Statistics and Data Science I (AB) CHEAT SHEET

CourseKata

Word Equations

```
outcome = explanatory + other stuff
Y = X + other stuff
```

Basics

```
print("Hello world!")

# assign value to object
myNumber <- 5
# combine values into vector
myVector <- c(1, 2, 3)
# first element in vector
myVector[1]

# orders values or cases
sort(myVector)

# arithmetic operations
sum(1, 2, 100), +, -, *, /
sqrt(157)
abs(data_set$Y)
# logical operations
>, <, >=, <=, ==, !=, |, &
# results in a variable with values
# of TRUE or FALSE
data_set$C <- data_set$A > data_set$B
```

Summary Tables

```
# compute five-number summary
favstats(~ Y, data = data_set)

# create frequency table
tally(data_set$Y)
tally(~ Y, data = data_set)

# tally by condition
tally(~ Y < 1900, data = data_set)

# two-way frequency table
tally(Y ~ X, data = data_set, margin = TRUE,
format = "proportion")
```

Simple Statistics

```
mean(data_set$Y)
var(data_set$Y)
sd(data_set$Y)

cohensD(Y ~ X, data = data_set)
cor(Y ~ X, data = data_set)

b1(Y ~ X, data = data_set)
b1(one_model)

pre(Y ~ X, data = data_set)
f(Y ~ X, data = data_set)
```

Probability Distribution

```
# calculate the probability area
xpnorm(65.1, data_set$mean, data_set$sd)

zscore(data_set$Y)
```

Simulation

```
# sample without replacement
sample(data_set, 6)

# sample with replacement
resample(data_set, 10)
```

Data Frame

```
# structure of data frame
str(data_set)

# view first/last six rows
head(data_set)
tail(data_set)

# select multiple variables
select(data_set, Y1, Y2)

# first six rows of selected variables
head(select(data_set, Y1, Y2))


# select variable (a column)
data_set$Y

# find rows that meet condition
data_set[data_set$Y > 40]
filter(data_set, Y > 300)
filter(data_set, Y != "NA")
```

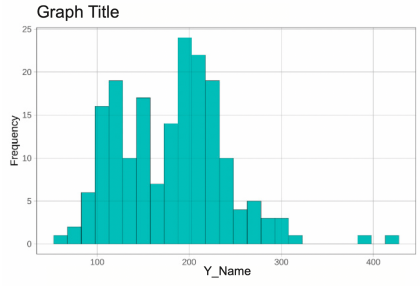
```
# arrange rows by variable
arrange(data_set, Y)

# creates data frame from csv file
data_set <- read.csv("file_name", header = TRUE)

# convert quantitative variable
# to categorical
factor(data_set$Y)
factor(data_set$Y, levels = c(1,2), labels =
c("A", "B"))

# transform values
recode(data_set$Y, "0" = 0, "1" = 50, "2" = 100)

# creates two equal sized groups
ntile(data_set$Y, 2)

# convert categorical variable
# to quantitative
as.numeric(data_set$Y)
```

Fitting & Evaluating Models

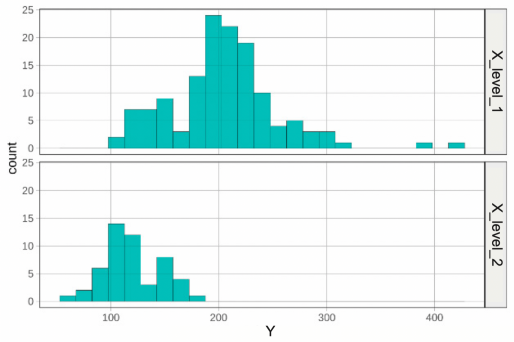
```
# empty model
empty_model <- lm(Y ~ NULL,
data = data_set)

# use one expanatory variable
one_model <- lm(Y ~ X, data = data_set)


# model predictions and residuals
data_set$empty_predict <- predict(empty_model)
data_set$empty_resid <- resid(empty_model)

supernova(empty_model)
```

Visualizations

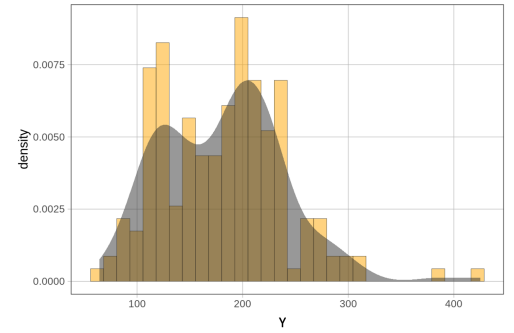
```
gf_histogram(~ Y, data = data_set) %>%
# change labels
  gf_labs(title = "Graph Title", x = "Y_Name",
y = "Frequency")
```



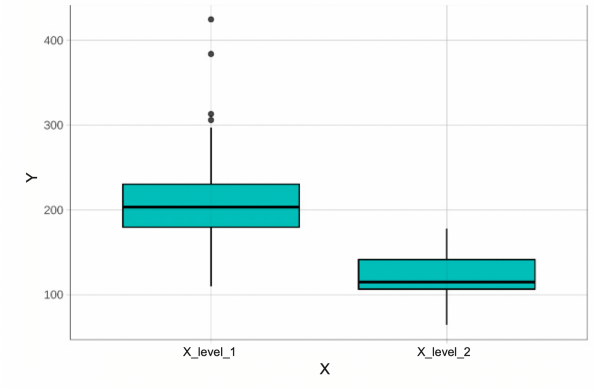

```
# faceted grid of histograms
gf_histogram(~ Y, data = data_set) %>%
  gf_facet_grid(X ~ .)
```



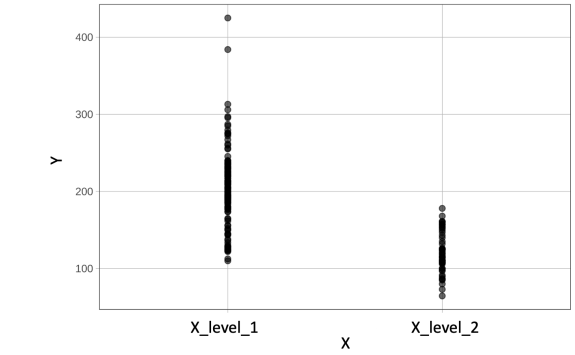

```
gf_dhistogram(~ Y, data = data_set, fill =
"orange") %>%
  gf_density()
```



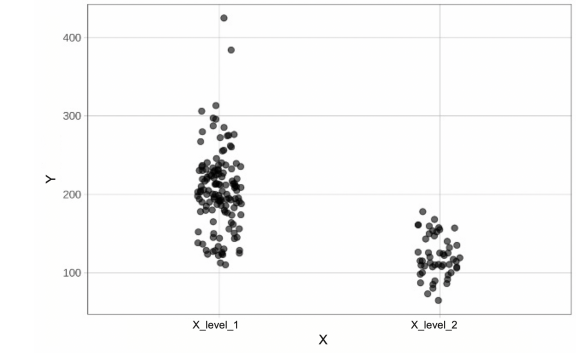

```
gf_boxplot(Y ~ X, data = data_set)
```



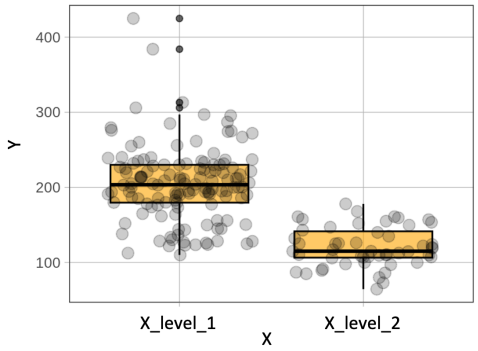

```
gf_point(Y ~ X, data = data_set)
```



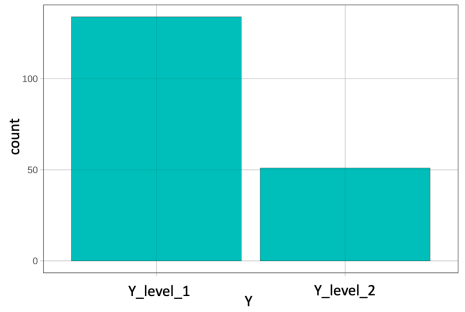

```
gf_jitter(Y ~ X, data = data_set)
```




```
gf_boxplot(Y ~ X, data = data_set, fill =
"orange") %>%
 gf_jitter(height = 0, alpha = .2, size =
3)
```




```
gf_bar( ~ Y, data = data_set)
```




```
gf_point(Y ~ X, data = data_set) %>%
 # add model predictions as red points
 gf_point(Y ~ X , shape = 1, size = 3,
color = "firebrick") %>%
 # add best fitting model as a red line
 gf_model(one_model, color = "red")
```