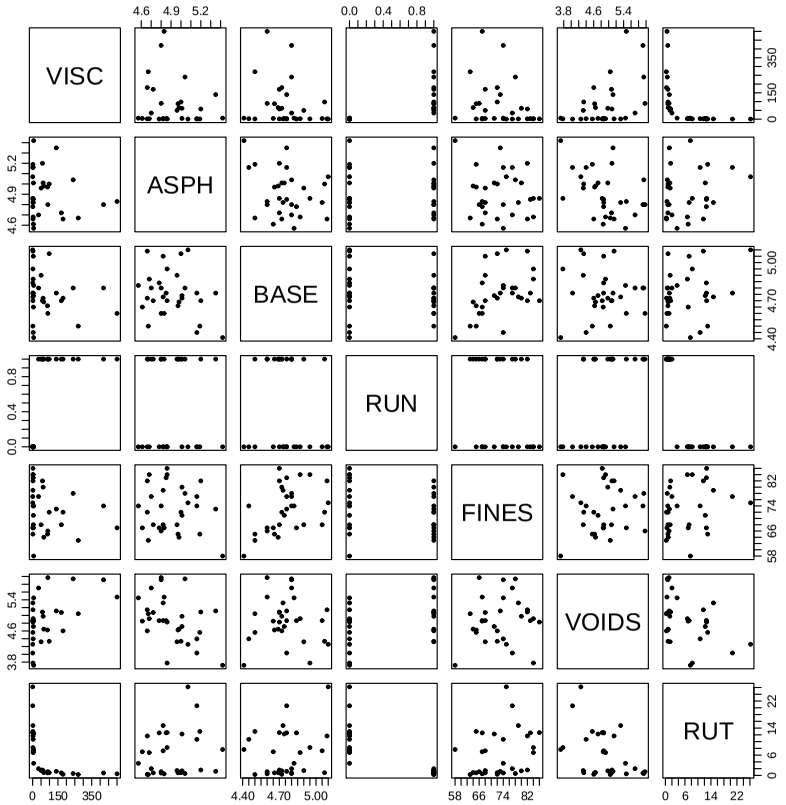
Lösungsskizze Serie 4

1. a) In der Scatterplot-Matrix fällt auf, dass `RUT` nicht linear von `VISC` abhängt. Werden diese Variablen logarithmiert, so ist der Zusammenhang linear. Alle anderen Variabeln, braucht man nicht zu transformieren.

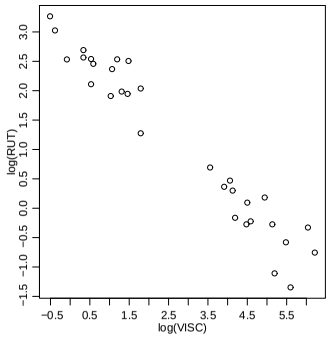


b) Schrittweise rückwärts:

Wir betrachten das volle Modell

$$\log(\texttt{RUT}) = \beta_0 + \beta_1 \log(\texttt{VISC}) + \beta_2\ \texttt{ASPH} + \beta_3\ \texttt{BASE} + \beta_4\ \texttt{FINES} + \beta_5\ \texttt{VOIDS} + \beta_6\ \texttt{RUN}$$

und eliminieren schrittweise die am wenigsten signifikante Variable (d.h diejenige Variable, nach deren Weglassen das verbleibende Modell den kleinsten AIC-Wert aufweist).




```
> asphalt.lm <- lm(log(RUT)~log(VISC)+ASPH+BASE+FINES+VOIDS+RUN, data=asphalt)
> summary(step(asphalt.lm, direction="backward"))

Start:  AIC= -77.35
 log(RUT) ~ log(VISC) + ASPH + BASE + FINES + VOIDS + RUN

            Df Sum of Sq     RSS     AIC
- FINES      1     0.021   1.648 -78.965
- BASE       1     0.034   1.662 -78.705
<none>                     1.628 -77.355
- RUN        1     0.300   1.927 -74.115
- VOIDS      1     0.559   2.186 -70.208
- ASPH       1     1.265   2.892 -61.530
- log(VISC)  1     3.348   4.976 -44.713

Step:  AIC= -78.96
 log(RUT) ~ log(VISC) + ASPH + BASE + VOIDS + RUN

            Df Sum of Sq     RSS     AIC
- BASE       1     0.065   1.713 -79.768
<none>                     1.648 -78.965
- RUN        1     0.288   1.936 -75.967
- VOIDS      1     0.692   2.340 -70.098
- ASPH       1     1.293   2.941 -63.016
- log(VISC)  1     3.629   5.277 -44.887

Step:  AIC= -79.77
 log(RUT) ~ log(VISC) + ASPH + VOIDS + RUN

            Df Sum of Sq     RSS     AIC
<none>                     1.713 -79.768
- RUN        1     0.231   1.944 -77.849
- VOIDS      1     0.675   2.388 -71.470
- ASPH       1     1.245   2.958 -64.836
- log(VISC)  1     4.608   6.321 -41.294

Call:
lm(formula = log(RUT) ~ log(VISC) + ASPH + VOIDS + RUN, data = asphalt)

Residuals:
     Min       1Q   Median       3Q      Max
-0.42912 -0.15258 -0.01832  0.17435  0.40451

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -4.01062    1.49034  -2.691 0.012284 *
log(VISC)   -0.54747    0.06547  -8.363 7.62e-09 ***
```

```
ASPH         1.07061    0.24631   4.347 0.000188 ***
VOIDS        0.33089    0.10338   3.201 0.003597 **
RUN         -0.51199    0.27355  -1.872 0.072545 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.2567 on 26 degrees of freedom
Multiple R-Squared: 0.9708,     Adjusted R-squared: 0.9663
F-statistic:   216 on 4 and 26 degrees of freedom,      p-value:     0
```

Das Endmodell lautet mit schrittweiser Variablenselektion rückwärts:

$$\log(\texttt{RUT}) = \beta_0 + \beta_1 \log(\texttt{VISC}) + \beta_2\, \texttt{ASPH} + \beta_3\, \texttt{VOIDS} + \beta_4\, \texttt{RUN}.$$

Schrittweise vorwärts:

Wir betrachten das Modell $\log(\texttt{RUT}) = \beta_0$ und fügen schrittweise die signifikanteste der verbleibenden Variable hinzu (d.h. die Variable mit dem kleinsten AIC).

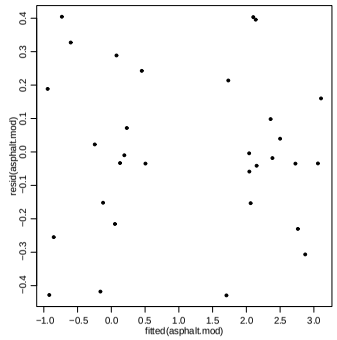
```
> start.lm <- lm(log(RUT)~1, asphalt)
> summary(step(start.lm,scope=list(upper=log(RUT)~log(VISC)+ASPH+BASE+
                  FINES+VOIDS+RUN), direction="forward"))
Start:  AIC= 21.76
 log(RUT) ~ 1

           Df Sum of Sq     RSS     AIC
+ log(VISC)  1    55.423   3.210 -66.302
+ RUN        1    50.565   8.067 -37.732
+ VOIDS      1     9.865  48.767  18.045
+ FINES      1     6.021  52.612  20.397
<none>                    58.632  21.756
+ BASE       1     2.095  56.537  22.628
+ ASPH       1     1.399  57.233  23.008

Step:  AIC= -66.3
 log(RUT) ~ log(VISC)

        Df Sum of Sq     RSS     AIC
+ ASPH   1     0.566   2.643 -70.321
<none>                 3.210 -66.302
+ VOIDS  1     0.158   3.052 -65.868
+ RUN    1     0.122   3.087 -65.508
+ FINES  1     0.045   3.165 -64.737
+ BASE   1     0.029   3.180 -64.587

Step:  AIC= -70.32
 log(RUT) ~ log(VISC) + ASPH

        Df Sum of Sq     RSS     AIC
+ VOIDS  1     0.699   1.944 -77.849
+ RUN    1     0.255   2.388 -71.470
<none>                 2.643 -70.321
+ FINES  1     0.115   2.529 -69.697
+ BASE   1     0.002   2.641 -68.343

Step:  AIC= -77.85
 log(RUT) ~ log(VISC) + ASPH + VOIDS

        Df Sum of Sq     RSS     AIC
+ RUN    1     0.231   1.713 -79.768
<none>                 1.944 -77.849
+ FINES  1     0.015   1.929 -76.089
+ BASE   1     0.007   1.936 -75.967
```



```
Step:  AIC= -79.77
 log(RUT) ~ log(VISC) + ASPH + VOIDS + RUN

        Df Sum of Sq     RSS     AIC
<none>                 1.713 -79.768
+ BASE   1     0.065   1.648 -78.965
+ FINES  1     0.051   1.662 -78.705

Call:
lm(formula = log(RUT) ~ log(VISC) + ASPH + VOIDS + RUN, data = asphalt)

Residuals:
     Min       1Q   Median       3Q      Max
-0.42912 -0.15258 -0.01832  0.17435  0.40451

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -4.01062    1.49034  -2.691 0.012284 *
log(VISC)   -0.54747    0.06547  -8.363 7.62e-09 ***
ASPH         1.07061    0.24631   4.347 0.000188 ***
VOIDS        0.33089    0.10338   3.201 0.003597 **
RUN         -0.51199    0.27355  -1.872 0.072545 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.2567 on 26 degrees of freedom
Multiple R-Squared: 0.9708,     Adjusted R-squared: 0.9663
F-statistic:   216 on 4 and 26 degrees of freedom,      p-value:     0
```

Das Resultat ist identisch mit demjenigen des Backward-Verfahrens.

2. **a)** Der Tukey-Anscombe Plot zeigt nichts unerwünschtes. Die Varianz der Fehler kann als konstant angesehen werden.

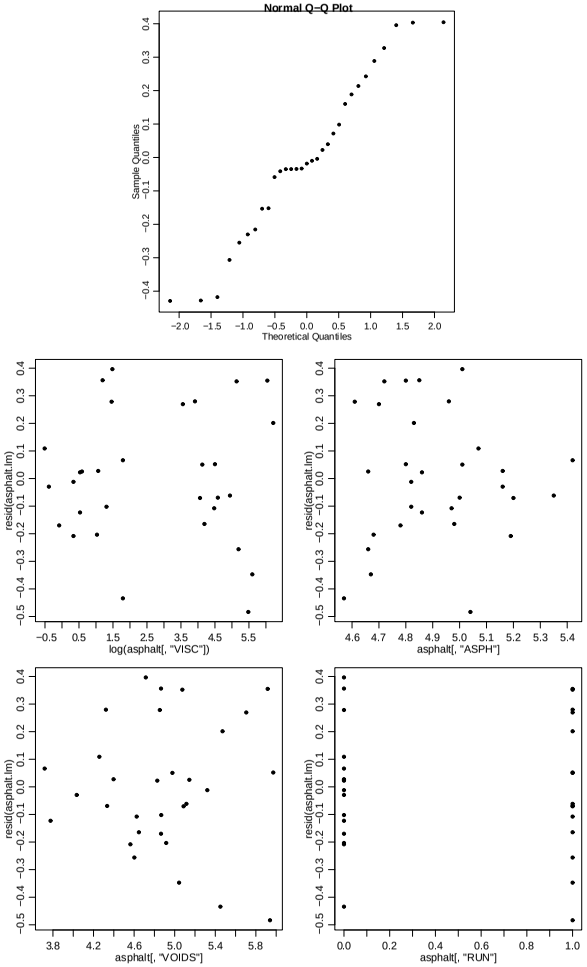


b) Die Fehler scheinen nicht normalverteilt zu sein. Diese leichte Inhomogenität kann aber rein daher resultieren, dass die Anzahl der Daten ($n = 31$) klein ist. Der Q-Q-Plot wird daher als 'in Ordnung' betrachtet.

c) Die Fehler sind mit keiner erklärenden Variablen korreliert, d.h. die Grafiken zeigen keine unerwünschte Struktur.

Das Modell scheint die Daten gut zu beschreiben.

R-Code:




```
> asphalt.mod <- lm(log(RUT)~log(VISC)+ASPH+VOIDS+RUN, data=asphalt)
> plot(fitted(asphalt.mod), resid(asphalt.mod))
> qqnorm(resid(asphalt.mod))
> par(mfrow=c(2,2))
> plot(log(asphalt[,'VISC']), resid(asphalt.lm))
> plot(asphalt[,'ASPH'], resid(asphalt.lm))
> plot(asphalt[,'VOIDS'], resid(asphalt.lm))
> plot(asphalt[,'RUN'], resid(asphalt.lm))
```

3. Daten einlesen:

```
> korrelation<-read.table.url(
        ''http://stat.ethz.ch/Teaching/Datasets/korrelation.dat'',header=T)
```

Die Matrix des linearen Modells berechnet man folgendermassen:


```
> eins<-rep(1,30)
> X.1 <- korrelation$X1-1
> X1<-cbind(eins,X.1)
> X.2 <- korrelation$X2-1
> X2<-cbind(eins,X.2)
> X.3 <- korrelation$X3-1
> X3<-cbind(eins,X.3)
```

Die Covarianzmatrix Σ bekommt man wie folgt:

```
> sigma<-0.8^(abs(outer(1:30,1:30,FUN=''-'')))
```

a) Im ersten Fall ist die Standardabweichung von $\widehat{\beta}$ gegeben durch: $\sqrt{((X^TX)^{-1})_{22}} \cdot 1$. Der R-Code ist:

```
> covX1.a <- sqrt(solve(t(X1)%*%X1)[2,2])          # für i),a)
> covX2.a <- sqrt(solve(t(X2)%*%X2)[2,2])          # für i),b)
> covX3.a <- sqrt(solve(t(X3)%*%X3)[2,2])          # für i),c)
```

Das Resultat ist:
i),a): 0.3651484, i),b): 0.3651484, i),c): 0.3651484
Da die Korrelation ignoriert wird und wir überall in allen drei Fällen die gleichen 30 Modellgleichungen haben (nur in einer anderen Reihenfolge), bekommen wir überall dasselbe Resultat.

b) Im zweiten Fall ist die Standardabweichung von $\widehat{\beta}$ gegeben durch: $\sqrt{((X^TX)^{-1}(X^T\Sigma X)(X^TX)^{-1})_{22}} \cdot 1$. Der R-Code ist:

```
> covX1.b <- sqrt((solve(t(X1)%*%X1)%*%(t(X1)%*%sigma
                    %*%X1)%*%solve(t(X1)%*%X1))[2,2])     # für ii),a)
> covX2.b <- sqrt((solve(t(X2)%*%X2)%*%(t(X2)%*%sigma
                    %*%X2)%*%solve(t(X2)%*%X2))[2,2])     # für ii),b)
> covX3.b <- sqrt((solve(t(X3)%*%X3)%*%(t(X3)%*%sigma
                    %*%X3)%*%solve(t(X3)%*%X3))[2,2])     # für ii),c)
```

Man bekommt:
ii),a): 0.8315447, ii),b): 0.4167435, ii),c): 0.1622576

c) Im letzten Fall ist die Standardabweichung von $\widehat{\beta}$ gegeben durch: $\sqrt{(X^T\Sigma^{-1}X)_{22}} \cdot 1$. Der R-Code ist:

```
> covX1.c <- sqrt(solve(t(X1)%*%solve(sigma)%*%X1)[2,2])     # für iii),a)
> covX2.c <- sqrt(solve(t(X2)%*%solve(sigma)%*%X2)[2,2])     # für iii),b)
> covX3.c <- sqrt(solve(t(X3)%*%solve(sigma)%*%X3)[2,2])     # für iii),c)
```

Das Resultat ist:
iii),a): 0.5523448, iii),b): 0.1900871, iii),c): 0.1415663

Zwei Dinge fallen auf:
1. Die verallgemeinerte kleinste Quadrate Methode, gibt genauere Schätzungen für $\widehat{\beta}$ als die Methode in c). Sie berücksichtigt also die Korrelation besser.

2. Die letzte Versuchsanordnung gibt (ausser mit a)) die besten Schätzungen. Grund: Durch die zufällige, aber trotzdem alternierende Wahl, der Gruppe wird die Möglichkeit der Korrelation vermindert.