

Abteilung für Mathematik
Seminar für Statistik

Wintersemester 2000/01

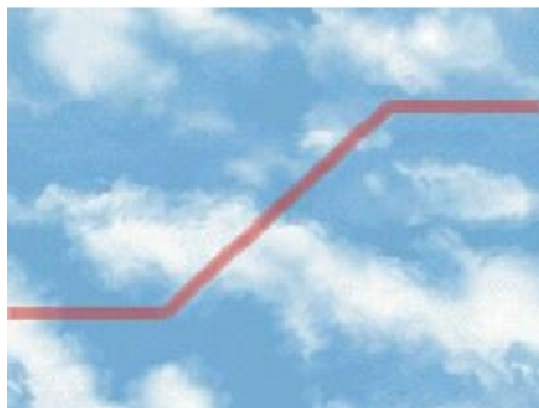
Diplomarbeit

Adrian Lüsi

Diplomprofessor: **Prof. Dr. Peter Bühlmann**
Betreuer: **Dr. Martin Mächler**
Dr. Peter Antal

Ausgabe: 13. November 2000
Abgabe: 23. März 2001

Modellierung von Wetterderivaten



An dieser Stelle möchte ich Herrn Prof. Dr. Peter Bühlmann für die Leitung dieser Diplomarbeit und seine wertvollen Hinweise herzlich danken. Ebenso bin ich Herrn Dr. Marin Mächler für seine sorgfältige Betreuung, sein Interesse und sein Engagement sehr zu Dank verpflichtet.

Herrn Dr. Peter Antal danke ich für die interessante angewandte Themenstellung und die Möglichkeit, dieses Projekt zu realisieren. Herrn Dr. Christian Straub danke ich für seine nützlichen Informationen und Hinweise betreffend Wetterderivate aus Sicht der Swiss Re. Den Mitarbeiterinnen und Mitarbeitern des Seminars für Statistik der ETH Zürich danke ich für die sehr angenehmen Arbeitsbedingungen.

Zusammenfassung

Die Wetterderivate bieten ein neuartiges derivatives Instrument zur Absicherung des Wetterrisikos. Meistens wird dabei die Temperaturdifferenz zu 65 F (Degree Days) als die entscheidende Einflussgrösse des Wetters auf das Unternehmensresultat betrachtet. Die Auszahlung eines Wetterderivats hängt von der Anzahl der kumulierten Degree Days über eine festgelegte Laufzeit ab. Damit ein Anbieter eines Wetterderivats (z.B. die Swiss Re) die Prämie festlegen kann, werden Kenntnisse der zukünftigen kumulierten Degree Days benötigt. Ein häufig verwendeter Ansatz ist die Untersuchung der kumulierten Degree Days der vergangenen Jahre. Diese Methode hat den Nachteil, dass nur etwa 35 Daten für die Analyse zur Verfügung stehen. In dieser Arbeit verfolgen wir die Modellierung der Temperaturzeitreihen. Damit bietet sich durch Simulation die Möglichkeit, eine beliebig grosse Anzahl von Realisationen der interessierenden kumulierten Degree Days zu erzeugen. Wir arbeiten hier mit den Datensätzen der 68 Standorte, für die die Swiss Re ihre Wetterderivate anbietet. Es handelt sich dabei um die mittleren täglichen Temperaturen seit 1965.

Unser Vorgehen besteht aus der Zerlegung der Temperaturzeitreihe in eine stationäre Reihe. Ein besonderes Augenmerk verdient dabei die Entfernung von Trend und Saisoneffekt. In Kapitel 3 vergleichen wir verschiedene Schätzmethoden für den Trend und Saisoneffekt. Dabei wählen wir aufgrund der Robustheitseigenschaften das auf dem Lowess basierende STL-Verfahren. In diesem Verfahren werden Trend und Saisoneffekt iterativ geschätzt. Eine besondere Eigenschaft ist die Veränderbarkeit der Saisonkomponente im Laufe der Zeit. Damit wird berücksichtigt, dass sich eine Klimaveränderung unterschiedlich auf die verschiedenen Jahreszeiten auswirken kann.

Nach der Entfernung von Trend und Saisoneffekt ist immer noch eine Saisonalität in den Residuen vorhanden. Diese rühren von der saisonalen Abhängigkeit der Standardabweichung her, die im Sommer kleinere Werte als im Winter annimmt. Im Kapitel 4 wird die Heteroskedastizität der Residuen beseitigt. Zudem muss das STL-Verfahren für heteroskedastische Residuen angepasst werden.

In Kapitel 5 analysieren wir die normierten Residuen. Dabei stellen wir eine schiefe Verteilung fest. Mit Hilfe einer Box-Cox Transformation beheben wir die Schiefe und erhalten annähernd normalverteilte Residuen.

Die durch die obigen Schritte erhaltene Residuen-Zeitreihe ist nun stationär. Jedoch besteht noch eine Abhängigkeitsstruktur. Diese wird in Kapitel 6 mit einem autoregressiven Prozess der Ordnung p modelliert. Die AR-Koeffizienten werden dabei mit dem Yule-Walker Verfahren geschätzt. Die AR-Ordnungswahl erfolgt mit Akaike's Information Criterion. Dieses Kriterium liefert aber manchmal unnötig hohe Ordnungen. Deshalb akzeptieren wir mit einer Schranke auch annähernd optimale tiefere Ordnungen.

In Kapitel 7 beschäftigen wir uns mit der Vorhersage der Temperaturzeitreihe. Dabei muss die in den Kapiteln 3-6 erhaltene Zerlegung ausgehend vom AR-Prozess rückgängig gemacht werden und für jeden Schritt eine Vorhersage erstellt werden. Zudem bestimmen wir die optimale Wahl der STL-Parameter, indem wir die Vorhersagegüte in früheren Jahren testen.

Im Kapitel 8 erstellen wir mithilfe eines Bootstrap-Verfahrens ein Vertrauensintervall für die optimale Vorhersage der Degree-Days.

Die Datenanalyse auf dem Computer erfolgte mit dem frei erhältlichen Statistikpaket R. Die für unsere Zwecke entwickelten Funktionen befinden sich im Anhang A.

Im Anhang C werden anhand von vier Stationen die einzelnen Schritte der Modellierung graphisch veranschaulicht.

Inhaltsverzeichnis

1	Einführung in die Wetterderivate	1
1.1	Sinn der Wetterderivate	1
1.2	Entwicklung der Wetterderivate	1
1.3	Wetter-Masse	2
1.4	Struktur eines Wetterderivats	2
1.5	Typen von Wetterderivaten	3
1.5.1	Degree-Day Floor	3
1.5.2	Degree-Day Cap	4
1.5.3	Degree-Day Collar	5
2	Modellwahl und Datenbasis	7
2.1	Problemstellung	7
2.2	Direkte Modellierung der Degree-Days	7
2.2.1	Anpassung einer Verteilung	7
2.2.2	Berechnung der Risikoprämie eines Wetterderivats	8
2.3	Modellierung der Temperaturzeitreihe	10
2.4	Datenbasis	10
2.4.1	Einteilung der Wetterstationen in Gruppen	11
3	Trend und Saisoneffekt	17
3.1	Der Trend	17
3.1.1	Lineare (oder polynomiale) Regression	18
3.1.2	Glättende Splines	18
3.1.3	Nadaraja-Watson-Kernschätzer	19
3.1.4	Der Lowess	20
3.1.5	Diskussion	23
3.2	Der Saisoneffekt	26
3.2.1	Bildung der Tagesmittelwerte	27
3.2.2	Schätzung des Saisoneffekts mit Fourieranalyse	29
3.2.3	Diskussion	33
3.3	Das STL-Verfahren	35
4	Die Heteroskedastizität der Residuen	43
4.1	Schätzung der saisonalen Standardabweichung	44
4.1.1	Die empirische Tages-Standardabweichung	44
4.1.2	Standardabweichungs-Schätzung mit Lowess	44
4.1.3	Schätzung der saisonalen Standardabweichung mit Fourieranalyse	46
4.1.4	Diskussion	48
4.2	Der modifizierte STL-Algorithmus	48
5	Die Box-Cox Transformation	53
6	Anpassung eines autoregressiven Modells	59
6.1	Die Autokorrelationsfunktion	59
6.2	AR-Modelle	61
6.3	Bestimmung der Parameter des AR-Modells	61
6.4	Wahl der AR-Ordnung	63
6.4.1	Wahl der AR-Ordnung mittels partieller Autokorrelationsfunktion	63
6.4.2	AR-Ordnungswahl mittels AIC	64
6.4.3	Güte des angepassten AR-Modells	68

7	Vorhersage der Temperaturzeitreihe	71
7.1	Vorhersage des AR-Prozesses	71
7.2	Rücktransformation des Residuenprozesses	73
7.2.1	Box-Cox Rücktransformation	73
7.2.2	Modellierung der Heteroskedastizität der Residuen	73
7.3	Vorhersage des Saisoneffekts	75
7.4	Vorhersage des Trends	75
7.5	Vorhersage der Temperatur	75
7.6	Wahl der optimalen STL-Parameter	78
8	Das Bootstrap-Verfahren	81
8.1	Der Bootstrap-Ansatz	81
8.2	Vertrauensintervall für $\hat{\theta}$	83
8.3	Vorhersageintervall für θ	85
8.4	Anwendung auf die Tarifierung eines Wetterderivats	85
9	Diskussion und Ausblick	87
A	Eigene R Funktionen	91
A.1	Einlesen und Bereinigen der Daten	91
A.1.1	Schaltjahrbereinigung	91
A.1.2	Umrechnung von Fahrenheit nach Celsius	91
A.1.3	Einlesen der Daten mit anschliessender Bereinigung	91
A.2	Schätzung von Trend und Saisoneffekt	92
A.2.1	Berechnung des Saisoneffekts mit Tagesmitteln	92
A.2.2	Berechnung des Saisoneffekts mit Fourieranalyse	93
A.2.3	Iterative Berechnung von Trend und Saisoneffekt	94
A.2.4	Berechnung von Trend und Saisoneffekt mit STL	96
A.3	Berücksichtigung der Heteroskedastizität der Residuen	97
A.3.1	Schätzung der saisonalen Standardabweichung	97
A.3.2	Modifiziertes STL-Verfahren	98
A.4	Box-Cox Transformation	100
A.4.1	Box-Cox Transformation durchführen	100
A.4.2	Optimales λ wählen	100
A.5	Zerlegung der Temperaturzeitreihe	101
A.6	Vorhersage der Temperatur	105
A.6.1	Vorhersage des Trends	105
A.6.2	Vorhersage der Temperatur	105
A.7	Bootstrap	106
A.7.1	Rücktransformation mit Bootstrap-Residuen	106
A.7.2	Berechnung der Bootstrap-Realisationen T^*	108
B	C-Code des modifizierten STL-Verfahrens	111
B.1	Hauptroutine	111
B.2	Bestimmung der Robustheitsgewichte	113
C	Beispiele der Temperaturzerlegung	115
	Literaturverzeichnis	124

Abbildungsverzeichnis

1.1	Heating Degree-Day Floor	4
1.2	Heating Degree-Day Cap	5
1.3	Heating Degree-Day Collar	6
2.1	Trend der HDD in New York	8
2.2	Histogramm der HDD nach Abzug des Trends von Vancouver und New York	9
2.3	Wetterstationen in Europa	13
2.4	Wetterstationen in den USA und in Kanada	14
2.5	Cluster Dendrogramm Korrelationen der Temperaturzeitreihen Europas . .	15
2.6	Cluster Dendrogramm der Korrelationen der Temperaturzeitreihen Amerikas	15
3.1	Tagestemperaturen in Zürich seit 1965	18
3.2	Veranschaulichung der lokalen Regression	21
3.3	Biquadratische Gewichtsfunktion $B(u)$	22
3.4	Konvergenzgeschwindigkeit des Lowess mit Robustheitsiterationen	22
3.5	Verschiedene Trendmethoden im Vergleich (Beispiel Zürich)	23
3.6	Vergleich der Trends in Amerika	25
3.7	Trends im Nordosten und Zentrum der USA	25
3.8	Histogramm der Temperaturen von Zürich	26
3.9	Geglättetes Spektrum von Zürich	27
3.10	Saisoneffekt mit Tagesmittel (geglättet und ungeglättet)	28
3.11	Vergleich des Saisoneffekts: arithmetisches Mittel und Median	29
3.12	Vergleich der Saisoneffekts für verschiedene k	31
3.13	Cook's Distanz des Modells $k = 2$	32
3.14	Funktion $\rho_c(u)$ des Huber-Schätzers	33
3.15	Saisoneffekt durch Bildung von Medianen (geglättet und ungeglättet) . . .	34
3.16	Spektrum nach Entfernung des Saisoneffekts	34
3.17	Lowess von $D_{152,j}^{(15)}$ mit Extrapolation der Jahre 1964 und 2001 (grün) . . .	36
3.18	Temperaturen, STL-Trend, STL-Saisoneffekt und Residuen von Zürich . . .	39
3.19	Geglätteter STL-Saisoneffekt von Zürich	40
3.20	Veränderung der Saisonkomponente in verschiedenen Monaten (Beispiel San Antonio)	41
4.1	Residuen R_t von San Antonio	43

4.2	Berechnung der saisonalen Standardabweichung mit Lowess (Beispiel San Antonio)	45
4.3	Vergleich der verschiedenen Schätzungsmethoden der Heteroskedastizität der Residuen	47
4.4	Normierte Residuen I_t von San Antonio	48
4.5	Modifizierter STL-Algorithmus: $(k + 1)$ -ter Durchgang	50
4.6	Unterschied STL / modifizierter STL in San Antonio	51
4.7	Unterschied STL / modifizierter STL in Zürich	51
5.1	QQ-Plot der normierten Residuen von Atlanta	54
5.2	Histogramm der normierten Residuen von Atlanta	54
5.3	Box-Cox Transformation für verschiedene Parameter λ	55
5.4	Log-Likelihoodfunktion für λ am Beispiel von Atlanta	57
5.5	QQ-Plot der normierten Residuen von Atlanta nach der Box-Cox Transformation	58
5.6	Histogramm der normierten Residuen von Atlanta nach der Box-Cox Transformation	58
6.1	Autokorrelationsfunktion der Residuen x_t von Zürich	60
6.2	PACF der Residuen x_t von Zürich	65
6.3	PACF der Residuen x_t von Vancouver	65
6.4	AIC-Differenz der Residuen von Zürich	67
6.5	AIC-Differenz der Residuen von Vancouver	67
6.6	ACF der AR-Residuen $\hat{\varepsilon}_t$	68
6.7	Geglättetes Spektrum der AR-Residuen $\hat{\varepsilon}_t$	69
6.8	Vergleich des Spektrums von x_t mit dem des angepassten $AR(p)$ (Zürich)	69
7.1	Beste Vorhersage (rot) und Prognoseintervall (grün) des $AR(p)$ -Modells	74
7.2	Trend in Zürich mit Vorhersage ab 1.9.1999	77
7.3	Vorhersage des Trends ab 1.9.1999 für zwei Jahre (Zürich)	77
7.4	Parameteroptimierung in Zürich	80
8.1	Histogramm der Bootstrap-Replikate $\hat{\theta}^{*(i)}$	84
8.2	95% Perzentilintervall für θ	86

Tabellenverzeichnis

3.1	Fourierkoeffizienten für verschiedene Werte von k (Zürich)	31
3.2	Koeffizienten bei robuster Regression für verschiedene Werte von k (Zürich)	33
4.1	Fourierkoeffizienten für verschiedene Werte von k (San Antonio)	47
6.1	AR-Koeffizienten von Zürich für verschiedene Ordnungen p	62

Kapitel 1

Einführung in die Wetterderivate

1.1 Sinn der Wetterderivate

Viele Unternehmen leiden unter wetterbedingten finanziellen Verlusten, die sie wegen der volatilen Natur des Wetters nicht kontrollieren können. Experten schätzen, dass rund 20% (2000 Mrd. \$) des amerikanischen Bruttoinlandprodukts Wettereinflüssen ausgesetzt sind und etwa 1 Trillion Dollar von den 7 Trillionen Dollar der U.S. Wirtschaft wettersensitiv ist. Abhängig von Wetter sind z.B die Versicherungen, die Energieunternehmen, das Baugewerbe, die Landwirtschaft und der Detailhandel.

Diese Unternehmen möchten sich gegen das Wetterrisiko absichern, um ein stabileres Jahresresultat zu erhalten. Ein solches Weather Risk Management kann durch den Einsatz von **Wetterderivaten** betrieben werden.

1.2 Entwicklung der Wetterderivate

Der Handel mit Wetterderivaten ist noch ein junger Markt. Die erste Transaktion mit Wetterderivaten fand im Jahre 1997 statt. Es wird geschätzt, dass seither 1600 Geschäfte mit Wetterderivaten im Wert von 3.5 Milliarden Dollar getätigt wurden.

Im September 1999 lancierte die Chicago Mercantile Exchange (CME) die ersten börsengehandelten Wetter-Futures. Zusätzlich gibt es mehrere Wetterderivat-Broker, die beim Abschliessen von "over the counter" (OTC) Geschäften assistieren.

Bis vor kurzem konzentrierte sich der Handel mit Wetterderivaten auf die USA. Dank der Deregulierung des Energiesektors, den Fortschritten des elektronischen Handels und dem zunehmenden Interesse der Banken und der Versicherungsindustrie breitet sich das Geschäft mit Wetterderivaten allmählich auch in Europa aus. Seit Mai 2000 können über das System ELRiX der Swiss Re Wetterderivate online gehandelt werden. Dieser Vertriebskanal bietet die Möglichkeit, die Geschäfte transparenter und effizienter abzuwickeln, sowie die bis anhin schwer erreichbaren Kleinunternehmen anzuziehen.

1.3 Wetter-Masse

Damit ein Wetterderivat das Wetterrisiko eines Unternehmens absichern kann, muss ein Wetter-Mass eingeführt werden. Dieses Mass soll den Einfluss des Wetters auf den Unternehmensgewinn möglichst gut repräsentieren.

Wir wollen als Beispiel einen Stromproduzenten betrachten. Der Stromverbrauch hängt stark von den Lufttemperaturen ab. Eine Temperatur von 65 F ($\cong 18.3^\circ$ C) wird als angenehme Raumtemperatur empfunden. Diese 65 F nennen wir **Referenztemperatur**. Man trifft die Annahme, dass die Haushalte in den USA bei Temperaturen über 65 F die Klimaanlage einschalten und bei Temperaturen unter 65 F zu heizen beginnen. Wir sehen also, dass mit der Zunahme der Abweichung zu 65 F auch der Stromverbrauch steigt. Deshalb korreliert der Unternehmensgewinn stark mit der Grösse der Temperaturabweichung.

Um das Risiko dieser Temperaturabweichung zu erfassen, wird das Konzept der Degree-Days eingeführt.

Die **Degree Days (DD)** messen die Abweichung der mittleren Tages-Temperatur zur Referenztemperatur. Wie oben erläutert, bilden die Degree Days eine geeignete Masseinheit für einen Wetterderivatsvertrag.

Liegt die mittlere Temperatur eines Tages unter 65 F, spricht man von einem **Heating Degree-Day** oder Heizgradtag. Die Temperaturdifferenz zur Referenztemperatur an einem solchen Tag bezeichnen wir mit **HDD**.

$$\text{HDD} = \max(0, \text{Referenztemp.} - \text{mittlere tägliche Temp.})$$

Wenn die mittlere Temperatur grösser als 65 F ist, handelt es sich um einen **Cooling Degree-Day** oder Kühlgradtag. Die Temperaturdifferenz zur Basistemperatur an einem solchen Tag nennen wir **CDD**.

$$\text{CDD} = \max(0, \text{mittlere tägliche Temp.} - \text{Referenztemp.})$$

Weist ein Tag zum Beispiel eine mittlerer Temperatur von 46 F auf, so hat er 19 HDD und 0 CDD. Ein Tag mit mittlerer Temperatur 72 F hat 7 CDD und 0 HDD.

In Europa und Kanada nimmt man als Referenztemperatur 18° C (was nicht genau 65 F entspricht).

Es auch möglich, die Feuchtigkeit, Niederschlags- oder Schneefallmenge als Wettermass zu nehmen, jedoch werden die Degree-Days in 98%-99% aller Verträge verwendet. Bei dieser Arbeit betrachten wir nur Verträge mit Degree-Days.

1.4 Struktur eines Wetterderivats

Folgende Elemente charakterisieren ein Wetterderivat:

- Man legt die dem Wetterderivat zugrundeliegend Variable (**Index**) fest: HDD oder CDD.

- Es wird eine **Periode** festgelegt, die das Wetterderivat absichern soll. Häufig nimmt man die Periode vom 1. November bis zum 31. März (HDD Saison) und die Periode vom 1. Mai bis zum 30. September (CDD Saison). Über diese Laufzeit wird die tägliche Anzahl der HDD bzw. CDD kumuliert, d.h.

$$\text{kumulierte DD} = \sum_{i \in \text{Periode}} \max(\delta \cdot (T_i - T_{Ref}), 0)$$

wobei T_i die mittlere Temperatur des Tages i ist und T_{Ref} die Referenztemperatur. Für CDD wird δ auf 1 gesetzt und für HDD gilt $\delta = -1$.

- Es muss eine Wetterstation bestimmt werden, auf deren Temperaturen sich das Wetterderivat bezieht. Meist befindet sich diese Station in einer grossen Stadt.
- Es wird der **Strike** festgesetzt: dies ist die Anzahl kumulierter HDD bzw. CDD, ab denen die Wetteroption wirksam wird.
- Für den Wert eines HDD bzw. CDD wird ein Geldbetrag (**Tick**) vereinbart.
- Die Grenze (**Limit**) gibt die maximale Anzahl HDD bzw. CDD an, die vom Vertrag gedeckt werden.

Beispiel:

Ort	New York / La Guardia
Index	HDD
Periode	1. November 1999-31. März 2000
Tick	\$20'000 pro HDD
Strike	3440 HDD
Limit	300 HDD
Maximale Auszahlung	\$6 Millionen (=Tick·Limit)

1.5 Typen von Wetterderivaten

Es gibt verschiedene Typen von Wetterderivaten, die auf die unterschiedlichen Bedürfnisse eines Kunden ausgerichtet sind. Wir betrachten im Folgenden die gebräuchlichsten Typen:

1.5.1 Degree-Day Floor

Beim Degree-Day Floor wird der Käufer ausbezahlt, wenn die Anzahl der kumulierten HDD bzw. CDD *unter* dem Strike liegt. Die Auszahlung berechnet sich nach folgender Formel:

$$\text{Auszahlung} = \min(\text{Tick} \cdot \max(\text{Strike} - \text{kumulierte DD}, 0), \text{maximale Auszahlung})$$

Die Auszahlung hängt also wesentlich von den kumulierten HDD/CDD ab. Diese sind das zufällige Element im Vertrag. Wenn wir die Angabe des obigen Beispiels verwenden, erhalten wir die in Abbildung 1.1 dargestellte, von den kumulierten HDD abhängige Auszahlung:

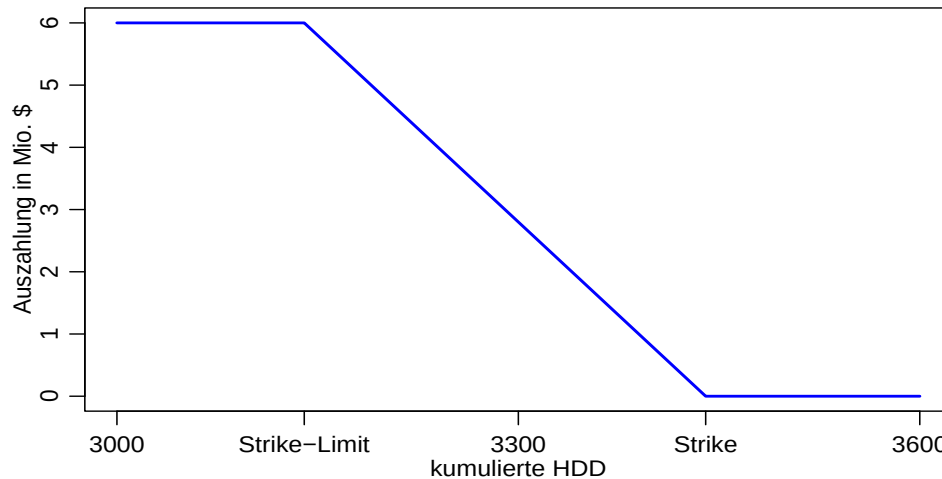


Abbildung 1.1: Heating Degree-Day Floor

Der Käufer dieses Vertrages möchte sich gegen einen zu warmen Winter absichern. Ein Energieunternehmen erleidet umso mehr Umsatzverluste, je wärmer der Winter ist (es wird weniger geheizt). Dafür erhält das Unternehmen durch den Heating Degree-Day Floor in diesem Fall umso mehr Geld ausbezahlt. Auf diese Weise können die Umsatzverluste kompensiert werden. Der Tick wird so festgesetzt, dass er dem Verlust entspricht, der durch ein HDD weniger entsteht. Analog wird dieses Unternehmen im Sommer einen Cooling Degree-Day Floor abschliessen (Klimaanlagen).

1.5.2 Degree-Day Cap

In einem solchen Vertrag wird der Käufer ausbezahlt, wenn die Anzahl der kumulierten HDD bzw. CDD über dem Strike liegt. Der Käufer erhält folgenden Betrag:

$$\text{Auszahlung} = \min(\text{Tick} \cdot \max(\text{kumulierte DD} - \text{Strike}, 0), \text{maximale Auszahlung})$$

Wir verwenden wiederum die Daten aus dem Beispiel, aber mit dem Strike bei 3900 HDD. Die Abbildung 1.2 zeigt die Auszahlungsfunktion eines Degree-day Caps, der auf diesen Daten basiert.

Der Käufer eines Degree-Day Caps möchte sich gegen zu kalte Winter (Heating Degree-Day Cap) bzw. zu heiße Sommer (Cooling Degree-Day Cap) absichern. Ein Cooling Degree-Cap ist zum Beispiel für ein Kinounternehmen attraktiv. In einem heißen Sommer hat ein Kino weniger Besucher, da die Leute sich dann anderen Freizeitaktivitäten widmen. Mit einem Cooling Degree-Cap kann aber der Verlust, der durch geringe Besucherzahlen entsteht, ausgeglichen werden, denn ein solches Wetterderivat hat in einem heißen Sommer die höchste Auszahlung.

Bei den oben vorgestellten Wetterderivaten profitiert der Käufer einseitig vom Vertrag.

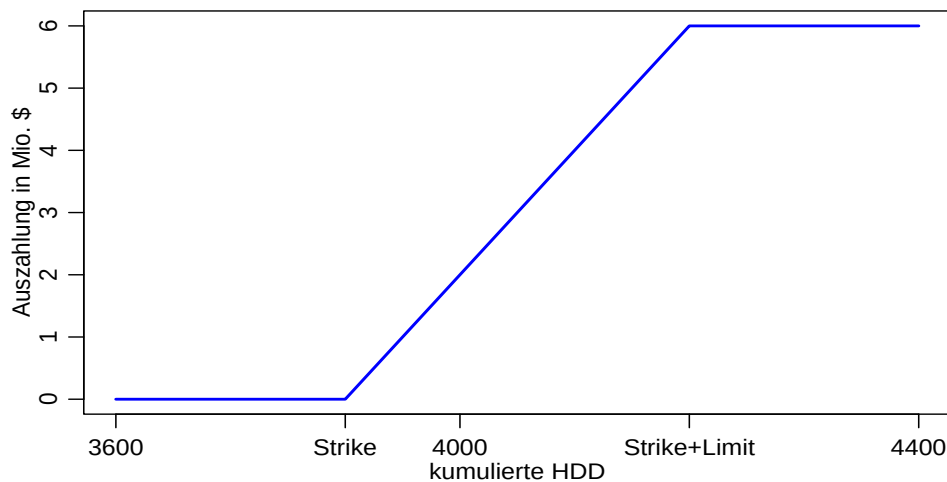


Abbildung 1.2: Heating Degree-Day Cap

Er kann Gewinne erzielen, jedoch keine Verluste. Deshalb fordert der Verkäufer eine angemessene **Prämie** für das Wetterderivat.

1.5.3 Degree-Day Collar

Bei einem Collar wird zugleich ein Floor gekauft und ein Cap verkauft (oder ein Floor verkauft und ein Cap gekauft).

Wir betrachten den gleichzeitigen Kauf des Floors aus dem Beispiel von Abschnitt 1.5.1 und des Caps aus Abschnitt 1.5.2. Das ergibt die in der Tabelle stehenden Daten.

Ort	New York / La Guardia
Index	HDD
Periode	1. November 1999-31.März 2000
Tick	\$20'000 pro HDD
Strike floor (tiefer)	3440 HDD
Strike cap (höher)	3900 HDD
Limit	300 HDD
Maximale Auszahlung	\$6 Millionen (=Tick·Limit)

In Abbildung 1.3 ist die Auszahlung des Collars in Abhängigkeit der kumulierten HDD aufgetragen.

Betrachten wir ein Energieunternehmen, das den obigen Collar besitzt. Wenn das Wetter im Winter ungewöhnlich kalt ist, macht die Firma mehr Gewinn. Ein Teil davon muss für die Deckung des verkauften Caps verwendet werden, der unter diesen Bedingungen an Wert gewinnt. Es wird also das Gewinnpotenzial nach oben geschmälert. Jedoch wird die Prämie, die die Firma für den verkauften Cap erhält, in einen Floor investiert. Der schützt

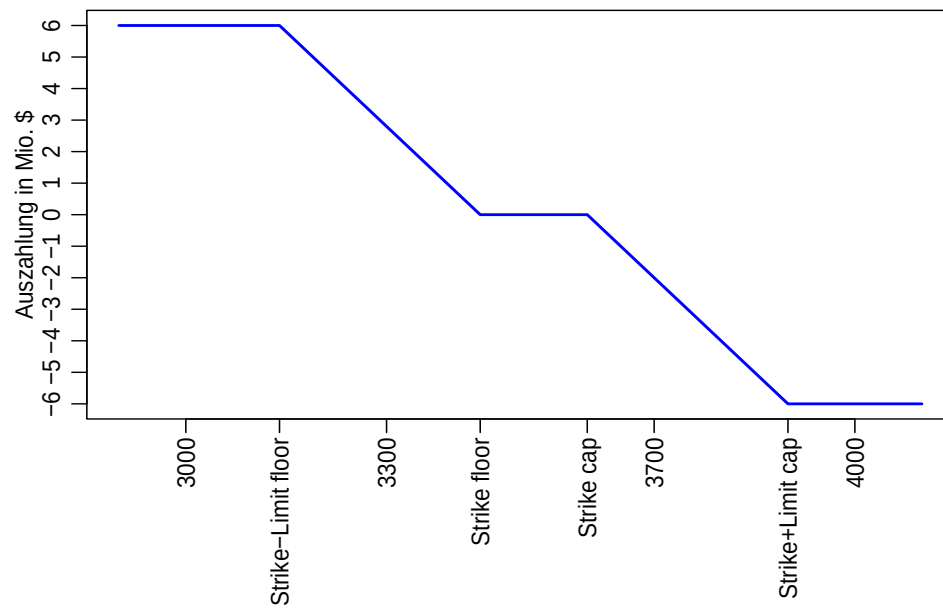


Abbildung 1.3: Heating Degree-Day Collar

die Unternehmung vor Verlusten in milden Wintern. Ein Collar begrenzt also sowohl das Gewinnpotenzial nach oben als auch das Verlustpotenzial nach unten. Wenn der Wert des Caps dem des Floor entspricht, handelt es sich um einen *Costless Collar*. Es muss also keine Prämie für den Erwerb des Collars bezahlt werden.

Kapitel 2

Modellwahl und Datenbasis

2.1 Problemstellung

In Kapitel 1 wurde festgestellt, dass der Wert, den ein Wetterderivat am Ende seiner Laufzeit besitzt, von den täglichen Temperaturen während dieser Zeit abhängt. Diese sind aber beim Abschluss des Wetterderivats noch nicht bekannt.

Trotzdem muss bei Vertragsabschluss für das Wetterderivat eine Prämie festgelegt werden. Die Prämie P ist wie folgt zusammengesetzt:

$$P = \mathbf{E}[V] + K + M.$$

Dabei bedeutet V die (stochastische) Auszahlung des Wetterderivats, K die Fixkosten des Wetterderivat-Anbieters und M eine Gewinnmarge.

In der vorliegenden Arbeit interessieren wir uns nur für die Risikoprämie $\mathbf{E}[V]$, d.h. die erwartete Auszahlung.

Um die erwartete Auszahlung zu berechnen, benötigen wir Kenntnisse über die Verteilung der kumulierten Degree-Days. In dieser Arbeit gilt es, diese Verteilung zu modellieren.

Ein häufig verwendeter Ansatz ist die Anpassung einer parametrischen Verteilung an die kumulierten Degree-Days früherer Jahre. Dieser wird im nächsten Abschnitt behandelt. Danach präsentieren und begründen wir unseren Ansatz zur Modellierung der kumulierten Degree-Days.

2.2 Direkte Modellierung der Degree-Days

2.2.1 Anpassung einer Verteilung

Bei diesem Vorgehen betrachten wir die in vergangenen Jahren realisierten Degree-Days in der selben Periode. An diese Daten wird mit einer Regression ein linearer Trend angepasst (Abbildung 2.1). Da dieser Trend nicht beliebig weit in die Vergangenheit dieselbe lineare Form besitzt, werden nur die letzten 25 Jahre betrachtet.

An die um den Trend bereinigten Degree-Days wird nun eine parametrische Verteilung angepasst. Meist wird dazu die Normalverteilung $\mathcal{N}(\mu, \sigma^2)$ gewählt.

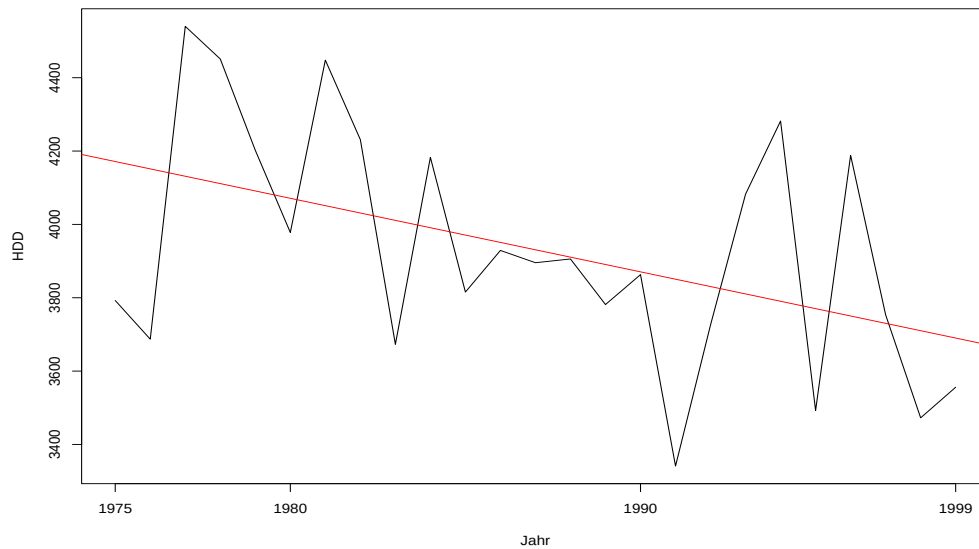


Abbildung 2.1: Trend der HDD in New York

Der Erwartungswert μ für das nächste Jahr wird durch die Fortsetzung der linearen Regression geschätzt.

Die Varianz σ^2 wird mit den um den Trend bereinigten Residuen der letzten 25 Jahre geschätzt. In Abbildung 2.2 sehen wir das Histogramm der um den Trend bereinigten HDD von Vancouver und New York.

In New York und Vancouver wurden die folgenden Parameter geschätzt:

	μ	σ
New York	3670 HDD	287
Vancouver	9003 HDD	129

Wir sehen, dass die Standardabweichung in New York mehr als doppelt so gross wie jene von Vancouver ist. Im folgenden Abschnitt sehen wir, wie sich die Normalverteilungsparameter auf die Preisbildung eines Wetterderivats auswirken.

2.2.2 Berechnung der Risikoprämie eines Wetterderivats

Unter der Normalverteilungsannahme kann die Risikoprämie analytisch berechnet werden. Dazu verwenden wir die folgende Notation:

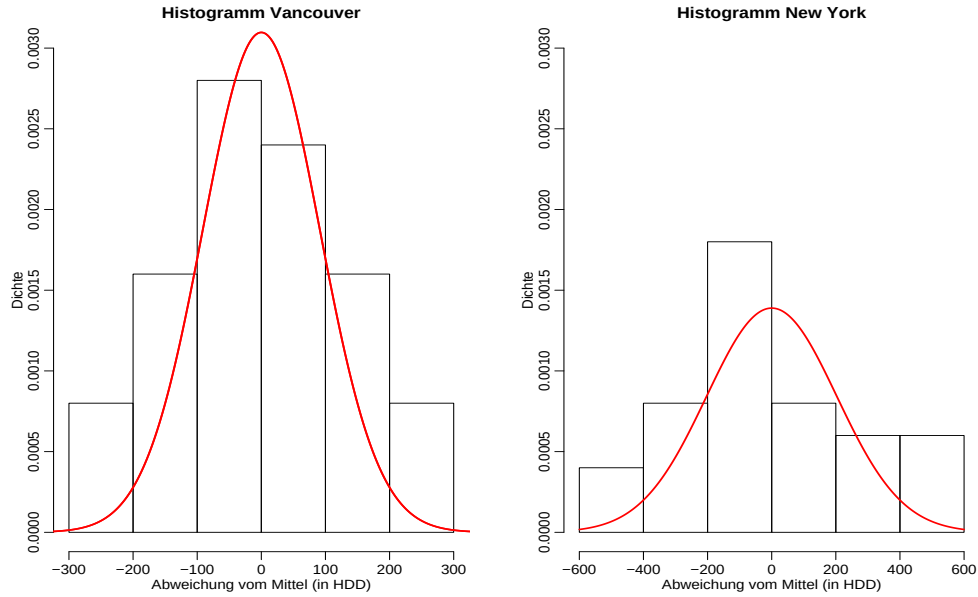


Abbildung 2.2: Histogramm der HDD nach Abzug des Trends von Vancouver und New York. Die rote Kurve ist die angepasste Normalverteilung.

- z : Kumulierte DD
- μ : Erwartungswert der kumulierten DD
- σ : Standardabweichung der kumulierten DD
- θ : Tick
- K : Strike
- L : Limit
- $\varphi(\cdot)$: Dichtefunktion der Normalverteilung $\mathcal{N}(\mu, \sigma^2)$
- $\Phi(\cdot)$: Kumulative Verteilungsfunktion der Standard-Normalverteilung

Im Falle eines *Degree-Day Caps* ist die erwartete Auszahlung wie folgt gegeben:

$$\begin{aligned}
 \mathbf{E}[V_{cap}] &= \int_{-\infty}^{\infty} \theta \cdot \min(\max(z - K, 0), K + L) \cdot \varphi(z) \, dz \\
 &\stackrel{NV}{=} \int_K^{K+L} \theta \cdot (z - K) \cdot \frac{1}{\sqrt{2\pi}\sigma} \cdot e^{-\frac{1}{2}\left(\frac{z-\mu}{\sigma}\right)^2} \, dz \\
 &\stackrel{Subst.}{=} \theta \cdot \left((\mu - K) \cdot \left(\Phi\left(\frac{K+L-\mu}{\sigma}\right) - \Phi\left(\frac{K-\mu}{\sigma}\right) \right) + \sigma^2 \cdot \left(\varphi(K) - \varphi(K+L) \right) \right)
 \end{aligned}$$

Eine analoge Formel kann für einen *Degree-Day Floor* hergeleitet werden:

$$\begin{aligned}
 \mathbf{E}[V_{floor}] &= \int_{-\infty}^{\infty} \theta \cdot \min(\max(K - z, 0), K - L) \cdot \varphi(z) \, dz \\
 &\stackrel{NV}{=} \int_{K-L}^K \theta \cdot (K - z) \cdot \frac{1}{\sqrt{2\pi}\sigma} \cdot e^{-\frac{1}{2}\left(\frac{z-\mu}{\sigma}\right)^2} \, dz \\
 &\stackrel{Subst.}{=} \theta \cdot \left((K - \mu) \cdot \left(\Phi\left(\frac{K-\mu}{\sigma}\right) - \Phi\left(\frac{K-L-\mu}{\sigma}\right) \right) + \sigma^2 \cdot \left(\varphi(K) - \varphi(K-L) \right) \right)
 \end{aligned}$$

Mit dieser Formel wollen wir nun die Riskiopremie eines HDD-Floors von New York berechnen. Dazu verwenden wir die selben Daten wie schon in Kapitel 1.3:

δ	\$20'000 pro HDD
K	3440 HDD
L	300 HDD

Für diesen Vertrag muss nach obiger Formel eine Risikoprämie von \$421'127 gefordert werden.

2.3 Modellierung der Temperaturzeitreihe

Das oben beschriebene Vorgehen zur Modellierung der Degree-Days besteht durch seine Einfachheit. Jedoch kann die Parameterschätzung aufgrund der wenigen verfügbaren Daten ungenau werden. Auch scheint die Annahme der Normalverteilung nicht an allen Orten zuzutreffen.

Abhilfe kann hier die direkte Modellierung der Temperaturzeitreihen schaffen. Dazu versuchen wir, die täglichen Temperaturen¹ mittels eines stochastischen Prozesses zu modellieren. Mit diesem Prozess wird dann eine Vorhersage für die aktuelle Vertragsperiode aufgestellt. Erst anschliessend werden die kumulierten Degree-Days gebildet.

Die genaue Vorgehensweise bei diesem Ansatz setzt sich aus folgenden Punkten zusammen:

1. Die Temperaturzeitreihe wird soweit zerlegt, bis sie stationär ist. Diese Zerlegung soll so gewählt werden, dass sie rückwärts durchführbar ist. Zudem muss darauf geachtet werden, dass bei allen Zerlegungsschritten eine vernünftige Vorhersage möglich ist, da wir an Temperaturprognosen interessiert sind.
2. An die stationären Residuen soll dann ein stochastischer Prozess angepasst werden.
3. Dieser stochastische Prozess wird mittels eines Bootstrap-Verfahrens simuliert, wobei die in Punkt 1 durchgeführten Zerlegungsschritte rückgängig gemacht werden. Für jeden Schritt wird eine Prognose über den Vorhersagezeitraum aufgestellt.

Die Modellierung der täglichen Temperaturen bietet einen besseren Einblick in das Verhalten der Temperaturen als die direkte Vorhersage der kumulierten HDD/CDD. Zudem ist die Datenbasis, um den Prozess zu modellieren und um Vorhersagen aufzustellen, um ein Vielfaches grösser. Die Temperaturzeitreihe bildet die Grundlage für die kumulierten HDD/CDD. Deshalb verfolgen wir in der vorliegenden Arbeit diesen Ansatz.

2.4 Datenbasis

Bei dieser Untersuchung stehen uns von 68 Wetterstationen die täglichen Temperaturen während 35 Jahren zur Verfügung. Für diese Standorte in Europa, Kanada und der USA

¹Als tägliche Temperatur wird in dieser Arbeit die mittlere Temperatur des Tages bezeichnet. Sie ist das arithmetische Mittel aus der maximalen und minimalen Temperatur des Tages.

bietet die Swiss Re Wetterderivate an. In den Abbildungen 2.3 und 2.4 sind diese Stationen eingezeichnet.

Beim den Wetterstationen in Europa und Kanada wird die Temperatur in Grad Celsius angegeben, in den USA in Fahrenheit.

Bei neun Orten gibt es fehlende Werte. Bei den Orten Bakersfield, Charleston, Colorado-Springs, Houston, Oklakoma City, Topeka und Tulsa fehlen jeweils ein Monat. In Montreal und Savannah fehlen ganze drei Jahre am Stück.

Diese fehlenden Daten sind vorallem in den 90er Jahren anzutreffen. Deshalb kann man mit der Zeitreihenanalyse nicht erst nach den fehlenden Daten beginnen. Eine Möglichkeit wäre die Auffüllung der Lücken aufgrund des Temperaturzeitreihenmodells der restlichen Daten. Diese Variante würde jedoch eine spätere Modellanpassung verfälschen.

Eine bessere Möglichkeit wäre die Anpassung der Temperaturdaten eines nahegelegenen Ortes (mit hoher Korrelation) an die Temperaturzeitreihe. Dadurch könnten die Lücken aufgefüllt werden.

Da für diese Arbeit ohnehin sehr viele Datensätze zur Verfügung stehen, haben wir auf die Ersetzung von fehlenden Werten verzichtet.

2.4.1 Einteilung der Wetterstationen in Gruppen

Wir wollen die zu analysierenden Wetterstationen in Gruppen mit ähnlichen klimatologischen Eigenschaften einteilen. Dazu betrachten wir die Korrelation zwischen den Temperaturzeitreihen der verschiedenen Orte. Zur Gruppen-Einteilung bilden wir ein *Hierarchisches Cluster*. Dabei bildet zu Beginn jede Station eine eigene Gruppe. Danach geht die Cluster-Bildung iterativ vonstatten. In jedem Schritt werden die beiden Gruppen mit dem kleinsten Abstand voneinander zu einer neuen Gruppe zusammengefasst. Als Abstandsmass dient in unserem Fall

$$dist(G1, G2) = \max_{x \in G1, y \in G2} (1 - cor(x, y)),$$

wobei $G1$ und $G2$ zwei Gruppen sind.

Diese Cluster-Bildung wurde separat für die Standorte in Europa (2.5) und jene in Amerika (2.6) durchgeführt.

Dabei ergeben sich für Europa folgende Gruppen:

- Nordeuropa
- Grossbritannien
- Zentraleuropa
 - südlicher Teil Zentraleuropas (Frankreichs)
 - nördlicher Teil Zentraleuropas

In Amerika werden folgende Gruppen gebildet:

- Südosten der USA
- Nordosten und Zentrum Amerikas

- Südlicher Teil des Gebiets um die Appalachen
 - Nördliche Teil des Gebietes um die Appalachen
 - Zentrum der USA
- Westen Amerikas
 - Nördlicher Teil
 - Südlicher Teil
 - Edmonton
 - San Francisco

In Amerika ist eine klare Trennung zwischen dem Gebiet östlich und westlich der Rocky Mountains zu erkennen.

Im Westen Amerikas bilden Edmonton und San Francisco eine eigene Gruppe. Edmonton liegt als einzige Station im innern Kanadas. Ebenso ist San Francisco ist der einzige Ort an der (südlichen) Westküste der USA.



Abbildung 2.3: Wetterstationen in Europa

<i>Finland</i>	<i>Norwegen</i>
1 Helsinki	13 Bergen
	14 Oslo
<i>Frankreich</i>	<i>Schweden</i>
2 Bordeaux	15 Gotheburg
3 Lille	16 Stockholm
4 Lyon	
5 Marseille	<i>Schweiz</i>
6 Paris	17 Zürich
7 Strasbourg	
8 Toulouse	<i>Grossbritannien</i>
<i>Deutschland</i>	18 Belfast
9 Berlin	19 Birmingham
10 Köln	20 Glasgow
11 Hamburg	21 Leeds
12 Stuttgart	22 London-Heathrow
	23 Manchester

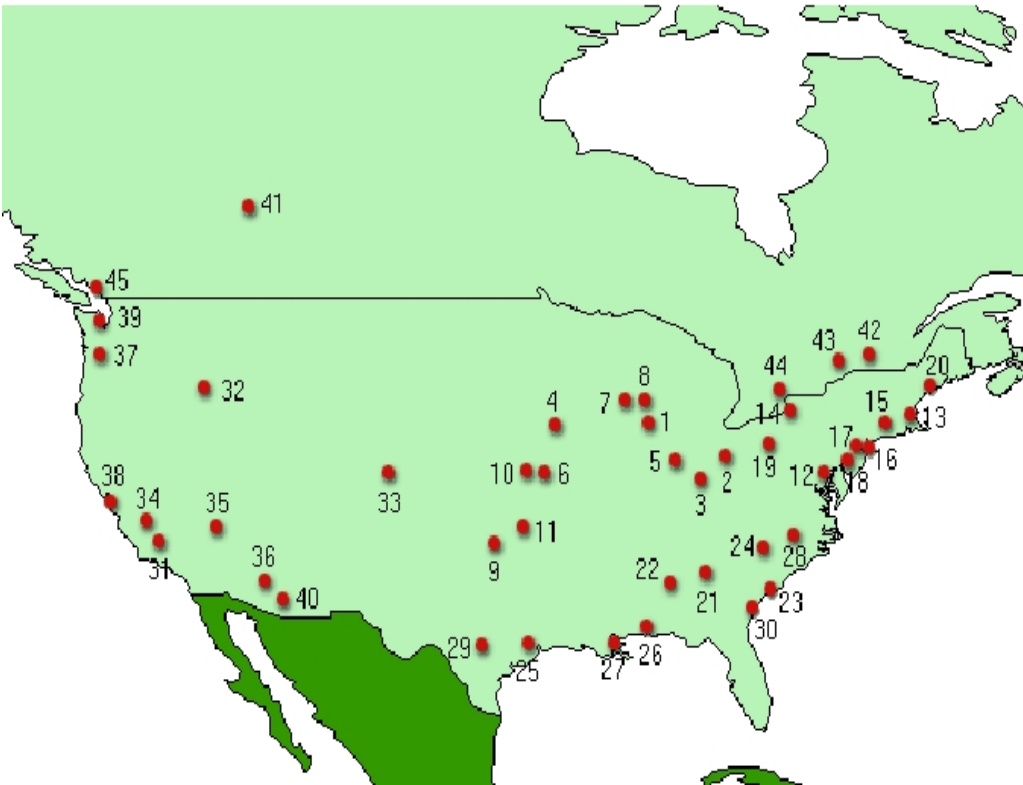


Abbildung 2.4: Wetterstationen in den USA und in Kanada

	<i>USA, Zentral</i>	16	New York, NY- La Guardia AP		<i>USA, Westen</i>
1	Chicago, IL - O'Hare	17	Newark, NJ	31	Bakersfield, CA
2	Columbus, OH	18	Philadelphia, PA	32	Boise, ID
3	Covington, KY	19	Pittsburgh, PA	33	Colorado Springs, CO
4	Des Moines, IA	20	Portland, ME	34	Fresno, CA
5	Indianapolis, IN			35	Las Vegas, NV
6	Kansas City, MO		<i>USA, Südosten</i>	36	Phoenix, AZ
7	Madison, WI	21	Atlanta, GA	37	Portland, OR
8	Milwaukee, WI	22	Birmingham, AL	38	San Francisco, CA
9	Oklahoma City, OK	23	Charleston, SC	39	Seattle, WA
10	Topeka, KS	24	Charlotte, NC	40	Tucson, AZ
11	Tulsa, OK	25	Houston, TX - Hobby AP		
		26	Mobile, AL		<i>Kanada</i>
	<i>USA, Nordosten</i>	27	New Orleans, LA	41	Edmonton
12	Baltimore, MD	28	Raleigh, NC	42	Montreal
13	Boston, MA	29	San Antonio, TX	43	Ottawa
14	Buffalo, NY	30	Savannah, GA	44	Toronto
15	Hartford, CT			45	Vancouver

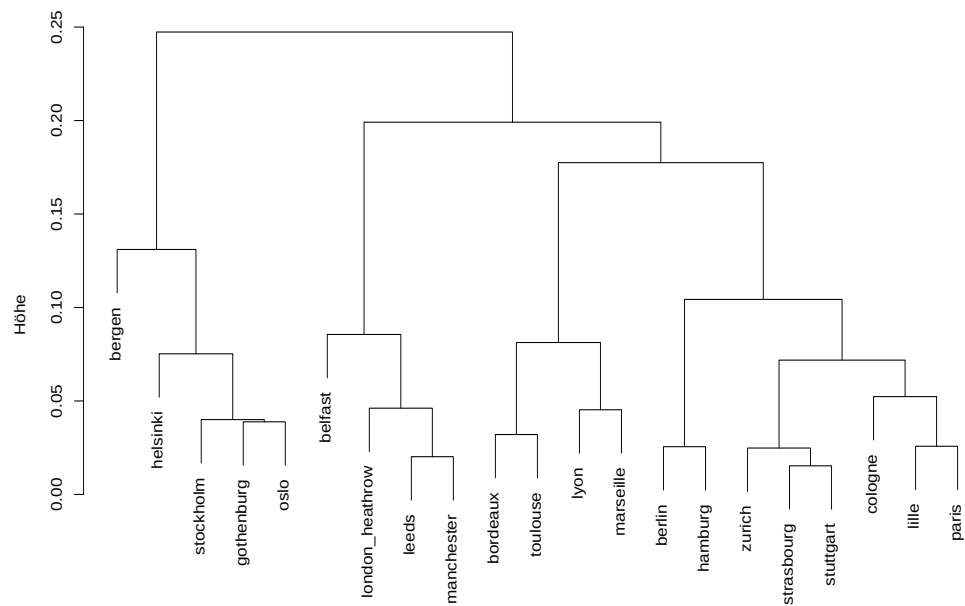


Abbildung 2.5: Cluster Dendrogramm Korrelationen der Temperaturzeitreihen Europas

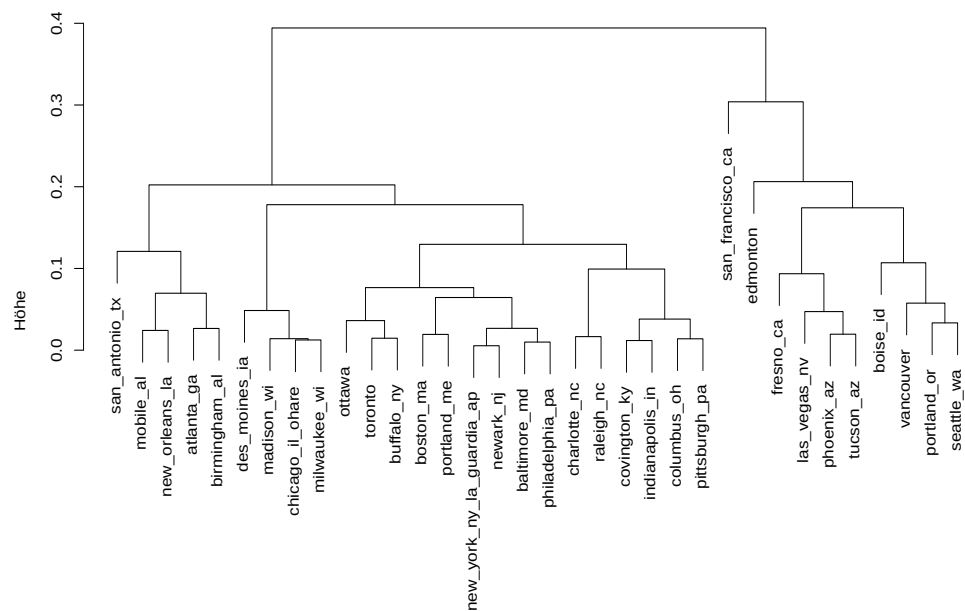


Abbildung 2.6: Cluster Dendrogramm der Korrelationen der Temperaturzeitreihen Amerikas

Kapitel 3

Trend und Saisoneffekt

In diesem Kapitel beginnen wir mit der Zerlegung der Temperaturzeitreihen. Das Ziel dieser Zerlegung ist eine stationäre Zeitreihe. Ein stationärer stochastischer Prozess wird wie folgt definiert:

Definition 3.1. Sei $(X_t)_{t \in \mathbb{Z}}$ ein stochastischer Prozess in diskreter Zeit. $(X_t)_{t \in \mathbb{Z}}$ heisst **stationär** (im strengen Sinne), falls

$$\mathbf{P}[X_{t_1} \leq x_1, \dots, X_{t_n} \leq x_n] = \mathbf{P}[X_{t_1+h} \leq x_1, \dots, X_{t_n+h} \leq x_n]$$

für $n = 1, 2, 3, \dots$, $t_1, \dots, t_n \in \mathbb{Z}$, $x_1, \dots, x_n \in \mathbb{R}$ und $h \in \mathbb{Z}$.

Die Verteilungsfunktion eines stationären stochastischen Prozesses ist also invariant gegenüber Zeitverschiebungen. Diese Eigenschaft machen wir uns zunutze, um den stochastischen Prozess zu modellieren und um Temperaturvorhersagen zu machen.

In Abbildung 3.1 sind die Temperaturen von Zürich seit 1965 aufgetragen. Diese Zeitreihe zeigt eine ausgeprägte saisonale Abhängigkeit. Ausserdem muss von einem Trend ausgegangen werden (Temperaturerwärmung).

Unser Temperaturzeitreihen-Modell hat vorläufig die Form

$$T_t = m_t + s_t + R_t, \quad \mathbf{E}[R_t] = 0$$

wobei T_t die Temperatur, m_t den Trend, s_t den Saisoneffekt und R_t die Residuen zur Zeit t darstellen. In der Abbildung 3.1 stellt der rote Balken die Spannweite des Trends, der blaue die des Saisoneffekts und der grüne die der Residuen R_t dar.

Trend und Saisoneffekt müssen entfernt werden, um Stationarität zu erlangen. Dazu müssen diese Komponenten aus den Daten geschätzt werden.

3.1 Der Trend

Der Trend und der Saisoneffekt müssen iterativ geschätzt werden, damit sie sich nicht gegenseitig beeinflussen. In den Iterationen wird der Saisoneffekt jeweils an die um den

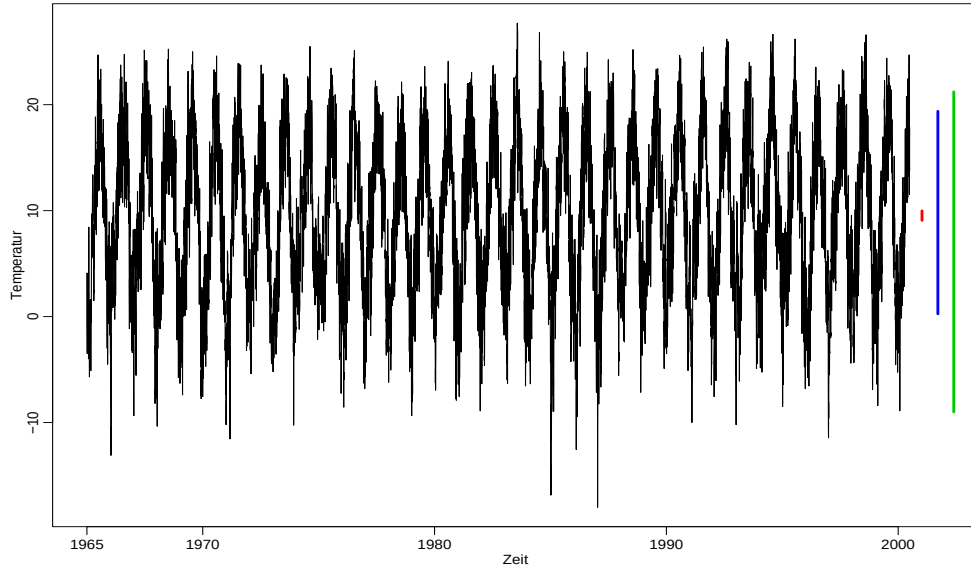


Abbildung 3.1: Tagestemperaturen in Zürich seit 1965

Trend (aus der vorangehenden Iteration) bereinigte Zeitreihe angepasst. Ebenso wird der Trend an die um den Saisoneffekt (aus der vorangehenden Iteration) bereinigte Zeitreihe angepasst.

Um die Trendkomponente zu schätzen, bieten sich verschiedene Verfahren an. Für die Analyse dieser Verfahren gehen wir davon aus, dass wir den wahren Saisoneffekt s_t kennen, und betrachten die Zeitreihe

$$E_t = T_t - s_t$$

3.1.1 Lineare (oder polynomiale) Regression

Die *lineare (oder polynomiale) Regression* ist die nächstliegende Methode zur Trendschätzung. Dabei wird an die Daten eine Gerade (bzw. ein Polynom vom Grad k) angepasst, d.h.

$$\hat{m}_t = a_0 + a_1 \cdot t + \dots + a_k \cdot t^k.$$

Die Regressionsparameter $a_0, \dots, a_k$ werden dabei mit dem Verfahren der Kleinsten Quadrate bestimmt.

3.1.2 Glättende Splines

Die *glättenden Splines* bieten sich an, wenn man keine funktionale Vorstellung über den Verlauf des Trends hat. Man geht lediglich davon aus, dass die Entwicklung von $\hat{m}_t$ relativ glatt verläuft. Die Glattheit kann dabei mit

$$\int \left(\frac{d^k}{dt^k} \hat{m}_t \right)^2 dt \quad (3.1)$$

gemessen werden. Je kleiner dieser Ausdruck ist, desto glatter ist die Trendfunktion. Häufig wird die Ableitungsordnung $k = 2$ gewählt. (3.1) misst dann die mittlere Krümmung von $\hat{m}_t$.

Andererseits soll die Trendfunktion $\hat{m}_t$ den Verlauf der Temperaturzeitreihe möglichst gut wiedergeben, d.h. einen möglichst kleinen Bias besitzen.

Dazu wird der Ausdruck

$$\sum_{i=1}^n (E_{t_i} - \hat{m}_{t_i})^2 \quad (3.2)$$

minimiert, wobei $t_1, \dots, t_n$ die sogenannten Knotenpunkte sind. Falls die Residuen R_{t_i} normalverteilt sind, entspricht (3.2) dem Negativen der Log-Likelihoodfunktion.

Die Minimierungen von (3.1) und (3.2) verhalten sich gegenläufig. Im Fall $k = 2$ wird das Glattheitsmass (3.2) null, falls $\hat{m}_t$ linear ist. Andererseits wird (3.2) minimal, falls $\hat{m}_t$ die Temperaturzeitreihe T_{t_i} , $i = 1 \dots n$ interpoliert. Deshalb wird ein positiver Parameter α eingeführt und die Funktion

$$\ell_\alpha(\tilde{m}_t) = \sum_{i=1}^n (E_{t_i} - \tilde{m}_{t_i})^2 + \alpha \cdot \int \left(\frac{d^k}{dt^k} \tilde{m}_t \right)^2 dt$$

nach $\tilde{m}_t$ minimiert, d.h. $\hat{m}_t = \operatorname{argmin}_{\tilde{m}_t} \ell_\alpha(\tilde{m}_t)$. Die Funktion $\ell_\alpha(\tilde{m}_t)$ ist eine sogenannte Penalized Log-Likelihoodfunktion. Der Parameter α legt dabei fest, wie viel Gewicht der Glattheit zukommen soll. Für $k = 2$ ist $\hat{m}_t$ ein natürlicher Spline (kubisch) mit Knoten $t_1, \dots, t_n$. $\hat{m}_t$ ist also ein stückweise kubisches Polynom.

3.1.3 Nadaraja-Watson-Kernschätzer

Ein weiteres nichtparametrisches Verfahren zur Trendschätzung ist der *Nadaraja-Watson-Kernschätzer*. Dieser bildet für jeden Zeitpunkt das gewichtete Mittel der (trendbereinigten) Temperaturen seiner Nachbarnpunkte. Dabei wird die sogenannte *k-nächste Nachbarschaft* verwendet, die wir für den Zeitpunkt t mit $N(t)$ bezeichnen.

Die *k-nächste Nachbarschaft* $N(t)$ wählt immer nur die k Zeitpunkte aus, die zu t am nächsten liegen. Dadurch werden zum Beispiel für ein t am linken Rand die k nächsten rechten Nachbarnpunkte gewählt.

Der Kernschätzer für den Trend hat die Form

$$\hat{m}_t = \frac{\sum_{i=1}^n K\left(\frac{|t-t_i|}{b_t}\right) \cdot E_{t_i}}{\sum_{i=1}^n K\left(\frac{|t-t_i|}{b_t}\right)},$$

wobei der Kern $K(\cdot)$ eine Gewichtung der Nachbarnpunkte bewirkt. Als Kern wird häufig

$$K(x) = \left((1 - x^2)_+ \right)^2$$

gewählt. Diese Funktion hat den Träger $[-1, 1]$.

$b_t = \max_{\tilde{t} \in N(t)} |t - \tilde{t}|$ bezeichnet den Abstand des Punktes t zum k -nächsten Punkt. Somit erhalten nur jene Punkte in der Nachbarschaft $N(t)$ ein Gewicht ungleich null. Häufig wird beim Nadaraja-Watson-Kernschätzer statt der k -nächsten Nachbarschaft (b_t) auch ein fester Parameter b verwendet.

3.1.4 Der Lowess

Der *Lowess* (Locally weighted regression) ist ein von Cleveland [5] entwickeltes nichtparametrisches Verfahren. Es beruht auf einer lokalen gewichteten Regression. Die Trendschätzung $\hat{m}_t$ wird durch folgende Schritte bestimmt:

1. *Definition der Nachbarschaft*

Zuerst muss die Nachbarschaft $N(t)$ für den Zeitpunkt t bestimmt werden. Dazu wird wie beim Kernschätzer die k -nächste Nachbarschaft verwendet.

$b_t = \max_{\tilde{t} \in N_t} |t - \tilde{t}|$ bezeichnet wieder den Abstand des Punktes t zum k -nächsten Punkt.

2. *Festlegung der Gewichte benachbarter Punkte*

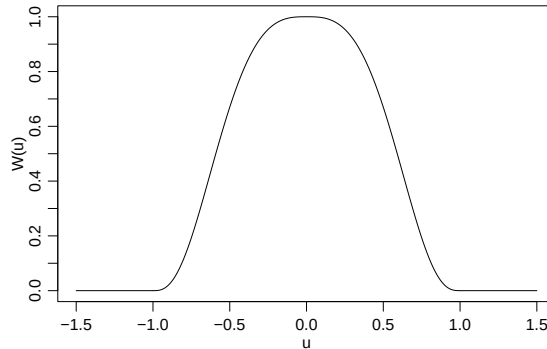
Wir führen für jeden Punkt ein Gewicht ein. Diese Gewichte werden durch

$$v_i(t) = W\left(\frac{|t - t_i|}{b_t}\right) \quad (3.3)$$

definiert. Dabei ist

$$W(u) := \begin{cases} (1 - u^3)^3 & |u| \leq 1 \\ 0 & |u| > 1 \end{cases} \quad (3.4)$$

die trikubische Gewichtsfunktion.



Das grösste Gewicht erhalten also die Daten, die dem zu schätzenden Zeitpunkt t am nächsten sind. Für die weiter entfernten Punkte wird das Gewicht reduziert. Die Punkte ausserhalb der Nachbarschaft $N(t)$ erhalten Gewicht 0. Dieser Gewichtung liegt die Idee zu Grunde, dass die Punkte nahe bei t mehr zu einem (lokalen) funktionalen Zusammenhang beitragen als die weiter entfernten.

Wir setzen $w_i(t) := v_i(t)$.

3. *Lokale gewichtete Regression*

Wir führen nun eine polynomiale Regression vom Grad d durch. Die Regression wird mit $w_i(t)$ gewichtet. Die Regressionskoeffizienten werden durch

$$(a_0, \dots, a_d) = \underset{(\tilde{a}_0, \dots, \tilde{a}_d) \in \mathbb{R}^d}{\operatorname{argmin}} \sum_{i=1}^n w_i(t) \cdot (T_{t_i} - \tilde{a}_0 - \tilde{a}_1 \cdot t_i - \dots - \tilde{a}_d \cdot t_i^d)^2 \quad (3.5)$$

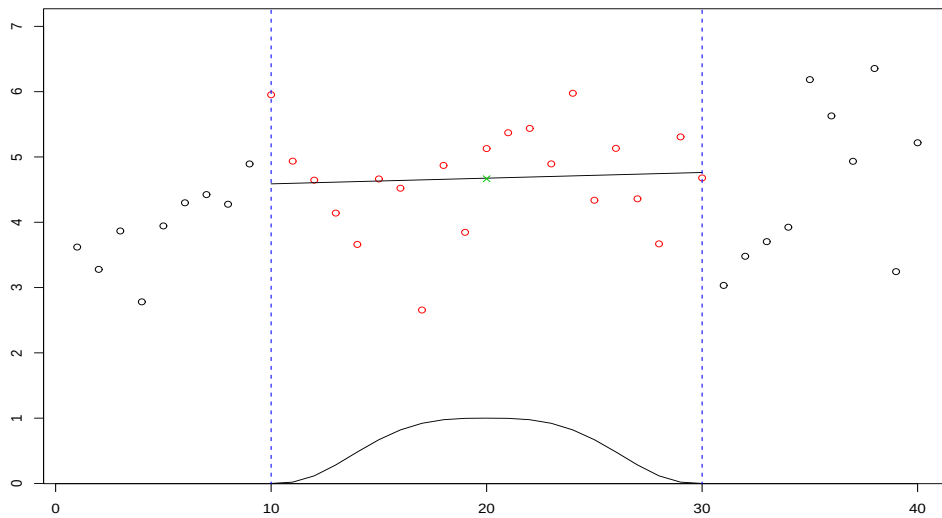


Abbildung 3.2: Veranschaulichung der lokalen Regression: Um den Wert des Lowess an der Stelle 20 zu berechnen, wird eine Regressionsgerade an die roten Punkte (Umgebung $N(20)$) angepasst. Die Gewichtung ist unten eingezeichnet. Der Wert des Lowess bei 20 ist der grüne Punkt auf der Regressionsgeraden.

bestimmt. Zur Bestimmung der Regression werden also nur die Punkte in der Nachbarschaft $N(t)$ miteinbezogen.

Der an der Stelle t angepasste Wert wird zur Schätzung des Trends verwendet, d.h.

$$\hat{m}_t = a_0 + a_1 \cdot t + \dots + a_d \cdot t^d. \quad (3.6)$$

4. Berechnung der Robustheitsgewichte

Das Lowess-Verfahren kann durch die sogenannten Robustheitsiterationen robustifiziert werden. Dazu betrachten wir die Residuen

$$R_t = E_t - \hat{m}_t$$

des Lowess-Verfahrens. Jene Beobachtungen, die ein grosses Residuum besitzen, werden heruntergewichtet. Dadurch kann der Einfluss dieser Ausreisser auf die lokale Regression reduziert werden. Die Robustheitsgewichte werden durch

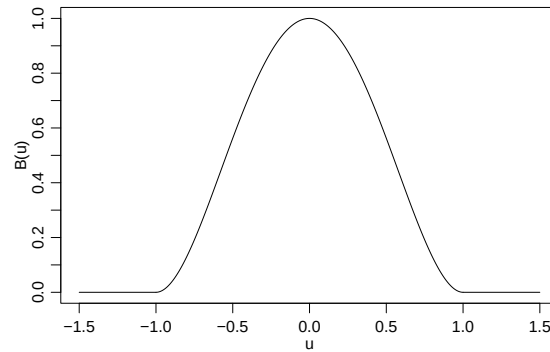
$$\varrho_t = B\left(\frac{|R_t|}{h}\right)$$

bestimmt, wobei

$$B(u) := \begin{cases} (1 - u^2)^2 & 0 \leq u \leq 1 \\ 0 & u > 1 \end{cases}$$

die biquadratische Gewichtsfunktion ist und

$$h = 6 \cdot \text{median}(|R_t|).$$

Abbildung 3.3: Biquadratische Gewichtsfunktion $B(u)$

5. Robustheitsiterationen

Wir führen die lokale Regression in Schritt 3) mit den Gewichten

$$w_i(t) = v_i(t) \cdot \varrho_t, \quad (3.7)$$

nochmals durch. Diese Iteration wird mehrmals hintereinander mit immer neu berechneten Robustheitsgewichten durchgeführt, bis die Lowess-Schätzung konvergiert.

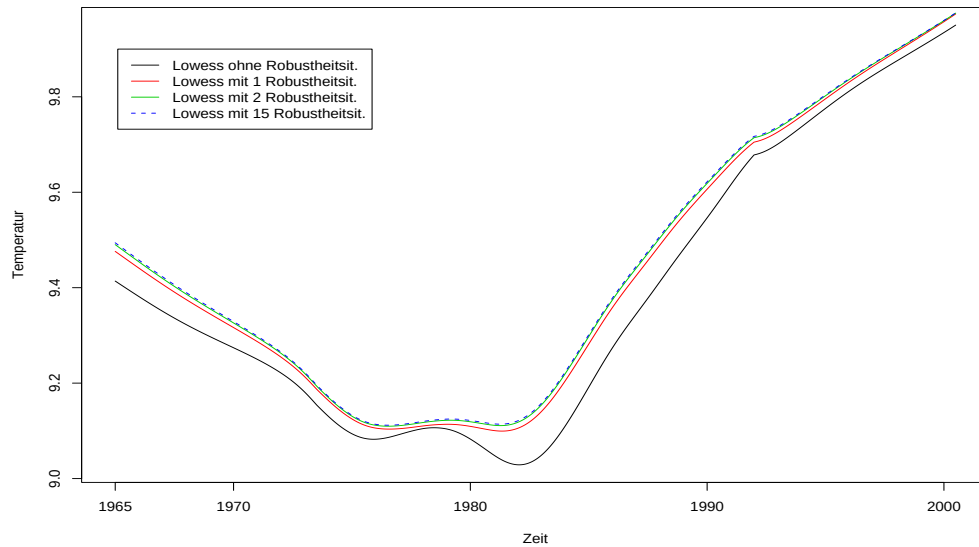


Abbildung 3.4: Konvergenzgeschwindigkeit des Lowess mit Robustheitsiterationen

Die Abbildung 3.4 vergleicht den Lowess für verschiedene Anzahlen von Robustheitsiterationen (Datensatz Zürich). Dabei zeigt sich, dass der grösste Unterschied nach der ersten Iteration feststellbar ist (dies kommt vom Einfluss der Robustheitsgewichte). Von der ersten zur zweiten Iteration ist noch ein kleinerer Unterschied zu verzeichnen. Zwischen der zweiten und der 15. Robustheitsiteration besteht der Unterschied nur noch im dritten Nachkommabereich. Für die Temperaturdaten konvergieren die Robustheitsiterationen also relativ schnell. Im wesentlichen reichen schon zwei Robustheitsiterationen aus.

3.1.5 Diskussion

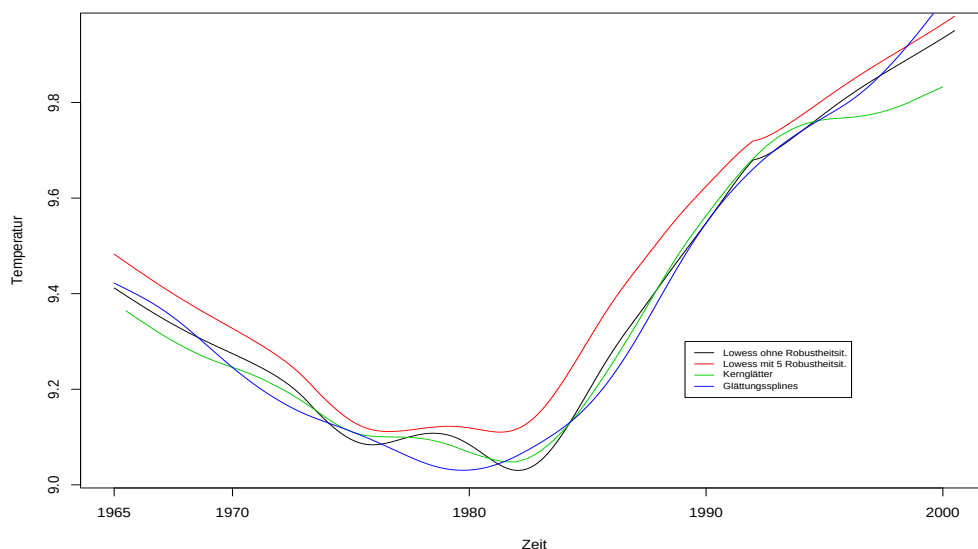


Abbildung 3.5: Verschiedene Trendmethoden im Vergleich (Beispiel Zürich)

Für die Trendschätzung muss eines dieser Verfahren gewählt werden. Die lineare (bzw. polynomiale) Regression hat den Nachteil, dass sie eine bestimmte Form des Trends voraussetzt. Zudem meinen die Klimatologen (z.B. in [8]), dass die Klimaerwärmung nicht linear vonstatten gehen muss.

Ausserdem kann es durch die Umstellung der Temperatur-Messstationen (andere Sensoren) zu einer systematischen Verschiebung der Temperatur kommen (z.B. 0.5°C mehr). An diese Verschiebung kann sich ein nichtparametrisches Verfahren besser anpassen. Deshalb fällt die Entscheidung zugunsten eines der nichtparametrischen Verfahren.

In Abbildung 3.5 sind die Trendschätzungen dieser Verfahren für Zürich aufgetragen. Dabei wurden die Parameter der Verfahren so gewählt, dass die *äquivalenten Freiheitsgrade* übereinstimmen. Es wird ersichtlich, dass der Trend bis zum Beginn der 80er Jahre fallend ist, danach aber auf ein höheres Niveau als 1965 ansteigt. Eine lineare Regression über alle Daten wäre also falsch gewesen. Allenfalls könnte eine stückweise lineare Funktion angepasst werden.

Die nichtparametrischen Verfahren zeigen alle dasselbe qualitative Verhalten des Trends. Der Lowess mit Robustheitsiterationen liegt etwas höher als die anderen Methoden. Dies kommt daher, dass bei unseren Daten die Ausreisser vor allem nach unten vorhanden sind (vgl. Abbildung 3.1). Diese Messungen sind jedoch legitim, denn diese tiefen Temperaturen können hin und wieder auftreten. Deshalb werden sie nicht entfernt, da sonst unser Modell verfälscht würde. Die Robustheitsiterationen des Lowess gewichten diese Ausreisser jedoch herunter, und dadurch wird das Niveau des Trends etwas höher als beim Lowess ohne Robustheitsiterationen.

Da für unsere Daten eine robuste Trendschätzung vorteilhaft ist (Ausreisser), fällt die Wahl auf den Lowess mit Robustheitsiterationen.

Das qualitative Verhalten des Trends bei verschiedenen Stationen fällt sehr unterschiedlich aus. Dies zeigt sich in der Abbildung (3.6), die alle Trends in Amerika miteinander

vergleicht. Dabei werden die Trends so verschoben, dass alle am 1. Januar 1965 bei 0 beginnen.

Wird jedoch nur eine einzelne Regionen betrachtet (vgl. Cluster in Kapitel 2), zeigt sich ein einheitlicheres Bild. Dies ist zum Beispiel bei den Stationen im Nordosten und Zentrum der USA der Fall (Abbildung 3.7).

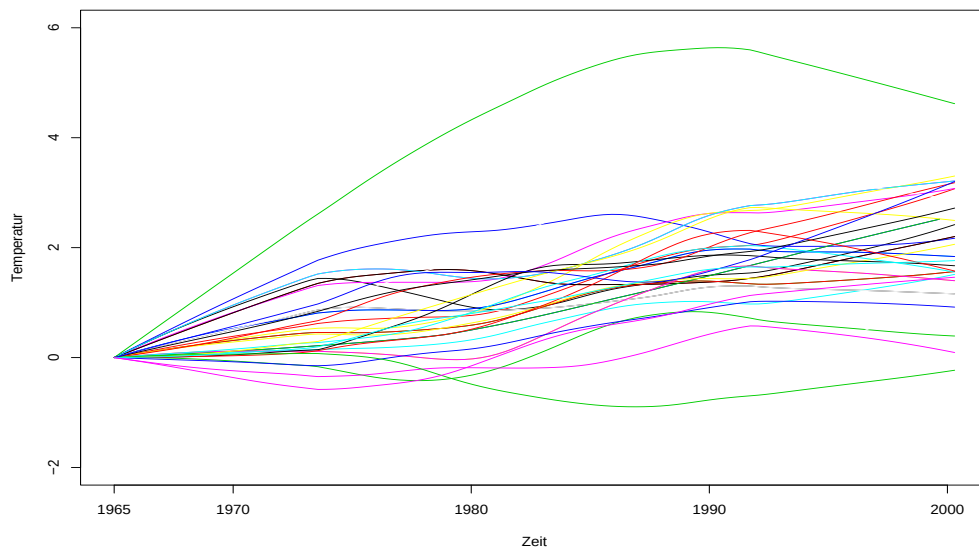


Abbildung 3.6: Vergleich der Trends in Amerika

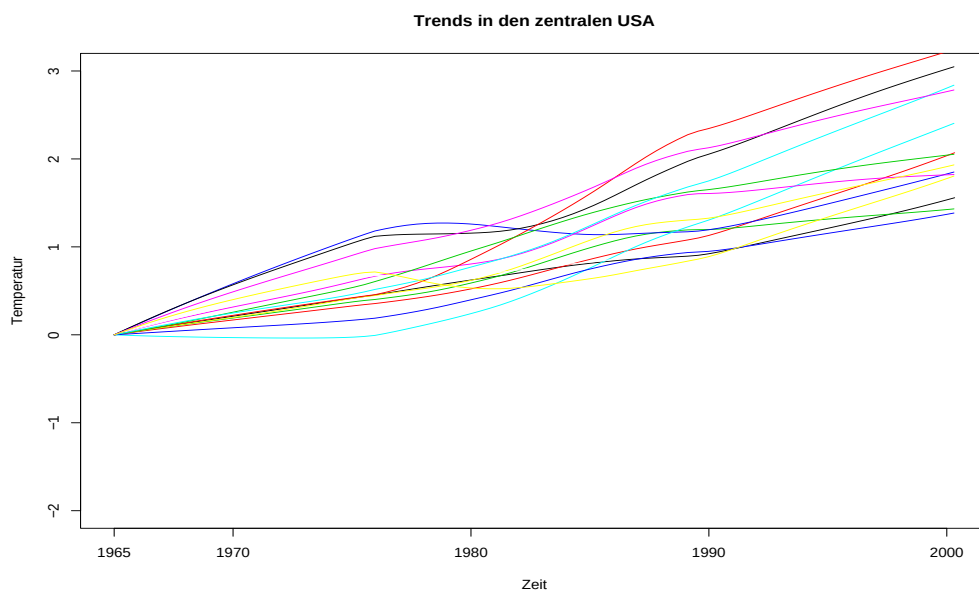


Abbildung 3.7: Trends im Nordosten und Zentrum der USA

3.2 Der Saisoneffekt

Die Temperaturen unterliegen einem ausgeprägten Jahresgang. Dieser jahreszeitliche Temperaturverlauf ist in seiner Spannweite um ein Vielfaches grösser als die des Trends. Deshalb ist es für ein Temperaturmodell wichtig, die Saisonkomponente möglichst gut zu schätzen.

Die Entfernung der Saisonkomponente eröffnet auch einen Einblick in die Strukturen des Temperaturprozesses. Die Abbildung 3.8 zeigt das Histogramm der Temperaturen von Zürich. Es sind darin zwei Gipfel erkennbar. Dies kommt daher, dass dieses Histogramm aus einer Überlagerung von verschiedenen Verteilungen zustande kommt. Je nach Jahreszeit hat die Verteilung der Temperaturen einen anderen Mittelwert.

Zur Schätzung der Saisonkomponente vergleichen wir wiederum verschiedene Verfahren. Wie schon früher erwähnt, müssen Trend und Saisoneffekt iterativ geschätzt werden. Für unsere Untersuchung in diesem Abschnitt gehen wir davon aus, dass der wahre Trend m_t bekannt ist, und wir betrachten die Zeitreihe

$$D_t = T_t - s_t.$$

Der geschätzte Saisoneffekt soll folgenden Anforderungen genügen:

1. Nach Abzug der Saisonkomponente von der (trendbereinigten) Temperaturzeitreihe soll das 1. Moment der Residuen vom Tag im Jahr unabhängig sein.
2. Die Saisonkomponente soll eine Periode von 365 Tagen haben.
3. Der Schätzer des Saisoneffekts soll genügend glatt sein.
4. Die Schätzung des Saisoneffekts soll robust sein.

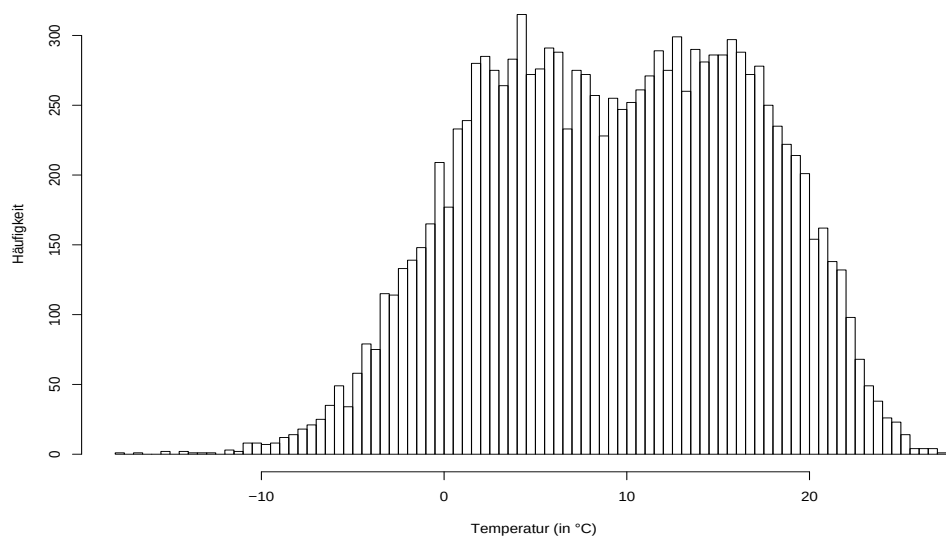


Abbildung 3.8: Histogramm der Temperaturen von Zürich

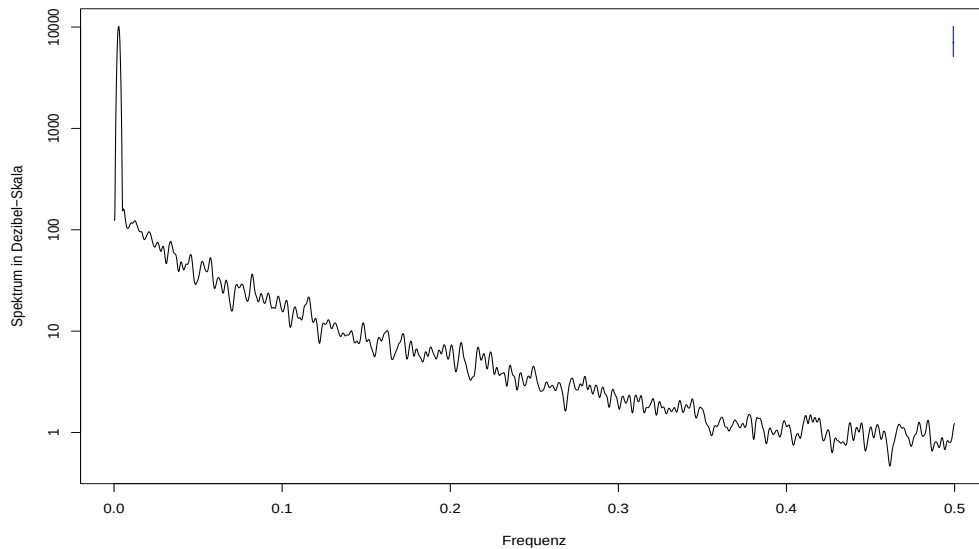


Abbildung 3.9: Geglättetes Spektrum von Zürich

Die Forderung 1) bringt uns dem Ziel eines stationären Prozess um einen grossen Schritt näher. Die Unabhängigkeit vom 2. Moment ist Thema des Kapitels 4.

Durch die Forderung 2) lässt sich der Saisoneffekt im Gegensatz zum Trend auch leicht für Temperaturvorhersagen verwenden. Die Periode 365 der Temperaturzeitreihe lässt sich auch sehr schön mit dem Spektrum nachweisen. In der Abbildung 3.9 ist das geglättete Spektrum von Zürich aufgetragen. Der Peak tritt bei der Frequenz $\frac{1}{365}$ auf.

In einem späteren Stadium werden wir auch eine langsame Veränderung des Saisoneffekts im Laufe der Zeit zulassen.

Die Forderung 3) stellt sicher, dass die geschätzte Saisonkomponente vom meteorologischen Standpunkt aus Sinn macht, denn grosse Schwankungen zwischen benachbarten Tagen sind nicht einsichtig.

Die Forderung 4) bewirkt, dass Ausreisser den Schätzer nicht verfälschen. Da in unseren Daten Ausreisser (v.a. in den Wintermonaten) vorhanden sind, ist diese Forderung berechtigt.

3.2.1 Bildung der Tagesmittelwerte

Eine für unsere Zwecke praktischere Parametrisierung von D_t ist $D_{\nu,j}$, wobei ν den Tag des Jahres ($1, \dots, 365$) und j das Jahr bezeichnet.

Wir betrachten nun den Saisoneffekt des Tages ν . Um der Forderung 1) zu genügen, sollte

$$s(\nu) = \mathbf{E}[D_{\nu,\cdot}] \quad (3.8)$$

gelten. Eine naheliegende Schätzung von (3.8) bildet der empirische Mittelwert von $D_{\nu,\cdot}$, d.h. in unserem Fall

$$\hat{s}(\nu) = \frac{1}{36} \sum_{j=1965}^{2000} D_{\nu,j}, \quad j = 1, \dots, 365. \quad (3.9)$$

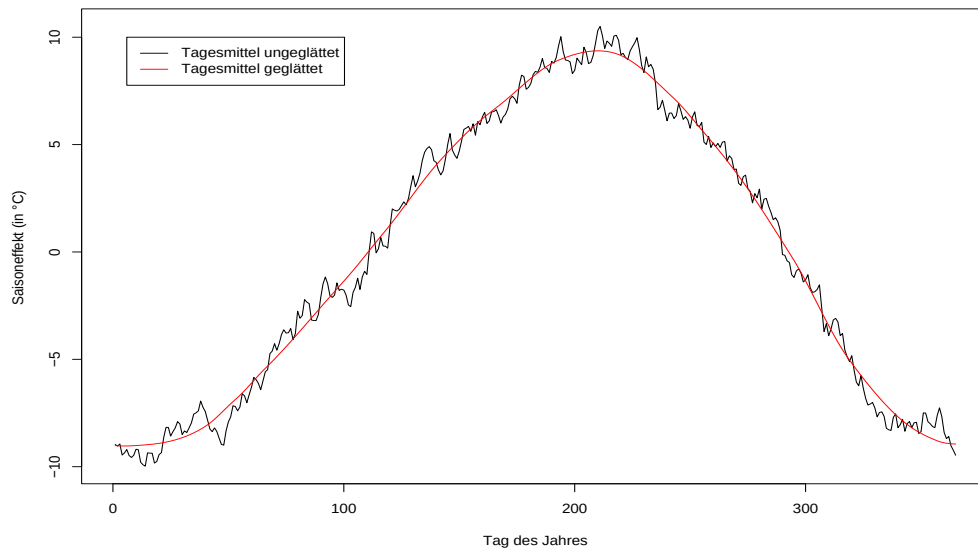


Abbildung 3.10: Saisoneffekt mit Tagesmittel (geglättet und ungeglättet)

Dieser Schätzer erfüllt nach Konstruktion die Forderung 2). Jedoch ist die Forderung 3) nicht erfüllt, wie man aus der Abbildung 3.10 (Datensatz Zürich) ersehen kann. Dies kommt daher, dass bei einem Stichprobenumfang von 36 Datenpunkten das arithmetische Mittel als Schätzung des Erwartungswerts eine zu grosse Streuung aufweist. Die Varianz des Schätzers beträgt

$$\frac{1}{36} \cdot \sigma_{\nu}^2$$

wobei σ_{ν} die Standardabweichung von $D_{\nu,\cdot}$ darstellt.

Um der Forderung 3) zu genügen, muss die Kurve der Tagesmittelwerte noch geglättet werden. Die rote Kurve in Abbildung 3.10 ist die Lowess-Glättung der Tagesmittel. Die Form des Saisoneffekts lässt nicht auf einen lokalen linearen Zusammenhang schliessen. Deshalb wird im Gegensatz zum Trend eine lokale *quadratische* Regression verwendet. Eine Verbesserung zeigt sich besonders an den Extrema des Saisoneffekts, da dort die Krümmung maximal ist.

Ein Problem der Schätzung des Saisoneffekts mit Tagesmitteln ist die mangelnde Robustheit des Verfahrens. Deshalb erfüllt die Saisonschätzung mit Tagesmitteln die Forderung 4) nicht.

Um auch die Forderung 4) zu erfüllen, kann das arithmetische Mittel aus (3.9) durch den Median ersetzt werden.

Die resultierende Schätzung des Saisoneffekts am Beispiel Zürichs zeigt die Abbildung 3.11.

Zum Vergleich wurde zusätzlich die Saisoneffektschätzung durch das arithmetische Mittel aufgetragen. Ein deutlicher Unterschied zeigt sich vor allem in den Wintermonaten. Dort

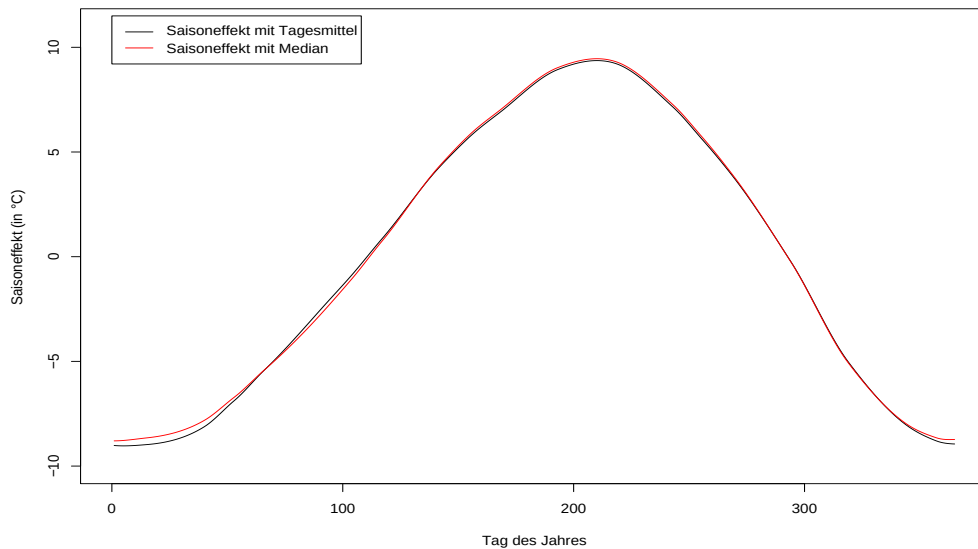


Abbildung 3.11: Vergleich des Saisoneffekts: arithmetisches Mittel und Median

ist der robuste Saisoneffekt etwas grösser. Dies liegt daran, dass in diesen Monaten Ausreisser nach unten vorkommen, die das arithmetische Mittel im Gegensatz zum Median stärker beeinflussen.

3.2.2 Schätzung des Saisoneffekts mit Fourieranalyse

Der *Satz von Fourier* besagt, dass jede Funktion $f(t)$ mit Periode 365 in der Form

$$f(t) = a_0 + \sum_{j=1}^{\infty} (a_j \cos(\omega_j t) + b_j \sin(\omega_j t)) \quad (3.10)$$

$$\omega_j = j \cdot \frac{2\pi}{365} \quad (3.11)$$

dargestellt werden kann. Mit anderen Worten kann jede periodische Schwingung als Überlagerung von harmonischen Schwingungen dargestellt werden, deren Grundfrequenz durch die Wiederholfrequenz der periodischen Schwingung gegeben ist.

Um eine solche Funktion zu glätten, kann man den hochfrequenten Anteil der Reihe weglassen, d.h. nur die k ersten Summanden der Funktion (3.10) betrachten. Die geglättete Funktion $\tilde{f}(t)$ hat also folgende Form:

$$\tilde{f}(t) = a_0 + \sum_{j=1}^k (a_j \cos(\omega_j t) + b_j \sin(\omega_j t)) \quad (3.12)$$

$$\omega_j = j \cdot \frac{2\pi}{365} \quad (3.13)$$

Der Saisoneffekt s_t ist nach Forderung 2) eine Funktion mit Periode 365. Er lässt sich somit nach dem Satz von Fourier in der Form (3.10) schreiben. Da wir nach Forderung 3)

von einem glatten Saisoneffekt ausgehen, verzichten wir auf den hochfrequenten Anteil und gehen von der Form (3.12) aus. Jedoch wird der Wert a_0 nicht benötigt, da der Saisoneffekt den Mittelwert 0 hat (der Trend wurde schon abgezogen).

Unter diesen Annahmen lässt sich die Reihe D_t als

$$D_t = \sum_{j=1}^k (a_j \cos(\omega_j t) + b_j \sin(\omega_j t)) + \tilde{R}_t \quad (3.14)$$

schreiben. D_t ist also zusammengesetzt aus dem deterministischen Saisoneffekt s_t und den stochastischen Grössen $\tilde{R}_t$. Aus dieser Reihe D_t muss also die Saisonkomponente s_t geschätzt werden. Durch die Gleichung 3.14 reduziert sich das Problem auf die Schätzung der Koeffizienten $a_1, \dots, a_k$ und $b_1, \dots, b_k$. Ferner muss eine vernünftige Wahl des Glattheitsparameters k getroffen werden.

Die Beziehung (3.14) lässt sich in Matrixform folgendermassen darstellen:

$$\begin{pmatrix} D_1 \\ \vdots \\ D_n \end{pmatrix} = \begin{pmatrix} \cos(\omega_1 \cdot 1) & \dots & \cos(\omega_k \cdot 1) & \sin(\omega_1 \cdot 1) & \dots & \sin(\omega_k \cdot 1) \\ \vdots & & \vdots & \vdots & & \vdots \\ \vdots & & \vdots & \vdots & & \vdots \\ \cos(\omega_1 \cdot n) & \dots & \cos(\omega_k \cdot n) & \sin(\omega_1 \cdot n) & \dots & \sin(\omega_k \cdot n) \end{pmatrix} \begin{pmatrix} a_1 \\ \vdots \\ a_k \\ b_1 \\ \vdots \\ b_k \end{pmatrix} + \begin{pmatrix} \tilde{R}_1 \\ \vdots \\ \tilde{R}_n \end{pmatrix} \quad (3.15)$$

Bei gegebenem k ist dies ein multiples Regressionsproblem. Zur Schätzung der Koeffizienten kann der Kleinste Quadrate-Schätzer verwendet werden, d.h.

$$(\hat{a}_1, \dots, \hat{a}_k, \hat{b}_1, \dots, \hat{b}_k) = \underset{\tilde{a}_1, \dots, \tilde{a}_k, \tilde{b}_1, \dots, \tilde{b}_k}{\operatorname{argmin}} \sum_{t=1}^n \left(D_t - \sum_{j=1}^k (\tilde{a}_j \cos(\omega_j t) + \tilde{b}_j \sin(\omega_j t)) \right)^2. \quad (3.16)$$

Der Spezialfall $k = 1$ entspricht einem sinusförmigen Saisoneffekt. Aus physikalischer Sicht (Sonnenstand) wäre die Sinusform korrekt.

Mit dem Kleinsten Quadrate-Schätzer erhält man für $k = 1$ die Koeffizienten

$$\hat{a}_1 = \frac{\sum_{t=1}^n (c_{ss} \cdot D_t \cdot \cos(\omega_1 t) - c_{cs} \cdot D_t \cdot \sin(\omega_1 t))}{c_{cc} \cdot c_{ss} - c_{cs}^2} \quad (3.17)$$

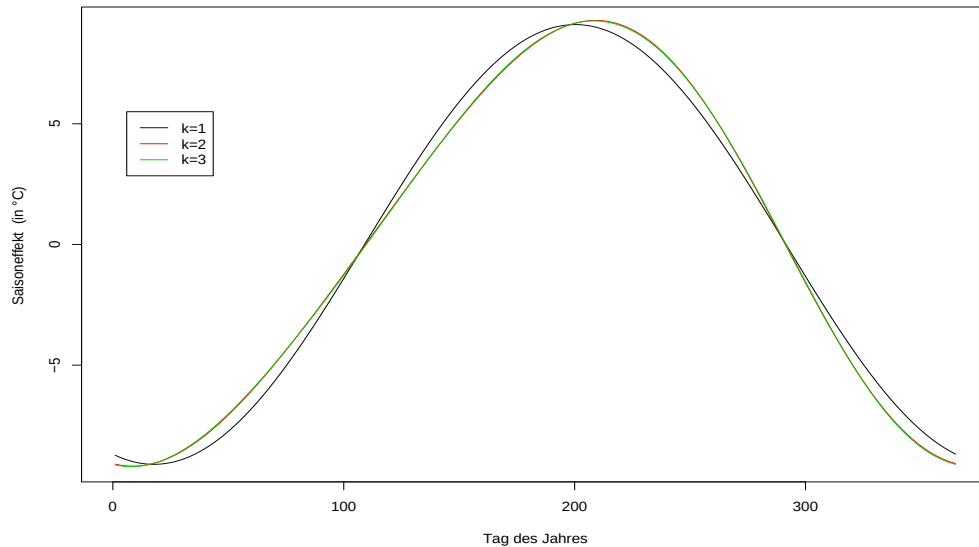
$$\hat{b}_1 = \frac{\sum_{t=1}^n (c_{cs} \cdot D_t \cdot \cos(\omega_1 t) - c_{cc} \cdot D_t \cdot \sin(\omega_1 t))}{c_{cs}^2 - c_{cc} \cdot c_{ss}}, \quad (3.18)$$

wobei $c_{ss} = \sum_{t=1}^n \sin(t)^2$, $c_{cc} = \sum_{t=1}^n \cos(t)^2$ und $c_{cs} = \sum_{t=1}^n \cos(t) \cdot \sin(t)$.

Für die Wahl des Parameters k betrachten wir die mit dem Kleinsten Quadrate-Schätzer bestimmten Koeffizienten für verschiedene Werte von k . In Tabelle (4.1) sind die Koeffizienten für $k = 1, 2, 3$ eingetragen (Datensatz Zürich). Dabei zeigt sich, dass $k = 2$ eine bedeutende Verbesserung gegenüber $k = 1$ darstellt. Dagegen liefert $k = 3$ gegenüber $k = 2$ nur eine geringfügige Verbesserung.

Dies bestätigt sich auch in Abbildung 3.12, wo die Saisoneffekte von $k = 1, 2, 3$ übereinander aufgetragen wurden.

	a_1	b_1	a_2	b_2	a_3	b_3
$k = 1$	-8.69	-2.74				
$k = 2$	-8.70	-2.75	-0.38	0.64		
$k = 3$	-8.69	-2.75	-0.38	0.63	-0.03	0.03

Tabelle 3.1: Fourierkoeffizienten für verschiedene Werte von k (Zürich)Abbildung 3.12: Vergleich der Saisoneffekte für verschiedene k

Der aus der Fourieranalyse gewonnene Schätzer des Saisoneffekts hat nach Konstruktion Periode 365 (Forderung 2) und ist auch glatt (Forderung 3). Das Problem bildet die Forderung 4), denn die Kleinste Quadrate Methode ist nicht robust gegenüber Ausreißern. Eine Möglichkeit festzustellen, ob es einflussreiche Beobachtungen gibt, und wie stark sie die Regression beeinflussen, bietet die Cook's Distanz. Sie ist definiert als

$$\delta_i = \frac{(\hat{\theta}^{(i)} - \hat{\theta})^T (\mathbf{X}^T \mathbf{X}) (\hat{\theta}^{(i)} - \hat{\theta})}{2k\hat{\sigma}^2}. \quad (3.19)$$

Dabei bezeichnet $\mathbf{X}$ die Matrix der erklärenden Variablen und $\hat{\theta}$ den Vektor der geschätzten Parameter. In unserem Fall gilt $\hat{\theta} = (\hat{a}_1, \dots, \hat{a}_k, \hat{b}_1, \dots, \hat{b}_k)^T$. Der Vektor $\hat{\theta}^{(i)}$ ist die Parameterschätzung unter Ausschluss der i -ten Beobachtung.

Die Cook's Distanz δ_i misst also den Einfluss der i -ten Beobachtung auf die Parameterschätzung. Als Abstandsmass wird die durch die Kovarianzmatrix von $\hat{\theta}$ definierte Metrik verwendet.

In Abbildung 3.13 ist die Cook's Distanz δ_i aus der Regression mit $k = 2$ für alle Beobachtungen aufgetragen (Datensatz Zürich). Wir stellen fest, dass einige einflussreiche Beobachtungen vorhanden sind, wobei die drei einflussreichsten mit der Beobachtungsnummer gekennzeichnet sind. Diese Beobachtungen fallen auch schon in der Abbildung 3.1 auf.

Der Einfluss dieser Beobachtungen auf das Regressionsmodell ist zwar nicht gewaltig, da viele Daten vorhanden sind. Es gibt jedoch Wetterstationen, die deutlich mehr Ausreisser

besitzen, welche die Schätzung stärker beeinflussen (vgl. Vancouver in Anhang C).

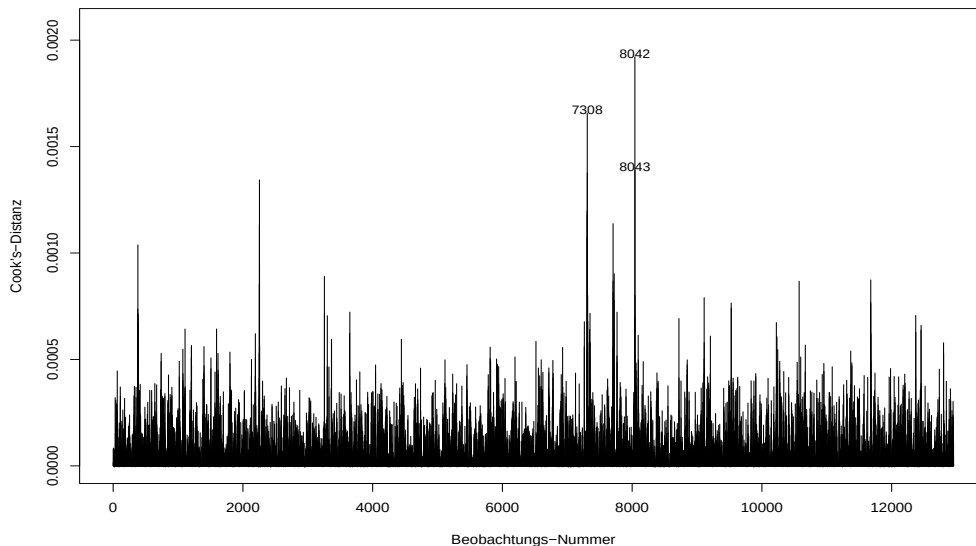


Abbildung 3.13: Cook's Distanz des Modells $k = 2$

Eine Variante zur Behandlung der Ausreisser wäre deren Entfernung. Dadurch würden aber die Vertrauensintervalle in Frage gestellt.

Eine bessere Lösung bietet die robuste Regression. In unserem Fall haben wir nur Ausreisser in der y-Richtung. Deshalb bietet sich der Huber-Regressionsschätzer an. Die Parameter werden dort durch

$$(\hat{a}_1, \dots, \hat{a}_k, \hat{b}_1, \dots, \hat{b}_k) = \underset{\tilde{a}_1, \dots, \tilde{a}_k, \tilde{b}_1, \dots, \tilde{b}_k}{\operatorname{argmin}} \sum_{t=1}^n \rho_c \left(\frac{D_t - \sum_{j=1}^k (\tilde{a}_j \cos(\omega_j t) + \tilde{b}_j \sin(\omega_j t))}{\hat{\sigma}} \right) \quad (3.20)$$

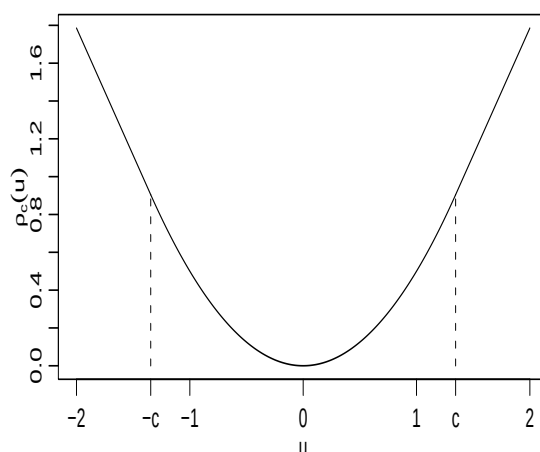
geschätzt. Dabei ist

$$\rho_c(u) = \begin{cases} \frac{1}{2}u^2 & \text{falls } |u| \leq c \\ c \cdot (|u| - \frac{c}{2}) & \text{falls } |u| \geq c \end{cases} \quad (3.21)$$

und $\hat{\sigma}$ eine Schätzung der Standardabweichung.

Beim Huber-Schätzer wird die Funktion $\rho_c(u)$ anstelle von u^2 beim Kleinsten Quadrate Schätzer verwendet. Diese Funktion ist in Abbildung 3.14 aufgetragen. Sie bildet einen stetig differenzierbaren Übergang von einer quadratischen zu einer linearen Funktion. Die Werte $|u|$ grösser als c werden also heruntergewichtet. Die Wahl des Parameters c hängt von der Standardabweichung der Residuen ab. Deshalb werden die Residuen in (3.20) normiert. Eine übliche Wahl für c ist dann 1.345.

Berechnet man die Koeffizienten für verschiedene k mit dem Huber-Schätzer nochmal neu, ergeben sich die Werte aus der Tabelle 3.2. Wiederum scheint $k = 2$ eine gute Wahl zu sein. Die Unterschiede zu den Koeffizienten in Tabelle 3.1 spielen sich im zweiten Nachkommabereich ab. Dennoch ist beim Vergleich des robusten mit dem Kleinsten Quadrate Saisoneffekt ein kleiner Unterschied feststellbar.

Abbildung 3.14: Funktion $\rho_c(u)$ des Huber-Schätzers

	a_1	b_1	a_2	b_2	a_3	b_3
$k = 1$	-8.65	-2.75				
$k = 2$	-8.65	-2.73	-0.34	0.68		
$k = 3$	-8.65	-2.73	-0.34	0.68	-0.04	0.07

Tabelle 3.2: Koeffizienten bei robuster Regression für verschiedene Werte von k (Zürich)

3.2.3 Diskussion

Wir haben mit den Tagesmitteln (bzw. Median) und der Fourieranalyse zwei Methoden zur Schätzung des Saisoneffekts zur Verfügung. Abbildung 3.15 vergleicht die Saisoneffekt-Schätzung des Medians mit der der Fourieranalyse mit $k = 2$. Die Verfahren zeigen eine erstaunliche Übereinstimmung. Am Beispiel Zürich beträgt die maximale Abweichung 0.34 °C.

Wir haben gesehen, dass der Sinus keine geeignete Schätzung des Saisoneffekts ist. Stattdessen ist die Variante mit $k = 2$ vorzuziehen. Der Unterschied des Saisoneffekts mit $k = 2$ zur idealen Sinusform lässt sich durch die Witterungsbedingungen erklären. So ist die Erwärmung im Frühling oft durch das schlechten Wetter verzögert.

Wir betrachten nun das geglättete Spektrum der um den Saisoneffekt bereinigten Zeitreihe. Dieses ist in der Abbildung 3.16 aufgetragen (Beispiel Zürich). Bei beiden oben vorgestellten Verfahren zeigt sich dasselbe Bild. Ein Vergleich mit dem Spektrum vor Entfernung der Saisonkomponente (Abbildung 3.9) zeigt, dass der Peak bei der Frequenz $\frac{1}{365}$ verschwunden ist. Wir haben also mit unseren Saisoneffekt-Schätzungen die 365-Tage Periodizität erfasst.

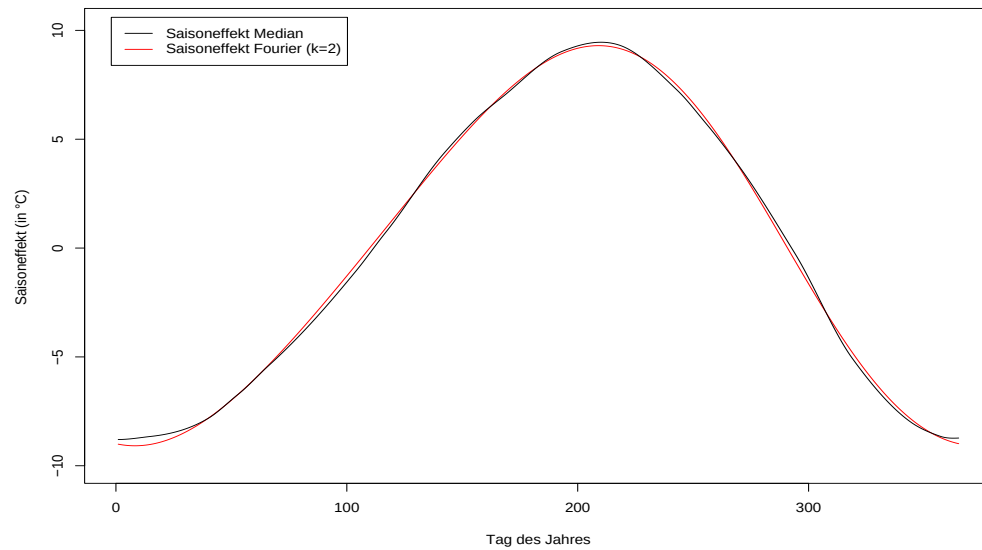


Abbildung 3.15: Saisoneffekt durch Bildung von Medianen (geglättet und ungeglättet)

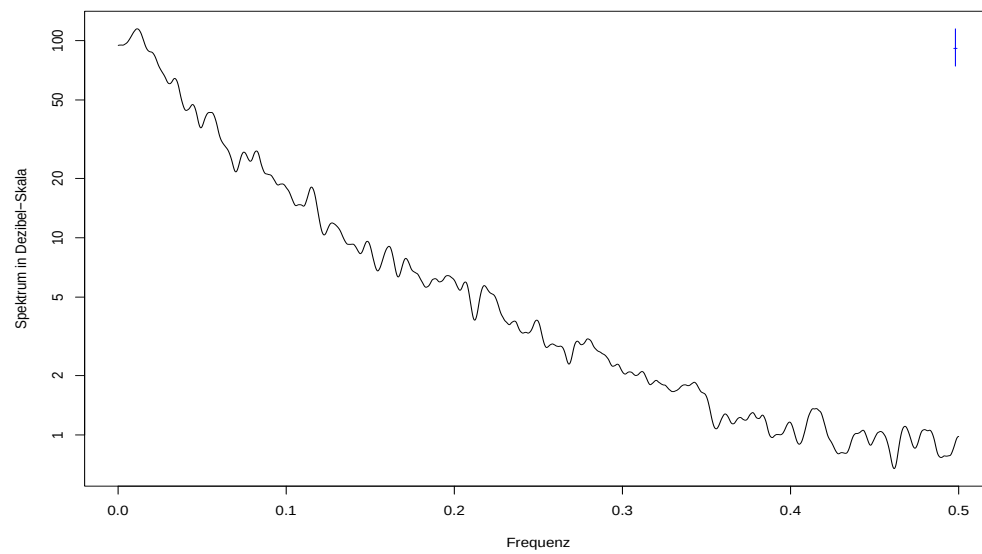


Abbildung 3.16: Spektrum nach Entfernung des Saisoneffekts

3.3 Das STL-Verfahren

In diesem Abschnitt soll das von Cleveland [6] entwickelte STL Verfahren erläutert werden. Es handelt sich um einen Algorithmus, der eine Zeitreihe iterativ in ihre Trend- und Saisonkomponente zerlegt.

Für die Trendkomponente wird ein Lowess verwendet. Dies ist genau das Verfahren, das wir in Abschnitt 3.1.4 für die Trendschätzung gewählt haben.

Die Schätzung der Saisonkomponente basiert auf einem raffinierten Verfahren, das wir nun näher betrachten werden. In diesem Verfahren fließt auch der Lowess mit ein. Der Vorteil dieser Schätzung der Saisonkomponente besteht darin, dass Veränderungen im Saisonereffekt zugelassen werden. Die bisher betrachteten Schätzer setzen alle einen periodischen Saisonereffekt voraus.

Zunächst müssen wir noch etwas Notation einführen:

- T_t sei die Temperaturzeitreihe ($t = 1, \dots, n$)
- $\hat{m}_t^{(k)}$ ist die Schätzung des Trends nach dem k -ten Durchgang.
- $\hat{s}_t^{(k)}$ ist die Schätzung der Saisonkomponente nach dem k -ten Durchgang.
- $w_t^{(k)}$ sind die Robustheitsgewichte nach dem k -ten Durchgang ($w_t^{(1)} \equiv 1$)
- $n_{(m)}$ ist die Fensterbreite (Grösse der Nachbarschaft) für die Lowess-Glättung des Trends
- $n_{(s)}$ ist die Fensterbreite für die Lowess-Glättung der Saisonkomponente
- $d_{(m)}$ ist der Grad der lokalen Regression beim Lowess des Trends
- $d_{(s)}$ ist der Grad der lokalen Regression beim Lowess des Saisonereffekts
- p bezeichnet die Periode des Saisonereffekts (hier: $p = 365$ Tage)

Wir betrachten nun den $k + 1$ -ten Durchgang des STL-Algorithmus:

1. Entfernung der Trendkomponente

Wir entfernen die im vorangehenden Durchgang bestimmte Trendschätzung $\hat{m}_t^{(k)}$ von der Zeitreihe T_t . Im ersten Durchgang ($k = 1$) existiert noch keine frühere Trendkomponente, und daher wird keine Subtraktion durchgeführt. Die um den Trend bereinigte Zeitreihe $D_t^{(k+1)}$ sieht also folgendermassen aus:

$$D_t^{(k+1)} := \begin{cases} T_t - \hat{m}_t^{(k)} & \text{falls } k > 1 \\ T_t & \text{falls } k = 1 \end{cases}$$

2. Glättung der Tageszeitreihen

Die Zeitreihe $D_t^{(k+1)}$ wird nun in die den Jahrestagen entsprechenden Teilreihen aufgeteilt. Dazu wird $D_t^{(k+1)}$ umparametrisiert: $D_{\nu,j}^{(k+1)}$ ist der Wert der Zeitreihe $D_t^{(k+1)}$ am Tag ν (ν nimmt Werte von 1 bis 365 an) des Jahres j . Bei festem ν ist $D_{\nu,j}^{(k+1)}$ die dem Tag ν entsprechende Unterreihe (j ist variabel). Als Beispiel ist in der Abbildung 3.17 die 1. Juni-Reihe $D_{152,j}^{(15)}$ von Zürich aufgetragen (Jahre 1965-2000).

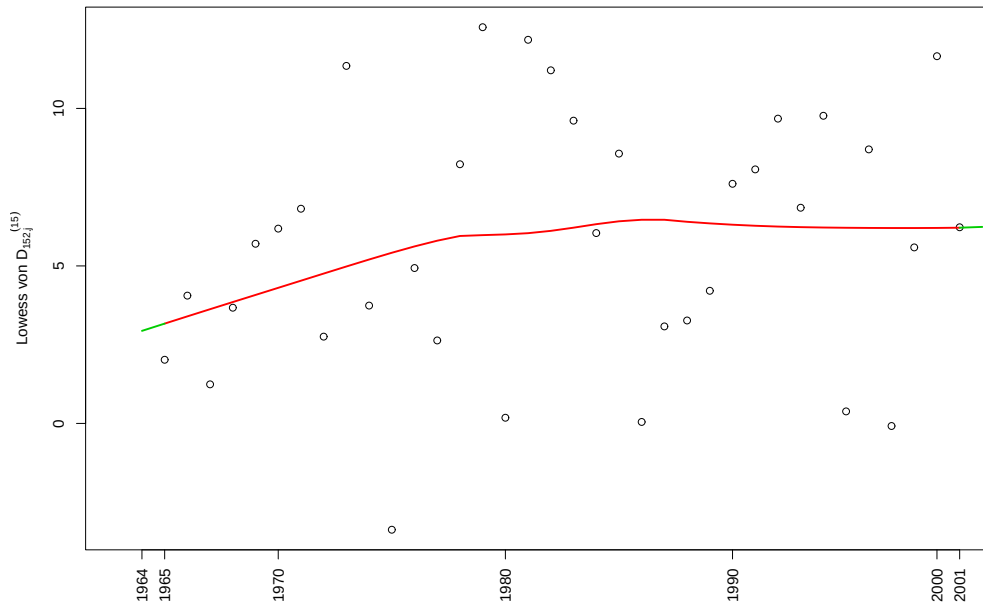


Abbildung 3.17: Lowess von $D_{152,j}^{(15)}$ mit Extrapolation der Jahre 1964 und 2001 (grün)

Jede der 365 Unterreihen $D_{\nu,j}^{(k+1)}$ ($\nu = 1, \dots, 365$) wird nun mit einem Lowess mit Fensterbreite $n_{(s)}$ geglättet. Beim Beispiel der Abbildung 3.17 ist diese Kurve rot eingezeichnet.

Der Grad der lokalen Regression sei $d_{(s)}$ und die $w_{\nu,j}^{(k)}$ werden als Robustheitsgewichte verwendet. Der verwendete Lowess führt selbst keine Robustheitsiterationen durch, da wir schon die Robustheitsgewichte $w_{\nu,j}^{(k)}$ zur Verfügung haben.

Der Lowess extrapoliert bei jeder der 365 Unterreihen an beiden Enden je einen Wert (Jahr). Bei unserem Beispiel ist die Extrapolation des 1. Junis der Jahre 1964 und 2001 grün eingetragen.

Die geglätteten und extrapolierten Unterreihen werden nun wieder zur Zeitreihe $C_t^{(k+1)}$ ($t = -p + 1, \dots, n + p$) zusammengesetzt. Diese Reihe hat durch die Extrapolation der Unterreihen nun $n + 2 \cdot p$ Werte.

3. Low-Pass-Filter auf $C_t^{(k+1)}$ anwenden

Wir wollen die Zeitreihe $C_t^{(k+1)}$ nun glätten. Dazu werden vier Glättungsschritte hintereinandergeschaltet:

- Zunächst wird die Reihe mit einem gleitenden Mittel (Moving Average) der Länge p geglättet. Dabei werden jeweils die Mittelwerte der p benachbarten Werte gebildet.

Das gleitende Mittel berücksichtigt im Gegensatz zum Lowess an beiden Seiten des Wertes eines Zeitpunkts t gleich viele Werte zur Mittelung. Deshalb kann das gleitende Mittel nicht an den Rändern der Zeitreihe verwendet werden. Die mit einem gleitenden Mittel der Länge p geglättete Reihe nimmt also nur noch Werte von $t = 1 - \frac{p+1}{2}, \dots, n + \frac{p+1}{2}$ an.

- Es wird ein zweites gleitendes Mittel der Längen p auf die im vorhergehenden Schritt geglättete Reihe angewendet. Die neue Reihe ist für die Zeiten $t = 1 - 1, \dots, n + 1$ definiert.
- Ein weiteres gleitendes Mittel der Länge 3 wird auf die oben erhaltene Reihe angewendet. Wir haben nun eine Zeitreihe, die für die Zeitpunkte $t = 1, \dots, n$ definiert ist.
Dies ist derselbe Bereich wie die ursprünglich betrachtete Temperaturzeitreihe T_t . Die Extrapolationen im Schritt 2 des STL-Algorithmus mussten nur darum gemacht werden, um die 3 gleitenden Mittel ohne Datenverlust verwenden zu können.
- Als letzter Glättungsschritt wird noch ein Lowess mit Fensterbreite $n_{(l)}$ auf die mit den gleitenden Mitteln erhaltene Reihe angewendet. Dabei entspricht die Fensterbreite $n_{(l)}$ der Periode der Zeitreihe, d.h. in unserem Fall gilt $n_{(l)} = 365$. Für die lokale Regression wird der Grad 1 gewählt (lokale lineare Regression).

Die mit den obigen 4 Glättungsschritten erhaltene Reihe bezeichnen wir mit $L_t^{(k+1)}$ $t = 1, \dots, n$.

4. Bildung der Saisonkomponente

Um die Saisonkomponente zu erhalten, wird von der im Schritt 2 erhaltenen Zeitreihe $C_t^{(k+1)}$ die im Schritt 3 geglättete Reihe $L_t^{(k+1)}$ abgezogen, d.h.

$$\hat{s}_t^{(k+1)} = C_t^{(k+1)} - L_t^{(k+1)}$$

Der Abzug von $L_t^{(k+1)}$ bewirkt, dass keine langfristigen Zyklen in den Saisoneffekt eingehen.

5. Entfernung der Saisonkomponente

Um den Trend an die Temperaturzeitreihe T_t anzupassen, muss zunächst die im Schritt 5 berechnete Saisonkomponente entfernt werden. Der Trend wird also an die Zeitreihe

$$E_t^{(k+1)} = T_t - \hat{s}_t^{(k+1)}$$

angepasst. Damit soll ausgeschlossen werden, dass der Saisoneffekt den Trend beeinflusst.

6. Trend anpassen

Um den Trend $\hat{m}_t^{(k+1)}$ zu schätzen, wird die Zeitreihe $E_t^{(k+1)}$ mit einem Lowess der Fensterbreite $n_{(m)}$ geglättet. Die lokale Regression hat den Grad $d_{(m)}$ und Robustheitsgewichte $w_t^{(k)}$. Wiederum sind keine zusätzlichen Robustheitsiterationen des Lowess nötig, da die Robustheitsgewichte $w_t^{(k)}$ schon vorhanden sind.

7. Robustheitsgewichte berechnen

Das STL Verfahren soll robust gegenüber Ausreißern sein. Dabei gilt es vor allem die an verschiedenen Orten eingesetzten Lowess-Verfahren mit Robustheitsgewichten zu versehen. Da der STL iterativ arbeitet, werden bei den einzelnen Lowess-Schätzern keine Robustheitsiterationen verwendet. Stattdessen werden pro Iteration des STL einmal die Robustheitsgewichte für die in der nächsten STL-Iteration verwendeten Lowess-Glätter bestimmt. Dies geschieht in diesem Schritt.

Es seien

$$R_t^{(k+1)} = T_t - \hat{m}_t^{(k+1)} - \hat{s}_t^{(k+1)}$$

die Residuen der Temperaturzeitreihe nach Abzug von Trend und Saisoneffekt. Die Robustheitsgewichte werden dann durch

$$\varrho_t^{(k+1)} = B\left(\frac{|R_t^{(k+1)}|}{h}\right)$$

bestimmt, wobei

$$B(u) := \begin{cases} (1 - u^2)^2 & 0 \leq |u| \leq 1 \\ 0 & |u| > 1 \end{cases}$$

und

$$h = 6 \cdot \text{median}(|R_t^{(k+1)}|).$$

Für lokale Regression beim Lowess werden also die Robustheitsgewichte $w_t^{(k+1)} := \varrho_t^{(k+1)}$ verwendet.

In Abbildung 3.18 ist die mit dem STL-Verfahren erzeugte Trend- und Saisonkomponente der Temperaturen von Zürich zu sehen. Für die Trendschätzung wurde der Glättungsparameter $f_{(m)} = 0.5$ verwendet. Dieser Parameter gibt an, welcher Prozentsatz der Daten für die lokale Regression verwendet wird, d.h. $n_{(m)} = f_{(m)} \cdot n$. Der Glättungsparameter der Saisonkomponente war $f_{(s)} = 0.8$. Für die Veränderung der Saisonkomponente darf nicht ein zu kleines $f_{(s)}$ gewählt werden, da die Saisonkomponente dann den Jahrestemperaturen zu stark folgt. Dies widerspricht aber dem Konzept des Saisoneffekts. Die Saisonkomponente soll über einen grösseren Zeitraum gültig sein, um Vorhersagen anzustellen. Zudem wird bei der lokalen Regression beim Saisoneffekt $d_{(s)} = 0$ gewählt. Die Wahl der Glättungsparameter $f_{(m)}$ und $f_{(s)}$ ist für die Güte der Temperaturvorhersagen von zentraler Bedeutung. Es gibt aber keine feste Regel, wie die Parameter zu wählen sind. Für jedes Anwendungsgebiet können sich andere Parameter als sinnvoll erweisen. Mit der optimalen Wahl von $f_{(t)}$ und $f_{(s)}$ werden wir uns deshalb in Kapitel 7 nochmals beschäftigen.

Die STL-Saisonkomponente in Abbildung 3.18 zeigt bei aufeinanderfolgenden Tagen hohe Schwankungen. Dies lässt sich aus meteorologischer Sicht nicht erklären. Deshalb glätten wir die Saisonkomponente mit einem Lowess mit dem kleinen Glättungsparameter $f_{(g)} = 0.0025$. Dieser Parameter wird deshalb so klein gewählt werden, da ja nur ein glatter Verlauf des Jahresgangs angestrebt wird. Der Saisoneffekt selbst soll aber beibehalten werden. Auch der Einfluss des Parameters $f_{(g)}$ auf die Vorhersagegüte wird in Kapitel 7 überprüft.

Das STL Verfahren zeichnet sich dadurch aus, dass sich der Saisoneffekt im Laufe der Zeit verändern kann. Dies ist zum Beispiel sinnvoll, wenn sich ein globaler Temperatur-Trend nicht auf alle Monate gleich auswirkt. Dies ist zum Beispiel in San Antonio der Fall (vgl. Anhang C), wo ein Anstieg des Saisoneffekts in den Wintermonaten zu verzeichnen ist. Im Sommer hingegen ist kein Anstieg des Saisoneffekts beobachtbar. Dies heisst aber nicht, dass sich die Temperaturen nicht erwärmen, denn die Erwärmung ist im Trend berücksichtigt (der Trend von San Antonio ist bis Mitte der 70er Jahre leicht abflachend und steigt danach stetig an). In den Sommermonaten ist die Erwärmung in San Antonio also höchstens so gross, wie der globale Trend m_t . Die Erwärmung des Winters in San Antonio

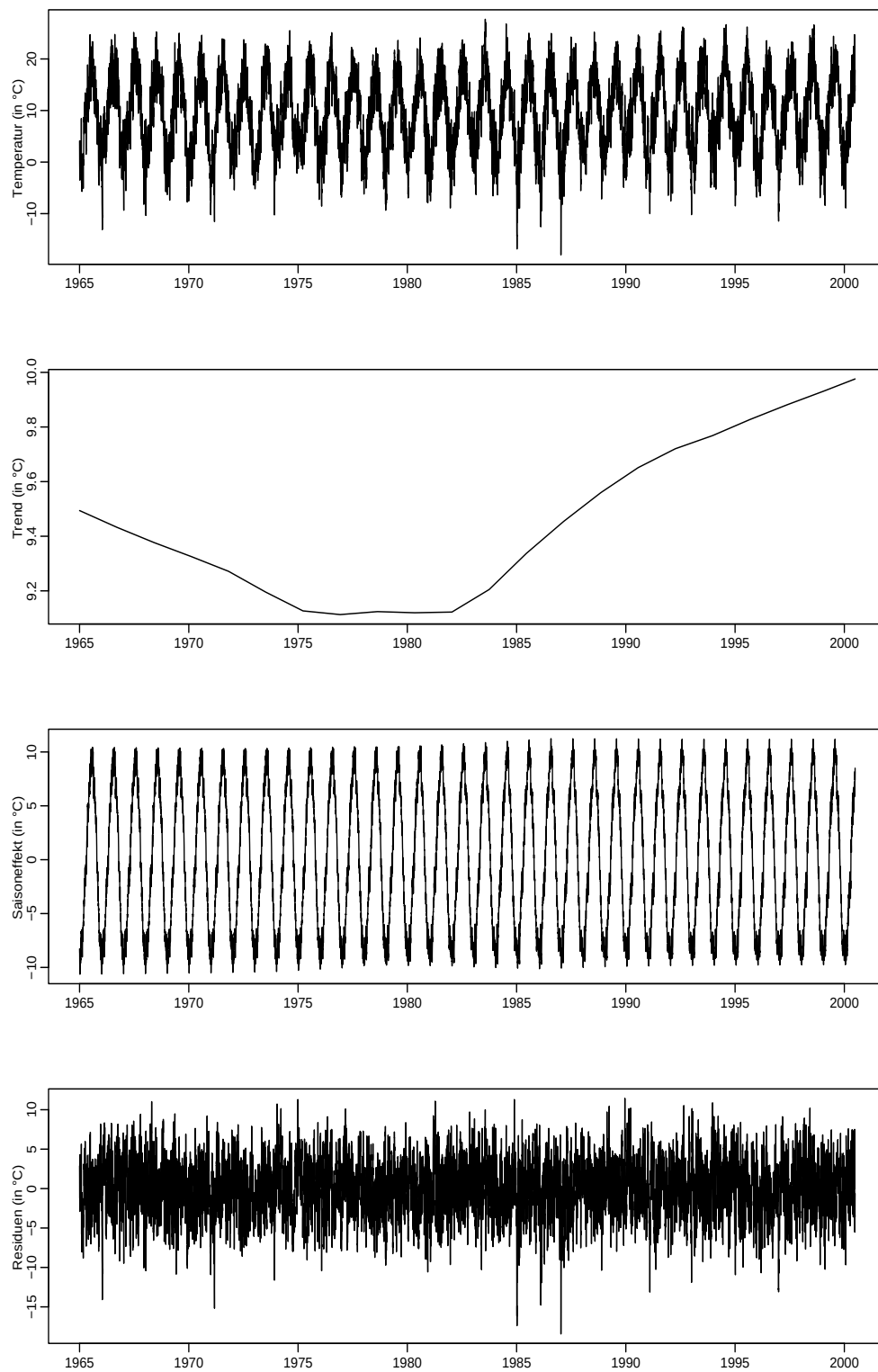


Abbildung 3.18: Temperaturen, STL-Trend, STL-Saisoneffekt und Residuen von Zürich

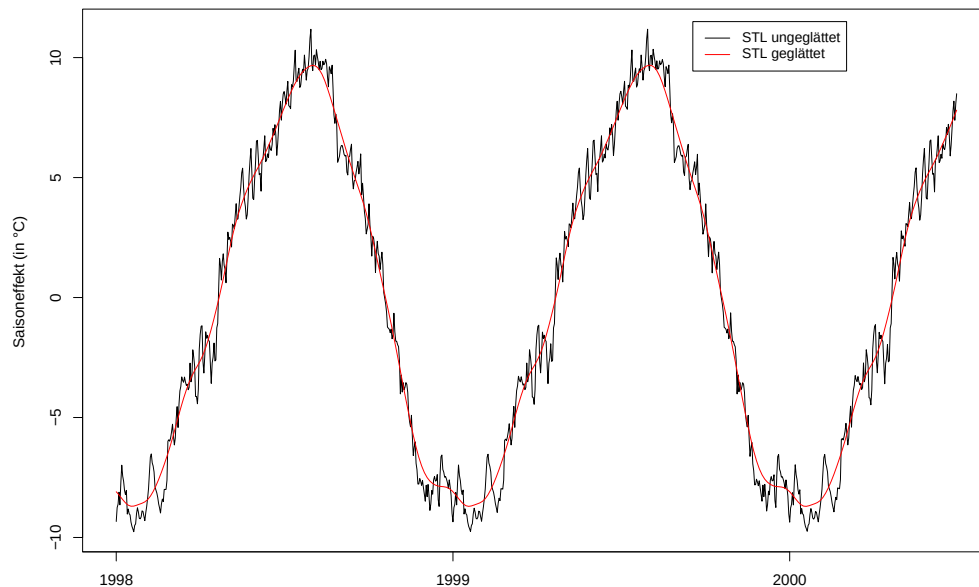


Abbildung 3.19: Geplätteter STL-Saisoneffekt von Zürich

fällt aber stärker aus, als der das ganze Jahr berücksichtigende Temperatur-Trend. Um diese Veränderung genauer zu untersuchen, könnte man die Veränderung der Saisonkomponente für jedem einzelnen Jahrestag betrachten. Um ein aussagekräftigeres Bild zu erhalten, werden die Tages-Temperaturen jedes Monats gemittelt. Dadurch entsteht eine neue Zeitreihe mit monatlichen Daten. Bei dieser Reihe betrachten wir nun die Veränderung des Saisoneffekts jedes Monats. Dies ist in der Abbildung 3.20 dargestellt. Die rote Linie ist der STL-Saisoneffekt, und die Punkte stellen die effektiven mittleren Temperaturen des jeweiligen Monats dar.

Diese Plots bestätigen, dass die Temperaturen im Dezember, Januar und Februar überdurchschnittlich ansteigen, in den Monaten Juni, August, September Oktober und November hingegen unterdurchschnittlich. Diese zwei gegenläufigen Bewegungen kompensieren sich, da der Trend den durchschnittlichen Temperaturanstieg darstellt.

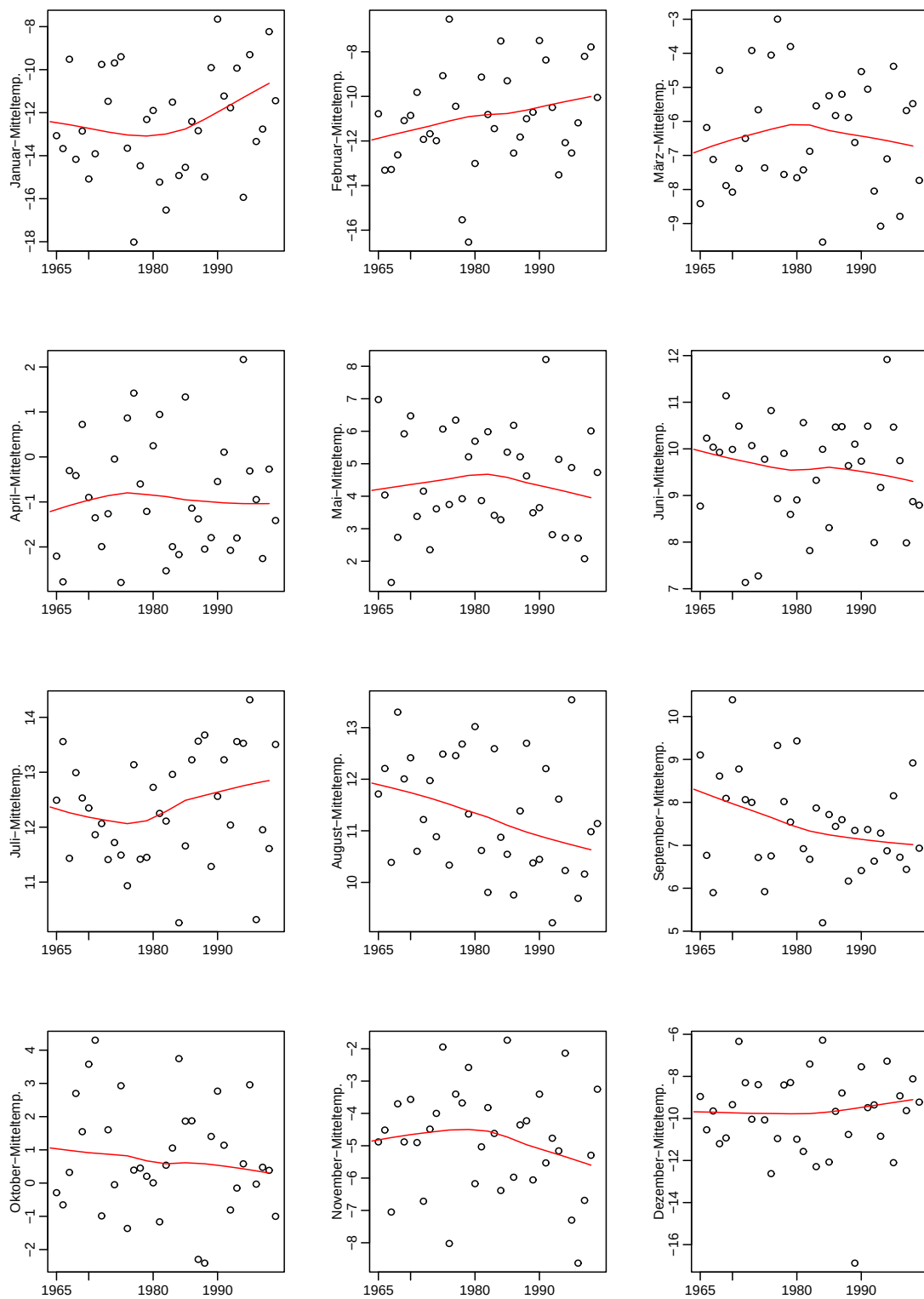


Abbildung 3.20: Veränderung der Saisonkomponente in verschiedenen Monaten (Beispiel San Antonio)

Kapitel 4

Die Heteroskedastizität der Residuen

Wir haben uns als Ziel gesetzt, eine stationäre Zeitreihe zu erhalten. Bisher haben wir dazu den Trend und den Saisoneffekt von den Temperaturzeitreihen abgezogen. Nun werden wir die resultierenden Residuen R_t genauer untersuchen. In Abbildung 3.18 im vorhergehenden Kapitel sehen wir die Residuen R_t von Zürich. Hier ist keine offensichtliche saisonale Abhängigkeit mehr erkennbar.

Bei den Residuen R_t von San Antonio (Abbildung 4.1) zeigt sich hingegen ein anderes Bild. Diese Residuen weisen nach wie vor eine ausgeprägte saisonale Abhängigkeit auf.

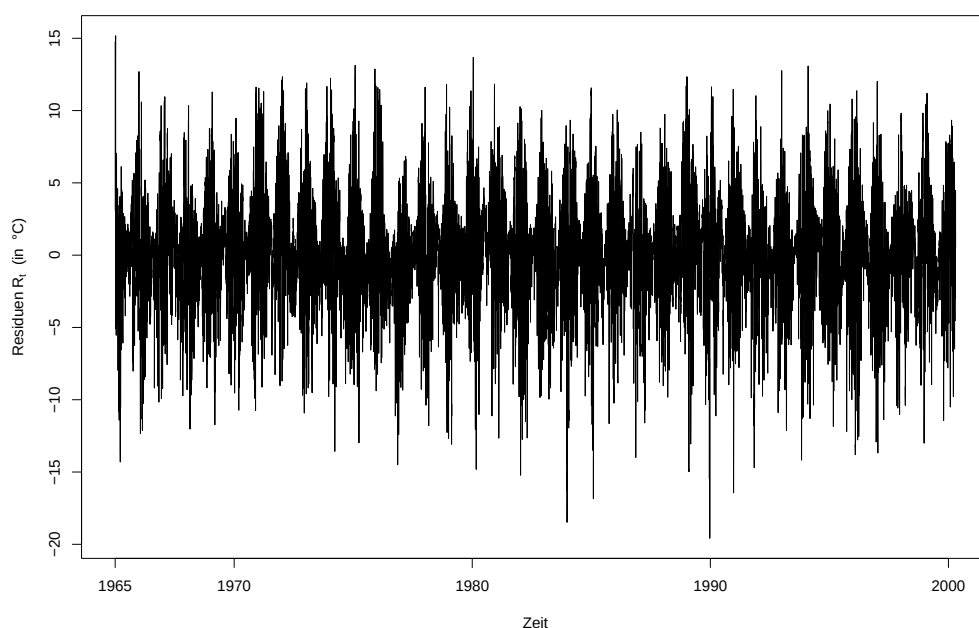


Abbildung 4.1: Residuen R_t von San Antonio

Wir stellen aber fest, dass dieser Effekt nicht wie der schon entfernte Saisoneffekt eine dem

Jahresgang entsprechende Anhebung bzw. Senkung der Residuen bewirkt. Die Residuen haben stets den Mittelwert 0.

Hier haben wir es mit einem Effekt zu tun, der die Residuen im Verlauf des Jahres mit einem sich verändernden Faktor multipliziert, d.h. die Standardabweichung der Residuen ist vom Jahresgang abhängig.

Deshalb betrachten wir nun die Veränderung des 2. Moments im Verlaufe des Jahres, nachdem in Kapitel 3 die Saisonalität des 1. Moments entfernt wurde.

Die Residuen R_t haben also folgende Form:

$$R_t = \sigma_t \cdot I_t. \quad (4.1)$$

Dabei ist σ_t die Standardabweichung des Tages t . Diese hat eine Periode von 365 Tagen, d.h. $\sigma_t = \sigma_{t+365}$. Mit I_t werden die normierten Residuen bezeichnet ($\mathbf{E}[I_t] = 0$ und $\text{Var}[I_t] = 1$).

4.1 Schätzung der saisonalen Standardabweichung

Um die Residuen R_t zu normieren, müssen wir nun die Standardabweichung σ_t schätzen. Dazu bieten sich wiederum verschiedene Verfahren an.

4.1.1 Die empirische Tages-Standardabweichung

Das nächstliegende Verfahren ist die Schätzung der Standardabweichung von jedem einzelnen Tag des Jahres. Als Parametrisierung von R_t verwenden wir wieder $R_{\nu,j}$, wobei ν den Tag des Jahres und j das Jahr bezeichnet.

Die empirische Varianz des Tages ν berechnet sich durch

$$\hat{\sigma}_{\nu}^2 = \frac{1}{(n-p)} \sum_{j=1965}^{2000} R_{\nu,j}^2. \quad (4.2)$$

In dieser Formel müssen wir keinen Mittelwert mehr abziehen. Dies haben wir mit der Trend- und Saisoneffekt-Schätzung schon erledigt. Um eine erwartungstreue Varianz-Schätzung zu erhalten, muss durch $(n-p)$ dividiert werden. Dabei bezeichnet p die Anzahl äquivalenter Freiheitsgrade, die zur Trend- und Saisoneffektschätzung verwendet wurden.

Um die Standardabweichung zu schätzen, muss die Wurzel aus (4.2) gezogen werden. Dies ergibt jedoch keinen erwartungstreuen Schätzer für die Standardabweichung.

In Abbildung 4.3 ist diese Schätzung für alle Jahrestage aufgetragen.

4.1.2 Standardabweichungs-Schätzung mit Lowess

Beim obigen Vorgehen ist die Schwankung der Tages-Standardabweichung von aufeinanderfolgenden Tagen relativ gross. Dies liegt daran, dass für einen einzelnen Tag nur 35 Daten zur Verfügung stehen. Durch Einbezug der Residuen von benachbarten Tagen kann

die Genauigkeit der Schätzung erhöht und die Schwankung reduziert werden. Eine Möglichkeit dazu ist die Glättung der geschätzten saisonalen Standardabweichung (z.B. mit Lowess).

Besser ist es jedoch, schon von vornherein die benachbarten Tage zur Schätzung der Tages-Standardabweichung einzubeziehen. Dazu gehen wir wie folgt vor:

1. Wir tragen für jeden Tag des Jahres die absoluten Residuen auf. Die ist in Abbildung 4.2 illustriert.
2. Durch die absoluten Residuen passen wir einen *Lowess* an (grüne Kurve in Abbildung 4.2). Mit dieser Wahl besitzt die Kurve eine vom meteorologischen Standpunkt aus sinnvolle Glattheit. Um die Periodizität an den Rändern ($\nu = 1, \nu = 365$) zu gewährleisten, fügen wir am linken und rechten Rand nochmals die absoluten Residuen an.
3. Die angepasste Kurve muss mit einer Konstante c multipliziert werden, um eine Schätzung der Standardabweichung zu erhalten.

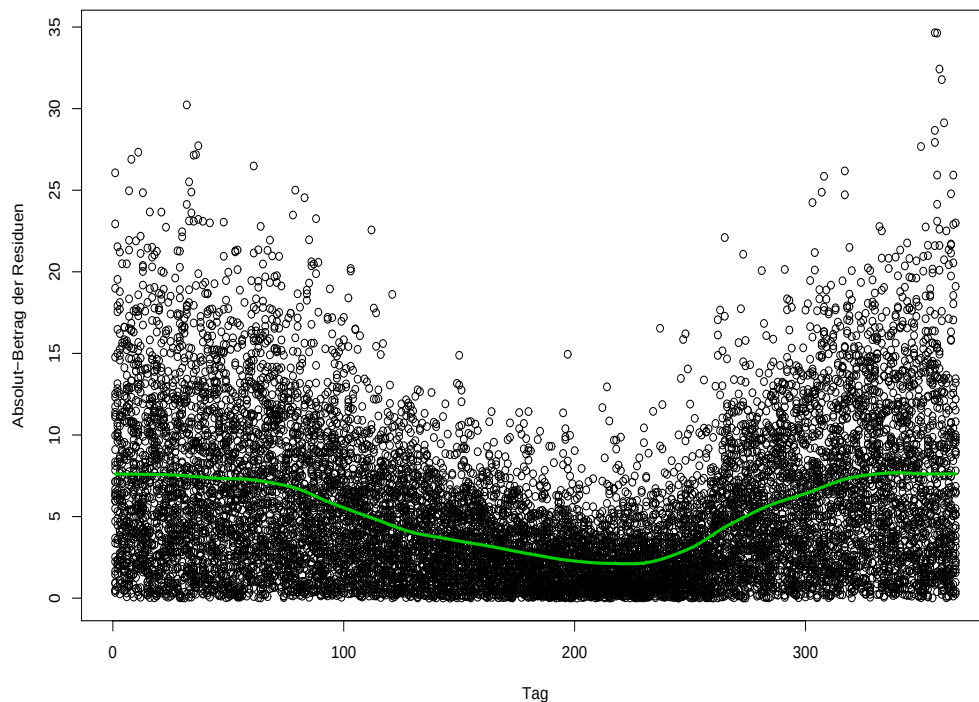


Abbildung 4.2: Berechnung der saisonalen Standardabweichung mit Lowess (Beispiel San Antonio)

Die Verwendung des Absolut-Betrags der Residuen hat zwei Vorteile gegenüber den quadrierten Residuen. Zum einen erhalten wir direkt eine Schätzung der Standardabweichung (und nicht der Varianz). Zum anderen würden Ausreisser durch eine Quadrierung noch mehr Gewicht auf die Standardabweichungsschätzung erhalten.

Gemäss Schritt 3 muss die Schätzung mit einer Konstante multipliziert werden, um die korrekte Standardabweichungsschätzung zu erhalten. Unter der Normalverteilungsannah-

me $R_t \sim \mathcal{N}(0, \sigma_t^2)$ beträgt diese Konstante $\sqrt{\frac{\pi}{2}}$, wie die folgende Rechnung zeigt:

$$\mathbf{E}[|R_t|] = \frac{1}{\sqrt{2\pi}\sigma_t} \int_{-\infty}^{\infty} |x| e^{-\frac{x^2}{2\sigma_t^2}} dx \quad (4.3)$$

$$= \frac{2}{\sqrt{2\pi}\sigma_t} \int_0^{\infty} x e^{-\frac{x^2}{2\sigma_t^2}} dx \quad (4.4)$$

$$\stackrel{\text{Substitution}}{=} \sqrt{\frac{2}{\pi}} \cdot \sigma_t \quad (4.5)$$

Es folgt also, dass $\sigma_t = \sqrt{\frac{\pi}{2}} \cdot \mathbf{E}[|R_t|]$ ist.

Die Multiplikation mit dieser Konstante ist nur von sekundärer Bedeutung. Wir sind hauptsächlich daran interessiert, die Heteroskedastizität der Residuen zu beseitigen, um eine stationäre Zeitreihe zu erhalten. Die Konstante c bewirkt lediglich, dass die Varianz der Residuen eins beträgt.

Der Lowess in Schritt 2 wurde ohne Robustheitsiterationen gewählt. Die Robustheitsiterationen des Lowess benötigen eine konstante Varianz der Residuen, um ein sinnvolles Resultat zu liefern. Ansonsten würden jene Werte, denen eine grosse Varianz zugrunde liegt, als Ausreisser heruntergewichtet. Die absoluten Residuen besitzen aber gerade eine heteroskedastische Struktur, die wir durch den Lowess schätzen wollen. Deshalb würden die grossen Varianzen systematisch zu klein geschätzt.

Die Abbildung 4.3 bestätigt, dass die Standardabweichungsschätzung beim Lowess mit Robustheitsiterationen systematisch zu tief liegt. Zudem nimmt die Schätzung an den Rändern, wo die grössten Residuen auftreten, ab. Dies erklärt sich dadurch, dass diese grossen Residuen Gewicht null erhalten.

4.1.3 Schätzung der saisonalen Standardabweichung mit Fourieranalyse

Die in Kapitel 3 vorgestellte Fourieranalyse zur Schätzung des Saisoneffekts lässt sich auch auf die Schätzung der saisonalen Standardabweichung anwenden.

Dabei gehen wir wie folgt vor:

1. Wir tragen für jeden Tag des Jahres die absoluten Residuen auf, wie in Abbildung 4.2 illustriert.
2. Durch die absoluten Residuen passen wir eine Funktion der Form

$$\tilde{f}(t) = a_0 + \sum_{j=1}^k (a_j \cos(\omega_j t) + b_j \sin(\omega_j t))$$

mit $\omega_j = j \cdot \frac{2\pi}{365}$ an.

3. Die angepasste Kurve muss mit einer Konstante c multipliziert werden (wobei $c = \sqrt{\frac{\pi}{2}}$ bei Normalverteilung), um eine Schätzung der Standardabweichung zu erhalten.

Aus denselben Gründen wie im obigen Abschnitt ist hier die Heruntergewichtung der grossen Residuen nicht angebracht. Deshalb verzichten wir auf eine robuste Regression.

Da der Mittelwert der absoluten Residuen nicht null ist, muss hier im Gegensatz zur Saisoneffekt-Schätzung noch ein Intercept a_0 geschätzt werden.

Für die Wahl des Parameters k betrachten wir wiederum die Koeffizienten für verschiedene Werte von k (am Beispiel San Antonios):

	a_0	a_1	b_1	a_2	b_2	a_3	b_3	a_4	b_4
$k = 1$	2.92	1.55	0.34						
$k = 2$	2.92	1.55	0.34	-0.08	-0.21				
$k = 3$	2.92	1.55	0.34	-0.08	-0.21	-0.18	0.00		
$k = 4$	2.92	1.55	0.34	-0.08	-0.21	-0.18	0.00	0.05	-0.02

Tabelle 4.1: Fourierkoeffizienten für verschiedene Werte von k (San Antonio)

Die Koeffizienten und der Vergleich der geschätzten Standardabweichungs-Kurven deuten auf eine Wahl von $k = 3$ hin. Die Veränderung bei höheren Werten von k ist nur noch geringfügig. Auch an den anderen Stationen bewährt sich die Wahl $k = 3$.

In der Abbildung 4.3 sehen wir, dass der Unterschied zwischen der Standardabweichungsschätzung mit dem Lowess (ohne Robustheitsiterationen) und mit der Fourieranalyse ($k = 3$) nur gering ist. Beide Verfahren würden sich für die Standardabweichungsschätzung eignen. Im weiteren Vorgehen verwenden wir aber die Schätzung mittels Lowess.

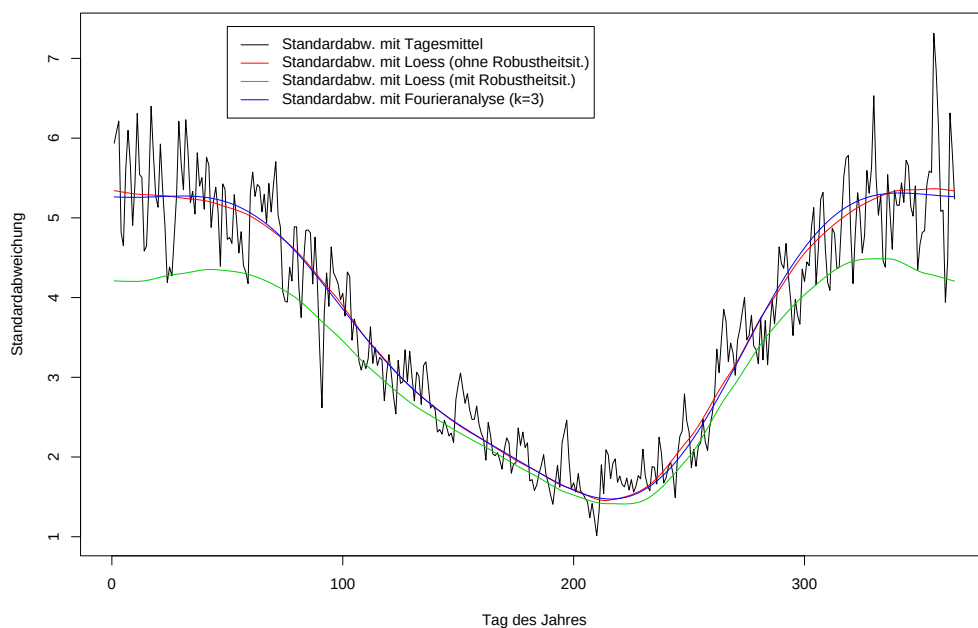


Abbildung 4.3: Vergleich der verschiedenen Schätzungsmethoden der Heteroskedastizität der Residuen

4.1.4 Diskussion

In Abbildung 4.3 erkennt man deutlich die saisonale Abhängigkeit der Standardabweichung. Im Winter ist die Standardabweichung höher als im Sommer. So finden in den Wintermonaten einerseits Kälteeinbrüche statt, andererseits können auch Temperaturen von bis zu 20 °C vorkommen.

Die Form der saisonalen Standardabweichung ist bei allen Standorten ähnlich wie in San Antonio, jedoch ist der Unterschied zwischen Sommer und Winter nicht überall gleich ausgeprägt. Deshalb ist die saisonale Abhängigkeit der Standardabweichung nicht überall schon aus den Residuen ersichtlich (z.B. in Zürich).

In Abbildung 4.4 sehen wir die mit der saisonalen Standardabweichung normierten Residuen I_t von San Antonio. Im Vergleich mit Abbildung 4.1 ist die saisonale Abhängigkeit nun grösstenteils verschwunden.

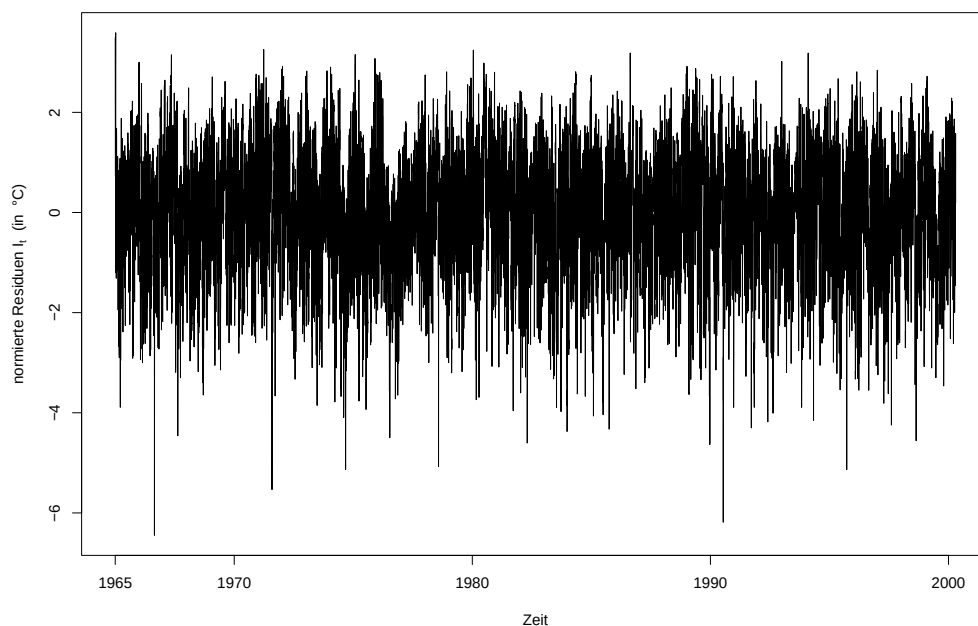


Abbildung 4.4: Normierte Residuen I_t von San Antonio

Das neue Modell für die Temperaturzeitreihen sieht nun folgendermassen aus:

$$T_t = m_t + s_t + \sigma_t \cdot I_t. \quad (4.6)$$

4.2 Der modifizierte STL-Algorithmus

Die Heteroskedastizität der Residuen R_t hat auch Konsequenzen auf die Schätzung von Trend und Saisoneffekt. Das in Kapitel 3 vorgestellte STL-Verfahren geht von Residuen mit konstanter Varianz aus. Die Heteroskedastizität der Residuen muss in die Gewichtung

bei der lokalen Regression einfließen.

Die Schritte 1) - 6) der $(k + 1)$ -ten Iteration des STL-Algorithmus aus Kapitel 3 bleiben unverändert.

Die übrigen Schritte sehen nun folgendermassen aus:

7. *Saisonale Standardabweichung $\hat{\sigma}_t^{(k+1)}$ berechnen*

Um die Heteroskedastizität zu berücksichtigen, muss zunächst die saisonale Standardabweichung von $R_t^{(k+1)}$ geschätzt werden. Dies geschieht mittels der in Abschnitt (4.1.2) vorgestellten Methode.

Da die Residuen $R_t^{(k+1)}$ nach jeder Iteration verschieden sind, muss auch die Standardabweichung $\hat{\sigma}_t^{(k+1)}$ jedesmal neu geschätzt werden.

8. *Robustheitsgewichte unter Berücksichtigung der Heteroskedastizität berechnen*

Die Robustheitsgewichte in Kapitel 3 hatten die Form

$$\varrho_t^{(k+1)} = B\left(\frac{|R_t^{(k+1)}|}{h}\right),$$

wobei

$$B(u) = \begin{cases} (1 - u^2)^2 & 0 \leq |u| \leq 1 \\ 0 & |u| > 1 \end{cases}$$

und

$$h = 6 \cdot \text{median}(|R_t|^{(k+1)}).$$

Diese Robustheitsgewichte bewirkten eine Heruntergewichtung von Ausreissern. Dabei erhielten nur jene Beobachtungen ein Gewicht, die kleiner als ein Vielfaches der (mittels MAD) geschätzten Standardabweichung waren.

Da wir es mit heteroskedastischen Residuen zu tun haben, ist dieses Vorgehen nicht exakt. Stattdessen müssen die Robustheitsgewichte aufgrund der normierten Residuen berechnet werden, d.h.

$$\varrho_t^{(k+1)} = B\left(\frac{|R_t^{(k+1)}|/\sigma_t^{(k+1)}}{h}\right),$$

und

$$h = 6 \cdot \text{median}(R_t^{(k+1)}/\sigma_t^{(k+1)}).$$

Erst jetzt kann bei einem grossen Residuum von einem Ausreisser gesprochen werden.

9. *Gewichtung des Lowess mit der saisonalen Standardabweichung*

Die lokale Regression beim Lowess basiert auf der Kleinsten Quadrate-Methode. Dabei wird ausgehend vom Modell $\mathbf{y} = \mathbf{X}\theta + \varepsilon$ die Summe

$$\sum_{i=1}^n (\mathbf{y} - \mathbf{X}\tilde{\theta})_i^2$$

nach dem Parameter $\tilde{\theta}$ minimiert. Vorausgesetzt wird dabei eine konstante Fehlervarianz. Sonst wird die Schätzung schnell einmal ungenau.

Bei Heteroskedastizität der ε_i , d.h. $\text{Var}(\varepsilon_i) = \sigma_i^2$, müssen die normierten Residuen minimiert werden. Die Parameter werden also durch

$$\underset{\tilde{\theta}}{\text{argmin}} = \sum_{i=1}^n \left(\frac{\mathbf{y} - \mathbf{X}\tilde{\theta}}{\sigma_i} \right)_i^2$$

geschätzt.

Die Normierung der Residuen kann auch als Gewichtung der Regression mit $\frac{1}{\sigma_i^2}$ interpretiert werden. Dadurch werden Werte mit grosser Standardabweichung weniger stark gewichtet. Zu den Robustheitsgewichten führen wir deshalb im Lowess noch die Gewichtung mit $\frac{1}{\sigma_i^2}$ ein. Es gilt also

$$w_t^{(k+1)} = \frac{1}{\sigma_t^2} \cdot \varrho_t^{(k+1)}.$$

Die Abbildung 4.5 gibt eine Zusammenfassung des modifizierten STL-Verfahrens.

1. Entfernung der Trendkomponente (falls $k > 0$): $D_t^{(k+1)} = T_t - \hat{m}_t^{(k)}$
2. Glättung und Extrapolation der Trend-bereinigten Tageszeitreihen $D_{\nu,j}^{(k+1)} \rightarrow C_t^{(k+1)}$
3. Low-Pass-Filter auf $C_t^{(k+1)}$ anwenden $\rightarrow L_t^{(k+1)}$
4. Berechnung der Saisonkomponente: $\hat{s}_t^{(k+1)} = C_t^{(k+1)} - L_t^{(k+1)}$
5. Entfernung der Saisonkomponente: $E_t^{(k+1)} = T_t - \hat{s}_t^{(k+1)}$
6. Anpassung des Trends an $E_t^{(k+1)} \rightarrow \hat{m}_t^{(k+1)}$
7. Saisonale Standardabweichung von $R_t^{(k+1)} = T_t - \hat{m}_t^{(k+1)} - \hat{s}_t^{(k+1)}$ berechnen $\rightarrow \hat{\sigma}_t^{(k+1)}$
8. Robustheitsgewichte unter Berücksichtigung der Heteroskedastizität berechnen $\rightarrow \varrho_t^{(k+1)}$
9. Verwende die Gewichte $w_t^{(k+1)} := \frac{1}{\hat{\sigma}_t^{(k+1)^2}} \cdot \varrho_t^{(k+1)}$ beim Lowess in 2), 3) und 6)

Abbildung 4.5: Modifizierter STL-Algorithmus: $(k + 1)$ -ter Durchgang

Um diese Modifikation des STL-Algorithmus zu implementieren, wurde der in FORTRAN vorliegende STL-Code nach C übersetzt. Danach konnte die Änderung in C vorgenommen werden. Im Anhang (A 3.2 und B) findet sich der Quellcode des modifizierten STL-Verfahrens.

Der Einfluss der Modifikation des STL-Verfahrens hängt von der Ausprägung der Saisonalität der Standardabweichung ab. In Abbildung 4.6 werden das STL- und das modifizierte STL-Verfahren miteinander am Beispiel San Antonios verglichen. Da die Standardabweichung in San Antonio eine starke Saisonalität aufweist, ist der Unterschied in der

Trendschätzung relativ gross. In Zürich hingegen, wo die Heteroskedastizität der Residuen kleiner ist, ist der Unterschied in der Trendschätzung gering (Abbildung 4.7).

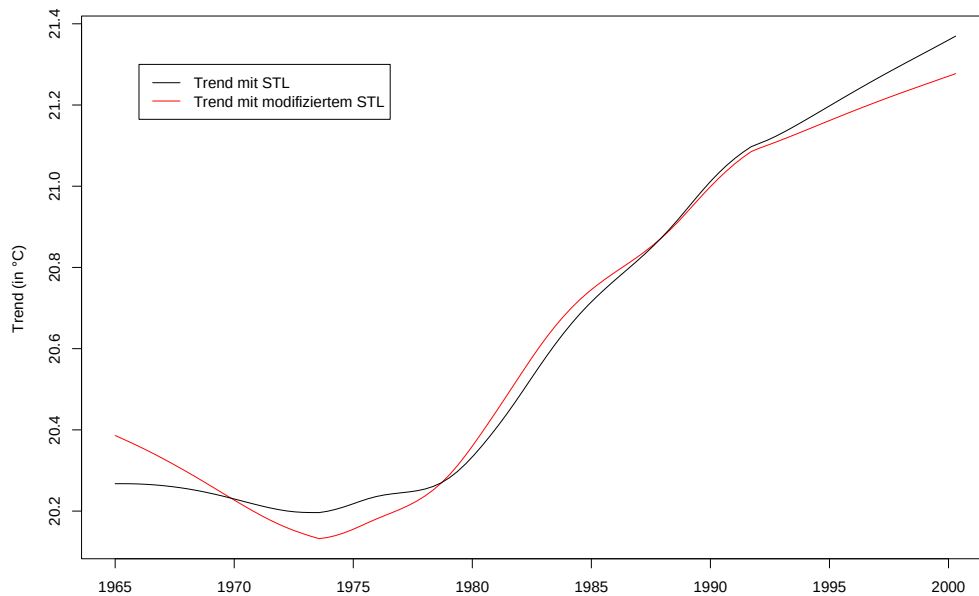


Abbildung 4.6: Unterschied STL / modifizierter STL in San Antonio

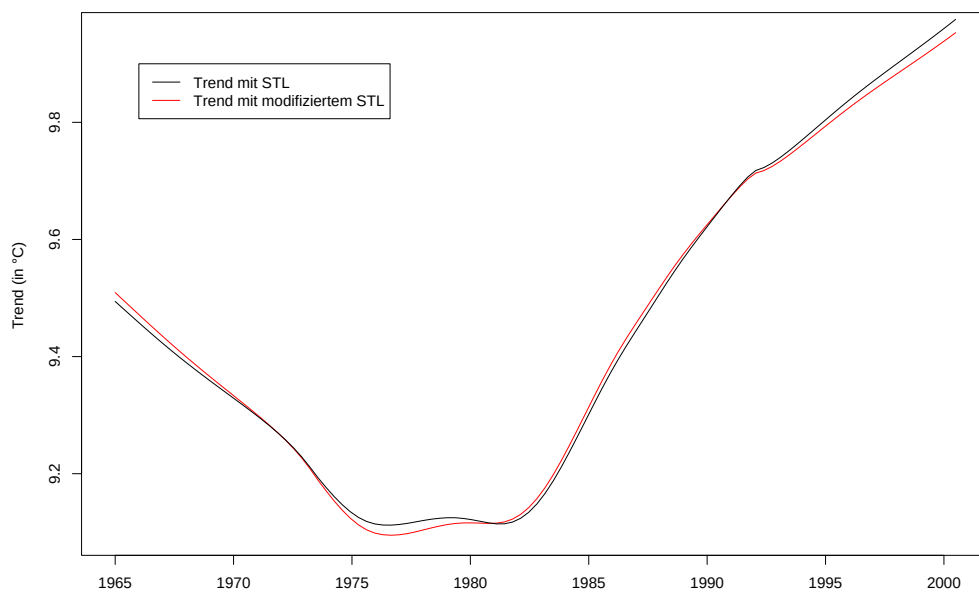


Abbildung 4.7: Unterschied STL / modifizierter STL in Zürich

Kapitel 5

Die Box-Cox Transformation

Für die weitere Analyse und Modellierung der Residuen I_t benötigen wir Kenntnisse über die Form ihrer Verteilung. Insbesondere interessiert uns die Abweichung von der Normalverteilung.

Um diese Abweichung zu erkennen, bedienen wir uns der grafischen Hilfsmittel QQ-Plot und Histogramm. Der **QQ-Plot** prüft, ob die Normalverteilungsannahme zutrifft. Wenn die Daten $I_1, \dots, I_n$ vorliegen, dann ist die empirische kumulative Verteilungsfunktion durch

$$F_n(x) = \frac{1}{n} \cdot \text{Anzahl}(I_i \leq x)$$

gegeben. Falls die I_i einer Normalverteilung $\mathcal{N}(\mu, \sigma^2)$ gehorchen, gilt nach dem Lemma von Glivenko-Cantelli für grosse n

$$F_n(x) \approx \Phi\left(\frac{x - \mu}{\sigma}\right)$$

wobei Φ die kumulative Verteilungsfunktion der Normalverteilung ist.

Wir definieren

$$z := \Phi^{-1}(F_n(x)) \approx \frac{x - \mu}{\sigma}$$

.

Beim QQ-Plot wird nun x gegen z aufgetragen. Falls die Normalverteilungsannahme zutrifft, sollten wir ungefähr eine Gerade erhalten.

Der QQ-Plot der normierten Residuen I_t von Atlanta in Abbildung 5.1 weicht deutlich von der Geraden ab und weist auf eine schiefe Verteilung hin. Dies bestätigt sich auch, wenn wir das Histogramm betrachten. Die Schiefe der Verteilung ist je nach Ort mehr oder weniger ausgeprägt.

Mit einer **Box-Cox Transformation** versuchen wir, diese Schiefe zu beheben.

Bei den Box-Cox Transformationen handelt es sich um eine parametrische Familie von Transformationen von y (> 0) nach $y^{(\lambda)}$, die folgendermassen definiert sind:

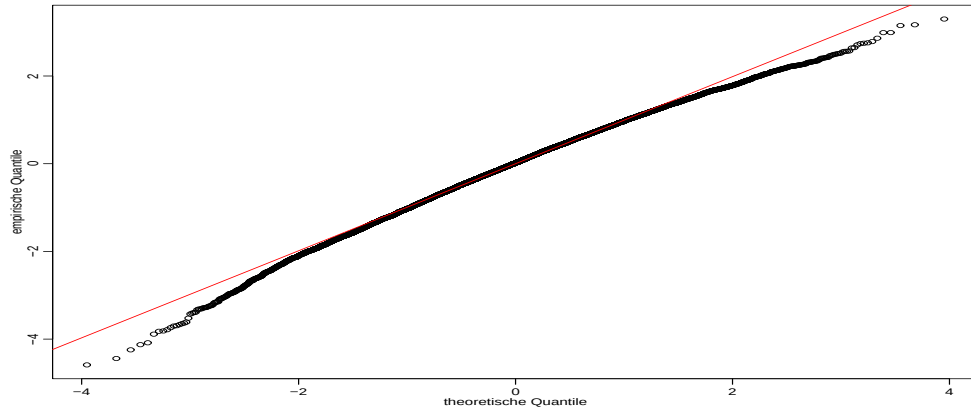


Abbildung 5.1: QQ-Plot der normierten Residuen von Atlanta

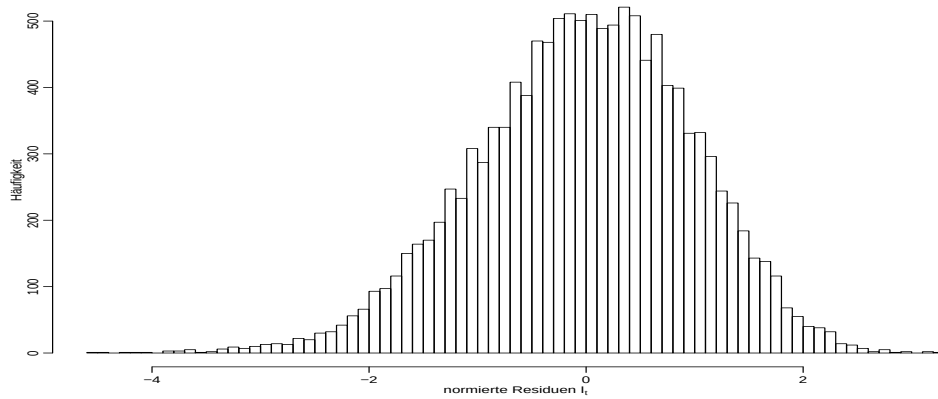


Abbildung 5.2: Histogramm der normierten Residuen von Atlanta

$$y^{(\lambda)} = \begin{cases} \frac{y^\lambda - 1}{\lambda} & (\lambda \neq 0) \\ \ln y & (\lambda = 0) \end{cases}$$

Der Fall $\lambda \neq 0$ entspricht bis auf eine lineare Transformation einer Potenzierung. Die Transformation wurde so gewählt, dass wir einen stetigen Übergang zum Logarithmus ($\lambda = 0$) haben:

$$\lim_{\lambda \rightarrow 0} \frac{y^\lambda - 1}{\lambda} = \lim_{\lambda \rightarrow 0} \frac{e^{\lambda \cdot \ln y} - 1}{\lambda} \stackrel{BdH}{=} \lim_{\lambda \rightarrow 0} \frac{\ln y \cdot y^\lambda}{1} = \ln y$$

In Abbildung 5.3 ist die Transformationsfunktion für verschiedene Parameter λ aufgetragen.

Bei der Box-Cox Transformation gilt es, den für die Daten optimalen Parameter λ zu ermitteln. Im Folgende leiten wir ein Kriterium zur Parameterwahl her.

Wir gehen dabei von einem linearen Regressionsmodell aus. Dabei seien $\mathbf{y} = (y_1, \dots, y_n)$ die abhängigen Variablen, wobei alle y_i positiv sind. $\mathbf{y}^{(\lambda)}$ seien die Box-Cox transformierten

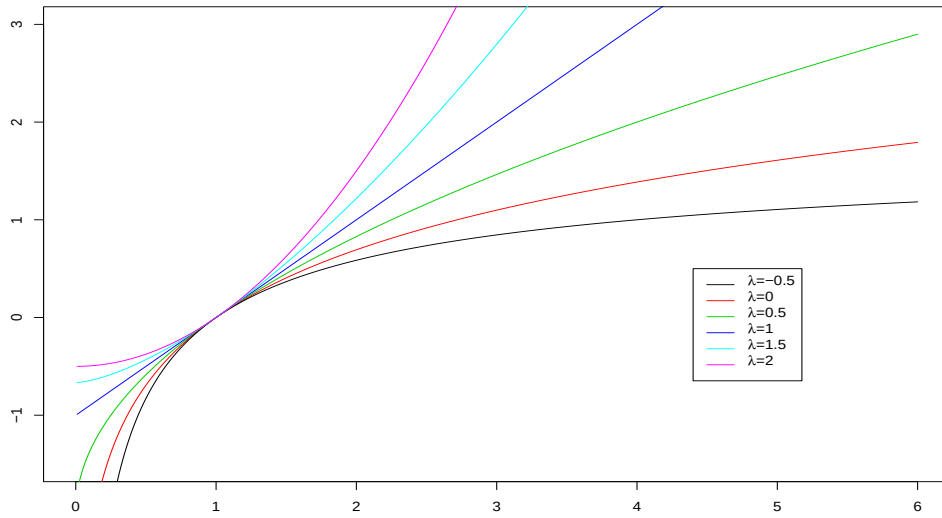


Abbildung 5.3: Box-Cox Transformation für verschiedene Parameter λ

abhängigen Variablen $\mathbf{y}$. Wir nehmen an, dass es einen Parameter λ_{opt} gibt, so dass $\mathbf{y}^{(\lambda_{opt})}$ die folgenden Bedingungen erfüllt:

1. $\mathbf{E}[\mathbf{y}^{(\lambda_{opt})}] = \mathbf{X}\boldsymbol{\theta}$, wobei $\mathbf{X}$ die Matrix der abhängigen Variablen ist und $\boldsymbol{\theta}$ der Parametervektor der linearen Regression.
2. Die transformierten Beobachtungen $\mathbf{y}^{(\lambda_{opt})}$ sind unabhängig normalverteilt mit konstanter Varianz σ^2 .

Die Wahrscheinlichkeitsdichte für $\mathbf{y}^{(\lambda_{opt})}$ unter diesen Annahmen wäre

$$\frac{1}{\sqrt{2\pi}^n \cdot \sigma^n} \cdot \exp\left(-\frac{(\mathbf{y}^{(\lambda_{opt})} - \mathbf{X}\boldsymbol{\theta})^T (\mathbf{y}^{(\lambda_{opt})} - \mathbf{X}\boldsymbol{\theta})}{2 \cdot \sigma^2}\right)$$

Für die untransformierten Beobachtungen $\mathbf{y}$ lautet die Wahrscheinlichkeitsdichte dann

$$f_{\lambda_{opt}, \sigma}(\mathbf{y}) = \frac{1}{\sqrt{2\pi}^n \cdot \sigma^n} \cdot \exp\left(-\frac{(\mathbf{y}^{(\lambda_{opt})} - \mathbf{X}\boldsymbol{\theta})^T (\mathbf{y}^{(\lambda_{opt})} - \mathbf{X}\boldsymbol{\theta})}{2 \cdot \sigma^2}\right) \cdot J(\lambda_{opt}, \mathbf{y}). \quad (5.1)$$

wobei

$$J(\lambda_{opt}, \mathbf{y}) = \prod_{i=1}^n \left| \frac{dy_i^{(\lambda_{opt})}}{dy_i} \right|$$

die Jakobi-Determinante ist. Bei gegebenem $\mathbf{y}$ ist $f_{\lambda_{opt}, \sigma}(\mathbf{y})$ auch die Likelihoodfunktion für die Parameter $\boldsymbol{\theta}$, σ und λ .

Bei gegebenem λ entspricht der Maximum-Likelihood Schätzer

$$\hat{\boldsymbol{\theta}}(\lambda) = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}^{(\lambda)} \quad (5.2)$$

für $\boldsymbol{\theta}$ der kleinsten Quadrate Schätzung. Die Maximum-Likelihood Schätzung von σ^2 bei festem λ ist durch

$$\hat{\sigma}^2(\lambda) = \frac{1}{n} \cdot \mathbf{y}^{(\lambda)T} \tilde{\mathbf{X}}^T \tilde{\mathbf{X}} \mathbf{y}^{(\lambda)} = \frac{RSS(\mathbf{y}^{(\lambda)})}{n} \quad (5.3)$$

gegeben, wobei

$$\tilde{\mathbf{X}} = \mathbf{I} - \mathbf{X}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T$$

und $RSS(\mathbf{y}^{(\lambda)})$ die Quatratsumme der Residuen im an $\mathbf{y}^{(\lambda)}$ angepassten Modell ist.

Für festes λ ist die nach $\boldsymbol{\theta}$ und σ maximierte Log-Likelihoodfunktion also

$$\begin{aligned} \ell_{\max}(\lambda) &\stackrel{(5.1)}{=} \ln \left((2\pi)^{-\frac{n}{2}} \cdot \hat{\sigma}(\lambda)^{-n} \cdot \exp \left(- \frac{(\mathbf{y}^{(\lambda)} - \mathbf{X}\hat{\boldsymbol{\theta}}(\lambda))^T (\mathbf{y}^{(\lambda)} - \mathbf{X}\hat{\boldsymbol{\theta}}(\lambda))}{2 \cdot \hat{\sigma}(\lambda)^2} \right) \cdot J(\lambda, \mathbf{y}) \right) \\ &\stackrel{(5.2), (5.3)}{=} \ln((2\pi)^{-\frac{n}{2}}) + \ln(\hat{\sigma}(\lambda)^{-n \cdot \frac{2}{2}}) - \frac{n \cdot \hat{\sigma}^2(\lambda)}{2 \cdot \hat{\sigma}^2(\lambda)} + \ln(J(\lambda, \mathbf{y})) \\ &= -\frac{n}{2} \cdot \ln \hat{\sigma}(\lambda)^2 + \ln J(\lambda, \mathbf{y}) + konst. \end{aligned}$$

Das optimale λ_{opt} wird also durch die Maximierung von

$$\ell(\lambda) = -\frac{n}{2} \cdot \ln \hat{\sigma}(\lambda)^2 + \ln J(\lambda, \mathbf{y}).$$

nach λ geschätzt.

Betrachten wir nun wieder die normierten Residuen I_t der Temperaturzeitreihe am Beispiel von Atlanta. Um eine Box-Cox Transformation anzuwenden, muss die zu transformierende Reihe positiv sein. Deshalb addieren wir eine Konstante α ($=12$) zu $-I_i$, d.h.

$$\mathbf{y}_i = -I_i + \alpha$$

Der Grund dafür, dass wir $-I_i$ statt I_i verwenden, liegt darin, dass die positiven Residuen weniger gross sind als die negativen. Deshalb können wir ein kleineres α verwenden. Um eine Schätzung $\hat{\lambda}$ von λ_{opt} zu erhalten, muss nach dem oben beschriebenen Vorgehen ein lineares Modell an die transformierte Grösse $\mathbf{y}^{(\lambda)}$ angepasst werden. Da wir nur normalverteilten Residuen erhalten wollen, wird ein Lokationsmodell gewählt, d.h. $\mathbf{X} = (1, \dots, 1)^T$. An der Stelle, wo $\ell(\lambda)$ den maximalen Wert annimmt, trifft die Normalverteilungs-Annahme am besten zu.

In Kapitel 6 werden wir sehen, dass die Residuen nicht unabhängig sind. Daher ist die Bedingung 2) für jedes λ nur approximativ erfüllt.

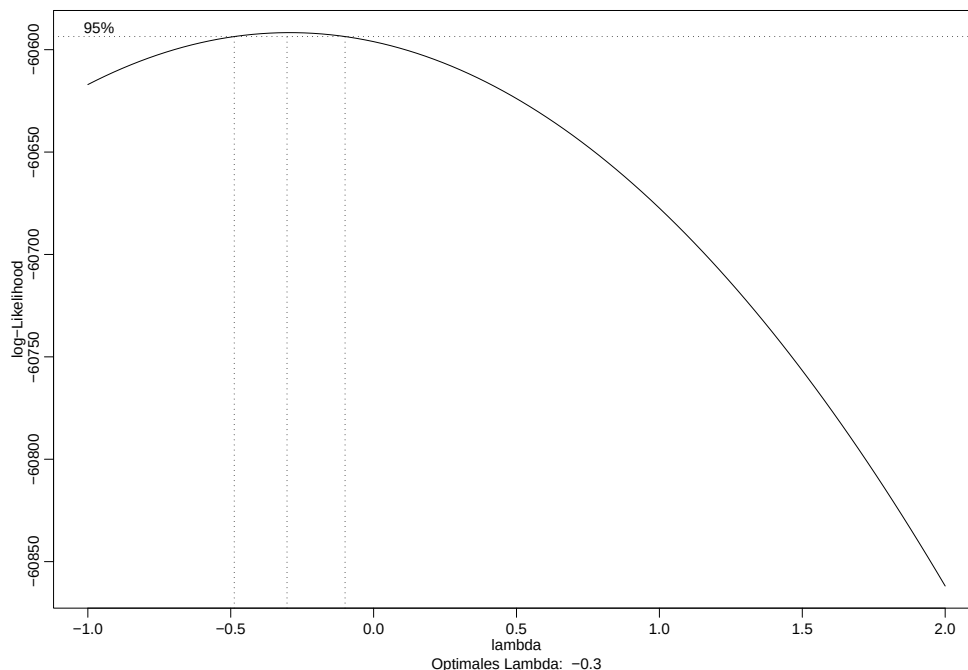


Abbildung 5.4: Log-Likelihoodfunktion für λ am Beispiel von Atlanta

Die Maximierung der Likelihood-Funktion erfolgt durch die Auswertung von $\ell(\lambda)$ an den Werten im Intervall $[-1, 2]$. Die Stelle mit dem maximalen Wert ergibt das geschätzte $\hat{\lambda}$.

In der Abbildung 5.4 ist Funktion $\ell(\lambda)$ von Atlanta aufgetragen. Hier wird der Transformationsparameter $\hat{\lambda} = -0.3$ gewählt. In dieser Abbildung ist auch ein 95% Vertrauensintervall für λ_{opt} eingezeichnet. Ein $100 \cdot (1 - \beta) \%$ Vertrauensintervall für λ_{opt} enthält alle Werte $\tilde{\lambda}$, die die folgende Beziehung erfüllen:

$$|\ell(\hat{\lambda}) - \ell(\tilde{\lambda})| < \frac{1}{2} \cdot \chi_{1, 1-\beta}^2.$$

Dabei ist $\chi_{1, 1-\beta}^2$ das $(1 - \beta)$ -Quantil der χ^2 -Verteilung mit einem Freiheitsgrad. Eine Herleitung dieses sogenannten Likelihoodquotienten-Vertrauensintervalls findet sich in [11].

Die Temperatur-Residuen $I_1, \dots, I_n$ haben nach der Box-Cox Transformation den Mittelwert $\hat{\mu}_{(b)}$. Nach Abzug dieses Mittelwerts und des Multiplikation mit -1 (wir haben $-I_t + \alpha - I_t$ transformiert) bezeichnen wir die Residuen mit $x_1, \dots, x_n$.

In Abbildung 5.5 sehen wir den QQ-Plot und das Histogramm der Residuen x_t von Atlanta, die aus einer Box-Cox Transformation mit $\hat{\lambda} = -0.3$ hervorgehen. Man erkennt im Vergleich zu Abbildung 5.1 eine deutliche Verbesserung hinsichtlich der Schiefe.

Die Box-Cox Transformation wenden wir auf alle Standorte an. Ähnlich wie oben stellt sich auch dort eine Verbesserung hinsichtlich der Schiefe ein. Die optimalen Parameter unterscheiden sich sehr stark für die verschiedenen Stationen. Deshalb kann kein einheitlicher Transformationsparameter gewählt werden. Immerhin zeigt sich innerhalb den im

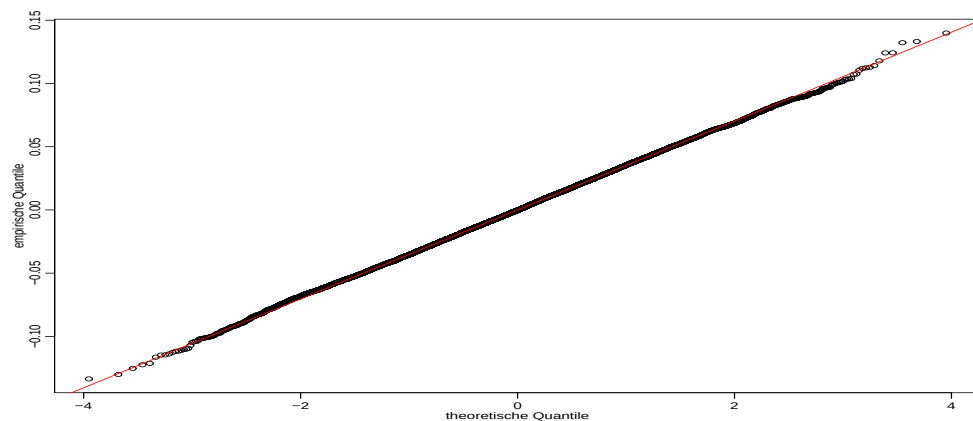


Abbildung 5.5: QQ-Plot der normierten Residuen von Atlanta nach der Box-Cox Transformation

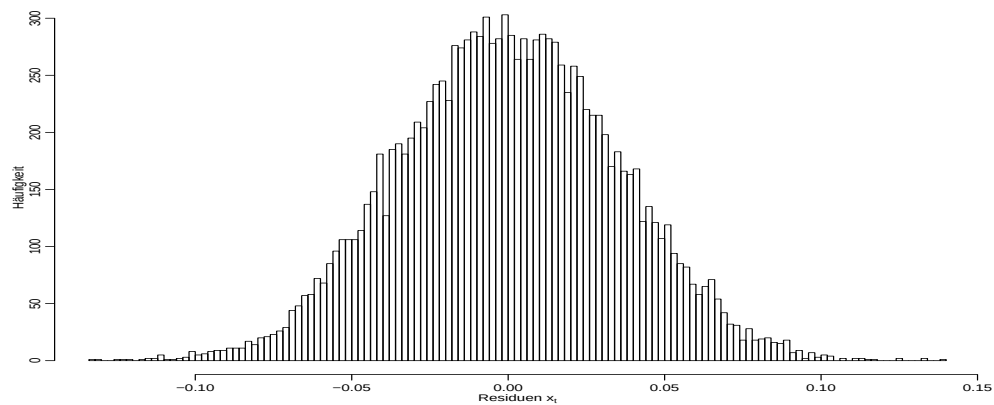


Abbildung 5.6: Histogramm der normierten Residuen von Atlanta nach der Box-Cox Transformation

Kapitel 2 gebildeten Gruppen ein einheitlicheres Bild. Im Südosten der USA werden zum Beispiel Werte im Intervall $[-0.7, -0.2]$ gewählt. Im Nordosten Amerikas liegen die Werte im Bereich $[1.2, 1.7]$.

Der Kolmogorov-Smirnov Test zeigt auf, dass bei einigen Orten die Normalverteilungsannahme verworfen werden muss. Bei einer zu grossen Anzahl von Ausreissern nach unten versagt die Box-Cox Transformation (vgl. Vancouver im Anhang C).

Bei der Schätzung $\hat{\lambda}$ werden die drei extremsten Beobachtungen an den Rändern nicht verwendet. Dies bezweckt, dass extreme Beobachtungen, die nicht mit einer Normalverteilung vereinbar sind, die Schätzung $\hat{\lambda}$ nicht verfälschen.

Kapitel 6

Anpassung eines autoregressiven Modells

Die Temperatur eines Tages hängt auch von der vorangehenden Tage ab, da die Witterungsbedingungen oft über einige Tage stabil bleiben. Diese Abhängigkeit ist noch in den Residuen x_t vorhanden. Um die Abhängigkeit statistisch zu untersuchen, führen wir die Autokorrelationsfunktion ein.

6.1 Die Autokorrelationsfunktion

Wir gehen davon aus, dass die Temperatur-Residuen x_t die Realisation eines stationären stochastischen Prozesses $(X_t)_{t \in \mathbb{Z}}$ mit Erwartungswert 0 ist. Die Autokorrelationsfunktion dieses Prozesses ist folgendermassen definiert:

Definition 6.1. Sei $(X_t)_{t \in \mathbb{Z}}$ ein stationärer Prozess mit $E[X_t^2] < \infty$ und sei $k \in \mathbb{Z}$. Definiert wird die **Autokorrelationsfunktion (ACF)** durch

$$\rho(k) = \frac{R(k)}{R(0)}$$

wobei

$$R(k) = \text{Cov}(X_t, X_{t+k})$$

die **Autokovarianzfunktion** darstellt.

Die Autokorrelationsfunktion $\rho(k)$ misst also die Korrelation zwischen zwei um k Zeitschritte (Tage) entfernten Temperaturen.

Da wir in unserem Fall den zugrunde liegenden stochastischen Prozess $(X_t)_{t \in \mathbb{Z}}$ nicht kennen, müssen wir die Autokorrelationsfunktion $\rho(k)$ aus den vorhandenen Beobachtungen $x_1, \dots, x_n$ schätzen:

$$\hat{\rho}(k) = \frac{\hat{R}(k)}{\hat{R}(0)}$$

wobei

$$\hat{R}(k) = \frac{1}{n} \sum_{t=1}^{n-|k|} (x_t - \bar{x})(x_{t+|k|} - \bar{x}).$$

Beim Schätzer der Autokovarianz $\hat{R}(k)$ wird $\frac{1}{n}$ anstelle von $\frac{1}{n-|k|}$ verwendet. Damit will man erreichen, dass die geschätzte Kovarianzmatrix positiv definit wird.

In Abbildung 6.1 ist die Autokorrelationsfunktion am Beispiel der Residuen von Zürich aufgetragen. Sie zeigt uns, dass die Temperatur eines Tages wie erwartet von den vorhergehenden abhängt. Denn nur wenn sich die ACF innerhalb des blau eingezeichneten 95%-Vertrauensintervalls befindet, darf die Nullhypothese der Unkorreliertheit beibehalten werden.

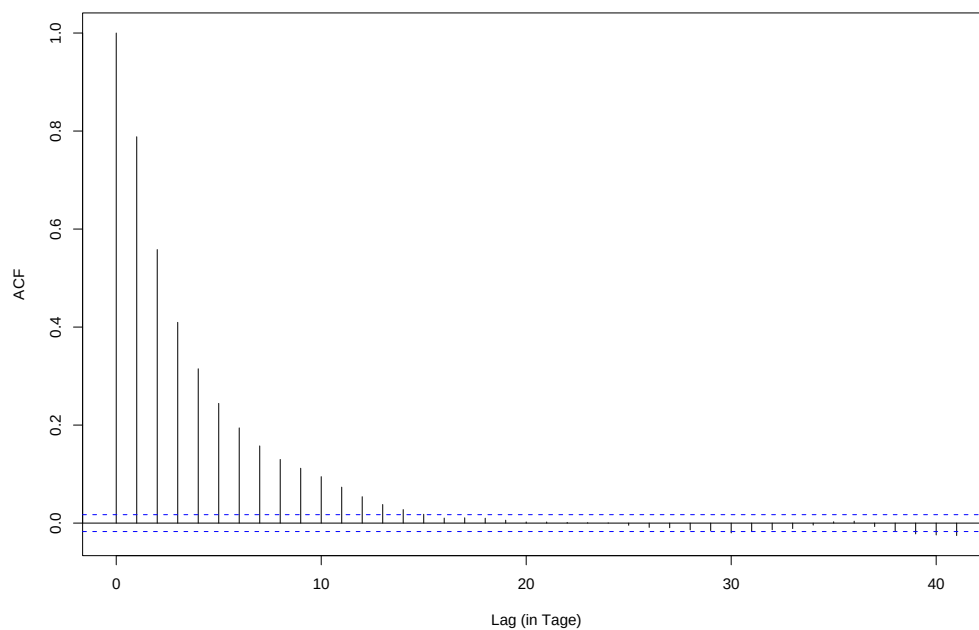


Abbildung 6.1: Autokorrelationsfunktion der Residuen x_t von Zürich

Diese Abhängigkeit der Residuen gilt es in unserem Modell zu berücksichtigen. Dies realisieren wir mittels eines autoregressiven Modells.

6.2 AR-Modelle

Definition 6.2. Ein stochastischer Prozess $(X_t)_{t \in \mathbb{Z}}$ heisst **autoregressiv** der Ordnung p , kurz **AR(p)**, falls

$$X_t = \sum_{j=1}^p \phi_j \cdot X_{t-j} + \varepsilon_t$$

wobei ε_t iid mit $\mathbf{E}[\varepsilon_t] = 0$ und $\text{Var}(\varepsilon_t) = \sigma_\varepsilon^2$.

Definition 6.3. Ein AR(p)-Prozess heisst **kausal**, falls die ε_t von allen X_u mit $u < t$ unabhängig sind.

Die Parameter ϕ_j bestimmen, auf welche Weise der Prozess von den vorangehenden Beobachtungen abhängt.

Es gilt nun, an die Zeitreihe der Residuen x_t einen kausalen autoregressiven Prozess anzupassen. Dabei müssen

1. die Ordnung p
2. die Parameter $\phi_1, \dots, \phi_p$

geschätzt werden.

Zuerst widmen wir uns der Bestimmung der Parameter $\phi_1, \dots, \phi_p$. Dazu nehmen wir an, dass die AR-Ordnung p bekannt sei.

6.3 Bestimmung der Parameter des AR-Modells

Zunächst betrachten wir die Autokovarianzfunktion $R(k)$ eines AR(p) Modells.

Für $k > 0$ gilt:

$$\begin{aligned} R(k) &= \text{Cov}(X_t, X_{t+k}) \\ &= \mathbf{E}[X_t \cdot X_{t+k}] \\ &= \mathbf{E}\left[X_t \cdot \left(\sum_{j=1}^p \phi_j \cdot X_{t+k-j} + \varepsilon_{t+k}\right)\right] \\ &= \sum_{j=1}^p \phi_j \cdot \mathbf{E}[X_t \cdot X_{t+k-j}] \\ &= \sum_{j=1}^p \phi_j \cdot R(k-j) \end{aligned}$$

Im Fall $k = 0$ gilt:

$$\begin{aligned}
R(0) &= \mathbf{E}[X_t^2] \\
&= \mathbf{E}[X_t \cdot (\sum_{j=1}^p \phi_j \cdot X_{t-j} + \varepsilon_t)] \\
&\stackrel{\text{Kausalität}}{=} \sum_{j=1}^p \phi_j \cdot R(j) + \sigma_\varepsilon^2
\end{aligned}$$

Sei $\mathbf{\Gamma} = [R(i-j)]_{i,j=1}^p$ die Kovarianzmatrix, $\boldsymbol{\gamma} = (R(1), \dots, R(p))^T$ und $\boldsymbol{\phi} = (\phi_1, \dots, \phi_p)^T$.

Mit dieser Notation haben die oben hergeleiteten Gleichungen folgende Form:

$$\begin{aligned}
\mathbf{\Gamma} \boldsymbol{\phi} &= \boldsymbol{\gamma} \\
R(0) &= \boldsymbol{\phi}^T \boldsymbol{\gamma} + \sigma_\varepsilon^2
\end{aligned}$$

Dieses System von $p+1$ Gleichungen nennt man die **Yule-Walker-Gleichungen**. Wenn $\mathbf{\Gamma}$ und $\boldsymbol{\gamma}$ bekannt wären, könnten wir aus dem Gleichungssystem $\boldsymbol{\phi}$ und σ_ε^2 berechnen. Da dies nicht der Fall ist, verwenden wir die schon eingeführten Momenten-Schätzer $\hat{R}(k)$ und erhalten $\hat{\mathbf{\Gamma}}$ und $\hat{\boldsymbol{\gamma}}$. Daraus ergibt sich der **Yule-Walker-Schätzer** für $\boldsymbol{\phi}$ und σ_ε^2 :

$$\begin{aligned}
\hat{\boldsymbol{\phi}} &= \hat{\mathbf{\Gamma}}^{-1} \hat{\boldsymbol{\gamma}} \\
\hat{\sigma}_\varepsilon^2 &= \hat{R}(0) - \hat{\boldsymbol{\phi}}^T \hat{\boldsymbol{\gamma}}
\end{aligned}$$

Der Schätzer hat unter der Voraussetzung, dass $(X_t)_{t \in \mathbb{Z}}$ ein kausaler $\text{AR}(p)$ Prozess ist, eine asymptotische Normalverteilung:

$$\sqrt{n} \cdot (\hat{\boldsymbol{\phi}} - \boldsymbol{\phi}) \rightarrow \mathcal{N}(0, \sigma_\varepsilon^2 \mathbf{\Gamma}^{-1})$$

Ein Beweis dafür findet sich in [2].

Die folgende Tabelle zeigt die mit dem Yule-Walker-Schätzer bestimmten AR-Koeffizienten von Zürich für verschiedene Ordnungen p :

	ϕ_1	ϕ_2	ϕ_3	ϕ_4	ϕ_5	ϕ_6
$p = 1$	0.79					
$p = 2$	0.92	-0.17				
$p = 3$	0.94	-0.24	0.07			
$p = 4$	0.94	-0.24	0.06	0.01		
$p = 5$	0.94	-0.26	0.06	0.01	0.00	
$p = 6$	0.94	-0.24	0.06	0.01	-0.02	0.02

Tabelle 6.1: AR-Koeffizienten von Zürich für verschiedene Ordnungen p

Die Abhängigkeit von der Vergangenheit ist dabei im wesentlichen durch die letzten drei Tage gegeben. Der Einfluss der weiter zurückliegenden Tage ist eher gering. Eine Gemeinsamkeit aller Stationen ist die starke positive Abhängigkeit vom Vortag und die negative Abhängigkeit des um zwei Tage zurückliegenden Tages.

6.4 Wahl der AR-Ordnung

Die AR-Ordnung kann mit verschiedenen Methoden bestimmt werden. Eine Möglichkeit zur explorativen Modellwahl bietet die partielle Autokorrelationsfunktion.

Zur automatischen Modellselektion kann das Akaike's Information Criterion (AIC) herbeigezogen werden.

6.4.1 Wahl der AR-Ordnung mittels partieller Autokorrelationsfunktion

Ein erster Eindruck von der AR-Ordnung ergibt sich durch das Betrachten der partiellen Autokorrelationsfunktion. Um diese Funktion einzuführen, benötigen wir noch Kenntnisse über die besten Vorhersagen bezüglich der mittleren quadratischen Abweichung. Diese Begriffe werden wir auch im Kapitel 7 wieder antreffen.

Definition 6.4. Sei $\hat{X}_{t|u:v}$ die **beste Vorhersage** von X_t gestützt auf $X_u, \dots, X_v$, d.h.

$$\mathbf{E}\left[\left(X_t - \hat{X}_{t|u:v}\right)^2\right] = \min_{\substack{g(\cdot) \text{ messbar} \\ \mathbf{E}[g(X_u, \dots, X_v)] < \infty}} \mathbf{E}\left[\left(X_t - (g(X_u, \dots, X_v))\right)^2\right]$$

Bei der besten linearen Vorhersage wird der Bereich der zulässigen Funktionen auf die linearen Funktionen eingeschränkt:

Definition 6.5. Sei $\tilde{X}_{t|u:v}$ die **beste lineare Vorhersage** von X_t gestützt auf $X_u, \dots, X_v$, d.h.

$$\mathbf{E}\left[\left(X_t - \tilde{X}_{t|u:v}\right)^2\right] = \min_{(a_0, a_u, \dots, a_v)} \mathbf{E}\left[\left(X_t - \left(a_0 + \sum_{i=u}^v a_i X_i\right)\right)^2\right]$$

Mithilfe des folgenden Satzes kann die beste Vorhersage berechnet werden:

Satz 6.1. Es sei $\mathbf{E}[X_i^2] < \infty \quad \forall i \in \{t, u, \dots, v\}$. Dann ist die beste Vorhersage durch

$$\hat{X}_{t|u:v} = \mathbf{E}[X_t | X_u, \dots, X_v]$$

gegeben, falls diese bedingte Erwartung existiert.

Ein Beweis für diesen nützlichen Satz findet sich in [2].

Wir können nun mit Hilfe der besten linearen Vorhersage die partielle Autokorrelationsfunktion definieren:

Definition 6.6. Die **partielle Autokorrelationsfunktion** (kurz **PACF**) eines stationären stochastischen Prozesses X_t ist

$$\alpha(k) = \text{Cor}(X_t - \tilde{X}_{t|(t+1):(t+k-1)}, X_{t+k} - \tilde{X}_{t+k|(t+1):(t+k-1)}).$$

Die partielle Autokorrelationsfunktion $\alpha(k)$ misst die Korrelation zwischen den X_t und X_{t+k} , wenn der Einfluss der dazwischenliegenden Werte $X_{t+1}, \dots, X_{t+k-1}$ entfernt wurde.

Der folgende Satz zeigt, wie mit Hilfe der partiellen Autokorrelation die AR-Ordnung bestimmt werden kann:

Satz 6.2. Sei X_t ein kausaler $AR(p)$ Prozess. Dann gilt $\alpha(k)=0$ für $k > p$.

Beweis: Für einen kausalen $AR(p)$ Prozess gilt

$$\begin{aligned} \hat{X}_{t|(t+1):(t+k-1)} &\stackrel{\text{Satz 6.1}}{=} \mathbf{E}[X_{t+k}|X_{t+1}, \dots, X_{t+k-1}] \\ &= \mathbf{E}[\phi_1 \cdot X_{t+k-1} + \dots + \phi_p \cdot X_{t+k-p} + \varepsilon_{t+k}|X_{t+1}, \dots, X_{t+k-1}] \\ &\stackrel{\text{Kausalität}}{=} \phi_1 \cdot X_{t+k-1} + \dots + \phi_p \cdot X_{t+k-p}. \end{aligned}$$

Diese beste Vorhersage ist zugleich die beste *lineare* Vorhersage, d.h.

$$\tilde{X}_{t|(t+1):(t+k-1)} = \phi_1 \cdot X_{t+k-1} + \dots + \phi_p \cdot X_{t+k-p}. \quad (6.1)$$

Aus dieser Beziehung folgt

$$\begin{aligned} \alpha(k) &= \text{Cor}(X_t - \tilde{X}_{t|(t+1):(t+k-1)}, X_{t+k} - \tilde{X}_{t+k|(t+1):(t+k-1)}) \\ &\stackrel{(6.1)}{=} \text{Cor}(X_t - \tilde{X}_{t|(t+1):(t+k-1)}, \varepsilon_{t+k}) \\ &\stackrel{\text{Kausalität}}{=} 0 \end{aligned}$$

□

Wenn ein $AR(p)$ -Prozess vorliegt, sollte also die geschätzte partielle Autokorrelation ab der Stelle p etwa null sein. Diese Stelle gilt es zu finden, um eine Schätzung $\hat{p}$ der AR -Ordnung zu erhalten.

In den Abbildungen 6.2 und 6.3 sehen wir die partielle Autokorrelationsfunktionen von Zürich und Vancouver. Die blauen Linien bilden das 95% Vertrauensintervall, das die Nullhypothese $\alpha(k) = 0$ überprüft. Bei Zürich liegen die Werte ab ca. $k = 4$ in diesem Intervall. Also ist nach dem obigen Kriterium die AR -Ordnung 3 für Zürich sinnvoll.

Bei Vancouver liegen die Werte ab ca. $k = 6$ im Vertrauensintervall. Daher wird hier die Ordnung 5 gewählt.

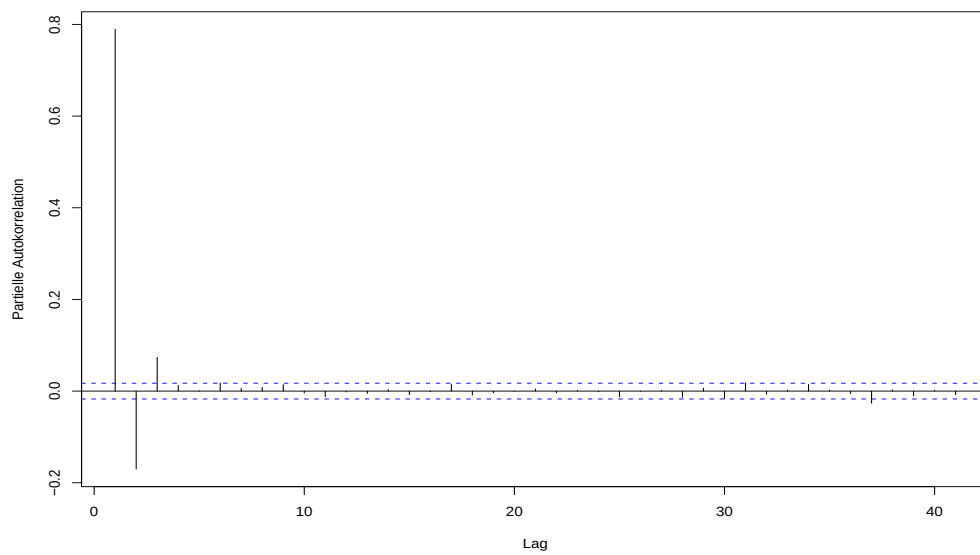
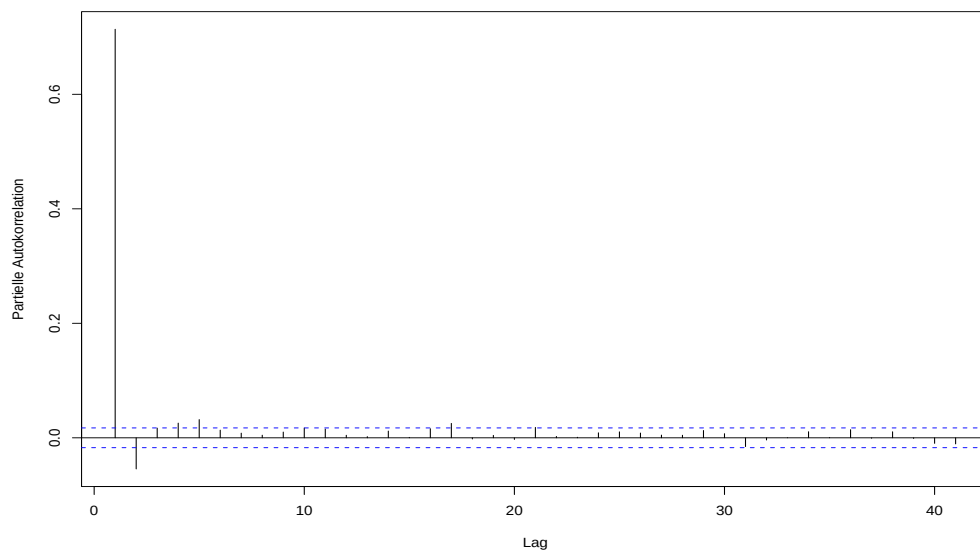
Bei allen Stationen geht aus der PACF hervor, dass die Abhängigkeit zum letzten Tag mit Abstand am grössten ist.

Die Bestimmung der AR -Ordnung mit der PACF hat den Nachteil, dass es zum Teil schwierig zu beurteilen ist, ab welchem k die wahre PACF den Wert null annimmt. Die Bestimmung der AR -Ordnung mit Hilfe des AIC besitzt diesen Nachteil nicht und bietet eine automatische Modellselektion:

6.4.2 AR-Ordnungswahl mittels AIC

Die Anpassung eines Modells von sehr hoher Ordnung ergibt Residuen mit einer kleinen Varianz. Wenn wir das angepasste Modell für Vorhersagen verwenden wollen, hängt der mittlere quadratische Vorhersagefehler nicht nur von der Varianz der Residuen ab, sondern auch von den Fehlern, die durch die Schätzung der Modellparameter entstehen. Diese Fehler werden grösser für Modelle höherer Ordnung.

Dem **Akaike's Information Criterion (AIC)** liegt nun die Idee zu Grunde, Modelle mit zu vielen Parametern zu bestrafen.

Abbildung 6.2: PACF der Residuen x_t von ZürichAbbildung 6.3: PACF der Residuen x_t von Vancouver

Seien $x_1, \dots, x_n$ (kurz $\mathbf{x}$) wieder die Temperatur-Residuen, die eine Realisation des stationären Prozesses $(X_t)_{t \in \mathbb{Z}}$ mit Wahrscheinlichkeitsverteilung P sind. Aufgrund von $\mathbf{x}$ sei $\hat{\phi}$ die Schätzung der Parameter $\phi_1, \dots, \phi_p$ in einem $\text{AR}(p)$ Modell. Aus den Daten $\mathbf{x}$ und der Annahme eines $\text{AR}(p)$ Modells mit Parametern $\hat{\phi}$ können wir die Wahrscheinlichkeitsdichte $dP_{\hat{\phi}}$ des Prozesses schätzen. Logarithmiert ergibt sich die Log-Likelihood Funktion, die an der Stelle $\hat{\phi}$ den Wert $\ell(\hat{\phi}, \mathbf{x})$ annimmt. Beim AIC suchen wir die Ordnung p , bei der die Grösse

$$AIC(p) = -2 \cdot \ell(\hat{\phi}, \mathbf{x}) + 2 \cdot p$$

minimal wird.

Das AIC setzt sich aus zwei Termen zusammen:

1. $-2 \cdot \ell(\hat{\phi}, \mathbf{x})$ hat die Rolle eines “*Goodness of fit*”-Masses
2. $2 \cdot p$ ist ein *Bestrafungsterm* für grosse Modelle.

Wir werden kurz den theoretische Hintergrund dieses Kriteriums erläutern:

Die **Kullback-Leibler-Information** ist eine Pseudo-Distanz zwischen zwei Verteilungen P und Q und ist wie folgt definiert:

$$I_n(P, Q) = \int_{R^n} \log\left(\frac{dP(\mathbf{y})}{dQ(\mathbf{y})}\right) dP(\mathbf{y})$$

Sie hat die Eigenschaft, dass $I_n(P, Q) \geq 0$, wobei Gleichheit genau dann gilt, wenn $dP = dQ$ erfüllt ist (vgl. [3]).

Sei nun P die Wahrscheinlichkeitsverteilung des wahren Modells und $P_{\sim}$ die Wahrscheinlichkeitsverteilung des $AR(p)$ Modells mit geschätztem Parameter $\hat{\phi}$. Die optimale AR-Ordnung ergibt eine minimale Pseudo-Distanz $I_n(P, P_{\sim})$.

Das AIC stellt eine Schätzung von $I_n(P, P_{\sim})$ dar, und deshalb entspricht die Minimierung des AIC einer Minimierung von $I_n(P, P_{\sim})$.

Wir betrachten nun das AIC von verschiedenen Stationen. Dazu bilden wir die AIC-Differenz

$$AIC(p) - \min_p AIC(p),$$

um die Abweichung einer Ordnung p von der optimalen Ordnung besser untersuchen zu können.

In Abbildung 6.4 ist die AIC-Differenz am Beispiel von Zürich aufgetragen. Hier wählt das AIC die optimale Ordnung 6.

Es gibt Fälle, wo eine tiefere Ordnung als die des minimalen AIC ein beinahe so kleines AIC besitzt. Ein solches Beispiel ist das AIC von Vancouver (Abbildung 6.5). Hier wird die hohe Ordnung 17 gewählt, obwohl wesentlich tiefere Ordnungen auch schon ein ähnlich kleines AIC aufweisen. Um in so einem Fall nicht eine unnötig hohe Ordnung zu erhalten, führen wir eine Schranke ein (blaue Linie in Abbildung 6.5). Die kleinste Ordnung, bei der die AIC-Differenz diese Schranke unterschreitet, wählen wir als geschätzte AR-Ordnung $\hat{p}$. Im Beispiel von Vancouver kann dadurch die AR-Ordnung von 17 auf 5 reduziert werden. Wir stellen auch fest, dass das AIC mit Schranke und die PACF bei diesen Beispielen hinsichtlich der geschätzten Ordnung übereinstimmen. Die Schranke lässt sich auch damit rechtfertigen, dass das AIC bekanntermassen zu hohe Ordnungen wählt, falls die wahre Verteilung P nicht in der Klasse $P_{\sim}$ enthalten ist.

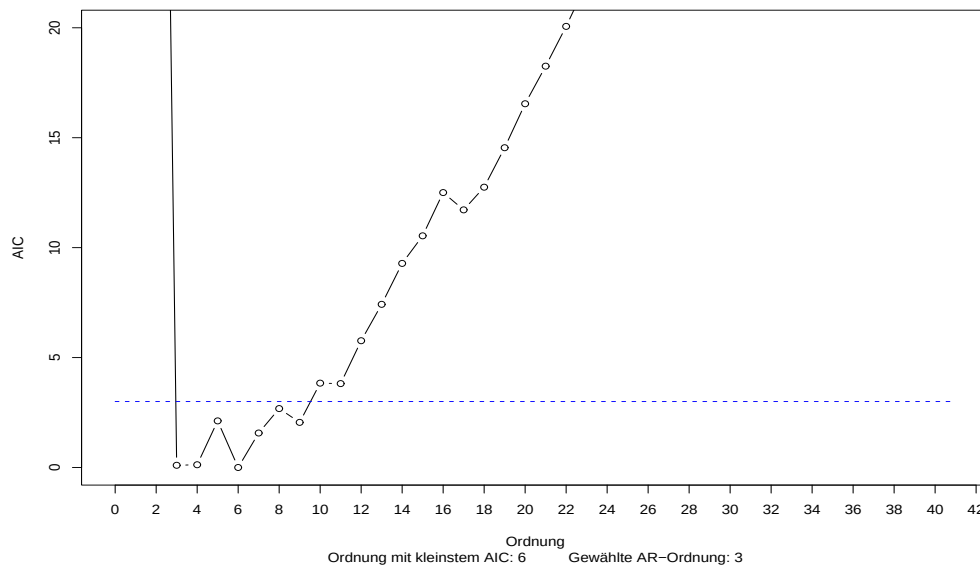


Abbildung 6.4: AIC-Differenz der Residuen von Zürich

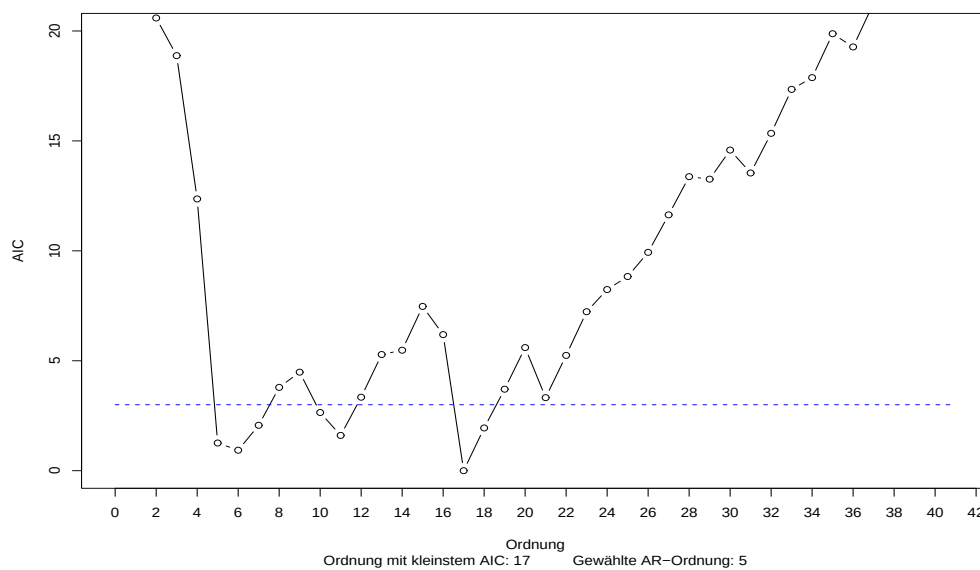


Abbildung 6.5: AIC-Differenz der Residuen von Vancouver

6.4.3 Güte des angepassten AR-Modells

An die Temperatur-Residuen x_t haben wir ein $\text{AR}(p)$ Modell mit Parametern $\hat{\phi}_1 \dots \hat{\phi}_p$ angepasst. Wir wollen nun die AR-Residuen

$$\hat{\varepsilon}_t = x_t - \hat{\phi}_1 \cdot x_{t-1} - \dots - \hat{\phi}_p \cdot x_{t-p}$$

genauer untersuchen.

Falls das angepasste AR-Modell zutrifft, sollten die $\hat{\varepsilon}_t$ unkorreliert sein.

Um dies zu überprüfen, ist in Abbildung 6.6 die Autokorrelationsfunktion der AR-Residuen $\hat{\varepsilon}_t$ aufgetragen (Datensatz Zürich). Wir stellen fest, dass die Residuen $\hat{\varepsilon}_t$ von Zürich nun

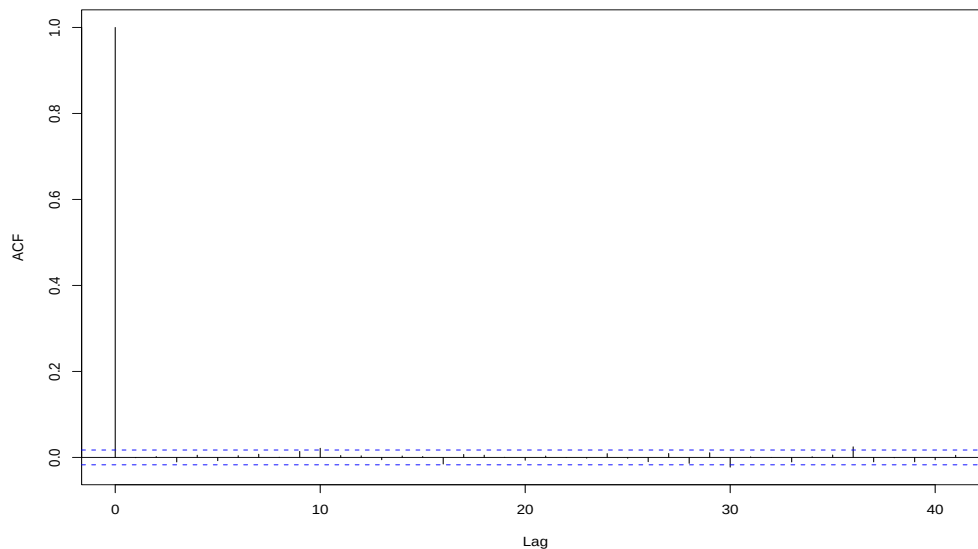


Abbildung 6.6: ACF der AR-Residuen $\hat{\varepsilon}_t$

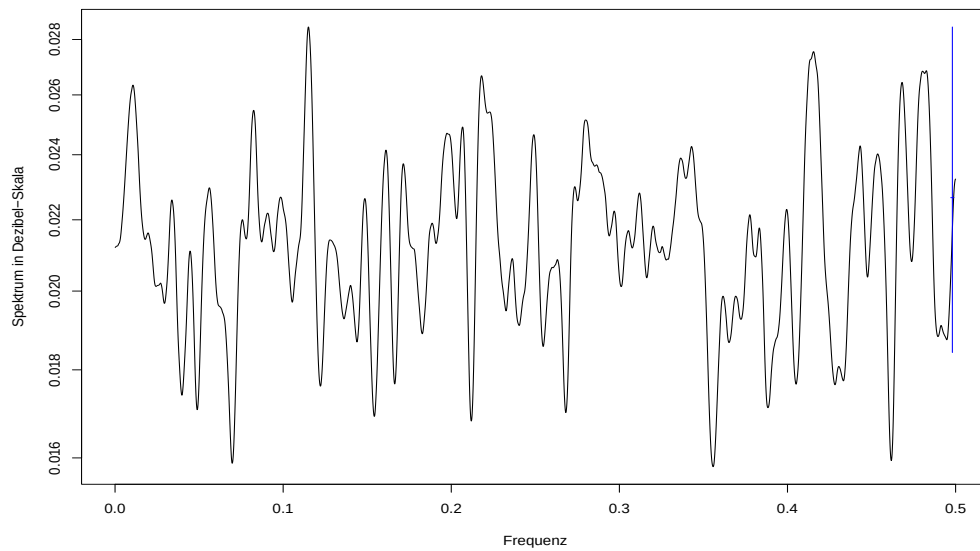
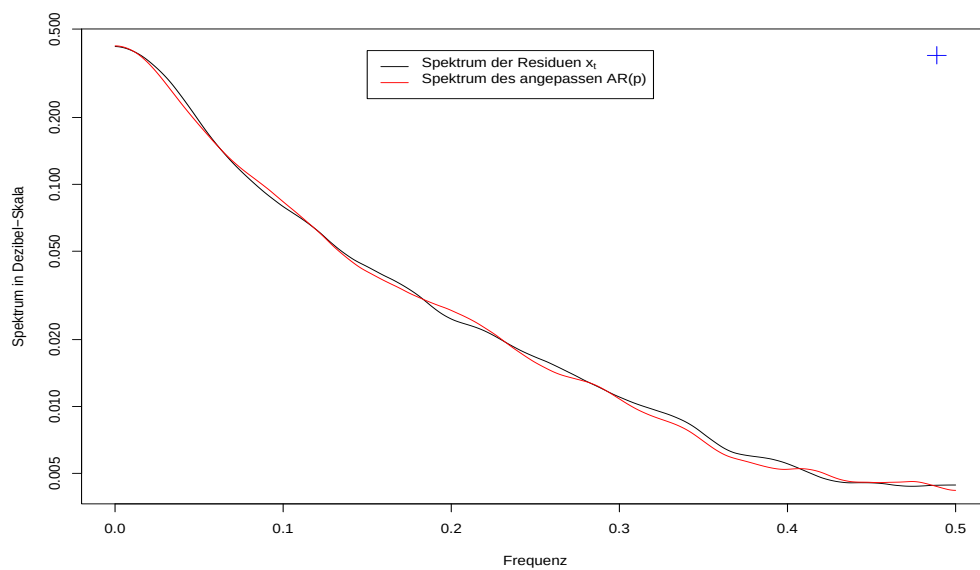
für alle Lags unkorreliert sind. Das gleiche Bild zeigt sich auch bei allen anderen Orten. Das AR-Modell scheint also gut zu unseren Daten zu passen.

Die Abbildung 6.7 zeigt das geglättete Spektrum der Residuen $\hat{\varepsilon}_t$ von Zürich. Auch hier ist keine Periodizität der Residuen mehr feststellbar. Diese trifft auch für alle anderen Stationen zu. Eine weitere Möglichkeit zur Überprüfung des AR-Modells bietet der Vergleich des (geglätteten) Spektrums des angepassten AR-Modells mit dem der Residuen x_t .

Zur Simulation des Prozesses wurden normalverteilte Zufallsvariablen ε_t mit der Varianz σ_ε^2 verwendet.

Die Abbildung 6.8 vergleicht diese Spektren am Beispiel Zürichs. Es zeigt sich eine schöne Übereinstimmung der beiden Spektren. Die Modellierung der Residuen x_t mit einem AR-Modell erfasst also den Frequenzbereich sehr gut.

Zusammenfassend lässt sich sagen, dass der AR-Prozess ein gutes Modell zur Modellierung der Residuen x_t darstellt das alle wesentlichen Informationen erfasst.

Abbildung 6.7: Geglättetes Spektrum der AR-Residuen $\hat{\varepsilon}_t$ Abbildung 6.8: Vergleich des Spektrums von x_t mit dem des angepassten $AR(p)$ (Zürich)

Kapitel 7

Vorhersage der Temperaturzeitreihe

Beim Abschluss eines Wetterderivats werden Informationen über den zukünftigen Temperaturverlauf benötigt. Der Vorhersagezeitraum liegt je nach Vertrag zwischen einem halben und mehreren Jahren.

Unser Temperaturmodell wurde bewusst so entwickelt, dass Vorhersagen möglich sind. Dabei werden die in den Kapiteln 3-6 behandelten Schritte in rückwärtiger Reihenfolge durchgeführt. Dabei muss bei jedem Schritt eine Vorhersage über die Laufzeit des Wetterderivats erstellt werden. Wir beginnen mit der Vorhersage des AR-Prozesses.

7.1 Vorhersage des AR-Prozesses

Wir haben in Kapitel 6 an die Residuen $x_1 \dots x_n$ ein AR-Modell angepasst. Es gilt nun eine Vorhersage dieses Prozesses zu bestimmen und für jeden Zeitpunkt $n + m$ in der Vertragsperiode den Wert X_{n+m} zu prognostizieren. Dazu wird die *beste Vorhersage* von X_{n+m} gestützt auf die Daten $x_1, \dots, x_n$ berechnet. Diese ist nach Satz 6.3 durch

$$\hat{X}_{n+m|1:n} = \mathbf{E}[X_{n+m}|x_1, \dots, x_n]$$

gegeben.

Für einen $AR(p)$ Prozess lässt sich diese Vorhersage mit der folgenden Rekursionsformel berechnen (wobei wir $n > p$ annehmen):

Satz 7.1. *Für einen $AR(p)$ Prozess gilt*

$$\hat{X}_{n+m|1:n} = \begin{cases} \sum_{i=1}^p \phi_i \cdot X_{n+1-i} & \text{falls } m = 1 \\ \sum_{i=1}^{m-1} \phi_i \cdot \hat{X}_{n+m-i|1:n} + \sum_{i=m}^p \phi_i \cdot X_{n+m-i} & \text{falls } m > 1 \end{cases}$$

Beweis:

Wir betrachten zunächst den Fall $m = 1$:

$$\hat{X}_{n+1|1:n} = \mathbf{E}[X_{n+1}|x_1, \dots, x_n]$$

$$\begin{aligned}
&= \mathbf{E}\left[\sum_{i=1}^p \phi_i \cdot X_{n+1-i} + \varepsilon_{n+1} \mid x_1, \dots, x_n\right] \\
&= \sum_{i=1}^p \phi_i \cdot X_{n+1-i}
\end{aligned}$$

Der Fall $m > 1$ lässt sich wie folgt beweisen:

$$\begin{aligned}
\hat{X}_{n+m|1:n} &= \mathbf{E}[X_{n+m} \mid x_1, \dots, x_n] \\
&= \mathbf{E}\left[\sum_{i=1}^p \phi_i \cdot X_{n+m-i} + \varepsilon_{n+m} \mid x_1, \dots, x_n\right] \\
&= \sum_{i=1}^{m-1} \phi_i \cdot \underbrace{\mathbf{E}[X_{n+m-i} \mid x_1, \dots, x_n]}_{\hat{X}_{n+m-i|1:n}} + \sum_{i=m}^p \phi_i \cdot X_{n+m-i}
\end{aligned}$$

□

Wir können nun die beste Vorhersage des AR-Prozesses bestimmen. Zusätzlich ist ein Prognoseintervall zum Niveau $1 - \beta$ unter der Annahme von $\mathcal{N}(0, \sigma_\varepsilon^2)$ -verteilten ε_t durch

$$V_{n+m} = \hat{X}_{n+m|1:n} \pm \Phi^{-1}\left(1 - \frac{\beta}{2}\right) \cdot \sigma_{n+m|-\infty:n} \quad (7.1)$$

gegeben. Mit Φ bezeichnen wir die kumulative Verteilungsfunktion der Standard-Normalverteilung und mit $\Phi^{-1}(1 - \frac{\beta}{2})$ das $(1 - \frac{\beta}{2})$ -Quantil.

Mit $\sigma_{n+m|-\infty:n}$ wird die Standardabweichung von X_{n+m} gegeben der Daten bis zum Zeitpunkt n bezeichnet. Diese lässt sich wie folgt berechnen:

Satz 7.2. *Es gilt $\sigma_{n+m|-\infty:n}^2 = \sigma_\varepsilon^2 \cdot \sum_{i=0}^{m-1} \psi_i^2$. Dabei sind die ψ_i durch $\sum_{i=0}^{\infty} \psi_i \cdot z^i = \frac{1}{1 - \sum_{i=1}^p \phi_i \cdot z^i}$ gegeben.*

Beweis: Ein kausaler AR-Prozess lässt sich durch

$$X_{n+m} = \sum_{i=0}^{\infty} \psi_i \cdot \varepsilon_{n+m-i} \quad (7.2)$$

darstellen (Beweis in [2]). Die beste Vorhersage lautet dann in dieser Form

$$\hat{X}_{n+m} = \sum_{i=m}^{\infty} \psi_i \cdot \varepsilon_{n+m-i}. \quad (7.3)$$

Nun gilt

$$\begin{aligned}
\sigma_{n+m|-\infty:n}^2 &= \text{Var}(X_{n+m} \mid X_n, X_{n-1}, \dots) \\
&= \mathbf{E}[(X_{n+m} - \hat{X}_{n+m|-\infty:n})^2 \mid X_n, X_{n-1}, \dots] \\
&\stackrel{(7.2), (7.3)}{=} \mathbf{E}\left[\left(\sum_{i=0}^{m-1} \psi_i \cdot \varepsilon_{n+m-i}\right)^2\right] \\
&\stackrel{\varepsilon_t \text{ iid}}{=} \sigma_\varepsilon^2 \cdot \sum_{i=0}^{m-1} \psi_i^2
\end{aligned}$$

□

Wir werden nun die beste Vorhersage bei einigen Orten berechnen. Dazu berücksichtigen wir die Daten von x_t bis zum 31. August 1999. Für den Zeitraum vom 1. September 1999 bis zu 31. März 2000 soll eine Vorhersage erstellt werden.

In der Abbildung 7.1 ist die beste Vorhersage für die Standorte Zürich, New York und San Antonio rot eingezeichnet. Wir stellen fest, dass die Vorhersage sehr rasch gegen den Mittelwert 0 des Prozesses konvergiert. Für die Prognose über einen langfristigen Zeitraum ist also die Information von $x_1, \dots, x_n$ nicht von einem grossem Nutzen.

Die grünen Linien bilden das 95 % Prognoseintervall unter der Annahme von normalverteilten ε_t . Dieses Intervall konvergiert ebenfalls ziemlich schnell gegen einen konstanten Wert.

Um die Güte dieses Vertrauensintervalls zu testen, wurden die wahren Residuen x_t von der Vorhersageperiode in die Abbildung 7.1 eingezeichnet (schwarze Kurve).

Das Vertrauensintervall scheint recht gut zu diesen Residuen zu passen.

Dieses Vertrauensintervall kann ebenso wie die beste Vorhersage zurücktransformiert werden. Dann erhalten wir ein Vertrauensintervall für die Temperaturzeitreihe. Dieses ist zwar nicht für die kumulierten Degree Days von Nutzen. Jedoch gibt es spezielle Wetterderivatsverträge, die jedesmal eine Auszahlung ergeben, wenn die Temperaturen ein bestimmtes Band verlassen.

7.2 Rücktransformation des Residuenprozesses

7.2.1 Box-Cox Rücktransformation

Auch die Box-Cox Transformation muss zur Temperaturvorhersage rückwärts durchgeführt werden. Dazu addieren wir zu den vorhergesagten Residuen $\hat{x}_t$ wieder den Mittelwert $\hat{\mu}_{(b)}$. Dieser wurde in Kapitel 5 abgezogen, damit ein stationärer Prozess mit Erwartungswert 0 angepasst werden konnte. Zudem müssen wir berücksichtigen, dass wir die Werte $-I_t + \alpha$ Box-Cox transformiert hatten. Somit erhalten wir die Residuen $\hat{I}_t$ durch

$$\hat{I}_t = \begin{cases} \alpha - (\hat{\lambda} \cdot (\hat{x}_t + \hat{\mu}_{(b)}) + 1)^{\frac{1}{\hat{\lambda}}} & \text{falls } \hat{\lambda} \neq 0 \\ \alpha - e^{\hat{x}_t + \hat{\mu}_{(b)}} & \text{falls } \hat{\lambda} = 0 \end{cases}$$

aus den vorhergesagten AR-Residuen $\hat{x}_t$ ($t = n + 1, \dots, n + l$).

Für diese neuen Residuen $\hat{I}_t$ müssen auch die im obigen Abschnitt behandelten Vertrauensintervalle transformiert werden.

7.2.2 Modellierung der Heteroskedastizität der Residuen

Zur Nachbildung des Temperaturprozesses müssen wir die saisonale Abhängigkeit der Residuen modellieren. Dies bewerkstelligen wir, indem wir die Residuen $\hat{I}_t$ mit der im Kapitel 4 geschätzten saisonalen Standardabweichung $\hat{\sigma}_t$ multiplizieren, d.h.

$$\hat{R}_t = \hat{\sigma}_t \cdot \hat{I}_t.$$

Wegen der 365-Tage Periodizität der saisonalen Standardabweichung dürfen zur Vorhersage die aus den früheren Daten geschätzten $\hat{\sigma}_t$ ($= \hat{\sigma}_{t-365}$) verwendet werden.

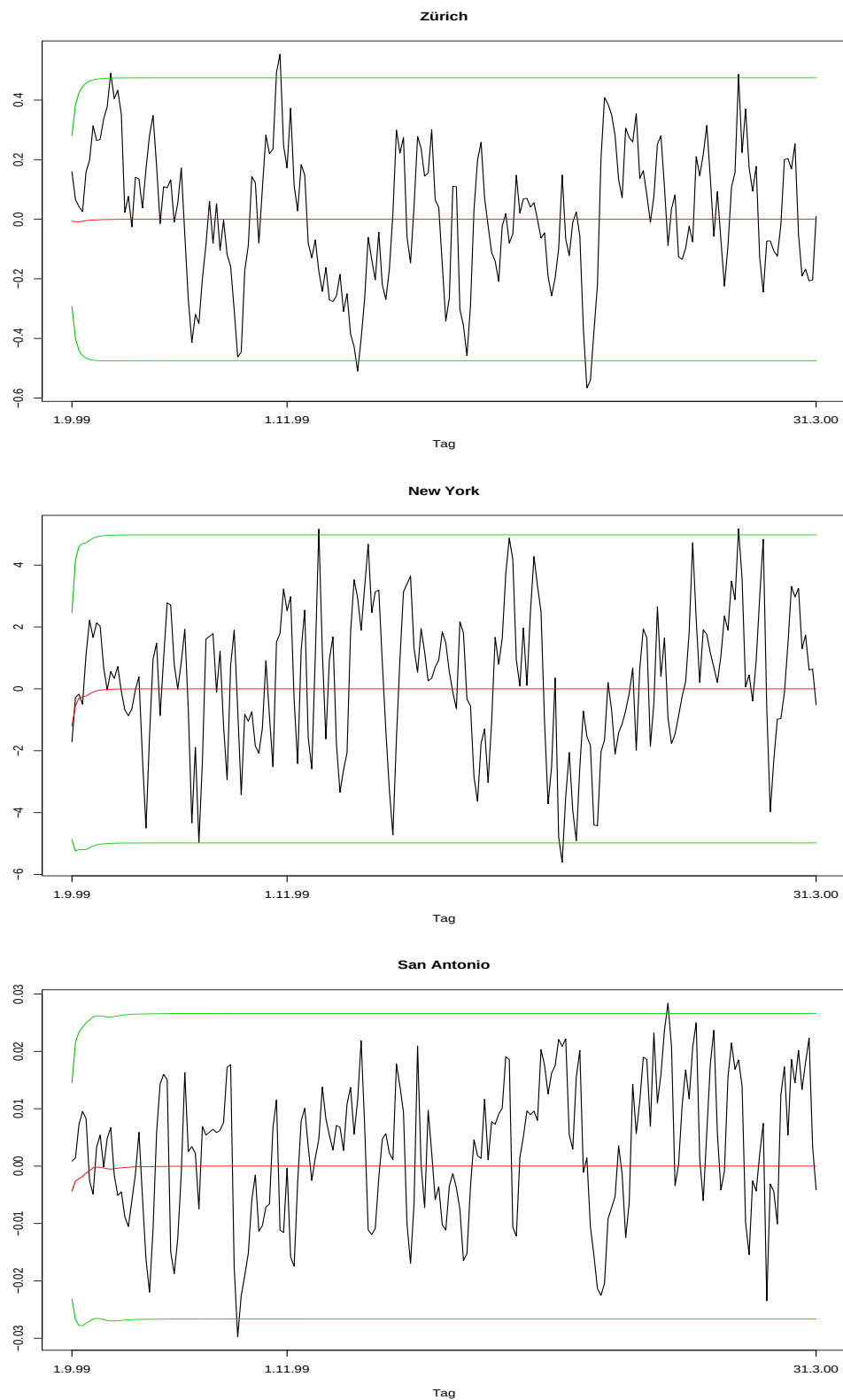


Abbildung 7.1: Beste Vorhersage (rot) und Prognoseintervall (grün) des $AR(p)$ -Modells. Zum Vergleich sind die wahren Residuen x_t (schwarz) der betrachteten Periode aufgetragen.

7.3 Vorhersage des Saisoneffekts

Bei einem periodischem Saisoneffekt $s_t = s_{t-365}$ kann wie schon bei der saisonalen Standardabweichung auf den aus früheren Jahren geschätzten Verlauf des Saisoneffekts zurückgegriffen werden.

Beim STL-Verfahren sind hingegen Änderungen des Saisoneffekts im Laufe der Zeit zugelassen. Beim Lowess zur Glättung der Tagesreihen haben wir den Grad $d_{(s)} = 0$ gewählt. Zur Vorhersage müsste wieder ein lokales gewichtetes Mittel aus den vergangenen Werten gebildet werden. Dabei wird von der k -nächsten Nachbarschaft Gebrauch gemacht.

Der Einfachheit halber verzichten wir auf diese Variante und verwenden zur Vorhersage das Mittel des geschätzten Saisoneffekts aus den letzten zwei bekannten Jahren.

Dies lässt sich dadurch rechtfertigen, dass bei der Lowess-Vorhersage mit $d_{(s)} = 0$ immer über die letzten k bekannten Werte gemittelt wird, und nur die Gewichtung sich ändert.

7.4 Vorhersage des Trends

Zur Vorhersage des Trends haben wir das modifizierte STL-Verfahren gewählt. Dort erfolgt die Trendschätzung mit einem Lowess. Für den Grad der lokalen Regression wurde $d_{(m)} = 1$ (linear) gewählt.

Bei der Vorhersage des Trends wird wieder die k -nächste Nachbarschaft verwendet. Diese Nachbarschaft wird im Fall der Vorhersage aus den k letzten bekannten Daten gebildet. Das Vorgehen bei der Berechnung der Trends an der Stelle $n+m$ bleibt dasselbe wie im Kapitel 3.1.4 (abgesehen von der zusätzlichen Gewichtung mit der saisonale Standardabweichung). Die Vorhersage berücksichtigt immer die gleichen k Datenpunkte, jedoch ändert sich die Nachbarschaftsgewichtung. Für langfristige Vorhersagen findet aber auch diese Änderung nur noch in einem bescheidenen Rahmen statt, da die biquadratische Gewichtsfunktion am Rand abflacht.

Im Gegensatz zum Saisoneffekt wird aber durch diese Daten eine Gerade angepasst. Somit verändert sich die Vorhersage für weiter entfernte Zeitpunkte sehr wohl noch. Jedoch wird die Vorhersage für weiter entfernte Zeitpunkte durch die sich angleichenden Gewichte immer linearer.

In Abbildung 7.2 ist der Trend von Zürich bis 31.9.1999 aufgetragen (schwarz). Ab den 1.9.99 wurde eine Trend-Vorhersage (rot) über zwei Jahre angestellt. Diese Vorhersage ist in Abbildung 7.3 noch etwas grösser eingezeichnet. Es wird hier ersichtlich, dass diese Trendvorhersage (zumindest zu Beginn) nicht linear ist.

7.5 Vorhersage der Temperatur

Wir haben nun die Schritte der Temperaturzerlegung rückgängig gemacht und jeweils eine Vorhersage erstellt. Für die beste Vorhersage der Temperatur erhalten wir

$$\hat{T} = \hat{m}_t + \hat{s}_t + \hat{R}_t \quad t = n+1, \dots, n+l.$$

Bei dieser besten Vorhersage hat der Residuenteil $\hat{R}_t$ nur einen kurzfristigen Einfluss. Dies würde sich ändern, wenn statt der besten AR-Vorhersage der zukünftige AR-Prozess

simuliert wird.

Da der Einfluss von Trend und Saisoneffekt auf die Vorhersage gross ist, werden wir die zu wählenden Parameter dieser Vorhersagen optimieren.

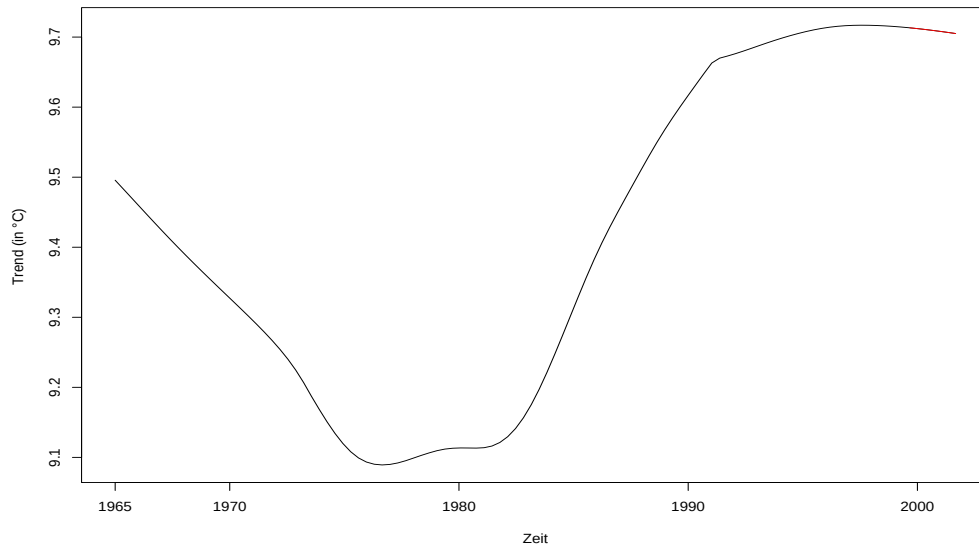


Abbildung 7.2: Trend in Zürich (schwarz) mit Vorhersage ab 1.9.1999 (rot)

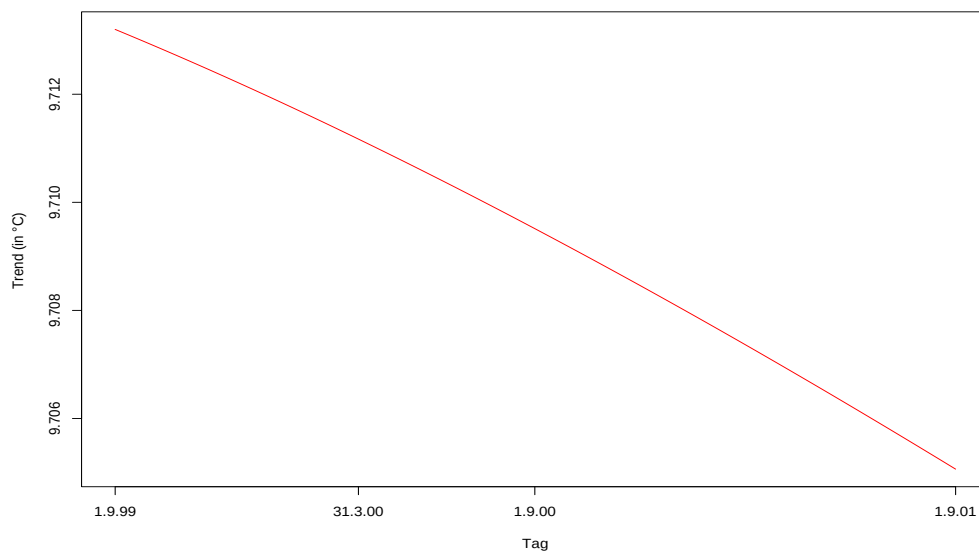


Abbildung 7.3: Vorhersage des Trends ab 1.9.1999 für zwei Jahre (Zürich)

7.6 Wahl der optimalen STL-Parameter

Beim STL-Verfahren müssen verschiedene Parameter gewählt werden. Im Detail sind dies

- der Glättungsparameter $f_{(m)}$ für den Trend,
- der Glättungsparameter $f_{(s)}$ für die Glättung der Tageszeitreihen und
- der Parameter $f_{(g)}$ für die Nachglättung des STL-Saisoneffekts.

Für die Wahl dieser Parameter gibt es keine festen Regeln. Je nach Anwendungsgebiet sind andere Werte sinnvoll.

In unserem Fall können meteorologische Hintergrundinformationen zur Wahl beitragen. So sollte der Saisoneffekt genügend glatt sein, um eine klimatologisch sinnvolle Grösse zu ergeben. Der Parameter $f_{(g)}$ darf folglich nicht zu klein gewählt werden.

Auch der Glättungsparameter $f_{(s)}$ darf nicht allzu klein gewählt werden. Sonst würde der Saisoneffekt zu genau den effektiven Temperaturen folgen. Dies widerspricht aber dem Konzept des Saisoneffekts, der über eine längere Frist gültig bleiben sollte.

Bei einer kleinen Wahl der Glättungsparameters $f_{(m)}$ folgt der Trend auch kurzfristigen Temperatur-Anomalien wie dem El Niño. Dies kann bei Prognosen über einen kurzfristigen Zeitraum durchaus Sinn machen. Jedoch könnten diese Anomalien bei einem glatteren Trend auch auf andere Weise (z.B. Anpassung des El Niño Index an die Temperaturdaten) berücksichtigt werden.

Bei langfristigen Prognosen (mehrere Jahre) kann aber das Folgen eines kurzfristigen Trends zu groben Fehlvorhersagen führen. Deshalb ist in diesem Fall eher die Wahl eines grösser Parameters $f_{(m)}$ angebracht.

Um die Parameterwahl zu optimieren, untersuchen wir, welche Parameter sich in der Vergangenheit am besten bewährt hätten. Da die optimale Wahl der Parameter von der Prognosedauer abhängt, müssen die Parameter für jede Dauer einzeln bestimmt werden. Bei unserer Untersuchung interessieren wir uns für die HDD-Periode vom 1. November bis zum 31. März. Da ein Wetterderivat typischerweise schon vor Beginn der Laufzeit abgeschlossen wird, muss die Tarifierung zum Zeitpunkt des Vertragsabschlusses erfolgen. Deshalb startet unsere Prognose schon am 1. September.

Wir passen nun unser Temperaturzeitreihen-Modell an die Daten bis zum 31. August 1998 an (Trainingsdatensatz). Aus diesem Modell berechnen wir wie im vorangehenden Abschnitt erläutert eine Vorhersage für die Periode vom 1. September 1998 bis zum 31. März 1999.

Da in dieser Periode auch die wahren Temperaturen (Testdaten) bekannt sind, können wir die mit unserer Temperaturvorhersage vergleichen.

Mit θ_{1998} bezeichnen wir die wahren kumulierten HDD der Laufzeit vom 1. November 1998 bis zum 31. März 1999 des Wetterderivats. $\hat{\theta}_{1998}$ bezeichnet die vorhergesagten kumulierten HDD derselben Periode. Dies ist die für die Auszahlung des Wetterderivats entscheidende Grösse. Als Gütemass für unsere Vorhersage nehmen wir nun die absolute Abweichung

$$|\hat{\theta}_{1998} - \theta_{1998}|.$$

des vorhergesagten zum wahren Wert.

Um zu prüfen, wie das Modell im langjährigen Vergleich abschneidet, wird dieser Vergleich analog für die Jahre 1989 bis 1997 durchgeführt.

Als Gütemass dafür verwenden wir

$$\theta = \sum_{j=1989}^{1998} |\hat{\theta}_j - \theta_j|.$$

Wenn bei der Vorhersage das STL-Verfahren mit den Parametern $f_{(m)}$, $f_{(s)}$ und $f_{(g)}$ verwendet wird, bezeichnen wir das Gütemass mit $\Delta_{f_{(m)}, f_{(s)}, f_{(g)}}$. Die optimalen STL-Parameter lassen sich durch die Minimierung von $\Delta_{f_{(m)}, f_{(s)}, f_{(g)}}$ nach $f_{(m)}$, $f_{(s)}$ und $f_{(g)}$ finden.

In der Abbildung 7.4 wird das Gütemass $\Delta_{f_{(m)}, f_{(s)}, f_{(g)}}$ in Zürich für verschiedene Parameterwahlen ausgerechnet. Dabei zeigt sich, dass die Wahl des Glättungsparameters f_g nur einen geringen Einfluss auf die Güte der Vorhersage besitzt.

Bei einigen Stationen (z.B. San Antonio) reduzieren grössere Werte von $f_{(g)}$ den Einfluss der Wahl von $f_{(s)}$ auf die Güte der Vorhersage. Die optimale Werte für $f_{(s)}$ bleibt aber derselbe. Auch auf den minimalen Wert des Gütemasses hat die Wahl von $f_{(g)}$ keinen Einfluss.

Da die Wahl des Parameters $f_{(g)}$ auf die Vorhersagegüte nur einen geringen Einfluss hat, treffen wir eine nach klimatologischen Kriterien (genügende Glattheit des Saisoneffekts) sinnvolle Wahl. Dies ist zum Beispiel beim Wert $f_{(g)} = 0.0025$ erfüllt.

Der optimale Wert des Parameters $f_{(s)}$ liegt in Zürich bei 0.8. Bei kleineren und grösseren Wahlen dieses Parameters ist eine Verschlechterung der Vorhersagegüte zu verzeichnen.

Der Trend-Glättungsparameter $f_{(m)}$ nimmt in Zürich einen optimalen Wert bei 1.4 an. Es ist jedoch zu bemerken, dass auch die Werte 0.8 bis 1.5 etwa dieselbe Vorhersagegüte besitzen. Ausserhalb dieses Bereichs verschlechtert sich die Vorhersage wieder.

In der folgenden Tabelle sehen wir die optimalen Parameter $f_{(m)}$ und $f_{(s)}$ für verschiedene Stationen:

	$f_{(m)}$	$f_{(s)}$
Zürich	1.4 (0.8-1.5)	0.8 (0.7-1.4)
New York	0.8 (0.6-1.6)	0.3 (0.4-0.8)
San Antonio	3.0 (1.4-2.9)	0.5 (0.3-0.7)

In den Klammern wurden jene Werte angegeben, die nahezu optimal sind.

Die optimalen Parameter unterscheiden sich je nach Station. Jedoch gibt es für jeden Parameter einen Bereich, der ebensogute Vorhersagen liefert. Deshalb stehen auch diese Parameter zur Wahl.

Die Wahl des Glättungsparameters $f_{(m)}$ im Bereich 0.8-1.5 liefert häufig gute Resultate. Für den Parameter $f_{(s)}$ kann keine generelle Empfehlung herausgegeben werden. Jedoch hat eine schlechte Wahl dieses Parameters weniger Einfluss auf die Vorhersage, als dies eine falsche Wahl von $f_{(m)}$ haben kann (vgl. Zürich mit $f_{(m)} = 0.2$).

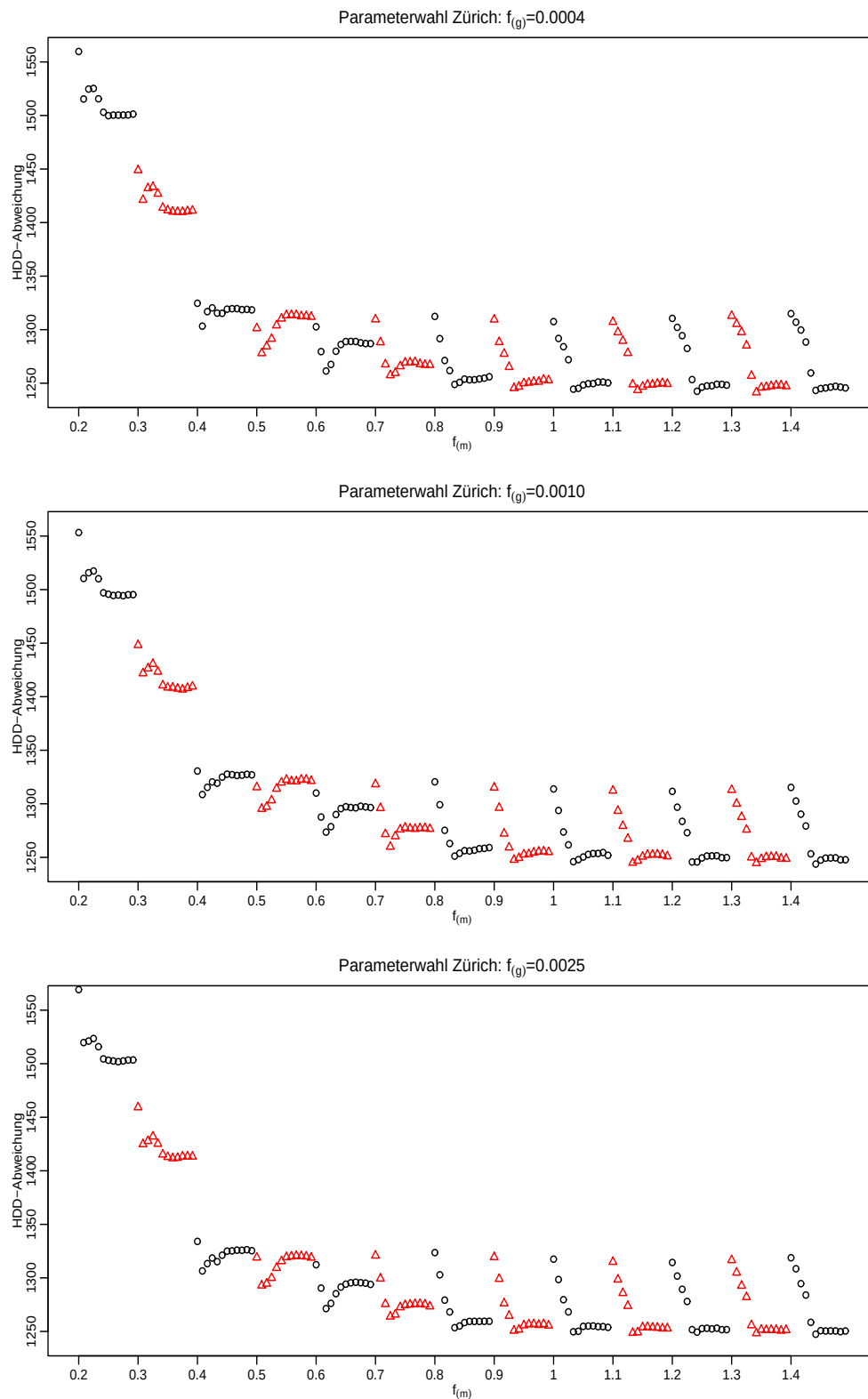


Abbildung 7.4: Parameteroptimierung in Zürich: Die Bilder zeigen die Werte von $\Delta_{f_{(m)}, f_{(s)}, f_{(g)}}$ bei drei verschiedenen Wahlen von $f_{(g)}$. In einem einzelnen Bild ($f_{(g)}$ fest) werden die Parameter $f_{(m)}$ und $f_{(s)}$ variiert. Bei einem festen Wert von $f_{(m)}$ wird das Gütemass für verschiedene Wahlen von $f_{(s)}$ (0.3-1.4) berechnet. Bei gewähltem $f_{(m)}$ werden die Auswertungen durch dieselbe Farbe gekennzeichnet.

Kapitel 8

Das Bootstrap-Verfahren

8.1 Der Bootstrap-Ansatz

Im vorangehenden Kapitel haben wir eine Methode hergeleitet, wie aus den bekannten Temperaturen $T_1, \dots, T_n$ eine Vorhersage $\hat{T}_{n+1}, \dots, \hat{T}_{n+l}$ berechnet werden kann. Aus diesen vorhergesagten Werten lassen sich die kumulierten Degree Days $\hat{\theta}$ schätzen, von denen die Auszahlung eines Wetterderivats abhängt.

Zur Modellierung des Temperaturprozesses werden

- der Trend m_t ,
- der Saisoneffekt s_t ,
- die saisonale Standardabweichung σ_t
- der optimale Box-Cox Parameter λ ,
- der Erwartungswert $\mu_{(b)}$ der Residuen nach der Box-Cox Transformation,
- die AR-Ordnung p und
- die AR-Parameter $\phi_1, \dots, \phi_p$

geschätzt. Die Schätzungen dieser Parameter hängen alle von den Zufallsvariablen $\varepsilon_1, \dots, \varepsilon_n$ ab, und unterliegen somit einer Verteilung. In die Schätzung der Grösse $\hat{\theta}$ fließt diese Unsicherheit auch mit ein. Jedoch ist es nicht möglich, die Verteilung der optimalen Schätzung analytisch zu ermitteln.

Stattdessen greifen wir auf das *Bootstrap-Verfahren* von Efron [9] zurück. Hier wird die Verteilung von $\hat{\theta}$ durch Simulation bestimmt.

Das genaue Vorgehen setzt sich aus den folgenden Punkten zusammen:

1. *Anpassung des Modells an die Temperaturzeitreihe T_t*

In einem ersten Schritt passen wir unser Modell an die Temperaturzeitreihe T_t an. Dadurch erhalten wir die Schätzungen $\hat{m}_t, \hat{s}_t, \hat{\sigma}_t$ ($t = 1, \dots, n$), $\hat{\lambda}, \hat{\mu}_{(b)}, \hat{p}$ und $\hat{\phi}_1, \dots, \hat{\phi}_{\hat{p}}$. Aus diesen Grössen ergeben sich die Residuen $\hat{\varepsilon}_{\hat{p}+1} \dots \hat{\varepsilon}_n$.

2. *Vorhersage der kumulierten Degree-Days $\hat{\theta}$*

Aus der Temperaturzeitreihe $T_1, \dots, T_n$ bilden wir wie in Kapitel 7 beschrieben ei-

ne Temperaturvorhersage $\hat{T}_{n+1}, \dots, \hat{T}_{n+l}$. Damit lassen sich die kumulierten Degree-Days durch

$$\hat{\theta} = \sum_{i=n+k}^{n+l} \max(\delta \cdot (\hat{T}_i - T_{Ref}), 0)$$

berechnen, wobei $\delta = 1$ für Cooling Degree-Days und $\delta = -1$ für Heating Degree-Days. Die Summation startet zum Zeitpunkt $n+k$ ($0 \leq k < l$), wo die Laufzeit des Wetterderivats beginnt. Die Temperaturvorhersage beginnt aber schon zum Zeitpunkt n des Vertragsabschlusses.

3. Simulation der Verteilung der AR-Residuen $\hat{\varepsilon}_t$

Die Verteilung der Residuen $\hat{\varepsilon}_t$ kann auf zwei Arten nachgebildet werden:

- Beim *parametrischen Bootstrap* wird eine Normalverteilung mit Erwartungswert 0 und Varianz σ_ε^2 angepasst. Dieser Ansatz lässt sich dadurch rechtfertigen, dass nach der Box-Cox Transformation auch die AR-Residuen genähert normalverteilt sein sollten. Mit dieser Normalverteilung generieren wir die Werte $\varepsilon_1^*, \dots, \varepsilon_n^*$.
- Beim *Bootstrap der Residuen* werden aus der Menge $\{\hat{\varepsilon}_{\hat{p}+1}, \dots, \hat{\varepsilon}_n\}$ n Stichproben mit Zurücklegen gezogen. Diese Stichprobe $\varepsilon_1^*, \dots, \varepsilon_n^*$ sollte nach dem Satz von Glivenko-Cantelli mit guter Näherung gemäss der wahren Verteilung verteilt sein. Der Bootstrap der Residuen eignet sich vor allem bei Orten, wo die AR-Residuen nur schlecht einer Normalverteilung gehorchen.

4. Simulation der Temperaturzeitreihe

Die im Schritt 3) generierten AR-Residuen $\varepsilon_1^*, \dots, \varepsilon_n^*$ können zur Simulation des Temperaturprozesses herangezogen werden. Dabei ist die Realisation des AR-Prozesses durch

$$x_t^* = \begin{cases} \hat{\phi}_1 \cdot x_{t-1}^* + \dots + \hat{\phi}_{\hat{p}} \cdot x_{t-\hat{p}}^* + \varepsilon_t^* & t = 1, \dots, n \\ 0 & t = 1 - \hat{p}, \dots, 0 \end{cases}$$

gegeben. Nach der Box-Cox Rücktransformation erhalten wir die Residuen I_t^* ($t=1, \dots, n$). Die Simulation des Temperaturprozesses ist durch

$$T_t^* = \hat{m}_t + \hat{s}_t + \hat{\sigma} \cdot I_t^* \quad t = 1, \dots, n$$

gegeben.

5. Anpassung des Modells an die simulierte Temperaturzeitreihe T_t^*

An die im Schritt 4) simulierte Temperaturzeitreihe T_t^* passen wir wieder unser Modell an. Die resultierenden Schätzungen bezeichnen wir mit m_t^* , s_t^* , σ_t^* ($t = 1, \dots, n$), λ^* , $\mu_{(b)}^*$, p^* und $\phi_1^*, \dots, \phi_{p^*}^*$.

6. Vorhersage der kumulierten Degree-Days θ^*

Aufgrund der Temperaturen $T_1^*, \dots, T_n^*$ bilden wir wie in Kapitel 7 erläutert eine Temperaturvorhersage $T_{n+1}^*, \dots, T_{n+l}^*$. Dazu werden die im Schritt 5) erhaltenen Schätzungen verwendet. Die Vorhersage der kumulierten Degree-Days ist durch

$$\theta^* = \sum_{i=n+k}^{n+l} \max(\delta \cdot (T_i^* - T_{Ref}), 0)$$

gegeben.

7. Wiederholte Erzeugung von Bootstrap-Replikaten θ^*

Die Schritte 3)-6) werden nun B -mal wiederholt und wir erhalten die Bootstrap-Replikate $\theta^{*(1)}, \dots, \theta^{*(B)}$. Diese können zur Analyse der Verteilung und der Varianz der geschätzten Grösse $\hat{\theta}$ dienen.

8.2 Vertrauensintervall für $\hat{\theta}$

Mit dem Bootstrap-Verfahren können wir nun beliebig viele Replikate von $\hat{\theta}$ simulieren. Eine Grösse von Interesse ist dabei die Varianz des Vorhersage-Schätzers. Diese wird durch

$$\frac{1}{B-1} \sum_{i=1}^B (\theta^{*(i)} - \bar{\theta}^*)^2$$

geschätzt. Dabei gilt $\bar{\theta}^* = \frac{1}{B} \sum_{i=1}^B \theta_i^*$.

Ebenfalls können wir ein 95%-Vertrauensintervall für die beste Vorhersage $\hat{\theta}$ bilden. Dieses wird durch jene Beobachtungen begrenzt, bei denen sich ausserhalb nur noch 2.5 % der Replikate $\theta^{*(i)}$ befinden.

Die Abbildung 8.1 zeigt das Histogramm der Replikate $\theta^{*(1)}, \dots, \theta^{*(1000)}$ bei verschiedenen Orten. Die Grösse $\hat{\theta}$ bezieht sich wieder auf die HDD-Periode vom 1. November 1999 bis zum 31. März 2000, wobei die Temperaturvorhersage am 1. September 1999 beginnt. Die STL-Parameter wurden gemäss Kapitel 7 gewählt. Da die AR-Residuen an diesen Orten gut einer Normalverteilung gehorchen, wurde ein parametrischer Bootstrap zur Simulation der Verteilung verwendet.

In der Abbildung 8.1 ist das 95%-Vertrauensintervall für $\hat{\theta}$ mit blauen Linien markiert. Die grüne Line in der Abbildung 8.1 kennzeichnet den Mittelwert $\bar{\theta}^*$ der Bootstrap-Replikate und der rote Kreis auf der x-Achse den Wert $\hat{\theta}$.

Unser Vorhersagemodell schätzt diesen Wert im Mittel relativ genau. Vermutlich würde der Mittelwert der Replikate bei mehr als 1000 Simulationen noch näher an den Wert $\hat{\theta}$ heranrücken.

Falls der geschätzte Bias $\bar{\theta}^* - \hat{\theta}$ grösser als in unserem Fall wäre, könnte sich eine Bias-Korrektur des Vertrauensintervalls als sinnvoll erweisen.

Die folgende Tabelle zeigt die Varianz-Schätzung und den Bias der Vorhersage $\hat{\theta}$:

	Varianz	Bias
Zürich	1211.8	3.8
New York	3896.1	20.2
San Antonio	7980.4	19.9

Es ist zu beachten, dass in Zürich die Temperatur in Grad Celsius gemessen wird.

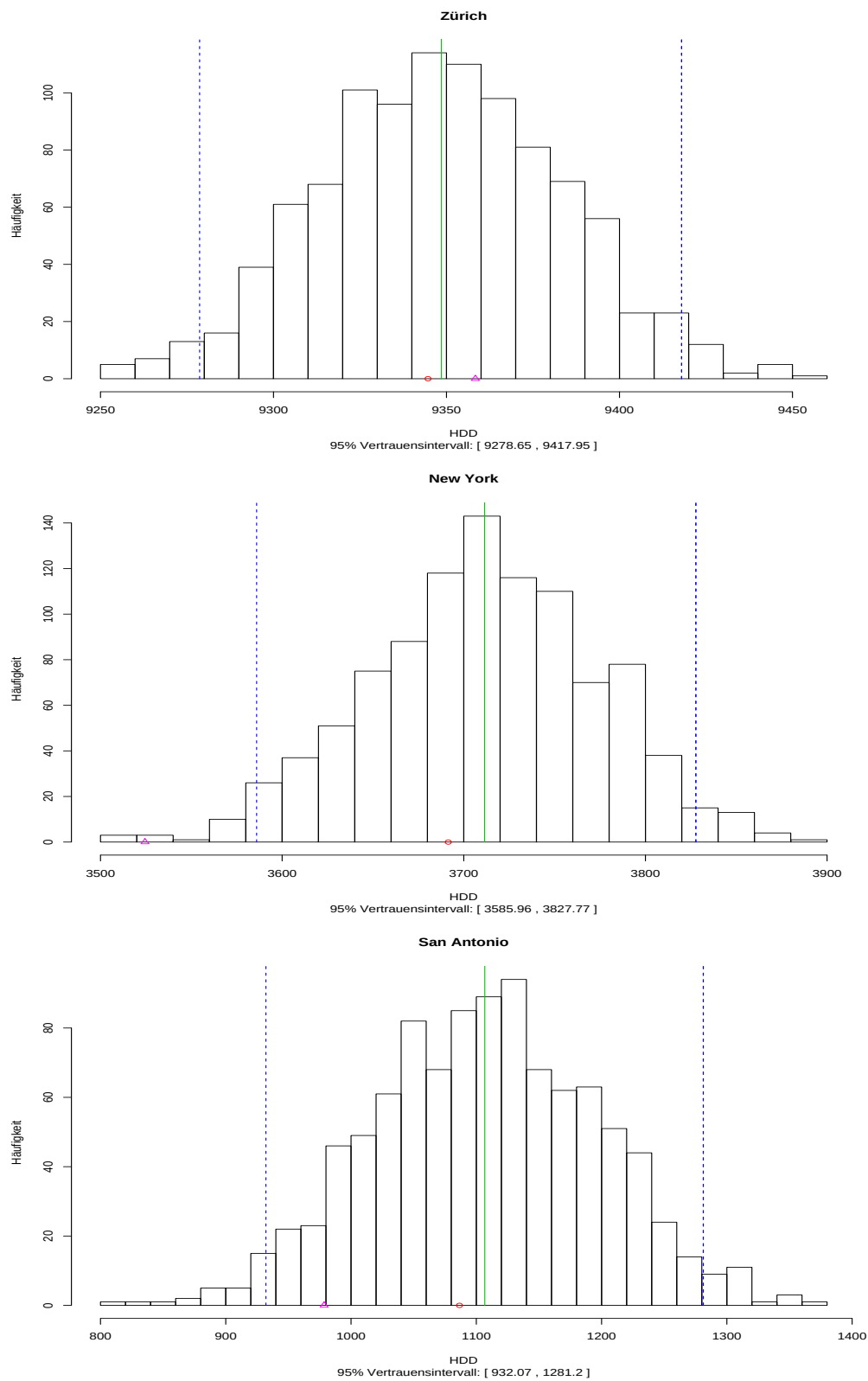


Abbildung 8.1: Histogramm der Bootstrap-Replikate $\hat{\theta}^{*(i)}$. Blau ist das 95% Perzentilintervall für $\hat{\theta}$ eingezeichnet. Der rote Kreis auf der x-Achse kennzeichnet $\hat{\theta}$ und das rosafarbene Dreieck den wahren Wert von θ in der betrachteten Periode.

8.3 Vorhersageintervall für θ

Wir haben nun ein Vertrauensintervall für die beste Vorhersage $\hat{\theta}$ ermittelt. Um ein Vorhersageintervall für den wahren Wert θ zu erhalten (der selber stochastisch ist), muss der stochastische Prozess X_t im Vorheragezeitraum $t = n + 1, \dots, n + l$ simuliert werden. Bis anhin haben wir nur die beste Vorhersage dieses Prozesses verwendet.

Das Vorgehen bleibt im wesentlichen das selbe wie bei der Bestimmung des Bootstrap-Vertrauensintervalls für $\hat{\theta}$. Jedoch muss in Schritt 6 die beste Vorhersage des AR-Prozesses X_t durch die stochastische Simulation des AR-Prozesses X_t ersetzt werden. Dabei kann ein parametrischer Bootstrap oder ein Bootstrap der Residuen verwendet werden.

Der Vorteil dieses Vorhersageintervalls für θ (im Gegensatz zur direkten Simulation aufgrund der Zerlegung von T_t) bildet die Tatsache, dass neben der Unsicherheit des AR-Prozesses X_t auch die Modellunsicherheit berücksichtigt wird.

In der Abbildung 8.2 ist dieses Vertrauensintervall für Zürich und New York eingezeichnet. Erwartungsgemäss ist es grösser als das für $\hat{\theta}$.

Der wahre Wert von θ liegt bei beiden Stationen innerhalb des Vorhersageintervalls.

8.4 Anwendung auf die Tarifierung eines Wetterderivats

Unser Modell bietet nun die Möglichkeit, aus einer beliebigen Temperaturzeitreihe die zukünftigen Temperaturen zu simulieren. Dabei wird auch die Modellunsicherheit berücksichtigt, die aus den Parameterschätzungen entsteht.

Die Simulationswerte der zukünftigen Degree-Days können dazu verwendet werden, um ein Wetterderivat zu tarifieren. Dazu kann wie in Kapitel 2 eine Verteilung an die Simulationswerte angepasst werden.

Da nun beliebig viele Bootstrap-Replikate zur Verfügung stehen, kann die erwartete Auszahlung des Wetterderivates auch aufgrund der empirischen Verteilungsfunktion von θ berechnet werden. Dies hat den Vorteil, dass auf die Wahl einer parametrischen Verteilung verzichtet werden kann.

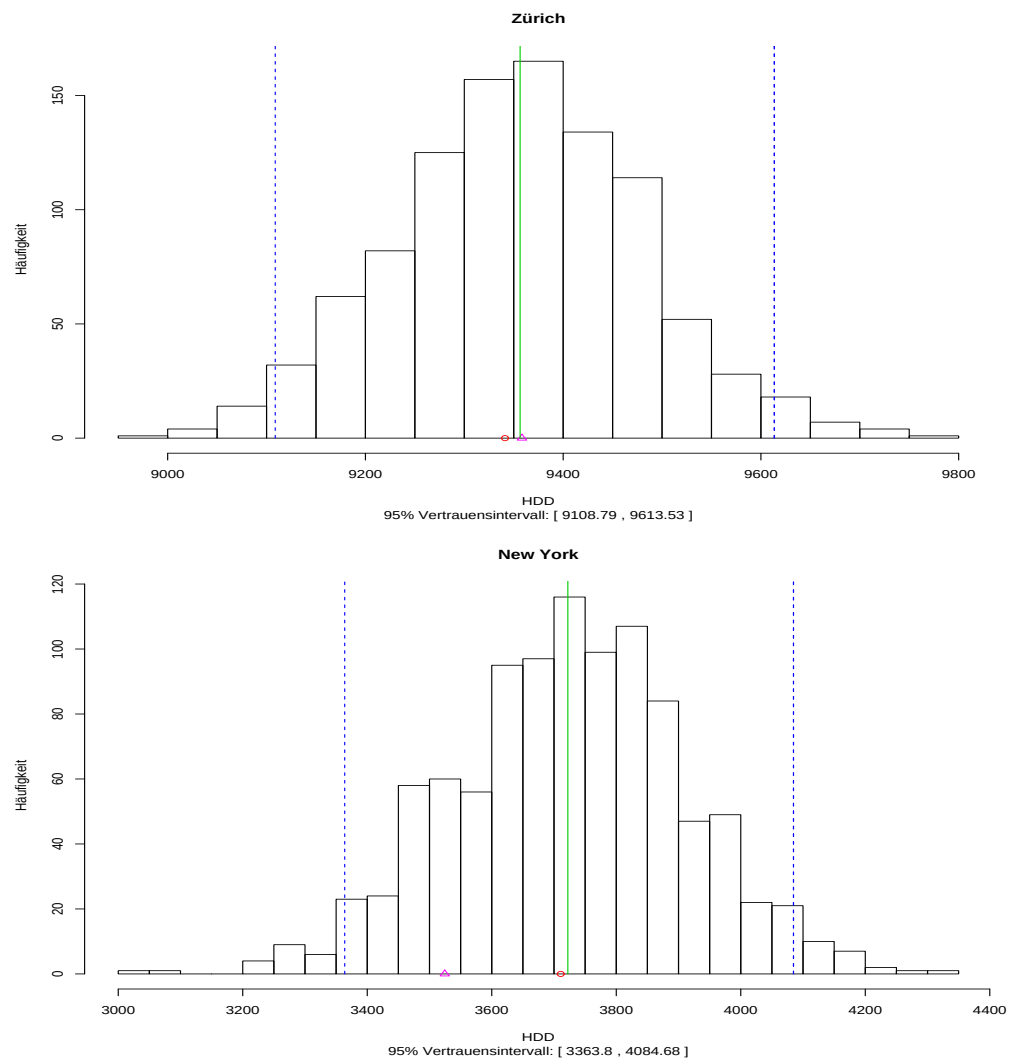


Abbildung 8.2: 95% Perzentilintervall für θ . Dabei wurde eine Simulation des Residuenprozesses im Vorhersagezeitraum verwendet.

Kapitel 9

Diskussion und Ausblick

In dieser Arbeit haben wir auf statistischer Basis ein Modell zur Vorhersage der Temperatur entwickelt. Dabei wurden die Temperaturen der Vergangenheit einer statistischen Analyse unterzogen, um die Struktur des Temperaturprozesses zu erfassen. Dazu wurde die Temperaturzeitreihe in eine stationäre Zeitreihe zerlegt.

Die genauen Schritte waren dabei die Folgenden:

1. Trend und Saisoneffekt wurden mit dem modifizierten STL-Verfahren entfernt. Dabei konnte die Heteroskedastizität der Residuen berücksichtigt werden. Dieses Verfahren wurde aufgrund seiner Robustheitseigenschaften und der Veränderbarkeit des Saison-effekts gewählt.
2. Die Heteroskedastizität der Residuen R_t wurde mit einem Lowess entfernt. Dabei ist es wichtig zu beachten, dass keine Robustheitsiterationen vorzunehmen sind.
3. Die Schiefe der normierten Residuen I_t wurde mit einer Box-Cox Transformation behoben. Dadurch konnte an vielen Orten eine Verbesserung der Residuen hinsichtlich einer Normalverteilung erzielt werden. Bei wenigen Orten (z.B. Vancouver) kann nicht von einer Normalverteilung ausgegangen werden.
4. An die Box-Cox transformierten Residuen x_t wurde ein AR-Modell angepasst. Zur Schätzung der Koeffizienten wurde das Yule-Walker Verfahren verwendet. Die Schätzung der AR-Ordnung erfolgte mit Hilfe des AIC, wobei eine Schranke eingeführt wurde. Dadurch konnte eine unnötig hohe AR-Ordnung vermieden werden und eine bessere Übereinstimmung mit dem PACF erzielt werden. Analysen haben ergeben, dass der angepasste AR-Prozess gut zu den Daten passt.

Bei der Modellierung wurde festgestellt, dass an den verschiedenen Standorten der Wetterstationen sehr unterschiedliche klimatologische Bedingungen herrschen. Dies wurde sowohl im Verhalten der Trends, als auch in der unterschiedlichen Wahl der Modell-Parameter deutlich. Deshalb musste ein für eine allgemeine Wetterstation gültiges Modell erstellt werden. Dies hat den Vorteil, dass sich das Modell neben der Modellierung von Wetterderivaten auch für weitere klimatologische Analysen einsetzen lässt. Da das STL-Verfahren einen veränderbaren Saisoneffekt zulässt, kann dieses Modell die Temperaturveränderung an verschiedenen Standorten untersuchen und vergleichen. Zudem kann durch die Verwendung eines nichtparametrischen Verfahrens auch die Temperaturzeitreihe weiter als 35 Jahre in die Vergangenheit untersucht werden.

Bei der Modellierung der Wetterderivate kann man sich mit unserer Methode durch Simulation ein Bild über die möglichen Verläufe der zukünftigen Temperatur verschaffen. Insbesondere können auch komplexere Wetterderivate als Degree-Day Caps und Degree-Day Floors modelliert werden. So gibt es z.B. Wetterderivate, die jedesmal eine Auszahlung ergeben, wenn der Temperaturprozess eine obere bzw. untere Schranke überschreitet.

Neben den statistischen Modellen können auch klimatologische Modelle zur Temperaturvorhersage beitragen. So ist zum Beispiel bei Anomalien wie dem El Niño mit diesen Modellen eine gewisse Vorhersage möglich. Solche Prognosen bietet die National Oceanic and Atmospheric Administration (NOAA) an. Über das Internet¹ können dabei Informationen über die prognostizierte Auswirkung der Anomalien auf die Temperatur abgerufen werden.

Eigene Analysen mit dem El Niño haben ergeben, dass der Einfluss des El Niño Index ENSO auf die Temperaturen Nordamerikas um etwa 3 Monate verzögert ist. Dazu wurde dieser Index mit Hilfe einer Regression an den STL-Trend angepasst. Der Glättungsparameter $f_{(m)}$ wurde dabei so klein gewählt, dass der Einfluss der Anomalien sichtbar wird. Bei der Anpassung wurde auch die dritte Potenz des ENSO-Index als abhängige Variable verwendet. Dadurch konnte die Anpassung (R^2) verbessert werden. Dies lässt sich dadurch begründen, dass besonders die extremen Werte dieses Index auf die Temperaturen einen Einfluss haben (El Niño und La Niña Ereignisse). Mit dieser Regression ergab sich der beste Fit (R^2), wenn die Temperatur etwa einen Lag von 3 Monaten zum ENSO Index hatte.

Dieser verzögerte Einfluss des ENSO-Index könnte dazu verwendet werden, um die Temperaturvorhersage zu verbessern. Jedoch ist die Vorhersage auf einen Zeitraum von 3 Monaten beschränkt. Für die meisten Wetterderivate wird aber ein längerer Zeitraum benötigt. Insbesondere findet der Vertragsabschluss häufig schon einige Monate vor Beginn der Laufzeit statt. Auch der Einfluss des ENSO auf die kurzfristigen Temperaturtrends war nicht an allen Standorten in Amerika gleich ausgeprägt. So ergab sich in San Antonio ein R^2 von etwa 20%, in anderen Orten wie Chicago war jedoch kaum ein Einfluss vorhanden.

Möglicherweise könnte die Vorhersage durch den Einbezug weiterer klimatologischer Variablen in die Regression verbessert werden.

Ein anderer Punkt, der noch untersucht werden könnte, ist die saisonale Abhängigkeit der AR-Koeffizienten und der Verteilung des Temperaturprozesses. So wäre es denkbar, für die CDD-Periode (Sommer) und HDD-Periode (Winter) ein separates AR-Modell anzupassen. Auch der Parameter λ der Box-Cox Transformation könnte separat für Sommer und Winter geschätzt werden.

Dies sind aber Effekte zweiter Ordnung, so dass kein allzugrosser Einfluss auf die Vorhersage zu erwarten ist.

Ein weiterer Punkt, der noch genauer untersucht werden könnte, ist die Ursache, warum bei der Parameterwahl in Kapitel 7 die HDD-Abweichung und die Quadratsumme der Residuen bei einigen Standorten nicht dieselbe Wahl für den Parameter $f_{(s)}$ liefert. Dies könnte damit zu tun haben, dass bei der HDD-Abweichung nur die vorhergesagten Tempe-

¹<http://www.noaa.gov>

raturen, die kleiner als die Referenztemperatur sind, einbezogen werden. Da die Wahl des Parameters $f_{(s)}$ nicht bei allen Stationen einen sehr grossen Einfluss auf die Vorhersagegüte hat, könnten die Temperaturen oberhalb der Referenztemperatur zu einer anderen Wahl des Parameters beitragen.

Anhang A

Eigene R Funktionen

A.1 Einlesen und Bereinigen der Daten

A.1.1 Schaltjahrbereinigung

```
\begin{small}
schaltjahr <- function(zeitreihe,aufteilen=FALSE){
  anz <- length(zeitreihe)
  anz.schalt <- floor((anz-1155)/1461)
  schalt <- c(1155,1155+1461*(1:anz.schalt)) # Schaltjahre nach 1965

  # Aufteilung des 29.2 auf 28.2 und 1.3
  if (aufteilen==TRUE){
    zeitreihe[schalt-1] <- (zeitreihe[schalt]+zeitreihe[schalt-1])/2
    zeitreihe[schalt+1] <- (zeitreihe[schalt]+zeitreihe[schalt+1])/2
  }

  res <- zeitreihe[-schalt]
}
```

A.1.2 Umrechnung von Fahrenheit nach Celsius

```
f2c <- function(zeitreihe){
  res <- 5/9*(zeitreihe-32)
}
```

A.1.3 Einlesen der Daten mit anschliessender Bereinigung

```
lesen <- function(stadt,schalt=TRUE, celsius=TRUE){

  # Dateinamen der Städte (Unterteil nach Europa, Kanada und USA

  europa <- c("belfast","bergen","berlin","bordeaux","cologne","gothenburg",
    "hamburg","helsinki","leeds","lille","london_heathrow","lyon","manchester",
    "marseille","oslo","paris","stockholm","strasbourg","stuttgart","toulouse",
```

```

"zurich")

kanada <- c("edmonton","ottawa","toronto","vancouver")

usa <- c("atlanta_ga","baltimore_md","birmingham_al","boise_id",
        "boston_ma","buffalo_ny","charlotte_nc","chicago_il_ohare","columbus_oh",
        "covington_ky","des_moinen_ia","fresno_ca","indianapolis_in","las_vegas_nv",
        "madison_wi","milwaukee_wi","mobile_al","new_orleans_la",
        "new_york_ny_la_guardia_ap","newark_nj","philadelphia_pa","phoenix_az",
        "pittsburgh_pa","portland_me","portland_or","raleigh_nc","san_antonio_tx",
        "san_francisco_ca","seattle_wa","tucson_az")

cities <- c(europa,kanada,usa)

# Datensatz einlesen (die ersten 3 Zeilen auslassen)
fil <- paste(Ddir, "/", cities[stadt], ".csv", sep="")
zeitreihe <- read.table(fil, header = TRUE, sep = ",", skip = 3)

# Schaltjahrbereinigung
if (schalt==TRUE){
  year <- schaltjahr(zeitreihe$YEAR)
  month <- schaltjahr(zeitreihe$MONTH)
  day <- schaltjahr(zeitreihe$DAY)
  avtemp <- schaltjahr(zeitreihe$Average.temp,aufteilen=TRUE)}

else{
  year <- zeitreihe$YEAR
  month <- zeitreihe$MONTH
  day <- zeitreihe$DAY
  avtemp <- zeitreihe$Average.temp}

#Fahrenheit nach Celsius umrechnen (in den USA)
if ((celsius==TRUE)&(stadt>=26)) {avtemp <- f2c(avtemp)}

res <- data.frame(YEAR=year,MONTH=month,DAY=day,Average.temp=avtemp)
}

```

A.2 Schätzung von Trend und Saisoneffekt

A.2.1 Berechnung des Saisoneffekts mit Tagesmitteln

Hier werden die Tagesmittelwerte zur Schätzung des Saisoneffekts herangezogen. Optional kann der resultierende Saisoneffekt mittels Loess geglättet werden.

```

tagesmittelwert <- function(zeitreihe,glact=TRUE){

  tagesmittel <- NA

  anz <- length(zeitreihe)
  jahre <- trunc(anz/365)

```

```

jahre1 <- trunc(anz/365)+1
angebrochen <- anz-trunc(anz/365)*365 # Anzahl Tage im angebrochenen Jahr

# Bildung der Mittelwerte (Unterscheidung bei Tagen mit Messung im
# angebrochenen Jahr

kum <- function(i){ sum(zeitreihe[i+(0:(jahre-1))])}
kum1 <- function(i){ sum(zeitreihe[i+(0:(jahre1-1))])}

for (i in (1:angebrochen)){
  tagesmittel[i] <- sum(zeitreihe[i+(0:(jahre1-1))*365])/jahre1
}

for (i in ((angebrochen+1):365)) {
  tagesmittel[i] <- sum(zeitreihe[i+(0:(jahre-1))*365])/jahre
}

# Glättung des Saisonseffekts mit Loess (Grad 2)
if (glaet==TRUE) {
  tagesmittel <- loess(saison.tr~c(1:anz),degree=2,span=0.01)$fitted
  tagesmittel <- rep(tagesmittel[366:730],jahre1)[1:anz]
}

res <- c(rep(tagesmittel,jahre),tagesmittel[1:angebrochen])
}

```

A.2.2 Berechnung des Saisonseffekts mit Fourieranalyse

Hier wird der Saisonseffekt mit Hilfe der Fourieranalyse ($k=1,2$) angepasst. Die Anpassung erfolgt mit dem robusten Huber-Regressions-Schätzer.

```

# Saisonseffektschätzung mit k=1

sinus.1 <- function(zeitreihe){
  anz <- length(zeitreihe)

  saison.fit <- rlm((zeitreihe)~cos(2*pi/365*(1:anz))+sin(2*pi/365*(1:anz))-1)

  res <- saison.fit$coefficients[1]*cos(2*pi/365*(1:anz))\
    +saison.fit$coefficients[2]*sin(2*pi/365*(1:anz))
}

# Saisonseffektschätzung mit k=2

sinus.2 <- function(zeitreihe){
  anz <- length(zeitreihe)
  mat <- cbind(cos(2*pi/365*(1:anz)),sin(2*pi/365*(1:anz)),\cos(2*2*pi/365
    *(1:anz)),sin(2*2*pi/365*(1:anz)))
}

```

```

saison.fit <- rlm((zeitreihe)~cos(2*pi/365*(1:anz))+sin(2*pi/365*(1:anz))
+cos(2*2*pi/365*(1:anz))+sin(2*2*pi/365*(1:anz))-1)

res <- saison.fit$coefficients[1]*cos(2*pi/365*(1:anz))+
saison.fit$coefficients[2]*sin(2*pi/365*(1:anz))+saison.fit$coefficients[3]
*cos(2*2*pi/365*(1:anz))+saison.fit$coefficients[4]*sin(2*2*pi/365*(1:anz))
}

```

A.2.3 Iterative Berechnung von Trend und Saisonereffekt

Hier werden mit verschiedenen Verfahren Trend und Saisonereffekt geschätzt. Die Schätzung erfolgt iterativ.

```

saison.trend.iter <- function(zeitreihe,iter=5,trend.window=NULL,df=NULL,
grad=NULL,trend.method=c("splines","kern","polynom","lowess"),
saison.method=c("sinus1","sinus2","tagesmittel","tagesmittel.glatt")){

  anz <- length(zeitreihe)
  zeit <- c(1:anz)

#Erster Durchlauf

  # Trend anpassen
  if (trend.method=="kern"){
    # quadratischer Kern
    trend <- NW.kernel(zeit,zeitreihe,(trend.window)/2,kernel=4)
  }

  if (trend.method=="splines"){
    trend <- sm.spline(zeit,zeitreihe,df=df)
  }

  if (trend.method=="polynom"){
    trend <- fitted(lm(zeitreihe~poly(zeit,grad)))
  }

  if (trend.method=="lowess"){
    trend <- lowess((1:anz),zeitreihe,f=trend.window/anz)$y
  }

  # Saisonereffekt anpassen
  if (saison.method=="sinus1"){
    saison <- sinus.1(zeitreihe-trend)
  }

  if (saison.method=="sinus2"){
    saison <- sinus.2(zeitreihe-trend)
  }

  if (saison.method=="tagesmittel"){
    saison <- tagesmittelwert(zeitreihe-trend)
  }
}

```

```
if (saison.method=="tagesmittel.glatt"){
  saison <- tagesmittelwert(zeitreihe-trend,glaet=TRUE)
}

# Iterationen

if (iter!=0){
  for (l in (1:iter)) {

    # Trend anpassen
    if (trend.method=="kern"){
      trend <- NW.kernel(zeit,zeitreihe-saison,(trend.window)/2,kernel=4)
    }

    if (trend.method=="splines"){
      trend <- sm.spline(zeit,zeitreihe-saison,df=df)
    }

    if (trend.method=="polynom"){
      trend <- fitted(lm(zeitreihe-saison~poly(zeit,grad)))
    }

    if (trend.method=="lowess"){
      trend <- lowess((1:anz),zeitreihe-saison,f=trend.window/anz)$y
    }

    # Saisoneffekt anpassen

    if (saison.method=="sinus1"){
      saison <- sinus.1(zeitreihe-trend)
    }

    if (saison.method=="sinus2"){
      saison <- sinus.2(zeitreihe-trend)
    }

    if (saison.method=="tagesmittel"){
      saison <- tagesmittelwert(zeitreihe-trend)
    }

    if (saison.method=="tagesmittel.glatt"){
      saison <- tagesmittelwert(zeitreihe-trend,glaet=TRUE)
    }

  }
}

list(tr=trend,sa=saison)

}
```

A.2.4 Berechnung von Trend und Saisoneffekt mit STL

```

stl1 <- function(zeitreihe,trend.window,saison.window="per",
heteroskedastisch=TRUE,saison.glaet=12){

  zeitreihe1 <- ts(zeitreihe,freq=365,start=c(1965,1))

  # Modifiziertes STL-Verfahren
  if (heteroskedastisch==TRUE) {
    trend.saison <- hstl(zeitreihe1, t.window=trend.window,
      s.window=saison.window, robust=TRUE)
  }

  # Gewöhnliches STL-Verfahren
  else {
    trend.saison <- stl(zeitreihe1,t.window=trend.window,
      s.window=saison.window,robust=TRUE)
  }

  anz <- length(zeitreihe)
  angebrochen <- anz-trunc(anz/365)*365
  jahre <- trunc(anz/365)

  sa <- trend.saison$time.series[(1:anz),1]
  sa.ungelaettet <- sa

  # Glättung des STL-Saisoneffekts: Wiederhole den periodischen
  # Saisoneffekt dreimal, um bei der Glättung die Periodizität
  # zu erhalten

  if (saison.window=="per"){
    x <- (1:1095)
    y <- rep(sa[1:365],3)
    sa <- lowess(x,y,f=(saison.glaet/365),delta=3)
    sa <- sa$y[366:730]
    sa <- c(rep(sa,jahre),sa[1:angebrochen])
  }

  else{
    sa.right <- sa[(anz-365+1):(anz-365+100)]
    sa <- lowess(c(1:(anz+100)), c(sa,sa.right),f=(saison.glaet/anz),
      delta=3,iter=0)$y
    sa <- sa[1:anz]
  }

  list(tr=trend.saison$time.series[(1:anz),2], sa=sa,
    sa.ungeglaettet=sa.ungelaettet, gewichte=trend.saison$weights)
}

```

A.3 Berücksichtigung der Heteroskedastizität der Residuen

A.3.1 Schätzung der saisonalen Standardabweichung

Schätzung der saisonalen Standardabweichung mit empirischem Standardabweichungsschätzer und mit Lowess.

```
standardabweichung <- function(zeitreihe,glaet=FALSE){

  anz <- length(zeitreihe)
  standardabw <- NA

  jahre <- trunc(anz/365)
  jahre1 <-trunc(anz/365)+1
  angebrochen <- anz-trunc(anz/365)*365 # Anzahl Tage im angebrochenen Jahr

  if (glaet==FALSE){

    for (i in (1:angebrochen)){
      standardabw[i] <- sqrt(sum((zeitreihe[i+(0:(jahre1-1))*365])^2)
        /jahre1)
    }

    for (i in ((angebrochen+1):365)){
      standardabw[i] <- sqrt(sum((zeitreihe[i+(0:(jahre-1))*365])^2)
        /jahre)
    }

    standardabw <- c(rep(standardabw,jahre),standardabw[1:angebrochen])
  }

  if (glaet==TRUE){

    x <- c(c(rep((1:365),jahre),(1:angebrochen)),c(rep((1:365),jahre),
      (1:angebrochen))+365,c(rep((1:365),jahre),(1:angebrochen))+730)
    y <- rep(abs(zeitreihe),3)

    # Glättung mit Loess (Grad 2, ohne Robustheitsiterationen)
    gl <- loess(y~x,span=(31*jahre)/length(zeitreihe))
    standardabw <- gl$fitted[(jahre*365+angebrochen+1):(jahre*365
      +angebrochen+365)]*1.253314
    standardabw <- c(rep(standardabw,jahre),standardabw[1:angebrochen])
  }

  res <- standardabw

}
```

A.3.2 Modifiziertes STL-Verfahren

Der nächste Programm-Code ist das modifizierte STL-Verfahren. Dabei wird bei der Gewichtung die Heteroskedastizität der Residuen berücksichtigt. Dieser R-Code ruft ein C-Programm auf (vgl. Anhang B), in dem Trend und Saisoneffekt (unter Einbezug der Heteroskedastizität) geschätzt werden. Die Robustheitsiterationen werden von R aus durchgeführt, um die saisonale Standardabweichung mit der bestehenden R Funktion zu schätzen.

```
hstl <- function (x, s.window = NULL, s.degree = 0, t.window = NULL,
  t.degree = 1, rob.iter=15, na.action = na.fail)
{
  # Bereinigung der Eingabe;
  # Fensterbreiten auf ungerade Zahlen setzen
  nextodd <- function(x) {
    x <- round(x)
    if (x%%2 == 0)
      x <- x + 1
    as.integer(x)
  }

  x <- na.action(as.ts(x))
  if (is.matrix(x))
    stop("only univariate series are allowed")
  n <- length(x)
  period <- frequency(x)
  if (is.null(s.window))
    stop("s.window is missing with no default")
  if (period < 2 || n <= 2 * period)
    stop("series is not periodic or has less than two periods")
  periodic <- FALSE
  if (is.character(s.window))
    if (is.na(pmatch(s.window, "periodic")))
      stop("unknown value for s.window")

  # Wenn periodisch sehr grosses Saisonfenster
  else {
    periodic <- TRUE
    s.window <- 10 * n + 1
    s.degree <- 0
  }
  s.degree <- as.integer(s.degree)
  if (s.degree < 0 || s.degree > 1)
    stop("s.degree must be 0 or 1")
  t.degree <- as.integer(t.degree)
  if (t.degree < 0 || t.degree > 1)
    stop("t.degree must be 0 or 1")

  ni <- 1          # nur je ein Durchlauf der inneren Iteration;
                  # dafür äussere Robustheitsiterationen

  niter <- 0       # Keine äusseren Iterationen im C Code:
                  # die äussere Iterationen werden von R
```

```

# aus durchgeführt

# Berechnung des Trendfensters wenn keine Eingabe
if (is.null(t.window)){
  t.window <- nextodd(ceiling((1.5 * period)/(1 - (1.5/s.window))))
}

# Grad und Fensterbreite des Low-Pass Filters
l.degree <- t.degree
l.window <- nextodd(period)

trend1      <- rep(0,length(x))      # Trend zu Beginn 0
standardabw <- rep(1,length(x))      # Standardabweichung zu Beginn 1

# Aufruf des modifizierten STL C-Codes
#
# Gegenüber dem gewöhnlichen STL wurde der Abstand zwischen zwei
# vom Lowess ausgewerteten Zeitpunkten (d.h. nsjump (Saisoneffekt),
# ntjump (Trend), nljump (Low-Pass Filter) ) erhöht.
# Dadurch müssen weniger Zeitpunkte linear interpoliert werden.

z <- .C("stlmod_", as.double(x), as.integer(n), as.integer(period),
  as.integer(s.window), as.integer(t.window),
  as.integer(l.window), s.degree, t.degree, l.degree,
  nsjump = as.integer(ceiling(s.window/50)),
  ntjump = as.integer(ceiling(t.window/50)),
  nljump = as.integer(ceiling(l.window/50)),
  as.integer(ni), niter = as.integer(niter),
  weights = double(n), seasonal = double(n),
  trend = as.double(trend1), double((n + 2 * period) * 5),
  stdev = as.double(standardabw),dl=as.integer(1))

# Äussere STL-Iterationen
# (mit Berücksichtigung der Heteroskedastizität)

for (i in (1:rob.iter)) {

  # Schätzung der saisonalen Standardabweichung
  standardabw <- standardabweichung(x-z$seasonal-z$trend)

  # Aufruf des modifizierten STL C-Codes (mit Übergabe der
  # saisonalen Standardabweichung)
  z <- .C("stlmod_", as.double(x), as.integer(n), as.integer(period),
    as.integer(s.window), as.integer(t.window),
    as.integer(l.window), s.degree, t.degree, l.degree,

```

```

    nsjump = as.integer(ceiling(s.window/50)),
    ntjump = as.integer(ceiling(t.window/50)),
    nljump = as.integer(ceiling(l.window/50)),
    as.integer(ni), niter = as.integer(niter),
    weights = double(n), seasonal = as.double(z$seasonal),
    trend = as.double(z$trend), double((n + 2 * period) * 5),
    stdev = as.double(standardabw),dl=as.integer(2))
  }

  # Mittelung der STL-Saisonkomponente wenn periodisch
  if (periodic) {
    which.cycle <- cycle(x)
    z$seasonal <- tapply(z$seasonal, which.cycle, mean)[which.cycle]
  }

  remainder <- as.vector(x) - z$seasonal - z$trend # Residuen

  y <- cbind(seasonal = z$seasonal, trend = z$trend, remainder = remainder)
  res <- list(time.series = ts(y, start = start(x), frequency = period),
    weights = z$weights, call = match.call())

  class(res) <- "stl"
  res
}

```

A.4 Box-Cox Transformation

A.4.1 Box-Cox Transformation durchführen

```

boxcox1 <- function(zeitreihe,lambda){
  if (lambda!=0) {zeitreihe <- ((abs(zeitreihe)^(lambda)-1)/lambda)}
  else {zeitreihe <- log(abs(zeitreihe))}

  res <- zeitreihe
}

```

A.4.2 Optimales λ wählen

```

boxcox.lambda <- function(zeitreihe,plotit=FALSE){
  alpha <- 13 # Verschiebung der Zeitreihe und Alpha
  trim <- 9 # Anzahl Messungen die an Rändern
  # abgeschnitten werden

  anz <- length(zeitreihe)

  # Messungen an Rändern nicht berücksichtigen
  zeitreihe1 <- sort(zeitreihe)
  zeitreihe1 <- zeitreihe1[(trim+1):(anz-trim)]
}

```

```

anz <- length(zeitreihe1)

# Log-Likelihoodfunktion berechnen
lambda.bc <- boxcox((alpha-zeitreihe1)~rep(1,anz) , lambda = seq(-1, 2,
  len = 100),plotit=plotit)

la <- which.min(lambda.bc$y)          # Optimales Lambda berechnen
lambdahat <- lambda.bc$x[la]

list(lambda=lambdahat,loglik=lambda.bc,alpha=alpha)

}

```

A.5 Zerlegung der Temperaturzeitreihe

Zerlegung einer Temperatur-Zeitreihe in eine stationäre Zeitreihe. Dabei werden verschiedene Trend- und Saisoneffekt-Schätzungen zugelassen. Anschliessend wird ein AR-Prozess angepasst (Ordnungswahl mit AIC und Schranke). Dies ist das Hauptprogramm für die Analyse der Temperaturzeitreihen.

```

datenanalyse <- function(ort,trend.method=c("stl","splines","kern",
  "polynom","lowess"), trend.window=NULL, df=NULL, grad=NULL, iter=5,
  saison.window="per", saison.glaet=12, saison.method=c("sinus1",
  "sinus2","tagesmittel","tagesmittel.glatt"), heteroskedastisch=TRUE,
  standardabweichung.glaet=TRUE, box.cox=TRUE, celsius=TRUE,
  zeitreihe=NULL, ende=-1, arfit=TRUE){

# ort                : Zu untersuchender Ort (Angabe der Nummer)
# zeitreihe          : Analyse einer vorgegebenen Zeitreihe (statt Angabe
#                      eines Orts)
# trend.method       : Methode zur Bestimmung des Trends
# trend.window       : Breite des Trend-Fensters bei stl, Lowess und
#                      Kernglätter
# df                 : Anzahl Freiheitsgrade bei Splines
# grad               : Grad der polynomialen Regression (bei polynomialer
#                      Trendbestimmung)
# iter               : Anzahl Iterationen (Trend-Saison) bei "splines",
#                      "kern" und "polynom"
# saison.method       : Methode zur Bestimmung des Saisoneffekts
#                      (nicht bei nötig stl)
# saison.window       : Fensterbreite für STL-Saisonschätzer
# standardabweichung.glaet : Saisonale Standardabweichung mit Lowess schätzen
# heteroskedastisch   : Modifizierter STL verwenden
#                      (für heteroskedastische Residuen)
# box.cox             : Box-Cox Transformation durchführen
# ende               : Letzte Messung, die für die Analyse berücksichtigt wird
# C                  : Celsius statt Fahrenheit verwenden
# arfit               : Ganze Information des AR-Fits in Resultat speichern
#                      (notwendig für Vorhersage)

  trend.method <- match.arg(trend.method)
  res <- list()

```

```

if (is.null(zeitreihe)==FALSE){
  res$Average.temp <- zeitreihe
}

else{
  dat <- lesen(ort,celsius=celsius)
  anz <- length(dat$Average.temp)

  if (ende==1){
    res$Average.temp <- dat$Average.temp[1:anz]
  }

  else {
    res$Average.temp      <- dat$Average.temp[1:ende]
    res$Average.temp.ganz <- dat$Average.temp[1:anz]
  }
}

res$trend.method <- trend.method
res$saizon.method <- saizon.method
if (is.null(trend.window)==FALSE) {res$trend.window <- trend.window}
if (is.null(saizon.window)==FALSE) {res$saizon.window <- saizon.window}

# Schätzung von Trend und Saisoneneffekt

switch(trend.method,

"stl"={
  stl.fit <- stl1(res$Average.temp,trend.window,
    saizon.window=saizon.window,saizon.glaet=saizon.glaet,
    heteroskedastisch=heteroskedastisch)

  res$trend      <- stl.fit$tr
  res$saizon     <- stl.fit$sa
  res$saizon.glaet <- saizon.glaet
  res$saizon.ungeglaettet <- stl.fit$sa.ungeglaettet
  res$gewichte   <- stl.fit$gewichte
  res$saizon.method <- trend.method

  residuen <- res$Average.temp-stl.fit$tr-stl.fit$sa
},

"splines"={
  spline.fit <- saizon.trend.iter(res$Average.temp,iter=iter,df=df,
    trend.method="splines",saizon.method=saizon.method)
  res$trend <- spline.fit$tr
  res$saizon <- spline.fit$sa

  residuen <- res$Average.temp-spline.fit$tr-spline.fit$sa
},

```

```

"kern"={
  kern.fit <- saison.trend.iter(res$Average.temp,iter=iter,
    trend.window=trend.window,trend.method="kern",
    saison.method=saison.method)
  res$trend <- kern.fit$str
  res$saison <- kern.fit$sa

  residuen <- res$Average.temp-kern.fit$str-kern.fit$sa
},

"lowess"={
  lowess.fit <- saison.trend.iter(res$Average.temp,iter=iter,
    trend.window=trend.window,trend.method="lowess",
    saison.method=saison.method)
  res$trend <- lowess.fit$str
  res$saison <- lowess.fit$sa

  residuen <- res$Average.temp-kern.fit$str-kern.fit$sa
},

"polynom"={
  polynom.fit <- saison.trend.iter(res$Average.temp,iter=iter,
    grad=grad,trend.method="polynom",saison.method=saison.method)
  res$trend <- polynom.fit$str
  res$saison <- polynom.fit$sa

  residuen <- res$Average.temp-polynom.fit$str-polynom.fit$sa
})

# saisonale Standardabweichung schätzen
res$standardabweichung <- standardabweichung(res$Average.temp
-res$trend-res$saison,glaet=standardabweichung.glaet)

# Normierte Residuen I_t berechnen
resid.I <- residuen/res$standardabweichung

# Normierte Residuen Box-Cox Transformieren
res$boxcox <- box.cox

if (box.cox==TRUE){
  lambda <- boxcox.lambda(resid.I)
  res$boxcox.lambda <- lambda$lambda
  res$boxcox.loglik <- lambda$loglik
  res$boxcox.alpha <- lambda$alpha

  resb <- boxcox1(res$boxcox.alpha-resid.I, res$boxcox.lambda)
  res$boxcox.mean <- mean(resb)
  resid.X <- (-resb+res$boxcox.mean)

```

```

}

else{
  resid.X      <- resid.I-mean(resid.I)
  res$boxcox.mean <- mean(resid.I)
}

# Bestimmung des AR-Modells (Ordnung mit AIC)
ar.fit <- ar(resid.X,demean=FALSE)

res$arb.coef      <- ar.fit$ar
res$arb.order     <- ar.fit$order
res$arb.aic       <- ar.fit$aic
res$arb.ordermax  <- ar.fit$order.max
res$arb.resid     <- ar.fit$resid

# Bestimmung der AR-Ordnung mit AIC und Schranke
opt.order <- 0
ind <- FALSE
p <- 0
grenze <- 3

while (ind==FALSE){
  if (res$arb.aic[p+1]<=grenze){
    opt.order <- p
    ind <- TRUE
  }

  else {p <- p+1}
}

# Bestimmung des AR-Modells mit (Ordnung mit AIC und Schranke)
res$arb.orderopt <- opt.order
ar.fit <- ar(resid.X,aic=FALSE,order.max=opt.order,demean=FALSE)

res$arb.coefopt <- ar.fit$ar
res$arb.orderopt <- opt.order
res$arb.residopt <- ar.fit$resid
res$arb.varpred <- ar.fit$var.pred
res$grenze      <- grenze

if (arfit==TRUE){
  res$ar.fit <- ar.fit
  res$xt     <- resid.X
}

mires      <- res$arb.orderopt+1
mares      <- length(res$arb.residopt)
arresiduen <- res$arb.residopt[mires:mares]
res$arresiduen <- arresiduen

# Speichern der Informationen des Orts

```

```

if (is.null(zeitreihe)==TRUE){
  assign(paste(cities[ort],".res",sep=""),res, env= .GlobalEnv)
}

else {res}

}

```

A.6 Vorhersage der Temperatur

A.6.1 Vorhersage des Trends

```

trend.predict <- function(res,nr=nr){

  anz <- length(res$Average.temp)
  zeit <- c(1:anz)

  resid.t <- res$Average.temp-res$saison.ungeglatt

  # Loess Objekt erzeugen (wird für Vorhersage benötigt)
  trend.fit <- loess(resid.t[1:5]~c(1:5),,c(1:5),span=1,degree=1,
    family=c("gaussian"),control=loess.control(surface="direct"))
  trend.fit$pars$robust <- rep(1,anz)
  trend.fit$y <- resid.t
  trend.fit$x <- zeit
  trend.fit$weights <- res$gewichte # Gewichte w_t aus STL einsetzen
  trend.fit$pars$span <- (res$trend.window/anz)

  # Vorhersage des Trend
  res$trend.pred <- predict(trend.fit, data.frame(zeit = seq((anz+1),
    (anz+nr), 1)), se = FALSE)

}

```

A.6.2 Vorhersage der Temperatur

In dieser Funktion wird die Temperatur ausgehend von den Residuen x_t vorhergesagt.

```

ruecktransformation.arpredict <- function(res,nr=nr){

# res: Datenanalyse-Objekt
# nr : Anzahl zu vorhersagende Tage

  anz <- length(res$Average.temp)
  angebrochen <- anz-trunc(anz/365)*365
  ganze.jahre <- trunc(nr/365)+1

  if (angebrochen!=365){
    anfangstag <- angebrochen+1
  }
}

```

```

}

if (angebrochen==365){anfangstag <- 1}

# AR Modell
assign("resid.X",res$xt, env=.GlobalEnv)
resid.X <- predict(res$ar.fit,n.ahead=nr)$pred

# Box Cox Rücktransformation
residst <- res$boxcox.mean-resid.X
if (res$boxcox.lambda!=0) {
  residst <- (res$boxcox.lambda*residst+1)^(1/res$boxcox.lambda)
}

if (res$boxcox.lambda==0) {
  residst <- exp(residst)
}

resid.I <- res$boxcox.alpha-residst

# Saisonale Standardabweichung
stabw <- c(res$standardabweichung[anfangstag:365],
  rep(res$standardabweichung[1:365],ganze.jahre))
stabw <- stabw[1:nr]
resid.R <- resid.I*stabw

# Saisoneffekt Vorhersage (Mittel der letzten 2 Jahre
s1 <- res$saizon[(anz-364):anz]
s2 <- res$saizon[(anz-729):(anz-365)]
saizon <- (s1+s2)/2

saizon <- c(rep(saizon,(ganze.jahre+1)))
saizon <- saizon[1:nr]
saizon <- saizon+resid.R

#Trend vorhersage
res$trend.pred <- trend.predict(res,nr)

return(saizon+res$trend.pred)
}

```

A.7 Bootstrap

A.7.1 Rücktransformation mit Bootstrap-Residuen

```

ruecktransformation.zeitreihe <- function(res,normal=FALSE){

# res      : Datenanalyse-Objekt der Original Temperatur

```

```

# normal : bei TRUE : Parametrischer Bootstrap
#           bei FALSE: Residuen Bootstrap

anz <- length(res$Average.temp)

# Parametrischer Bootstrap
if (normal==TRUE){eps <- rnorm(anz)
                  eps <- eps*sqrt(res$arb.varpred)}

# Residuen Bootstrap
if (normal==FALSE){
  samp <- res$arresiduen
  samp.n <- sample((1:length(samp)),anz,replace=TRUE)
  eps <- samp[samp.n]
}

# AR-Prozess (aus Bootstrap-Sample)
init <- rep(0,res$arb.orderopt)
resid.X <- filter(ts(eps),filter=res$arb.coefopt,method="recursive",
  init=init)

# Box Cox Rücktransformation
residst <- res$boxcox.mean-resid.X

if (res$boxcox.lambda!=0) {
  residst <- (res$boxcox.lambda*residst+1)^(1/res$boxcox.lambda)
}

if (res$boxcox.lambda==0) {
  residst <- exp(residst)
}

resid.I <- res$boxcox.alpha-residst

# Saisonale Standardabweichung
stabw <- res$standardabweichung
resid.R <- resid.I*stabw

# Saisoneneffekt
saison <- res$saison
saison <- saison+resid.R

#Trend
temp.star <- saison+res$trend
}

```

A.7.2 Berechnung der Bootstrap-Realisationen T^*

```
simulation.bootstrap <- function(ort,ende=12714,f=0.8,sf=0.9,
  saison.glaet=12,nr=100,anz.sim=100,start=1,normal=FALSE,celsius=TRUE,
  heteroskedastisch=FALSE,theta.wahr=TRUE,mass="EDD"){

# ort          : Datensatz für Bootstrap
# ende         : Letzte Messung, die für Analyse berücksichtigt wird
# f            : Glättungsparameter Trend
# sf           : Glättungsparameter Saisoneffekt
# saison.glaet : Glättungsparameter STL-Nachglättung
# nr           : Anzahl zu vorhersagende Tage
# start        : Tag der Vorhersage, am dem die Laufzeit beginnt
# anz.sim      : Anzahl Bootstrap-Realisationen
# normal       : Parameter Bootstrap (TRUE), Residuen Bootstrap (FALSE)
# celsius      : Fahrenheit nach Celsius umrechnen
# heteroskedastisch: Modifiziertes STL-Verfahren verwenden
# theta.wahr   : Wahre Temperaturen im Vorhersagezeitraum bekannt (Vergleich)
# mass         : Verwendete Einheit (EDD, HDD oder CDD)

  if (celsius==TRUE) {ref.temp <- 18}
  else {ref.temp <- 65}

  jahre <- trunc(ende/365)+2

  datenanalyse((ort),trend.method="stl",trend.window=ende*f,
    saison.window=jahre*sf,saison.glaet=saison.glaet,
    standardabweichung.glaet=TRUE,heteroskedastisch=heteroskedastisch,
    box.cox=TRUE,ende=ende,arfit=TRUE,celsius=celsius)
  res <- get(paste(cities[ort],".res",sep=""))

  # Temperaturvorhersage Hut
  temp.pred <- rucktransformation.arpredict(res,nr)

  for (i in (1:anz.sim)){

    # Temperatur Hut (mit Bootstrap Residuen)
    t.hut <- rucktransformation.zeitreihe(res,normal=normal)

    # Modell an Temperatur Hut anpassen
    parameter.stern <- datenanalyse(0,trend.method="stl",trend.window=
      ende*f,saison.window=jahre*sf,saison.glaet=saison.glaet,
      heteroskedastisch=heteroskedastisch,standardabweichung.glaet=TRUE,
      box.cox=TRUE,ende=ende,arfit=TRUE,zeitreihe=t.hut)

    # Temperaturvorhersage Stern
    t.stern <- rucktransformation.arpredict(parameter.stern,nr)

    # Berechnung der vorhergesagten DD in Modell Stern
    if (i==1){
      if (mass=="EDD"){
        edd <- sum(abs(ref.temp-t.stern)[start:nr])
      }
    }
  }
}
```

```

    }
    if (mass=="CDD"){
      edd <- sum(pmax((t.stern-ref.temp)[start:nr],rep(0,nr-start+1)))
    }
    if (mass=="HDD"){
      edd <- sum(pmax((-t.stern+ref.temp)[start:nr],rep(0,nr-start+1)))
    }
  }
}

else{

  if (mass=="EDD"){
    edd <- c(edd, sum(abs(ref.temp-t.stern)[start:nr]))
  }
  if (mass=="CDD"){
    edd <- c(edd, sum(pmax((t.stern-ref.temp)[start:nr],rep(0,nr-
      start+1))))
  }
  if (mass=="HDD"){
    edd <- c(edd, sum(pmax((-t.stern+ref.temp)[start:nr],rep(0,nr-
      start+1))))
  }
}
}

# Berechnung vorhergesagten DD im Modell Hut
if (mass=="EDD"){
  theta.hut <- sum(abs(ref.temp-temp.pred)[start:nr])
}
if (mass=="CDD"){
  theta.hut <- sum(pmax((temp.pred-ref.temp)[start:nr],rep(0,nr-
    start+1)))
}
if (mass=="HDD"){
  theta.hut <- sum(pmax((-temp.pred+ref.temp)[start:nr],rep(0,nr-
    start+1)))
}

# Berechnung der wahren DD (falls wahre DD bekannt)
if (theta.wahr==TRUE){
  tempwahr <- res$Average.temp.ganz[(ende+1):(ende+nr)]

  if (mass=="EDD"){
    theta.wahr <- sum(abs(ref.temp-tempwahr)[start:nr])
  }
  if (mass=="CDD"){
    theta.wahr <- sum(pmax((tempwahr-ref.temp)[start:nr],rep(0,nr-
      start+1)))
  }
  if (mass=="HDD"){
    theta.wahr <- sum(pmax((-tempwahr+ref.temp)[start:nr],rep(0,nr-
      start+1)))
  }
}
}

```

```
list(theta.stern=edd,theta.hut=theta.hut,theta.wahr=theta.wahr)
}
```

Anhang B

C-Code des modifizierten STL-Verfahrens

In diesem Kapitel wird der C-Code des modifizierten STL-Verfahrens angegeben. Dabei werden nur die beiden Unterrouninen abgedruckt, die von der Änderung betroffen sind.

B.1 Hauptroutine

```
/* Subroutine */ int stlmod_(y, n, np, ns, nt, nl, isdeg, itdeg, ildeg, nsjump,
    ntjump, nljump, ni, no, rw, season, trend, work, stdev,dl)
doublereal *y;
integer *n, *np, *ns, *nt, *nl, *isdeg, *itdeg, *ildeg, *nsjump, *ntjump, *nljump, *ni, *no;
doublereal *rw, *season, *trend, *work,*stdev;
integer *dl;
{
    /* System generated locals */
    integer work_dim1, work_offset, i__1;

    /* Local variables */
    extern /* Subroutine */ int stlrwts_();
    static integer i__, k, newnl, newnp, newns, newnt;
    static logical userw;
    extern /* Subroutine */ int stl1stp_();

    /*
        n                : length(y) */
    /*
        ns, nt, nl        : spans          for 's', 't' and 'l' smoother */
    /*
        isdeg, itdeg, ildeg : local degree for 's', 't' and 'l' smoother */
    /*
        nsjump,ntjump,nljump: length of interpolation   for 's', 't' and 'l' smoother */
    /*
        ni, no            : number of inner and outer (robust) iterations */

    /* Parameter adjustments */
    --trend;
    --season;
    --rw;
    --y;
    --stdev;
```

```

    work_dim1 = *n + 2 * *np;
    work_offset = 1 + work_dim1 * 1;
    work -= work_offset;

    /* first run ? */
    if (*dl == 1) { userw = FALSE_ ;}
    if (*dl == 2) { userw = TRUE_ ;}

    i__1 = *n;

/* the three spans must be at least three and odd: */
    newns = max(3,*ns);
    newnt = max(3,*nt);
    newnl = max(3,*nl);
    if (newns % 2 == 0) {
        ++newns;
    }
    if (newnt % 2 == 0) {
        ++newnt;
    }
    if (newnl % 2 == 0) {
        ++newnl;
    }
/* periodicity at least 2: */
    newnp = max(2,*np);
    k = 0;

/* compute robustness weights if not first run */

if (userw==TRUE_){
    i__1 = *n;
    for (i__ = 1; i__ <= i__1; ++i__) {
        work[i__ + work_dim1] = trend[i__] + season[i__];
    }
    stlrwts_(&y[1], n, &work[work_dim1 + 1], &rw[1],&stdev[1]);
}

L100:
    stl1stp_(&y[1], n, &newnp, &newns, &newnt, &newnl, isdeg, itdeg, ildeg,
        nsjump, ntjump, nljump, ni, &userw, &rw[1], &season[1], &trend[1],
        &work[work_offset]);
    ++k;
    if (k > *no) {
        goto L10; /* outer iterations are computed in R */
    }

    goto L100;

```

L10:

```
    return 0;
}
```

B.2 Bestimmung der Robustheitsgewichte

```
/* Subroutine */ int stlrwts_(y, n, fit, rw, stdev)
double real *y;
integer *n;
double real *fit, *rw, *stdev;
{
    /* System generated locals */
    integer i__1;
    double real d__1, d__2;

    /* Local variables */
    static double real cmad;
    static integer i__;
    static double real r__, c1, c9;
    extern /* Subroutine */ int psort_();
    static integer mid[2];

    /* Robustness Weights */
    /*      rw_i := B( |(y_i - fit_i)/stdev_i| / (6 M) )/(stdev_i)^2,   i = 1,2,...,n */
    /*      where B(u) = (1 - u^2)^2 * 1[|u| < 1]   {Tukey's biweight} */
    /*      and      M := median{ |(y_i - fit_i)/stdev_i| } */

    /* Parameter adjustments */
    --rw;
    --fit;
    --y;
    --stdev;

    /* Function Body */
    i__1 = *n;
    for (i__ = 1; i__ <= i__1; ++i__) {
        rw[i__] = (d__1 = (y[i__] - fit[i__])/stdev[i__], abs(d__1));
/* L7: */
    }
    mid[0] = *n / 2 + 1;
    mid[1] = *n - mid[0] + 1;
    psort_(&rw[1], n, mid, &c__2);
    cmad = (rw[mid[0]] + rw[mid[1]]) * (float)3.;
/*      = 6 * MAD */
    c9 = cmad * (float).999;
    c1 = cmad * (float).001;
    i__1 = *n;
    for (i__ = 1; i__ <= i__1; ++i__) {
        r__ = (d__1 = (y[i__] - fit[i__])/stdev[i__], abs(d__1));
```

```
        if (r__ <= c1) {
            rw[i__] = (float)1. / (stdev[i__] * stdev[i__])    ;
        } else if (r__ <= c9) {
/* Computing 2nd power */
            d__2 = r__ / cmad;
/* Computing 2nd power */
            d__1 = (float)1. - d__2 * d__2;
            rw[i__] = d__1 * d__1 / (stdev[i__] * stdev[i__]);
        } else {

            rw[i__] = (float)0.;
        }

    }
    return 0;
}
```

Anhang C

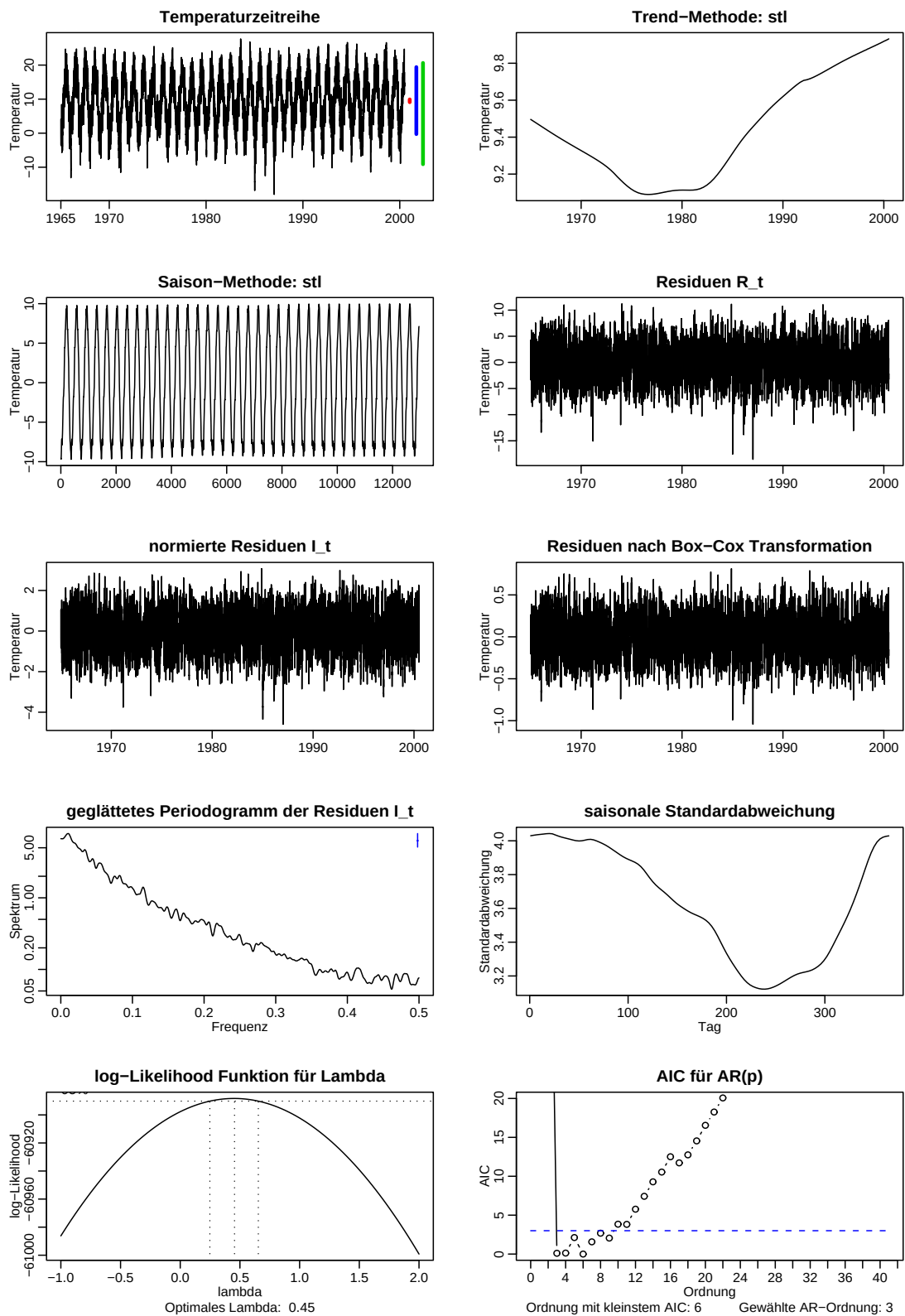
Beispiele der Temperaturzerlegung

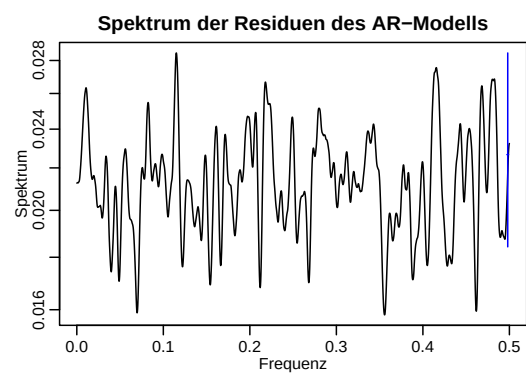
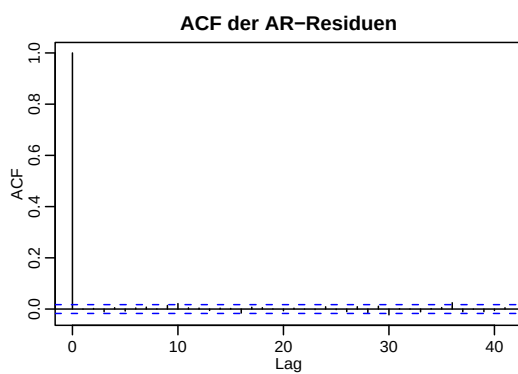
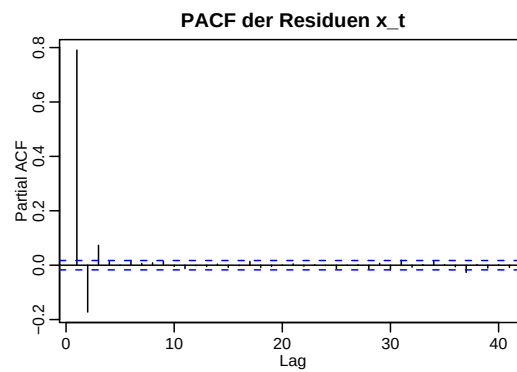
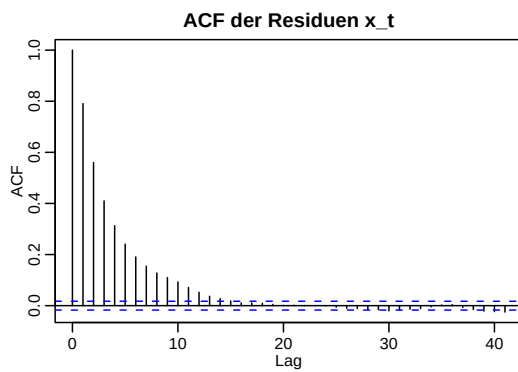
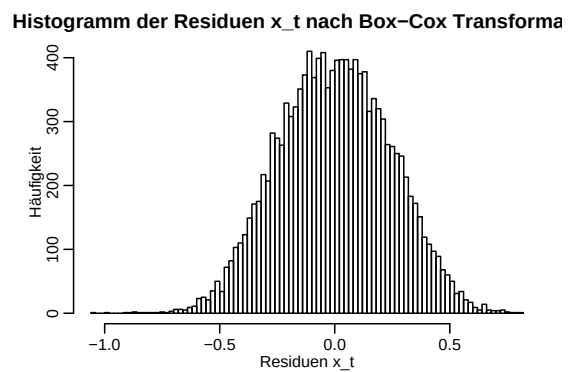
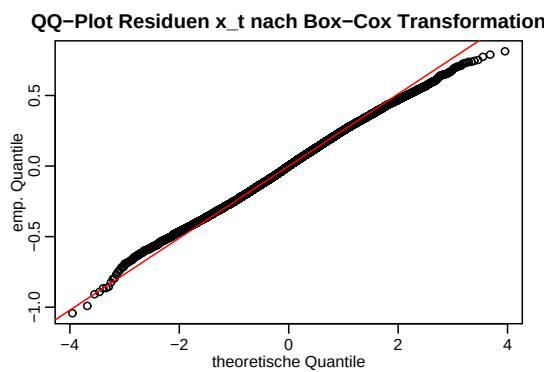
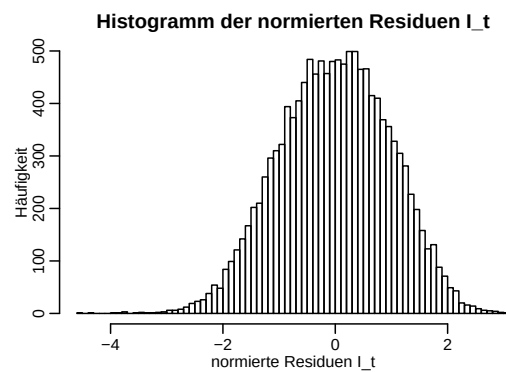
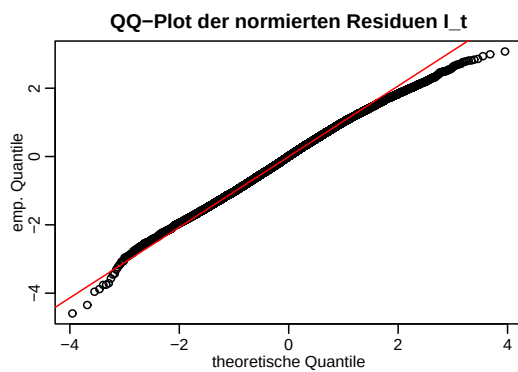
In diesem Anhang wird das Temperaturzeitreihenmodell auf vier Datensätze angewandt. Es handelt sich dabei um folgende Orte:

- Zürich
- New York
- San Antonio
- Vancouver

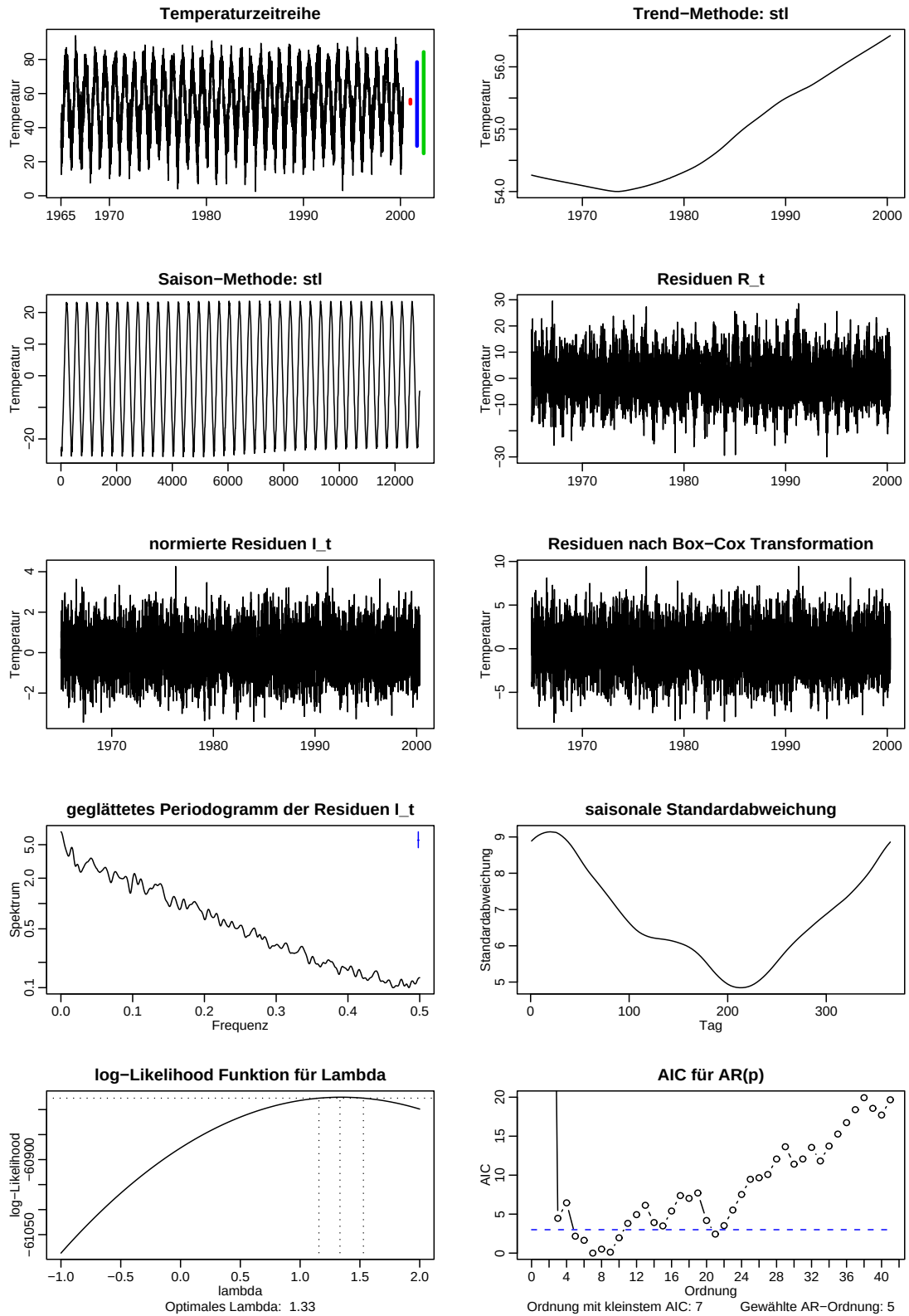
Für jede dieser Stationen werden die Plots mit den wesentlichen Modell-Informationen gezeigt.

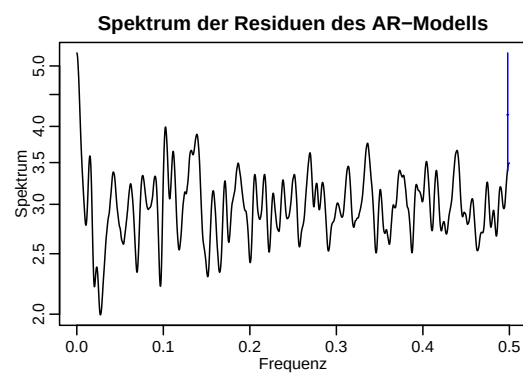
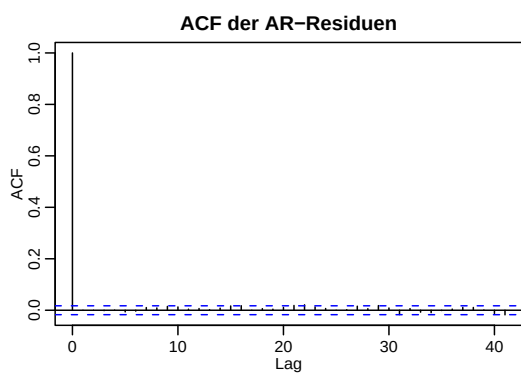
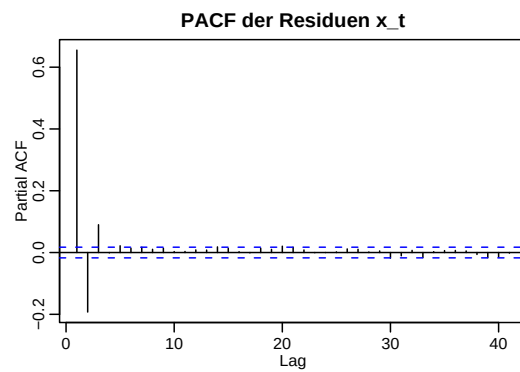
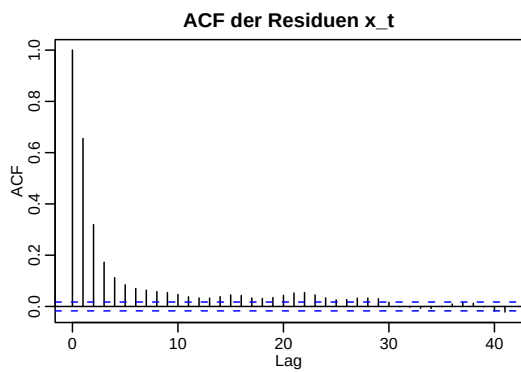
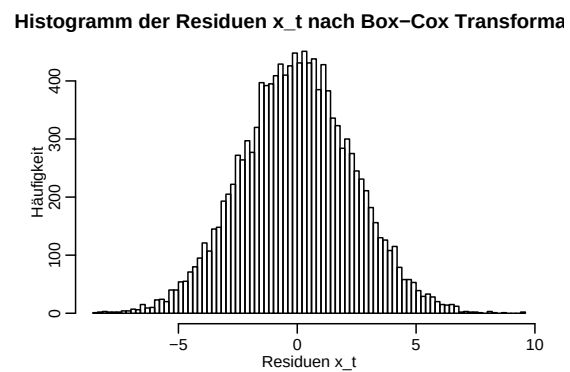
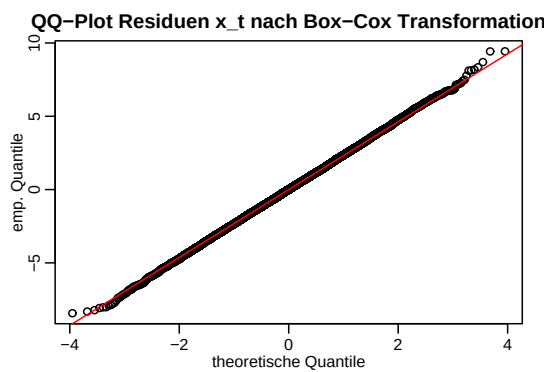
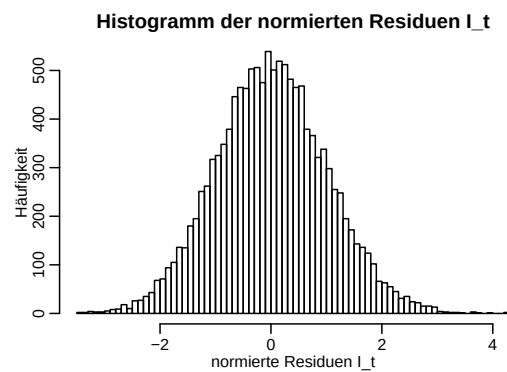
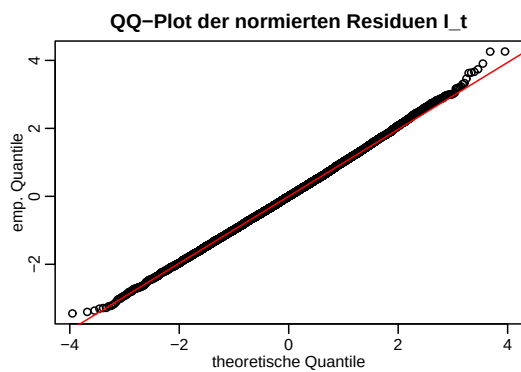
Datensatz Zürich



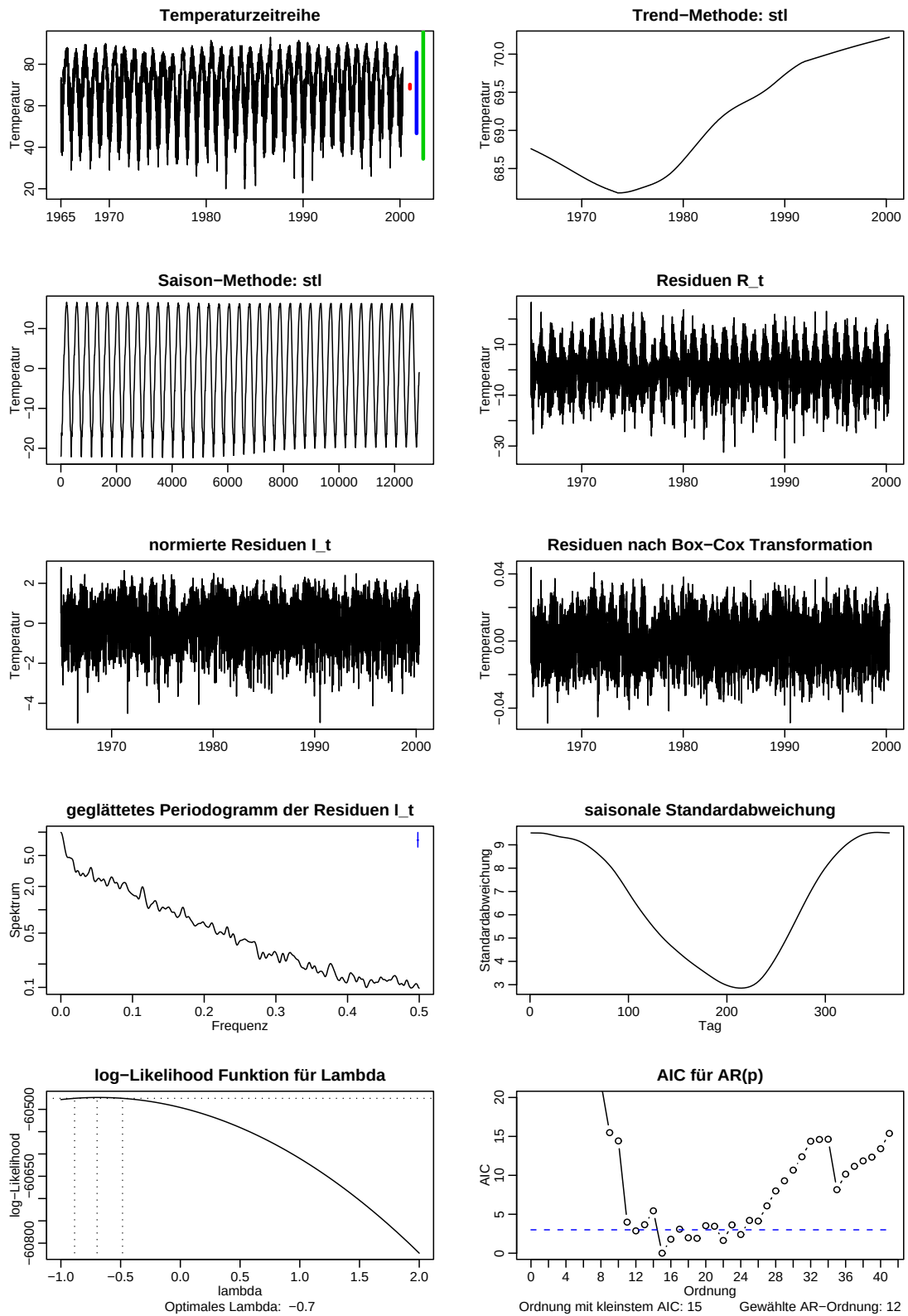


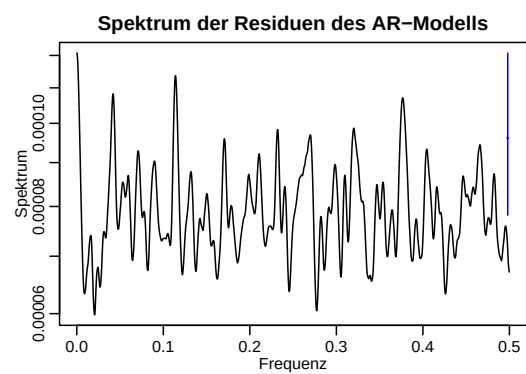
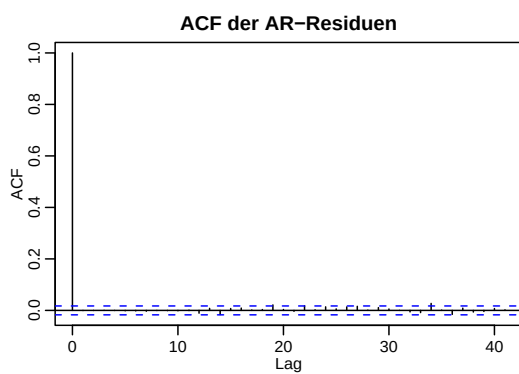
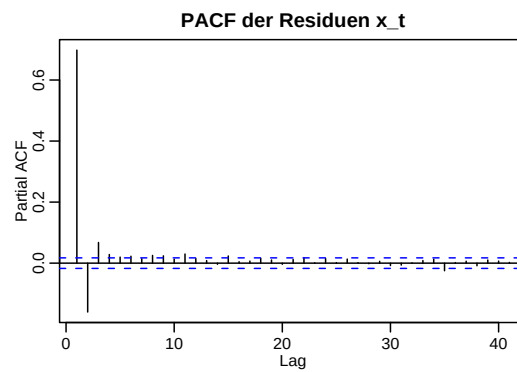
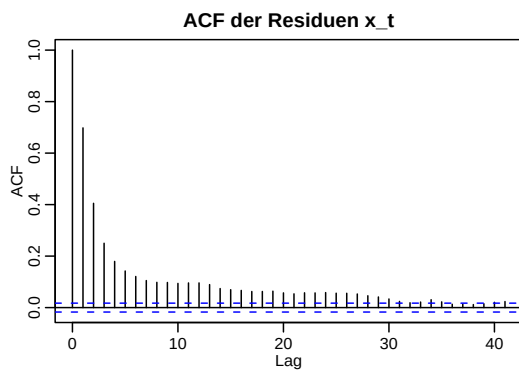
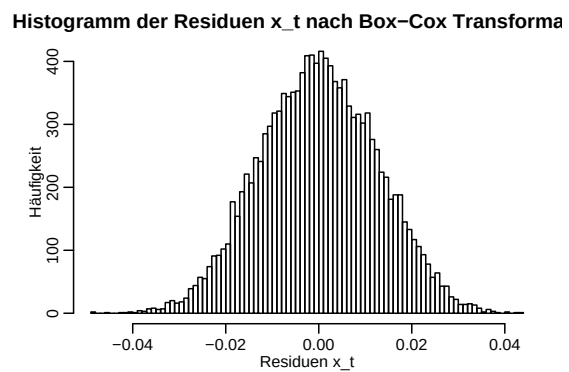
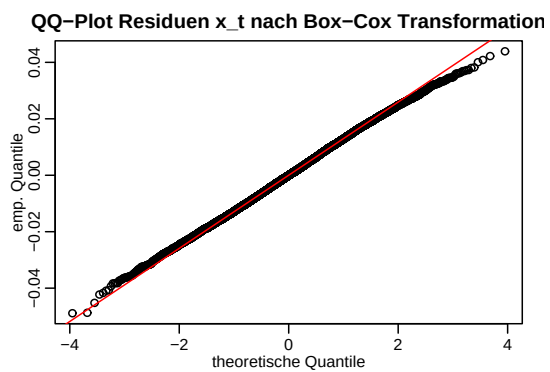
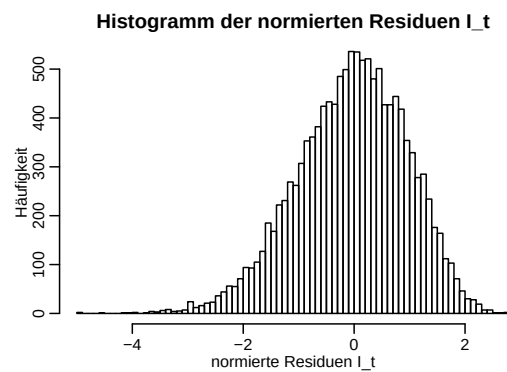
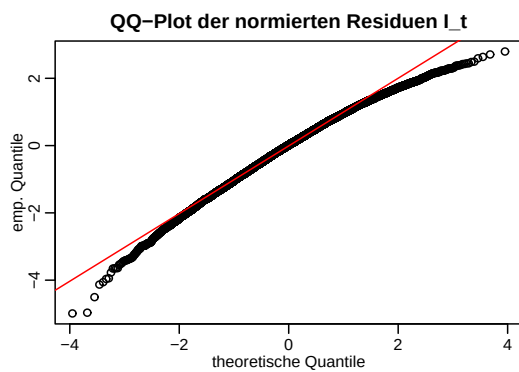
Datensatz New York



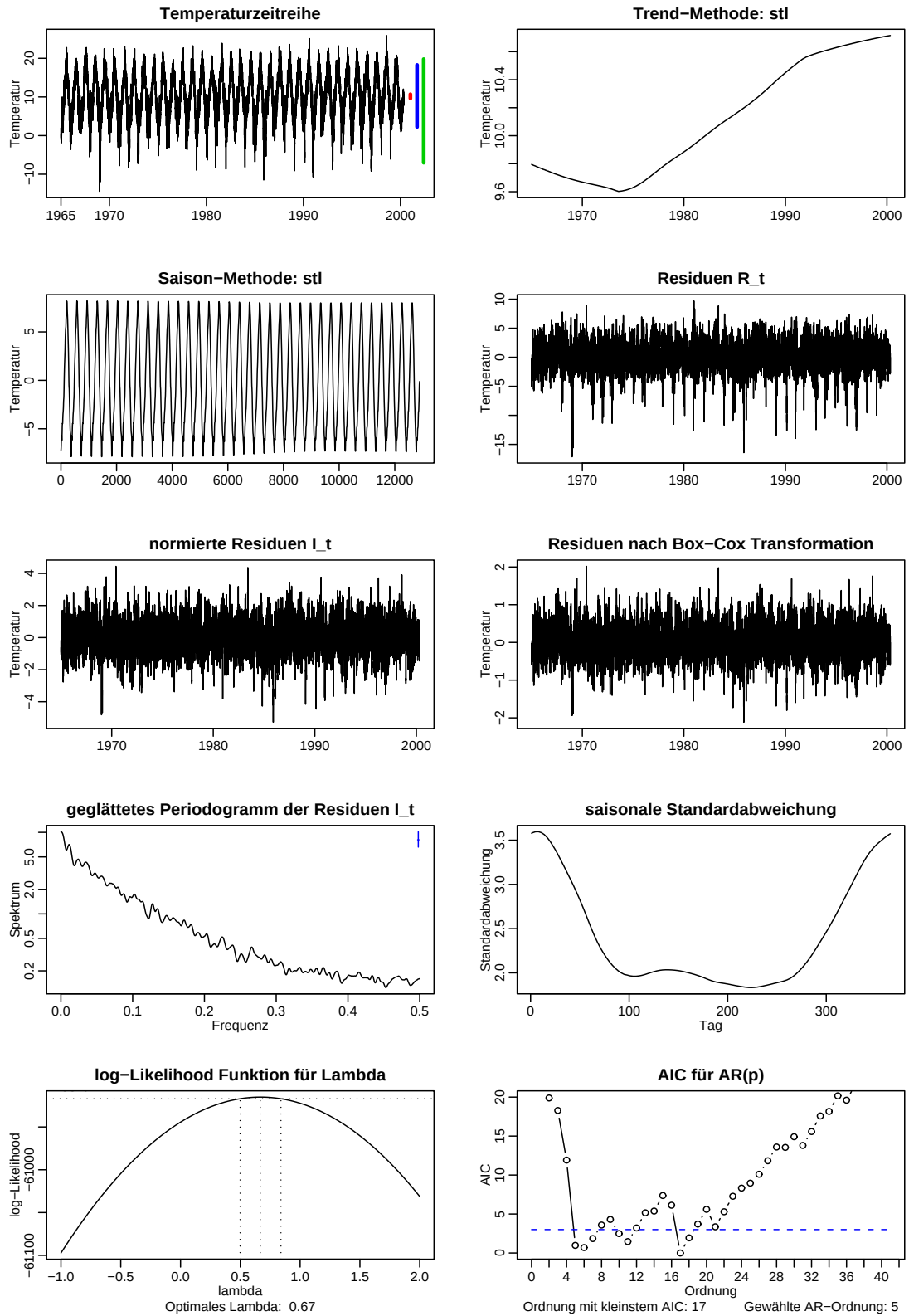


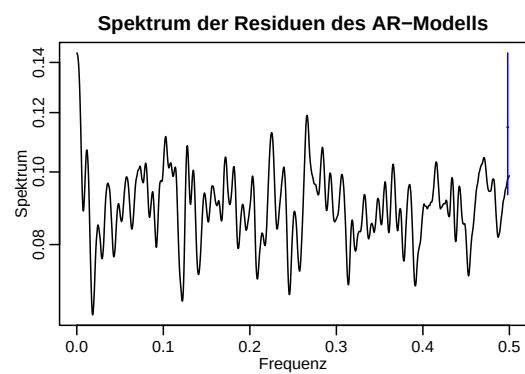
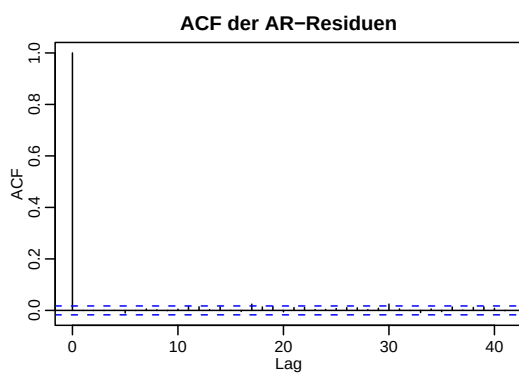
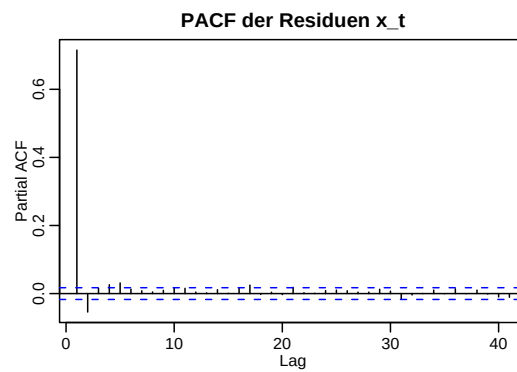
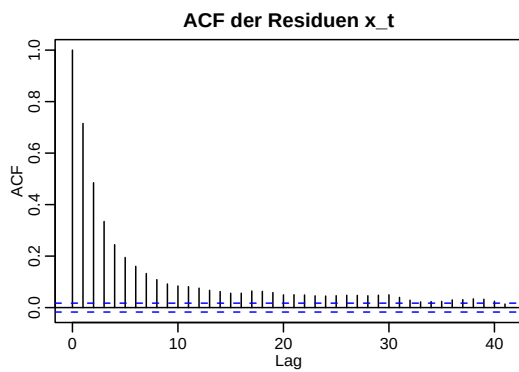
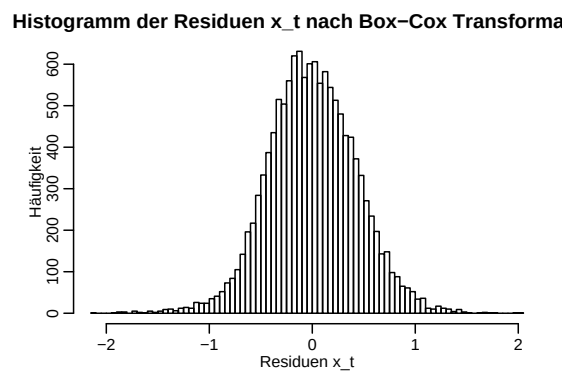
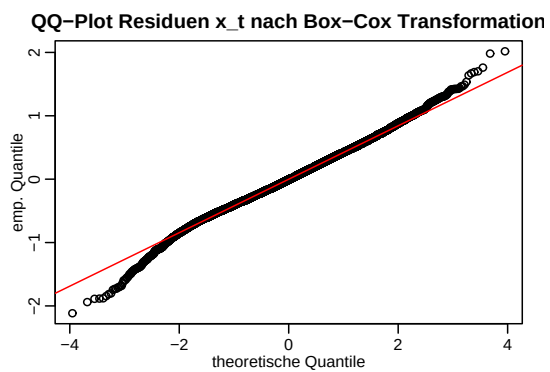
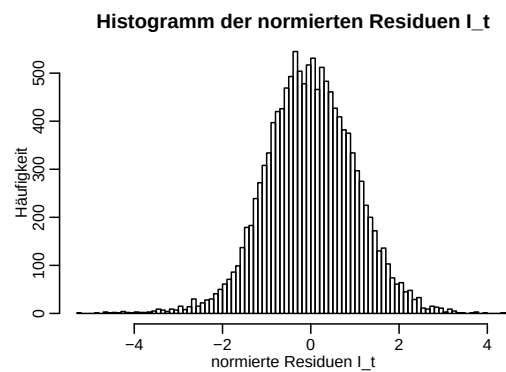
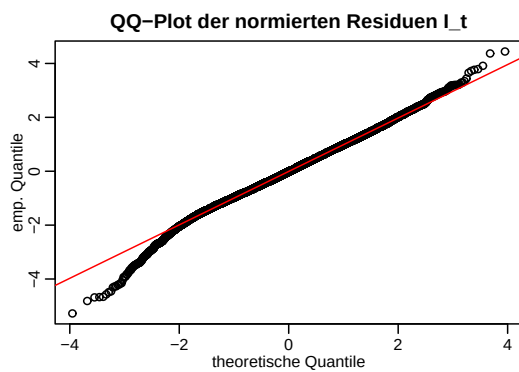
Datensatz San Antonio





Datensatz Vancouver





Literaturverzeichnis

- [1] G. E. P. Box, D. R. Cox, 1964, *An Analysis of Transformations*, Journal of the Royal Statistical Society B **26**, 211-252
- [2] P. J. Brockwell, R. A. Davis, 1987, *Time Series: Theory and Methods*, New York, Berlin: Springer-Verlag.
- [3] P. J. Brockwell, R. A. Davis, 1996, *Introduction to Time Series and Forecasting*, New York: Springer-Verlag.
- [4] C. Chatfield, 1996, *The Analysis of Time Series; An Introduction*, London: Chapman & Hall.
- [5] W. S. Cleveland, 1979, *Robust locally weighted regression and smoothing scatterplots*, Journal of the American Statistical Association, 75, 829-836.
- [6] R. B. Cleveland, W. S. Cleveland et al., 1990, *STL: A Seasonal-Trend Decomposition Procedure Based on Loess*, Journal of Official Statistics, Vol. 6, No. 1, 3-73.
- [7] W. S. Cleveland, 1993, *Visualizing Data*, Murray Hill, New Jersey: AT&T Bell Laboratories.
- [8] D. R. Easterling, G. A. Meehl et al., 2000, *Climate Extremes: Observations, Modeling and Impacts*, Science, Vol. 289, 2068-2074
- [9] B. Efron, R. J. Tibshirani, 1993, *An Introduction to Bootstrap*, New York, London: Chapman & Hall.
- [10] J. W. Hurrell, 1996, *Influence of variations in extratropical wintertime teleconnections on Northern Hemisphere temperature*, Geophysical Research Letters, Vol. 23, No. 6, 665-668
- [11] H.R. Künsch, 1997, *Skript zur Vorlesung Mathematische Statistik*, Zürich: ETH.
- [12] R. Schlittgen, B. H. J. Streitberg, 1999, *Zeitreihenanalyse*, München, Wien: Oldenbourg.
- [13] W.N Venables, B.D. Ripley, 1994, *Modern Applied Statistics with S-Plus*, New York: Springer-Verlag.