2 Das klassische Komponentenmodell

Das klassische Komponentenmodell ▪ Überprüfung der Stationaritätsannahme 2.1

Stationäre Prozesse weisen eine approximative Konstanz der interessierenden empirischen Momente auf.

Stationarität	**Nicht-Stationarität**
approx. konstantes Niveau	Trend
approx. konstante Streuung	die Varianz steigt mit dem Niveau an
approx. konstante Abhängigkeitsstruktur	Saison

Die Stationaritätsannahme kann überprüft werden

- durch visuelle Inspektion anhand des Zeitreihenplots
- rechnerisch – die Zeitreihe in gleich lange Segmente zerlegen und die approx. Konstanz der empirischen Maßzahlen für die Segmente überprüfen

Das klassische Komponentenmodell 2.2

Wir beschreiben die Daten (y_t) mit dem Modell:

$$Y_t := T_t + S_t + \varepsilon_t.$$

- Die Trendkomponente T_t erfasst die langfristige systematische Veränderung des mittleren Niveaus der Zeitreihe. Diese kann als polynomialer Trend oder als glatte Komponente modelliert werden.
- Die Saisonkomponente S_t erfasst jahreszeitlich bedingte Schwankungen, die sich relativ unverändert jedes Jahr wiederholen.
- Der Rest ε_t enthält die nicht zu erklärenden Einflüsse oder Störungen.

Anmerkung: Das klassische Komponentenmodell ist kein eindeutig definiertes Modell sondern ein Konzept. Daher große Flexibilität in der Modellierung der einzelnen Komponenten.

Ziel: Bestimmung einer trend- und saisonbereinigten Reihe, die mit Hilfe eines stationären und ergodischen Prozesses modelliert werden kann.

Das klassische Komponentenmodell ▪ Fallbeispiel Anzahlen von Luchspelzen – Überprüfung der Stationaritätsannahme 2.3

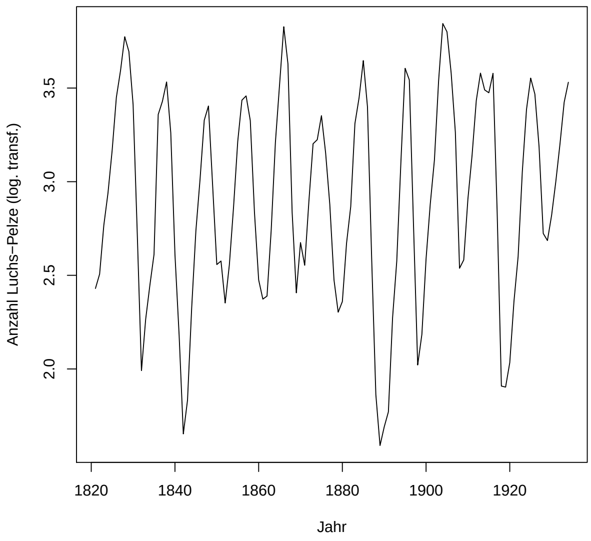

Abb. 2.1: Die Zeitreihe *Luchs*

Das klassische Komponentenmodell ▪ Fallbeispiel Anzahlen von Luchspelzen – Überprüfung der Stationaritätsannahme 2.4

	Segment-Mittelwerte	Segment-Standardabweichungen
[1,]	2.888403	0.5478154
[2,]	2.823095	0.5815392
[3,]	2.999493	0.5456087

Das klassische Komponentenmodell ▪ Fallbeispiel Anzahlen von Luchspelzen – Überprüfung der Stationaritätsannahme 2.5

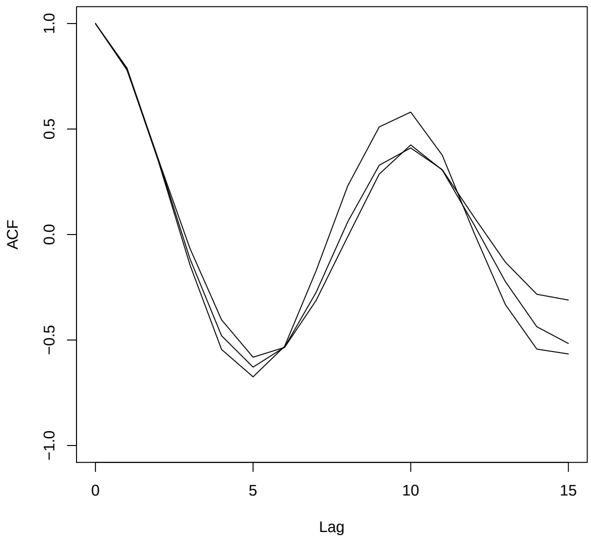

Abb. 2.2: Die ACFs für 3 Segmente der Zeitreihe *Luchs*

Das klassische Komponentenmodell ▪ Fallbeispiel Anzahlen von Luchspelzen – Code 2.6

```
y <- scan("C:/ZRA/Luchs.dat")
#Die Varianz steigt mit dem Niveau an.
#Durch die logarithmische Transformation wird die Varianz stabilisiert.
y <- ts(log10(y), start=1821, frequency=1)
tsp(y)
ts.plot(y, xlab="Jahr", ylab="Anzahl Luchs-Pelze (log.transf.)")

source("C:/ZRA/tsutil.r")
statcheck(y,3)
```

Das klassische Komponentenmodell ▪ Fallbeispiel Verkehrsunfälle – Überprüfung der Stationaritätsannahme 2.7

Die monatlichen Anzahlen der Verkehrsunfälle mit Personenschäden in Deutschland, Jan. 1979 – Dez. 2017.

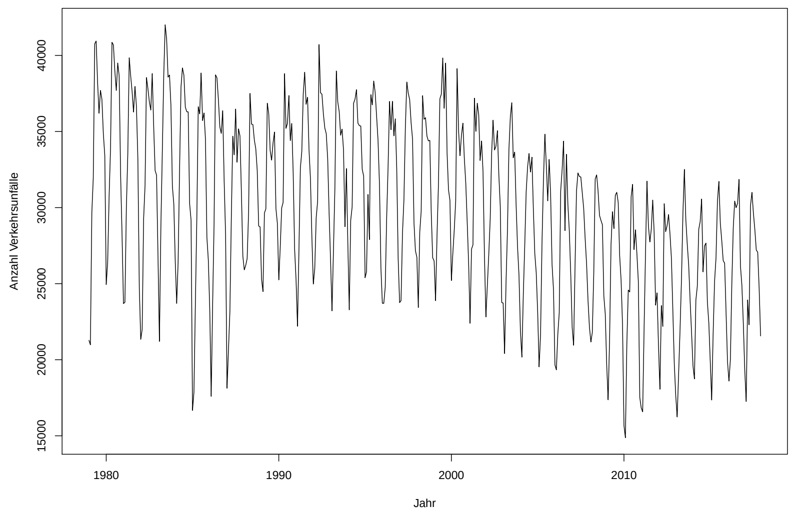

Abb. 2.3: Die Zeitreihe *Unfall*

Das klassische Komponentenmodell ▪ Fallbeispiel Verkehrsunfälle – Überprüfung der Stationaritätsannahme 2.8

	Segment-Mittelwerte	Segment-Standardabweichungen
[1,]	33470.22	6094.125
[2,]	31941.89	6659.370
[3,]	31028.96	4676.784
[4,]	32141.57	4789.325
[5,]	32059.17	4979.130
[6,]	32197.15	4458.553
[7,]	29166.13	4832.825
[8,]	26986.78	4535.740
[9,]	24538.76	4835.763
[10,]	25223.02	4197.525

Das klassische Komponentenmodell ▪ Fallbeispiel Verkehrsunfälle – Überprüfung der Stationaritätsannahme 2.9

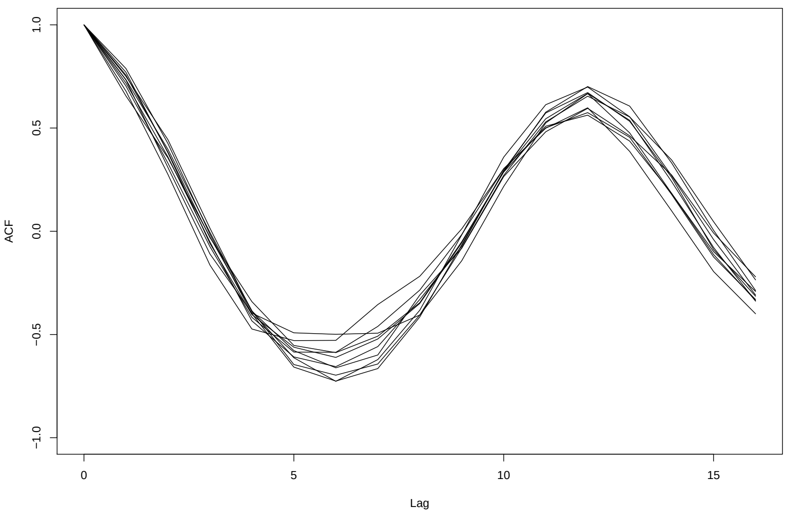

Abb. 2.4: Die ACFs für 10 Segmente der Zeitreihe *Unfall*

Das klassische Komponentenmodell ▪ Fallbeispiel Verkehrsunfälle – Code 2.10

```
Unfall <- scan("C:/ZRA/Unfall.dat")
Unfall.ts <- ts(Unfall, start=c(1979,1), frequency=12)
ts.plot(Unfall.ts, xlab="Jahr", ylab="Anzahl Verkehrsunfälle")
source("C:/ZRA/tsutil.r")
statcheck(Unfall.ts, 10)
```

2.1 Die Trendkomponente

2.1.1 Polynomialer Trend

Die Trendkomponente ▪ Polynomialer Trend 2.11

Wir beschreiben die Daten (y_t) mit dem Modell:

$$Y_t := T_t + \varepsilon_t \tag{2.1}$$

mit dem Trend

$$T_t := \beta_0 + \beta_1 t$$

oder allgemein, mit einem polynomialen Trend der Ordnung m

$$T_t := \beta_0 + \beta_1 t + \beta_2 t^2 + \ldots + \beta_m t^m.$$

- Das Modell (2.1) ist ein lineares Modell.
- Schätzung der Modellparameter mit Hilfe der Methode der kleinsten Quadrate. Dazu noch mehr in Kapitel 'Regressionsmodelle für Zeitreihen'.

Die Trendkomponente ▪ Fallbeispiel Verkehrsunfälle 2.12

$$Y_t := \beta_0 + \beta_1 t + \varepsilon_t$$

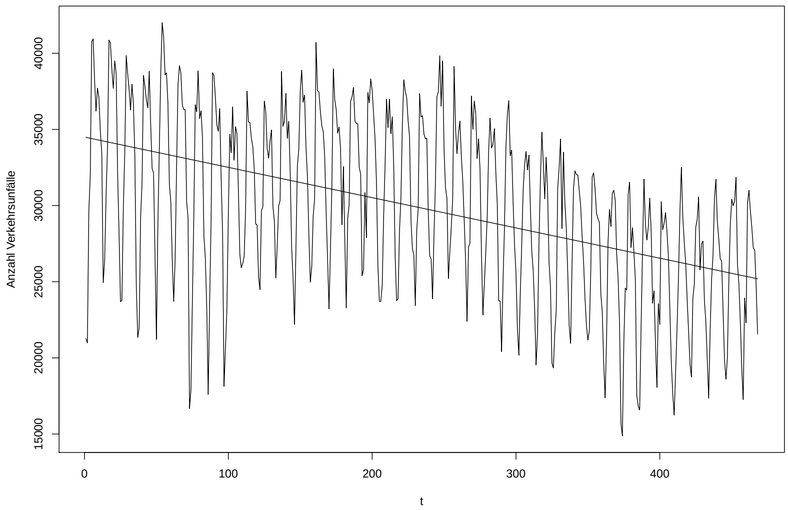

Abb. 2.5: Die Zeitreihe *Unfall* mit polynomialem Trend 1. Ordnung

Die Trendkomponente ▪ Fallbeispiel Verkehrsunfälle 2.13

$$Y_t := \beta_0 + \beta_1 t + \beta_2 t^2 + \beta_3 t^3 + \beta_4 t^4 + \varepsilon_t$$

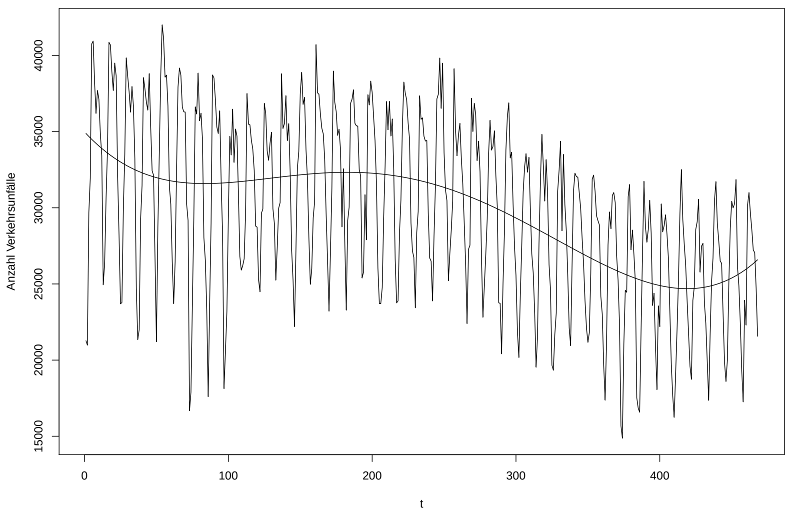

Abb. 2.6: Die Zeitreihe *Unfall* mit polynomialem Trend 4. Ordnung

Die Trendkomponente ▪ Fallbeispiel Verkehrsunfälle – Code

2.14

```
n <- length(Unfall)
t <- c(1:n)
out1 <- lm(Unfall ~1 + t)
plot(Unfall, xlab="t", ylab="Anzahl Verkehrsunfälle", type="l")
lines(t, out1$fitted.values)

t2 <- t^2
t3 <- t^3
t4 <- t^4
out2 <- lm(Unfall ~1 + t + t2 + t3 + t4)
plot(Unfall, xlab="t", ylab="Anzahl Verkehrsunfälle", type="l")
lines(t, out2$fitted.values)
```

2.1.2 Die glatte Komponente

Die Trendkomponente ▪ Die glatte Komponente 2.15

Die glatte Komponente kann mit dem Verfahren der gleitenden Durchschnitte, wie folgt, bestimmt werden:

1. Die Zeitreihe in gleich lange Segmente zerlegen
2. Das arithmetische Mittel für jedes Segment bestimmen
3. Die arithmetischen Mittel den jeweilligen zeitlichen Mittelpunkten der Segmente zuordnen
4. Wir erhalten eine Zeitreihe aus (gleitenden) arithmetischen Mitteln. Diese ist die glatte Komponente der Ausgangsreihe.

Die Trendkomponente ▪ Fallbeispiel Verkehrsunfälle 2.16

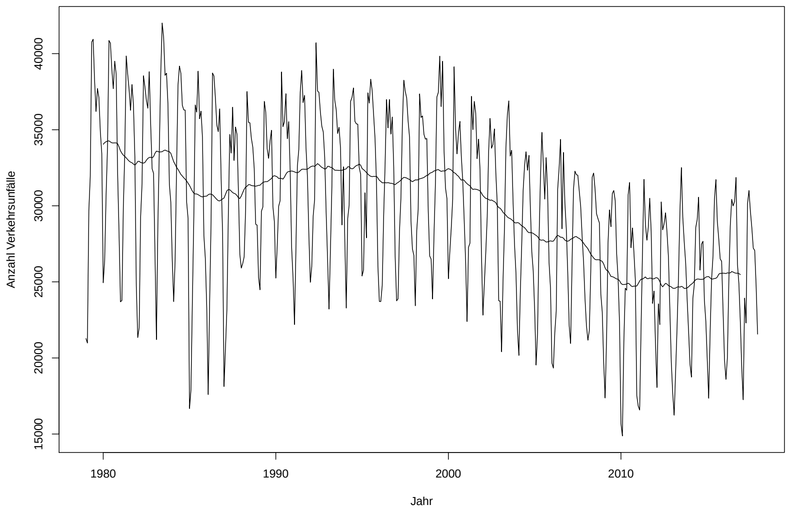

Abb. 2.7: Die Zeitreihe *Unfall* mit gleitenden 24er Durchschnitten

Die Trendkomponente ▪ Fallbeispiel Verkehrsunfälle – Code 2.17

```
source("C:/ZRA/tsutil.r")

glatt <- runmean(Unfall.ts, 24)

ts.plot(Unfall.ts, xlab="Jahr", ylab="Anzahl Verkehrsunfälle")

lines(glatt)
```

2.2 Die Saisonkomponente

Die Saisonkomponente 2.18

Sei s die Saisonperiode ($s = 4$ für Quartalsdaten, $s = 12$ für Monatsdaten).

Wir betrachten die Dummy-Variablen $S_{i,t}$, $i = 1, 2, \ldots, s$

$$S_{i,t} := \begin{cases} 1 & t \text{ findet in der Jahresperiode } i \text{ statt} \\ 0 & \text{sonst} \end{cases} .$$

Für eine zentrierte Reihe (y_t), die keinen Trend aufweist, verwenden wir das Modell:

$$Y_t := S_t + \varepsilon_t. \tag{2.2}$$

Alternativ für eine Reihe (y_t) mit polynomialem Trend:

$$Y_t := \alpha_0 + \alpha_1 t + \alpha_2 t^2 + \ldots + \alpha_p t^p + S_t + \varepsilon_t. \tag{2.3}$$

Die Saisonkomponente ▪ Modellierung der Saisonkomponente (1) 2.19

$$S_t := \beta_1 S_{1,t} + \beta_2 S_{2,t} + \ldots + \beta_{s-1} S_{s-1,t}$$

Die Regressoren für Quartalsdaten ($s = 4$) sind:

t	$S_{1,t}$	$S_{2,t}$	$S_{3,t}$
1	1	0	0
2	0	1	0
3	0	0	1
4	0	0	0
5	1	0	0
6	0	1	0
7	0	0	1
8	0	0	0
9	1	0	0
10	0	1	0
⋮	⋮	⋮	⋮

Die Saisonkomponente ▪ Modellierung der Saisonkomponente (1) 2.20

Wir schätzen das Modell (2.2) bzw. (2.3) mit der Methode der kleinsten Quadrate.

Die geschätzte Saison für Quartalsdaten ist

$$\widehat{S}_t = \widehat{\beta}_1 S_{1,t} + \widehat{\beta}_2 S_{2,t} + \widehat{\beta}_3 S_{3,t}.$$

Die saisonbereinigte Reihe ist

$$y_{t,b} = y_t - \widehat{S}_t + \left(\widehat{\beta}_1 + \widehat{\beta}_2 + \widehat{\beta}_3\right) S_{4,t}.$$

Die Saisonkomponente ■ Modellierung der Saisonkomponente (2) 2.21

Die Regressoren für Quartalsdaten sind:

t	$S^*_{1,t}$	$S^*_{2,t}$	$S^*_{3,t}$
1	1	0	0
2	0	1	0
3	0	0	1
4	*-1*	*-1*	*-1*
5	1	0	0
6	0	1	0
7	0	0	1
8	*-1*	*-1*	*-1*
9	1	0	0
10	0	1	0
⋮	⋮	⋮	⋮

Die Saisonkomponente ■ Modellierung der Saisonkomponente (2) 2.22

Wir schätzen das Modell (2.2) bzw. (2.3) mit der Methode der kleinsten Quadrate.

Die geschätzte Saison für Quartalsdaten ist

$$\widehat{S}_t = \widehat{\beta}_1 S^*_{1,t} + \widehat{\beta}_2 S^*_{2,t} + \widehat{\beta}_3 S^*_{3,t}.$$

Die saisonbereinigte Reihe ist

$$y_{t,b} = y_t - \widehat{S}_t.$$

Die Saisonkomponente ■ Fallbeispiel Steuern 2.23

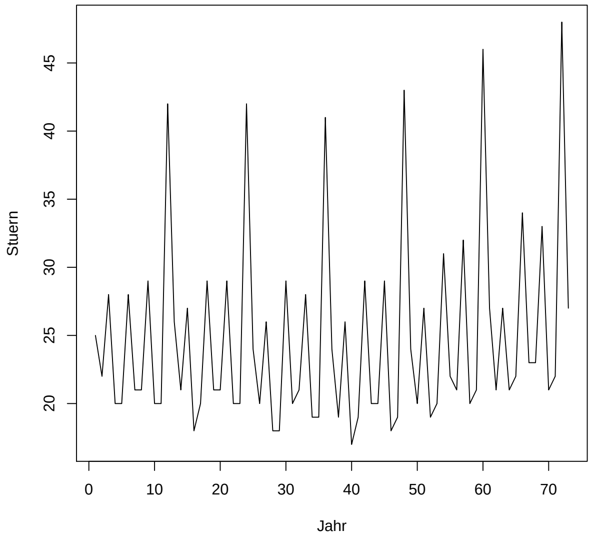

Abb. 2.8: Die Zeitreihe *Steuern*

Die Saisonkomponente ▪ Fallbeispiel Steuern 2.24

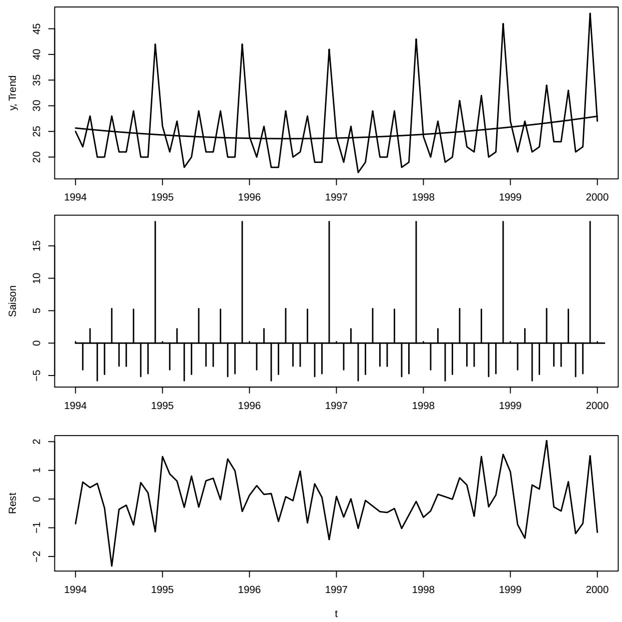

Abb. 2.9: Zerlegung der Zeitreihe *Steuern* mit polynomialem Trend 2. Ordnung und Saison

Die Saisonkomponente ▪ Fallbeispiel Steuern – Code 2.25

```
steuern <-scan("C:/ZRA/steuern.dat")
y <- ts(steuern, start=c(1994,1), frequency=12)
ts.plot(steuern, xlab="Jahr", ylab="Steuern")
source("C:/ZRA/tsutil.r")
simpledecomp(y,trend=2, saison=12)
```

Literaturhinweise 2.26

Schlittgen (2015), Angewandte Zeitreihenanalyse mit R: Kapitel 2.2, 2.3, 2.4