Beispiele in R: Einfacher gleitender Durchschnitt und Exponentielles Glätten

Regina Tüchler & Thomas Rusch

November 2, 2009

Beispiel: Einfacher Gleitender Durchschnitt der Nil-Daten:

Wir haben Daten über den Wasserstand des Nils bei Assuan für die Jahre 1901 bis 1970. Die Daten `nil.rda` werden geladen. Sie müssen sich dafür im working directory befinden.

```
> load("nil.rda")
```

Die Daten sind jetzt im Arbeitsspeicher

```
> ls()
```

und sind unter dem Namen `nil` verfügbar.

Die Daten sollen mit Hilfe des einfachen gleitenden Durchschnitts geglättet werden. Dabei werden wir das Verfahren mit verschieden großen Zeitfenstern mehrmals durchführen und die Ergebnisse in einem Plot vergleichen.

In `R` steht für den einfachen gleitenden Durchschnitt die Funktion `filter` zur Verfügung. Die Daten müssen dafür eine `R`-Zeitreihe sein (in `R`: `ts`). Das kann nachgeprüft werden:

```
> is.ts(nil)
```

```
[1] TRUE
```

Das Ergebnis der Abfrage ist: `TRUE`.

In der Argumentliste von `filter` stehen:
Die zu filternde ZR, `nil`.
Die Gewichte für den Filter. Zuerst wählen wir hier ein Zeitfenster $[t-2, t+2]$. Das entspricht 5 Zeitpunkten, die linear mit Gewichten $\frac{1}{5}$ kombiniert werden. Daher steht im Argument: `rep(1/5, 5)`
Die Optionen `method="convolution",sides=2` sind die Default-Werte und könnten daher auch weggelassen werden.

Die Ergebnisse sind unter dem Namen `nil.ma5` verfügbar.

```
> nil.ma5 <- filter(nil, rep(1/5, 5), method = "convolution", sides = 2)
```

Zum Vergleich glätten wir die ZR mit einem Zeitfenster der Länge 11 und speichern das Ergebnis unter `nil.ma11`, und mit einem Zeitfenster der Länge 3 (in `nil.ma3`).

```
> nil.ma11 <- filter(nil, rep(1/11, 11))
> nil.ma3 <- filter(nil, rep(1/3, 3))
```

In Fig. 1 zeichnen wir die ursprüngliche ZR und fügen die 3 geglätteten Reihen dazu.

```
> plot(nil)
> lines(nil.ma5, col = "blue")
> lines(nil.ma11, col = "red")
> lines(nil.ma3, col = "green")
```

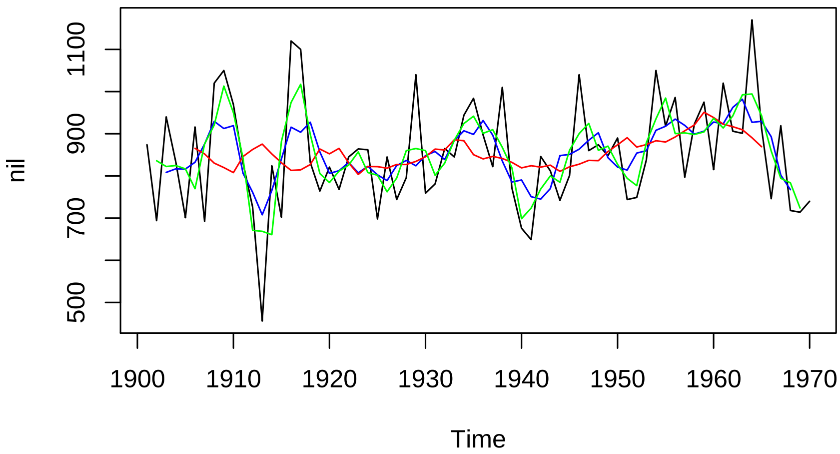

Figure 1: Nil-Daten und ihre Glättung mittels einfachem gleitenden Durchschnitt.

Beispiel: Exponentielles Glätten der Nil-Daten:

Beim einfachen gleitenden Durchschnitt wird ein um den Zeitpunkt t symmetrisches Zeitfenster $[t-q, t+q]$ beim Filtern verwendet. Die Outputreihe wird daher an beiden Enden um q Werte kürzer. Beim Verfahren des exponentiellen Glättens handelt es sich um ein rekursives Filterverfahren. Wir erhalten Outputwerte bis zum Endzeitpunkt T und können dieses Verfahren daher für eine Prognose über den Zeitpunkt T hinaus verwenden.

In diesem Beispiel sollen die Nil-Daten der Jahre 1901-1960 als Input für das exponentielle Glätten verwendet werden, um eine Prognose für die Jahre 1961-1975 zu erstellen.

1. Schritt: Zeitreihenplot wird in Fig. 2 gezeichnet.

2. Schritt: Analyse des Zeitreihenplots, Wahl einer Methode:

Welche Strukturen sind in Fig. 2 zu erkennen? Sollen die Daten transformiert werden? Gibt es Ausreißer? Sind Brüche erkennbar?

Es ist kein Trend bei diesen Jahresdaten erkennbar. Die Daten müssen nicht transformiert werden. Es sind keine Brüche erkennbar. Eher am Beginn der ZR, im Jahr 1913, ist ein auffallend kleiner Wert zu beobachten, den wir hier einfach mitmodellieren können, da er die Prognose ab dem Jahr 1961 kaum mehr beeinflusst.

Ein Verfahren, das das Niveau modelliert, kann gut verwendet werden. Wir wählen daher das exponentielle Glätten.

3. Schritt: Die Daten der Jahre 1901-1960 sollen Input für das exponentielle Glätten sein, um eine Prognose für die Jahre 1961-1975 zu erstellen. Die Daten teilen wir in `past` und in `future`:

```
> past <- window(nil, end = 1960)
```

```
> plot(nil)
```

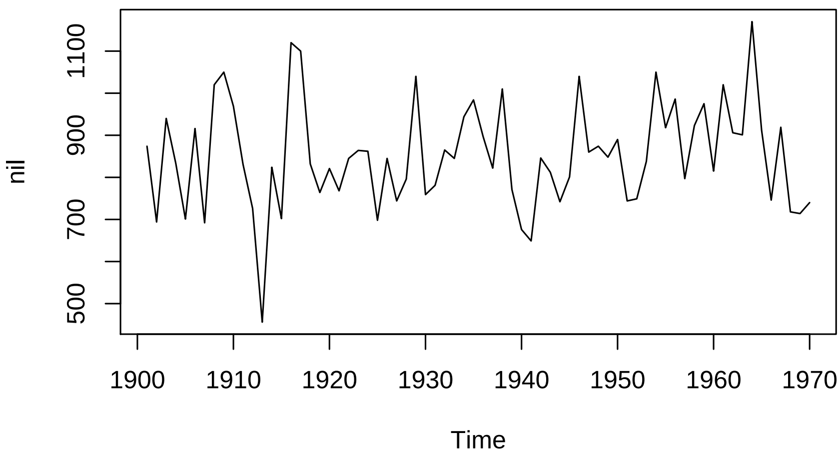

Figure 2: Zeitreihenplot der Nil-Daten.

```
> future <- window(nil, start = 1961)
```

Für den Glättungsparameter nehmen wir den Wert 0.2 an.

Die Argumente in der `R`-Funktion `HoltWinters` sind die Daten der Jahre 1901-1960: `past` und der Glättungsparameter `alpha=0.2`. Um das exponentielle Glätten auszuwählen, muss der Wert der Parameter `beta` und `gamma` auf `FALSE` gesetzt werden!

Der Output wird hier unter dem Namen `model` gespeichert.

```
> model <- HoltWinters(past, alpha = 0.2, beta = FALSE, gamma = FALSE)
```

Nachdem die Modellparameter mit der Funktion `HoltWinters` berechnet wurden, können die Werte für den Prognosezeitraum über die Funktion `predict` automatisch berechnet werden. Dabei muss die Anzahl der zu prognostizierenden Zeitpunkte (in unserem Beispiel 15) im Argument `n.ahead = 15` angegeben werden. Die Vorhersage für 1961-1975 wird hier im Parameter `progn` gespeichert:

```
> progn <- predict(model, n.ahead = 15)
```

In Fig. 3 wird mit dem Befehl `plot(model)` die ZR gemeinsam mit der geglätteten ZR für die Jahre 1901-1960 gezeichnet. Soll auch das prognostizierte Niveau für 1961-1975 eingezeichnet werden, muss im Argument auch die Option `predicted.values = progn` dazugeschrieben werden. Schließlich können mit `lines(future)` auch die Originaldaten der Jahre 1961-1970 dazugezeichnet werden.

Ergebnis:

Im Help-Menü zur Funktion HoltWinters findet man unter der Überschrift `Value` den Output der Funktion beschrieben. Diese Outputparameter und ihre Struktur, die im Objekt `model` gespeichert wurden, können auch angezeigt werden:

```
> str(model)
```

```
> plot(model, predicted.values = progn)
> lines(future)
```

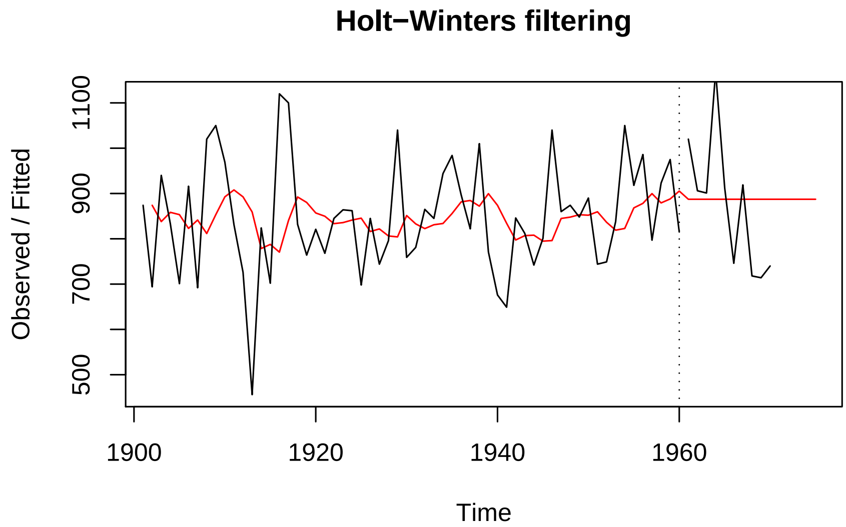

Figure 3: Die Nil-Daten, ihre Glättung und die Prognose für 1961-1975.

```
List of 9
 $ fitted      : mts [1:59, 1:2] 874 838 858 853 823 ...
  ..- attr(*, "dimnames")=List of 2
  .. ..$ : NULL
  .. ..$ : chr [1:2] "xhat" "level"
  ..- attr(*, "tsp")= num [1:3] 1902 1960 1
  ..- attr(*, "class")= chr [1:2] "mts" "ts"
 $ x           : Time-Series [1:60] from 1901 to 1960: 874 694 940 833 701 916 692 1020 1050 969
 $ alpha       : num 0.2
 $ beta        : logi FALSE
 $ gamma       : logi FALSE
 $ coefficients: Named num 887
  ..- attr(*, "names")= chr "a"
 $ seasonal    : chr "additive"
 $ SSE         : num 1030706
 $ call        : language HoltWinters(x = past, alpha = 0.2, beta = FALSE, gamma = FALSE)
 - attr(*, "class")= chr "HoltWinters"
NULL
```

Der Zugriff auf die einzelnen Elemente des Objekts `model` erfolgt immer unter Angabe des Objekts. So werden z.B. mit

```
> model$fitted[50:59, ]
```

```
          xhat    level
 [1,] 859.6619 859.6619
 [2,] 836.5295 836.5295
 [3,] 819.0236 819.0236
 [4,] 822.8189 822.8189
```

```
 [5,] 868.2551 868.2551
 [6,] 878.2041 878.2041
 [7,] 899.7633 899.7633
 [8,] 879.2106 879.2106
 [9,] 887.9685 887.9685
[10,] 905.3748 905.3748
```

spaltenweise die ZR der geglätteten Werte: `xhat` und die Niveaukomponente `level` für die letzten 10 Jahre, also für 1951-1960, ausgegeben. Mit

```
> model$SSE
```

```
[1] 1030706
```

kann die Fehlerquadratsumme zwischen den wahren Werten der ZR und den geglätteten Werten im Modellierungszeitraum 1901-1960 ausgegeben werden.

Wie wir aus den Folien der LV wissen, erfolgt die h-Schritt-Prognose vom Jahr 1960 ausgehend nach

$$\hat{x}_{1960}(h) = \hat{a}_{1961}, \quad h = 1, 2, \ldots, 15.$$

Der Wert des konstanten Prognoseniveaus $\hat{a}_{1961}$ steht in

```
> model$coeff
```

```
       a
887.2998
```

Die Prognose für die Jahre 1961-1975 beträgt daher 887.2998.

Genau diese Prognose wurde im Schritt 3 auch in `progn` gespeichert. Dort sind die prognostizierten Werte für 1961-1975 in der Spalte `fit`, zu sehen.

```
> progn
```

```
Time Series:
Start = 1961
End = 1975
Frequency = 1
           fit
 [1,] 887.2998
 [2,] 887.2998
 [3,] 887.2998
 [4,] 887.2998
 [5,] 887.2998
 [6,] 887.2998
 [7,] 887.2998
 [8,] 887.2998
 [9,] 887.2998
[10,] 887.2998
[11,] 887.2998
[12,] 887.2998
[13,] 887.2998
[14,] 887.2998
[15,] 887.2998
```

4. Schritt: Wahl des Glättungsparameters im Beispiel Nil-Daten:

In obigem Beispiel wurden die Nil-Daten mit exponentiellem Glätten und Glättungsparameter $\alpha = 0.2$ geglättet. Üblicherweise wird α so gewählt, dass die Fehlerquadratsumme $SSE(\alpha)$ möglichst klein wird (vgl. Folien der LV).

Im folgenden modellieren wir die Daten mit $\alpha = 0.3$:

```
> model2 <- HoltWinters(past, alpha = 0.3, beta = FALSE, gamma = FALSE)
```

Diesmal beträgt die mittlere Fehlerquadratsumme

```
> model2$SSE
[1] 1081658
```

Dieser Wert ist größer als jener Wert, der für $\alpha = 0.2$ errechnet wurde. Daher würde man $\alpha = 0.2$ einem Glättungsparameter $\alpha = 0.3$ vorziehen.

Die Funktion `HoltWinters` bietet auch die Möglichkeit den optimalen Glättungsparameter automatisch zu berechnen. Dazu wird im Argument kein Wert für `alpha` eingesetzt:

```
> modelopt <- HoltWinters(past, beta = FALSE, gamma = FALSE)
```

Der optimale Wert für α beträgt:

```
> modelopt$alpha
[1] 0.01377027
```

und die minimale mittlere Fehlerquadratsumme beträgt:

```
> modelopt$SSE
[1] 928741.9
```