TECHNISCHE UNIVERSITÄT DRESDEN

Fakultät Informatik Institut für Systemarchitektur, Professur für Datenbanken

Diplomarbeit

METHODISCHE UNTERSTÜTZUNG DER MODELLAUSWAHL FÜR ENERGIEPROGNOSEN

Stefan Hinkel
Matr.-Nr.: 3402299

Betreut durch:
Dipl.-Wirtsch.-Inf. Robert Ulbricht

und:
Prof. Dr.-Ing. Wolfgang Lehner

Eingereicht am 19. Dezember 2016

ERKLÄRUNG

Ich erkläre, dass ich die vorliegende Arbeit selbständig, unter Angabe aller Zitate und nur unter Verwendung der angegebenen Literatur und Hilfsmittel angefertigt habe.

Dresden, 19. Dezember 2016

DANKSAGUNG

An dieser Stelle möchte ich mich von Herzen bei allen Begleitern und Partnern bedanken, welche mir geholfen haben, meinen Weg bis hin zu dieser Diplomarbeit zu gehen. Zunächst geht mein Dank an Robert Ulbricht, Dirk Habich, Martin Hahmann und Claudio Hartmann, welche mir gerade im universitären Umfeld eine große Hilfe waren. Hinsichtlich meiner Abschlussarbeit möchte ich mich bei der Robotron Datenbank-Software GmbH bedanken, die mir als Unternehmenspartner bei der Erstellung dieser Diplomarbeit unterstützend zur Seite stand. Weiterhin möchte ich meinen Dank auch an Katja Gruhnert richten. Danke, dass du mein wandelnder Duden warst. Ich danke ebenso sehr meiner Familie, die mich zu jeder Zeit in vollstem Vertrauen nach Kräften unterstützt hat. Das Wissen über diesen Rückhalt hat mich manche schlaflose Nacht mit nicht enden wollendem Lernstoff überstehen lassen. Ein besonderer Dank geht schließlich an meine Verlobte Peggy Strohbach, die mich mit unermüdlichem Einsatz immer wieder motiviert hat, sodass ich nie mein Ziel aus den Augen verlieren konnte. Danke, dass du stets für mich da warst und bist!

INHALTSVERZEICHNIS

1 Einleitung **1**

1.1 Motivation 1

1.2 Strukturierung der Arbeit 2

2 Themenbezogene Arbeiten **5**

2.1 Klassische Distanz- und Ähnlichkeitsmaße 5

2.2 Eigenschaftsextraktion und Dimensionsreduktion 10

2.3 Allgemeines Vorgehensmodell 12

3 Eigenschaftsbasierte Repräsentation von Zeitreihen **15**

3.1 Numerische und statistische Eigenschaften von Zeitreihen 15

3.2 Aufbau und Zusammensetzung einer Datenbasis für Zeitreihen 21

3.3 Analyse der Eigenschaften von Zeitreihen 24

4 Methodische Unterstützung der Modellauswahl für Prognoseprozesse **29**

4.1 Das Benchmark Framework ECAST 29

4.2 Zeitreihenklassifikation anhand von Vorhersagemethodiken 31

4.3 Architekturerweiterung des Systems ECAST 32

5 Evaluation des 1-Nearest-Neighbor Klassifikators **35**

5.1 Methodologie 35

5.1.1 Entwurf und Realisierung einer Evaluationsdatenbank 36

5.1.2 Verwendete Modelle und Methodik zur experimentellen Zeitreihenvorhersage 37

5.1.3 Aufbau einer Experimentenserie zur empirischen Klassifikatorevaluation . 40

5.2 Evaluation des 1-Nearest-Neighbor Klassifikators 41

5.2.1 Kontextfreie Evaluation 41

5.2.2 Evaluation unter Einbeziehung nicht-numerischer Kontextinformationen . 45

6 Evaluation weiterer Klassifikatoren **49**

6.1 Entscheidungsbaumbasierte Klassifikation 49

6.2 Verbundklassifikator TP1NN 54

6.3 Künstliches Neuronales Netz 59

6.4 Evaluation gegen Random-Walk-Methodiken 61

6.5 Evaluation gegen rohwertbasierte euklidische Distanz 64

6.6 Gegenüberstellung von Laufzeit und Performanz aller betrachteter Klassifikatoren 66

7 Zusammenfassung und Ausblick **69**

7.1 Fazit 69

7.2 Anregungen für zukünftige Arbeiten und Analysen 71

Literaturverzeichnis **73**

Abbildungsverzeichnis **79**

Tabellenverzeichnis **81**

Abkürzungsverzeichnis **83**

A SMAPE-Diagramme aller Klassifikatoren **85**

1 EINLEITUNG

1.1 MOTIVATION

In der heutigen Welt stehen wir einer Unmenge von erfassten Daten gegenüber. Mit der stetig wachsenden Anzahl von persönlichen elektronischen Wegbegleitern im Alltag, wie Smartphones, Tablets und ähnlichem, und der wachsenden Anzahl technischer Neuerungen zur schnellen Datenerfassung und -speicherung, wie etwa die Geräte der Smart-Home-Bewegung, bildete sich über die Jahre ein eigenständiger Marktzweig zur Analyse solcher Daten. Mittlerweile haben solche Analysen und das daraus gewonnene Wissen ihren Weg in die moderne Unternehmensplanung gefunden, sodass zukünftige Anforderungen des Marktes frühzeitig erkannt werden und in die Unternehmensstrategie mit einfließen können. Aus diesem Grund ist es von essentieller Wichtigkeit, solide Methodiken zur Daten- und somit Anforderungsvorhersage verfügbar zu haben.

Ein besonderes Interesse diesbezüglich weist der Energiesektor auf. In diesem müssen erwartete Anforderungen an die zu erzeugende Energiemenge so präzise wie möglich vorhergesagt werden. Mit der Einführung erneuerbarer Energiequellen wie etwa Windkraft-, Wasserkraft- und Solaranlagen, bekommt die Produktionsvorhersage jedoch einen weiteren Schwierigkeitsgrad hinzu, da die Energieeinspeisung durch solche Quellen zumeist durch ein unstetes Verhalten charakterisiert ist. Dies begründet sich dadurch, dass erneuerbare Energiequellen im Regelfall von unkontrollierbaren Umweltfaktoren abhängen, welche einen direkten Zusammenhang zur erzeugbaren Energiemenge aufweisen. Kurzum, die Vorhersage des zu erwartenden Energiebedarfs ist ein äußerst komplexes Unterfangen und bedarf geeigneter komplexer Modelle zur Beschreibung.

Aus diesem Grund wurde in Industrie und Forschung in den vergangenen Jahren eine Vielzahl von Prognosemodellen entwickelt, welche diese Aufgabe meistern sollen. Da jedoch solche Modelle meist spezielle Charakteristiken und Aspekte einzelner Prognoseprozesse begünstigen, ergeben sich mitunter drastische Unterschiede in der Vorhersagequalität bei der bereichsübergreifenden Verwendung. Es stellt sich daher die Frage, ob es Ähnlichkeiten zwischen unterschiedlichen Prognoseprozessen gibt und inwieweit sich diese erkennen und effektiv ausnutzen lassen,

um die Auswahl des besten Modells aus der Vielzahl verfügbarer Modelle zu unterstützen.

Als triviale Lösung dieses Problems bietet sich unmittelbar ein Verarbeitungsmodell an, bei dem die gesammelten zeitbezogenen Datenaufzeichnungen, sogenannte Zeitreihen, mittels eines strukturbasierten Vergleichs analysiert werden. Hierbei kommt ein mathematisches Modell zur Anwendung, in welchem Zeitreihen als hochdimensionale Vektoren existieren und ein sogenanntes Distanzmaß, wie beispielsweise die euklidische Distanz, einen standardisierten räumlichen Abstand zwischen den Vektoren ermittelt. Damit ist es grundlegend möglich, Gruppen von ähnlichen Zeitreihen durch Anwendung von Methoden des Clusterings und der Klassifikation zu isolieren. So kann zu jeder Zeitreihe, welcher noch kein konkretes Vorhersagemodell zugeordnet ist, ein Modell vorgeschlagen werden, welches mit dem Modell der ähnlichsten bisher analysierten Zeitreihe korreliert.

Die Nachteile dieses Verarbeitungsmodells zeigen sich jedoch sehr schnell, gerade im Energiesektor. Da die dort vorkommenden Datenerfassungen nicht selten aus einer Menge von mehreren tausend bis zehntausend Beobachtungen bestehen, ergibt sich im Vektormodell eine enorme Dimensionalität der einzelnen Zeitreihen. Dies führt unmittelbar zu einer drastischen Erhöhung des individuellen Berechnungsaufwandes, wodurch sich die Mehrarbeit einer Ähnlichkeitssuche als Vorverarbeitung immer weniger amortisiert.

Überdies unterliegen klassische Distanzmaße einer stringenten Restriktion, da sie grundlegend nur auf Vektoren derselben Dimensionalität definiert sind. In realen Anwendungsszenarien kann sich so schnell die potentielle Vergleichsmenge ungewollt reduzieren, da beispielsweise durch technisches Versagen Aufzeichnungsfehler oder -ausfälle entstehen, welche zu einer lückenhaften Messreihe führen. In diesen Fällen ist ein solchartiger Vergleich unmöglich, wodurch die ähnlichste Zeitreihe möglicherweise nicht gefunden werden kann. Die Vorverarbeitung hätte somit lediglich eine Verzögerung der allumfassenden „Trial and Error“-Methodik der Anwendung und Evaluation aller verfügbaren Vorhersagemodelle erwirkt.

Aus diesem Grund versuchten einige Arbeiten Möglichkeiten und Wege zu finden, um die Suche nach einem optimalen Modell wieder zu begünstigen. Hierbei stützten sich die Betrachtungen und Analysen alternativer Verarbeitungsansätze auf die Repräsentation von Zeitreihen unter Zuhilfenahme numerischer Zeitreiheneigenschaften. Nach wie vor stellt sich jedoch die Frage, wie und auf Basis welcher Eigenschaften eine Modellempfehlung für Prognoseprozesse zu realisieren ist. Diese Frage bildet den Ausgangspunkt dieser Arbeit.

1.2 STRUKTURIERUNG DER ARBEIT

Diese Arbeit widmet sich der methodischen Unterstützung der Modellauswahl für generelle Prognoseprozesse und speziell für Energieprognosen. Hierzu erfolgt in Kapitel 2 die Betrachtung und Diskussion von Arbeiten, welche sich bereits mit dieser Thematik befasst haben. Gegenstand sind dabei sowohl Arbeiten zu Distanzmaßen im Allgemeinen als auch zu bestehenden Ansätzen eigenschaftsbasierter Darstellungen und Verarbeitungen von Zeitreihen. Anschließend wird eine Menge numerischer und statistischer Kennzahlen auf Zeitreihen zusammengetragen und vorgestellt (Kapitel 3). Der Aufbau einer Datenbasis, welche später zu einer Fallbasis avanciert,

ist hierin ebenso thematisiert wie die Verdichtung einer Ausgangsmenge von Eigenschaftswerten auf ein Extrakt der vielversprechendsten Kandidaten für die Modellklassifikation. Das reduzierte Extrakt bildet den Ausgangspunkt für Kapitel 4, in welchem sukzessive auf die Architektur einer Modellempfehlungskomponente hingearbeitet wird. Die Kapitel 5 und 6 widmen sich der Evaluation der Leistung der neu entwickelten Empfehlungskomponente über einer exemplarischen Fallbasis. Schließlich werden alle Erkenntnisse der Arbeit zusammengefasst dargestellt und ein Ausblick darauf gegeben, welche unbetrachteten Aspekte der Thematik Ansatzpunkte für zukünftige Arbeiten liefern können (Kapitel 7).

2 THEMENBEZOGENE ARBEITEN

In dieser Arbeit wird die methodische Unterstützung der Modellauswahl für Prognoseprozesse am Beispiel von Energieprognosen in heterogenen Datensätzen untersucht. Wie in Kapitel 1 bereits erwähnt wurde, basiert eine erfolgreiche Empfehlung zur Modellauswahl auf einer hinreichend soliden Klassifikation von Prognoseprozessen anhand ihrer optimalen Modelle. Auf Grund dessen, dass bereits einige Erkenntnisse in dieser Thematik existieren, widmet sich dieses Kapitel der Aufstellung verwandter Arbeiten. Hierfür wird in Abschnitt 2.1 zunächst eine Übersicht über klassische Distanz- und Ähnlichkeitsmaße gegeben, welche zur rohwertbasierten Zeitreihenklassifikation verwendet werden können. Da die rohwertbasierte Verarbeitung einige Nachteile mit sich bringt, werden anschließend Arbeiten zur Eigenschaftsextraktion und Dimensionsreduktion gezeigt, welche diese Schwächen überwinden (Abschnitt 2.2). Schließlich enthält Abschnitt 2.3 eine Vorstellung des Vorgehensmodells Case-based Reasoning, als Methode des maschinellen Lernens und spätere Grundlage des weiteren Systementwurfs (siehe Kapitel 4).

2.1 KLASSISCHE DISTANZ- UND ÄHNLICHKEITSMASSE

Die Frage nach der Bestimmung ähnlicher Zeitreihen aus einer Menge verfügbarer Reihen ist in der Literatur bereits eingängig behandelt worden. Zentraler Ausgangspunkt ist hierbei zumeist die Tatsache, dass eine Zeitreihe aus n Beobachtungen problemlos als n-dimensionaler Vektor darstellbar ist. Zur Lösung des Problems der Ähnlichkeitsbestimmung bieten sich damit unmittelbar sämtliche Möglichkeiten unter Verwendung klassischer Distanzmaße an.

Bis heute entstand eine Vielzahl von Koeffizienten zur Bestimmung des räumlichen Abstands zwischen zwei Vektoren eines n-dimensionalen Raumes. Einige der wohl bekanntesten Maße sind die *euklidische Distanz* und die *Manhattan-Metrik*, welche beide Spezialfälle der *Minkowski-Distanz* ([JMF+00]) darstellen. Diese und weitere teils komplexe Distanz- und Ähnlichkeitsmaße – wie der Kreuzkorrelationskoeffizient nach Pearson ([War05], [RN88]) oder die Hauptkomponentenanalyse (engl. Principal Component Analysis (PCA)) ([Pea01], [Hot33]) – sind ausführlich in [LVVC11] betrachtet. Die jeweiligen formalen Definitionen einiger dieser Maße sind in Tabelle 2.1 enthalten.

Minkowski-Distanz	$d_{minkowski}(x,y) = \left(\sum_{i=1}^{n} \lvert x_i - y_i \rvert^p \right)^{\frac{1}{p}}$
Manhattan-Metrik	$d_{manhattan}(x,y) = \sum_{i=1}^{n} \lvert x_i - y_i \rvert$
Euklidische Distanz	$d_{euklid}(x,y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$
Kreuzkorrelationskoeffizient nach Pearson	$d_{CC}(x,y) = \frac{\sum_{i=1}^{n} [(x_i - \overline{x}) \cdot (y_{i-l} - \overline{y})]}{\sqrt{\sum_{i=1}^{n} (x_i - \overline{x})^2} \cdot \sqrt{\sum_{i=1}^{n} (y_{i-l} - \overline{y})^2}}$
Kosinus-Ähnlichkeit	$d_{kosinus}(x,y) = \frac{\sum_{i=1}^{n} x_i \cdot y_i}{\sqrt{\sum_{i=1}^{n} x_i^2} \cdot \sqrt{\sum_{i=1}^{n} y_i^2}}$

Tabelle 2.1: Formale Definitionen gebräuchlicher Distanz- und Ähnlichkeitsmaße

Mit Hilfe dieser Maße ist es möglich, eine Aussage darüber zu treffen, welche der zur Verfügung stehenden Zeitreihen zu einer ausgewählten Reihe *strukturell* am ähnlichsten ist. Zur Illustration dieses Vorgehens kann Abbildung 2.1 betrachtet werden. Die dort dargestellte Menge besteht aus vier Zeitreihen, welche alle eine Länge von 20 Beobachtungen aufweisen. Durch die gleichen Längen sind alle Maße aus Tabelle 2.1 anwendbar. Zu jedem Maß ist eine Distanzmatrix erstellbar, welche die räumlichen Abstände je möglichem Paar von Zeitreihen aufzeigt. Abbildung 2.2 beinhaltet diese Distanzmatrizen.

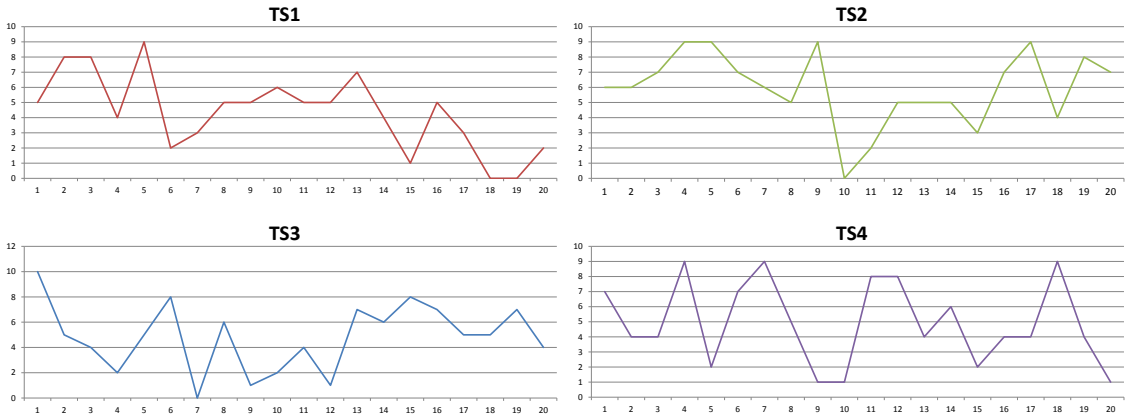


Abbildung 2.1: Eine Menge von Zeitreihen gleicher Länge

Euklid				
	TS1	**TS2**	**TS3**	**TS4**
TS1	0,00	**16,73**	17,44	18,55
TS2	16,73	0,00	**16,55**	17,20
TS3	17,44	**16,55**	0,00	17,49
TS4	18,55	**17,20**	17,49	0,00

Manhattan				
	TS1	**TS2**	**TS3**	**TS4**
TS1	0	**60**	68	70
TS2	**60**	0	**60**	**60**
TS3	68	**60**	0	**60**
TS4	70	**60**	**60**	0

Minkowski (p = 3)				
	TS1	**TS2**	**TS3**	**TS4**
TS1	0,00	**11,54**	11,62	12,64
TS2	**11,54**	0,00	**11,54**	11,99
TS3	11,62	**11,54**	0,00	12,44
TS4	12,64	**11,99**	12,44	0,00

Minkowski (p = 5)				
	TS1	**TS2**	**TS3**	**TS4**
TS1	0,00	9,04	**8,82**	9,96
TS2	**9,04**	0,00	9,18	9,40
TS3	**8,82**	9,18	0,00	10,08
TS4	9,96	**9,40**	10,08	0,00

Kreuzkorrelation				
	TS1	**TS2**	**TS3**	**TS4**
TS1	0,00	**0,96**	1,15	1,24
TS2	**0,96**	0,00	1,01	1,06
TS3	1,15	**1,01**	0,00	1,09
TS4	1,24	**1,06**	1,09	0,00

Kosinus				
	TS1	**TS2**	**TS3**	**TS4**
TS1	0,00	**0,19**	0,27	0,30
TS2	0,19	0,00	**0,18**	0,20
TS3	0,27	**0,18**	0,00	0,25
TS4	0,30	**0,20**	0,25	0,00

Abbildung 2.2: Distanzmatrizen der Zeitreihen aus Abbildung 2.1 unter Verwendung der Distanz- und Ähnlichkeitsmaße aus Tabelle 2.1

Bei genauerer Betrachtung der behandelten Distanz- und Ähnlichkeitsmaße fällt auf, dass diese spezielle Anforderungen an eine darunterliegende Zeitreihendatenbank stellen, um korrekt funktionieren zu können. Distanz- und Ähnlichkeitsmaße sind überwiegend nur auf zwei Vektoren derselben Dimensionalität definiert. Tabelle 2.1 veranschaulicht diese Aussage, da als obere Grenze der Summen in den formalen Definitionen der Term n angegeben ist, welcher die Länge der verarbeiteten Vektoren bezeichnet. Die Anwendung dieser Verfahren in Zeitreihendatenbanken der realen Welt führt jedoch dazu, dass Fälle auftreten, bei denen eine Ähnlichkeitssuche entweder nur eingeschränkt oder sogar überhaupt nicht möglich ist.

Dieser Umstand wird vor allem dann deutlich, wenn folgendes Beispiel betrachtet wird. Sei $Y' = \{Y_1, \cdots, Y_n\}$ eine Menge von Zeitreihen $Y_i = \{y_1, \cdots, y_{t_i}\}$, wobei $1 \leq i \leq n$ und $t_i = |Y_i|$. Des Weiteren seien alle Y_i sowohl aus ein und derselben Domäne als auch aus ein und demselben Datensatz. Auf Grund nicht kontrollierbarer Einflüsse[1] beinhaltet Y_1 einige Beobachtungen weniger als die verbleibenden Zeitreihen in Y'. Für die Menge verbleibender Zeitreihen $Y' \setminus Y_1$ gilt $\forall Y_v, Y_w \in Y' \setminus Y_1 \colon |Y_v| = |Y_w|$. Ohne Beschränkung der Allgemeinheit ist $Y^C = \{Y_1, \cdots, Y_k\}$ mit $1 \leq k < n$ objektiv betrachtet ein idealer Cluster in diesem Szenario.

Im eben eingeführten Beispiel könnte unter Verwendung eines klassischen Distanzmaßes zu Y_1 nie eine ähnliche Zeitreihe gefunden werden. Dies ist damit zu begründen, dass Y_1 eine andere Dimensionalität besitzt als alle sowohl in Y^C als auch in Y' verbleibenden Zeitreihen. Das Verfah-

[1] Solche Einflüsse können beispielsweise Störungen bei der Datenerhebung durch Materialversagen oder wirtschaftliche bzw. politische Entscheidungen sein.

ren scheitert für Zeitreihe Y_1 an dieser stringenten Restriktion.

Zusätzlich zu klassischen Distanzmaßen existieren auch rohwertbasierte Distanzverfahren, welche robust gegenüber einer Verzerrung der Dimensionierung sind. Das wohl bekannteste Verfahren bildet der Dynamic Time Warping (DTW) Algorithmus ([SC78]). Mit Hilfe von DTW ist es möglich, zwei Signale aufeinander abzubilden, indem ein sogenannter Warping-Pfad ermittelt wird. Dieser Pfad entsteht durch die Erstellung einer 2-dimensionalen Matrix D, deren Dimensionen durch je ein Signal bestimmt sind. Zu Beginn des Algorithmus erfolgt die Initialisierung von D, welche der Formel

$$D_{i,j} = \begin{cases} 0 & \text{, wenn } i = 0 \text{ und } j = 0 \\ \infty & \text{, sonst} \end{cases}$$

folgt. Anschließend wird die Matrix nach folgender Formel ausgefüllt:

$$D_{i,j} = \left(x_i - y_j\right)^2 + min\left\{D_{i,j-1}, D_{i-1,j}, D_{i-1,j-1}\right\} \tag{2.1}$$

Die Variablen x und y bezeichnen dabei die beiden Signale mit den jeweiligen Längen $n = |x|$ und $m = |y|$. Es ist zu erkennen, dass DTW nicht nur die bloße Distanz zwischen zwei Ausprägungen einer Dimension berücksichtigt, sondern auch die „Umgebung“ der jeweiligen Zellen der Matrix betrachtet. Wie in Formel 2.1 zu sehen ist, wird zur quadrierten Distanz der minimale Wert aus den jeweils relativ westlich, nord-westlich und nördlich benachbarten Zellen aus D addiert. Terminiert der Algorithmus in $D_{n-1,m-1}$, lässt sich ausgehend von diesem Feld der Warping-Pfad durch Rückverfolgung des jeweils ausgewählten, addierten Minimums rekonstruieren. Wird DTW als Distanzmaß angewandt, stellt die Länge des Warping-Pfades die entsprechende Distanz zwischen den beiden Eingangssignalen dar.

Folgendes Beispiel dient dazu, das Vorgehen von DTW zu demonstrieren. Abbildung 2.3 zeigt zwei Signale, welche mittels DTW aufeinander abgebildet werden sollen. Nach den oben genann-

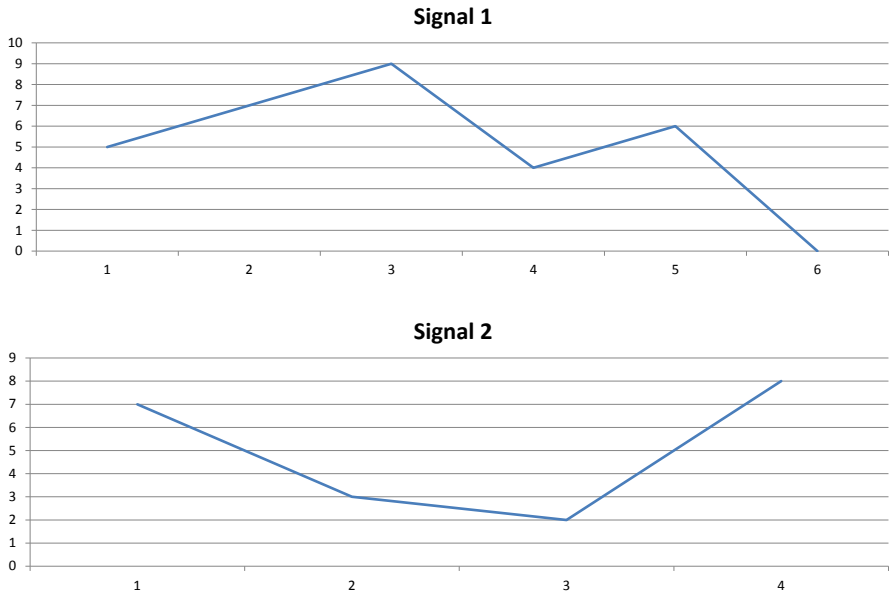


Abbildung 2.3: Zwei Signale unterschiedlicher Länge

ten Definitionen erfolgt die Erstellung und Ausfüllung einer Warp-Matrix, welche in Tabelle 2.2 dargestellt ist. Aus dieser Tabelle ist ersichtlich, dass zur Abbildung der beiden Signale aus Abbildung 2.3 ein Warping-Pfad der Länge[2] 6 entsteht. Wären diese Signale nun Zeitreihen und DTW das verwendete Distanzmaß, wäre der Wert 6 der korrespondierende Distanzwert beider Zeitreihen.

DTW Warp-Matrix				
	7	**3**	**2**	**8**
5	4	4	6	9
7	0	16	29	7
9	4	36	65	8
4	9	5	9	24
6	1	10	21	13
0	49	10	14	77

Tabelle 2.2: Warp-Matrix der Signale aus Abbildung 2.3

DTW bietet somit eine robuste Möglichkeit, Zeitreihen miteinander zu vergleichen, die beispielsweise dieselbe Größe protokollieren, jedoch eine unterschiedliche Auflösung besitzen. Laut den Angaben in [SA14] wurde der Algorithmus in der Vergangenheit bereits zur Zeitreihenähnlichkeit betrachtet und verwendet ([BC94], [TGQS09]). Der operative Einsatz von DTW als verzerrungsrobustes Distanzmaß ist allerdings auf Grund der hohen Zeit- und Speicherkomplexität von $O(n \cdot m)$ nicht praktikabel. Es gibt nach den Angaben in [SA14] zwar Ansätze, diese Komplexitäten zu reduzieren ([SC07], [KR05], [Lem09]), jedoch gehen diese zumeist mit Kompromissen in der Genauigkeit der Ergebnisse einher.

Zur Demonstration der Zeitkomplexität von DTW soll folgendes minimalistisches Experiment dienen. Der Versuch wurde auf einem Laptop unter Windows 8.1 Professional x64 unter Verwendung einer Intel(R) Core(TM) i7-3630QM CPU @ 2.40GHz und 8GB RAM durchgeführt. Als ausführende Software kam die freie Statistiksoftware *R*[3] in der Version 3.3.1 zum Einsatz.

In einem exemplarischen Benchmark wurden die euklidische Distanz d_{euklid} und DTW d_{dtw} auf einer aus 10 Zeitreihen bestehenden, generierten Menge Z getestet. Die Zeitreihen weisen je eine Länge von 8.760 Beobachtungen auf. Dies könnte beispielsweise dadurch zustande gekommen sein, dass ein Messsystem über einen Zeitraum von 365 Tagen stündlich einen Messwert erhoben hat. Alle Werte wurden in *R* in eine Matrix generiert unter Verwendung der Funktion `rnorm`. Zusätzlich zur Matrix ist mit der gleichen Methodik ein 8.760-dimensionaler Vektor v als Vergleichszeitreihe erstellt worden. Mit Hilfe der Funktion `dist` erfolgte anschließend die Berechnung der Distanzmatrix $v \times Z$. Diese degeneriert auf Grund dessen, dass nur von einer Zeitreihe die Distanz zu allen Zeitreihen aus Z ermittelt werden muss, zu einem Distanzvektor.

Als Ergebnis des Versuchs stehen die Zeiten $t(d_{euklid}) = 0,003sec$ und $t(d_{dtw}) = 80,498sec$, welche die Berechnungszeiten der Distanzvektoren unter Verwendung des jeweiligen Distanzmaßes bezeichnen. Schon dieses Minimalbeispiel zeigt, dass DTW signifikant längere Berechnungszeiten als klassische Distanzmaße wie etwa die euklidische Distanz aufweist – in diesem Fall (gerundet)

[2] Als Länge des Warping-Pfades wird die Anzahl der Schritte in der Matrix festgelegt.
[3] `https://www.r-project.org`

um den Faktor 26.800! Dies demonstriert deutlich, dass DTW keine praktikable Lösung in einem operativen System darstellt.

Ein weiteres Problem ist der gleichartig expandierende Speicherbedarf von DTW. Während der Distanzvektor bei Benutzung der euklidischen Distanz quasi in-place ermittelt werden kann, entsteht bei DTW je Dimension des Distanzvektors eine 8.760×8.760 große Warp-Matrix. Wird nun in Betracht gezogen, dass Energiezeitreihen häufig eine wesentlich feinere Auflösung von bis zu mehreren zehntausend Beobachtungen je Reihe aufweisen, ist unmittelbar zu erkennen, dass die zugehörigen Warp-Matrizen eine nicht unwesentliche Menge an Speicher benötigen. Zu klein dimensionierte Systeme wären somit gezwungen Methoden des Speicher-Swappings bis hin zu komplett plattenbasierten Berechnungsverfahren anzuwenden, um mit solch großen Datenobjekten umgehen zu können. Die Anwendung dieser Verfahren reduziert die Berechnungsperformanz jedoch zusätzlich.

Eine weitere Aussage in [SA14] ist, dass einige Autoren eine spekulative Meinung im Bezug auf klassische Distanzmaße vertreten. In [WMD+13] werde beispielsweise konstatiert, dass andere Verfahren zur Ähnlichkeitsbestimmung die Genauigkeit eines Distanzmaßes – besonders der euklidischen Distanz – wohl nicht erreichen könnten. Daraus ließe sich schlussfolgern, dass aus Gründen der Genauigkeit ein Distanzmaß gegenüber einem alternativen Ansatz stets zu bevorzugen ist. Hinsichtlich der stringenten Restriktionen bzw. der hohen Zeit- und Speicherkomplexität klassischer Distanz- und Ähnlichkeitsmaße, ist von einer operativen Verwendung solcher Verfahren jedoch abzuraten.

2.2 EIGENSCHAFTSEXTRAKTION UND DIMENSIONSREDUKTION

Angesichts der in Abschnitt 2.1 erläuterten Nachteile wertebasierter Zeitreihenverarbeitung, widmeten sich einige Arbeiten der Eigenschaftsextraktion und Dimensionsreduktion. [KCPM01] zeigt, dass schon simple Transformationen der originalen Zeitreihen in niederdimensionale Problemräume viele Vorteile gegenüber einer Verarbeitung der Rohwerte mit sich bringen. Diese sind unter anderem eine wesentliche Steigerung der Berechnungsperformanz und die Längenunabhängigkeit beim Vergleich von einzelnen Zeitreihen.

[NAM01] demonstriert eine Klassifikation von Zeitreihen auf Basis von Eigenschaftswerten. Darin betrachten die Autoren grundlegende statistische Eigenschaften, wie das arithmetische Mittel, die Standardabweichung, die Schiefe und die Wölbung. Mit Hilfe der genannten Merkmale, unter Einbeziehung eines Multi-layer Perceptron (MLP) Neuronalen Netzwerks als Klassifikator, wurde gezeigt, dass auf sogenannten Control Chart Patterns (CCPs)[4] wesentlich effizientere Klassifikationen erreichbar sind als mit einer trivialen Klassifikation auf den Beobachtungen der Zeitreihen. Im Fall von verrauschten Daten erweist sich, laut den Autoren, die Klassifikationsgenauigkeit dieses eigenschaftsbasierten Ansatzes der Genauigkeit der trivialen Methodik als überlegen. Hinsichtlich großer Zeitreihen – also solchen hoher Dimensionalität – werden sowohl schnellere als auch bessere Klassifikationsergebnisse konstatiert.

[4] CCPs sind Zeitreihen, welche zur Repräsentation kritischer Systemparameter verwendet werden, wie zum Beispiel Elektrokardiogramm (EKG)-Aufzeichnungen, oder Wettermessungen.

Die Autoren weisen jedoch darauf hin, dass die alleinige Verwendung der vier eingangs erwähnten Eigenschaften nicht notwendigerweise die schlussendliche Menge berechenbarer Merkmale darstellt, um Zeitreihen hinreichend akkurat zu klassifizieren. Sie empfehlen daher die Betrachtung weiterer Eigenschaften, wie beispielsweise der zentralen Momente. Darüber hinaus sprechen sie die Empfehlung aus, das MLP Neuronale Netzwerk durch andere Klassifikatoren zu ersetzen, um so die Klassifikationsgenauigkeit möglicherweise noch zu erhöhen.

In [WSH06] wird das Framework Characteristic-Based Clustering (CBC) vorgeschlagen, welches zum Clustering von Zeitreihen verwendet werden kann. Hierbei zeigen die Autoren, dass eine Menge von 13 ausgewählten Eigenschaften Clusterings von hoher Güte erzeugen kann. Der Clustering-Algorithmus hängt in diesem Ansatz nicht mehr von den Rohwerten der Zeitreihe ab, sondern arbeitet direkt auf den Eigenschaftswerten. Laut den Autoren überwindet die postulierte Methodik so die in Abschnitt 2.1 erwähnten Beschränkungen.

Die Erkenntnis der Arbeit aus [WSH06] endet jedoch im Clustering von Zeitreihen auf Basis der extrahierten Merkmale. Durch die Einteilung in kompakte Cluster ist ein guter Ausgangspunkt für die Klassifikation von Zeitreihen unter Verwendung numerischer Eigenschaften gegeben. Allerdings gründet diese Klassenbildung auf der Verteilung der Eigenschaftswerte im n-dimensionalen Eigenschaftsraum. Es wird nicht klar, ob diese Verteilung auch eine hinreichend homogene Verteilung der besten Vorhersagemodelle impliziert.

In [KHSm16] wird die Idee der eigenschaftsbasierten Darstellung von Zeitreihen weiter geführt, indem die extrahierten statistischen Werte zur Visualisierung von Zeitreihen im 2-dimensionalen Raum verwendet werden. Hierfür berechnen die Autoren insgesamt sechs Kennzahlen, welche durch eine PCA in den 2-dimensionalen Raum transformiert werden. Des Weiteren wird erläutert, dass mittels der Extraktion von charakteristischen Eigenschaften jede Zeitreihe durch eine vergleichsweise kleine Menge numerischer Werte darstellbar ist. Dies eliminiert das Problem fluktuierender Längen bei Zeitreihenvergleichen, wie schon in [WSH06] bemerkt wurde.

Neben der Herausstellung der Vorteile eigenschaftsbasierter Zeitreihendarstellungen verfolgt diese Arbeit jedoch ein anders geartetes Ziel, da die Eigenschaften als Grundlage einer Visualisierung benutzt werden. Auch die Angabe eines Algorithmus zur kontrollierten Erzeugung neuer Zeitreihen weicht vom Ziel dieser Diplomarbeit ab. Einzig die Demonstration guter Leistungen der Visualisierung in der Clusterbildung auf den getesteten Daten fördert das Interesse an den vorgestellten statistischen Eigenschaften. Aus diesem Grund werden diese in Kapitel 3 detaillierter analysiert.

Andere, frühe Ansätze wie [AFS93] und [FF94] vollziehen die Transformation von Zeitreihen in sogenannte Fourier-Koeffizienten. Auf Basis dieser Koeffizienten wird dann die euklidische Distanz zwischen den transformierten Darstellungen ermittelt und als Ähnlichkeitsgrundlage verwendet. Die Autoren ziehen dabei sowohl generierte als auch reale Finanzdaten in Betracht.

Obwohl die gezeigten Ansätze gute Ergebnisse erzielen, ist die Transformation von Zeitreihen in Fourier-Koeffizienten ein parametriertes Verfahren. Diese Parameter, wie beispielsweise die Anzahl der zu verwendenden Fourier-Koeffizienten, müssten in einem laufenden System geschätzt und optimiert werden. Dadurch erscheinen die dort gezeigten Ansätze unattraktiv und werden nicht weiter betrachtet. Nichtsdestotrotz zeigen Arbeiten wie [AFS93], [FF94] und [SA14], dass

hohes Potential in der Eigenschaftsextraktion und Dimensionsreduktion liegt, um Zeitreihen effektiv und effizient miteinander vergleichen zu können.

2.3 ALLGEMEINES VORGEHENSMODELL

In der Zeitreihenvorhersage ist es kein Novum, Methodiken und Algorithmen des maschinellen Lernens anzuwenden ([ASEA99], [BTB14]). Das gleiche gilt auch im Bezug auf die Klassifikation von Zeitreihen. Besonders interessant erscheint aus der Literatur der Ansatz des fallbasierten Schließens (engl. Case-based Reasoning (CBR)). CBR beschreibt nach [ASEA99] einen Prozess, dessen zentrales Konstrukt eine sogenannte Fallbasis darstellt. Das Modell ist in vier grundlegende Elemente unterteilt, welche den sogenannten CBR-Zyklus formen. Diese Elemente sind als die vier *Rs* – Retrieve (I), Reuse (II), Revise (III) und Retain (IV) – bekannt.

Teil (I) dieses viergliedrigen Verfahrens bezeichnet den Schritt *Retrieve*, also „gewinnen“. Dabei soll aus besagter Fallbasis ein schon gelöstes Problem gewonnen werden, welches dem neuen zu lösenden Problem am ähnlichsten ist. Dieses herausgefilterte, ähnlichste Problem wird anschließend in Teil (II) wiederverwendet (engl. *reuse*), um das neue Problem anhand der „alten“ Lösung zu behandeln. Es wird prinzipiell kaum einen Fall geben, bei dem ein neues Problem in vollem Maße einem bereits eruierten Problem entspricht und sich durch dessen bekanntes Ergebnis vollumfänglich zufriedenstellend lösen lässt. Aus diesem Grund erfolgt in Teil (III) die Revision (engl. *to revise*) des Ergebnisses der durch die Fallbasis vorgeschlagenen Lösung. Nach etwaigen manuellen Änderungen an den Lösungsparametern, schließt sich als Teil (IV) die Einbringung des tatsächlichen Lösungsweges in die Fallbasis an. Diese beinhaltet (engl. *to retain*) den neuen Fall anschließend, sodass er in die Erzeugung von Lösungsvorschlägen für neue Problemstellungen einbezogen werden kann. Dieser zyklische Systemablauf ist in Abbildung 2.4 illustriert.

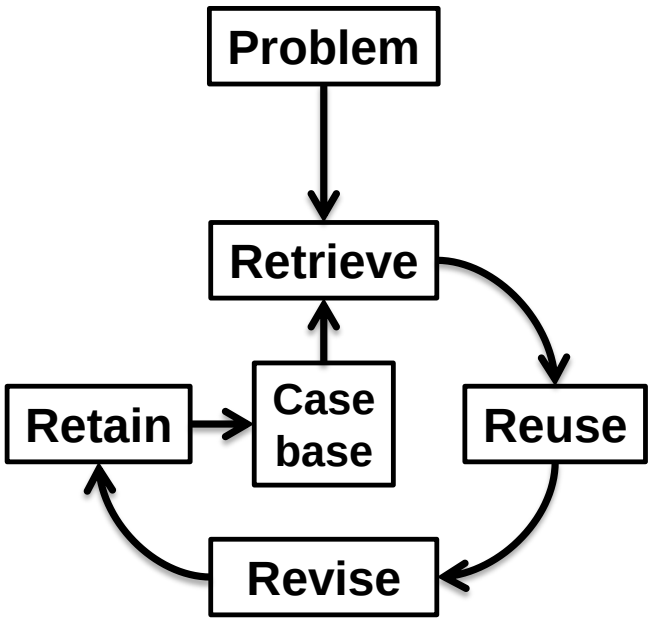


Abbildung 2.4: Der CBR-Zyklus des Case-based Reasoning

Auf Grund der eben vorgestellten Funktionsweise erscheint CBR als sehr gut geeignet, um als Grundlage der methodischen Unterstützung der Modellauswahl für Prognoseprozesse zu fungieren. Ein so fundiertes System würde in einem stetig lernenden Modul resultieren, welches auf Grund verfügbarer Modellklassifikationen neue Modellempfehlungen ableitet. Da die eingebrachten Modellklassifikationen entweder durch Expertenwissen oder durch sukzessives Testen einer Vielzahl verfügbarer Modelle entstanden sind, verspricht CBR bei geeignetem Ähnlichkeitsverfahren qualitativ hochwertige Empfehlungen.

Schließlich wird durch die Betrachtung der aufgeführten themenbezogenen Arbeiten deutlich, dass eine effiziente Ähnlichkeitsbestimmung zwischen Zeitreihen auf der Grundlage der Extraktion charakteristischer Merkmale zu erfolgen hat. Die Anwendung eines solchen Verfahrens reduziert die Problemgröße durch Transformation in überschaubare niederdimensionale Räume mitunter signifikant. Darüber hinaus verringert sich die Sensibilität gegenüber fehlenden Beobachtungen innerhalb der Zeitreihen. Das Problem der stringenten Längenabhängigkeit üblicher, rohwertbasierter Ähnlichkeitsverfahren wird ebenfalls eliminiert.

Zum gegenwärtigen Stand wird aus der Literaturrecherche jedoch nicht ersichtlich, welche konkreten Eigenschaften das Problem der Klassifikation von Zeitreihen allgemein zufriedenstellend lösen. Es stellt sich eher die Frage, ob eine solche Menge charakteristischer Eigenschaften überhaupt existiert, oder ob weitere Faktoren zur Vorgruppierung bzw. Vorklassifizierung in Betracht gezogen werden müssen.

3 EIGENSCHAFTSBASIERTE REPRÄSENTATION VON ZEITREIHEN

In Kapitel 2 wurden die Vorteile der Verwendung von repräsentativen Eigenschaften einer Zeitreihe diskutiert. Die Möglichkeiten einer signifikanten Reduzierung der Problemdimensionierung und einer Eliminierung der stringenten Suchraumverkleinerung auf Grund von Längenabhängigkeit, lassen diesen Ansatz besonders attraktiv erscheinen. In diesem Kapitel werden zunächst einige numerische und statistische Eigenschaften von Zeitreihen aufgezeigt (Abschnitt 3.1). Anschließend erfolgt in Abschnitt 3.2 die Einführung einer Datenbasis aus Zeitreihen verschiedenster Herkunft. Mit Hilfe dieser Datensammlung erfolgt schließlich eine Analyse, welche der in 3.1 identifizierten Merkmale als aussagekräftiges Charakteristikum geeignet sind (Abschnitt 3.3).

3.1 NUMERISCHE UND STATISTISCHE EIGENSCHAFTEN VON ZEITREIHEN

Hinsichtlich der schlussendlichen Menge repräsentativer Eigenschaften bietet sich eine Vielzahl von Kandidaten an. Die Spannweite der Kennzahlen reicht dabei von einfachen numerischen Merkmalen bis hin zu komplexeren statistischen Werten. Im Folgenden werden einige Charakteristiken vorgestellt, die auf Grund ihrer Semantik für eine aussagekräftige Darstellung von Zeitreihen erfolgversprechend erscheinen. Hierzu wird folgende Definition einer endlichen Zeitreihe $T = t_1, \cdots, t_n$ aller Beobachtungen t_i der Beobachtungszeitpunkte i mit $1 \leq i \leq n$ als gegeben vorausgesetzt.

Numerische Kennzahlen
Jede Zeitreihe ist eine kontinuierliche Folge numerischer Werte. Aus diesem Grund können sofort elementare numerische Eigenschaften bestimmt werden, welche es ermöglichen, die Ausprägung

der Zeitreihe grob zu umschreiben. Sie liefern einerseits Auskunft über den vorliegenden Wertebereich und darüber hinaus andererseits auch Anhaltspunkte zur generellen Datenverteilung. Zu diesen Eigenschaften gehören:

- das Minimum $min(T)$
- das 25% Quartil (auch unteres Quartil bzw. lower Quartil) $lq(T)$
- der Median $\widetilde{T}$
- das 75% Quartil (auch oberes Quartil bzw. upper Quartil) $uq(T)$
- das Maximum $max(T)$
- das arithmetische Mittel[1] $\overline{T}$

Die beiden Darstellungen in Abbildung 3.1 verdeutlichen die Aussage der eben genannten Zeitreiheneigenschaften. Sie beinhalten nicht nur die Streudiagramme der Zeitreihen *Geburten*[2] und *Getreide*[3], sondern auch einen eingebetteten Boxplot der jeweiligen Datenverteilungen auf der linken Seite der Grafik. Während Abbildung 3.1a die tägliche Anzahl neugeborener Kinder des Jahres 1959 im US-Bundesstaat Kalifornien zeigt, ist in Abbildung 3.1b die monatliche Entwicklung des Getreide-Preisindexes in Deutschland in den Jahren 1995 bis 2010 dargestellt. Aus den Diagrammen ist unmittelbar zu erkennen, dass beide Zeitreihen in völlig verschiedenen „Datenregionen" verlaufen. Da dieser Fall illustriert, dass die gezeigten Eigenschaften Trennungscharakter aufweisen, werden sie in den folgenden Analysen betrachtet.

Als weitere triviale Eigenschaften können von jeder Zeitreihe die *Länge* $l(T)$ und die *Anzahl unterschiedlicher Werte* $dv(T)$ bestimmt werden. Dabei ist die Länge als $l(T) = |T|$ und die Anzahl unterschiedlicher Werte als $dv(T) = |\{v|v \in W(T) \wedge v \in T\}|$ definiert, wobei $W(T)$ den Wertebereich der Zeitreihe T darstellt. Durch die Aufnahme der Länge als beschreibendes Merkmal kann eine Art „Vorgruppierung" erreicht werden. Diese drückt sich dadurch aus, dass in der Ähnlichkeitsbetrachtung Zeitreihen von etwa gleicher Länge einen Vorzug erhalten. Die Idee der Betrachtung der Anzahl unterschiedlicher Werte folgt einem analogen Ansatz. Auch hier wird eine Vorgruppierung anhand der Ausprägung dieser Eigenschaft angenommen.

Statistische Kennzahlen

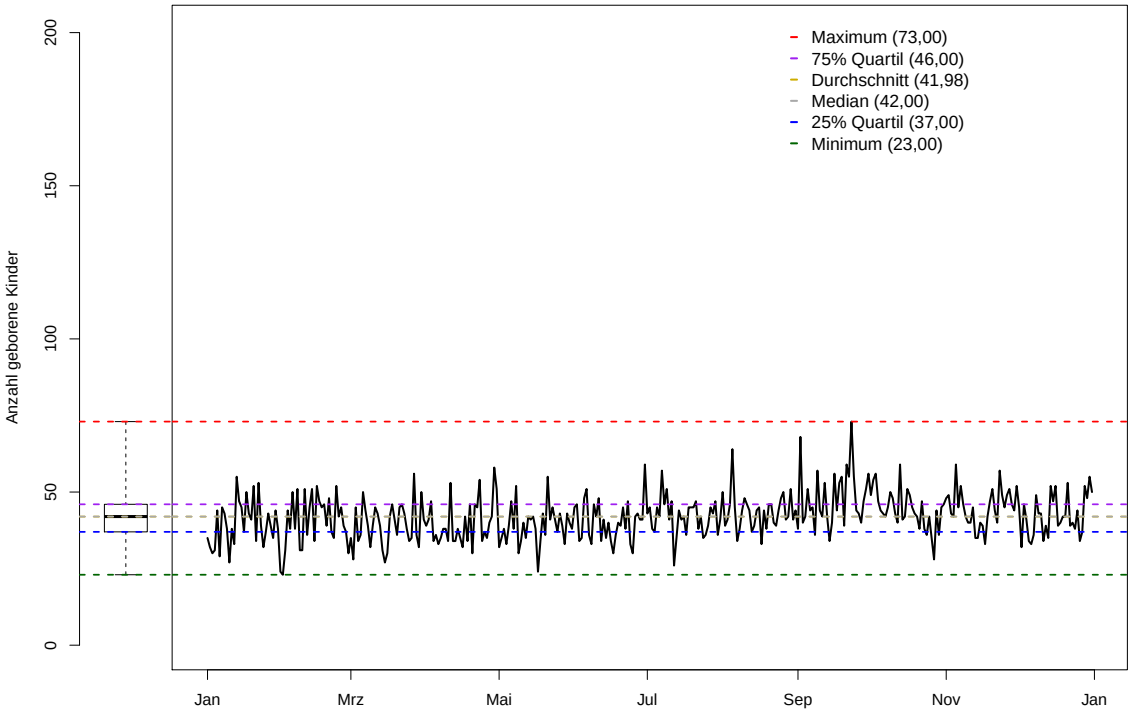
Zusätzlich zu diesen trivialen Eigenschaften können noch weitere statistische Werte auf einer Zeitreihe ermittelt und in Betracht gezogen werden. Im Folgenden werden die Standardabweichung, die empirische Varianz, die empirische Schiefe und die empirische Wölbung betrachtet. Sie sind in einer Vielzahl von Lehrbüchern zur Statistik aufgeführt und werden als grundlegende statistische Eigenschaften von Verteilungen genannt. Im Besonderen wird an dieser Stelle Bezug auf die Angaben in [HS16] genommen.

Die *Standardabweichung* $sd(T)$ gilt als Maß für die Streuung einer Datenverteilung. Aus diesem

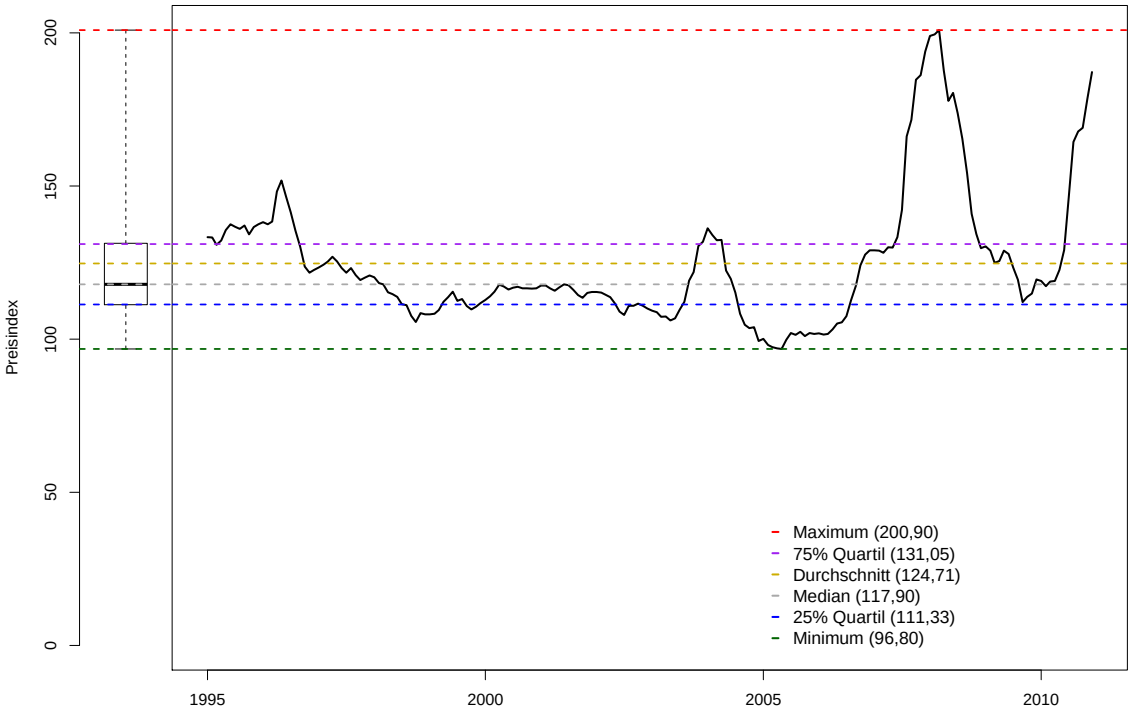
[1] auch empirisches zentrales Moment erster Ordnung, siehe [HS16]

[2] Quelle: `https://datamarket.com/data/set/235k/daily-total-female-births-in-california-1959#!ds=235k&display=line`

[3] Quelle: `http://www.faes.de/Basis/Basis-Statistik/Basis-Statistik-Zeitreihenanal/BS_Zeitreihenanalyse_R/bs_zeitreihenanalyse_r.html`



(a) Streudiagramm der täglichen Anzahl neugeborener Kinder des Jahres 1959 im US-Bundesstaat Kalifornien



(b) Streudiagramm der monatlichen Entwicklung des Getreide-Preisindexes in Deutschland in den Jahren 1995 bis 2010

Abbildung 3.1: Streudiagramme der Zeitreihen *Geburten* und *Getreide*

Grund erscheint sie als attraktiv, um als repräsentative Eigenschaft für eine Zeitreihe ausgewählt zu werden. Mittels der in [HS16] angegebenen Formel

$$sd(T) = \sqrt{\frac{\sum_{i=1}^{n} \left(t_i - \overline{T}\right)^2}{n-1}} \tag{3.1}$$

kann $sd(T)$ berechnet werden.

Durch das Kürzel $var(T)$ wird die *empirische Varianz* – auch zentrales Moment zweiter Ordnung – bezeichnet. Sie ist die Maßzahl für die Variabilität einer Datenverteilung, weswegen sie ebenfalls als interessantes beschreibendes Maß einer Zeitreihe angesehen werden kann. Die Berechnung von $var(T)$ ergibt sich als:

$$var(T) = \frac{\sum_{i<j} \left(t_i - t_j\right)^2}{n\,(n-1)} \tag{3.2}$$

Als zentrales Moment dritter Ordnung ist die *empirische Schiefe* (auch Skewness) $sk(T)$ anzusehen. Sie gibt Auskunft über das Maß an Asymmetrie der Verteilung der einzelnen Beobachtungswerte. Hierbei steht der Wert 0 für eine symmetrische Verteilung der Beobachtungen um ihren Erwartungswert (siehe [HS16]). Negative Werte der Schiefe signalisieren sogenannte „rechtssteile Datenverteilungen“, wobei in einer solchen Zeitreihe viele „große“ Werte und verhältnismäßig wenige „kleine“ Werte vorkommen. Der umgekehrte Fall, also positive Werte der Schiefe, ist ein Indikator für sogenannte „linkssteile Datenverteilungen“. Mit Hilfe der folgenden Formel ist die empirische Schiefe zu ermitteln:

$$sk(T) = \frac{\frac{1}{n}\sum_{i=1}^{n} \left(t_i - \overline{T}\right)^3}{sd(T)^3} \tag{3.3}$$

Die *empirische Wölbung* (auch Kurtosis) $k(T)$, als zentrales Moment vierter Ordnung, ist das Maß für die Steilheit oder auch Spitzigkeit einer Zeitreihe. Bei einem geringen Wert der Wölbung kann von einer recht gleichmäßigen Streuung der Beobachtungen ausgegangen werden. Hohe Werte lassen wiederum auf die Existenz einzelner Extreme in den Beobachtungen schließen. Die empirische Wölbung einer Zeitreihe wird nach [HS16] durch die Formel

$$k_H(T) = \frac{1}{n}\sum_{i=1}^{n} \left(\frac{t_i - \overline{T}}{sd(T)}\right)^4 - 3 \tag{3.4}$$

ermittelt. Zur Vermeidung negativer Kurtosis-Werte wird in dieser Arbeit eine leicht abgeänderte Version der vorangegangenen Formel benutzt, welche im *R*-Paket `fBasics` (siehe [TWSC14]) verwendet wird. Diese Version, welche auf der Definition der Kurtosis für Verteilungen basiert, ist als

$$k(T) = \frac{1}{n}\sum_{i=1}^{n} \left(\frac{t_i - \overline{T}}{sd(T)}\right)^4 \tag{3.5}$$

definiert.

Wie in Kapitel 2 bereits besprochen, werden in [KHSm16] sechs weitere statistische Eigenschaften von Zeitreihen vorgeschlagen. Auf Grund der Tatsache, dass die dort verwendeten Kennzahlen gute Leistungen im Bereich der eigenschaftsbasierten Visualisierung erbringen, scheinen sie ebenso geeignet zu sein, um eine längeninvariante Repräsentation einer Zeitreihe zu erzielen. Als erste Eigenschaft in [KHSm16] ist die *spektrale Entropie* $se(T)$ genannt. Sie ist durch die Formel

$$se(T) = \int_{-\pi}^{\pi} \hat{f}_T(\lambda) \log \hat{f}_T(\lambda) d\lambda \tag{3.6}$$

erklärt. Die spektrale Entropie wird als Maß für die Vorhersagbarkeit einer Zeitreihe angesehen und ist in verschiedenen Arbeiten ([GJB14], [Goe13], [MR02]) betrachtet worden. Ihre Ermittlung kann mit Hilfe der Funktion `spectral_entropy` des *R*-Pakets `ForeCA` (siehe [Goe16]) erfolgen, welche in den weiteren Untersuchungen dieser Arbeit zur Anwendung kommt.

Des Weiteren sind sowohl die *Trendstärke* $sot(T)$ als auch die *Saisonstärke* $sos(T)$ in [KHSm16] aufgeführt. Sie werden auf Basis der drei Komponenten berechnet, welche durch eine „Seasonal and Trend decomposition using Loess (STL)“ (siehe [CCMT90]) entstehen. Die eigentliche Zeitreihe zerfällt dabei in eine Saisonkomponente S_T, eine Trendkomponente T_T und eine Restkomponente R_T, welche additiv miteinander verknüpft werden. Der Trendstärke ist die Berechnungsvorschrift

$$sot(T) = 1 - \frac{var(R_T)}{var(T - S_T)} \tag{3.7}$$

zu Grunde gelegt, wobei $var(R_T)$ die Varianz der Restkomponente – und damit der saison- und trendbereinigten Zeitreihe – und $var(T - S_T)$ die Varianz der saisonbereinigten Zeitreihe darstellt. Durch Ersetzung der saisonbereinigten Zeitreihe in der Varianz im Nenner durch die trendbereinigte Zeitreihe, ergibt sich die Saisonstärke wie folgt:

$$sos(T) = 1 - \frac{var(R_T)}{var(T - T_T)} \tag{3.8}$$

Zur Erfassung der Länge periodischer Muster einer Zeitreihe schlagen die Autoren in [KHSm16] die Betrachtung der *Periodizität* (oder auch Frequenz) $f(T)$ vor. Diese Eigenschaft prägt sich grundlegend kategorial aus und ist als numerischer Wert hinsichtlich des Beobachtungszeitraums eines ganzen Jahres anzugeben. Somit ist ein einjähriges periodisches Muster mit dem Wert $f(T) = 1$ anzugeben. Quartalsmuster hingegen tragen den Wert $f(T) = 4$, während Monatsmuster durch den Wert $f(T) = 12$ identifiziert sind.

Die Verwendung des *Autokorrelationswerts erster Ordnung* wird ebenfalls als beschreibende Eigenschaft einer Zeitreihe eingeführt. Dieser Wert beschreibt den linearen Zusammenhang zwischen der Restkomponente R_T einer Zeitreihe T und ihrer zugehörigen verschobenen Restkomponente $R_T^{'}$ zum Lag 1 und berechnet sich als $foac(T) = Corr(R_T, R_T^{'})$. Da dieser Wert eine Aussage darüber trifft, inwieweit zukünftige Beobachtungen von vergangenen abhängen, zählt er als zusätzlicher Indikator für die Vorhersagbarkeit einer Zeitreihe.

Schließlich wird der *Parameterwert* $\lambda(T)$ *der Box-Cox-Transformation* nach [BC64] als letztes beschreibendes Merkmal einer Zeitreihe in [KHSm16] genannt. Diese statistische Eigenschaft gibt

Auskunft über das Ausmaß der Variationsänderung innerhalb einer Zeitreihe. Der Wert für $\lambda(T)$ kann mit Hilfe der *R*-Funktion `BoxCox.lambda` des Pakets `forecast` (siehe [HK08] und [Hyn16]) ermittelt werden.

Tabelle 3.1 zeigt eine zusammenfassende Aufstellung der Eigenschaften, welche in dieser Arbeit hinsichtlich ihrer Eignung zur längeninvarianten Repräsentation einer Zeitreihe untersucht werden. Zusätzlich zu den verwendeten Kürzeln und Namen sind die jeweiligen *R*-Funktionen in der Notation `Paketname::Funktionsname` aufgeführt. Es ist zu beachten, dass im Moment noch keine dedizierten Funktionen existieren, um sowohl die Trendstärke als auch die Saisonstärke berechnen zu lassen. Diese beiden Werte sind somit durch explizite Formulierung der Formeln 3.7 und 3.8 zu ermitteln.

Kürzel	**Eigenschaftsname**	**Verwendete *R*-Funktion**
$min(T)$	Minimum	`base::min`
$lq(T)$	25%-Quartil	`stats::quantile`
$\widetilde{T}$	Median	`stats::median`
$uq(T)$	75%-Quartil	`stats::quantile`
$max(T)$	Maximum	`base::max`
$\overline{T}$	arithmetisches Mittel	`base::mean`
$l(T)$	Länge	`base::length`
$dv(T)$	Anzahl unterschiedlicher Werte	`base::length` und `base::table`
$sd(T)$	Standardabweichung	`stats::sd`
$var(T)$	empirische Varianz	`stats::var`
$sk(T)$	empirische Schiefe	`timeDate::skewness`
$k(T)$	empirische Wölbung	`timeDate::kurtosis`
$se(T)$	spektrale Entropie	`ForeCA::spectral_entropy`
$sot(T)$	Trendstärke	-
$sos(T)$	Saisonstärke	-
$f(T)$	Periodizität	`timeSeries::frequency`
$foac(T)$	Autokorrelationswert erster Ordnung	`stats::acf`
$\lambda(T)$	Parameter λ der Box-Cox-Transformation	`forecast::BoxCox.lambda`

Tabelle 3.1: Aufstellung aller zu analysierenden Zeitreiheneigenschaften

3.2 AUFBAU UND ZUSAMMENSETZUNG EINER DATENBASIS FÜR ZEITREIHEN

Zur Analyse der in Abschnitt 3.1 aufgeführten Eigenschaften wird eine Datenbasis aus Zeitreihen angelegt. Um ein möglichst breites Datenspektrum abzudecken, sind verschiedene Datensätze unterschiedlicher Herkunft Bestandteil der Sammlung. Diese Sets werden im Folgenden beschrieben.

Beginnend im Jahr 1982 entstanden die sogenannten M-Competitions (siehe dazu [MAC+82], [MCH+93] und [MH00]). Sie gelten bis heute als international anerkannte Standardsammlung von Zeitreihen und dienen der Evaluation der Genauigkeit von Vorhersagemodellen. In ihrer jüngsten Form, der M3-Competition[4] aus dem Jahr 2000, besteht die zugehörige Datenbasis aus 3003 Datenreihen. Diese organisieren sich vorrangig in den Domänen *Micro*, *Industry*, *Macro*, *Finance* und *Demographic*. Es sind jedoch auch Zeitreihen enthalten, welche keiner genauen Domäne zugeordnet werden können und durch die Domänenbezeichnung „Other" zusammengefasst werden. Abbildung 3.2 zeigt die tatsächliche Aufteilung der Zeitreihen anhand der eben genannten Domänen.

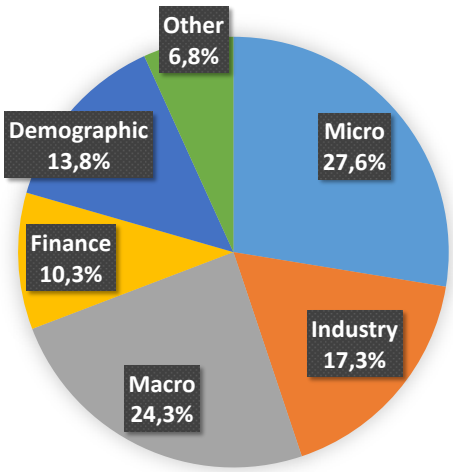


Abbildung 3.2: Domänenspezifische Einteilung der Zeitreihen der M3-Competition

Die Einteilung der M3-Zeitreihen erfolgt nicht nur domänenspezifisch, sondern zusätzlich auch auf Basis der zeitlichen Auflösung der jeweiligen Beobachtungen. Aus diesem Grund werden die Daten in jährliche, vierteljährliche und monatliche Beobachtungsintervalle unterteilt. Darüber hinaus gibt es auch Zeitreihen, deren Intervall sich in keine der drei vorangegangenen Kategorien einordnen lässt. Sie werden unter der Bezeichnung „andere" zusammengefasst. Die so entstandene zweistufige Klassifizierung der M3-Daten ist in Tabelle 3.2 illustriert.

In Anbetracht der jahrelangen Beständigkeit der Verwendung der eben vorgestellten Zeitreihen in der Forschung, kann der M3-Datensatz als obligatorischer Kern einer Datenbasis zum Zweck der empirischen Analyse angesehen werden. Die schlussendliche Datenmenge soll allerdings nicht nur auf diesen Standard reduziert bleiben. Da es das Ziel dieser Arbeit ist, ein Verfahren

[4]Quelle: `https://forecasters.org/resources/time-series-data/m3-competition`

Intervall	Domäne						Gesamt
	Micro	**Industry**	**Macro**	**Finance**	**Demographic**	**Other**	
jährlich	146	102	83	58	245	11	645
vierteljährlich	204	83	336	76	57	0	756
monatlich	474	334	312	145	111	52	1.428
andere	4	0	0	29	0	141	174
Gesamt	**828**	**519**	**731**	**308**	**413**	**204**	**3.003**

Tabelle 3.2: Domänenspezifische und intervallbasierte Klassifizierung der M3-Daten

zur methodischen Unterstützung der Modellauswahl für Prognosen aus dem Energieumfeld zu entwickeln, müssen die M3-Daten zunächst um Zeitreihen aus diesem Sektor erweitert werden. Hierzu finden Daten aus zwei verschiedenen Datensätzen Verwendung.

Der erste Datensatz stammt aus tatsächlichen Erhebungen der DREWAG - Stadtwerke Dresden GmbH und wurde von dieser freundlicherweise bereitgestellt. Er beinhaltet Messreihen zum Energieertrag verschiedener Energieerzeugungsanlagen im Raum Dresden des Jahres 2012. Die Messdaten bestehen aus insgesamt 30 Zeitreihen, welche in viertelstündlicher Auflösung vorliegen.

Als zweiter Bestandteil der neu hinzugekommenen Energiedomäne wird ein Datensatz des National Renewable Energy Laboratory (NREL), USA genutzt. Das NREL ist durch das U.S. Department of Energy subventioniert und wurde im Jahr 1974 gegründet. Forschungen auf den Gebieten der erneuerbaren Energien und der Energieeffizienz gehören zu den Kernkompetenzen dieser Einrichtung. Im Datenbestand des NREL befindet sich unter anderem das sogenannte Wind Integration Data Set (WIDS), welches simulierte Windenergiedaten beinhaltet. Ähnlich wie im Fall der Messdaten der DREWAG, ist auch in diesen Reihen der Energieertrag in Megawatt (MW) je Anlage aufgezeichnet. Anders als bei den DREWAG-Daten ist die zeitliche Auflösung in Form von 10-Minuten-Intervallen gegeben. Zusätzlich zum Energieertrag sind in diesen multivariaten Zeitreihen auch Angaben zur Windstärke enthalten. In dieser Arbeit wird der Fokus jedoch auf univariate Zeitreihen gelegt, weshalb die Windstärke keinerlei weitere Beachtung erfährt. Als Beobachtungszeitraum für den hier verwendeten Auszug von 186 Zeitreihen des WIDS ist das Kalenderjahr 2006 zu nennen.

Zur Simulation eines progressiven Systemverhaltens, bei dem die Datenbasis sukzessive durch reguläres Nutzungsverhalten vergrößert wird, werden abermals Zeitreihen zur bestehenden Datensammlung hinzugenommen. Die Datensätze *Browser Market Share*, *Yahoo*, *Currency*, *Getreide* und *EEX* bilden die neue Domäne Economics. In ihr sind unterschiedliche wirtschaftsbezogene Daten zusammengefasst. So umfasst der Satz *Browser Market Share*[5] Datenerhebungen über die Marktanteile verschiedener Versionen der Web-Browser Internet Explorer, Firefox, Chrome und Safari. Die Daten liegen in täglicher Auflösung vor und reichen vom 01.07.2008 bis zum 29.03.2012. In der Zeitreihe *Yahoo*[6] hingegen sind die täglichen Preise der Yahoo!-Inc.-Aktie in den Jahren von 1997 bis 2015 verzeichnet. Ein Auszug der Eurowechselkurse vom 04.01.1999 bis zum 23.06.2016 ist im Datensatz *Currency*[7] zu finden. Jede der 41 Zeitreihen beschreibt den täg-

[5]Quelle: `http://www.economicswebinstitute.org/data/browsermarket01.xls`
[6]Quelle: `http://finance.yahoo.com/quote/YHOO/history?ltr=1`
[7]Quelle: `https://www.ecb.europa.eu/stats/exchange/eurofxref/html/index.en.html`

lichen Verlauf des Wechselkurses der zugehörigen Fremdwährung in dieser Zeitspanne. *Getreide* bezieht sich auf die in Abbildung 3.1 verwendete Zeitreihe und repräsentiert die Entwicklung des Getreide-Preisindexes in Deutschland von 1995 bis 2010. Die zugehörigen Daten stammen vom Statistischen Bundesamt[8] und liegen in monatlichen Beobachtungsintervallen vor. Als letzter Datensatz der Domäne *Economics*, ist in *EEX* die Entwicklung des Energie Spotpreises für Deutschland zwischen 2012 und 2014 an der European Energy Exchange (EEX) Börse beschrieben. Diese Zeitreihe liegt in stündlicher Auflösung vor.

Abschließend werden der Datenbasis die Domäne *Statistics*, bestehend aus dem Datensatz *Births*, und die Domäne *Weather*, bestehend aus dem Datensatz *Weather*, hinzugefügt. Sie sollen dünn besetzte Domänen verkörpern, wie sie in realen Datenbasen durchaus vorkommen. Aus diesem Grund beinhalten beide Domänen jeweils nur zwei Zeitreihen. *Births* beinhaltet dabei unterschiedliche Geburtenraten, wohingegen *Weather* die Aufzeichnung natur- bzw. wetterbezogener Messungen aufzeigt. Alle Daten in diesen beiden Domänen wurden der Time Series Data Library (TSDL) des Internetportals DataMarket[9] entnommen. Ihre zugrunde liegende Auflösung ist durch tägliche Beobachtungen gegeben.

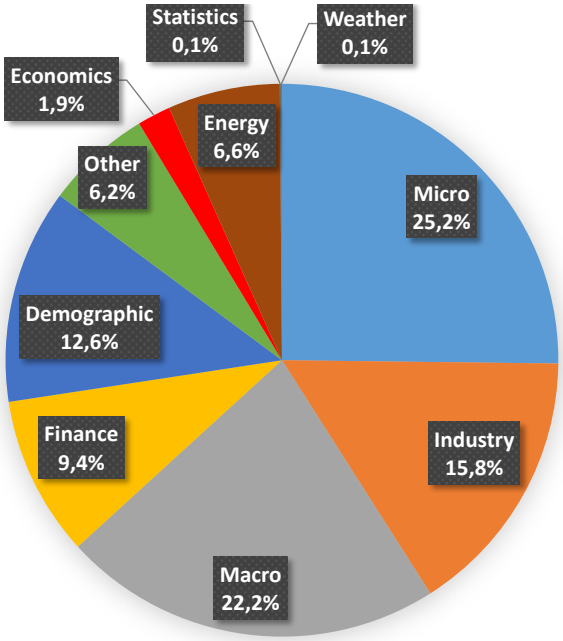


Abbildung 3.3: Domänenspezifische Einteilung der Zeitreihen der Datenbasis

Durch die vorangegangenen Erläuterungen ist der Referenzrahmen, gegeben durch den M3-Datensatz, zu einer Datenbasis aus insgesamt 3.287 Zeitreihen avanciert. Diese Grundgesamtheit spannt zehn Domänen auf, welche unterschiedlich stark vertreten sind. Alle enthaltenen Zeitreihen teilen sich des Weiteren in insgesamt sieben Beobachtungsintervalle auf.

Eine Übersicht der prozentualen Anteile jeder Domäne an der Datenbasis kann in Abbildung 3.3 gefunden werden. Für einen detailierteren Einblick in die datensatzbezogene Zusammensetzung der Datenbasis ist Tabelle 3.3a zu konsultieren. Tabelle 3.3b illustriert die auf das Beobachtungsintervall bezogene Zusammensetzung der Datenbasis.

[8] Quelle: `http://www.destatis.de/jetspeed/portal/cms/`
[9] Quelle: `https://datamarket.com/data/list/?q=provider:tsdl`

Datensatz	Domäne										Gesamt
	Micro	Industry	Macro	Finance	Demographic	Other	Economics	Energy	Statistics	Weather	
M3	828	519	731	308	413	204	0	0	0	0	3.003
WIDS	0	0	0	0	0	0	0	186	0	0	186
DREWAG	0	0	0	0	0	0	0	30	0	0	30
Browser Market Share	0	0	0	0	0	0	20	0	0	0	20
Currency	0	0	0	0	0	0	41	0	0	0	41
EEX	0	0	0	0	0	0	1	0	0	0	1
Getreide	0	0	0	0	0	0	1	0	0	0	1
Yahoo	0	0	0	0	0	0	1	0	0	0	1
Births	0	0	0	0	0	0	0	0	2	0	2
Weather	0	0	0	0	0	0	0	0	0	2	2
Gesamt	**828**	**519**	**731**	**308**	**413**	**204**	**64**	**216**	**2**	**2**	**3.287**

(a) Domänen- und datensatzspezifische Klassifizierung der Datenbasis

Intervall	Domäne										Gesamt
	Micro	Industry	Macro	Finance	Demographic	Other	Economics	Energy	Statistics	Weather	
jährlich	146	102	83	58	245	11	0	0	0	0	645
vierteljährlich	208	83	336	105	57	141	0	0	0	0	930
monatlich	474	334	312	145	111	52	0	0	0	0	1.428
täglich	0	0	0	0	0	0	63	0	2	2	67
stündlich	0	0	0	0	0	0	1	0	0	0	1
viertelstündlich	0	0	0	0	0	0	0	30	0	0	30
zehnminütig	0	0	0	0	0	0	0	186	0	0	186
Gesamt	**828**	**519**	**731**	**308**	**413**	**204**	**64**	**216**	**2**	**2**	**3.287**

(b) Klassifikation der Datenbasis hinsichtlich des Beobachtungsintervalls

Tabelle 3.3: Klassifikationen der Datenbasis

3.3 ANALYSE DER EIGENSCHAFTEN VON ZEITREIHEN

Im ersten Abschnitt dieses Kapitels wurden insgesamt 18 Eigenschaften konstatiert, welche zur längeninvarianten Darstellung von Zeitreihen verwendet werden können. Mit Hilfe der Datenbasis aus Abschnitt 3.2 soll nun eine Überprüfung erfolgen, ob diese Menge der finalen Eigenschaftsmenge P_{rel} entspricht. Würden unterschiedliche Eigenschaften einen linearen Zusammenhang aufweisen, wäre dem nicht so und die schlussendliche Menge könnte in ihrer Mächtigkeit reduziert werden. Da alle 18 Eigenschaften intervallskalierte Merkmale darstellen, ist der Korre-

lationskoeffizient r nach Pearson ([Pea95]) als Maß für den Grad des linearen Zusammenhangs anwendbar. Dies gilt auch für die Periodizität $f(T)$, obwohl sie in Abschnitt 3.1 als kategoriales Attribut charakterisiert wurde. Tatsächlich ist sie als „quasi-kategorial“ anzusehen, da sich ihr Wertebereich durchaus numerisch ausprägt.

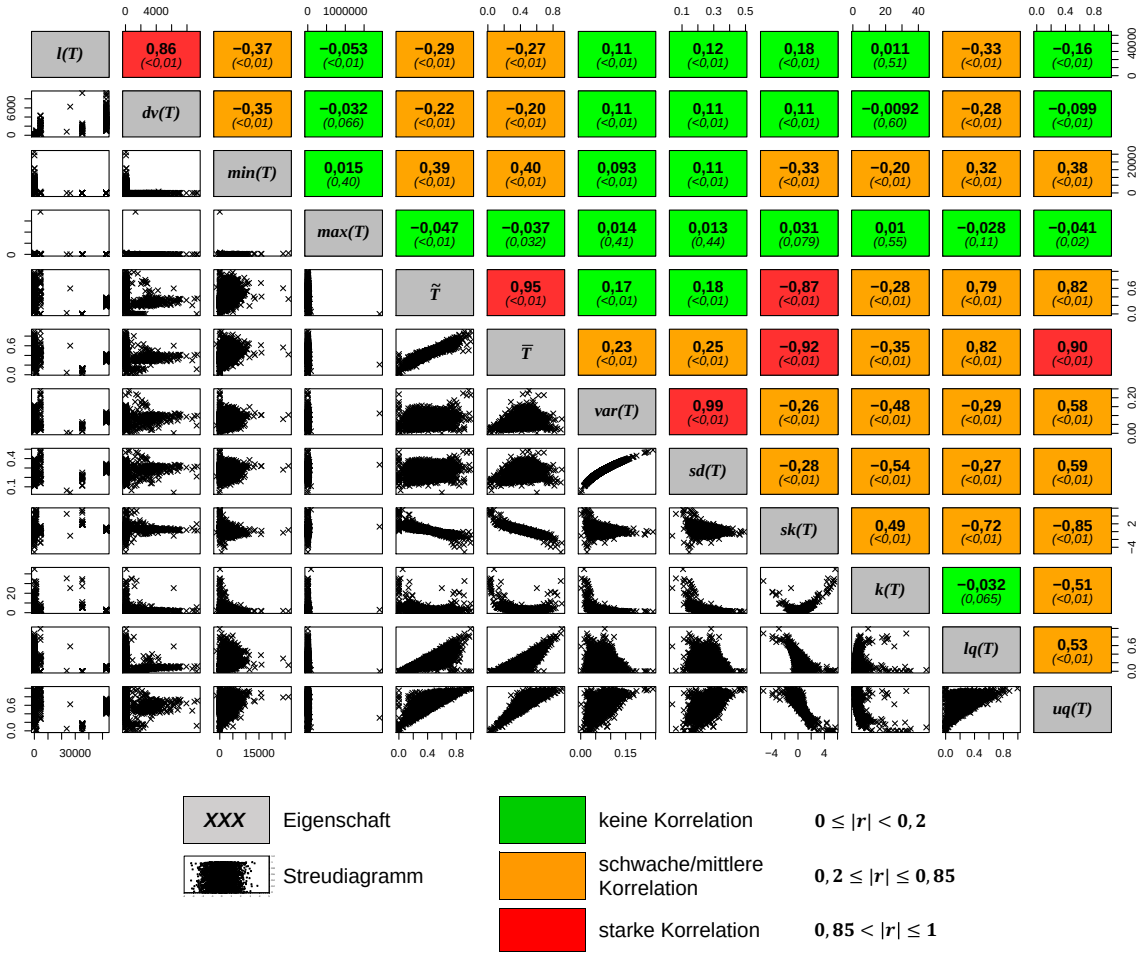


Abbildung 3.4: Korrelationsmatrix der ersten 12 Eigenschaften aus Tabelle 3.1 auf der Datenbasis

Je Streudiagramm ergibt sich die Skalierung von X und Y anhand des Wertebereichs der zugehörigen Eigenschaft auf der Hauptdiagonalen

Abbildung 3.4 zeigt das Ergebnis der Korrelationsanalyse der ersten 12 Eigenschaften aus Tabelle 3.1. Die Grafik besteht aus 144 Paneelen und ist in drei Bereiche unterteilt. Diese sind die untere Dreiecksmatrix dm_u, die Hauptdiagonale h und die obere Dreiecksmatrix dm_o. In den Paneelen der Hauptdiagonalen h sind die jeweiligen Eigenschaften aufgeführt, welche in der Analyse betrachtet wurden. Der Bereich dm_u beinhaltet die Streudiagramme jedes einzelnen Eigenschaftspaares aus h. Die zu dm_u an h gespiegelten Paneele in dm_o zeigen in der ersten Zeile fett gedruckt den jeweiligen Korrelationskoeffizienten r und in runden Klammern den korrespondierenden Signifikanzwert (engl. p-value).

Alle Paneele in dm_o weisen zusätzlich eine farbliche Kategorisierung auf. Diese Markierung spiegelt die Bewertung der „Stärke“ der Korrelation zweier Eigenschaften zueinander wider. In der

Literatur schwanken die Angaben zu den Grenzen zur starken Korrelation mitunter horrend. Frühe Arbeiten wie [Coh88] nennen einen Korrelationskoeffizienten von mehr als 0,5 schon einen hinreichenden Indikator auf einen sehr starken Zusammenhang. In späteren Veröffentlichungen desselben Autors ([CMM05]) wird diese Schwelle auf 0,85 nach oben korrigiert. Einige Online-Ressourcen[10] postulieren sogar einen Wert von 0,9.

Die Farbkodierung in Abbildung 3.4 basiert auf den folgenden, durch [CMM05] gestützten Intervallen: Korrelationswerte im Bereich $0 \leq |r| < 0,2$ signalisieren, dass keine Korrelation (grüne Färbung) vorliegt. Im Bereich $0,2 \leq |r| \leq 0,85$ herrscht eine schwache bis mittlere Korrelation (orange Färbung), während im Bereich $0,85 < |r| \leq 1$ eine starke Korrelation (rote Färbung) erkennbar ist.

Anhand der eingeführten Grenzen der Indikation starker Korrelation zeigt sich in der Korrelationsmatrix in Abbildung 3.4, dass einige Merkmalspaare unter hinreichendem Verdacht eines linearen Zusammenhangs stehen. So zum Beispiel das Paar $\left(\widetilde{T}, \overline{T}\right)$, welches einen Korrelationskoeffizienten von 0,95 aufweist. Ein solch hoher Korrelationswert lässt darauf schließen, dass zwischen diesen beiden Kennzahlen ein linearer Zusammenhang besteht. Auch das entsprechende Streudiagramm stützt diese Erkenntnis. Durch den aufgezeigten Zusammenhang ist es möglich, eines der beiden Merkmale aus der Finalmenge zu streichen. Auf Grund der Tatsache, dass der Median in seiner Funktion als Zentralwert robust gegenüber Extremwerten in einer Datenverteilung ist (siehe [HS16]), wird dieser anstelle des arithmetischen Mittels in P_{rel} aufgenommen.

Ein ähnliches Szenario findet sich bei zwei weiteren Paaren in der Matrix – $(l(T), dv(T))$ und $(var(T), sd(T))$. Für das letztgenannte Paar war dieses Ergebnis zu erwarten, da sich die Standardabweichung aus der Wurzel der Varianz berechnen lässt. Der Korrelationswert von 0,99 ist daher nicht überraschend. Da die Varianz – im Gegensatz zur Standardabweichung – eines der zentralen Momente einer Zufallsvariable darstellt, wird sie in die finale Eigenschaftsmenge aufgenommen.

Die eben geführte Argumentation müsste prinzipiell auch auf das Merkmalspaar $\left(\widetilde{T}, sk(T)\right)$ angewendet werden. Ein Korrelationswert von -0,87 spricht nach oben genannter Definition für einen Ausschluss einer der beiden Kennzahlen. Es werden im Folgenden dennoch sowohl die empirische Schiefe als auch der Median zu P_{rel} hinzugefügt. Dieser Entschluss ist jeweils mit der Relevanz der semantischen Aussage beider Kennzahlen und ihren Funktionen als zentrale Momente einer Datenverteilung zu begründen. Im Hinblick auf eine Grenze, welche einerseits nur knapp überschritten wurde und andererseits eine überaus unstete Fundierung aufweist, überwiegen diese beiden Aspekte und sprechen für eine Aufnahme beider Werte.

Wird die Korrelationsmatrix jeweils für die Eigenschaften $min(T)$ und $max(T)$ konsultiert, ist erkennbar, dass bei beiden Merkmalen keine nennenswerte Korrelation zu einer der verbleibenden Kennzahlen existiert. Aus Sicht der Korrelationsanalyse spricht demnach nichts gegen eine Aufnahme beider Merkmale in die Finalmenge. Eine genauere Betrachtung der Aussage beider Eigenschaften führt jedoch zu einer abweichenden Erkenntnis. Hierzu ist zu bemerken, dass sowohl das Minimum als auch das Maximum nicht notwendigerweise eine solide Auskunft über die Eingrenzung der Verlaufsregion einer Datenverteilung widerspiegeln. Ähnlich wie das arithmetische Mittel sind auch diese beiden Werte nicht robust gegenüber Extremwerten.

[10] Beispielsweise: `https://www.univie.ac.at/ksa/elearning/cp/quantitative/quantitative-104.html`

Das Minimum einer Messreihe im Energieumfeld wird beispielsweise sehr häufig den Wert 0 annehmen. Dies ist der allgemeinen Natur solcher Datenerhebungen geschuldet, da beispielsweise kontinuierliche Aufzeichnungen des Energieertrags von Solaranlagen eine Abhängigkeit zur Sonneneinstrahlung aufweisen. Ein Minimum von 0 ist technisch korrekt, jedoch repräsentiert es mitnichten eine realistische Aussage der minimalen Erzeugungsleistung einer solchen Anlage. Abgesehen vom technisch korrekten Auftreten des Wertes 0, existieren auch Fälle, in denen Aufzeichnungslücken in der Datenerhebung stellvertretend durch 0 ersetzt wurden. In diesem Fall stellt ein solcher Minimalwert ebenso keine repräsentative Aussage zur Eingrenzung der Verteilung dar.

Gegen die Aufnahme des Maximums als Charakteristikum ist analog zum Fall des Minimums zu argumentieren. Reale Messanlagen sind in ihrer Funktion als technische Systeme anfällig für Störeinflüsse (siehe [Czi15], Kapitel 2), welche eine Verfälschung der Messwerte zur Folge haben können. Die Existenz von Extremwerten als obere und untere Schranken einer Datenverteilung kann somit nicht ignoriert werden. Das bedeutet schlussendlich, dass gemessene Maximalwerte ebenfalls keine hinreichende Eignung zur charakteristischen Darstellung einer Zeitreihe besitzen. Aus den eben genannten Gründen werden weder das Minimum noch das Maximum in die Finalmenge aufgenommen.

Schließlich verbleibt in dieser Analyse die Überprüfung der Eigenschaften $l(T)$ und $dv(T)$ hinsichtlich ihrer Aussagekraft. Eine Begutachtung der Korrelation beider Kennzahlen zeigt, dass der ermittelte Wert von -0,86 für die Aufnahme nur eines der beiden Merkmale spricht. Tatsächlich erscheint jedoch die Aufnahme keines der beiden Werte als besonders geeignet. Hierzu ist zu betrachten, dass die reine Länge als Anzahl der Beobachtungen stark von der Auflösung der Zeitreihe abhängt.

Als Beispiel seien die Aufzeichnungen verschiedener fiktiver Temperaturmessstationen genannt. Eine Station A sei in diesem Szenario so konfiguriert, dass sie jeden Tag ein einziges Mal zu einer definierten Zeit die aktuelle Umgebungstemperatur messen soll. Station B hingegen besitzt eine Konfiguration derart, dass sie stündlich die Umgebungstemperatur misst. Sammeln beide Stationen für 365 Tage unterbrechungsfrei Messdaten, so ergeben sich die Längen $l(A) = 365$ und $l(B) = 365 \cdot 24 = 8.760$. Ein rein numerischer Vergleich dieser beiden Werte würde hier keine erkennbare Ähnlichkeit signalisieren, da sich beide Längen um den Faktor 24 unterscheiden. Tatsächlich könnte durch Aggregation der Messreihe von Station B eine Reihe B' konstruiert werden, welche die grundlegende Verlaufscharakteristik von B erhält, jedoch dieselbe Länge wie A aufweist. Die jeweiligen Beobachtungen B'_i, mit $1 \leq i \leq (|B| - 1)$ div k sind definiert als

$$B'_i = \frac{1}{k} \cdot \sum_{j=i*k}^{(i*(k+1))-1} B_j$$

mit $k = 24$. Eine Verwendung der Länge wäre in diesem Fall somit irreführend gewesen und hätte einen potentiellen Vergleichspartner ausgeschlossen. Da sich für die Anzahl unterschiedlicher Werte $dv(T)$ analoge Überlegungen ergeben, werden sowohl $l(T)$ als auch $dv(T)$ aus P_{rel} ausgeschlossen.

Durch die vorangegangene Analyse wurde die finale Menge repräsentativer Zeitreiheneigenschaften P_{rel} um sechs Merkmale reduziert. Eine weitere Korrelationsanalyse mit den letzten

sechs Eigenschaften aus Tabelle 3.1 zeigt, dass, mit Ausnahme der geduldeten starken Korrelation im Paar $\left(\widetilde{T}, sk(T)\right)$, keine weiteren starken Korrelationen zweifelsfrei nachgewiesen werden können. Abbildung 3.5 illustriert diese Erkenntnis. Die semantische Aussage eines jeden Merkmals aus [KHSm16] ist ebenfalls solide und schlüssig.

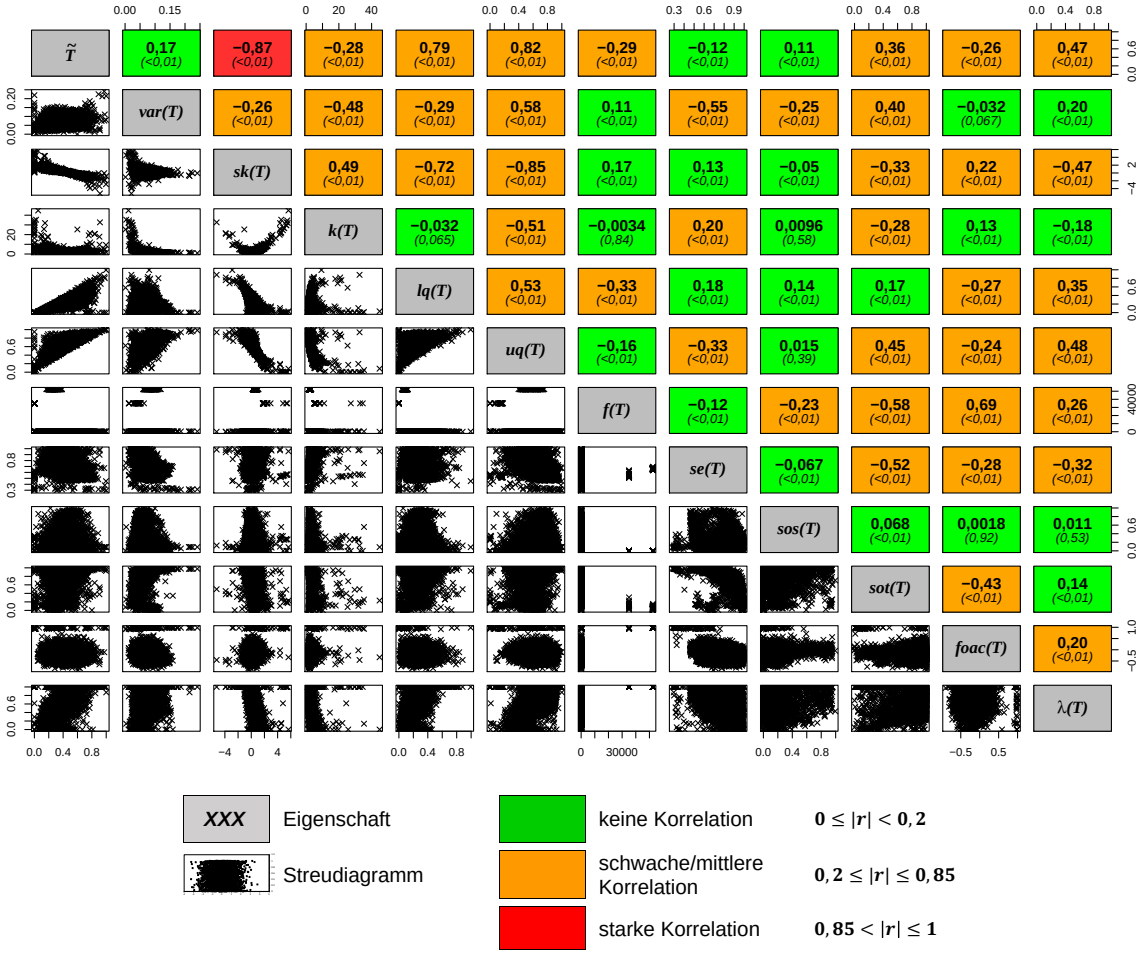


Abbildung 3.5: Korrelationsmatrix der finalen Eigenschaftsmenge P_{rel} auf der Datenbasis

Je Streudiagramm ergibt sich die Skalierung von X und Y anhand des Wertebereichs der zugehörigen Eigenschaft auf der Hauptdiagonalen

Auf Grund dieser Aspekte werden alle sechs verbleibenden Merkmale zur finalen Menge repräsentativer Eigenschaften von Zeitreihen hinzugefügt. Unter Verwendung dieser Menge ist es möglich, jede Zeitreihe T als Vektor eines 12-dimensionalen Eigenschaftsraumes zu betrachten. Die grundlegende Form einer solchen Darstellung $P_{rel}(T)$ wird in Formel 3.9 demonstriert.

$$P_{rel}(T) = \left(lq(T), uq(T), \widetilde{T}, var(T), sk(T), k(T), se(T), sot(T), sos(T), f(T), foac(T), \lambda(T)\right) \quad (3.9)$$

Mit Formel 3.9 ist eine längeninvariante, eigenschaftsbasierte Darstellung einer Zeitreihe T gefunden. Im Folgenden wird sie als Grundlage zur systematischen Modellempfehlung für neue, unbekannte Zeitreihen verwendet.

4 METHODISCHE UNTERSTÜTZUNG DER MODELLAUSWAHL FÜR PROGNOSEPROZESSE

Im vorangegangenen Kapitel wurde sukzessive eine Menge numerischer Kennzahlen ermittelt, welche eine hinreichend erfolgversprechende Eignung zur längeninvarianten, eigenschaftsbasierten Darstellung von Zeitreihen auszeichnet. Gegenstand dieses Kapitels ist es, diese Darstellung als Grundlage einer systematischen Unterstützung des Auswahlprozesses für Zeitreihenvorhersagemodelle zu nutzen. Da die Erkenntnisse der methodischen Unterstützung der Modellauswahl später als neue Komponente in das Benchmark Framework ECAST einfließen sollen, wird in Abschnitt 4.1 zunächst eine Vorstellung dieses Frameworks vorgenommen. Im Anschluss daran erfolgt eine Erläuterung der Klassifikation von Zeitreihen anhand von Prognosemodellen (Abschnitt 4.2). Schließlich zeigt Abschnitt 4.3 die grundlegende Struktur und Funktionsweise der entwickelten Klassifikationskomponente SEMPER, welche alle vorangegangenen Erkenntnisse bündelt, um Modellempfehlungen für Zeitreihen zu ermitteln.

4.1 DAS BENCHMARK FRAMEWORK ECAST

Das Energy Forecasting Benchmark Framework (ECAST) ([UFK+14], [UHH+16]) wurde als Software-Werkzeug zum systematischen Benchmarking von Energieversorgungsvorhersagen entwickelt. Es ermöglicht Benutzern die Planung und Ausführung von Vorhersageexperimenten unter Verwendung verschiedenster Informationen. Diese reichen von der Zeitreihe selbst bis hin zu beeinflussenden Kontextinformationen. Sowohl die Eingabedaten zur Ausführung der Experimente als auch deren jeweilige Resultate werden in der darunterliegenden Datenbank persistiert. Abbildung 4.1, welche [UHH+16] entnommen ist, zeigt die grundlegende Architektur des Sys-

tems ECAST. Es besteht aus insgesamt vier Komponenten – dem Web Interface (WI), der Core Logic Component (CLC), dem Use Case Repository (UCR) und der Prediction API (PAPI).

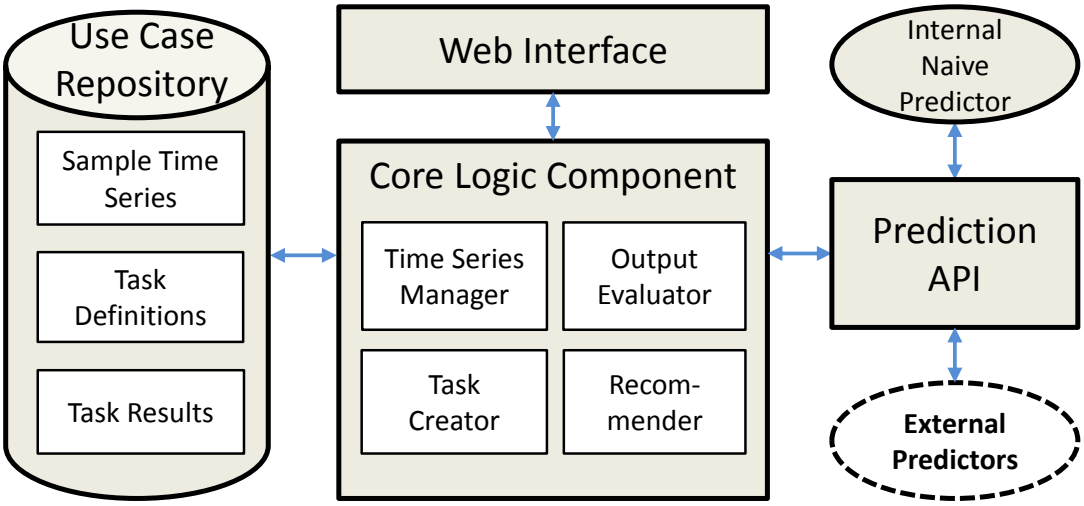


Abbildung 4.1: Grundlegende Architektur des Systems ECAST (Quelle: [UHH+16])

Das UCR bildet die Persistenzschicht in Form einer relationalen PostgreSQL-Datenbank. In ihr werden sämtliche Daten, von den Zeitreihenbeobachtungen selbst über zugehörige Kontextparameter bis hin zu den experimentell vorhergesagten Werten, gespeichert. Zugehörige Werte ausgewählter Fehlermaße, wie etwa dem Symmetric Mean Average Percentage Error (SMAPE) oder dem Root Mean Square Error (RMSE), sowie die einzelnen Ausführungszeiten der Experimente sind ebenfalls im UCR zu finden.

Die CLC ist die funktionale Kernkomponente des Gesamtsystems. Sie ist ihrerseits in vier weitere logische Einheiten unterteilt – den Time Series Manager (TSM), den Recommender, den Task Creator (TC) und den Output Evaluator (OE). Ein typischer Durchlaufzyklus beginnt in der CLC im TSM, welcher die Daten in ein internes Format zur effizienten Speicherung und Verarbeitung umwandelt. Anschließend durchsucht der Recommender das UCR und vergleicht die neue Zeitreihe mit allen vorhandenen Anwendungsfällen. Hierbei werden sowohl die strukturelle Ähnlichkeit der Zeitreihen als auch die angegebenen Kontextinformationen verwendet. Als Ergebnis der Suche des Recommenders werden dem Nutzer die Parameter des ähnlichsten Experiments aus dem UCR als empfohlene Standardeinstellungen im WI präsentiert. Diese fließen in den TC zur Erzeugung einzelner Forecasting-Tasks ein. Nachdem die Task von der PAPI bearbeitet wurde, berechnet der OE verschiedene statistische Fehlermetriken (SMAPE, RMSE etc.) zur Evaluation der Vorhersagegenauigkeit.

Die PAPI dient als Verbindungsbrücke zu den einzelnen Implementierungen der Vorhersagemodelle. In ECAST ist es möglich, sowohl eigenständige kommerzielle wie wissenschaftliche Vorhersagewerkzeuge als auch benutzerspezifische, in der Statistiksprache *R* implementierte Algorithmen anzubinden. Durch die PAPI werden alle notwendigen Eingabedaten an den Vorhersagealgorithmus übergeben. Dieser führt seine Berechnungen aus und leitet seine Ergebnisse zur PAPI zurück.

4.2 ZEITREIHENKLASSIFIKATION ANHAND VON VORHERSAGEMETHODIKEN

Im Laufe der Jahre entwickelte sich eine Vielzahl an Modellen zur Vorhersage von Zeitreihen (vgl. [KEDH01], [Hyn09]). Auf Grund unterschiedlicher Funktionsweisen der Modelle ergeben sich auf ein und derselben Messreihe mitunter signifikante Unterschiede in der Qualität der prognostizierten Werte. Diese Qualität zeichnet sich im Allgemeinen durch den berechneten Forecasting-Fehler $E_T = Y_T - F_T$ aus, wobei Y_T die tatsächlichen Beobachtungen und F_T die Vorhersagewerte einer Zeitreihe T bezeichnen.

Wird der Fehler mehrerer Modelle auf ein und derselben Zeitreihe erfasst und verglichen, ist es unmittelbar möglich, ein oder mehrere Modelle zu identifizieren, welche den ermittelten Fehler minimieren. Durch wiederholte Anwendung dieser Prozedur auf eine Menge von Zeitreihen ergibt sich ein Szenario, bei dem alle betrachteten Reihen mindestens einem Vorhersagemodell mit minimalem Fehler zugeordnet werden können.

Wie in Abschnitt 4.1 erwähnt, ist ECAST aktuell so implementiert, dass die Ergebnisse einzelner Forecasting-Experimente (also E_T, Y_T und F_T) in der eingebundenen Datenbank abgelegt werden. Die jeweiligen Daten sind, wie Abbildung 4.1 zeigt, im UCR zu finden (siehe *Sample Time Series*, *Task Definitions* und *Task Results*). Dieses „Protokoll“ aller Resultate aller Experimente auf den eingespielten Daten kann formal als eine Abbildung $experiment\colon TS \times M \to E$ angesehen werden. Hierbei bezeichnet TS eine Menge betrachteter Zeitreihen, M eine Menge zur Verfügung stehender Modelle und E den Wertebereich des verwendeten Fehlermaßes.

Genau dann, wenn sowohl die Mengen TS und M als auch die Abbildung $experiment$ existieren, gibt es eine totale Abbildung $label\colon TS \to \mathcal{P}^+(M)$, wobei $\mathcal{P}^+(M) = \mathcal{P}(M) \setminus \emptyset$ und $\mathcal{P}(M)$ die Potenzmenge der Menge M bezeichnet, sodass gilt: $\forall ts \in TS\, \exists m \in M\colon m \in label(ts)$. Die Klassifikationslabels von ts sind durch

$$label(ts) = \{m_i \mid experiment(ts, m_i) = min\,(\{experiment(ts, m_k) \mid m_k \in M\})\}$$

zu ermitteln. Fortan bezeichnet m ein *Klassifikationslabel* oder eine *Klasse* von ts.

Folgendes Beispiel soll die eben eingeführten Definitionen illustrieren. Es sei $TS = \{ts_1, ts_2, ts_3\}$ die Menge von Zeitreihen und $M = \{m_1, m_2\}$ die Menge zur Verfügung stehender Vorhersagemodelle. In diesem speziellen Szenario besitzt die Funktion $experiment$ ohne Beschränkung der Allgemeinheit folgende Ausprägung:

$$\begin{array}{lll} experiment(ts_1, m_1) = 25 & experiment(ts_2, m_1) = 66 & experiment(ts_3, m_1) = 13 \\ experiment(ts_1, m_2) = 22 & experiment(ts_2, m_2) = 105 & experiment(ts_3, m_2) = 13 \end{array}$$

Ausgehend von jeder Zeitreihe in TS gibt die Funktion $label$ die Menge aller Modelle aus M an, welche für die jeweilige Zeitreihe den niedrigsten Fehlerwert erzeugen. In diesem Beispiel hätte $label$ somit folgende Ausprägung: $label(ts_1) = \{m_2\}$, $label(ts_2) = \{m_1\}$, $label(ts_3) = \{m_1, m_2\}$. Es ist zu beachten, dass $label(ts_3)$ hier eine zweielementige Menge zurück gibt, da sowohl das Modell m_1 als auch das Modell m_2 für ts_3 denselben Fehlerwert erzeugen.

Ausgehend von jedem Modell in M lassen sich Modellklassen c_{m_l} definieren, welche die Gesamtheit aller Zeitreihen beschreiben, deren minimaler Fehlerwert durch m_l erzeugt wird. Diese Modellklassen sind als $c_{m_l} = \bigcup ts \mid m_l \in label(ts)$ definiert, wobei die einzelnen c_{m_l} jedoch nicht notwendigerweise disjunkt sind. Dies zeigt sich ebenfalls am gegebenen Beispiel, da sich hier die Modellklassen $c_{m_1} = \{ts_2, ts_3\}$ und $c_{m_2} = \{ts_1, ts_3\}$ ergeben. Auf Grund der Tatsache, dass nach diesem Prinzip jede Zeitreihe mindestens einer Modellklasse zugeordnet werden kann, gilt $TS = \bigcup c_{m_l} \mid 1 \leq l \leq |M|$. Aus dem Beispiel zeigt sich, dass über die Abbildung *label* Zeitreihen erfolgreich ihren Klassifikationslabels zugeführt werden. Somit ist die Fallbasis zufriedenstellend durch sie realisiert.

Mit Hilfe der eben eingeführten Fallbasis soll zukünftig die Unterstützung der Modellauswahl für neue Zeitreihen in ECAST gestützt werden. Die zentrale Annahme hierbei ist, dass die Eigenschaften aus P_{rel} (vgl. Kapitel 3) durch die Verteilung ihrer Ausprägungen die Zuordnung der einzelnen Zeitreihen zu ihren korrespondierenden Klassenlabels widerspiegeln. Unter dieser Voraussetzung ist es möglich, einen oder mehrere Klassifikatoren auf der Fallbasis zu bestimmen, unter deren Verwendung neue Zeitreihen ihren mutmaßlichen Labels zugeordnet werden können.

Es verbleibt zum gegenwärtigen Zeitpunkt die Frage nach der konkreten Art des zu verwendenden Klassifikators. Unter der Voraussetzung, dass P_{rel} die Fallbasis in hinreichend kompakte Modellcluster separiert, wäre eine triviale Klassifikation nach dem 1-Nearest-Neighbor-Prinzip (siehe [LLK85], [Ert13], [Run12]) gangbar und ausreichend. Wenn dies nicht der Fall ist, kann es notwendig werden, andere Verfahren zur Klassifikation in Betracht zu ziehen, um ausreichend gute Ergebnisse in der Modellvorhersage zu erzeugen. Nichtsdestotrotz besteht die Kernkompetenz der methodischen Unterstützung der Modellauswahl für Prognoseprozesse in der korrekten Modellklassifikation neuer Zeitreihen, unter Zuhilfenahme einer zentralen Fallbasis.

4.3 ARCHITEKTURERWEITERUNG DES SYSTEMS ECAST

Die vorangegangenen Kapitel und Abschnitte führten zu dem Schluss, dass eine Unterstützung der Modellauswahl für Prognoseprozesse durch eine hinreichend stabile Modellklassifikation unter Verwendung einer eigenschaftsbasierten Repäsentation erreichbar ist. Das System ECAST bietet in seiner aktuellen Realisierung eine gute Grundlage, um diesen Klassifikationsmechanismus zu integrieren. Im Folgenden wird die Konzeption dieser Systemerweiterung vorgestellt. Die tatsächliche Implementierung der Erweiterung in ECAST ist jedoch nicht Teil dieser Arbeit.

Auf Grund ihrer Funktionsweise kann die zukünftige Empfehlungskomponente in ECAST als systematisches Modul zur Modellklassifikation angesehen werden. Es arbeitet mit eigenschaftsbasierten Zeitreihendarstellungen und erzeugt empirisch ermittelte Empfehlungen. Infolgedessen erhält diese Komponente den Namen „Systematic Engine for Model classification on Property based Empirical Recommendations (SEMPER)“.

Abbildung 4.2 zeigt die grundlegende Architektur von SEMPER als logische Komponente. Sie demonstriert einen exemplarischen Durchlauf, bei dem die Zeitreihe *Query* in das System eingebracht wurde. Zunächst erfolgt die Transformation der Zeitreihe *Query* in ihre korrespondierende

vektorisierte Darstellung V_Q durch Berechnung der Eigenschaftswerte gemäß P_{rel}. Anschließend fließt V_Q in den Klassifikationsmechnismus ein. Dieser konsultiert, je nach konkreter Funktionsweise, die zentrale Fallbasis und ermittelt mit deren Hilfe das zu erwartende Klassifikationslabel. Besagtes Label bildet die Ausgabe von SEMPER, welche zur aufrufenden Instanz zurückgegeben wird. Auf das Beispiel in Abbildung 4.2 bezogen, liefert die Komponente *Modell B* als Empfehlung zurück.

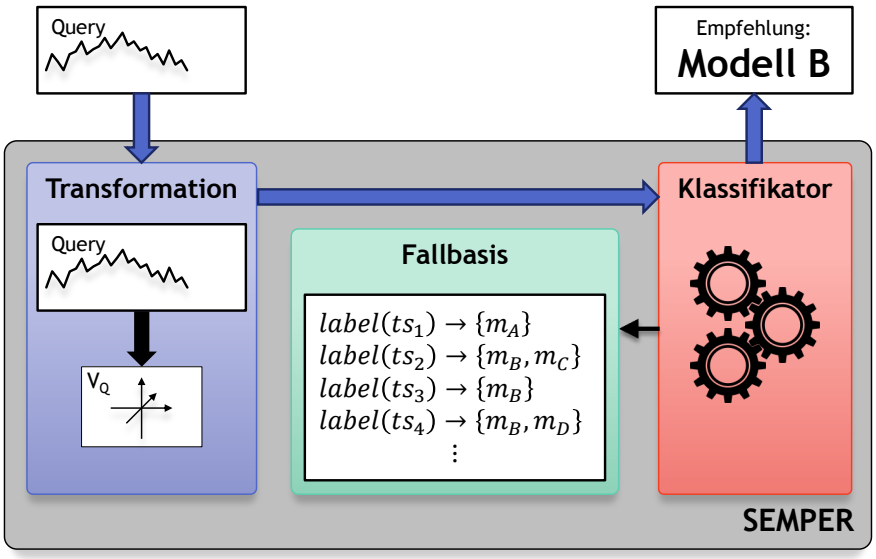


Abbildung 4.2: Logische Architektur der Empfehlungskomponente SEMPER

In ECAST wird SEMPER jedoch nicht als zusammenhängende Komponente erkennbar sein. Die Elemente der Empfehlungskomponente verteilen sich auf die Strukturbausteine in ECAST und erweitern diese. Abbildung 4.3 zeigt in Anlehnung an Abbildung 4.1 eine schematische Darstellung der theoretischen Integration von SEMPER in ECAST. Farblich zu Abbildung 4.2 passende Einfärbungen markieren die Verteilung der Funktionalität der Klassifikationskomponente im Grundsystem und indizieren somit die zu verändernden bzw. zu erweiternden Teile.

Hinsichtlich der besagten Systemerweiterung ist die Fallbasis (engl. case base) als Teil des UCR anzusehen. In diesem befinden sich nach gegenwärtigem Entwurf schon alle notwendigen Daten in einer relationalen Datenbank. Da die Fallbasis lediglich eine andere Sicht auf die vorhandenen Daten im UCR darstellt, könnte sie mittels einer View im verwendeten PostgreSQL realisiert werden.

Die Überführung der Zeitreihe in eine vektorisierte Darstellung ist in den TSM zu implementieren. Auf Grund der Tatsache, dass diese Komponente der CLC bereits die Umwandlung der übermittelten Werte in ein internes Format vornimmt, erschließt sie sich als logischer Integrationspunkt der Transformationsfunktionalität im Systemzyklus. Alle berechneten Eigenschaftswerte werden, gemäß dem ursprünglichen Entwurf von ECAST, ins UCR persistiert.

Schließlich hat die Eingliederung des Klassifikationsmechanismus im Recommender zu erfolgen. Analog zu den Überlegungen im Fall des TSM, bildet diese Komponente die passende Stelle zur Aufnahme der Modellklassifikation. Nachdem das empfohlene Modell zur eingegebenen

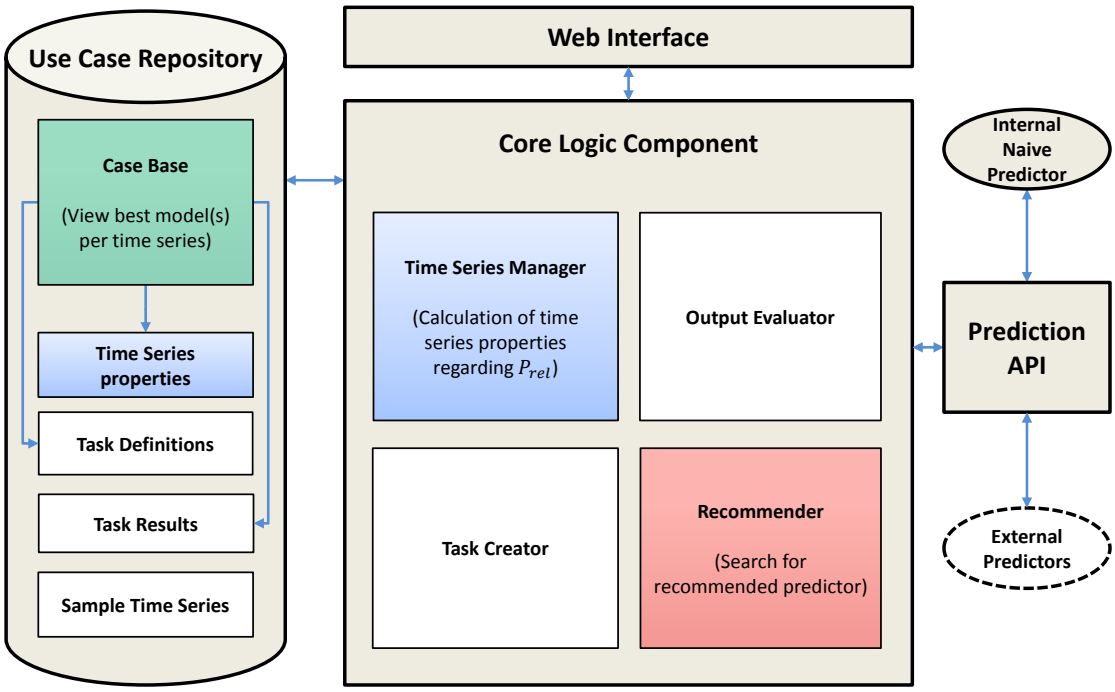


Abbildung 4.3: Integration der Komponente SEMPER in das System ECAST

Zeitreihe ermittelt wurde, kann diese Information in die Standardeinstellungen für ein neues Forecasting-Experiment einfließen.

Die Funktionsweise von SEMPER fügt sich nahtlos in den bisherigen Kontroll- und Datenfluss von ECAST ein. Das ermittelte Verfahren ist einfach in seiner Durchführung und orientiert sich stark am Vorgehen des CBR. Darüber hinaus kann das algorithmische Vorgehen von SEMPER als vollständig charakterisiert werden. Dies ist dadurch zu begründen, dass zu jeder neuen Zeitreihe eine Modellempfehlung auf Grundlage des Ähnlichkeitsprinzips ermittelbar ist, solange eine Möglichkeit besteht, die Eigenschaftsmenge P_{rel} auf ihr zu bestimmen.

5 EVALUATION DES 1-NEAREST-NEIGHBOR KLASSIFIKATORS

In den vorangegangenen Kapiteln wurde zunächst eine längeninvariante, eigenschaftsbasierte Darstellung von Zeitreihen ermittelt (Kapitel 3), welche anschließend in den Entwurf einer Komponente zur Unterstützung der Modellauswahl für Prognoseprozesse eingeflossen ist (Kapitel 4). Ziel dieses Kapitels ist es, einen geeigneten Klassifikator zu finden, welcher eine solide Zuordnung von Zeitreihen zu ihren optimalen Vorhersagemodellen ermöglicht. Hierfür erläutert Abschnitt 5.1 zunächst den Aufbau einer Reihe von Experimenten, welche dazu dienen, die Zuordnungsleistung des 1-Nearest-Neighbor (1NN) Klassifikators zu evaluieren. Die Evaluation des 1NN-Ansatzes erfolgt direkt im Anschluss (Abschnitt 5.2).

5.1 METHODOLOGIE

Zur Evaluation der in Kapitel 4 erarbeiteten Empfehlungskomponente wird eine Reihe von Experimenten genutzt, welche im Folgenden erläutert sind. Alle Experimente werden auf einem Laptop unter Windows 8.1 Professional x64 unter Verwendung einer Intel(R) Core(TM) i7-3630QM CPU @ 2.40GHz und 8GB RAM durchgeführt und finden außerhalb der ECAST-Infrastruktur statt. Daher ist zunächst der Entwurf und die Realisierung einer Datenbank notwendig, um alle Daten bereit zu stellen (Abschnitt 5.1.1). Anschließend ist zu erläutern, welche Vorhersagemodelle in dieser Evaluation zur Anwendung kommen (Abschnitt 5.1.2). Die dabei zusammengestellte Modellmenge stellt keinerlei Anspruch auf Vollständigkeit, sondern soll einen Auszug aus gebräuchlichen Algorithmen zur Zeitreihenvorhersage widerspiegeln. Schließlich zeigt Abschnitt 5.1.3 den Aufbau der Experimentenserie zur weiteren Evaluation.

5.1.1 Entwurf und Realisierung einer Evaluationsdatenbank

Zunächst müssen alle 3.287 Zeitreihen aus 3.2 und ihre berechneten Eigenschaften aus P_{rel} (vgl. Abschnitt 3.3) in eine Datenbank gespeichert werden. Hierzu ist ein geeignetes Datenbank Management System (DBMS) zu verwenden. In dieser Arbeit wird MySQL in der Version 5.7 als solches verwendet. Jede einzelne Zeitreihe stellt eine Tabelle in der Datenbank dar (*TS1*, *TS2*, *TS...*). Neben einer künstlich vergebenen ID besteht das Tabellenschema aus dem Zeitstempel der entsprechenden Beobachtungen und dem zugehörigen Beobachtungswert. Alle Reihen werden unverändert in ihren korrespondierenden Tabellen abgespeichert. Die Berechnung der Eigenschaftswerte erfolgt jedoch auf den durch $T_i^{norm} = (T_i - min(T)) \,/\, (max(T) - min(T))$ normalisierten Werten. Diese Methodik wurde gewählt, um einer Verzerrung der Werte der Y-Achse unterschiedlicher Zeitreihen zueinander entgegenzuwirken.

Die Repräsentation der berechneten Zeitreiheneigenschaften ist in einer separaten Tabelle (*P_Rel*) vorgesehen. Ihr Schema besteht aus einer ID-Spalte *SeriesID*, welche den Tabellennamen jeder einzelnen Zeitreihe beinhaltet. In der Systemtabelle *INFORMATION_SCHEMA.Tables* bekommt die Spalte *TABLE_NAME* damit partiell eine logische Fremdschlüsselbeziehung zur Eigenschaftstabelle. Des Weiteren ist für jede der Eigenschaften aus P_{rel} eine separate Spalte zur Speicherung der korrespondierenden Werte vorgesehen. Alle Spaltennamen entsprechen dabei den in Formel 3.9 verwendeten Abkürzungen.

Jede der Zeitreihen soll mit insgesamt acht Modellen vorhergesagt werden. Der ermittelte Forecasting-Fehler jeder dieser Vorhersagen ist ebenfalls in einer speziellen Tabelle (*Error*) in der Datenbank enthalten. Diese ist durch einen Verbundschlüssel aus der Zeitreihen-ID und dem verwendeten Vorhersagemodell bestimmt. Als letztes Attribut in dieser Tabelle schließt sich eine Spalte zur Speicherung des tatsächlichen Zahlenwertes des berechneten Forecasting-Fehlers an.

Schließlich wird die zentrale Fallbasis als Sicht in der Datenbank (*Casebase*) realisiert. Diese Sicht basiert auf der Tabelle der berechneten Forecasting-Fehler und listet zu jeder Zeitreihe das Modell (bzw. die Modelle), welches den niedrigsten Fehlerwert produziert hat. Abbildung 5.1 zeigt die schematische Darstellung des zugrunde liegenden Datenbankentwurfs.

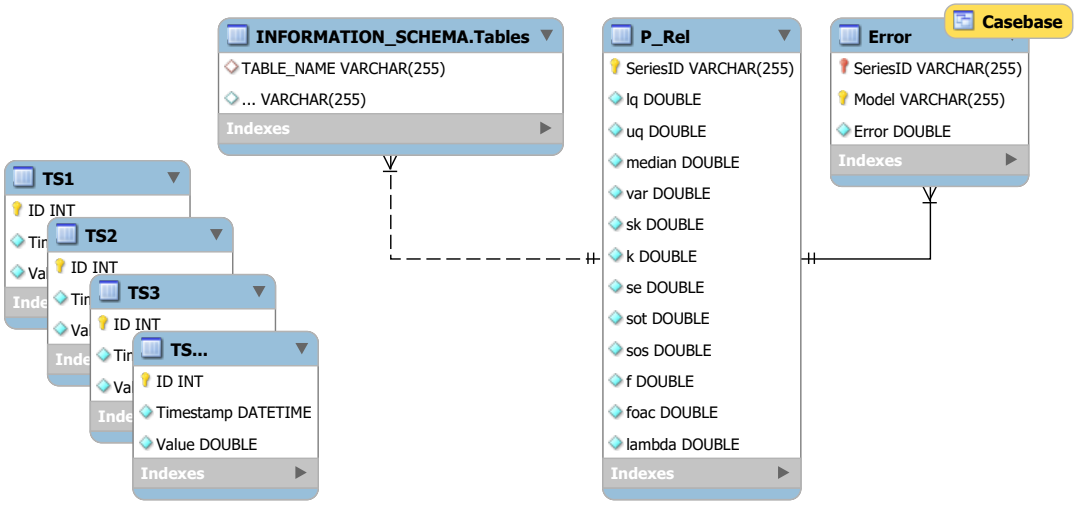


Abbildung 5.1: Schema der Evaluationsdatenbank

5.1.2 Verwendete Modelle und Methodik zur experimentellen Zeitreihenvorhersage

Zur Initialisierung der Fallbasis werden auf jeder Zeitreihe acht Vorhersagemodelle angewendet. Zunächst werden einige Varianten eines autoregressiven Modells (siehe [CC08]) auf die jeweiligen Zeitreihen angewendet. Diese Modelle definieren vorhergesagte Werte derart, dass sie linear von den Trainingsdaten und einem stochastischen Term abhängen. Ein AR-Modell p-ter Ordnung $AR(p)$ ist als

$$X_t = c + \sum_{i=1}^{p} \varphi_i X_{t-i} + \epsilon_t$$

definiert, wobei $\varphi_1, \cdots, \varphi_i$ die Modellparameter darstellen, c eine Konstante und ϵ_t Rauschen bezeichnet. Im Folgenden werden auf allen Zeitreihen die Modelle $AR(1), AR(2), AR(3)$ und $AR(4)$ berechnet.

Zusätzlich zu den vier autoregressiven Modellen wird ein optimales sogenanntes ARIMA-Modell (siehe [HA14]) zu jeder Zeitreihe geschätzt. Dies geschieht mit Hilfe der Funktion `auto.arima` des *R*-Pakets `forecast` (siehe [HK08], [Hyn16]). ARIMA-Modelle sind eine Verallgemeinerung der ARMA-Modelle und betrachten neben dem autoregressiven Teil des Modells auch einen „integrierten gleitenden Durchschnitt“ (engl. integrated moving average).

Als Beispiel eines Verfahrens zur exponentiellen Glättung wird das Holt-Winters Modell (siehe [HA14]) benutzt. Holt-Winters ist ein Verfahren, bei dem eine sogenannte doppelte exponentielle Glättung durchgeführt wird. Das bedeutet, dass ein exponentieller Filter rekursiv auf den Trainingsdaten angewendet und so das entsprechende Modell abgeleitet wird.

Weiterhin kommt eine automatisierte Methode zur Schätzung des optimalen Exponential Smoothing State Space (ETS) Modells (siehe [HAA08], [HKOS08]) auf jeder Zeitreihe zur Anwendung. Die Berechnung eines solchen Modells läuft vollautomatisch durch Aufruf der Funktion `ets` in *R*. Laut [Hyn02] zeigte diese Methode auf den M3-Daten hervorragende Ergebnisse. Da die M3-Daten einen großen Teil der Datenbasis dieser Arbeit ausmachen, ist eine Aufnahme dieser Vorhersagemethode sinnvoll.

Schließlich wird als letztes Modell der naive Forecast verwendet. Bei diesem Modell wird lediglich der letzte Wert der Trainingsdaten kontinuierlich fortgeschrieben. Es wird in Betracht gezogen, da es ein sehr einfaches Modell darstellt, jedoch laut [HA14] häufig erstaunlich gute Leistungen in der Vorhersage erzielt.

Jede der 3.287 Zeitreihen der Evaluationsdatenbank wurde in zwei Segmente unterteilt – Trainingsdaten und Testdaten. Die Trainingsdaten dienen dazu, die jeweiligen Vorhersagemodelle zu berechnen. Mit Hilfe der Testdaten (auch Evaluierungszeitraum genannt) erfolgt die Bestimmung des jeweiligen Forecasting-Fehlers.

Für die M3-Daten ist je Zeitreihe der Evaluierungszeitraum genau festgelegt. Dieser bezieht sich immer auf die Anzahl der letzten Beobachtungen der jeweiligen Zeitreihe und schwankt dabei zwischen 8% und 33% der korrespondierenden Länge. Alle verbleibenden Zeitreihen wurden manuell durch Abgrenzung der letzten 5% der Beobachtungen in die beiden Segmente unterteilt.

Diese Grenze ist vor allen Dingen im Hinblick auf die Längen der Energiezeitreihen sinnvoll. Die Evaluierungszeiträume sollten zwar einen gewissen messbaren Teil der Gesamtreihe ausmachen, jedoch nicht eine exorbitante Anzahl von Werten aufweisen.

Als zugehöriges Fehlermaß kommt der Symmetric Mean Average Percentage Error (SMAPE) zum Einsatz. Diese Entscheidung ist damit zu begründen, dass der SMAPE ein relatives Maß darstellt und somit einen prozentualen Fehler ermittelt, der sich problemlos zwischen Zeitreihen vergleichen lässt, auch wenn diese unterschiedliche Wertebereiche aufweisen. Anders als der Mean Average Percentage Error (MAPE) ist der SMAPE robuster gegenüber auftretenden 0-Werten in den Beobachtungen. Da dies gerade in Energiezeitreihen häufig vorkommen kann, zeigt sich der SMAPE als geeignetes relatives Fehlermaß für diese Evaluation. Der SMAPE ist definiert als:

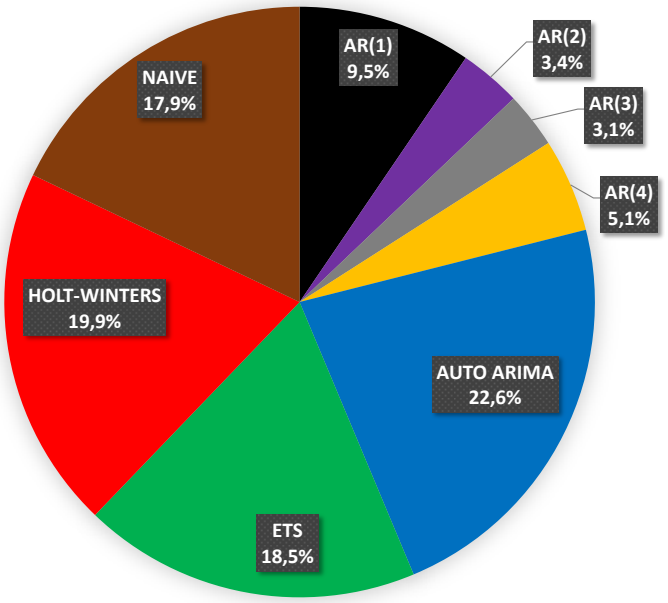
$$SMAPE(F_t, A_t) = \frac{1}{n} \sum_{t=1}^{n} \frac{|F_t - A_t|}{(F_t + A_t)/2}$$

Hobei bezeichnet F_t den durch das jeweilige Modell vorhergesagten Wert und A_t den tatsächlich beobachteten Wert zum Zeitpukt t.

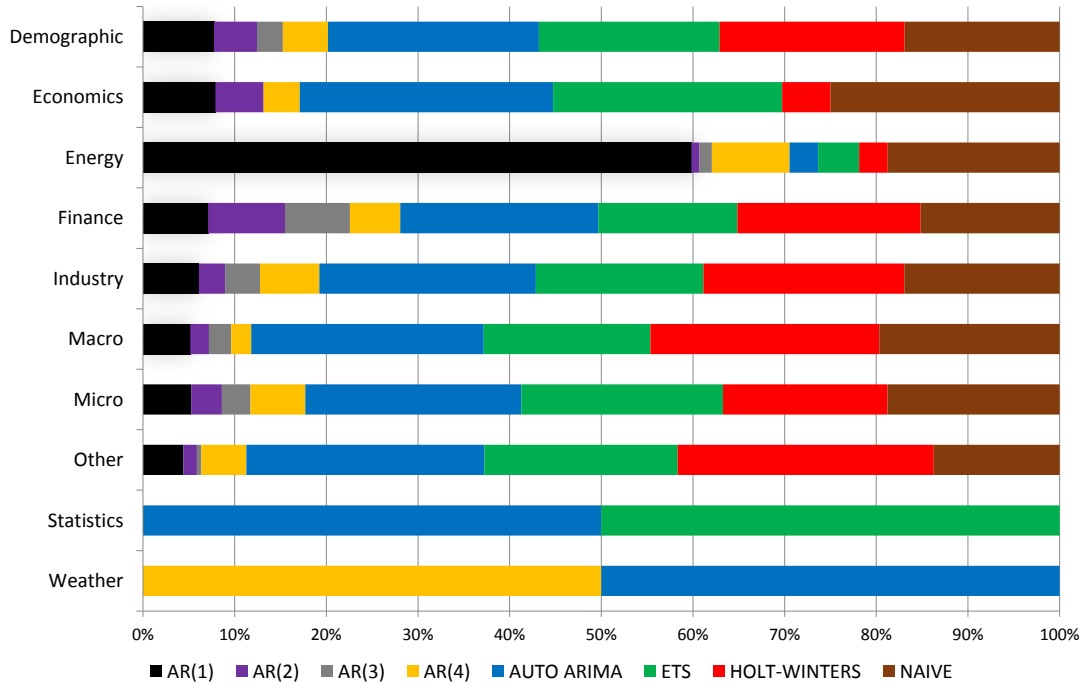
Durch sukzessive Berechnung der Fehler aller Modelle je Zeitreihe, ergibt sich deren Zuordnung gemäß der Funktion *label* (vgl. Abschnitt 4.2). Eine detaillierte Auflistung der jeweiligen Zuordnungen je Zeitreihe würde den Rahmen dieser Arbeit übersteigen. Stattdessen sollen die Diagrammdarstellungen in Abbildung 5.2 auf Basis von Tabelle 5.1 einen Einblick in die Verteilung der einzelnen Modelle geben.

Modell	**Domäne**										**Fallbasis**
	Micro	**Industry**	**Macro**	**Finance**	**Demographic**	**Other**	**Economics**	**Energy**	**Statistics**	**Weather**	
AR(1)	33	6	134	22	32	38	44	9	0	0	318
AR(2)	20	4	2	26	15	15	28	3	0	0	113
AR(3)	12	0	3	22	20	18	26	1	0	0	102
AR(4)	21	3	19	17	34	16	50	10	0	1	171
AUTO ARIMA	98	21	7	67	124	187	197	53	1	1	756
ETS	84	19	10	47	96	134	184	43	1	0	618
HOLT-WINTERS	86	4	7	62	115	184	150	57	0	0	665
NAIVE	72	19	42	47	89	145	157	28	0	0	599
Gesamt	**426**	**76**	**224**	**310**	**525**	**737**	**836**	**204**	**2**	**2**	**3.342**

Tabelle 5.1: Verteilung der Modelle als Klassifikationslabels in der Evaluationsdatenbank



(a) Verteilung der Modelle als Klassifikationslabels in der Fallbasis



(b) Verteilung der Modelle als Klassifikationslabels je Domäne

Abbildung 5.2: Grafische Darstellung der Verteilung der Modelle als Klassifikationslabels in der Evaluationsdatenbank

Bei genauerer Betrachtung von Tabelle 5.1 fällt auf, dass der berechnete Gesamtwert der Modellvorkommen in der Fallbasis von der Mächtigkeit der Fallbasis von 3.287 Zeitreihen um 55 zusätzliche Vorkommen abweicht. Dieser Wert von insgesamt 3.342 Modellvorkommen ist dadurch zu erklären, dass einige Zeitreihen[1] – wie bereits in Abschnitt 4.2 erwähnt – mehreren Modellen zugeordnet werden können. Wenn in den nachfolgenden Untersuchungen der Klassifikatoren eine Zeitreihe mit mehr als einem Klassifikationslabel als ähnlichste Reihe bestimmt wird, erfolgt stets die Auswahl des ersten gefundenen Labels als Empfehlung. Im späteren operativen Einsatz sollte jedoch ein entsprechender Hinweis ausgegeben werden, dass die ermittelte Empfehlung die Auswahlmenge lediglich auf eine Teilmenge eingrenzen konnte.

5.1.3 Aufbau einer Experimentenserie zur empirischen Klassifikatorevaluation

Zur Evaluation der einzelnen Klassifikatoren werden unterschiedliche Experimente vorbereitet. Zunächst sollen die Klassifikatoren kontextunabhängig eine generelle Klassifikation der Fallbasis ermitteln. Hierfür kommt eine Serie von 100 Experimenten zum Einsatz, wobei jedes Experiment eine stratifizierte Stichprobe von 100 Zeitreihen aus der gesamten Datenbasis extrahiert. Die Zusammensetzung richtet sich dabei nach den prozentualen Verhältnissen, welche in Abbildung 3.3 dokumentiert sind, und spiegelt diese in jeder zufällig gewählten Stichprobe wider. Die Zeitreihen der Datenbasis, abzüglich derer, welche als Stichprobe zur Klassifikation extrahiert wurden, dienen dazu, den spezifischen Klassifikator zu ermitteln.

In einem zweiten Schritt erfolgt eine weitere Serie von Experimenten. Hierbei wird die Berechnung der Klassifikatoren jeweils innerhalb der spezifizierten Datendomänen durchgeführt. Die einzigen Ausnahmen in diesem zweiten Schritt bilden die Domänen *Statistics* und *Weather*. Auf Grund ihrer dünnen Besetzung von je nur zwei Zeitreihen wäre eine Unterteilung in Trainingsdaten und Stichprobe unsinnig, da der trainierte Klassifikator hier dem Phänomen der Überanpassung erliegen würde.

Des Weiteren wäre es nicht möglich gewesen, bei nur zwei Zeitreihen je Domäne eine Unterteilung in 90% Trainingsdaten und 10% Testdaten vorzunehmen. Diese Aufteilung ist je Domäne bei allen Experimenten verwendet worden. Um ein analoges Vorgehen wie im ersten Schritt zu wahren, werden auch je Domäne 100 Experimente durchgeführt. Die Menge von Testzeitreihen ist je Experiment zufällig nach dem eben eingeführten Verteilungsschlüssel bestimmt. Abbildung 5.3 zeigt die jeweiligen Größen der Trainings- und Testdatenmengen je Domäne.

Für alle durchgeführten Experimente gilt, dass jeder Klassifikator auf denselben Trainings- und Testdaten arbeitet. Das bedeutet, wenn Experiment $E_1 = \left(Training_{E_1}, Test_{E_1}\right)$ durch die Klassifikatoren K_1 und K_2 bearbeitet wird, finden sie jeweils dieselben Datensets $Training_{E_1}$ und $Test_{E_1}$ vor. Dieses Vorgehen ist obligatorisch, um eine Vergleichbarkeit der Ergebnisse der einzelnen Verfahren zu erwirken.

[1] Insgesamt haben 51 Zeitreihen je 2 Klassifikationslabels und 2 Zeitreihen besitzen 3 Klassifikationslabels. Alle anderen Zeitreihen der Evaluationsdatenbank werden durch ein einziges Label klassifiziert.

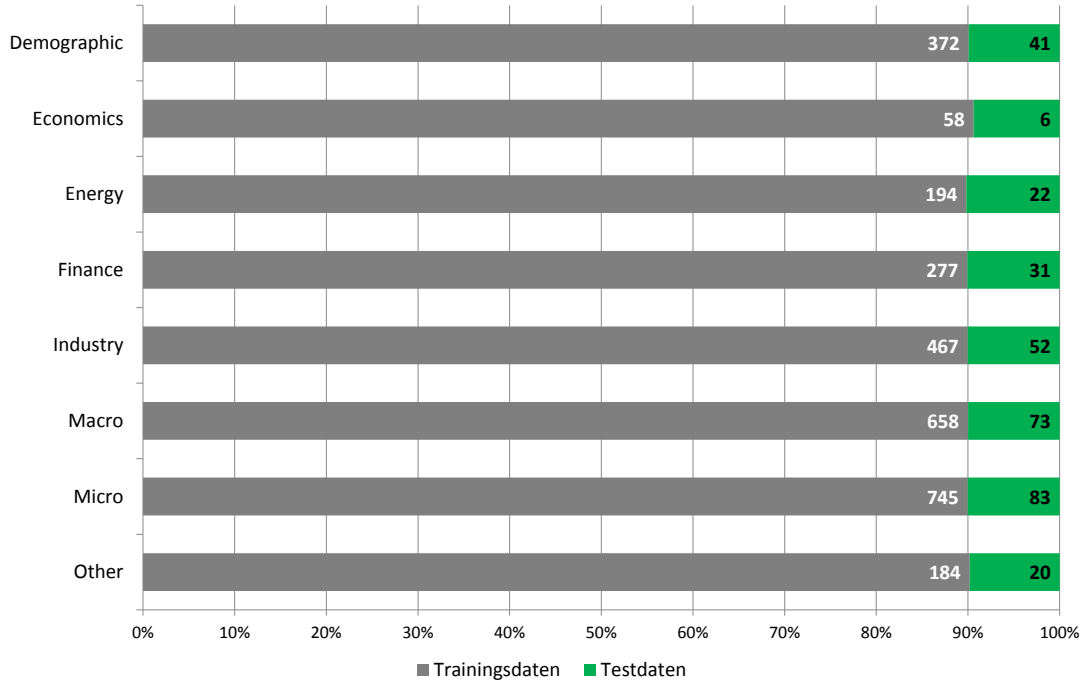


Abbildung 5.3: Aufteilung der Zeitreihen je Domäne in Trainings- und Testdaten

5.2 EVALUATION DES 1-NEAREST-NEIGHBOR KLASSIFIKATORS

Nachdem im vorangegangenen Abschnitt ausführlich die verwendete Methodik und Grundlage der Evaluation erläutert wurde, erfolgt in diesem Abschnitt die Betrachtung des ersten Klassifikators – einem trivialen 1NN-Ansatz. Als Ergebnis von Kapitel 4 stand der 12-dimensionale Vektor P_{rel}, welcher die herausgefilterten, als relevant befundenen numerischen Eigenschaften einer Zeitreihe repräsentiert. Mit Hilfe der in Kapitel 3 aufgebauten Datenbasis wurden bereits zu jeder Zeitreihe die zugehörigen Eigenschaften und zu acht Modellen die entsprechenden SMAPE-Werte ermittelt. In der folgenden Evaluation betrachtet Abschnitt 5.2.1 den Ansatz in einem kontextfreien Szenario. Abschnitt 5.2.2 beinhaltet schließlich die Analyse des 1NN-Klassifikators unter Einbeziehung der Datendomäne als nicht-numerische Kontextinformation.

5.2.1 Kontextfreie Evaluation

Zunächst soll der 1NN-Klassifikator in einem kontextfreien Szenario untersucht werden. Das bedeutet, dieses Verfahren berechnet im korrespondierenden Suchraum die euklidische Distanz eines „anfragenden“ Zeitreihenvektors zu *allen* existierenden Vektoren einer Vergleichsmenge. Etwaige nicht-numerische Kontextinformationen werden hierbei außer Acht gelassen. Der räumlich nächste Nachbar im jeweiligen Suchraum – und nur dieser – bildet als ähnlichster Vektor zum Anfragevektor das Ergebnis der Suche. Ohne Beschränkung der Allgemeinheit wird hier der zuerst gefundene Vektor als Ausgabe verwendet, sollte mehr als ein Vergleichsvektor dieselbe Distanz aufweisen.

Diese Methodik extrahiert zu jeder Zeitreihe ein ähnlichstes, eigenschaftsrepräsentiertes Pendant

aus der jeweiligen Vergleichsmenge. Der einzig unklare Faktor zum gegenwärtigen Stand ist, ob die alleinige Betrachtung des 12-dimensionalen Eigenschaftsraumes tatsächlich einer erschöpfenden Ähnlichkeitssuche Rechnung trägt. Grundlegend wäre es auch denkbar, dass nicht alle der 12 ermittelten Eigenschaften für eine erfolgreiche Klassifikation benötigt werden. Um diesen Faktor zu eliminieren findet eine Untersuchung aller möglichen Teilräume von P_{rel} statt.

An dieser Stelle zeigt sich ein Vorteil der analysebedingten Reduktion der Eigenschaftsmenge aus Kapitel 3. Bei der gegenwärtigen Anzahl von 12 Dimensionen müssen insgesamt

$$\#R = \sum_{i=1}^{12} \left(\begin{array}{c} 12 \\ i \end{array} \right) = 4.095$$

Räume durchsucht werden. Wäre es bei der ursprünglichen Menge von 18 Eigenschaften geblieben, läge die Anzahl zu durchsuchender Räume bei $\#R = 262.143$! Diese signifikante Reduzierung der zu durchsuchenden Räume bekommt vor allem dann einen beachtlichen Stellenwert, wenn betrachtet wird, dass jedes einzelne Experiment ein Mal für jeden der möglichen Teilräume gestartet werden muss. Nur so ist der Teilraum, welcher die besten Klassifikationsergebnisse erzielt, sicher identifizierbar.

Ein idealer n-dimensionaler Klassifikationsraum sollte grundlegend in der Lage sein, die Zeitreihenvektoren räumlich so zu trennen, dass jeder separate Cluster von einem spezifischen Modell hinreichend dominiert wird. Um den optimalen Klassifikationsraum zu finden, wird nun jedes Experiment in jedem der 4.095 Teilräume ausgeführt. Auf Grund der ermittelten Vorhersagemodelle wird der Forecasting-Fehler als SMAPE-Wert je Zeitreihe der Testmenge des Experiments in der Datenbank nachgeschlagen. Sei $TS_{j,i}^{S_l} \in Test\left(E_i^{S_l}\right)$ die j-te zu klassifizierende Zeitreihe des Experiments E_i im Teilraum S_l. Für diesen Teilraum gilt $S_l \in \mathcal{P}^+(P_{rel})$ mit $\mathcal{P}^+(P_{rel}) = \mathcal{P}(P_{rel}) \setminus \emptyset$, wobei $\mathcal{P}(P_{rel})$ die Potenzmenge der Menge P_{rel} bezeichnet. Des Weiteren sei $TS_{k,i\sim j,i}^{S_l} \in Training\left(E_i^{S_l}\right)$ die zu $TS_{j,i}^{S_l}$ ähnlichste Zeitreihe der Trainingsmenge des Experiments E_i im Teilraum S_l. Die Beobachtungszeitpunkte des Evaluierungszeitraums der Zeitreihe $TS_{j,i}^{S_l}$ seien mit h bezeichnet. A_h entspricht damit den tatsächlichen Messwerten der Zeitreihe $TS_{j,i}^{S_l}$ im Evaluierungszeitraum h, während F_h^M die vorhergesagten Werte von $TS_{j,i}^{S_l}$ im Evaluierungszeitraum h unter der Verwendung des Modells $M \in label\left(TS_{k,i\sim j,i}^{S_l}\right)$ definiert. Dieses Modell M ist das Klassifikationslabel[2] der ähnlichsten Zeitreihe $TS_{k,i\sim j,i}^{S_l}$ und stellt damit das empfohlene Vorhersagemodell zur Reihe $TS_{j,i}^{S_l}$.

Der Durchschnitt aller SMAPE-Werte je Experiment bildet den erreichten gemittelten Fehlerwert $\overline{E_i^{S_l}}$ des Teilraums S_l in Experiment E_i und wird durch

$$\overline{E_i^{S_l}} = \frac{1}{100} \sum_{j=1}^{100} SMAPE\left(F_h^M, A_h\right)$$

berechnet. Durch die 100 ausgeführten Experimente entstehen je Teilraum insgesamt 100 gemittelte SMAPE-Werte. Deren Mittelwert definiert die Bewertung der Gesamtperformanz des Teilraums S_l, welche sich durch $\overline{S_l} = \left(\sum_{i=1}^{100} \overline{E_i^{S_l}}\right) / 100$ bestimmt.

[2] Bei mehr als einem Klassifikationslabel der ähnlichsten Zeitreihe wird das erste gefundene Modell verwendet.

Mit Hilfe der Performanzbewertungen aller möglichen Teilräume kann der beste Teilraum mittels $S_{opt} = S_l \mid \overline{S_l} = min\{S_h \mid 1 \leq h \leq 4.095\}$ ermittelt werden. Hierbei stellt S_l den zu $\overline{S_l}$ gehörigen Teilraum dar. Sollten mehrere Teilräume denselben optimalen gemittelten Fehler erzielen, erhält der Teilraum mit der geringsten Dimensionalität den Vorzug.

Durch Anwendung der eben erläuterten Methodik, stellte sich der 6-dimensionale Eigenschaftsraum $S_{opt}^{casebase} = foac(T), se(T), sk(T), sos(T), sot(T), var(T)$ als der Teilraum mit den geringsten gemittelten SMAPE-Werten und der geringsten Dimensionalität heraus[3]. In Anbetracht dieses Ergebnisses zeigt sich, dass die Entscheidung, das Merkmal $sk(T)$ trotz indizierter starker Korrelation dennoch in P_{rel} aufzunehmen, gerechtfertigt war.

Abbildung 5.4 zeigt in den blau gefärbten Säulen eine Übersicht der erzielten SMAPE-Werte auf allen 100 Experimenten. Zur Einschätzung der Lage des erzielten Ergebnisses, sind darüber hinaus zwei weitere Säulenarten abgebildet. Die grün gefärbten Säulen in Abbildung 5.4 bilden das jeweils zu erreichende Optimum. Sie zeichnen den gemittelten Fehler unter ausschließlicher Verwendung des jeweils besten Modells einer Zeitreihe ab. Durch analoges Vorgehen unter ausschließlicher Verwendung des jeweils schlechtesten Modells, erschließen sich die rot gefärbten Säulen. Wie in der Abbildung zu sehen ist, wird durch die Verwendung der Empfehlung durchschnittlich ein um ca. 32 Prozentpunkte besserer Forecasting-Fehler erreicht als im schlechtesten Fall. Die durchschnittliche Abweichung vom Optimum beträgt rund 10 Prozentpunkte.

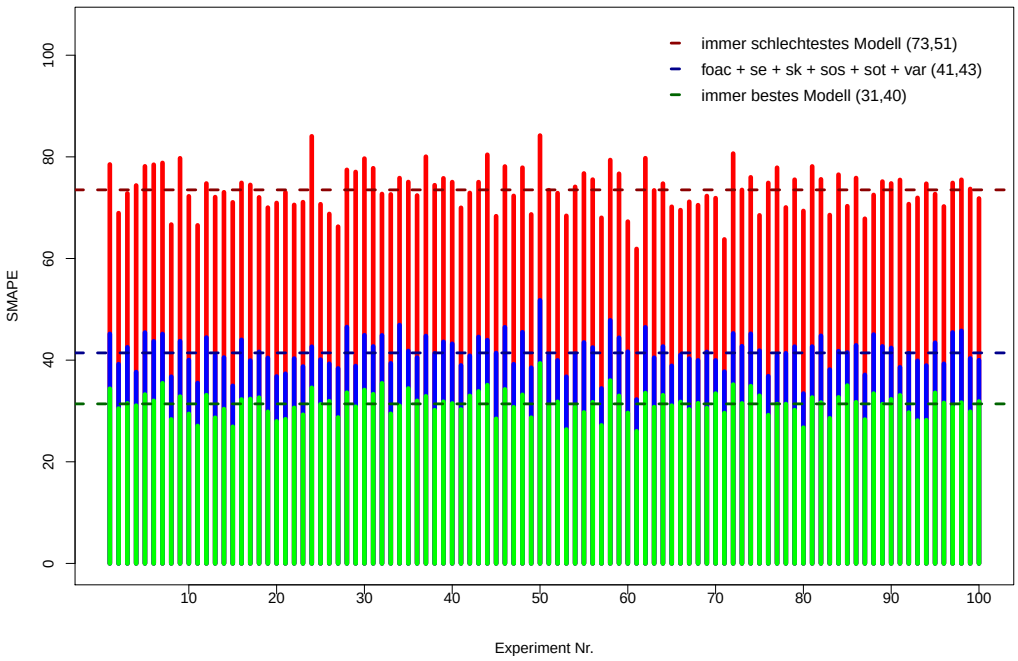


Abbildung 5.4: Erzielte SMAPE-Werte des besten Teilraumes auf der Datenbasis

Zur Abgrenzung des besten ermittelten 1NN-Klassifikationsraums illustriert Abbildung 5.5 den schlechtesten ermittelten Teilraum (schwarze Säulen). Dieser stellt sich als der 1-dimensionale Teilraum $S_{worst}^{casebase} = f(T)$ heraus. Wie die Abbildung zeigt, heben sich die erzielten Fehlerwerte zwar weiterhin signifikant von den Werten der roten Säulen ab, jedoch ist hier die Abweichung zum Optimum mit rund 17 Prozentpunkten deutlich höher als beim besten Teilraum.

[3]Tatsächlich erzielte der 7-dimensionale Teilraum $f(T), foac(T), se(T), sk(T), sos(T), sot(T), var(T)$ den gleichen SMAPE-Wert, verlor jedoch auf Grund seiner höheren Dimensionalität gegen den wertgleichen 6-dimensionalen Raum.

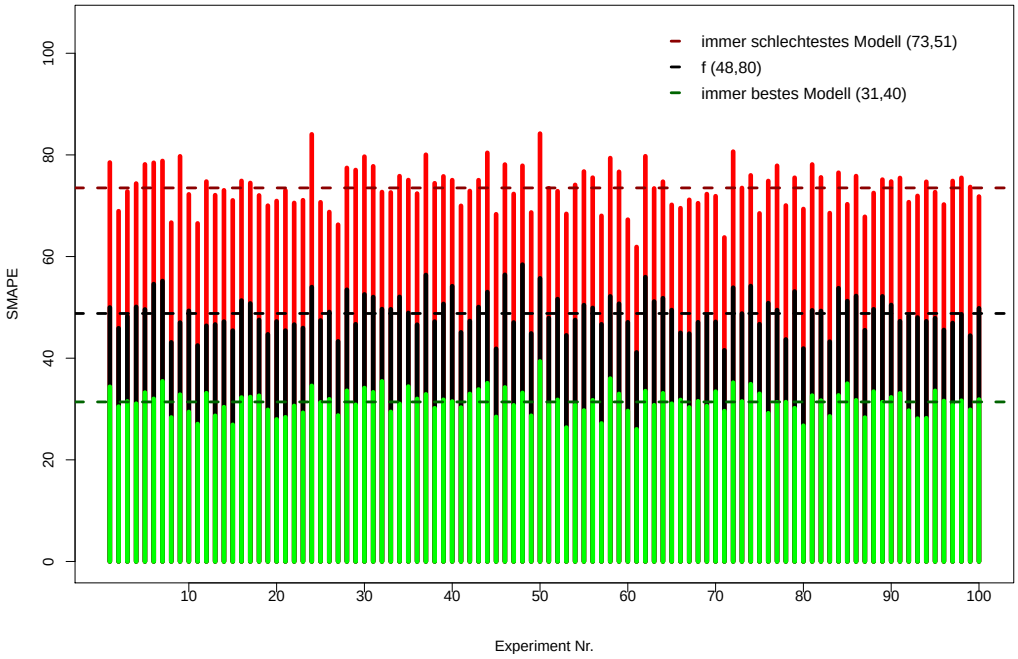


Abbildung 5.5: Erzielte SMAPE-Werte des schlechtesten Teilraumes auf der Datenbasis

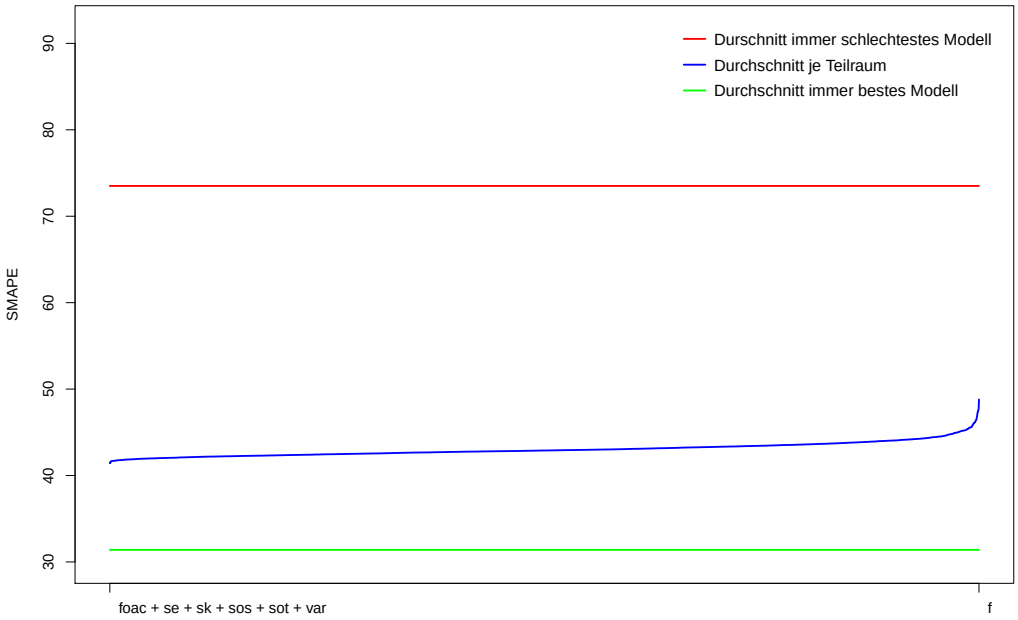


Abbildung 5.6: Streuung der SMAPE-Werte aller Teilräume auf der Datenbasis

Abbildung 5.6 beinhaltet eine Übersicht der Streuung der erzielten SMAPE-Werte je Teilraum. Die Räume wurden dabei nach ihren erzielten gemittelten Fehlern aufsteigend sortiert. Demnach liegt $S_{opt}^{casebase}$ in dieser Grafik ganz links, während $S_{worst}^{casebase}$ ganz rechts verzeichnet ist. Es lässt sich aus diesem Diagramm erkennen, dass der Großteil der Räume sehr dicht beieinander liegt und nur schleppend eine Verbesserung der SMAPE-Werte erreicht wird. Lediglich ein geringer Teil der Modelle (rechts zu sehen) hebt sich deutlich ab und streut in einem ähnlichen Ausmaß wie die restlichen rund 98% der Räume.

5.2.2 Evaluation unter Einbeziehung nicht-numerischer Kontextinformationen

Die vorangegangene kontextfreie Evaluation offenbarte einen 6-dimensionalen Teilraum der Eigenschaftsmenge P_{rel} als besten 1NN-Klassifikationsraum über der gesamten Datenbasis. Eine Betrachtung der Streuung der einzelnen Teilräume anhand ihrer SMAPE-Werte zeigte jedoch, dass alle Teilräume in ihrer Klassifikationsleistung sehr dicht beieinander lagen. Es stellt sich unmittelbar die Frage, ob es weitere Einflussgrößen gibt, welche zu einer deutlicheren Abgrenzung der Klassifikationsleitungen unterstützend beitragen. Solche Einflussgrößen sind grundlegend nicht-numerischer Art, da sie sonst bereits in die Betrachtung der numerischen Kennzahlen zur eigenschaftsgestützten Repräsentation eingeflossen wären.

Als sofortiger Kandidat einer solchen nicht-numerischen Kontextinformation, bietet sich die Datendomäne an. Sie ist nicht quantifizierbar und realisiert eine disjunkte Kategorisierung auf den zugrunde liegenden Daten. Da für alle verfügbaren Zeitreihen der Evaluationsdatenbank lediglich die Datendomäne als nicht-numerische Kontextinformation verfügbar ist, beschränkt sich die folgende Evaluation auf die alleinige Analyse dieser.

Sollte es tatsächlich domänenspezifische Eigenarten in den Streuungen der Eigenschaftswerte geben, müsste sich (I) je Domäne ein abweichender n-dimensionaler Teilraum ermitteln lassen. Darüber hinaus müsste besagter Raum (II) auf den domänenspezifischen Experimenten bessere Klassifikationsergebnisse erzielen, als der in Abschnitt 5.2.1 bestimmte kontextfreie Teilraum.

Zur Überprüfung der Annahmen (I) und (II) werden die in Abschnitt 5.1.3 erläuterten domänenspezifischen Experimente durchgeführt. Die Ermittlung der besten Teilräume $S_{D_{opt}}$ je Domäne D erfolgt analog zu der in Abschnitt 5.2.1 eingeführten Methodik. Der einzige Unterschied ist, dass alle Vorkommen des Terms 100 in der Berechnung der $\overline{E_{D_i}^{S_l}}$ durch die in Abbildung 5.3 gezeigten Testmengenmächtigkeiten ersetzt werden.

Die Berechnung des gemittelten Fehlers $\overline{E_{D_i}^{S_l}}$ des Experiments E_{D_i} der Domäne $D = Energy$ im Teilraum S_l, ergibt sich mit $\left|Test\left(E_{D_i}\right)\right| = 22$ als $\overline{E_{D_i}^{S_l}} = \left(\sum_{j=1}^{22} SMAPE\left(F_h^M, A_h\right)\right)/22$. An der Berechnung der Teilraumperformanz $\overline{S_{D_l}}$ ändert sich hingegen nichts. Sie ist weiterhin als $\overline{S_{D_l}} = \left(\sum_{i=1}^{100} \overline{E_{D_i}^{S_l}}\right)/100$ definiert, da auch hier jede Experimentenserie aus 100 Einzelexperimenten besteht.

Die Auswertung der einzelnen Experimentenserien resultierte in den durch Tabelle 5.2 dokumentierten, besten domänenspezifischen Klassifikationsräumen. Eine Übersicht über die entsprechenden Streuungen der SMAPE-Werte jeder Experimentenserie ist in Abbildung 5.7 enthalten. Durch die Analyse zeigte sich, dass in jeder Domäne ein anderer als der global beste Klassifikationsraum die geringsten gemittelten Fehlerwerte erzeugt. Daher gilt die vormals erhobene Annahme (I) als bestätigt.

Es zeigt sich jedoch, dass keine konstante Aussage zur Dimensionalität der jeweils besten Räume getroffen werden kann. Tatsächlich fluktuiert die Raumdimensionalität stark zwischen den verschiedenen Domänen. Der beste Klassifikationsraum der Domäne *Industry* bildet in dieser Betrachtung einen klaren Extremwert. Das errechnete Ergebnis konstatiert, dass der Klassifikati-

Domäne	**Klassifikationsraum**	**Dimensionalität**
Demographic	$f(T), foac(T), \lambda(T), lq(T), \widetilde{T}, se(T), sos(T), sot(T), uq(T), var(T)$	10
Economics	$f(T), \lambda(T), sk(T), uq(T)$	4
Energy	$k(T), \widetilde{T}, sot(T), var(T)$	4
Finance	$k(T), \lambda(T), lq(T), se(T), sos(T), sot(T), var(T)$	7
Industry	$f(T)$	1
Macro	$f(T), \lambda(T), lq(T), \widetilde{T}, se(T), sk(T), sos(T), var(T)$	8
Micro	$k(T), \lambda(T), \widetilde{T}, sk(T), sos(T), sot(T), var(T)$	7
Other	$f(T), \lambda(T), lq(T), se(T)$	4

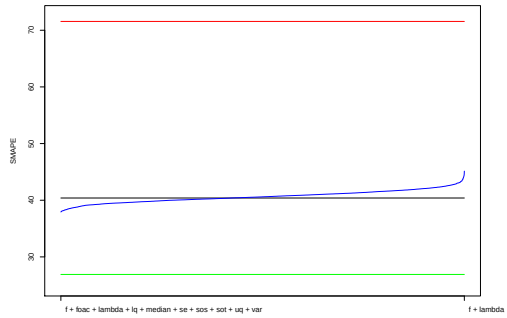
Tabelle 5.2: Beste domänenspezifische Klassifikationsräume

onsraum $S_{opt}^{Industry} = f(T)$, also der alleinige Vergleich der Periodizität, zur Modellklassifikation ausreichend ist. Tatsächlich wurde dieser Raum in drei anderen domänenspezifischen Analysen sogar als schlechtester Klassifikationsraum bestimmt. Die Wahl zum besten Raum in der Domäne *Industry* ist somit überraschend und könnte ein Indiz dafür sein, dass die gewählten Eigenschaften aus P_{rel} diese Domäne möglicherweise nicht ausreichend charakterisieren.

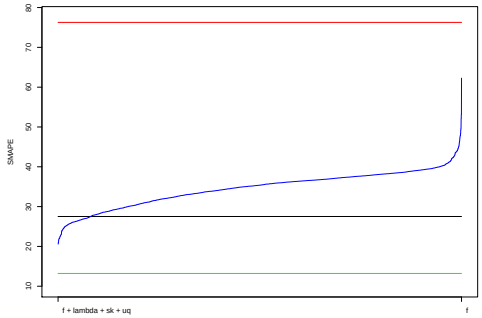
Tabelle 5.3 dokumentiert die erzielten Verringerungen der SMAPE-Werte durch Anwendung des errechneten domänenspezifischen Klassifikationsraums. Durch die Verwendung der speziellen Räume in allen Domänen ist stets eine Verbesserung der durch die empfohlenen Modelle entstandenen Fehlerwerte zu verzeichnen. Dies gilt sogar für die Domäne *Industry*, wenn auch die Verringerung lediglich 1,77 Prozentpunkte beträgt. Erstaunlicherweise ist die dort erzielte Verringerung im Vergleich nicht die geringste. Obwohl durch die domänenspezifischen Klassifikationsräume nur teils marginale Verbesserungen gegenüber dem global besten Klassifikationsraum erzielt werden konnten, ist die Verbesserung in allen untersuchten Domänen vorhanden. Dies bestätigt auch die vormals aufgestellte Annahme (II), dass die Einbeziehung nicht-numerischer Kontextinformationen zu einer Steigerung der Klassifikationsgüte führt.

Domäne	**bester globaler Raum**	**bester domänenspezifischer Raum**	**Differenz**
Demographic	40,39	37,93	-2,47
Economics	27,51	19,98	-7,53
Energy	74,69	73,08	-1,61
Finance	40,10	38,84	-1,26
Industry	33,87	32,10	-1,77
Macro	18,36	17,62	-0,74
Micro	46,41	46,05	-0,36
Other	86,07	81,69	-4,37

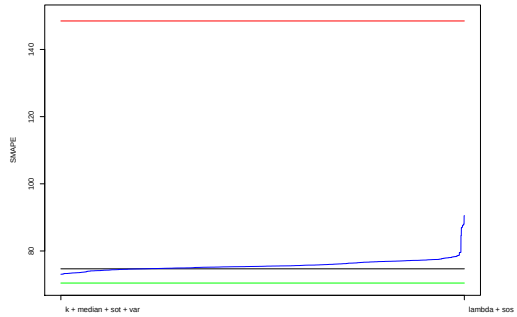
Tabelle 5.3: Erzielte Verringerungen der SMAPE-Werte durch domänenspezifische Klassifikationsräume



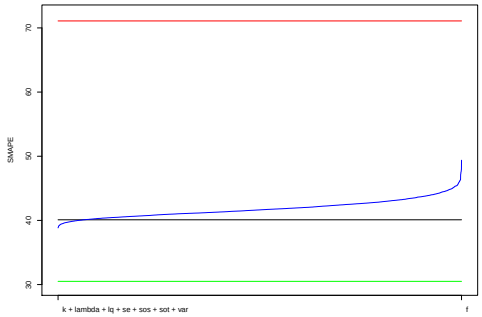
(a) Streuung der Domäne *Demographic*



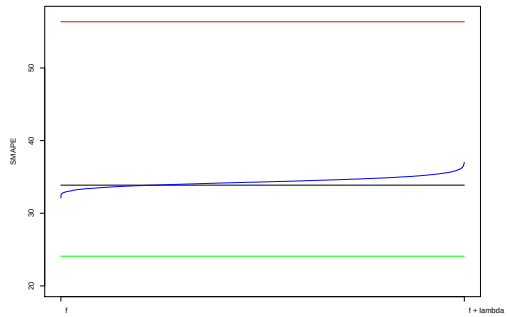
(b) Streuung der Domäne *Economics*



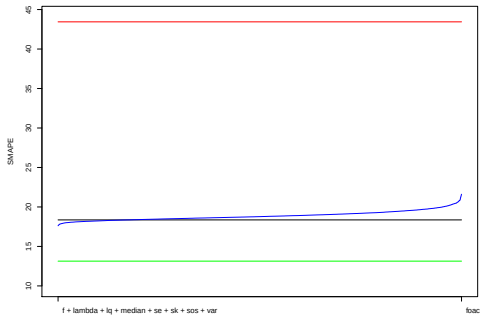
(c) Streuung der Domäne *Energy*



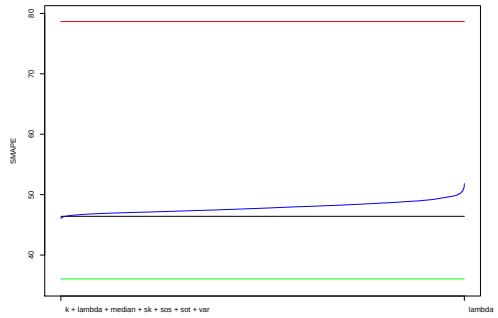
(d) Streuung der Domäne *Finance*



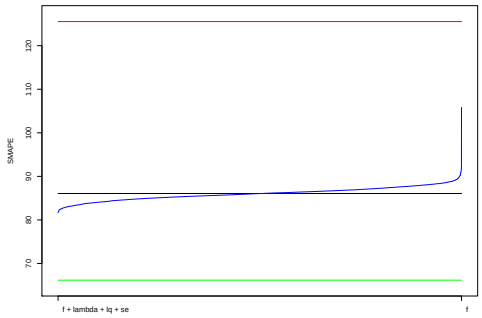
(e) Streuung der Domäne *Industry*



(f) Streuung der Domäne *Macro*



(g) Streuung der Domäne *Micro*



(h) Streuung der Domäne *Other*

Abbildung 5.7: Streuung der SMAPE-Werte aller Teilräume je Domäne

Hinsichtlich der Streuung der SMAPE-Werte der einzelnen Teilräume offenbart Abbildung 5.7, dass die jeweiligen Kurven keine grundlegenden Unterschiede in ihrem Verlauf gegenüber der Kurve aus Abbildung 5.6 aufweisen. Es lässt sich jedoch erkennen, dass die einzelnen Streubereiche variieren. Der überwiegende Teil dieser Bereiche charakterisiert sich als nah beieinanderliegend, so wie in den Abbildungen 5.7a, 5.7c, 5.7d, 5.7e, 5.7f und 5.7g zu sehen ist. Die Verteilung der Werte könnte in allen diesen Domänen als nahezu linear beschrieben werden.

Bei den beiden verbleibenden SMAPE-Kurven aus den Abbildungen 5.7b und 5.7h verhält sich dies jedoch anders. Hier zeigt sich ein stark an Abbildung 5.6 angelehntes Verlaufsbild, mit dem Unterschied, dass die Ausläufer an der äußeren rechten Seite einen wesentlich höheren Ausschlag aufweisen. Leider befindet sich dieser signifikante Bruch ebenfalls bei den letzten rund 2% der analysierten Räume. Damit zeigt sich auch in dieser Betrachtung kein Verhalten, in welchem der am besten operierende Teilraum eine deutliche Abhebung in seiner Klassifikationsleistung von seinen Konkurrenten signalisiert.

Nichtsdestotrotz bleibt die Erkenntnis, dass die Trennung der Daten nach ihrer Herkunft eine Verbesserung in der Modellklassifikation erwirkt. Dies zeigt auch Tabelle 5.4. Aus den dort aufgeführten Werten wird ersichtlich, dass alle domänenspezifischen Klassifikationsräume näher an das jeweilige gemittelte Optimum heranreichen, als der global beste Klassifikationsraum.

Domäne	**immer schlechtestes Modell**	**bester globaler Raum**	**bester domänenspezifischer Raum**	**immer bestes Modell (Optimum)**
Datenbasis	73,51	41,43	41,43	31,40
Demographic	71,55	40,39	37,93	26,90
Economics	75,87	27,51	19,98	13,21
Energy	148,49	74,69	73,08	70,45
Finance	71,09	40,10	38,84	30,50
Industry	56,36	33,87	32,10	24,09
Macro	43,44	18,36	17,62	13,14
Micro	78,65	46,41	46,05	36,05
Other	125,53	86,07	81,69	66,17

Tabelle 5.4: Vergleich der ermittelten SMAPE-Werte der globalen und domänenspezifischen Experimente

6 EVALUATION WEITERER KLASSIFIKATOREN

In Kapitel 5 wurde ausführlich die experimentelle Evaluation des 1-Nearest-Neighbor Klassifikators gezeigt. Sie beinhaltete einen kontextfreien und einen den Kontext einbeziehenden Teil. Dieses Kapitel widmet sich dem Vergleich des 1NN-Verfahrens mit anderen gebräuchlichen Verfahren zur Klassifikation.

Die zusätzliche Analyse weiterer Klassifikatoren ist dadurch motiviert, dass das bereits betrachtete Verfahren der 1NN-basierten Klassifikation einen eher trivialen Ansatz darstellt. Es besteht die Möglichkeit, mit sensibleren Verfahren des maschinellen Lernens, eine höhere Klassifikationsgüte zu erzielen. Diese würde sich in einer weiteren Annäherung des gemittelten Fehlers an das globale Optimum zu erkennen geben.

Um dieses Ziel zu erreichen, behandelt Abschnitt 6.1 die Vorstellung und Evaluation von Klassifikationsbäumen. Anschließend wird dieses Verfahren mit dem in Kapitel 5 besprochenen 1NN-Klassifikator kombiniert (Abschnitt 6.2), um so einen Verbundklassifikator zu erzeugen und zu evaluieren. Abschnitt 6.3 zeigt und diskutiert die Verwendung eines MLP Neuronalen Netzwerks, als Variante eines Klassifikators des sogenannten *Deep Learnings*. Zur Abgrenzung gegen naive Greedy-Verfahren diskutiert Abschnitt 6.4 zwei konstruierte sogenannte Random-Walk-Verfahren. Schließlich werden alle bis zu dieser Stelle betrachteten eigenschaftsbasierten Verfahren gegen ein klassisches Distanzmaß evaluiert (Abschnitt 6.5). Hierbei kommen zwei Varianten der euklidischen Distanz als Kriterium der strukturbasierten Ähnlichkeitssuche zur Anwendung.

6.1 ENTSCHEIDUNGSBAUMBASIERTE KLASSIFIKATION

Allgemeine „Classification And Regression Trees (CARTs)“ (vgl. [BFO+84], [Qui86], [YWI99], [RM14]) sind anhand ihrer grundlegenden Konzeption so ausgelegt, dass sie entweder einen Klassifikationsbaum oder einen Regressionsbaum auf einer gegebenen Trainingsdatenmenge er-

mitteln. Die Wahl, welcher spezielle Baum aus einer Trainingsdatenmenge generierbar ist, korreliert mit der Art der abhängigen Variable einer Fallbasis. Ist diese ein kategoriales Attribut, kann über ihr ein Klassifikationsbaum ermittelt werden. Im Fall eines numerischen Attributs ist ein Regressionsbaum das Ergebnis der Baumerstellung.

Auf Grund der Tatsache, dass die Modelle, auf welche die jeweiligen Zeitreihen der Evaluationsdatenbank abgebildet werden, eine Menge kategorialer Ausprägungen darstellen, kann für sie ein Klassifikationsbaum erstellt werden. Zur Erzeugung eines solchen Baumes kommen Algorithmen wie „Iterative Dichotomiser 3 (ID3)“ (siehe [Qui86]) oder seine Erweiterung „C4.5“ (siehe [Qui93]) zum Einsatz. Diese Algorithmen basieren grundlegend auf dem Prinzip der attributbasierten Trennung der Gesamtdatenmenge.

ID3 beispielsweise beginnt seine Ausführung auf der originalen Trainingsdatenmenge TR und berechnet für jedes ungenutzte Attribut a den sogenannten Informationszuwachs (engl. Information Gain (IG); siehe [Mit12]) als:

$$IG(TR, a) = H(T) - \sum_{v \in values(a)} \frac{|\{x \in TR \mid x_a = v\}|}{|TR|} \cdot H\left(\{x \in TR \mid x_a = v\}\right)$$

Hierbei bezeichnet $H()$ jeweils die Entropie. Steht diese Metrik für alle Attribute der Trainingsmenge fest, wählt ID3 das Attribut mit dem höchsten IG aus und trennt die Datenmenge anhand dessen. Durch rekursive Anwendung des Algorithmus auf den erzeugten Teildatenmengen, entsteht sukzessive ein Entscheidungsbaum als erlernter Klassifikator auf den Trainingsdaten. Tabelle 6.1 zeigt das aus [Qui86] entnommene Standardbeispiel, dessen ebenfalls entnommener Klassifikationsbaum in Abbildung 6.1 illustriert ist. *Klasse* stellt hierbei die abhängige Variable dar, welche sich durch die Werte „N“ und „P“ ausprägt.

Fall	**Attribute**				**Klasse**
	Aussicht	**Temperatur**	**Luftfeuchtigkeit**	**windig**	
1	sonnig	heiß	hoch	nein	N
2	sonnig	heiß	hoch	ja	N
3	bewölkt	heiß	hoch	nein	P
4	regnerisch	mild	hoch	nein	P
5	regnerisch	kühl	normal	nein	P
6	regnerisch	kühl	normal	ja	N
7	bewölkt	kühl	normal	ja	P
8	sonnig	mild	hoch	nein	N
9	sonnig	kühl	normal	nein	P
10	regnerisch	mild	normal	nein	P
11	sonnig	mild	normal	ja	P
12	bewölkt	mild	hoch	ja	P
13	bewölkt	heiß	normal	nein	P
14	regnerisch	mild	hoch	ja	N

Tabelle 6.1: Eine beispielhafte Fallbasis

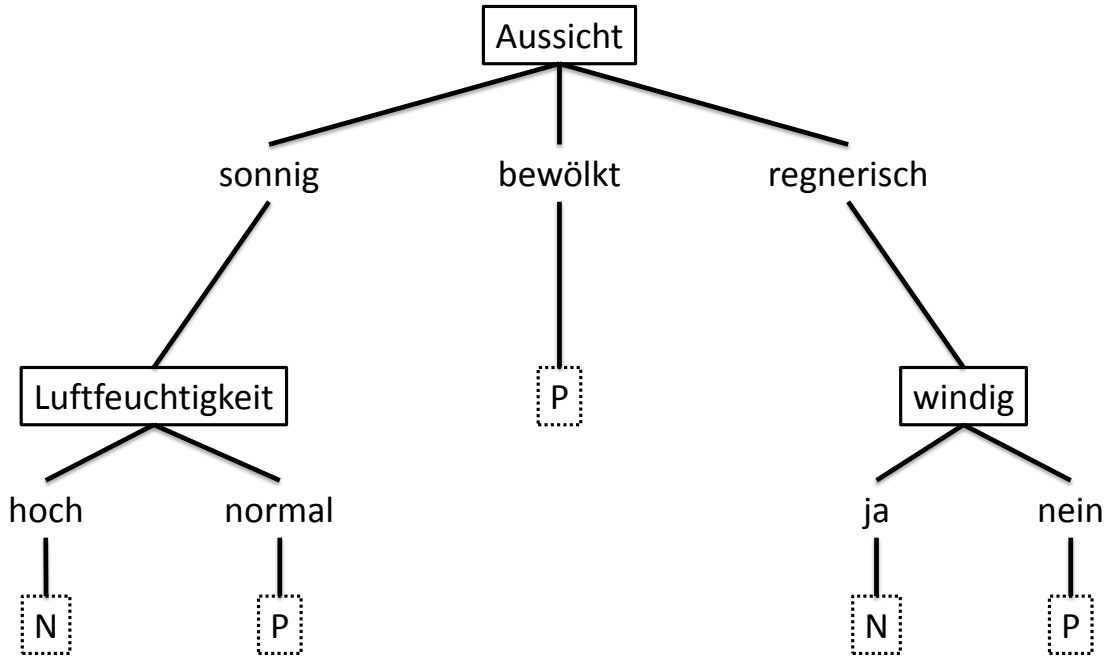


Abbildung 6.1: Der zu Tabelle 6.1 generierte Klassifikationsbaum

Mittlerweile ist es nicht mehr notwendig einen Erstellungsalgorithmus, wie ID3, eigenständig zu implementieren, da es fertige Lösungen gibt, welche komfortabel eingebunden und verwendet werden können. Die freie Statistiksoftware *R* beispielsweise bietet ein Paket namens *rpart* (siehe [Fal13]) an, welches entsprechende Funktionen zur automatisierten Erstellung von Klassifikations- und Regressionsbäumen bereitstellt. Die folgenden Untersuchungen basieren auf der Verwendung der Funktionen dieses Pakets und ermitteln auf jedem der in Abschnitt 5.1.3 beschriebenen Experimente einen Klassifikationsbaum. Analog zu den in Abschnitt 5.2 durchgeführten Analysen wird auch jetzt zu jeder durch Klassifikation entstandenen Empfehlung der erzielte SMAPE-Wert protokolliert.

Abbildung 6.2 zeigt das Ergebnis der Auswertung der Evaluationsexperimente unter Verwendung eines Klassifikationsbaumes auf der gesamten Datenbasis. Es ist zu erkennen, dass der Baum in diesem Szenario keine Verbesserung der erreichten SMAPE-Werte im Vergleich zum global besten Klassifikationsraum erzielen konnte. Jedoch ist sein Abstand auf diesen Klassifikationsraum mit einer Differenz von nur 0,6 Prozentpunkten sehr gering. Auf Grund dessen kann der Baumansatz in diesem Szenario als Klassifikationsverfahren von gleichwertiger Güte zum besten Klassifikationsraum angesehen werden.

Es ist überraschend, dass ein Verfahren des maschinellen Lernens keine reale Verbesserung der Klassifikationsleistung erwirken konnte. Über die Gründe dieses Ergebnisses kann zum gegenwärtigen Zeitpunkt lediglich spekuliert werden, da eine detaillierte Untersuchung dieser den Rahmen dieser Arbeit übersteigen würde. Eine mögliche Ursache wäre eine ungünstige Datenlage, welche es dem Baumerzeugungsalgorithmus unmöglich macht, ein optimales Modell zu lernen. Weiterhin wäre es denkbar, dass die Baumerzeugung, trotz Anwendung von Pruning, ein zu spezifisches Modell anhand der Trainingsdaten ermittelt – ein Phänomen, welches in der Literatur als „Überanpassung“ (engl. overfitting; siehe [Bra07]) bekannt ist.

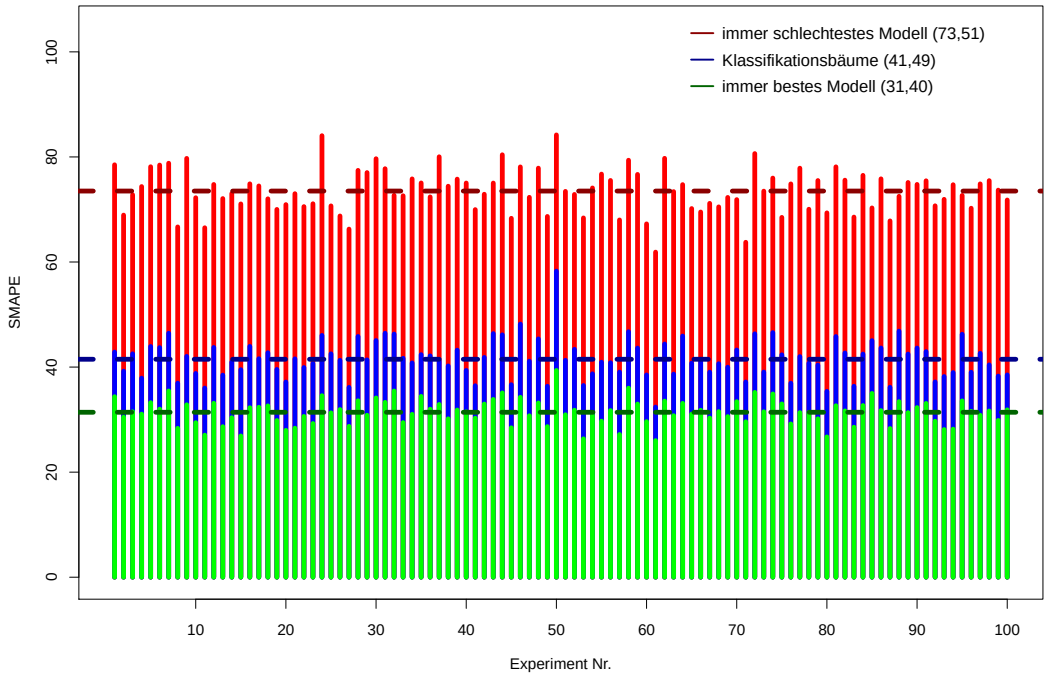


Abbildung 6.2: Erzielte SMAPE-Werte des Klassifikationsbaumes auf der gesamten Datenbasis

Analog zur Vorgehensweise in Kapitel 5 wurde auch für den Klassifikationsbaum eine domänenspezifische Klassifikatorbestimmung analysiert. Tabelle 6.2 zeigt die jeweiligen Ergebnisse je Domäne. Zur Bewertung der Veränderung der SMAPE-Werte gegenüber dem zuvor ermittelten besten domänenspezifischen Klassifikationsraum, wurden die entsprechenden Werte in diese Tabelle mit aufgenommen.

Ein direkter Vergleich offenbart, dass der Baumansatz lediglich in der Domäne *Energy* eine Verbesserung der Fehlerwerte erzielen konnte. Eine Betrachtung der Grafik in Abbildung 6.3 stützt diese Erkenntnis und illustriert darüber hinaus die hervorragende Annäherung der Empfehlung an die Kurve des Optimums. Nach erneuter Konsultation von Abbildung 5.2 könnte diese nahezu perfekte Modellklassifikation jedoch auf die Dominanz der $AR(p)$-Modelle in dieser Domäne zurückzuführen sein. Dadurch erscheint die Repräsentativität dieses Ergebnisses fraglich. Es wäre zu klären, ob eine Dominanz der $AR(p)$-Modelle im Energiesektor charakteristisch ist oder ob eine ungünstige Zusammensetzung der Testdatenbank das Ergebnis verrauscht. Diese Analyse ist jedoch nicht Teil dieser Arbeit.

In allen anderen Domänen erhöht sich der erzielte SMAPE-Wert teilweise drastisch. Dies ist am Beispiel der Domäne *Economics* zu erkennen. Die erreichte Verschlechterung der gemittelten Fehlerwerte ist hier mit rund 16,5 Prozentpunkten am höchsten.

Schließlich zeigt Tabelle 6.3 eine detaillierte Einordnung der je Domäne gemittelten SMAPE-Werte in den Grenzen des schlechtesten gemittelten Fehlerwertes[1] und des Optimums. Aus beiden Ta-

[1] Hierbei ist der gemittelte Fehlerwert gemeint, welcher entsteht, wenn stets das schlechteste zur Verfügung stehende Modell verwendet wird.

Domäne	bester domänen-spezifischer Raum	Klassifika-tionsbaum	Unterschied
Demographic	37,93	39,35	1,42
Economics	19,98	36,43	16,45
Energy	73,08	71,11	-1,97
Finance	38,84	43,27	4,43
Industry	32,10	32,66	0,56
Macro	17,62	18,55	0,93
Micro	46,05	47,16	1,11
Other	81,69	83,96	2,27

Tabelle 6.2: Ermittelte Unterschiede der SMAPE-Werte durch domänenspezifische Klassifikationsbäume

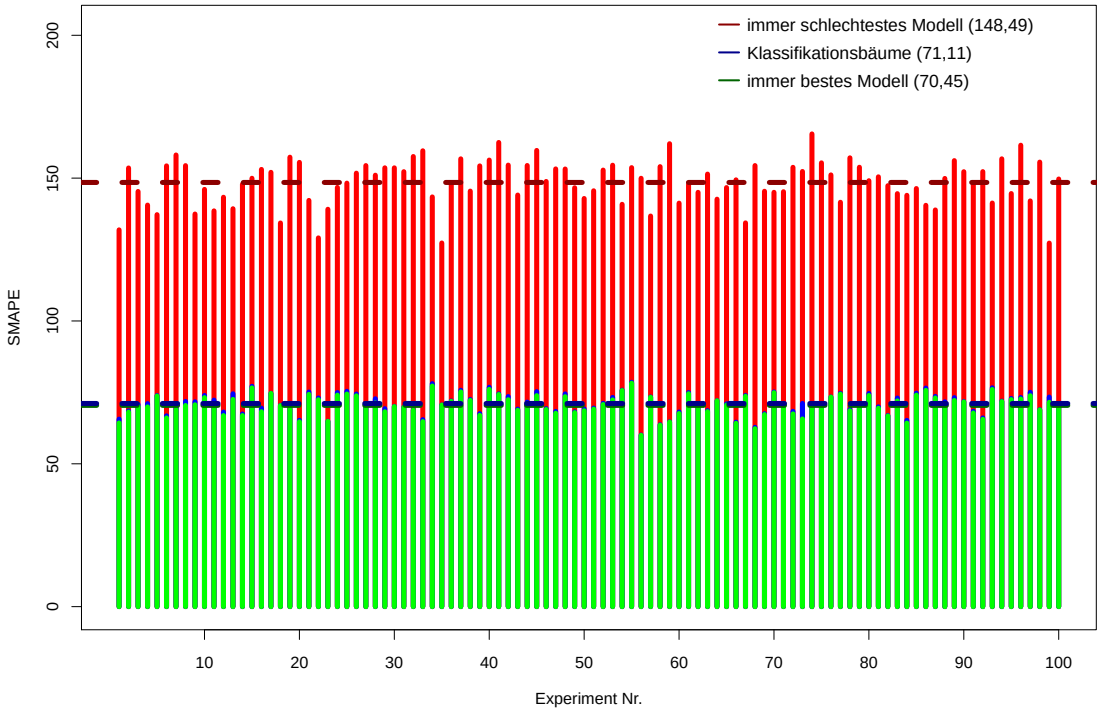


Abbildung 6.3: Erzielte SMAPE-Werte des Klassifikationsbaumes in der Domäne *Energy*

bellen geht hervor, dass der Baumansatz überwiegend hinreichend nah an den besten domänenspezifischen Klassifikationsraum heranreicht. Im Fall der Domäne *Energy* konnte sogar eine Verbesserung der erzielten Fehlerwerte erreicht werden. Aus diesem Grund wird die Verwendung eines Klassifikationsbaumes auf Zeitreihen aus dieser Domäne empfohlen.

Domäne	immer schlechtestes Modell	bester domänenspezifischer Raum	Klassifikationsbaum	immer bestes Modell (Optimum)
Datenbasis	73,51	41,43	41,49	31,40
Demographic	71,55	37,93	39,35	26,90
Economics	75,87	19,98	36,43	13,21
Energy	148,49	73,08	71,11	70,45
Finance	71,09	38,84	43,27	30,50
Industry	56,36	32,10	32,66	24,09
Macro	43,44	17,62	18,55	13,14
Micro	78,65	46,05	47,16	36,05
Other	125,53	81,69	83,96	66,17

Tabelle 6.3: Vergleich der ermittelten SMAPE-Werte der domänenspezifischen Klassifikationsräume und Klassifikationsbäume

6.2 VERBUNDKLASSIFIKATOR TP1NN

Die vorangegangene Analyse offenbarte, dass die Verwendung eines Klassifikationsbaumes in den meisten Domänen und kontextfrei eine ähnlich gute Klassifikationsgüte aufweist, wie der jeweils beste ermittelte 1NN-Klassifikationsraum. Bis auf die Domäne *Energy* lagen die erzielten Fehlerwerte jedoch über denen der besten Räume, weswegen diese zumeist zu bevorzugen sind.

Ein Makel an der 1NN-Klassifikation in den besten eigenschaftsbasierten Teilräumen ist jedoch der hohe Analyse- und damit Berechnungsaufwand. Die Untersuchung aller 4.095 Teilräume zur Identifikation der jeweils besten Räume ist sehr zeitaufwändig. Es stellt sich daher die Frage, ob es eine Möglichkeit gibt, die Anzahl der zu analysierenden Räume zu begrenzen.

Unter Berücksichtigung der Tatsache, dass die erzielten Fehlerwerte der baumbasierten Klassifikation teilweise sehr nah an den korrespondierenden besten Klassifikationsraum heranreichen, könnte eine Vorverarbeitung unter Verwendung eben dieser Bäume möglicherweise eine Eingrenzung leisten. Zunächst wäre bei einem solchen Vorgehen von Interesse, welche Eigenschaften tatsächlich in den verschiedenen Klassifikationsbäumen verwendet wurden. Sollte sich herausstellen, dass nicht alle Eigenschaften Beachtung finden, könnte dadurch eine Reduktion der zu betrachtenden Raummenge vollzogen werden. Auf Grund der Tatsache, dass der Baum somit die Analyse des besten 1NN-Klassifikationsraumes vorverarbeitet bzw. vorbereitet, ist dieser Klassifikator als „Tree Prepared 1-Nearest-Neighbor (TP1NN)“ zu bezeichnen.

In den Tabellen 6.4a bis 6.4i sind die Ergebnisse der Bestimmung der relativen Signifikanz aller verfügbaren Eigenschaften gezeigt[2]. Die Bewertung arbeitet dabei nach dem Prinzip eines Bestrafungssystems, wobei jeder Klassifikationsbaum sukzessive von der Wurzel bis zu den Blättern unter Berücksichtigung aller innenliegenden Knoten durchlaufen wird. Diese innenliegenden Knoten beinhalten laut Erzeugungsalgorithmus die jeweiligen Eigenschaften, welche zur Tren-

[2]Es ist zu beachten, dass zusätzlich zu den 12 Eigenschaften aus P_{rel} bei der Baumerstellung das kategoriale Attribut $domain(T)$ in die Betrachtung einbezogen wurde. Es bezeichnet die Datendomäne einer Zeitreihe T.

Eigenschaft	Strafwert
$f(T)$	100,00
$sos(T)$	242,00
$sot(T)$	411,50
$var(T)$	421,00
$se(T)$	822,00
$domain(T)$	1.296,00
$uq(T)$	1.464,00
$k(T)$	1.489,00
$lq(T)$	1.500,00
$foac(T)$	1.500,00
$\lambda(T)$	1.500,00
$median(T)$	1.500,00
$sk(T)$	1.500,00

(a) Datenbasis

Eigenschaft	Strafwert
$sot(T)$	350,75
$sos(T)$	408,75
$uq(T)$	502,75
$var(T)$	655,69
$median(T)$	786,00
$se(T)$	792,75
$\lambda(T)$	821,00
$foac(T)$	850,75
$lq(T)$	882,50
$sk(T)$	1.029,50
$k(T)$	1.044,50
$f(T)$	1.433,00
$domain(T)$	1.500,00

(b) Demographic

Eigenschaft	Strafwert
$foac(T)$	260,00
$var(T)$	352,00
$sos(T)$	870,50
$uq(T)$	966,00
$median(T)$	1.201,00
$sk(T)$	1.231,00
$se(T)$	1.244,50
$k(T)$	1.377,00
$\lambda(T)$	1.379,00
$sot(T)$	1.427,00
$lq(T)$	1.440,00
$f(T)$	1.500,00
$domain(T)$	1.500,00

(c) Economics

Eigenschaft	Strafwert
$median(T)$	286,00
$se(T)$	300,25
$var(T)$	483,00
$sot(T)$	1.339,00
$k(T)$	1.344,00
$sk(T)$	1.381,00
$uq(T)$	1.448,00
$foac(T)$	1.462,00
$lq(T)$	1.463,00
$f(T)$	1.486,00
$sos(T)$	1.500,00
$\lambda(T)$	1.500,00
$domain(T)$	1.500,00

(d) Energy

Eigenschaft	Strafwert
$sos(T)$	436,00
$foac(T)$	557,13
$\lambda(T)$	704,00
$lq(T)$	709,44
$median(T)$	715,00
$var(T)$	744,50
$uq(T)$	765,50
$sot(T)$	818,00
$se(T)$	875,50
$k(T)$	1.086,25
$f(T)$	1.290,00
$sk(T)$	1.306,50
$domain(T)$	1.500,00

(e) Finance

Eigenschaft	Strafwert
$sos(T)$	225,25
$uq(T)$	350,50
$se(T)$	484,00
$sot(T)$	631,50
$foac(T)$	639,00
$\lambda(T)$	658,75
$var(T)$	760,25
$lq(T)$	786,50
$sk(T)$	872,50
$k(T)$	956,00
$median(T)$	989,25
$f(T)$	1.500,00
$domain(T)$	1.500,00

(f) Industry

Eigenschaft	Strafwert
$var(T)$	404,63
$sot(T)$	565,25
$foac(T)$	576,25
$uq(T)$	608,50
$sk(T)$	781,25
$sos(T)$	857,13
$\lambda(T)$	861,00
$lq(T)$	982,63
$k(T)$	1.012,00
$median(T)$	1.168,50
$se(T)$	1.176,00
$f(T)$	1.393,00
$domain(T)$	1.500,00

(g) Macro

Eigenschaft	Strafwert
$sos(T)$	192,50
$se(T)$	333,00
$f(T)$	368,00
$foac(T)$	604,00
$uq(T)$	611,50
$sk(T)$	889,50
$sot(T)$	1.339,00
$lq(T)$	1.351,00
$\lambda(T)$	1.387,00
$median(T)$	1.402,00
$k(T)$	1.436,00
$var(T)$	1.491,00
$domain(T)$	1.500,00

(h) Micro

Eigenschaft	Strafwert
$var(T)$	369,00
$sos(T)$	378,25
$se(T)$	482,88
$k(T)$	544,75
$\lambda(T)$	817,63
$sk(T)$	931,50
$sot(T)$	951,75
$foac(T)$	1.049,50
$lq(T)$	1.122,13
$uq(T)$	1.227,00
$median(T)$	1.250,00
$f(T)$	1.486,00
$domain(T)$	1.500,00

(i) Other

Tabelle 6.4: Ermittelte Strafwerte je Eigenschaft je Domäne

nung der Datenmenge dienen. Da in jedem der rekursiven Teilschritte zur Baumerstellung jeweils das Attribut mit dem höchsten IG ausgewählt wird, trennt diese Eigenschaft die Daten besonders aussagekräftig und hat somit eine hohe Signifikanz. Daraus folgt, dass je tiefer die Ebene des Auftretens einer Eigenschaft ist, desto weniger signifikant sich die darin vorkommende Eigenschaft charakterisiert. Aus diesem Grund wird im Folgenden eine Eigenschaft $p \in P'_{rel}$ mit $P'_{rel} = P_{rel} \cup domain(T)$ mittels der Funktion $punishment\colon P'_{rel} \to \mathbb{R}$ durch

$$punishment(p) = \frac{l(p)}{n(p)}$$

mit einer Strafe belegt, falls p in diesem Baum in Ebene $l(p)$ zum $n(p)$-ten Mal ausgewählt wurde. Die vergebenen Strafen relativieren sich bei jedem erneuten Auftreten einer Eigenschaft im Baum, damit vorangegangene, höhergelagerte Vorkommen durch weitere Vorkommen in späteren Ebenen nicht so stark kompromittiert werden. Hat p jedoch kein einziges Auftreten im Baum, bestimmt sich seine Strafe als $punishment(p) = 15$. In Anbetracht einer Anzahl von 100 Experimenten – und damit 100 Klassifikationsbäumen – würde eine Eigenschaft, welche nie in einem Klassifikationsbaum vorgekommen ist, die Strafbewertung 1.500 aufweisen.

Eine Betrachtung der Tabellen 6.4a bis 6.4i offenbart, dass in einigen Analysen tatsächlich nicht die gesamte Menge P'_{rel} in den Klassifikationsbäumen verwendet wurde. Es entspricht den Erwartungen, dass die kontinuierliche Strafbewertung des Attributs $domain(T)$ in allen Analysen, außer der über der gesamten Datenbasis, einen Wert von 1.500 aufweist. Da bei einer domänenspezifischen Analyse die Ausprägungen dieser Eigenschaft immer denselben Wert annehmen, ist sie für eine Klassifikation irrelevant.

Die deutlichste Reduktion der Eigenschaftsmenge zeigt sich bei der globalen Analyse über der Datenbasis. P'_{rel} wurde durch sie um fünf Eigenschaften verkleinert, was einer Reduktion um rund 38% entspricht. Abzüglich der Vorkommen des nicht-numerischen Attributs $domain(T)$ verbleiben sieben Eigenschaften in der Menge, was die Anzahl der zu untersuchenden Teilräume auf 127 absinken lässt. Somit hätte sich durch den Vorverarbeitungsschritt die Komplexität der Hauptanalyse drastisch verringert.

Es zeigt sich leider, dass diese Vorgehensweise mit Kompromissen in der Genauigkeit einhergeht, da die baumgestützte Vorverarbeitung im genannten Fall die Eigenschaft $foac(T)$ als irrelevant charakterisiert hätte. Tatsächlich ist dieses Merkmal jedoch eine Dimension des besten Teilraums, welcher in Kapitel 5 ermittelt wurde. Dies ist leider kein Einzelfall, da sich ein ähnliches Verhalten im Fall der Domäne *Industry* zeigt, wobei exakt die Eigenschaft als irrelevant befunden wird, welche nach vollständiger Teilraumanalyse als alleiniges Vergleichscharakteristikum isoliert wurde.

In anderen Domänen, wie beispielsweise *Demographic* und *Finance*, kann der Vorverarbeitungsschritt sogar keines der numerischen Attribute als irrelevant einstufen. Somit würde der Klassifikator in diesen Fällen durch ein Versagen der Vorverarbeitung zum klassischen 1NN-Klassifikator degenerieren. Die Gesamtanalyse wäre hier auf Grund der geleisteten Arbeit zur Vorverarbeitung sogar komplexer und aufwändiger.

Da die baumbasierte Vorverarbeitung zur Reduktion des Suchaufwandes des besten 1NN-Klassifikationsraumes nicht geeignet scheint, stellt sich die Frage ob es möglich ist, mit Hilfe der

Vorverarbeitung den besten Klassifikationsraum hinreichend zu approximieren. Um den Berechnungsaufwand so klein wie möglich zu halten, wird hierbei die anschließende Teilraumsuche ausgelassen und stattdessen der Teilraum unter Verwendung aller durch die Vorverarbeitung herausgefilterten Eigenschaften als 1NN-Klassifikationsraum genutzt. Tabelle 6.5 zeigt die jeweiligen Teilräume, welche sich auf Basis der baumbasierten Vorverarbeitung ergeben. Etwaige Vorkommen der Eigenschaft $domain(T)$ wurden dabei ignoriert, da sich dieses Merkmal kategorial ausprägt und somit keine Grundlage zur Einbeziehung in die euklidische Distanz als Ähnlichkeitsmaß bietet.

Domäne	**Teilraum**
Datenbasis	$f(T), se(T), sos(T), sot(T), var(T)$
Demographic	$foac(T), k(T), \lambda(T), lq(T), \widetilde{T}, se(T), sk(T), sos(T), sot(T), uq(T), var(T)$
Economics	$foac(T), var(T)$
Energy	$\widetilde{T}, se(T), var(T)$
Finance	$foac(T), \lambda(T), lq(T), \widetilde{T}, se(T), sos(T), sot(T), uq(T), var(T)$
Industry	$foac(T), k(T), \lambda(T), lq(T), \widetilde{T}, se(T), sk(T), sos(T), sot(T), uq(T), var(T)$
Macro	$foac(T), k(T), \lambda(T), lq(T), \widetilde{T}, se(T), sk(T), sos(T), sot(T), uq(T), var(T)$
Micro	$f(T), foac(T), se(T), sk(T), sos(T), uq(T)$
Other	$k(T), se(T), sos(T), var(T)$

Tabelle 6.5: Durch baumbasierte Vorverarbeitung ermittelte Teilräume

Die baumbasierte Vorverarbeitung, welche zu den Resultaten in Tabelle 6.5 führte, erfuhr eine leichte Modifikation, wodurch die jeweilige Raumdimensionalität minimiert wurde. Die Tabellen 6.6a bis 6.6i demonstrieren die verwendete modifizierte Vorverarbeitung. Jede der Tabellen beinhaltet eine farbig markierte Zeile. Der jeweilige Strafwert in dieser Zeile verzeichnet den größten Anstieg zu dem Wert der vorhergehenden Eigenschaft. Dadurch zeichnet sich an dieser Stelle erkennbar die Abtrennung der Eigenschaften ab, welche am häufigsten in den verschiedenen Baumklassifikatoren ausgewählt wurden.

Schließlich dokumentiert Tabelle 6.7 den Vergleich des TP1NN-Klassifikators mit den vorher betrachteten Klassifikatoren. Darin zeigt eine fett gedruckte Zahl, welcher Ansatz in einer bestimmten Domäne den geringsten Fehler erzielt hat und somit in dieser Domäne verwendet werden sollte. Unterstrichene Zahlen markieren, ob der klassische Baumklassifikator oder der TP1NN-Klassifikator bessere SMAPE-Werte in einer Domäne erzielen konnte.

Es zeigt sich, dass der TP1NN-Klassifikator in keiner Domäne den besten zu verwendenden Ansatz darstellt. Somit bleibt das Ergebnis der Evaluation nach Abschnitt 6.1 bestehen, wodurch der 1NN-Klassifikator, mit Ausnahme der Domäne *Energy*, stets verwendet werden sollte. Im Vergleich zum klassischen Baumklassifikator kann sich der TP1NN-Ansatz auf der gesamten Datenbasis nicht durchsetzen, erreicht mit einer Abweichung von 0,39 Prozentpunkten jedoch ein vergleichbar gutes Ergebnis. Hinsichtlich der domänenspezifischen Analysen herrscht ein Gleichgewicht zwischen der Anzahl der Vorkommen, bei denen der Baumklassifikator die bes-

Eigenschaft	Strafwert
$f(T)$	100,00
$sos(T)$	242,00
$sot(T)$	411,50
$var(T)$	421,00
$se(T)$	822,00
$domain(T)$	1.296,00
$uq(T)$	1.464,00
$k(T)$	1.489,00
$lq(T)$	1.500,00
$foac(T)$	1.500,00
$\lambda(T)$	1.500,00
$median(T)$	1.500,00
$sk(T)$	1.500,00

(a) Datenbasis

Eigenschaft	Strafwert
$sot(T)$	350,75
$sos(T)$	408,75
$uq(T)$	502,75
$var(T)$	655,69
$median(T)$	786,00
$se(T)$	792,75
$\lambda(T)$	821,00
$foac(T)$	850,75
$lq(T)$	882,50
$sk(T)$	1.029,50
$k(T)$	1.044,50
$f(T)$	1.433,00
$domain(T)$	1.500,00

(b) Demographic

Eigenschaft	Strafwert
$foac(T)$	260,00
$var(T)$	352,00
$sos(T)$	870,50
$uq(T)$	966,00
$median(T)$	1.201,00
$sk(T)$	1.231,00
$se(T)$	1.244,50
$k(T)$	1.377,00
$\lambda(T)$	1.379,00
$sot(T)$	1.427,00
$lq(T)$	1.440,00
$f(T)$	1.500,00
$domain(T)$	1.500,00

(c) Economics

Eigenschaft	Strafwert
$median(T)$	286,00
$se(T)$	300,25
$var(T)$	483,00
$sot(T)$	1.339,00
$k(T)$	1.344,00
$sk(T)$	1.381,00
$uq(T)$	1.448,00
$foac(T)$	1.462,00
$lq(T)$	1.463,00
$f(T)$	1.486,00
$sos(T)$	1.500,00
$\lambda(T)$	1.500,00
$domain(T)$	1.500,00

(d) Energy

Eigenschaft	Strafwert
$sos(T)$	436,00
$foac(T)$	557,13
$\lambda(T)$	704,00
$lq(T)$	709,44
$median(T)$	715,00
$var(T)$	744,50
$uq(T)$	765,50
$sot(T)$	818,00
$se(T)$	875,50
$k(T)$	1.086,25
$f(T)$	1.290,00
$sk(T)$	1.306,50
$domain(T)$	1.500,00

(e) Finance

Eigenschaft	Strafwert
$sos(T)$	225,25
$uq(T)$	350,50
$se(T)$	484,00
$sot(T)$	631,50
$foac(T)$	639,00
$\lambda(T)$	658,75
$var(T)$	760,25
$lq(T)$	786,50
$sk(T)$	872,50
$k(T)$	956,00
$median(T)$	989,25
$f(T)$	1.500,00
$domain(T)$	1.500,00

(f) Industry

Eigenschaft	Strafwert
$var(T)$	404,63
$sot(T)$	565,25
$foac(T)$	576,25
$uq(T)$	608,50
$sk(T)$	781,25
$sos(T)$	857,13
$\lambda(T)$	861,00
$lq(T)$	982,63
$k(T)$	1.012,00
$median(T)$	1.168,50
$se(T)$	1.176,00
$f(T)$	1.393,00
$domain(T)$	1.500,00

(g) Macro

Eigenschaft	Strafwert
$sos(T)$	192,50
$se(T)$	333,00
$f(T)$	368,00
$foac(T)$	604,00
$uq(T)$	611,50
$sk(T)$	889,50
$sot(T)$	1.339,00
$lq(T)$	1.351,00
$\lambda(T)$	1.387,00
$median(T)$	1.402,00
$k(T)$	1.436,00
$var(T)$	1.491,00
$domain(T)$	1.500,00

(h) Micro

Eigenschaft	Strafwert
$var(T)$	369,00
$sos(T)$	378,25
$se(T)$	482,88
$k(T)$	544,75
$\lambda(T)$	817,63
$sk(T)$	931,50
$sot(T)$	951,75
$foac(T)$	1.049,50
$lq(T)$	1.122,13
$uq(T)$	1.227,00
$median(T)$	1.250,00
$f(T)$	1.486,00
$domain(T)$	1.500,00

(i) Other

Tabelle 6.6: Markierung der Abtrennungseigenschaft zur Identifikation der Menge häufigster Teilungseigenschaften in den Klassifikationsbäumen

Domäne	immer schlechtestes Modell	bester domänenspezifischer Raum	Baumklassifikator	TP1NN	immer bestes Modell (Optimum)
Datenbasis	73,51	**41,43**	41,49	41,88	31,40
Demographic	71,55	**37,93**	39,35	39,53	26,90
Economics	75,87	**19,98**	36,43	26,38	13,21
Energy	148,49	73,08	**71,11**	74,65	70,45
Finance	71,09	**38,84**	43,27	39,37	30,50
Industry	56,36	**32,10**	32,66	33,95	24,09
Macro	43,44	**17,62**	18,55	18,82	13,14
Micro	78,65	**46,05**	47,16	46,93	36,05
Other	125,53	**81,69**	83,96	83,61	66,17

Tabelle 6.7: Vergleich der ermittelten SMAPE-Werte der besten Klassifikationsräume, des Baumklassifikators und des TP1NN-Ansatzes

seren SMAPE-Werte erzielt, und derer, bei denen TP1NN das bessere Ergebnis produziert. Die jeweils erreichten Verbesserungen sind mit Ausnahme der Domäne *Economics* nicht sehr hoch. Im genannten Ausnahmefall kann die Klassifikationsgüte jedoch mit einer Verbesserung um rund 10 Prozentpunkte signifikant gesteigert werden. Mit einer Gesamtabweichung von nur rund 5 Prozentpunkten auf das Optimum, ließe sich TP1NN im Bezug auf den Analyse- und Berechnungsaufwand hier durchaus als Alternative zum 1NN-Ansatz betrachten.

Allgemein lässt sich jedoch sagen, dass TP1NN keine konkurrenzfähige Alternative zum klassischen Baumklassifikator darstellt. Dies ist dadurch zu begründen, dass eine Verwendung klassischer Baumklassifikatoren in der Mehrheit der Analysen bessere Ergebnisse erzielt hat als TP1NN. Darüber hinaus ist bei letztgenanntem Ansatz der Berechnungsaufwand höher als bei klassischen Baumklassifikatoren.

6.3 KÜNSTLICHES NEURONALES NETZ

Als letzter Klassifikator soll ein sogenanntes Künstliches Neuronales Netz (engl. „Artificial Neural Network (ANN)“) (siehe [BH00], [Has95]) evaluiert werden. ANNs stammen aus einem gemeinsamen Teilbereich sowohl der künstlichen Intelligenz als auch der Neurobiologie und gehen auf Arbeiten beginnend in den frühen 1940er Jahren zurück (siehe [MP43], [Heb49], [Ros58], [WH60], [MP88]). Sie können auf Grund ihrer Struktur und Funktionsweise dem Bereich des sogenannten *Deep Learning* zugeordnet werden.

Deep Learning (siehe [GBC16], [Sch15], [DY13]) ist ein Teil das maschinellen Lernens. Dabei kommen Algorithmen zum Einsatz, welche auf einem Graphen basieren, der eine Vielzahl von zwischengelagerten Verarbeitungsschichten zwischen den dedizierten Eingabe- und Ausgabeschichten aufweist. Ein ANN besitzt die Struktur eines solchen Graphen, da es aus einer Vielzahl soge-

nannter Neuronen besteht, welche grundlegend Verarbeitungsknoten in einem hochkomplexen Netzwerk darstellen. Die Knoten sind über Kanten – den Verbindungskanälen – miteinander verbunden. Auf diese Art und Weise versuchen ANNs, die Nervenzellenverbindungen in Gehirn und Rückenmark eines biologischen Organismus zu simulieren und dadurch deren Funktionsweise künstlich nachzubilden.

Wie im Fall der CARTs gibt es auch für die Verwendung von ANNs mittlerweile verfügbare Implementierungen, welche komfortabel in Analysewerkzeugen eingebunden und benutzt werden können. Die folgende Evaluation basiert auf der Verwendung des Deep Learning Pakets für die Statistiksoftware *R* names *h2o* (siehe [AKMHt16]). Dieses Paket beinhaltet die Funktion *h2o.deeplearning*, mit welcher ein Modell in Form eines ANNs aus Trainingsdaten erlernt werden kann. Anschließend lassen sich Testdaten mit diesem Modell über die Funktion *h2o.predict* ihren erwarteten Klassifikationslabels zuführen.

Bei der Benutzung der Lernfunktion zur Modellerzeugung wurde bewusst darauf geachtet, die Standardparametereinstellungen weitestgehend beizubehalten. Eine Optimierung dieser Parameter sollte zwar höchstwahrscheinlich in einer Erhöhung der Performanz des Ansatzes resultieren, jedoch müssten diese Parameter im laufenden Betrieb geschätzt oder optimiert werden. Im Zweifelsfall würde die Festlegung dieser Parameter in die Verantwortlichkeit des Benutzers fallen, was allerdings dem gewünschten Ziel einer systematischen Unterstützungsleistung abträglich ist. Ob und inwieweit es möglich ist, die Parameter systematisch schätzen und optimieren zu lassen, ist nicht Teil dieser Arbeit.

Domäne	**bester domänenspezifischer Raum**	**Neuronales Netz**	**Abweichung**
Datenbasis	41,43	42,45	1,02
Demographic	37,93	40,43	2,50
Economics	19,98	37,81	17,83
Energy	73,08	73,89	0,81
Finance	38,84	47,91	9,07
Industry	32,10	34,14	2,04
Macro	17,62	18,28	0,66
Micro	46,05	48,41	2,36
Other	81,69	85,71	4,02

Tabelle 6.8: Ermittelte Abweichungen der SMAPE-Werte durch ANNs

Tabelle 6.8 zeigt das Ergebnis der Auswertung aller insgesamt 900 Experimente unter Verwendung eines ANNs als Klassifikator. Es ist zu erkennen, dass sich dieser Ansatz leider in keiner spezifischen Domäne und auch nicht auf der gesamten Datenbasis als bester Klassifikator herausstellt. In den meisten Fällen erbringt das modellierte ANN jedoch ähnlich gute Klassifikationsleistungen wie der jeweilige beste 1NN-Klassifikationsraum in einer Domäne. Es gibt jedoch auch Fälle, bei denen der gemittelte Fehlerwert eine signifikante Abweichung zum besten An-

satz der Domäne aufweist. Das Extrembeispiel bildet hierbei die Domäne *Economics*, in welcher der ANN-Ansatz um fast 18 Prozentpunkte schlechtere gemittelte Fehlerwerte erzielt als der beste Klassifikationsraum. Abbildung 6.4 veranschaulicht diesen Sachverhalt deutlich. In der Domäne *Macro* hingegen stellt sich ANN in der Gesamtrangfolge der verwendeten Klassifikatoren als zweitbester verwendeter Ansatz heraus. Dies wird durch Tabelle 6.9 ersichtlich, in welcher die Einordnung des ANN-Ansatzes in die Menge der anderen untersuchten Klassifikatoren erfolgt.

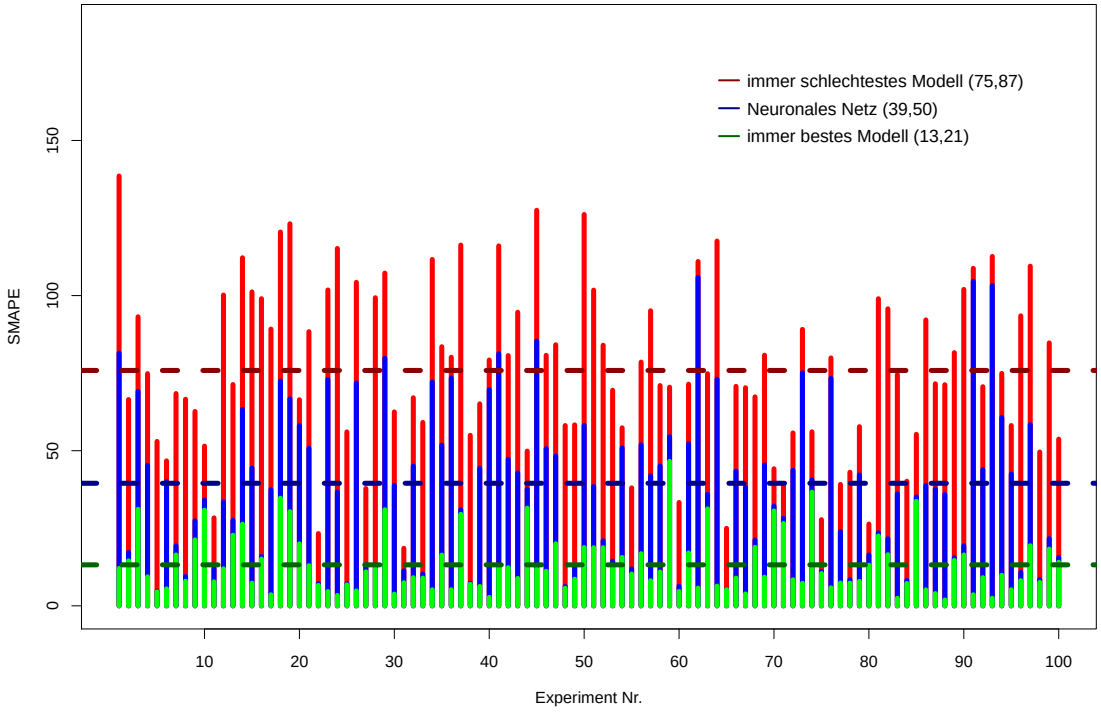


Abbildung 6.4: Erzielte SMAPE-Werte des Neuronalen Netzes in der Domäne *Economics*

Insgesamt kann ANN als Klassifikator in dieser Evaluation nicht überzeugen. Die Komplexität der Erstellung solcher Modelle erscheint auf Grund der Vielzahl von Parametrierungen im Hinblick auf eine einfache Unterstützungsmethodik als nicht zielführend. Mit den Standardparametereinstellungen schwankt die Leistung des Ansatzes domänenabhängig teilweise sehr stark. Darüber hinaus kann der verwendete ANN-Klassifikator zumeist keinen der vorher untersuchten Ansätze im gemittelten Fehler verbessern.

6.4 EVALUATION GEGEN RANDOM-WALK-METHODIKEN

Als eine weitere grundlegende Technik, gegen welche es sich abzugrenzen gilt, existieren sogenannte Random-Walk-Methodiken (siehe [Spi76]). Diese Ansätze können grundlegend in den Bereich der naiven und gierigen Verfahren eingeordnet werden, da sie ohne größere Untersuchungen oder Vorüberlegungen schnelle Ergebnisse erzielen sollen. Auch für die Unterstützung der Modellauswahl können solche randomisierte Verfahren konstruiert werden.

Domäne	immer schlechtestes Modell	bester domänenspezifischer Raum	Baumklassifikator	Neuronales Netz	immer bestes Modell (Optimum)
Datenbasis	73,51	**41,43**	41,49	42,45	31,40
Demographic	71,55	**37,93**	39,35	40,43	26,90
Economics	75,87	**19,98**	36,43	37,81	13,21
Energy	148,49	73,08	**71,11**	73,89	70,45
Finance	71,09	**38,84**	43,27	47,91	30,50
Industry	56,36	**32,10**	32,66	34,14	24,09
Macro	43,44	**17,62**	18,55	18,28	13,14
Micro	78,65	**46,05**	47,16	48,41	36,05
Other	125,53	**81,69**	83,96	85,71	66,17

Tabelle 6.9: Vergleich der ermittelten SMAPE-Werte der besten Klassifikationsräume, des Baumklassifikators und des Neuronalen Netzes

Im Folgenden wird ein Messexperiment beschrieben und durchgeführt, mit welchem die Performanz zweier solcher Random-Walk-Verfahren evaluiert werden soll. Gegenstand dieses Experiments sind dabei die Methoden „strikte Wiederverwendung ein und desselben Vorhersagemodells“ (I) und „strikte Verwendung des am häufigsten auftretenden Modells“ (II). Aus Gründen der Vereinfachung, wird (I) fortan durch „Forced Single Model Reuse (FSMR)“ und (II) mittels „Forced Most common Model Reuse (FMMR)“, unter Verwendung der jeweiligen Abkürzung, bezeichnet. Der Vergleich dieser beiden Ansätze erfolgt gegen die Verwendung der bisher ermittelten besten domänenspezifischen Klassifikatoren. Das Experiment soll Klarheit darüber bringen, ob die durch Analysen hervorgebrachten Klassifikatoren überhaupt einen Mehrwert zu simplen und naiven Techniken erbringen.

Als Basis dienen erneut die 100 Stichproben, wie sie in Kapitel 5 eingeführt wurden. Da nach Abbildung 5.1 das Datenmodell der Evaluationsdatenbank die Tabelle *Error* und die Sicht *Casebase* enthält, ist die Bestimmung der gemittelten Fehler für die Methodiken FSMR und FMMR durch wenige Berechnungen schnell ermittelt. Die korrespondierenden Werte des jeweils besten domänenspezifischen Klassifikators können aus den vorangegangenen Tabellen übernommen werden. Infolgedessen zeigt Tabelle 6.10 die Ergebnisse dieses Experiments.

In dieser Ergebnistabelle ist der jeweils beste domänenspezifische Ansatz durch eine fett gedruckte Zahl mit grauer Zellenmarkierung versehen. Es zeigt sich in der Tat, dass in einigen Domänen die Ansätze FSMR und FMMR tatsächlich die SMAPE-Werte des besten analysierten domänenspezifischen Klassifikators verbessern können. Allerdings ist dies kein Indiz auf eine gute oder gar vertretbare Leistung der Techniken. Wird die Tabelle pro Ansatz gelesen, ergibt sich ein deutlich anderes Bild, in welchem die naiven Methodiken FSMR und FMMR je in nur höchstens zwei Domänen diese Verbesserung erzielen. Darüber hinaus sind die erreichten Verbesserungen jeweils so minimal, dass sie kaum die ihnen zugrunde liegende mangelnde Flexibilität rechtfertigen. Des Weiteren ist zu beachten, dass der Ansatz FMMR in fast allen Fällen zu einer FSMR-Methodik, bezogen auf das Modell „Auto-Arima“, degeneriert. Dieser Umstand ist

Domäne	immer schlechtestes Modell	immer Empfehlung	immer häufigstes Label	immer AR(1)	immer AR(2)	immer AR(3)	immer AR(4)	immer AUTO-ARIMA	immer HOLT-WINTERS	immer ETS	immer NAIVE	immer bestes Modell (Optimum)
Datenbasis	73,51	**41,43**	43,48	56,00	54,30	53,99	54,20	43,48	52,76	43,07	45,15	31,40
Demographic	71,55	37,93	40,93	52,82	53,56	54,69	55,83	40,93	48,13	**37,74**	38,23	26,90
Economics	75,87	19,98	54,94	73,62	80,23	78,19	79,42	68,68	85,11	19,32	**19,28**	13,21
Energy	148,49	**71,11**	103,50	89,17	89,48	89,49	89,51	103,50	142,02	105,72	84,99	70,45
Finance	71,09	**38,84**	42,11	51,68	48,59	48,33	48,42	42,11	53,05	43,22	43,61	30,50
Industry	56,36	32,10	**32,01**	44,22	41,56	40,49	40,47	**32,01**	38,53	32,43	35,63	24,09
Macro	43,44	**17,62**	19,42	38,18	35,19	35,03	35,08	19,42	20,60	19,96	21,43	13,14
Micro	78,65	46,05	**45,39**	59,72	58,41	57,73	57,79	**45,39**	56,61	46,04	51,86	36,05
Other	125,53	81,69	84,74	112,43	108,64	108,63	108,53	84,74	89,53	**80,50**	99,81	66,17

Tabelle 6.10: Ermittelte SMAPE-Werte der Ansätze FSMR und FMMR

jedoch wenig überraschend, wenn Abbildung 5.2a erneut betrachtet wird, in welcher zu erkennen ist, dass das Modell „Auto-Arima“ die Klassifikationen der Datenbasis dominiert.

Im Gegensatz zu den beiden Random-Walk-Methoden, erreicht der beste analysierte Klassifikator allerdings in vier Domänen die besten SMAPE-Werte. Darüber hinaus sind die jeweils besten ermittelten Klassifikatoren durch Analysen entstanden, welche stets gewissen Einschränkungen unterliegen. Daher ist es nachvollziehbar, dass die ermittelten Klassifikatoren in einigen Fällen marginal schlechtere Ergebnisse erzielten als die naiven Ansätze. Im Hinblick auf die vormals erwähnte mangelnde Flexibilität von FSMR und FMMR und kaum merkliche Unterschiede in der Klassifikationsgüte, erweisen sich diese beiden Random-Walk-Methoden als ungeeignet und werden daher nicht weiter betrachtet.

6.5 EVALUATION GEGEN ROHWERTBASIERTE EUKLIDISCHE DISTANZ

Die vorangegangenen Abschnitte beinhalteten die Evaluation verschiedenster Klassifikatoren, welche auf Basis der in Kapitel 3 herausgestellten Eigenschaften eine Ähnlichkeitssuche realisiert haben. Aus diesen Analysen ging der 1-Nearest-Neighbor Klassifikator zumeist als bester Ansatz hervor. In der Domäne *Energy* konnte sich der Klassifikationsbaum durchsetzen.

Zum gegenwärtigen Zeitpunkt ist noch unklar, wie sich die untersuchten Ansätze im Vergleich zur Verwendung eines klassischen Distanzmaßes auf den Rohwerten der Zeitreihen verhalten. Sollten sich die Aussagen in [WMD+13] bewahrheiten, könnte sich ergeben, dass keiner der eigenschaftsbasierten Ansätze die Klassifikationsgüte eines solchen Maßes verbessern kann. Zu diesem Zweck werden die in Abschnitt 5.1.3 erläuterten Experimente erneut mit zwei Varianten der euklidischen Distanz durchgeführt. Variante (I) bezeichnet dabei die herkömmliche euklidische Distanz mit der damit einhergehenden Vergleichsmengenreduktion. Um diese zu umgehen, zeigt Variante (II) ein leicht modifiziertes Verhalten. Sie vergleicht zwar alle Zeitreihen miteinander, betrachtet dabei aber je Zeitreihe lediglich den letzten längsten gemeinsamen Teil der Beobachtungen, gemessen an der kürzesten Zeitreihe in der Gesamtvergleichsmenge. Auf Grund dieser Funktionsweisen wird Variante (I) als „Euclidean distance on Same Length Series (ESLS)“ und Variante (II) als „Euclidean distance on Last Common Series Part (ELCSP)“ bezeichnet.

Die Auswertung der Experimente für ESLS ist in Tabelle 6.11 dokumentiert. Es lässt sich erkennen, dass ESLS in der Mehrheit der Analysen die Klassifikationsgüte im Vergleich zum besten eigenschaftsbasierten Ansatz verbessern konnte. Die größte Verbesserung um rund 2 Prozentpunkte, wurde dabei auf den Experimenten über der gesamten Datenbasis erzielt. In der Domäne *Economics* zeichnet sich hingegen die größte Verschlechterung der Klassifikationsgüte ab, mit einer Abweichung um mehr als 11 Prozentpunkte. Die verbleibenden Domänen, in welchen ESLS schlechter operiert als der beste eigenschaftsbasierte Ansatz, zeigen mit Abweichungen im Bereich von rund 2 bis rund 6 Prozentpunkten ähnlich deutliche, wenn auch schwächere Verschlechterungen in der Performanz.

Zusätzlich zu diesen Ergebnissen in der Performanzbegutachtung, sind auch die Ergebnisse zu beachten, welche durch die fünfte Spalte von Tabelle 6.11 gezeigt werden. Darin ist die Anzahl

Domäne	bester eigenschaftsbasierter Ansatz	ESLS	Abweichung	Anzahl leerer Vergleichsmengen
Datenbasis	41,43	39,46	-1,97	80 ($\hat{=}0,008\%$)
Demographic	37,93	36,61	-1,32	104 ($\hat{=}0,025\%$)
Economics	19,98	31,26	11,28	37 ($\hat{=}0,062\%$)
Energy	71,11	73,03	1.92	0 ($\hat{=}0,000\%$)
Finance	38,84	44,60	5,76	366 ($\hat{=}0,118\%$)
Industry	32,10	31,39	-0,71	150 ($\hat{=}0,029\%$)
Macro	17,62	16,42	-1,20	61 ($\hat{=}0,008\%$)
Micro	46,05	44,49	-1,56	20 ($\hat{=}0,003\%$)
Other	81,69	86,07	4,38	13 ($\hat{=}0,007\%$)

Tabelle 6.11: Ermittelte Abweichungen der SMAPE-Werte durch ESLS

der Vorkommen vermerkt, in welchen der ESLS-Ansatz keine ähnlichste Zeitreihe zu einer neuen Zeitreihe finden konnte. Dies lag in den jeweiligen Fällen daran, dass die neue Zeitreihe in ihrer Länge einzigartig unter den Reihen der Gesamtvergleichsmenge war. Die Forderung der Längengleichheit führte somit dazu, dass keine Vergleichsmenge aus der Datenbasis extrahiert werden konnte. In einem operativen System wäre in diesen Fällen die Unterstützung fehlgeschlagen. Auf Grund der Vielzahl der Experimente, zeigen sich solche Empfehlungsversagen jedoch domänenabhängig in nur 0,003% bis 0,118% aller Klassifikationsversuche.

In Tabelle 6.12 ist die Auswertung der Experimente durch die Variante ELCSP aufgeführt. Die letzte Spalte zeigt, wie in den vorhergehenden Tabellen auch, die Abweichung des untersuchten Ansatzes – hier ELCSP – zum besten domänenspezifischen Ansatz. Als Ergebnis der Auswertung steht fest, dass ELCSP in der knappen Mehrheit der Domänen ein besseres Ergebnis erzielt als die Variante ESLS.

Es ist allerdings zu beachten, dass die überraschend gute Performanz des gewählten ELCSP-Verfahrens durchaus zufällig aufgetreten sein kann. Die hier verwendete Variante betrachtet tatsächlich immer nur den längsten verfügbaren Beobachtungsbereich unter allen zu vergleichenden Zeitreihen, gemessen an der kürzesten Reihe in der Vergleichsmenge. Ist diese beispielsweise 20 Beobachtungen lang, so werden von jeder zu vergleichenden Zeitreihe ausschließlich die letzten 20 Beobachtungen verglichen. Hinsichtlich der teilweise sehr hohen Dimensionalität von Zeitreihen (beispielsweise im Fall der Domäne *Energy*) ist es leicht vorstellbar, dass dieser Vergleich alles andere als aussagekräftig ist.

Um das ELCSP-Verfahren erschöpfend und korrekt durchzuführen, müssten alle kürzeren Zeitreihen stufenweise mit jedem möglichen zusammenhängenden Teilstück jeder längeren Zeitreihe verglichen werden. Mit dieser Methodik wäre sicher, dass tatsächlich die Zeitreihe aus der Vergleichsmenge ermittelt wird, welche das Teilstück mit der besten Übereinstimmung beinhaltet. Dadurch könnte problemlos ein Szenario entstehen, bei dem beispielsweise eine 52.000-dimensionale Energiezeitreihe auf Basis eines Teilstückvergleichs der Beobachtungslänge 20 als

Domäne	bester eigenschaftsbasierter Ansatz	ELCSP	Abweichung
Datenbasis	41,43	40,37	-1,06
Demographic	37,93	34,42	-3,51
Economics	19,98	17,35	-2,63
Energy	71,11	73,05	1,94
Finance	38,84	37,86	-0,98
Industry	32,10	30,35	-1,75
Macro	17,62	16,53	-1,09
Micro	46,05	45,35	-0,70
Other	81,69	82,17	0,48

Tabelle 6.12: Ermittelte Abweichungen der SMAPE-Werte durch ELCSP

ähnlichste Zeitreihe bestimmt wird. Es ist jedoch höchst spekulativ, ob diese Ähnlichkeit wirklich repräsentativ ist.

6.6 GEGENÜBERSTELLUNG VON LAUFZEIT UND PERFORMANZ ALLER BETRACHTETER KLASSIFIKATOREN

Nachdem alle Ansätze in den einzelnen Experimenten bewertet wurden, kann nun eine gemeinsame Gegenüberstellung aller untersuchten Ansätze erfolgen. Diese ist durch Tabelle 6.13 gegeben. Im direkten Vergleich scheint sich die Aussage von [WMD+13], dass eine Verbesserung der durch die euklidische Distanz erreichten Fehlerwerte sehr schwierig ist, bewahrheitet zu haben. In der Tat gelingt den eigenschaftsbasierten Ansätzen diese Leistung insgesamt in nur zwei Domänen. Diese sind die Domäne *Energy*, deren bester Ansatz der klassische Klassifikationsbaum darstellt, und die Domäne *Other*, in welcher der domänenspezifische 1NN-Klassifikationsraum das beste Ergebnis erwirkt.

Unter Nichtbeachtung der Euklid-Verfahren stellt sich mit Ausnahme der Domäne *Energy* heraus, dass in allen Domänen des lokal ermittelten 1NN-Klassifikationsraums die besten SMAPE-Werte erwirkt werden. Des Weiteren ist in der Auswertung zu erkennen, dass sich das ANN-Verfahren in fast allen Domänen als insgesamt schlechtester Ansatz herausstellt. Eine Ausnahme dieser Regel bildet die Domäne *Macro*, bei der das Neuronale Netz als zweitbester eigenschaftsbasierter Ansatz ausgezeichnet wurde.

Bei der Betrachtung der absoluten Abstände zwischen den besten Euklid-Verfahren und den besten eigenschaftsbasierten Verfahren fällt auf, dass diese mit Werten zwischen rund 1 Prozentpunkt und rund 3,5 Prozentpunkten eher klein sind. Das bedeutet, dass sie im Regelfall sehr nah beieinanderliegende Ergebnisse produzieren. Dies führt unmittelbar zu der Frage, inwieweit sich die Ansätze bei einem operativen Einsatz hinsichtlich der Ausführungszeiten für die Modellklas-

Domäne	Datenbasis	Demographic	Economics	Energy	Finance	Industry	Macro	Micro	Other
immer bestes Modell	31,40	26,90	13,21	70,45	30,50	24,09	13,14	36,05	66,17
bester dom.-spez. Raum	41,43 3	37,93 3	19,98 2	73,08 4	38,84 2	32,10 3	17,62 3	46,05 3	81,69 1
global bester Raum	41,43 3	40,39 6	26,85 4	74,69 7	40,10 4	33,87 5	18,36 5	46,41 4	86,07 6
Klassifikationsbaum	41,49 5	39,35 4	36,43 6	71,11 1	43,27 5	32,66 4	18,55 6	47,16 6	83,96 4
TP1NN	41,88 6	39,53 5	26,38 3	74,65 6	39,37 3	33,95 6	18,82 7	46,93 5	83,61 3
KNN	42,45 7	40,43 7	37,81 7	73,89 5	47,91 7	34,14 7	18,28 4	48,41 7	85,71 5
ESLS	39,46 1	36,61 2	31,26 5	73,03 2	44,60 6	31,39 2	16,42 1	44,49 1	86,07 7
ELCSP	40,37 2	34,42 1	17,35 1	73,05 3	37,86 1	30,35 1	16,53 2	45,35 2	82,17 2
immer schlechtestes Modell	73,51	71,55	75,87	148,49	71,09	56,36	43,44	78,65	125,53

Bester eigenschaftsbasierter Klassifikator

Bester Klassifikator unter Einbeziehung der euklidischen Distanzen

X Rang des jeweiligen Klassifikators unter allen Ansätzen in einer Domäne

Tabelle 6.13: Finale Gegenüberstellung der SMAPE-Werte aller Klassifikatoren

sifikation verhalten.

Zu diesem Zweck wird ein weiteres Messexperiment durchgeführt. Bei diesem wird ohne Beschränkung der Allgemeinheit aus jeder der insgesamt 100 Stichproben die erste Zeitreihe entnommen und anhand der zur Stichprobe gehörenden Trainingsdatenmenge durch jeden der untersuchten Ansätze klassifiziert. Die jeweiligen Einzellaufzeiten werden addiert und das Ergebnis anschließend gemittelt. Aus diesem Mittel ergibt sich die durchschnittliche Laufzeit zur Klassifikation einer Zeitreihe unter Verwendung der jeweiligen Klassifikatoren.

Die unterschiedlichen eigenschaftsraumbasierten Klassifikatoren (alle Varianten von 1NN und TP1NN) werden dabei zusammengefasst. Sowohl bei Klassifikationsbäumen als auch bei ANNs ist es möglich, die jeweiligen Klassifikationsstrukturen einerseits operativ zur Anfragezeit zu berechnen und andererseits in vorberechneter Form aus einer Datenquelle zu laden. Aus diesem Grund ist deren Analyse nach den eben erläuterten Prinzipien getrennt aufgeführt.

Tabelle 6.14 protokolliert die entsprechenden gemessenen Laufzeiten der einzelnen Klassifikatoren. Das erhaltene Ergebnis entspricht im Wesentlichen den Erwartungen und zeigt deutlich, dass die eigenschaftsbasierten Ansätze eine signifikant kleinere Laufzeit aufweisen als die rohwertbasierten Euklid-Methoden. Der gemessene Faktor des Laufzeitunterschieds zwischen den Euklid- und den eigenschaftsraumbasierten Verfahren schwankt beispielsweise zwischen Faktor 18 und Faktor 2.695. Schon in der verhältnismäßig kleinen Datenbasis der Evaluationsdatenbank zeigt sich somit ein deutliches lineares Skalierungsverhalten der Euklid-Ansätze, welches bei anwachsenden Datenbasen zu einer raschen Vervielfachung der Systemantwortzeiten führen würde. Bleibt der ermittelte Trend bestehen, liegt die Berechnungszeit für eine anfragende Zeitreihe in einer Zeitreihendatenbank von 31.000 Vergleichsreihen schon bei mehr als 90 Sekunden. Dies bestätigt erneut die mangelnde Eignung rohwertbasierter Ähnlichkeitsmaße in operativen Systemen und führt auch in dieser Betrachtung zu der deutlichen Empfehlung, im Sinne der Wahrung vertretbarer Systemanwortzeiten eine Verwendung dieser zu vermeiden. Auf Grund

Domäne	1NN	Klassifikationsbaum (mit Erstellung)	Klassifikationsbaum (nur Klassifikation)	ANN (mit Erstellung)	ANN (nur Klassifikation)	ESLS	ELCSP
Datenbasis	0,0034	0,2857	0,0022	4,1877	0,3837	0,1580	9,0590
Demographic	0,0022	0,0265	0,0023	2,7488	0,3669	0,0384	1,0672
Economics	0,0011	0,0073	0,0022	1,6058	0,3788	0,0754	0,1739
Energy	0,0013	0,0128	0,0021	2,7303	0,3804	0,5394	0,6819
Finance	0,0016	0,0205	0,0022	2,6141	0,3660	0,0346	0,7929
Industry	0,0011	0,0328	0,0023	2,8491	0,4003	0,0429	1,3376
Macro	0,0024	0,0535	0,0022	2,8505	0,3743	0,1777	1,8755
Micro	0,0021	0,0549	0,0020	2,8911	0,3941	0,3505	2,1186
Other	0,0013	0,0143	0,0020	2,7600	0,3879	0,0299	0,5272

Tabelle 6.14: Durchschnittliche Laufzeiten in Sekunden zur Klassifikation einer Zeitreihe

der Tatsache, dass die eigenschaftsbasierten Ansätze eine vergleichbar gute Klassifikationsleistung erbringen, sind sie an dieser Stelle klar zu bevorzugen.

Dies wird vor allem dann deutlich, wenn die jeweiligen Messergebnisse der alleinigen Klassifikation durch wahlweise den Klassifikationsbaum oder das ANN betrachtet werden. Es zeigt sich, dass diese Klassifikatoren nach ihrer Erstellung in der Wiederverwendung eine grundlegend stabil bleibende Laufzeit aufweisen. In Anbetracht der Tatsache, dass für jede Zeitreihe immer ein und dieselbe Klassifikationsstruktur wiederverwendet wird, spiegelt das aufgezeichnete Resultat die mitgebrachte Erwartungshaltung wider. Es zeigt sich, dass die Berechnungszeiten zur Erstellung der einzelnen Klassifikationsstrukturen mit der jeweiligen Trainingsdatenmenge skalieren und daher in ihrer Ausprägung variieren. Auch dies erscheint unter den gegebenen Voraussetzungen schlüssig.

Insgesamt betrachtet zeigt sich, dass die Klassifikationsgüte der euklidischen Distanz in den meisten Fällen nicht zu verbessern ist. In Systemen, bei welchen der Zeitaspekt keine vordergründige Rolle spielt, sollte daher einer Verwendung dieser Algorithmen – unter Zuhilfenahme geeigneter Parallelisierungsmechanismen – durchaus Beachtung geschenkt werden. Im Regelfall sollte der Zeitaspekt jedoch durchaus von Belang sein. Daher gelten die getesteten eigenschaftsbasierten Verfahren, im Hinblick auf die signifikante Laufzeitreduzierung und die hinreichend gleichartige Klassifikationsgüte, als konkurrenzfähige Methoden und sollten in diesem Fall zum Einsatz kommen.

7 ZUSAMMENFASSUNG UND AUSBLICK

7.1 FAZIT

In dieser Arbeit wurde die methodische Unterstützung der Modellauswahl für Prognoseprozesse am Beispiel von Energiezeitreihen in heterogenen Datensätzen analysiert. Dabei hat sich gezeigt, dass klassische Distanzmaße, wie die euklidische Distanz, unter bestimmten Voraussetzungen als qualitativ hochwertiges Verfahren bei der Modellklassifikation auf Basis einer Ähnlichkeitssuche angesehen werden können. Diese Rahmenbedingungen können jedoch nicht ausnahmslos garantiert werden. Des Weiteren gelten die Berechnungszeiten dieser Verfahren allgemein als ausreichend unpraktikabel, weshalb die Verwendung solcher Ansätze im operativen Betrieb keine gangbare Lösung darstellt.

Eigenschaftsbasierte Verfahren, wie der 1-Nearest-Neighbor-Klassifikator in Eigenschaftsräumen und der Klassifikationsbaum, erreichen eine vergleichbar gute und teilweise sogar bessere Leistung und sind darüber hinaus robust gegenüber den Beschränkungen von Distanzmaßen. Im Gegensatz zu letztgenannten, können bei der Verwendung eigenschaftsbasierter Verfahren berechnete Zwischenergebnisse persistiert werden. Dadurch lassen sich reale Berechnungszeiten der Ähnlichkeitsbeziehung und Modellklassifikation teilweise auf ein stabiles Minimum reduzieren. Im Regelfall wird die Problemkomplexität durch den Eigenschaftsansatz signifikant verringert.

Es zeigte sich ebenfalls, dass eine Repräsentation durch charakteristische Eigenschaften in Verbindung mit der Ausführung gängiger Ähnlichkeitsverfahren und Klassifikationsmethodiken ein sinnvolles Unterfangen darstellt. Dies begründete sich nicht allein durch die Reduktion von Problemkomplexität und Ausführungszeit. Ein Vergleich gegen naive Greedy-Verfahren – wie die ständige Verwendung eines spezifischen Modells oder die ständige Verwendung des häufigsten Modells – offenbarte ebenfalls eine häufige Verbesserung der erzielten Ergebnisse und in jedem Fall eine Erhöhung der Dynamik des zugrunde gelegten Ansatzes.

Im Hinblick auf die Suche nach *einem optimalen Klassifikator* ist zu bemerken, dass kein robuster, global anwendbarer Klassifikator aus der Menge der untersuchten Eigenschaften isoliert werden konnte. Stattdessen variieren die Klassifikatoren domänen- und damit kontextabhängig in ihrer Ausprägung und teilweise auch in ihrer Art. Für die vorliegenden Testdaten konnten sowohl domänenspezifisch als auch über der gesamten Datenbasis Klassifikatoren identifiziert werden, welche hinreichend gute Ergebnisse in der Klassifikation und damit in der Modellempfehlung erzielen.

Die Gesamtheit der Resultate zur Empfehlungskomponente können wie in Abbildung 7.1 gezeigt zusammengefasst werden. Auf diese Weise wird unter Berücksichtigung der jeweils vorliegenden Kontextinformationen der jeweils beste Klassifikator gewählt und mit dessen Hilfe die Modellempfehlung vorgenommen. Der so entstandene Entscheidungsprozess ist leicht in das Framework ECAST im Recommender zu integrieren, wie Abbildung 7.2 illustriert.

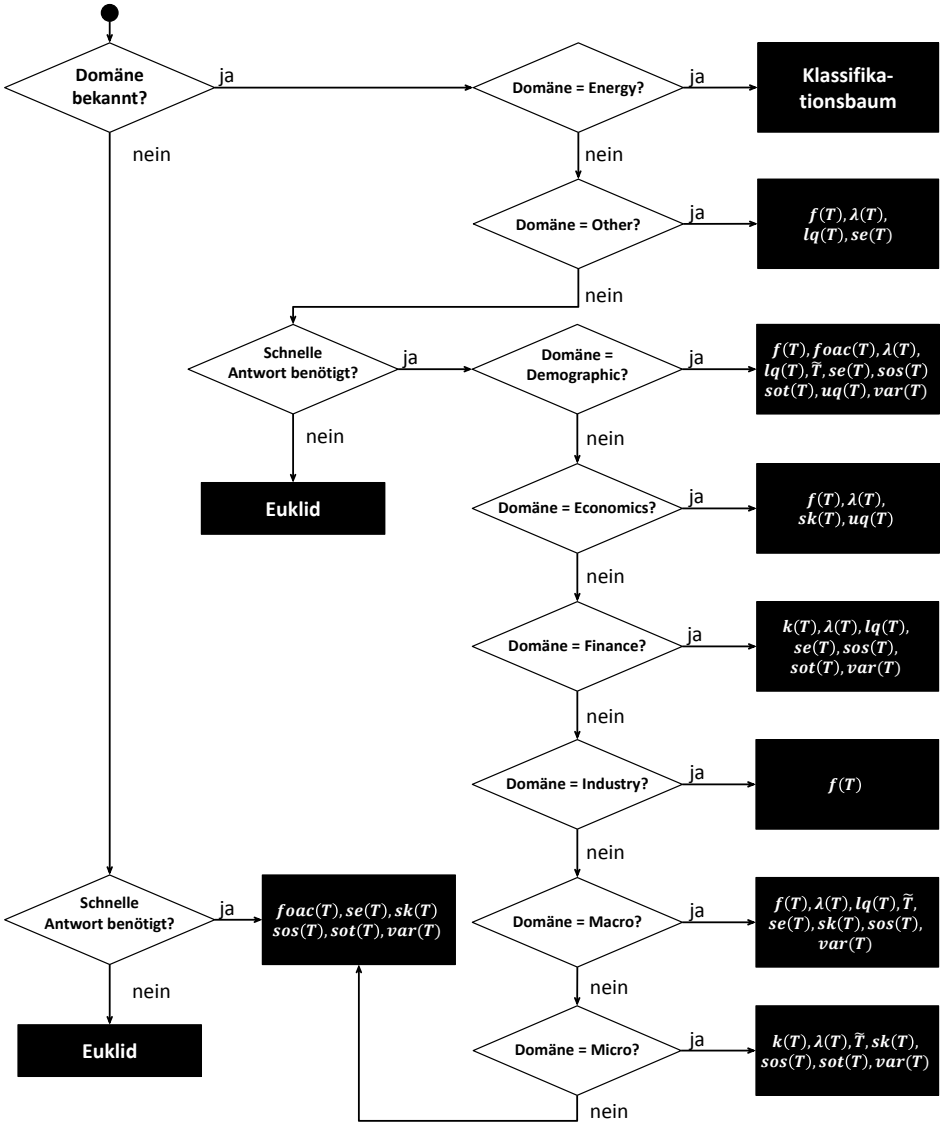


Abbildung 7.1: Ablaufplan des Entscheidungsprozesses zur Auswahl eines Klassifikators

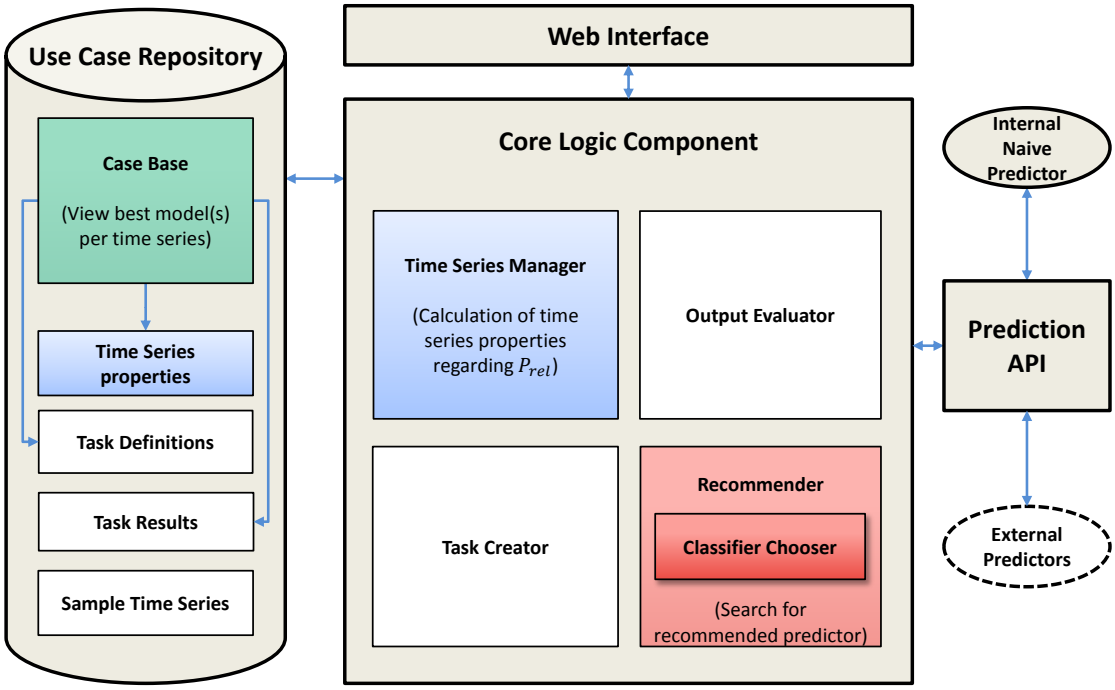


Abbildung 7.2: Eingliederung der Klassifikatorauswahl in das durch SEMPER erweiterte System ECAST

7.2 ANREGUNGEN FÜR ZUKÜNFTIGE ARBEITEN UND ANALYSEN

In der Literatur werden stetig neue Ansätze zur Bestimmung charakteristischer Eigenschaften von Zeitreihen beschrieben. Viele von ihnen werden durch komplexe statistische Verfahren ermittelt und sind damit möglicherweise in der Lage, die inhärente Charakteristik einer einzelnen Zeitreihe noch tiefer zu erfassen. Eine erschöpfende Analyse aller bekannten Kennzahlen hätte den Rahmen dieser Arbeit überstiegen, weshalb die betrachteten Merkmale ein Extrakt der relevanten Zeitreiheneigenschaften darstellen. Unter diesen Gesichtspunkten erscheint es ratsam, in zukünftigen Arbeiten weitere dieser komplexen statistischen Kennzahlen hinsichtlich ihres Repräsentationscharakters zu untersuchen.

Weiterhin besteht die Möglichkeit, die identifizierten Domänen weiter zu unterteilen. Hier sei die Energiedomäne exemplarisch genannt. Sie kann unter anderem durch die Unterscheidung der jeweiligen Energieart, wie etwa Windenergie, Solarenergie, Wasserkraft, Kernkraft, geothermische Energie, Energie aus fossilen Brennstoffen etc., in Unterdomänen zerlegt werden. Es besteht die Chance, dass sich durch solch eine Unterkategorisierung noch bessere und angepasstere Klassifikatoren ermitteln lassen.

Die Auswahl der insgesamt 100 stratifizierten Stichproben und deren Verwendung in den durchgeführten Experimenten simulierte einen laufenden Systembetrieb mit stetiger Erweiterung der Datenbasis. Gegenwärtig ist jedoch noch unklar, wie sich die Klassifikatoren verhalten, wenn die Datenbasis durch längerfristigen Systemeinsatz stark anwächst. Von besonderem Interesse ist hierbei, wie der global beste Klassifikator mit Zeitreihen verfährt, deren (möglicherweise un-

bekannte) Domäne noch nicht durch die Testdaten erfasst wurde. Im schlimmsten Fall müsste in regelmäßigen Abständen eine Wartungsphase angestoßen werden. In dieser wären die in den Kapiteln 5 und 6 gezeigten Analysen zu großen Teilen erneut durchzuführen, um eine eventuelle Anpassung der Klassifikatoren zu erreichen. Diesbezüglich stellt sich unmittelbar die Frage, wie groß die Abstände der Wartungsphase sein dürfen, um ein vertretbares Gleichgewicht zwischen Arbeitsaufwand und Empfehlungsleistung zu wahren.

Schlussendlich besteht aktuell auch noch kein Anspruch auf Vollständigkeit in der Menge der untersuchten Klassifikationsmodelle. Die Verwendung und Betrachtung verschiedener sowohl trivialer als auch systematischer Ansätze bildet zwar ein hinreichend breites Spektrum, kann jedoch noch erweitert werden. Da es im Bereich der Klassifikation auf dem Gebiet des maschinellen Lernens weitere Algorithmen und Verfahren gibt, welche Lösungsansätze zu dieser grundlegenden Problematik anbieten, eröffnet sich hier ein weiterer guter Ansatzpunkt für zukünftige Analysen.

LITERATURVERZEICHNIS

[AFS93] AGRAWAL, Rakesh ; FALOUTSOS, Christos ; SWAMI, Arun: Efficient similarity search in sequence databases. In: *Proceedings of the 4th International Conference on Foundations of Data Organization and Algorithms (FODO '93)* 8958546 (1993), 69–84. http://dx.doi.org/10.1007/3-540-57301-1_5. – DOI 10.1007/3–540–57301–1_5. ISBN 9783540573012

[AKMHt16] AIELLO, Spencer ; KRALJEVIC, Tom ; MAJ, Petr ; H2O.AI TEAM with contributions from t.: *h2o: R Interface for H2O*, 2016. https://cran.r-project.org/package=h2o

[ASEA99] AL-SABA, Tawfiq ; EL-AMIN, Ibrahim: Artificial neural networks as applied to long-term demand forecasting. In: *Artificial Intelligence in Engineering* 13 (1999), Nr. April 1998, S. 189–197. http://dx.doi.org/10.1016/S0954-1810(98)00018-1. – DOI 10.1016/S0954–1810(98)00018–1. – ISBN 09541810 (ISSN)

[BC64] BOX, G. E. P. ; COX, D. R.: An analysis of transformations. In: *Journal of the Royal Statistical Society. Series B (Methodological* 26 (1964), Nr. 2, 211–252. http://dx.doi.org/10.2307/2287791. – DOI 10.2307/2287791. – ISBN EE000178 00359246 DI993152 99P02493

[BC94] BERNDT, Donald J. ; CLIFFORD, James: Using Dynamic Time Warping to Find Patterns in Time Series. In: *KDD workshop* Bd. 10 Seattle, WA, 1994, S. 359–370

[BFO+84] BREIMAN, Leo ; FRIEDMAN, Jerome H. ; OLSHEN, Richard A. ; STONE, Charles J. ; FREIDMAN, J ; OLSEN, R ; STONE, Charles J.: Classification and regression trees. In: *CA: Wadsworth International Group* (1984). http://dx.doi.org/10.1371/journal.pone.0015807. – DOI 10.1371/journal.pone.0015807. – ISBN 978–0534980535

[BH00] BASHEER, I. A. ; HAJMEER, M.: Artificial neural networks: Fundamentals, computing, design, and application. In: *Journal of Microbiological Methods* 43 (2000), Nr. 1, S. 3–31. http://dx.doi.org/10.1016/S0167-7012(00)00201-3. – DOI 10.1016/S0167–7012(00)00201–3. – ISBN 0167–7012 (Print)\n0167–7012 (Linking)

[Bra07] BRAMER, Max: *Principles of Data Mining*. 1. Aufl. Berlin Heidelberg : Springer Science & Business Media, 2007. – ISBN 978–1–846–28766–4

[BTB14] BONTEMPI, Gianluca ; TAIEB, Sb ; BORGNE, Ya L.: Machine learning strategies for time series forecasting. In: *Business Intelligence* (2014), Nr. September, 62–77. http://dx.doi.org/10.1007/978-3-642-36318-4_3. – DOI 10.1007/978–3–642–36318–4_3. – ISBN 978–3–642–36317–7

[CC08] CRYER, Jonathan D. ; CHAN, Kung-Sik: *Time Series Analysis - With Applications in R*. 2. Aufl. Berlin Heidelberg : Springer Science & Business Media, 2008. – ISBN 978–0–387–75958–6

[CCMT90] CLEVELAND, Robert B. ; CLEVELAND, William S. ; MCRAE, Jean E. ; TERPENNING, Irma: STL: A seasonal-trend decomposition procedure based on loess. In: *Journal of Official Statistics* 6 (1990), Nr. 1, S. 3–73. http://dx.doi.org/citeulike-article-id:1435502. – DOI citeulike–article–id:1435502. – ISBN 0282–423X

[CMM05] COHEN, Louis ; MANION, Lawrence ; MORRISON, Keith: *Research Methods in Education*. 5th Editio. 2005. – 226–258 S. http://dx.doi.org/10.1080/19415257.2011.643130. http://dx.doi.org/10.1080/19415257.2011.643130. – ISBN 0203224345

[Coh88] COHEN, Jacob: *Statistical power analysis for the behavioral sciences*. Hillsdale, N.J. : L. Erlbaum Associates, 1988. – ISBN 9780805802832 0805802835

[Czi15] CZICHOS, Horst: *Mechatronik: Grundlagen und Anwendungen technischer Systeme*. 3. Auflage. Springer Fachmedien Wiesbaden, 2015. – 351 S. http://dx.doi.org/10.1007/978-3-658-09950-3. http://dx.doi.org/10.1007/978-3-658-09950-3. – ISBN 978–3–658–09949–7

[DY13] DENG, Li ; YU, Dong: Deep Learning: Methods and Applications. In: *Foundations and Trends® in Signal Processing* 7 (2013), Nr. 3-4, S. 197—-387. http://dx.doi.org/10.1136/bmj.319.7209.0a. – DOI 10.1136/bmj.319.7209.0a. – ISBN 9781405161251

[Ert13] ERTEL, Wolfgang: *Grundkurs Künstliche Intelligenz - Eine praxisorientierte Einführung*. 3. Aufl. Berlin Heidelberg New York : Springer-Verlag, 2013. – ISBN 978–3–834–82157–7

[Fal13] FALLIS, A.G: An introduction to recursive partitioning using the rpart routines. In: *Journal of Chemical Information and Modeling* 53 (2013), Nr. 9, S. 1689–1699. http://dx.doi.org/10.1017/CBO9781107415324.004. – DOI 10.1017/CBO9781107415324.004. – ISBN 9788578110796

[FF94] FALOUTSOS, Christos ; FALOUTSOS, Christos: Fast subsequence matching in time-series databases. In: *ACM SIGMOD Record* 23 (1994), Nr. 2, S. 419–429. http://dx.doi.org/10.1145/191843.191925. – DOI 10.1145/191843.191925. – ISBN 9781595936493

[GBC16] GOODFELLOW, Ian ; BENGIO, Yoshua ; COURVILLE, Aaron: Deep Learning. In: *2015* (2016), 1. http://files.sig2d.org/sig2d14.pdf{#}page=5. ISBN 9781491925614

[GJB14] GARLAND, Joshua ; JAMES, Ryan ; BRADLEY, Elizabeth: Model-free quantification of time-series predictability. In: *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics* 90 (2014), Nr. 5. http://dx.doi.org/10.1103/PhysRevE.90.052910. – DOI 10.1103/PhysRevE.90.052910. – ISSN 15502376

[Goe13] GOERG, Georg M.: Forecastable Component Analysis (ForeCA). In: *arXiv preprint* (2013), 1–9. http://arxiv.org/abs/1205.4591

[Goe16] GOERG, Georg M.: *ForeCA: An R package for Forecastable Component Analysis*, 2016

[HA14] HYNDMAN, Rob J. ; ATHANASOPOULOS, George: *Forecasting: Principles and Practice*. 2014. – 46–51 S. http://dx.doi.org/10.1017/CBO9781107415324.004. http://dx.doi.org/10.1017/CBO9781107415324.004. – ISBN 978–0–98–75071–0–5

[HAA08] HYNDMAN, Rob J. ; AKRAM, Muhammad ; ARCHIBALD, Blyth C.: The admissible parameter space for exponential smoothing models. In: *Annals of the Institute of Statistical Mathematics* 60 (2008), Nr. 2, S. 407–426. http://dx.doi.org/10.1007/s10463-006-0109-x. – DOI 10.1007/s10463–006–0109–x. – ISBN 1046300601

[Has95] HASSOUN, Mohamad H.: *Fundamentals of artificial neural networks*. MIT press, 1995

[Heb49] HEBB, D O.: The Organization of Behavior. In: *The Organization of Behavior* 911 (1949), Nr. 1, S. 335. http://dx.doi.org/10.2307/1418888. – DOI 10.2307/1418888. – ISBN 0805843000

[HK08] HYNDMAN, Rob J. ; KHANDAKAR, Yeasmin: Automatic time series forecasting: the forecast package for {R}. In: *Journal of Statistical Software* 26 (2008), Nr. 3, 1–22. http://www.jstatsoft.org/article/view/v027i03

[HKOS08] HYNDMAN, Rob ; KOEHLER, Anne B. ; ORD, J K. ; SNYDER, Ralph D.: *Forecasting with exponential smoothing: the state space approach*. Springer Science & Business Media, 2008

[Hot33] HOTELLING, Harold: Analysis of a complex of statistical variables into principal components. In: *Journal of educational psychology* 24 (1933), Nr. 6, S. 417

[HS16] HEDDERICH, Jürgen ; SACHS, Lothar: *Angewandte Statistik - Methodensammlung mit R*. 15. Aufl. Berlin Heidelberg New York : Springer-Verlag, 2016. – ISBN 978–3–662–45691–0

[Hyn02] HYNDMAN, R.J.; KOEHLER, A.B.; SNYDER, R.D.; GROSE, S.: A state space framework for automatic forecasting using exponential smoothing methods. In: *International Journal of Forecasting* 18 (2002), Nr. 3, S. 439–454

[Hyn09] HYNDMAN, Rob J.: Forecasting: An Overview. In: *International Encyclopedia of Statistical Science* (2009), 5. http://dx.doi.org/10.1007/978-3-642-04898-2_256. – DOI 10.1007/978–3–642–04898–2_256

[Hyn16] HYNDMAN, Rob J.: *{forecast}: Forecasting functions for time series and linear models*, 2016. http://github.com/robjhyndman/forecast

[JMF+00] JAIN, A K. ; MURTY, M N. ; FLYNN, P J. ; ROSENFELD, Azriel ; BOWYER, K ; AHUJA, N ; JAIN, A: Data Clustering: A Review. (2000). http://dx.doi.org/10.1145/331499.331504. – DOI 10.1145/331499.331504. – ISBN 0360–0300

[KCPM01] KEOGH, Eamonn J. ; CHAKRABARTI, Kaushik ; PAZZANI, Michael J. ; MEHROTRA, Sharad: Dimensionality Reduction for Fast Similarity Search in Large Time Series Databases. In: *Knowledge and Information Systems (KAIS)* 3 (2001), Nr. 3, S. 263–286. http://dx.doi.org/10.1007/PL00011669. – DOI 10.1007/PL00011669. – ISBN 1581133324

[KEDH01] KANDIL, M. S. ; EL-DEBEIKY, S. M. ; HASANIEN, N. E.: Overview and comparison of long-term forecasting techniques for a fast developing utility: Part I. In: *Electric Power Systems Research* 58 (2001), Nr. 1, S. 11–17. http://dx.doi.org/10.1016/S0378-7796(01)00097-9. – DOI 10.1016/S0378–7796(01)00097–9. – ISBN 0378–7796

[KHSm16] KANG, Yanfei ; HYNDMAN, Rob J. ; SMITH-MILES, Kate: Visualising Forecasting Algorithm Performance using Time Series Instance Spaces. (2016), Nr. May

[KR05] KEOGH, Eamonn ; RATANAMAHATANA, Chotirat A.: Exact indexing of dynamic time warping. In: *Knowledge and Information Systems* 7 (2005), mar, Nr. 3, 358–386. http://dx.doi.org/10.1007/s10115-004-0154-9. – DOI 10.1007/s10115–004–0154–9. – ISBN 9781920682828

[Lem09] LEMIRE, Daniel: Faster Retrieval with a Two-Pass Dynamic-Time-Warping Lower Bound. (2009), Nr. June 2009, S. 1–26. http://dx.doi.org/10.1016/j.patcog.2008.11.030. – DOI 10.1016/j.patcog.2008.11.030

[LLK85] LAWLER, E L. ; LENSTRA, Jan K. ; KAN, A H G R.: *The Traveling Salesman Problem - A Guided Tour of Combinatorial Optimization*. Weinheim : John Wiley & Sons, Incorporated, 1985. – ISBN 978–0–471–90413–7

[LVVC11] LHERMITTE, S. ; VERBESSELT, J. ; VERSTRAETEN, W. W. ; COPPIN, P.: A comparison of time series similarity measures for classification and change detection of ecosystem dynamics. In: *Remote Sensing of Environment* 115 (2011), Nr. 12, 3129–3152. http://dx.doi.org/10.1016/j.rse.2011.06.020. – DOI 10.1016/j.rse.2011.06.020. – ISBN 0034–4257

[MAC+82] MAKRIDAKIS, S. ; ANDERSEN, A. ; CARBONE, R. ; FILDES, R. ; HIBON, M. ; LEWANDOWSKI, R. ; NEWTON, J. ; PARZEN, E. ; WINKLER, R.: The Accuracy of Extrapolative (Time Series) Methods - Results of a Forecasting Competition. In: *Journal of Forecasting* 1 (1982), Nr. 2, S. 111–153. http://dx.doi.org/10.1002/for.3980010202. – DOI 10.1002/for.3980010202. – ISBN 1099–131X

[MCH+93] MAKRIDAKIS, Spyros ; CHATFIELD, Chris ; HIBON, Michèle ; LAWRENCE, Michael ; MILLS, Terence ; ORD, Keith ; SIMMONS, LeRoy F.: The M2-competition: A real-time judgmentally based forecasting study. In: *International Journal of Forecasting* 9 (1993), Nr. 1, S. 5–22. http://dx.doi.org/10.1016/0169-2070(93)90044-N. – DOI 10.1016/0169–2070(93)90044–N. – ISBN 0169–2070

[MH00] MAKRIDAKIS, Spyros ; HIBON, Michèle: The M3-Competition: results, conclusions and implications. In: *International Journal of Forecasting* 16 (2000), Nr. 4, S. 451–476. http://dx.doi.org/10.1016/S0169-2070(00)00057-1. – DOI 10.1016/S0169–2070(00)00057–1. – ISBN 0169–2070

[Mit12] MITCHELL, Thomas: Machine Learning. In: *Machine Learning* (2012), Nr. X, S. 639–644. http://dx.doi.org/10.1145/242224.242229. – DOI 10.1145/242224.242229. – ISBN 1101100111111

[MP43] MCCULLOCH, W S. ; PITTS, W: A Logical Calculus of the Idea Immanent in Nervous Activity. In: *Bulletin of Mathematical Biophysics* 5 (1943), 115–133. http://dx.doi.org/10.1007/BF02478259. – DOI 10.1007/BF02478259. – ISBN 0007–4985

[MP88] MINSKY, Marvin L. ; PAPERT, Seymour: *Perceptrons : an Introduction to Computational Geometry*. Expanded e. MIT : MIT Press, 1988. – ISBN 0–262–63111–3

[MR02] MAASOUMI, Esfandiar ; RACINE, Jeff: Entropy and predictability of stock market returns. In: *Journal of Econometrics* 107 (2002), Nr. 1-2, S. 291–312. http://dx.doi.org/10.1016/S0304-4076(01)00125-7. – DOI 10.1016/S0304–4076(01)00125–7. – ISBN 0304–4076

[NAM01] NANOPOULOS, Alex ; ALCOCK, R. ; MANOLOPOULOS, Yannis: Feature-based classification of time-series data. In: *Information processing and Management* 0056 (2001), 49–61. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.73.9555{&}rep=rep1{&}type=pdf. ISBN 1–59033–116–8

[Pea95] PEARSON, Karl: Note on Regression and Inheritance in the Case of Two Parents. In: *Proceedings of the Royal Society of London* 58 (1895), 240–242. http://www.jstor.org/stable/115794. – ISSN 03701662

[Pea01] PEASON, K: On lines and planes of closest fit to systems of point in space. In: *Philosophical Magazine* 2 (1901), S. 559–572

[Qui86] QUINLAN, J. R.: Induction of Decision Trees. In: *Machine Learning* 1 (1986), Nr. 1, S. 81–106. http://dx.doi.org/10.1023/A:1022643204877. – DOI 10.1023/A:1022643204877. – ISBN 0885–6125

[Qui93] QUINLAN, Jr: *C4.5: programs for machine learning*. Bd. 240. 1993. – 302 S. http://dx.doi.org/10.1016/S0019-9958(62)90649-6. http://dx.doi.org/10.1016/S0019-9958(62)90649-6. – ISBN 1558602380

[RM14] ROKACH, Lior ; MAIMON, Oded: *Data mining with decision trees: theory and applications*. World scientific, 2014

[RN88] RODGERS, Joseph L. ; NICEWANDER, W A.: Thirteen Ways to Look at the Correlation Coefficient. In: *The American Statistician* 42 (1988), Nr. 1, 59. http://dx.doi.org/10.2307/2685263. – DOI 10.2307/2685263. – ISBN 00031305

[Ros58] ROSENBLATT, Frank: The perceptron: a probabilistic model for information storage and organization in the brain. In: *Psychological review* 65 (1958), Nr. 6, S. 386

[Run12] RUNKLER, Thomas A.: *Data Analytics - Models and Algorithms for Intelligent Data Analysis*. 2012. Aufl. Berlin Heidelberg : Springer Science & Business Media, 2012. – ISBN 978–3–834–82589–6

[SA14] SERRÀ, Joan ; ARCOS, Josep L.: An empirical evaluation of similarity measures for time series classification. In: *Knowledge-Based Systems* 67 (2014), S. 305–314. http://dx.doi.org/10.1016/j.knosys.2014.04.035. – DOI 10.1016/j.knosys.2014.04.035. – ISSN 09507051

[SC78] SAKOE, Hiroaki ; CHIBA, Seibi: Dynamic Programming Algorithm Optimization for Spoken Word Recognition. In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 26 (1978), Nr. 1, S. 43–49. http://dx.doi.org/10.1109/TASSP.1978.1163055. – DOI 10.1109/TASSP.1978.1163055. – ISBN 9788578110796

[SC07] SALVADOR, S ; CHAN, P: FastDTW : Toward Accurate Dynamic Time Warping in Linear Time and Space. In: *Intelligent Data Analysis* 11 (2007), Nr. 5, S. 561–580

[Sch15] SCHMIDHUBER, Jürgen: Deep Learning in neural networks: An overview. In: *Neural Networks* 61 (2015), S. 85–117. http://dx.doi.org/10.1016/j.neunet.2014.09.003. – DOI 10.1016/j.neunet.2014.09.003. – ISBN 0893–6080

[Spi76] SPITZER, Frank: *Principles of random walk*. Bd. 34. Springer Science & Business Media, 1976. – 416 S. http://dx.doi.org/10.1007/978-1-4757-4229-9. http://dx.doi.org/10.1007/978-1-4757-4229-9. – ISBN 9781475742299

[TGQS09] TORMENE, Paolo ; GIORGINO, Toni ; QUAGLINI, Silvana ; STEFANELLI, Mario: Matching incomplete time series with dynamic time warping: an algorithm and an application to post-stroke rehabilitation. In: *Artificial intelligence in medicine* 45 (2009), Nr. 1, S. 11–34

[TWSC14] TEAM, Rmetrics C. ; WUERTZ, Diethelm ; SETZ, Tobias ; CHALABI, Yohan: *fBasics: Rmetrics - Markets and Basic Statistics*, 2014. https://cran.r-project.org/package=fBasics

[UFK+14] ULBRICHT, Robert ; FISCHER, Ulrike ; KEGEL, Lars ; HABICH, Dirk ; DONKER, Hilko ; LEHNER, Wolfgang: ECAST: A benchmark framework for renewable energy forecasting systems. In: *CEUR Workshop Proceedings* 1133 (2014), S. 148–155. – ISSN 16130073

[UHH+16] ULBRICHT, Robert ; HARTMANN, Claudio ; HAHMANN, Martin ; DONKER, Hilko ; LEHNER, Wolfgang: Web-based Benchmarks for Forecasting Systems - The ECAST Platform. In: *Sigmod '16* (2016), S. 2–5. http://dx.doi.org/10.1145/2882903.2899399. – DOI 10.1145/2882903.2899399. – ISBN 9781450335317

[War05] WARREN LIAO, T.: Clustering of time series data - A survey. In: *Pattern Recognition* 38 (2005), Nr. 11, S. 1857–1874. http://dx.doi.org/10.1016/j.patcog.2005.01.025. – DOI 10.1016/j.patcog.2005.01.025. – ISBN 0031–3203

[WH60] WIDROW, B ; HOFF, Me: *Adaptive switching circuits.* 1960

[WMD+13] WANG, Xiaoyue ; MUEEN, Abdullah ; DING, Hui ; TRAJCEVSKI, Goce ; SCHEUERMANN, Peter ; KEOGH, Eamonn: Experimental comparison of representation methods and distance measures for time series data. In: *Data Mining and Knowledge Discovery* 26 (2013), Nr. 2, S. 275–309. http://dx.doi.org/10.1007/s10618-012-0250-5. – DOI 10.1007/s10618–012–0250–5. – ISBN 1384–5810

[WSH06] WANG, Xiaozhe ; SMITH, Kate ; HYNDMAN, Rob: Characteristic-based clustering for time series data. In: *Data Mining and Knowledge Discovery* 13 (2006), Nr. 3, S. 335–364. http://dx.doi.org/10.1007/s10618-005-0039-x. – DOI 10.1007/s10618–005–0039–x. – ISBN 1384–5810

[YWI99] YOHANNES, Yisehac ; WEBB, Patrick ; INSTITUTE, International Food Policy R.: *Classification and Regression Trees, CART - A User Manual for Identifying Indicators of Vulnerability to Famine and Chronic Food Insecurity*. Pap/Dsk. Washington, D.C. : Intl Food Policy Res Inst, 1999. – ISBN 978–0–896–29337–3

ABBILDUNGSVERZEICHNIS

2.1 Eine Menge von Zeitreihen gleicher Länge 6

2.2 Distanzmatrizen der Zeitreihen aus Abbildung 2.1 unter Verwendung der Distanz- und Ähnlichkeitsmaße aus Tabelle 2.1 7

2.3 Zwei Signale unterschiedlicher Länge 8

2.4 Der CBR-Zyklus des Case-based Reasoning 12

3.1 Streudiagramme der Zeitreihen *Geburten* und *Getreide* 17

3.2 Domänenspezifische Einteilung der Zeitreihen der M3-Competition 21

3.3 Domänenspezifische Einteilung der Zeitreihen der Datenbasis 23

3.4 Korrelationsmatrix der ersten 12 Eigenschaften aus Tabelle 3.1 auf der Datenbasis . 25

3.5 Korrelationsmatrix der finalen Eigenschaftsmenge P_{rel} auf der Datenbasis 28

4.1 Grundlegende Architektur des Systems ECAST 30

4.2 Logische Architektur der Empfehlungskomponente SEMPER 33

4.3 Integration der Komponente SEMPER in das System ECAST 34

5.1 Schema der Evaluationsdatenbank 36

5.2 Grafische Darstellung der Verteilung der Modelle als Klassifikationslabels in der Evaluationsdatenbank 39

5.3 Aufteilung der Zeitreihen je Domäne in Trainings- und Testdaten 41

5.4 Erzielte SMAPE-Werte des besten Teilraumes auf der Datenbasis 43

5.5 Erzielte SMAPE-Werte des schlechtesten Teilraumes auf der Datenbasis 44

5.6 Streuung der SMAPE-Werte aller Teilräume auf der Datenbasis 44

5.7 Streuung der SMAPE-Werte aller Teilräume je Domäne 47

6.1 Der zu Tabelle 6.1 generierte Klassifikationsbaum 51

6.2 Erzielte SMAPE-Werte des Klassifikationsbaumes auf der gesamten Datenbasis . . 52

6.3 Erzielte SMAPE-Werte des Klassifikationsbaumes in der Domäne *Energy* 53

6.4 Erzielte SMAPE-Werte des Neuronalen Netzes in der Domäne *Economics* 61

7.1 Ablaufplan des Entscheidungsprozesses zur Auswahl eines Klassifikators 70

7.2 Eingliederung der Klassifikatorauswahl in das durch SEMPER erweiterte System ECAST 71

A.1 SMAPE-Diagramme aller Klassifikatoren auf der Datenbasis 86

A.2 SMAPE-Diagramme aller Klassifikatoren auf der Domäne *Demographic* 87

A.3 SMAPE-Diagramme aller Klassifikatoren auf der Domäne *Economics* 88

A.4 SMAPE-Diagramme aller Klassifikatoren auf der Domäne *Energy* 89

A.5 SMAPE-Diagramme aller Klassifikatoren auf der Domäne *Finance* 90

A.6 SMAPE-Diagramme aller Klassifikatoren auf der Domäne *Industry* 91

A.7 SMAPE-Diagramme aller Klassifikatoren auf der Domäne *Macro* 92

A.8 SMAPE-Diagramme aller Klassifikatoren auf der Domäne *Micro* 93

A.9 SMAPE-Diagramme aller Klassifikatoren auf der Domäne *Other* 94

TABELLENVERZEICHNIS

2.1 Formale Definitionen gebräuchlicher Distanz- und Ähnlichkeitsmaße 6

2.2 Warp-Matrix der Signale aus Abbildung 2.3 9

3.1 Aufstellung aller zu analysierenden Zeitreiheneigenschaften 20

3.2 Domänenspezifische und intervallbasierte Klassifizierung der M3-Daten 22

3.3 Klassifikationen der Datenbasis 24

5.1 Verteilung der Modelle als Klassifikationslabels in der Evaluationsdatenbank . . . 38

5.2 Beste domänenspezifische Klassifikationsräume 46

5.3 Erzielte Verringerungen der SMAPE-Werte durch domänenspezifische Klassifikationsräume 46

5.4 Vergleich der ermittelten SMAPE-Werte der globalen und domänenspezifischen Experimente 48

6.1 Eine beispielhafte Fallbasis 50

6.2 Ermittelte Unterschiede der SMAPE-Werte durch domänenspezifische Klassifikationsbäume 53

6.3 Vergleich der ermittelten SMAPE-Werte der domänenspezifischen Klassifikationsräume und Klassifikationsbäume 54

6.4 Ermittelte Strafwerte je Eigenschaft je Domäne 55

6.5 Durch baumbasierte Vorverarbeitung ermittelte Teilräume 57

6.6 Markierung der Abtrennungseigenschaft zur Identifikation der Menge häufigster Teilungseigenschaften in den Klassifikationsbäumen 58

6.7 Vergleich der ermittelten SMAPE-Werte der besten Klassifikationsräume, des Baumklassifikators und des TP1NN-Ansatzes 59

6.8 Ermittelte Abweichungen der SMAPE-Werte durch ANNs 60

6.9 Vergleich der ermittelten SMAPE-Werte der besten Klassifikationsräume, des Baumklassifikators und des Neuronalen Netzes 62

6.10 Ermittelte SMAPE-Werte der Ansätze FSMR und FMMR 63

6.11 Ermittelte Abweichungen der SMAPE-Werte durch ESLS 65

6.12 Ermittelte Abweichungen der SMAPE-Werte durch ELCSP 66

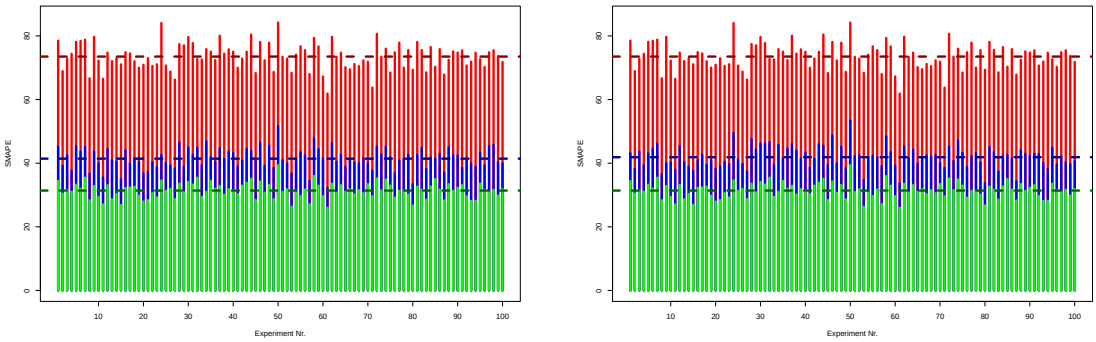
6.13 Finale Gegenüberstellung der SMAPE-Werte aller Klassifikatoren 67

6.14 Durchschnittliche Laufzeiten in Sekunden zur Klassifikation einer Zeitreihe 68

ABKÜRZUNGSVERZEICHNIS

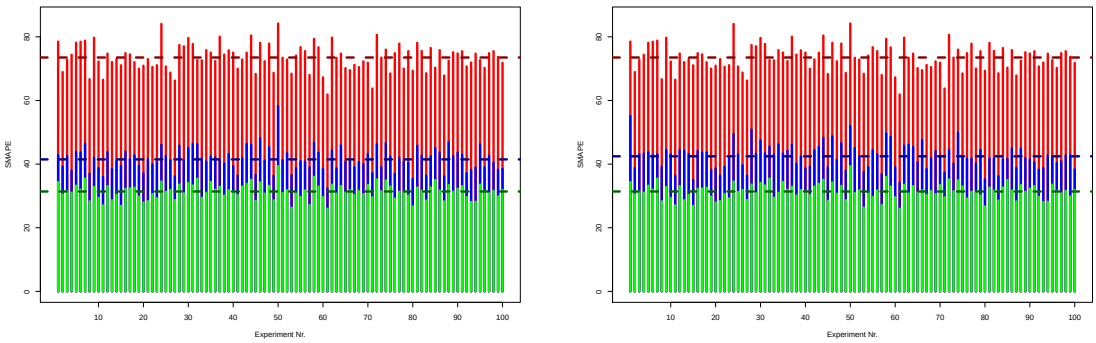
1NN	1-Nearest-Neighbor 35
ANN	Artificial Neural Network 59
CART	Classification And Regression Tree 49
CBC	Characteristic-Based Clustering 11
CBR	Case-based Reasoning 12
CCP	Control Chart Pattern 10
CLC	Core Logic Component 30
DBMS	Datenbank Management System 36
DTW	Dynamic Time Warping 8
ECAST	Energy Forecasting Benchmark Framework 29
EEX	European Energy Exchange 23
EKG	Elektrokardiogramm 10
ELCSP	Euclidean distance on Last Common Series Part 64
ESLS	Euclidean distance on Same Length Series 64
ETS	Exponential Smoothing State Space 37
FMMR	Forced Most common Model Reuse 62
FSMR	Forced Single Model Reuse 62
ID3	Iterative Dichotomiser 3 50
IG	Information Gain 50
MAPE	Mean Average Percentage Error 38
MLP	Multi-layer Perceptron 10
MW	Megawatt 22
NREL	National Renewable Energy Laboratory 22
OE	Output Evaluator 30
PAPI	Prediction API 30
PCA	Principal Component Analysis 5
RMSE	Root Mean Square Error 30
SEMPER	Systematic Engine for Model classification on Property based Empirical Recommendations 32
SMAPE	Symmetric Mean Average Percentage Error 30
STL	Seasonal and Trend decomposition using Loess 19
TC	Task Creator 30
TP1NN	Tree Prepared 1-Nearest-Neighbor 54
TSDL	Time Series Data Library 23
TSM	Time Series Manager 30
UCR	Use Case Repository 30
WI	Web Interface 30
WIDS	Wind Integration Data Set 22

A SMAPE-DIAGRAMME ALLER KLASSIFIKATOREN



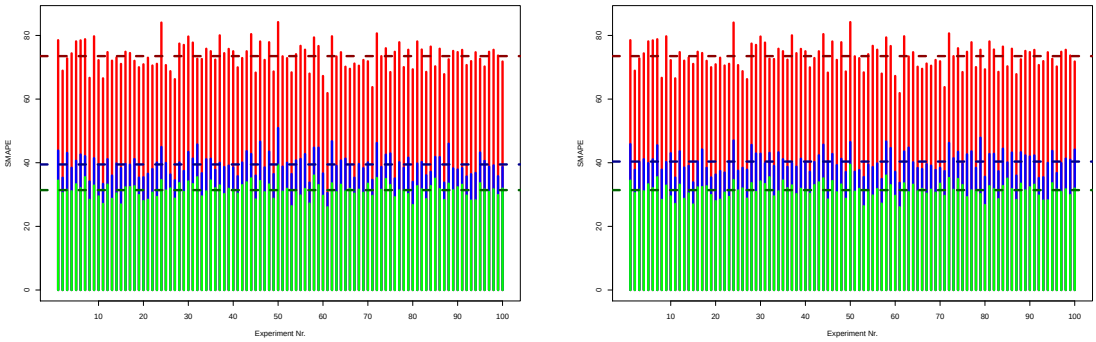
(a) bester 1NN-Raum

(b) TP1NN-Raum



(c) Klassifikationsbaum

(d) KNN!-Klassifikator



(e) ESLS-Ansatz

(f) ELCSP-Ansatz

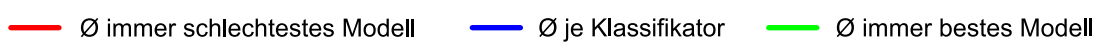
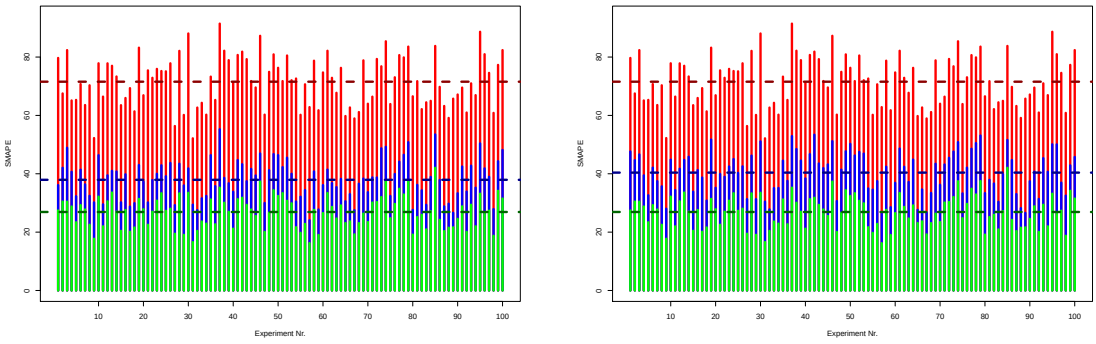
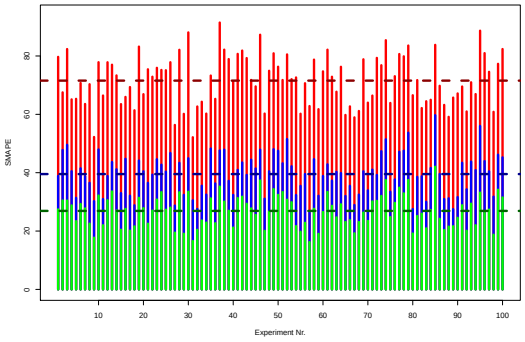


Abbildung A.1: SMAPE-Diagramme aller Klassifikatoren auf der Datenbasis

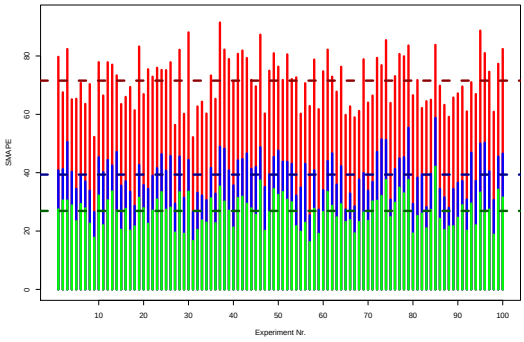


(a) bester domänenspezifischer 1NN-Raum

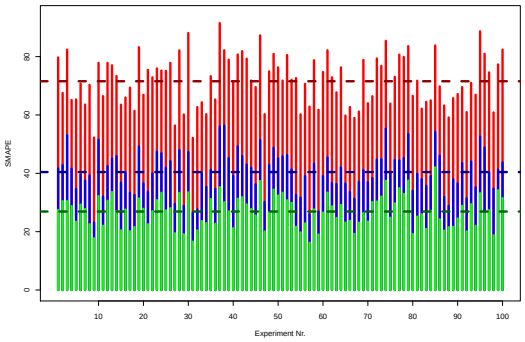
(b) global bester 1NN-Raum



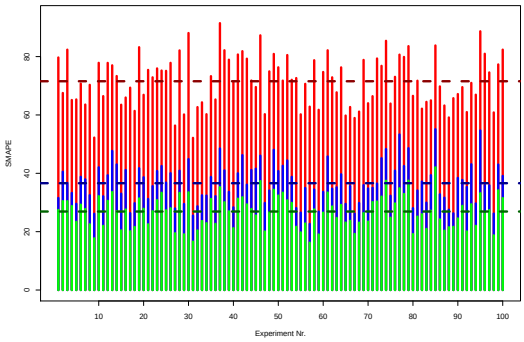
(c) TP1NN-Raum



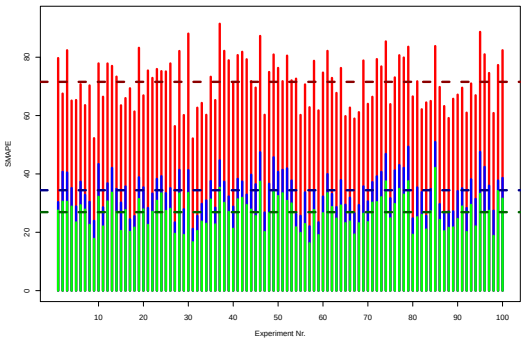
(d) Klassifikationsbaum



(e) KNN!-Klassifikator



(f) ESLS-Ansatz



(g) ELCSP-Ansatz

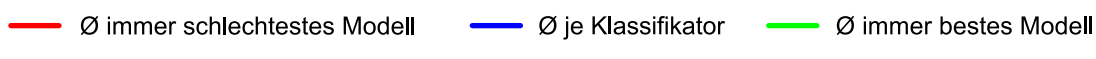
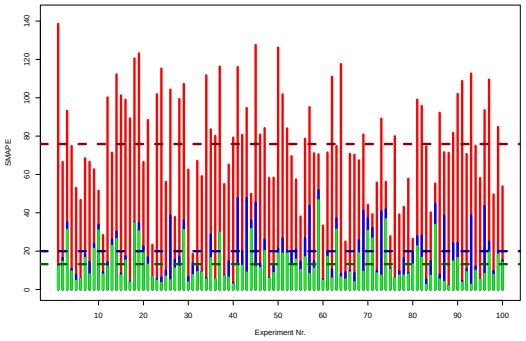
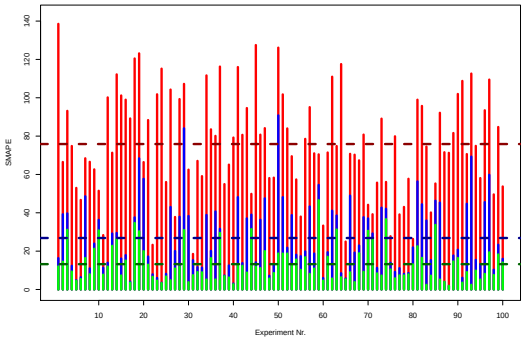


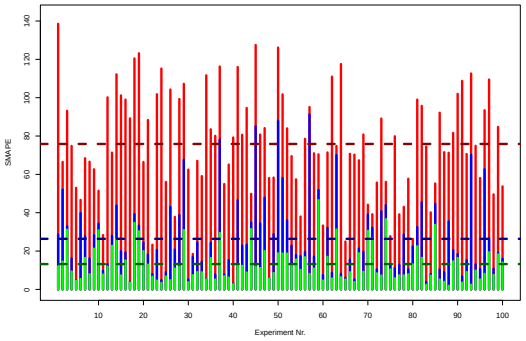
Abbildung A.2: SMAPE-Diagramme aller Klassifikatoren auf der Domäne *Demographic*



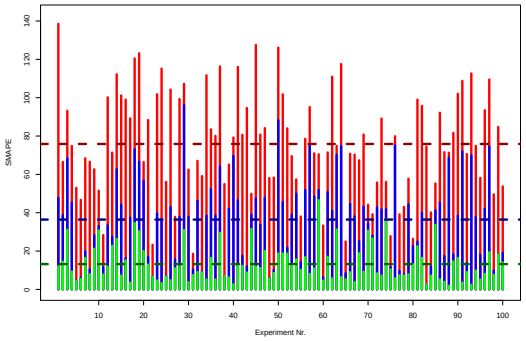
(a) bester domänenspezifischer 1NN-Raum



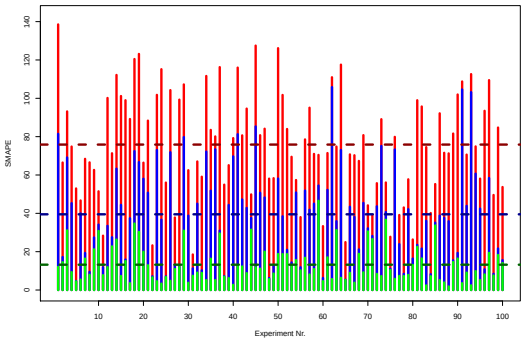
(b) global bester 1NN-Raum



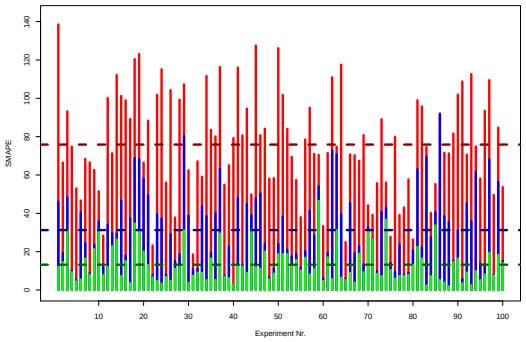
(c) TP1NN-Raum



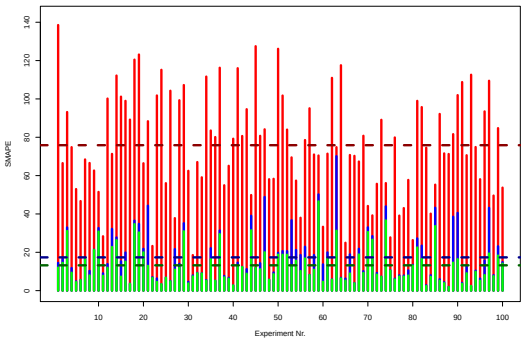
(d) Klassifikationsbaum



(e) **KNN!**-Klassifikator



(f) ESLS-Ansatz



(g) ELCSP-Ansatz

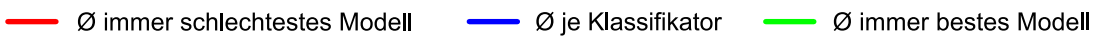
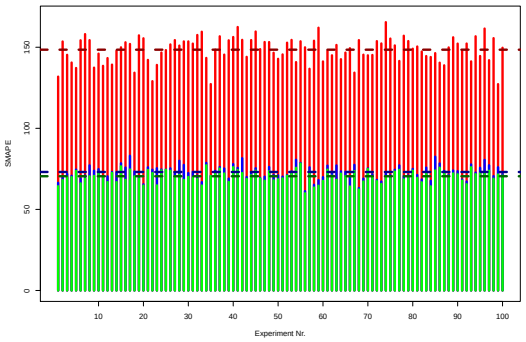
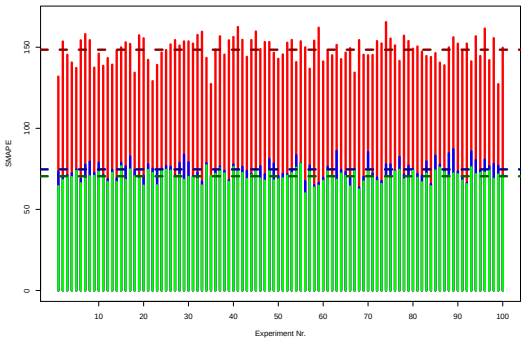


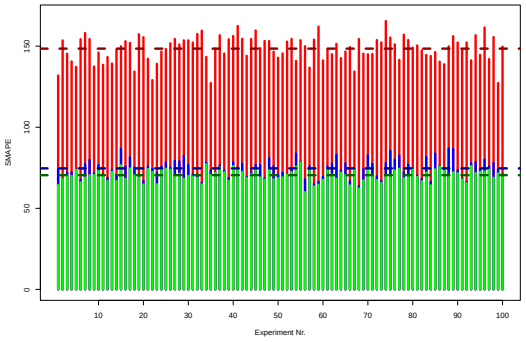
Abbildung A.3: SMAPE-Diagramme aller Klassifikatoren auf der Domäne *Economics*



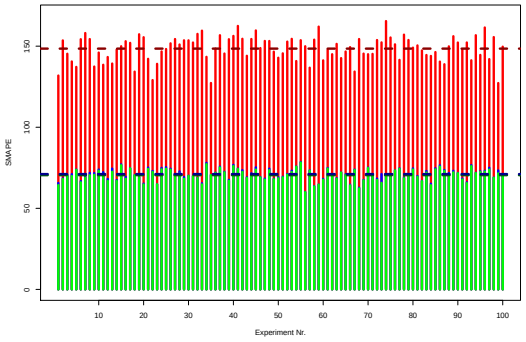
(a) bester domänenspezifischer 1NN-Raum



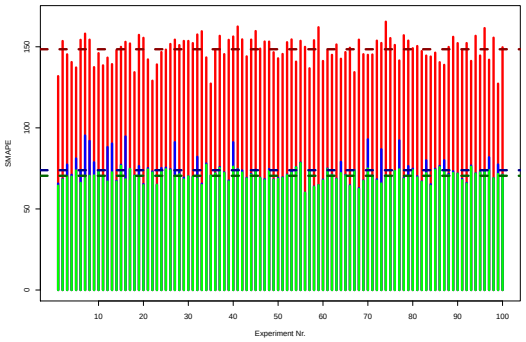
(b) global bester 1NN-Raum



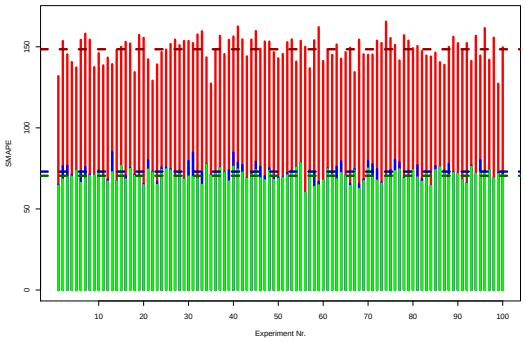
(c) TP1NN-Raum



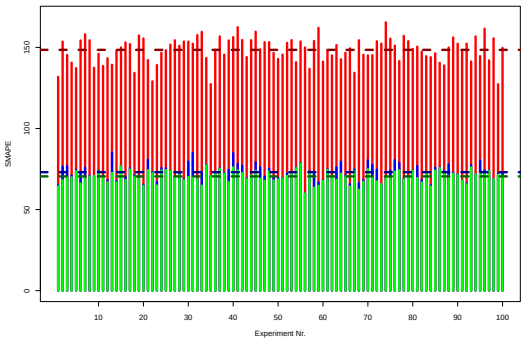
(d) Klassifikationsbaum



(e) KNN!-Klassifikator



(f) ESLS-Ansatz



(g) ELCSP-Ansatz

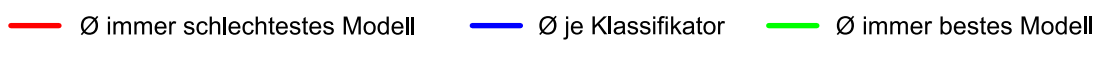
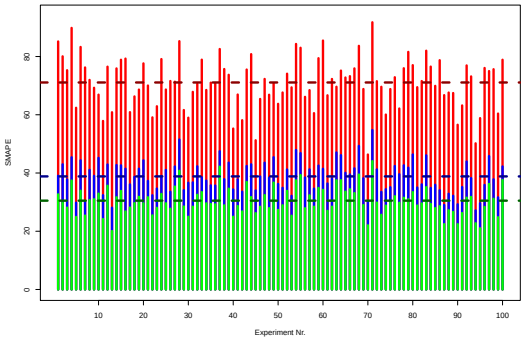
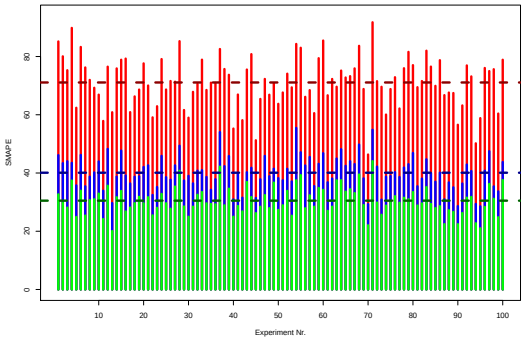


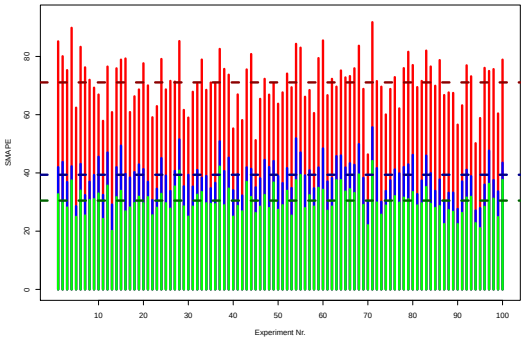
Abbildung A.4: SMAPE-Diagramme aller Klassifikatoren auf der Domäne *Energy*



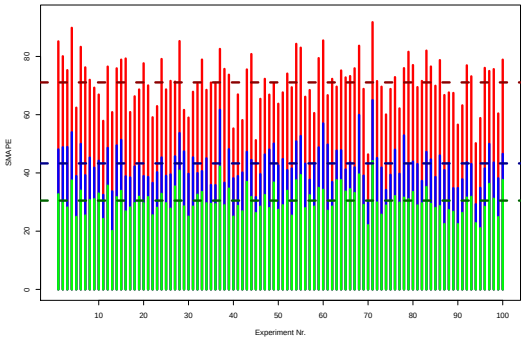
(a) bester domänenspezifischer 1NN-Raum



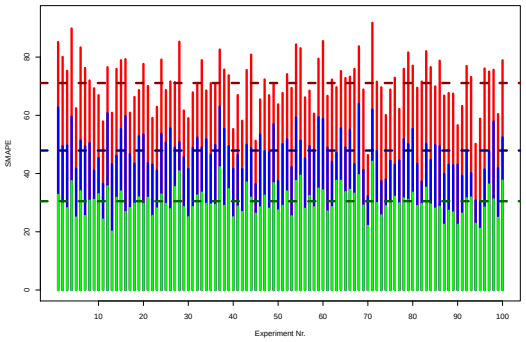
(b) global bester 1NN-Raum



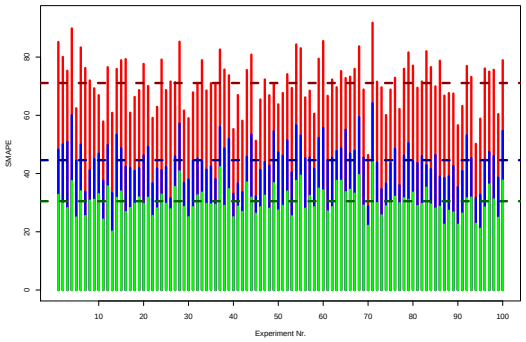
(c) TP1NN-Raum



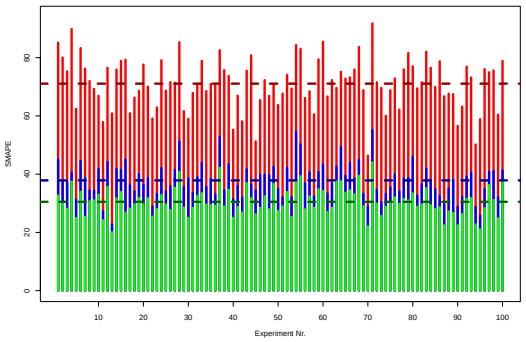
(d) Klassifikationsbaum



(e) **KNN!**-Klassifikator



(f) ESLS-Ansatz



(g) ELCSP-Ansatz

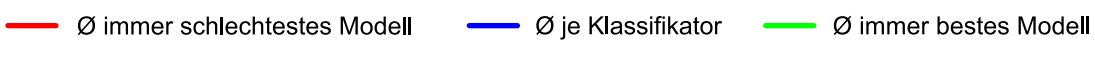
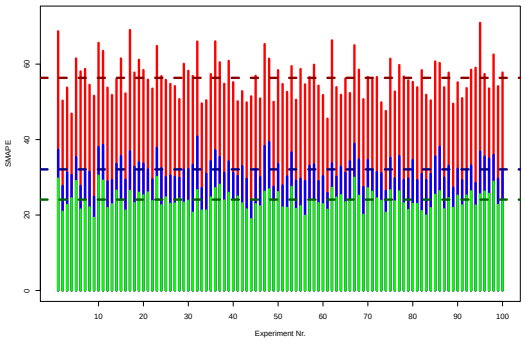
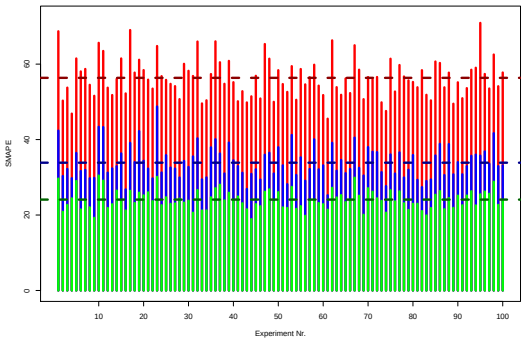


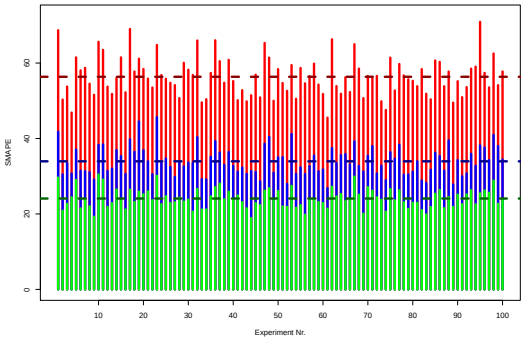
Abbildung A.5: SMAPE-Diagramme aller Klassifikatoren auf der Domäne *Finance*



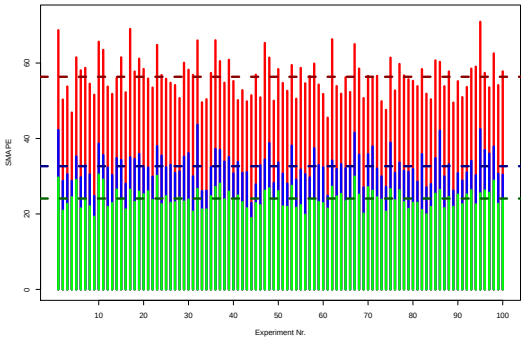
(a) bester domänenspezifischer 1NN-Raum



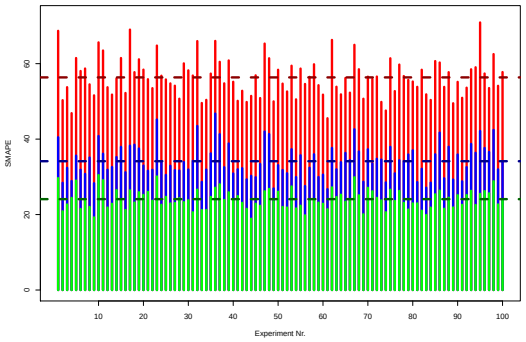
(b) global bester 1NN-Raum



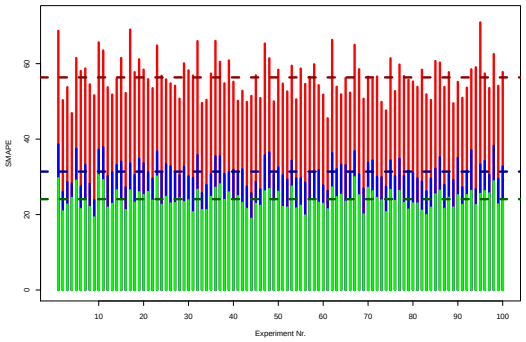
(c) TP1NN-Raum



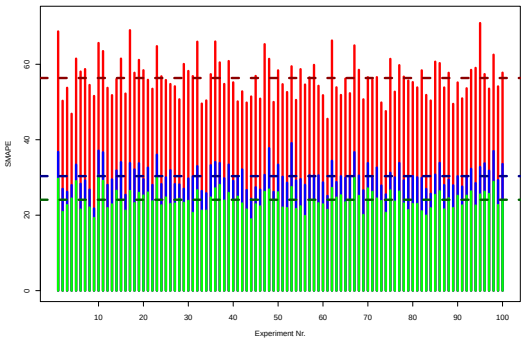
(d) Klassifikationsbaum



(e) **KNN!**-Klassifikator



(f) ESLS-Ansatz



(g) ELCSP-Ansatz

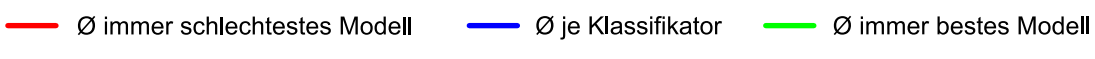
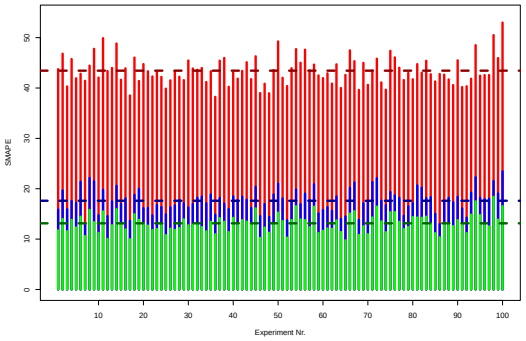
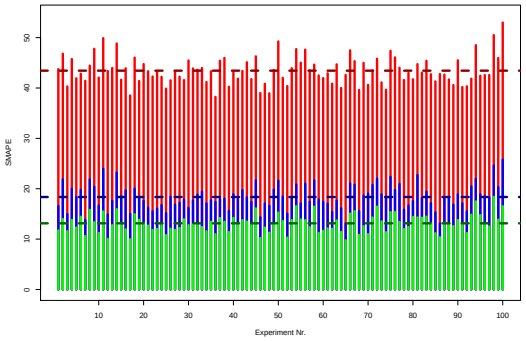


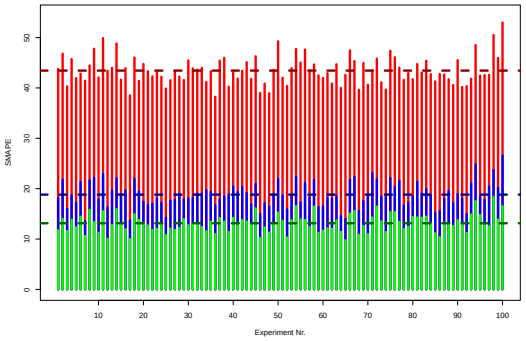
Abbildung A.6: SMAPE-Diagramme aller Klassifikatoren auf der Domäne *Industry*



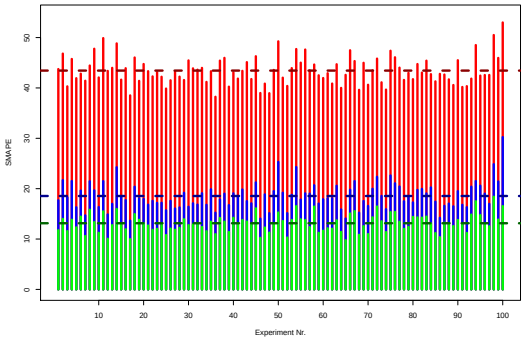
(a) bester domänenspezifischer 1NN-Raum



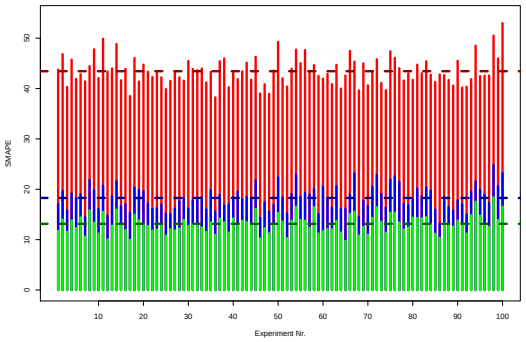
(b) global bester 1NN-Raum



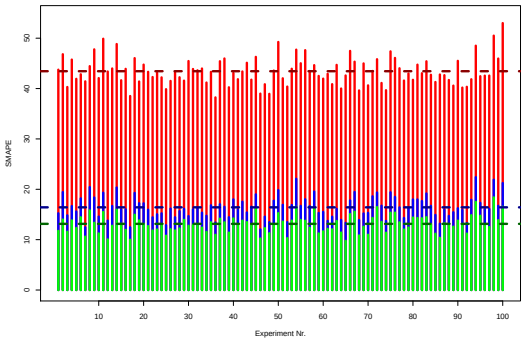
(c) TP1NN-Raum



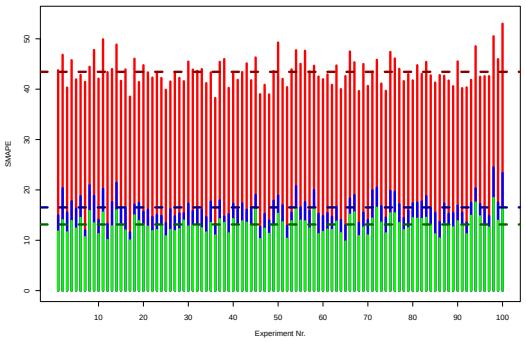
(d) Klassifikationsbaum



(e) **KNN!**-Klassifikator



(f) ESLS-Ansatz



(g) ELCSP-Ansatz

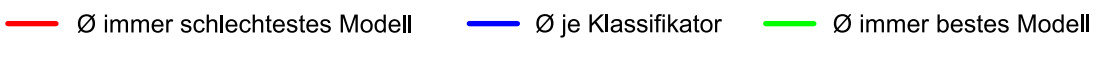
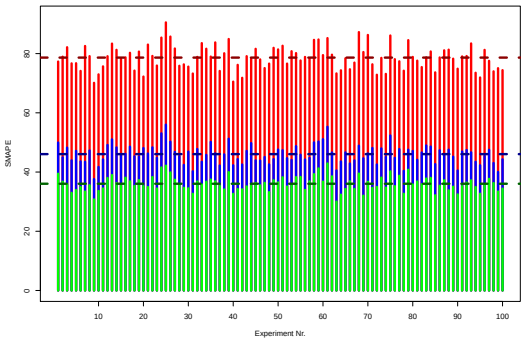
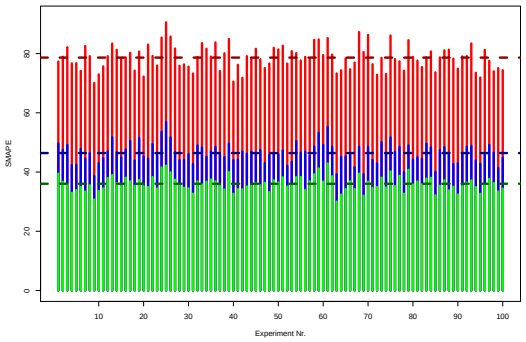


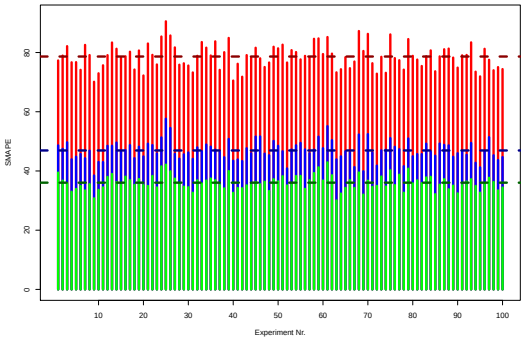
Abbildung A.7: SMAPE-Diagramme aller Klassifikatoren auf der Domäne *Macro*



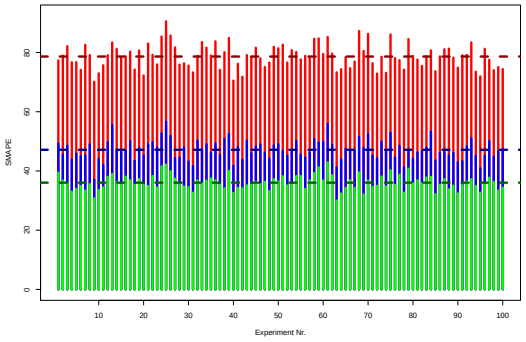
(a) bester domänenspezifischer 1NN-Raum



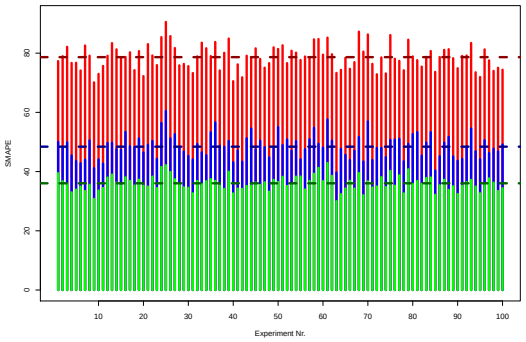
(b) global bester 1NN-Raum



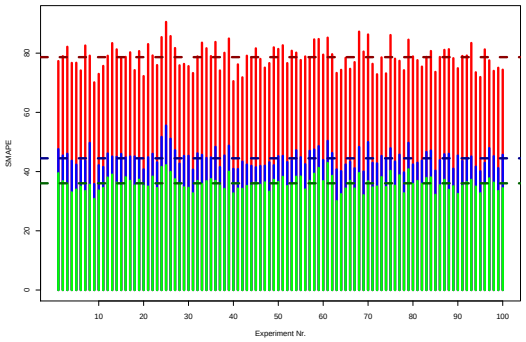
(c) TP1NN-Raum



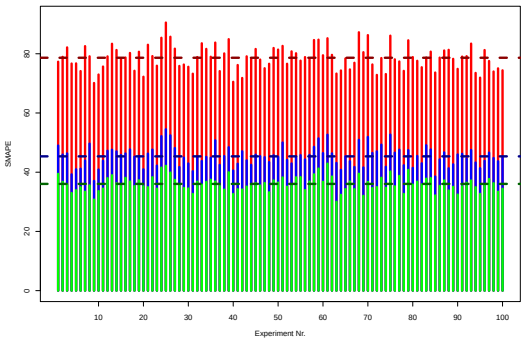
(d) Klassifikationsbaum



(e) KNN!-Klassifikator



(f) ESLS-Ansatz



(g) ELCSP-Ansatz

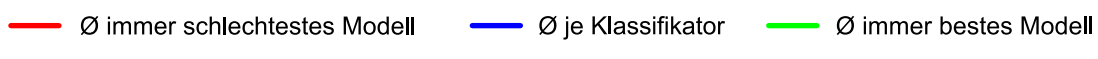
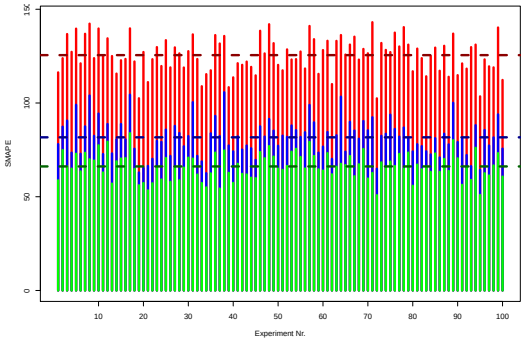
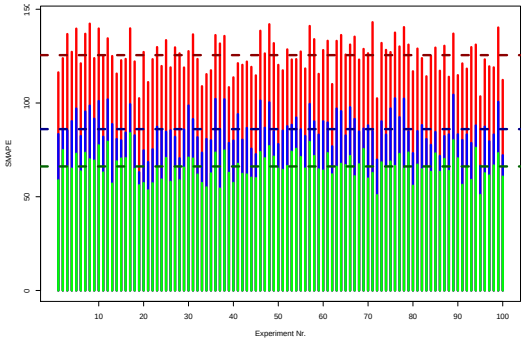


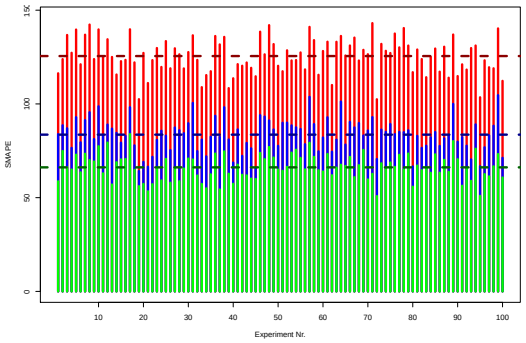
Abbildung A.8: SMAPE-Diagramme aller Klassifikatoren auf der Domäne *Micro*



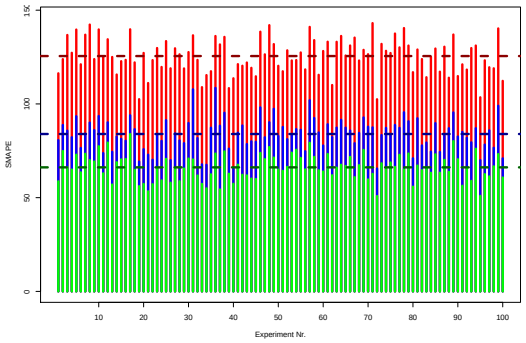
(a) bester domänenspezifischer 1NN-Raum



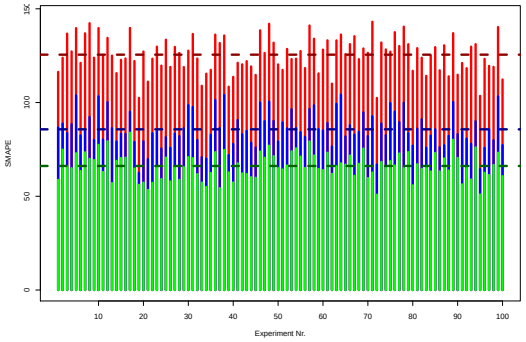
(b) global bester 1NN-Raum



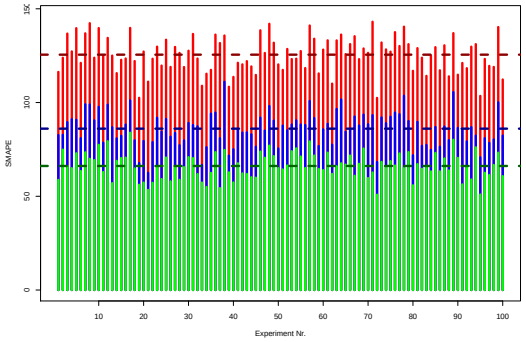
(c) TP1NN-Raum



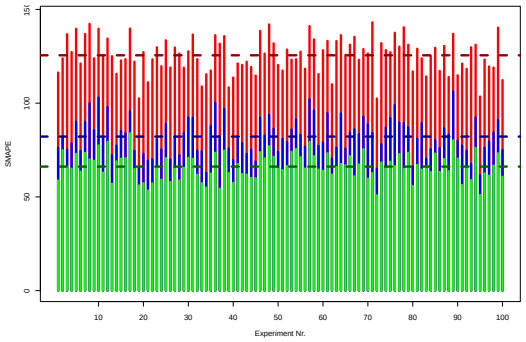
(d) Klassifikationsbaum



(e) **KNN!**-Klassifikator



(f) ESLS-Ansatz



(g) ELCSP-Ansatz

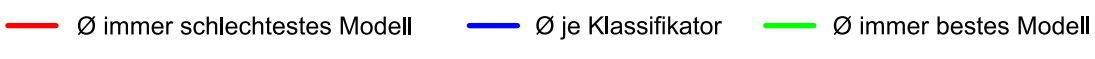


Abbildung A.9: SMAPE-Diagramme aller Klassifikatoren auf der Domäne *Other*