Course: ALY6020 Name: Pragati Koladiya
Term: A_Fall_2020 NUID: 001029445

Assignment 4

Introduction:

The Naive Bayes Classifier is a well-known machine learning classifier which has many applications in Natural Language Processing (NLP) and other areas. Despite its simplicity, it is able to achieve above average performance in different tasks like sentiment analysis.

The objectives of this assignment are as follows,

- To perform data visualization techniques to understand the insight of the data.
- This led us to implement the naive base model for the chosen dataset.
- Apply various R tools to get a visual understanding of the data
- Clean it to make it ready to apply machine learning model thus predicting the outcome.

Problem Statement:

Consider the campus students who have taken part in campus requirements. The data included secondary and higher secondary school percentage. It also includes degree specialization type, work experience and salary offers to the placed students.

Essentially, the campus wants,

- To identify the variables affecting placement status,
- Show some student placement information details with Exploratory Data Analysis(EDA)
- Implement the naive base model to predict the student placement status.
- The goal of predictive analysis is to avoid overfitting and find the model that gives the highest accuracy

Data Acquisition :

The dataset is chosen from kaggle with file name “Placement.CSV”. This dataset contains of student record of placement.

- Link: https://www.kaggle.com/benroshan/factors-affecting-campus-placement

Data Information:

The data is about student placement.

- The data contains 215 observations (student placement sample) and 15 attributes related to the placement.

Data Dictionary: Campus Recruitment

1. **sl_no** = Serial Number
2. **gender** = Gender: Male='M', Female='F'
3. **ssc_p** = Secondary Education percentage- 10th Grade
4. **ssc_b** = Board of Education- Central/ Others
5. **hsc_p** = Higher Secondary Education percentage- 12th Grade
6. **hsc_b** = Board of Education- Central/ Others
7. **hsc_s** = Specialization in Higher Secondary Education
8. **degree_p** = Degree Percentage
9. **degree_t** = Under Graduation(Degree type)- Field of degree education

10. **workex** = Work Experience
11. **etest_p** = Employability test percentage (conducted by college)
12. **specialisation** = Post Graduation(MBA)- Specialization
13. **mba_p** = MBA percentage
14. **status** = Status of placement- Placed/Not placed
15. **salary** = Salary offered by corporate to candidates

Data Overview:

Required packages have been imported before loading the dataset into R.

- Head of the dataset with columns and first few rows. I have removed the space from the variables labels for simplicity.

```
> head(placement)
# A tibble: 6 x 15
  sl_no gender ssc_p ssc_b hsc_p hsc_b hsc_s degree_p degree_t workex etest_p
  <dbl> <chr>  <dbl> <chr> <dbl> <chr> <chr>    <dbl> <chr>    <chr>    <dbl>
1     1 M       67   Othe…  91   Othe… Comm…     58   Sci&Tech No        55
2     2 M       79.3 Cent…  78.3 Othe… Scie…     77.5 Sci&Tech Yes       86.5
3     3 M       65   Cent…  68   Cent… Arts      64   Comm&Mg… No        75
4     4 M       56   Cent…  52   Cent… Scie…     52   Sci&Tech No        66
5     5 M       85.8 Cent…  73.6 Cent… Comm…     73.3 Comm&Mg… No        96.8
6     6 M       55   Othe…  49.8 Othe… Scie…     67.2 Sci&Tech Yes       55
# … with 4 more variables: specialisation <chr>, mba_p <dbl>, status <chr>,
#   salary <dbl>
```

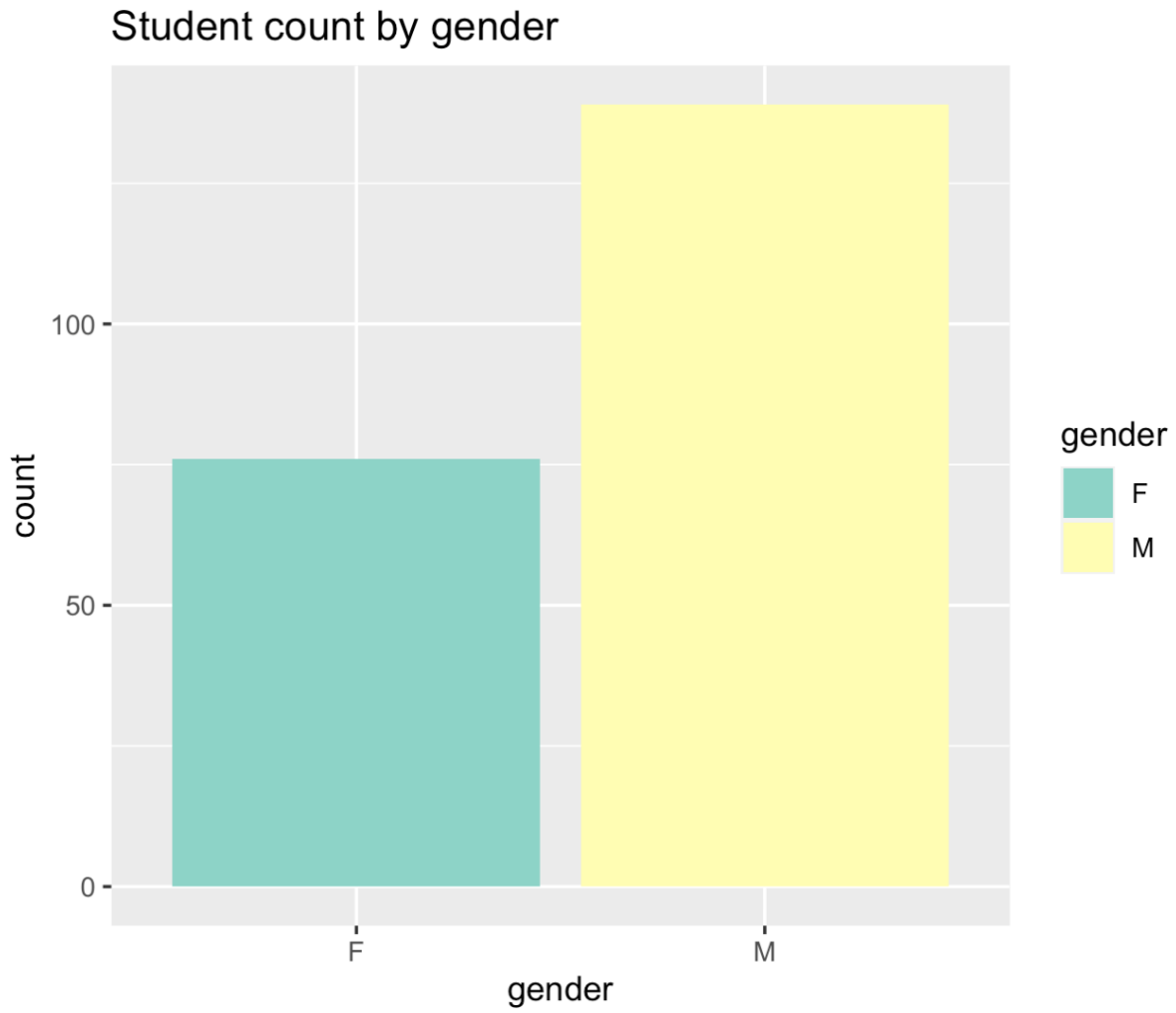
Course: ALY6020
Term: A_Fall_2020

Name: Pragati Koladiya
NUID: 001029445

Exploratory Data Analysis(EDA):

The most important step is to understand the data and identify if there is some obvious multicollinearity present. Here's where we will also identify if predictors have a strong association with the outcome variable.

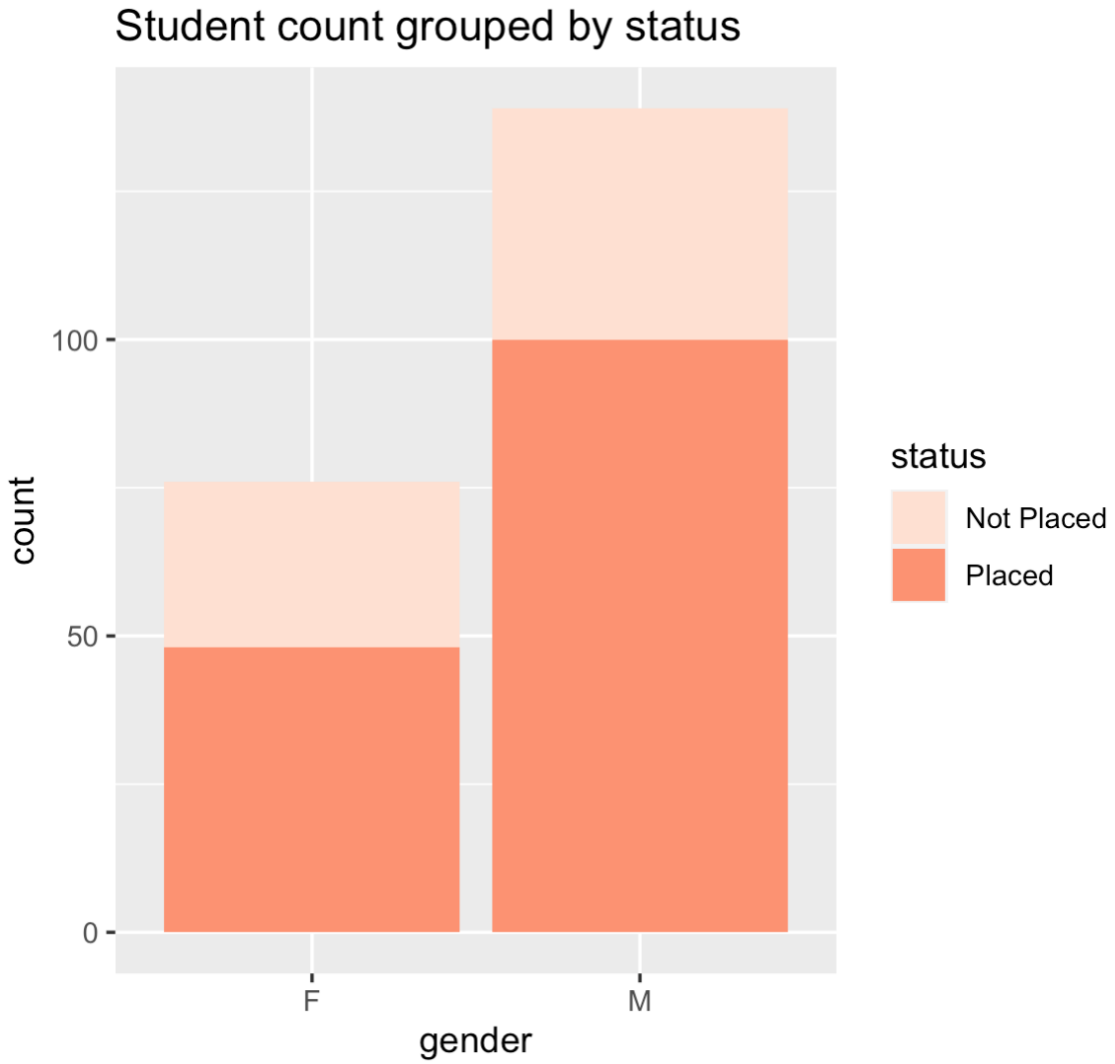
1) Student count by gender



- The above graph shows the gender distribution of students for a given campus.
- Majority students are male count is approximately 175.
- The female student count is approximately 75.

Name: Pragati Koladiya
NUID: 001029445

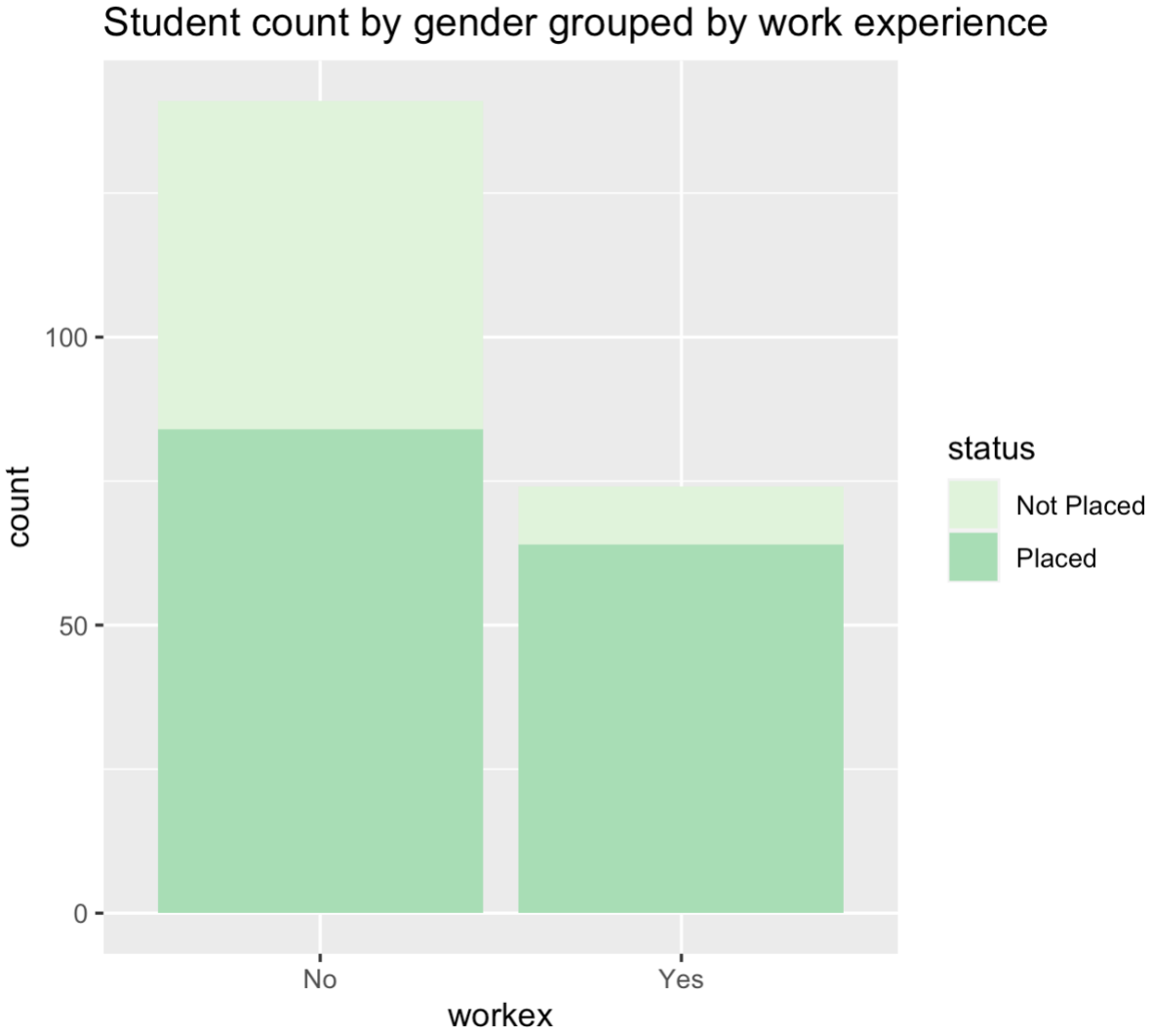
2) Student count grouped by status



- Out of 75 female students approximately 50 female students were placed.
- Out of 175 male students 100 male students were placed.
- Over all, placement rate is much higher among female students.

Name: Pragati Koladiya
NUID: 001029445

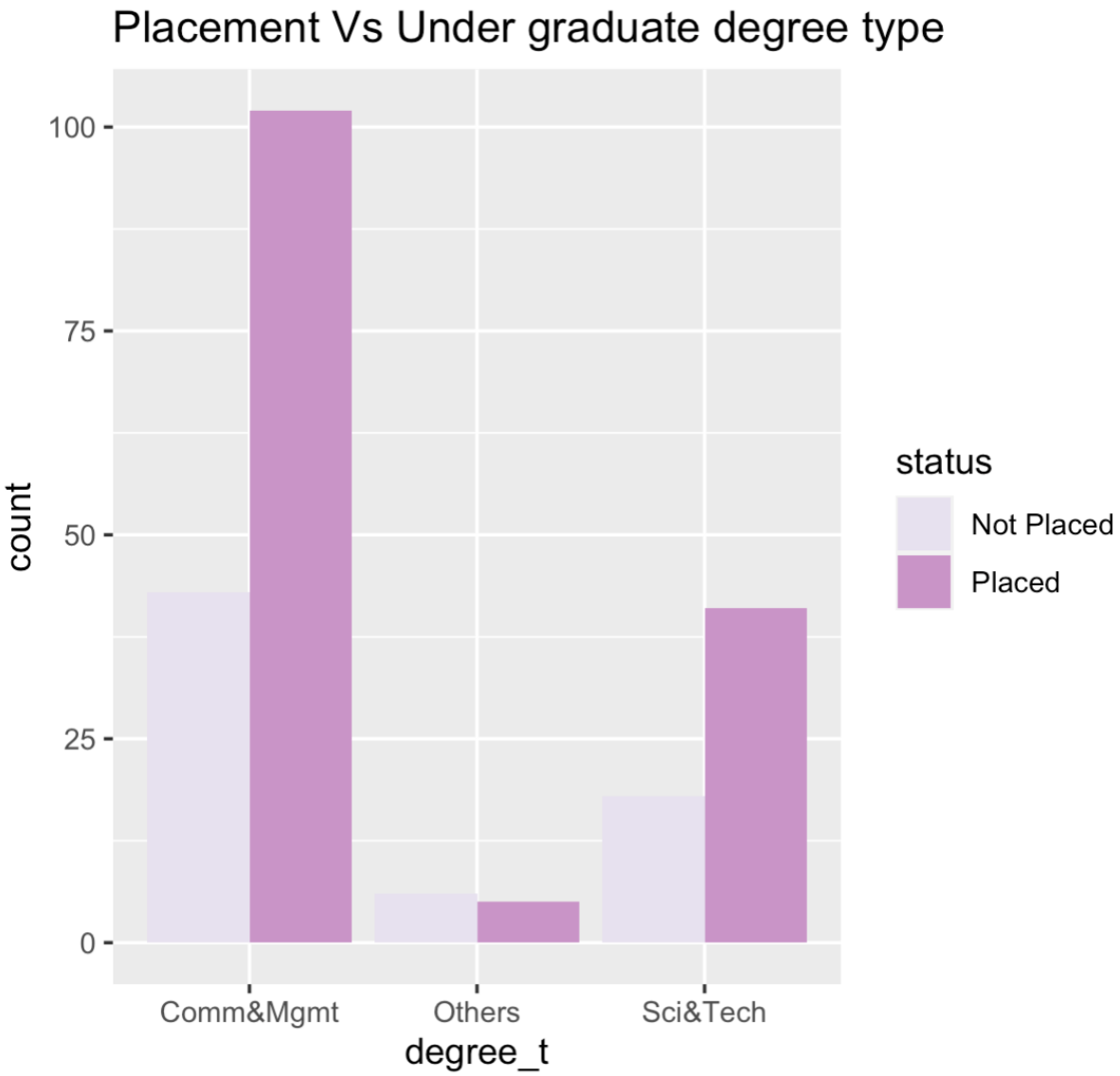
3) Student Vs Work experience



- Majority of the students do not have work experience.
- 75 students have work experience out of those 65 got placed.
- More number of students having work experience got placed compare to students with no work experience.
- The placement rate is approximately 50% among freshers.

Name: Pragati Koladiya
NUID: 001029445

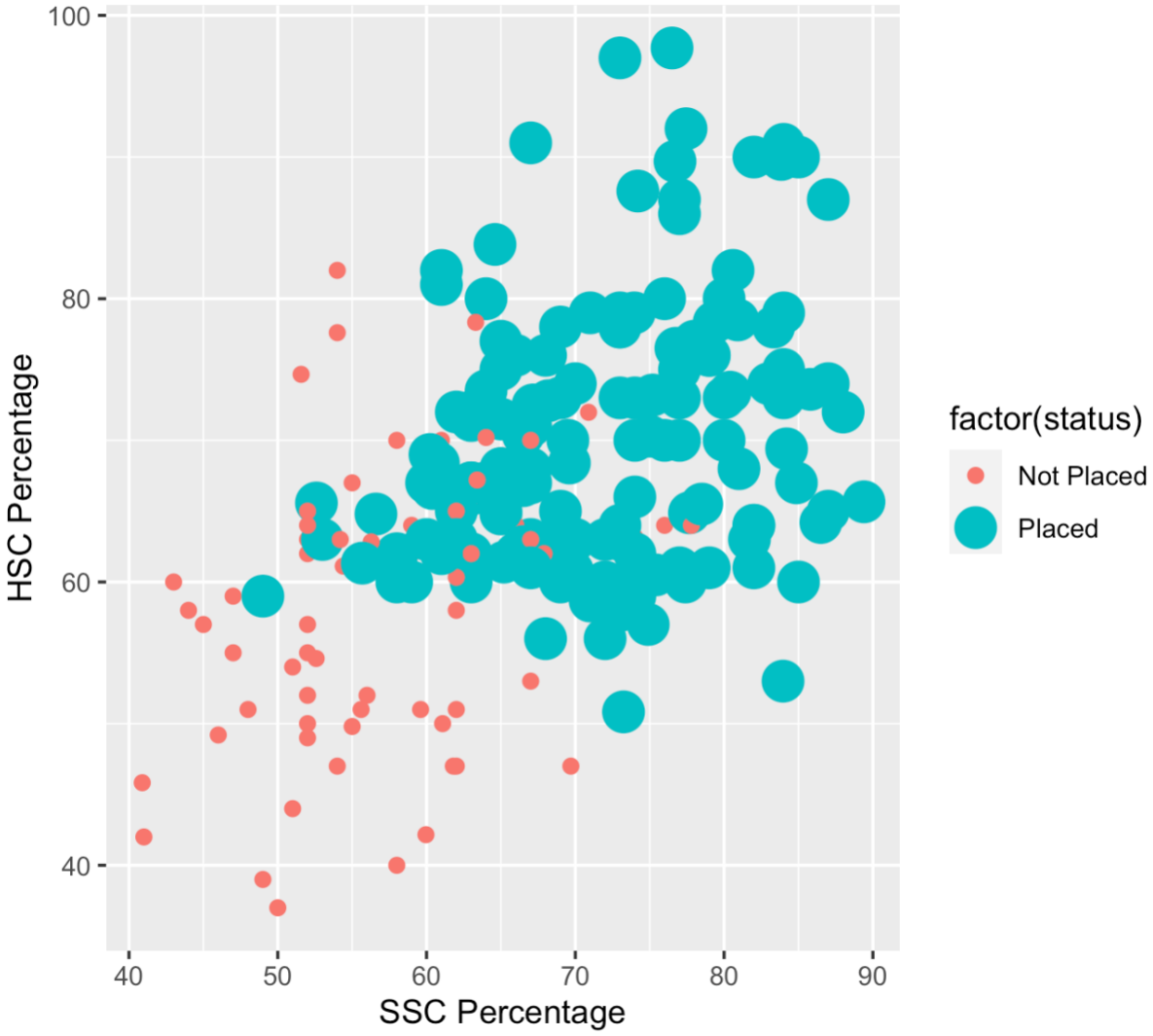
4) Students placement status by degree type



- The above graph shows the students with degree from Commerce and management has the highest placement rate.
- Students from technologies have the lowest placement rate for particular campus.

Name: Pragati Koladiya
NUID: 001029445

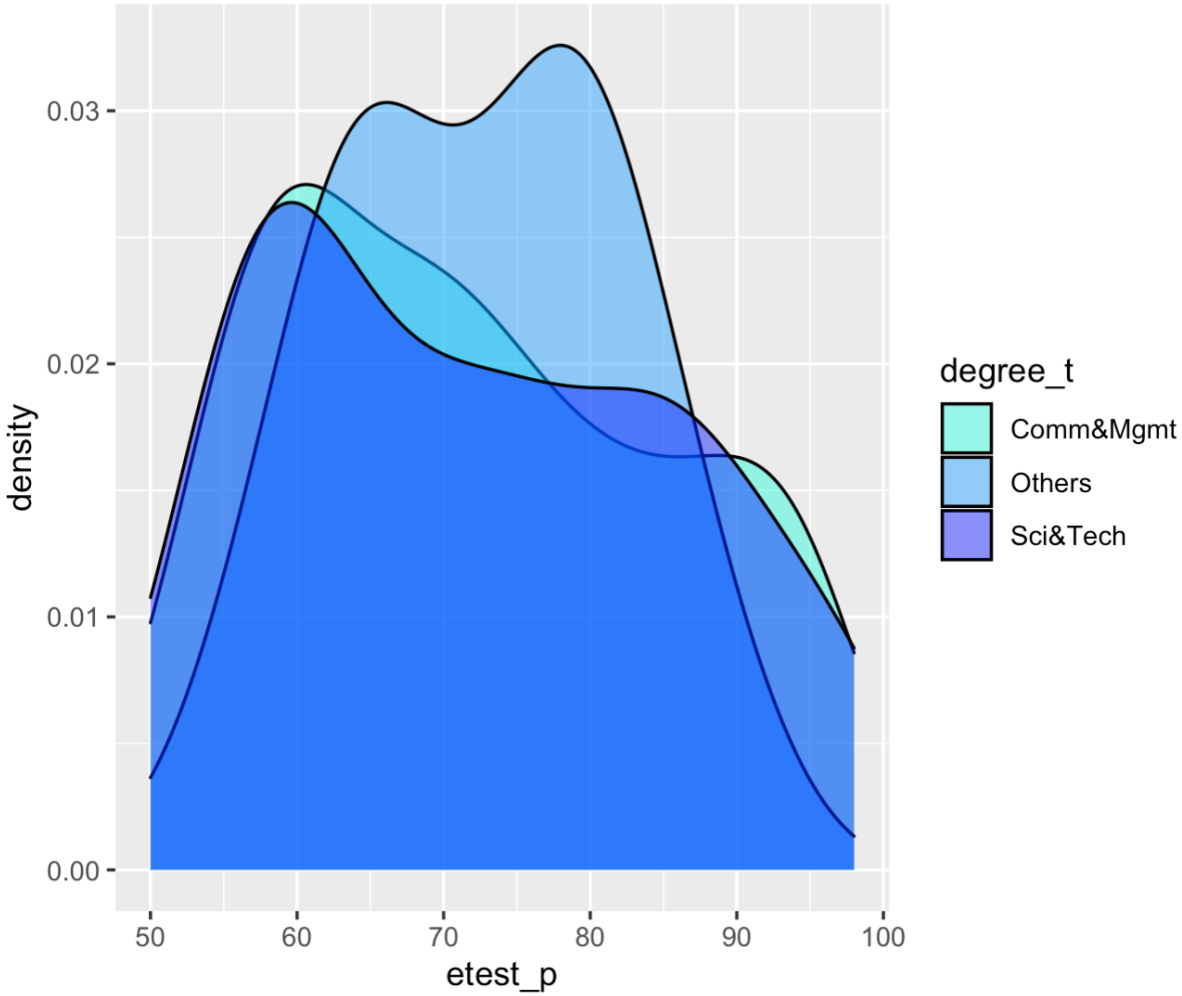
5) SSC Vs HSC Percentage by status



- Students who are not pleased has the HSC - Higher Sedentary Certificate and SSC - Secondary School Certificate percentage are very less pointed with carrot red colour.
- Most of the students who are placed have good SSC and HSC percentage.

Name: Pragati Koladiya
NUID: 001029445

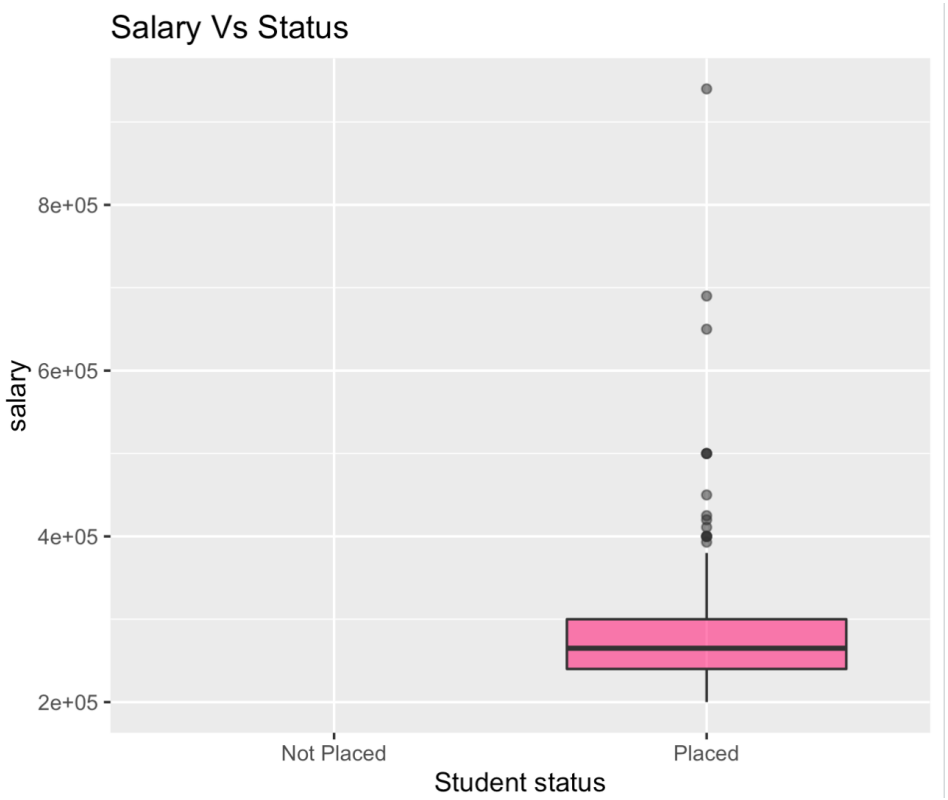
6) E-Test percentage distribution by density plot



- The above density plot shows the distribution of etest_p - e test percentage by degree type.
- The students who have performed well have other kinds of degree type and the distribution is equal.

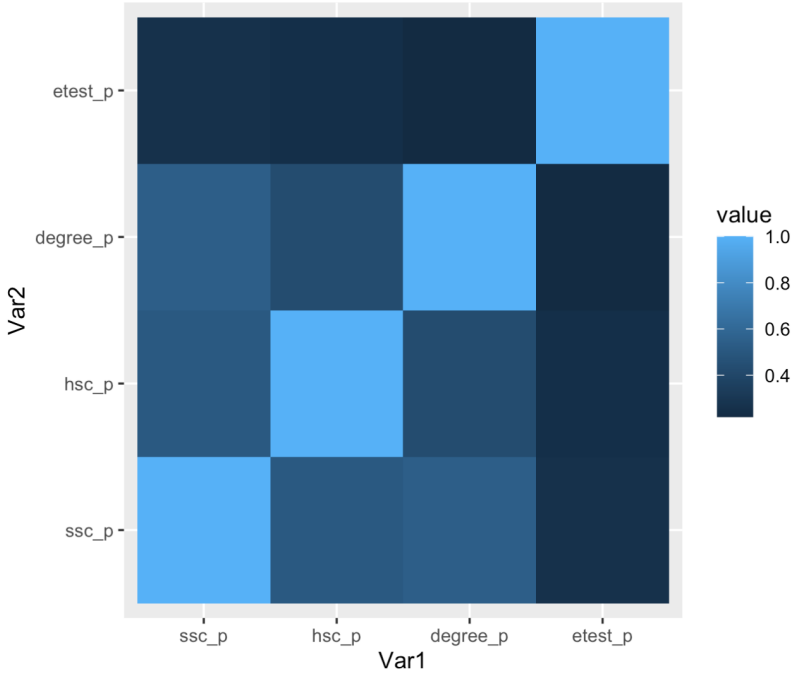
Name: Pragati Koladiya
NUID: 001029445

7) Salary Vs Status using box plot



- The students who are placed has salary between 2k to 3k.

8) Heat map to show correlation between numerical data



- All the features which has numeric values are shown in the above heat map. Higher the correlation lighter the blue shade will be.
- As we can observe all the features has positive correlation between each variable.

Name: Pragati Koladiya
NUID: 001029445

Data Preparation:

The first step is to check for null values in the dataset.

```
> placement[placement==""] <- NA
> sapply(placement,function(x) sum(is.na(x)))
         sl_no         gender          ssc_p          ssc_b 
            0              0              0              0 
        hsc_p          hsc_b          hsc_s       degree_p 
            0              0              0              0 
     degree_t         workex       etest_p specialisation 
            0              0              0              0 
      mba_p          status         salary 
            0              0             67 
```

- We see that the sum of null values is 0 for all the columns except salary.

```
#Dropping Salary column
placement =placement[,!names(placement) %in% 'salary']
```

- We do not require salary column as the aim of the model is to predict whether the student will be placed or not!

```
placement$gender <- factor(placement$gender, levels=c('M','F'),labels=c(0,1))
table(placement$gender)
typeof(placement$gender)
```

- Converting categorical columns into numerical columns.
- There are eight different columns which are converted to numerical columns which includes, gender, ssc_b, hsc_b, hsc_s, degree_t, workex, specialisation, and status

After converting dataset into numerical

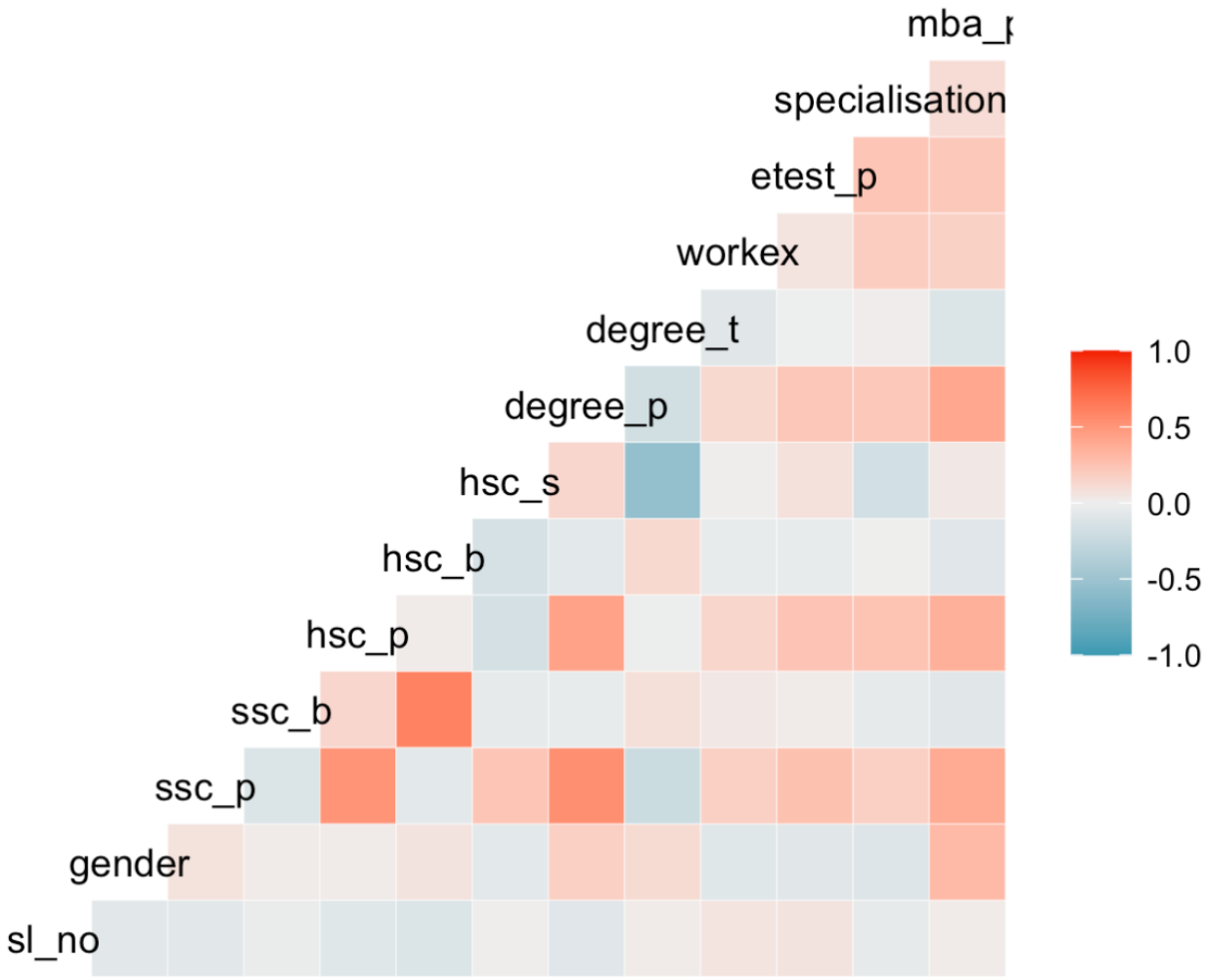
```
> head(placement_Num_Data)
  sl_no gender ssc_p ssc_b hsc_p hsc_b hsc_s degree_p degree_t
1     1      0 67.00     0 91.00     0     1    58.00        0
2     2      0 79.33     1 78.33     0     2    77.48        0
3     3      0 65.00     1 68.00     1     0    64.00        1
4     4      0 56.00     1 52.00     1     2    52.00        0
5     5      0 85.80     1 73.60     1     1    73.30        1
6     6      0 55.00     0 49.80     0     2    67.25        0
  workex etest_p specialisation mba_p status
1      0    55.0              0 58.80      0
2      1    86.5              1 66.28      0
3      0    75.0              1 57.80      0
4      0    66.0              0 59.43      1
5      0    96.8              1 55.50      0
6      1    55.0              1 51.58      1
```

- All the columns are being converted to numerical.

Before moving further let's consider the correlation plot which allows highlighting most (positively or negatively) correlated.

As we are going to predict the student placement status knowing the correlation between each variable is very important. This will help the model to predict the price with higher accuracy.

Correlation plot:



Darker the grid higher the correlation.

- The positive correlation has been observed between –
  ssc_p, ssc_b and status

The negative correlation is in a lighter shade of blue.

- Few of them are,
  sl_no, hsc_b, degree_t, and status

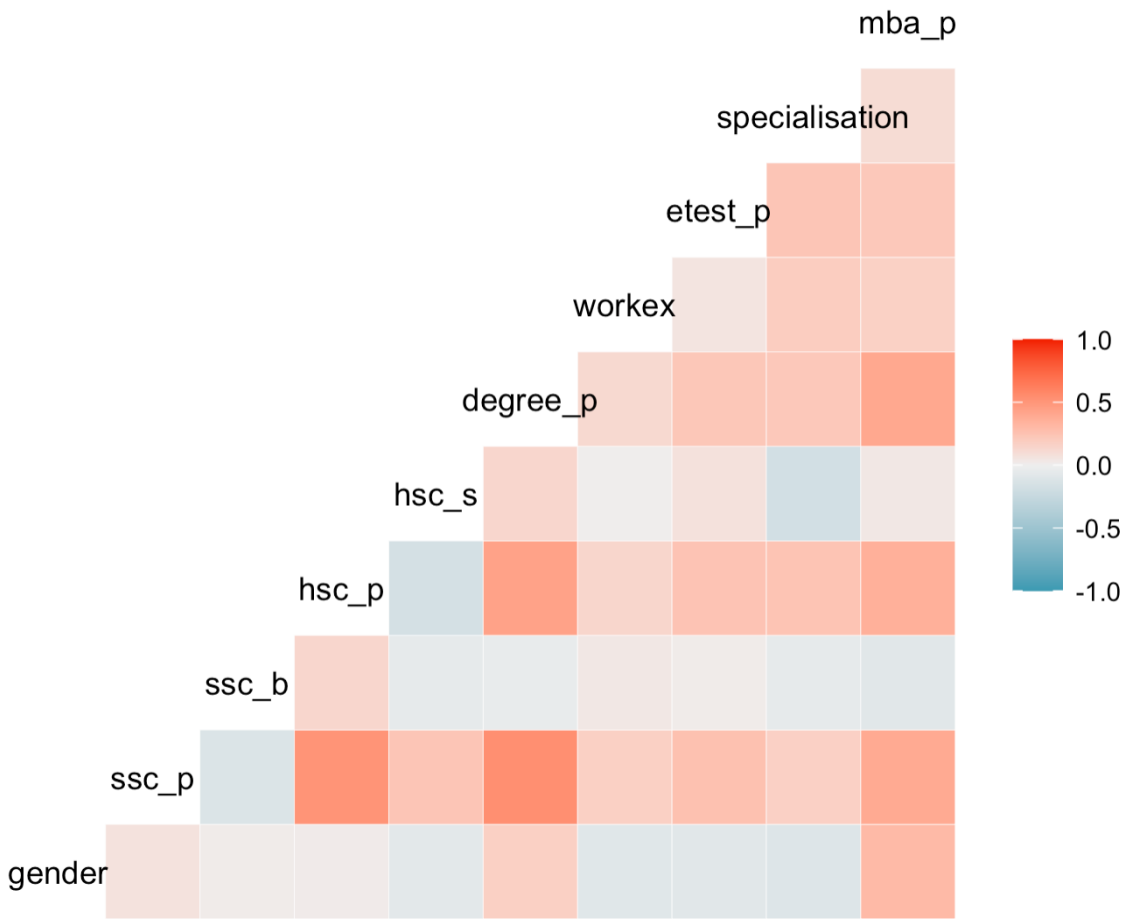
Dropping columns which has lower correlation with status.

```
> placement_Num_Data =placement_Num_Data[,!names(placement_Num_Data) %in% 'sl_no']
> placement_Num_Data =placement_Num_Data[,!names(placement_Num_Data) %in% 'hsc_b']
> placement_Num_Data =placement_Num_Data[,!names(placement_Num_Data) %in% 'degree_t']
```

- Column name 'sl_no', 'hsc_b' and 'degree_t' have been dropped.

Name: Pragati Koladiya
NUID: 001029445

Let's observe the correlation between each variable after dropping few columns



- It is apparent that most of the columns have positive correlation with dependent variable status.

Implementing Naive Bayes Model:

Naive Bayes classifiers work on the principle of conditional probability as given by the base theorem.

Naive Bayes come under the category of supervised machine learning classification algorithms. It can be used for facial recognition, weather prediction, a medical diagnosis like cancer patient is at high rise or not, news classification like identifying political or weather news, and many more.

Formula: `naiveBayes(formula, data, laplace = 0, ..., subset, na.action = na.pass)`

Split data into train and test set:

The first step in the model implementation is to split the dataset to eliminate the bias to training data in machine learning algorithms.

```
# Splitting the data set into the Training set and Test set
set.seed(123)
split = sample.split(placement_Num_Data$status, SplitRatio = 0.75)
training_set = subset(placement_Num_Data, split == TRUE)
test_set = subset(placement_Num_Data, split == FALSE)
> dim(training_set)
[1] 161  11
> dim(test_set)
[1] 54 11
```

Train set has 161 x 11 and the test set has 54 x 11. The data is split into a 75:25 ratio.

The caTools package provides a method sample.split() for partitioning our data into train and test sets. We are passing 2 parameters. The "y" parameter takes the value of the variable according to which data needs to be partitioned. In our case, the target variable is quality, so we are passing *placement_Num_Data$status* and the *SplitRatio* is .75.

```
classifier = naiveBayes(status ~ . ,data = training_set)
classifier
```

- In this model we are going to use all the columns present in data frame name '*placement_Num_Data'*.
- To implement the model naiveBayes() function is from *library(e1071)* and caTool package has been used.

```
# Predicting the Test set results
prediction <-predict(classifier, test_set)
prediction
```

Here we will predict the student placement status using test data. The predict() function will helps to predict it.

```
> prediction
 [1] 0 0 0 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 1 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 1 0 1 0 0 1 0 0 1
[46] 0 0 1 0 1 0 0 1 0
Levels: 0 1
```

The predicted values are in two categories 0 and 1 where 0 indicates student is not placed and 1 indicates student is placed.

After predicting the student placement status lets view the confusion matrix with accuracy so that we can decide whether we will trust the model or not.

```
# Making the Confusion Matrix
ConfusionMatrix = table(test_set[,11], prediction)
ConfusionMatrix
```

Name: Pragati Koladiya
NUID: 001029445

```
> ConfusionMatrix
   prediction
     0  1
  0 35  2
  1  7 10
```

For more clear understanding I have plotted a confusion matrix table which shows the detail row and column count with total observation.

```
> CrossTable(prediction, test_set[,11],
+            prop.chisq = FALSE, prop.t = FALSE,
+            dnn = c('predicted', 'actual'))


   Cell Contents
|-------------------------|
|                       N |
|           N / Row Total |
|           N / Col Total |
|-------------------------|


Total Observations in Table:  54


             | actual
   predicted |         0 |         1 | Row Total |
-------------|-----------|-----------|-----------|
           0 |        35 |         7 |        42 |
             |     0.833 |     0.167 |     0.778 |
             |     0.946 |     0.412 |           |
-------------|-----------|-----------|-----------|
           1 |         2 |        10 |        12 |
             |     0.167 |     0.833 |     0.222 |
             |     0.054 |     0.588 |           |
-------------|-----------|-----------|-----------|
Column Total |        37 |        17 |        54 |
             |     0.685 |     0.315 |           |
-------------|-----------|-----------|-----------|
```

True positives (TP): Correctly predicted values.

- The first row and first column shows the true positive (TP) cases, means the students that already not placed and Naive Bayes predicts they are not got placement.

True negatives (TN): Correctly rejected the prediction.

- The second row and first column shows student is placed in real world but Naive Bayes predict they are not placed(FP).

False positives (FP): We predicted yes, but the correct answer is no.
False negatives (FN): We predicted no, but the correct answer is yes.

- Last column and last row are False Negative (FN) that means students who are placed and Naive Bayes predict as they are actually placed.

Important statistics has been calculated from confusion matrix using formula which are as follows,

```
> Accuracy = (35+10)/(35+2+7+10)
> Accuracy
[1] 0.8333333
>
> Sentivity = (10)/(10+7)
> Sentivity
[1] 0.5882353
>
> Specificity = (35)/(35+2)
> Specificity
[1] 0.9459459
>
> Precision = (10)/(10+2)
> Precision
[1] 0.8333333
```

As we can observe that our model gives 83% accuracy with good specificity and precision. The accuracy is much better, so we can trust our model.

Discussion:

What makes the problem interesting from the viewpoint of analytics?

Considering the problem statement the data includes secondary and higher secondary school percentage. It also includes degree specialization type, work experience and salary offers to the placed students. The bayes model can be effective because

- The Naïve Bayes model helps to solve the problem with a small amount of training data, it can achieve better results than other classifiers because it has a low tendency to overfit the data.
- Training is much faster, and consists of computing to the priors.
- The model treats all variables by the same order, by ignoring the relationship among variables, it has a high bias.
- Probability is predicting the chance for a future event to take place.
- Statistics involves the analysis of the frequency of past events.

A combination of these is used for the problem thus making it interesting from the analytics point of view.

How did the chosen technique help to illuminate, or solve the problem?

- Data analysis and its technique is very helpful when we wonder how our email provider filters the spam mail, how the online news channel provides the text classification, and how companies perform sentimental analysis on social media all of this and more is done through machine learning algorithm Naive Bayes classifier.

What analysis do you think should be conducted next?

- Sentiment analysis adds a whole new dimension to the company's performance insights. It enables to uncover of the audience's emotions about the brand, which can shape marketing strategy and make it much more effective.

Reference:

1]https://www.datanovia.com/en/blog/top-r-color-palettes-to-know-for-great-data-visualization/

2]https://www.saedsayad.com/naive_bayesian.htm

3]https://bookdown.org/max/FES/naive-bayes.html