1) Observations made of the dataset provided for this competition.

The dataset provided contains training and testing data. The training dataset has 34 data columns which represent the features. We need to predict the Attribute feature for the testing dataset given the 33 other features.

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1028 entries, 0 to 1027
Data columns (total 34 columns):
Age                         1028 non-null int64
Attrition                   1028 non-null int64
BusinessTravel              1028 non-null object
DailyRate                   1028 non-null int64
Department                  1028 non-null object
DistanceFromHome            1028 non-null int64
Education                   1028 non-null int64
EducationField              1028 non-null object
EmployeeCount               1028 non-null int64
EmployeeNumber              1028 non-null int64
EnvironmentSatisfaction     1028 non-null int64
Gender                      1028 non-null object
HourlyRate                  1028 non-null int64
JobInvolvement              1028 non-null int64
JobLevel                    1028 non-null int64
JobRole                     1028 non-null object
JobSatisfaction             1028 non-null int64
MaritalStatus               1028 non-null object
MonthlyIncome               1028 non-null int64
MonthlyRate                 1028 non-null int64
NumCompaniesWorked          1028 non-null int64
OverTime                    1028 non-null object
PercentSalaryHike           1028 non-null int64
PerformanceRating           1028 non-null int64
RelationshipSatisfaction    1028 non-null int64
StockOptionLevel            1028 non-null int64
TotalWorkingYears           1028 non-null int64
TrainingTimesLastYear       1028 non-null int64
WorkLifeBalance             1028 non-null int64
YearsAtCompany              1028 non-null int64
YearsInCurrentRole          1028 non-null int64
YearsSinceLastPromotion     1028 non-null int64
YearsWithCurrManager        1028 non-null int64
ID                          1028 non-null int64
dtypes: int64(27), object(7)
memory usage: 273.2+ KB
```

Fig 1 data.info()

From the above Fig 1 it is clear that there are 27 integer type features and 7 object type features.During preprocessing we deal with modelling of the object type features.

[6]:
```
data.describe().T
```

Out[6]:

	count	mean	std	min	25%	50%	75%	max
Age	1028.0	36.999027	9.444297	18.0	30.00	36.0	43.00	60.0
Attrition	1028.0	0.167315	0.373439	0.0	0.00	0.0	0.00	1.0
DailyRate	1028.0	806.551556	407.043735	102.0	465.75	813.0	1157.25	1499.0
DistanceFromHome	1028.0	9.010700	8.078418	1.0	2.00	7.0	13.00	29.0
Education	1028.0	2.873541	1.032840	1.0	2.00	3.0	4.00	5.0
EmployeeCount	1028.0	1.000000	0.000000	1.0	1.00	1.0	1.00	1.0
EmployeeNumber	1028.0	710.198444	418.513656	1.0	351.75	701.5	1069.25	1447.0
EnvironmentSatisfaction	1028.0	2.719844	1.069614	1.0	2.00	3.0	4.00	4.0
HourlyRate	1028.0	65.451362	20.274229	30.0	48.00	65.0	83.00	100.0
JobInvolvement	1028.0	2.731518	0.703240	1.0	2.00	3.0	3.00	4.0
JobLevel	1028.0	2.093385	1.141854	1.0	1.00	2.0	3.00	5.0
JobSatisfaction	1028.0	2.757782	1.105306	1.0	2.00	3.0	4.00	4.0
MonthlyIncome	1028.0	6632.573930	4855.747841	1009.0	2886.00	4907.5	8729.50	19999.0
MonthlyRate	1028.0	14243.531128	7048.768076	2094.0	8259.75	14074.5	20342.25	26999.0
NumCompaniesWorked	1028.0	2.697471	2.527534	0.0	1.00	2.0	4.00	9.0
PercentSalaryHike	1028.0	15.172179	3.647641	11.0	12.00	14.0	18.00	25.0
PerformanceRating	1028.0	3.152724	0.359896	3.0	3.00	3.0	3.00	4.0
RelationshipSatisfaction	1028.0	2.732490	1.088774	1.0	2.00	3.0	4.00	4.0
StockOptionLevel	1028.0	0.758755	0.836236	0.0	0.00	1.0	1.00	3.0
TotalWorkingYears	1028.0	11.417315	8.015441	0.0	6.00	10.0	16.00	40.0
TrainingTimesLastYear	1028.0	2.769455	1.302518	0.0	2.00	3.0	3.00	6.0
WorkLifeBalance	1028.0	2.763619	0.703005	1.0	2.00	3.0	3.00	4.0
YearsAtCompany	1028.0	7.100195	6.316289	0.0	3.00	5.0	9.00	40.0
YearsInCurrentRole	1028.0	4.263619	3.630387	0.0	2.00	3.0	7.00	18.0
YearsSinceLastPromotion	1028.0	2.210117	3.268559	0.0	0.00	1.0	3.00	15.0
YearsWithCurrManager	1028.0	4.141051	3.608460	0.0	2.00	3.0	7.00	17.0
ID	1028.0	513.500000	296.902341	0.0	256.75	513.5	770.25	1027.0

Fig 2: data.describe()

Fig 2 is a clear description of our data describing their mean,standard deviation and other major distribution parameters like 25%,50%,min,max etc.

Attrition	0	1
Age	37.712617	33.447674
BusinessTravel	1.609813	1.610465
DailyRate	815.773364	760.656977
Department	1.251168	1.337209
DistanceFromHome	8.696262	10.575581
Education	2.892523	2.779070
EducationField	2.217290	2.354651
EmployeeCount	1.000000	1.000000
EmployeeNumber	705.167056	735.238372
EnvironmentSatisfaction	2.778037	2.430233
Gender	0.574766	0.610465
HourlyRate	65.619159	64.616279
JobInvolvement	2.781542	2.482558
JobLevel	2.179907	1.662791
JobRole	4.427570	4.808140
JobSatisfaction	2.806075	2.517442
MaritalStatus	1.047897	1.401163
MonthlyIncome	6988.733645	4860.058140
MonthlyRate	14162.245327	14648.069767
NumCompaniesWorked	2.655374	2.906977
OverTime	0.240654	0.558140
PercentSalaryHike	15.156879	15.238372
PerformanceRating	3.151869	3.156977
RelationshipSatisfaction	2.757009	2.610465
StockOptionLevel	0.818925	0.459302
TotalWorkingYears	12.059579	8.220930
TrainingTimesLastYear	2.792056	2.656977
WorkLifeBalance	2.783879	2.662791
YearsAtCompany	7.476636	5.226744
YearsInCurrentRole	4.523364	2.970930
YearsSinceLastPromotion	2.257009	1.976744
YearsWithCurrManager	4.385514	2.924419

Fig 3: Grouped w.r.t Attrition

From Fig 3 we can clearly observe a major difference in mean of parameters having attrition 0 and 1.

For example consider Job level mean for attrition value 0 is 2.1799 and for attrition 1 is 1.662.This obviously indicates chances of an employee leaving the company is higher incase of low employee level.

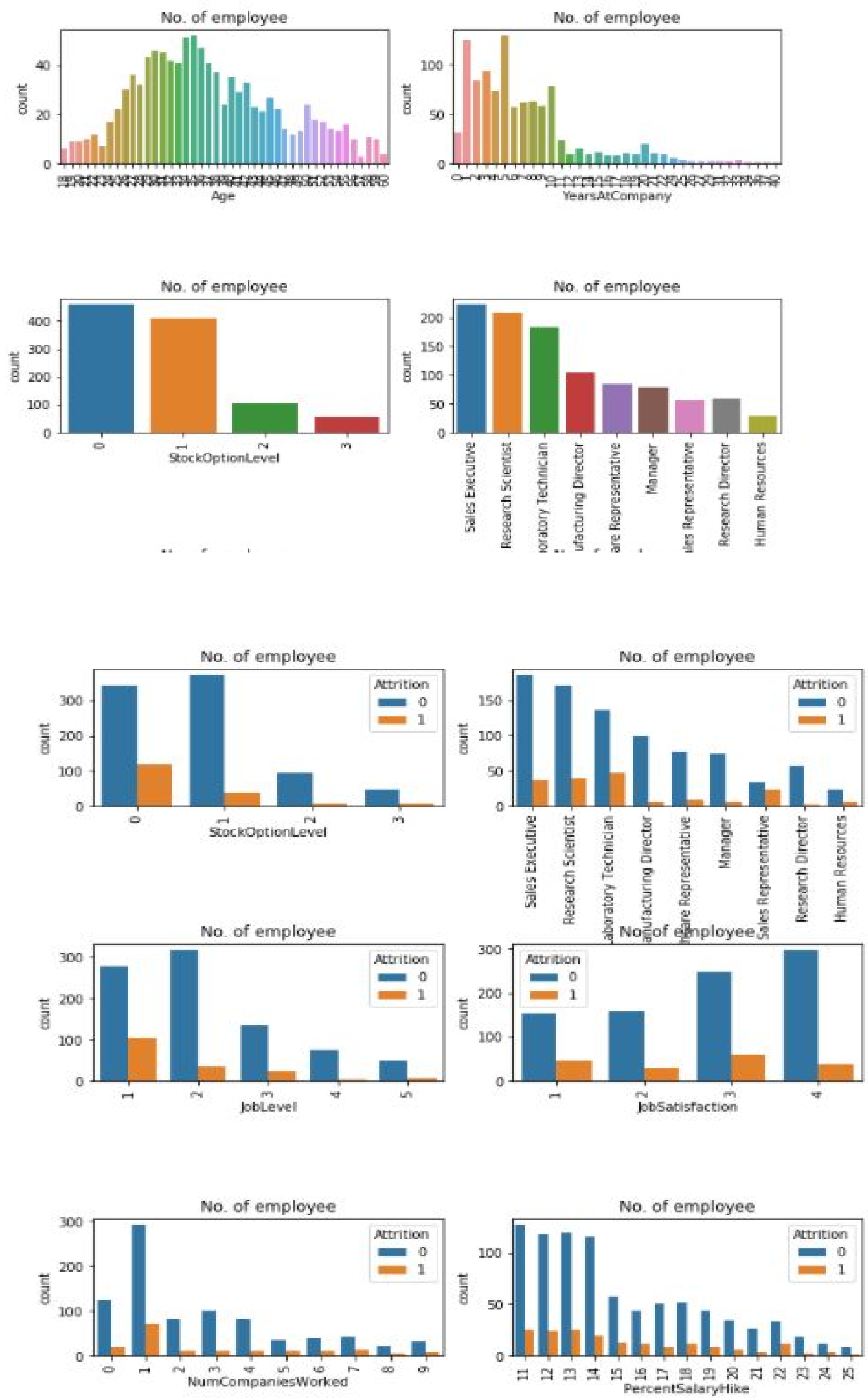


Fig 4: Observations comparing various features with attrition values and count of employees

Some such observations made are:

- Young employees with ages in the 25-35 range generally leave the company.
- Chances of leaving the company are high after working for 1 year in the company. Generally after working for 10 years employees don't leave the company.
- Also people who have worked only in 1 company are likely to leave.
- Lower job level employees to be specific 0 leaves more than others.
- Employees with zero stock options leave.
- Sales representative leaves most of the time. Manufacturing director,manager,HR etc don't leave.

2)Preprocessing method used

Machines don’t understand free text, image or video data as it is, they understand 1s and 0s. So it probably won’t be good enough if we put on a slideshow of all our images and expect our machine learning model to get trained just by that.

Our data contains categorical and numerical features.

- **Categorical :** Features whose values are taken from a defined set of values. For instance, days in a week : {Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, Sunday} is a category because its value is always taken from this set.
- **Numerical :** Features whose values are continuous or integer-valued. They are represented by numbers and possess most of the properties of numbers. For instance, the number of steps you walk in a day, or the speed at which you are driving your car at.

Various preprocessing schemes for categorical data:

One hot encoding:

It refers to splitting the column which contains *categorical data* to many columns depending on the number of categories present in that column. Each column contains “0” or “1” corresponding to which column it has been placed.

Label Encoding:

It refers to converting the labels into numeric form so as to convert it into the machine-readable form. Machine learning algorithms can then decide in a better way on how those labels must be operated. It is an important preprocessing step for the structured dataset in supervised learning.

My observation and choosing a preprocessing method:

I first tried using both these methods but finally chose Label encoding. Incase of One hot encoding each categorical data is split into various columns. For example consider the feature JobRole, it can take various options like Sales Executive, Research Scientist, Laboratory Technician, Research Director, Manufacturing director, Manager, Human Resources, Sales Representative i.e by using one hot encoding we develop 8 columns instead of 1. The number of features to train the model becomes quite difficult. Whereas in the label encoding each of the above is given a particular integer value from 1 to 8 specifying each job role thus making it easier. So, I have chosen a linear encoding scheme for the 7 categorical data given in the assignment.

3) List of various approaches used for better accuracy:

Decision tree:

A decision tree is a flowchart-like structure in which each internal node represents a “test” on an attribute (e.g. whether a coin flip comes up heads or tails), each branch represents the outcome of the test, and each leaf node represents a class label (decision taken after computing all attributes). The paths from root to leaf represent classification rules.

By using 20% validation data the prediction was as follows using the Decision tree classifier.

```
Accuracy: 0.7669902912621359
Precision: 0.2631578947368421
Recall: 0.3333333333333333
```

Gradient boosting classifier:

Gradient boosting classifiers are a group of machine learning algorithms that combine many weak learning models together to create a strong predictive model. Decision trees are usually used when doing gradient boosting. Gradient boosting models are becoming popular because of their effectiveness at classifying complex datasets, and have recently been used to win many Kaggle data science competitions.

By using 20% validation data the prediction was as follows using Gradient boosting classifier.

```
Accuracy: 0.8543689320388349
Precision: 0.5
Recall: 0.26666666666666666
```

LogisticRegression:

Logistic regression is the appropriate regression analysis to conduct when the

dependent variable is dichotomous (binary). Like all regression analyses, the logistic regression is a predictive analysis. Logistic regression is used to describe data and to explain the relationship between one dependent binary variable and one or more nominal, ordinal, interval or ratio-level independent variables.
By using 20% validation data the prediction was as follows using Logistic regression.

```
Accuracy: 0.8737864077669902
Precision: 0.625
Recall: 0.3333333333333333
```

The precision and recall values means the following:

$$\text{Precision} = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

$$\text{Recall} = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

By comparing the above accuracy values it is more beneficial to use logistic regression and gradient boosting algorithms. Also I observed the output scores for various percentages of validation and all these 3 methods.

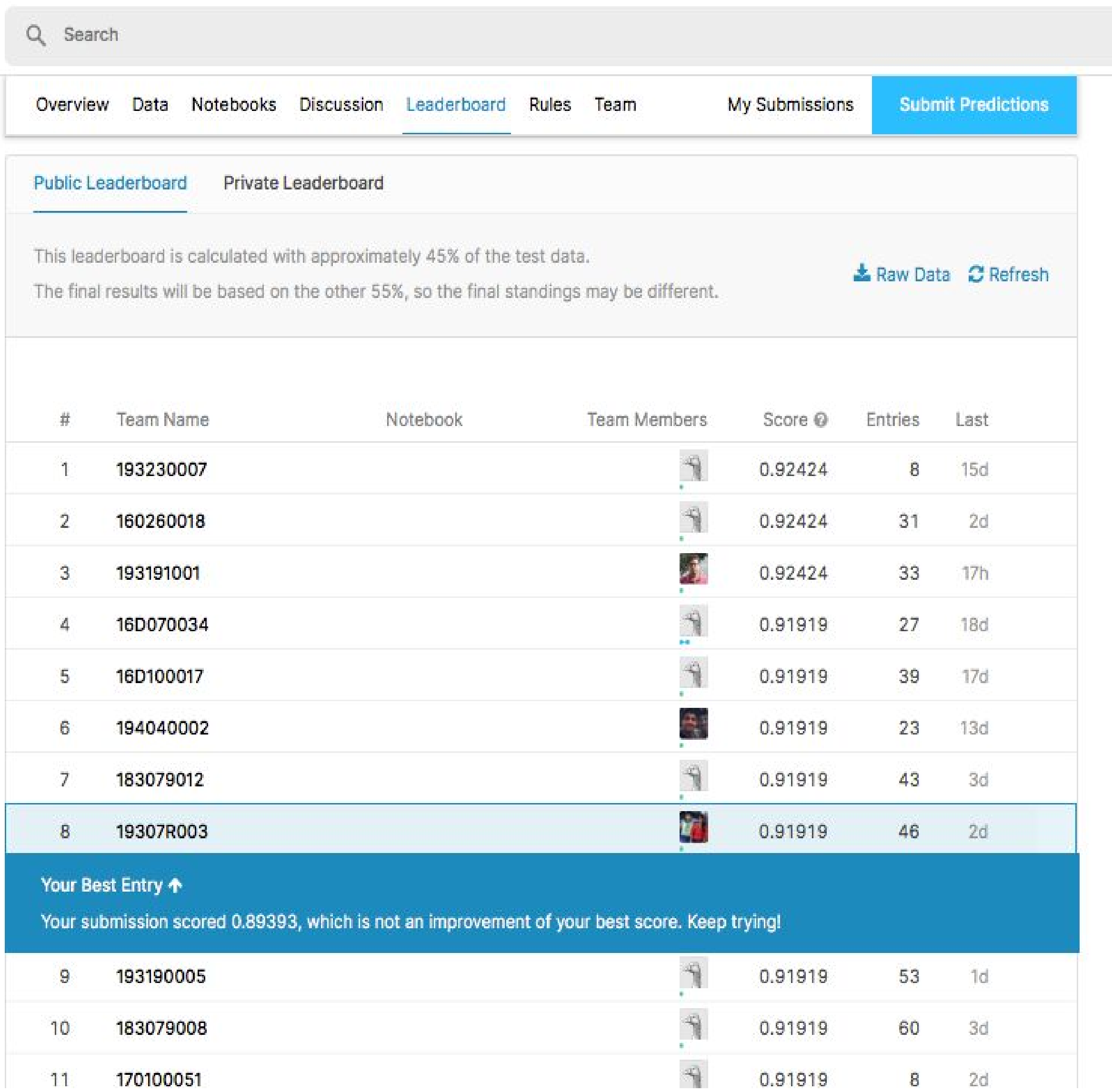
4)Results and final learning

Final learning:

This assignment was a hands-on experience in ML where the problem statement was basic but quite realistic. It helped me learn how to use the kaggle environment and I also observed how different classifying algorithms worked. My project for this course is based on categorical data classification. So, this assignment was helpful for me to get a clear idea on how to proceed with my project and I also understood the types of data used.

Results:

My best score in the public leaderboard of the kaggle competition is 0.91919.



Search

Overview Data Notebooks Discussion Leaderboard Rules Team My Submissions Submit Predictions

Public Leaderboard Private Leaderboard

This leaderboard is calculated with approximately 45% of the test data.
The final results will be based on the other 55%, so the final standings may be different.

Raw Data Refresh

#	Team Name	Notebook	Team Members	Score	Entries	Last
1	193230007			0.92424	8	15d
2	160260018			0.92424	31	2d
3	193191001			0.92424	33	17h
4	16D070034			0.91919	27	18d
5	16D100017			0.91919	39	17d
6	194040002			0.91919	23	13d
7	183079012			0.91919	43	3d
8	19307R003			0.91919	46	2d
9	193190005			0.91919	53	1d
10	183079008			0.91919	60	3d
11	170100051			0.91919	8	2d

Your Best Entry ↑
Your submission scored 0.89393, which is not an improvement of your best score. Keep trying!