# Advanced Regression: 2b Variable selection with correlated predictors

Verena Zuber

Epidemiology and Biostatistics, Imperial College London

24th January 2019

What is multicollinearity?

How to detect multicollinearity?

- Correlation matrix
- Variance inflation factor
- The rank of a matrix

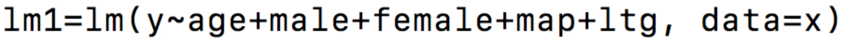
How to prevent multicollinearity?

- Grouping
- Partial least squares
- Pre-whitening

We consider again the diabetes outcome looking at the outcome disease progression $y$ and we try to fit the following linear model

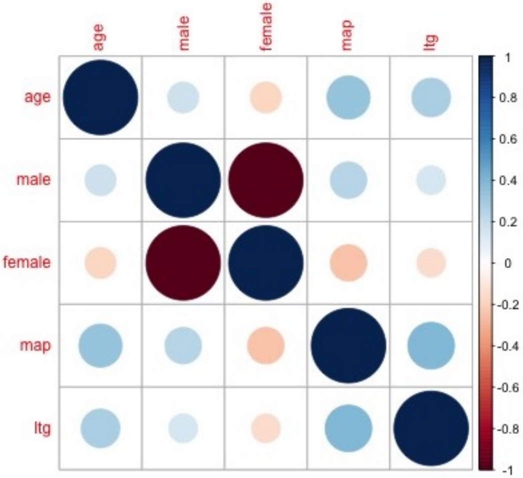
$$y = \alpha + age + male + female + map + ltg$$

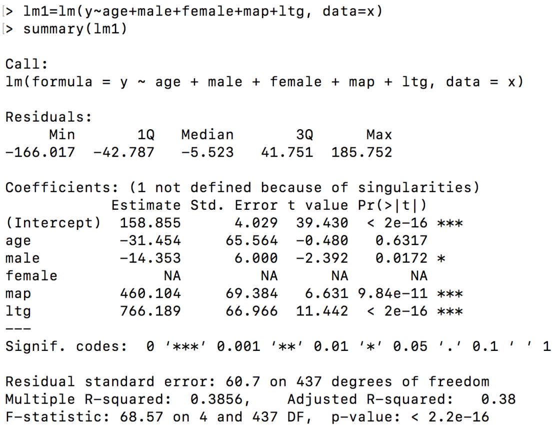
- age: age of the subject
- male: binary indicator if male
- female: binary indicator if female
- map: blood pressure
- ltg: triglycerides



```
lm1=lm(y~age+male+female+map+ltg, data=x)
```

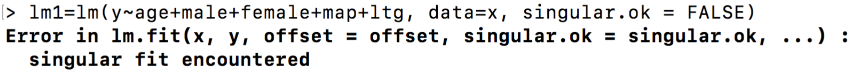
- Visualise the correlation structure using `corrplot()`





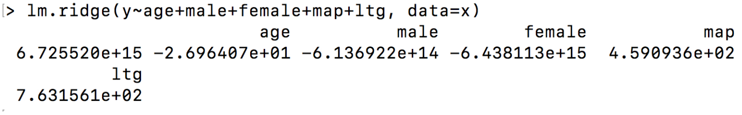
```
> lm1=lm(y~age+male+female+map+ltg, data=x)
> summary(lm1)

Call:
lm(formula = y ~ age + male + female + map + ltg, data = x)

Residuals:
     Min       1Q   Median       3Q      Max 
-166.017  -42.787   -5.523   41.751  185.752 

Coefficients: (1 not defined because of singularities)
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  158.855      4.029  39.430  < 2e-16 ***
age          -31.454     65.564  -0.480   0.6317    
male         -14.353      6.000  -2.392   0.0172 *  
female            NA         NA      NA       NA    
map          460.104     69.384   6.631 9.84e-11 ***
ltg          766.189     66.966  11.442  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 60.7 on 437 degrees of freedom
Multiple R-squared:  0.3856,	Adjusted R-squared:   0.38 
F-statistic: 68.57 on 4 and 437 DF,  p-value: < 2.2e-16
```

- Option in lm() function: singular.ok = TRUE automatically removes ‘female’.



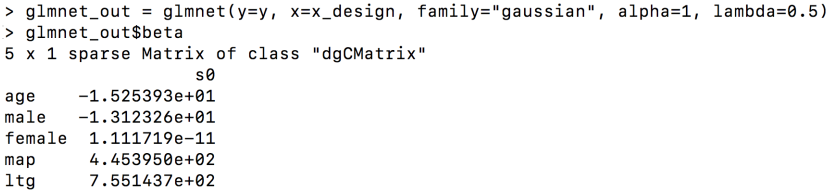
```
> lm1=lm(y~age+male+female+map+ltg, data=x, singular.ok = FALSE)
Error in lm.fit(x, y, offset = offset, singular.ok = singular.ok, ...) : 
  singular fit encountered
```

- The lm() function checks for singularities in the design matrix $x$, but not all methods have this safety check.
- Example: Ridge regression



```
> lm.ridge(y~age+male+female+map+ltg, data=x)
                         age          male        female           map 
 6.725520e+15 -2.696407e+01 -6.136922e+14 -6.438113e+15  4.590936e+02 
          ltg 
 7.631561e+02 
```

- Example: Lasso regression



```
> glmnet_out = glmnet(y=y, x=x_design, family="gaussian", alpha=1, lambda=0.5)
> glmnet_out$beta
5 x 1 sparse Matrix of class "dgCMatrix"
                 s0
age    -1.525393e+01
male   -1.312326e+01
female  1.111719e-11
map     4.453950e+02
ltg     7.551437e+02
```

# What is multicollinearity?

## Singularity

One predictor variable in a multiple regression model can be exactly explained by the other $p - 1$ predictor variables.

## Multicollinearity

One predictor variable in a multiple regression model can be linearly explained by the other $p - 1$ predictor variables with high accuracy.

What can cause singularity?

- Dummy-coding of categorical variables. Make sure not to add redundant information.
- Do not include multiple measurements that are measured on different scales (e.g. mol and mmol).

# What is the impact of multicollinearity?

## True biological processes

do not cause singularity (because they are random, not deterministic), but can cause multicollinearity.

- The computation of the ordinary least squares estimate requires an inversion of the $p \times p$-dimensional correlation matrix $x^t x$.
- $x^t x$ cannot be inverted when the $x^t x$ is singular.
- When there is multicollinearity, $x^t x$ can be inverted, but the estimate will show a high variance and will be highly instable.
- Multicollinearity can distort a linear model and impact the interpretation.

## How to detect multicollinearity?

1. Variance inflation factor
2. Rank of the between predictor correlation matrix
3. Condition number based on the ratio of largest over smallest singular value

# Covariance matrix

Computing the sample covariance matrix using matrix multiplication

$$\hat{cov}(x) = \frac{1}{n-1} \underbrace{x_c^t}_{p\times n} \underbrace{x_c}_{n\times p}$$

- $x_c$ is centred (mean is zero) $x_c = x - 1_n\bar{x} = cx$
  - where $\bar{x} = (\bar{x}_1, ..., \bar{x}_p)$ is the vector of means
  - and $1_n$ is a vector of ones
  - and $c = I_n - \frac{1}{n} 1_n 1_n^t$
  - and $I_n$ is the $n \times n$ identity matrix with ones on the diagonal
- $x$ predictor matrix of $n$ rows and $p$ columns
- $x^t$ transposed predictor matrix of $p$ rows and $n$ columns

# Matrix multiplication

Matrix multiplication: $c = ab$

$$c_{ij} = \sum_{k=1}^{m} a_{ik} b_{kj}$$

- $a$ is a $n \times m$ and $b$ is a $m \times p$ matrix
- $c$ is a $n \times m \times m \times p = n \times p$ matrix
- Make sure your matrices have the correct dimensions, number of columns of the left matrix must be equal to the number of rows on the right.
- Can be computed in R using the $\% * \%$ command.

# Correlation matrix

Computing the sample correlation matrix using matrix multiplication

$$\hat{cor}(x) = \frac{1}{n-1} \underbrace{x_s^t}_{p\times n} \underbrace{x_s}_{n\times p}$$

where $x_s$ is a centred and scaled matrix $x_s = cxd^{-1}$

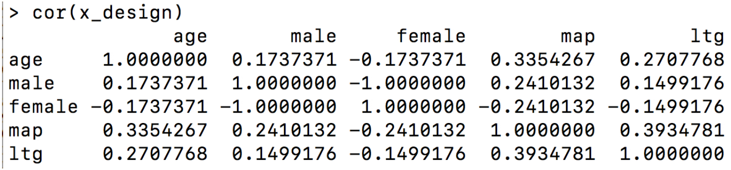
- where $d = diag(s)$ is a diagonal matrix
- with the sample standard deviation $s$ on the diagonal.

This is equivalent to writing

$$\hat{cor}(x_j, x_k) = \frac{\sum_{i=1}^n (x_{ij} - \bar{x}_j)(x_{ik} - \bar{x}_k)}{\sqrt{\sum_{i=1}^n (x_{ij} - \bar{x}_j)^2}\sqrt{\sum_{i=1}^n (x_{ik} - \bar{x}_k)^2}}$$

# Correlation matrix

- Correlation matrices are symmetric and have a vector of 1's on the diagonal.



```
> cor(x_design)
              age       male     female        map        ltg
age     1.0000000  0.1737371 -0.1737371  0.3354267  0.2707768
male    0.1737371  1.0000000 -1.0000000  0.2410132  0.1499176
female -0.1737371 -1.0000000  1.0000000 -0.2410132 -0.1499176
map     0.3354267  0.2410132 -0.2410132  1.0000000  0.3934781
ltg     0.2707768  0.1499176 -0.1499176  0.3934781  1.0000000
```

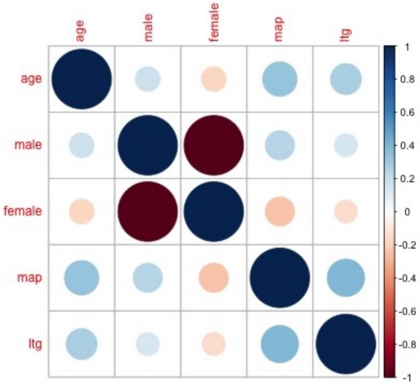
- Note the following correlation matrix captures the correlation between the samples and is of dimension $n \times n$

$$c\hat{o}r(x^t) = \frac{1}{p-1} \underbrace{x_s}_{n\times p} \underbrace{x_s^t}_{p\times n}$$

# Correlation matrix

R commands

- `cov()` sample covariance matrix
- `cor()` sample correlation matrix
- `corrplot()` to visualise



# Variance inflation factor (VIF)

- The VIF is the ratio of the variance of $\beta_j$ when fitting the full model divided by the variance of $\beta_{UNI}(j)$ in a unvariable linear model.
- Lowest possible value is 1 (no collinearity).
- Rule of thumb: If VIF $> 10$, this indicates strong multicollinearity, but already smaller VIF can impact the analysis.
- It provides an indication how much the variance of an estimated regression coefficient is increased because of multicollinearity.

# Variance inflation factor (VIF)

Consider the following linear model

$$y = \alpha + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_j x_j + \ldots + \beta_p x_p + \epsilon$$

1. For the first variable $j = 1$ fit the following linear model

   $$x_1 = \alpha + \beta_2 x_2 + \ldots + \beta_j x_j + \ldots + \beta_p x_p + \epsilon$$

   and estimate the proportion of variance explained ($R_2(1)$).
2. The VIF for variable 1 is defined as

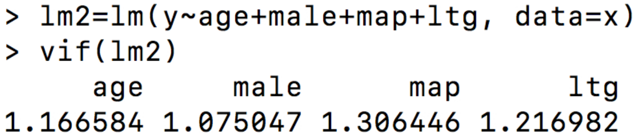
   $$VIF_1 = \frac{1}{1 - R_2(1)}$$

3. Repeat for the other $j \in 2, \ldots, p$.

# Variance inflation factor

R commands

- `vif()` in the *R*-package `car`
- Computes variance-inflation and generalized variance-inflation factors for linear and generalized linear models.



```
> lm2=lm(y~age+male+map+ltg, data=x)
> vif(lm2)
     age     male      map      ltg
1.166584 1.075047 1.306446 1.216982
```

- Interpretation: No variable has a VIF $> 10$, with around 1 they are rather low and there is no indication of multicollinearity.

# The rank of a matrix

- Consider a matrix $x$ of dimension $n \times p$.

$$\underbrace{x}_{n \times p}$$

- The rank of matrix $x$ is the minimum of $n$ and $p$.
- If we have more samples than variables $(n > p)$ the rank is $p$.
- If we have less samples than variables $(n < p)$ the rank is $n$.

# The rank of the correlation matrix

- Let us consider again the correlation matrix

$$\hat{cor}(x) = \frac{1}{n-1} \underbrace{x_s^t}_{p \times n} \underbrace{x_s}_{n \times p}$$

- The theoretical rank of the correlation matrix is the minimum of $n$ and $p$.
- To test the rank of a matrix in R: `rankMatrix()`

If the rank of a correlation matrix is smaller than $min(n, p)$ the correlation matrix is singular and thus cannot be inverted.

# Outlook: Big data ($n << p$)

- Assume we are considering a big data set with much more variables than observations $n << p$

$$\hat{cor}(x) = \frac{1}{n-1} \underbrace{x_s^t}_{p \times n} \underbrace{x_s}_{n \times p}$$

- The theoretical rank of the correlation matrix is the minimum of $n$ and $p$.
- In case of big data, the rank of the matrix is $n$, which is much smaller than $p$.
- Thus the correlation matrix (and also $x_s^t x_s$) are singular and cannot be inverted.
- It is not possible to compute the ordinary least squares estimate for big data.

# Singular value decomposition and condition number

Singular value decomposition of a matrix $m$ (dimension $n \times p$) is defined as

$$m = u\Sigma v$$

- $\Sigma$: Matrix of singular values (dimension $n \times p$)
- $u$: Left-singular vectors (dimension $n \times n$)
- $v$: Right-singular vectors (dimension $p \times p$)

# Singular value decomposition and condition number

- A singular value decomposition of the sample correlation matrix produces $p$ singular values $d_1$ to $d_p$.
- After sorting the eigenvalues in decreasing order

$$d_{[1]} > \ldots > d_{[j]} > \ldots > d_{[p]}$$

  - ◇ $d_{[1]}$ is the largest singular value
  - ◇ $d_{[p]}$ is the smallest singular value

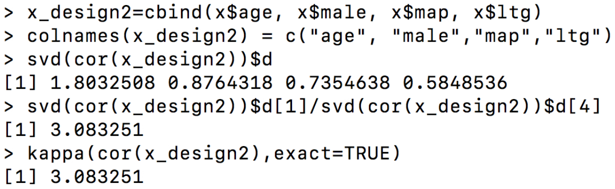
## Condition number

Ratio of largest over smallest singular value.

$$\kappa = d_{[1]}/d_{[p]}$$

# Singular value decomposition in R

- Singular value decomposition: `svd()`
  Value $*d*$ extracts the singular values
- Condition number: `kappa()`
  Use argument `exact=TRUE`



```
> x_design2=cbind(x$age, x$male, x$map, x$ltg)
> colnames(x_design2) = c("age", "male","map","ltg")
> svd(cor(x_design2))$d
[1] 1.8032508 0.8764318 0.7354638 0.5848536
> svd(cor(x_design2))$d[1]/svd(cor(x_design2))$d[4]
[1] 3.083251
> kappa(cor(x_design2),exact=TRUE)
[1] 3.083251
```

- Interpretation: The condition number is far below 30, which is often used as a rule of thumb. There is no sign of multicollinearity.

# Example: Diabetes data

- age: age of the subject
- male: binary indicator if male
- female1: binary indicator if female, but one sample is wrongly annotated

→ cor(male,female1) = -0.9954659

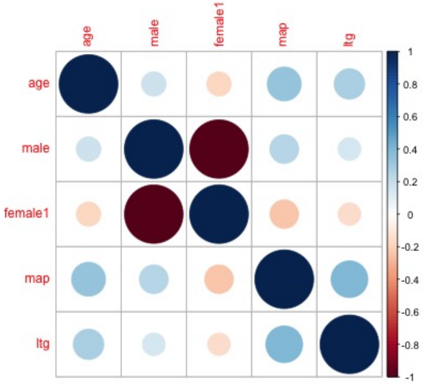
- map: blood pressure
- ltg: triglycerides

Correlation matrix

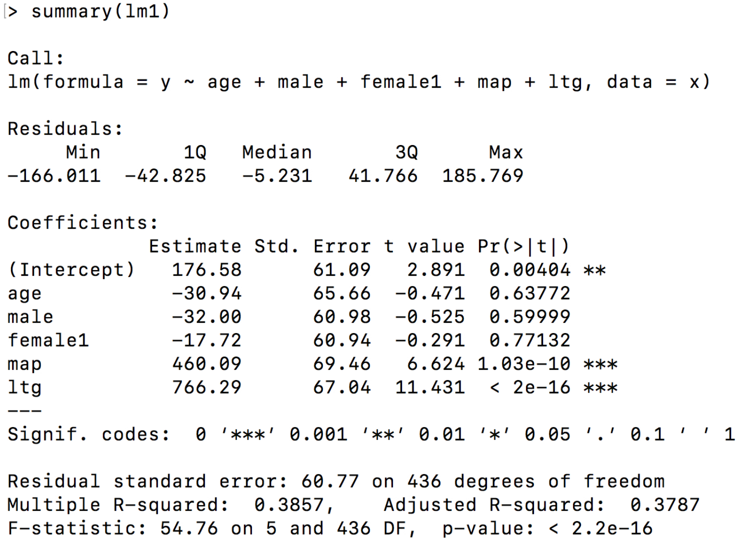
```
> x_design1=cbind(x$age, x$male, x$female1, x$map, x$ltg)
> colnames(x_design1) = c("age", "male", "female1","map","ltg")
> cor(x_design1)
               age       male    female1        map        ltg
age      1.0000000  0.1737371 -0.1701584  0.3354267  0.2707768
male     0.1737371  1.0000000 -0.9954659  0.2410132  0.1499176
female1 -0.1701584 -0.9954659  1.0000000 -0.2389995 -0.1480640
map      0.3354267  0.2410132 -0.2389995  1.0000000  0.3934781
ltg      0.2707768  0.1499176 -0.1480640  0.3934781  1.0000000
```

# Example: Diabetes data

## Correlation matrix corrplot



# Example: Diabetes data



```
> summary(lm1)

Call:
lm(formula = y ~ age + male + female1 + map + ltg, data = x)

Residuals:
     Min       1Q   Median       3Q      Max 
-166.011  -42.825   -5.231   41.766  185.769 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   176.58      61.09   2.891  0.00404 ** 
age           -30.94      65.66  -0.471  0.63772    
male          -32.00      60.98  -0.525  0.59999    
female1       -17.72      60.94  -0.291  0.77132    
map           460.09      69.46   6.624 1.03e-10 ***
ltg           766.29      67.04  11.431  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 60.77 on 436 degrees of freedom
Multiple R-squared:  0.3857,	Adjusted R-squared:  0.3787 
F-statistic: 54.76 on 5 and 436 DF,  p-value: < 2.2e-16
```

# Example: Diabetes data

- The linear model can be calculated now since there is no exact collinearity.
- Note that male and female have both a negative regression coefficient.
- VIF()

```
> lm1=lm(y~age+male+female1+map+ltg, data=x)
> vif(lm1)
       age       male    female1        map        ltg
  1.167432 110.819281 110.626114   1.306447   1.217015
```

- Interpretation: VIF for male and female1 is highly inflated and indicates strong multicollinearity.
- This inflation distorts the linear model and hinders the interpretation of the male and female1 regression coefficients.

# Example: Diabetes data

- Singular value decomposition and condition number

```
> x_design2=cbind(x$age, x$male, x$female1, x$map, x$ltg)
> colnames(x_design2) = c("age", "male","female1","map","ltg")
> svd(cor(x_design2))$d
[1] 2.323503750 1.346325644 0.736162810 0.589481033 0.004526764
> svd(cor(x_design2))$d[1]/svd(cor(x_design2))$d[5]
[1] 513.2814
> kappa(cor(x_design2),exact=TRUE)
[1] 513.2814
```

- Interpretation: The condition number is very high ($>>$ 30) and indicates strong multicollinearity.

# How to prevent multicollinearity?

- Grouping
- Partial least squares
- Pre-whitening

# Grouping

- When there is biological knowledge of pre-defined groups of variables (e.g. genes within a pathways, lipid characteristics of specific subfractions), it is advised to group them and use only one variable within the group as representative.
- Group structures can be defined using unsupervised learning approaches such as clustering.
- Projections into a lower-dimensional space
  - Principle component analysis (PCA)
  - Independent component analysis
  - Non-negative matrix factorisation

# Partial least squares

- PLS is a dimension reduction approach that is coupled with a regression model.
  1. Create latent components $t$ as a linear transformation of $x$ and latent components $u$ as a linear transformation of $y$

$$\begin{aligned}\underbrace{x}_{n\times p} &= \underbrace{t}_{n\times l}\underbrace{q_x^t}_{l\times p}+\epsilon_x\\ \underbrace{y}_{n\times 1} &= \underbrace{u}_{n\times l}\underbrace{q_y^t}_{l\times 1}+\epsilon_y\end{aligned}$$

  2. Idea: The latent components have a smaller dimension $l < p$.
  3. Aim: Find decompositions of both $x$ and $y$ that maximise the covariance between $t$ and $u$.
- R-package: `pls`

# Pre-whitening

- Prewhitening transformations are matrix operations that ‘remove’ correlation.
- Suppose $x$ has a mean vector of 0 and covariance matrix $\Sigma$.
- There exists a whitening matrix $w$ that satisfies $w^t w = \Sigma^{-1}$.
- The prewhitened data $x^\star$ is defined as

  $$x^\star = wx,$$

  where the covariance of $x^\star$ is diagonal.
- There are several algorithms to compute $w$
  - Mahalanobis transformation $w = \Sigma^{-1/2}$
  - Cholesky decomposition of $\Sigma$
  - PCA based
- R-package: `whitening`

# Take away: Variable selection with correlated predictors

- Exact correlation between predictors can cause singularity of the correlation and covariance matrix.
- Strong correlation between predictors can cause multicollinearity.
- Multicollinearity can distort linear regression models and inflate the variance of the estimate.
- How to detect multicollinearity?
  - ◇ Variance inflation factor
  - ◇ Rank of the between predictor correlation matrix
  - ◇ Condition number based on the ratio of largest over smallest singular value