---

title: 'CWRU DSCI351-351m-451: Lab Exercise LE7 NAME' subtitle: 'Inference, Linear Regression, Timeseries Analysis' author: "Prof.:Roger French, TA: Raymond Wieser, Sameera Nalin Venkat" date: "03 December, 2021" output: pdf_document: latex_engine: xelatex toc: TRUE number_sections: TRUE toc_depth: 6 highlight: tango html_notebook: html_document: css: ../lab.css highlight: pygments theme: cerulean toc: yes toc_depth: 6 toc_float: yes df_print: paged urlcolor: blue always_allow_html: true —

LE7, 10 points, questions.

- Q1 - OIS: Numerical inference, 1 pt.
- Q2 - OIS: Linear regression, 1 pt.
- Q3 - OIS: Logistic regression, 1 pt.
- Q4 - Logistic regression: Palmer's penguins, 3 pts.
- Q5 - Houston crime data, 4 pts.

```
set.seed(1)
library(tidyverse)
```

Lab Exercise (LE) 7

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.4     v dplyr   1.0.7
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   2.0.1     v forcats 0.5.1
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

Q1. OIS: Numerical inference (1 point)

OIS v3 5.44: Teaching descriptive statistics.

A study compared five different methods for teaching descriptive statistics.

- The five methods were
  - traditional lecture and discussion,
  - programmed textbook instruction,
  - programmed text with lectures,
  - computer instruction,
  - and computer instruction with lectures.
- 45 students were randomly assigned,
  - 9 to each method.
- After completing the course,
  - students took a 1-hour exam.

What are the hypotheses for evaluating - if the average test scores are different - for the different teaching methods?

What are the degrees of freedom associated with the F -test

- for evaluating these hypotheses?

Suppose the p-value for this test is 0.0168.

- What is the conclusion?

```
### Hypothesis
#
# H0: The mean score is the same across all groups.
# HA: At least one mean score is different.

# Generally we must check three conditions on the data before performing ANOVA:
# • the observations are independent within and across groups,
# • the data within each group are nearly normal, and
# • the variability across the groups is about equal.

# There are two degree of freedoms associated with these tests
# mean square between groups (MSG), and it has an associated
# degrees of freedom,dfG k-1, when there are k groups

k= 5

dFg =  k -1

# We need a benchmark value for how much variability should be expected among the sample means if the n
n = 45

dFE = n - k
```

ANSWER:

Hypothesis

H0: The mean score is the same across all groups. HA: At least one mean score is different.

There are two degree of freedoms associated with these tests:

Mean square between groups (MSG), and it has an associated degrees of freedom,dfG = k-1, when there are k groups.

dFg : 4

We need a benchmark value for how much variability should be expected among the sample means if the null hypothesis is true. To this end, we compute a pooled variance estimate, often abbreviated as the mean square error (MSE), which has an associated degrees of freedom value dfE = n - k.

dFE : 40

The p-value is smaller than 0.05, indicating the evidence is strong enough to reject the null hypothesis at a significance level of 0.05.

Q2. OIS: Linear regression (1 point)

OIS v3 7.12: Trees.

This dataset shows the relationship between height, diameter, and volume of timber in 31 felled black cherry trees. The diameter of the tree is measured 4.5 feet above the ground.

Visualize the relationships in the data

- height vs. volume
- diameter vs. volume
- your visualizations should contain all important info (labels, etc.)
- (hint: scatterplots)

Let's answer questions using linear regression

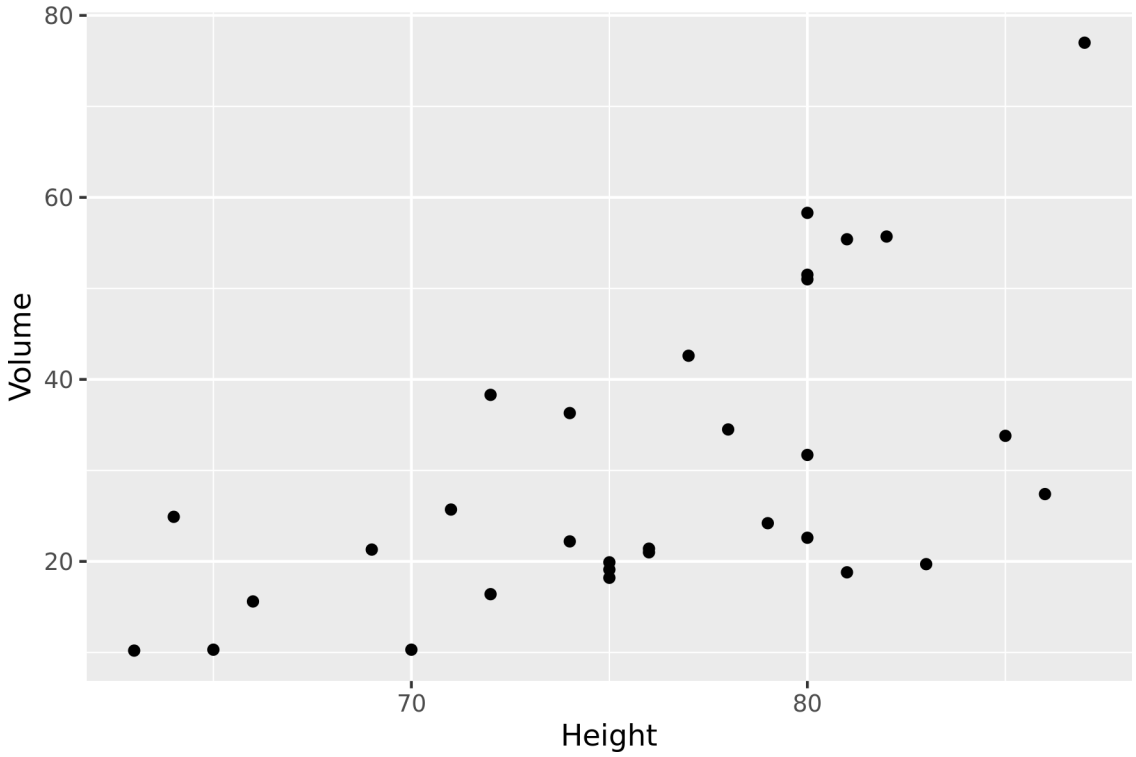
- Describe the relationship between volume and height of these trees.
- Describe the relationship between volume and diameter of these trees.
- The summarizing the model results from the lm() function will provide valuable numerical insights.

Suppose you have height and diameter measurements for another black cherry tree. Which of these variables would be preferable to use to predict the volume of timber in this tree using a simple linear regression model? Explain your reasoning.

```
library(tidyverse)
datasets::trees
```

```
##    Girth Height Volume
## 1    8.3     70   10.3
## 2    8.6     65   10.3
## 3    8.8     63   10.2
## 4   10.5     72   16.4
## 5   10.7     81   18.8
## 6   10.8     83   19.7
## 7   11.0     66   15.6
## 8   11.0     75   18.2
## 9   11.1     80   22.6
## 10  11.2     75   19.9
## 11  11.3     79   24.2
## 12  11.4     76   21.0
## 13  11.4     76   21.4
## 14  11.7     69   21.3
## 15  12.0     75   19.1
## 16  12.9     74   22.2
## 17  12.9     85   33.8
## 18  13.3     86   27.4
## 19  13.7     71   25.7
## 20  13.8     64   24.9
## 21  14.0     78   34.5
## 22  14.2     80   31.7
## 23  14.5     74   36.3
## 24  16.0     72   38.3
## 25  16.3     77   42.6
## 26  17.3     81   55.4
## 27  17.5     82   55.7
## 28  17.9     80   58.3
## 29  18.0     80   51.5
## 30  18.0     80   51.0
## 31  20.6     87   77.0
```

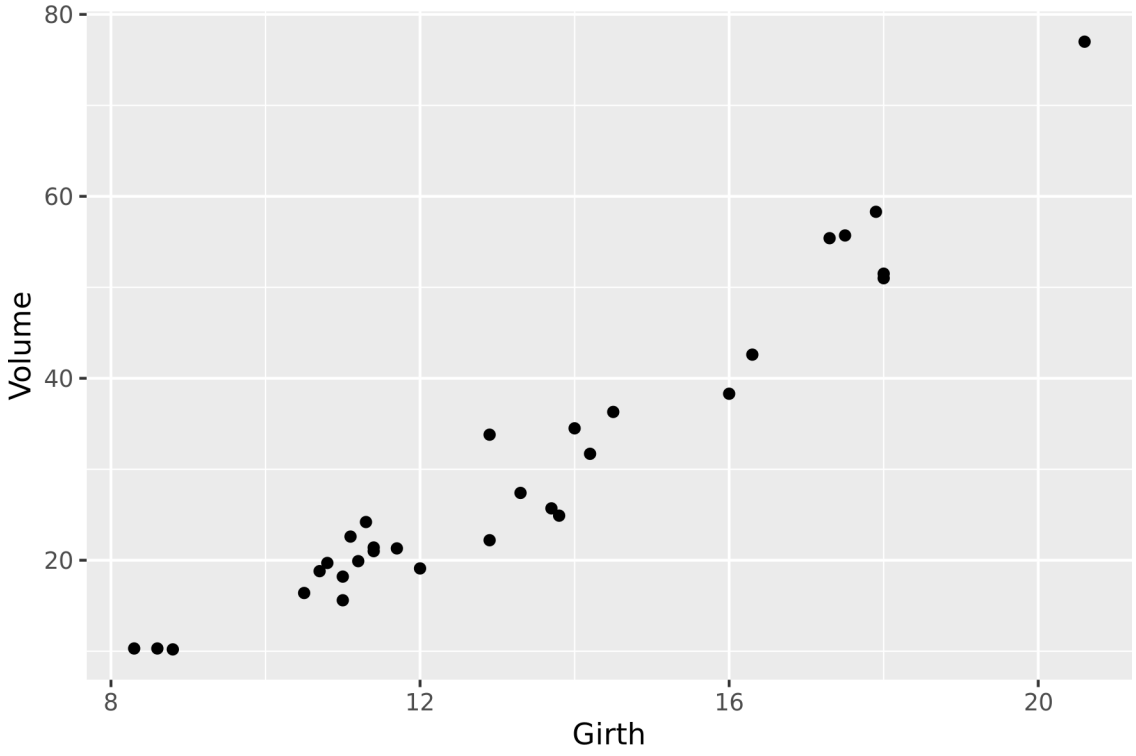
```
tree_dt <- as.data.frame(datasets::trees)

# visualizing the data with scatterplots
#- height vs. volume
tree_dt  %>%
  ggplot(aes(x = Height , y = Volume)) + geom_point()
```




```
cor(tree_dt$Volume, tree_dt$Height)
```

```
## [1] 0.5982497
```

```
#.- diameter vs. volume
tree_dt  %>%
  ggplot(aes(x = Girth , y = Volume)) + geom_point()
```



```r
cor(tree_dt$Volume, tree_dt$Girth)
```

```
## [1] 0.9671194
```

```r
# Let's answer questions using linear regression

  # - Describe the relationship between volume and height of these trees.
  # - Describe the relationship between volume and diameter of these trees.
  # - The summarizing the model results from the lm() function will provide valuable numerical insights
# analyzing the data with lm()

mdla <- lm(Volume ~ Height, tree_dt)
mdlb <- lm(Volume ~ Girth, tree_dt)

summary(mdla)
```

```
##
## Call:
## lm(formula = Volume ~ Height, data = tree_dt)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -21.274  -9.894  -2.894  12.068  29.852
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -87.1236    29.2731  -2.976 0.005835 **
## Height        1.5433     0.3839   4.021 0.000378 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.4 on 29 degrees of freedom
## Multiple R-squared:  0.3579, Adjusted R-squared:  0.3358
## F-statistic: 16.16 on 1 and 29 DF,  p-value: 0.0003784
```

```r
summary(mdlb)
```

```
##
## Call:
## lm(formula = Volume ~ Girth, data = tree_dt)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -8.065 -3.107  0.152  3.495  9.587
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -36.9435     3.3651  -10.98 7.62e-12 ***
## Girth         5.0659     0.2474   20.48  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.252 on 29 degrees of freedom
## Multiple R-squared:  0.9353, Adjusted R-squared:  0.9331
## F-statistic: 419.4 on 1 and 29 DF,  p-value: < 2.2e-16
```

ANSWER: -

- Direction : Both Height and Girth have a positive direction relationhip: meaning a positive change in either of these variables gives a positive relationship in Volume of tree
- Form : Both have a linear relationship with volume
- Strength : correlation : 0.5982497 for Height correlation : 0.9671194 for Girth
- Girth is significanty correlated with Volume
- Looking at the summary of model statistics,
- Adjusted R-squared: 0.3358 p-value -0.000378 ***
- Adjusted R-squared: 0.9331 p-value: < 2.2e-16 ***
- Girth explains 0.9331 of the variability of the volume of tree
- and has much smaller p - value
- Because of these above reasons, i will prefer to use Girth to predict Volume of
- another set of cherry trees based on linear regression model.

Q3. OIS: Logistic regression (1 point)

OIS v3 8.16 Challenger disaster, Part I.

On January 28, 1986, a routine launch was anticipated for the Challenger space shuttle. Seventy-three seconds into the flight, disaster happened: the shuttle broke apart, killing all seven crew members on board. An investigation into the cause of the disaster focused on a critical seal called an O-ring, and it is believed that damage to these O-rings during a shuttle launch may be related to the ambient temperature during the launch.

The orings.txt file in the data folder contains data on the temperature and number of damaged O-rings for 23 shuttle missions, where the mission order is based on the temperature at the time of the launch. Temp gives the temperature in Fahrenheit, Damaged represents the number of damaged O-rings. There are 6 O-rings total, so the number of undamaged O-rings can be calculated.

Visualize the data.

- what relationships do you observe between temperature and failure?

Create a logistic regression model.

- classify each case as having either damaged or undamaged O-rings (1 or 0)
  - a binary "failure" variable will help us determine probability of failure as a result
- use temperature as a predictor
- use the glm() function
- display the summary statistics of your model

Based on the model, do you think concerns regarding O-rings are justified? Explain.

- what does the p-value tell you?

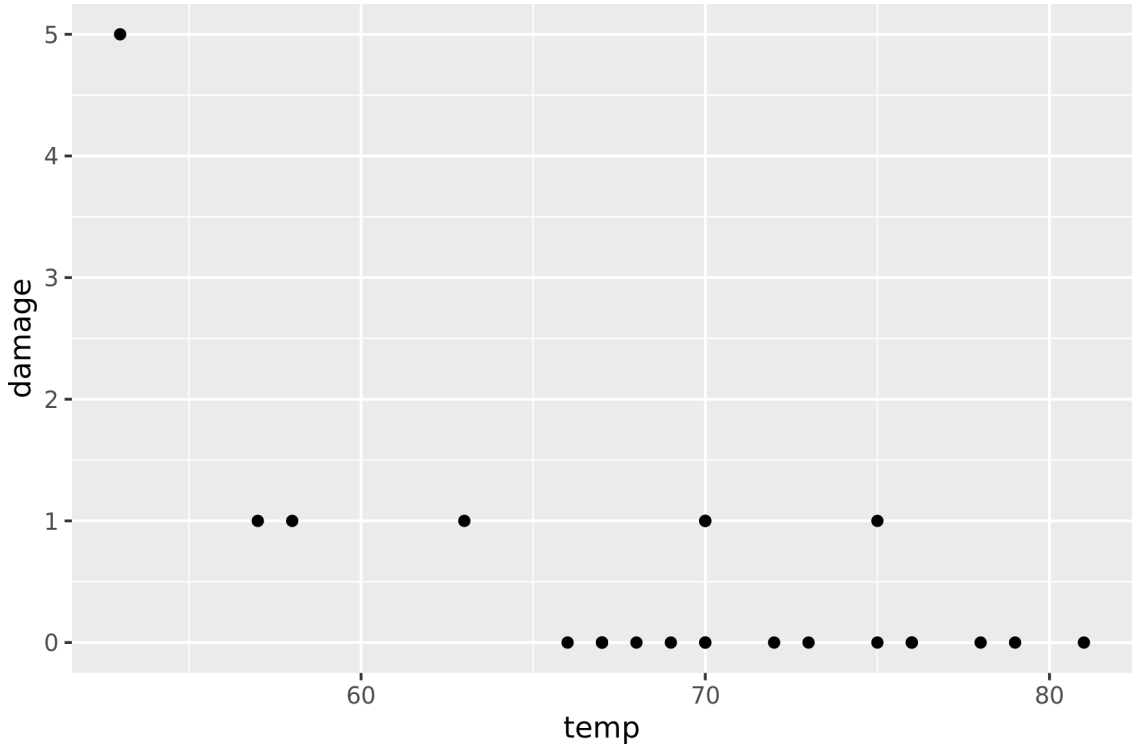
What assumption has to be made for logistic regression to hold valid in this case?

```
read.table("data/orings.txt", header = TRUE)
```

```
##    temp damage
## 1    53      5
## 2    57      1
```

```
## 3    58      1
## 4    63      1
## 5    66      0
## 6    67      0
## 7    67      0
## 8    67      0
## 9    68      0
## 10   69      0
## 11   70      1
## 12   70      0
## 13   70      1
## 14   70      0
## 15   72      0
## 16   73      0
## 17   75      0
## 18   75      1
## 19   76      0
## 20   76      0
## 21   78      0
## 22   79      0
## 23   81      0
```

```r
oring <- read.table("data/orings.txt", header = TRUE)



# initial EDA
# Visualize the data.
#
#   - what relationships do you observe between temperature and failure?

# In general, lower temp is associated with damaged O-rings but at higher temp they
# all seem to be undamaged.
ggplot(oring, aes(x = temp, y = damage)) + geom_point()
```




```
# logistic regression model

??glm()


## create failure column (binary, 1 or 0)
oring <- oring %>%
  mutate(failure = if_else(damage > 0, 1 ,0))


## use and summarize the glm() function

mdl <- glm(failure ~ temp,  data = oring, family = "binomial")
summary(mdl)
```

```
##
## Call:
## glm(formula = failure ~ temp, family = "binomial", data = oring)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.0611  -0.7613  -0.3783   0.4524   2.2175
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  15.0429     7.3786   2.039   0.0415 *
## temp         -0.2322     0.1082  -2.145   0.0320 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 28.267  on 22  degrees of freedom
## Residual deviance: 20.315  on 21  degrees of freedom
## AIC: 24.315
##
## Number of Fisher Scoring iterations: 5
```

ANSWER:

In general, lower temp is associated with damaged O-rings but at higher temp they all seem to be undamaged.

Statistically significant: 0.0320 * p- value is less than 0.05 There is some relationship between temp and faiure rate.

Independence of errors, linearity in the logit for continuous variables, absence of multicollinearity, and lack of strongly influential outliers.

Q4. Logistic regression: Palmer's penguins (3 points)

Let's make some logistic models on Palmer's penguins.

- we've looked at regression with a single predictor
- let's make a logistic model with multiple predictors
  - we're increasing the dimensions of the model in order to get more information out of the data
- we want to create a model that can predict penguin species

We will used the package nnet

- This package is used for machine learning
- But it also has a function
  - nnet::multinom()
    - Which works like stats::lm()
    - see ?nnet::multinom() for more help
- This is because logistic models
  - Are used to baseline more complex Machine Learning Models
    - For performance
- nnet::multinom() is used to build multiple logistic models
  - Which can be used to classify multiple outputs
  - Instead of a binary like a logistic model

```
library(nnet)
library(caret)
```

```
## Loading required package: lattice
```

```
##
## Attaching package: 'caret'
```

```
## The following object is masked from 'package:purrr':
##
##     lift
```

```
glimpse(palmerpenguins::penguins)
```

```
## Rows: 344
## Columns: 8
```

```
## $ species           <fct> Adelie, Adelie, Adelie, Adelie, Adelie, Adelie, Adel~
## $ island            <fct> Torgersen, Torgersen, Torgersen, Torgersen, Torgerse~
## $ bill_length_mm    <dbl> 39.1, 39.5, 40.3, NA, 36.7, 39.3, 38.9, 39.2, 34.1, ~
## $ bill_depth_mm     <dbl> 18.7, 17.4, 18.0, NA, 19.3, 20.6, 17.8, 19.6, 18.1, ~
## $ flipper_length_mm <int> 181, 186, 195, NA, 193, 190, 181, 195, 193, 190, 186~
## $ body_mass_g       <int> 3750, 3800, 3250, NA, 3450, 3650, 3625, 4675, 3475, ~
## $ sex               <fct> male, female, female, NA, female, male, female, male~
## $ year              <int> 2007, 2007, 2007, 2007, 2007, 2007, 2007, 2007, 2007~
```

Q4.1 Setup Training & Testing Dataframes

Processing the data

- divide the data into training and testing datasets
  - Using caret::createDataPartiton()
  - Which will divide the data into groups
    - So we can train the model on one subset
    - And use the other subset to test the models accuracy

```
# Get the partition
# Gives a list of indices that equally represent the column,on which it is partitioned

df <- palmerpenguins::penguins
idx <- createDataPartition(df$species, p = 0.8, list = FALSE,
                                    )
#Subset the data
#Create training and testing dfs
training <- df[idx, ] # use the indices to obtain the ful rows
testing <- anti_join(df, training) # get the leftover rows
```

```
## Joining, by = c("species", "island", "bill_length_mm", "bill_depth_mm", "flipper_length_mm", "body_ma
```

ANSWER: Divided the main dataset into training and testing, with 80% in training and 20 % testing.

Q4.2 Build a logistic regression model

Build a logistic model

- to predict species
- Based on all the predictors in the dataset
  - The short hand way of doing this is to use
  - model_function_call(Response_Variable ~ .)
  - Where the . will automatically use the rest of the columns
    - As predictors
- Once you have obtained your model object
- It's time to use stats::predict()
  - Which takes in a model object
  - Then applies it to new data (a testing subset)
- Different model types can have different prediction classes
  - Here we are trying to predict class

```
#Build your model

fit <- multinom(species ~ ., data = training)
```

```
## # weights:  30 (18 variable)
## initial  value 296.625318
## iter  10 value 28.299855
## iter  20 value 0.236814
## iter  30 value 0.117415
## iter  40 value 0.075803
## iter  50 value 0.003229
## iter  60 value 0.000717
## iter  70 value 0.000557
## iter  80 value 0.000338
## iter  90 value 0.000311
## iter 100 value 0.000140
## final  value 0.000140
## stopped after 100 iterations
```

```
## what does binomial do

summary(fit)
```

```
## Call:
## multinom(formula = species ~ ., data = training)
##
## Coefficients:
##            (Intercept) islandDream islandTorgersen bill_length_mm bill_depth_mm
## Chinstrap -0.012167049    10.35766       -3.982476      12.366254     -6.565064
## Gentoo     0.001310889    -3.30042       -3.280596       3.743364     -7.130813
##           flipper_length_mm body_mass_g   sexmale       year
## Chinstrap         0.6887773 -0.05471645 -8.168277 -0.1746627
## Gentoo            2.8599981  0.03474133 -3.673958 -0.3810283
##
## Std. Errors:
##           (Intercept) islandDream islandTorgersen bill_length_mm bill_depth_mm
## Chinstrap  0.01301750  101.225906        87.83527       21.76273      68.35890
## Gentoo     0.01404213    9.333284        16.21564      120.23252      74.57756
##           flipper_length_mm body_mass_g  sexmale      year
## Chinstrap          17.97934   0.4404898 32.95738  1.530114
## Gentoo            101.29747   2.0214080 12.73849 12.532492
##
## Residual Deviance: 0.000279657
## AIC: 36.00028
```

```
#Predict classes
predict (fit, testing)
```

```
##  [1] <NA>      Adelie    Adelie    <NA>      <NA>      Adelie    Adelie
##  [8] Adelie    Adelie    Adelie    Adelie    Adelie    Adelie    Adelie
## [15] Adelie    Adelie    Adelie    Adelie    Adelie    Adelie    Adelie
## [22] Adelie    Adelie    Adelie    Adelie    Adelie    Adelie    Gentoo
## [29] Adelie    Adelie    Gentoo    Gentoo    Gentoo    Gentoo    Gentoo
## [36] Gentoo    Gentoo    Gentoo    Gentoo    Gentoo    Gentoo    Gentoo
## [43] Gentoo    Gentoo    Gentoo    Gentoo    Gentoo    <NA>      Gentoo
## [50] Gentoo    Gentoo    Gentoo    Gentoo    Gentoo    Chinstrap Chinstrap
## [57] Chinstrap Adelie    Chinstrap Chinstrap Chinstrap Chinstrap Gentoo
## [64] Chinstrap Chinstrap Chinstrap Chinstrap
## Levels: Adelie Chinstrap Gentoo
```

ANSWER:

Q4.3 Evaluate accuracy on your test data

Evaluate the accuracy of your model against test data

- create a confusion matrix to evaluate your results
- using caret::confusionMatrix()
  - This compares the predicted class against the actual class
  - Which shows how the model classified the data

Are there any things that were commonly misclassified?

- Why do you think the model had trouble with these?
- What can be done to improve this model?

```
#Create Confusion Matrix
#From prediction and observed data


confusionMatrix(predict (fit, testing), testing$species)
```

```
## Confusion Matrix and Statistics
##
##            Reference
## Prediction  Adelie Chinstrap Gentoo
##   Adelie        26         1      0
##   Chinstrap      0        11      0
##   Gentoo         1         1     23
##
## Overall Statistics
##
##                Accuracy : 0.9524
##                  95% CI : (0.8671, 0.9901)
##     No Information Rate : 0.4286
##     P-Value [Acc > NIR] : <2e-16
##
##                   Kappa : 0.9251
##
##  Mcnemar's Test P-Value : 0.3916
##
## Statistics by Class:
##
##                      Class: Adelie Class: Chinstrap Class: Gentoo
## Sensitivity                 0.9630           0.8462        1.0000
## Specificity                 0.9722           1.0000        0.9500
## Pos Pred Value              0.9630           1.0000        0.9200
## Neg Pred Value              0.9722           0.9615        1.0000
## Prevalence                  0.4286           0.2063        0.3651
## Detection Rate              0.4127           0.1746        0.3651
## Detection Prevalence        0.4286           0.1746        0.3968
## Balanced Accuracy           0.9676           0.9231        0.9750
```

ANSWER:

- Prediction Adelie Chinstrap Gentoo
- Adelie 29 2 0

- Chinstrap 0 11 0
- Gentoo 0 0 22

There are 2 Chinstraps that were misclassified as Adelie. Since, I know that year is an uneccessary variable ansd cannot possibly determine the species, i remove it to get a accuarcy of 100 %

- Prediction Adelie Chinstrap Gentoo
- Adelie 29 0 0
- Chinstrap 0 13 0
- Gentoo 0 0 22

```
#Build your model

fit2 <- multinom(species ~ island  + bill_length_mm + bill_depth_mm + flipper_length_mm + body_mass_g +
```

```
## # weights:  27 (16 variable)
## initial  value 296.625318
## iter  10 value 12.663556
## iter  20 value 2.643019
## iter  30 value 0.052425
## iter  40 value 0.007243
## iter  50 value 0.000710
## iter  60 value 0.000659
## iter  70 value 0.000624
## iter  80 value 0.000589
## iter  90 value 0.000350
## iter 100 value 0.000161
## final  value 0.000161
## stopped after 100 iterations
```

```
## what does binomial do

summary(fit2)
```

```
## Call:
## multinom(formula = species ~ island + bill_length_mm + bill_depth_mm +
##     flipper_length_mm + body_mass_g + sex, data = training)
##
## Coefficients:
##           (Intercept) islandDream islandTorgersen bill_length_mm bill_depth_mm
## Chinstrap  -172.01570    46.84830       -4.496149       13.88097     -20.54374
## Gentoo      -10.02645   -46.92715      -37.108729       10.01853     -19.40838
##           flipper_length_mm body_mass_g   sexmale
## Chinstrap        -0.2034052 -0.01632525 -22.36892
## Gentoo           -1.3655929  0.04494848 -12.62046
##
## Std. Errors:
##           (Intercept)  islandDream islandTorgersen bill_length_mm bill_depth_mm
## Chinstrap   4.1527827 4.486238e+00             NaN     142.077849     66.014857
## Gentoo      0.1916737 9.665459e-15    5.087399e-07       7.459623      3.291453
##           flipper_length_mm body_mass_g   sexmale
## Chinstrap          24.01204    1.110742 0.2258676
## Gentoo             28.28494    1.289562 0.3486594
##
## Residual Deviance: 0.0003225712
## AIC: 32.00032
```

```
#Predict classes
predict (fit2, testing)
```

```
##  [1] <NA>      Adelie    Adelie    <NA>      <NA>      Adelie    Adelie
##  [8] Adelie    Adelie    Adelie    Adelie    Adelie    Adelie    Adelie
## [15] Adelie    Adelie    Adelie    Adelie    Adelie    Adelie    Adelie
## [22] Adelie    Adelie    Adelie    Adelie    Adelie    Adelie    Adelie
## [29] Adelie    Adelie    Gentoo    Gentoo    Gentoo    Gentoo    Gentoo
## [36] Gentoo    Gentoo    Gentoo    Gentoo    Gentoo    Gentoo    Gentoo
## [43] Gentoo    Gentoo    Gentoo    Gentoo    Gentoo    <NA>      Gentoo
## [50] Gentoo    Gentoo    Gentoo    Gentoo    Gentoo    Chinstrap Chinstrap
## [57] Chinstrap Chinstrap Chinstrap Chinstrap Chinstrap Chinstrap Chinstrap
## [64] Chinstrap Chinstrap Chinstrap Chinstrap
## Levels: Adelie Chinstrap Gentoo
```

```
confusionMatrix(predict (fit2, testing), testing$species)
```

```
## Confusion Matrix and Statistics
##
##            Reference
## Prediction  Adelie Chinstrap Gentoo
##   Adelie        27         0      0
##   Chinstrap      0        13      0
##   Gentoo         0         0     23
##
## Overall Statistics
##
##                Accuracy : 1
##                  95% CI : (0.9431, 1)
##     No Information Rate : 0.4286
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 1
##
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: Adelie Class: Chinstrap Class: Gentoo
## Sensitivity                 1.0000           1.0000        1.0000
## Specificity                 1.0000           1.0000        1.0000
## Pos Pred Value              1.0000           1.0000        1.0000
## Neg Pred Value              1.0000           1.0000        1.0000
## Prevalence                  0.4286           0.2063        0.3651
## Detection Rate              0.4286           0.2063        0.3651
## Detection Prevalence        0.4286           0.2063        0.3651
## Balanced Accuracy           1.0000           1.0000        1.0000
```

Q5 Houston Crime Reports

We will be working with the Huston crime data file provided by the ggmap package, ggmap::crimes.

- This CSV file contains the location (latitude and longitude)

- For crimes reported
- From January 2010 - August 2010

```
library(ggmap)
```

```
## Google's Terms of Service: https://cloud.google.com/maps-platform/terms/.
```

```
## Please cite ggmap if you use it! See citation("ggmap") for details.
```

```
library(sp)
library(rgdal)
```

```
## rgdal: version: 1.5-23, (SVN revision 1121)
## Geospatial Data Abstraction Library extensions to R successfully loaded
## Loaded GDAL runtime: GDAL 2.4.2, released 2019/06/28
## Path to GDAL shared files: /usr/share/gdal
## GDAL binary built with GEOS: TRUE
## Loaded PROJ runtime: Rel. 5.2.0, September 15th, 2018, [PJ_VERSION: 520]
## Path to PROJ shared files: (autodetected)
## Linking to sp version:1.4-5
```

```
library(leaflet)
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
```

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

```
library(RColorBrewer)
library(classInt)
library(tidyverse)
```

Q5.1 EDA to identify trends

Exploratory Data Analysis (EDA)

Let's do some exploratory data analysis on the data

- we'll start with the temporal aspect of the crime data.
- what can we say about **when** people commit crimes?

What trends do you see looking at different time frames?

- what months have particularly high crime rates?
- what times of day have increased crime rates?
- what days of the week have higher crime rates?
- produce three different visuals that represent each of these trends.

(hint: histograms are helpful for showing distributions)

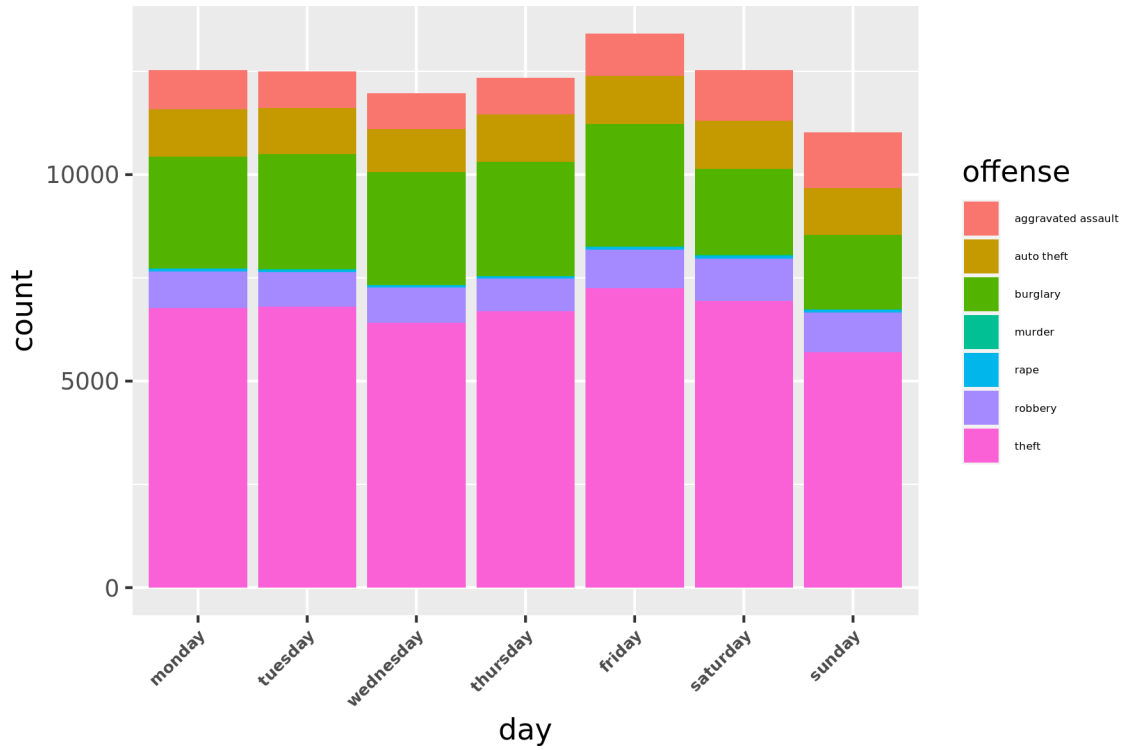
Which of these trends could you have predicted? Does anything surprise you?

Are there any relationships between types of crime and time of day?

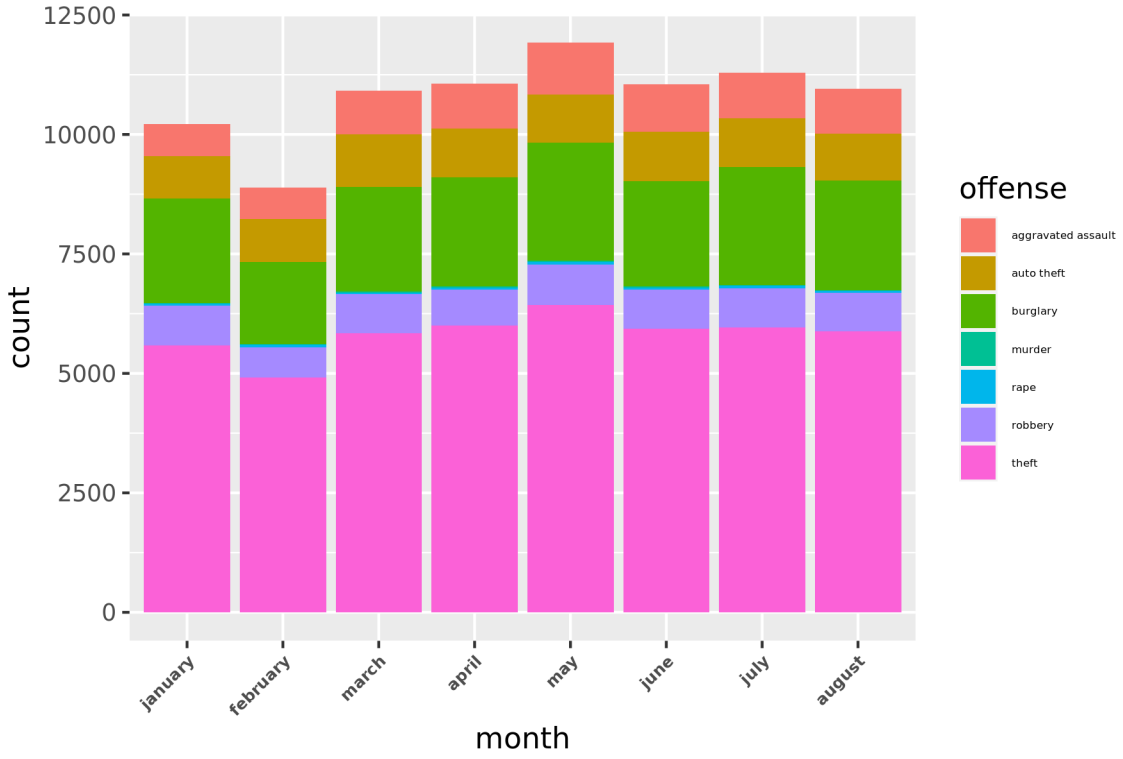
- produce a stacked histogram and comment on the results

```
# Load Data
t <- theme(legend.text = element_text(size = 4) ) +
```

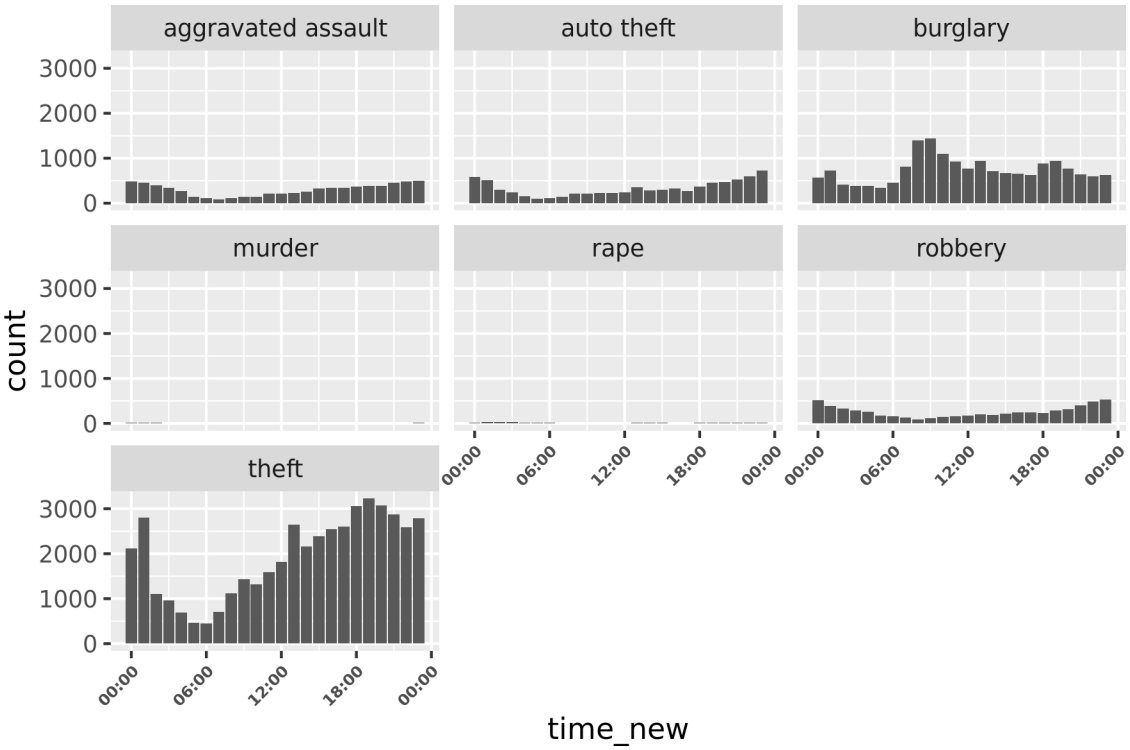
```
theme(legend.key.size = unit(0.5, 'cm') ,
axis.text.x = element_text(size = 6,
face = "bold", angle = 45, hjust = 1))

Houston_Crime_Reports <- ggmap::crime

#Crime per Day
# what days of the week have higher crime rates?
Houston_Crime_Reports <- Houston_Crime_Reports %>%
separate(time, c("date_new", "time_new"), " ", extra = "merge")%>%
mutate(day_week = wday(date_new, label=T) )

Houston_Crime_Reports %>%
  ggplot(aes(x = day, fill = offense)) + geom_bar(stat = "count") + t
```



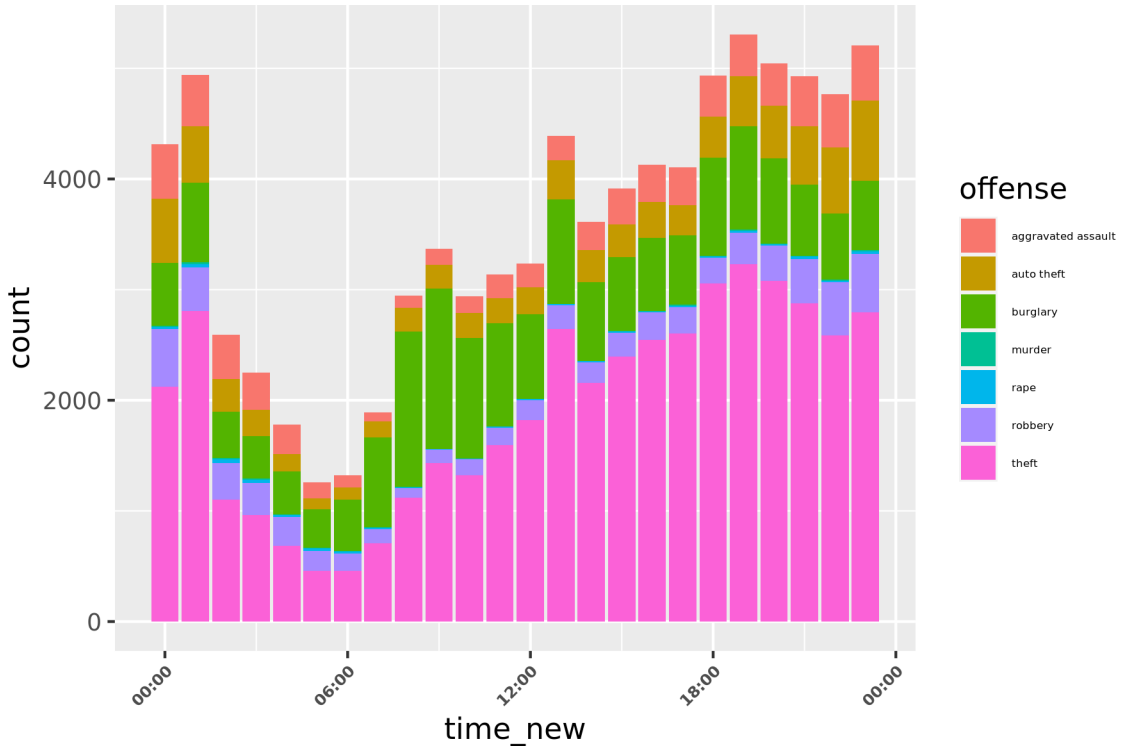

```
## why are there so many Nas
na_fil <-Houston_Crime_Reports %>%
  filter(is.na(day))


#Crime Per Month
# - what months have particularly high crimes rates?
Houston_Crime_Reports %>%
  ggplot(aes(x = month, fill = offense)) + geom_bar(stat = "count") + t
```




```
#Crime Per Hour
Houston_Crime_Reports %>%
mutate(time_new = as.POSIXct(hms::parse_hm(time_new))) %>%
ggplot(aes(x = time_new)  ) +
geom_bar() +
scale_x_datetime(date_labels = "%H:%M") + t + facet_wrap(~ offense)
```



```
#Crime Per Hour
Houston_Crime_Reports %>%
mutate(time_new = as.POSIXct(hms::parse_hm(time_new))) %>%
  ggplot(aes(x = time_new , fill = offense)  ) +
geom_bar() +
scale_x_datetime(date_labels = "%H:%M") + t
```



ANSWER: The highest number of crimes happen on Friday, while the lowest number is on Sunday. The highest number of crimes happen in may , while the lowest happen in february. Crimes reduce as the morning approaches and are at the least between 5 to 7am, In the afternoon there is peak at 1 pm , there is peak in the evening at 7pm, when the highest number of crimes are reported. Crimes again peak as pitch black night approaches, from 11 pm to 1am.

Looking at the stacked histograms

Across the week, types of crime and their variations remains constant, with theft, burglary being the highest and eveything else varying equally and murder being the lowest.

Across the year, types of crime and their variations remains constant, with theft, burglary being the highest and eveything else varying equally and murder being the lowest.

Across a day, rapes happen only at night,robberies happen mostly at night, burglaries seem to peak 8 to 10. Theft peaks at 1pm in the afternoon and happens all the night hours Agravated assault minimises from 6 to 8 am

I could have predicted that rapes happen only during the night hours. Sundays are low crime days and Fridys are high crime days.

I could not have predicted the months of the highest and lowest crimes.

Q5.2 Geospatial Analysis

```
library(sp)
library(rgdal)
library(maptools)
```

```
## Checking rgeos availability: TRUE
```

```
library(broom)
```

Geospatial Analysis

Next we'll look at the spatial distribution of this data.

- Plot the data on a OpenStreetMap
  - Using source = "stamen"
- You will have to specify the location in the function call
  - This is because ggmap::get_map()
  - Defaults to Google Maps when a bounding box is not specified
  - A bounding box
    - Gives the boundaries of the map that is downloaded
    - Specified with a list
    - c(left = ' ' , right = ' ' , top = '', bottom =")
- Color the map based on
  - Type of Crime Reported

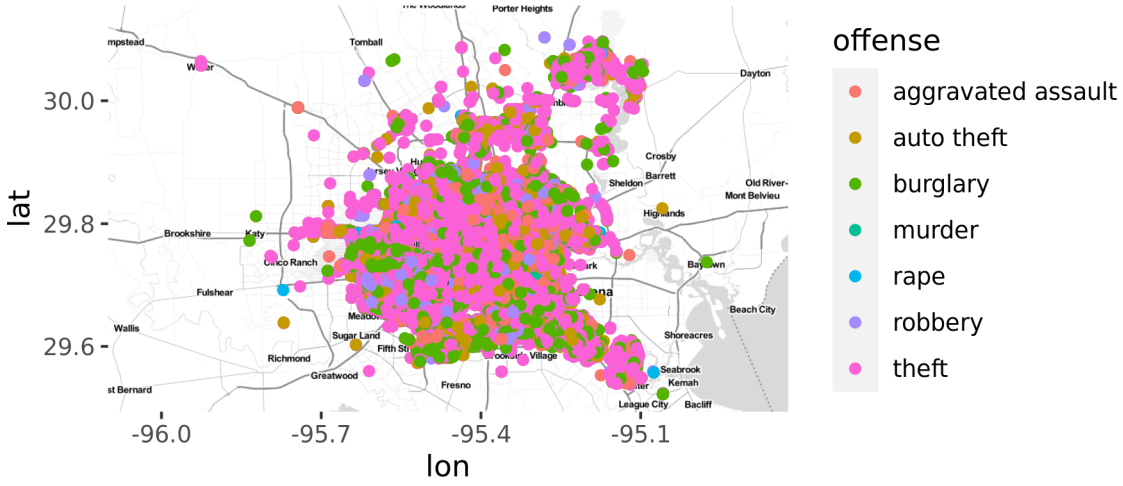
```
#Get the bbox

bbox = c(left = -96.1002 , bottom = 29.4934, right = -94.8230, top = 30.1546)


#Retrieve the map

houston_map <- get_stamenmap(
bbox,
maptype = "toner-lite",
zoom = 10
)
```

```
## Source : http://tile.stamen.com/toner-lite/10/238/421.png
## Source : http://tile.stamen.com/toner-lite/10/239/421.png
## Source : http://tile.stamen.com/toner-lite/10/240/421.png
## Source : http://tile.stamen.com/toner-lite/10/241/421.png
## Source : http://tile.stamen.com/toner-lite/10/242/421.png
## Source : http://tile.stamen.com/toner-lite/10/238/422.png
## Source : http://tile.stamen.com/toner-lite/10/239/422.png
## Source : http://tile.stamen.com/toner-lite/10/240/422.png
## Source : http://tile.stamen.com/toner-lite/10/241/422.png
## Source : http://tile.stamen.com/toner-lite/10/242/422.png
## Source : http://tile.stamen.com/toner-lite/10/238/423.png
## Source : http://tile.stamen.com/toner-lite/10/239/423.png
```

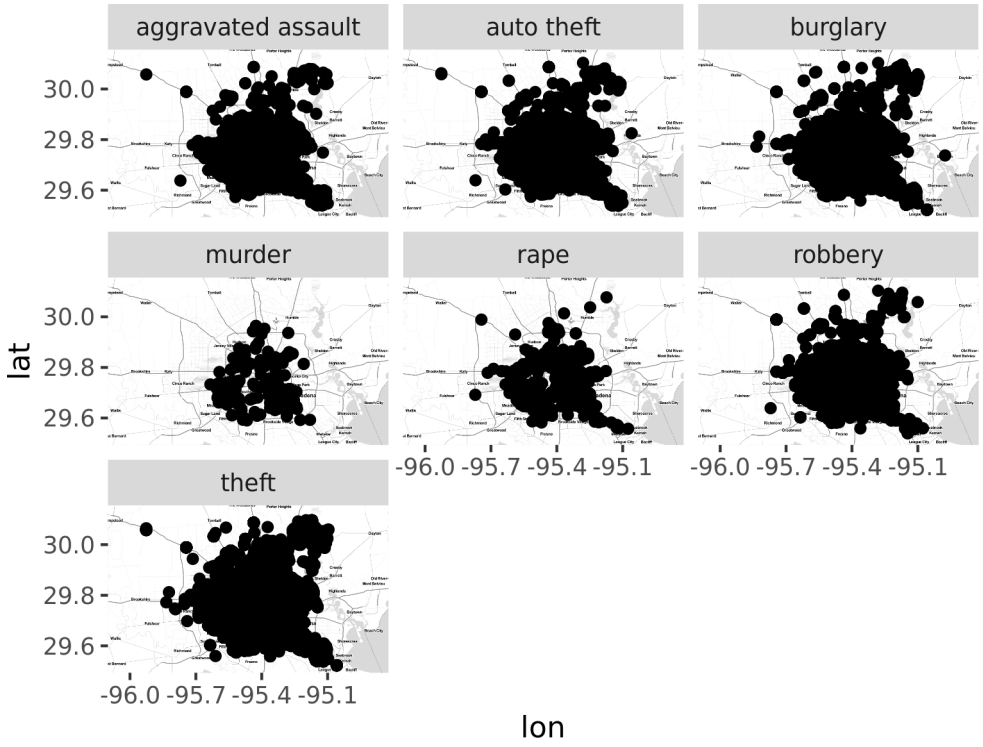
```
## Source : http://tile.stamen.com/toner-lite/10/240/423.png
## Source : http://tile.stamen.com/toner-lite/10/241/423.png
## Source : http://tile.stamen.com/toner-lite/10/242/423.png
## Source : http://tile.stamen.com/toner-lite/10/238/424.png
## Source : http://tile.stamen.com/toner-lite/10/239/424.png
## Source : http://tile.stamen.com/toner-lite/10/240/424.png
## Source : http://tile.stamen.com/toner-lite/10/241/424.png
## Source : http://tile.stamen.com/toner-lite/10/242/424.png
```

```
#Plot

rep1 <- ggmap(houston_map) +
  geom_point(data = Houston_Crime_Reports, aes(x = lon, y = lat, color = offense))

rep2 <- ggmap(houston_map) +
  geom_point(data = Houston_Crime_Reports, aes(x = lon, y = lat) ) + facet_wrap( ~ offense)
```

```
rep1
```




```
rep2
```



ANSWER:

Q5.3a Modify the incident occurrence layer, to better see whats happening

In the last map, it was a bit tricky

- to see the density of the incidents
  - because all the graphed points
  - were sitting on top of each other.

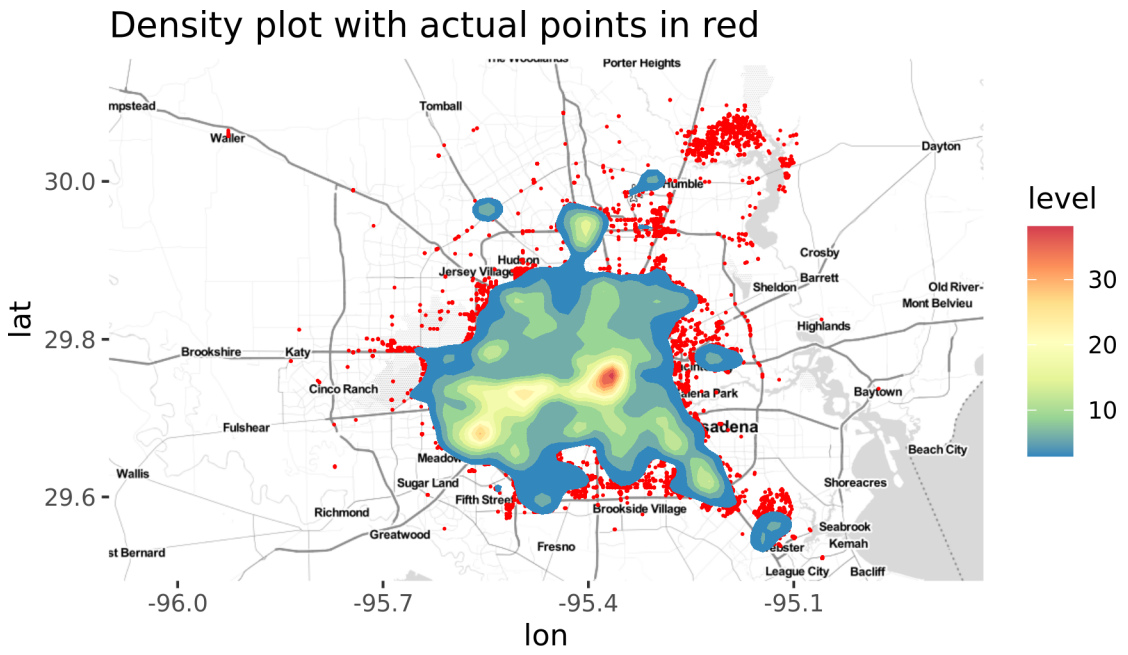
We're going to now modify the incident occurrence layer

- to plot the density of points
  - vs plotting each incident individually.

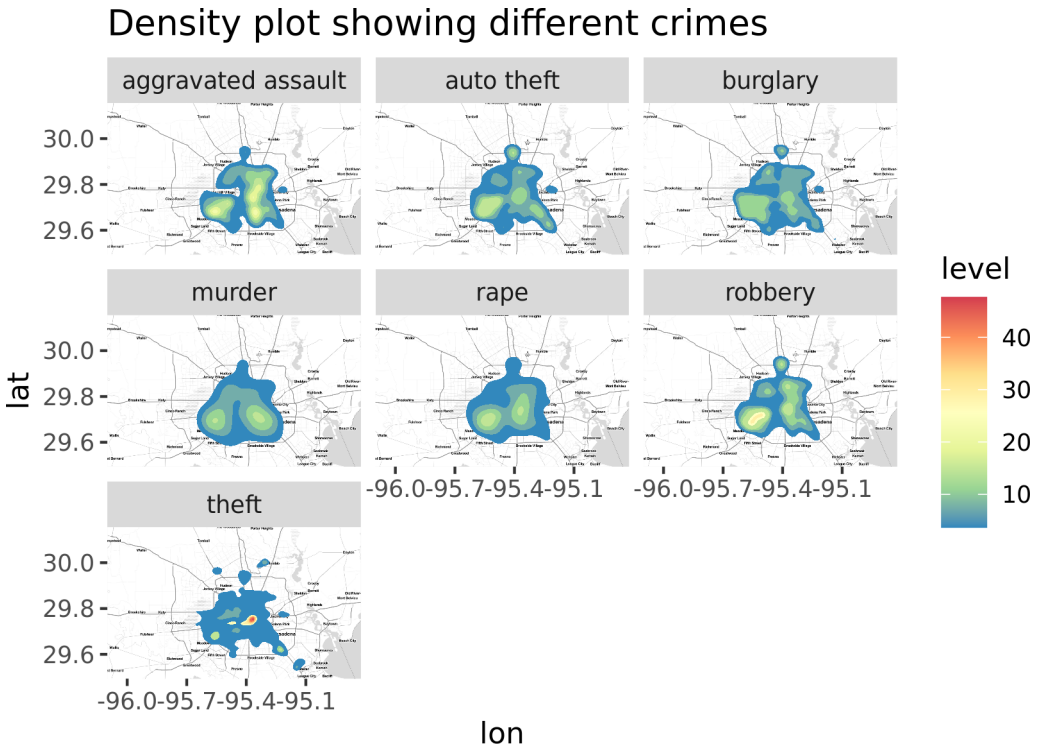
- We accomplish this with
  - the ggplot2::stat_density2d() function vs ggplot2::geom_point().

```
#Plot using stat_density2d()

??stat_density2d()



ggmap(houston_map) +
geom_point( data = Houston_Crime_Reports, aes(x = lon, y = lat ),
            color = "red", size =0.01)  +
stat_density2d(
aes(x = lon, y = lat , fill = ..level..),
size = 0.02, bins = 15, data = Houston_Crime_Reports,
geom = "polygon"
```

```
) + scale_fill_gradientn(colours=rev(brewer.pal(7,"Spectral")))+
  labs(title = "Density plot with actual points in red")
```




```
ggmap(houston_map)  +
stat_density2d(
aes(x = lon, y = lat , fill = ..level..),
size = 0.02, bins = 15, data = Houston_Crime_Reports,
geom = "polygon"
) +   scale_fill_gradientn(colours=rev(brewer.pal(7,"Spectral")))+
facet_wrap(~offense) + labs(title = "Density plot showing different crimes")
```



ANSWER:

Q5.3b What does ..level.. do

- What does ..level.. do
  - in the ggplot2::stat_density2d() function call?
  - Hint: Look at the help topics for this function

ANSWER: ..level.. is a parameter that is assigned to a point based on the number of data points in the neighborhood of that point, in a way a number that denotes density, higher the ..level.. more the data points around that neighborhood.

Q5.4 What is the safest and most dangerous neighborhoods

Finally

- Filter out a specific crime of your choosing
- Plot the crime density

We will use a new package that assists in geospatial analysis

- rdgal
- This package is used to transform and project geospatial objects
- It also has some nice functions for working with .shp files
  - .shp files contain information about regions on a map
  - i.e .shp files can contain the information
    - the size, shape, and location of countries or states
- Add Polygons for the specific Neighborhoods

    - Using NeighborShapefile
        * and rdgal::readOGR()
        * or mapdata::readShapeSpatial()

- What is the most dangerous Neighborhood for your crime
- Where is the safest neighborhood?
- Label the map with the neighborhoods
    - Hint: It's OK to remove some of the labels, there's a lot
        * ggplot2::geom_text() has a built in function for this
        * check overlap = TRUE

```
# Filter
# filter a particular crime  - burglaries
burglary_dt <- Houston_Crime_Reports %>%
  filter(offense == "burglary")




#Read Shapefile
texas_shp <- readOGR(dsn = "data/shp",
                     layer = "COH_SUPER_NEIGHBORHOODS")
```

```
## OGR data source with driver: ESRI Shapefile
## Source: "/home/kxg360/Git/21f-dsci351-351m-451-e1451-e2451-kxg360_backup/1-assignments/lab-exercise/l
## with 88 features
## It has 14 fields
```

```
#Find the location of the center of the neighborhood

cent  <- coordinates(texas_shp)
county <- texas_shp@data$SNBNAME
coord_tex <- cbind( coordinates(texas_shp), as.character(texas_shp@data$SNBNAME))
coord_tex <-as.data.frame(coord_tex)
head(coord_tex)
```

```
##                   V1               V2                          V3
## 0 -95.3807661413748 29.7571155590013                FOURTH WARD
## 1 -95.3279386713314 29.7509623114363                SECOND WARD
## 2 -95.3590963165381  29.756688573148                   DOWNTOWN
## 3 -95.2627140945746 29.7483969910759 CLINTON PARK TRI-COMMUNITY
## 4 -95.4756135530745 29.7565515822066              GREATER UPTOWN
## 5 -95.4811830907025   29.87441143736              GREATER INWOOD
```

```
coords <- SpatialPoints(burglary_dt[, c("lon", "lat")])
crime_spatial_df <- SpatialPointsDataFrame(coords, burglary_dt)
proj4string(crime_spatial_df) <- CRS("+proj=longlat +ellps=WGS84")

crime_df <- data.frame(crime_spatial_df)
texas_shp_df <- fortify(texas_shp)
```

```
## Regions defined for each Polygons
```

```
#Get the neigborhood names
texas_shp@data[["SNBNAME"]]
```

```
##  [1] FOURTH WARD
##  [2] SECOND WARD
##  [3] DOWNTOWN
##  [4] CLINTON PARK TRI-COMMUNITY
##  [5] GREATER UPTOWN
##  [6] GREATER INWOOD
##  [7] GREATER HOBBY AREA
##  [8] GOLFCREST / BELLFORT / REVEILLE
##  [9] ELDRIDGE / WEST OAKS
## [10] WASHINGTON AVENUE COALITION / MEMORIAL PARK
## [11] GREATER FIFTH WARD
## [12] DENVER HARBOR / PORT HOUSTON
## [13] PLEASANTVILLE AREA
## [14] NORTHSHORE
## [15] LAZYBROOK / TIMBERGROVE
## [16] GREATER HEIGHTS
## [17] KASHMERE GARDENS
## [18] MINNETEX
## [19] NEAR NORTHSIDE
## [20] SPRING BRANCH EAST
## [21] SPRING BRANCH NORTH
## [22] EL DORADO / OATES PRAIRIE
## [23] SPRING BRANCH CENTRAL
## [24] HUNTERWOOD
## [25] SETTEGAST
## [26] LANGWOOD
## [27] INDEPENDENCE HEIGHTS
## [28] CENTRAL NORTHWEST
## [29] TRINITY / HOUSTON GARDENS
## [30] CARVERDALE
## [31] EASTEX - JENSEN AREA
## [32] EAST HOUSTON
## [33] ACRES HOME
## [34] NORTHSIDE/NORTHLINE
## [35] HIDDEN VALLEY
## [36] EAST LITTLE YORK / HOMESTEAD
## [37] WILLOWBROOK
## [38] GREATER GREENSPOINT
## [39] IAH / AIRPORT AREA
## [40] KINGWOOD AREA
## [41] LAKE HOUSTON
## [42] FAIRBANKS / NORTHWEST CROSSING
## [43] WESTBRANCH
## [44] SHARPSTOWN
## [45] WESTWOOD
## [46] FORT BEND HOUSTON
## [47] FONDREN GARDENS
## [48] SOUTH BELT / ELLINGTON
## [49] SOUTH ACRES / CRESTMONT PARK
## [50] BRAYS OAKS
## [51] CENTRAL SOUTHWEST
## [52] SUNNYSIDE
## [53] ALIEF
## [54] PECAN PARK
```

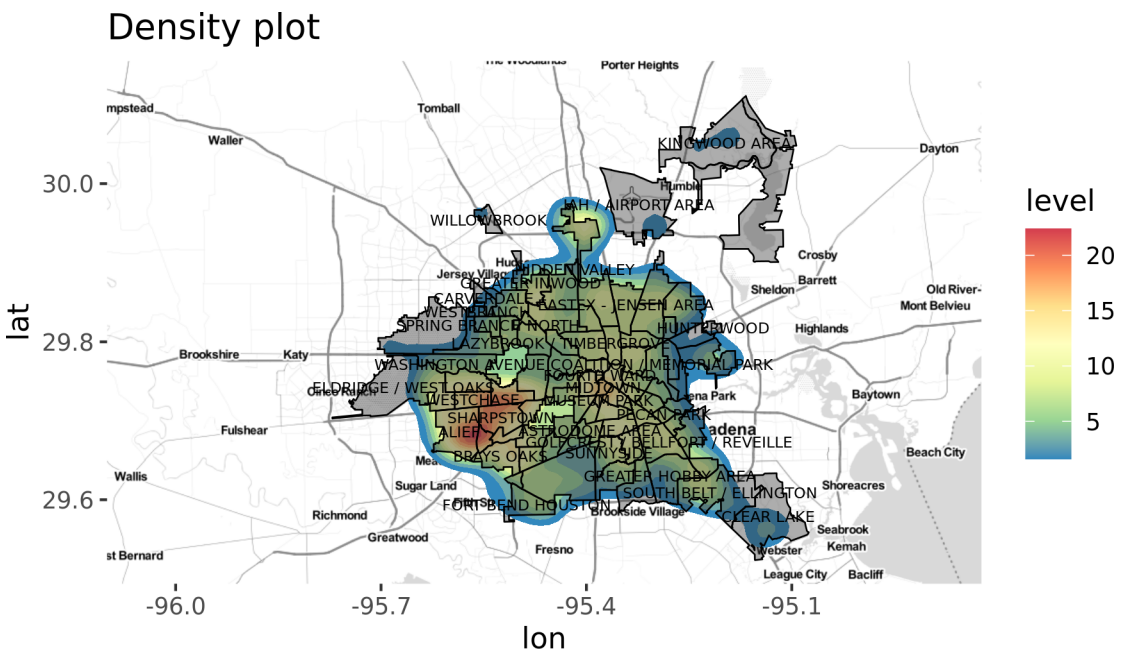
```
## [55] CLEAR LAKE
## [56] WESTBURY
## [57] WILLOW MEADOWS / WILLOWBEND AREA
## [58] BRAEBURN
## [59] SOUTH MAIN
## [60] SOUTH PARK
## [61] ASTRODOME AREA
## [62] GREATER OST / SOUTH UNION
## [63] PARK PLACE
## [64] MEADOWBROOK / ALLENDALE
## [65] MEDICAL CENTER AREA
## [66] GULFTON
## [67] MACGREGOR
## [68] GULFGATE RIVERVIEW / PINE VALLEY
## [69] HARRISBURG / MANCHESTER
## [70] UNIVERSITY PLACE
## [71] WESTCHASE
## [72] MUSEUM PARK
## [73] LAWNDALE / WAYSIDE
## [74] GREENWAY / UPPER KIRBY AREA
## [75] GREATER THIRD WARD
## [76] MID WEST
## [77] GREATER EASTWOOD
## [78] MIDTOWN
## [79] BRAESWOOD
## [80] MEYERLAND AREA
## [81] EDGEBROOK AREA
## [82] MAGNOLIA PARK
## [83] AFTON OAKS / RIVER OAKS AREA
## [84] BRIAR FOREST
## [85] NEARTOWN - MONTROSE
## [86] MEMORIAL
## [87] SPRING BRANCH WEST
## [88] ADDICKS PARK TEN
## 88 Levels: ACRES HOME ADDICKS PARK TEN AFTON OAKS / RIVER OAKS AREA ... WILLOWBROOK
```

```
# data_prep
coord_tex <- rename(coord_tex, lon = V1, lat = V2 , county = V3)

coord_tex$lon <- as.numeric(as.character(coord_tex$lon))
coord_tex$lat <- as.numeric(as.character(coord_tex$lat))
coord_tex$county <- as.character(coord_tex$county)

crime_df_new <- right_join(crime_df, coord_tex )
```

```
## Joining, by = c("lon", "lat")
```

```
#Plot with density + neighborhood + label
stat <-  ggmap(houston_map)  +
stat_density2d(aes(x = lon, y = lat , fill = ..level.. ),
size = 0.02, bins = 15, data = crime_df, geom = "polygon") +   scale_fill_gradientn(colours=rev(brewer.
  labs(title = "Density plot")


# county <- ggmap(unitedStatesmap) +  geom_polygon(aes( x = long, y = lat, group = group, label = as.ch
```

```
# alpha = .4, size = .3)


k <- stat + geom_polygon(aes( x = long, y = lat, group = group),
                         data = texas_shp_df, colour = "black",
alpha = .4, size = .3)

k + geom_text(data = crime_df_new,aes(x = lon, y = lat, label = county) ,
              check_overlap = TRUE, size= 2)
```



ANSWER: Based on an rudimentary look at the plot obtained.

The most dangerous neighborhood for burglaries :

The most safest neighborhood for burglaries : SHARPSTOWN , ALIEF

There are several neighbourhoods in the edge of houston , marked in blue that are the safest

Links

https://blog.dominodatalab.com/applied-spatial-data-science-with-r/

http://www.r-project.org

http://rmarkdown.rstudio.com/

https://www.openintro.org/stat/textbook.php?stat_book=os

https://en.wikipedia.org/wiki/Multivariate_kernel_density_estimation