# Study Verification Test (SSP) Results

The study verification test (SSP) was conducted 15th of September 2017 for 1 hour on all MED1 student, in Aalborg at 9 am and in Copenhagen at 12:30 am. The immediate result of the test showed that a few students (around 7) did not show for the test, as they had dropped out in MED1. These dropouts will be replaced with new students. All active students took the test as requested.

The goals of analysing SSP:

A) Detecting students with high risk of dropping out. We will inform the student counselors who is the high-risk dropout students so they can take action. The student counselors need the results from SSP as an overview for each high-risk dropout students, e.g. a percentile rank for each category as in https://www.ruffalonl.com/complete-enrollment-management/student-success/rnl-retention-management-system-plus/samples.
B) Creating a tool for predicting dropouts (low priority).

TODOS:

- Compute different types of data analysis to describe the dropout rate (with aggegation of variables)
- Mail merge: https://www.r-bloggers.com/mail-merge-with-rmarkdown/
- Import remaining data from Moodle of the newcoming students

This script imports, cleans, and analyses the data from SSP.

### Importing and Cleaning Data

Setting the working directory to the SVN folder.

```
SVNData<-if(grepl("BiancaClavio", getwd())){
  'C:/Users/BiancaClavio/Documents/SVN/01Projects/SSP/'} else {"~/SVN/01Projects/SSP/"}
setwd(SVNData)
```

Importing a csv file that gives and overview of quiz, questions, possible answers, and grading for each answer (We might not use this in Rstudio, but it is still useful). The document is created in Excel manually based on the SSP Moodle quiz, as the quiz questions from Moodle can be exported to anything useful outside Moodle.

```
dfQAGrades<-read.csv("QuestionsOverview.csv",
  header = TRUE, fill=TRUE, sep = ",", check.names=FALSE)
```

Importing SSP grades and answers via Moodle in Aalborg and Copenhagen. All text strings for the grading attributes are converted to NA's, e.g. Moodle write "-" when a student skipped a question. The data analysis further down ignores NA's in the computations.

```
dfSSPgradesCPH<-read.csv("SSPgradesTestCPH 28-09.csv",
  header = TRUE, fill=TRUE, sep = ",", check.names=FALSE,
  na.strings = c("-","Requires grading"))
dfSSPgradesAAL<-read.csv("SSPgradesTestAAL 28-09.csv",
  header = TRUE, fill=TRUE, sep = ",", check.names=FALSE,
  na.strings = c("-","Requires grading"))
dfSSPanswersCPH<-read.csv("SSPanswersTestCPH 28-09.csv",
  header = TRUE, fill=TRUE, sep = ",", check.names=FALSE)
dfSSPanswersAAL<-read.csv("SSPanswersTestAAL 28-09.csv",
  header = TRUE, fill=TRUE, sep = ",", check.names=FALSE)
```

This snippet changes some answer types in AAL data set to match the ones in CPH. The answer types from questions 24-32 in AAL (No influence, Limited influence, Some influence, Decisive influence) are converted to

the ones in CHP (Not at all true, Slightly true, Somewhat true, Completely true).

```
dfSSPanswersAAL[35:43] <- apply(dfSSPanswersAAL[35:43],2, function(dfSSPanswersAAL)
  gsub("No influence","Not at all true",dfSSPanswersAAL))
dfSSPanswersAAL[35:43] <- apply(dfSSPanswersAAL[35:43],2, function(dfSSPanswersAAL)
  gsub("Limited influence","Slightly true",dfSSPanswersAAL))
dfSSPanswersAAL[35:43] <- apply(dfSSPanswersAAL[35:43],2, function(dfSSPanswersAAL)
  gsub("Some influence","Some influence",dfSSPanswersAAL))
dfSSPanswersAAL[35:43] <- apply(dfSSPanswersAAL[35:43],2, function(dfSSPanswersAAL)
  gsub("Decisive influence","Completely true",dfSSPanswersAAL))
```

Adds the campus variable:

```
dfSSPgradesCPH["Campus"]<-"CPH"
dfSSPgradesAAL["Campus"]<-"AAL"
dfSSPanswersCPH["Campus"]<-"CPH"
dfSSPanswersAAL["Campus"]<-"AAL"
```

Removes the last row for the dataset with grades (called overall).

```
dfSSPgradesAAL <- dfSSPgradesAAL[-nrow(dfSSPgradesAAL),]
dfSSPgradesCPH <- dfSSPgradesCPH[-nrow(dfSSPgradesCPH),]
```

Combines data from AAL and CPH. The two variables should have the same number of observations, otherwise something went wrong in the previous steps.

```
names(dfSSPgradesAAL) <- names(dfSSPgradesCPH) # debugging: weird column names
dfSSPgrades <- rbind(dfSSPgradesAAL,dfSSPgradesCPH)
dfSSPanswers <- rbind(dfSSPanswersAAL,dfSSPanswersCPH)
```

Combines Surname with First name

```
dfSSPgrades["Name"] <- paste(dfSSPgrades$'First name', dfSSPgrades$Surname)
dfSSPanswers["Name"] <- paste(dfSSPanswers$'First name', dfSSPanswers$Surname)
```

Removes unnecessary variables, i.e. Surname, First name, Institution, and Department columns:

```
dfSSPgrades <- dfSSPgrades[ ,-c(1:4)]
dfSSPanswers <- dfSSPanswers[ ,-c(1:4)]
```

Replaces all values with NA for questions about study/working hours:

```
dfSSPgrades$`Q. 93 /0.09` <- NA
dfSSPgrades$`Q. 95 /0.09` <- NA
dfSSPgrades$`Q. 96 /0.09` <- NA
```

### Data Aanlysis

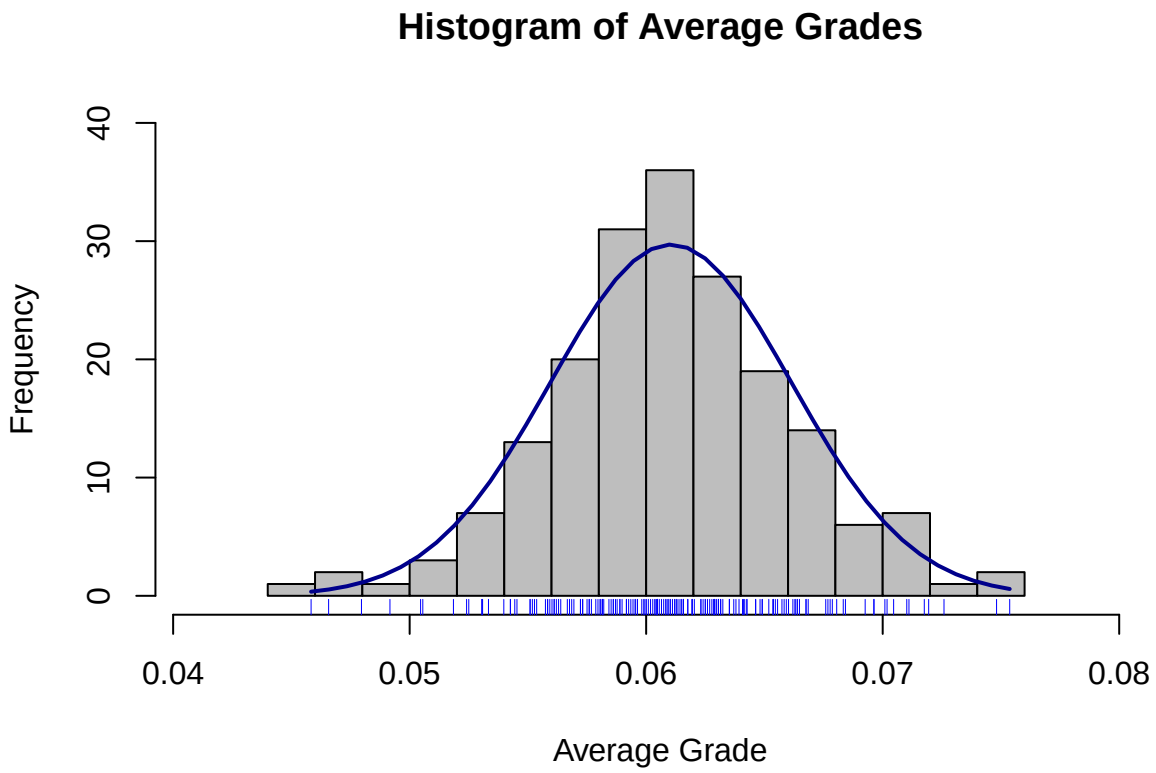
Creates a data frame with only the grades and computes the average grades for each student:

```
dfSSPgradesStat <- data.frame(dfSSPgrades[7:117])
dfSSPgradesStat["Mean"] <- rowMeans(dfSSPgradesStat[1:111], na.rm=TRUE)
```

### Overview with Hisrograms

Plotting a histogram with normal curve of the average grades:

```
x <- dfSSPgradesStat$Mean
h<-hist(x, breaks=(nrow(dfSSPgradesStat))/10, col="grey", xlab="Average Grade",
  main="Histogram of Average Grades", xlim=c((min(dfSSPgradesStat$Mean)-0.005),
  (max(dfSSPgradesStat$Mean)+0.005)), ylim=c(0,40))
xfit<-seq(min(x),max(x),length=40)
yfit<-dnorm(xfit,mean=mean(x),sd=sd(x))
yfit <- yfit*diff(h$mids[1:2])*length(x)
lines(xfit, yfit, col="darkblue", lwd=2)
rug(dfSSPgradesStat$Mean,col="blue")
```




```
#points(density(dfSSPgradesStat$Mean),type="l",col="red")
```

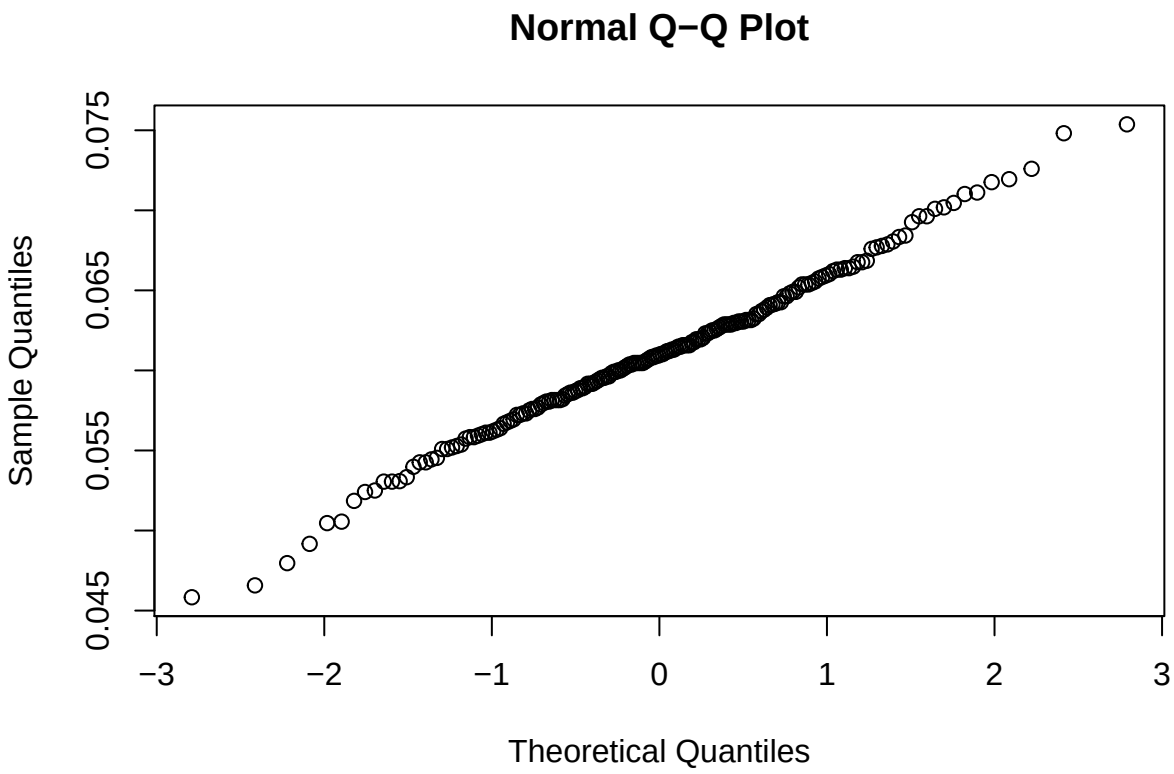
Descriptive statistics - summary of average grade:

```
summary(dfSSPgradesStat$Mean)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.04583 0.05806 0.06097 0.06105 0.06410 0.07537
```

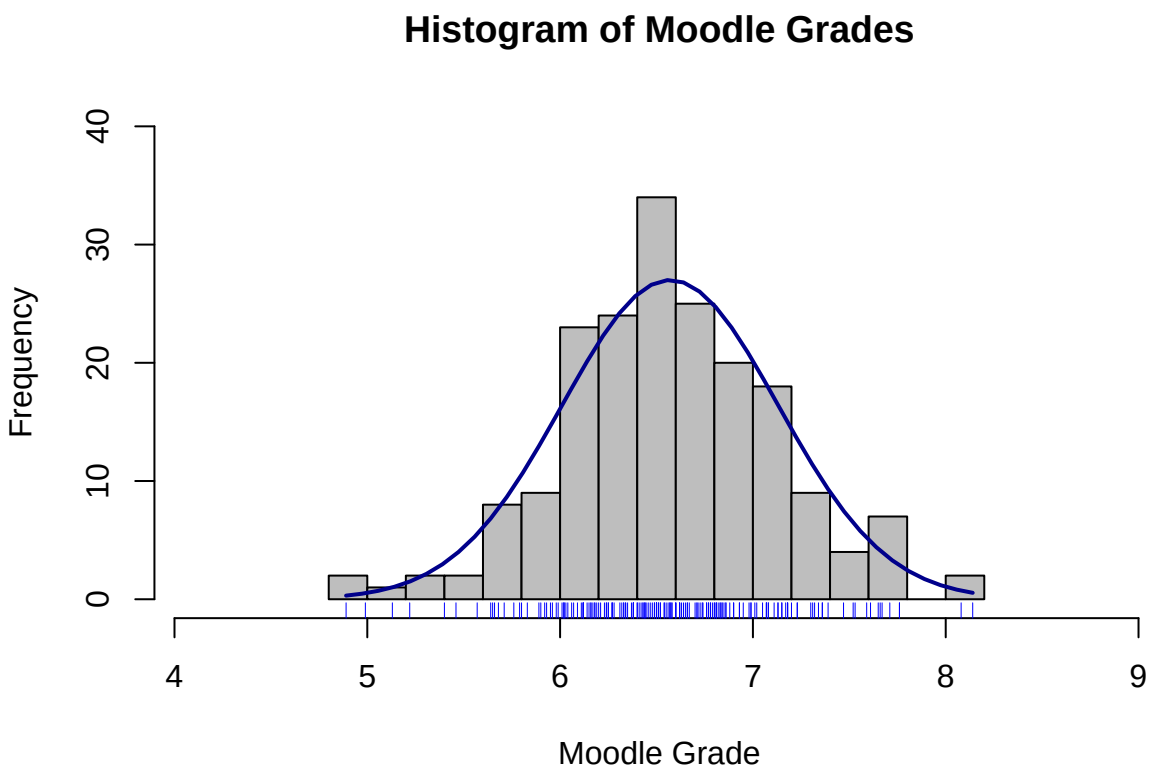
QQ-plot of the average grades:

```
qqnorm(dfSSPgradesStat$Mean)
```




Plotting a histogram with normal curve of the mean grade computed by Moodle (in comparison):

```
x <- dfSSPgrades$`Grade/10.00`
h<-hist(x, breaks=(nrow(dfSSPgrades))/10, col="grey", xlab="Moodle Grade",
  main="Histogram of Moodle Grades", ylim=c(0,40),
  xlim=c((min(dfSSPgrades$`Grade/10.00`)-0.8), (max(dfSSPgrades$`Grade/10.00`)+0.8)))
xfit<-seq(min(x),max(x),length=40)
yfit<-dnorm(xfit,mean=mean(x),sd=sd(x))
yfit <- yfit*diff(h$mids[1:2])*length(x)
lines(xfit, yfit, col="darkblue", lwd=2)
rug(dfSSPgrades$`Grade/10.00`,col="blue")
```




```
#points(density(dfSSPgrades$`Grade/10.00`),type="l",col="red")
```

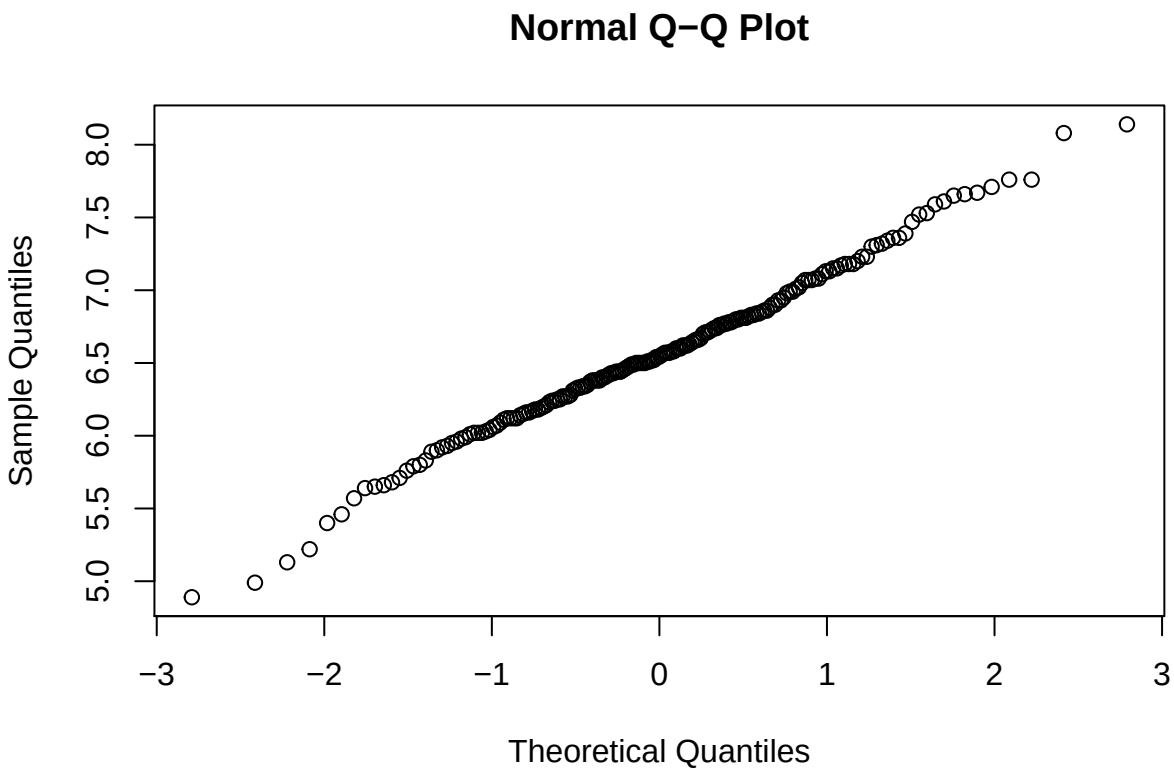
Descriptive statistics - summary of average grade:

```
summary(dfSSPgrades$`Grade/10.00`)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   4.890   6.215   6.545   6.571   6.895   8.140
```

QQ-plot of the Moodle grades:

```
qqnorm(dfSSPgrades$`Grade/10.00`)
```




TODO: compute avg, sd,

**Individual Student**

TODO: Compute grades statistics (avg, percent rank, frequency..) for each student

## work with this!

quantile(dfSSPgradesStat′*Q*..2..0.09′)*length*(*dfSSPgradesStat*'Q..2..0.09'[dfSSPgradesStat$‘Q..2..0.09’ <= 0.02]) / length(dfSSPgradesStat$‘Q..2..0.09’) * 100

## TODO: Compute grades statistics(avg, percent rank, frequency) for each category

dfSSPgradesStat[“DemographicsMean”] <- rowMeans(dfSSPgradesStat[1:9], na.rm=TRUE) dfSSP-gradesStat[“AttitudeMean”] <- rowMeans(dfSSPgradesStat[10:15], na.rm=TRUE) dfSSPgrades-Stat[“ReasonsMean”] <- rowMeans(dfSSPgradesStat[16:23], na.rm=TRUE) dfSSPgradesStat[“ChoiceMean”] <- rowMeans(dfSSPgradesStat[24:33], na.rm=TRUE) dfSSPgradesStat[“HSBehaveMean”] <- rowMeans(dfSSPgradesStat[34:45] na.rm=TRUE) dfSSPgradesStat[“HSTrustMean”] <- rowMeans(dfSSPgradesStat[46:50], na.rm=TRUE) dfSSPgradesStat[“BelongingMean”] <- rowMeans(dfSSPgradesStat[51:56], na.rm=TRUE) dfSSPgrades-Stat[“GritMean”] <- rowMeans(dfSSPgradesStat[57:61], na.rm=TRUE) dfSSPgradesStat[“GrowthMean”] <- rowMeans(dfSSPgradesStat[62:64], na.rm=TRUE) dfSSPgradesStat[“ControlMean”] <- rowMeans(dfSSPgradesStat[65:74],

na.rm=TRUE) dfSSPgradesStat[“TraitsMean”] <- rowMeans(dfSSPgradesStat[75:87], na.rm=TRUE) dfSSPgradesStat[“AcademicMean”] <- rowMeans(dfSSPgradesStat[88:92], na.rm=TRUE) dfSSPgradesStat[“HoursMean”] <- rowMeans(dfSSPgradesStat[93:97], na.rm=TRUE) dfSSPgradesStat[“MedialogyMean”] <- rowMeans(dfSSPgradesStat[98:11], na.rm=TRUE)

dfSSPgradesClassMeans <- sqldf(“select DemographicsMean, AttitudeMean, ReasonsMean, ChoiceMean, HSBehaveMean, HSTrustMean, BelongingMean, GritMean, GrowthMean, ControlMean, TraitsMean, AcademicMean, HoursMean, MedialogyMean from dfSSPgradesStat”)

## TODO: Perform one tailed t-test

## TODO: General summery and individual summary compared to the cohort.

## write.csv(dfSSPgrades,file = “SSPgrades.csv”)