Doing Data Science

Unit 9

Machine Learning - 1

Dr. Jacquie Cheun
Data Science @ SMU

Admin notes

Case Study 1-Delayed

Machine Learning – general concepts and linear regression

Case Study 1

Everyone did well

Github: minor issues with readme, project organization

#Q1: How many breweries are present in each state

```
ct <- data.frame(count(brews, brews$State))
names(ct) <- c("State", "Count")
ct[order(-ct$Count),]
```

	State	Count
6	CO	47
5	CA	39
23	MI	32
38	OR	29

…

#Q2 Print first 6 and last 6 observations

rbind(head(df),tail(df))

```
     BreweryID                BeerName BeerID   ABV IBU                               Style Ounces
1            1            Get Together   2692 0.045  50                      American IPA     16
2            1            Maggie's Leap  2691 0.049  26                  Milk / Sweet Stout     16
3            1              Wall's End   2690 0.048  19                   English Brown Ale     16
4            1                 Pumpion   2689 0.060  38                         Pumpkin Ale     16
5            1              Stronghold   2688 0.060  25                     American Porter     16
6            1             Parapet ESB   2687 0.056  47 Extra Special / Strong Bitter (ESB)     16
2405       556           Pilsner Ukiah     98 0.055  NA                     German Pilsener     12
2406       557  Heinnieweisse Weissebier     52 0.049  NA                          Hefeweizen     12
2407       557          Snapperhead IPA     51 0.068  NA                        American IPA     12
2408       557        Moo Thunder Stout     50 0.049  NA                  Milk / Sweet Stout     12
2409       557        Porkslap Pale Ale     49 0.043  NA             American Pale Ale (APA)     12
2410       558 Urban Wilderness Pale Ale     30 0.049  NA                    English Pale Ale     12
                 Company        City State
1      NorthGate Brewing Minneapolis    MN
2      NorthGate Brewing Minneapolis    MN
3      NorthGate Brewing Minneapolis    MN
```

#Q3 Report the number of NAs

```
names(df)<- c("BreweryID","BeerName", "BeerID",
"ABV","IBU","Style","Ounces","Company","City","State")

sapply(X=df, FUN=function(x) sum(is.na(x))) # This is equivalent...

colSums(is.na(df)) # ...to this!
```

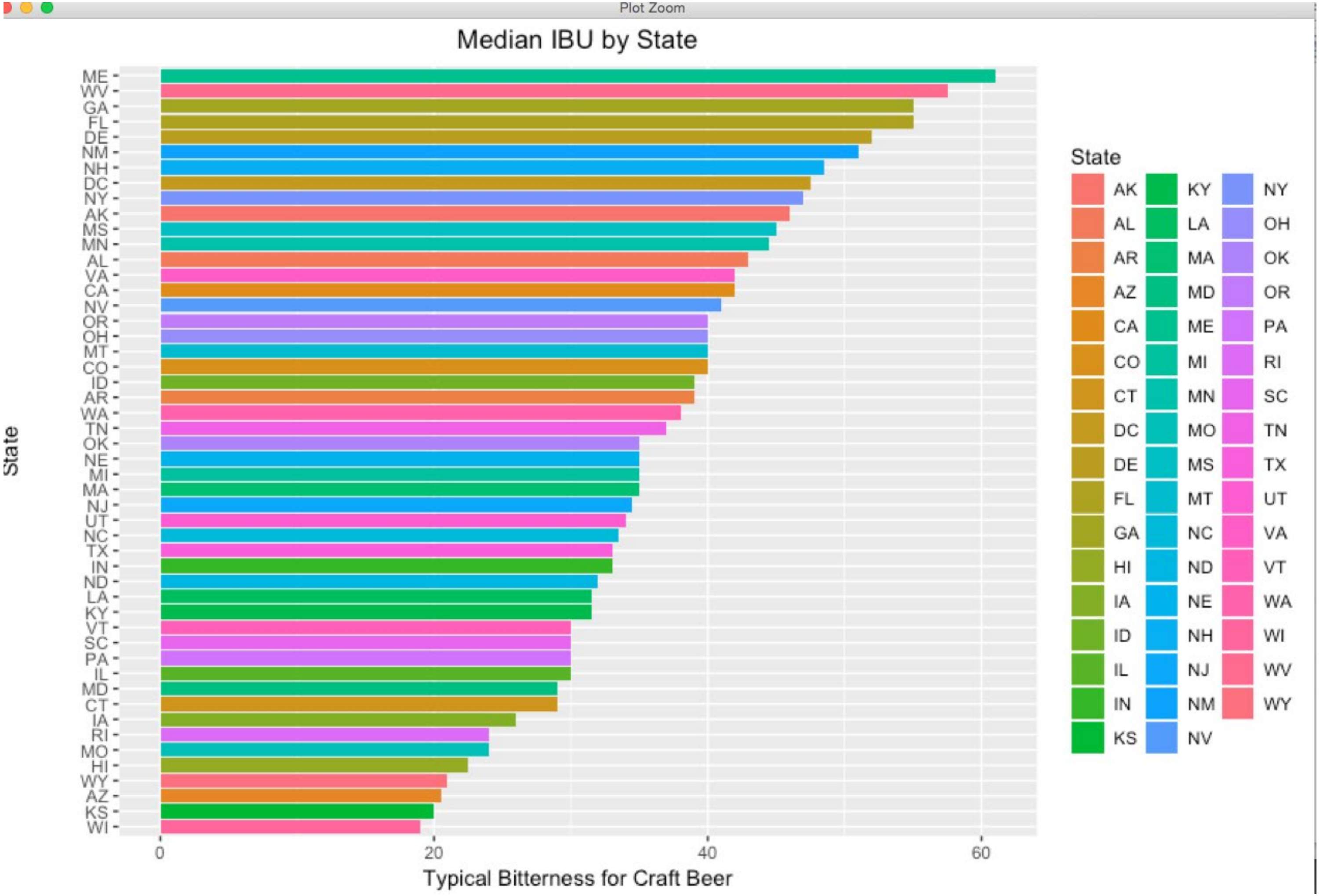
BreweryID	BeerName	BeerID	ABV	IBU	Style	Ounces	Company	City	State
0	0	0	62	1005	0	0	0	0	0

#Q4 Compute the median alcohol content and international bitterness unit for each state

```
median(df$IBU, na.rm=TRUE)

[1] 35

IB<-df %>% # I use dplyr here to do all my steps simultaneously, but you can do this piecemeal in base R, too
    select(State, IBU) %>%
    group_by(State) %>%
    summarize(MedianIBU=round(median(IBU, na.rm=TRUE),4)) %>%
                                          # I chose to round to four digits here
  arrange(desc(MedianIBU))
  # I put them in descending order because my clients are most interested in higher values
data.frame(IB)
```

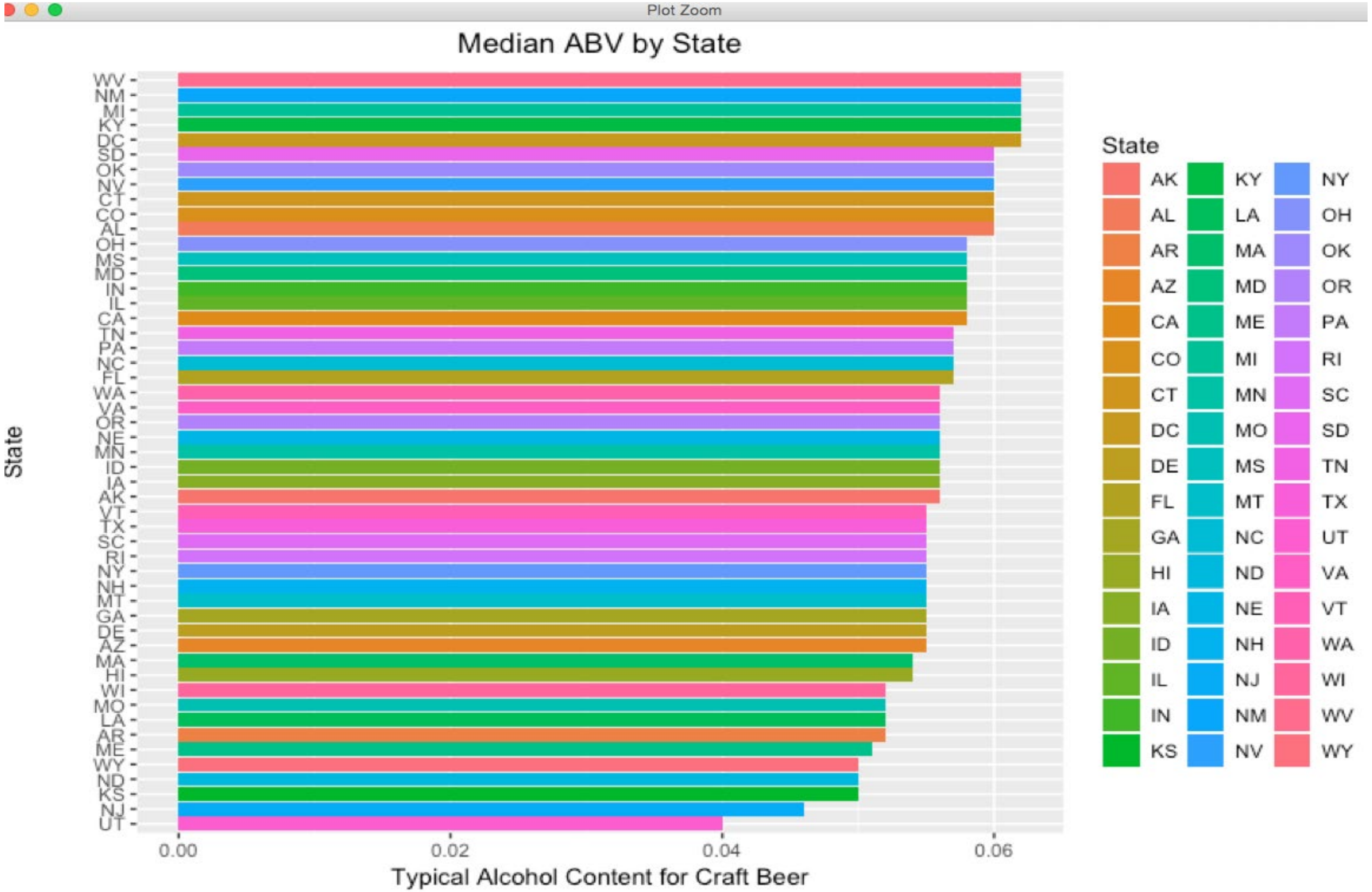

Plot Zoom
Median IBU by State
State
ME
WV
GA
FL
DE
NM
NH
DC
NY
AK
MS
MN
AL
VA
CA
NV
OR
OH
MT
CO
ID
AR
WA
TN
OK
NE
MI
MA
NJ
UT
NC
TX
IN
ND
LA
KY
VT
SC
PA
IL
MD
CT
IA
RI
MO
HI
WY
AZ
KS
WI
0
20
40
60
Typical Bitterness for Craft Beer
State
AK AL AR AZ CA CO CT DC DE FL GA HI IA ID IL IN KS
KY LA MA MD ME MI MN MO MS MT NC ND NE NH NJ NM NV
NY OH OK OR PA RI SC TN TX UT VA VT WA WI WV WY

```
median(df$ABV, na.rm=TRUE)

[1] 0.056

AB<-df %>%
  select(State, ABV) %>%
  group_by(State) %>%
  summarize(MedianABV=round(median(ABV, na.rm=TRUE),3)) %>%
  arrange(desc(MedianABV))
data.frame(AB)

ggplot(AB, aes(reorder(State, MedianABV), MedianABV)) +
# I chose to reorder the bars in descending order, rather than alphabetical
  geom_bar(aes(fill=State), stat="identity") + # the bar colors and what the values mean
  ggtitle("Median ABV by State") + # The Title
  theme(plot.title = element_text(hjust = 0.5)) + # Centers the Title
  xlab("State\n\n") + # Gives an X-axis name
  ylab("Typical Alcohol Content for Craft Beer") + # Gives an informative Y-axis name
  coord_flip() # Flips the coordinates - I do this because it's easier to see States on the Y-Axis
```


Plot Zoom
Median ABV by State
State
WV
NM
MI
KY
DC
SD
OK
NV
CT
CO
AL
OH
MS
MD
IN
IL
CA
TN
PA
NC
FL
WA
VA
OR
NE
MN
ID
IA
AK
VT
TX
SC
RI
NY
NH
MT
GA
DE
AZ
MA
HI
WI
MO
LA
AR
ME
WY
ND
KS
NJ
UT
0.00
0.02
0.04
0.06
Typical Alcohol Content for Craft Beer
State
AK
AL
AR
AZ
CA
CO
CT
DC
DE
FL
GA
HI
IA
ID
IL
IN
KS
KY
LA
MA
MD
ME
MI
MN
MO
MS
MT
NC
ND
NE
NH
NJ
NM
NV
NY
OH
OK
OR
PA
RI
SC
SD
TN
TX
UT
VA
VT
WA
WI
WV
WY

#Q5 which state has the maximum ABV beer? Which state has the most bitter beer?

Which state has the maximum alcoholic beer?

```
df[which.max(df$ABV),c("State","BeerName","ABV")]
```

	State	BeerName	ABV
375	CO	Lee Hill Series Vol. 5 - Belgian Style Quadrupel Ale	0.128

Which state has the most bitter beer?

```
df[which.max(df$IBU),c("State","BeerName","IBU")]
```

	State	BeerName	IBU
1857	OR	Bitter Bitch Imperial IPA	138

Q6: Summary statistics for the ABV variable

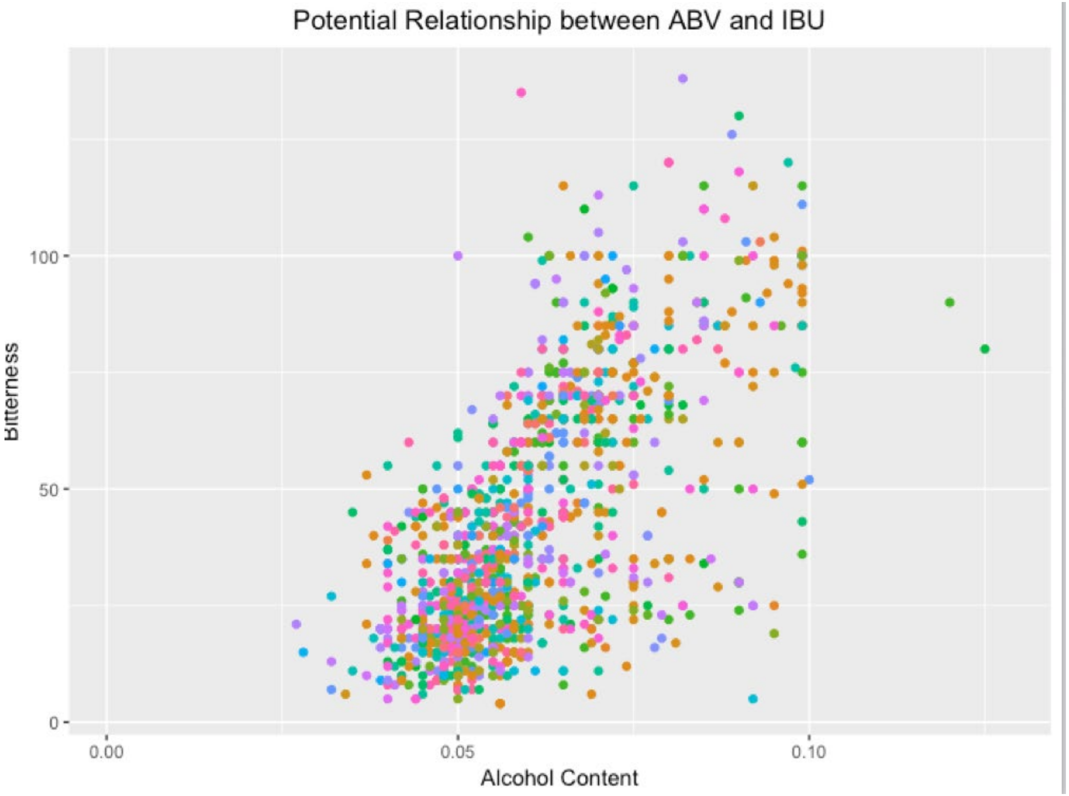
```
summABV<-data.frame(cbind(summary(df$ABV)))

names(summABV)<-"Summary Statistics (ABV)"

round(summABV,3)
```

	Summary Statistics (ABV)
Min.	0.001
1st Qu.	0.050
Median	0.056
Mean	0.060
3rd Qu.	0.067
Max.	0.128
NA's	62.000

#Q7: # Is there an apparent relationship between the IBU and ABV?



Machine learning- Automating Automation

https://medium.com/machine-learning-for-humans/why-machine-learning-matters-6164faf1df12

Gives “**computers the ability to learn without being explicitly programmed**” (Arthur Samuel, 1959)

Study of identifying patterns in data and building models to explain and predict those patterns.

Subset of Artificial Intelligence (study of intelligent agents)

Applications: predict, classify or cluster

Statistical Modeling (Statistics) vs Machine Learning Algorithms (CS), Model (generative process) vs Classify

Machine learning components

Training data (e.g. implicit and explicit feedback on job ads)

Target function (e.g. probability of user clicking and/or applying for a job)

Metrics (e.g. precision vs. recall, or any ranking metric that correlates to AB test metrics)

Supervised Learning

Requires training examples with labels

A learning algorithm that classifies new observations to one or more classes

Regression: predicting a continuous value (e.g., predicting house prices)

Classification: predicting a label (e.g., predicting a the category of an item on an ecommerce site), **“Samsung 7.5 cu. Ft. Electric Dryer in White” → ”Laundry Care | Dryers”**

Supervised Learning

Binary classification (two classes), e.g., spam classification

Multi-class classification (multiple classes, mutually exclusive), e.g., ecommerce product categorization

Multi-label classification: assigning multiple classes to an observation, e.g., Google Images: object classification in an image

- **Regression:** predict a continuous numerical value. *Credit scoring or predicting house prices*

- **Classification**: assign a label. *Is this a hammer or a light bulb?*

Unsupervised learning

Does not require training data or labels

Used for finding structure and patterns in unlabeled data

Clustering: cluster data by some similarity measure (e.g., customer segmentation, identifying customer clusters based on purchase history, location, etc.)

Dimensionality reduction: reduce the number of variables (high dimension to lower dimension mappings) – advantages?

Metrics . . . if you can’t measure you cant improve!

- **Confusion matrix: for classification problems …almost all metrics are based on it!**

		Predicted class	
		P	*N*
Actual Class	*P*	True Positives (TP)	False Negatives (FN)
	N	False Positives (FP)	True Negatives (TN)

True Positive: both actual and predicted classes are true

True Negative: both actual and predicted classes are false

False Positive: actual class is false and predicted is true

False Negative: actual class is true and predicted is false

Spam example:

1: spam
0: not spam

An important email classified as spam (i.e., false positive), is worse than a spam email classified as not spam (i.e., false negative)

Hence, **minimizing false positives** is more important for this problem. Any examples where **minimizing false negatives** is more important?

Accuracy

- Total number of correct predictions made by the model
- Accuracy = (TP + TN) / (TP + TN + FN + FP)
- Use when classes are balanced
- Don't use for unbalanced classes (i.e when one class is a majority, as in disease prediction or spam classification)

Precision and Recall

- Precision: what proportion of positive instances identified were actually correct
- Precision = TP / (TP + FP)

- Recall: the proportion of positive instances correctly identified
- Recall = TP / (TP + FN)

- Being precise vs capturing all cases

Linear Regression

What is Prediction All About?

- Correlations can be used as a basis for the prediction of the value of one variable from the value of another
    - Correlation can be determined by using a set of previously collected data (such as data on variables X and Y)
    - Calculate how correlated these variables are with one another
    - Use that correlation and the knowledge of X to predict Y with a new set of data
- Therefore, we describe the relationship using the equation of a straight line.

Remember…

- The greater the strength of the relationship between two variables (the higher the absolute value of the correlation coefficient) the more accurate the predictive relationship

- Why???
  - The more two variables share in common (shared variance) the more you know about one variable from the other. ☺

Measurement and Linear Regression

- Correlation and regression are both concerned with the relationship between at least two interval level variables

- While Pearson’s correlation coefficient examines the covariance and computes the correlation coefficient, linear regression extends this hypothesis and uses the regression line to predict what will occur in variable y at a given value of x

Measurement and Linear Regression

- **Linear regression** = concerned with finding the line that most closely fits the data, and assessing how well it does so; attempts to determine what will occur in the variable y at a given value of x
  - Assumptions:

1. Both variables are measured at the interval level
2. Only a linear relationship is appropriate
3. Random sample
4. Normally distributed variables and a reasonable sample size

Measurement and Linear Regression

- Y is literally a function of X
  - As X increases, Y increases by a set, known, and constant amount
  - Constant amount = slope (by how much the values of y increase with each increase in x)
- Bivariate regression when dealing with only one dependent variable and one independent variable

Describing a Straight Line

$$Y_i = b_0 + b_i X_i + \varepsilon_i$$

- b_i
  - Regression coefficient for the predictor
  - Gradient (slope) of the regression line
  - Direction/strength of relationship
- b_0
  - Intercept (value of Y when $X = 0$)
  - Point at which the regression line crosses the Y-axis (ordinate)

Drawing the World’s Best Line

- Linear Regression Formula
  - $Y = bX + a$

- ***Y*** = dependent variable
  - the predicted score or criterion
- ***X*** = independent variable
  - the score being used as the predictor
- ***b*** = the slope
  - direction and “steepness” of the line
- ***a*** = the intercept
  - point at which the line crosses the y-axis

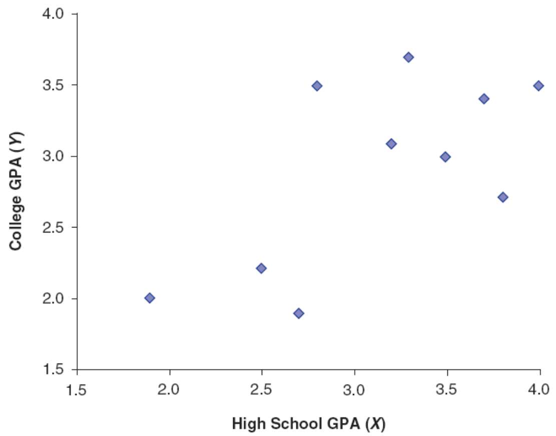
The Logic of Prediction

- Prediction is an activity that computes future outcomes from present ones
    - What if you wanted to predict college GPA based on high school GPA?

High School GPA	First-Year College GPA
3.50	3.30
2.50	2.20
4.00	3.50
3.80	2.70
2.80	3.50
1.90	2.00
3.20	3.10
3.70	3.40
2.70	1.90
3.30	3.70

Scatterplot

Figure 16.1 Scatterplot of High School GPA and College GPA



- Used to look at two variables at the same time.
- Represents each variable on a pair of axes
- Slope of the line reveals the direction of the relationship
- Start to understand the strength of the correlation by examining how close the points are to the imaginary line that runs through them

Regression Line

- Regression line – reflects our best guess as to what score on the Y variable would be predicted by the X variable.
- Line that minimizes the distance between the line and each of the points on the predicted Y variable
- If correlation were perfect, all data points would align themselves along a 45 degree angle and pass through each point
    - Also known as the “line of best fit.”

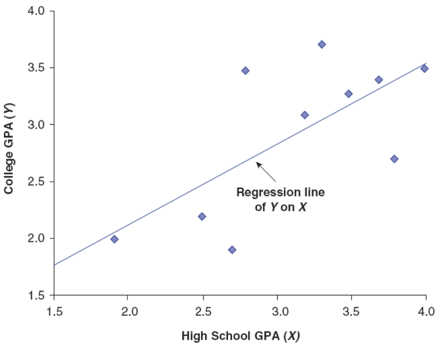


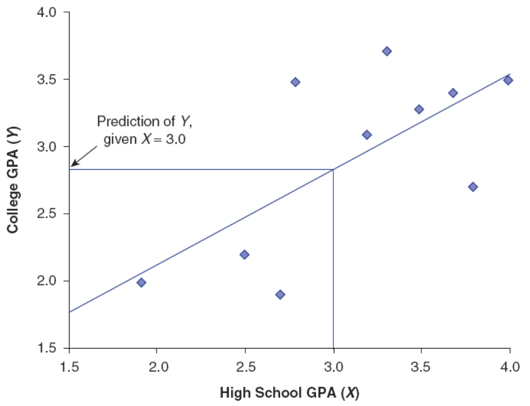
Figure 16.2 Regression Line of College GPA (*Y*) on High School GPA (*X*)

- Y (college GPA) predicted from X (high school GPA)

Prediction of Y given X = 3.0

- Regression line represents our best guess at estimating Y given X
- If high school GPA is 3.0 then college GPA should be around 2.8

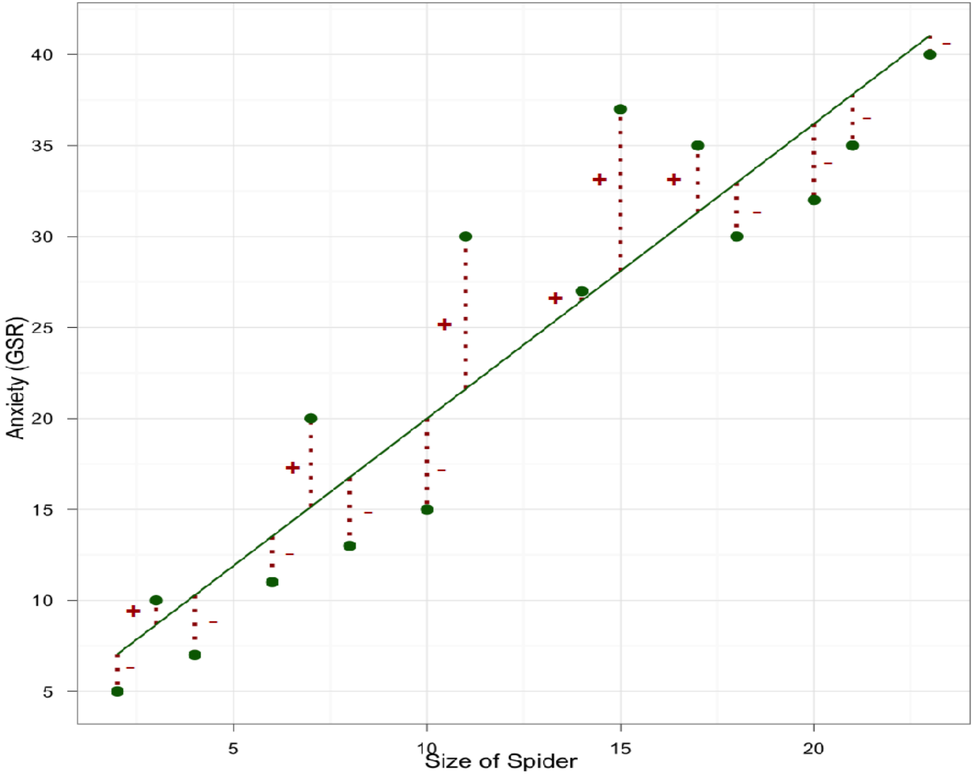
Figure 16.3 Estimating College GPA Given High School GPA



The Method of Least Squares

This graph shows a scatterplot of some data with a line representing the general trend.

The vertical lines (dotted) represent the differences (or residuals) between the line and the actual data

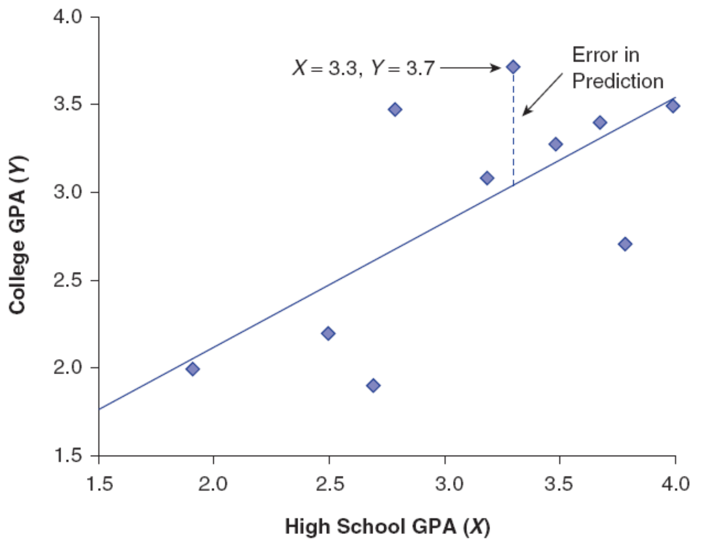


Error in Prediction

- The distance between each individual data point and the regression line is the error in prediction – or a direct reflection of the correlation between two variables

- Where would predicted points fall if perfect prediction?

Figure 16.4 Prediction Is Rarely Perfect: Estimating the Error in Prediction



How Good Is the Model?

- The regression line is only a model based on the data.
- This model might not reflect reality.
  - We need some way of testing how well the model fits the observed data.
  - How?
- Standard error of estimate
  - the measure of how much each data point (on average) differs from the predicted data point or a standard deviation of all the error scores
- The higher the correlation between two variables (and the better the prediction), the lower the error will be
  - Perfect predictive relationship, error = 0

Sums of Squares

Diagram showing from where the regression sums of squares derive

SS_T

Total variability (variability between scores and the mean).

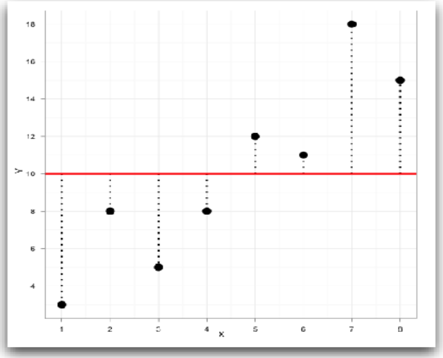
SS_R

Residual/error variability (variability between the regression model and the actual data).

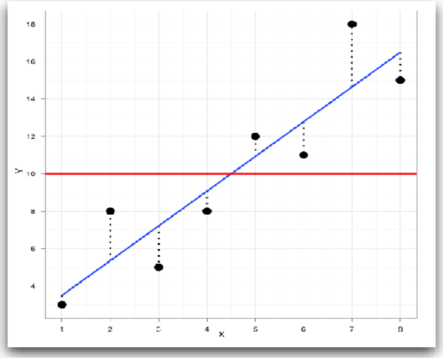
SS_M

Model variability (difference in variability between the model and the mean).

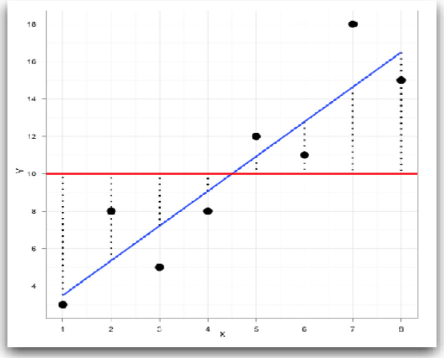
If the model results in better prediction than using the mean, then we expect SS_M to be much greater than SS_R



SS_T uses the differences between the observed data and the mean value of Y



SS_R uses the differences between the observed data and the regression line



SS_M uses the differences between the mean value of Y and the regression line

Testing the Model: ANOVA

- Mean squared error
  - Sums of squares are total values.
  - They can be expressed as averages.
  - These are called mean squares, MS.

$$\boxed{F = \frac{MS_M}{MS_R}}$$

Testing the Model: R^2

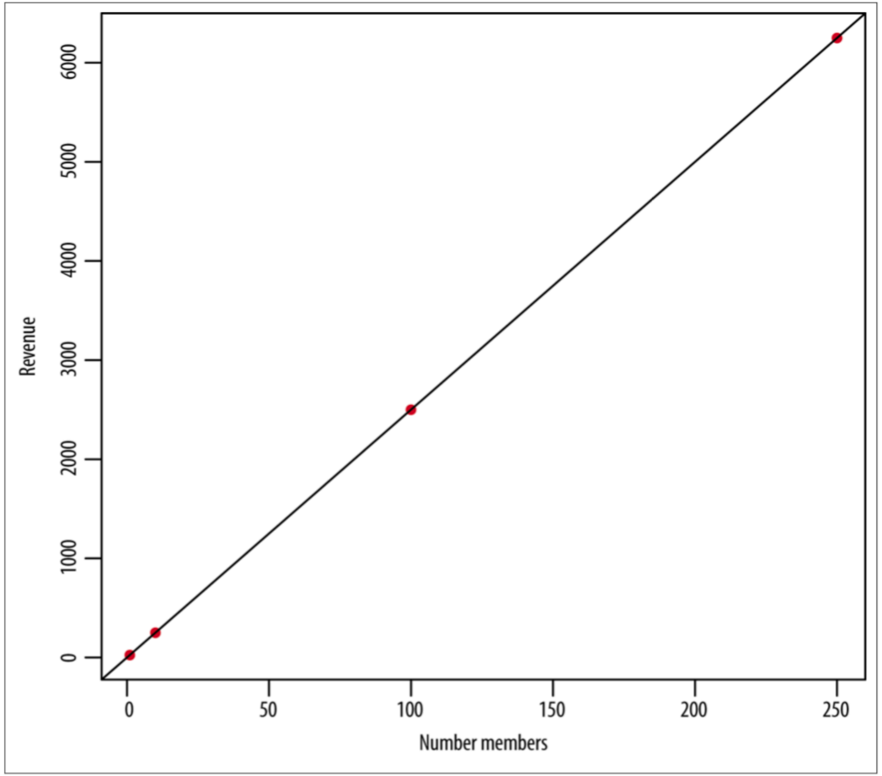
- R^2
  - The proportion of variance accounted for by the regression model.
  - The Pearson Correlation Coefficient Squared

$$R^2 = \frac{SS_M}{SS_T}$$

Regression Examples

Linear Regression

- Expressing a (linear) relationship between a label (outcome variable, dependent variable) and 1 (“simple”) or more features (predictors) (independent variable
- Examples: sell more items, make more money; others?
- *S*= (*x*,*y*) = (1,25), (10,250), (100,2500), (200,5000)
  y = m(x)
- **Find the line that minimizes the distances between all the points and the line**



- Need to capture trend and variation in the model
- *y* = *β*0 + *β*1*x*. → *find best choices for β*0 and *β*1 using the observed data
- *y* = x.*β (matrix form)*

Least squares estimation (LSS): $RSS(\beta) = \sum_i (y_i - \beta x_i)^2$

- Minimizes the sum of the squares of the vertical distances between the predicted and the observed values (to minimize prediction errors)

- Example: **"the more new friends you have, the more time you might spend on the site. "**

- Modeled the trend but have not modeled the variation.

- Everyone with 5 new friends is not guaranteed to spend a certain fixed amount of time on the site.

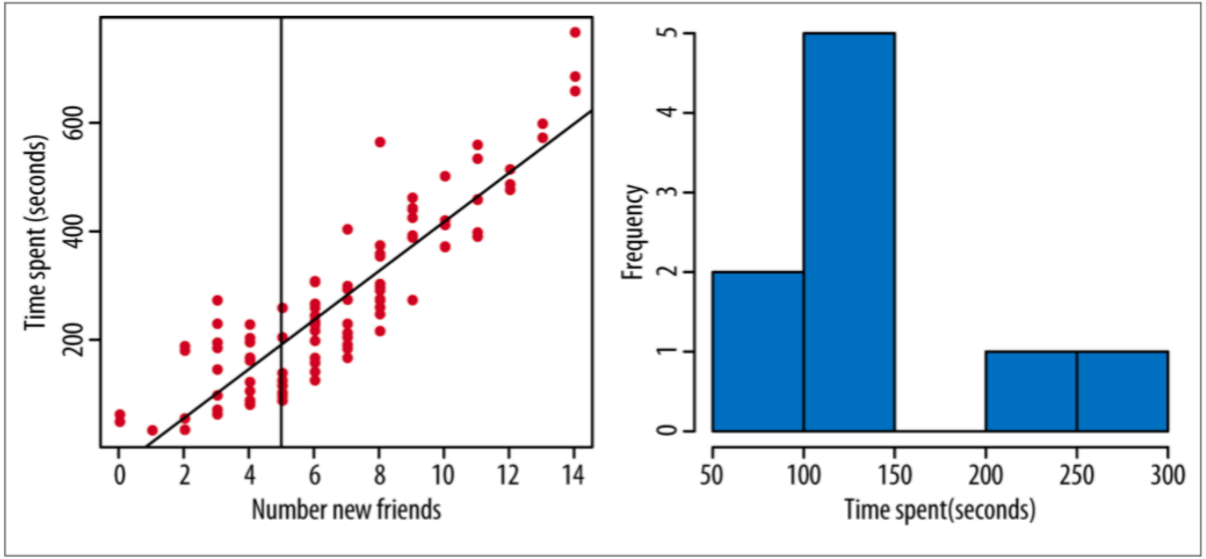


Figure 3-5. On the left is the fitted line. We can see that for any fixed value, say 5, the values for y vary. For people with 5 new friends, we display their time spent in the plot on the right.

Extending beyond Least Squares

- Adding in modeling assumption about errors (variability)

- ***y* = *β*0 + *β*1*x* + ε** where **ε** = noise or error term

- **$e_i = y_i - est(y_i) = y_i - \beta_0 + \beta_1 x_i$ for i=1,...,n.**

Variance of errors = $\sum_i e_i^2 / (n-2)$

- **Aka <u>Mean Squared Error:</u> captures how much the predicted values varies from the observed value. Is an example of a loss function.**

Regression in R

Regression in R

Example

- A record company boss was interested in predicting record sales from advertising.
- Data
  - 200 different album releases
- Outcome variable:
  - Sales (CDs and downloads) in the week after release
- Predictor variable:
  - The amount (in units of £1000) spent promoting the record before release.

- We run a regression analysis using the *lm()* function – lm stands for 'linear model'. This function takes the general form:

```
newModel<-lm(outcome ~ predictor(s), data = dataFrame, na.action = an action))
```

```
albumSales.1 <- lm(album1$sales ~ album1$adverts)
```

- or we can tell **R** what dataframe to use (using *data = nameOfDataFrame*), and then specify the variables without the *dataFrameName$* before them:

```
albumSales.1 <- lm(sales ~ adverts, data = album1)
```

- We have created an object called *albumSales.1* that contains the results of our analysis. We can show the object by executing:

```
summary(albumSales.1)

>Coefficients:
             Estimate    Std. Error   t value    Pr(>|t|)
(Intercept) 1.341e+02   7.537e+00   17.799     <2e-16 ***
adverts     9.612e-02   9.632e-03    9.979     <2e-16 ***

Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 65.99 on 198 degrees of freedom
Multiple R-squared: 0.3346, Adjusted R-squared: 0.3313
F-statistic: 99.59 on 1 and 198 DF,  p-value: < 2.2e-16
```

The More Predictors the Better? Multiple Regression

- Multiple Regression Formula
  - $Y = bX_1 + bX_2 + a$
    - Y = the value of the predicted score
    - X_1 = the value of the first independent variable
    - X_2 = the value of the second independent variable
    - b = the regression weight for each variable
- Predicting an outcome from two or more independent variables

Using the Model

$$\begin{aligned}\text{Record Sales}_i &= b_0 + b_1\text{Advertising Budget}_i \\ &= 134.14 + \left(0.09612 \times \text{Advertising Budget}_i\right)\end{aligned}$$

$$\begin{aligned}\text{Record Sales}_i &= 134.14 + \left(0.09612 \times \text{Advertising Budget}_i\right) \\ &= 134.14 + \left(0.09612 \times 100\right) \\ &= 143.75\end{aligned}$$

Things to Remember When Using Multiple Predictors

- Your independent variables (X_{1}, X_{2} , X_{3} , etc.) should be related to the dependent variable (Y)…they should have something in common

- However…the independent variables should not be related to each other…they should be "uncorrelated" so that they provide a "unique" contribution to the variance in the outcome of interest.
    - Independent variables should be related to the dependent variable, but unrelated to each other

- **Adding other predictors : Multiple linear regression**
- $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \varepsilon$
- **Example: use other factors such as age, gender, etc to model time spent on a website**

- **Adding transformations:**
- **Example:** $y = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3$
- But this is polynomial, not linear.
  Create a new variable: $z = x^3$
- Other common transformation: log, using thresholds, etc.

Boston Housing linear regression demo using MLR

What did you learn today?

Questions?