1. [Introduction](file:///C:\\Users\\Andrew\\Downloads\\ProjectProgress%20Report-OPerera.docx" \l "iq0g4obyk) (Andrew)
   1. [Project Background and Description](file:///C:\Users\Andrew\Downloads\ProjectProgress%20Report-OPerera.docx#jdbro0a3nys3)
   2. [Project Scope and Context for Analysis](file:///C:\Users\Andrew\Downloads\ProjectProgress%20Report-OPerera.docx#fr1qdv0mijd)
   3. [Business Questions](file:///C:\\Users\\Andrew\\Downloads\\ProjectProgress%20Report-OPerera.docx" \l "xz2gj2s5n1ft)
2. [Data Overview](file:///C:\Users\Andrew\Downloads\ProjectProgress%20Report-OPerera.docx#ke7qc7g7ymm4)
   1. [Data Source](file:///C:\Users\Andrew\Downloads\ProjectProgress%20Report-OPerera.docx#rz88riuq57lk) (Andrew)

Provide details of the IBM site and how data was created

<https://www.ibm.com/communities/analytics/watson-analytics-blog/hr-employee-attrition/>

* 1. [Data Quality](file:///C:\Users\Andrew\Downloads\ProjectProgress%20Report-OPerera.docx#mgmx7dt7lkht) (Andrew)

Discuss dataset was reviewed and no blanks found – high quality dataset

Discuss that the dataset was created by IBM’s Watson team, which may be why it is so complete.

Discuss that there is no way to know if it was based on any real dataset.

(does anyone have any thought to add about the dataset in general

* 1. [Data Dictionary](file:///C:\Users\Andrew\Downloads\ProjectProgress%20Report-OPerera.docx#io1soyf4zg7y) (Andrew – clean up descriptions to make them more readable)

|  |  |
| --- | --- |
| **Variable** | **Description** |
| Age | Numerical Value |
| Attrition | Employee leaving the company (0=no, 1=yes) |
| BusinessTravel | (1=No Travel, 2=Travel Frequently, 3=Tavel Rarely) |
| DailyRate | Numerical Value - Salary Level |
| Department | (1=HR, 2=R&D, 3=Sales) |
| DistanceFromHome | Numerical Value - THE DISTANCE FROM WORK TO HOME |
| Education | Numerical Value |
| EducationField | (1=HR, 2=LIFE SCIENCES, 3=MARKETING, 4=MEDICAL SCIENCES, 5=OTHERS, 6= TEHCNICAL) |
| EmployeeCount | Numerical Value |
| EmployeeNumber | Numerical Value - EMPLOYEE ID |
| EnvironmentSatisfaction | Numerical Value - SATISFACTION WITH THE ENVIROMENT |
| Gender | (1=FEMALE, 2=MALE) |
| HourlyRate | Numerical Value - HOURLY SALARY |
| JobInvolvement | Numerical Value - JOB INVOLVEMENT |
| JobLevel | Numerical Value - LEVEL OF JOB |
| JobRole | (1=HC REP, 2=HR, 3=LAB TECHNICIAN, 4=MANAGER, 5= MANAGING DIRECTOR, 6= REASEARCH DIRECTOR, 7= RESEARCH SCIENTIST, 8=SALES EXECUTIEVE, 9= SALES REPRESENTATIVE) |
| JobSatisfaction | Numerical Value - SATISFACTION WITH THE JOB |
| MaritalStatus | (1=DIVORCED, 2=MARRIED, 3=SINGLE) |
| MonthlyIncome | Numerical Value - MONTHLY SALARY |
| MonthlyRate | Numerical Value - MONTHY RATE |
| NumCompaniesWorked | Numerical Value - NO. OF COMPANIES WORKED AT |
| Over18 | (1=YES, 2=NO) |
| OverTime | (1=NO, 2=YES) |
| PercentSalaryHike | Numerical Value - PERCENTAGE INCREASE IN SALARY.  The parentage of change in salary between 2 year (2017, 2018). |
| PerformanceRating | Numerical Value - ERFORMANCE RATING |
| RelationshipSatisfaction | Numerical Value - RELATIONS SATISFACTION |
| StandardHours | Numerical Value - STANDARD HOURS |
| StockOptionLevel | Numerical Value - STOCK OPTIONS. |
| TotalWorkingYears | Numerical Value - TOTAL YEARS WORKED |
| TrainingTimesLastYear | Numerical Value - HOURS SPENT TRAINING |
| WorkLifeBalance | Numerical Value - TIME SPENT BEWTWEEN WORK AND OUTSIDE |
| YearsAtCompany | Numerical Value - TOTAL NUMBER OF YEARS AT THE COMPNAY |
| YearsInCurrentRole | Numerical Value -YEARS IN CURRENT ROLE |
| YearsSinceLastPromotion | Numerical Value - LAST PROMOTION |
| YearsWithCurrManager | Numerical Value - YEARS SPENT WITH CURRENT MANAGER |

* 1. Data Summary and Structure
     1. Data Summary

## Age Attrition BusinessTravel DailyRate   
## Min. :18.00 Length:1470 Length:1470 Min. : 102.0   
## 1st Qu.:30.00 Class :character Class :character 1st Qu.: 465.0   
## Median :36.00 Mode :character Mode :character Median : 802.0   
## Mean :36.92 Mean : 802.5   
## 3rd Qu.:43.00 3rd Qu.:1157.0   
## Max. :60.00 Max. :1499.0

## Department DistanceFromHome Education EducationField   
## Length:1470 Min. : 1.000 Min. :1.000 Length:1470   
## Class :character 1st Qu.: 2.000 1st Qu.:2.000 Class :character   
## Mode :character Median : 7.000 Median :3.000 Mode :character   
## Mean : 9.193 Mean :2.913   
## 3rd Qu.:14.000 3rd Qu.:4.000   
## Max. :29.000 Max. :5.000

## EmployeeCount EmployeeNumber EnvironmentSatisfaction Gender   
## Min. :1 Min. : 1.0 Min. :1.000 Length:1470   
## 1st Qu.:1 1st Qu.: 491.2 1st Qu.:2.000 Class :character   
## Median :1 Median :1020.5 Median :3.000 Mode :character   
## Mean :1 Mean :1024.9 Mean :2.722   
## 3rd Qu.:1 3rd Qu.:1555.8 3rd Qu.:4.000   
## Max. :1 Max. :2068.0 Max. :4.000

## HourlyRate JobInvolvement JobLevel JobRole   
## Min. : 30.00 Min. :1.00 Min. :1.000 Length:1470   
## 1st Qu.: 48.00 1st Qu.:2.00 1st Qu.:1.000 Class :character   
## Median : 66.00 Median :3.00 Median :2.000 Mode :character   
## Mean : 65.89 Mean :2.73 Mean :2.064   
## 3rd Qu.: 83.75 3rd Qu.:3.00 3rd Qu.:3.000   
## Max. :100.00 Max. :4.00 Max. :5.000

## JobSatisfaction MaritalStatus MonthlyIncome MonthlyRate   
## Min. :1.000 Length:1470 Min. : 1009 Min. : 2094   
## 1st Qu.:2.000 Class :character 1st Qu.: 2911 1st Qu.: 8047   
## Median :3.000 Mode :character Median : 4919 Median :14236   
## Mean :2.729 Mean : 6503 Mean :14313   
## 3rd Qu.:4.000 3rd Qu.: 8379 3rd Qu.:20462   
## Max. :4.000 Max. :19999 Max. :26999

## NumCompaniesWorked Over18 OverTime   
## Min. :0.000 Length:1470 Length:1470   
## 1st Qu.:1.000 Class :character Class :character   
## Median :2.000 Mode :character Mode :character   
## Mean :2.693   
## 3rd Qu.:4.000   
## Max. :9.000

## PercentSalaryHike PerformanceRating RelationshipSatisfaction  
## Min. :11.00 Min. :3.000 Min. :1.000   
## 1st Qu.:12.00 1st Qu.:3.000 1st Qu.:2.000   
## Median :14.00 Median :3.000 Median :3.000   
## Mean :15.21 Mean :3.154 Mean :2.712   
## 3rd Qu.:18.00 3rd Qu.:3.000 3rd Qu.:4.000   
## Max. :25.00 Max. :4.000 Max. :4.000

## StandardHours StockOptionLevel TotalWorkingYears TrainingTimesLastYear  
## Min. :80 Min. :0.0000 Min. : 0.00 Min. :0.000   
## 1st Qu.:80 1st Qu.:0.0000 1st Qu.: 6.00 1st Qu.:2.000   
## Median :80 Median :1.0000 Median :10.00 Median :3.000   
## Mean :80 Mean :0.7939 Mean :11.28 Mean :2.799   
## 3rd Qu.:80 3rd Qu.:1.0000 3rd Qu.:15.00 3rd Qu.:3.000   
## Max. :80 Max. :3.0000 Max. :40.00 Max. :6.000

## WorkLifeBalance YearsAtCompany YearsInCurrentRole  
## Min. :1.000 Min. : 0.000 Min. : 0.000   
## 1st Qu.:2.000 1st Qu.: 3.000 1st Qu.: 2.000   
## Median :3.000 Median : 5.000 Median : 3.000   
## Mean :2.761 Mean : 7.008 Mean : 4.229   
## 3rd Qu.:3.000 3rd Qu.: 9.000 3rd Qu.: 7.000   
## Max. :4.000 Max. :40.000 Max. :18.000

## YearsSinceLastPromotion YearsWithCurrManager  
## Min. : 0.000 Min. : 0.000   
## 1st Qu.: 0.000 1st Qu.: 2.000   
## Median : 1.000 Median : 3.000   
## Mean : 2.188 Mean : 4.123   
## 3rd Qu.: 3.000 3rd Qu.: 7.000   
## Max. :15.000 Max. :17.000

* + 1. Data Structure (Transformation question - Looks like all data is factorized to numeric factors from the start. Why did we do that? How does it help to do that at the very beginning? Why does changing from actual values or discretized values to numeric factors a good idea at this point?)

## Classes 'tbl\_df', 'tbl' and 'data.frame': 1470 obs. of 28 variables:  
## $ Age : Factor w/ 6 levels "1","2","3","4",..: 1 2 1 1 1 1 1 1 1 2 ...  
## $ Attrition : Factor w/ 2 levels "1","2": 2 2 1 1 2 2 2 1 1 1 ...  
## $ BusinessTravel : Factor w/ 3 levels "1","2","3": 1 3 3 1 3 2 2 1 1 3 ...  
## $ Department : Factor w/ 3 levels "1","2","3": 2 3 3 2 2 3 3 2 2 2 ...  
## $ DistanceFromHome : Factor w/ 6 levels "1","2","3","4",..: 2 5 2 1 1 1 1 3 1 3 ...  
## $ Education : Factor w/ 5 levels "1","2","3","4",..: 1 1 3 2 3 2 3 3 3 2 ...  
## $ EducationField : Factor w/ 6 levels "1","2","3","4",..: 4 3 4 2 2 4 3 4 2 2 ...  
## $ EnvironmentSatisfaction : Factor w/ 4 levels "1","2","3","4": 3 4 4 2 3 2 2 2 4 3 ...  
## $ Gender : Factor w/ 2 levels "1","2": 2 2 1 2 2 1 2 1 1 2 ...  
## $ JobInvolvement : Factor w/ 4 levels "1","2","3","4": 3 3 2 3 3 3 3 3 3 3 ...  
## $ JobLevel : Factor w/ 5 levels "1","2","3","4",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ JobRole : Factor w/ 9 levels "1","2","3","4",..: 3 9 9 7 3 9 9 7 3 7 ...  
## $ JobSatisfaction : Factor w/ 4 levels "1","2","3","4": 3 3 3 4 3 4 2 3 4 4 ...  
## $ MaritalStatus : Factor w/ 3 levels "1","2","3": 3 3 3 3 3 3 3 3 3 3 ...  
## $ MonthlyIncome : Factor w/ 6 levels "1","2","3","4",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ NumCompaniesWorked : Factor w/ 10 levels "0","1","2","3",..: 2 2 2 2 2 2 2 2 2 2 ...  
## $ OverTime : Factor w/ 2 levels "1","2": 1 2 1 1 1 2 2 1 1 1 ...  
## $ PercentSalaryHike : Factor w/ 6 levels "2","3","4","5",..: 1 5 1 3 2 1 2 3 3 2 ...  
## $ PerformanceRating : Factor w/ 2 levels "3","4": 1 1 1 1 1 1 1 1 1 1 ...  
## $ RelationshipSatisfaction: Factor w/ 4 levels "1","2","3","4": 4 4 1 4 3 3 4 3 3 4 ...  
## $ StockOptionLevel : Factor w/ 4 levels "0","1","2","3": 1 1 1 1 1 1 1 1 1 1 ...  
## $ TotalWorkingYears : Factor w/ 8 levels "1","2","3","4",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ TrainingTimesLastYear : Factor w/ 7 levels "0","1","2","3",..: 1 3 3 3 3 3 4 5 6 7 ...  
## $ WorkLifeBalance : Factor w/ 4 levels "1","2","3","4": 3 2 3 3 3 4 3 1 4 3 ...  
## $ YearsAtCompany : Factor w/ 10 levels "1","2","3","4",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ YearsInCurrentRole : Factor w/ 7 levels "1","2","3","4",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ YearsSinceLastPromotion : Factor w/ 7 levels "1","2","3","4",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ YearsWithCurrManager : Factor w/ 7 levels "1","2","3","4",..: 1 1 1 1 1 1 1 1 1 1 ...

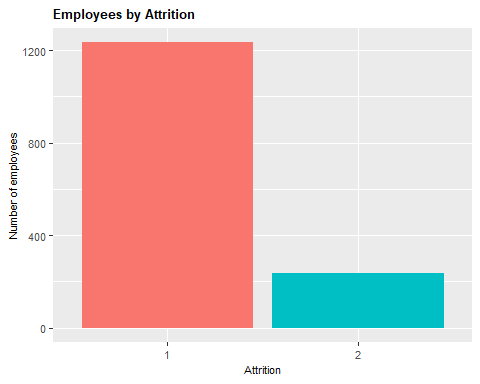
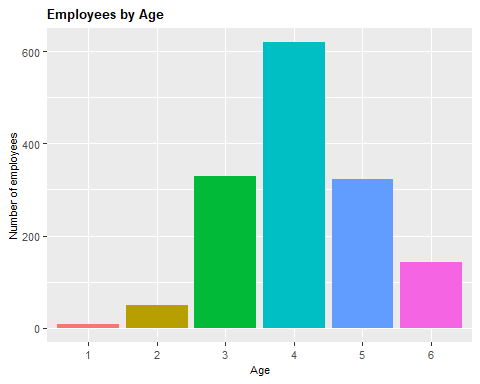
* 1. Data Selection (Andrew)

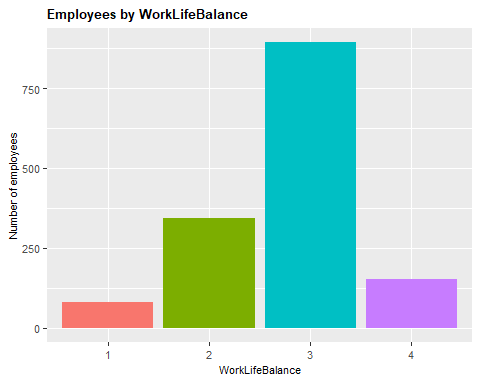
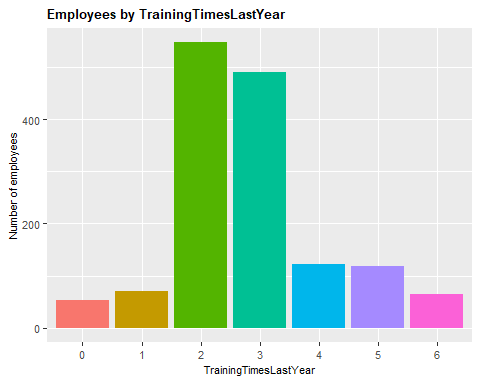
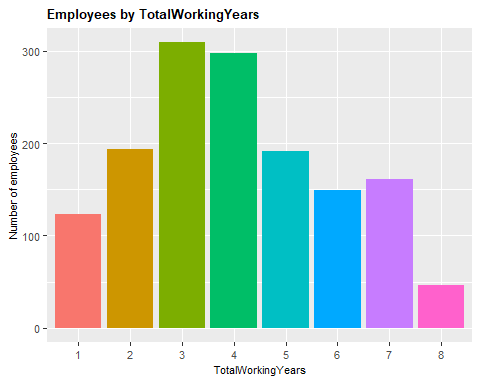
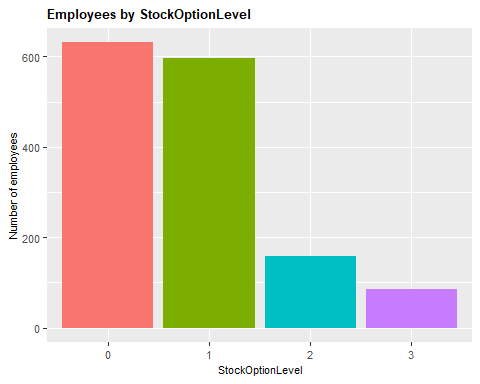
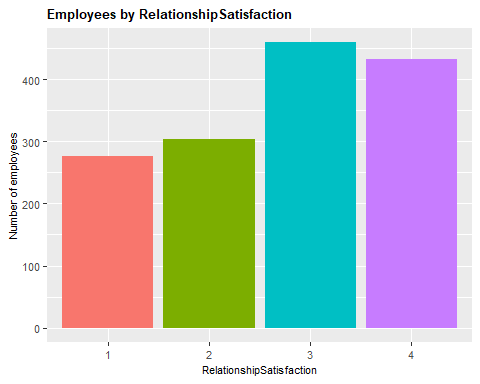
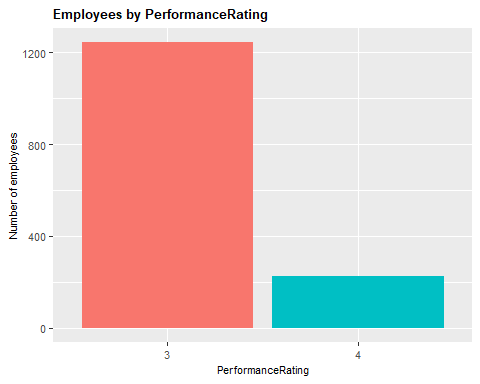
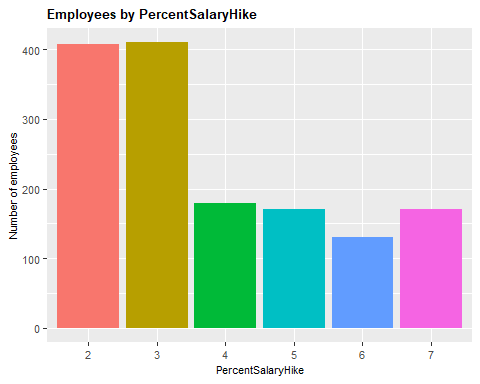
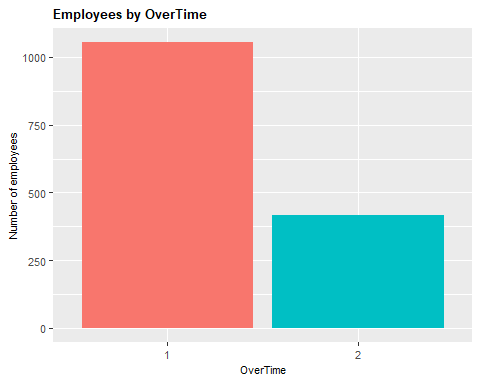
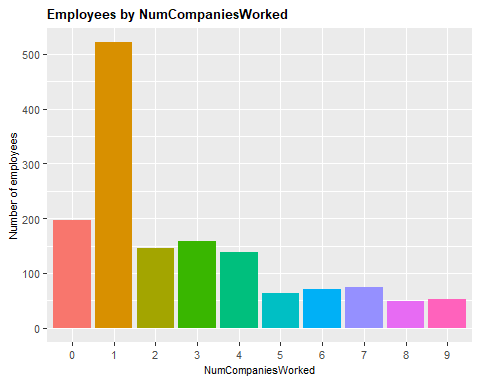
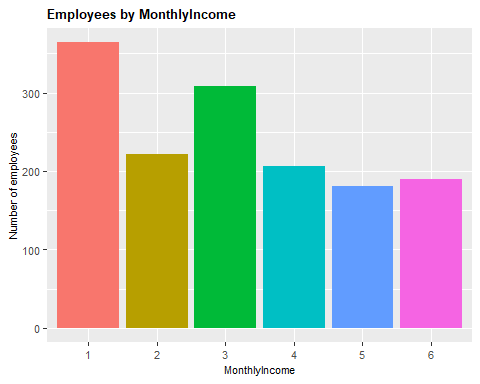
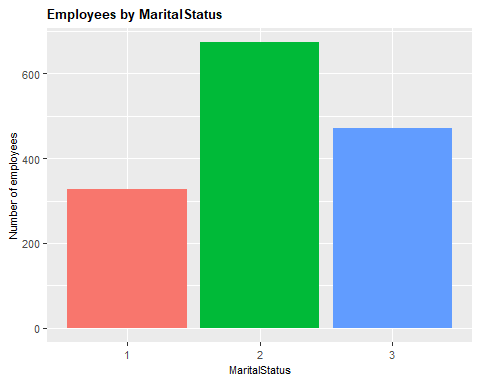
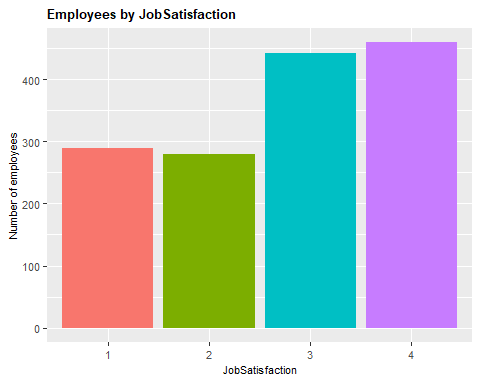
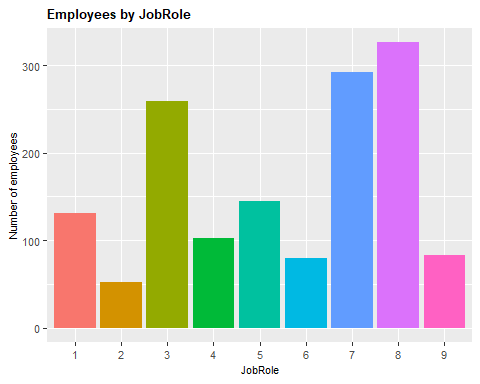
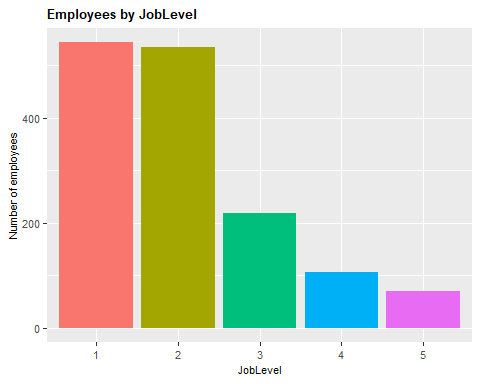
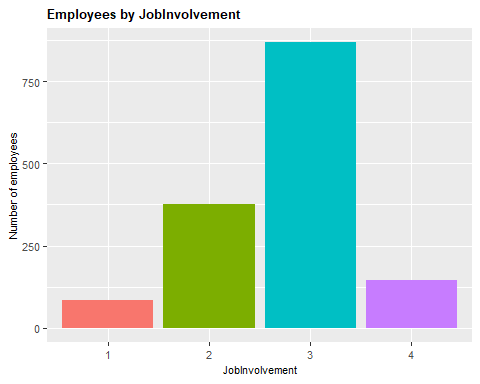
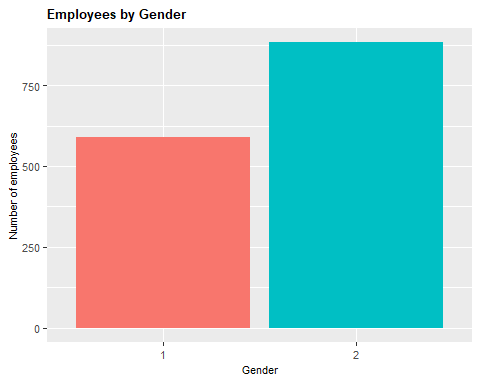
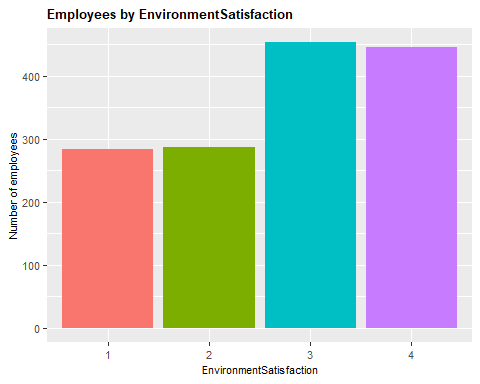
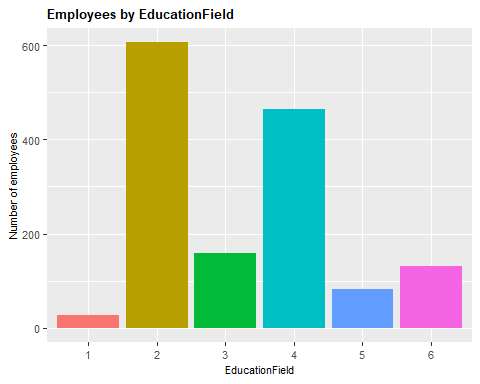
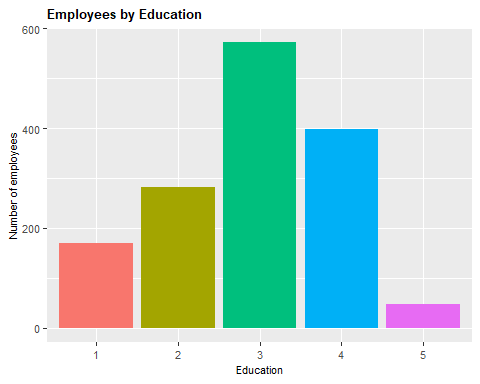
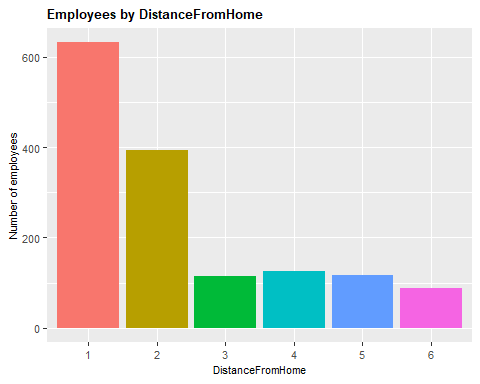
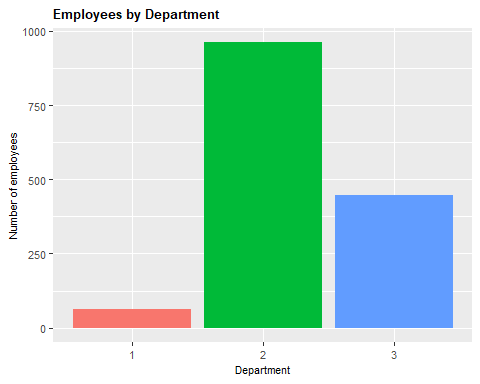
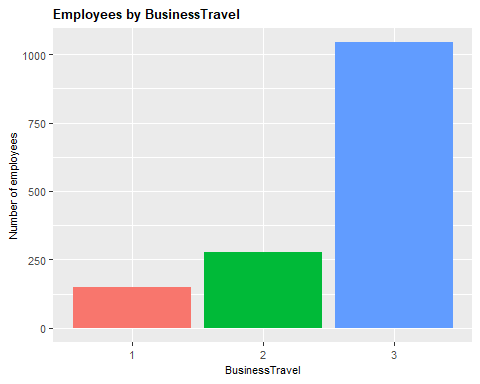
Removed data elements (Andrew - provide explanation for each):

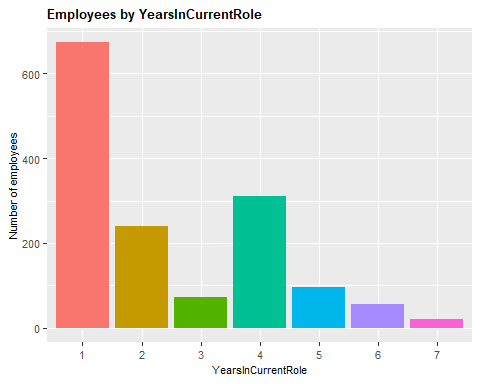
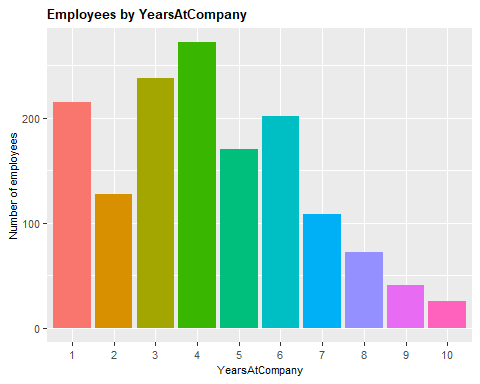
* DailyRate
* HourlyRate
* MonthlyRate
* Over18
* EmployeeCount', # 1 for everyone
* EmployeeNumber
* StandardHours

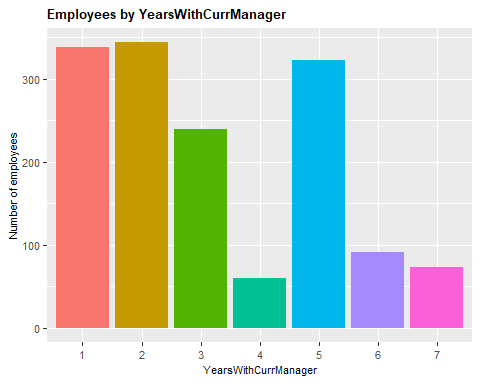
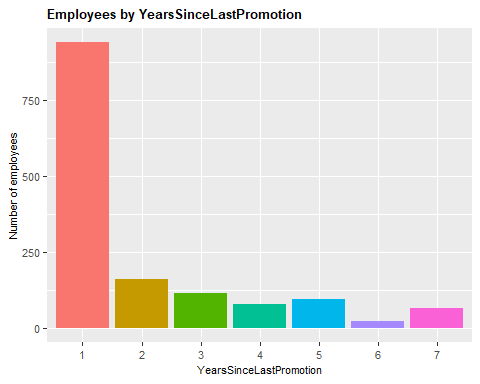
Explain how some of the variables were convergent (Relationship Satisfaction, Job Satisfaction, Job Involvement may all be proxy for engagement, but no way to know)

* 1. Data Visualizations (Andrew to add figure names to each visualization)









* 1. Data Transformation (Andrew)

Explain purpose for discretizing each of the following variables:

Age

DistanceFromHome

MonthlyIncome

PercentSalaryHike

TotalWorkingYears

YearsAtCompany

YearsInCurrentRole

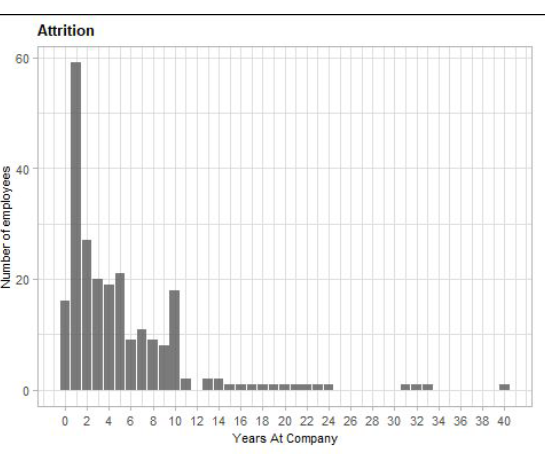
YearsSinceLastPromotion

YearsWithCurrManager

* 1. Additional transformation for <2 and <5 year attrition

Explain why we went after the two subsections of <2 and <5 based on what we saw in turnover and realte back to costs/investments in hiring and developing employees.

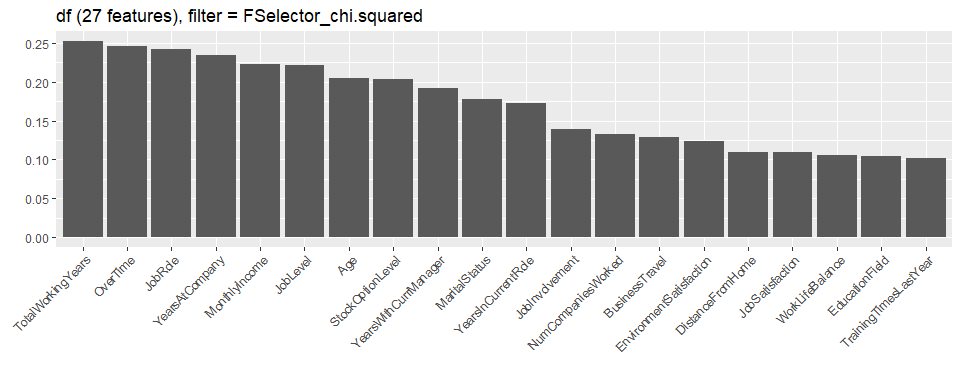
(Need the code and graphic for the graph from slide 11 of the deck that shows that the attrition rate is highest in the first years of employement. This one:)



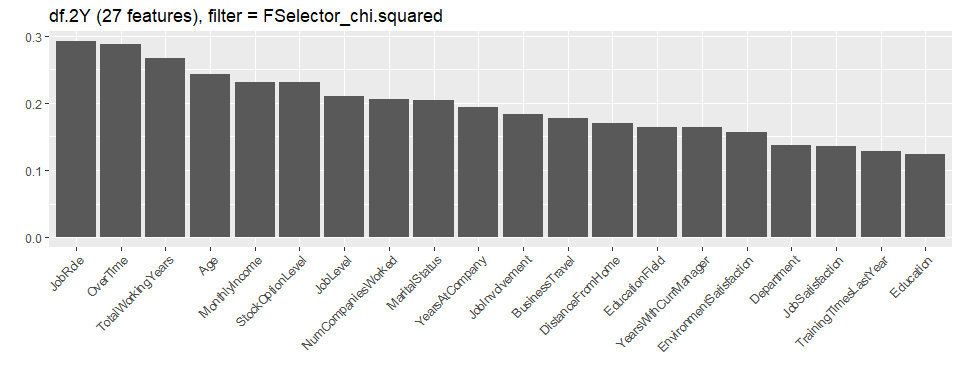
* 1. Feature Selection

(can you explain how these features were selected and confirm that we should only show the top 20?)

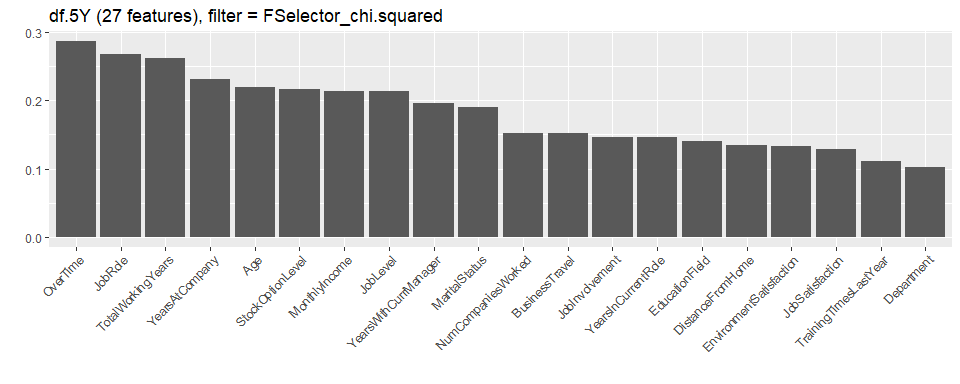
Feature selection for the entire dataset



Feature selection for employees leaving with less than 2 years of service:



Feature selection for employees leaving with less than 5 years of service:



1. **Data Mining**
   1. Association Rule Mining

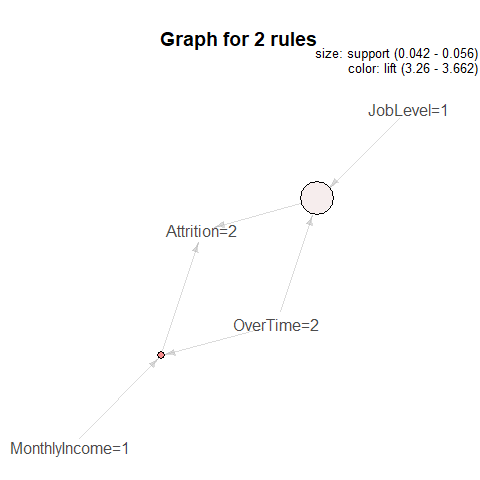
The Apriori algorithm was used to perform association rules mining on the data. Positive attrition was placed on the right hand and with support >= 0.3, and confidence >= 0.5. (why did we pick these levels for support and confidence?)

Only 2 rules were created for the entire population, but 51 rules were created for the dataset where employees left the company in less than 2 years.

(this is another example where using numeric factors vs. discretized factors akes it a little harder to understand what is going on. IF the LHS for the first rule income = 1 and Overtime = 2, there is no way of knowing whether that is a low or high income and whether they sis or did not work overtime.)

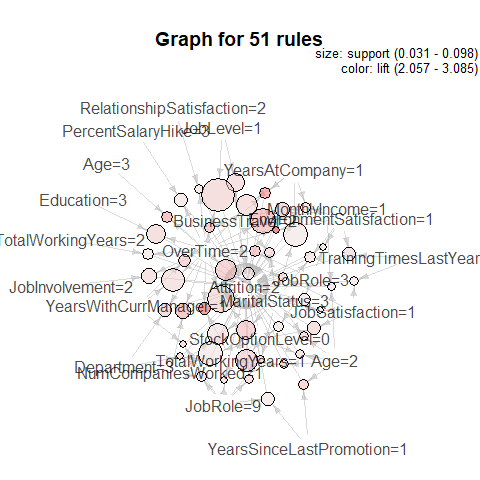
Association Rules – Entire Population

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| lhs | Rhs | support | confidence | lift | count |
| {MonthlyIncome=1,OverTime=2} | {Attrition=2} | 0.042 | 0.590 | 3.662 | 62 |
| {JobLevel=1,OverTime=2} | {Attrition=2} | 0.056 | 0.526 | 3.260 | 82 |



Association Rules – Employees leaving with less than 2 years of service:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| lhs | rhs | support | confidence |  | lift | count |
| {EnvironmentSatisfaction=1,OverTime=2} | {Attrition=2} | 0.031 | 0.750 |  | 3.085 | 18 |
| {OverTime=2,TotalWorkingYears=1} | {Attrition=2} | 0.043 | 0.714 |  | 2.938 | 25 |
| {BusinessTravel=2,YearsAtCompany=1} | {Attrition=2} | 0.041 | 0.667 |  | 2.742 | 24 |
| {Age=3,OverTime=2} | {Attrition=2} | 0.040 | 0.639 |  | 2.628 | 23 |
| {NumCompaniesWorked=1,OverTime=2} | {Attrition=2} | 0.052 | 0.625 |  | 2.571 | 30 |
| {MonthlyIncome=1,OverTime=2} | {Attrition=2} | 0.078 | 0.616 |  | 2.536 | 45 |
| {BusinessTravel=2,OverTime=2} | {Attrition=2} | 0.038 | 0.611 |  | 2.514 | 22 |
| {JobRole=3,OverTime=2} | {Attrition=2} | 0.040 | 0.605 |  | 2.490 | 23 |
| {MaritalStatus=3,OverTime=2} | {Attrition=2} | 0.066 | 0.585 |  | 2.405 | 38 |
| {MaritalStatus=3,TotalWorkingYears=1} | {Attrition=2} | 0.060 | 0.583 |  | 2.400 | 35 |
| {Age=2,YearsSinceLastPromotion=1} | {Attrition=2} | 0.038 | 0.579 |  | 2.381 | 22 |
| {Age=2,TotalWorkingYears=1} | {Attrition=2} | 0.034 | 0.571 |  | 2.351 | 20 |
| {BusinessTravel=2,YearsWithCurrManager=1} | {Attrition=2} | 0.043 | 0.568 |  | 2.337 | 25 |
| {Department=3,OverTime=2} | {Attrition=2} | 0.047 | 0.562 |  | 2.314 | 27 |
| {BusinessTravel=2,MonthlyIncome=1} | {Attrition=2} | 0.048 | 0.560 |  | 2.304 | 28 |
| {JobRole=9,MaritalStatus=3} | {Attrition=2} | 0.033 | 0.559 |  | 2.299 | 19 |
| {StockOptionLevel=0,TotalWorkingYears=1} | {Attrition=2} | 0.069 | 0.556 |  | 2.285 | 40 |
| {JobLevel=1,OverTime=2} | {Attrition=2} | 0.098 | 0.553 |  | 2.276 | 57 |
| {JobRole=9,StockOptionLevel=0} | {Attrition=2} | 0.036 | 0.553 |  | 2.273 | 21 |
| {OverTime=2,YearsAtCompany=1} | {Attrition=2} | 0.066 | 0.551 |  | 2.265 | 38 |
| {JobRole=9,YearsWithCurrManager=1} | {Attrition=2} | 0.031 | 0.545 |  | 2.244 | 18 |
| {Education=3,OverTime=2} | {Attrition=2} | 0.062 | 0.545 |  | 2.244 | 36 |
| {OverTime=2,StockOptionLevel=0} | {Attrition=2} | 0.083 | 0.545 |  | 2.244 | 48 |
| {JobSatisfaction=1,StockOptionLevel=0} | {Attrition=2} | 0.045 | 0.542 |  | 2.228 | 26 |
| {Age=2,MaritalStatus=3} | {Attrition=2} | 0.034 | 0.541 |  | 2.224 | 20 |
| {BusinessTravel=2,JobLevel=1} | {Attrition=2} | 0.059 | 0.540 |  | 2.220 | 34 |
| {JobInvolvement=2,OverTime=2} | {Attrition=2} | 0.050 | 0.537 |  | 2.209 | 29 |
| {BusinessTravel=2,MaritalStatus=3} | {Attrition=2} | 0.038 | 0.537 |  | 2.207 | 22 |
| {OverTime=2,YearsWithCurrManager=1} | {Attrition=2} | 0.072 | 0.532 |  | 2.187 | 42 |
| {JobRole=9,TotalWorkingYears=1} | {Attrition=2} | 0.031 | 0.529 |  | 2.178 | 18 |
| {Department=3,TotalWorkingYears=1} | {Attrition=2} | 0.031 | 0.529 |  | 2.178 | 18 |
| {EnvironmentSatisfaction=1,TrainingTimesLastYear=2} | {Attrition=2} | 0.047 | 0.529 |  | 2.178 | 27 |
| {Age=2,StockOptionLevel=0} | {Attrition=2} | 0.036 | 0.525 |  | 2.160 | 21 |
| {EnvironmentSatisfaction=1,YearsAtCompany=1} | {Attrition=2} | 0.038 | 0.524 |  | 2.155 | 22 |
| {BusinessTravel=2,StockOptionLevel=0} | {Attrition=2} | 0.043 | 0.521 |  | 2.142 | 25 |
| {Age=2,MonthlyIncome=1} | {Attrition=2} | 0.033 | 0.514 |  | 2.112 | 19 |
| {MaritalStatus=3,NumCompaniesWorked=1} | {Attrition=2} | 0.067 | 0.513 |  | 2.111 | 39 |
| {OverTime=2,RelationshipSatisfaction=2} | {Attrition=2} | 0.034 | 0.513 |  | 2.109 | 20 |
| {JobSatisfaction=1,OverTime=2} | {Attrition=2} | 0.034 | 0.513 |  | 2.109 | 20 |
| {JobRole=9,NumCompaniesWorked=1} | {Attrition=2} | 0.038 | 0.512 |  | 2.105 | 22 |
| {EnvironmentSatisfaction=1,MaritalStatus=3} | {Attrition=2} | 0.040 | 0.511 |  | 2.102 | 23 |
| {OverTime=2,PercentSalaryHike=3} | {Attrition=2} | 0.045 | 0.510 |  | 2.097 | 26 |
| {JobRole=9,YearsSinceLastPromotion=1} | {Attrition=2} | 0.047 | 0.500 |  | 2.057 | 27 |
| {JobSatisfaction=1,MaritalStatus=3} | {Attrition=2} | 0.031 | 0.500 |  | 2.057 | 18 |
| {JobSatisfaction=1,TrainingTimesLastYear=2} | {Attrition=2} | 0.034 | 0.500 |  | 2.057 | 20 |
| {JobRole=3,TotalWorkingYears=1} | {Attrition=2} | 0.038 | 0.500 |  | 2.057 | 22 |
| {EnvironmentSatisfaction=1,JobRole=3} | {Attrition=2} | 0.031 | 0.500 |  | 2.057 | 18 |
| {JobRole=3,MaritalStatus=3} | {Attrition=2} | 0.043 | 0.500 |  | 2.057 | 25 |
| {OverTime=2,TotalWorkingYears=2} | {Attrition=2} | 0.040 | 0.500 |  | 2.057 | 23 |
| {NumCompaniesWorked=1,StockOptionLevel=0} | {Attrition=2} | 0.076 | 0.500 |  | 2.057 | 44 |
| {MaritalStatus=3,YearsAtCompany=1} | {Attrition=2} | 0.076 | 0.500 |  | 2.057 | 44 |



Support – measures how much historical data supports the rule and it is calculated as joint probability of features: A and B when finding support for these two items together.

Confidence – measures a fraction of rows containing B or conditional probability of feature B given A.

Lift – measures ratio Confidence is to Support. Lift is expected to be greater than 1 meaning that feature A and B positively correlated.

While I don't think we can claim that there are 25 times as many things to drive an employee to leave within 2 years, but this 51 vs 2 number do mean something.

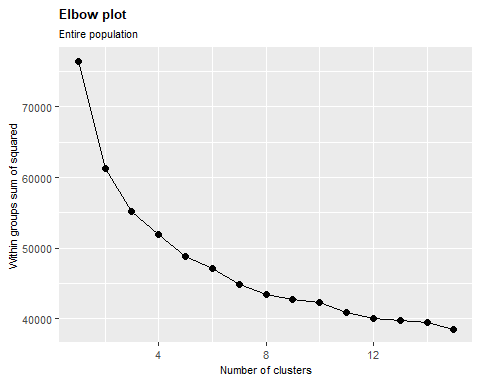
(what can we say about this?)

* 1. Clustering

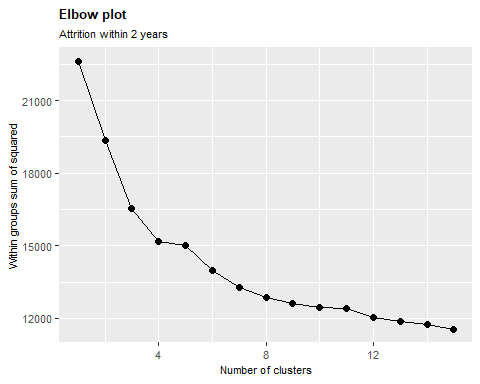
Choosing the k-value

The elbow method was used to determine the optimal number of clusters for each of the three datasets.

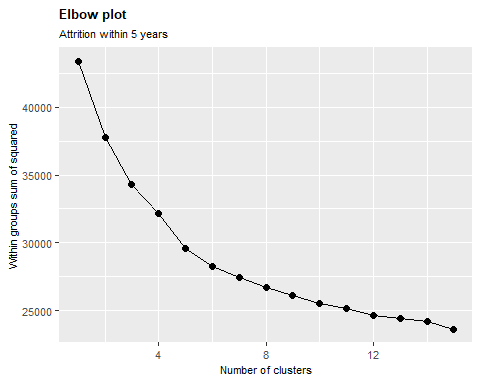
Employee attrition – Entire population



Employee attrition with less than 2 years of service



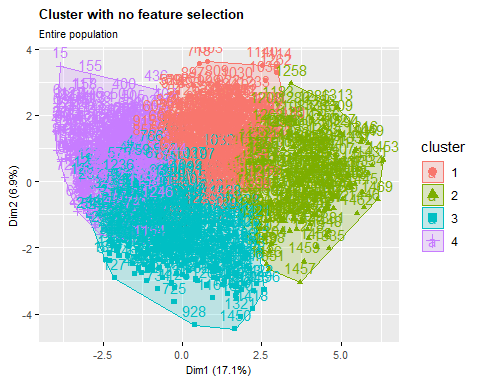
Employee attrition with less than 5 years of service



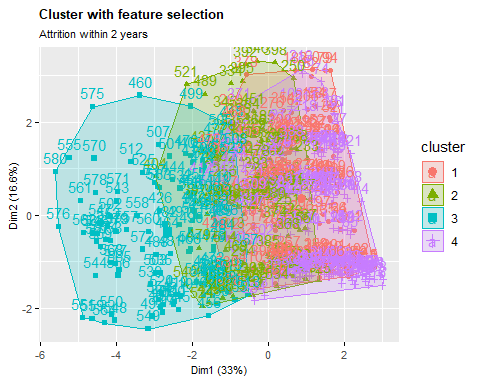
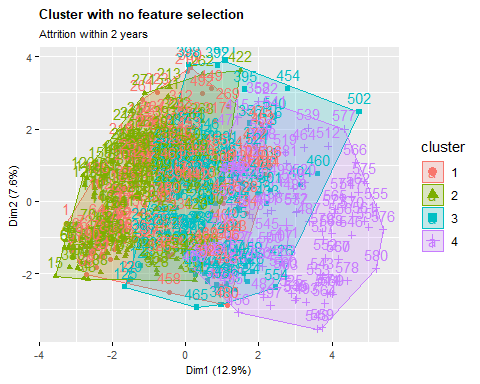
Clustering with and without feature selection

(explain the purpose of feature selection in clustering and how feature selection changes the way these graphs look)

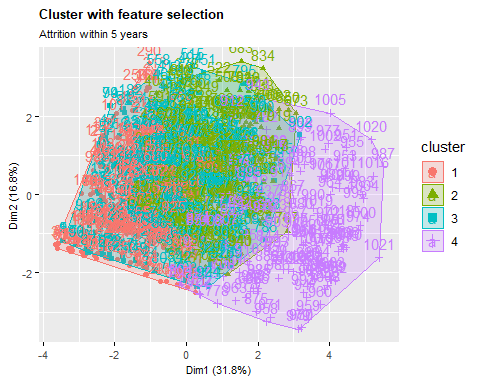
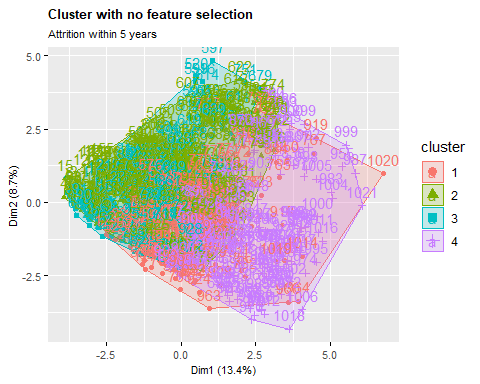
Employee attrition – Entire population



Employee attrition with less than 2 years of service



Employee attrition with less than 5 years of service



Models

The model selection process was done through nested resampling. This was done to ensure that the model is as unbiased as possible.

Model results:

(Is there any way to make the results more clear or explain what each of the 4 variables in each test result mean? What does each label mean e.g. How can we identify which models were the best?)

* 1. Naive Bayes

|  |  |  |
| --- | --- | --- |
| **All Employees** | **Employees with attrition < 2 years** | **Employees with attrition < 5 years** |
| Aggregate performance: auc.test.mean=0.7305575  mmce.test.mean=0.1890910,  acc.test.mean=0.8109090,  acc.train.mean=0.8231315 | Aggregate performance: auc.test.mean=0.7374655  mmce.test.mean=0.2481349  acc.test.mean=0.7518651  acc.train.mean=0.7909498 | Aggregate performance: auc.test.mean=0.7695777  mmce.test.mean=0.1908338  acc.test.mean=0.8091662  acc.train.mean=0.8273060 |

* 1. Rpart

|  |  |  |
| --- | --- | --- |
| **All Employees** | **Employees with attrition < 2 years** | **Employees with attrition < 5 years** |
| Aggregate performance:  auc.test.mean=0.7280833  mmce.test.mean=0.1625742  acc.test.mean=0.8374258  acc.train.mean=0.8780622 | Aggregate performance:  auc.test.mean=0.7341376  mmce.test.mean=0.2309969  acc.test.mean=0.7690031  acc.train.mean=0.8284517 | Aggregate performance:  auc.test.mean=0.7297605  mmce.test.mean=0.1917856  acc.test.mean=0.8082144  acc.train.mean=0.8701088 |

* 1. kNN

|  |  |  |
| --- | --- | --- |
| **All Employees** | **Employees with attrition < 2 years** | **Employees with attrition < 5 years** |
| Aggregate performance:  auc.test.mean=0.6866792  mmce.test.mean=0.1673454  acc.test.mean=0.8326546  acc.train.mean=0.9175124 | Aggregate performance:  auc.test.mean=0.6422862  mmce.test.mean=0.2535154  acc.test.mean=0.7464846  acc.train.mean=0.8538837 | Aggregate performance  auc.test.mean=0.7357503  mmce.test.mean=0.1829812  acc.test.mean=0.8170188  acc.train.mean=0.9173127 |

* 1. Random Forest

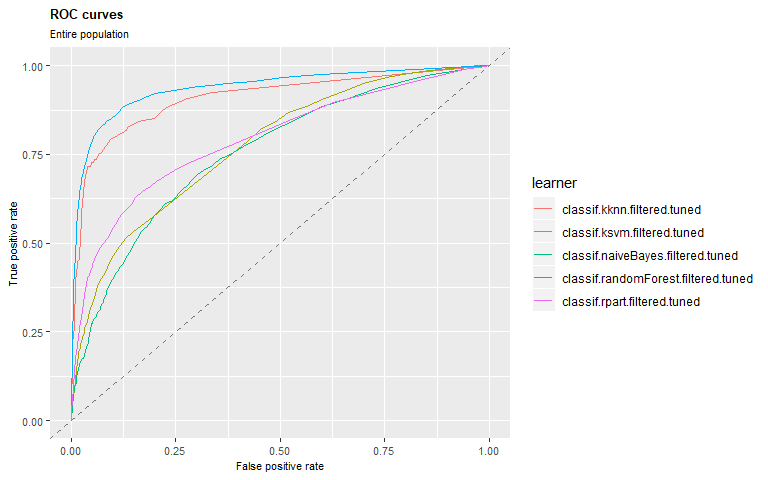
|  |  |  |
| --- | --- | --- |
| **All Employees** | **Employees with attrition < 2 years** | **Employees with attrition < 5 years** |
| Aggregate performance:auc.test.mean=0.7319680,mmce.test.mean=0.1564517,acc.test.mean=0.8435483,acc.train.mean=0.9402976 | Aggregate performance:auc.test.mean=0.7209943,mmce.test.mean=0.2258095,acc.test.mean=0.7741905,acc.train.mean=0.9004338 | Aggregate performance:auc.test.mean=0.7856206,mmce.test.mean=0.1663430,acc.test.mean=0.8336570,acc.train.mean=0.9628152 |

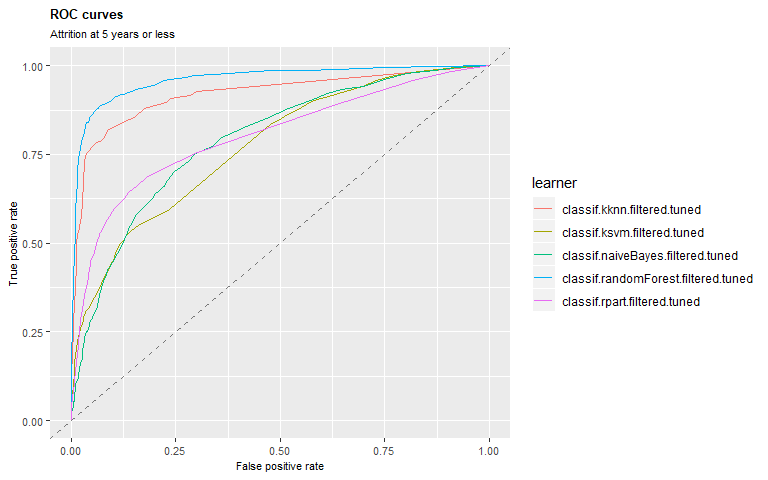
* 1. kSVM

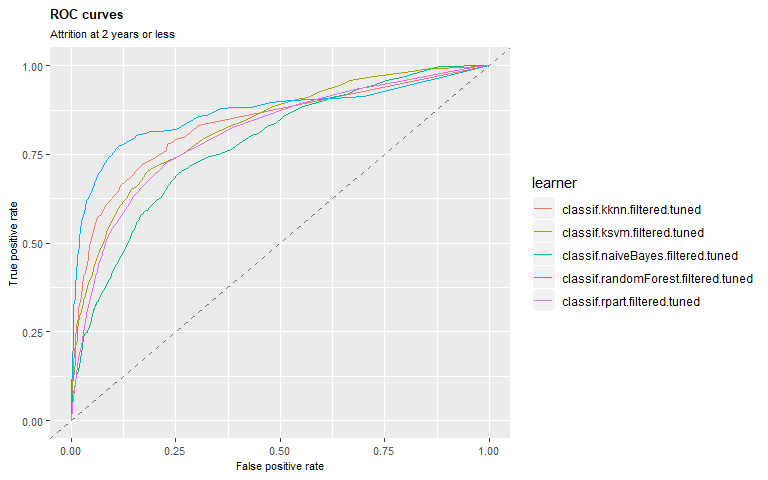
|  |  |  |
| --- | --- | --- |
| **All Employees** | **Employees with attrition < 2 years** | **Employees with attrition < 5 years** |
| Aggregate performance:  auc.test.mean=0.7720970  mmce.test.mean=0.1598601  acc.test.mean=0.8401399  acc.train.mean=0.8500023 | Aggregate performance:  auc.test.mean=0.7430675  mmce.test.mean=0.2224354  acc.test.mean=0.7775646  acc.train.mean=0.8297615 | Aggregate performance:  auc.test.mean=0.7679828  mmce.test.mean=0.1633543  acc.test.mean=0.8366457  acc.train.mean=0.8448967 |

ROC Curves

(Are these curves for the tuned on untuned models?)







1. **Results**

(which model gave the best accuracy both before and after the tuning? Looks like Random Forest worked best?)

**Tuning Strategy**

Parameter tuning and feature selection were done within the inner loop and the performance was estimated with the outer loop.

Additionally, tuning and feature selection were accomplished using a holdout test. The tuning strategy is a random search with 100 iterations. The feature selection was done as part of the tuning, allowing for 3 to 10 predictors being used for each model.

Hyperparameter tuning is the process of choosing a set of optimal hyperparameters for a learning algorithm. An actual hyperparameter is parameter whose value is set before the learning process begins. The other hyperparameters used for training each model are as follows:

(can you explain what a hyperparameter is? I don’t think we learned these in class. Is the definition I added above enough?)

(Look like we have 7 models here (3 SVMs), but the results in the next section and the ROC only reflect 5 models. Can you explain why?)

* Naïve Bayes: Laplace
* Decision Tree: complexity parameter
* kNN: k
* Linear SVM: cost
* Polynomial SVM: degree of polynomial and cost
* RBF SVM: sigma and cost
* Random Forest: number of trees

**Tuning Results**

(I’m assuming the results above in sections 3.4 – 3.7 were pre-tuned results. The tables below look like they are showing the results of the various models based on different tuning variables. Is that correct? If so, what is the best way to identify the best results for each model? Should I build another set up tables like above with the tunes results? Or are we saying that the results above are a summary of the tuned results? I want to make sure I tell the story in the right order.)

**Entire Population**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| task.id | learner.id | iter | fw.abs | laplace | auc.test.mean | mmce.test.mean | acc.test.mean | acc.train.mean | cp | k | ntree | C | kernel | sigma |
| df | classif.ksvm.filtered.tuned | 3 | 8 | NA | 0.8183677 | 0.1529745 | 0.8470255 | 0.8420413 | NA | NA | NA | 0.0002802 | vanilladot | NA |
| df | classif.ksvm.filtered.tuned | 5 | 9 | NA | 0.8093508 | 0.1497175 | 0.8502825 | 0.8625304 | NA | NA | NA | 0.9134832 | vanilladot | NA |
| df | classif.ksvm.filtered.tuned | 1 | 9 | NA | 0.8058914 | 0.1671388 | 0.8328612 | 0.8527981 | NA | NA | NA | 0.8758431 | rbfdot | 0.0106392 |
| df | classif.randomForest.filtered.tuned | 2 | 6 | NA | 0.7748637 | 0.1586402 | 0.8413598 | 0.8967193 | NA | NA | 99 | NA | NA | NA |
| df | classif.naiveBayes.filtered.tuned | 1 | 8 | 5 | 0.7739154 | 0.1529745 | 0.8470255 | 0.8394161 | NA | NA | NA | NA | NA | NA |
| df | classif.ksvm.filtered.tuned | 2 | 8 | NA | 0.7707148 | 0.1586402 | 0.8413598 | 0.8541920 | NA | NA | NA | 0.1047149 | vanilladot | NA |
| df | classif.kknn.filtered.tuned | 2 | 10 | NA | 0.7689960 | 0.1869688 | 0.8130312 | 0.9392467 | NA | 5 | NA | NA | NA | NA |
| df | classif.naiveBayes.filtered.tuned | 5 | 3 | 5 | 0.7667316 | 0.1610169 | 0.8389831 | 0.8418491 | NA | NA | NA | NA | NA | NA |
| df | classif.rpart.filtered.tuned | 4 | 5 | NA | 0.7632170 | 0.1553672 | 0.8446328 | 0.8699878 | 0.0000012 | NA | NA | NA | NA | NA |
| df | classif.rpart.filtered.tuned | 2 | 10 | NA | 0.7620614 | 0.1529745 | 0.8470255 | 0.9040097 | 0.0017395 | NA | NA | NA | NA | NA |
| df | classif.randomForest.filtered.tuned | 5 | 8 | NA | 0.7553311 | 0.1440678 | 0.8559322 | 0.9513382 | NA | NA | 251 | NA | NA | NA |
| df | classif.randomForest.filtered.tuned | 4 | 10 | NA | 0.7545041 | 0.1610169 | 0.8389831 | 0.9866343 | NA | NA | 181 | NA | NA | NA |
| df | classif.randomForest.filtered.tuned | 3 | 9 | NA | 0.7530524 | 0.1501416 | 0.8498584 | 0.9781288 | NA | NA | 169 | NA | NA | NA |
| df | classif.kknn.filtered.tuned | 1 | 5 | NA | 0.7491110 | 0.1586402 | 0.8413598 | 0.9002433 | NA | 5 | NA | NA | NA | NA |
| df | classif.ksvm.filtered.tuned | 4 | 8 | NA | 0.7485971 | 0.1610169 | 0.8389831 | 0.8383961 | NA | NA | NA | 0.3820851 | vanilladot | NA |
| df | classif.naiveBayes.filtered.tuned | 2 | 10 | 5 | 0.7468587 | 0.2152975 | 0.7847025 | 0.8165249 | NA | NA | NA | NA | NA | NA |
| df | classif.rpart.filtered.tuned | 1 | 5 | NA | 0.7460289 | 0.1614731 | 0.8385269 | 0.8734793 | 0.0008320 | NA | NA | NA | NA | NA |
| df | classif.randomForest.filtered.tuned | 1 | 8 | NA | 0.7441619 | 0.1756374 | 0.8243626 | 0.9282238 | NA | NA | 296 | NA | NA | NA |
| df | classif.rpart.filtered.tuned | 3 | 8 | NA | 0.7433321 | 0.1614731 | 0.8385269 | 0.8736330 | 0.0000001 | NA | NA | NA | NA | NA |
| df | classif.kknn.filtered.tuned | 3 | 6 | NA | 0.7406354 | 0.1529745 | 0.8470255 | 0.9088700 | NA | 5 | NA | NA | NA | NA |
| df | classif.kknn.filtered.tuned | 5 | 10 | NA | 0.7233446 | 0.1553672 | 0.8446328 | 0.9428224 | NA | 5 | NA | NA | NA | NA |
| df | classif.kknn.filtered.tuned | 4 | 8 | NA | 0.7181168 | 0.1468927 | 0.8531073 | 0.9100851 | NA | 5 | NA | NA | NA | NA |
| df | classif.rpart.filtered.tuned | 5 | 6 | NA | 0.7175852 | 0.1751412 | 0.8248588 | 0.8771290 | 0.0000000 | NA | NA | NA | NA | NA |
| df | classif.naiveBayes.filtered.tuned | 3 | 8 | 5 | 0.7117117 | 0.2011331 | 0.7988669 | 0.8347509 | NA | NA | NA | NA | NA | NA |
| df | classif.naiveBayes.filtered.tuned | 4 | 8 | 5 | 0.6708607 | 0.2457627 | 0.7542373 | 0.8262454 | NA | NA | NA | NA | NA | NA |

**Attrition less than 5 years**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| task.id | learner.id | iter | fw.abs | laplace | auc.test.mean | mmce.test.mean | acc.test.mean | acc.train.mean | cp | k | ntree | C | kernel | sigma |
| df.5Y | classif.ksvm.filtered.tuned | 1 | 5 | NA | 0.8602174 | 0.1504065 | 0.8495935 | 0.8391608 | NA | NA | NA | 0.0284500 | rbfdot | 0.0080308 |
| df.5Y | classif.ksvm.filtered.tuned | 2 | 6 | NA | 0.8411957 | 0.1666667 | 0.8333333 | 0.8444056 | NA | NA | NA | 9.5325859 | vanilladot | NA |
| df.5Y | classif.ksvm.filtered.tuned | 3 | 7 | NA | 0.8302174 | 0.1869919 | 0.8130081 | 0.8140351 | NA | NA | NA | 0.0012971 | rbfdot | 0.0005085 |
| df.5Y | classif.naiveBayes.filtered.tuned | 5 | 4 | 3 | 0.8241848 | 0.1544715 | 0.8455285 | 0.8374126 | NA | NA | NA | NA | NA | NA |
| df.5Y | classif.ksvm.filtered.tuned | 4 | 10 | NA | 0.8170652 | 0.1626016 | 0.8373984 | 0.8426573 | NA | NA | NA | 1098.5668930 | vanilladot | NA |
| df.5Y | classif.naiveBayes.filtered.tuned | 2 | 3 | 2 | 0.8149457 | 0.1626016 | 0.8373984 | 0.8251748 | NA | NA | NA | NA | NA | NA |
| df.5Y | classif.randomForest.filtered.tuned | 5 | 10 | NA | 0.8129891 | 0.1707317 | 0.8292683 | 0.9877622 | NA | NA | 135 | NA | NA | NA |
| df.5Y | classif.ksvm.filtered.tuned | 5 | 10 | NA | 0.8090761 | 0.1707317 | 0.8292683 | 0.8776224 | NA | NA | NA | 0.0081901 | vanilladot | NA |
| df.5Y | classif.randomForest.filtered.tuned | 3 | 9 | NA | 0.8018478 | 0.1504065 | 0.8495935 | 0.9771930 | NA | NA | 241 | NA | NA | NA |
| df.5Y | classif.naiveBayes.filtered.tuned | 3 | 4 | 0 | 0.8003804 | 0.1666667 | 0.8333333 | 0.8473684 | NA | NA | NA | NA | NA | NA |
| df.5Y | classif.rpart.filtered.tuned | 1 | 10 | NA | 0.7975543 | 0.1707317 | 0.8292683 | 0.8706294 | 0.0000026 | NA | NA | NA | NA | NA |
| df.5Y | classif.rpart.filtered.tuned | 2 | 5 | NA | 0.7906522 | 0.1626016 | 0.8373984 | 0.8583916 | 0.0000000 | NA | NA | NA | NA | NA |
| df.5Y | classif.randomForest.filtered.tuned | 1 | 8 | NA | 0.7857065 | 0.1829268 | 0.8170732 | 0.9195804 | NA | NA | 51 | NA | NA | NA |
| df.5Y | classif.rpart.filtered.tuned | 5 | 5 | NA | 0.7777717 | 0.1707317 | 0.8292683 | 0.8601399 | 0.0000000 | NA | NA | NA | NA | NA |
| df.5Y | classif.naiveBayes.filtered.tuned | 4 | 6 | 0 | 0.7757609 | 0.2195122 | 0.7804878 | 0.8129371 | NA | NA | NA | NA | NA | NA |
| df.5Y | classif.kknn.filtered.tuned | 2 | 9 | NA | 0.7757609 | 0.1504065 | 0.8495935 | 0.9143357 | NA | 5 | NA | NA | NA | NA |
| df.5Y | classif.randomForest.filtered.tuned | 4 | 10 | NA | 0.7675543 | 0.1829268 | 0.8170732 | 0.9842657 | NA | NA | 410 | NA | NA | NA |
| df.5Y | classif.randomForest.filtered.tuned | 2 | 9 | NA | 0.7581522 | 0.1707317 | 0.8292683 | 0.9772727 | NA | NA | 411 | NA | NA | NA |
| df.5Y | classif.kknn.filtered.tuned | 4 | 10 | NA | 0.7536413 | 0.1910569 | 0.8089431 | 0.9353147 | NA | 5 | NA | NA | NA | NA |
| df.5Y | classif.kknn.filtered.tuned | 5 | 9 | NA | 0.7430435 | 0.1788618 | 0.8211382 | 0.9125874 | NA | 5 | NA | NA | NA | NA |
| df.5Y | classif.kknn.filtered.tuned | 1 | 9 | NA | 0.7423370 | 0.1788618 | 0.8211382 | 0.9178322 | NA | 5 | NA | NA | NA | NA |
| df.5Y | classif.naiveBayes.filtered.tuned | 1 | 3 | 0 | 0.7356522 | 0.1747967 | 0.8252033 | 0.8339161 | NA | NA | NA | NA | NA | NA |
| df.5Y | classif.kknn.filtered.tuned | 3 | 7 | NA | 0.7316848 | 0.2154472 | 0.7845528 | 0.9596491 | NA | 4 | NA | NA | NA | NA |
| df.5Y | classif.rpart.filtered.tuned | 4 | 8 | NA | 0.6799457 | 0.1991870 | 0.8008130 | 0.8811189 | 0.0050928 | NA | NA | NA | NA | NA |
| df.5Y | classif.rpart.filtered.tuned | 3 | 10 | NA | 0.6554348 | 0.2073171 | 0.7926829 | 0.8807018 | 0.0000028 | NA | NA | NA | NA | NA |

**Attrition less than 2 years**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| task.id | learner.id | iter | fw.abs | laplace | auc.test.mean | mmce.test.mean | acc.test.mean | acc.train.mean | cp | k | ntree | C | kernel | degree |
| df.2Y | classif.randomForest.filtered.tuned | 5 | 3 | NA | 0.8485017 | 0.1928571 | 0.8071429 | 0.7962963 | NA | NA | 19 | NA | NA | NA |
| df.2Y | classif.naiveBayes.filtered.tuned | 5 | 3 | 4 | 0.8319922 | 0.2071429 | 0.7928571 | 0.7777778 | NA | NA | NA | NA | NA | NA |
| df.2Y | classif.randomForest.filtered.tuned | 2 | 3 | NA | 0.8308824 | 0.2357143 | 0.7642857 | 0.7932099 | NA | NA | 406 | NA | NA | NA |
| df.2Y | classif.ksvm.filtered.tuned | 2 | 10 | NA | 0.8188124 | 0.2071429 | 0.7928571 | 0.7746914 | NA | NA | NA | 799.9874592 | vanilladot | NA |
| df.2Y | classif.randomForest.filtered.tuned | 1 | 10 | NA | 0.8043840 | 0.2142857 | 0.7857143 | 0.9938272 | NA | NA | 222 | NA | NA | NA |
| df.2Y | classif.naiveBayes.filtered.tuned | 4 | 10 | 2 | 0.8029967 | 0.1714286 | 0.8285714 | 0.8049536 | NA | NA | NA | NA | NA | NA |
| df.2Y | classif.ksvm.filtered.tuned | 3 | 3 | NA | 0.7913430 | 0.1928571 | 0.8071429 | 0.7661538 | NA | NA | NA | 0.0025128 | vanilladot | NA |
| df.2Y | classif.naiveBayes.filtered.tuned | 1 | 10 | 1 | 0.7816315 | 0.2000000 | 0.8000000 | 0.8148148 | NA | NA | NA | NA | NA | NA |
| df.2Y | classif.ksvm.filtered.tuned | 5 | 9 | NA | 0.7759434 | 0.2357143 | 0.7642857 | 0.8611111 | NA | NA | NA | 7.4940983 | vanilladot | NA |
| df.2Y | classif.rpart.filtered.tuned | 2 | 5 | NA | 0.7608213 | 0.2285714 | 0.7714286 | 0.8271605 | 0.0005499 | NA | NA | NA | NA | NA |
| df.2Y | classif.ksvm.filtered.tuned | 4 | 8 | NA | 0.7595727 | 0.2000000 | 0.8000000 | 0.8142415 | NA | NA | NA | 6.5191791 | vanilladot | NA |
| df.2Y | classif.naiveBayes.filtered.tuned | 3 | 3 | 0 | 0.7583241 | 0.2214286 | 0.7785714 | 0.7969231 | NA | NA | NA | NA | NA | NA |
| df.2Y | classif.randomForest.filtered.tuned | 3 | 10 | NA | 0.7520810 | 0.2642857 | 0.7357143 | 0.9938462 | NA | NA | 194 | NA | NA | NA |
| df.2Y | classif.kknn.filtered.tuned | 3 | 5 | NA | 0.7437569 | 0.2285714 | 0.7714286 | 0.8430769 | NA | 5 | NA | NA | NA | NA |
| df.2Y | classif.ksvm.filtered.tuned | 1 | 6 | NA | 0.7412597 | 0.2357143 | 0.7642857 | 0.8271605 | NA | NA | NA | 0.0023492 | polydot | 2 |
| df.2Y | classif.naiveBayes.filtered.tuned | 2 | 6 | 2 | 0.7258602 | 0.2285714 | 0.7714286 | 0.7685185 | NA | NA | NA | NA | NA | NA |
| df.2Y | classif.kknn.filtered.tuned | 5 | 5 | NA | 0.7132353 | 0.2357143 | 0.7642857 | 0.9043210 | NA | 4 | NA | NA | NA | NA |
| df.2Y | classif.kknn.filtered.tuned | 4 | 7 | NA | 0.7047725 | 0.2571429 | 0.7428571 | 0.8699690 | NA | 5 | NA | NA | NA | NA |
| df.2Y | classif.rpart.filtered.tuned | 1 | 3 | NA | 0.6945061 | 0.2571429 | 0.7428571 | 0.8117284 | 0.0028463 | NA | NA | NA | NA | NA |
| df.2Y | classif.rpart.filtered.tuned | 3 | 5 | NA | 0.6839623 | 0.2571429 | 0.7428571 | 0.8184615 | 0.0016646 | NA | NA | NA | NA | NA |
| df.2Y | classif.randomForest.filtered.tuned | 4 | 10 | NA | 0.6839623 | 0.2142857 | 0.7857143 | 0.9814241 | NA | NA | 349 | NA | NA | NA |
| df.2Y | classif.rpart.filtered.tuned | 5 | 5 | NA | 0.6800777 | 0.2214286 | 0.7785714 | 0.8302469 | 0.0045201 | NA | NA | NA | NA | NA |
| df.2Y | classif.kknn.filtered.tuned | 2 | 5 | NA | 0.6684240 | 0.2714286 | 0.7285714 | 0.9259259 | NA | 4 | NA | NA | NA | NA |
| df.2Y | classif.kknn.filtered.tuned | 1 | 3 | NA | 0.6680078 | 0.2642857 | 0.7357143 | 0.8117284 | NA | 5 | NA | NA | NA | NA |
| df.2Y | classif.rpart.filtered.tuned | 4 | 8 | NA | 0.6528857 | 0.3357143 | 0.6642857 | 0.8452012 | 0.0101160 | NA | NA | NA | NA | NA |

1. **Conclusion (Andrew)**

(we need the code for the graphs that we had on pages 10-11 of the presentation that showed the accuracy and top drivers for the 3 populations

Indicate which model was the most accurate.

Can we add any thoughts behind why we believe that model was the most accurate?)

Andrew to enter the language for the recommendations we are making