Arbeitsblatt 5

Multiple lineare Regression (Vielfalt)

Aufgabe 1: Einkommen

Der Datensatz `salary.dat` enthält die Zielvariable y, die den Jahreslohn eines Angestellten in einer Firma beschreibt. Als kontinuierliche erklärende Variable dienen die Variable `experience`, welche die Berufserfahrung (in Anzahl Jahren) enthält und die kategorielle Variable `education` mit den Ausprägungen 1 = Berufslehre, 2 = Maturität und 3 = Hochschulabschluss.

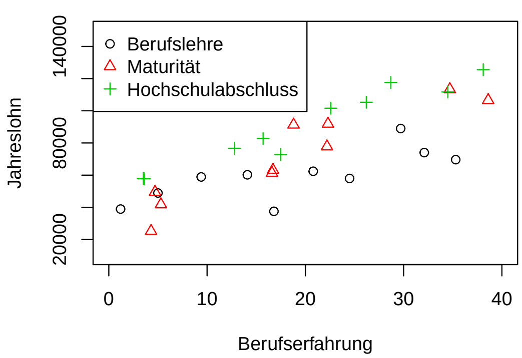
Einlesen der Daten

```
salary <- read.csv("data/salary.csv", header=TRUE)
```

(a)

Verschaffen Sie sich einen Überblick über die Daten. Erstellen Sie hierfür ein Streudiagramm mit Jahreslohn gegen Berufserfahrung und färben Sie die Punkte in Abhängigkeit der Ausbildung unterschiedlich ein. Kommentieren Sie den Plot.

```
plot(salary$experience, salary$y,
     col = salary$education, pch = salary$education,
     xlim = c(0,40), ylim = c(10000,150000),
     ylab = "Jahreslohn", xlab = "Berufserfahrung")
legend("topleft", c("Berufslehre","Maturität", "Hochschulabschluss"),
       col = 1:3, pch = 1:3)
```



Kommentar: Je mehr Berufserfahrung und je höher der Abschluss (mit Berufslehre < Maturität < Hochschulabschluss), desto höher ist das Einkommen.

(b)

Passen Sie ein multiples lineares Modell mit den erklärenden Variablen `experience` und `education` an. Stellen Sie dabei sicher, dass die Variable `education` eine Faktorvariable ist. Welche und wie viele Dummy-Variablen werden von R zur Modellierung der Variable `education` verwendet?

Zuerst stellen wir sicher, dass die Variable `education` in R richtig kodiert ist:

```
salary$education <- as.factor(salary$education)
class(salary$education)
```

```
[1] "factor"
```

Anpassung des Modells:

```
salary.fit <- lm(y ~ experience + education, data = salary)
```

Die Dummy-Variablen sind wie folgt:

$$d_i^{(1)} = \texttt{education2}_i = \left\{ \begin{array}{l} 1, \text{falls } \texttt{education}_i = \text{Maturität} \\ 0, \text{sonst} \end{array} \right\}$$

$$d_i^{(2)} = \texttt{education3}_i = \left\{ \begin{array}{l} 1, \text{falls } \texttt{education}_i = \text{Hochschulabschluss} \\ 0, \text{sonst} \end{array} \right\}$$

Die Anzahl der Dummy-Variablen lässt sich als "Anzahl Levels" - 1 bestimmen:

```
nlevels(salary$education) - 1
```

```
[1] 2
```

Es braucht also 2 Dummy-Variablen für `Education`.

(c)

Geben Sie die angepassten Gleichungen für Berufslehre, Maturität und Hochschulabschluss an.

Die geschätzten Koeffizieten sind:

```
coef(salary.fit)
```

```
(Intercept)  experience  education2  education3
  26320.897    1769.402   13273.719   28709.555
```

```
# Oder "einfacher" dargestellt mit
dummy.coef(salary.fit)
```

```
Full coefficients are

(Intercept):     26320.9
experience:     1769.402
education:             1        2        3
                    0.00 13273.72 28709.56
```

Damit lauten die drei geschätzten Geradengleichungen:

Berufslehre: $y = 26320.897 + 1769.402 \cdot \texttt{experience}$

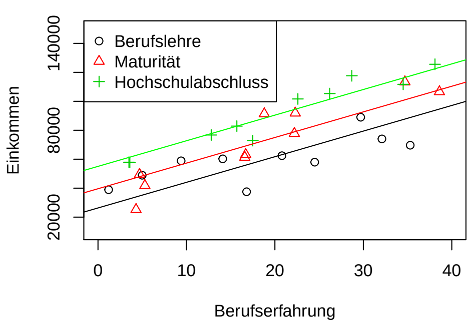
Maturität: $y = (26320.897 + 13273.719) + 1769.402 \cdot \texttt{experience} = 39594.62 + 1769.402 \cdot \texttt{experience}$

Hochschulabschluss: $y = (26320.897 + 28709.555) + 1769.402 \cdot \texttt{experience} = 55030.45 + 1769.402 \cdot \texttt{experience}$

(d)

Zeichnen Sie das geschätzte Modell ins Streudiagramm aus Teilaufgabe (a) ein.

```
plot(salary$experience, salary$y,
     col = as.numeric(salary$education),
     pch = as.numeric(salary$education),
     xlim = c(0,40), ylim = c(10000,150000),
     ylab = "Einkommen", xlab = "Berufserfahrung")
legend("topleft", c("Berufslehre","Maturität","Hochschulabschluss"),
       col = 1:3, pch = 1:3)
abline(a = coef(salary.fit)["(Intercept)"],
       b = coef(salary.fit)["experience"])
abline(a = coef(salary.fit)["(Intercept)"] + coef(salary.fit)["education2"],
       b = coef(salary.fit)["experience"], col = "red")
abline(a = coef(salary.fit)["(Intercept)"] + coef(salary.fit)["education3"],
       b = coef(salary.fit)["experience"], col = "green")
```



(e)

Haben die erklärenden Variablen einen auf 5% signifikanten Einfluss? Beurteilen Sie die Güte des Modells mit R^2.

Die letzte Zeile von `summary(salary.fit)` zeigt den globalen F-Test:

```
F-statistic: 42.89 on 3 and 26 DF, p-value:  3.29e-10
```

Der globale F-Test sagt, dass mindestens eine erklärende Variable einen auf 5% signifikanten Einfluss zur Erklärung des Einkommens beiträgt, da der p-Wert kleiner als 0.05 ist. Da das Modell kategorielle Variablen beinhaltet, testen wir die Signifikanz der einzelnen erklärenden Variablen mittels F-Test:

```
drop1(salary.fit, test = "F")
```

```
Single term deletions

Model:
y ~ experience + education
```

```
           Df  Sum of Sq        RSS    AIC F value    Pr(>F)
<none>                    3.3796e+09 564.19
experience  1 1.1779e+10 1.5159e+10 607.22  90.618 5.854e-10 ***
education   2 4.1169e+09 7.4965e+09 584.10  15.836 3.178e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Beide F-Tests zu den Koeffizienten haben einen p-Wert kleiner als 0.05 und man kann daraus schliessen, dass beide Variablen einen nachweisbaren Einfluss auf das Einkommen haben.

R^2 ist 0.81. Der Erklärungsanteil des Modells ist recht gut.

(f)

Was ist der Unterschied im Jahreslohn zwischen einer Person mit 20 Jahren Berufserfahrung und Berufslehre und einer Person mit ebenfalls 20 Jahren Berufserfahrung und einem Hochschulabschluss? Ist der Unterschied auf 5% signifikant?

Der Unterschied ist durch den Koeffizient der Dummyvariable education3 gegeben:

```r
coef(salary.fit)["education3"]
```

```
education3
  28709.56
```

Da der t-Test zur Dummy-Variable `education3` (Teststatistik: 5.623, p-Wert: 6.54e-06 - siehe in `summary(salary.fit)` Zeile zu `education3`) auf dem 5% Niveau signifikant ist, ist der Unterschied im Jahreslohn signifikant.

Aufgabe 2: Preis von Diamenten

Die wesentlichen Einflussfaktoren, die den Preis von Diamanten bestimmen, sind das Gewicht, die Reinheit, die Farbe und der Schliff. Im Englischen spricht man von den vier C's: Carat, Clarity, Colour and Cut. Diesem Sachverhalt wollen wir anhand von 307 runden Diamantsteinen nachgehen, die am 18. Februar 2000 in Singapur gehandelt wurden. Die Preisangaben sind in Singapur-Dollars und das Gewicht in Karat (ein Karat entspricht 0.2 Gramm). Die Daten dazu sind in `diamant.dat` gespeichert.

Einlesen der Daten

```r
FourC <- read.table("data/diamant.dat", header=TRUE)
```

(a)

Als erstes betrachten wir nur den Einfluss des Gewichts auf den Preis. Logarithmieren Sie die beiden Variablen `Carat` und `Preis` und passen ein quadratisches Polynom an, d.h.

$$\log(\text{Price}_i) = \beta_0 + \beta_1 \log(\text{Carat}_i) + \beta_2 (\log(\text{Carat}_i))^2 + E_i$$

Wie lautet das "rücktransformierte" Modell, d.h. $\text{Price}_i = \ldots$? Weshalb wird von Beginn an mit Logarithmus transformiert?

Das Modell lautet:

$$
\begin{aligned}
\log(\text{Price}_i) &= \beta_0 + \beta_1 \log(\text{Carat}_i) + \beta_2(\log(\text{Carat}_i))^2 + E_i \\
&\Leftrightarrow \\
\text{Price}_i &= e^{\beta_0} \cdot e^{\beta_1 \log(\text{Carat}_i)} \cdot e^{(\beta_2(\log(\text{Carat}_i))^2} \cdot e^{E_i} \\
&= e^{\beta_0} \cdot (\text{Carat}_i)^{\beta_1} \cdot e^{(\beta_2(\log(\text{Carat}_i))(\log(\text{Carat}_i)))} \cdot e_i^E \\
&= e^{\beta_0} \cdot (\text{Carat}_i)^{\beta_1} \cdot \text{Carat}_i^{(\beta_2(\log(\text{Carat}_i))} \cdot e_i^E
\end{aligned}
$$

Wir logarithmieren von Beginn an, weil es sich bei beiden relevanten Grössen um Beträge (d.h. um positive Intervallvariablen) handelt. In den meisten Fällen führt eine Logarithmus-Transformation zu einem geeigneten Regressionsmodell.

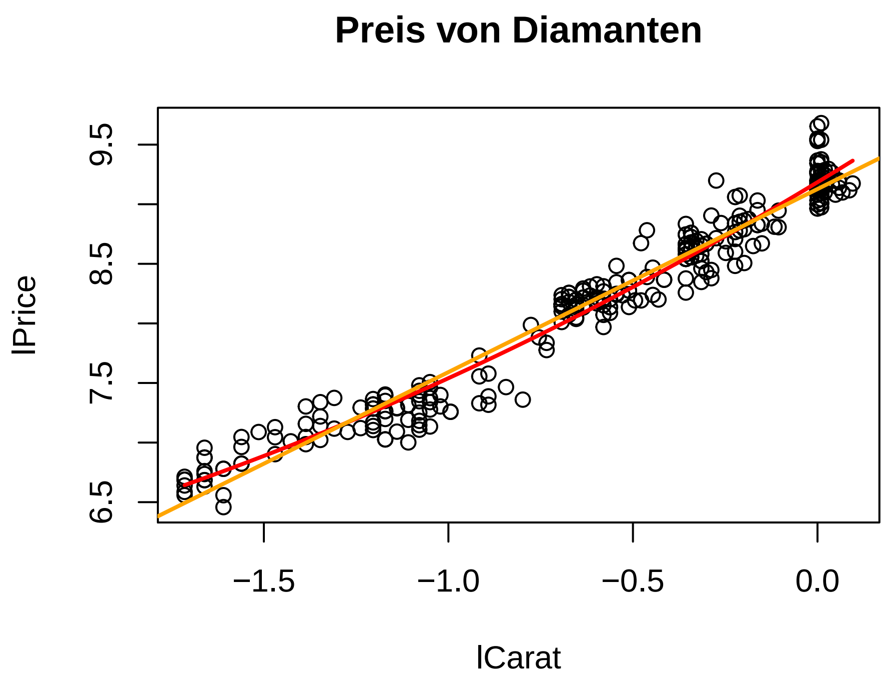
```r
# Transformationen
FourC$lCarat <- log(FourC$Carat)
FourC$lCaratH2 <- log(FourC$Carat)^2
FourC$lPrice <- log(FourC$Price)

## Regressionsfit
fit.Diamant1 <- lm(lPrice ~ lCarat + lCaratH2, data = FourC)
coef(fit.Diamant1)
```

```
(Intercept)      lCarat    lCaratH2
   9.184821    1.871764    0.227029
```

(b)

Erstellen Sie ein Streudiagramm mit dem angepassten Modell.

```r
plot(lPrice ~ lCarat, data = FourC, main = "Preis von Diamanten")
# Einzeichnen des angepassten quadr. Polynoms
x <- seq(min(FourC$lCarat), max(FourC$lCarat), length=50)
dat <- data.frame(lCarat=x, lCaratH2=x^2)
lines(x, predict(fit.Diamant1, newdata = dat), col=2, lwd = 2)
# Als Vergleich kann man noch die Regressionsgeraden
# ohne den quadratischen Term einzeichen, um den
# quadratischen Effekt zu verdeutlichen.
FourC.lmE <- lm(lPrice ~ lCarat, data=FourC)
abline(FourC.lmE , col="orange", lwd = 2)
```



Der Unterschied dieser beiden Kurven ist bezüglich der Streuung in den Beobachtungen zwar klein, aber immer noch statistisch nachweisbar (siehe Teilaufgabe (c)).

(c)

Ist der quadratische Term wesentlich zur Modellierung dieser Daten?

Ob der quadratische Term wesentlich ist, testen wir mit dem $H_0 : \beta_2 = 0$ gegen $H_A : \beta_2 \neq 0$ auf dem 5%-Niveau. Die Teststatistik mit p-Wert entnehmen wir aus `summary(fit.Diamant1)`:

```
    Estimate   Std. Error      t value     Pr(>|t|)
2.270290e-01 3.560582e-02 6.376177e+00 6.736756e-10
```

Da der p-Wert kleiner als das Niveau 5% ist, wird die Null-Hypothese verworfen. Der quadratische Teil ist deshalb statistisch wesentlich, um diese Daten zu modellieren. Damit wächst der Preis nachweisbar stärker als ein Potenzgesetz.

(d)

Passen Sie nun ein etwas komplexeres Regressionsmodell an, in dem Sie `Colour`, `Clarity` und `CBody` zum quadratischen Polynom hinzufügen. Um wie viele Parameter wird das Modell erweitert? Bringen die zusätzlichen Variablen eine Verbesserung?

Die zusätzlichen drei erklärenden Variablen sind kategorielle Variablen und werden in Form von binären Variablen im Modell aufgenommen. Aus der Anzahl Levels können wir bestimmen, wie viele binären Variablen nötig sind:

```
nlevels(FourC$Colour)
```

```
[1] 6
```

```r
nlevels(FourC$Clarity)
```

```
[1] 5
```

```r
nlevels(FourC$CBody)
```

```
[1] 3
```

Es braucht 5 Dummy-Variablen für `Colour`, 4 Dummy-Variablen für `Clarity` und 2 Dummy-Variablen für `CBody`. Insgesamt wird das Modell also um 11 Parameter erweitert.

```r
## Anpassen der Regression
fit.Diamant2 <- lm(lPrice ~ lCarat + lCaratH2 + Colour + Clarity + CBody,
                   data = FourC)
drop1(fit.Diamant2, test="F") # Signifikanz der Variablen
```

```
Single term deletions

Model:
lPrice ~ lCarat + lCaratH2 + Colour + Clarity + CBody
         Df Sum of Sq     RSS     AIC  F value Pr(>F)
<none>                 0.9321 -1751.7
lCarat    1   28.0095 28.9416  -699.0 8804.437 <2e-16 ***
lCaratH2  1    0.7440  1.6761 -1573.6  233.858 <2e-16 ***
Colour    5    4.7347  5.6668 -1207.6  297.657 <2e-16 ***
Clarity   4    2.3446  3.2767 -1373.8  184.251 <2e-16 ***
CBody     2    0.0073  0.9394 -1753.3    1.141 0.3209
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
## Fehlervarianzen
summary(fit.Diamant1)$sigma
```

```
[1] 0.1585453
```

```r
summary(fit.Diamant2)$sigma
```

```
[1] 0.05640299
```

```r
## R-Squared
summary(fit.Diamant1)$adj.r.squared
```

```
[1] 0.9620033
```

```r
summary(fit.Diamant2)$adj.r.squared
```

```
[1] 0.9951911
```

Bis auf CBody sind alle Variablen auf dem 5% Niveau statistisch notwendig. Die geschätzte Standardabweichung σ ist beim komplexeren Modell fast um den Faktor 3 kleiner - eine deutliche Verbesserung. Auch ist der Anteil der erklärten Schwankung im Preis etwas höher, wobei das R^2 im kleineren Modell schon sehr gut war.

(e)

Ein Käufer möchte einen runden 0.4 karätigen Diamanten mit GIA Zertifikat von Reinheit "IF" kaufen. Bei der Farbe ist er sich noch etwas unschlüssig zwischen "D" und "E". Geben Sie ihm, ein

95%-Prognoseintervall für den jeweiligen Preis.

```
# Datenpunkte für Vorhersage
x0 <- data.frame(lCarat = log(c(0.4,0.4)),
                 lCaratH2= log(c(0.4,0.4))^2,
                 Colour = c("D","E"),
                 Clarity = c("IF","IF"),
                 CBody = c("GIA","GIA"))
h <- predict(fit.Diamant2, newdata = x0, interval = "prediction")
h # Achtung, dies ist der logarithmierte Preis!
```

```
       fit      lwr      upr
1 8.072965 7.955505 8.190425
2 7.972762 7.858172 8.087352
```

```
# Rücktransformation
exp(h)
```

```
       fit      lwr      upr
1 3206.595 2851.226 3606.255
2 2900.858 2586.788 3253.061
```

Zu beachten ist, dass die Punkt-Vorhersage nicht der Erwartungswert der Zielgrösse ist, sondern nur deren Median (da Log-Response). Wenn wir den erwarteten Wert vorhersagen wollen, so müssen wir noch speziell rücktransformieren.

```
exp(h[1,1] + (summary(fit.Diamant2)$sigma^2)/2) # Farbe D
```

```
[1] 3211.699
```

```
exp(h[2,1] + (summary(fit.Diamant2)$sigma^2)/2) # Farbe E
```

```
[1] 2905.476
```

Der erwartete Preis für den Diamanten der Farbe D liegt bei 3211.70 Singapur Dollar. Mit 95% liegt dieser zwischen 2851.23 und 3606.26 Singapur Dollar. Für den Diamanten der Farbe E liegt der erwartete Preis bei 2905.476 Singapur Dollar und mit 95% zwischen 2586.79 und 3253.06.