```
## Some examples

head(mtcars)



## Statistical summarization


mean(mtcars$hp)
quantile(mtcars$hp)


## Statistical summarization


median(mtcars$wt)
quantile(mtcars$wt, probs = 0.6)

## Statistical summarization

x = c(1,5,7,NA,4,2, 8,10,45,42)
mean(x)
mean(x,na.rm=TRUE)
quantile(x,na.rm=TRUE)


## Data Summarization on matrices/data frames


circ = read.csv("Charm_City_Circulator_Ridership.csv",
            header=TRUE,as.is=TRUE)


## Subsetting to specific columns


library(dplyr,quietly = TRUE)
circ2 = select(circ, date, day, ends_with("Average"))

## column and row means


avgs = select(circ2, ends_with("Average"))
colMeans(avgs,na.rm=TRUE)
circ2$daily = rowMeans(avgs,na.rm=TRUE)
head(circ2$daily)

## Summary

summary(circ2)


## Apply statements
```

```
apply(X, MARGIN, FUN, ...)

## Apply statements


apply(avgs,2,mean,na.rm=TRUE) # column means
apply(avgs,2,sd,na.rm=TRUE) # columns sds
apply(avgs,2,max,na.rm=TRUE) # column maxs


## Other Apply Statements


## `tapply()`

tapply(X, INDEX, FUN = NULL, ..., simplify = TRUE)

## `tapply()`


tapply(circ2$daily, circ2$day, max, na.rm=TRUE)



## Data Summarization

## Basic Plots

## Scatterplot

plot(mtcars$mpg, mtcars$disp)


## Histograms

hist(circ2$daily)


## Density

## plot(density(circ2$daily))
plot(density(circ2$daily,na.rm=TRUE))


## Boxplots

boxplot(circ2$daily ~ circ2$day)


## Boxplots


boxplot(daily ~ day, data=circ2)


## Data Summarization for data.frames
```

```
## Matrix plot

matplot(avgs)

#Part 2


##Getting started: birthwt data set

library(MASS)
str(birthwt)


##Renaming the variables

colnames(birthwt)

# The default names are not very descriptive

colnames(birthwt) <- c("birthwt.below.2500", "mother.age",
"mother.weight",
    "race", "mother.smokes", "previous.prem.labor", "hypertension",
"uterine.irr",
    "physician.visits", "birthwt.grams")



##Renaming the factors

library(plyr)
birthwt <- transform(birthwt,
            race = as.factor(mapvalues(race, c(1, 2, 3),
                              c("white","black", "other"))),
            mother.smokes = as.factor(mapvalues(mother.smokes,
                              c(0,1), c("no", "yes"))),
            hypertension = as.factor(mapvalues(hypertension,
                              c(0,1), c("no", "yes"))),
            uterine.irr = as.factor(mapvalues(uterine.irr,
                              c(0,1), c("no", "yes"))),
            birthwt.below.2500 = as.factor(mapvalues(birthwt.below.2500,
                              c(0,1), c("no", "yes")))
            )

##Summary of the data

summary(birthwt)


##A simple table

with(birthwt, tapply(birthwt.grams, INDEX = list(race, mother.smokes), FUN
= mean))

##What if we wanted nicer looking output?
```

```
##aggregate() function

##Example: tapply vs aggregate

library(MASS)
with(birthwt, tapply(birthwt.grams, INDEX = list(race, mother.smokes), FUN
= mean)) # tapply
with(birthwt, aggregate(birthwt.grams, by = list(race, mother.smokes), FUN
= mean)) # aggregate


##Example: different syntax

aggregate(birthwt.grams ~ race + mother.smokes, FUN=mean, data=birthwt)


weight.smoke.tbl <- with(birthwt, table(birthwt.below.2500,
mother.smokes))
weight.smoke.tbl

or.smoke.bwt <- (weight.smoke.tbl[2,2] / weight.smoke.tbl[1,2]) /
(weight.smoke.tbl[2,1] / weight.smoke.tbl[1,1])
or.smoke.bwt

with(birthwt, cor(birthwt.grams, mother.age))  # Calculate correlation

with(birthwt, cor(birthwt.grams[mother.smokes == "yes"],
mother.age[mother.smokes == "yes"]))

with(birthwt, cor(birthwt.grams[mother.smokes == "no"],
mother.age[mother.smokes == "no"]))

##Faster way: by() function

by(data = birthwt[c("birthwt.grams", "mother.age")],
   INDICES = birthwt["mother.smokes"],
   FUN = function(x) {cor(x[,1], x[,2])})

##Standard graphics in R

## Single-variable plots

par(mfrow = c(2,2)) # Display plots in a single 2 x 2 figure
plot(birthwt$mother.age)
with(birthwt, hist(mother.age))
plot(birthwt$mother.smokes)
plot(birthwt$birthwt.grams)


par(mfrow = c(1,1))
plot(birthwt$mother.smokes,
     main = "Mothers Who Smoked In Pregnancy",
     xlab = "Smoking during pregnancy",
     ylab = "Count of Mothers",
     col = "lightgrey")

## (much) better graphics with ggplot2
```

```
## Introduction to ggplot2


library(ggplot2)


## plot vs qplot


with(birthwt, plot(mother.age, birthwt.grams))  # Base graphics
qplot(x=mother.age, y=birthwt.grams, data=birthwt)  # using qplot from
ggplot2

qplot(x=mother.age, y=birthwt.grams, data=birthwt,
      color = mother.smokes,
      shape = mother.smokes,
      xlab = "Mother's age (years)",
      ylab = "Baby's birthweight (grams)"
      )


## ggplot function


dim(diamonds)
head(diamonds)

diamond.plot <- ggplot(data=diamonds, aes(x=carat, y=price))
diamond.plot + geom_point()

diamond.plot + geom_point(size = 0.7, alpha = 0.3)

diamond.plot <- ggplot(data=diamonds, aes(x=carat, y=price, colour =
color))
diamond.plot + geom_point()


cbPalette <- c("#999999", "#E69F00", "#56B4E9", "#009E73", "#F0E442",
"#0072B2", "#D55E00", "#CC79A7")
diamond.plot <- ggplot(data=diamonds, aes(x=carat, y=price, colour =
color))
diamond.plot + geom_point() + scale_colour_manual(values=cbPalette)

diamond.plot + geom_point() +
  coord_trans(x = "log10", y = "log10")

## Conditional plots

diamond.plot <- ggplot(data=diamonds, aes(x=carat, y=price, colour =
color))

diamond.plot + geom_point() + facet_wrap(~ cut)

diamond.plot + geom_point() + facet_grid(. ~ cut)

diamond.plot + geom_point() + facet_grid(cut ~ .)
```

```
# Function | Description
# --------------------|-----------------------------------------------
# `geom_point(...)` | Points, i.e., scatterplot
# `geom_bar(...)` | Bar chart
# `geom_line(...)` | Line chart
# `geom_boxplot(...)` | Boxplot
# `geom_violin(...)` | Violin plot
# `geom_density(...)` | Density plot with one variable
# `geom_density2d(...)` | Density plot with two variables
# `geom_histogram(...)` | Histogram

## A bar chart


qplot(x = race, data = birthwt, geom = "bar")




## Histograms and density plots

base.plot <- ggplot(birthwt, aes(x = mother.age)) +
  xlab("Mother's age")
base.plot + geom_histogram()
base.plot + geom_histogram(aes(fill = race))
base.plot + geom_density()
base.plot + geom_density(aes(fill = race), alpha = 0.5)


## Box plots and violin plots


base.plot <- ggplot(birthwt, aes(x = as.factor(physician.visits), y =
birthwt.grams)) +
  xlab("Number of first trimester physician visits") +
  ylab("Baby's birthweight (grams)")

# Box plot
base.plot + geom_boxplot()

# Violin plot
base.plot + geom_violin()

## Visualizing means


bwt.summary <- aggregate(birthwt.grams ~ race + mother.smokes, data =
birthwt, FUN = mean) # aggregate
bwt.summary

# Define basic aesthetic parameters
p.bwt <- ggplot(data = bwt.summary, aes(y = birthwt.grams, x = race, fill
= mother.smokes))

# Pick colors for the bars
bwt.colors <- c("#009E73", "#999999")
```

```
# Display barchart
p.bwt + geom_bar(stat = "identity", position = "dodge") +
  ylab("Average birthweight") +
  xlab("Mother's race") +
  guides(fill = guide_legend(title = "Mother's smoking status")) +
  scale_fill_manual(values=bwt.colors)

by(data = birthwt[c("birthwt.grams", "mother.age")],
   INDICES = birthwt["mother.smokes"],
   FUN = function(x) {cor(x[,1], x[,2])})

ggplot(birthwt, aes(x=mother.age, y=birthwt.grams, shape=mother.smokes,
color=mother.smokes)) +
  geom_point() + # Adds points (scatterplot)
  geom_smooth(method = "lm") + # Adds regression lines
  ylab("Birth Weight (grams)") + # Changes y-axis label
  xlab("Mother's Age (years)") + # Changes x-axis label
  ggtitle("Birth Weight by Mother's Age") # Changes plot title
```