DegreesThatPayYouBack

Which college majors will pay the bills?

Wondering if that Philosophy major will really help you pay the bills? Think you're set with an Engineering degree? Choosing a college major is a complex decision evaluating personal interest, difficulty, and career prospects. Your first paycheck right out of college might say a lot about your salary potential by mid-career. Whether you're in school or navigating the postgrad world, join me as we explore the short and long term financial implications of this major decision.

In this notebook, we'll be using data collected from a year-long survey of 1.2 million people with only a bachelor's degree by PayScale Inc., made available here by the Wall Street Journal for their article Ivy League's Big Edge: Starting Pay. After doing some data clean up, we'll compare the recommendations from three different methods for determining the optimal number of clusters, apply a k-means clustering analysis, and visualize the results.

To begin, let's prepare by loading the following packages: tidyverse, dplyr, readr, ggplot2, cluster, and factoextra. We'll then import the data from degrees-that-pay-back.csv (which is stored in a folder called datasets), and take a quick look at what we're working with.

Read in the dataset

```
## Read in the dataset
degrees <- read_csv("./data/degrees-that-pay-back.csv",
col_names= c("College.Major", "Starting.Median.Salary", "Mid.Career.Median.Salary", "Career.Percent.Grou
```

Display the first few rows and a summary of the data frame

```
head(degrees)
```

```
## # A tibble: 6 x 8
##   College.Major Starting.Median~ Mid.Career.Medi~ Career.Percent.~ Percentile.10
##   <chr>         <chr>            <chr>                       <dbl> <chr>
## 1 Accounting    $46,000.00       $77,100.00                   67.6 $42,200.00
## 2 Aerospace En~ $57,700.00       $101,000.00                  75   $64,300.00
## 3 Agriculture   $42,600.00       $71,900.00                   68.8 $36,300.00
## 4 Anthropology  $36,800.00       $61,500.00                   67.1 $33,800.00
## 5 Architecture  $41,600.00       $76,800.00                   84.6 $50,600.00
## 6 Art History   $35,800.00       $64,900.00                   81.3 $28,800.00
## # ... with 3 more variables: Percentile.25 <chr>, Percentile.75 <chr>,
## #   Percentile.90 <chr>
```

```
summary(degrees)
```

```
##  College.Major      Starting.Median.Salary Mid.Career.Median.Salary
##  Length:50          Length:50              Length:50
##  Class :character   Class :character       Class :character
```

```
##  Mode  :character   Mode  :character       Mode  :character
##
##
##
##  Career.Percent.Growth Percentile.10      Percentile.25      Percentile.75
##  Min.   : 23.40        Length:50          Length:50          Length:50
##  1st Qu.: 59.12        Class :character   Class :character   Class :character
##  Median : 67.80        Mode  :character   Mode  :character   Mode  :character
##  Mean   : 69.27
##  3rd Qu.: 82.42
##  Max.   :103.50
##  Percentile.90
##  Length:50
##  Class :character
##  Mode  :character
##
##
##
```

Currency and strings and percents, oh my!

Notice that our salary data is in currency format, which R considers a string. Let's strip those special characters using the gsub function and convert all of our columns except College.Major to numeric.

While we're at it, we can also convert the Career.Percent.Growth column to a decimal value.

```
# Clean up the data
degrees_clean <- degrees %>%
    mutate_at(vars(Starting.Median.Salary:Percentile.90), function(x) as.numeric(gsub("[\\$,]","",x))) %
mutate(Career.Percent.Growth = Career.Percent.Growth / 100)
```

The elbow method

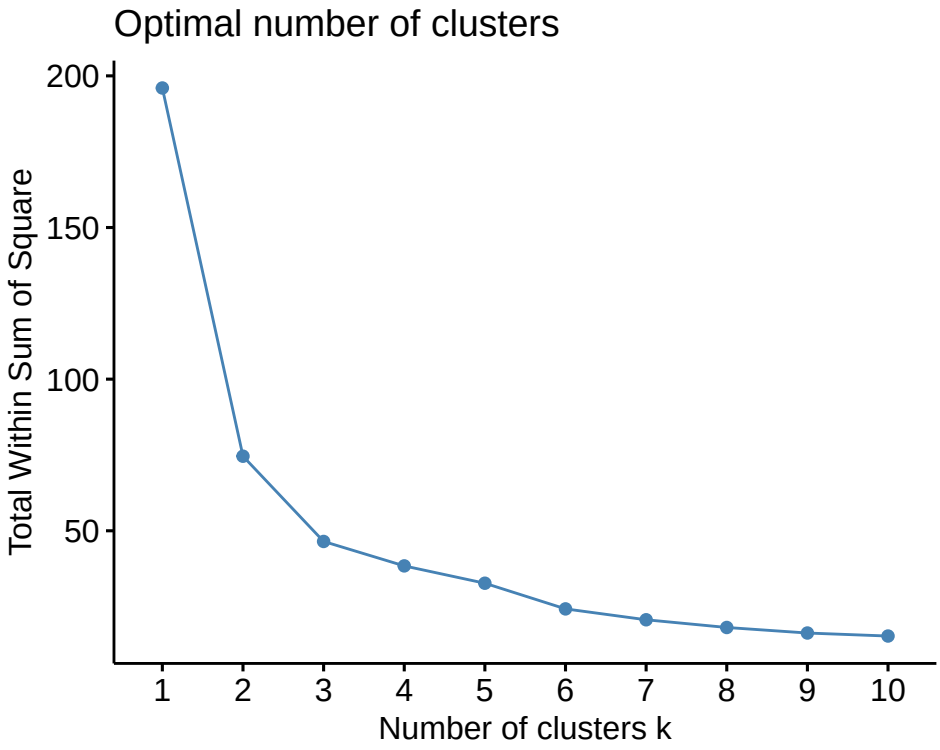
Great! Now that we have a more manageable dataset, let's begin our clustering analysis by determining how many clusters we should be modeling. The best number of clusters for an unlabeled dataset is not always a clear-cut answer, but fortunately there are several techniques to help us optimize. We'll work with three different methods to compare recommendations:

- Elbow Method
- Silhouette Method
- Gap Statistic Method

First up will be the Elbow Method. This method plots the percent variance against the number of clusters. The "elbow" bend of the curve indicates the optimal point at which adding more clusters will no longer explain a significant amount of the variance. To begin, let's select and scale the following features to base our clusters on: Starting.Median.Salary, Mid.Career.Median.Salary, Perc.10, and Perc.90. Then we'll use the fancy fviz_nbclust function from the factoextra library to determine and visualize the optimal number of clusters.

```
# Select and scale the relevant features and store as k_means_data
k_means_data <- degrees_clean %>% select(Starting.Median.Salary, Mid.Career.Median.Salary, Percentile.1
```

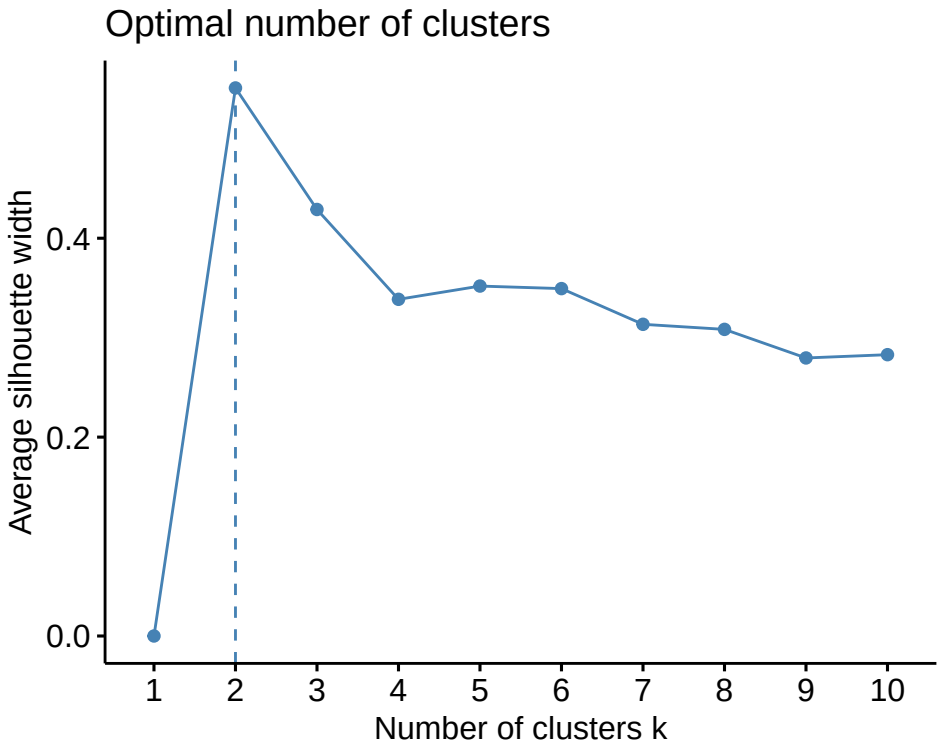
```r
# Run the fviz_nbclust function with our selected data and method "wss"
elbow_method <- factoextra::fviz_nbclust(k_means_data, FUNcluster = kmeans, method = "wss")

# View the plot
elbow_method
```



The silhouette method

Wow, that fviz_nbclust function was pretty nifty. Instead of needing to "manually" apply the elbow method by running multiple k_means models and plotting the calculated the total within cluster sum of squares for each potential value of k, fviz_nbclust handled all of this for us behind the scenes. Can we use it for the Silhouette Method as well? The Silhouette Method will evaluate the quality of clusters by how well each point fits within a cluster, maximizing average "silhouette" width.

```r
# Run the fviz_nbclust function with the method "silhouette"
silhouette_method <- factoextra::fviz_nbclust(k_means_data, FUNcluster = kmeans, method = "silhouette")

# View the plot
silhouette_method
```



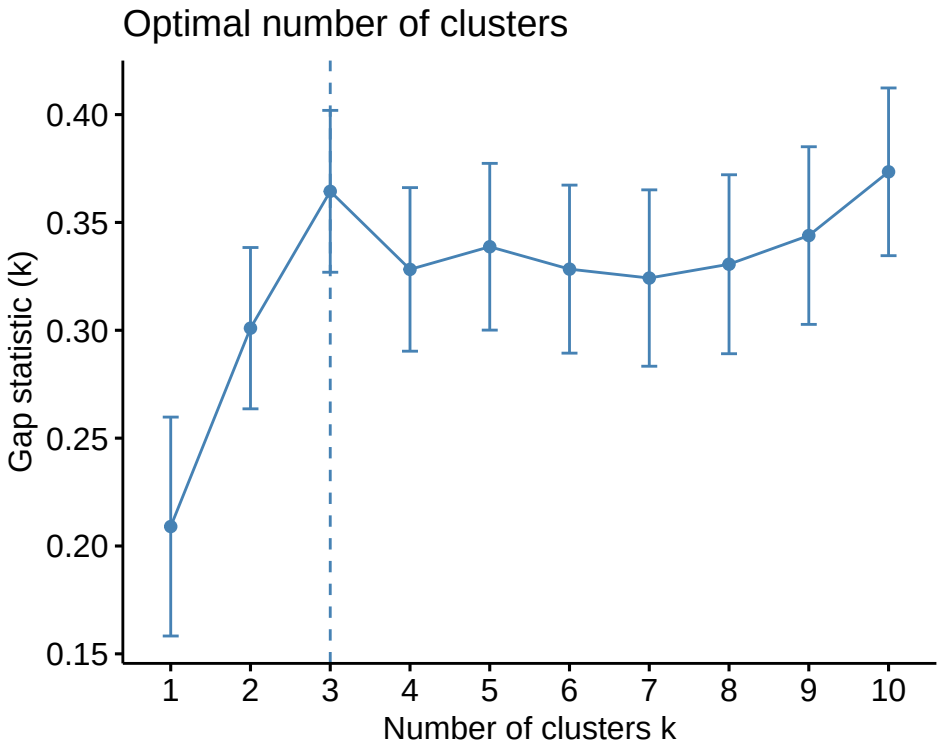
The gap statistic method

Marvelous! But hmm, it seems that our two methods so far disagree on the optimal number of clusters... Time to pull out the tie breaker.

For our final method, let's see what the Gap Statistic Method has to say about this. The Gap Statistic Method will compare the total variation within clusters for different values of k to the null hypothesis, maximizing the "gap." The "null hypothesis" refers to a uniformly distributed simulated reference dataset with no observable clusters, generated by aligning with the principle components of our original dataset. In other words, how much more variance is explained by k clusters in our dataset than in a fake dataset where all majors have equal salary potential?

Fortunately, we have the clusGap function to calculate this behind the scenes and the fviz_gap_stat function to visualize the results.

```
# Use the clusGap function to apply the Gap Statistic Method
gap_stat <- cluster::clusGap(k_means_data,
FUN = kmeans, nstart = 25, K.max = 10, B = 50)

# Use the fviz_gap_stat function to vizualize the results
gap_stat_method <- factoextra::fviz_gap_stat(gap_stat)

# View the plot
gap_stat_method
```



K-means algorithm

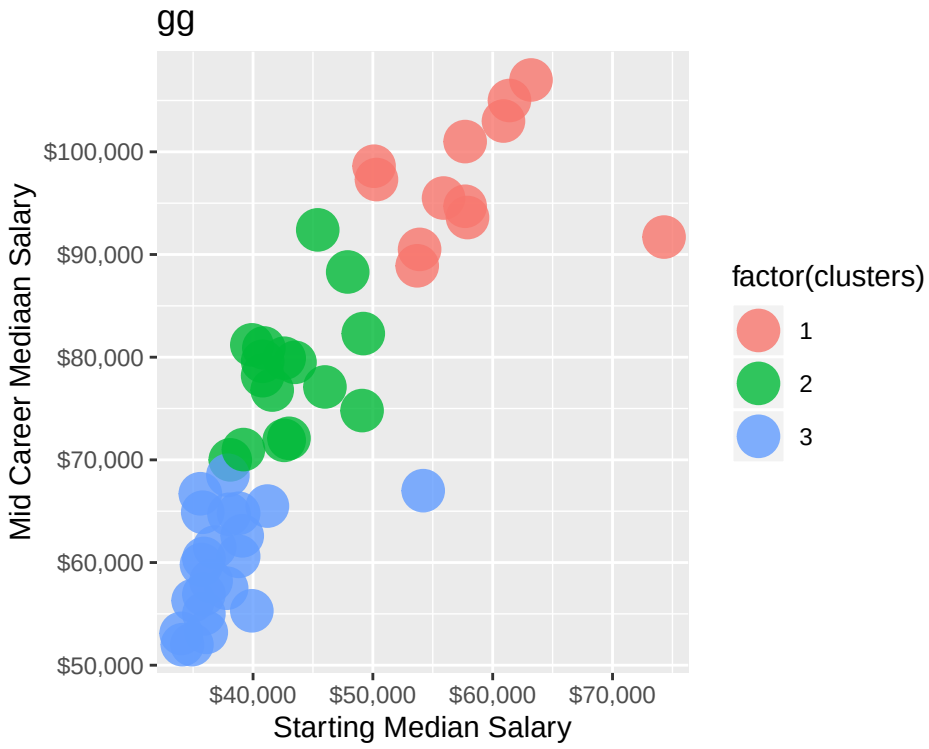
Looks like the Gap Statistic Method agreed with the Elbow Method! According to majority rule, let's use 3 for our optimal number of clusters. With this information, we can now run our k-means algorithm on the selected data. We will then add the resulting cluster information to label our original dataframe.

```
# Set a random seed
set.seed = 111

# Set k equal to the optimal number of clusters
num_clusters <- 3

# Run the k-means algorithm
k_means <- kmeans(x = k_means_data, centers = num_clusters, iter.max = 15,
    nstart = 25)

# Label the clusters of degrees_clean
degrees_labeled <- degrees_clean %>%
    mutate(clusters = k_means$cluster)
```

Visualizing the clusters

Now for the pretty part: visualizing our results. First let's take a look at how each cluster compares in Starting vs. Mid Career Median Salaries. What do the clusters say about the relationship between Starting and Mid Career salaries?

```r
# Graph the clusters by Starting and Mid Career Median Salaries
career_growth <- ggplot(degrees_labeled, aes(Starting.Median.Salary, Mid.Career.Median.Salary, color = f
                        geom_point(aes(), alpha = 4/5, size = 7) +
                        xlab("Starting Median Salary") +
                        ylab("Mid Career Mediaan Salary") +
                        ggtitle("gg") +
                        scale_x_continuous(labels = scales::dollar) +
                        scale_y_continuous(labels = scales::dollar)


# View the plot
career_growth
```



A deeper dive into the clusters

Unsurprisingly, most of the data points are hovering in the top left corner, with a relatively linear relationship. In other words, the higher your starting salary, the higher your mid career salary. The three clusters provide a level of delineation that intuitively supports this.

How might the clusters reflect potential mid career growth? There are also a couple curious outliers from clusters 1 and 3... perhaps this can be explained by investigating the mid career percentiles further, and exploring which majors fall in each cluster.

Right now, we have a column for each percentile salary value. In order to visualize the clusters and majors by mid career percentiles, we'll need to reshape the degrees_labeled data using tidyr's gather function to make a percentile key column and a salary value column to use for the axes of our following graphs. We'll then be able to examine the contents of each cluster to see what stories they might be telling us about the majors.

```
# Use the gather function to reshape degrees and
# use mutate() to reorder the new percentile column
degrees_perc <- degrees_labeled %>%
    select(College.Major, Percentile.10, Percentile.25, Mid.Career.Median.Salary, Percentile.75, Percen
    gather(key = percentile, value = salary,
           -c(College.Major, clusters)) %>%
            mutate(percentile = factor(percentile, levels=c("Percentile.10", "Percentile.25", "Mid.Caree
```

The liberal arts cluster

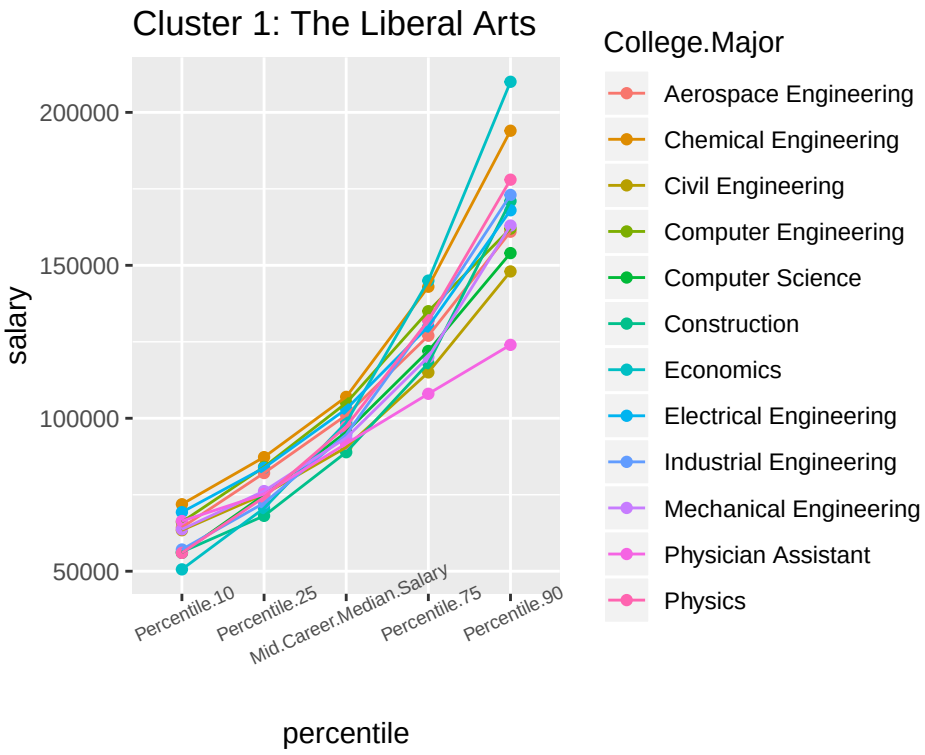
Let's graph Cluster 1 and examine the results. These Liberal Arts majors may represent the lowest percentiles with limited growth opportunity, but there is hope for those who make it! Music is our riskiest major with the lowest 10th percentile salary, but Drama wins the highest growth potential in the 90th percentile for this cluster (so don't let go of those Hollywood dreams!). Nursing is the outlier culprit of cluster number 1, with a higher safety net in the lowest percentile to the median. Otherwise, this cluster does represent the majors with limited growth opportunity.

An aside: It's worth noting that most of these majors leading to lower-paying jobs are women-dominated, according to this Glassdoor study. According to the research:

"The single biggest cause of the gender pay gap is occupation and industry sorting of men and women into jobs that pay differently throughout the economy. In the U.S., occupation and industry sorting explains 54 percent of the overall pay gap—by far the largest factor."

Does this imply that women are statistically choosing majors with lower pay potential, or do certain jobs pay less because women choose them. . . ?

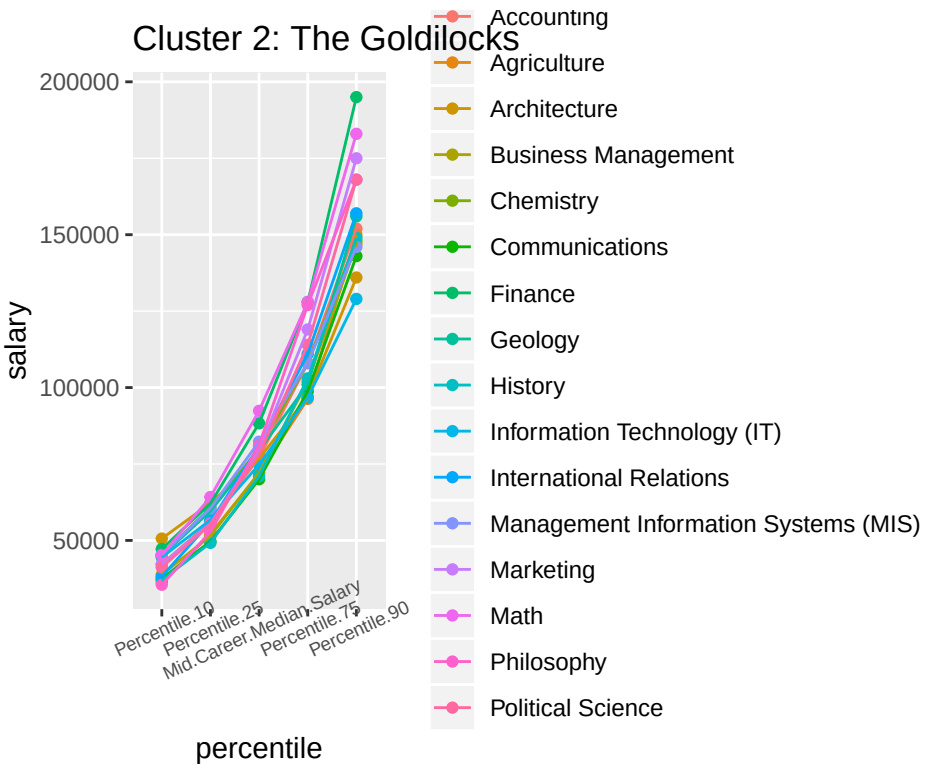
```
# Graph the majors of Cluster 1 by percentile
cluster_1 <- degrees_perc %>% filter(clusters == 1) %>%
    ggplot(aes(x= percentile, y = salary, color = College.Major, group = College.Major)) +
    geom_point()+
    geom_line() +
    ggtitle("Cluster 1: The Liberal Arts")+
    theme(axis.text.x = element_text(size = 7, angle = 25))


# View the plot
cluster_1
```



The goldilocks cluster

On to Cluster 2, right in the middle! Accountants are known for having stable job security, but once you're in the big leagues you may be surprised to find that Marketing or Philosophy can ultimately result in higher salaries. The majors of this cluster are fairly middle of the road in our dataset, starting off not too low and not too high in the lowest percentile. However, this cluster also represents the majors with the greatest differential between the lowest and highest percentiles.

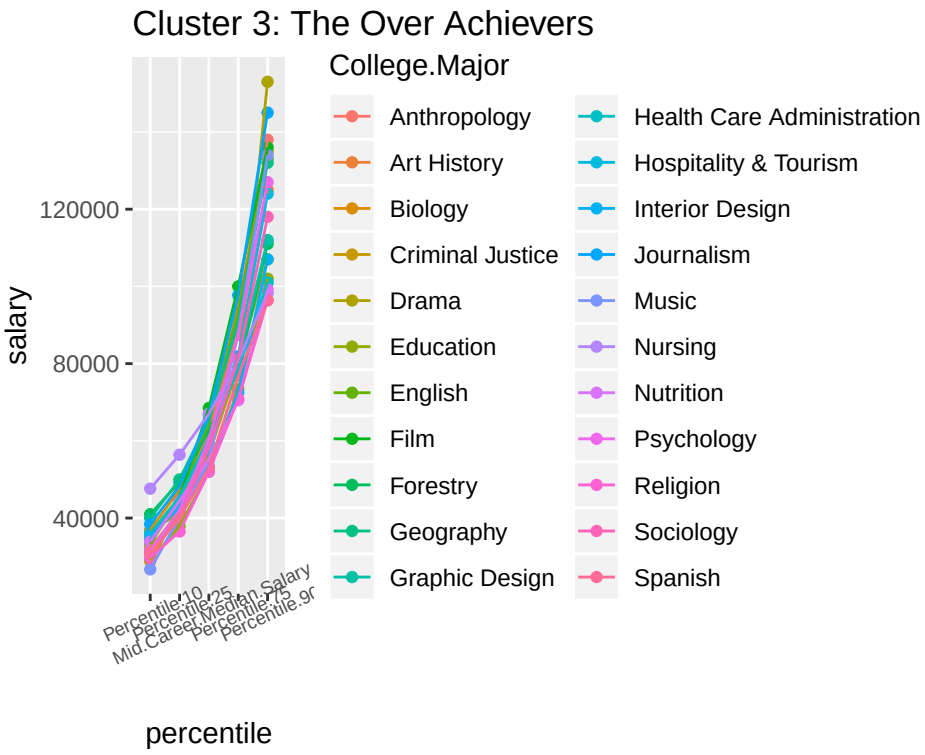
```
# Modify the previous plot to display Cluster 2
cluster_2 <-  degrees_perc %>% filter(clusters == 2) %>%
    ggplot(aes(x= percentile, y = salary, color = College.Major, group = College.Major)) +
    geom_point()+
    geom_line() +
    ggtitle("Cluster 2: The Goldilocks")+
    theme(axis.text.x = element_text(size = 7, angle = 25))

# View the plot
cluster_2
```



The over achiever cluster

Finally, let's visualize Cluster 3. If you want financial security, these are the majors to choose from. Besides our one previously observed outlier now identifiable as Physician Assistant lagging in the highest percentiles, these heavy hitters and solid engineers represent the highest growth potential in the 90th percentile, as well as the best security in the 10th percentile rankings. Maybe those Freakonomics guys are on to something...

```
# Modify the previous plot to display Cluster 3
cluster_3 <-  degrees_perc %>% filter(clusters == 3) %>%
    ggplot(aes(x= percentile, y = salary, color = College.Major, group = College.Major)) +
    geom_point()+
    geom_line() +
    ggtitle("Cluster 3: The Over Achievers")+
    theme(axis.text.x = element_text(size = 7, angle = 25))

# View the plot
cluster_3
```



Every major's wonderful

Thus concludes our journey exploring salary projections by college major via a k-means clustering analysis! Dealing with unsupervized data always requires a bit of creativity, such as our usage of three popular methods to determine the optimal number of clusters. We also used visualizations to interpret the patterns revealed by our three clusters and tell a story.

Which two careers tied for the highest career percent growth? While it's tempting to focus on starting career salaries when choosing a major, it's important to also consider the growth potential down the road. Keep in mind that whether a major falls into the Liberal Arts, Goldilocks, or Over Achievers cluster, one's financial destiny will certainly be influenced by numerous other factors including the school attended, location, passion or talent for the subject, and of course the actual career(s) pursued.

A similar analysis to evaluate these factors may be conducted on the additional data provided by the Wall Street Journal article, comparing salary potential by type and region of college attended. But in the meantime, here's some inspiration from xkcd for any students out there still struggling to choose a major.

```
# Sort degrees by Career.Percent.Growth
degrees_labeled %>% arrange(desc(Career.Percent.Growth))
```

```
## # A tibble: 50 x 9
##    College.Major Starting.Median~ Mid.Career.Medi~ Career.Percent.~
##    <chr>                    <dbl>            <dbl>            <dbl>
##  1 Math                     45400            92400            1.03
##  2 Philosophy               39900            81200            1.03
##  3 Internationa~            40900            80900            0.978
##  4 Economics                50100            98600            0.968
##  5 Marketing                40800            79600            0.951
##  6 Physics                  50300            97300            0.934
##  7 Political Sc~            40800            78200            0.917
```

```
##  8 Chemistry                42600            79900            0.876
##  9 Journalism               35600            66700            0.874
## 10 Architecture             41600            76800            0.846
## # ... with 40 more rows, and 5 more variables: Percentile.10 <dbl>,
## #   Percentile.25 <dbl>, Percentile.75 <dbl>, Percentile.90 <dbl>,
## #   clusters <int>
```

```r
# Identify the two majors tied for highest career growth potential
highest_career_growth <- c('Math','Philosophy')
```