Objective

Data Exploration

Data Preparation

Build Models

Conclusion

DATA 622 Homework 4

Ramnivas Singh

December 24, 2022

```
library("viridis")
```

```
## Loading required package: viridisLite
```

```
library(tidyverse)
```

```
## ── Attaching packages
## ─────────────────────────────────────
## tidyverse 1.3.2 ──
```

```
## ✓ ggplot2 3.3.6      ✓ purrr   0.3.5
## ✓ tibble  3.1.8      ✓ dplyr   1.0.10
## ✓ tidyr   1.2.1      ✓ stringr 1.4.1
## ✓ readr   2.1.3      ✓ forcats 0.5.2
## ── Conflicts ──────────────────────────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
```

```
library(DataExplorer)
library(e1071)
library(caret)
```

```
## Loading required package: lattice
##
## Attaching package: 'caret'
##
## The following object is masked from 'package:purrr':
##
##     lift
```

Objective

You get to decide which dataset you want to work on. The data set must be different from the ones used in previous homeworks You can work on a problem from your job, or something you are interested in. You may also obtain a dataset from sites such as Kaggle, Data.Gov, Census Bureau, USGS or other open data portals. Select one of the methodologies studied in weeks 1-10, and one methodology from weeks 11-15 to apply in the new dataset selected. To complete this task:

- describe the problem you are trying to solve.
- describe your dataset and what you did to prepare the data for analysis.
- methodologies you used for analyzing the data
- what's the purpose of the analysis performed
- make your conclusions from your analysis. Please be sure to address the business impact (it could be of any domain) of your solution.

Data Exploration

This Analysis will focus on determining how many funds will be needed for future academic years. This could potentially help allocate funds more efficiently.

I have decided to use data from "data.ny.gov" website, which has a collection of different data sets to choose from and work with. Within this website I found a data set with the Tuition Assistance Program recipients and dollar amount by college and sector group from the year 2000.

We will first load the data, which I have saved in a local folder, and explore it.

```
data <- readr::read_csv("TAP_Recipients_Beginning_2000.csv")
```

```
## Rows: 7965 Columns: 9
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: ","
## chr (3): TAP College Name, Sector Type, TAP Sector Group
## dbl (6): Academic Year, TAP College Code, Federal School Code, TAP Recipient...
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

The data consists of 7,965 observations and 9 variables.

```
dim(data)
```

```
## [1] 7965    9
```

The data contain the variables "Academic Year, TAP College Code, Federal School Code, TAP College Name, Sector Type, TAP Sector Group, TAP Recipient Headcount, TAP Recipient FTEs and TAP Recipient Dollars".

```
head(data)
```

```
## # A tibble: 6 × 9
##   Academic Yea…¹ TAP C…² Feder…³ TAP C…⁴ Secto…⁵ TAP S…⁶ TAP R…⁷ TAP R…⁸ TAP R…⁹
##            <dbl>   <dbl>   <dbl> <chr>   <chr>   <chr>     <dbl>   <dbl>   <dbl>
## 1           2020      10    2666 ADELPH… PRIVATE 5-INDE…    1343 1248.   3827369
## 2           2020    2000    2860 ADIRON… PUBLIC  4-SUNY…     603  460.   1247648
## 3           2020     995    2885 ALBANY… PRIVATE 5-INDE…     163  162.    449133
## 4           2020      20    2668 ALFRED… PRIVATE 5-INDE…     442  395.   1217828
## 5           2020    7952   10813 AMERIC… PRIVATE 6-BUS.…      36   27.3    80168
## 6           2020    1075    7465 AMERIC… PRIVATE 5-INDE…       3    3.06   12364
## # … with abbreviated variable names ¹`Academic Year`, ²`TAP College Code`,
## #   ³`Federal School Code`, ⁴`TAP College Name`, ⁵`Sector Type`,
## #   ⁶`TAP Sector Group`, ⁷`TAP Recipient Headcount`, ⁸`TAP Recipient FTEs`,
## #   ⁹`TAP Recipient Dollars`
```

Since “Sector Type” and “TAP Sector Group” seem like categorical variables we may want to use in our model, let’s take a look at how many different categories we would find within each of these variables.

```
data %>%
  distinct(`Sector Type`)
```

```
## # A tibble: 2 × 1
##   `Sector Type`
##   <chr>
## 1 PRIVATE
## 2 PUBLIC
```

```
data %>%
  distinct(`TAP Sector Group`)
```

```
## # A tibble: 9 × 1
##   `TAP Sector Group`
##   <chr>
## 1 5-INDEPENDENT
## 2 4-SUNY CC
## 3 6-BUS. DEGREE
## 4 8-OTHER
## 5 9-CHAPTER XXII
## 6 7-BUS. NON-DEG
## 7 3-SUNY SO
## 8 1-CUNY SR
## 9 2-CUNY CC
```

From the results above, we see that there are 2 categories for “Sector Type” and 9 for “TAP Sector Group”.

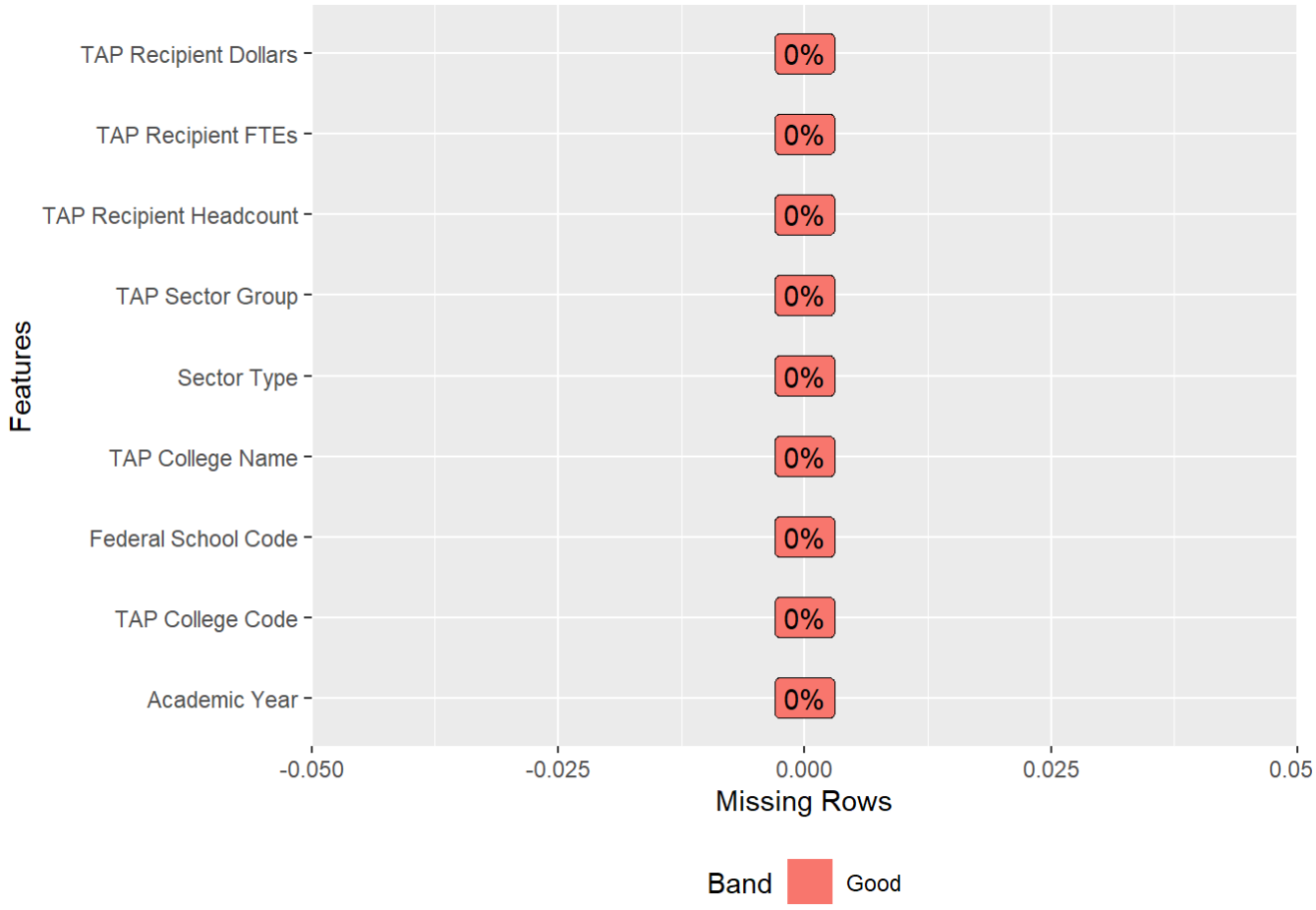
Let’s also take a look at the structure of the data to determine if any of the variables need transformations:

```
str(data)
```

```
## spec_tbl_df [7,965 × 9] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ Academic Year          : num [1:7965] 2020 2020 2020 2020 2020 2020 2020 2020 2020 2
020 ...
##  $ TAP College Code       : num [1:7965] 10 2000 995 20 7952 ...
##  $ Federal School Code    : num [1:7965] 2666 2860 2885 2668 10813 ...
##  $ TAP College Name       : chr [1:7965] "ADELPHI UNIVERSITY 4 YR UNDERGRAD" "ADIRONDAC
K COMMUNITY COLLEGE" "ALBANY COLLEGE OF PHARMACY 4YR UG" "ALFRED UNIVERSITY 4YR UNDERGRAD"
...
##  $ Sector Type            : chr [1:7965] "PRIVATE" "PUBLIC" "PRIVATE" "PRIVATE" ...
##  $ TAP Sector Group       : chr [1:7965] "5-INDEPENDENT" "4-SUNY CC" "5-INDEPENDENT" "5
-INDEPENDENT" ...
##  $ TAP Recipient Headcount: num [1:7965] 1343 603 163 442 36 ...
##  $ TAP Recipient FTEs     : num [1:7965] 1247.9 459.9 162.4 395.2 27.3 ...
##  $ TAP Recipient Dollars  : num [1:7965] 3827369 1247648 449133 1217828 80168 ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   `Academic Year` = col_double(),
##   ..   `TAP College Code` = col_double(),
##   ..   `Federal School Code` = col_double(),
##   ..   `TAP College Name` = col_character(),
##   ..   `Sector Type` = col_character(),
##   ..   `TAP Sector Group` = col_character(),
##   ..   `TAP Recipient Headcount` = col_double(),
##   ..   `TAP Recipient FTEs` = col_double(),
##   ..   `TAP Recipient Dollars` = col_double()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

Additionally, the data do not seem to have any missing values as observed in the plot below.

```
plot_missing(data)
```



We will also explore any trends that might be found in the “TAP Recipient Dollars” throughout the years.

```
attrYear <- data %>% group_by(`Academic Year`) %>% summarise("Average TAP Dollar" = mean(`
TAP Recipient Dollars`), Count = n())
```

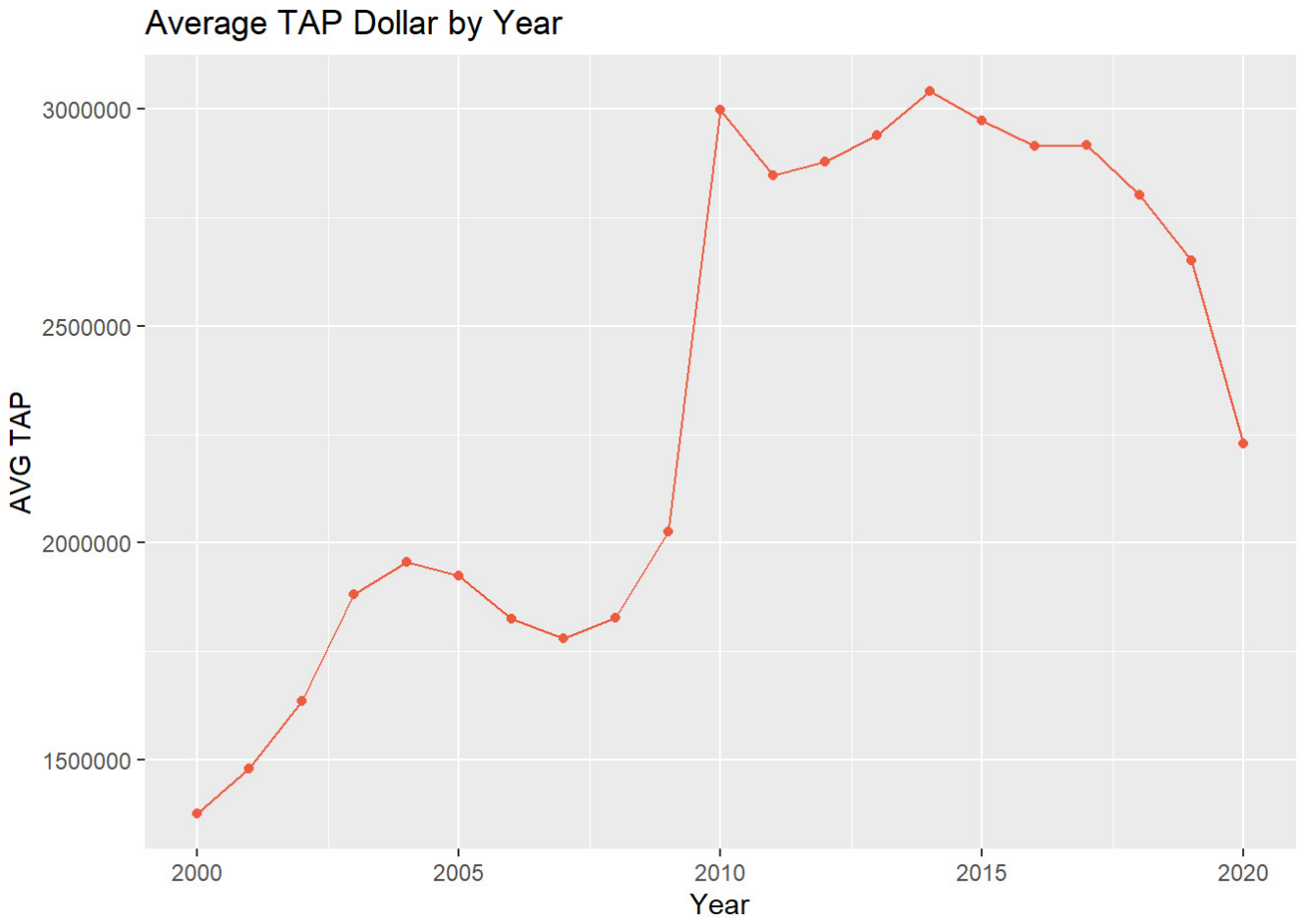
Below are the first few rows of the subset of the data I created summarizing the average amount of TAP Dollar per year.

```
head(attrYear)
```

```
## # A tibble: 6 × 3
##   `Academic Year` `Average TAP Dollar` Count
##             <dbl>                <dbl> <int>
## 1            2000             1376963.   461
## 2            2001             1480101.   455
## 3            2002             1635832.   444
## 4            2003             1882418.   449
## 5            2004             1955027.   447
## 6            2005             1924555.   448
```

It seems that TAP Dollar had a steep increase between the years 2009 and 2010 where it reached its peak. From then, it’s had its ups and downs but we can see a clear decline from year 2017 and on.

```
ggplot(attrYear, aes(x=`Academic Year`, y=`Average TAP Dollar`)) + geom_line(color = "toma
to2") + labs(x = "Year", y = "AVG TAP", title = "Average TAP Dollar by Year") + geom_point
(color = "tomato2")
```



Data Preparation

We will create a separate data set in order to maintain the original data and make all the necessary transformations there. First we'll transform the previous variables of interest that were of type character into factor.

```
data_prepared <- data

#Data type change for columns of interest
data_prepared$`Sector Type` <- as.factor(data_prepared$`Sector Type`)
data_prepared$`TAP Sector Group` <- as.factor(data_prepared$`TAP Sector Group`)
```

We have changed the structure of our variables as it is shown below.

```
str(data_prepared)
```

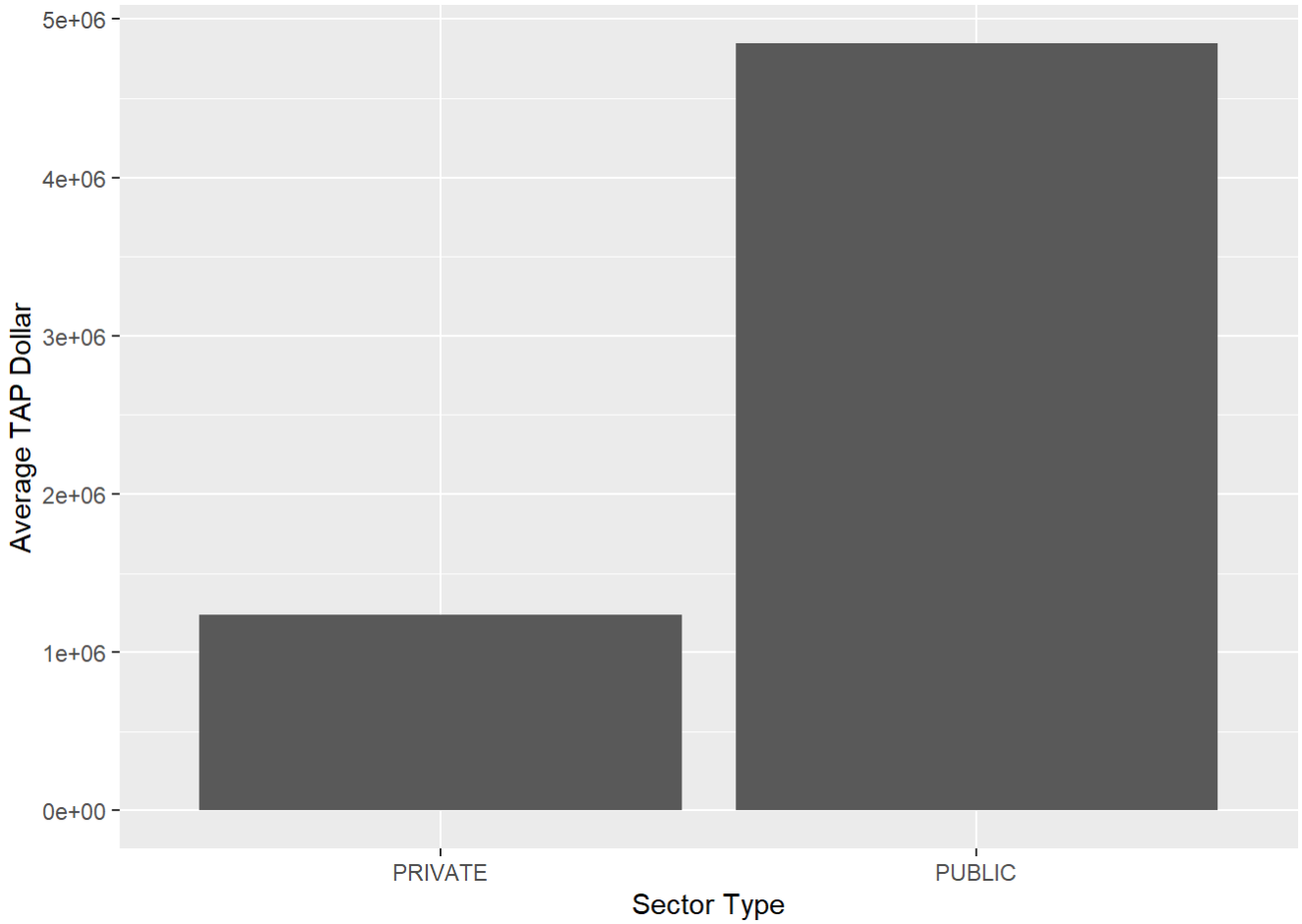
```
## spec_tbl_df [7,965 × 9] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ Academic Year          : num [1:7965] 2020 2020 2020 2020 2020 2020 2020 2020 2020 2
020 ...
##  $ TAP College Code       : num [1:7965] 10 2000 995 20 7952 ...
##  $ Federal School Code    : num [1:7965] 2666 2860 2885 2668 10813 ...
##  $ TAP College Name       : chr [1:7965] "ADELPHI UNIVERSITY 4 YR UNDERGRAD" "ADIRONDAC
K COMMUNITY COLLEGE" "ALBANY COLLEGE OF PHARMACY 4YR UG" "ALFRED UNIVERSITY 4YR UNDERGRAD"
...
##  $ Sector Type            : Factor w/ 2 levels "PRIVATE","PUBLIC": 1 2 1 1 1 1 1 1 1 1
...
##  $ TAP Sector Group       : Factor w/ 9 levels "1-CUNY SR","2-CUNY CC",..: 5 4 5 5 6 5
8 8 6 5 ...
##  $ TAP Recipient Headcount: num [1:7965] 1343 603 163 442 36 ...
##  $ TAP Recipient FTEs     : num [1:7965] 1247.9 459.9 162.4 395.2 27.3 ...
##  $ TAP Recipient Dollars  : num [1:7965] 3827369 1247648 449133 1217828 80168 ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   `Academic Year` = col_double(),
##   ..   `TAP College Code` = col_double(),
##   ..   `Federal School Code` = col_double(),
##   ..   `TAP College Name` = col_character(),
##   ..   `Sector Type` = col_character(),
##   ..   `TAP Sector Group` = col_character(),
##   ..   `TAP Recipient Headcount` = col_double(),
##   ..   `TAP Recipient FTEs` = col_double(),
##   ..   `TAP Recipient Dollars` = col_double()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

Additionally, the plots below show that more of the TAP allocation goes to the “Public” sector type and the “CUNY” TAP sector group. Something to consider when reviewing our results.

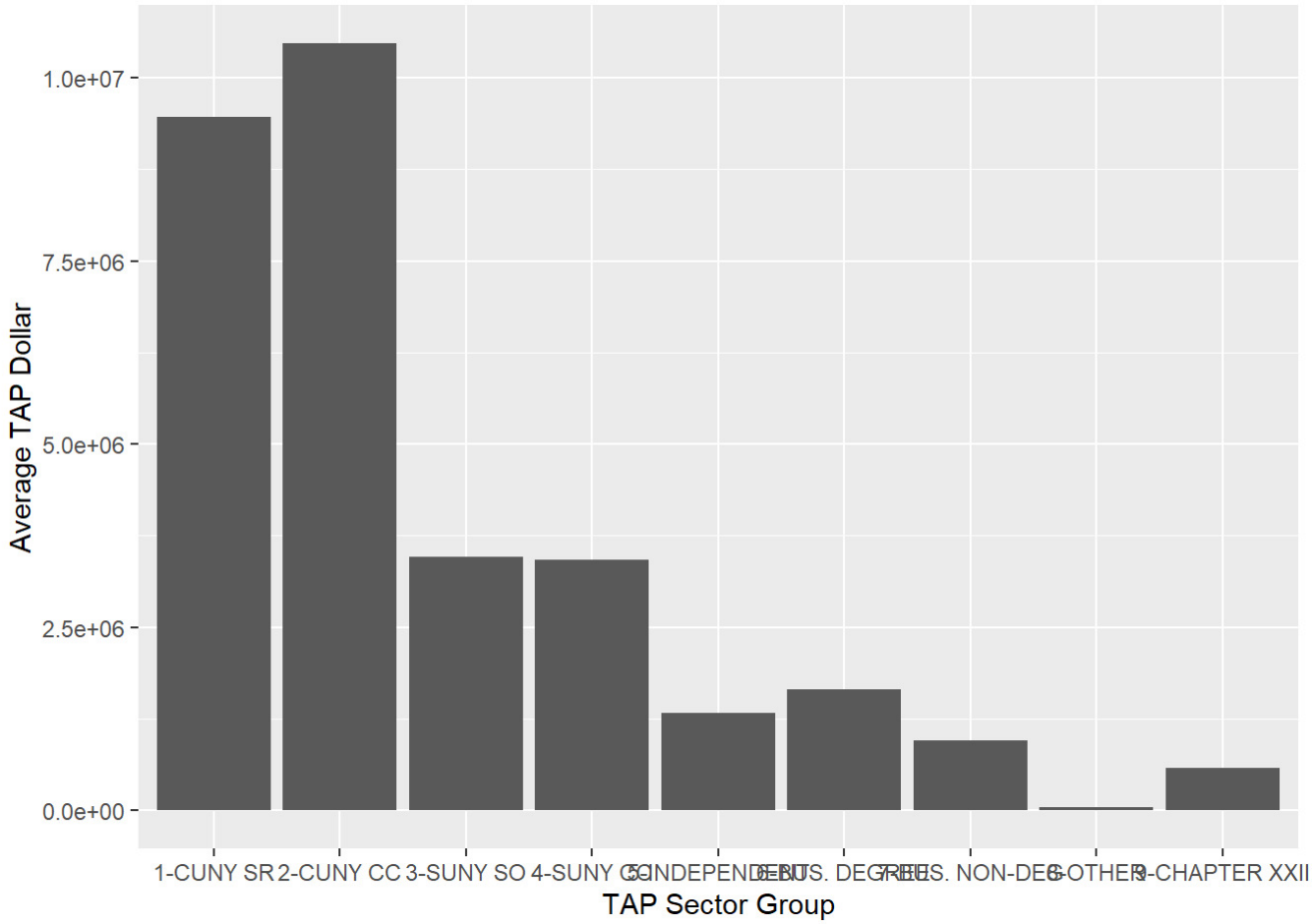
```
attrSector <- data %>% group_by(`Sector Type`) %>% summarise("Average TAP Dollar" = mean(`
TAP Recipient Dollars`), Count = n())

attrGroup <- data %>% group_by(`TAP Sector Group`) %>% summarise("Average TAP Dollar" = me
an(`TAP Recipient Dollars`), Count = n())
```

```
ggplot(attrSector, aes(x=`Sector Type`, y=`Average TAP Dollar`)) + geom_bar(stat="identit
y")
```




```
ggplot(attrGroup, aes(x=`TAP Sector Group`, y=`Average TAP Dollar`)) + geom_bar(stat="iden
tity")
```



Build Models

Here we will train a SVM model and a Neural Networks model and compare their performance to determine which of the two is the most appropriate for our data.

For this assignment I have picked to use a SVM model to train the data as it has been one of the better performing models I have used in previous homeworks. On the other hand, the Neural Network model is one of the more advance techniques in machine learning and we could potentially obtain some interesting results and performance.

First we will partition the data into 80% train and 20% test.

```
# Partition data - train (80%) & test (20%)
set.seed(1234)
ind <- sample(2, nrow(data_prepared), replace = T, prob = c(0.8, 0.2))
train <- data_prepared[ind==1,]
test <- data_prepared[ind==2,]
```

We will also “center” and “scale” our data as we will be using variables with different scales in our models.

```
trans_train <- preProcess(train, method = c("center", "scale"))
trans_test <- preProcess(test, method = c("center", "scale"))

train_prep <- predict(trans_train, train)
test_prep <- predict(trans_test, test)
```

Next we will tune our SVM model and perform a grid search.

```
# perform a grid search
svm_tune <- tune(svm, `TAP Recipient Dollars` ~ + `Academic Year` + `TAP College Code` + `
Sector Type` + `TAP Sector Group`, data = train_prep)
```

As evident from the results below, the SVM model gave us an R-square of 0.54, RMSE of 0.68 and MAE of 0.35.

```
#The best model
best_mod <- svm_tune$best.model
best_mod_pred <- predict(best_mod, test_prep)

postResample(pred=best_mod_pred, obs = test_prep$`TAP Recipient Dollars`)
```

```
##      RMSE  Rsquared       MAE
## 0.6836889 0.5416867 0.3534940
```

Secondly, we will train a Neural Networks model to compare the results. The first step will be to “one-hot” encode the categorical variables in our data, as the Neural Network model only takes numerical variables.

```
#Select only those variables we'll use in the model
new_data1 <- select(train_prep, `Academic Year`, `TAP College Code`, `Sector Type`, `TAP S
ector Group`, `TAP Recipient Dollars`)

#Use the 'dummyVars' from the caret package to encode
dmy <- dummyVars(" ~ .", data = new_data1, fullRank = T)
train_transformed <- data.frame(predict(dmy, newdata = new_data1))

glimpse(train_transformed)
```

```
## Rows: 6,380
## Columns: 12
## $ X.Academic.Year.                  <dbl> 1.810535, 1.810535, 1.810535, 1.8105…
## $ X.TAP.College.Code.               <dbl> -1.2237154, -0.5129988, -0.8719285, …
## $ X.Sector.Type.PUBLIC              <dbl> 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, …
## $ X.TAP.Sector.Group.2.CUNY.CC      <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, …
## $ X.TAP.Sector.Group.3.SUNY.SO      <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, …
## $ X.TAP.Sector.Group.4.SUNY.CC      <dbl> 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, …
## $ X.TAP.Sector.Group.5.INDEPENDENT  <dbl> 1, 0, 1, 1, 1, 0, 0, 0, 1, 0, 0, 0, …
## $ X.TAP.Sector.Group.6.BUS..DEGREE  <dbl> 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, …
## $ X.TAP.Sector.Group.7.BUS..NON.DEG <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, …
## $ X.TAP.Sector.Group.8.OTHER        <dbl> 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, …
## $ X.TAP.Sector.Group.9.CHAPTER.XXII <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, …
## $ X.TAP.Recipient.Dollars.          <dbl> 0.41017085, -0.25121303, -0.45593477…
```

Now we’ll do the same for the test set.

```
new_data2 <- select(test_prep, `Academic Year`, `TAP College Code`, `Sector Type`, `TAP Se
ctor Group`, `TAP Recipient Dollars`)

dmy <- dummyVars(" ~ .", data = new_data2, fullRank = T)
test_transformed <- data.frame(predict(dmy, newdata = new_data2))
```

And now we build the Neural Network model:

```
set.seed(145)
nnetAvg <- avNNet(train_transformed[1:11], train_transformed$X.TAP.Recipient.Dollars.,
                  size = 5,
                  decay = 0.01,
                  repeats = 5,
                  linout = TRUE,
                  trace = FALSE,
                  maxit = 500)
```

```
## Warning: executing %dopar% sequentially: no parallel backend registered
```

```
nnetPred <- predict(nnetAvg, newdata = test_transformed[1:11])

postResample(pred = nnetPred, obs = test_transformed$X.TAP.Recipient.Dollars.)
```

```
##      RMSE  Rsquared       MAE
## 0.6640800 0.5828910 0.3650667
```

The Neural Network model gave us an R-square of 0.58, RMSE of 0.66 and MAE of 0.36.

Conclusion

As we are able to see above, both models gave us very good R-square values, and they're both very close in their performance based on RMSE and MAE. I would recommend that either model could be used to predict values for "TAP Recipient Dollars" for future academic years. The R-square for both models is also low enough that I would not be concerned about over-fit, and high enough to give us confidence that the model can predict values accurately.

After reviewing and training two different models on the "Tuition Assistance Program recipients and dollar amount by college and sector group" data, I would conclude that any of these two models would be an excellent choice to predict funds for future academic years.

These results could potentially help allocate funds more effectively and efficiently as it could give administrators a better sense of how much will be needed for future academic years. Our analysis could also help derive predictions for how much funds will be needed in each "Sector Type", "TAP Sector Group" and College, all of which would aid in reducing any shortages or surpluses of funds.