ADAPT Program Day 3

# Loading packages

library(readxl)  
library(dplyr)

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

library(tidyr)  
library(ggplot2)  
library(reshape2)

##   
## Attaching package: 'reshape2'

## The following object is masked from 'package:tidyr':  
##   
## smiths

library(stringr)  
library(lubridate)

##   
## Attaching package: 'lubridate'

## The following object is masked from 'package:base':  
##   
## date

library(data.table) #This is to use fread instead of read.csv (faster for big files)

##   
## Attaching package: 'data.table'

## The following objects are masked from 'package:lubridate':  
##   
## hour, isoweek, mday, minute, month, quarter, second, wday,  
## week, yday, year

## The following objects are masked from 'package:reshape2':  
##   
## dcast, melt

## The following objects are masked from 'package:dplyr':  
##   
## between, first, last

# Reading files

df\_Data <- fread(file="./data/ADAPT\_data\_for\_day3.csv", stringsAsFactors = F)  
  
Port\_codes <- read\_xlsx(path="./data/Portcodes Dictionary.xlsx")

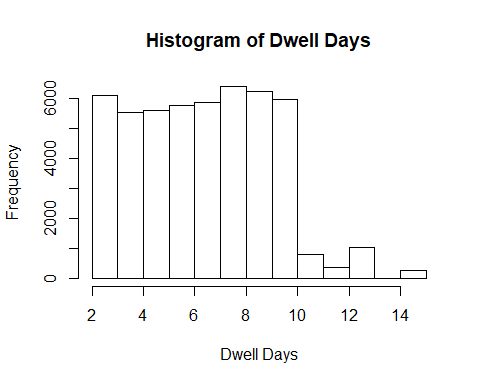
# Transformations for plotting

df\_Data$CNTR\_FOE <- as.factor(df\_Data$CNTR\_FOE)

# Some visualizations

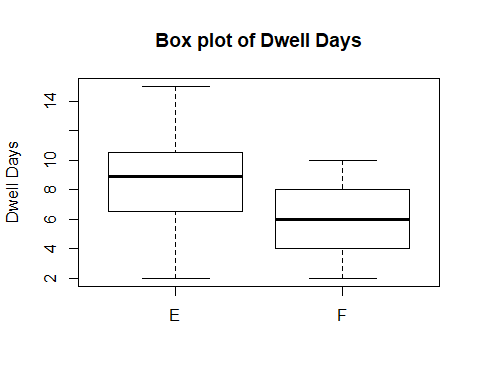
## Histogram

hist(df\_Data$DWELL\_DAYS, main="Histogram of Dwell Days", xlab="Dwell Days")



## Boxplot

boxplot(df\_Data$DWELL\_DAYS ~ df\_Data$CNTR\_FOE, main="Box plot of Dwell Days", ylab="Dwell Days")



# Working with the information

## Advanced Data Summarizing

df\_Data %>% group\_by(POL) %>% summarise(AVG\_WGT=mean(CNTR\_WGT\_KG)) %>%  
 arrange(AVG\_WGT) %>% top\_n(5,wt=AVG\_WGT)

## # A tibble: 5 x 2  
## POL AVG\_WGT  
## <chr> <dbl>  
## 1 GTPRQ 31900  
## 2 EETLL 32200  
## 3 ARPMY 32800  
## 4 ARZAE 33000  
## 5 NOAES 34600

## Data Agreggation

df\_Data %>% group\_by(POL,CNTR\_FOE) %>% summarise(AVG\_WGT=mean(CNTR\_WGT\_KG))

## # A tibble: 660 x 3  
## # Groups: POL [433]  
## POL CNTR\_FOE AVG\_WGT  
## <chr> <fct> <dbl>  
## 1 "" E 3671.  
## 2 "" F 19842.  
## 3 AEAUH E 3212.  
## 4 AEAUH F 25595.  
## 5 AEDXB E 3400   
## 6 AEDXB F 24370   
## 7 AEJEA E 3431.  
## 8 AEJEA F 22401.  
## 9 AEKHL E 3795   
## 10 AEKHL F 16330   
## # ... with 650 more rows

## Data restructuring

df\_Data %>% group\_by(POL,CNTR\_FOE) %>% summarise(AVG\_WGT=mean(CNTR\_WGT\_KG)) %>%  
 spread(CNTR\_FOE, AVG\_WGT)

## # A tibble: 433 x 3  
## # Groups: POL [433]  
## POL E F  
## <chr> <dbl> <dbl>  
## 1 "" 3671. 19842.  
## 2 AEAUH 3212. 25595.  
## 3 AEDXB 3400 24370   
## 4 AEJEA 3431. 22401.  
## 5 AEKHL 3795 16330   
## 6 AEKLF 2790 NA   
## 7 ALDRZ NA 26200   
## 8 AOLAD 2967. 27389.  
## 9 AOLOB 2110 NA   
## 10 AOLUA 2629. 23032.  
## # ... with 423 more rows

## Data subsetting

df\_IDBUN <- df\_Data %>% filter(POL=="IDBUN")  
head(df\_IDBUN)

## V1 ID CNTR\_N CNTR\_OPR\_C CNTR\_FOE PURPOSE\_C CNTR\_LENGTH  
## 1 251 30254 PCIU 1027618 PI F T 20  
## 2 668 30671 HAMU 1105799 HL F T 20  
## 3 699 30702 NYKU 3169565 ON F T 20  
## 4 732 30735 TGBU 2152590 HL F T 20  
## 5 983 30986 TEMU 4255514 AP F T 20  
## 6 1017 31020 PCIU 2599346 PI F T 20  
## CNTR\_TYPE\_C CNTR\_WGT\_KG POL POD DEST\_PORT\_C DISC\_VSL\_NAME  
## 1 GP 22440 IDBUN CNNGB CNNGB MARCOPOLO292  
## 2 GP 19189 IDBUN EGDAM EGALY MARCOPOLO294  
## 3 GP 22440 IDBUN CNSHA CNJIU MARCOPOLO296  
## 4 GP 16272 IDBUN EGDAM EGALY MARCOPOLO294  
## 5 GP 17693 IDBUN PKKH9 PKKHI MARCOPOLO888  
## 6 GP 22481 IDBUN BDCGP BDCGP   
## DISC\_SERVICE\_C LOAD\_VSL\_NAME LOAD\_SERVICE\_C LOAD\_DT  
## 1 IDO KOTA PERKASA RS2 20181012063518  
## 2 IDO AL RAWDAH NEN 20181127191704  
## 3 IDO UMM SALAL AGX 20181209233702  
## 4 IDO NA  
## 5 IDO MS SALINA ASC 20180814230457  
## 6 KOTA AZAM CCS 20181002110855  
## DISC\_DT GATE\_OUT\_DT GATE\_IN\_DT BATCH\_ID LOGISTICS\_PARTNER  
## 1 20181006145516 <NA> <NA> 201810 <NA>  
## 2 20181121223011 20181124165209 20181124170116 201811 <NA>  
## 3 20181205204710 20181207052758 20181207053755 201812 <NA>  
## 4 20181121231623 20181127141014 20181127142018 201811 <NA>  
## 5 20180807201410 <NA> <NA> 201808 <NA>  
## 6 NA <NA> <NA> 201810 <NA>  
## PRODUCT COMMODITY CARGO\_OWNER RF\_TEMP  
## 1 Greenling Sea Food Guangzhou Automobile Industry Group NA  
## 2 Snail Sea Food Mitsubishi NA  
## 3 Pomegranate Fruits China Baowu Steel Group NA  
## 4 Peacock Bass Sea Food Tata Motors NA  
## 5 Kiwi Fruits IBM NA  
## 6 Snail Sea Food Mitsubishi NA  
## DWELL\_DAYS CNTR\_HEIGHT dt\_LOAD\_DT dt\_DISC\_DT  
## 1 9.8 9'6"" 2018-10-12 06:35:18 2018-10-06 14:55:16  
## 2 5.4 8'6"" 2018-11-27 19:17:04 2018-11-21 22:30:11  
## 3 4.3 8'6"" 2018-12-09 23:37:02 2018-12-05 20:47:10  
## 4 8.9 8'6"" <NA> 2018-11-21 23:16:23  
## 5 2.9 8'6"" 2018-08-14 23:04:57 2018-08-07 20:14:10  
## 6 7.1 8'6"" 2018-10-02 11:08:55 <NA>  
## dt\_TRANSACTION CNTR\_HEIGHT\_CONV  
## 1 2018-10-06 9.6  
## 2 2018-11-21 8.6  
## 3 2018-12-05 8.6  
## 4 2018-11-21 8.6  
## 5 2018-08-07 8.6  
## 6 2018-10-02 8.6

## Data subsetting

df\_Data %>% filter(grepl("BUN",POL)) %>% head()

## V1 ID CNTR\_N CNTR\_OPR\_C CNTR\_FOE PURPOSE\_C CNTR\_LENGTH  
## 1 251 30254 PCIU 1027618 PI F T 20  
## 2 668 30671 HAMU 1105799 HL F T 20  
## 3 699 30702 NYKU 3169565 ON F T 20  
## 4 732 30735 TGBU 2152590 HL F T 20  
## 5 983 30986 TEMU 4255514 AP F T 20  
## 6 1017 31020 PCIU 2599346 PI F T 20  
## CNTR\_TYPE\_C CNTR\_WGT\_KG POL POD DEST\_PORT\_C DISC\_VSL\_NAME  
## 1 GP 22440 IDBUN CNNGB CNNGB MARCOPOLO292  
## 2 GP 19189 IDBUN EGDAM EGALY MARCOPOLO294  
## 3 GP 22440 IDBUN CNSHA CNJIU MARCOPOLO296  
## 4 GP 16272 IDBUN EGDAM EGALY MARCOPOLO294  
## 5 GP 17693 IDBUN PKKH9 PKKHI MARCOPOLO888  
## 6 GP 22481 IDBUN BDCGP BDCGP   
## DISC\_SERVICE\_C LOAD\_VSL\_NAME LOAD\_SERVICE\_C LOAD\_DT  
## 1 IDO KOTA PERKASA RS2 20181012063518  
## 2 IDO AL RAWDAH NEN 20181127191704  
## 3 IDO UMM SALAL AGX 20181209233702  
## 4 IDO NA  
## 5 IDO MS SALINA ASC 20180814230457  
## 6 KOTA AZAM CCS 20181002110855  
## DISC\_DT GATE\_OUT\_DT GATE\_IN\_DT BATCH\_ID LOGISTICS\_PARTNER  
## 1 20181006145516 <NA> <NA> 201810 <NA>  
## 2 20181121223011 20181124165209 20181124170116 201811 <NA>  
## 3 20181205204710 20181207052758 20181207053755 201812 <NA>  
## 4 20181121231623 20181127141014 20181127142018 201811 <NA>  
## 5 20180807201410 <NA> <NA> 201808 <NA>  
## 6 NA <NA> <NA> 201810 <NA>  
## PRODUCT COMMODITY CARGO\_OWNER RF\_TEMP  
## 1 Greenling Sea Food Guangzhou Automobile Industry Group NA  
## 2 Snail Sea Food Mitsubishi NA  
## 3 Pomegranate Fruits China Baowu Steel Group NA  
## 4 Peacock Bass Sea Food Tata Motors NA  
## 5 Kiwi Fruits IBM NA  
## 6 Snail Sea Food Mitsubishi NA  
## DWELL\_DAYS CNTR\_HEIGHT dt\_LOAD\_DT dt\_DISC\_DT  
## 1 9.8 9'6"" 2018-10-12 06:35:18 2018-10-06 14:55:16  
## 2 5.4 8'6"" 2018-11-27 19:17:04 2018-11-21 22:30:11  
## 3 4.3 8'6"" 2018-12-09 23:37:02 2018-12-05 20:47:10  
## 4 8.9 8'6"" <NA> 2018-11-21 23:16:23  
## 5 2.9 8'6"" 2018-08-14 23:04:57 2018-08-07 20:14:10  
## 6 7.1 8'6"" 2018-10-02 11:08:55 <NA>  
## dt\_TRANSACTION CNTR\_HEIGHT\_CONV  
## 1 2018-10-06 9.6  
## 2 2018-11-21 8.6  
## 3 2018-12-05 8.6  
## 4 2018-11-21 8.6  
## 5 2018-08-07 8.6  
## 6 2018-10-02 8.6

## Random data subsetting

sample1 <- sample\_n(df\_Data,5) #Number of row  
sample2 <- sample\_frac(df\_Data,0.05) #Percentage of the dataset  
  
nrow(sample1)

## [1] 5

nrow(sample2)

## [1] 2500

## Merging data frames

#Merge for POL  
df\_Data\_names <- left\_join(df\_Data,Port\_codes,by=c("POL"="PORT\_CODE"))  
names(df\_Data\_names)[33:36] <- paste0("POL\_",names(df\_Data\_names)[33:36])  
  
#Merge for POD  
df\_Data\_names <- left\_join(df\_Data\_names,Port\_codes,by=c("POD"="PORT\_CODE"))  
names(df\_Data\_names)[37:40] <- paste0("POD\_",names(df\_Data\_names)[37:40])  
  
df\_Data\_names[,c("POL","POL\_CITY\_NAME","POD","POD\_CITY\_NAME")] %>% head(10)

## POL POL\_CITY\_NAME POD POD\_CITY\_NAME  
## 1 PKKHI Karachi,PAKISTAN <NA>  
## 2 INKAT KATTUPALLI,INDIA CGPNR POINTE NOIRE,CONGO  
## 3 KRPUS PUSAN,KOREA AUBNE BRISBANE,AUSTRALIA  
## 4 MYWSP PORT KLANG, Malaysia PKKHI Karachi,PAKISTAN  
## 5 KRPUS PUSAN,KOREA INCCU KOLKATA (EX CALCUTTA),INDIA  
## 6 ITGOA GENOA,ITALY IDJKT JAKARTA, indonesia  
## 7 SAJUB JUBAIL,SAUDI ARABIA VNSGN HO CHI MINH CITY,VIET NAM  
## 8 IDBLW BELAWAN, SUMATRA NLRTM ROTTERDAM,NETHERLANDS  
## 9 CNSHK SHEKOU,China ZADUR DURBAN,SOUTH AFRICA  
## 10 AUMEL MELBOURNE,AUSTRALIA IDSUB TANJUNG PERAK, Java

names(df\_Data\_names)

## [1] "V1" "ID" "CNTR\_N"   
## [4] "CNTR\_OPR\_C" "CNTR\_FOE" "PURPOSE\_C"   
## [7] "CNTR\_LENGTH" "CNTR\_TYPE\_C" "CNTR\_WGT\_KG"   
## [10] "POL" "POD" "DEST\_PORT\_C"   
## [13] "DISC\_VSL\_NAME" "DISC\_SERVICE\_C" "LOAD\_VSL\_NAME"   
## [16] "LOAD\_SERVICE\_C" "LOAD\_DT" "DISC\_DT"   
## [19] "GATE\_OUT\_DT" "GATE\_IN\_DT" "BATCH\_ID"   
## [22] "LOGISTICS\_PARTNER" "PRODUCT" "COMMODITY"   
## [25] "CARGO\_OWNER" "RF\_TEMP" "DWELL\_DAYS"   
## [28] "CNTR\_HEIGHT" "dt\_LOAD\_DT" "dt\_DISC\_DT"   
## [31] "dt\_TRANSACTION" "CNTR\_HEIGHT\_CONV" "POL\_COUNTRY\_CODE"   
## [34] "POL\_PORT\_NAME" "POL\_COUNTRY\_NAME" "POL\_CITY\_NAME"   
## [37] "POD\_COUNTRY\_CODE" "POD\_PORT\_NAME" "POD\_COUNTRY\_NAME"   
## [40] "POD\_CITY\_NAME"

# Removing duplicates

## Founding the duplicated lines

df\_Data[duplicated(df\_Data[,c("CNTR\_N","CNTR\_WGT\_KG","POL","POD","LOAD\_VSL\_NAME","DISC\_VSL\_NAME")]),]

## V1 ID CNTR\_N CNTR\_OPR\_C CNTR\_FOE PURPOSE\_C CNTR\_LENGTH  
## 1: 45862 10873 TRLU 8044390 CC F T 40  
## CNTR\_TYPE\_C CNTR\_WGT\_KG POL POD DEST\_PORT\_C DISC\_VSL\_NAME  
## 1: GP 29760 THBKK SAJED SAJED GSS YANGON  
## DISC\_SERVICE\_C LOAD\_VSL\_NAME LOAD\_SERVICE\_C LOAD\_DT  
## 1: PRG COSCO FRANCE RS1 20180909205318  
## DISC\_DT GATE\_OUT\_DT GATE\_IN\_DT BATCH\_ID LOGISTICS\_PARTNER  
## 1: 20180901014708 <NA> <NA> 201809 <NA>  
## PRODUCT COMMODITY CARGO\_OWNER RF\_TEMP DWELL\_DAYS CNTR\_HEIGHT  
## 1: Clarias Fish Sea Food Mitsui NA 4.4 9'6""  
## dt\_LOAD\_DT dt\_DISC\_DT dt\_TRANSACTION CNTR\_HEIGHT\_CONV  
## 1: 2018-09-09 20:53:18 2018-09-01 01:47:08 2018-08-31 9.6

## Removing the duplicated lines

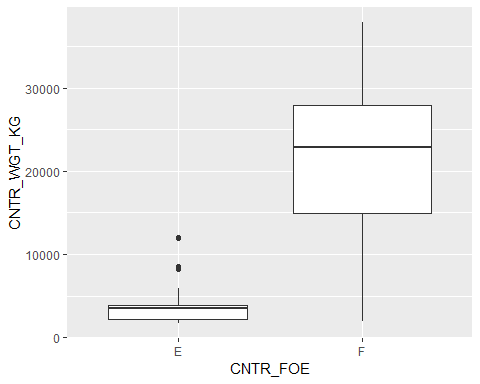
df\_Data\_unique <- df\_Data[!duplicated(df\_Data[,c("CNTR\_N","CNTR\_WGT\_KG","POL","POD","LOAD\_VSL\_NAME","DISC\_VSL\_NAME")]),]  
nrow(df\_Data\_unique)

## [1] 49989

# Using ggplot to make some graphics

## Boxplot

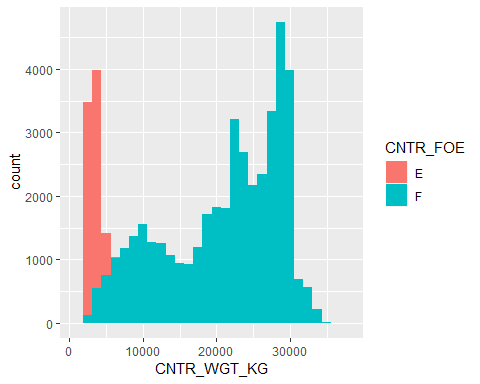
g <- ggplot(df\_Data,aes(CNTR\_FOE,CNTR\_WGT\_KG))  
g + geom\_boxplot()



## Histogram

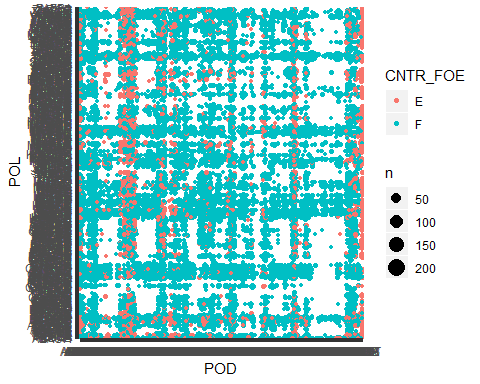
g <- ggplot(df\_Data,aes(CNTR\_WGT\_KG))  
g + geom\_histogram(aes(fill=CNTR\_FOE))

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



## Association plot

g <- ggplot(df\_Data,aes(POD,POL))  
g + geom\_count(aes(col=CNTR\_FOE))



# Descriptive Statistics

## Undestanding the structure of the dataset

### How many data items are there?

ncol(mtcars)

## [1] 11

### How many rows of data?

nrow(mtcars)

## [1] 32

### How many numerical and how many categorical data?

#Determining the quantity of diferent values of each column  
sapply(mtcars,function(x) {length(unique(x))})

## mpg cyl disp hp drat wt qsec vs am gear carb   
## 25 3 27 22 22 29 30 2 2 3 6

#Smaller figures could be factorized, the others are numerical data

### Transforming variables

mtcars2 <- within(mtcars, {  
 vs <- factor(vs, labels = c("V", "S"))  
 am <- factor(am, labels = c("automatic", "manual"))  
 cyl <- ordered(cyl)  
 gear <- ordered(gear)  
 carb <- ordered(carb)  
 })  
  
str(mtcars2)

## 'data.frame': 32 obs. of 11 variables:  
## $ mpg : num 21 21 22.8 21.4 18.7 18.1 14.3 24.4 22.8 19.2 ...  
## $ cyl : Ord.factor w/ 3 levels "4"<"6"<"8": 2 2 1 2 3 2 3 1 1 2 ...  
## $ disp: num 160 160 108 258 360 ...  
## $ hp : num 110 110 93 110 175 105 245 62 95 123 ...  
## $ drat: num 3.9 3.9 3.85 3.08 3.15 2.76 3.21 3.69 3.92 3.92 ...  
## $ wt : num 2.62 2.88 2.32 3.21 3.44 ...  
## $ qsec: num 16.5 17 18.6 19.4 17 ...  
## $ vs : Factor w/ 2 levels "V","S": 1 1 2 2 1 2 1 2 2 2 ...  
## $ am : Factor w/ 2 levels "automatic","manual": 2 2 2 1 1 1 1 1 1 1 ...  
## $ gear: Ord.factor w/ 3 levels "3"<"4"<"5": 2 2 2 1 1 1 1 2 2 2 ...  
## $ carb: Ord.factor w/ 6 levels "1"<"2"<"3"<"4"<..: 4 4 1 1 2 1 4 2 2 4 ...

### Getting quantiles of mpg

quantile(mtcars2$mpg, prob=c(0.15,0.25,0.35))

## 15% 25% 35%   
## 14.895 15.425 17.165

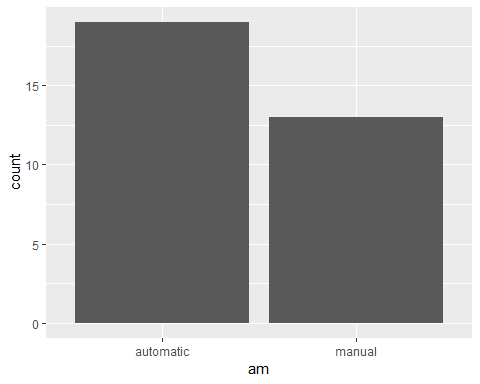
### Getting mode of am

i <- match(mtcars2$am,unique(mtcars2$am)) %>% tabulate() %>% which.max()  
unique(mtcars2$am)[i]

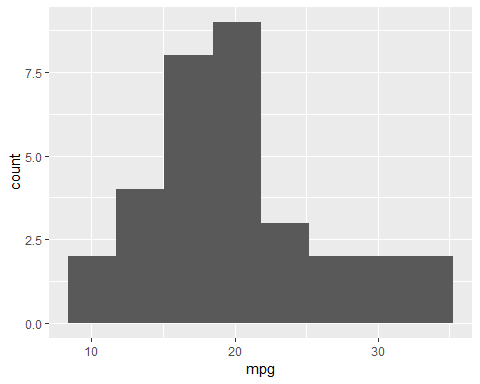
## [1] automatic  
## Levels: automatic manual

## Making different graphs

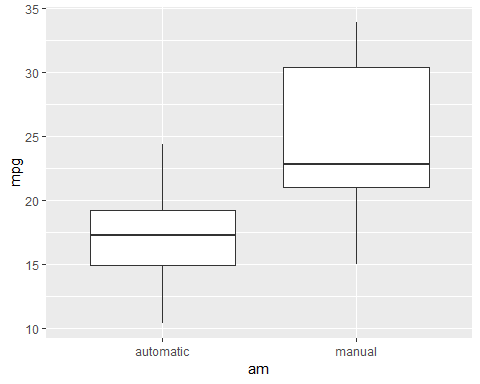
ggplot(mtcars2,aes(am)) + geom\_bar()



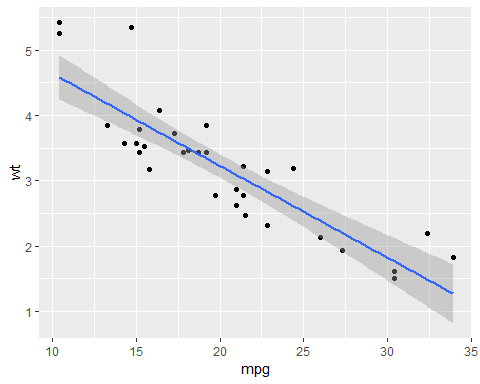
ggplot(mtcars2,aes(mpg)) + geom\_histogram(bins=8)



ggplot(mtcars2,aes(am,mpg)) + geom\_boxplot()



ggplot(mtcars2,aes(mpg,wt)) + geom\_point() + geom\_smooth(method="lm")



## Some statistics

### Variance

var(mtcars2$mpg)

## [1] 36.3241

### Standard deviation

sd(mtcars2$mpg)

## [1] 6.026948

### Skewness

ggplot(mtcars2,aes(mpg)) + geom\_density()



### Correlation

cor(mtcars2[,c("mpg","disp","hp","drat","wt","qsec")])

## mpg disp hp drat wt qsec  
## mpg 1.0000000 -0.8475514 -0.7761684 0.68117191 -0.8676594 0.41868403  
## disp -0.8475514 1.0000000 0.7909486 -0.71021393 0.8879799 -0.43369788  
## hp -0.7761684 0.7909486 1.0000000 -0.44875912 0.6587479 -0.70822339  
## drat 0.6811719 -0.7102139 -0.4487591 1.00000000 -0.7124406 0.09120476  
## wt -0.8676594 0.8879799 0.6587479 -0.71244065 1.0000000 -0.17471588  
## qsec 0.4186840 -0.4336979 -0.7082234 0.09120476 -0.1747159 1.00000000

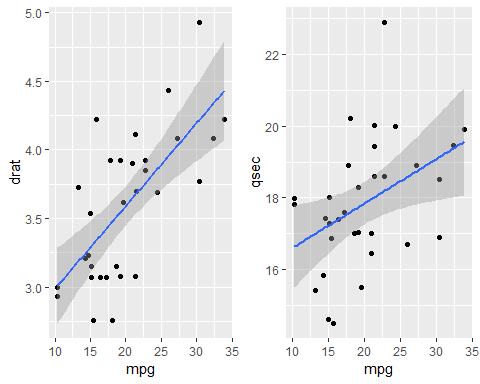
### Positive correlation

library(gridExtra)

##   
## Attaching package: 'gridExtra'

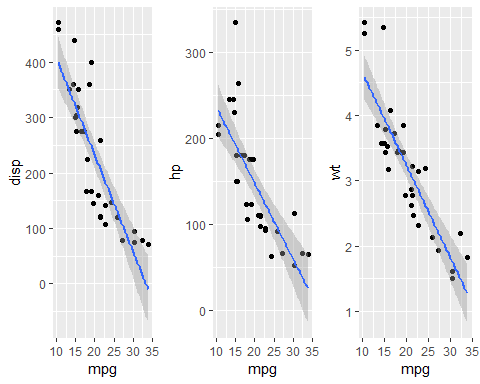
## The following object is masked from 'package:dplyr':  
##   
## combine

grid.arrange(ggplot(mtcars2,aes(mpg,drat)) + geom\_point() + geom\_smooth(method="lm"),  
 ggplot(mtcars2,aes(mpg,qsec)) + geom\_point() + geom\_smooth(method="lm"),nrow=1)



### Negative correlation

grid.arrange(ggplot(mtcars2,aes(mpg,disp)) + geom\_point() + geom\_smooth(method="lm"),  
 ggplot(mtcars2,aes(mpg,hp)) + geom\_point() + geom\_smooth(method="lm"),  
 ggplot(mtcars2,aes(mpg,wt)) + geom\_point() + geom\_smooth(method="lm"),nrow=1)



### Covariance

cov(mtcars2$mpg,mtcars2$wt)

## [1] -5.116685

### Correlation

cor(mtcars2$mpg,mtcars2$wt)

## [1] -0.8676594

### Correlation test

cor.test(mtcars2$mpg,mtcars2$wt, method = "pearson")

##   
## Pearson's product-moment correlation  
##   
## data: mtcars2$mpg and mtcars2$wt  
## t = -9.559, df = 30, p-value = 1.294e-10  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## -0.9338264 -0.7440872  
## sample estimates:  
## cor   
## -0.8676594