LAB 2

Michael Kummer

Setup

Start by installing the necessary packages. In Lab 1, we installed "data.table". Today we install two packages: **ggplot2** and **stargazer**. Once installed, you activate the packages using "the function"library":

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.1.3
```

```
library(stargazer)
```

```
##
## Please cite as:
##  Hlavac, Marek (2022). stargazer: Well-Formatted Regression and Summary Statistics Tables.
##  R package version 5.2.3. https://CRAN.R-project.org/package=stargazer
```

```
library(data.table)
```

```
## Warning: package 'data.table' was built under R version 4.1.3
```

Then set your working directory.

```
setwd("C:/TopicsDig/Labs")
# change the file's path to your own
```

Data Analysis

We will use the same data set from Lab 1 to compute descriptive statistics. To start, load Lab 1's data set and convert it to the data.table format:

```
load("C:/TopicsDig/Labs/datasets/ceosal2.RData")
dt.ceo.salaries <- data.table(data)
rm(data)
```

Descriptive Statistics

How many CEOS are in the sample?

As there are only CEOS in this data set, we can count the number of CEOS by simply counting the number of observations in the data set.

```
nrow(dt.ceo.salaries)
```

```
## [1] 177
```

How many CEOS have a graduate degree?

Whenever a CEO has a graduate degree the variable "grad" takes the value 1 (0 otherwise). Thus, to count how many CEOs have a graduate degree we can simply compute the sum of the variable "grad".

```
dt.ceo.salaries[, sum(grad)]
```

```
## [1] 94
```

Alternatively, we can count the number of rows in which the variable grad takes the value of 1.

```
nrow(dt.ceo.salaries[grad==1,])
```

```
## [1] 94
```

What is the percentage of CEOs with graduate degrees?

```
dt.ceo.salaries[, sum(grad)]/nrow(dt.ceo.salaries)
```

```
## [1] 0.5310734
```

Or we can simply calculate the mean of grad:

```
dt.ceo.salaries[, mean(grad)]
```

```
## [1] 0.5310734
```

What is the average CEO salary?

```
dt.ceo.salaries[, mean(salary)]
```

```
## [1] 865.8644
```

Alternatively:

```
mean(dt.ceo.salaries[, salary])
```

```
## [1] 865.8644
```

What is the mean CEO salary for those with a graduate degree?

```
dt.ceo.salaries[grad==1, mean(salary)]
```

```
## [1] 864.2128
```

What is the mean CEO salary for those without a graduate degree?

```
dt.ceo.salaries[grad==0, mean(salary)]
```

```
## [1] 867.7349
```

How many CEOs are have/don't have a college degree?

```
dt.ceo.salaries[ , list(n_ceo=.N), by = college]
```

```
##    college n_ceo
## 1:       1   172
## 2:       0     5
```

Can we say that the mean salary is statistically different from 800?

```
t.test(dt.ceo.salaries[, salary], mu = 800)
```

```
##
##  One Sample t-test
##
## data:  dt.ceo.salaries[, salary]
## t = 1.4913, df = 176, p-value = 0.1377
```

```
## alternative hypothesis: true mean is not equal to 800
## 95 percent confidence interval:
##  778.7015 953.0274
## sample estimates:
## mean of x
##  865.8644
```

The p-value is greater than 0.05 so we cannot reject the null that the population mean is 800. You can also see that 800 is contained in the confidence interval.

Is the average salary different for CEOs with a graduate degree and those without?

```
t.test(dt.ceo.salaries[, salary] ~ dt.ceo.salaries[, grad])
```

```
##
##  Welch Two Sample t-test
##
## data:  dt.ceo.salaries[, salary] by dt.ceo.salaries[, grad]
## t = 0.038973, df = 149.94, p-value = 0.969
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
##  -175.0489  182.0932
## sample estimates:
## mean in group 0 mean in group 1
##        867.7349        864.2128
```

Alternatively:

```
dt.ceo.salaries[ , t.test (salary ~ grad)]
```

```
##
##  Welch Two Sample t-test
##
## data:  salary by grad
## t = 0.038973, df = 149.94, p-value = 0.969
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
##  -175.0489  182.0932
## sample estimates:
## mean in group 0 mean in group 1
##        867.7349        864.2128
```

Alternatively:

```
t.test(dt.ceo.salaries[grad==0, salary] , dt.ceo.salaries[grad==1, salary])
```

```
##
##  Welch Two Sample t-test
##
## data:  dt.ceo.salaries[grad == 0, salary] and dt.ceo.salaries[grad == 1, salary]
## t = 0.038973, df = 149.94, p-value = 0.969
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -175.0489  182.0932
## sample estimates:
## mean of x mean of y
##  867.7349  864.2128
```

The p-value is greater than 0.05 so we cannot reject the null that the two population means are the same. You can also see that the value 0 (no difference between means) is contained in the confidence interval.

Creating a table with descriptive statistics:

```
dt.ceo.salaries[, list(  mean_salary = mean(salary)
                       , sd_salary = sd(salary)
                       , min_salary =  min(salary)
                       , max_salary = max(salary)
                       , median_salary = median(salary))]
```

```
##    mean_salary sd_salary min_salary max_salary median_salary
## 1:    865.8644  587.5893        100       5299           707
```

You can also compute the summary statistics for different groups:

```
dt.ceo.salaries[, list(  mean_salary = mean(salary)
                       , sd_salary = sd(salary)
                       , min_salary =  min(salary)
                       , max_salary = max(salary)), by = list(grad, college)]
```

```
##    grad college mean_salary sd_salary min_salary max_salary
## 1:    1       1    864.2128  501.3924        100       2265
## 2:    0       1    853.0897  679.0268        174       5299
## 3:    0       0   1096.2000  633.4569        300       1738
```

Alternatively, you can use the stargazer function to get summary statistics for all variables in your data set:

```
stargazer(dt.ceo.salaries, type = "text")
```

```
##
## ================================================
## Statistic  N    Mean    St. Dev.    Min     Max
## ------------------------------------------------
## salary    177  865.864   587.589    100    5,299
## age       177  56.429     8.422      33      86
## college   177   0.972     0.166      0       1
## grad      177   0.531     0.500      0       1
## comten    177  22.503    12.295      2       58
## ceoten    177   7.955     7.151      0       37
## sales     177 3,529.463 6,088.654    29    51,300
## profits   177  207.831   404.454    -463   2,700
## mktval    177 3,600.316 6,442.276   387    45,400
## lsalary   177   6.583     0.606    4.605   8.575
## lsales    177   7.231     1.432    3.367   10.845
## lmktval   177   7.399     1.133    5.958   10.723
## comtensq  177  656.684   577.123     4     3,364
## ceotensq  177  114.124   212.566     0     1,369
## profmarg  177   6.420    17.861   -203.077 47.458
## ------------------------------------------------
```

You can also get summary statistics for a subset of your observations and for a specific list of variables.

```
stargazer(dt.ceo.salaries[grad==1, list(age, salary)], type = "text")
```

```
##
## ======================================
## Statistic N   Mean   St. Dev. Min  Max
## --------------------------------------
```

```
## age       94 55.457   8.155   38   86
## salary    94 864.213 501.392  100 2,265
## ----------------------------------------
```

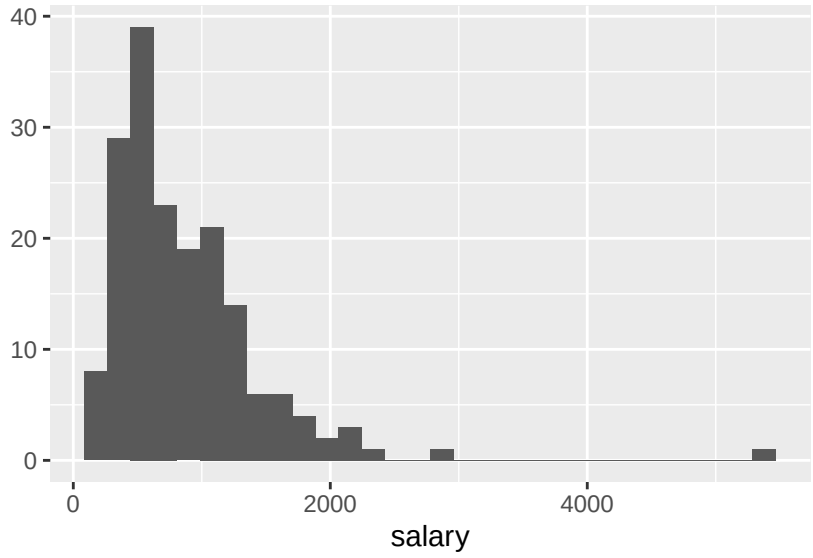
Quick plots

Histogram

Salary

```
qplot(   data = dt.ceo.salaries
       , x = salary
       , geom = "histogram")
```

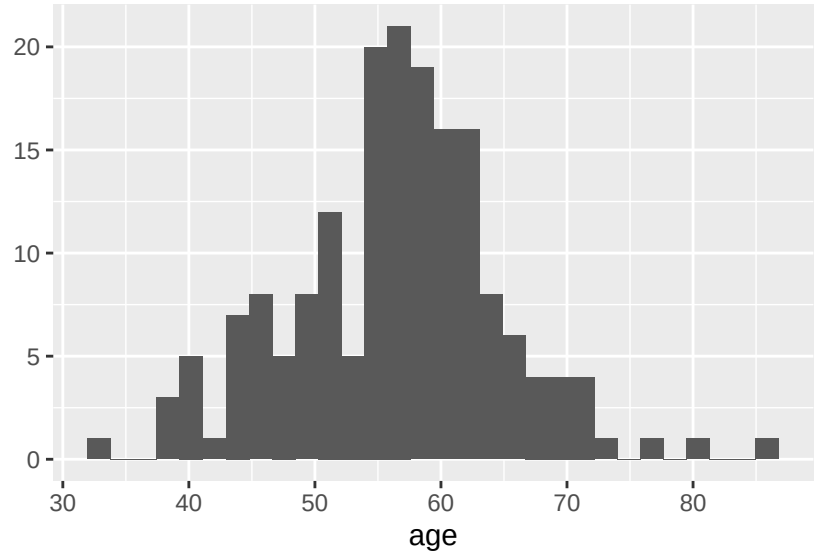
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Age

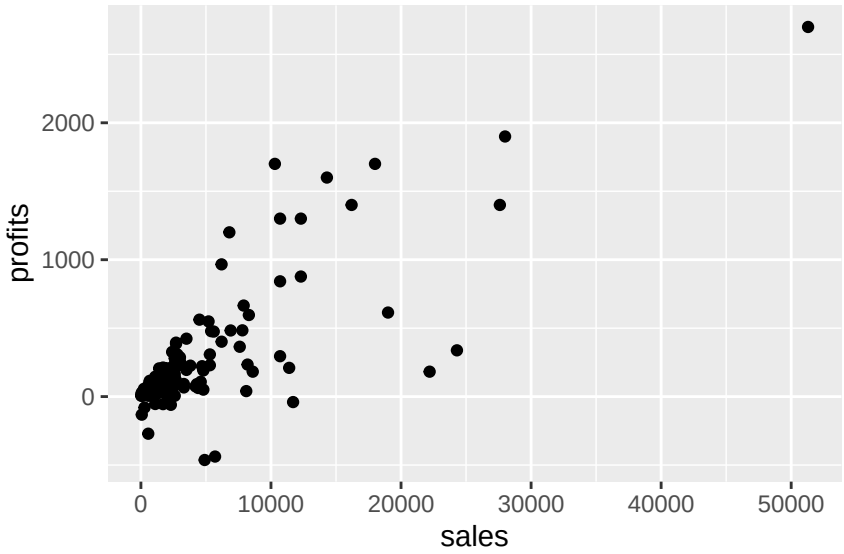
```
qplot(   data = dt.ceo.salaries
       , x = age
       , geom = "histogram")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Scatterplot

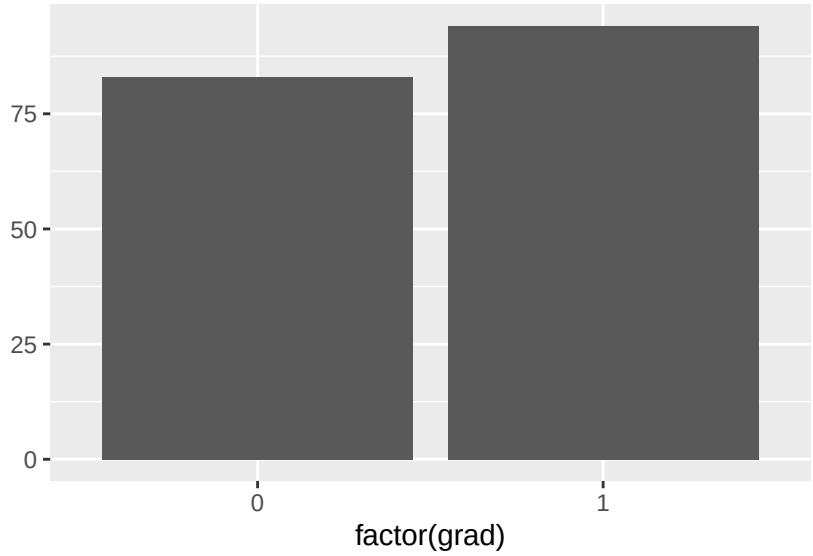
```
qplot(   data = dt.ceo.salaries
       , x = sales
       , y = profits
       , geom = "point")
```



Barplot

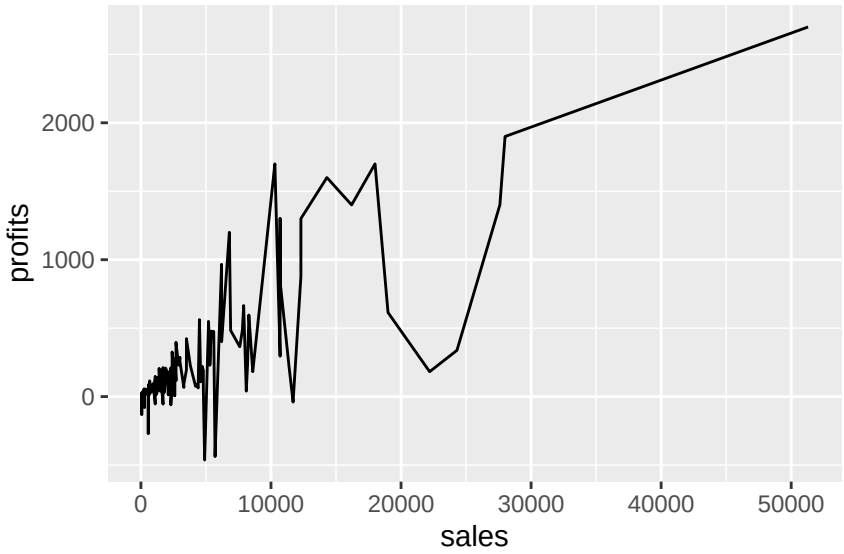
Graduate

```
qplot(   data = dt.ceo.salaries
       , x = factor(grad)
       , geom = "bar")
```



Line

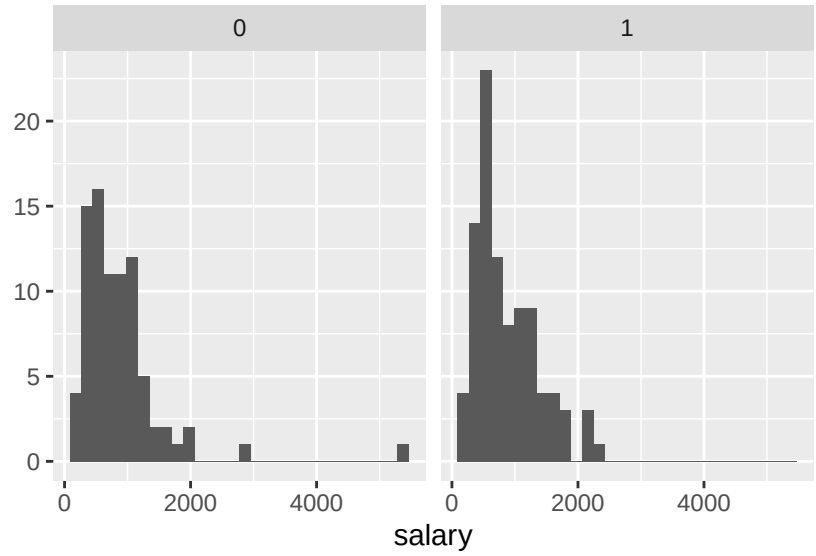
```
qplot(   data = dt.ceo.salaries
       , x = sales
       , y = profits
       , geom = "line")
```



Facet Wrap

```
qplot(   data = dt.ceo.salaries
       , x = salary
       , geom = "histogram") + facet_wrap(~ grad)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Customizing plots:

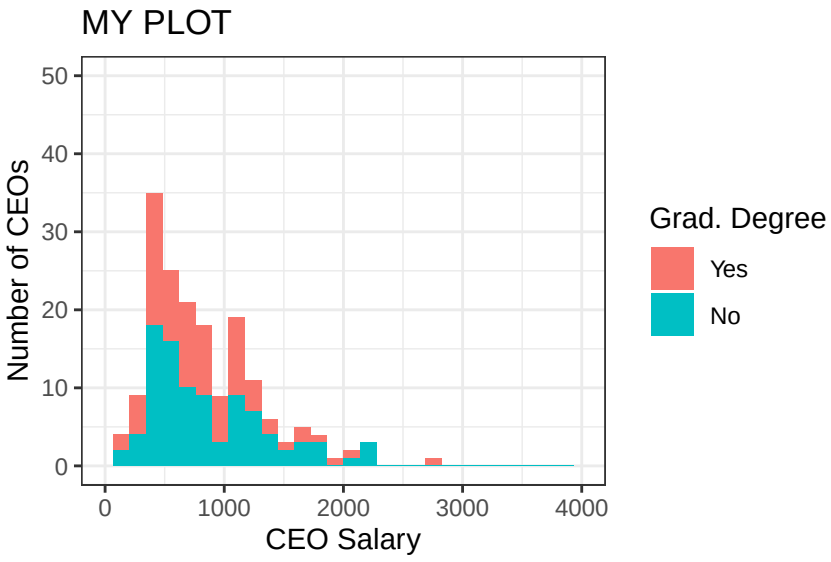
Salary

```
qplot(   data = dt.ceo.salaries
       , x = salary
       , geom = "histogram"
       ,  fill = factor(grad, levels = c(0,1), labels = c("Yes", "No"))) +
  theme_bw() +
  ylim(0,50) +
  xlim(0, 4000) +
  labs( title = "MY PLOT", x = "CEO Salary", y = "Number of CEOs", fill = "Grad. Degree")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 1 rows containing non-finite values (stat_bin).
```

```
## Warning: Removed 4 rows containing missing values (geom_bar).
```



Acknowledgements and Thanks:

This lab is based on material by M Godinho de Matos, R Belo and F Reis. Gratefully acknowledged!