DAV 6150 Final Project
Predict Employee Attrition

Group members:
Manling Yang, Xiaojia He, Qi Sun

Introduction

Attrition is one of the important issues in an organization. Stovel, M. and Bontis, N. (2002) draw attention on controlling attrition, they state that the value of employees to an organization is a very crucial element in the success of the organization. If organizations know why their employees leave, they can develop effective strategies for employee retention. The purposes of this study are to find out why employees left the organization and to develop a best model that can predict how likely the employee quit the job.

Data Overview:

- Kaggle
- 1470 observations with 22 attributes. (Attrition is response variable, binary “Yes/No’)
- Data types contain continuous, boolean, and dictionary

Research Questions

1. What are the relationships between attrition and other variables?
2. What is the best model to predict employees' attrition?
3. Which features are most contributed to the response variable?

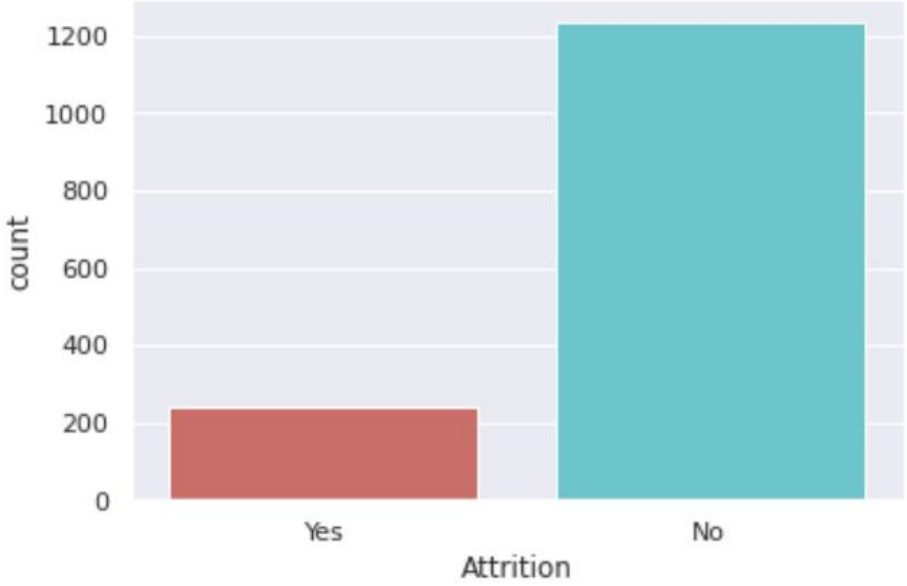
EDA Works

1. **Data exploration**: check shape, describe, dtypes, isnull, etc..
2. **Check distribution of dependent variable**
3. **Check duplicate data** - no duplicated data
4. **Check the relationship between response variable and independent variable by bar chart, box plot and pointplot**. (11 features)

- Education, EnvironmentSatisfaction, JobInvolvement, JobLevel, JobSatisfaction, worklifebalance, relationshipSatisfaction, Age, DailRate, DistanceFromHome, TotalWorkingYears maybe more predictable.

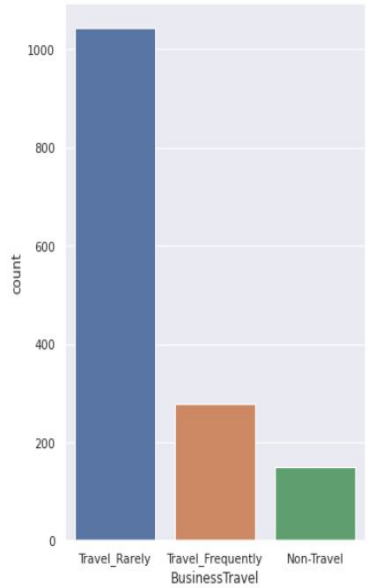
EDA Works -Response variable

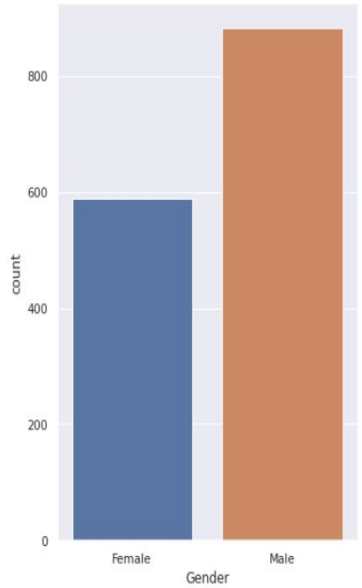
- The Attrition is imbalance, 84% are No attrition.

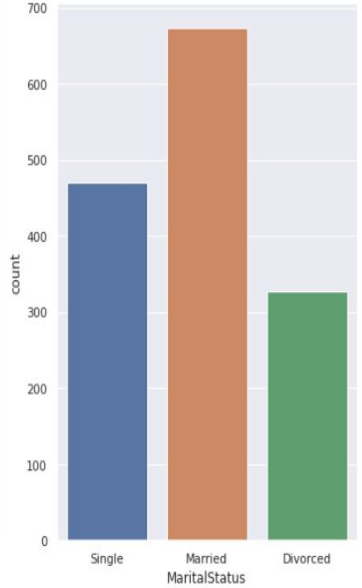


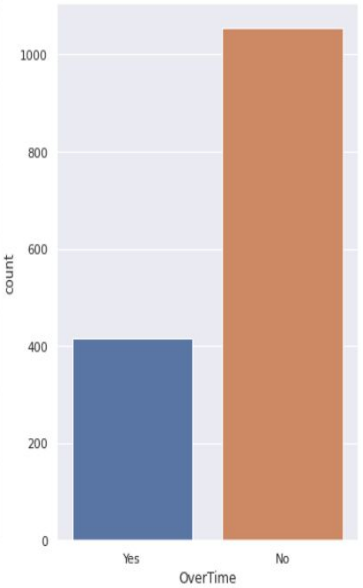
EDA Works - Categorical Data

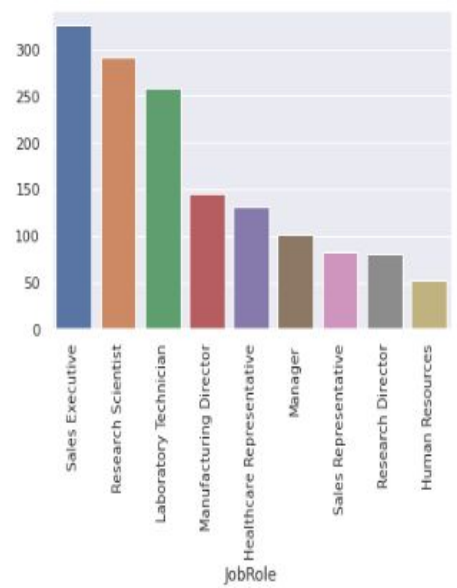
- In the dataset, most observations are married male employees, they travel rarely and do not work overtime.
- Sales Executive, Research Scientist, and Laboratory Technician are the top 3 main job role in the company.









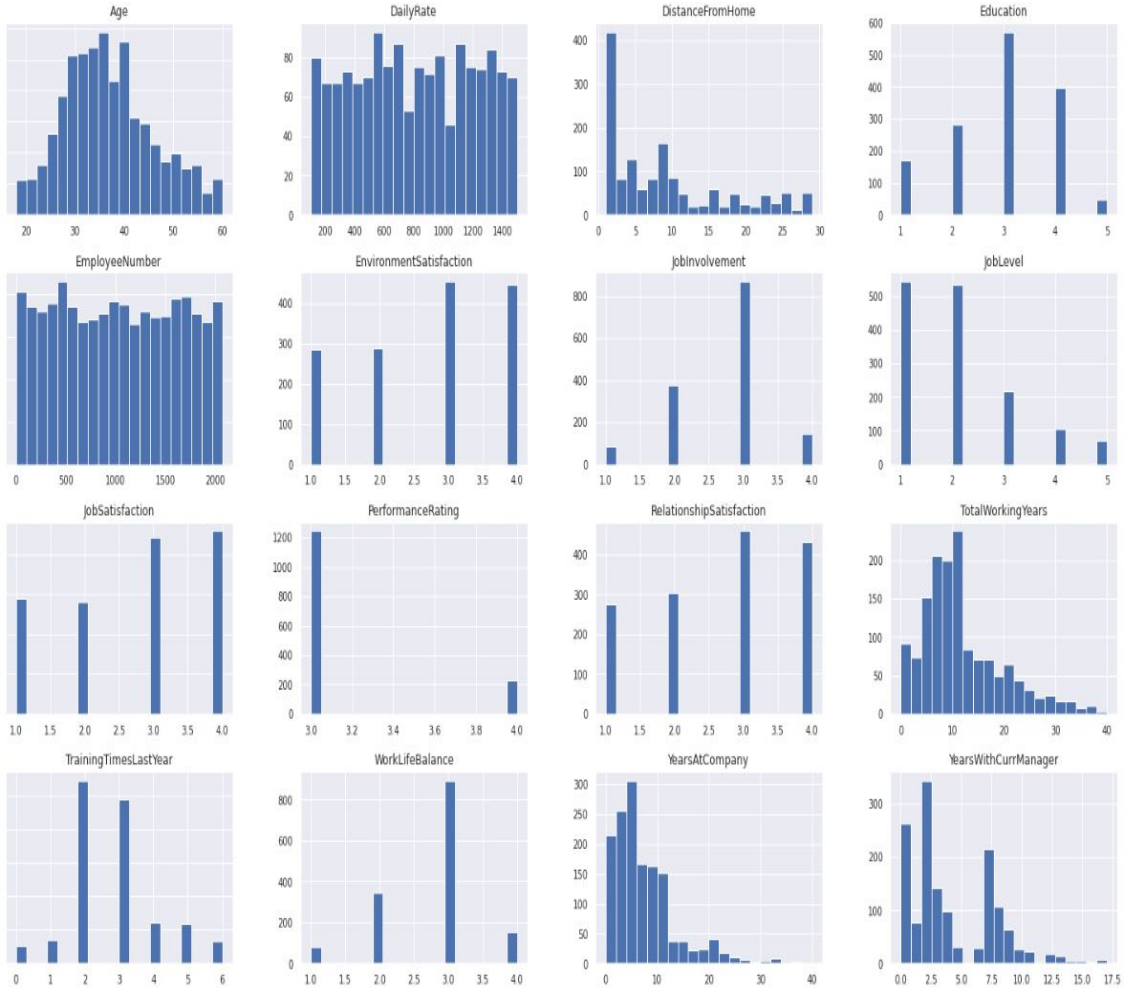


EDA Works -Numerical

Most employees have high environment, Job, relationship satisfaction. And they also have an excellent performance rating and work-life balance.

Most observations employees are working no more than ten years totally and stay with current manager no more than 2.5 years.

Most employees trained 2 or 3 times at last year.



Data Prep & Feature Selection

Data Preparation:

1. Drop irrelevant columns 'EmployeeNumber'
2. Label the binary dependent variable (Attrition)
3. Convert categorical variable into dummy variables
4. Split data into training and testing subsets
5. Scaling numerical values using MinMaxScaler (except dummy variables)

Feature Selection:

1. Correct imbalance response variable by SMOTE
2. Use SelectFromModel RandomForest to select 15 features

'Age', 'DailyRate', 'DistanceFromHome', 'Education', 'EnvironmentSatisfaction', 'JobInvolvement', 'JobLevel', 'JobSatisfaction', 'RelationshipSatisfaction', 'TotalWorkingYears', 'TrainingTimesLastYear', 'WorkLifeBalance', 'YearsAtCompany', 'YearsWithCurrManager', 'OverTime_Yes'

Build Machine Learning Models and Optimize Parameters

We use Logistic Regression, KNN, SVM, Decision Tree, and Random Forest to build models.

The same process was used for each model and optimizing the parameters:

1. Build base models with default parameters
2. Use **RandomizedSearchCV** for optimizing parameters. We picked some important hyperparameters to tune
3. Build tuned models with best parameters
4. Model Evaluation - 5-fold cross validation to evaluate models based on accuracy, recall, F1, precision, and roc_auc scores.

Model Evaluation - Logistic Regression

Metric	Logistic Regression base model	Logistic Regression tuned model
# Indep. Vars	15	15
The Accuracy of the model	0.749	0.756
Recall	0.769	0.771
F1 Score	0.757	0.762
precision	0.745	0.755
roc_auc Score	0.822	0.820

For Logistic Regression model, the tuned model performs better than the base model, since their Accuracy, Recall, F1 and Precision scores are higher than that in base model.

So we choose tuned model as our preferred model for logistic regression model building.

Model Evaluation - KNN

Metric	KNN base model	KNN tuned model
# Indep. Vars	15	15
The Accuracy of the model	0.807	0.877
Recall	0.943	0.951
F1 Score	0.832	0.888
precision	0.943	0.951
roc_auc Score	0.899	0.935

For KNN model, the tuned model performs better than the base model, since their Accuracy, Recall, F1, Precision and Roc_auc scores are higher than that in base model.

So we choose tuned model as our preferred model for KNN model building.

Model Evaluation - SVM

Metric	SVM base model	SVM tuned model
# Indep. Vars	15	15
The Accuracy of the model	0.824	0.870
Recall	0.812	0.906
F1 Score	0.824	0.876
precision	0.837	0.849
roc_auc Score	0.904	0.942

For SVM model, the tuned model performs better than the base model, since their Accuracy, Recall, F1, Precision and Roc_auc scores are higher than that in base model.

So we choose tuned model as our preferred model for SVM model building.

Model Evaluation - Decision Tree

Metric	Decision Tree base model	Decision Tree tuned model
# Indep. Vars	15	15
The Accuracy of the model	0.838	0.766
Recall	0.849	0.752
F1 Score	0.843	0.765
precision	0.838	0.781
roc_auc Score	0.838	0.823

For Decision tree model, the base model performs better than the tuned model, since their Accuracy, Recall, F1, Precision and Roc_auc scores are higher than that in tuned model.

So we choose base model as our preferred model for Decision tree model building.

Model Evaluation - Random Forest

Metric	Random forest base model	Random forest tuned model
# Indep. Vars	15	15
The Accuracy of the model	0.911	0.916
Recall	0.878	0.887
F1 Score	0.910	0.915
precision	0.944	0.956
roc_auc Score	0.969	0.969

For Random Forest model, the tuned model performs better than the base model, since their Accuracy, Recall, F1, Precision scores are higher than that in base model.

So we choose tuned model as our preferred model for Random forest model building.

Model Selection

Metric	Logistic Regression tuned model	KNN tuned model	SVC tuned model	Decision Tree base model	Random Forest tuned model
# Indep. Vars	15	15	15	15	15
The Accuracy of the model	0.756	0.877	0.870	0.838	0.916
Recall	0.771	0.951	0.906	0.849	0.887
F1 Score	0.762	0.888	0.876	0.843	0.915
precision	0.755	0.951	0.849	0.838	0.956
roc_auc Score	0.820	0.935	0.942	0.838	0.969

What is our preferred model in this step?

Comparing all the preferred models above, the tuned Random Forest model performs better than other models. So we choose the tuned Random Forest model as our final model.

Apply preferred model to training and testing

Confusion Matrix

```
[[777   0]
 [  4 798]]
```

Classification Report

	precision	recall	f1-score	support
0	0.99	1.00	1.00	777
1	1.00	1.00	1.00	802
accuracy			1.00	1579
macro avg	1.00	1.00	1.00	1579
weighted avg	1.00	1.00	1.00	1579

Random Forest Tuned model:

Confusion Matrix

```
[[202   8]
 [ 24 161]]
```

Classification Report

	precision	recall	f1-score	support
0	0.89	0.96	0.93	210
1	0.95	0.87	0.91	185
accuracy			0.92	395
macro avg	0.92	0.92	0.92	395
weighted avg	0.92	0.92	0.92	395

Comparing the results for our training and testing dataset, the recall, precision and f1 scores are 1 which is higher than in our testing dataset which is 0.92. But 0.92 is still a very high score and close to 1. So our final model performs well as we expected.

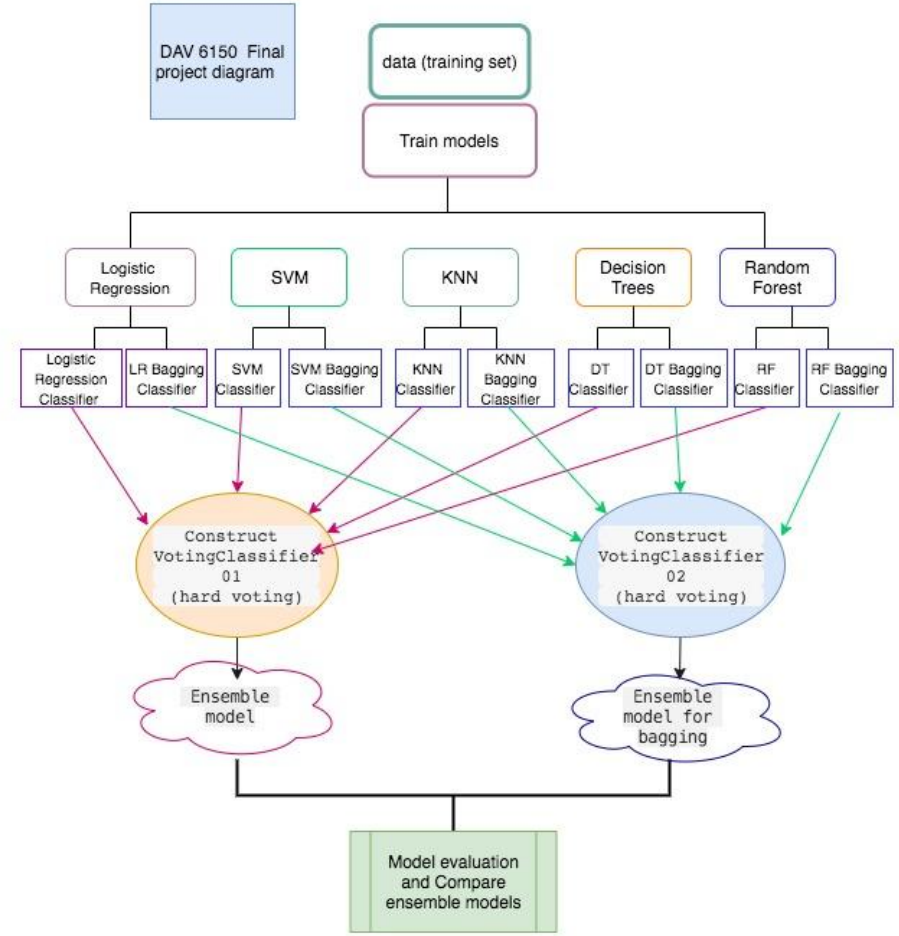
Ensemble Model

We created two Ensemble Models:

1.Ensemble Model comprising all of the models selected from last step.

2. Ensemble Model comprising Bagging models. We created Bagging models by using Scikit-learn's BaggingClassifier.

We used sklearn's VotingClassifier to combine different classifiers and perform a vote.



Model Evaluation Results:

1) Compare Ensemble Model to the Models selected from Step 5:

Metric	Logistic Regression tuned model	KNN tuned model	SVM tuned model	Decision Tree base model	Random Forest tuned model	Ensemble model
# Indep. Vars	15	15	15	15	15	15
The Accuracy of the model	0.7555	0.8771	0.8702	0.8385	0.9164	0.9056
Recall	0.7706	0.9514	0.9064	0.8491	0.8865	0.9189
F1 Score	0.7623	0.8876	0.8763	0.8425	0.9151	0.9083
precision	0.7545	0.832	0.8485	0.8377	0.9456	0.8982

2) Compare Bagging Ensemble Model to the Bagging Models:

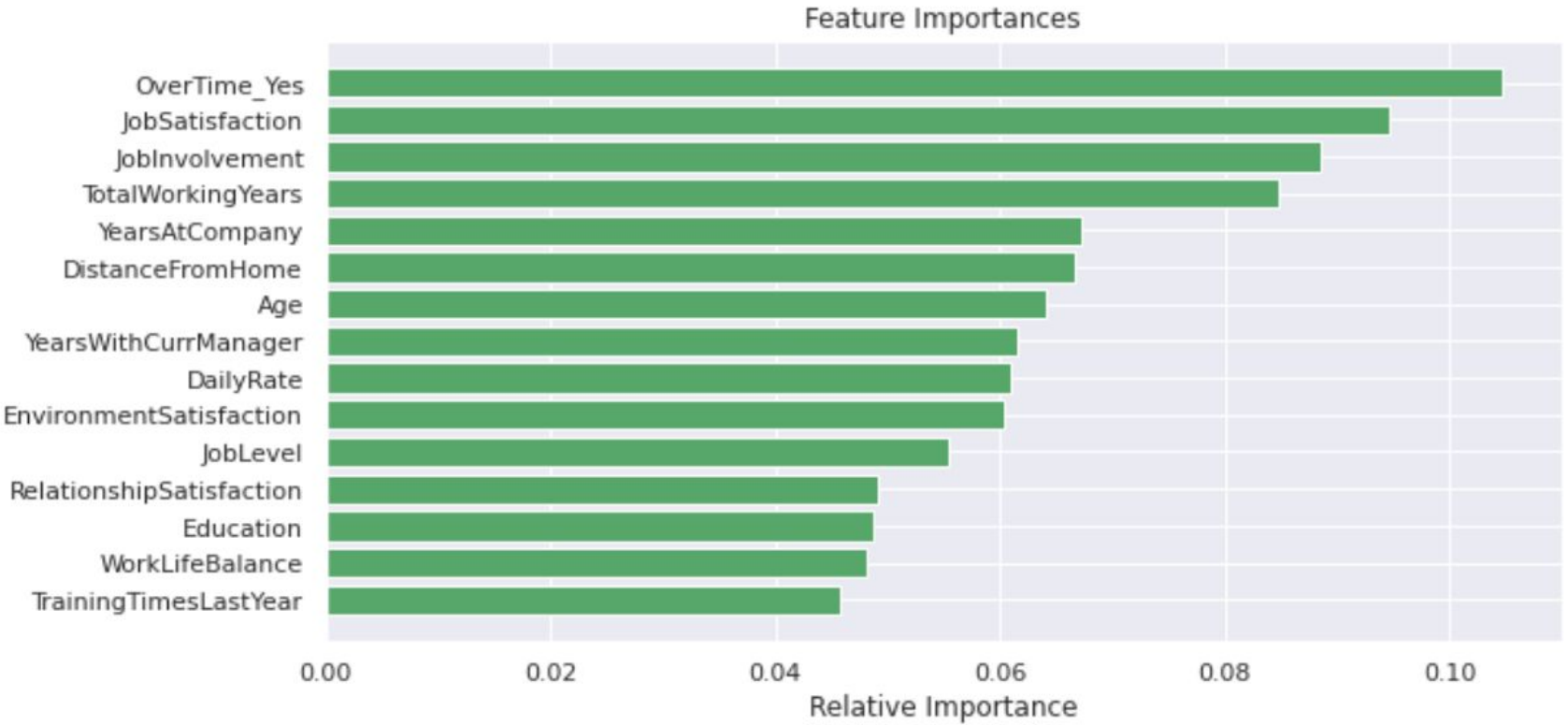
Metric	Bagging Logistic Regression model	Bagging KNN model	Bagging SVM model	Bagging Decision Tree model	Bagging Random Forest model	Bagging Ensemble model
# Indep. Vars	15	15	15	15	15	15
The Accuracy of the model	0.7570	0.8828	0.8708	0.9006	0.9139	0.9062
Recall	0.7718	0.9551	0.9090	0.8791	0.8828	0.9152
F1 Score	0.7635	0.8925	0.8771	0.8999	0.9125	0.9085
precision	0.7557	0.8382	0.8477	0.9219	0.9446	0.9023

Research Question 2: What is the best model to predict employees' attrition?

The Random Forest tuned Model is our final model to predict employee attrition.

Feature Importance:

Research Question 3: Which features are most contributed to the response variable?



Explain the Model by using LIME

Lime supports explanations for individual predictions.

For example, we used LIME to predict the case #18. This case attrition value is 0.

Let's see what our model predicts this employee attrition.

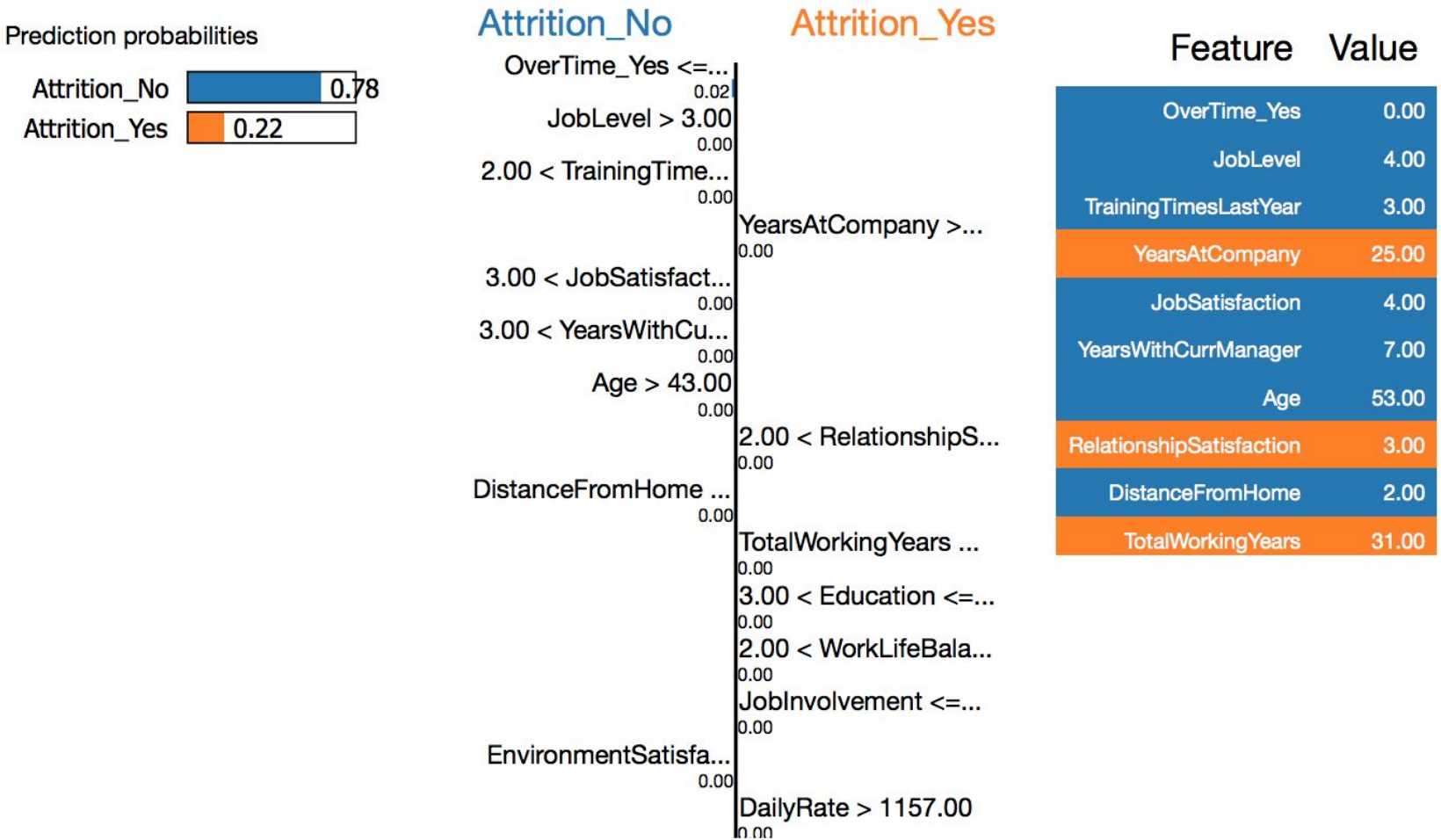
```
# example
i=18

df_new.loc[[i]]
```

	Age	DailyRate	DistanceFromHome	Education	EnvironmentSatisfaction	JobInvolv
18	53	1219	2	4	1	

```
# attrition actual value
y.loc[[i]]
```

	Attrition
18	0


Prediction probabilities
Attrition_No
0.78
Attrition_Yes
0.22
Attrition_No
Attrition_Yes
OverTime_Yes <=...
0.02
JobLevel > 3.00
0.00
2.00 < TrainingTime...
0.00
YearsAtCompany >...
0.00
3.00 < JobSatisfact...
0.00
3.00 < YearsWithCu...
0.00
Age > 43.00
0.00
2.00 < RelationshipS...
0.00
DistanceFromHome ...
0.00
TotalWorkingYears ...
0.00
3.00 < Education <=...
0.00
2.00 < WorkLifeBala...
0.00
JobInvolvement <=...
0.00
EnvironmentSatisfa...
0.00
DailyRate > 1157.00
0.00
Feature
Value
OverTime_Yes 0.00
JobLevel 4.00
TrainingTimesLastYear 3.00
YearsAtCompany 25.00
JobSatisfaction 4.00
YearsWithCurrManager 7.00
Age 53.00
RelationshipSatisfaction 3.00
DistanceFromHome 2.00
TotalWorkingYears 31.00

Challenge

- To identify which categorical data should be encoded.
- The steps to construct ensemble model.
- How to explain the final model? And how to use LIME to explain the model?

THANK YOU!