$ steht für einen Schritt nach links, $(i, j-1)$ steht für einen Schritt nach unten und in dem Fall $(i-1, j-1)$ bewegt man sich diagonal.

Nun beginnt man, den Wert der Startposition mit den Werten der drei möglichen Nachbarn zu vergleichen. Als erstes werden die Werte von Position $(1, L)$ und $(i, j-1)$ verglichen. Sind diese Werte gleich, so erfolgen keine weiteren Vergleiche. Die Stelle $(i, j-1)$ ist nun die Startposition und ein Durchlauf ist beendet.

Von der neuen Startposition beginnt der nächste Durchlauf mit wiederum drei möglichen Nachbarn. Als erstes wird der Wert unterhalb der Startposition verglichen. Ist er kleiner als der Wert an der Startposition, wird die Position $(i-1, j)$ verglichen, was einem Schritt nach rechts entspricht. Wenn der Wert an dieser Stelle gleich dem Wert der Startposition ist, erfolgen keine

weiteren Vergleiche. Die Position $(i-1, j)$ ist nun neue Startposition und ein weiterer Durchlauf ist beendet.
Von der neuen Startposition startet dann ein weiterer Durchlauf. Zunächst werden wieder die Werte an den Positionen $(i, j-1)$ und $(i-1, j)$ mit dem Wert an der Startposition verglichen. Sind diese Werte kleiner kommt es zum Vergleich der Position $(i-1, j-1)$, dies entspräche einem diagonalen Schritt. Ist dieser Wert ebenfalls kleiner oder gleich dem Wert der Startposition, so ist die Position $(i-1, j-1)$ neue Startposition und der Durchlauf ist beendet. Am Ende kann aus dem Pfad eine Struktur mit minimaler Anzahl an Basenpaaren zusammengesetzt werden.

Anzahl der Basenpaare:	3
Punkt-Klammer Notation:	`. ( ( ( . . ) ) )`
Sequenz:	`G G G A A A U C C`
Graph:	



Die Stacking-Wechselwirkung ist erfahrungsgemäß nur vom unmittelbar nächsten Nachbarn abhängig. Daraus ergibt sich, dass für die Basenpaare {C:G, A:U, G:U} 12 unterschiedliche Werte benötigt werden. Auf ein beliebiges Basenpaar kann jedes der drei möglichen Paare {C:G, A:U, G:U} gestapelt werden. Desweiteren ist auch die umgekehrte Basenanordnung {G:C, U:A, U:G} möglich, was zusammen sechs Werte ergibt. Nun existieren zwei Stapelseiten für das betrachtete Basenpaar. So entstehen 12 unterschiedliche

Energieparameter.

Zusätzlich werden noch Energiewerte für die verschiedenen Looptypen benötigt.

$$5' \longrightarrow 3'$$
$$CX$$
$$GY$$
$$3' \longrightarrow 5'$$

	A	C	G	U
A	.	.	.	-2,1
C	.	.	-3,3	.
G	.	-2,4	.	-1,4
U	-2,1	.	-2,1	.

Abbildung 2.8: In der Tabelle sind die Energiewerte für das Stacking angegeben. Auf das Basenpaar (C:G) kann ein weiteres beliebiges Paar gestapelt werden. Aus der Tabelle kann der entsprechende Energiewert x-vertikal und y-horizontal abgelesen werden. Die Energiewerte sind in $kcal/mol$ angegeben. (Zuker A. M., Mathews B. D. H., Turner C. D. H. 1999.)

Die Stabilität eines Hairpinloops hängt außerdem noch vom letzten regulär gepaarten Basenpaar und von der Loopsequenz ab. Die Sequenz GGGGAC beschreibt beispielsweise einen Tetrahairpinloop und hat eine Energie von $\Delta G = -3,0\ kcal/mol$ (Zuker und Turner 1999).

Größe	Internal Loop	Bulge Loop	Hairpin Loop
1	-	3,8	-
2	-	2,8	-
3	-	3,2	5,6
4	1,7	3,6	5,5
5	1,8	4,0	5,6
6	2,0	4,4	5,3
7	2,2	4,6	5,8
8	2,3	4,7	5,4
⋮	⋮	⋮	⋮
30	3,7	6,1	7,7

Tabelle 2.2: In der Tabelle sind Energiewerte verschiedener Looptypen eingetragen. Die Energiewerte sind in *kcal/mol* angegeben. Diese Parameter wurden experimentell gemessen und für größere Loops extrapoliert. (Zuker und Turner 1999.)

2.5 Problematik der Fenstermethode

Das Programm `RNALfold` arbeitet mit einem speziellen Parameter `-L`, mit welchem eine Fenstergröße festgesetzt werden kann. Das Fenster wird Nukleotid für Nukleotid entlang der Sequenz geschoben und bei jedem Schritt eine Struktur berechnet. Am Ende erhält man eine Aufstellung mit lokal stabilen Strukturen.

Problematisch ist diese Technik, weil unter bestimmten Umständen nicht alle Strukturen berücksichtigt werden. Würde man zum Beispiel ein zu kleines `-L`-Fenster wählen, können größere zusammenhängende Strukturen nicht korrekt berechnet werden. Ist das Fenster jedoch zu groß gewählt, können unter Umständen mehrere kürzere Strukturen nicht mehr differenziert werden. Die optimale Einstellung des `-L` Parameters ist entscheidend für das Auffinden von lokal stabilen Strukturen mit dem Programm `RNALfold`.

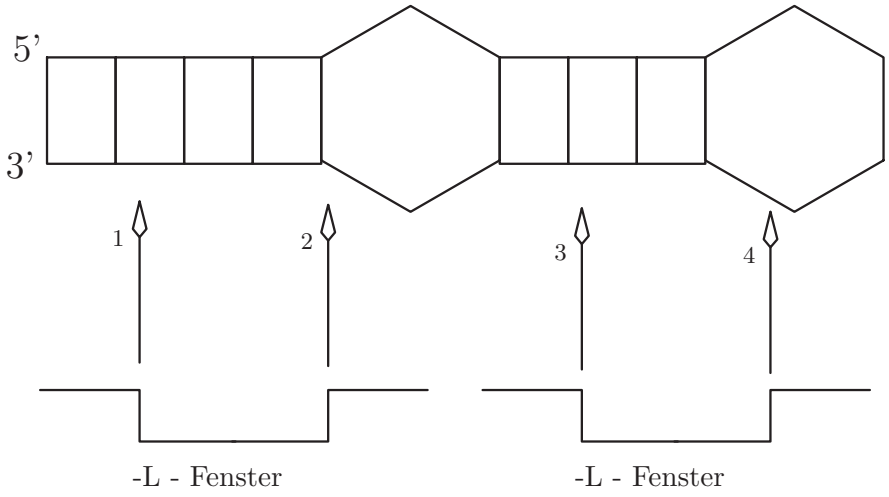


Abbildung 2.9: Hier ist ein längeres Strukturelement dargestellt. Das -L-Fenster ist in diesem Fall zu klein gewählt, da das gesuchte Motiv (Internalloop und Hairpinloop) größer als das -L-Fenster sind. Daher kann es nicht als zusammenhängendes Element berechnet werden.

In unserem Fall ist ein positives bzw. negatives Beispiel 3000 Nukleotide lang und das Fenster von `RNALfold` hat eine Spanne von 100 Nukleotiden. Das Programm fängt an Position eins an und endet an der Position 3000. Das Genom ist aber in Wirklichkeit um ein vielfaches länger, das heißt, dass hier nur ein Ausschnitt aus dem Genom betrachtet wird. Strukturen, die am Anfang des Ausschnittes bzw. am Ende des Ausschnittes vorhergesagt werden, sind unter Umständen nicht korrekt, denn Basen an den Randpositionen können zu Strukturen außerhalb des betrachteten Ausschnitts gehören.
Das Problem kann umgangen werden, in dem ein ausreichend großer Rahmen um die relevante Stelle im Genom gewählt wird. In diesem Fall paaren die Basen am Rand des Ausschnittes ebenfalls fälschlicherweise mit Basen innerhalb des betrachteten Ausschnittes, aber sie paaren nicht mit Nukleotiden, die zu den relevanten Stellen im Genom gehören (Stadler *et al.*, 2004).

2.6 RNAsubopt

In der Natur muss sich nicht zwangsläufig die mfe-Struktur ausbilden, vielmehr liegt in den meisten Fällen eine RNA-Strukturverteilung vor. Es können mehrere andere Moleküle an den RNA-Strang gebunden sein, so dass er sich nicht in die optimale Struktur falten kann, was durchaus auch einen biologischen Sinn haben kann. Wenn sich die Umgebungsparameter geringfügig ändern, kann der RNA-Strang eine andere Struktur annehmen und somit auch eine andere Funktion ausführen (z.B. Ribozyme).
Desweiteren gibt es Sequenzen, die verschiedene Strukturen mit ähnlicher Energie ausbilden können, das heißt die mfe-Struktur und suboptimale Strukturen mit ähnlich niedriger freier Energie wie die mfe-Struktur.

```
I   = GGCCCCUUUGGGGGCCAGACCCCUAAAGGGGUC
S0  = ((((((((((((((.....))))))))))))))
S1  = ((((((....)))))).((((((....))))))
```

Abbildung 2.10: Hier sieht man eine speziell generierte Sequenz, die zwei Strukturen mit ähnlichen Energiewerten ausbilden kann. ($S_0 = 26,30$ *kcal/mol* mfe-Struktur und $S_1 = 25,30$ *kcal/mol* suboptimale Struktur mir ähnlicher freier Energie).

Die mfe-Struktur S_0 (siehe Abbildung 2.10) besteht aus einem langen Hairpin, während die suboptimale Struktur S_1 mit ähnlich niedriger Energie aus zwei kurzen Hairpins besteht. Bei der Struktur S_1 handelt es sich um einen langlebigen metastabilen Zustand. Bei der Umlagerung in die mfe-Struktur müssten energetisch ungünstige Zustände (Sattelpunkte) durchlaufen werden. Um die Sattelpunkte zu erreichen, wäre eine Aktivierungsenergie notwendig, also ist die Struktur in einer Energiefalle gefangen.
Die Struktur S_1 gelangt in diesen langlebigen metastabilen Zustand, weil sie an zwei verschiedenen Nukleationszentren beginnen kann, eine Struktur auszubilden. Die mfe-Struktur S_0 kann nur an einem Nukleationszentrum mit

der Ausbildung einer Struktur beginnen. Dies kann der Grund dafür sein, dass in Lösung die Struktur S_1 zweimal öfter vorliegt, als die Struktur S_0 (Flamm *et al.*, 2000).

Zur Vorhersage von suboptimalen RNA-Sekundärstrukturen erweiterte S. Wuchty den Zuker-Algorithmus und implementierte ihn in das Programm `RNAsubopt` des Vienna Packages (Wuchty *et al.*, 1999).

Das Ziel des Wuchty-Algorithmus ist das Auffinden von möglichst allen suboptimalen RNA-Sekundärstrukturen innerhalb eines Energierahmens. Der Energierahmen wird mit dem Parameter `-e` eingestellt und bezieht sich auf den Energiewert der mfe-Struktur. Der Wert ist angegeben in $kcal/mol$ und gibt den Abstand zu dem Energiewert der mfe-Struktur wieder. In dem festgestellten Rahmen werden dann alle suboptimalen Strukturen berechnet. Die Ausgabe erfolgt in der üblichen Punkt-Klammer-Notation. Die Zahl der möglichen Strukturen wächst exponentiell. In der folgenden Tabelle 2.3 ist die Zahl der möglichen Strukturen im Verhältnis zu der Länge der Sequenz und dem Energierahmen angegeben.

Energierahmen-e	**5**	**10**	**12**	**15**	**17**	**20**
Sequenz Länge						
25	17	187	441	1299	2569	6048
50	9	108	254	900	2178	6477
75	86	1664	5056	24,299	67,601	295,722
100	121	4439	16,567	103,935	341,054	1,864,633

Tabelle 2.3: In der Tabelle ist die exponentielle Zunahme vorhergesagter suboptimaler Strukturen in Abhängigkeit von Sequenzlänge und Energierahmen angegeben. Der Energierahmen gibt die Differenz zur mfe-Struktur in $kcal/mol$ an.

Eine Möglichkeit, die gefundenen suboptimalen RNA-Sekundärstrukturen zu limitieren, bietet der Parameter `-p`. Damit kann die Anzahl an Strukturen, die ausgegeben werden sollen, festgelegt werden. Es wird dann eine Anzahl

an zufällig ausgewählten suboptimalen Strukturen ausgegeben.

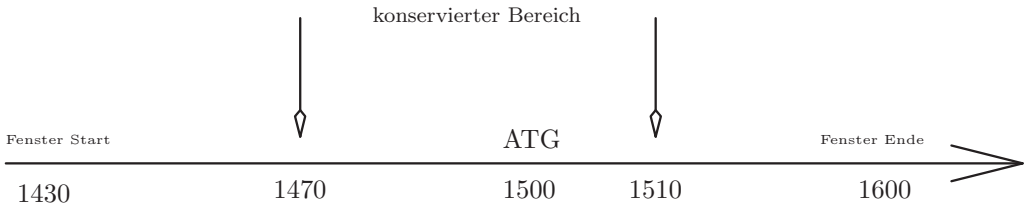
2.7 Konservierte Positionen

Mit dem Programm `RNALfold` wurden jeweils die positiven und die negativen Beispiele nach lokal stabilen Strukturen untersucht. Dabei wurde der Parameter-`-L` auf 100 Nukleotide eingestellt.
`RNALfold` gibt, neben den lokal stabilen Strukturen in der Punkt-Klammer-Notation, noch den Startpunkt der gefundenen Struktur an. Dazu wird auch noch der ΔG^0-Wert in $kcal/mol$ angegeben. Wie lang die gefundenen Strukturen sind, hängt vom Parameter `-L` ab. Er wird beim Aufruf des Programms übergeben und damit die Länge der Sequenz bzw. auch die Länge der Struktur festgelegt. Kürzere Strukturen sind natürlich möglich.
Die konservierten Positionen sind die TIS und die Shine-Dalgarno Sequenz. Es wird eine Struktur gesucht, die diese Positionen beinhaltet. Bei den positiven Beispielen befindet sich die „wahre“ TIS an der Stelle 1500 und die Shine-Dalgarno Sequenz ungefähr an Position 1480 - 1490. Es werden, da die genauen Startpositionen der Strukturen bekannt sind, zusätzlich auch noch die zu den Strukturen gehörenden Sequenzen ausgegeben, welche als Eingabe für andere Programme dienen.

2.8 Kontaktpunkte der ersten Klassifikation

Mit dem Programm `RNAfold` kann für eine Sequenz die mfe-Struktur berechnet werden. Mit dem Parameter `-p` wird zusätzlich noch die Partition function berechnet. Wie unter Punkt 2.3, S. 22 beschrieben, wird eine Basenpaarwahrscheinlichkeit in Form einer Pseudo-Bracket-Notation und einer Basenpaar Wahrscheinlichkeitsmatrix angegeben.
Als Eingabe bekommt das Programm `RNAfold` eine Sequenz übergeben. Sie

stammt aus einer Struktur, die mit dem Programm `RNALfold` vorhergesagt wurde. Die konservierten Positionen sind die TIS und die Shine-Dalgarno Sequenz. Es wurden Strukturen ausgesucht, die diese Positionen beinhalten, weil untersucht werden sollte, ob sich Hinweise auf konservierte RNA-Sekundärstrukturen in diesem Bereich vorhersagen lassen.
Der mögliche konservierte Bereich beginnt bei der Position 1470 und endet bei Position 1510. Die TIS befindet sich bei einem positiven Beispiel an der Position 1500. Das betrachtete Fenster beginnt an der Position 1430 und endet an der Position 1600. Folglich wird ein 170 Nukleotide langes Fenster betrachtet.
Strukturen, die in diesem Fenster von dem Programm `RNALfold` vorhergesagt wurden, sollten eine möglichst große Überlappung zwischen den Positionen 1470 und 1510 haben. In dem Bereich 1470 bis 1510 befindet sich der Genstart.
Um eine möglichst große Überlappung zu gewährleisten, wurden bevorzugt Strukturen benutzt, die bei der Position 1470 ihren Startpunkt hatten. Gab es jedoch keine durch das Programm `RNALfold` vorhergesagte Struktur, die ihren Startpunkt bei 1470 hatte, wurde geprüft, ob eine Struktur vorhergesagt wurde, die ihren Startpunkt an der Position 1471 hattte. Falls das nicht der Fall war, wurde die Position 1469 untersucht u.s.w., bis schließlich ein



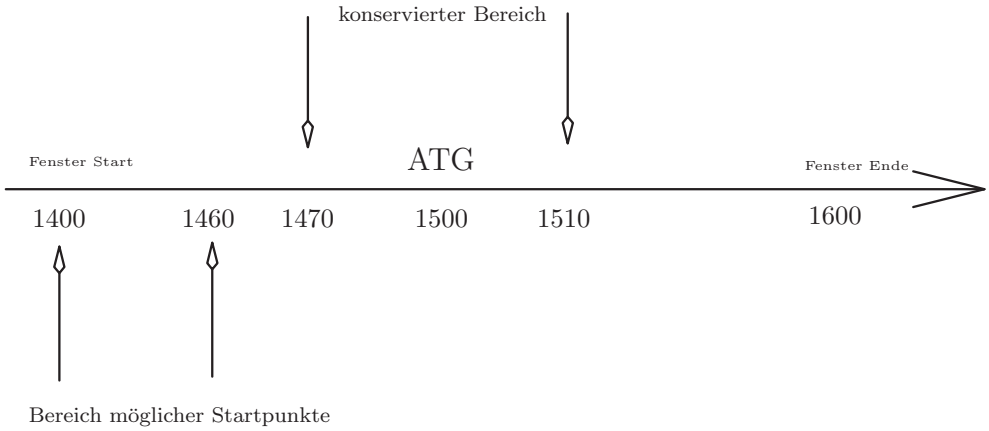
Startpunkt einer lokal stabilen Struktur gefunden wurde.
Zusätzlich zur mfe-Struktur wurden Basenpaarungswahrscheinlichkeiten mit der Partition function berechnet. Aus der Pseudo-Bracket-Notation können

wahrscheinliche Basenpaarungen von weniger wahrscheinlichen Basenpaarungen unterschieden werden. Desweiteren wurden nur Basenpaarungen betrachtet, die eine hohe Paarungswahrscheinlichkeit hatten. Zusätzlich musste auch noch eine eindeutige Basenpaarungsorientierung vorhergesagt werden. Basen, die von dem Programm `RNAfold` mit einem „|“ charakterisiert wurden, wurden vernachlässigt. Dies hatte zur Folge, dass der Datensatz um ca. 25 - 30% verkleinert wurde.
Hier wurden also die Ergebnisse von `RNALfold` und der Partition function von `RNAfold` verknüpft. Anschließend wurden Kontaktpunkte aus den wahrscheinlichsten Basenpaaren berechnet. Weniger wahrscheinliche Basenpaarungen wurden nicht in Kontaktpunkte siehe Abschnitt 2.15.1, S. 48 übersetzt.

2.9 Kontaktpunkte der zweiten Klassifikation

Mit dem Programm `RNAsubopt` können alle suboptimalen RNA-Sekundärstrukturen berechnet werden. Als Eingabe bekommt das Programm eine Sequenz übergeben.
Die Sequenz stammt von einer Struktur, die mit Hilfe des Programms `RNALfold` berechnet wurde. Mit den in Abschnitt 2.6, S. 35 beschriebenen Parametern ist es möglich, eine bestimmte Anzahl oder alle suboptimalen RNA-Sekundärstrukturen zu berechnen.
Mit dem Programm `RNAsubopt` wurden pro Sequenz 100 suboptimale RNA-Sekundärstrukturen berechnet. Der Energierahmen, in dem sich die suboptimalen RNA-Sekundärstrukturen von der mfe-Struktur unterscheiden durften, lag bei 10 $kcal/mol$.
Der Unterschied bei der Verwendung des Programms `RNAfold` in Abschnitt 2.8, S. 37 und des Programms `RNAsubopt` lag darin, dass versucht wurde,

möglichst lange Strukturen, die vom Programm `RNALfold` vorhergesagt wurden, zu benutzen.
Hierzu wurde ein 200 Nukleotide langes Fenster betrachtet. Das betrachtete Fenster beginnt an der Position 1400 und endet an der Position 1600. Ein Genstart befindet sich bei den positiven Beispielen an der Position 1500. Das entspricht der Position 100 in dem betrachteten Fenster. Der mögliche konservierte Bereich umfasst die Position 1470 - 1510.
Strukturen, die in diesem Fenster vom dem Programm `RNALfold` vorhergesagt wurden, durften an der Position 1400 - 1460 ihren Startpunkt haben. Denn auch sie sollten eine möglichst große Überlappung mit den Positionen 1470 - 1510 haben. Die maximale Strukturlänge (Parameter `-L`) lag bei 100 Nukleotiden.
Das Programm `RNALfold` sagte für das betrachtete Fenster mehrere Strukturen vorher. Es wurde die längste Struktur ermittelt, die ihren Startpunkt in dem Bereich 1400 - 1460 hatte. Die Sequenz dieser Struktur diente als Eingabe für das Programm `RNAsubopt`. Auf diese Weise wurden 722 Sequenzen



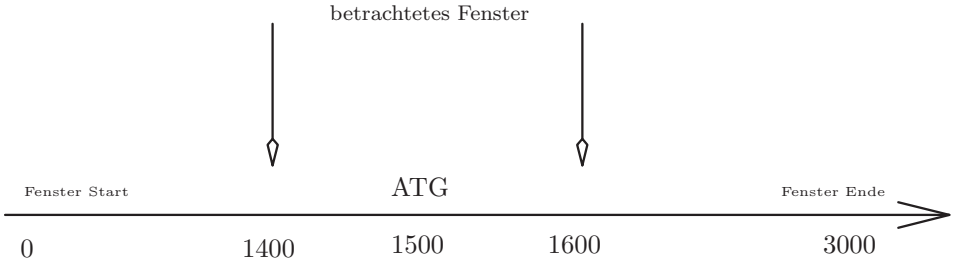
für die positiven und 854 Sequenzen für die negativen Beispiele ermittelt.
Pro Sequenz wurden 100 suboptimale RNA-Sekundärstrukturen berechnet, welche in der Punkt-Klammer-Notation ausgegeben wurden. Von den 100 suboptimalen Strukturen wurde die Struktur ausgewählt, die den höchsten

Anteil an konservierten Basenpaaren besitzt.
Hierzu wurde ein „Score“ definiert. Ein Basenpaar, welches zweimal in den 100 suboptimalen Strukturen vorkam, erhielt den Score zwei. Ein Basenpaar, welches 18 mal in den 100 suboptimalen Strukturen vorkam erhielt den Score 18. Für jedes vorkommen eines Basenpaares wurde der Score um eins erhöht. Es ist also ein maximaler Score von 100 pro Basenpaar möglich gewesen. Dies wäre möglich, wenn ein Basenpaar in allen 100 suboptimalen Strukturen vorkommen würde. Der Score entspricht der absoluten Häufigkeit in den 100 suboptimalen RNA-Sekundärstrukturen.
Für die Bewertung einer suboptimalen RNA-Sekundärstruktur wurde der Score von jedem Basenpaar in dieser suboptimalen Struktur zusammenaddiert (K-Score). Es wurden 100 K-Score Werte für 100 suboptimale Strukturen berechnet, die Struktur mit dem höchsten K-Score Wert ausgewählt und dies für alle positiven Beispiele wiederholt, so dass für jede Sequenz wieder eine Struktur vorlag. Für die positiven Beispiele wurden 722 suboptimale Strukturen und für die negativen Beispielen analog 854 suboptimale Strukturen ermittelt.

2.10 Kontaktpunkte der dritten Klassifikation

Mit dem Programm `RNALfold` wurde nach lokal stabilen RNA-Sekundärstrukturen innerhalb der positiven und negativen Beispiele gesucht. Als Eingabe bekam das Programm eine 3000 Nukleotide lange Sequenz übergeben. Eine „wahre“ TIS ist bei den positiven Beispielen an der Position 1500 lokalisiert. Bei den negativen Beispielen ist eine „falsche“ TIS wie, schon in Abschnitt 2.1.2, S. 20 beschrieben, an der Position 1500 lokalisiert.
Das Programm `RNALfold` sagt über den gesamten Bereich (0-3000) lokal stabile RNA-Sekundärstrukturen vorher. Zusätzlich wird auch der Startpunkt

der vorhergesagten RNA-Sekundärstruktur ausgegeben. Der Parameter `-L` wurde auf 100 Nukleotide eingestellt (siehe Abbildung 2.5, S. 33).



Desweiteren wurde ein 200 Nukleotide langes Fenster betrachtet. Das Fenster hatte einen Startpunkt an der Position 1400 und einen Endpunkt an der Position 1600. Eine vorhergesagte RNA-Sekundärstruktur konnte ihren möglichen Startpunkt in dem betrachteten Fenster haben. Der dazugehörige Endpunkt der vorhergesagten RNA-Sekundärstruktur durfte nicht außerhalb des betrachteten Fensters liegen. Unter diesen Voraussetzungen wurden mehrere RNA-Sekundärstrukturen pro Beispiel von dem Programm `RNALfold` vorhergesagt.

2.11 Erzeugung von Zufallssequenzen

Für die Erzeugung von Zufallssequenzen wurden die Sequenzen der Negativbeispiele zufällig permutiert. Unter Permutation wird die mögliche Anordnung von Elementen, in unserem Fall sind es Nukleotide, ohne Wiederholung verstanden. Die Anzahl der Permutationen P von n Elementen ist $P(n) = n!$. Dies gibt die Anzahl aller möglichen Anordnungen von Nukleotiden innerhalb der Sequenz wieder. Die Anzahl der unterschiedlichen Nukleotide in der Sequenz bleibt dabei erhalten, lediglich die Position der Nukleotide wird verändert.
Somit sind bei den Zufallssequenzen alle Informationen, die in den Negativbeispielen vorhanden waren, gelöscht worden. In unserem Fall gab es 3000!

mögliche Kombinationen, die Nukleotide in einer einzelnen Sequenz anzuordnen.
Die permutierten Sequenzen wurden als Eingabe für das Programm `RNALfold` verwendet, womit nach lokal stabilen Strukturen innerhalb dieser Sequenzen gesucht werden sollte. Die Ergebnisse dienten einmal als Eingabe für das Programm `RNAfold` und ein zweites Mal als Eingabe für das Programm `RNAsubopt`. Die weitere Vorgehensweise ist analog zu der in Abschnitt 2.8, S. 37 und Abschnitt 2.9, S. 39.

2.12 Histogramm

Ein Problem ist die Bestimmung der Verteilung von gegebenen Zufallsvariablen. Beispielsweise ist die Augenzahl eines Würfels eine Zufallsvariable, denn sie kann die Werte {1,2,3,4,5,6} annehmen. Die Zufallsvariable heißt in diesem Fall diskret, weil die einzelnen Werte, die die Zufallsvariable annehmen kann, aus einer abgezählten Menge stammen. Die Wahrscheinlichkeiten der möglichen Werte einer diskreten Zufallsvariablen bilden eine Wahrscheinlichkeitsverteilung.
Den möglichen Werten kontinuierlicher Zufallsvariablen können dagegen keine Wahrscheinlichkeiten zugeordnet werden. Um eine Wahrscheinlichkeitsverteilung zu betrachten, können Dichteschätzer verwendet werden. Einer der einfachsten Dichteschätzer ist das Histogramm. Es ist „*die Darstellung der Häufigkeiten klassierter Daten einer stetigen Zufallsvariablen*“. (Thadewald *et al.*, 1998)
Ein Histogramm ist eine graphische Darstellung von Daten, die in Abschnitte bzw. Intervalle zerlegt werden, wobei die Intervalle alle die gleiche Breite h haben. Dadurch werden die kontinuierlichen Werte diskret dargestellt. Formal kann ein Histogramm so beschrieben werden:

$$\begin{aligned} \hat{f}(x) &= \frac{1}{nh} * X_i \\ &= \frac{1}{nh} \sum_{i=1}^{k} n_i I_i(x) \end{aligned} \tag{2.3}$$

$$\text{mit } I_i(x) = \begin{cases} 1, & \text{falls } x \text{ in dem } i \text{ -ten Intervall liegt} \\ 0, & \text{sonst.} \end{cases}$$

$I_i(x)$: Indikatorfunktion

h: Inervallbreite

n: Flächeninhalt

In einem Histogramm wird für jede Beobachtung ein Block mit der Fläche $1/n$ und der Breite h auf die Intervallmitte gestapelt. Liegen z.B. zwei Beobachtungen in einem Intervall, so stapeln sich zwei Blöcke über die Intervallmitte, in der die Beobachtungen liegen. Der Flächeninhalt der gesamten gestapelten Blöcke über einem Intervall gibt dann die relative Häufigkeit an.

2.13 Vom Histogramm zum Kerndichteschätzer

Ein etwas besserer Dichteschätzer könnte so aussehen, dass die Blöcke nicht einfach auf die Intervallmitte gestapelt werden, sondern dass die Blöcke direkt auf die Beobachtung selbst stapeln. Dadurch kommt es zu Überschneidungen.

Danach wird eine Kurve über die äußeren Konturen der Blöcke gelegt, wodurch eine bessere Dichteschätzung als mit einem Histogramm erhalten wird. Für die Abschätzung einer stetigen Dichtefunktion ist dieses Verfahren jedoch immer noch nicht optimal. Deswegen werden die Blöcke durch „stetige-Funktionen“ ersetzt, welche auf jede Beobachtung platziert werden. Dies ist das Prinzip des Kerndichteschätzers.

2.14 Kerndichteschätzer

Die Benutzung von Kernfunktionen an Stelle von Blöcken ist die Idee des Kerndichteschätzers. Die Kernfunktionen werden durch eine Funktion K beschrieben. Alle Kernfunktionen haben in der Regel Gemeinsamkeiten. Sie sind alle um einen bestimmten Wert symmetrisch. Dies kann zum Beispiel der Ursprung sein. Die Unterschiedlichen Kernfunktionen liefern somit alle ähnliche Ergebnisse. Über jede Beobachtung wird eine Kernfunktion gelegt. Ein Kerndichteschätzer kann formal so beschrieben werden:

$$\hat{f}(x) = \frac{1}{nh} \sum_{i=1}^{n} K\left(\frac{x - X_i}{h}\right) \tag{2.4}$$

K: Kernfunktion
h: Bandbreite
x: Beobachtung
X_i: Anzahl der Beobachtungen

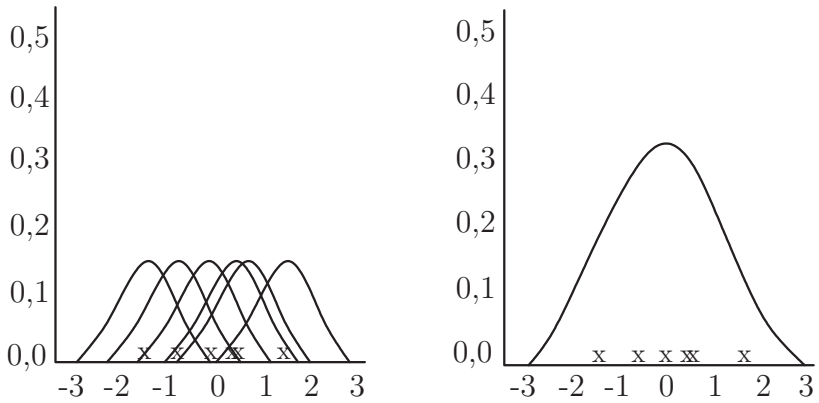


Abbildung 2.11: Hier sieht man eine Kerndichteschätzerfunktion. Auf jede Beobachtung (x) ist eine „Kernfunktion“ gelegt worden. In der zweiten Abbildung wurden die Kernfunktionen addiert (Thadewald *et al.*, 1998).

2.14.1 Arten von Kernfunktionen

Es gibt viele mögliche Kernfunktionen. Sie müssen die Eigenschaften einer Dichtefunktion erfüllen. Dies bedeutet, das Integral der Kernfunktion muss eins sein:

$$\int_{-\infty}^{+\infty} K(x)dx = 1 \text{ und } K(x) \geq 0. \tag{2.5}$$

In der Regel sind sie um Null symmetrisch. Beispiele oft verwendeter Kernfunktionen sind die Dichte der Standardnormalverteilung, auch Gauss-Kern genannt, desweiteren der Epanechnikov-Kern, die Dreiecksdichte, oder der Rechteckskern:

Gauss-Kern: $\varphi(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2}$

Epanechnikov-Kern: $e(x) = \begin{cases} \frac{3}{4\sqrt{5}}\left(1 - \frac{1}{5}x^2\right) & \text{für } x^2 < 5 \\ 0 & \text{sonst.} \end{cases}$

Dreiecksdichte: $d(x) = \begin{cases} 1 - |x| & \text{für } |x| < 1 \\ 0 & \text{sonst.} \end{cases}$

Rechteckskern: $r(x) = \begin{cases} \frac{1}{2} & \text{für } |x| < 1 \\ 0 & \text{sonst.} \end{cases}$

Wenn man die verschiedenen Kernfunktionen nutzt, um die Dichte einer Menge zu schätzen, liefern, mit Ausnahme des Rechteckskerns, alle Kernfunktionen ähnliche Ergebnisse. Dieser liefert in der Regel ein etwas „raueres“ Ergebnis (Hafner *et al.*, 2001). Die Wahl der Kernfunktion spielt jedoch eine vergleichsweise geringe Rolle im Gegensatz zur Wahl des Parameters h. Der Parameter h wird auch als Bandbreite oder Glättungsparameter bezeichnet.

2.14.2 Einfluss der Bandbreite

Bei Kerndichteschätzern spielt die Wahl der Bandbreite h eine zentrale Rolle. Das Ergebnis hängt von der Wahl der Kernbreite ab. Für kleine h zeigt die Dichteschätzung eine raue Kurve, für große Glättungsparameter wird die Dichteschätzung glatt.
Wenn h zu groß gewählt wird, kann es zum so genannten Überglätten (Oversmoothing) kommen, was heißt, dass die Verteilung zu sehr geglättet wird und möglicherweise wichtige Strukturen verloren gehen. Dementsprechend kommt es zum Unterglätten (Undersmoothing), wenn h zu klein gewählt wird. Lokale Gegebenheiten der Daten haben in diesem Fall einen zu großen Einfluss auf den Verlauf der Dichtekurve (Härdle und Müller 1993).
Das einfachste Verfahren, das Over- oder Undersmoothing zu umgehen, ist die Wahl des Glättungsparameters nach Augenmaß (smoothing by eye). Hierzu werden verschiedene Bandbreiten ausgewählt und die unterschiedlichen Graphen betrachtet. Darauf wird ein h ausgewählt, bei dem die Dichtefunktion als am sinnreichsten erachtet wird. Glättung nach Augenmaß kann durchaus gute Ergebnisse liefern. Bei vielen Anwendungen wird aber ein automatisiertes Verfahren zur Bestimmung der Bandbreite, wenn z.B. mehrere verschiedene Datensätze analysiert werden sollen, benötigt. Dabei müssen unter Umständen sehr viele Bandbreiten bestimmt werden, folglich sind dann automatisierte Verfahren um ein vielfaches effektiver.

2.15 Die Matrixform der RNA-Struktur

RNA-Strukturen können in Form von Kontaktmatrizen dargestellt werden. Diese sind wie folgt definiert: Seien $\mathbf{M}_i = \mathbf{M}_1 ... \mathbf{M}_n$ die Kontaktmatrizen der RNA-Strukturen von Sequenzen der Länge d, für die gilt:

$$\mathbf{M}_i(t_1, t_2) = \begin{cases} 1, & \text{falls die Base } t_1 \text{ mit der Base } t_2 \text{ der Sequenz } i \text{ gepaart ist.} \\ 0, & \text{sonst} \end{cases}$$

n	Anzahl aller Sequenzen
d	Länge der Sequenzen
$i = 1...n$	Index der Sequenzen
$t_1, t_2 = 1...d$	Positionen innerhalb der Sequenzen
	t_1 : Zeilenindizes der Kontaktmatrix $\mathbf{M}$
	t_2 : Spaltenindizes der Kontaktmatrix $\mathbf{M}$

Die Positionen einer Sequenz werden also zeilen- und spaltenweise in eine $d \times d$ - Matrix eingetragen. Bildet die Base an einer Position t_1 mit der an einer Position t_2 liegenden Base ein Basenpaar aus, so ist der Eintrag der Matrix $\mathbf{M}(t_1, t_2) = 1$. Sind die beiden Basen nicht gepaart, so ist $\mathbf{M}(t_1, t_2) = 0$.

2.15.1 Kontaktpunkte

Da nach der Definition der RNA-Sekundärstruktur jede Base maximal mit einer anderen Base gepaart ist, kann folglich maximal ein Eintag pro Zeile und pro Spalte der Kontaktmatrix 1 sein. Alle anderen Einträge sind 0. Weiterhin folgt daraus, dass die Matrizen symmetrisch sind. Es gilt also $\mathbf{M}_i(t_1, t_2) = \mathbf{M}_i(t_2, t_1)$. Der Effizienz halber werden statt der vollständigen Kontaktmatrizen $\mathbf{M}_i$ bei der Realisierung nur die nichtredundanten Mengen der Kontaktpunkte Z_i betrachtet, also die Indizes, für die gilt $\mathbf{M}_i(t_1, t_2) = 1$,

mit $t_1 < t_2$. Das heißt, dass nur die „obere“ Hälfte der Matrizen also alles, was oberhalb ihrer Hauptdiagonalen liegt, betrachtet wird.
Die Kontaktpunkte entsprechen den zweidimensionalen Vektoren $\vec{z} = \binom{t_1}{t_2}$, wobei t_1 den jeweiligen Zeilenindex des Kontaktpunktes und t_2 den Spaltenindex bezeichnet.

Darstellung einer zufällig erzeugten Kontaktmatrix:

$$\mathbf{M}_i = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mathbf{1} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mathbf{1} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mathbf{1} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathbf{1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \mathbf{1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \mathbf{1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \begin{matrix} \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \end{matrix} t_1$$

$$\begin{matrix} & & & & & & & & & & & \\ & & & & & & & & & & & \\ & & & & & & & & & & & \\ & & & & & & & & & & & \\ & & & & & & & & & & & \\ & & & & & & & & & & & \\ & & & & & & & & & & & \\ & & & & & & & & & & & \\ & & & & & & & & & & & \\ & & & & & & & & & & & \\ & & & & & & & & & & & \\ & & & & & & & t_2 & & & & \end{matrix}$$

Die gleiche Sequenz in Punkt - Klammer - Notation

. (((. . . .))) .

Darstellung in Form von Kontaktpunkten

Die Menge Z_i der Kontaktpunkte von der Sequenz i

$$Z_i = \left\{ \binom{2}{11}, \binom{3}{10}, \binom{4}{9} \right\},$$

mit den Kontaktpunkten $\vec{z}$ als Vektoren des zweidimensionalen Raums.

2.15.2 Matrixmultiplikation

Die Multiplikation einer $(m \times n)$-Matrix A mit einer $(n \times k)$-Matrix B ist nur definiert, wenn die Anzahl der Spalten der ersten Matrix mit der Anzahl der Zeilen der zweiten Matrix übereinstimmt. In diesem Fall erhält man durch Multiplikation beider Matrizen eine $(m \times k)$-Matrix als Ergebnis. Das heißt, dass die Ergebnismatrix die gleiche Anzahl Zeilen hat, wie die erste zu multiplizierende Matrix und die gleiche Anzahl Spalten, wie die zweite.
Die Multiplikation einer Zeile der ersten Matrix mit einer Spalte der zweiten Matrix kann auch als Multiplikation des entsprechenden Zeilenvektors aus der ersten Matrix mit dem entsprechenden Spaltenvektor aus der zweiten Matrix dargestellt werden (Skalarprodukt):

$$[x_{i1}, \ldots, x_{in}] \times \begin{bmatrix} y_{j1} \\ \vdots \\ y_{jn} \end{bmatrix} = [z_{i,j}]$$

Das Ergebnis einer solchen Multiplikation erhält man durch Addition der folgenden Produkte:

$$\sum_{l=1}^{n} x_{il} * y_{il} = z_{ij}$$

Das Element in der ersten Spalte des ersten Vektors wird also mit dem Element in der ersten Zeile des zweiten Vektors multipliziert, das Element in der zweiten Spalte des ersten Vektors mit dem Element in der zweiten Zeile des zweiten Vektors, usw.

2.15.3 Glättung der Kontaktmatrizen

In ähnlicher weise wie unter Abschnitt 2.14, S. 45 beschrieben wird über jede Beobachtung eine Kernfunktion gelegt. In unserem Fall ist eine Beobachtung ein Basenpaar einer RNA-Sekundärstruktur. Die Basenpaare bzw. die RNA-Sekundärstrukturen sind durch die jeweilige Kontaktmatrix repräsentiert. Mit Hilfe einer Glättungsmatrix ist es möglich eine Kernfunktion über

jede Beobachtung zu legen. Hierzu wird die Kontaktmatrix $\mathbf{M}_i$ mit einer Glättungsmatrix $\mathbf{G}$ multipliziert.

$$\mathbf{M}'_i = \mathbf{G}^\top \mathbf{M}_i \mathbf{G} \tag{2.6}$$

$\mathbf{G}^\top$: Transponierte Glättungsmatrix
$\mathbf{M}_i$: Kontaktmatrix
$\mathbf{G}$: Glättungsmatrix

Beispiel einer Glättungsmatrix:

3	**2**	**1**	0	0	0	0	0	0	0
2	**3**	**2**	**1**	0	0	0	0	0	0
1	**2**	**3**	**2**	**1**	0	0	0	0	0
0	**1**	**2**	**3**	**2**	**1**	0	0	0	0
0	0	**1**	**2**	**3**	**2**	**1**	0	0	0
0	0	0	**1**	**2**	**3**	**2**	**1**	0	0
0	0	0	0	**1**	**2**	**3**	**2**	**1**	0
0	0	0	0	0	**1**	**2**	**3**	**2**	**1**
0	0	0	0	0	0	**1**	**2**	**3**	**2**
0	0	0	0	0	0	0	**1**	**2**	**3**

Mit der transponierten Glättungsmatrix $\mathbf{G}^\top$ werden die Spalten der Kontaktmatrix $\mathbf{M}_i$ geglättet. Dies geschieht bei der ersten Multiplikation $\mathbf{G}^\top \mathbf{M}_i$. Bei der zweiten Multiplikation $\mathbf{M}_i \mathbf{G}$ werden die Zeilen der Kontaktmatrix geglättet. Wie schon in Abschnitt 2.15.2, S. 50 beschrieben, ist das Ergebnis einer Matrixmultiplikation wieder eine Matrix. In unserem Fall eine "geglättete Kontaktmatrix".

2.15.4 Merkmalsvektoren

Aus der „geglätteten Kontaktmatrix“ entsteht durch die *vec* Operation ein Merkmalsvektor. Definition der *vec* Operation:

$$\mathbf{M}_i = [\vec{m}_{i1}, \ldots, \vec{m}_{id}]$$

$$vec\,(\mathbf{M}_i) = \begin{bmatrix} \vec{m}_{i1} \\ \vec{m}_{i2} \\ \vdots \\ \vec{m}_{id} \end{bmatrix}$$

Ein Merkmalsvektor kann formal so beschrieben werden:

$$\vec{x}_i = vec\,(\mathbf{M}_i)$$

2.16 Aufteilung des Datensatzes für eine Klassifikation

Der vorhandene Datensatz wird in drei disjunkte Mengen aufgeteilt (Trainingsmenge, Validierungsmenge und Testmenge). Jede der drei Mengen besteht aus positiven und negativen Beispielen.
Es wurden 50 Läufe (Klassifikationen) mit jeweils einer Trainingsmenge, Validierungsmenge und einer Testmenge durchgeführt. Diese Mengen haben, wie in Abschnitt 2.16.4, S. 53 beschrieben, in jedem neuen Lauf die gleichen Anteile an positiven und negativen Beispielen. Bei jedem neuen Lauf wurden per Zufall die drei Mengen aus dem gesamten Datensatz neu bestimmt.

2.16.1 Trainingsmenge

Die Trainingsmenge wird verwendet, um einen linearen Klassifikator zu konstruieren. Das heißt es wird aus der Trainigsmenge mit Hilfe eines Lernalgorithmus eine Trennebene bzw. der Normalenvektor berechnet.

2.16.2 Validierungsmenge

Wie in Abschnitt 2.15.3, S. 50 beschrieben, werden die Kontaktmatrizen geglättet, was heißt, dass eine Kernfunktion über jede Beobachtung gelegt wird, in unserem Fall über jedes Basenpaar.
Die „Stärke“ der Glättung hängt von dem Hyperparameter oder Glättungsparameter ab, der die „Breite“ der verwendeten Kernfunktion bestimmt. Bei der Dichtefunktion der Standardnormalverteilung (Gauss-Kern) bestimmt der Parameter σ, wie stark die Funktion gestaucht wird. Wird der Gauss-Kern als Kernfunktion benutzt, ist σ der Glättungsparameter.
Welches σ die besten Ergebnisse bei einer anschließenden Klassifikation liefert, kann nicht vorab berechnet werden. Mit den Elementen der Validierungsmenge werden verschiedene Glättungsparameter ausprobiert. Der Parameter, der die beste Klassifikation auf der Validierungsmenge liefert, wird für die Klassifikation auf der Testmenge verwendet.

2.16.3 Testmenge

Die Testmenge dient zur Abschätzung der Vorhersagegenauigkeit des Klassifikators. Auf der Testmenge wird der letztendliche Klassifikationsfehler bestimmt.

2.16.4 Training, Validierung, Test

Die Datenmenge enthielt 722 positive und 854 negative Beispiele. Es wurden jeweils 50 Läufe durchgeführt, bei denen die Datenmenge in drei disjunkte Gruppen aufgeteilt wurde.
Die Testmenge enthielt 40% der positiven und 40% der der negativen Beispiele. Jeweils 40% der restlichen Daten wurde zur Validierung verwendet. Die verbleibenden Daten dienten als Trainingsmenge. Eine genaue Aufteilung ist in der Tabelle 2.4, S. 54 zu sehen.

Testmenge	pos: $\texttt{round}(0,4 * 722) = 289$
	neg: $\texttt{round}(0,4 * 854) = 342$
Validierungsmenge	pos: $\texttt{round}(0,4 * (722 - 289)) = 173$
	neg: $\texttt{round}(0,4 * (854 - 342)) = 205$
Trainingsmenge	pos: $722 - 289 - 173 = 260$
	neg: $854 - 342 - 205 = 307$

Tabelle 2.4: Hier ist die genaue Aufteilung der drei Mengen (Trainings-, Validierungs- und Testmenge) zu sehen.

2.17 Maschinelles Lernen (ML)

Unter Maschinellem Lernen wird eine „künstliche“ Generierung von Wissen aus Beispielen verstanden. Man schließt dabei aus einer Reihe von beobachteten Eingangsgrößen auf bestimmte Ausgangsgrößen. Ein künstliches System

Eingangsgrößen	$\Longrightarrow$ $\Longrightarrow$ $\Longrightarrow$	„Blackbox“	$\Longrightarrow$ $\Longrightarrow$ $\Longrightarrow$	Ausgangsgrößen

lernt aus Beispielen und kann nach Beendigung der Lernphase verallgemeinern. Das bedeutet, es lernt nicht einfach die Beispiele auswendig, sondern es „erkennt“ Gesetzmäßigkeiten in den Lerndaten. So kann das System auch unbekannte Daten beurteilen, das heißt bei neuen Daten von den beobachteten Eingangsgrößen auf die nicht beobachteten Ausgangsgrößen schließen.

2.17.1 Erläuterung der Klassifikation

Ein Merkmalsvektor (feature vector) besteht aus Attributen. In unserem Fall ist ein Merkmalsvektor eine RNA-Sekundärstruktur und die Attribute sind Basenpaare.

Ein Klassifikator ordnet einem Merkmalsvektor $\vec{x}$ eine Klasse H^+ oder H^- zu. Das Label y kommt in der Lernphase zum Einsatz.
In der Lernphase (Erstellung eines Klassifikators) werden aus dem Datensatz zufällig einige Merkmalsvektoren ausgewählt und zu einer Trainingsmenge zusammengestellt. Zu jedem Trainingsobjekt muss in einem zusätzlichen Attribut die Klasse vorgegeben bzw. vermerkt werden, in die es gehört. Dies geschieht mit dem Label y. Im Allgemeinen wird dieses Lernverfahren als überwachtes Lernen (supervised learning) bezeichnet. Anhand der klassifizierten Trainingsdaten wird mittels eines Algorithmus ein Modell erstellt, das zu Merkmalskombinationen die zugehörige Klasse angeben kann. Dieses Modell wird als Klassifikator bezeichnet.

$$y \, \varepsilon \, \{-1, 1\} \tag{2.7}$$

$$c(\vec{x}) = \begin{cases} 1, \text{ falls } \vec{x} \ \varepsilon \ H^+ \\ -1, \text{ sonst} \end{cases} \tag{2.8}$$

Die Dimension des Merkmalsraumes, in dem sich alle Merkmalsvektoren befinden, wird durch die Anzahl von Merkmalen in den Merkmalsvektoren repräsentiert. Je größer die Anzahl an Merkmalen in dem Merkmalsvektor, desto größer die Dimension des Merkmalsraums. Die Dimension der Trennebene ist um eins kleiner als die Dimension des Merkmalsraumes. In einem zwei dimensionalen Merkmalsraum ist die Trennebene eine Gerade.
Im Idealfall bilden die Merkmalsvektoren der zu den Klassen gehörenden Objekte getrennte Punktwolken (Cluster) aus.
Im folgenden bezieht sich das Zeichen „$*$“ auf das Skalarprodukt. Das Skalarprodukt ist wie folgt definiert:

$$\vec{a} \cdot \vec{b} = a_1 \cdot b_1 + a_2 \cdot b_2 + \ldots + a_d \cdot b_d = \sum_{i=1}^{d} a_i * b_i$$

Die Normalform der oben erwähnten Trennebene, die den Merkmalsraum in zwei Halbräume aufteilt, hat die Form:

$$\vec{w} * \vec{p} + b = 0 \qquad \vec{w}, \vec{p}\,\varepsilon\,\mathbf{R}^d,\ b\,\varepsilon\,\mathbf{R}$$

das heißt für alle Punkte $\vec{p}$ der Ebene gilt:

$$\vec{w} * \vec{p} = -b$$

Wobei $\vec{w}$ die Richtung des Normalenvektors bezeichnet, der senkrecht (orthogonal) auf der Ebene steht (siehe Abbildung 2.12, S. 58).
Im Folgenden wird die Projektion von $\vec{a}$ auf $\vec{b}$ veranschaulicht.
Die Länge eines Vektors kann so beschreiben werden:

$$\|\vec{a}\| = \sqrt{\vec{a} * \vec{a}}$$

Einmal ist der Winkel φ zwischen den Vektoren größer 90° und einmal kleiner 90°.

$$\cos\varphi = \frac{\vec{a} * \vec{b}}{\|\vec{a}\|\|\vec{b}\|}$$

$$\Leftrightarrow \vec{a} * \vec{b} = \|\vec{a}\| * \|\vec{b}\| * \cos(\varphi)$$

Die Projektion von $\vec{a}$ auf $\vec{b}$ wird im Allgemeinen so ausgedrückt ($\vec{a}_{\vec{b}}$).

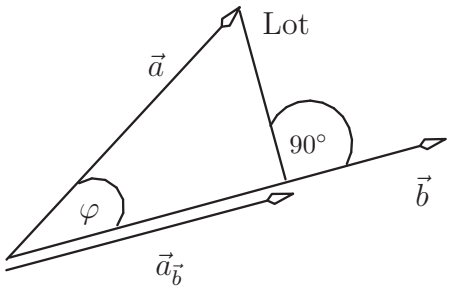
$$\vec{a}_{\vec{b}} = \frac{\vec{b}}{\|\vec{b}\|}\|\vec{a}\|\cos\varphi,$$

wobei $\frac{\vec{b}}{\|\vec{b}\|}$ die Orientierung und $\|\vec{a}\|\cos\varphi$ die Länge $(+/-)$ angibt. Bei gleicher Orientierung von $\vec{a}_{\vec{b}}$ und $\vec{b}$ gilt:

$$\cos\varphi = \frac{\|\vec{a}_{\vec{b}}\|}{\|\vec{a}\|}$$

$$0 \leq \varphi < \frac{\pi}{2}$$

$$\Rightarrow \vec{a} * \vec{b} > 0$$



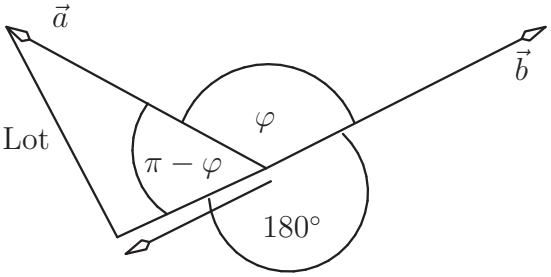
Projektion von $\vec{a}$ auf $\vec{b}$

Wenn $\vec{a}_{\vec{b}}$ und $\vec{b}$ nicht die gleiche Orientierung besitzen gilt:

$$\frac{\pi}{2} \leq \varphi < \pi$$

$$\cos(\pi - \varphi) = \frac{\|\vec{a}_{\vec{b}}\|}{\|\vec{a}\|}$$

$$\Leftrightarrow \cos\varphi = -\frac{\|\vec{a}_{\vec{b}}\|}{\|\vec{a}\|}$$



Projektion von $\vec{a}$ auf $\vec{b}$

$$\Rightarrow \vec{a} * \vec{b} \leq 0$$

Für ein beliebigen Punkt $\vec{p}$ der Trennebene gilt bezüglich der Projektion:

$$\|\vec{p}_{\vec{w}}\| = \frac{\vec{w} * \vec{p}}{\|\vec{w}\|} = c \qquad \text{(konstant)}$$

= Abstand der Ebene zum Ursprung.

Eine Trennebene teilt den Merkmalsraum in zwei Halbräume auf. Ein Punkt

$\vec{x}$ liegt entweder auf der einen , Seite

$$\vec{w} * \vec{x} + b > 0$$

oder auf der anderen Seite

$$\vec{w} * \vec{x} + b \leq 0$$

Alle $\vec{x}$ mit

$$\vec{w} * \vec{x} + b = 0$$

liegen direkt auf der Trennebene.

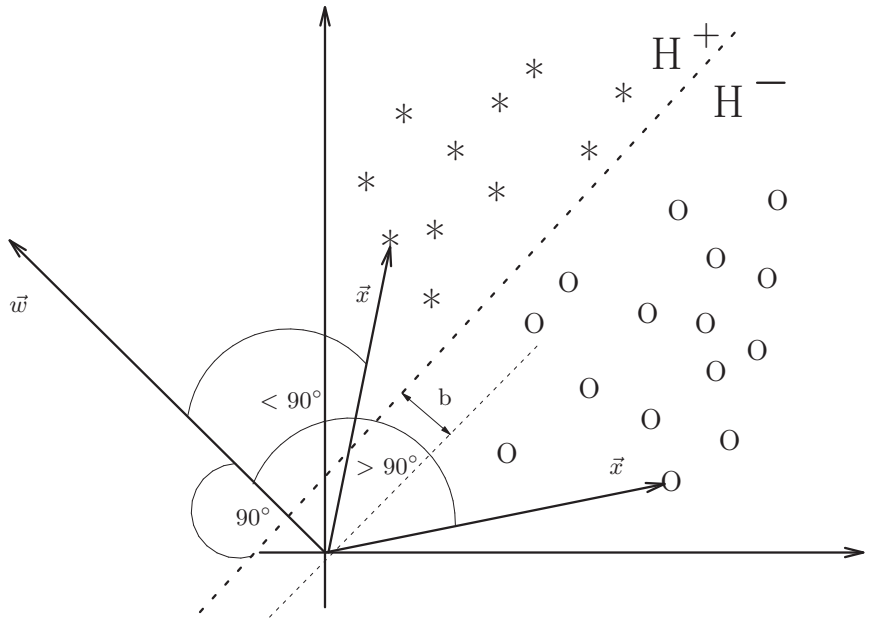


Abbildung 2.12: Hier ist ein zweidimensionaler Merkmalsraum zu sehen. Die Vektoren $\vec{x}$, $\vec{w}$ und das Skalar b sind dargestellt. Zusätzlich sind die Winkelbeziehungen und eine Trenngerade eingezeichnet. Der Datensatz ist linear separabel.

Betrachtet wird ein Merkmalsvektor $\vec{x}$. Wenn der Winkel zwischen $\vec{w}$ und $\vec{x}$ größer als 90° ist, liegt der betrachtete Punkt rechts von der Trennebene.

Dies entspräche der Klassenzugehörigkeit von H^-. Ist der Winkel zwischen $\vec{w}$ und $\vec{x}$ kleiner als 90°, liegt der betrachtete Punkt links der Trenngeraden und er würde zur Klasse H^+ gehören. Ist der Winkel genau 90°, so liegt der Punkt auf der Trennebene.
Ist das Skalar b gleich Null geht die Trennebene durch den Ursprung. Wenn b größer Null ist, verschiebt sich die Trennebene vom Ursprung weg. Damit ein Merkmalsvektor immer noch im gleichen jetzt entfernten Halbraum eingeordnet werden kann, muss der Winkel zwischen dem repräsentierenden $\vec{x}$ und dem Normalenvektor $\vec{w}$ groß genug sein, um die Addition von b zum Skalarprodukt zu kompensieren.
Ein linearer Klassifikator realisiert eine Entscheidungsregel auf Grundlage einer linearen Diskriminante.

$$f(\vec{x}) = \vec{w} * \vec{x} + b$$

$$\text{Linearer Klassifikator:} \quad c : \mathbf{R}^d \rightarrow \{-1, 1\}$$

$$c(\vec{x}) = \begin{cases} 1, \text{falls } f(\vec{x}) > 0 \\ -1, \text{falls } f(\vec{x}) \leq 0 \end{cases}$$

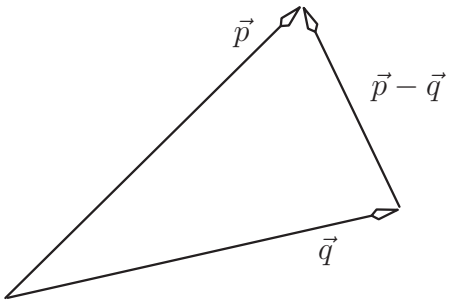
$f(\vec{x})$: Trennfunktion
$\vec{w}$: Normalenvektor
$\vec{x}$: Merkmalsvektor
b : Verschiebungsparameter

2.17.2 Mittelwertbasierte Klassifikation

Bei der mittelwertbasierten Klassifikation werden zwei gegebene Klassen durch zwei besonders charakteristische Vektoren $\vec{m}_+$ und $\vec{m}_-$ repräsentiert. Ein Datenbeispiel $\vec{x}$ wird nach der minimalen euklidischen Distanz klassifiziert.

Die euklidische Distanz zwischen zwei Vektoren ($\vec{p}$ und $\vec{q}$) kann wie folgt beschrieben werden:

$$D(\vec{p},\vec{q}) = \|\vec{p}-\vec{q}\| = \sqrt{(\vec{p}-\vec{q}) * (\vec{p}-\vec{q})} \tag{2.9}$$



Es gilt: **D** $(\vec{p},\vec{q}) \geq 0$
D$(\vec{p},\vec{q}) = 0 \Rightarrow \vec{p} = \vec{q}$

In der Trainingsphase werden die beiden Mittelwertsvektoren ($\vec{m}_-$ und $\vec{m}_+$) geschätzt. Die Mittelwertsvektoren der Trainingsmenge repräsentieren den „Schwerpunkt“ der Verteilung und gelten als charakteristische Vertreter einer Klasse.

$$c(\vec{x}) = \begin{cases} 1, \text{falls } \|\vec{x}-\vec{m}_+\| < \|\vec{x}-\vec{m}_-\| \\ -1, \text{sonst} \end{cases} \tag{2.10}$$

Der Mittelwertsvektor ist kein Datenbeispiel sondern ein Punkt im Merkmalsraum, dessen Distanz zu einem neuen Datenbeispiel angibt, ob ein Datenbeispiel in die eine oder in die andere Klasse klassifiziert wird.

Mittelwertsvektor für $\{\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_n\}$

$$\vec{m} = \sum_{i=1}^{n} \frac{\vec{x}_i}{n}$$

In unserem konkreten Fall werden zwei Mittelwertsvektoren bestimmt, einer repräsentiert die positiven Beispiele und der andere die negativen Beispiele:

$$\vec{m}_+ = \sum_{i:y_i=1} \frac{\vec{x}_i}{|\{i : y_i = 1\}|}$$

$$\vec{m}_- = \sum_{i:y_i=-1} \frac{\vec{x}_i}{|\{i : y_i = -1\}|}$$

Für einen Merkmalsvektor $\vec{x}$, der als positiv klassifiziert werden soll, bedeu-

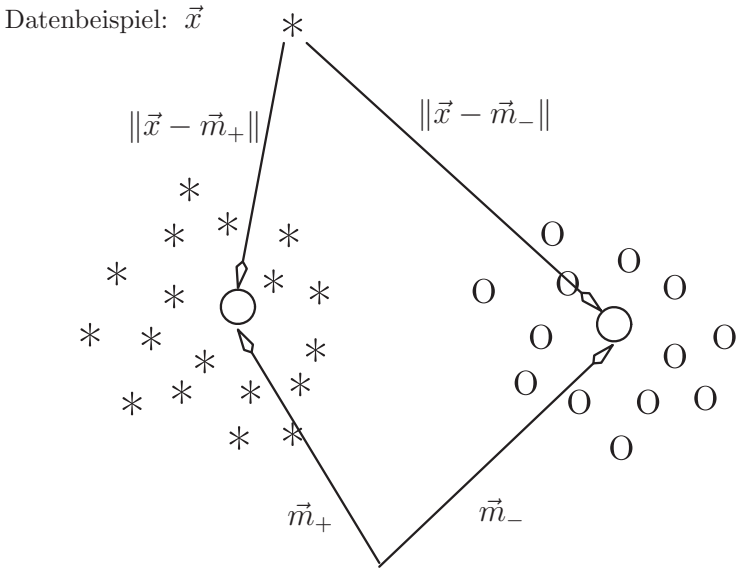


Abbildung 2.13: Hier sind die beiden „geschätzten“ Mittelwertsvektoren $\vec{m}_+$ und $\vec{m}_-$ dargestellt. Desweiteren ist ein noch nicht klassifiziertes Datenbeispiel dargestellt, welches nach der minimalen Distanz zu den geschätzten Mittelwertsvektoren klassifiziert werden soll.

tet es, dass die folgende Ungleichung erfüllt sein muss:

$$\|\vec{x} - \vec{m}_+\| < \|\vec{x} - \vec{m}_-\|$$

Man kann zeigen, dass ein mittelwertbasierter Klassifikator als linearer Klassifikator betrachtet werden kann.

Mit den berechneten Mittelwertsvektoren können direkt $\vec{w}$ und b bestimmt werden:

$$\vec{w} = \vec{m}_+ - \vec{m}_-$$
$$b = \frac{\|\vec{m}_-\|^2 - \|\vec{m}_+\|^2}{2}$$

2.17.3 Trennebene beim mittelwertbasierten Klassifikator

Der Vektor $\vec{p}$ ist der Mittelpunkt der Verbindungslinie zwischen $\vec{m}_+$ und $\vec{m}_-$. Eine Trennebene geht durch diesen Schnittpunkt und teilt den Merkmalsraum in zwei Halbräume auf. Die Trennebene steht orthogonal zum Vektor $\vec{w}$.

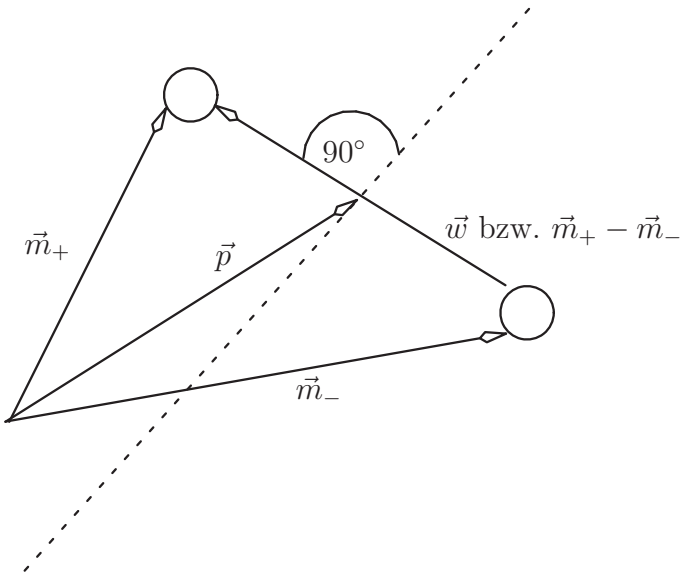


Abbildung 2.14: Hier sind die beiden Mittelwertsvektoren und die Trennebene dargestellt. Der Vektor $\vec{p}$ teilt den Normalenvektor $\vec{w}$ in zwei gleich große Teilstücke. Die Trennebene steht orthogonal auf dem Vektor $\vec{w}$ und geht durch diesen Schnittpunkt.

Die Trennebene lässt sich folgendermaßen beschreiben:

$$\text{H: } (\vec{m}_+ - \vec{m}_-) * \vec{x} + \frac{\|\vec{m}_-\|^2 - \|\vec{m}_+\|^2}{2} = 0 \tag{2.11}$$

Wobei der erste Faktor $\vec{w}$ entspricht:

$$\vec{w} = \vec{m}_+ - \vec{m}_-$$

Wie oben schon erwähnt kann das Skalar b so beschrieben werden.

$$b = \frac{\|\vec{m}_-\|^2 - \|\vec{m}_+\|^2}{2}$$

Auf diese Weise lassen sich optimale Trennebenen für zwei gleichwahrscheinliche Klassen mit radial symmetrischen Normalverteilungen bestimmen.

2.18 Verwendung des Kerndichteschätzers

Eine RNA-Sekundärstruktur in der Punkt-Klammer-Notation kann, wie in Abschnitt 2.15.1, S. 48 schon beschrieben wurde, ohne Informationsverlust in Kontaktpunkte überführt werden. Die Kontaktpunkte werden als zweidimensionale Vektoren dargestellt. Falls es konservierte Kontaktpunkte in den betrachteten Strukturen gibt, können diese mit Hilfe eines Kerndichteschätzers visualisiert werden.
Betrachtet wird, wie in dem Abschnitt 2.8, S. 37, 2.9, S. 39 und 2.10, S. 41 beschrieben, ein Rahmen von 170 bzw. 200 Nukleotiden. Existieren mögliche konservierte Kontaktpunkte in diesem Rahmen, würde man an den gleichen Positionen in verschiedenen Strukturen die gleichen Kontaktpunkte finden.
Bei den negativen Beispielen sind mögliche konservierte Kontaktpunkte, die mit einem „wahren“ Genstart assoziiert sein könnten, in einem Rahmen von 60 Nukleotiden um die „falsche“ TIS angeordnet. Eine Anordnung dieser Art würde Aufschluss darüber geben ob es überhaupt mögliche konservierte Kontaktpunkte gibt, die mit einen Genstart assoziiert sind.

Bei den positiven Beispielen wären mögliche konservierte Kontaktpunkte zentriert, da der „wahre“ Genstart zentriert in dem betrachteten Fenster liegt. Bei den permutierten Beispielen sollten die benutzten Programme nur zufällige RNA-Sekundärstrukturen vorhersagen können. Die entsprechenden Kontaktpunkte müssten sich gleichmäßig über das betrachtete Fenster verteilen, denn in den permutierten Beispielen ist der Leserahmen, die TIS und alles andere was in der Sequenz konserviert war, gelöscht worden. Man würde also ein Plateau bzw. eine gleichmäßige Verteilung der Kontaktpunkte in einem erstellten Dichteplot erwarten.
Um die Unterschiede sichtbar zu machen, wurde ein Kerndichteschätzer benutzt. Der hier verwendete Kerndichteschätzer ist frei im Internet[5] verfügbar. Er wurde in der MATLAB©-Umgebung implementiert.

[5] http://euler.ntu.ac.uk/maths.html

Kapitel 3

Ergebnisse

3.1 Darstellung der Kontaktpunkte

Die hier dargestellten Dichtplots sind mit „ungeglätteten“ Kontaktpunkten, wie in Abschnitt 2.15.1, S. 48 beschrieben, erstellt worden. Dies diente als Voruntersuchung um festzustellen, ob mögliche konservierte Kontaktpunkte im Bereich der untersuchten TIS existieren.
Da die Kontaktpunkte zweidimensionale Vektoren sind, können sie in x-,y-Richtung aufgetragen werden. Liegen zwei Kontaktpunkte übereinander oder in direkter Nachbarschaft zueinander, werden die überlappenden Kernfunktionen dementsprechend addiert. Die so entstandene Landschaft ist in z-Richtung aufgetragen. Dort, wo viele Kontaktpunkte übereinander oder in enger Nachbarschaft nebeneinander liegen, entstehen Hügel bzw. Berge. Kommt es zur Ausprägung solcher Berge bzw. Hügel, ist dies ein Hinweis dafür, dass sich in vielen Strukturen an derselben Position Basenpaare gebildet haben. Somit könnte vermutet werden, dass es sich hierbei um mögliche konservierte RNA-Sekundärstrukturen handelt.
Der Datensatz besteht aus 722 bzw. 854 Beispielen. Die Strukturen aller Beispiele umfassen insgesamt ca. 20.000 - 25.000 Basenpaare. Um diese Menge

an Basenpaaren bzw. Kontaktpunkten vergleichen zu können, wurde eine 3d-Dichteplotfunktion unter MATLAB© verwendet.

3.2 Kontaktpunkte aus der ersten Klassifikation

3.2.1 Dichteplot von Kontaktpunkten der negativen Beispiele

Bei der Abbildung 3.1, S. 67 handelt es sich um einen Dichteplot von negativen Beispielen, die durch die Verknüpfung der Programme `RNALfold` und `RNAfold` berechnet wurden.
Es wird, wie in Abschnitt 2.8, S. 37 beschrieben wird, ein 170 Nukleotide langes Fenster betrachtet. Eine „falsche TIS“ ist an der Position 70 in dem betrachteten Fenster lokalisiert.
Es gibt keinen oder nur einen gering spitzen Zulauf in Richtung Berggipfel. Entlang des Bereichs (50,60) - (110,120) existieren drei Spitzen und kleine Täler in der Landschaft.
Links und rechts der Position (70,80) existieren mögliche konservierte Kontaktpunkte. Dies ist an den addierten Werten der Kernfunktionen zu sehen, die diese Spitzen bilden. Die Spitzen liegen in einem Rahmen von 60 Nukleotiden upstream bzw. downstream einer „falschen“ TIS.

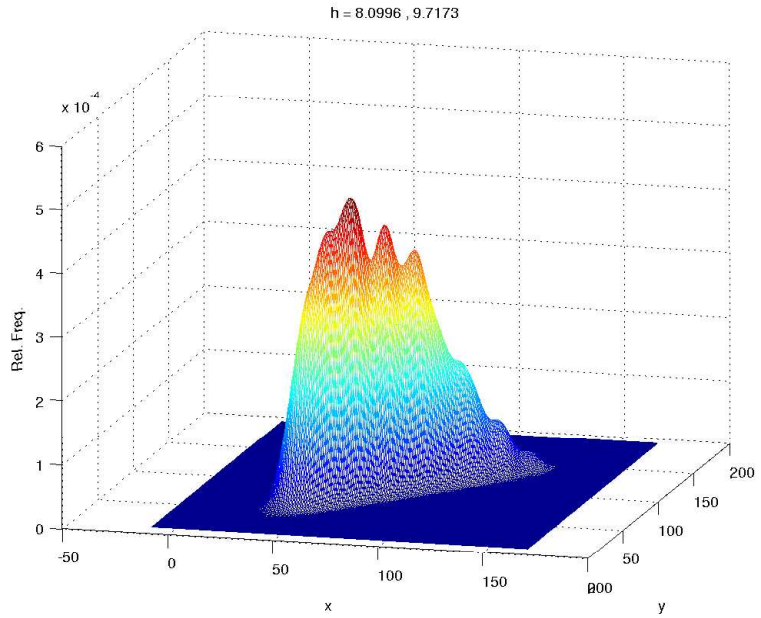


Abbildung 3.1: Hier ist ein Dichteplot negativer Beispiele zu sehen. Der Dichteplot wurde unter MATLAB© geplottet. Die benötigten Kontaktpunkte sind mit Hilfe der Programme `RNALfold` und `RNAfold` berechnet worden.

3.2.2 Dichteplot von Kontaktpunkten der positiven Beispiele

Bei der Abbildung 3.2, S. 68 handelt es sich um einen Dichteplot von positiven Beispielen, die durch die Verknüpfung der Programme `RNALfold` und `RNAfold` berechnet wurden.

Es wurde ein 170 Nukleotide langes Fenster, wie in Abschnitt 2.8, S. 37 beschrieben, betrachtet. Ein „wahrer“ Genstart ist an Position 70 in dem betrachteten Fenster lokalisiert.

Eine absolutes Maximum ist an der Position (50,50) zu sehen, und in dem Bereich (60,80) - (90,100) existieren zwei kleine lokale Maxima. An der Position (80,80) geht es steil bergab bis zur Position (100,100), da hier nur wenige Kontaktpunkte übereinander bzw. in näherer Nachbarschaft zueinander lie-

gen.

An der Position (130,140) existiert ein weiteres lokales Maximum. Zwischen diesen beiden lokalen Maxima gibt ein tiefes Tal. Der Bergfuß ist dadurch insgesamt etwas breiter als der Bergfuß der negativen Beispiele in Abbildung 3.1, S. 67.

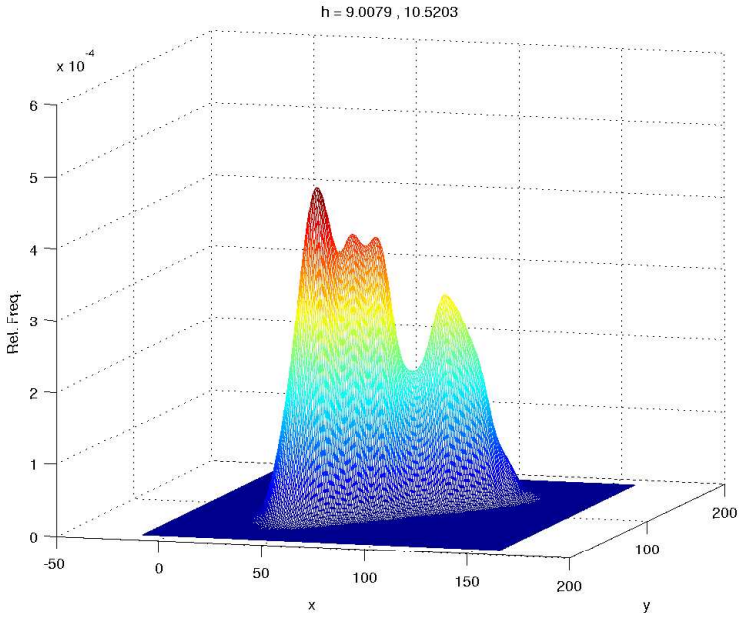


Abbildung 3.2: Hier ist ein Dichteplot positiver Beispiele, die durch die Verknüpfung der Programme `RNALfold` und `RNAfold` berechnet wurden, dargestellt. Der Dichteplot wurde unter MATLAB© geplottet.

3.2.3 Dichteplot von Kontaktpunkten aus Zufallssequenzen

In der Abbildung 3.3 ist ein Dichteplot der permutierten Beispiele zu sehen. Im Abschnitt 2.11, S. 42 ist aufgeführt, wie diese Sequenzen erzeugt wurden. An Position (70,90) ist ein lokales Maximum zu sehen und ein zweites an Position (90,110). Es ist aber nicht so ausgeprägt wie das lokale Maximum an der Stelle (70,90). Der Bergfuß ist im Gesamten sehr breit und umfasst den Bereich (45,50) - (160,160).

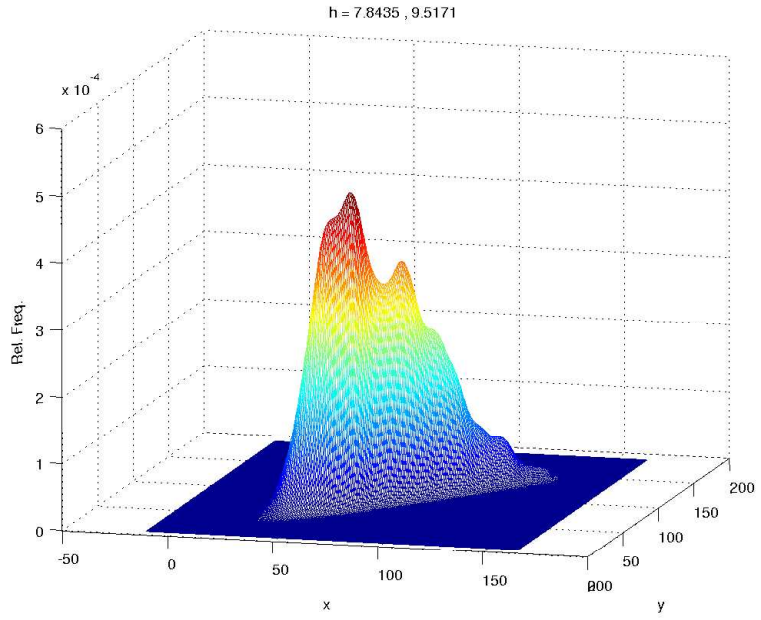


Abbildung 3.3: Wie auch bei den Dichteplots der negativen und positiven Beispiele ist hier ist ein Dichteplot von Zufallssequenzen zu sehen. Die Zufallssequenzen sind, wie in Abschnitt 2.11, S. 42 beschrieben, berechnet worden. Der Dichteplot wurde unter MATLAB© geplottet.

3.3 Kontaktpunkte aus der zweiten Klassifikation

3.3.1 Dichteplot von Kontaktpunkten der negativen Beispiele

In Abbildung 3.4, S. 71 ist ein Dichteplot von negativen Beispielen zu sehen. Die Kontaktpunkte, die für diesen Plot verwendet wurden, sind mit Hilfe der Programme `RNALfold` und `RNAsubopt` generiert worden.
Im Gegensatz zu den vorherigen Dichteplots wird hier ein 200 Nukleotide langes Fenster betrachtet. Im Abschnitt 2.9, S. 39 wurde beschrieben, wie das Fenster entstanden ist.
Eine „falsche“ TIS ist im betrachteten Fenster an der Position 100 lokalisiert. Im Bereich (0,0) - (150,150) sind Kontaktpunkte angeordnet, wodurch der Bergfuß sehr breit ist und ebenfalls über den gesamten Bereich reicht.
Es existieren zwei lokale Maxima an den Positionen (65,90) und (85,110), welche allerdings nicht sehr ausgeprägt sind. Zwischen den lokalen Maxima gibt es ein relativ kleines Tal.
Einem schwachen lokalen Maximum auf der x-Seite liegt ein schwaches lokales Maximum auf der y-Seite gegenüber. Insgesamt weist der Plot keine ausgeprägten Konturen auf.

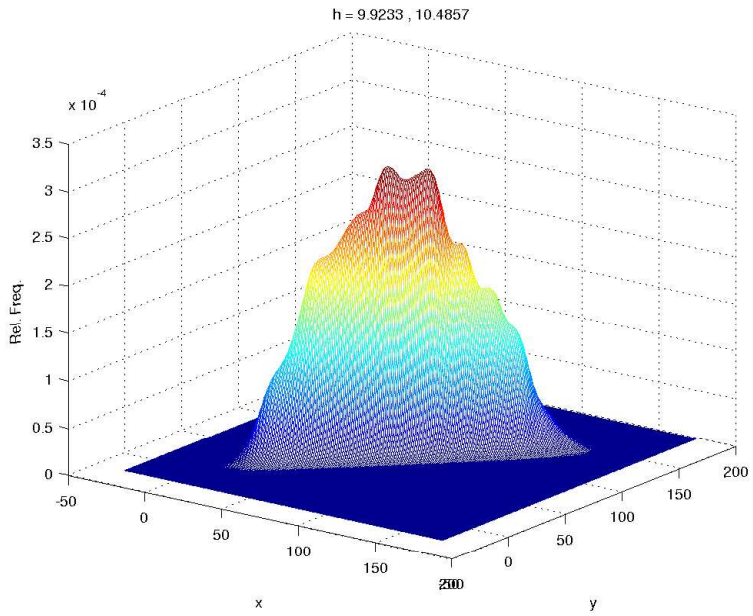


Abbildung 3.4: Hier ist ein Dichteplot von Kontaktpunkten, die mit den Programmen `RNALfold` und `RNAsubopt` berechnet wurden, zu sehen. Es sind die Kontaktpunkte der negativen Beispiele, wie in Abschnitt 2.9, S. 39 beschrieben, zu sehen. Der Dichteplot wurde unter MATLAB© geplottet.

3.3.2 Dichteplot von Kontaktpunkten der positiven Beispiele

In Abbildung 3.5, S. 72 ist ein Dichteplot der Kontaktpunkte der positiven Beispiele zu sehen. Die Kontaktpunkte sind mit Hilfe der Programme `RNAfold` und `RNAsubopt` berechnet worden.

Ein „wahrer“ Genstart ist in dem betrachteten Fenster an Position 100 lokalisiert. An Position (50,70) ist ein starkes lokales Maximum zu sehen. Der Plot weist einen relativ tiefen Einschnitt an der Position (60,90) auf. Ein zweites lokales Maximum befindet sich an Position (70,100). Der Bergfuß ist insgesamt schmaler als der Bergfuß in Abbildung 3.4, S. 71.

Man kann dem Plot entnehmen, dass sich mögliche konservierte Kontakt-

punkte in dem Bereich (50,70) - (80,110) befinden.

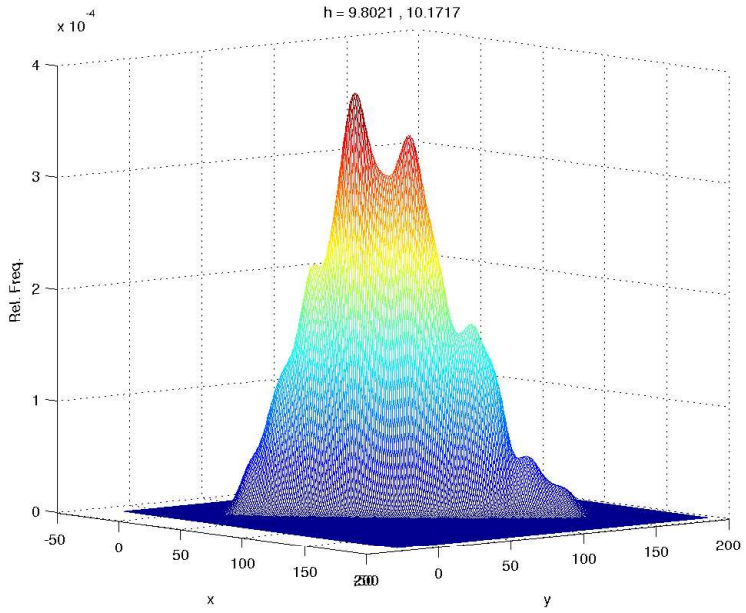


Abbildung 3.5: Hier ist ein Dichteplot der Kontaktpunkte der positiven Beispiele zu sehen. Die positiven Beispiele sind mit den Programmen `RNALfold` und `RNAsubopt` berechnet worden. Der Dichteplot wurde unter MATLAB© geplottet.

3.3.3 Dichteplot von Kontaktpunkten aus Zufallssequenzen

Die Abbildung 3.6 zeigt einen Dichteplot von Zufallssequenzen. Diese sind, wie in Abschnitt 2.11, S. 42 beschrieben, erzeugt worden. Der Bergfuß ist sehr breit und reicht über die gesamte Breite Position (0,0) - (150,150). Ein schwaches lokales Maximum ist an Position (60,100) zu sehen. Um das lokale Maximum an der Position (100,110) bilden sich weitere sehr schwache lokale Maxima aus. Der Dichteplot weist insgesamt keine starken Konturen auf.

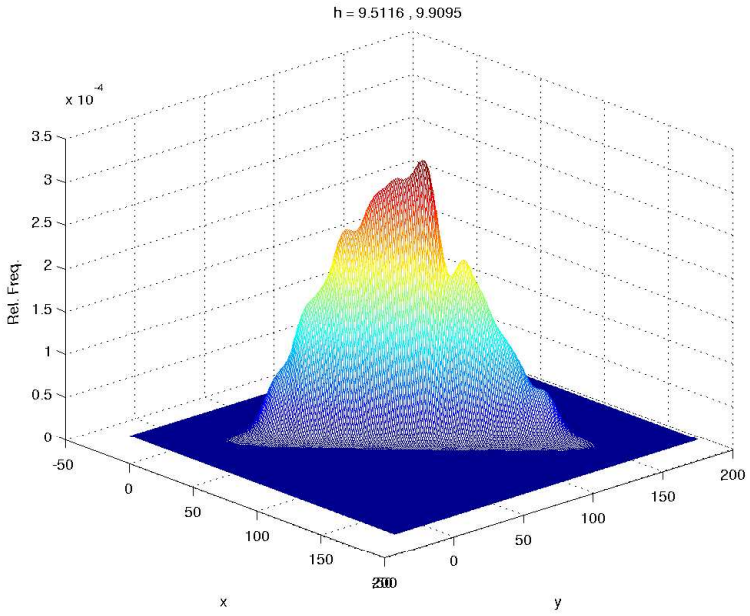


Abbildung 3.6: Hier ist ein Dichteplot von Kontaktpunkten aus zufällig generierten Sequenzen zu sehen. Der Dichteplot wurde unter MATLAB© geplottet.

3.4 Mittelwertbasierte Klassifikationen

Nachdem die erstellten Dichteplots gezeigt haben, dass es zu Häufungen von Kontaktpunkten kommt, die mit einem Genstart assoziiert sein könnten, ist eine mittelwertbasierte Klassifikation durchgeführt worden. Hierzu wurde der Datensatz in Trainings-, Validierungs-, und Testmenge, wie in Abschnitt 2.16, S. 52 beschrieben, aufgeteilt.
Es wurden insgesamt drei unterschiedliche Klassifikationen durchgeführt. Wie die unterschiedlichen Kontaktpunkte für die verschiedenen Klassifikationen entstanden sind, ist in den Abschnitten 2.8, S. 37, 2.9, S. 39 und 2.10, S. 41 beschrieben.

3.5 Ergebnisse der ersten Klassifikation

Wie schon in Abschnitt 2.16, S. 52 beschrieben, wird der vorhandene Datensatz in drei disjunkte Mengen aufgeteilt. Es wird eine Trainingsmenge, Validierungsmenge und eine Testmenge erzeugt.
In der Trainingsphase wird der Klassifikator konstruiert, das heißt es werden die Mittelwertsvektoren und eine Trennebene berechnet. Danach wird in der Validierungsphase ein optimaler Glättungsparameter σ bestimmt (siehe Abschnitt 2.16.2, S. 53). Um den optimalen Glättungsparameter zu bestimmen, werden für ihn verschiedene Werte getestet. Die beste Einstellung für σ wird für die Klassifikation auf der Testmenge verwendet. Somit wird der Klassifikationsfehler mit optimalem σ bestimmt.
Die Abbildung 3.7, S. 75 zeigt den gemittelten Klassifikationsfehler der Validierung. Da die einzelnen Datenbeispiele durch das Label y gekennzeichnet sind, kann der Fehler der Klassifikation berechnet werden. Je mehr positive Beispiele in den H^- Halbraum klassifiziert werden, desto größer wird der Klassifikationsfehler. Das gleiche gilt auch für die negativen Beispiele, die falsch in den H^+-Halbraum klassifiziert werden.

Die Kontaktpunkte für diese Klassifikation sind wie in Abschnitt 2.8, S. 37 beschrieben, erzeugt worden. Es werden für jede neue Einstellung des Glättungsparamerts σ 50 Läufe (Klassifikationen) durchgeführt. Der Klassifi-

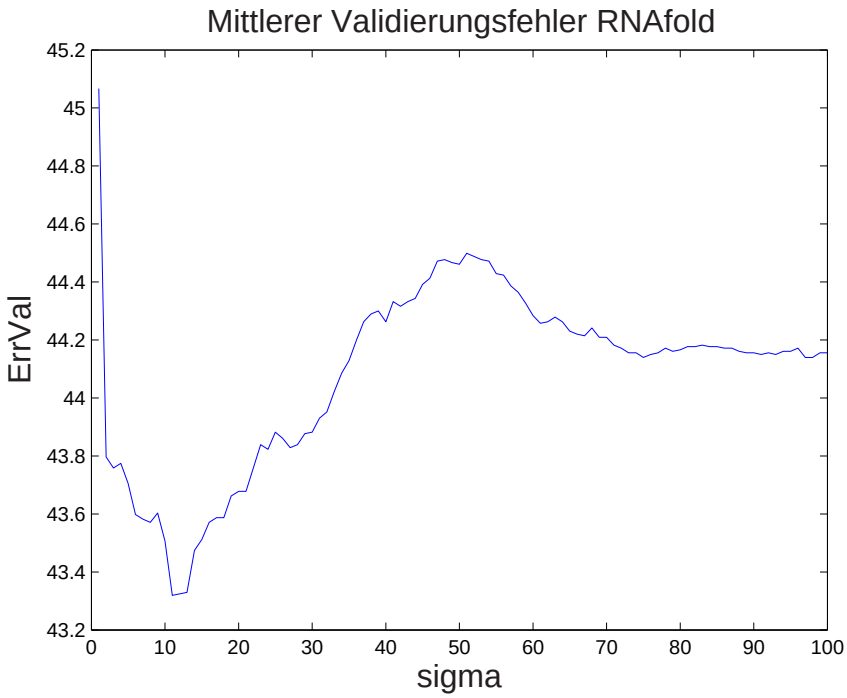


Abbildung 3.7: Hier ist der gemittelte Klassifikationsfehler der Validierung (ErrVal) zu sehen. Auf der x-Achse sind die getesteten σ Werte aufgetragen und auf der y-Achse ist der Klassifikationsfehler dargestellt.

kationsfehler wird nach der Klassifikation bestimmt. Bei jedem neu getesteten σ Wert sind 50 Läufe durchgeführt worden. Bei den unterschiedlichen Läufen wurden verschiedene Trainingsmengen, Validierungsmengen und Testmengen erzeugt. Dadurch entstehen unterschiedliche Klassifikationsfehler.
Auf der x-Achse sind die unterschiedlichen σ Einstellungen aufgetragen und auf der y-Achse die jeweiligen Klassifikationsfehler dargestellt. Bei der Validierung ist ein optimales σ von 14.5 berechnet worden. Bei der anschießenden Klassifikation auf der Testmenge ist ein gemittelter Klassifikationsfehler von 43.7% berechnet worden. Die dazugehörige Standardabweichung betrug 1.9.

3.6 Ergebnisse der zweiten Klassifikation

Eine zweite mittelwertbasierte Klassifikation ist mit Kontaktpunkten siehe Abschnitt 2.9, S. 39 durchgeführt worden.

In Abbildung 3.8 sind alle Klassifikationsfehler der Validierung dargestellt.

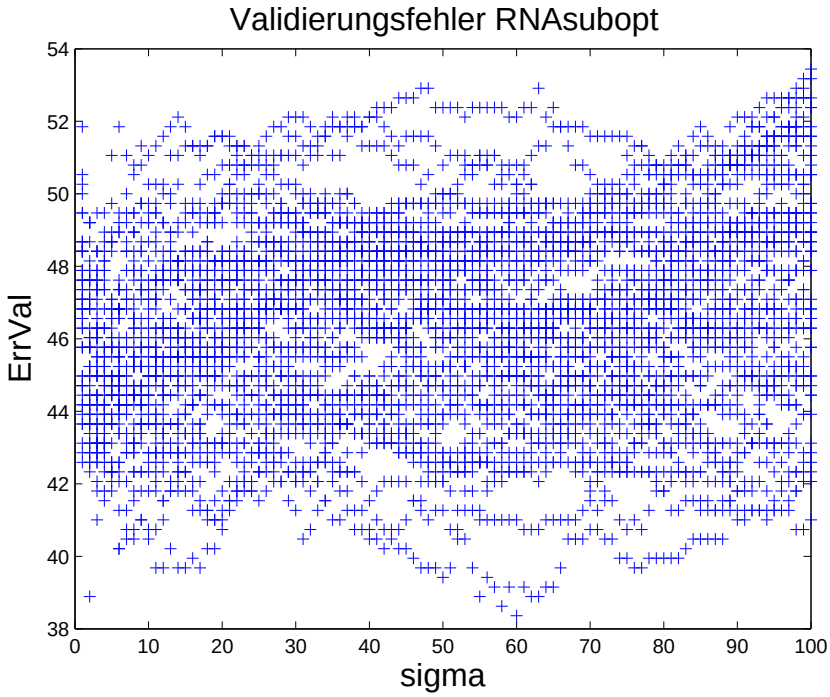


Abbildung 3.8: In diesem Plot sind alle Klassifikationsfehler der Validierung (ErrVal) zu sehen. Auf der x-Achse sind die getesteten σ Werte aufgetragen und auf der y-Achse ist der Klassifikationsfehler dargestellt.

Auf der x-Achse sind die getesteten σ Werte aufgetragen (0-100) und auf der y-Achse sind die Klassifikationsfehler dargestellt. Es wurden 50 Läufe pro getesteten σ Wert durchgeführt und die entsprechenden Klassifikationsfehler in den Plot eingetragen. Es wurde ein optimales σ von 32.4 berechnet. Bei der anschließenden Klassifikation auf der Testmenge ergab sich ein gemittelter Klassifikationsfehler von 45.4%. Die dazugehörige Standardabweichung lag bei 2.1.

Die Abbildung 3.9 zeigt den gemittelten Klassifikationsfehler der Validierung. Aus den Fehlerwerten der einzelnen σ Werte in der Abbildung 3.8, S. 76 ist der Mittelwert berechnet worden. Auf der x-Achse sind die unterschiedlichen σ Werte aufgetragen und auf der y-Achse die Klassifikationsfehler dargestellt.

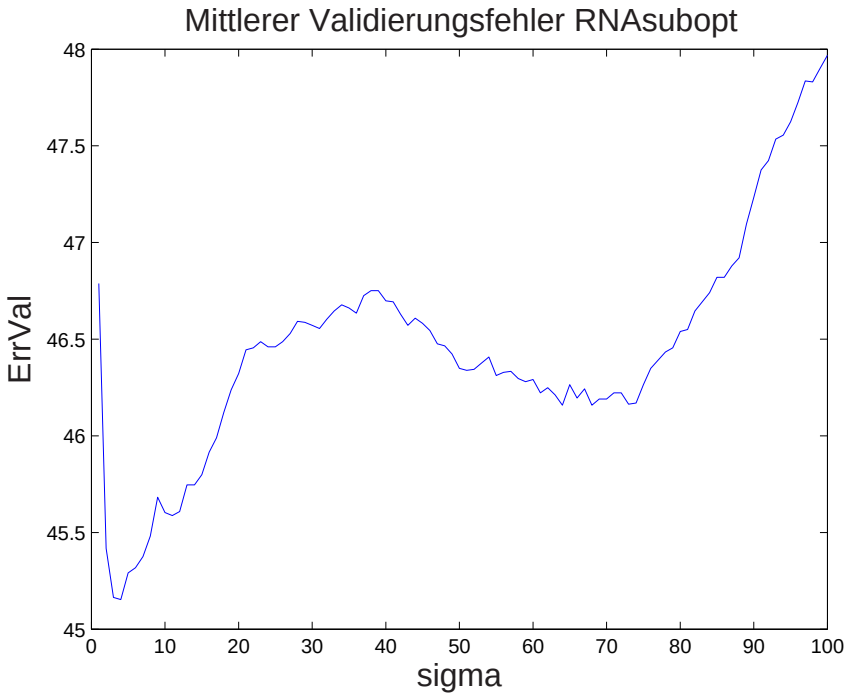


Abbildung 3.9: Diese Abbildung zeigt den gemittelten Klassifikationsfehler (ErrVal). Der Mittelwert des Klassifikationsfehlers wurde aus den Daten der Abbildung 3.8, S. 76 bestimmt.

3.7 Ergebnisse der dritten Klassifikation

In der Abbildung 3.10 ist der gemittelte Klassifikationsfehler der Validierung dargestellt. Auf der y-Achse ist der Klassifikationsfehler aufgetragen und auf der x-Achse sind die unterschiedlichen σ Werte dargestellt. Es wurden 50 Läufe pro getesteten σ Wert durchgeführt.
Dies hat das Ziel, eine σ Einstellung zu finden, bei der der Klassifikationsfehler so klein wie möglich ist. Es wurde ein optimales σ von 9.4 berechnet. Die Abbildung 3.11, S. 79 zeigt den gemittelten Klassifikationsfehler der Va-

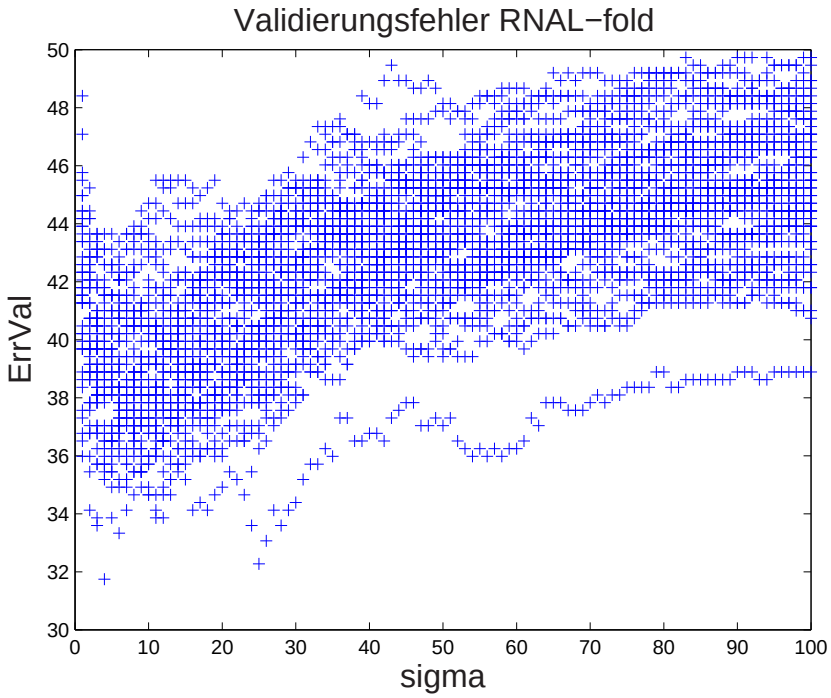


Abbildung 3.10: Diese Abbildung zeigt alle Klassifikationsfehler der Validierung (ErrVal). Er ist analog zu dem in Abbildung 3.8, S. 76 berechnet worden.

lidierung. Pro Spalte wurde ein neuer σ-Wert getestet (0-100) und für jeden neuen σ-Wert 50 Läufe durchgeführt. Aus den entstandenen Klassifikationsfehlern wurde der Mittelwert berechnet, aufgetragen und ein optimales σ

von 9.4 berechnet. Das optimale σ wurde dann für die Klassifikation auf der Testmenge verwendet.

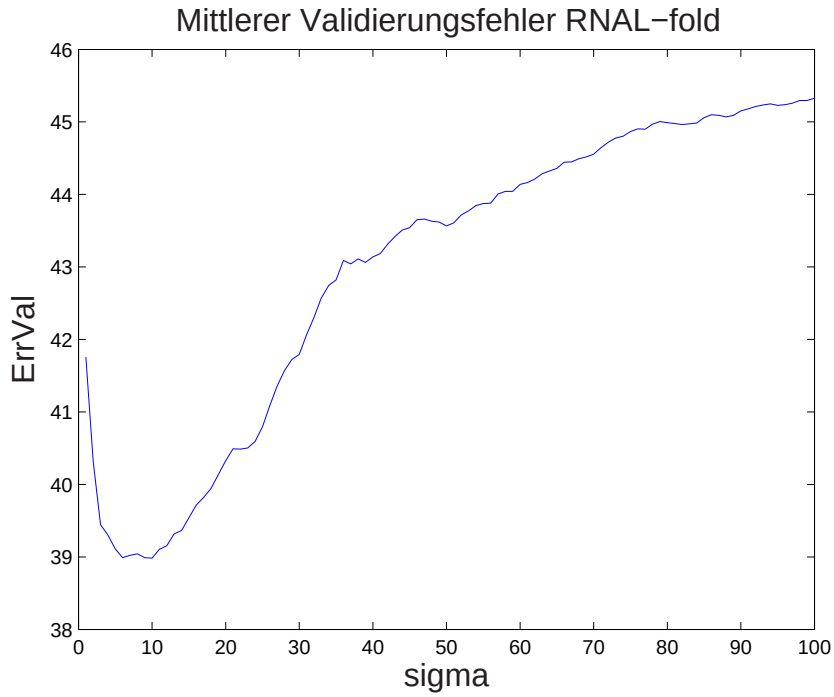


Abbildung 3.11: Diese Abbildung zeigt den Mittelwert aller Klassifikationsfehler (ErrVal). Der Mittelwert des Klassifikationsfehlers wurde aus den Daten der Abbildung 3.10, S. 78 bestimmt.

3.7.1 Visualisierung des Gewichtsvektors

Aus den beiden Mittelwertsvektoren ($\vec{m}_+$, $\vec{m}_-$) wird die euklidische Distanz berechnet. Die euklidische Distanz zwischen $\vec{m}_+$ und $\vec{m}_-$ beschreibt den Normalenvektor $\vec{w}$. Die Trennebene steht orthogonal zum Normalenvektor $\vec{w}$ und teilt den Normalenvektor in zwei gleiche Teilstücke. Der Normalenvektor wird mit einer umgekehrten *vec*-Operation aus der Vektordarstellung in eine Matrixdarstellung überführt.

Die Kontaktmatrizen beinhalten die vorhergesagten RNA-Sekundärstruktu-

ren siehe Abschnitt 2.15.1, S. 48 und werden, wie unter Abschnitt 2.15.3, S. 50 beschrieben, mit einer Glättungsmatrix multipliziert.

Mit Hilfe der Validierungsmenge wurde der optimale Glättungsparameter

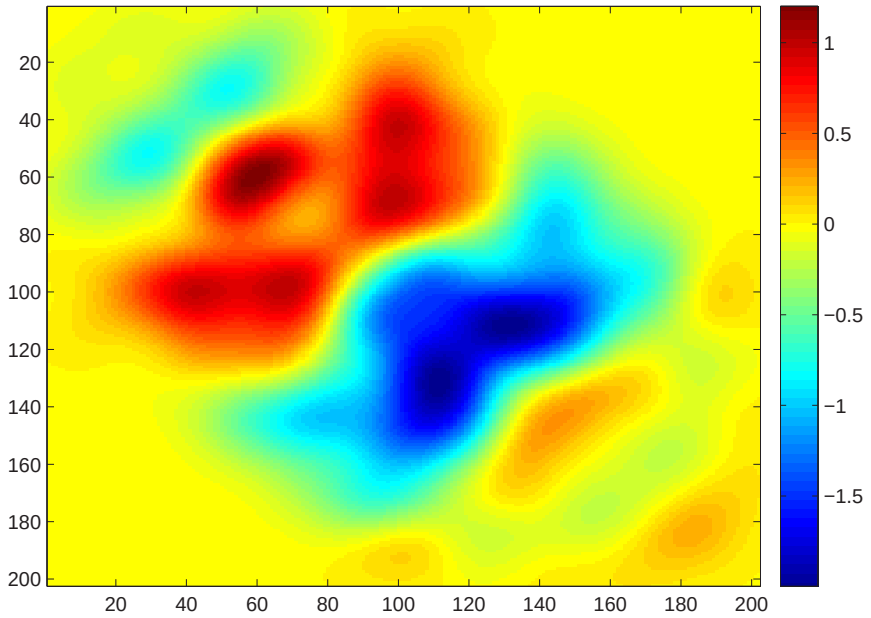


Abbildung 3.12: Hier ist eine Visualisierung des Gewichtsvektors einer mittelwertbasierten Klassifikation zu sehen. Die roten Flächen gehen als positive Merkmale in die Klassifikation ein, die blauen negativ und die orangenen Flächen sind kaum gewichtet.

σ ermittelt. Die so geglätteten Kontaktmatrizen der Testmenge werden mit der Matrix des Normalenvektors elementweise multipliziert und positionsweise addiert (Skalarprodukt wie definiert in Abschnitt 2.17.1, S. 54). Erhält man für ein Beispiel der Testmenge eine positive Zahl, so wird dieses Beispiel als positives Beispiel klassifiziert. Es liegt im positiven Halbraum (H^+ siehe Abschnitt 2.17.1, S. 54). Je größer die positive Zahl ist, desto weiter ist das Testbeispiel von der Trennebene entfernt.

Erhält man für ein Beispiel der Testmenge eine negative Zahl, so wird es als negatives Beispiel klassifiziert, da es im negativen Halbraum (H^-) liegt. Je

größer die negative Zahl ist, desto weiter liegt ein entsprechendes negatives Beispiel von der Trennebene im negativen Halbraum (H^-) entfernt.
Der Normalenvektor $\vec{w}$ entspricht einem Gewichtsvektor. In dem Plot 3.12, S. 80 sind alle Beispiele der Testmenge dargestellt. Der optimale Glättungsparameter σ wurde bei 9.4 validiert. Die TIS ist an der Postion 100 lokalisiert. Die dunkelroten Bereiche des Plots 3.12, S. 80 gehen als positive Merkmale (Kontaktpunkte) mit großen positiven Gewicht in die Klassifikation ein. Konservierte Kontaktpunkte befinden sich an der Position (50,70), was an dem lokalen Maximum im Plot zu erkennen ist (dunkelrote Färbung). Ein zweites bzw. drittes lokales Maximum ist an der Position (40,90) und (60,100) zu erkennen. An diesen Positionen sind ebenfalls mögliche konservierte Kontaktpunkte zu beobachten.
Die blauen bzw. dunkelblauen Bereiche gehen als negative Merkmale mit entsprechendem Gewicht in die Klassifikation ein. Die möglichen konservierten Kontaktpunkte liegen hinter der TIS an der Position (110,140), dies ist an dem lokalen Maximum (dunkelblaue Färbung) zu erkennen.
Dagegen entsprechen die orangenen Flächen den Positionen, die kaum oder gar nicht in die Diskrimination eingehen.

3.7.2 Klassifikationsfehler auf den Testdaten

Nachdem mit unterschiedlichen Trainingsmengen die Mittelwertsvektoren ($\vec{m}_+$ und $\vec{m}_-$) berechnet und mit unterschiedlichen Validierungsmengen ein optimales σ bestimmt wurde (siehe Abschnitt 2.16, S. 52), ergab der gemittelte Klassifikationsfehler auf der Testmenge 39.8%. Die Standardabweichung des gemittelten Klassifikationsfehlers war 2.2.

3.8 Ergebnistabelle

	optimales σ	Testfehler (mittel) %	Std
RNAfold	14,5	43,7	1,9
RNAsubopt	32,4	45,4	2,1
RNAL-fold	9,4	39,8	2,2

Tabelle 3.1: In der Tabelle ist das optimale σ und der prozentuale mittlere Testfehler mit Standardabweichung (std) über 50 Läufe für alle drei Klassifikationen angegeben. Das optimale σ wurde aus den Werten $1, 2, \ldots, 100$ bestimmt. Mit `RNAfold` ist die erste, mit `RNAsubopt` die zweite und mit `RNALfold` die dritte Klassifikation bezeichnet.

Kapitel 4

Diskussion

Die aus dem Genom von *Escherichia coli* entnommenen Sequenzen dienten als Datensatz für die beschriebenen Experimente. Hierzu wurden Sequenzen aus dem Genom betrachtet die eine verifizierte TIS beinhalten.
Für die Sequenzen wurden RNA-Sekundärstrukturen vorhergesagt und diese auf mögliche Informationen untersucht, die zur Verbesserung der Vorhersage des Genstarts eingesetzt werden könnten. Dazu wurden die vorhergesagten RNA-Sekundärstrukturen in Kontaktpunkte übersetzt (siehe Abschnitt 2.15.1, S. 48).
Ob es zu einer Häufung von Kontaktpunkten im Bereich einer TIS kommt, sollten speziell angefertigte 3d-Dichteplots zeigen. Nachdem die Dichteplots gezeigt haben, dass es mögliche konservierte Kontaktpunkte geben könnte, die mit einem Genstart assoziiert sind, wurde eine mittelwertbasierte Klassifikation durchgeführt.

4.1 Interpretation der Dichteplots

Die erstellten Dichteplots dienten als Voruntersuchung, ob es mögliche konservierte Kontaktpunkte gibt, die mit einer TIS assoziiert sein könnten.
Positive und negative Beispiele unterscheiden sich dadurch, dass die positiven

Beispiele einen „wahren“ Genstart immer an derselben Position im betrachteten Fenster lokalisiert haben (siehe Abbildung 2.1.2, S. 20).
Die negativen Beispiele haben einen „wahren“ Genstart nicht immer an derselben Position im betrachteten Fenster lokalisiert. Der „wahre“ Genstart liegt bei den negativen Beispielen in einem Bereich von etwa 60 Nukleotiden upstream und 60 Nukleotiden downstream einer „falschen“ TIS.
Ist eine RNA-Sekundärstruktur mit einem „wahren“ Genstart assoziiert, müssten sich die Kontaktpunkte der positiven und der negativen Beispiele diesbezüglich unterscheiden.
Existiert eine solche RNA-Sekundärstruktur, so müssten sich die Kontaktpunkte der positiven Beispiele in dieser Region in einem Dichteplot häufen. Ist die konservierte RNA-Sekundärstruktur mit einem Genstart assoziiert, müsste sich im Bereich einer „wahren“ TIS eine Häufung von Kontaktpunkten ergeben. Die Häufung der möglichen Kontaktpunkte, die diese konservierte RNA-Sekundärstruktur beschreibt, sollte in einem Dichteplot sichtbar gemacht werden.
Bei den negativen Beispielen ist dagegen eine Streuung der Kontaktpunkte zu erwarten, da die „wahre“ TIS bei den negativen Beispielen in einem Rahmen von 60 Nukleotiden up- oder downstream von einer „falschen“ TIS lokalisiert ist. Die möglichen Kontaktpunkte der konservierten RNA-Sekundärstruktur, die mit einem „wahren“ Genstart assoziiert ist, wären um die „falsche“ TIS verstreut. Die Streuung der Kontaktpunkte sollte in einem Dichteplot sichtbar gemacht werden.
Als Gegenprobe wurden Strukturvorhersagen auf Zufallssequenzen gemacht und geplottet. Damit sollte der Informationsgehalt der angewendeten RNA-Sekundärstrukturvorhersage Programme getestet werden. Die Experimente wurden unter gleichen Bedingungen aber nun mit den zufällig permutierten Sequenzen wiederholt.
Bei den Beispielen aus den permutierten Sequenzen können die Vorhersage-

programme nur zufällige Strukturen finden, da die Information, die in der Sequenz gespeichert war, gelöscht wurde. Die Kontaktpunkte der zufällig gefundenen RNA-Sekundärstrukturen müssten sich gleichmäßig über das betrachtete Fenster verteilen. In Abbildung 3.6, S. 73 ist ein Dichteplot von Kontaktpunkten der Zufallssequenzen dargestellt. Aus diesem Plot lässt sich entnehmen, dass sich die Kontaktpunkte über die gesamte Breite verteilen und dass der Plot keine starken Konturen aufweist.
Der Unterschied zwischen einem Dichteplot negativer Beispiele und einem Dichteplot positiver Beispiele ist der, dass sich in dem positiven Dichteplot „Spitzen“ an denselben Stellen des betrachteten Fensters ausbilden sollten. Die Spitzen repräsentieren mögliche konservierte Kontaktpunkte, da sie sich in vielen positiven Beispielen an ähnlicher Position befinden. Dieser Unterschied kann an den Plots 3.1, S. 67 und 3.2, S. 68 erkannt werden. Der Plot 3.1 hat keine starken Konturen und die Kontaktpunkte sind relativ breit über das gesamte Fenster verteilt. Anders als bei dem Plot 3.2, denn hier lassen sich tiefe „Einschnitte“ und eine „Spitzenbildung“ beobachten. In den positiven Beispielen liegen demnach viele Kontaktpunkte an denselben Positionen, was auf mögliche konservierte Kontaktpunkte schließen lässt.

4.2 Auswertung der Klassifikationen

4.2.1 Kontaktpunkte der ersten und zweiten Klassifikation

Aufgrund der erstellten Dichteplots liegt die Vermutung nahe, dass es mögliche konservierte Kontaktpunkte geben könnte, die mit dem Genstart assoziiert sind.
Mit den „geglätteten“ Kontaktpunkten aus der Verknüpfung der Programme

`RNALfold` mit `RNAfold` und `RNALfold` mit `RNAsubopt` wurde ein Klassifikator trainiert und eine mittelwertbasierte Klassifikation durchgeführt.
Die drei durchgeführten Klassifikationen unterschieden sich wie folgt: Während bei den ersten beiden Klassifikationen *eine* RNA-Sekundärstruktur pro positives oder negatives Beispiel in Kontaktpunkte übersetzt wurde, wurden bei der dritten Klassifikation sämtliche von dem Programm `RNALfold` vorhergesagten RNA-Sekundärstrukturen in Kontaktpunkte übersetzt.
Die Ergebnisse der ersten beiden Klassifikationen sind in Abschnitt 3.5, S. 74 und Abschnitt 3.6, S. 76 aufgeführt. Die Fehlerraten der ersten beiden Klassifikationen waren ähnlich hoch. Sie lagen bei ca. 42% - 45%. Ein Klassifikator, der keinerlei Information über die zu klassifizierenden Objekte besitzt, hätte eine Fehlerrate von 50%.
Vorhergehende Experimente konnten zeigen, dass sich mit Methoden des Maschinellen Lernens gute Klassifikationsergebnisse auf entsprechenden Sequenzen erzielen ließen. Ein Klassifikator konnte verifizierte Sequenzen aus dem EcoGene Datensatz bezüglich ihrer beinhalteten TIS unterscheiden. Der mittlere Klassifikationsfehler lag bei ca. 8% (Meinicke *et al.*, 2004).
Ein Grund für die unter Abschnitt 3.5, S. 74 und 3.6, S. 76 aufgeführten Ergebnisse könnte sein, dass mit dem Genstart in Prokaryoten eine RNA-Tertiärstruktur assoziiert ist. Ob ein ATG ein Startcodon ist oder nicht, könnte dem Ribosom durch ein Tertiärstruktur-Element übermittelt werden, verschiedene Tertiärstruktur-Elemente sind unter Abschnitt 1.4, S. 17 aufgeführt. RNA-Tertiärstrukturen können mit den hier verwendeten Programmen nicht mitberechnet werden.
In vielen Fällen ist die RNA-Tertiärstruktur für die biologische Aktivität verantwortlich. Die RNA-Sekundärstruktur kann sich erheblich schneller als die RNA-Tertiärstruktur ausbilden, weswegen angenommen wird, dass der Strukturbildende Prozess über die RNA-Sekundärstruktur zur komplexeren RNA-Tertiärstruktur verläuft. Die Ausbildung einer RNA-Tertiärstruk-

tur kann zu einer geringen Änderung der Grundstruktur führen. Die RNA-Tertiärstruktur kann somit aus einer suboptimalen RNA-Sekundärstruktur aufgebaut werden (Thirumalai *et al.*, 1998).

Desweiteren werden für die Vorhersage der RNA-Sekundärstrukturen die thermodynamischen Energieparameter (siehe Tabelle 2.2, S. 33) in entsprechender Genauigkeit benötigt. Für einige Looptypen existieren keine thermodynamischen Energieparameter. Die Energieparameter für Loops, welche größer als fünf Nukleotide sind, wurden extrapoliert.

Die Energien von Hairpinloops hängen von ihrer Größe ab, da größere Loops die Struktur mehr destabilisieren als kleine Loops. Prinzipiell destabilisieren aber alle Loops die Struktur. Wie stark ein Hairpinloop die Struktur destabilisiert, hängt zusätzlich auch noch vom letzten regulär gepaartem Basenpaar ab. Das Basenpaar G:C hat drei Wasserstoffbrückenbindungen und schließt einen Hairpinloop „stärker“ ein als ein abschließendes Basenpaar A:U. Das Basenpaar A:U besitzt im Gegensatz zum Basenpaar G:C eine Wasserstoffbrückenbindung weniger. Bei einigen Loop-Typen spielt zusätzlich auch noch die Loop Sequenz eine Rolle, auch diese Energieparameter sind nicht ausreichend bekannt und stehen den RNA-Strukturvorhersage Programmen nicht zur Verfügung. Aus diesen Gründen ist die Vorhersage für die RNA-Sekundärstrukturen mit Ungenauigkeiten behaftet.

Desweiteren ist es nicht möglich alle suboptimalen RNA-Sekundärstrukturen für lange Sequenzen berechnen zu lassen, denn die Anzahl der möglichen Strukturen wächst exponentiell mit Länge der betrachteten Sequenz (siehe Tabelle 2.3, S. 36). Der Vorteil einer Verteilung, die alle suboptimalen RNA-Sekundärstrukturen beinhaltet liegt darin, dass für ein bestimmtes Basenpaar eine Paarungswahrscheinlichkeit berechnet werden könnte. Dies ist ohne diese Verteilung nur bedingt möglich.

4.2.2 Kontaktpunkte der dritten Klassifikation

Wie in Abschnitt 2.1.2, S. 20 schon beschrieben, werden positive und negative Beispiele aus einem vorhandenen Datensatz erzeugt. Mit den verwendeten Vorhersage Programmen werden RNA-Sekundärstrukturen bzw. -verteilungen vorhergesagt. Die Strukturen werden in Kontaktpunkte übersetzt. Mit den „geglätteten“ Kontaktpunkten wird ein Klassifikator trainiert und eine mittelwertbasierte Klassifikation durchgeführt (siehe Abschnitt 2.17.1, S. 54).

Bei der dritten Klassifikation wurden sämtliche RNA-Sekundärstrukturen, die das Programm `RNALfold` in dem Bereich der TIS vorhergesagt hatte, in Kontaktpunkte übersetzt. Da eine RNA in einer thermodynamischen Strukturverteilung vorliegt, sollte sich durch das Einbeziehen aller vorhergesagten lokal stabilen RNA-Sekundärstrukturen die Klassifikationsrate verbessern. Der Klassifikationsfehler der dritten Klassifikation lag bei 39.8% dies entspricht einer Verbesserung von ca. 5% gegenüber den ersten Klassifikationsergebnissen.

`RNALfold` verwendet ein Fenster, welches Nukleotid für Nukleotid durch die Sequenz geschoben werden kann. In einem aktuell betrachteten Fenster wird dann eine lokal stabile RNA-Sekundärstruktur berechnet. Am Ende werden mehrere lokal stabile Strukturen für eine Sequenz ausgegeben. Zum einem spielt die Größe des Fensters und zu anderen die Basen an den Randpositionen des betrachteten Fensters eine Rolle (siehe Abschnitt 2.5, S. 33). Der Fensterparameter `-L` war bei den Experimenten auf 100 Nukleotide eingestellt. Strukturen mit mehr als 100 Nukleotiden konnten somit nicht mitberechnet werden.

Desweiteren bilden positive und negative Beispiele jeweils eine Klasse. Eine Klasse wird durch einen charakteristischen Mittelwertsvektor $\vec{m}_+$ oder $\vec{m}_-$ (siehe Abschnitt 2.17.2, S. 59) repräsentiert. Die beiden Mittelwertsvektoren werden aus der Trainingsmenge berechnet. Die RNA-Sekundärstrukturen

bzw. die daraus entstandenen Kontaktpunkte der Trainingsbeispiele wurden von dem Programm `RNALfold` vorhergesagt. Wie charakteristisch die Mittelwertsvektoren die entsprechende Klasse repräsentieren, nimmt großen Einfluss auf die Klassifikation (siehe 2.17.2, S. 59). Die Verteilung der Trainingsbeispiele in den dazugehörigen Klassen (positive Beispiele oder negative Beispiele) nehmen Einfluss auf die Berechnung der Mittelwertsvektoren. Im Idealfall bilden die jeweiligen Klassen (siehe Abschnitt 2.17.1, S. 54) so genannte Punktwolken oder Cluster aus. Von diesen Clustern wird der Mittelpunkt errechnet und dieser dient dann als charakteristischer Mittelwertsvektor. Bei dem Mittelwertsvektor handelt es sich nicht um ein konkretes positives oder negatives Beispiel, sondern um einen Punkt im Merkmalsraum, der in der Mitte der betrachteten Klasse liegt. Repräsentiert der Mittelwertsvektor die entsprechende Klasse nicht ausreichend, ist der Klassifikationsfehler dementsprechend hoch.

4.3 Zusammenfassung

In Genomen von Prokaryoten gibt es Sequenzen, die mit der Translationsinitiation assoziiert sind. Zum einen ist es die Shine-Dalgarno Sequenz und zum anderen das Startcodon ATG.
Vorhergehende Experimente konnten zeigen, dass sich mit Methoden des Maschinellen Lernens gute Klassifikationsergebnisse auf entsprechenden Sequenzen erzielen lassen. (Meinicke *et al.*, 2004).
Desweiteren wäre es denkbar, dass nicht nur die Shine-Dalgarno Sequenz bzw. das Startcodon ATG alleine für die Übermittlung der Genstart-Information innerhalb einer Prokaryotenzelle verantwortlich sind.
Ziel der vorliegenden Arbeit war es, RNA-Sekundärstrukturvorhersagen auf mögliche Informationen zu untersuchen, die zur Verbesserung der Vorhersa-

ge der Startposition eingesetzt werden könnten. Diese Informationen könnten beispielsweise mögliche konservierte Kontaktpunkte sein.
Für die jeweiligen Beispiele (positiv und negativ) wurden RNA-Sekundärstrukturen mit unterschiedlichen Programmen vorhergesagt. Die RNA-Sekundärstrukturen wurden anschließend in Kontaktpunkte übersetzt. Die Kontaktpunkte wurden in einem 3d-Dichteplot dargestellt. Bei den positiven Beispielen sollten sich „Spitzen“ ausbilden bzw. es sollte zu einer Häufung von Kontaktpunkten im Bereich einer TIS kommen. Bei den negativen Beispielen wurde eine Streuung der Kontaktpunkte erwartet.
Mit den erstellten Dichtplots wurde geprüft, ob mögliche konservierte Kontaktpunkte im Bereich einer TIS existieren. Diese Häufung von Kontaktpunkten könnte zu der Annahme führen, dass es mögliche konservierte RNA-Sekundärstrukturen geben könnte, die mit einem Genstart assoziiert sein könnten.
Nachdem die Dichteplots erstellt wurden, wurde eine mittelwertbasierte Klassifikation durchgeführt. Die Klassifikation sollte eine TIS Vorhersage auf der Grundlage von vorhergesagten RNA-Sekundärstrukturen machen.
Es konnte gezeigt werden, dass die Problematik der RNA-Sekundärstrukturvorhersage mit den hier verwendeten Programmen nicht vollständig gelöst ist. Aufgrund der beschriebenen Ungenauigkeiten (siehe Abschnitt 4.2.1, S. 85 und Abschnitt 4.2.2, S. 88) stellte sich ein Klassifikationsfehler von 39.8% ein.
Die RNA-Sekundärstrukturvorhersage Programme könnten zum einen durch genauere Energieparameter (siehe Tabelle 2.2, S. 33) und zum anderen durch das Einbeziehen der RNA-Tertiärstruktur erweitert werden. Damit wäre es denkbar eine Verbesserung der Vorhersage zu erzielen.

Literaturverzeichnis

[1] Barrik D., Villanueba K., Childs J., Kalil R., Schneider T.D. 1994. *Nucleic Acids Res.* **22**: 1287-1295.

[2] Cate J., Gooding A., Podell E., Zhou K., Golden B., Kundrot C., Cech T., Doudna J. 1996. Crystal structure of a group I ribozyme domain: principles of RNA packing. *Science,* **273**, S. 1678-1685.

[3] Conn, G. L., Draper E. D. 1998. RNA strukture. *Current Opinion in Structural Biology* Vol. **8**, S. 278-285.

[4] Flamm C., Fontana W., Hofacker I. L., Schuster P. 2000. RNA folding at elementary step resolution. *RNA*, **6**, 325-338.

[5] Gursinsky T., Jager J., Andreesen J. R., Sohling B. 2000. A selDABC cluster for selenocysteine incorporation in Eubacterium acidaminophilum. *Arch Microbiol.* **174**(3):200-12.

[6] Hafner R. 2001. Nichtparametrische Verfahren der Statistik. *Springer-Verlag Vienna* S. 75-95.

[7] Härdle W., Müller M. 1993. Nichtparametrische Glättungsmethoden in der alltäglichen statistischen Praxis. In *Allgemeines Statistisches Archiv,* **77**. Jg., 1993, S. 9-31.

[8] Hofacker I. L., Priwitzer B., and Stadler P. F. 2004. Prediction of Locally Stable RNA Secondary Struktures for Genome-Wide Surveys. *Bioinformatics*, Vol. **20**, 186-190.

[9] Hofacker I. L., Fontana W., Stadler P. F., Bonhoeffer S.,Tracker S., Schuster P. 1994. Fast folding and comparison of RNA secondary struktures. *Monatsh. Chemie*, **125**, 167-188.

[10] Hofacker I. L. 2003. Vienna RNA secondary structure server. In *Nucleic Acids Research* Vol. **31**, No. **13**, 3429-3431.

[11] Hofacker L. I., Bernhart H. F., Stadler F. P. 2004. Alignment of RNA Base Pairing Probability Matrices. *Bioinformatics* Vol. **20**, No. **14**, S. 2222-2227.

[12] Kenneth E. Rudd 2000. EcoGene: a genome sequence database for *Escherichia coli* K-12. In *Nucleic Acids Research*, Vol. **28**, No. **1**, 60-64.

[13] McCaskill J. S. 1990. The equilibrium Partition function and base pair binding probabilities for RNA secondary structure. *Biopolymers*, **29**, 1105-1119.

[14] Meinicke P., Tech M.,Morgenstern B., Merkl R. 2004. Oligo Kernels for Datamining on Biological Sequences: A Case Study on Prokaryotic Translation Initiation Sites. *BMC Bioinformatics* Vol. **5**:169.

[15] Nielsen H., Engelbrecht J., Brunak S., Heijne G. 1997. Identifation of prokaryotic and eukaryotic signal peptides and prediction of their cleavage sites. *Protein Engineering* **10**:1-6.

[16] Nussinov R., Piecznik G., Griggs J. R., Kleitman D. J. 1978. Algorithmus for loop matching. *SIAM j. Appl. Math.*, **35**, 68-82.

[17] Pleij C., Rietveld K., Bosch L. 1985. A new principle of RNA folding based on pseudoknotting. *Nucleic Acids Res.*, **13**, 1717-1731.

[18] Thadewald T. 1998. Uni- und bivariate Dichteschätzung. *Wirtschaftswissenschaftliche Dissertation, Berlin*, S. 4.

[19] Thirumalei D. 1998. Native secondary structure formation in RNA may be a slave to tertiary folding. *Proc. Nat. Acad. Sci. U.S.A.*, **95**, 11506-11508.

[20] Turner D. H., Sugimoto N., Freier S. M. 1990. Thermodynamics and kinetics of base-pairing and of DNA and RNA self-assembly and helix coil transition. In *Nucleic Acids, Subvolume c, Physical Data I, Spectroscopic and Kinetic Data.* (Saenger W., Hrsg.), Landolt-Börnstein, Group VII Biophysics, Vol I. Springer-Verlag, Berlin, S. 201-212.

[21] Varshavsky A. 1996. *Proc. Natl. Acad. Sci. USA* **93**: 12142-12149.

[22] Watson J. D., Crick F. H. C. 1953. Molecular strukture of nucleic acids: a structure for deoxyribose nucleic acid. *Nature*, **171**, 737-738.

[23] Wuchty S., Fontana W.,Hofacker I. L., Schuster P. 1999. Complete subobtimal folding of RNA and the stability of secondary structures. *Biopolymers*, **49**, 145-165.

[24] Zuker M.,Mathewes D. H. and Turner D. H. 1999. Algorithms and thermodynamics for RNA secondary structure prediction: A practical guide. In *RNA Biochemistry and Biotechnology.* (Barciszewski J., Clark B. F. C. Hrsg.). NATO ASI Series, Kluwer.

[25] Zuker M., Stiegler P 1981. Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information. *Nucleic Acids Res.*, **9**, 133-148.

Danksagung

Zunächst möchte ich mich bei Prof. Burkhard Morgenstern für die freundliche Aufnahme in seine Arbeitsgruppe und die intensive Betreuung dieser Arbeit bedanken. Weiterhin danke ich Dr. Peter Meinicke für seine wertvollen Ratschläge und intensive Unterstützung. Mein aufrichtiger Dank gebührt auch Maike Tech, die mir mit ihrer Betreuung stets zur Seite stand und diese Arbeit mit vielen interessanten Ideen und zahlreichen Diskussionen unterstützt hat.

Ein ganz besonderer Dank geht an meine Mutter, die mir das Studium der Biologie ermöglicht hat.

Nicht zuletzt möchte ich meiner Freundin Nina danken für alles, was sie in den letzten Jahren für mich getan hat.

Göttingen, den 14.02.2005