# TMA4268 Statistical Learning

## Module 6: Solution sketches

Thiago G. Martins, Department of Mathematical Sciences, NTNU

Spring 2022

# Recommended exercise 1

For the least square estimator, the solution can be found in the first session here.

For the maximum likelihood estimator, the solution can be found here.

# Recommended exercise 2

```
library(ISLR) # Package with data for an Introduction to St
              # Learning with Applications in R
```

```
# Load Credit dataset
data(Credit)


# Check column names
names(Credit)
```

```
##  [1] "ID"        "Income"    "Limit"     "Rating"    "Ca
##  [7] "Education" "Gender"    "Student"   "Married"   "Et
```

```
# Check dataset shape
dim(Credit)
```

```
## [1] 400  12
```

```
head(Credit)
```

# Recommended exercise 3

```r
# Exclude 'ID' column
credit_data <- subset(Credit, select=-c(ID))

# Counting the dummy variables as well
credit_data_number_predictors <- 11

# Take a look at the data
head(credit_data)
```

```
##    Income Limit Rating Cards Age Education Gender Studen
## 1  14.891  3606    283     2  34        11   Male      N
## 2 106.025  6645    483     3  82        15 Female     Ye
## 3 104.593  7075    514     4  71        11   Male      N
## 4 148.924  9504    681     3  36        11 Female      N
## 5  55.882  4897    357     2  68        16   Male      N
## 6  80.180  8047    569     4  77        10   Male      N
##   Balance
## 1     333
```

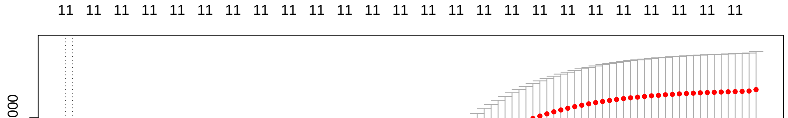
# Recommended exercise 4

Similar analysis as previous exercise, simply replace Best Subset Selection
`(best_subset_method=regsubsets(Balance~.,credit_data,nvmax=`
by Forward Stepwise Selection
`(regfit.fwd=regsubsets(Balance~.,credit_data,nvmax=credit_d`
Backward Stepwise Selection
`(regfit.fwd=regsubsets(Balance~.,credit_data,nvmax=credit_d`
and Hybrid Stepwise Selection
`(regfit.fwd=regsubsets(Balance~.,credit_data,nvmax=credit_d`

# Recommended exercise 5

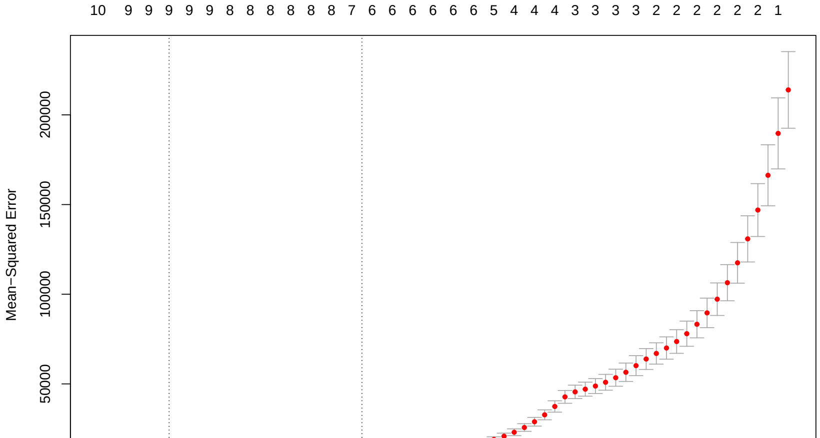
```r
library(glmnet) # Package Lasso and Elastic-Net Regularize
                # Generalized Linear Models
```

```r
x_train <- model.matrix(Balance~.,credit_data_training)[,-1
y_train <- credit_data_training$Balance

x_test <- model.matrix(Balance~.,credit_data_testing)[,-1]
y_test <- credit_data_testing$Balance

ridge_mod <- glmnet(x_train,y_train,alpha=0)

set.seed(1)
cv.out=cv.glmnet(x_train, y_train,alpha=0)
plot(cv.out)
```



# Recommended exercise 6

```
lasso_mod <- glmnet(x_train,y_train,alpha=1)

set.seed(1)
cv.out=cv.glmnet(x_train, y_train,alpha=1)
plot(cv.out)
```



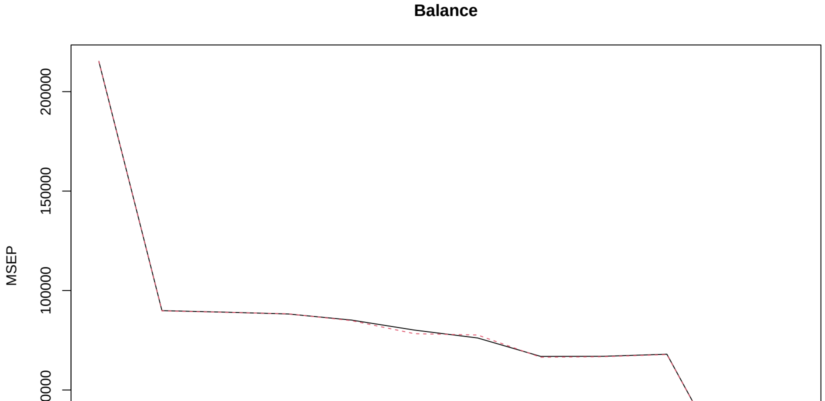
# Recommended exercise 7

```r
x <- model.matrix(Balance~.,credit_data)[,-1]

credit_pca <- prcomp(x, center = TRUE, scale. = TRUE)

print(credit_pca)
```

```
## Standard deviations (1, .., p=11):
##  [1] 1.66007642 1.26685832 1.05356810 1.04926273 1.0032
##  [7] 0.97830708 0.90714714 0.63722533 0.51174012 0.0461
##
## Rotation (n x k) = (11 x 11):
##                          PC1          PC2          P
## Income          -0.542206953  0.029036783 -0.0332706
## Limit           -0.586332930  0.017502630 -0.0243517
## Rating          -0.586751867  0.014971105 -0.0046307
## Cards           -0.019086978 -0.008549632  0.4790057
## Age             -0.122783390 -0.071116603  0.1071884
## Education        0.026797471  0.096557225 -0.4754183
```

# Recommended exercise 8

```
library(pls)

set.seed(1)

pcr_model <- pcr(Balance~., data=credit_data_training,scale
validationplot(pcr_model,val.type="MSEP")
```



# Recommended exercise 9

```
library(pls)

set.seed(1)

plsr_model <- plsr(Balance~., data=credit_data_training,sca
validationplot(plsr_model,val.type="MSEP")
```