Degrees That Pay

Shreya Rao

Data collected from a year-long survey of 1.2 million people with only a bachelor's degree by PayScale Inc., made available by the Wall Street Journal for their article Ivy League's Big Edge: Starting Pay.

```r
# Load relevant packages
library(tidyr)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

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

```r
library(readr)
library(ggplot2)
library(cluster)
library(factoextra)
```

```
## Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WBa
```

```r
# Read in the dataset
degrees <- read.table("C:/Users/Shreya/Documents/Projects/degrees.txt", sep = ",", header = T)
colnames(degrees) <- c("College.Major", "Starting.Median.Salary",
                       "Mid.Career.Median.Salary", "Career.Percent.Growth",
                       "Percentile.10", "Percentile.25", "Percentile.75",
                       "Percentile.90")

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

Data Cleaning:

```r
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)
```

Elbow Method:

```r
# 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.10, Percentile.90)

k_means_data = scale(k_means_data)

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

# View the plot
elbow_method
```

Optimal number of clusters
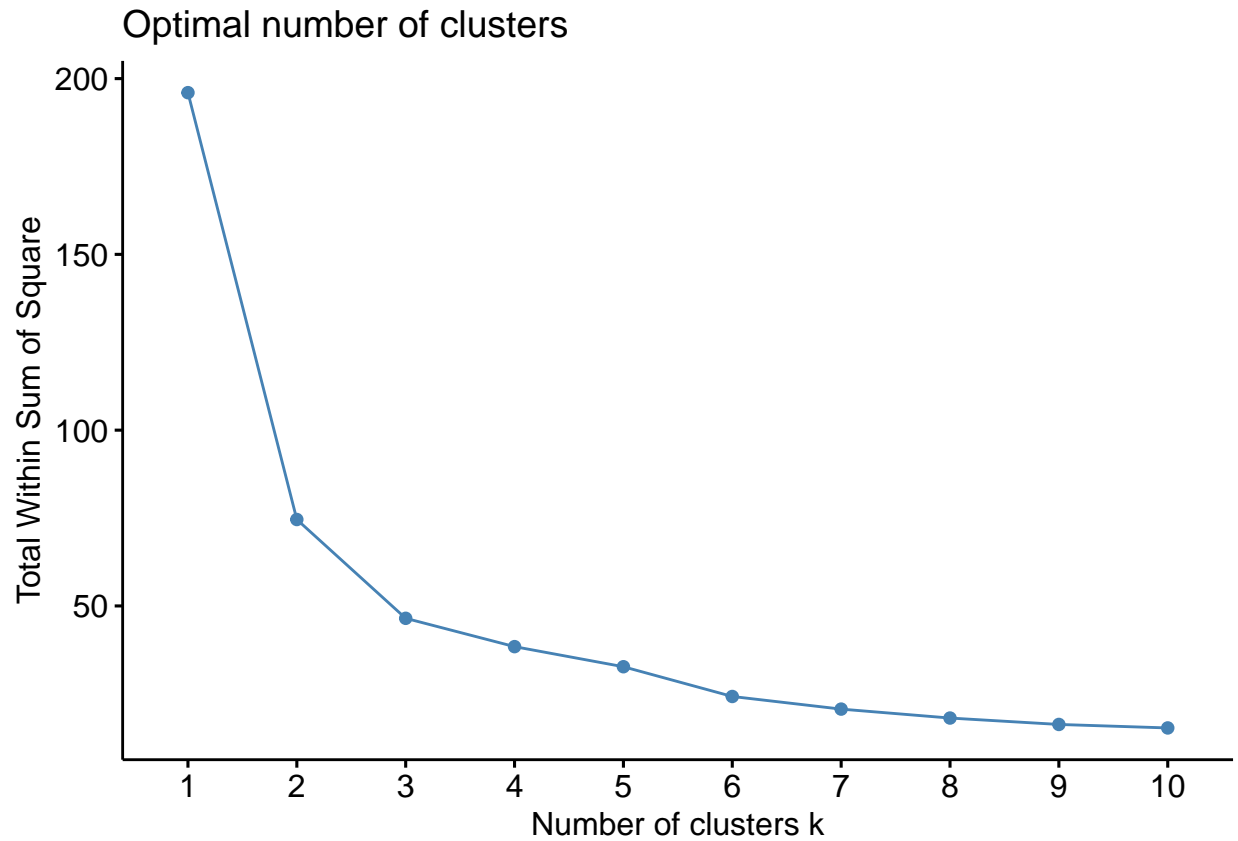
The Silhouette Method: The Silhouette Method will evaluate the quality of clusters by how well each point fits within a cluster, maximizing average "silhouette" width.

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

# View the plot
silhouette_method
```
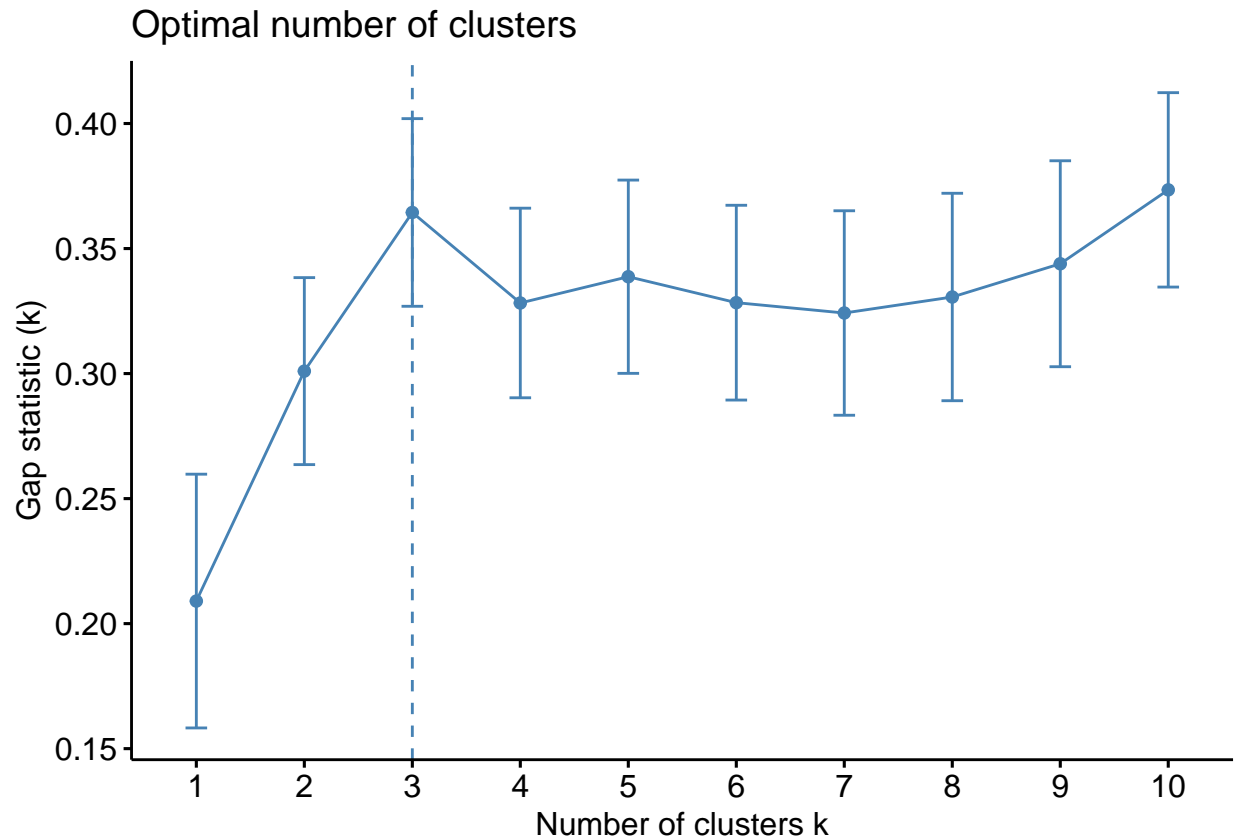
Optimal number of clusters
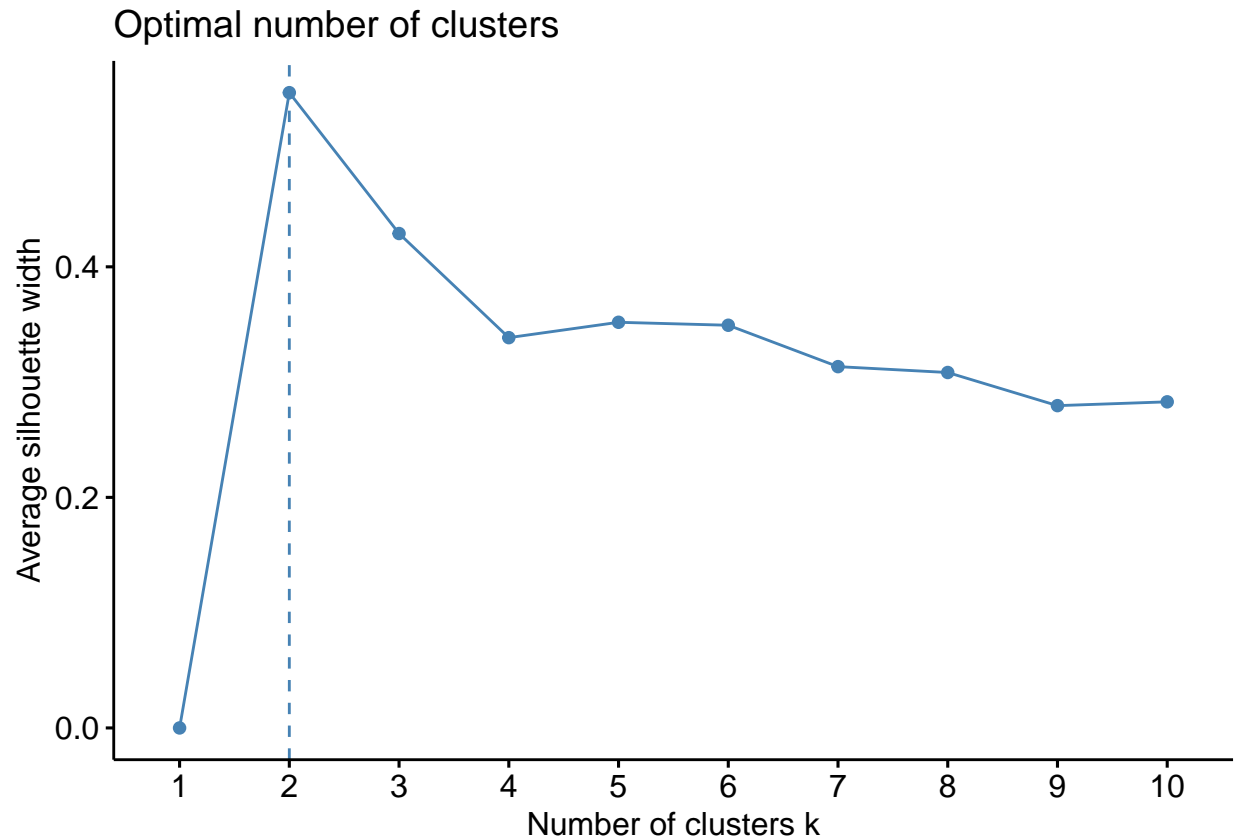
The Gap Statistic Method: The Gap Statistic Method will compare the total variation within clusters for different values of k to the null hypothesis, maximizing the "gap." The "null hypothesis" refers to a uniformly distributed simulated reference dataset with no observable clusters, generated by aligning with the principle components of the original dataset. In other words, how much more variance is explained by k clusters in the dataset than in a fake dataset where all majors have equal salary potential?

```r
# Use the clusGap function to apply the Gap Statistic Method
gap_stat <- 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 <- fviz_gap_stat(gap_stat)

# View the plot
gap_stat_method
```

Optimal number of clusters

The Gap Statistic Method agreed with the Elbow Method. So I'm using 3 for optimal number of clusters.

K-Means Clustering:

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

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

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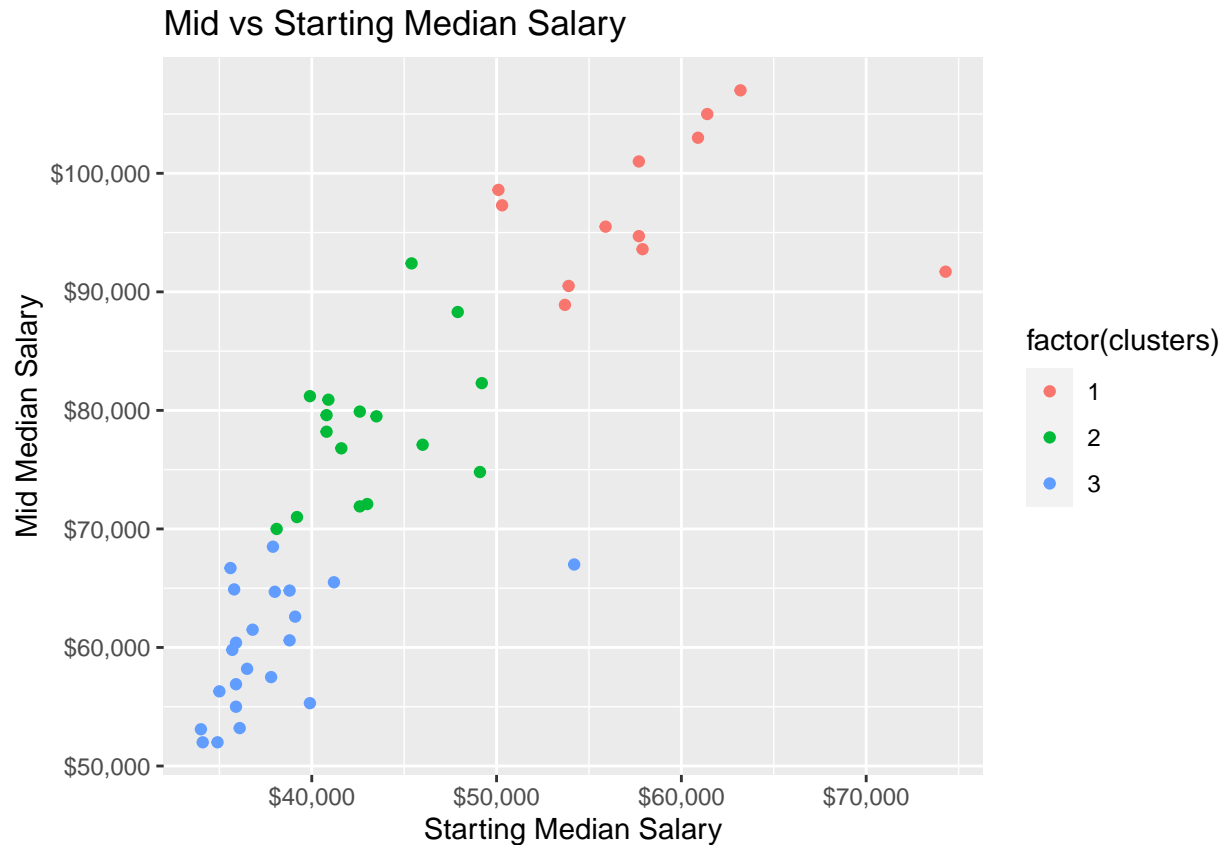
Visualizing The Clusters

```
# Graph the clusters by Starting and Mid Career Median Salaries
career_growth <- ggplot(degrees_labeled, aes(x=Starting.Median.Salary,y=Mid.Career.Median.Salary,
                                             color=factor(clusters))) +
geom_point() +
xlab("Starting Median Salary") + ylab("Mid Median Salary") +
ggtitle("Mid vs Starting Median Salary") +
scale_x_continuous(labels = scales::dollar) + scale_y_continuous(labels = scales::dollar)

# View the plot
career_growth
```

Mid vs Starting Median Salary

Analyzing The Clusters

```
# 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, Percentile.90, clusters) %>%
    gather(key=percentile, value=salary, -c(College.Major, clusters)) %>%
    mutate(percentile = factor(percentile,
                               levels=c('Percentile.10', 'Percentile.25',
                                  'Mid.Career.Median.Salary','Percentile.75',
                                  'Percentile.90')))
```

Liberal Arts Cluster: Music is the riskiest major with the lowest 10th percentile salary, but Drama wins the highest growth potential in the 90th percentile for this cluster. 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.

```
# Graph the majors of Cluster 1 by percentile
cluster_1 <- degrees_perc %>% filter(clusters == 3) %>%
    ggplot(aes(x=percentile, y=salary, group=College.Major, color=College.Major)) +
    geom_point() +
    geom_line() +
    ggtitle("Cluster 1: The Liberal Arts") +
    theme(axis.text.x = element_text(size=7, angle=25))
```
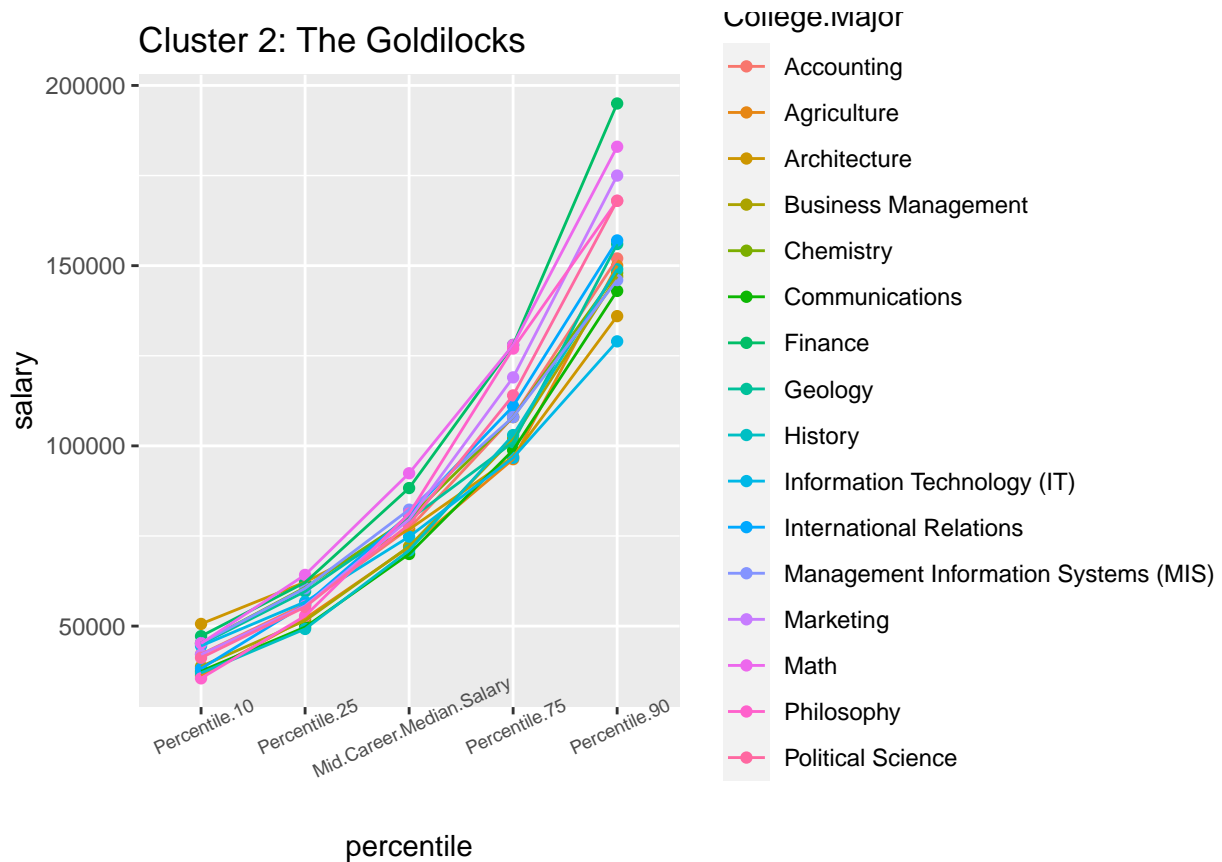
```
# View the plot
cluster_1
```


Cluster 1: The Liberal Arts
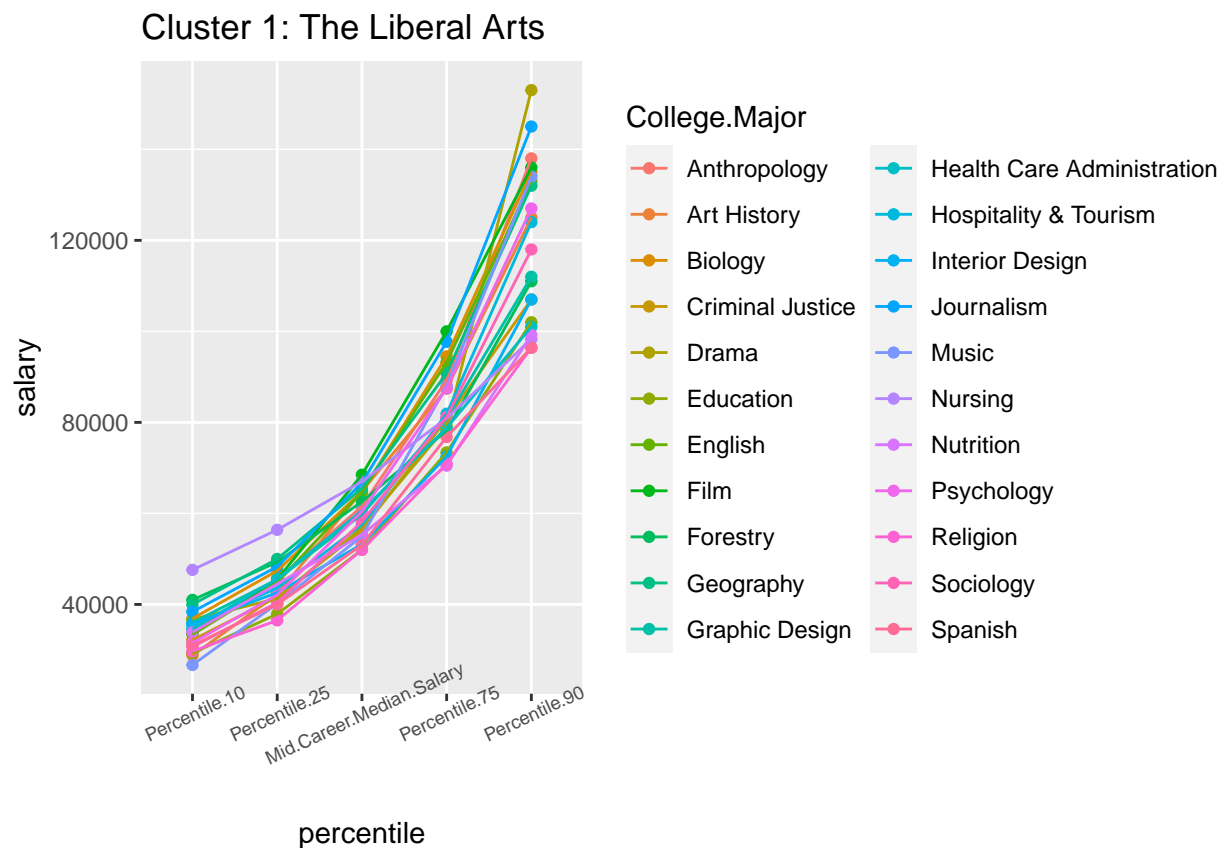
Goldilocks Cluster: Accountants are known for having stable job security, but once you're in the big leagues Marketing or Philosophy can ultimately result in higher salaries. The majors of this cluster are fairly middle of the road in the dataset, starting off not too low and not too high in the lowest 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, group=College.Major, color=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
```

Cluster 2: The Goldilocks

College.Major
- Accounting
- Agriculture
- Architecture
- Business Management
- Chemistry
- Communications
- Finance
- Geology
- History
- Information Technology (IT)
- International Relations
- Management Information Systems (MIS)
- Marketing
- Math
- Philosophy
- Political Science

Over Achiever Cluster: If you want financial security, these are the majors to choose from. Besides 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.

```
# Modify the previous plot to display Cluster 3
cluster_3 <- degrees_perc %>% filter(clusters == 1) %>%
    ggplot(aes(x=percentile, y=salary, group=College.Major, color=College.Major)) +
    geom_point() +
    geom_line() +
    ggtitle("Cluster 3: The Overachievers") +
    theme(axis.text.x = element_text(size=7, angle=25))

# View the plot
cluster_3
```

Cluster 3: The Overachievers