ADVANCED R WORKSHOP

Bruin Actuarial Society



AGENDA

- 1. R Markdown
- 2. Simple and Multiple Linear Regression
- 3. Logistic Regression
- 4. K-nearest Neighbor
- 5. K-Mean Clustering



R MARKDOWN



WHAT IS R MARKDOWN?

- R Markdown is a type of file that allows to you to typeset your analysis and code in R in a presentable way
- Output: HTML or PDF

Part e
Use rexp function in R to generate 1000 samples with the λ you dervied from (d), and then draw a EDF plot with ecdf function. Finally, add confidence interval to your EDF plot.
Solution
<pre>///generate the data rand_wet <- resp(1000, (1/6)) r_edf -= cod(rand_wet) //plot //plot //plot(read, main = "EDP Graph") //percate CI cl_wet <- r_edf(sot(rand_wet)) = 1.96*sqrt(sd(r_edf(sot(rand_wet)))^2/1000) cl_wet <- r_edf(sot(rand_wet)) = 1.96*sqrt(sd(r_edf(sot(rand_wet)))^2/1000) //add CI plot(r_edf, main = "EDP Graph with Confidence Interval") lines(sot(rand_wet), cl_upper, type ="1", typ = 3) lines(sot(rand_wet), cl_upper, type ="1", typ = 3) </pre>
EDF Graph

Part a

Draw a scatterplot of Wt on the vertical axis versus Ht on the horizontal axis. On the basis of this plot, does a simple linear regression model make sense for these data? Why or why not? **Solution** We will first plot the data

plot(ht, wt)



On the basis of the scatterplot, a simple linear model does not seem appropriate since the relation is not purely linear.

BRUIN ACTUARIAL SOCIETY

CREATING R MARKDOWN

• Go to File -> New File -> R Markdown

é	RStudio	File	Edit	Code	View	Plots	Sessio	n Build	Debug	Profile	Tools	Windo
••		New F	ile				>	R Script		<mark>ሰ</mark> ቼ N		hw4
*	1	New F	Project.					R Noteboo	ok		€ ⊂ In	sert 👻
	Pierce	Open	File				жo				0.03	036962
	Jeffers	Reope	en with	Encodin	g			R Markdov	wn		0.03	602810
	Madis	Recer	nt Files				>	Shiny Web	o App		0.03	357032
	Jeffers							Plumber A	VPI		0.03	036962
	Willar	Open	Projec	t				C File			0.03	009134
	Irving	Open	Projec	t in New	Session.			C++ File			0.03	238297
	McKir	Recer	nt Proje	cts			>	Header Fi	ام		0.03	009134
	9 rows	Impor	t Datas	set			>					
								Markdowr	n File			
69	c	Save					ЖS	HTML File				
70 71	So ther€	Save	As					CSS File				



CREATING R MARKDOWN

- Specify title, author and type of file
 - Need LaTex to create PDF





SIMPLE & MULTIPLE LINEAR REGRESSION



WHAT IS REGRESSION?

- Regression is a technique for estimating the relationships between variable
- A linear regression is regression technique where you use a linear approach to estimate the response variable
- Examples
 - $y = b_0 + b_1 x_1$ (Simple linear regression)
 - $y = b_0 + b_1 x_1 + ... + b_n x_n$ (Multiple linear regression)



LINEAR MODEL IN R

- For simple linear regression
 - M1 <- Im(y~x, data)
- For multiple linear regression
 - M2 <- Im(y~x1 + x2 + ... + xn, data)
- Use summary() to view the statistics of the model
- Use plot() to view the diagnostic graphs associated with the model

```
linear_mod = lm(cty ~ displ, data = mpg)
summary(linear_mod)
```

plot(linear_mod)

Call:							
lm(formula =	cty ~ displ	, data = r	npg)				
D · · · · ·							
Residuals:							
Min	1Q Median	3Q	Max				
-6.3109 -1.40	695 -0.2566	1.1087 14	1.0064				
Coefficients	:						
	Estimate Std	. Error t	value Pr	'(> t)			
(Intercept)	25.9915	0.4821	53.91	<2e-16	***		
displ	-2.6305	0.1302	-20.20	<2e-16	***		
Signif. codes	s: 0 '***' (0.001 '**	' 0.01 '*	° 0.05	'.' 0.1	''1	
Residual sta	ndard error:	2.567 on	232 dear	ees of	freedom		

Multiple R-squared: 0.6376, Adjusted R-squared: 0.6361 F-statistic: 408.2 on 1 and 232 DF, p-value: < 2.2e-16



DIAGNOSTIC PLOTS

- Red Flags to look out for:
 - Obvious patterns in residuals
 - Nonconstant variance
 - Normality of residuals
 - Influential points
- Any violations could make your model invalid





PRACTICE TIME!





- 1. Examine the variables within the iris dataset
 - a. Read the description (?iris)
 - b. Glance through the dataset
- 2. We will practice creating a MLR model on the iris dataset
 - a. Use Petal Length as the response variable
 - b. Use Sepal Length and Species as the predictors
 - c. Plot the diagnostics
 - d. Do you think your model did a good job?



LOGISTIC REGRESSION



WHAT IS LOGISTIC REGRESSION

- Logistic regression is a regression technique for estimating binary output
 - The range is bounded between 0 and 1
 - Useful for predicting probabilities
- General form

•
$$p = \frac{\exp(b_0 + b_1 x_1 + \dots + b_n x_n)}{1 + \exp(b_0 + b_1 x_1 + \dots + b_n x_n)}$$



LOGISTIC REGRESSION IN R

• Code:

 M3 <- glm(y~x1 + .. + xn, data, family = "binomial") log_mod = glm(efficiency ~ displ, data = new_mpg, family = "binomial")
summary(log_mod)

Call:

glm(formula = efficiency ~ displ, data = new_mpg)

Deviance Residuals:

Min 1Q Median 3Q Max -0.9183 -0.2835 0.0817 0.2548 0.6588

Coefficients:

Estimate Std. Error t value Pr(>|t|) (Intercept) 1.43770 0.06184 23.25 <2e-16 *** displ -0.28856 0.01670 -17.28 <2e-16 *** ---Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for gaussian family taken to be 0.1084307)

Null deviance: 57.538 on 233 degrees of freedom Residual deviance: 25.156 on 232 degrees of freedom AIC: 148.19

Number of Fisher Scoring iterations: 2



K-NEAREST NEIGHBOR



WHAT IS K-NEAREST NEIGHBOR

- K-Nearest Neighbor (KNN) is a supervised classification technique
 - Classification: predict data points into different existing classes/category from data
 - Supervised: the data you use to construct the model are labelled
- General Idea: Suppose you are given data, and you want to classify a new
 unknown data point
 - 1. Fix K
 - 2. Find K "closest" data points from the new data
 - 3. Return the mode of the K labels



KNN IN R

- First load the class library
 - library(class)
- To predict the class of a new point test_case_a
 - knn(train = training_x, cl = training_y, test = test_case_a, k = 5)

test_case_a				iris	;[24,]				
Sepal.Length 24 5.1	Sepal.Width	Petal.Length 1.7	Petal.Width 0.5	24	Sepal.Lengtl 5.2	n Sepal.Width L 3.3	Petal.Length	Petal.Width 0.5	Species setosa

```{r} library(class) knn(train = training\_x, cl = training\_y, test = test\_case\_a, k = 5)

[1] setosa
Levels: setosa versicolor virginica



# PRACTICE TIME!





- 1. Examine Test Case B and Test Case C
  - a. Output the contents of test\_case\_b and test\_case\_c
  - b. Look at the true species for the test cases
- 2. Use the knn function to predict the Species of test cases B and C
  - a. Use K = 5 (5 closest data points)
  - b. Predict the species
  - c. Did the knn model get the correct result?



## K-MEAN CLUSTERING



## WHAT IS K-MEANS CLUSTERING?

- K-Means Clustering is clustering technique
  - Clustering: group data points together into a category
- This technique is unsupervised, meaning you do not have the category of the data
- General Idea:





## K-MEAN CLUSTERING IN R

To perform the clustering algorithm

- km.res <- kmeans(data, center, nstart)</li>
- The parameter you input for nstart will determine how many random starting point R will use
  - Default is 1, use bigger number for more stable result



## K-MEAN CLUSTERING IN R - EXAMPLE

Suppose for iris, we do not know the true species for the data. However, we do know that there are 3 species

- We will use all the features to cluster
- Since there are 3 species -> center = 3
- For more stable result, we arbitrarily choose nstart = 20

```
df <- iris
set.seed(101)
irisCluster <- kmeans(df[,1:4], center=3, nstart=20)
irisCluster</pre>
```



### K-MEAN CLUSTERING IN R - EXAMPLE

#### Let's look at the result!

table(irisCluster\$cluster, df\$Species)

| ## |   |        |            |           |
|----|---|--------|------------|-----------|
| ## |   | setosa | versicolor | virginica |
| ## | 1 | 0      | 2          | 36        |
| ## | 2 | 0      | 48         | 14        |
| ## | 3 | 50     | 0          | 0         |
|    |   |        |            |           |



# PRACTICE TIME!



### EXERCISE

- 1. Examine the variables within the mtcars dataset
  - a. Read the description (?mtcars)
  - b. Glance through the dataset
- 2. We will implement K-Mean clustering to predict transmission (am)
  - a. Type set.seed(100) for reproducible result
  - b. Use mpg and disp to train the model
  - c. Use nstart = 20
  - d. Store the result into the variable mtcarCluster
  - e. Type table(mtcarCluster\$cluster, mtcars\$am) to analyze the accuracy



## QUESTIONS?

