Deliverable #3

Micah A. Perez

12/12/2020

Deliverable 3

For Deliverable 3, I didn't want to recycle any of my work from previous projects, but to instead add to and filter out data that I didn't need, as well as add more visualizations and models that are better built than my last deliverables. I also went through all my peer reviews to analyze potential adjustments and ideas that I could use for this deliverable. In my previous deliverables, I did a few things wrong. Some of my data was missing pieces, which I decided, in the end, to remove for this assignment. My models were also not as well thought out as I would have hoped, which inspired me to try to create a more interesting and better predicting model this time around (a good learning experience, but it unfortunately did not pan out how I had hoped).

Research Question - Deliverable 3

In this deliverable, I try to answer questions about the prevelance about the IT Sector in the US Economy. In my visualizations figure, I map out what areas tech economies are growing the fastest, and in my model, I attempt to answer the question: Is there a correlation between GDP per capita growth and IT sector growth?

Data

For my data in this deliverable, I wanted to add something new and interesting, that could be used for modeling. In the most recent peer review, one student brought up the idea of adding technology GDP, or population data to my data. Luckily, the same ".gov" website I used for my last deliverable (https://apps.bea.gov/iTable/index_regional.cfm) had GDP data on the information sector. I took this data through a csv, added it to my existing data, and then cut out the columns I felt that I didn't need. There was a slight lack of data on information GDP in smaller metro and micropolitian areas, so I cut those rows out as well to make it more accurate and clean.

```
library(tidyverse)
library(modelr)
library(tidyr)
#library(dplyr)
library(ggmap)
library(devtools)

#Read in Data Left off at Deliverable 2
Deliverable3_Data <- read.csv(file = 'final_data/Deliverable_3_Data.csv')

#Deliverable3_Data$X <- NULL
```

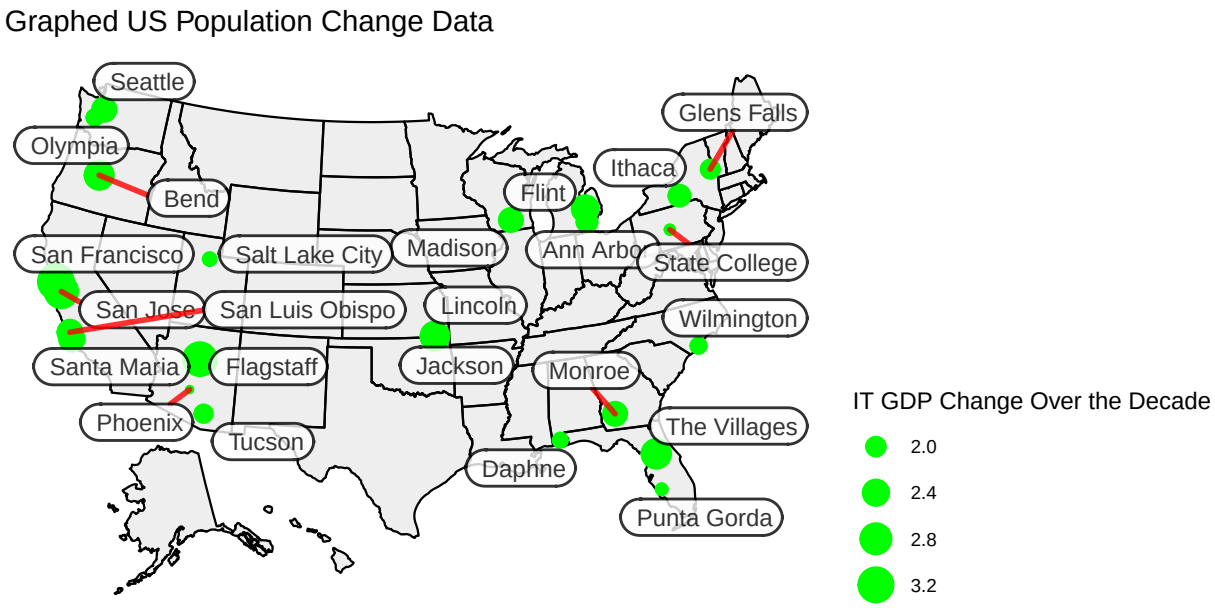
```r
Deliverable3_Data[5:20] <- list(NULL)

#Read In New IT Data from CSV's
Metro_IT_2010 <- read.csv(file = 'final_data/Metro_IT_GDP_2010.csv')
Metro_IT_2018 <- read.csv(file = 'final_data/Metro_IT_GDP_2018.csv')

#Created Dataframe with both columns of data
Metro_IT_Data <- data.frame("City" = Metro_IT_2010$City, "IT_GDP_2010" =  Metro_IT_2010$Tech_2010 , "IT_

#Make an index on GDP_2018 data to left_join with the Mains Index
Metro_IT_Data$X <- seq.int(nrow(Metro_IT_Data))
Metro_IT_Data$City <- NULL

#Do a Left join combining the city names and then delete a column
mainData <- left_join(Deliverable3_Data, Metro_IT_Data, by="X")
mainData$X <- NULL

#Delete rows with no IT Data
mainData <- subset(mainData, IT_GDP_2010 != "(D)")
mainData <- subset(mainData, IT_GDP_2018 != "(D)")

#Make IT GDP Data doubles (for future use)
mainData$IT_GDP_2010 <- as.double(as.character(mainData$IT_GDP_2010))
mainData$IT_GDP_2018 <- as.double(as.character(mainData$IT_GDP_2018))

#Multipy them both by 1000 (since original data was in 1000's)
mainData$IT_GDP_2010 <- (mainData$IT_GDP_2010*1000)
mainData$IT_GDP_2018 <- (mainData$IT_GDP_2018*1000)

#My Final Data
head(mainData)
```

```
##             City State      Region     GDP_2010     GDP_2018 Population_2010
## 1  United States    US          US 1.384544e+13 1.650475e+13       309321666
## 3          Akron    OH     Midwest 3.012429e+10 3.291284e+10          703031
## 5         Albany    OR        West 3.572329e+09 4.418178e+09          116891
## 6         Albany    NY   Northeast 4.889281e+10 5.236762e+10          871082
## 8     Alexandria    LA       South 5.612875e+09 5.660041e+09          154096
## 9      Allentown    PA   Northeast 3.717669e+10 4.051170e+10          821923
##   Population_2018 Decade_GDP_Change Population_Percent_Change
## 1       326687501      2.659305e+12                 1.0561417
## 3          703855      2.788549e+09                 1.0011721
## 5          127451      8.458490e+08                 1.0903406
## 6          882263      3.474810e+09                 1.0128358
## 8          152762      4.716600e+07                 0.9913431
## 9          842626      3.335019e+09                 1.0251885
##   GDP_Percent_Change  IT_GDP_2010  IT_GDP_2018
## 1           1.192071 727020547000 1.031010e+12
## 3           1.092568    752966000 1.158392e+09
## 5           1.236778     50296000 7.020900e+07
## 6           1.071070   1739709000 2.286471e+09
## 8           1.008403     92844000 6.744400e+07
## 9           1.089707   1834283000 2.541458e+09
```

Visualization

For this data, I was intrested in extracting some of it and creating new columns. For the visualization part of this, I decided to add the total information technology percentage change, as well as its total change. I also created a new column marked as "IT Importance". This data shows the amount of a metro area's infomation GDP as a percentage of its total GDP. My first visualization shows the top 25 Cities in the United States that had the highest IT Growth from 2010 to 2018. I also reused the Google api and usmap library to show them on a map view.

```
library(usmap) #Use USA MAP Library to Better visualize results

mainData$IT_Percent_Change <- (mainData$IT_GDP_2018/mainData$IT_GDP_2010)
mainData$IT_GDP_Change <- (mainData$IT_GDP_2018 - mainData$IT_GDP_2010)


Top_50_IT <- mainData %>%
    arrange(desc(IT_Percent_Change)) %>%
    slice(1:25)


#Google Maps API Key to have acess to data
register_google(key = "AIzaSyByXCki-hIHBM_HzbK_IE8d2xMZZYXEGLM")

#Used City names and Google Maps API to get coordinates and binded that to my data frame
Top_50_IT<- cbind(geocode(as.character(Top_50_IT$City)), Top_50_IT)

#Made all NA vaules 0 to avoid errors in code
Top_50_IT[is.na(Top_50_IT)] <- 0

#Transformed coordinate data to be readable by usmap plot
City_data_transformed <- usmap_transform(Top_50_IT)

City_data_transformed <- City_data_transformed[-c(2),]

plot_usmap(fill = "grey", alpha = 0.25) +
  geom_point(data=City_data_transformed, aes(x=lon.1, y=lat.1, size=IT_Percent_Change), color="green") +
  labs(title = "Graphed US Population Change Data", size = "IT GDP Change Over the Decade") +
  theme(legend.position = "right") +
  ggrepel::geom_label_repel(data = City_data_transformed,
            aes(x = lon.1, y = lat.1, label = City),
            size = 3, alpha = 0.8,
            label.r = unit(0.5, "lines"), label.size = 0.5,
            segment.color = "red", segment.size = 1,
            seed = 1002)
```

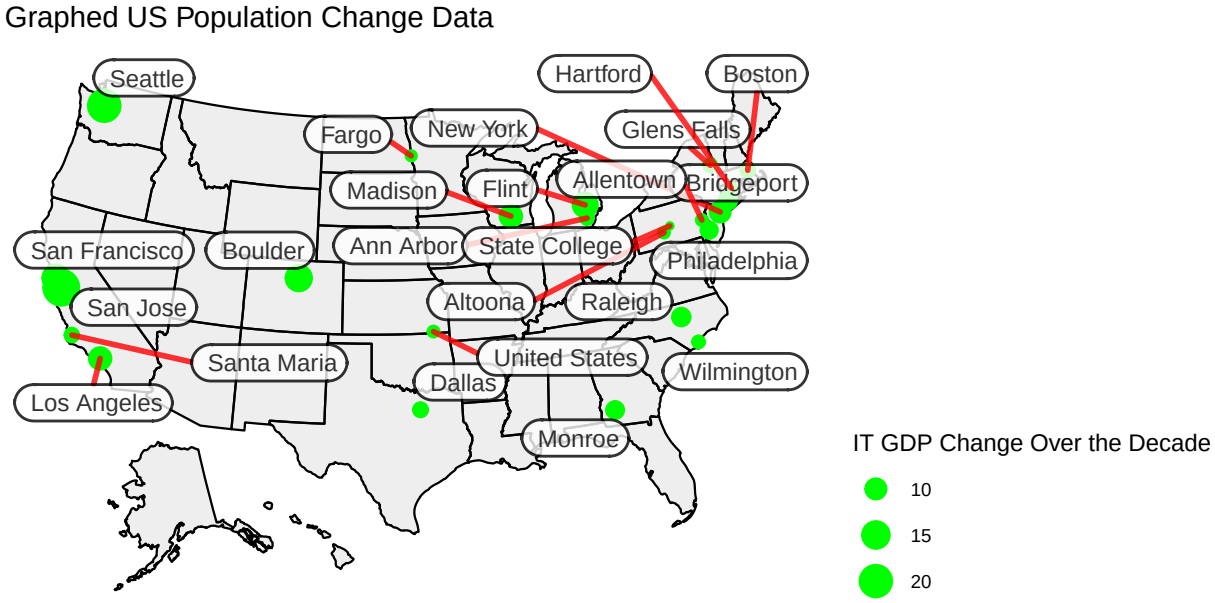


Visualization #2

Here, I created my IT importance column and visualized it on a us map similar to the graph before. I also used a gg plot and a bar chart to show the cities where the IT sector has the most prevelance over the total GDP. I was actually very interested in some of my findings and learned just how much of an impact information jobs have on cities, even outside of Silicon Valley and Seattle.

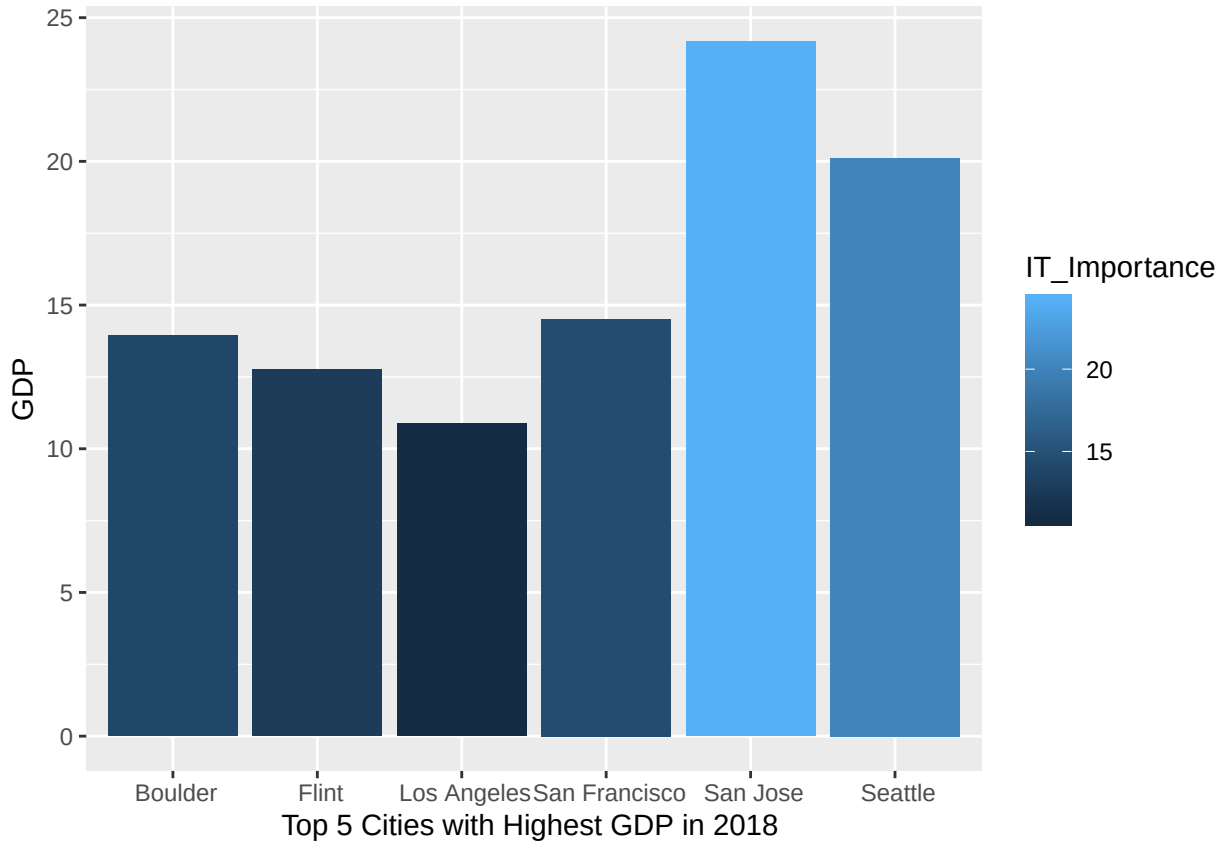
```
mainData$IT_Importance <- (mainData$IT_GDP_2018/mainData$GDP_2018 * 100)

Top_50_IT <- mainData %>%
    arrange(desc(IT_Importance)) %>%
    slice(1:26)

#Used City names and Google Maps API to get coordinates and binded that to my data frame
Top_50_IT<- cbind(geocode(as.character(Top_50_IT$City)), Top_50_IT)

#Transformed coordinate data to be readable by usmap plot
City_data_transformed <- usmap_transform(Top_50_IT)


City_data_transformed <- City_data_transformed[-c(24,25),]


plot_usmap(fill = "grey", alpha = 0.25) +
  geom_point(data=City_data_transformed, aes(x=lon.1, y=lat.1, size=IT_Importance), color="green") +
```

```
  labs(title = "Graphed US Population Change Data", size = "IT GDP Change Over the Decade") +
  theme(legend.position = "right") +
  ggrepel::geom_label_repel(data = City_data_transformed,
            aes(x = lon.1, y = lat.1, label = City),
            size = 3, alpha = 0.8,
            label.r = unit(0.5, "lines"), label.size = 0.5,
            segment.color = "red", segment.size = 1,
            seed = 1002)
```




```
mainData %>%
    arrange(desc(IT_Importance)) %>%
    slice(1:6) %>%
    ggplot(., aes(x=City, y=IT_Importance))+
              geom_bar(stat='identity', aes(fill = IT_Importance)) +
              print(labs(y="GDP", x = "Top 5 Cities with Highest GDP in 2018 ")) +
              scale_y_continuous(labels = scales::label_number_si())
```

```
## $y
## [1] "GDP"
##
## $x
## [1] "Top 5 Cities with Highest GDP in 2018 "
##
## attr(,"class")
## [1] "labels"
```



Model

When presenting my Deliverable 2, a fellow student brought up that I should do something with GDP per capita change or population density. After looking around for government data on population density, I realized that I was unable to find any, so I simply extracted the GDP per capita data from my existing data and created new columns. I also created a column on the percentage change, which is the data that I ended up using in my model.

```
library(caret)

mainData$GDP_Per_Capita_2010 <- (mainData$GDP_2010/mainData$Population_2010)
mainData$GDP_Per_Capita_2018 <- (mainData$GDP_2018/mainData$Population_2018)
mainData$GDP_Per_Capita_Percentage_Change <- (mainData$GDP_Per_Capita_2018/mainData$GDP_Per_Capita_2010

head(mainData)
```

```
##             City State      Region     GDP_2010     GDP_2018 Population_2010
## 1  United States    US          US 1.384544e+13 1.650475e+13       309321666
## 3          Akron    OH     Midwest 3.012429e+10 3.291284e+10          703031
## 5         Albany    OR        West 3.572329e+09 4.418178e+09          116891
## 6         Albany    NY   Northeast 4.889281e+10 5.236762e+10          871082
## 8     Alexandria    LA       South 5.612875e+09 5.660041e+09          154096
## 9      Allentown    PA   Northeast 3.717669e+10 4.051170e+10          821923
##   Population_2018 Decade_GDP_Change Population_Percent_Change
## 1       326687501      2.659305e+12                 1.0561417
```

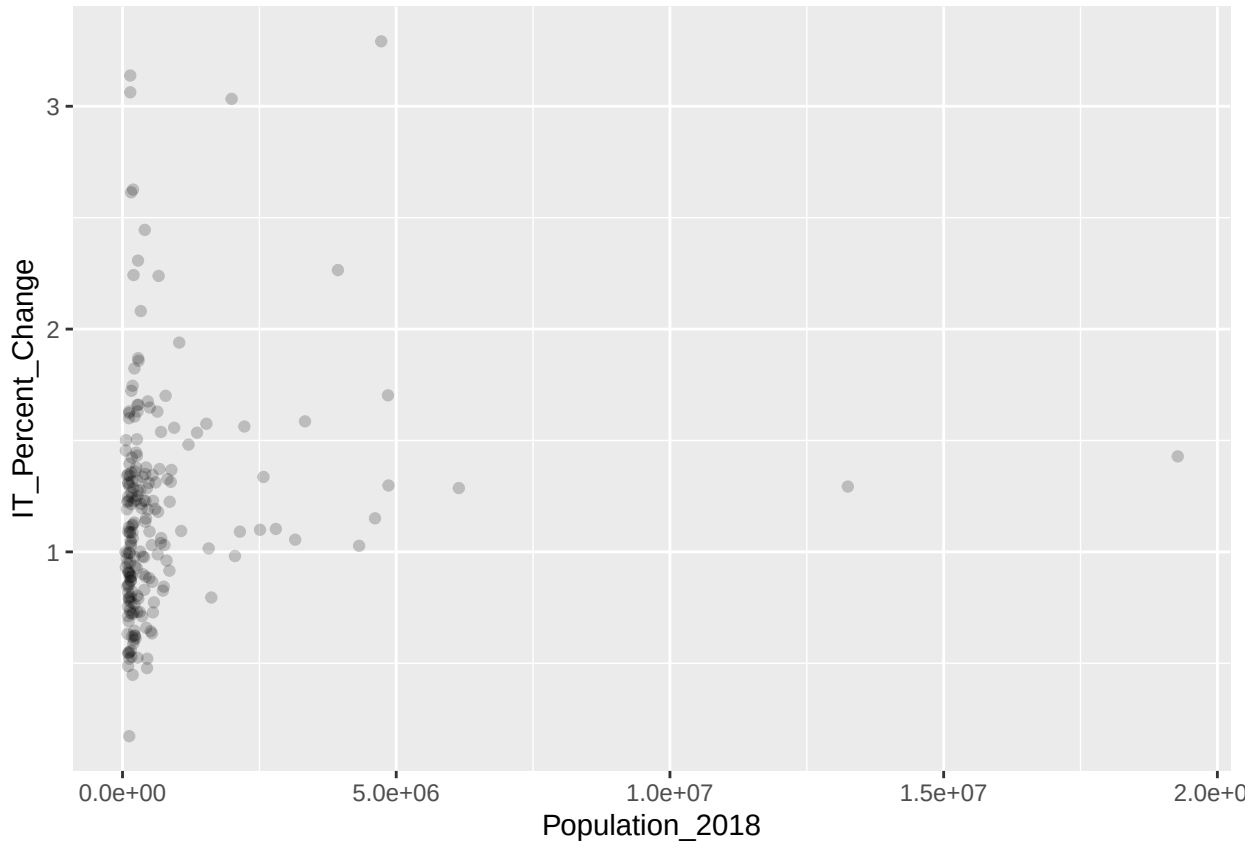
```
## 3          703855      2.788549e+09                1.0011721
## 5          127451      8.458490e+08                1.0903406
## 6          882263      3.474810e+09                1.0128358
## 8          152762      4.716600e+07                0.9913431
## 9          842626      3.335019e+09                1.0251885
##   GDP_Percent_Change  IT_GDP_2010  IT_GDP_2018 IT_Percent_Change IT_GDP_Change
## 1           1.192071 727020547000 1.031010e+12         1.4181308  303989704000
## 3           1.092568    752966000 1.158392e+09         1.5384387     405426000
## 5           1.236778     50296000 7.020900e+07         1.3959162      19913000
## 6           1.071070   1739709000 2.286471e+09         1.3142836     546762000
## 8           1.008403     92844000 6.744400e+07         0.7264228     -25400000
## 9           1.089707   1834283000 2.541458e+09         1.3855321     707175000
##   IT_Importance GDP_Per_Capita_2010 GDP_Per_Capita_2018
## 1      6.246750            44760.66            50521.51
## 3      3.519575            42849.16            46760.82
## 5      1.589094            30561.20            34665.70
## 6      4.366192            56128.82            59356.02
## 8      1.191581            36424.53            37051.37
## 9      6.273392            45231.35            48077.92
##   GDP_Per_Capita_Percentage_Change
## 1                         1.128703
## 3                         1.091289
## 5                         1.134304
## 6                         1.057496
## 8                         1.017209
## 9                         1.062934
```

```
#Spliting My test and rest data
rest_rows <- as.vector(createDataPartition(mainData$IT_Percent_Change, p = 0.75, list = FALSE))
test <- mainData[-rest_rows, ]
rest <- mainData[rest_rows, ]
```

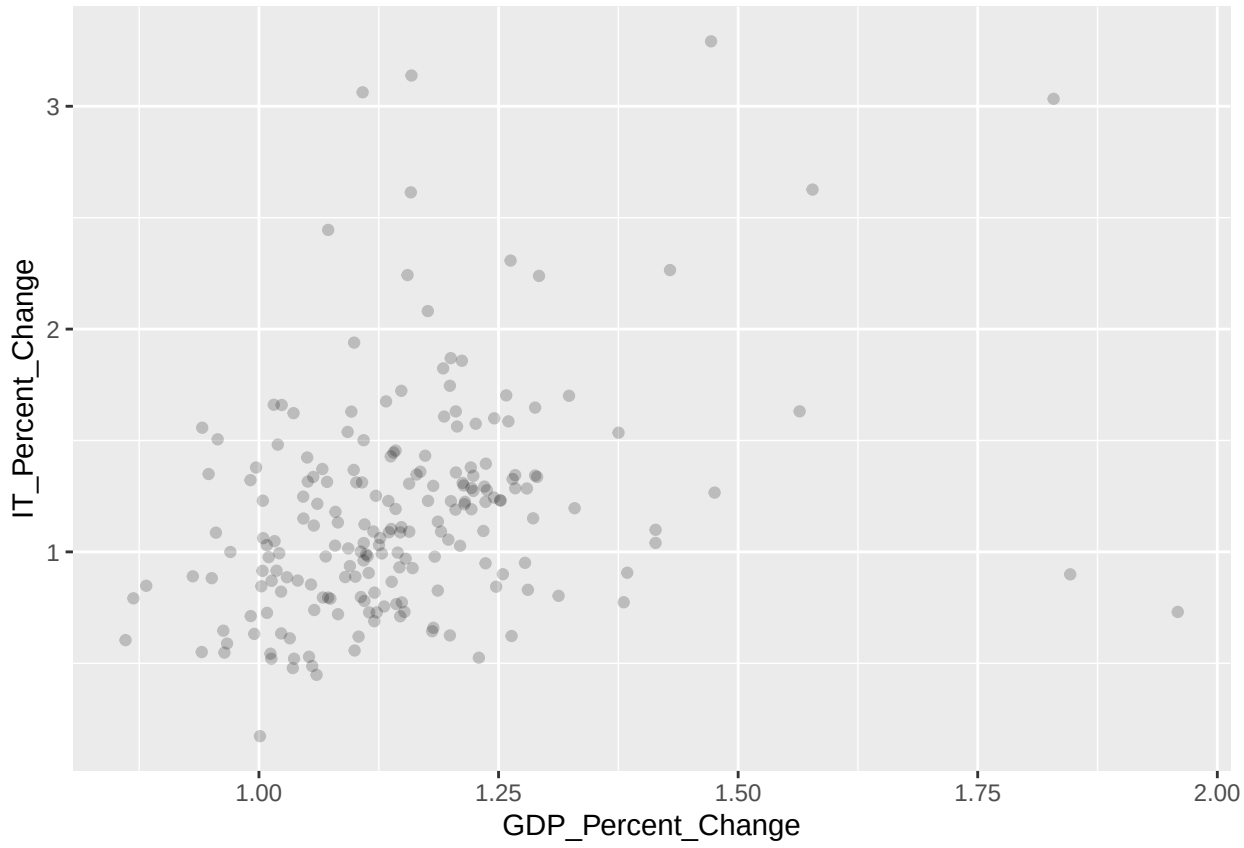
Exploritory Data Analysis

After splitting my data into a test and a rest set, I looked for potential variables to use in my model to predict the change of IT sectors. After making basic plots of four variables, I decided to either choose GDP percentage change or GDP per capita percentage change, but that I shouldn't continue with both. I felt that these were distinct enough to be considered independent variables, but still good variabeles to use to predict the IT percentage change. After throwing it into two models and graphing residuals, I decided that using GDP per capita percentage change was the better varible to predict because I feel that it is more independent and informative by including a population aspect.

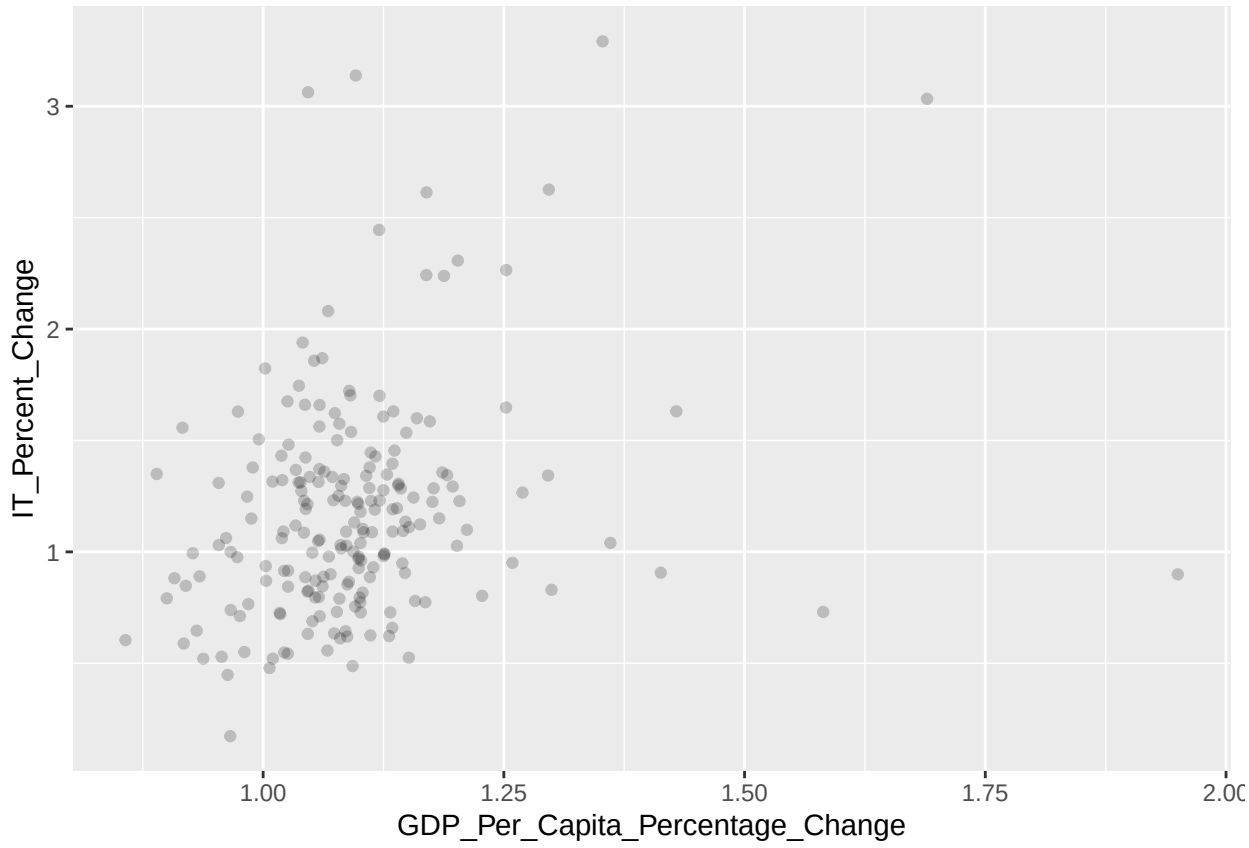
```
#Exploritory Data Analysis

#Total Population (2018)
ggplot(data = rest) +
  geom_point(mapping = aes(x = Population_2018, y = IT_Percent_Change), alpha = 0.2)
```



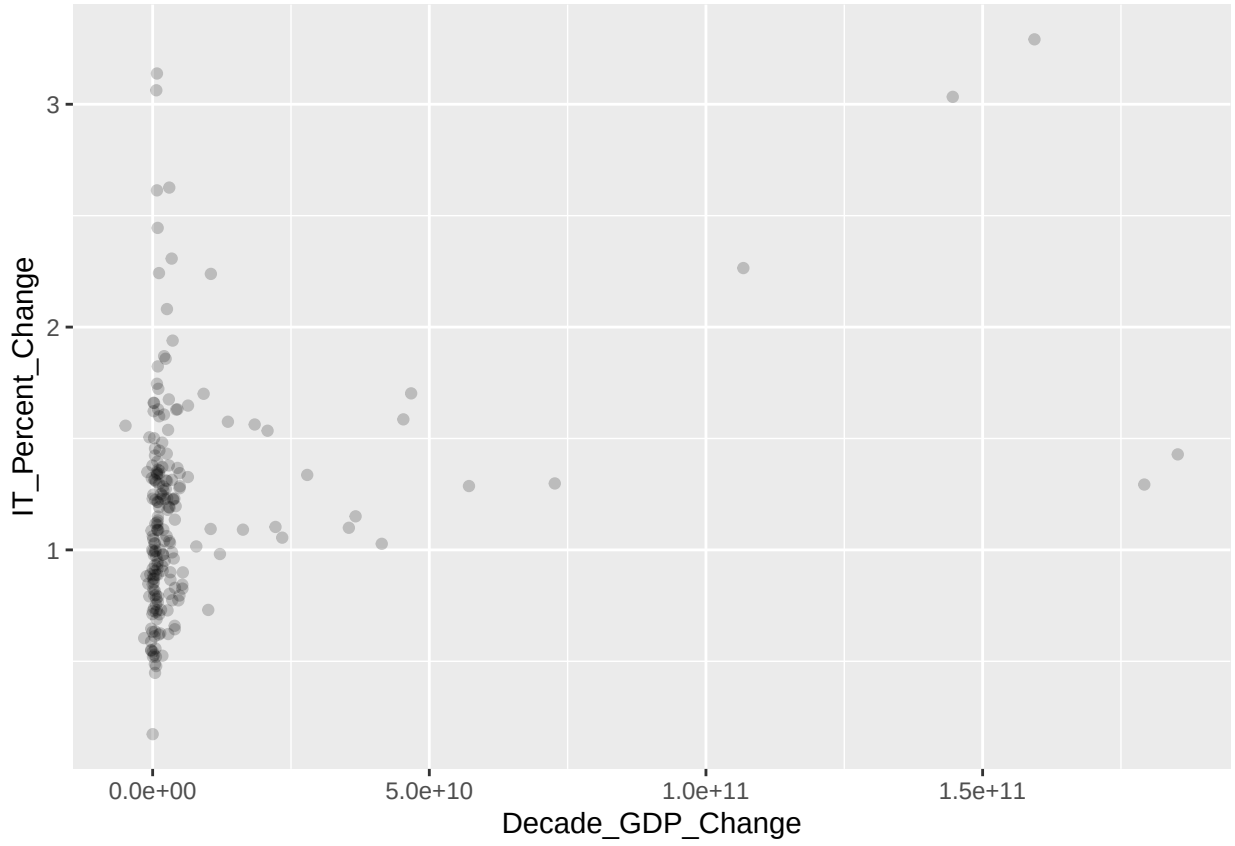

```
#GDP percentage change (2010-2018)
ggplot(data = rest) +
  geom_point(mapping = aes(x = GDP_Percent_Change, y = IT_Percent_Change), alpha = 0.2)
```



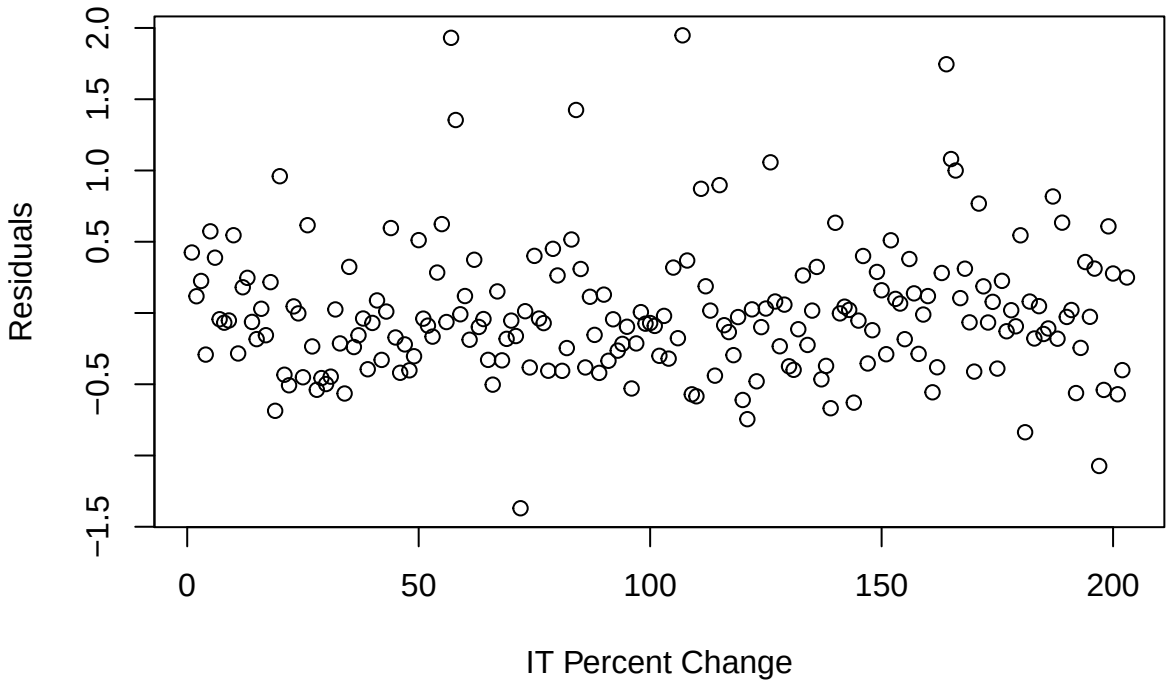

```
#Population percentage change (2010-2018)
ggplot(data = rest) +
  geom_point(mapping = aes(x = GDP_Per_Capita_Percentage_Change, y = IT_Percent_Change), alpha = 0.2)
```




```
#Decade GDP Change (2010-2018)
ggplot(data = rest) +
  geom_point(mapping = aes(x = Decade_GDP_Change, y = IT_Percent_Change), alpha = 0.2)
```

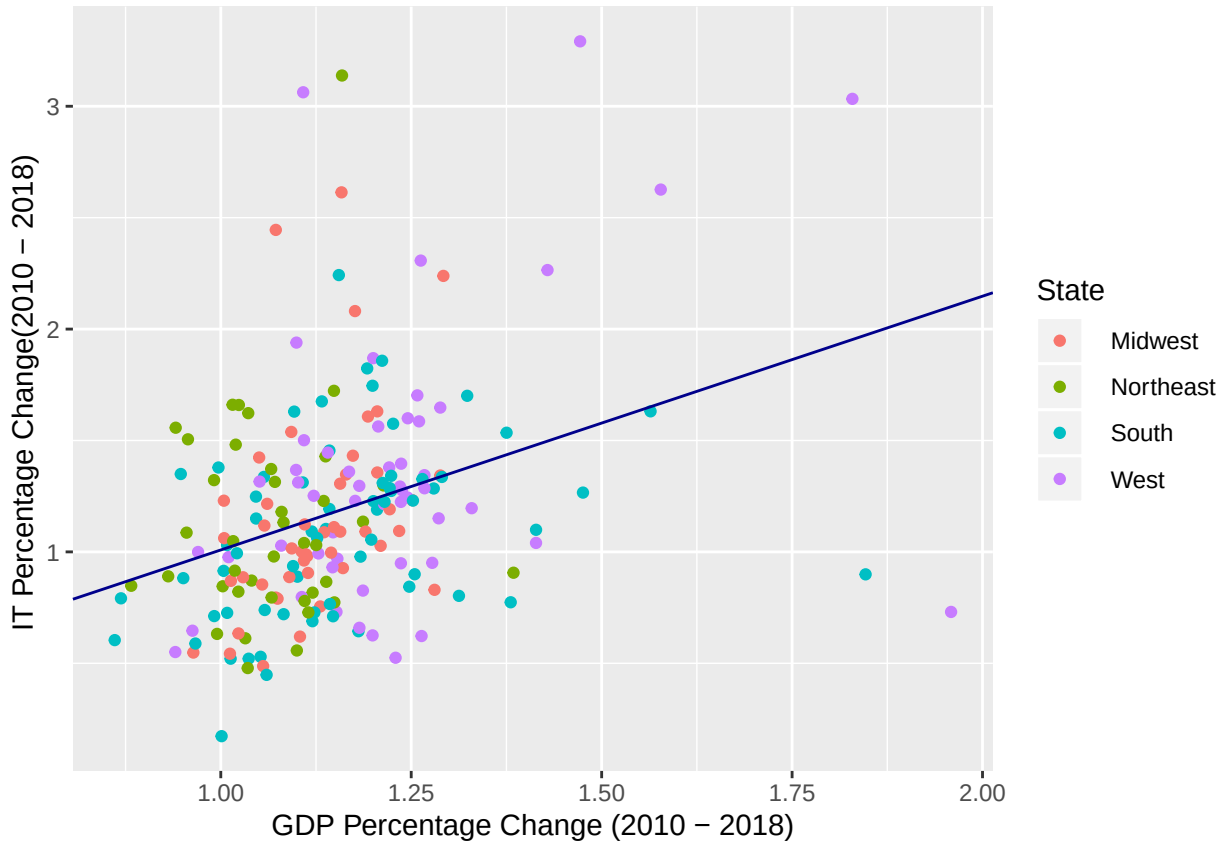



```
EDA_model_1 <- lm(IT_Percent_Change ~ GDP_Percent_Change, data = rest)

EDA_model_1_res <- resid(EDA_model_1)

plot(EDA_model_1_res, ylab="Residuals", xlab="IT Percent Change")
```

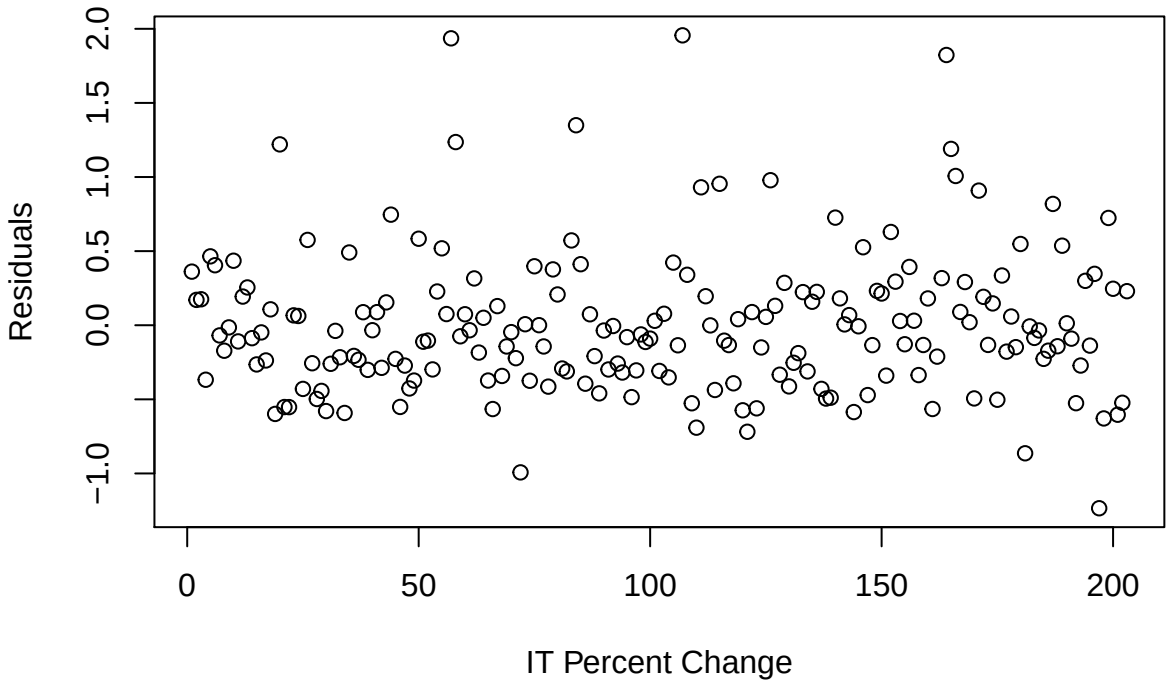



```
ggplot(data=rest, mapping=aes(x=rest$GDP_Percent_Change, y=rest$IT_Percent_Change, color = rest$Region))
  geom_point() +
  geom_abline(intercept = EDA_model_1$coefficients[1], slope = EDA_model_1$coefficients[2], color = "da
  print(labs(x="GDP Percentage Change (2010 - 2018)", y = "IT Percentage Change(2010 - 2018) ", color =
```

```
## $x
## [1] "GDP Percentage Change (2010 - 2018)"
##
## $y
## [1] "IT Percentage Change(2010 - 2018) "
##
## $colour
## [1] "State"
##
## attr(,"class")
## [1] "labels"
```

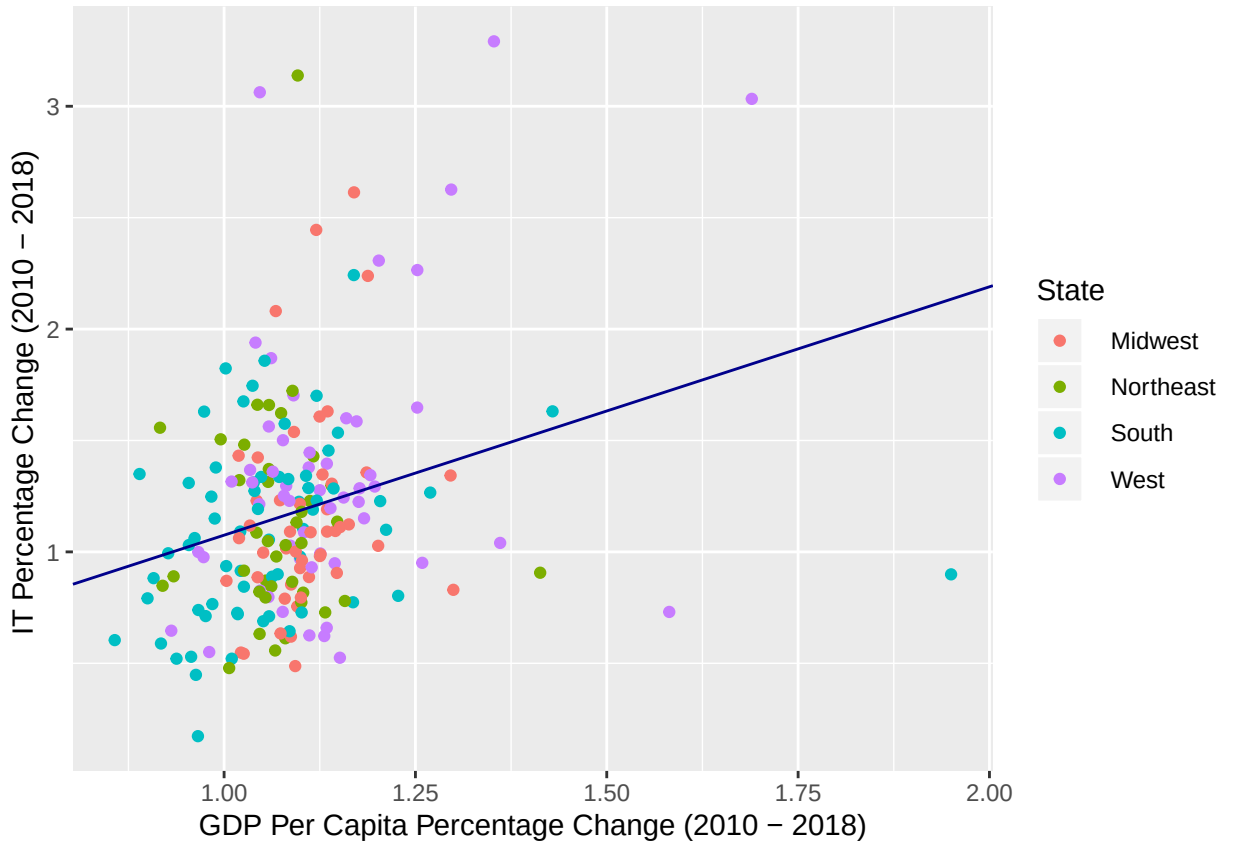



```
EDA_model_2 <- lm(IT_Percent_Change ~ GDP_Per_Capita_Percentage_Change, data = rest)

EDA_model_2_res <- resid(EDA_model_2)

plot(EDA_model_2_res, ylab="Residuals", xlab="IT Percent Change")
```




```
ggplot(data=rest, mapping=aes(x=rest$GDP_Per_Capita_Percentage_Change, y=rest$IT_Percent_Change, color =
  geom_point() +
  geom_abline(intercept = EDA_model_2$coefficients[1], slope = EDA_model_2$coefficients[2], color = "da
  print(labs(x="GDP Per Capita Percentage Change (2010 - 2018)", y = "IT Percentage Change (2010 - 2018
```

```
## $x
## [1] "GDP Per Capita Percentage Change (2010 - 2018)"
##
## $y
## [1] "IT Percentage Change (2010 - 2018) "
##
## $colour
## [1] "State"
##
## attr(,"class")
## [1] "labels"
```



Cross Validation + Model Analysis

To cross validate, I split the rest set into 5 folds and tested that against my independent variable. I also measured the goodness of fit variables for my model. After graphing the residuals and reading my goodness-of-fit measures, I decided that there isn't much correlation betweeen GDP per capita percent change and overall growth of the IT industry. This model is what I would call a failure and probably wouldn't be the best tool for trying to predict certain data. This model isn't useful and doesn't tell me too much about the research question. It does, however, show that the average GDP per capita growth doesn't have too much to do with just the IT sector - something that might suprise many, especially judging by how much the IT sector is growing.

```
#K-Fold Cross Validation (using 5 fold cross validation)
train.control <- trainControl(method = "cv", number = 5)
model <- train(IT_Percent_Change ~ GDP_Per_Capita_Percentage_Change, data = rest, method = "lm",
               trControl = train.control)

# Calculate goodness-of-fit measures for the model
model
```

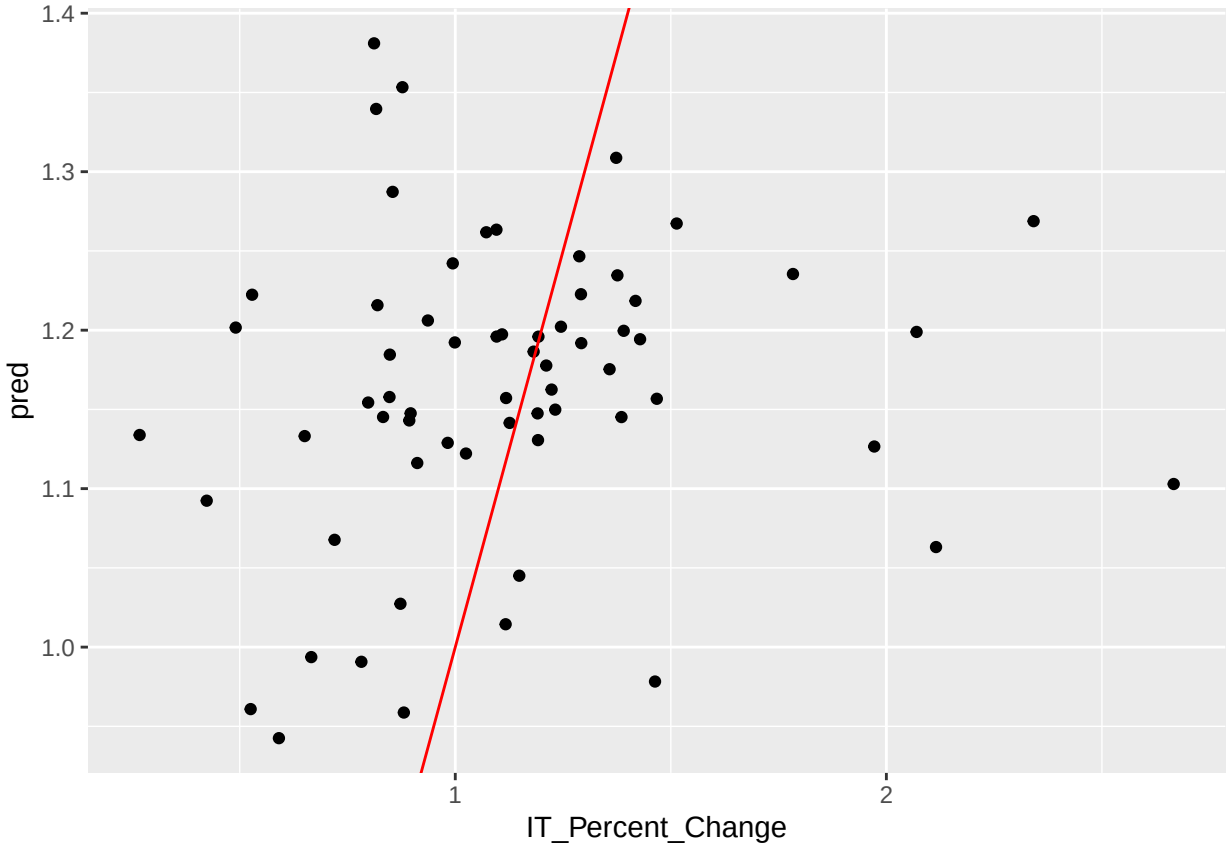
```
## Linear Regression
##
## 203 samples
##   1 predictor
##
## No pre-processing
```

```
## Resampling: Cross-Validated (5 fold)
## Summary of sample sizes: 162, 161, 163, 163, 163
## Resampling results:
##
##   RMSE       Rsquared   MAE
##   0.4750418  0.0701408  0.3501571
##
## Tuning parameter 'intercept' was held constant at a value of TRUE
```

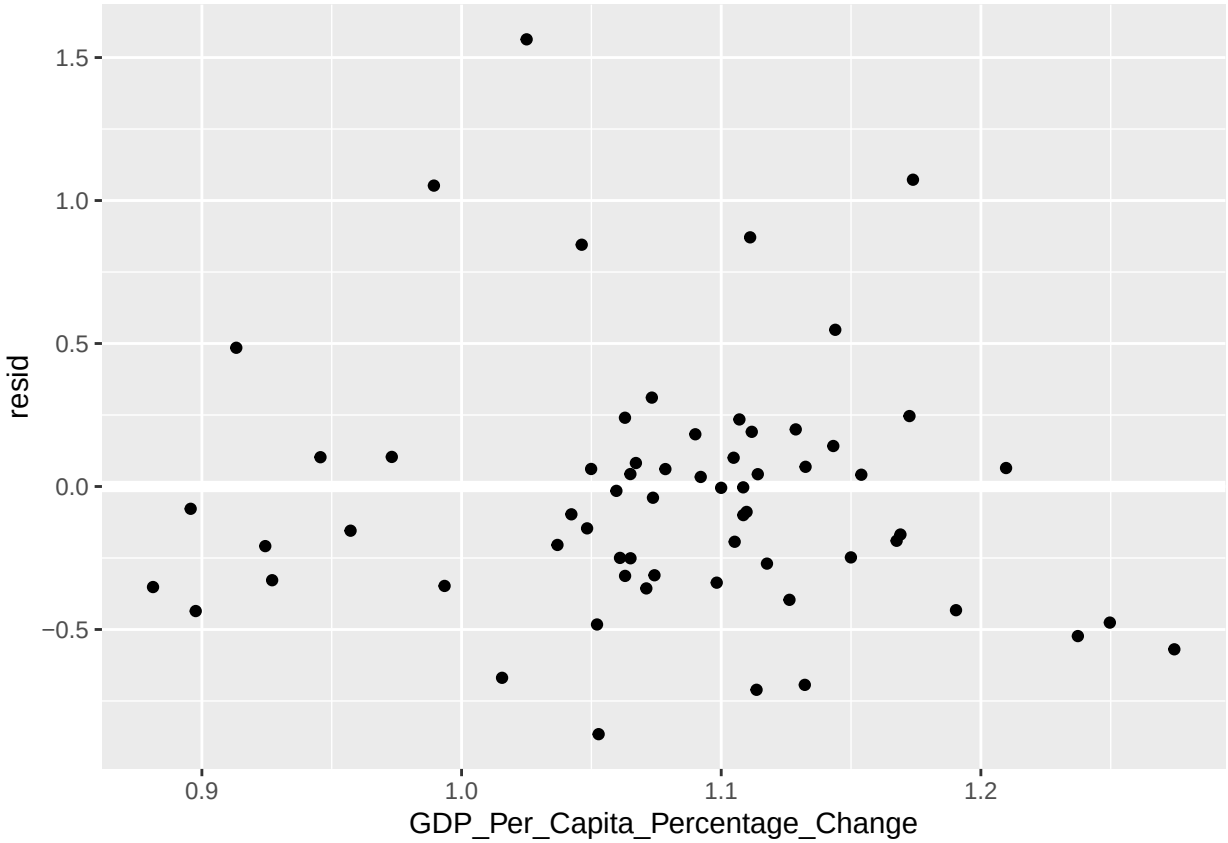
```
summary(model)
```

```
##
## Call:
## lm(formula = .outcome ~ ., data = dat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.23470 -0.30658 -0.06863  0.21081  1.95585
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                     -0.03975    0.30846  -0.129    0.898
## GDP_Per_Capita_Percentage_Change  1.11475    0.28012   3.979 9.64e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4808 on 201 degrees of freedom
## Multiple R-squared:  0.07303,    Adjusted R-squared:  0.06842
## F-statistic: 15.84 on 1 and 201 DF,  p-value: 9.641e-05
```

```
#For making my model predictions
model_predictions <- add_predictions(test, model)

ggplot(data = model_predictions, mapping = aes(x = IT_Percent_Change, y = pred)) +
  geom_point() +
  geom_abline(intercept = 0, slope = 1, color = "red")
```




```
#Creating and graphing the residuals
resids <- add_residuals(test, model)


ggplot(data = resids, mapping = aes(x = GDP_Per_Capita_Percentage_Change, y = resid)) +
  geom_ref_line(h = 0) +
  geom_point()
```




```
#Making predictions based off of changes in the Independent Variable
predict(model, data.frame(GDP_Per_Capita_Percentage_Change = c(1.00, 1.25, 1.50)))
```

```
##        1        2        3
## 1.075002 1.353689 1.632377
```

Conclusion + Potential Social Implications

Through working on this deliverable, I felt like I refined my previous skills in making data tidy and us

In terms of social implications, data regarding population and GDP is definitely useful. Data analysis