

## Angewandte Regression — Musterlösungen zur Serie 6

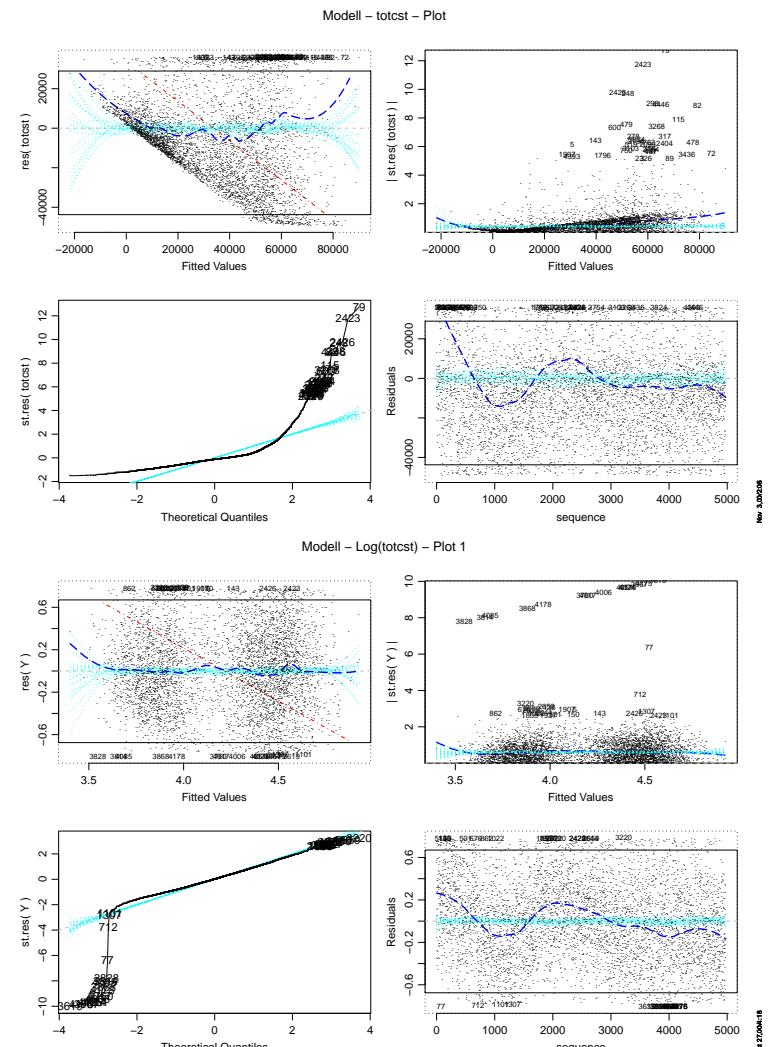
1. a) First-Aid-Transformation sind möglich.

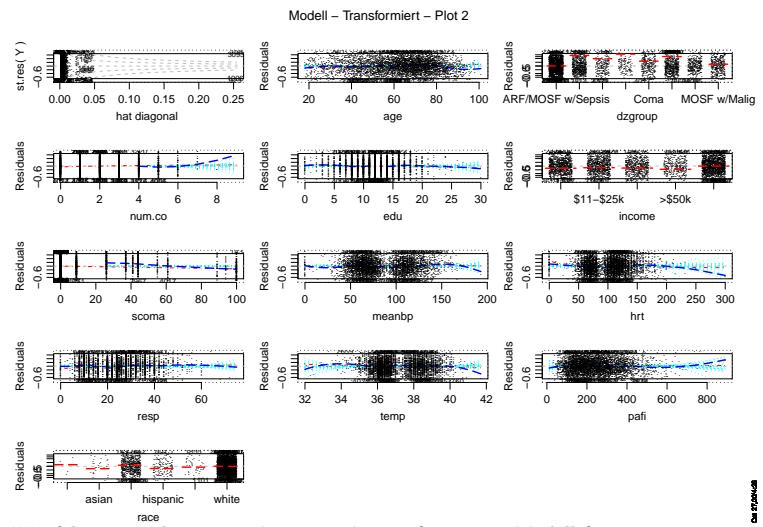
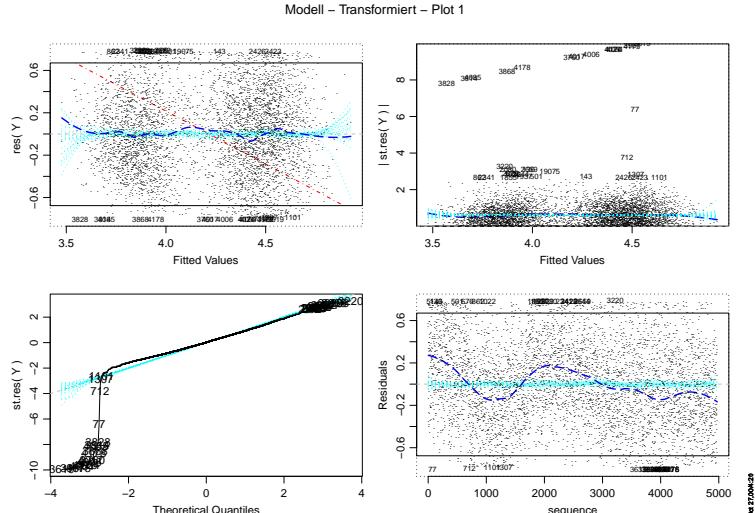
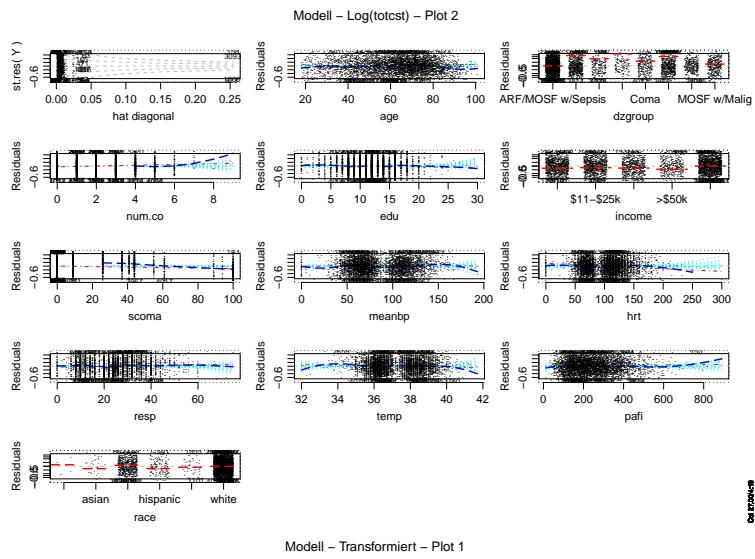
Variablen	Bedeutung	mögliche First-Aid-Transformationen
totcst	totale Kosten	$\log_{10}(\text{totcst}+1)$
age	Alter	
dzgroup	Krankheitsgruppe	
num.co	Anzahl von Komorbidität (Mehrfachdiagnose)	$\sqrt{\cdot}$
edu	Jahre der Ausbildung	
income	Einkommen	
scoma	Äquivalentes Mass für Glasgow-Koma-Wert	
meanbp	Mittelwert Blutdruck	$\log_{10}(\text{meanbp}+1)$
hrt	Puls	$\log_{10}(\text{hrt}+1)$
resp	Atemfrequenz	$\log_{10}(\text{resp}+1)$
temp	Temperatur	$\log_{10}(\text{temp}+1)$
race	Rasse	
pafi	Verhältnis PaO <sub>2</sub> /FiO <sub>2</sub> (Blut-Gasmischung)	$\log_{10}(\text{pafi}+1)$

R-Code:

```
.libPaths(c("/u/rsimon/R/Rlibrary/"))
library(regr0)
library(MASS)
d.supp <- read.table("http://stat.ethz.ch/Teaching/Datasets/WBL/support3.csv",
                      sep=",", header=T)
r.supp <- regr(totcst ~ age + dzgroup + num.co + edu + income
                + scoma + meanbp + hrt + resp + temp + pafi + race, data=d.supp)
summary(r.supp)
r.supp <- regr(log10(totcst+1) ~ age + dzgroup + sqrt(num.co) + edu + income
                + scoma + log10(meanbp+1) + log10(hrt+1) + log10(resp+1)
                + log10(temp+1) + log10(pafi+1) + race, data=d.supp)
summary(r.supp)
r.supp <- regr(log10(totcst+1) ~ age + dzgroup + num.co + edu + income
                + scoma + meanbp + hrt + resp + temp + pafi + race, data=d.supp)
summary(r.supp)
```

Eine Residuenanalyse zeigt, dass eine logarithmus-Transformation für die Zielvariable notwendig ist. Transformationen für die erklärenden Variablen sind nicht nötig.





Wir fahren mit dem  $\log_{10}(\text{totcst}+1)$ -transformierten Modell fort.

R-Output:

```
> Call:  
regr(formula = log10(totcst + 1) ~ age + dzgroup + num.co + edu +  
    income + scoma + meanbp + hrt + resp + temp + pafi + race,  
    data = d.suppl, na.action = na.exclude)
```

Fitting function lm

Terms:

	coef	stcoef	signif	R2.x	df	p.value	
(Intercept)	3.904796e+00	0.00000000	6.5390840	NA	1	0.0000	
age	-2.460941e-03	-0.06969102	-2.8279849	0.0824	1	0.0000	
dzgroup		NA	NA	35.5303505	0.0569	7	0.0000
num.co	-2.105147e-02	-0.05247788	-2.0374749	0.1220	1	0.0001	
edu	5.330446e-03	0.03259470	1.2930201	0.1030	1	0.0113	
income		NA	NA	7.2888103	0.0333	4	0.0000
scoma	1.365300e-03	0.05929679	2.1114530	0.1948	1	0.0000	
meanbp	-6.618862e-05	-0.00342917	-0.1483996	0.0214	1	0.7711	
hrt	1.436177e-03	0.08339005	3.4486921	0.0648	1	0.0000	
resp	-2.009884e-03	-0.03733652	-1.6114780	0.0240	1	0.0016	
temp	1.610870e-02	0.03740140	1.5555617	0.0595	1	0.0023	
pafi	-1.402376e-04	-0.02759589	-1.1654773	0.0450	1	0.0224	
race		NA	NA	3.7138886	0.0144	5	0.0001

Coefficients for factors:

\$dzgroup	ARF/MOSF w/Sepsis	CHF	Cirrhosis	Colon Cancer
-----------	-------------------	-----	-----------	--------------

```
0.00000000 -0.5563261 -0.3875634 -0.6413479
Coma          COPD      Lung Cancer    MOSF w/Malig
-0.2529050   -0.5067236 -0.7366444 -0.1042627
```

```
$income
$11-$25k $25-$50k >$50k under $11k
0.00000000 -0.02996724 -0.00998261 0.05811613 -0.11697248

$race
asian black hispanic other white
0.00000000 0.21044000 0.02305824 0.20159490 0.13189077 0.07902814
```

```
St.dev.error: 0.4563 on 4948 degrees of freedom
Multiple R^2: 0.3417 Adjusted R-squared: 0.3384
F-statistic: 102.7 on 25 and 4948 d.f., p.value: 0
```

b) R-Code:

```
#backward
r.supp <- regr(log10(totcst+1) ~ age + dzgroup + num.co + edu + income
                 + scoma + meanbp + hrt + resp + temp + pafi
                 + race, data=d.supp)
summary(r.supp)
```

```
r.supp1.step <- step(r.supp, direction = "backward")
r.supp1.step.formula <- formula(r.supp1.step)
r.supp1.final <- regr(r.supp1.step.formula, data=d.supp)
summary(r.supp1.final)
```

```
#forward
r.supp2 <- regr(log10(totcst+1) ~ 1, data=d.supp)
r.supp2.step <- step(r.supp2, scope=formula(r.supp), direction="forward")
r.supp2.final <- regr(r.supp2.step.formula, data=d.supp)
summary(r.supp2.final)
```

R-Output: Zusammengefasst erhalten wir für den *backward*

```
> Call:
regr(formula = r.supp1.step.formula, data = d.supp)
Fitting function lm
```

Terms:

	coef	stcoef	signif	R2.x	df	p.value
(Intercept)	3.8988894703	0.00000000	6.544350	NA	1	0.0000
age	-0.0024513095	-0.06941826	-2.825046	0.0798	1	0.0000
dzgroup	NA	NA	35.558519	0.0560	7	0.0000
num.co	-0.0210404285	-0.05245035	-2.036648	0.1220	1	0.0001
edu	0.0053421542	0.03266629	1.296218	0.1028	1	0.0111
income	NA	NA	7.296197	0.0332	4	0.0000
scoma	0.0013732541	0.05964224	2.131285	0.1920	1	0.0000
hrt	0.0014317651	0.08313388	3.447212	0.0624	1	0.0000
resp	-0.0020187937	-0.03750203	-1.620650	0.0229	1	0.0015
temp	0.0161199182	0.03742745	1.556831	0.0595	1	0.0023

```
pafi      -0.0001419996 -0.02794262 -1.186018 0.0403 1 0.0201
race        NA             NA             3.731986 0.0143 5 0.0001
```

Coefficients for factors:

\$dzgroup	ARF/MOSF w/Sepsis	CHF	Cirrhosis	Colon Cancer
	0.00000000	-0.5562096	-0.3874499	-0.6418558
Coma	0.25366555	-0.5068907	-0.7366678	-0.1039340
COPD				
Lung Cancer				
MOSF w/Malig				

```
$income
$11-$25k $25-$50k >$50k under $11k
0.00000000 -0.030063384 -0.009936136 0.058097576 -0.117062701
```

```
$race
asian black hispanic other white
0.00000000 0.21102422 0.02312897 0.20201848 0.13240204 0.07924477
```

```
St.dev.error: 0.4562 on 4949 degrees of freedom
Multiple R^2: 0.3417 Adjusted R-squared: 0.3385
F-statistic: 107 on 24 and 4949 d.f., p.value: 0
```

R-Output: Beim *forward* erhalten wir folgende Fehlerliste:

```
...
Step: AIC=-13412.42
log10(totcst + 1) ~ dzgroup + income + hrt + age + race + scoma +
num.co + temp + resp
```

Df	Sum of Sq	RSS	AIC
+ edu	1	1.38566	1031.3 -7778.1
+ pafi	1	1.16682	1031.5 -7777.1
<none>			1032.7 -7773.4
+ meanbp	1	0.06816	1032.6 -7771.8

Error in step(r.supp2, scope = formula(r.supp1), direction = "forward") :
number of rows in use has changed: remove missing values?

In den Helps finden wir eine allgemeine Warnung:

*Warning: The model fitting must apply the models to the same dataset. This may be a problem if there are missing values and R's default of 'na.action = na.omit' is used. We suggest you remove the missing values first.*

dh, wir müssten die Daten bereinigen, was wir hier aber nicht machen und auch nicht besprechen. Dazu müsste man einiges über die Daten wissen, um solche *Reinigung* durchzuführen zu können. Wir geben uns mit dem *backward* zu frieden.

- Lasso: R-Code:

```
r.lasso<-lasso(log10(totcst+1) ~ age + dzgroup + num.co + edu + income
                 + scoma + meanbp + hrt + resp + temp
                 + pafi + race, data=d.supp)
```

R-Output:

\* Number of observations: 4974

```
* Penalty parameter lambda:
      1.1      1.10      1.20
905.93833923 274.046348  2.264846

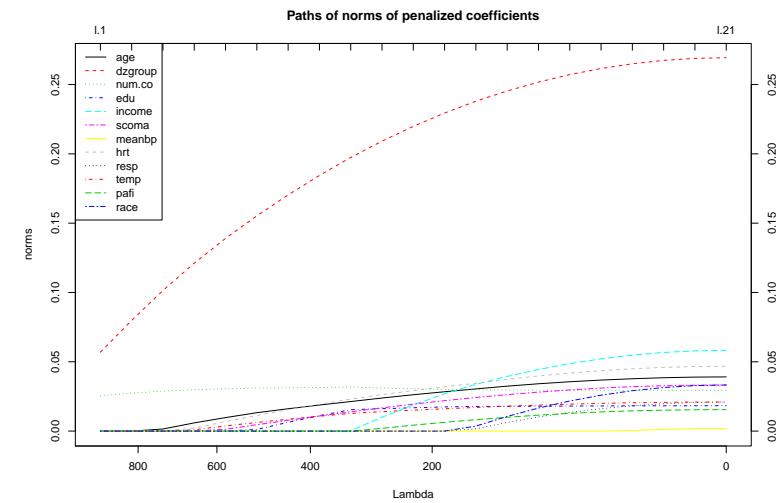
* Predictor groups : 13
* Penalized predictors :
[1] age      dzgroup num.co  edu      income   scoma    meanbp   hrt      resp
[10] temp     pafi    race
not penalized:
[1] (Intercept)

* Coefficients (*: p = penalized) :
            1.1      1.10      1.20
* lambda      905.93833923 2.740463e+02 2.264846e+00
(Intercept) 4.30441513 4.019428e+00 3.905574e+00
p age        0.00000000 -1.501908e-03 -2.451656e-03
p dzgroupCHF -0.12319560 -4.442281e-01 -5.554892e-01
p dzgroupCirrhosis -0.07636932 -2.929602e-01 -3.867234e-01
p dzgroupColon Cancer -0.12215270 -4.666770e-01 -6.398256e-01
p dzgroupComa -0.03759247 -1.686804e-01 -2.521645e-01
p dzgroupCOPD -0.11197140 -4.046780e-01 -5.059523e-01
p dzgroupLung Cancer -0.14346273 -5.489392e-01 -7.350585e-01
p dzgroupMOSF w/Malig -0.01623095 -7.025333e-02 -1.039534e-01
p num.co      -0.01829201 -2.222836e-02 -2.104861e-02
p edu        0.00000000 4.710914e-03 5.327635e-03
p income$11-$25k 0.00000000 -5.079782e-03 -2.975330e-02
p income$25-$50k 0.00000000 -3.458223e-05 -9.804076e-03
p income>$50k 0.00000000 1.007669e-02 5.773524e-02
p incomeunder $11k 0.00000000 -1.973644e-02 -1.161833e-01
p scoma      0.00000000 6.875550e-04 1.359628e-03
p meanbp     0.00000000 0.000000e+00 -6.037169e-05
p hrt        0.00000000 8.077068e-04 1.429942e-03
p resp       0.00000000 0.000000e+00 -1.981581e-03
p temp       0.00000000 1.075193e-02 1.606109e-02
p pafi       0.00000000 -1.784066e-05 -1.393418e-04
p raceasian  0.00000000 0.000000e+00 2.077249e-01
p raceblack  0.00000000 0.000000e+00 2.271800e-02
p racehispanic 0.00000000 0.000000e+00 1.987516e-01
p raceother  0.00000000 0.000000e+00 1.300163e-01
p racewhite  0.00000000 0.000000e+00 7.792639e-02
```

Der Befehl `r.lasso[c(1,10,20)]` zeigt uns die Koeffizienten für die Werte `lambda`  
`905.93833923 2.740463e+02 2.264846e+00` (mehr dazu siehe das pdf-File).

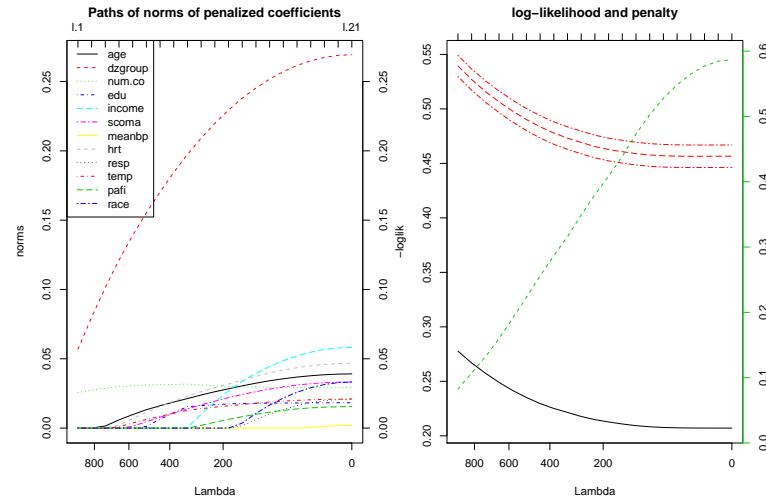
- Plot:

```
plot(r.lasso,type="norms")
```



Bemerkung:

- `plot(r.lasso,type="norms")`: der Plot zeigt uns den Betrag der **standardisierten Koeffizienten** der einzelnen Variablen. Einen Plot der wirklichen Koeffizienten: `plot(r.lasso,type="coefficients")` (ist noch unvollständig)
- `plot(r.lasso,type="criteria", cv=TRUE)`: der Plot zeigt uns die Graphik der Terme  $\sum R_i^2$  (Schwarz),  $\sum |\beta_i^*|$  (Grün) und die Graphik der **Cross-Validation** (Rot). Dieser Plot kann dazu bestimmt werden, das optimale  $\lambda$  zu wählen: das Minimum der roten Kurve. Dazu aber mehr in der Vorlesung. Eine Plot ohne die Cross-Validation: `plot(r.lasso,type="criteria")` oder `plot(r.lasso,type="criteria", cv=FALLS)`



- Wahl  $\lambda$ : ich wähle den Wert  $\lambda = 100$ . Dh, ich fahre fort mit dem Modell:
- ```
r.supp3 <- regr(formula = log10(totcst + 1) ~ age + dzgroup + num.co + edu + income + scoma + hrt + resp + temp + pafi + race, data = d.suppl)
```

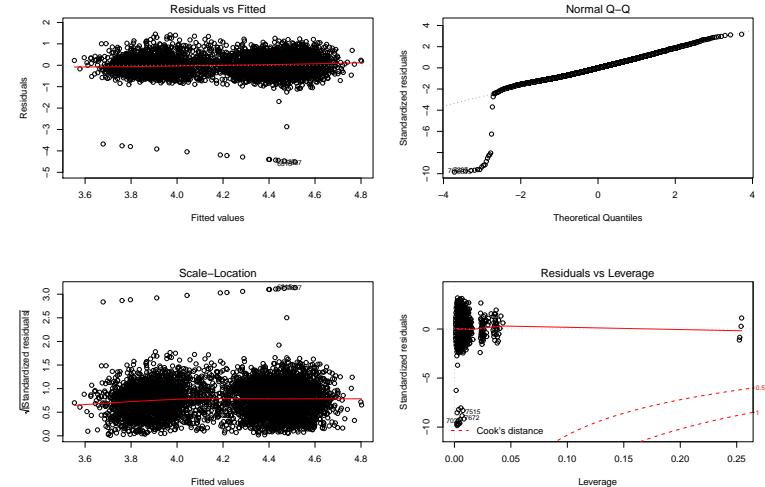
- Residuenanalyse:

```
e.lasso<-extract.lassogrp(r.lasso,lambda=100)
par(mfrow=c(2,2))
plot(e.lasso)
```

Bemerkung: Die Residuenanalysen zeigen, dass das Modell die Voraussetzungen im Grossen und Ganzen gut erfüllt. Jedoch:

QQ-Plot: im unteren Bereich schlecht, dh diejenigen mit kleinen Totalkosten. Dies ersieht man auch aus dem Histogramm.

Ausreisser: betrachtet man zb die Daten 7515 und 7672, so gilt  $totcst=0$ . Diese Erkenntnis deckt sich mit dem QQ-Plot. Diese Ausreisser (kleine Totalkosten) müssten jetzt lokalisiert und genauer analysiert werden. Wir machen diese Analyse hier nicht (dazu haben wir zu wenig Kenntnisse über die Datenentstehung).



- c) Die Hinzunahme von quadratischen Termen und Wechselwirkungen

R-Code:

```
r.supp3 <- regr(formula = log10(totcst + 1) ~ age + dzgroup + num.co + edu + income + add1(r.supp3))
```

Ich addiere die so gewonnenen Einzelterme zu meinem Modell (ich nehme nur diejenigen mit  $**$  oder  $***$ , addiere aber zusätzliche Terme dritten Grades für **age**, **scoma** und **resp3**) und benütze backwards und Lasso.

R-Code:

```
r.supp3 <- lm(formula = log10(totcst + 1) ~ age + dzgroup + num.co + edu + income + scoma + hrt + temp + I(age^2) + I(age^3) + I(scoma^2) + I(scoma^3) + I(resp^2) + I(resp^3) + age:dzgroup + age:income + age:scoma + dzgroup:num.co + dzgroup:scoma + dzgroup:temp + resp:temp , data = d.suppl)
```

- *backwards*-Output:

```
> r.supp3.step.formula <- log10(totcst + 1) ~ age + dzgroup + num.co + edu + income + scoma + hrt + temp + I(age^2) + I(age^3) + I(scoma^2) + I(scoma^3) + I(resp^2) + I(resp^3) + age:dzgroup + age:income + age:scoma + dzgroup:num.co + dzgroup:scoma + dzgroup:temp + resp:temp
>
Call:
regr(formula = r.supp3.step.formula, data = d.suppl)
Fitting function lm
```

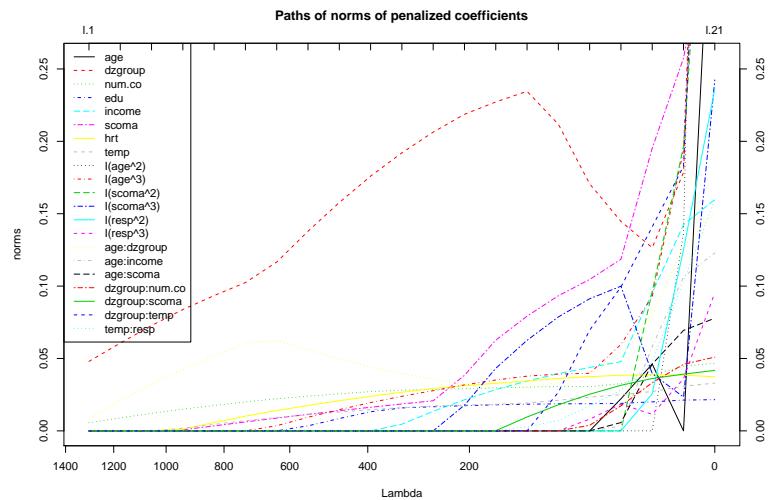
Terms:

| coef | stcoef | signif | R2.x | df | p.value |
|------|--------|--------|------|----|---------|
|------|--------|--------|------|----|---------|

```
(Intercept) 4.076004e+00 0.00000000 6.731818 NA 1 0.0000
age -3.289705e-02 -0.91474316 -1.923909 0.9605 1 0.0002
dzgroup NA NA 2.257877 0.9687 7 0.0120
num.co -3.424305e-02 -0.08401324 -2.600750 0.4189 1 0.0000
edu 6.297949e-03 0.03881044 1.844551 0.1079 1 0.0003
income NA NA 6.206637 0.7662 4 0.0000
scoma 1.620045e-02 0.67359056 3.523104 0.9018 1 0.0000
hrt 1.202673e-03 0.06723982 3.316212 0.0742 1 0.0000
temp 2.626791e-02 0.05890595 1.835752 0.4150 1 0.0003
I(age^2) 6.802161e-04 2.25004115 2.206561 0.9816 1 0.0000
I(age^3) -4.567035e-06 -1.45842065 -2.599995 0.9665 1 0.0000
I(scoma^2) -2.603963e-04 -0.92610072 -1.969288 0.9601 1 0.0001
I(scoma^3) 1.287650e-06 0.43743716 1.348627 0.9421 1 0.0082
I(resp^2) 4.655122e-04 0.42406205 1.492459 0.9339 1 0.0034
I(resp^3) -3.701088e-06 -0.17132603 -1.093395 0.8802 1 0.0321
age:dzgroup NA NA 4.345752 0.8060 7 0.0000
age:income NA NA 4.180468 0.7697 4 0.0000
age:scoma -5.175653e-05 -0.13966009 -1.487251 0.8001 1 0.0036
dzgroup:num.co NA NA 2.425211 0.5283 7 0.0076
dzgroup:scoma NA NA 3.948213 0.1924 7 0.0000
dzgroup:temp NA NA 2.200597 0.9680 7 0.0139
temp:resp -4.457397e-04 -0.28652520 -1.939245 0.8730 1 0.0001

St.dev.error: 0.4372 on 6679 degrees of freedom
Multiple R^2: 0.3876 Adjusted R-squared: 0.3824
F-statistic: 74.17 on 57 and 6679 d.f., p.value: 0
```

- Lasso-Output:

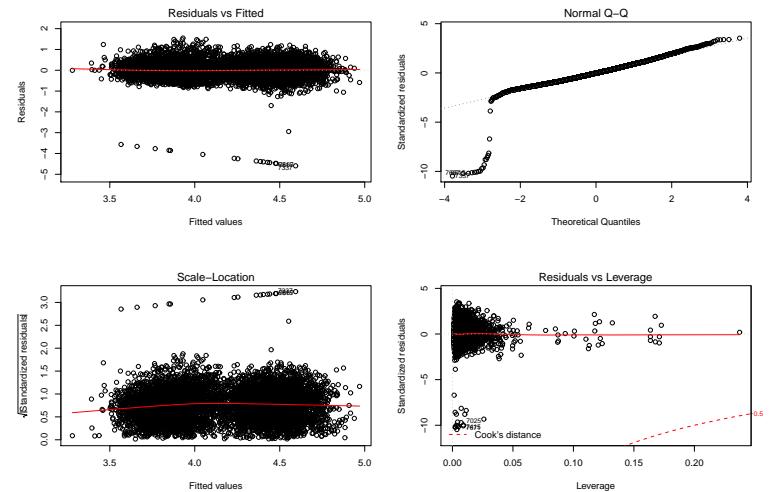


und wähle  $\lambda = 50$  - leider konnte kein plot der Cross-Validation gemacht werden

(geht zu lange und stürzt ab). Ich erhalte ein Modell

```
r.supp.l <- lm(formula = log10(totcst + 1) ~ dzgroup + num.co
+ edu + income + scoma + hrt + temp + I(age^3) + I(scoma^3)
+ I(resp^3) + age:dzgroup + dzgroup:num.co + dzgroup:scoma
+ dzgroup:temp + resp:temp , data = d.suppl)
```

- Residuenanalyse:



Hat sich nicht (viel) geändert. Die Probleme scheinen immer noch bei den kleinen Totalkosten zu sein. Hier müsste man auf jedenfall einen Stopp machen und die Daten bereinigen - ansonsten erstellt man ein Modell, dass dank den unteren Ausreisern nicht mehr haltbar ist.

- d) Ridge Regression:

- R-Code:

```
ridge.supp <- lm.ridge(log10(totcst+1) ~ age + dzgroup + num.co + edu
+ income + scoma + meanbp + hrt
+ resp + temp + pafi + race, data=d.suppl, lambda=seq(0,840,1))
```

```
ridge.supp$coef[,1:3]
```

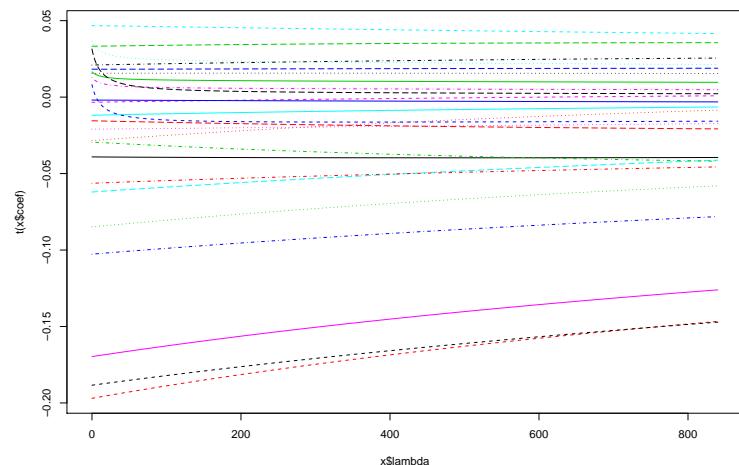
R-Output:

|                     | 0            | 1            | 2            |
|---------------------|--------------|--------------|--------------|
| age                 | -0.039089509 | -0.039095733 | -0.039101437 |
| dzgroupCHF          | -0.196978522 | -0.196892813 | -0.196807197 |
| dzgroupCirrhosis    | -0.084836513 | -0.084795488 | -0.084753728 |
| dzgroupColon Cancer | -0.102702442 | -0.102662097 | -0.102621901 |
| dzgroupComa         | -0.062049710 | -0.062015469 | -0.061981413 |
| dzgroupCOPD         | -0.169667954 | -0.169597321 | -0.169526366 |
| dzgroupLung Cancer  | -0.188406143 | -0.188340176 | -0.188274327 |
| dzgroupMOSF w/Malig | -0.028385464 | -0.028352881 | -0.028319877 |

|                   |              |              |              |
|-------------------|--------------|--------------|--------------|
| num.co            | -0.029434704 | -0.029457793 | -0.029481373 |
| edu               | 0.018282282  | 0.018276399  | 0.018271551  |
| income\$11-\$25k  | -0.011891878 | -0.011876467 | -0.011861871 |
| income\$25-\$50k  | -0.003408096 | -0.003394872 | -0.003382498 |
| income>\$50k      | 0.015585610  | 0.015591589  | 0.015596779  |
| incomeunder \$11k | -0.056350123 | -0.056327266 | -0.056305143 |
| scoma             | 0.033259414  | 0.033266174  | 0.033272972  |
| meanbp            | -0.001923412 | -0.001925910 | -0.001928335 |
| hrt               | 0.046773261  | 0.046768745  | 0.046763789  |
| resp              | -0.020941954 | -0.020938923 | -0.020935655 |
| temp              | 0.020978348  | 0.020989713  | 0.021000703  |
| pafi              | -0.015478464 | -0.015487710 | -0.015497183 |
| raceasian         | 0.016561461  | 0.016111120  | 0.015725168  |
| raceblack         | 0.008168855  | 0.006150508  | 0.004422174  |
| racehispanic      | 0.036626863  | 0.035588097  | 0.034698057  |
| raceother         | 0.012759946  | 0.012206882  | 0.011733011  |
| racewhite         | 0.031481831  | 0.029214670  | 0.027272873  |

dh, die Koeffizienten der erklärenden Variablen für  $\lambda \in \{0, 1, 2\}$ .

- `plot(ridge.supp)`



Bemerkung: im Gegensatz zum Lasso-Plot zeigt uns dieser Plot die standartisierten Koeffizienten der Variablen und Faktoren, deshalb diese vielen Linien. Aber es ist deutlich zu sehen, dass die viele der Linien gegen Null konvergieren, aber nicht ab einem  $\lambda$  Null werden wie im Lasso. Deshalb nützt diese Methode wenig für die Modellwahl.