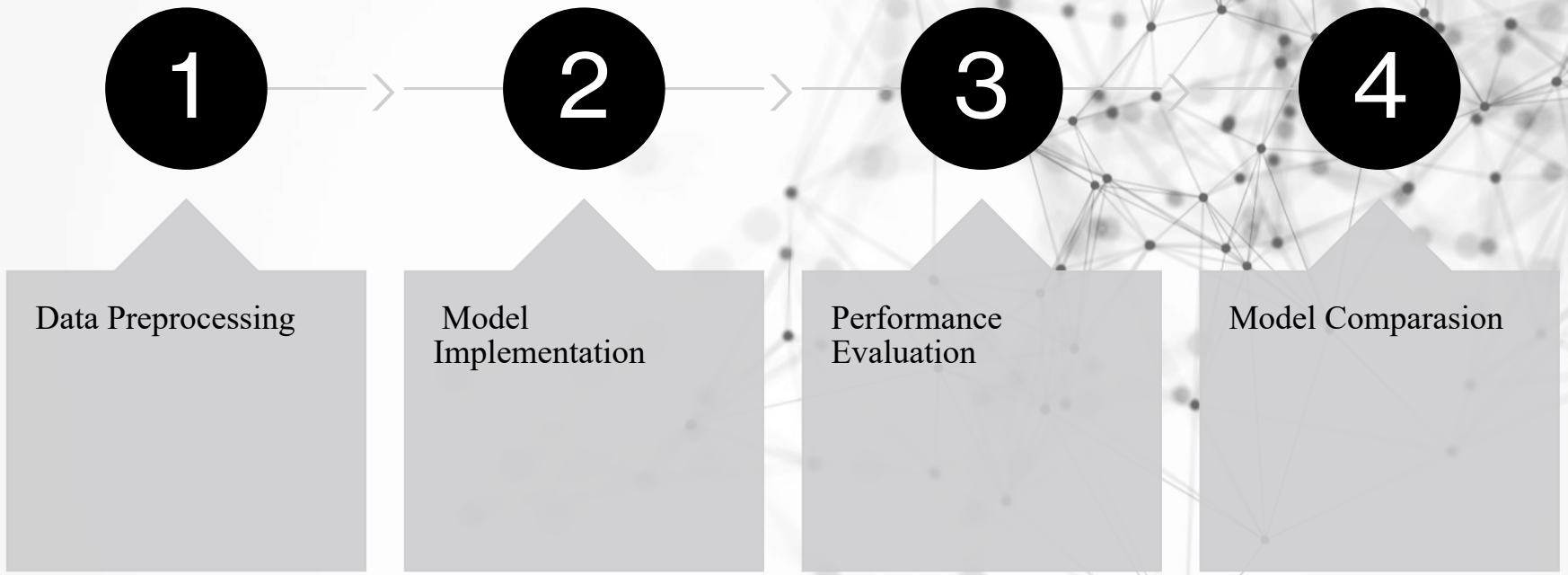
ENS 491
Machine Learning Exercise Part 1 Report

- Zeynep Derya Selçuk (29391)
- Darwin Serdar Dikici (30051)
- Selin Tıraş (2949)
- Efe Çelik (29268)
- Tunahan Arslan (29210)


1
2
3
4
Data Preprocessing
Model Implementation
Performance Evaluation
Model Comparasion

Data Preprocessing

- Used Python via Google Collab
- Upload Excel file
- Checked for missing values
- Understand dataset distribution
- Focused on possible inconsistencies
- Applied normalization
- The data was split into 85% training data and 15% test data

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 45 entries, 0 to 44
Data columns (total 4 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   X1      45 non-null     float64
 1   X2      45 non-null     int64
 2   X3      45 non-null     int64
 3   Y1      45 non-null     int64
dtypes: float64(1), int64(3)
memory usage: 1.5 KB
None
X1    0
X2    0
X3    0
Y1    0
dtype: int64
              X1         X2         X3           Y1
count  45.000000  45.000000  45.000000    45.000000
mean    0.050000  30.000000  58.333333   600.711111
std     0.028604  16.514456  31.532811   410.605242
min     0.010000  10.000000  25.000000   183.000000
25%     0.030000  10.000000  25.000000   263.000000
50%     0.050000  30.000000  50.000000   410.000000
75%     0.070000  50.000000 100.000000  1120.000000
max     0.090000  50.000000 100.000000  1211.000000
```

```
[[0.         0.         0.        ]
 [0.         0.         0.33333333]
 [0.         0.         1.        ]
 [0.         0.5        0.        ]
 [0.         0.5        0.33333333]]
```

```
Training set size: (38, 3)
Test set size: (7, 3)
```

Model Implementation

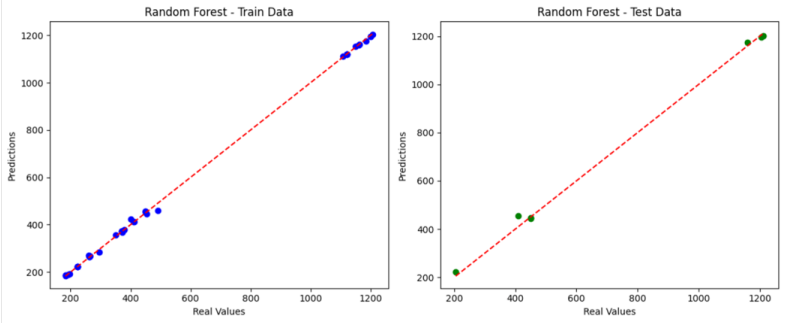
- In order to enhance accuracy and decrease errors, the Random Forest model was trained by averaging the outputs of several decision trees.
- Sequential training allowed LSBoost to deal with complex patterns better by enabling each new tree to concentrate on fixing mistakes of their predecessors.
- Predictions were produced for both the training and test data to reflect the models' performance on fresh, unseen data as well as their ability to learn from the data.

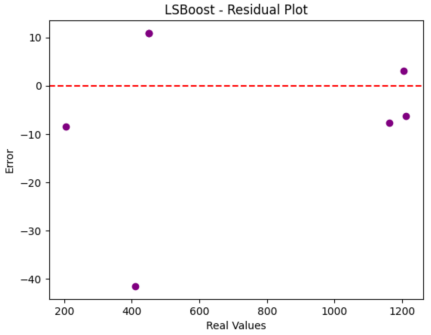
```
Random Forest - R² (Train): 1.00, R² (Test): 1.00
AME: 15.08, RMSE: 20.12
```

```
LSBoost - R² (Train): 1.00, R² (Test): 1.00
AME: 12.70, RMSE: 17.49
```

Performance Evaluation

- The R^2 (Coefficient of Determination) reflects how well the models capture the variability in the target values.
- The average gap between the expected and actual numbers is measured by AME (Absolute Mean Error), which offers a sense of typical forecast error.
- RMSE (Root Mean Squared Error) on the other hand gives more weight to the effect of larger errors, emphasizing their impact by highlighting instances with higher prediction inconsistencies.
- Regression curves were plotted for both the training and test datasets to better assess the models' performance. A red dashed line representing perfect predictions made it visibly easy to compare the predicted values with the actual ones.
- In order to investigate the distribution of the prediction errors, we generated a residual plot for LSBoost. In this plot, randomly distributed errors around 0 indicate that the model is objective and operates reliably across the data.





	Model	R^2 (Train)	R^2 (Test)	AME	RMSE
0	Random Forest	0.999636	0.997592	15.078571	20.121620
1	LSBoost	0.999569	0.998180	12.703682	17.494185

Model Comparasion

- Good outcomes were produced by both models, and their high R2 values demonstrated their good predictive power. On the test data, however, LSBoost performed better than Random Forest, indicating that it is more effective at generalizing to unseen data.
- Compared to Random Forest, LSBoost made fewer and smaller mistakes, as demonstrated by its lower AME and RMSE values. While LSBoost performed more consistently on both the training and test datasets, Random Forest was more likely to overfit the training data despite its accuracy.
- The residual plot for LSBoost revealed no indications of systematic bias in its predictions, given that its errors were randomly distributed around zero.
- In conclusion, we can suggest that LSBoost is the more suited model for this challenge because of its better generalization and reduced error rates.

END OF THE REPORT