Experimental Design and Data Analysis, Lecture 8

Eduard Belitser

VU Amsterdam

Lecture Overview

1. strategies to choose the variables
   - step up
   - step down
2. diagnostics in linear regression
3. problems in linear regression
   - outliers and influence points
   - collinearity

strategies to choose the variables

Strategies to choose the variables

An important issue in multiple linear regression is how to find a suitable model. That is, how to select explanatory variables $X_1, \ldots, X_p$ such that

$$Y_n = \beta_0 + \beta_1 X_{n1} + \beta_2 X_{n2} + \cdots + \beta_p X_{np} + e_n, \quad n = 1, \ldots, N,$$

is a good model for the given data.

A good model should be as precise and as concise as possible. It should

- contain all explanatory variables X_j that are essential in explaining Y
- not contain any variable X_j that does not contribute significantly.

Common strategies to build a model are:

- step-up
- step-down
- lasso (next lecture)

The coefficient of determination $R^2 \in [0, 1]$ yields a global check on the linear regression model. The higher R^2 the more variation the model explains.

Step-up method

In the step-up method one starts with fitting all p possible simple linear regression models ($j = 1, \dots, p$):

$$Y_n = \beta_0 + \beta_1 X_{nj} + e_n, \quad n = 1, \dots, N,$$

and selects the explanatory variable X_{j_o} that delivers the highest R^2 value.

In all next steps one explanatory variable is added as follows:

- compute R^2 for the obtained model extended with X_j for each X_j that is not (yet) in the model,
- select the X_j that yields the highest R^2 increase,
- stop when a newly added X_j yields insignificant explanatory variables.

Step-down method

In the step-down method one starts with fitting all explanatory variables in the so called full model:

$$Y_n = \beta_0 + \beta_1 X_{n1} + \beta_2 X_{n2} + \cdots + \beta_p X_{np} + e_n, \quad n = 1, \ldots, N.$$

In all next steps one explanatory variable is removed as follows:

- find the X_j that has the highest p-value for $H_0 : \beta_j = 0$.
- if that p-value is larger than 0.05, remove the X_j.
- stop when all remaining explanatory variables in the model are significant.

Analysis in R: data input

In the `sat` data there are 4 possible explanatory variables, `expend`, `ratio`, `salary` and `takers`. The response variable is `total`.

```
> sat[1:4,]
         expend ratio salary takers verbal math total
Alabama   4.405  17.2 31.144      8    491  538  1029
Alaska    8.963  17.6 47.951     47    445  489   934
Arizona   4.778  19.3 32.175     27    448  496   944
Arkansas  4.459  17.1 28.934      6    482  523  1005
> pairs(sat[,c(1:4,7)])
```

The variable `ratio` is the average pupil/teacher ratio in the state. The variable `salary` is the average salary of the teachers in the state. `verbal` and `math` denote subscores of the test.

Analysis in R: step-up (1)

The step-up method (only relevant output)

```
> summary(lm(total~expend,data=sat))
            Estimate Std. Error t value Pr(>|t|)
expend       -20.892      7.328  -2.851  0.00641 **

Multiple R-squared: 0.1448

> summary(lm(total~ratio,data=sat))
            Estimate Std. Error t value Pr(>|t|)
ratio          2.682      4.749   0.565    0.575

Multiple R-squared: 0.006602

> summary(lm(total~salary,data=sat))
            Estimate Std. Error t value Pr(>|t|)
salary        -5.540      1.632  -3.394  0.00139 **

Multiple R-squared: 0.1935

> summary(lm(total~takers,data=sat))
             Estimate Std. Error t value Pr(>|t|)
takers        -2.4801     0.1862  -13.32   <2e-16 ***

Multiple R-squared: 0.787
```

Analysis in R: step-up (2)

The step-up method (continued)

```
> summary(lm(total~takers+expend,data=sat))
            Estimate Std. Error t value Pr(>|t|)
takers       -2.8509     0.2151 -13.253  < 2e-16 ***
expend       12.2865     4.2243   2.909  0.00553 **

Multiple R-squared: 0.8195

> summary(lm(total~takers+ratio,data=sat))
             Estimate Std. Error t value Pr(>|t|)
takers        -2.5474     0.1871 -13.618   <2e-16 ***
ratio         -3.7264     2.2089  -1.687   0.0982 .

Multiple R-squared: 0.7991

> summary(lm(total~takers+salary,data=sat))
            Estimate Std. Error t value Pr(>|t|)
takers       -2.7787     0.2285 -12.163    4e-16 ***
salary        2.1804     1.0291   2.119   0.0394 *

Multiple R-squared: 0.8056
```

Analysis in R:step-up (3)

The step-up method (continued)

```
> summary(lm(total~takers+expend+ratio,data=sat))
             Estimate Std. Error t value Pr(>|t|)
takers        -2.8491     0.2155 -13.222   <2e-16 ***
expend        11.0140     4.4521   2.474   0.0171 *
ratio         -2.0282     2.2071  -0.919   0.3629

Multiple R-squared: 0.8227

> summary(lm(total~takers+expend+salary,data=sat))
            Estimate Std. Error t value Pr(>|t|)
takers       -2.8402     0.2248 -12.635   <2e-16 ***
expend       13.3326     7.0421   1.893   0.0646 .
salary       -0.3087     1.6530  -0.187   0.8527

Multiple R-squared: 0.8196
```

Adding either `ratio` or `salary` yields insignificant explanatory variables. Therefore, we should stop at the previous step.

Analysis in R: step-up (4)

The step-up method (continued)

```
> summary(lm(total~takers+expend,data=sat))

Call:
lm(formula = total ~ takers + expend, data = sat)

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 993.8317    21.8332  45.519  < 2e-16 ***
takers       -2.8509     0.2151 -13.253  < 2e-16 ***
expend       12.2865     4.2243   2.909  0.00553 **

Multiple R-squared: 0.8195.
```

The resulting model of the step-up method is

`total = 993.8317 + 12.2865*expend - 2.8509*takers + error`

Analysis in R: step-down (5)

The step-down method (only relevant output)

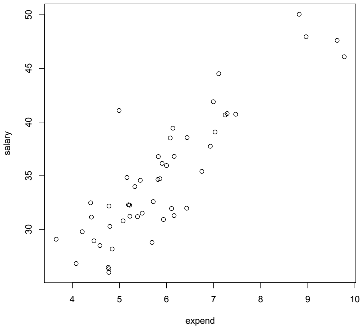
```
> summary(lm(total~expend+ratio+salary+takers,data=sat))
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 1045.9715    52.8698  19.784  < 2e-16 ***
expend         4.4626    10.5465   0.423    0.674
ratio         -3.6242     3.2154  -1.127    0.266
salary         1.6379     2.3872   0.686    0.496
takers        -2.9045     0.2313 -12.559 2.61e-16 ***
> summary(lm(total~ratio+salary+takers,data=sat))
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 1057.8982    44.3287  23.865   <2e-16 ***
ratio         -4.6394     2.1215  -2.187   0.0339 *
salary         2.5525     1.0045   2.541   0.0145 *
takers        -2.9134     0.2282 -12.764   <2e-16 ***

Multiple R-squared: 0.8239
```

The resulting model of the step-down method is
`total=1057.8982-4.6394*ratio+2.5525*salary-2.9134*takers+error`

Discussion (1)

We have found two different models by the two different strategies.
The reason is that `salary` and `expend` explain more or less the same.



If the plot of two explanatory variables shows (nearly) a straight line, the two variables are called collinear. Collinear variables should never be together in one model.

The amount of collinearity can be expressed based on the eigenvalues of the matrix containing all X_{ni} values.

Discussion (2)

Finding different models by different strategies is exemplary for linear regression: there is no golden strategy to resolve this.

In such a case one should compare

- R^2 values of both models (higher is better),
- plots of fitted values versus residuals of both plots (should be no specific structure),
- the number of explanatory variables in both models (fewer is better),
- the character of the explanatory variables in both models (easy to measure?),
- interpretation of both models,
- ...

and choose the one that is most appropriate.

diagnostics in linear regression

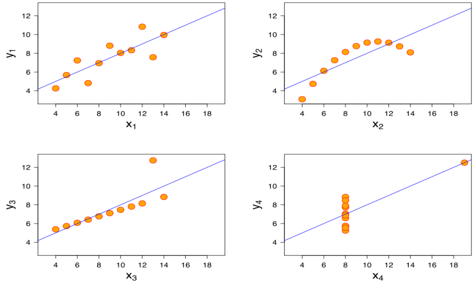
Example

Checking the fit in the linear regression by looking at the (adjusted) R^2 is not sufficient, we need to check the model assumptions: the linearity of the relation and the normality of the errors. We consider both graphical and numerical tools.

In the following 4 examples of artificial data, the fitted model is
`y = 3.0 + 0.5*x + error`, $\hat{\sigma}^2 = 1.5$ and $R^2 = 0.67$.

The differences between the 4 situations illustrate the need for a diagnostic tool, apart from $R^2, \hat{\sigma}$.

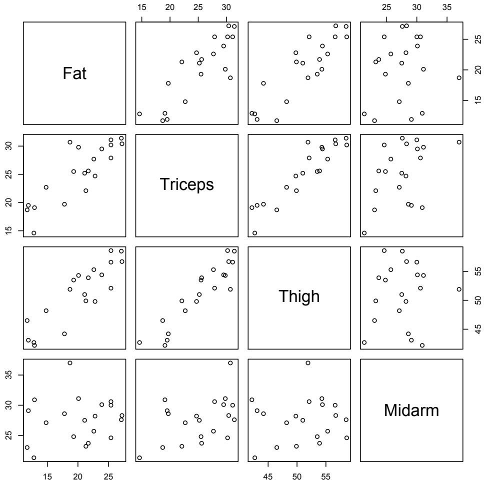
1. The first looks ok.
2. No lin. relation between X, Y.
3. Outlying point in Y.
4. Only one X is different.



The bodyfat data set

Example. Dataset `bodyfat` contains body measures of 20 females: `Fat`, `Triceps`, `Thigh` and `Midarm`. The variable `Fat` is difficult to measure.
Goal: predict the variable `Fat` from other (easy to measure) variables.

Using step down and step up we get
Model 1: ($R^2 = 0.771$)
`Fat = -23.6345 + 0.8565*Thigh`
Model 2: ($R^2 = 0.7862$)
`Fat = 6.7916 + 1.0006*Triceps -0.4314*Midarm`
Question: Which one do we prefer?
Answer: Model 1 is preferred, as it has less variables, and an only slightly lower value of R^2.

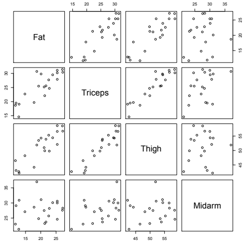


Diagnostic plots

To check the model quality look at

1. scatter plot: plot Y against each X_k separately (this yields overall picture, and shows outlying values)
2. scatter plot: plot residuals against each X_k in the model separately (look at pattern (curved?) and spread)
3. added variable plot (partial regression plot, see Velleman and Welsch (1981)): plot residuals of X_j against residuals of Y with omitted X_j (to show the effect of adding X_j to the model.) (Or, to show the relationship between Y and X_j, once all other predictors have been accounted for.)
4. scatter plot: plot residuals against each X_k not in the model separately (look at pattern — linear? then include!)
5. scatter plot: plot residuals against Y and $\hat{Y}$ (look at spread)
6. normal QQ-plot of the residuals (check normality assumption)

Example: bodyfat data (1)



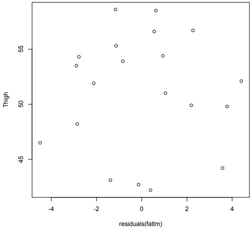
Read in the data.

```
>bodyfat=read.table("bodyfat.txt",header=T)
>attach(bodyfat)
```

1. Scatter plot of Y against each X_k separately.
   `> pairs(bodyfat)`

2. Scatter plot of residuals against each X_k in the model separately.
   `> bodyfatlm=lm(Fat∼Thigh)`
   `> plot(residuals(bodyfatlm),Thigh)`

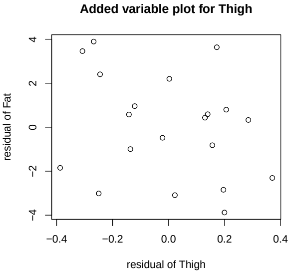
   If a curved pattern is visible, include, e.g., X_j^2 or transform X_j (e.g., $\log(X_j)$, $\sqrt{X_j}$).



Example: bodyfat data (2)

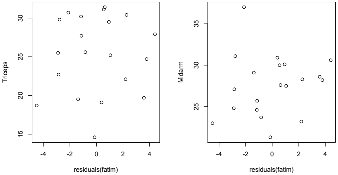
3. Added variable plot of residuals of X_j against residuals of Y with omitted X_j.

```
> x=residuals(lm(Thigh~Midarm+Triceps))
> y=residuals(lm(Fat~Midarm+Triceps))
> plot(x,y,main="Added variable plot for
  + Thigh", xlab="residual of Thigh",
  + ylab="residual of Fat"))
```

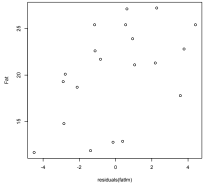


The slope in this plot reflects the regression coefficients β_j from the original multiple regression model, and the residuals in this plot are precisely the residuals from the original multiple regression. Outliers and heteroskedasticity (caused by X_j) can be identified by looking at the plot of a simple rather than multiple regression model.

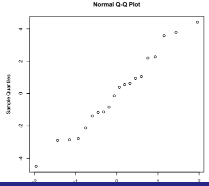
Example: bodyfat data (3)



4. Scatter plot of residuals against each X_k not in the model separately.
   ```
   > plot(residuals(bodyfatlm),Triceps)
   > plot(residuals(bodyfatlm),Midarm)
   ```



5. Scatter plot of residuals against Y (and $\hat{Y}$).
   ```
   > plot(residuals(bodyfatlm),Fat)
   > plot(res(bodyfatlm),fitted(bodyfatlm))
   ```



6. Normal QQ-plot of the residuals.
   ```
   > qqnorm(residuals(bodyfatlm))
   ```

   Also: `shapiro.test(residuals(bodyfatlm))`. If residuals are not normally distributed, go back to scatter plots and start with different model, possibly apply transforms.

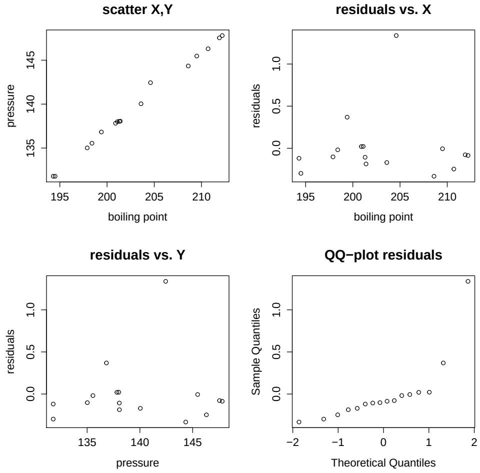
outliers and influence points

Outlier – Forbes' data (1)

An outlier is an observation with an extremely high or low response value, compared to what is expected under the model.

Consider Forbes' data which describe the relation between boiling point (`X`) of water and pressure (`Y`).

Residuals are for the simple linear regression model.



Outlier – Forbes' data (2)

```
> x=forbes[,2];y=forbes[,3];forbeslm=lm(y~x);round(residuals(forbeslm),2)
    1     2     3     4     5     6     7     8     9    10    11
-0.30 -0.12 -0.10 -0.02  0.37  0.02  0.02 -0.19 -0.11 -0.17  1.34
   12    13    14    15    16
-0.01 -0.33 -0.25 -0.08 -0.09
```

The 11-th data point seems to be an outlier. The command `order(abs(residuals(model)))` gives the indices of the ordered absolute values of residuals from smallest to largest. The last one(s) corresponds to the outlier(s).

```
> order(abs(residuals(forbeslm)))
[1] 12 4 6 7 15 16 3 9 2 10 8 14 1 13 5 11
```

Outlier – Forbes' data (3)

The mean shift outlier model can be applied to test whether the k-th point significantly deviates from the other points in a linear regression setting.

```
> u11=rep(0,16); u11[11]=1; u11
 [1] 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0
> forbeslm11=lm(y~x+u11); summary(forbeslm11)
...
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -40.787278   1.530216 -26.655 9.87e-13 ***
x             0.888534   0.007533 117.950  < 2e-16 ***
u11           1.433143   0.177565   8.071 2.03e-06 ***
...
```

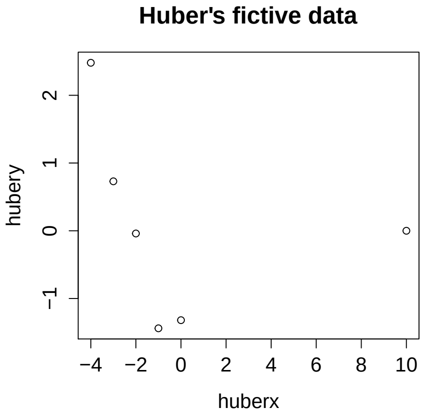
Since the coefficient for explanatory variable `u11` is significantly different from 0, the outlier is significant (it is common to apply a *one-sided* version of this test — we *know* whether the Y-value is very small or very big).

Definition potential point

A potential point (or leverage point) is an observation with an outlying value in an explanatory variable X_i.

Huber's fictive data.

Question: What is the influence of the observation with x=10?



Definition of influence point

- To study the effect of a leverage point one can fit the model with and without that data point. If the estimated parameters change drastically by deleting the leverage point, the observation is called an influence point.
- The Cook's distance for the i^{th} data point is

$$D_i = \frac{1}{(p+1)\hat{\sigma}^2} \sum_{j=1}^{n} (\hat{Y}_{(i),j} - \hat{Y}_j)^2,$$

with $\hat{Y}_{(i),j}$ the predicted j-th response based on the model without the i-th data point, p is the number of explanatory variables.

- The Cook's distance D_i quantifies the influence of observation i on the predictions.
- Rule of thumb: if the Cook's distance for some data point is close to or larger than 1, it is considered to be an influence point.

Example – Huber's data (2)

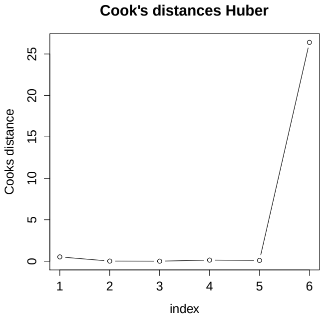
We compute the Cook's distances for Huber's data set:

```
> round(cooks.distance(huberlm),2)
   1    2    3    4    5     6
0.52 0.01 0.00 0.13 0.10 26.40
```

A plot of Cook's distances is usually insightful.

```
> plot(1:6,cooks.distance(huberlm),type="b")
```

Here we clearly see an influence point: the Cook's distance is 26.40 for the leverage point.



collinearity

Collinearity

Collinearity is the problem of linear relations between explanatory variables. A straight line in a scatter plot of two variables means they explain the same.

Example. Suppose we have a response variable Y and one explanatory variable X_1. Now we add a second explanatory variable$X_2 = 2X_1$. Can we do a meaningful analysis using the model $Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + e$? No, in this model we cannot uniquely estimate β_1 and β_2, because

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + e = \beta_0 + (\beta_1 + 2\beta_2) X_1 + e$$

and only the sum $\beta_1 + 2\beta_2$ is estimable.

If X_1 and X_2 are close to collinear then β_1 and β_2 are difficult to estimate. This is reflected in large variances and large confidence intervals of $\hat{\beta}_1$ and $\hat{\beta}_2$.

If the confidence interval of $\hat{\beta}_j$ is large, the estimate is not reliable.

We can have collinearity amongst a set of more than two explanatory variables (multicollinearity).

Ways to investigate and remove collinearity

Graphical ways to investigate collinearity:

- scatter plot of X_i against X_j for all i, j (pairwise collinearities).

Numerical way to investigate collinearity:

- pairwise linear correlation of X_i and X_j for all combinations i, j.
- variance inflation factor of β_j for all j (check whether these are high).

There are more advanced numerical ways to investigate collinearity (special packages in R like `car`), e.g.: condition indices, variance decomposition.

When there is collinearity amongst the explan. variables $X_1, \ldots, X_p$ one should

- avoid having two collinear explanatory variables in the model
- choose a model with a small number of explanatory variables
- choose a model that intuitively/practically makes sense

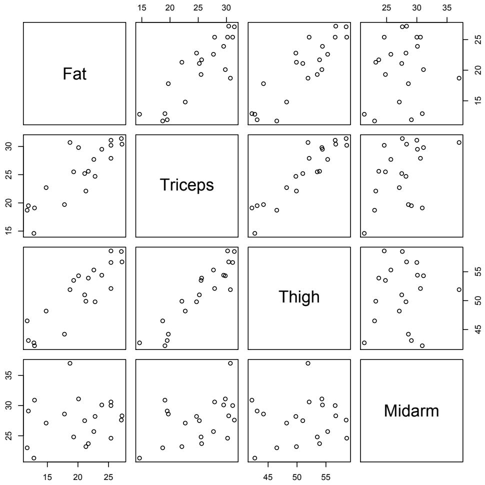
Without plots, one may not detect collinearity, use graphical checks!

Example - Bodyfat data

Apply these checks to the bodyfat data:

```
> round(cor(bodyfat),2)
         Fat Triceps Thigh Midarm
Fat     1.00    0.84  0.88   0.14
Triceps 0.84    1.00  0.92   0.46
Thigh   0.88    0.92  1.00   0.08
Midarm  0.14    0.46  0.08   1.00
```

```
> pairs(bodyfat)
```



Clearly `Triceps` and `Thigh` are collinear, both from the plot and from the correlation value of 0.92.

Variance inflation factor

To see which predictor variables are involved in collinearity we can look at the residuals of X_j regressed on the other explanatory variables (cf. added variable plot). If these residuals are very small, X_j is (nearly) a linear combination of other X's.

This is quantified in the variance inflation factor

$$VIF_j = \frac{1}{1 - R_j^2}, \qquad j = 1, \ldots, k,$$

with R_j^2 the determination coefficient of the mentioned regression.

Rule of thumb: VIF_j's larger than 5 indicate that $\hat{\beta}_j$ is unreliable.

Remark: these values do not give information about which variables are in the same collinear group of variables.

Example - Bodyfat data (8)

We compute the *VIF*-values for the bodyfat data.

```
> bodyfatlm=lm(Fat~Thigh+Triceps+Midarm, data=bodyfat)
> vif(bodyfatlm)
   Thigh  Triceps   Midarm
564.3434 708.8429 104.6060
> bodyfatlm2=lm(Fat~Triceps+Midarm, data=bodyfat)
> vif(bodyfatlm2)
 Triceps   Midarm
1.265118 1.265118
> bodyfatlm3=lm(Fat~Thigh, data=bodyfat)
> vif(bodyfatlm23)
Error in vif.default(bodyfatlm3) : model contains fewer than 2 terms
```

If we fit the full model all 3 VIF's are large, so there is a collinearity problem (as we saw in the scatter plots). The other 2 models are ok with respect to collinearity problems.

to finish

To wrap up

Today we learned:

- strategies to choose the variables (step up, step down)
- diagnostics in linear regression
- problems in linear regression (outliers and influence points, collinearity)

Next time: Lasso, ANCOVA.