

Homework:

Please complete this homework by filling the template provided on Canvas. Any submission which is not R will not be accepted. The code will need to properly run without any error

Exercise 1 - LR and Out of Sample Prediction

We generate 99 independent variables uniformly distributed between -100 and 100 of size 500 observations each. (see code below) We also generate the dependent variable $y = 3 + 10 \cdot V_{99}$, where V_{99} is the last covariate and add some noise (see code below).

- Construct 3 models: one linear model with no variables, one with all the variables and one with only the variable V_{99}
- Compute the MSE of each model

Hint: for the first two points code is provided below.

```
In [1]: set.seed(123)
n <- 500
p <- 99
x <- matrix(runif(n*p, min=-100,max=100), n , p)

## Generate the output variable as a linear combination of x

## With jitter() you add random noise
y <- jitter(3 + 10*x[,99], factor=10000)
```

- Pick from your data only 1/10th random observations (see code below)
- Use the remaining 9/10th observations to rebuild the three models
- Make prediction on the 1/10th observations
- What do you observe now?

```
In [2]: ## Pick randomly 1/10th of observations
## Hint
#ii <- sample(nrow(x), floor(nrow(x)/10))
```

Exercise 2 Cross Validation

We are interested in predicting the quality of wines using chemical indicators. To do so, we have a disposal two data sets for white and red wine, reporting the variable quality on a scale from 0 to 10.

- find the file read wine
- Find three models you might think are meaningful for the prediction with different number of variables
- Compute the in-sample mean squared
- Compute the out-of-sample mean squared error using a test-training set approach (remember to set the seeds)
- (challenging) Compute the out-of-sample mean squared error using 10-folds cross validation

Hint: see the lecture on cross validation and fill in the skeleton below.

```
In [3]: #wine.red <- read.table("./data/wine-red.txt")
#y <- wine.white$quality

## Skeleton : fill the missing entries
#n <- nrow(wine.red)
#k <-
#set.seed(123)
# ii <-
# mspe1 <- mspe2 <- mspe3 <-

# for (j in 1:k){

# hold <-
# train <-

# reg1 <-
# reg2 <-
# reg3 <-

# pr1 <-
# pr2 <-
# pr3 <-

# mspe1[j] <-
# mspe2[j] <-
# mspe3[j] <-
#}

## Return the mspe
#mean(mspe1)
#mean(mspe2)
#mean(mspe3)
```

Exercise 3 Ridge Regression

We are interested in predicting the level of alchol consumption during the weekend for students, controlling for many social and academic indicators. Some of them are the average grades for three years, the income of the family, the age, etc. In total we have 32 variables, but we want to find just the ones most correlated with alchol consumption.

Do the following:

- Download and open the student txt file

Note: your Y and X are provided in the code below.

```
In [4]: #student <- read.table('insert your path')
```

```
In [5]: ## Hint code for the first part of the exercise

## These must be your X and y
#X <- model.matrix(~. ,data=student)[, -27])[-1]
#y <- student[,27]
```

- Construct a sequence of lambda from e^{-4} to e^1 (see below)
- Use cross validation to find the best lambda to be used for estimating ridge regression (hint: use cv.glmnet function)
- Construct a ridge regression with the lambda with minimum error
- Plot the result of your cross validated function as function of the mean squared prediction error (hint: you can just use the command plot() and pass the estimated object)