Cairo University - Faculty of Engineering
Computer Engineering Department
CMP4040 – Machine Intelligence
Spring 2024

Machine Learning Project Report

Team #12

Employee Attrition Prediction

Team members:

Name	Section	BN
Donia Gameel	1	24
Shaza Mohammed	1	32
Heba Ashraf Raslan	2	32

Team members contribution:

Name	
Donia Gameel	Preprocessing + AdaBoost + ZeroR
Shaza Mohammed	Logistic Regression
Heba Ashraf Raslan	Preprocessing + SVM + linear SVM

Problem Definition and Motivation:

Employee attrition, or the rate at which employees leave a company, is a significant concern for organizations due to the high costs associated with hiring and training new employees. According to recent data, the average cost per hire rose to $4,700 in 2023. For specialized positions such as cybersecurity, engineering, or nursing, the cost per hire can be even higher, reaching up to $28,329 for executive positions. These costs, combined with factors such as ultra-low unemployment rates and an aging workforce, highlight the importance of predicting and mitigating employee attrition. [source: https://toggl.com/blog/cost-of-hiring-an-employee]

By developing a machine learning model that can predict employee attrition, organizations can take proactive measures to retain valuable talent and reduce turnover costs. This project aims to leverage machine learning techniques to analyze factors such as job satisfaction, salary, work-life balance, etc., and predict which employees are most likely to leave the company. By doing so, organizations can optimize their hiring and retention strategies, ultimately reducing the financial burden of employee turnover.

Evaluation Metrics:

The performance of our machine learning model will be evaluated using metrics such as accuracy, precision, recall, and F1-score. Given that the classes are imbalanced, we will use weighted f1-score as our main metric of evaluation

Dataset and References:

We will use a publicly available dataset on employee attrition, such as the IBM HR Analytics Employee Attrition & Performance dataset, which contains information about employees' demographics, job role, satisfaction levels, etc.
We will also refer to relevant research papers and articles on employee attrition prediction for guidance and insights.

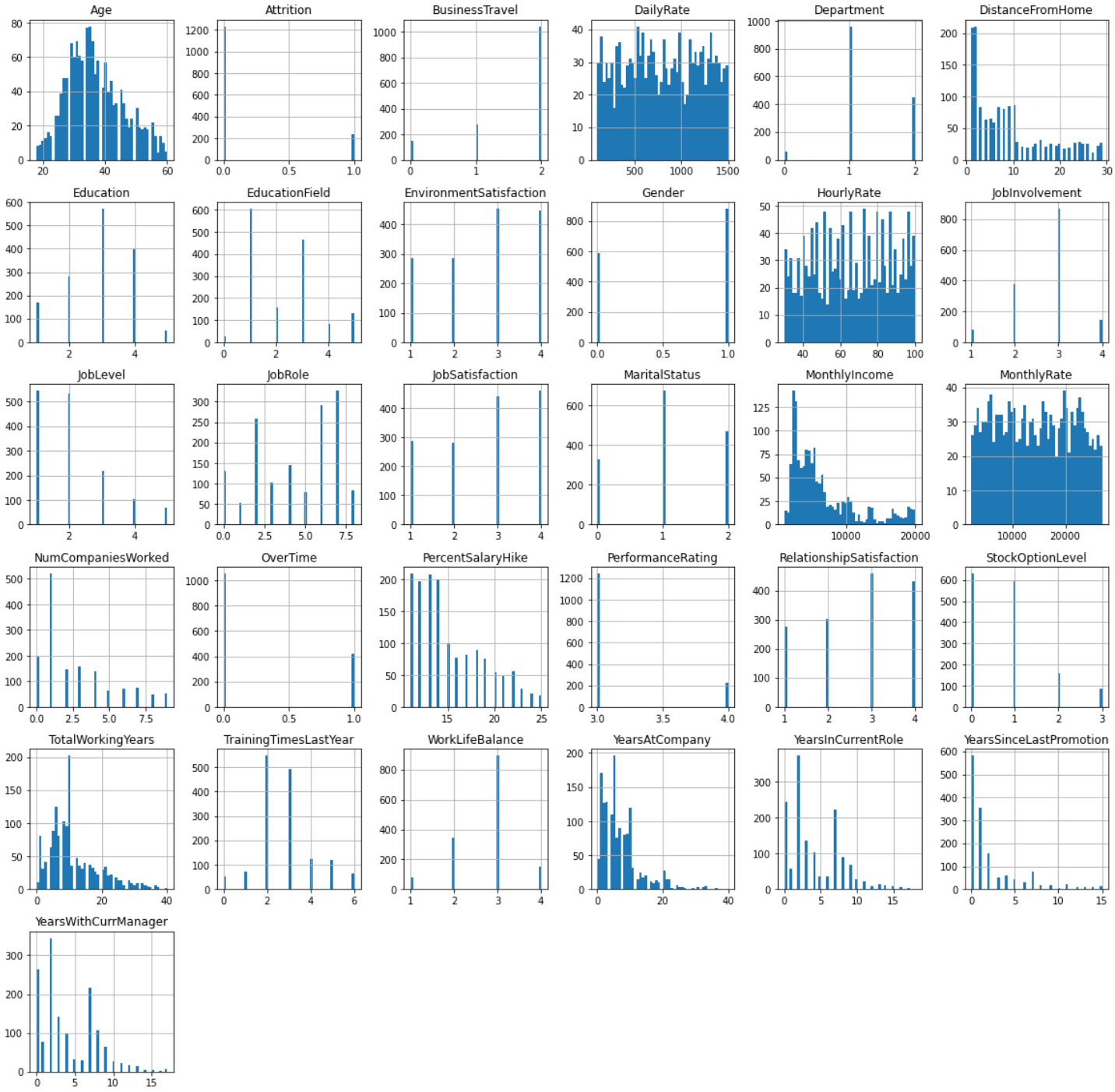
(1) Predicting Employee Attrition Using Machine Learning Technique
(2) Predicting Employee Attrition Using Machine Learning Approaches

Data Exploring:

Data columns (total 35 columns):

#	Column	Non-Null Count	Dtype	Unique values
0	Age	1470 non-null	int64	43
1	Attrition	1470 non-null	object	2
2	BusinessTravel	1470 non-null	object	3
3	DailyRate	1470 non-null	int64	886
4	Department	1470 non-null	object	3
5	DistanceFromHome	1470 non-null	int64	29
6	Education	1470 non-null	int64	5
7	EducationField	1470 non-null	object	6
8	EmployeeCount	1470 non-null	int64	1
9	EmployeeNumber	1470 non-null	int64	1470
10	EnvironmentSatisfaction	1470 non-null	int64	4
11	Gender	1470 non-null	object	2
12	HourlyRate	1470 non-null	int64	71
13	JobInvolvement	1470 non-null	int64	4
14	JobLevel	1470 non-null	int64	5
15	JobRole	1470 non-null	object	9
16	JobSatisfaction	1470 non-null	int64	4
17	MaritalStatus	1470 non-null	object	3
18	MonthlyIncome	1470 non-null	int64	1394
19	MonthlyRate	1470 non-null	int64	1427
20	NumCompaniesWorked	1470 non-null	int64	10
21	Over18	1470 non-null	object	1
22	OverTime	1470 non-null	object	2
23	PercentSalaryHike	1470 non-null	int64	15
24	PerformanceRating	1470 non-null	int64	2
25	RelationshipSatisfaction	1470 non-null	int64	4
26	StandardHours	1470 non-null	int64	1
27	StockOptionLevel	1470 non-null	int64	4
28	TotalWorkingYears	1470 non-null	int64	40
29	TrainingTimesLastYear	1470 non-null	int64	7
30	WorkLifeBalance	1470 non-null	int64	4
31	YearsAtCompany	1470 non-null	int64	37
32	YearsInCurrentRole	1470 non-null	int64	19
33	YearsSinceLastPromotion	1470 non-null	int64	16
34	YearsWithCurrManager	1470 non-null	int64	18

- **histogram for each numeric variable/feature of the dataset**



- **Check for nulls & duplicates:**
  - Total number of duplicates : 0
  - Total number of missing values : 0

- **Explore categorical features:**

number of categorical variable : 8

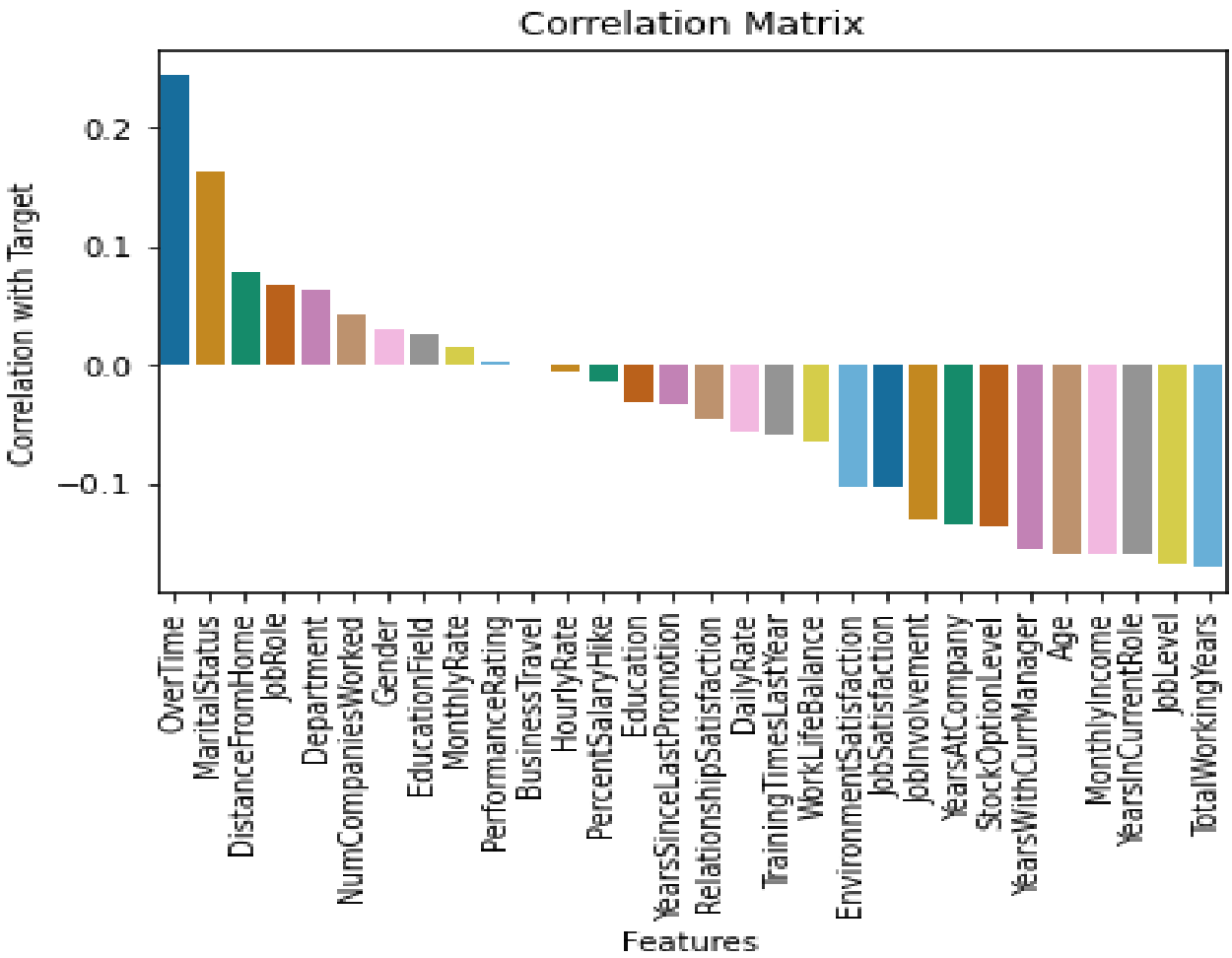
Attrition	['Yes' 'No']
BusinessTravel	['Travel_Rarely' 'Travel_Frequently' 'Non-Travel']
Department	['Sales' 'Research & Development' 'Human Resources']
EducationField	['Life Sciences' 'Other' 'Medical' 'Marketing' 'Technical Degree' 'Human Resources']
Gender	['Female' 'Male']
JobRole	['Sales Executive' 'Research Scientist' 'Laboratory Technician' 'Manufacturing Director' 'Healthcare Representative' 'Manager' 'Sales Representative' 'Research Director' 'Human Resources']
MaritalStatus	['Single' 'Married' 'Divorced']
OverTime	['Yes' 'No']

- **Exploring outliers:**

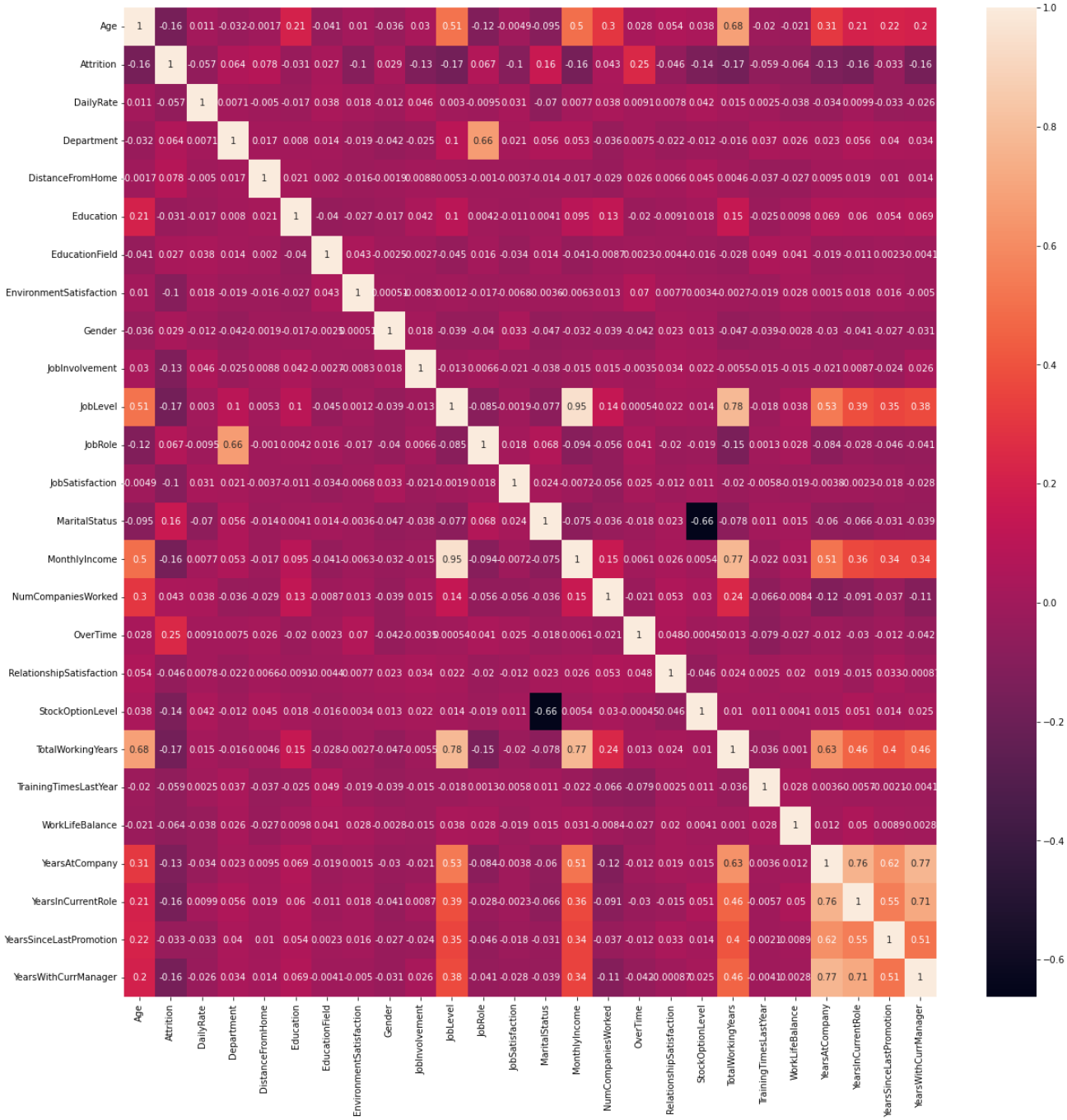
Number of rows with outliers in each column:

Attrition	237
MonthlyIncome	114
NumCompaniesWorked	52
PerformanceRating	226
StockOptionLevel	85
TotalWorkingYears	63
TrainingTimesLastYear	238
YearsAtCompany	104
YearsInCurrentRole	21
YearsSinceLastPromotion	107
YearsWithCurrManager	14

Show Correlation Between the target variables and each feature



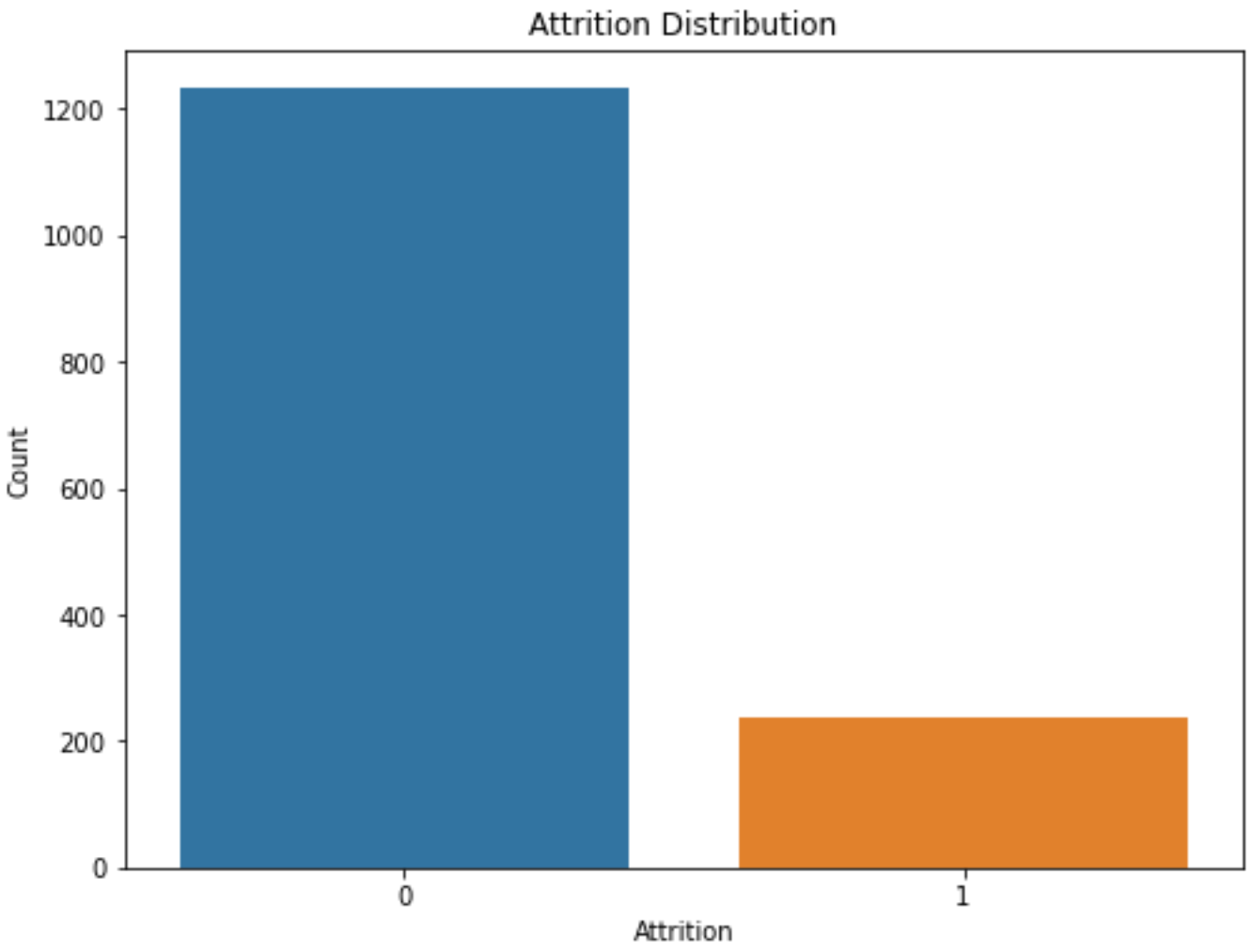
- **Correlation matrix**



There are high correlation between some features:

MonthlyIncome & joblevel	drop MonthlyIncome
Total working years & Joblevel	drop TotalWorkingYears
years in current role, years at company & years with current manager	Feature engineering

- **Check for imbalance**



As expected, the 'Attrition' column shows a significant class imbalance with 1233 'No' and 237 'Yes'.
This imbalance should be addressed prior to model training
to prevent overemphasis towards 'No' class when our study focus is on the employees who attrite - 'Yes'.

Preprocessing:

1- ('EmployeeCount', 'Over18', 'StandardHours') were found to have constant values for all 1470 rows.
   Also, 'EmployeeNumber' is a unique identifier for all 1470 rows.
   These 4 columns should be dropped as they would not be helpful in predicting attrition.
2- Encode categorical variables.
3- Remove MonthlyIncome,TotalWorkingYears, YearsInCurrentRole and YearsWithCurrManager
   taking a cutoff of 0.7 correlation coefficient.
   This will retain JobLevel and YearsAtCompany and remove possibility of multicollinearity from the features.
4- Scale the date.
5- Split The Data into training, validation, and testing sets.
6- Resampling.

Models:

- **ZeroR (baseline model):** **(weighted f1-score)**

	Classifier	Training Accuracy	Validation Accuracy	Testing Accuracy	Training F1 Score	Validation F1 Score	Testing F1 Score
0	without resampling	0.837234	0.809322	0.867347	0.763061	0.72403	0.805732
1	SMOTE	0.500000	0.809322	0.867347	0.333333	0.72403	0.805732
2	RandomOverSampler	0.500000	0.809322	0.867347	0.333333	0.72403	0.805732
3	SMOTETomek	0.500000	0.809322	0.867347	0.333333	0.72403	0.805732
4	RandomUnderSampler	0.500000	0.809322	0.867347	0.333333	0.72403	0.805732

- **AdaBoost:**

Performing grid search to get the best parameters:

Base_estimator	DecisionTreeClassifier(max_depth=1), DecisionTreeClassifier(max_depth=2), DecisionTreeClassifier(max_depth=3), DecisionTreeClassifier(max_depth=4), DecisionTreeClassifier(max_depth=5)
N_estimators	10, 20, 50, 100, 200
Learning_rate	0.01, 0.1, 0.2, 0.5, 1.0

Parameters of Best Model:

Base_estimator	DecisionTreeClassifier(max_depth=1)
N_estimator	50
Learning_rate	0.5

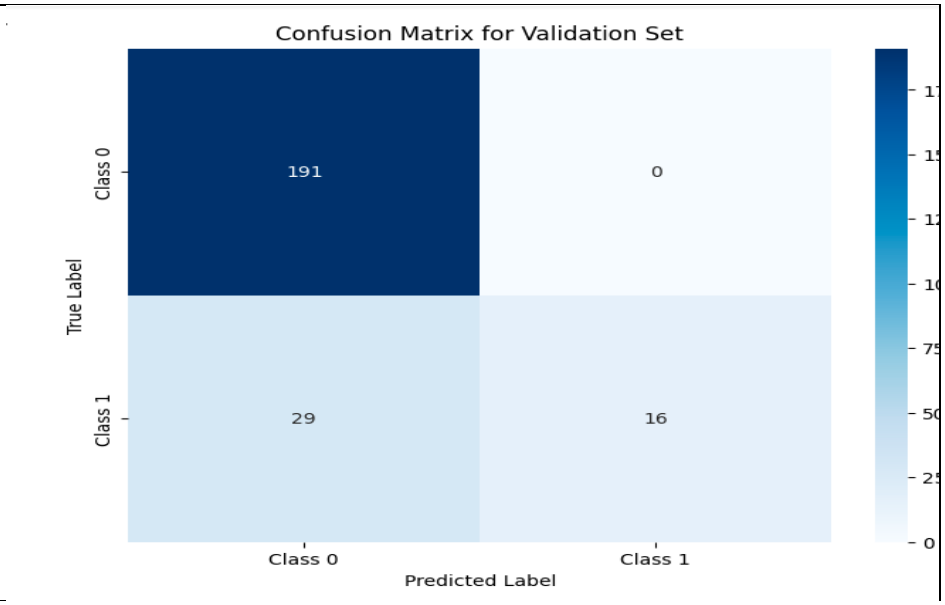
Evaluation metrics:

Training Accuracy	88.19%
Validation Accuracy	87.71%
Weighted F1-Score on training set	86.15%
Mean cross validation score	85.22%

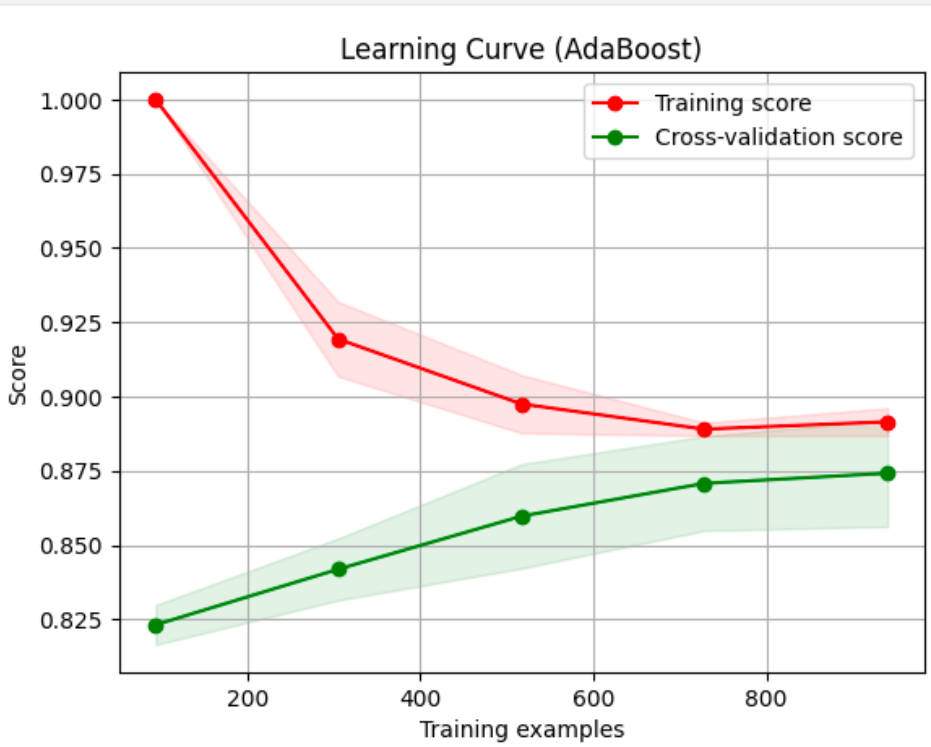
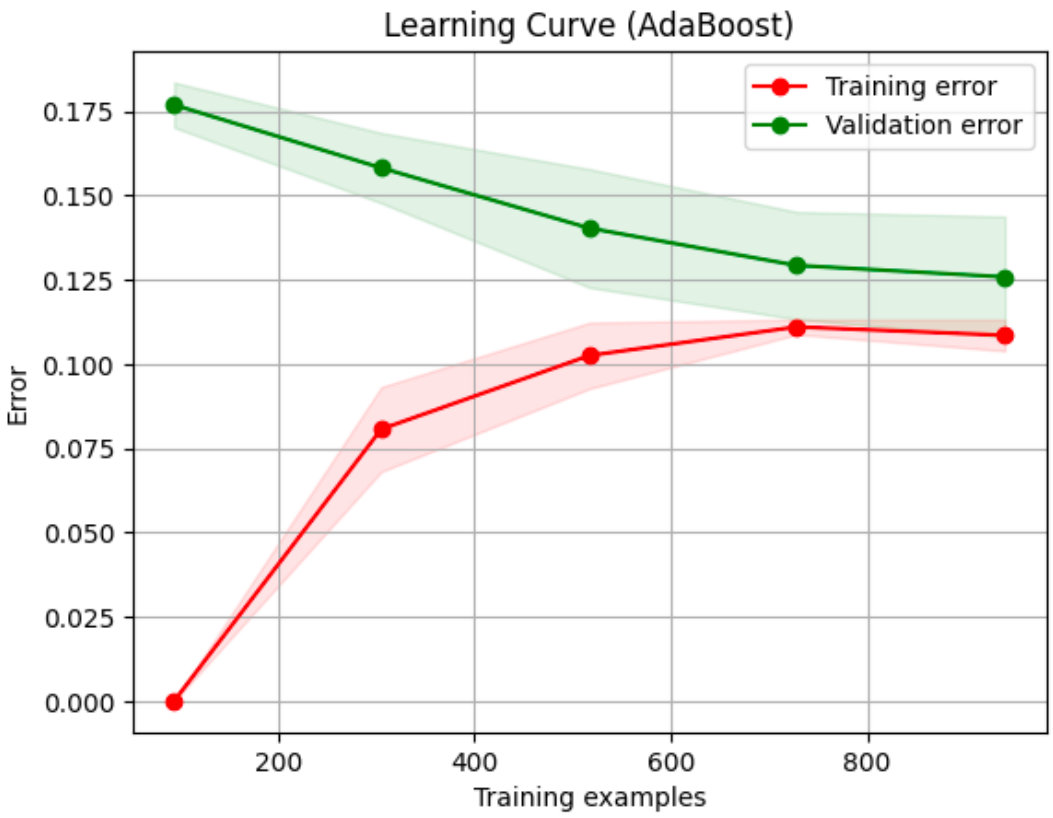
Classification report for validation set:

	precision	recall	f1-score	support
Class 0	0.87	1.00	0.93	191
Class 1	1.00	0.36	0.52	45
accuracy			0.88	236
macro avg	0.93	0.68	0.73	236
weighted avg	0.89	0.88	0.85	236

Confusion Matrix for validation set



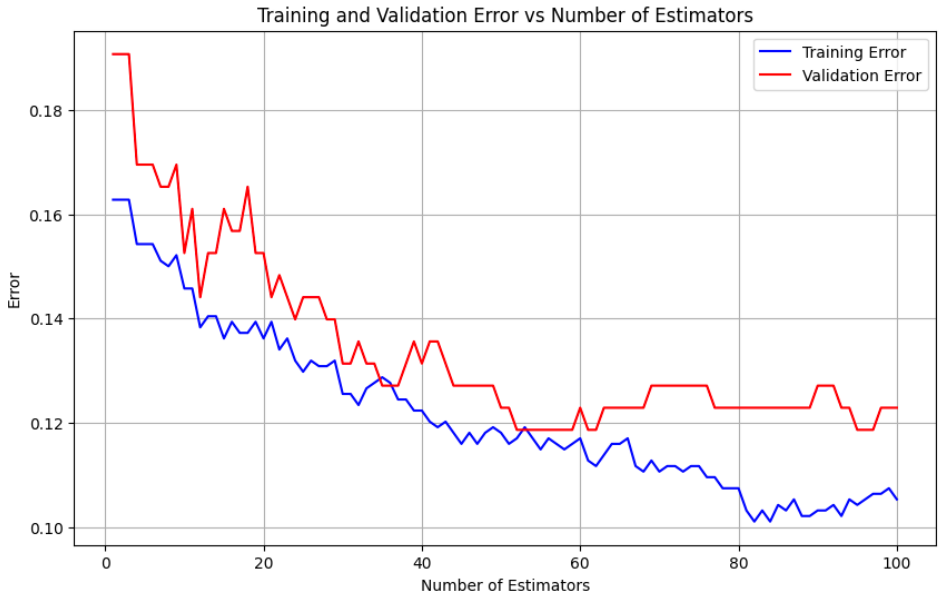
AdaBoost Plots:

Learning Curve (Scores)	
Learning Curve (Error)	

AdaBoost Learning Curve Insights:

- The decreasing training score suggests that as more training examples are provided, the model is exposed to a wider variety of instances and is learning to generalize better.
- This is a positive sign as it indicates that the model is not memorizing the training data but rather learning meaningful patterns.
- The increasing validation score indicates that the model's performance on unseen data is improving as more training examples are provided.
- This suggests that the model is generalizing well to new instances and is not overfitting to the training data.
- The small gap between the training and validation scores suggests that the model is not suffering from significant overfitting.
- The fact that both scores are increasing with a small gap indicates that the model is learning to generalize well to unseen data without excessively fitting to the training data.
- The fact that both the training and validation scores are eventually increasing slowly that the model's performance is stabilizing as more data is provided.
- This is a desirable outcome, indicating that additional data may not significantly improve the model's performance further.

Training & Validation errors vs number of learning estimators:



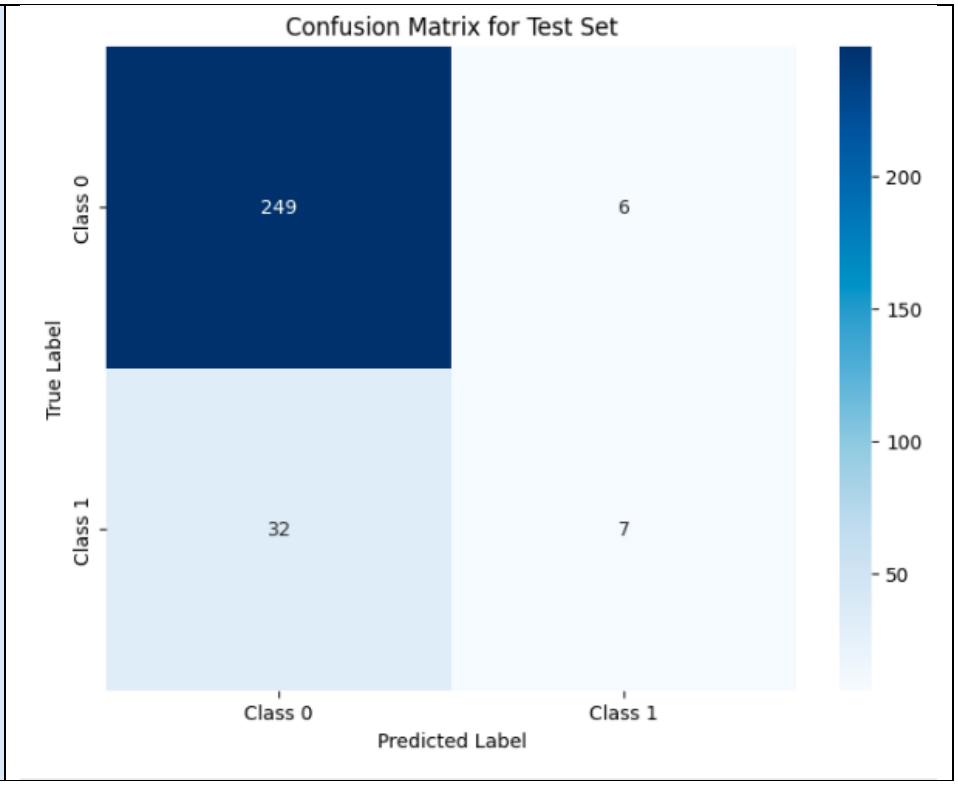
- Boosting is often robust to overfitting.
- Test set error decreases even after training error is almost zero

After Training over all the data (training + validation):

Testing Accuracy	87.07%
F1-Score on test set	84.15%

Classification Report for Test Set:

```
Classification Report for Test Set:
              precision    recall  f1-score   support

     Class 0       0.89      0.98      0.93       255
     Class 1       0.54      0.18      0.27        39

    accuracy                           0.87       294
   macro avg       0.71      0.58      0.60       294
weighted avg       0.84      0.87      0.84       294
```

Confusion Matrix for Testset	

- **SVM:**

Performing grid search to get the best parameters:

C	0.1,1,10,100
Gamma	0.001,0.01,0.1,1,10,100
Kernel	Linear, Poly, rbf, Sigmoid

Parameters of Best Model:

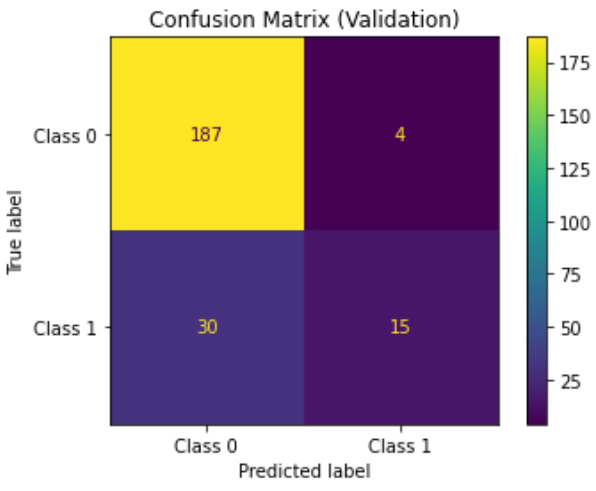
C	100
Gamma	0.011
Kernel	Linear

Evaluation metrics:

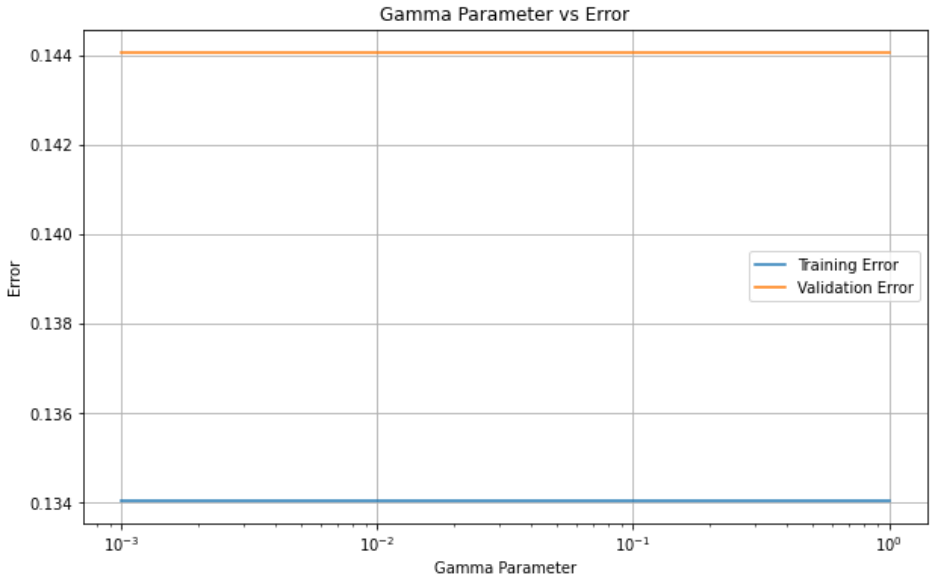
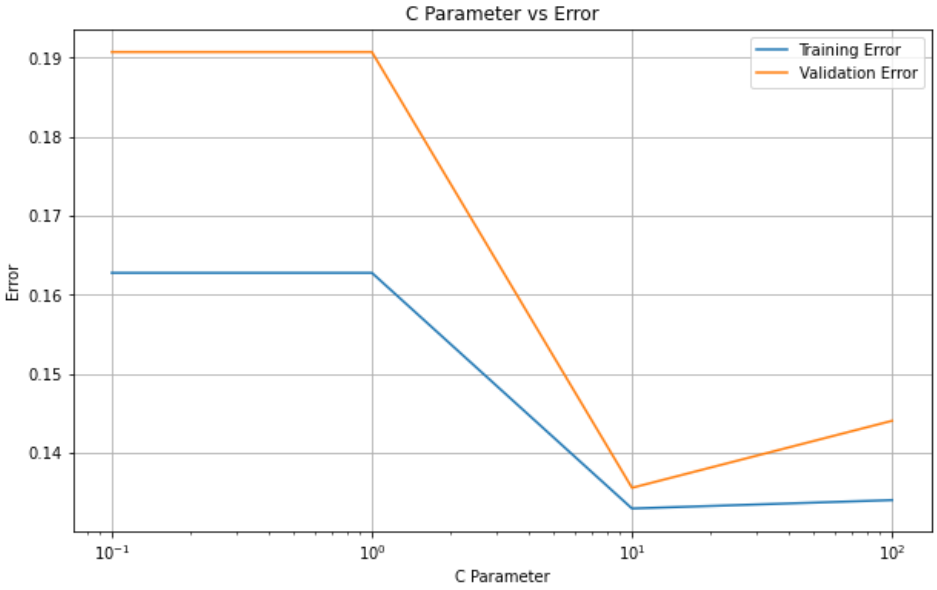
Training Accuracy	0.8659
Validation Accuracy	0.8559
F1-Score on validation	0.8313

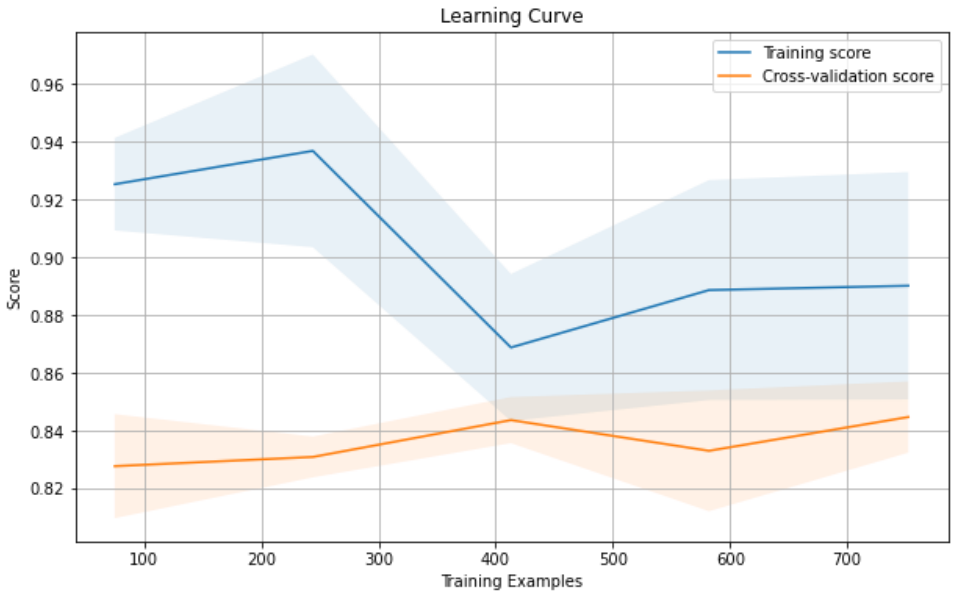
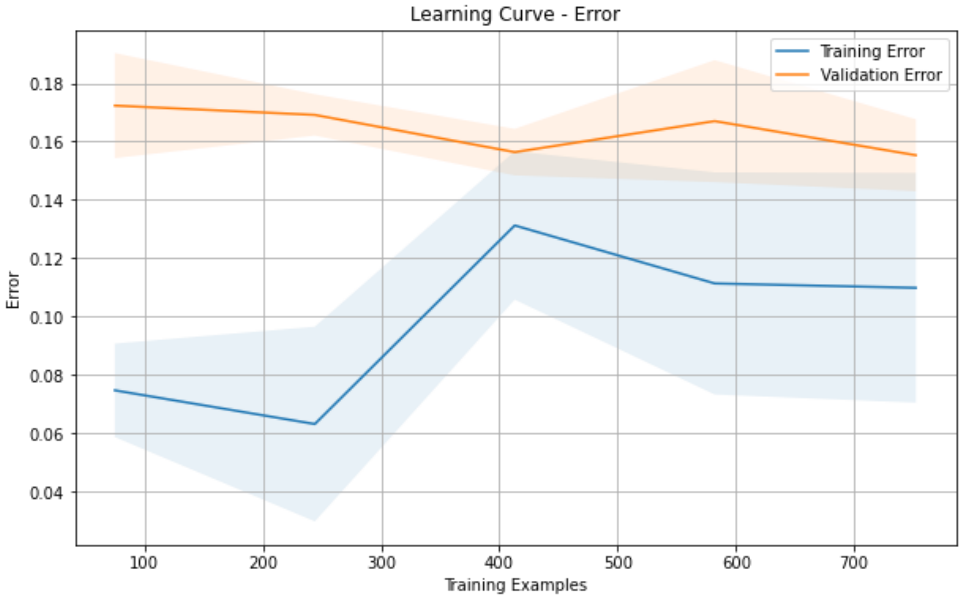
Classification report for validation set:

	precision	recall	f1-score	support
0	0.86	0.98	0.92	191
1	0.79	0.33	0.47	45
accuracy			0.86	236
macro avg	0.83	0.66	0.69	236
weighted avg	0.85	0.86	0.83	236

Confusion Matrix for validation set	

SVM Plots:

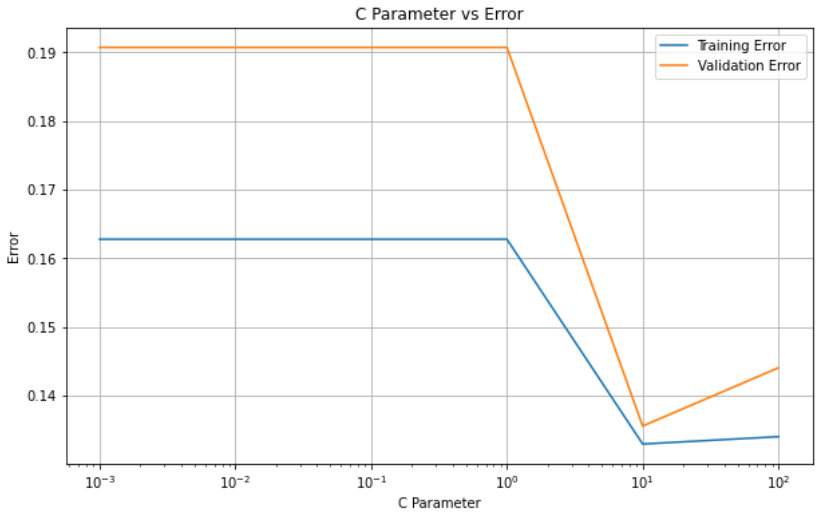
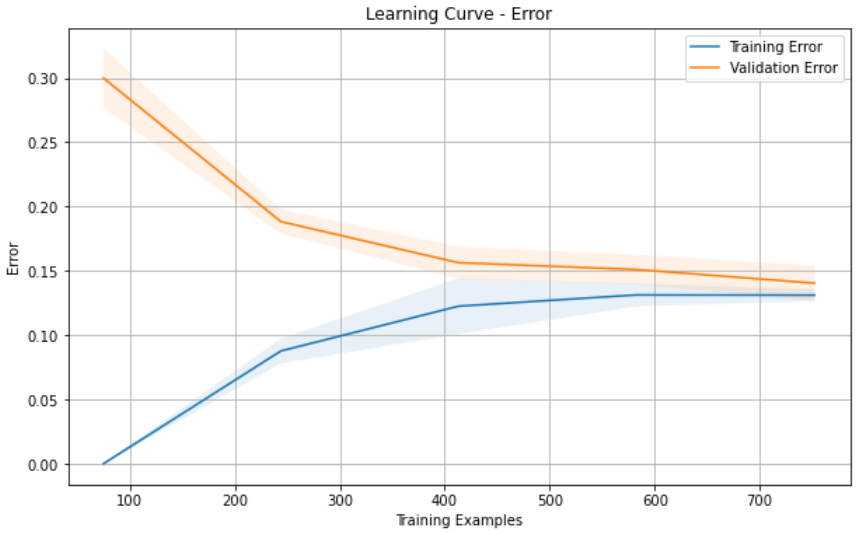
Gamma vs Error	 Changing the gamma parameter does not significantly affect the model's performance on either the training or validation data.
C vs Error	 **-Initial Decrease**: At lower values of C, both the training and validation errors decrease. This indicates that the model is underfitting. **-Optimal C Value**: There is a point where the validation error reaches its minimum value, indicating the best regularization parameter (C) for the model where the model achieves the best balance between bias and variance, leading to optimal generalization to unseen data. **-Increase after Optimal C**: Beyond the optimal C value, both the training and validation errors start to increase. This is because higher values of C lead to overfitting, where the model becomes too complex.

Learning Curve(scores):	
Learning Curve(errors):	 - The decreasing validation score indicates that the model's performance on unseen data is not improving as more training examples are provided. - This suggests that the model is not generalizing well to new instances and is not overfitting to the training data. - The gap between the training and validation scores is not small which suggests that the model is suffering from overfitting.

Overall, the learning curve suggests that the model's performance improves with more training examples up to a certain point, after which further increases in the training set size may lead to overfitting. It highlights the importance of balancing model complexity and dataset size for optimal performance.

- Because of this overfitting i started to perform **regularization** by choosing the optimal value for C.

I plotted the C vs Error again but this time i appended smaller values to the C list. Then I plotted the Learning curve again to ensure that the overfitting is solved.

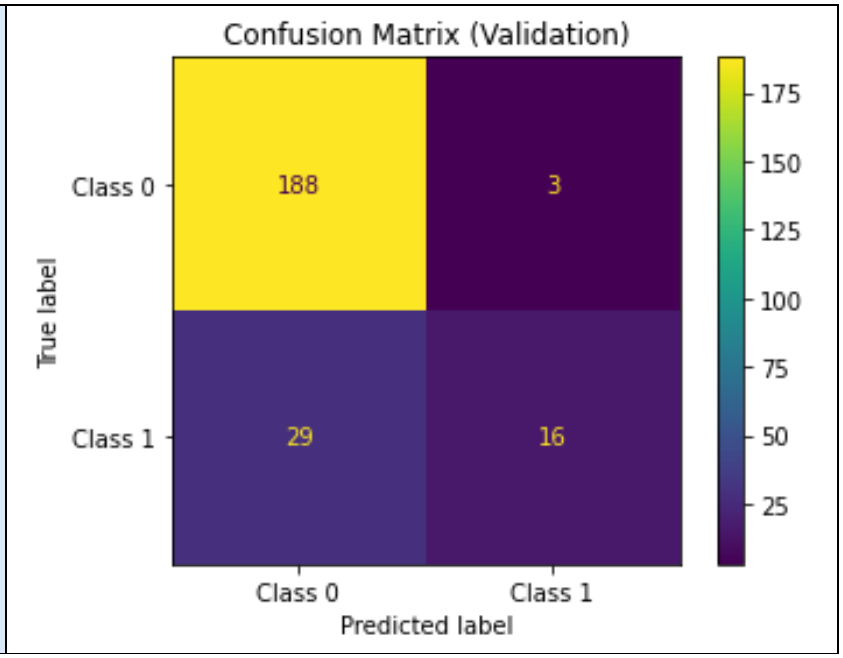
C vs Error	 **Optimal C value=10**
Learning Curve	 Overfitting Solved.

Evaluation metrics After Regularization

Training Accuracy	0.86702
Validation Accuracy	0.8644
F1-Score on validation	0.84118

Classification report for validation set after regularization:

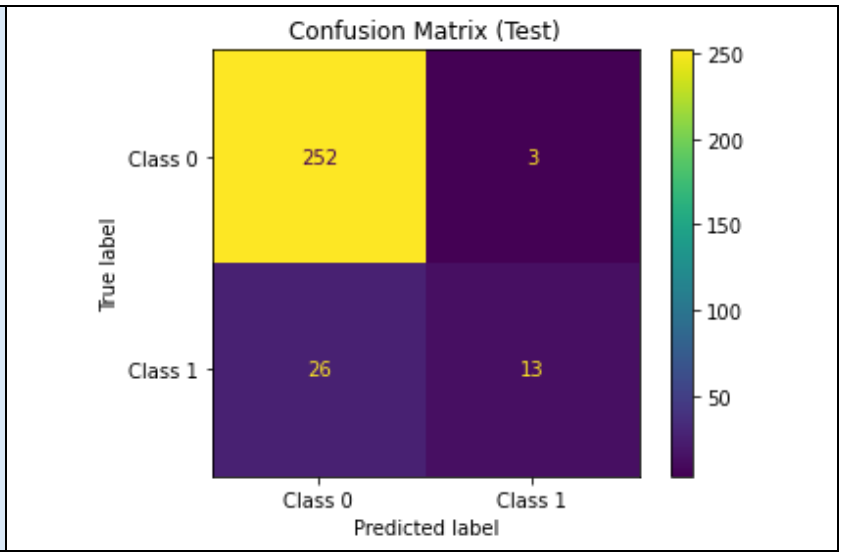
```
Classification Report (Validation):
              precision    recall  f1-score   support

           0       0.87      0.98      0.92       191
           1       0.84      0.36      0.50        45

    accuracy                           0.86       236
   macro avg       0.85      0.67      0.71       236
weighted avg       0.86      0.86      0.84       236
```

Confusion Matrix for validation set After Regularization	

Testing Accuracy	0.90136
F1_Score on test set	0.88286

Classification Report (Test):

	precision	recall	f1-score	support
0	0.91	0.99	0.95	255
1	0.81	0.33	0.47	39
accuracy			0.90	294
macro avg	0.86	0.66	0.71	294
weighted avg	0.89	0.90	0.88	294

Confusion Matrix for Test set	

- ## Linear SVM:

Performing grid search to get the best parameters:

C	0.1,1,10,100
Penalty	L1, L2
Loss	hinge,log,exponential
Dual	True,false
Tol	1e-4,1e-3,1e-2

Parameters of Best Model:

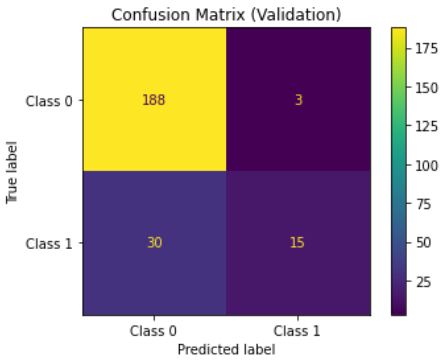
C	10
Penalty	L2
Loss	hinge
Dual	True
Tol	1e-2

Evaluation metrics:

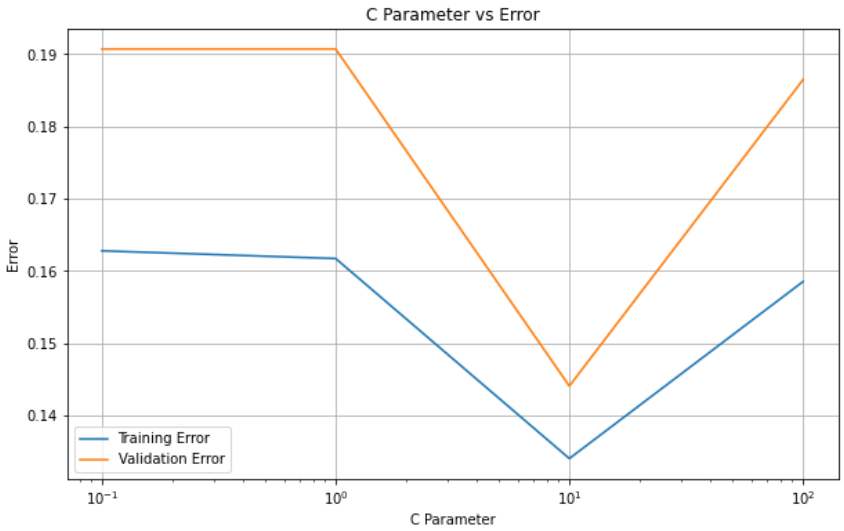
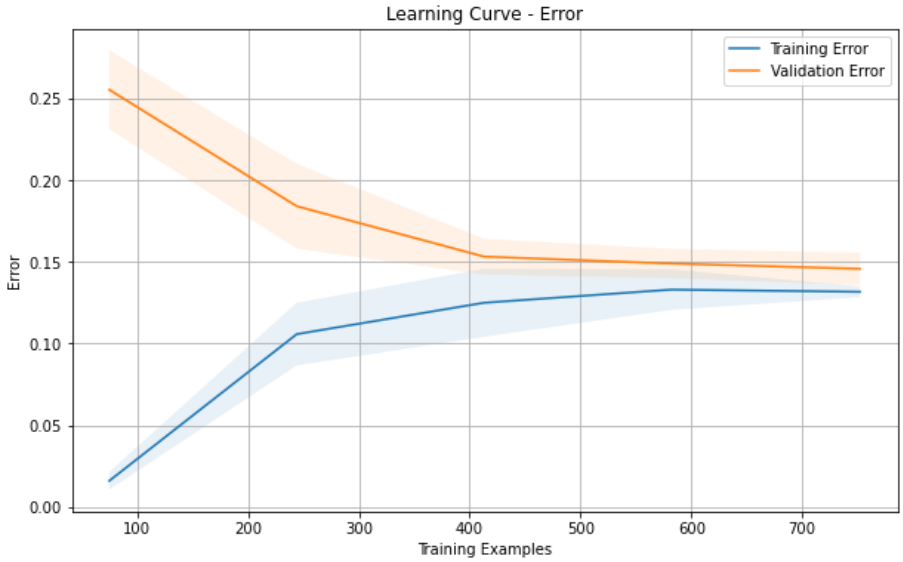
Training Accuracy	0.8659
Validation Accuracy	0.8602
F1-Score on validation	0.83757

Classification report for validation set:

	precision	recall	f1-score	support
0	0.86	0.98	0.92	191
1	0.83	0.33	0.48	45
accuracy			0.86	236
macro avg	0.85	0.66	0.70	236
weighted avg	0.86	0.86	0.83	236

Confusion Matrix for validation set	

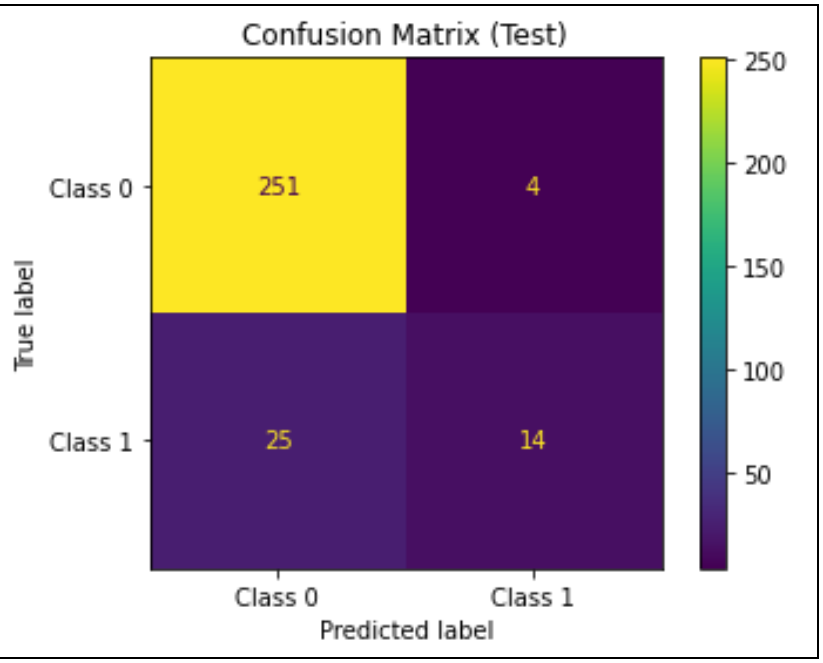
LinearSVM plots:

C vs Error	 **-Initial Decrease**: At lower values of C, both the training and validation errors decrease. This indicates that the model is underfitting. **-Optimal C Value**: There is a point where the validation error reaches its minimum value, indicating the best regularization parameter(C)for the model where the model achieves the best balance between bias and variance, leading to optimal generalization. **-Increase after Optimal C**: Both the training and validation errors start to increase. This is because higher values of C lead to overfitting, where the model becomes too complex.
Learning Curve	 The model initially struggles to generalize well to the validation set, as indicated by the increasing training error and decreasing validation error. However, as the training set size increases, the model begins to generalize better, leading to both errors stabilizing. This indicates that the model has learned the underlying patterns in the data and is not overfitting.

Testing Accuracy	0.90136
F1_Score on Test set	0.88514

Classification Report (Test):

	precision	recall	f1-score	support
0	0.91	0.99	0.95	255
1	0.81	0.33	0.47	39
accuracy			0.90	294
macro avg	0.86	0.66	0.71	294
weighted avg	0.89	0.90	0.88	294

Confusion Matrix for Test set	

- ## LogisticRegression:

Performing grid search to get the best parameters:

C	0.1, 1.0, 10.0
Max_iter	100, 200, 300, 400, 500
Solver	'sag', 'saga'

Parameters of Best Model:

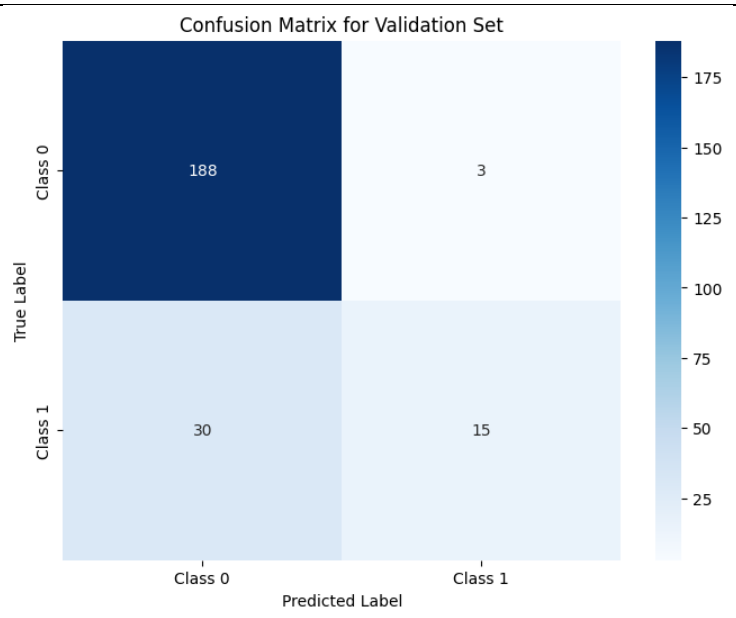
C	1
Max_iter	100
Solver	sag

Evaluation metrics:

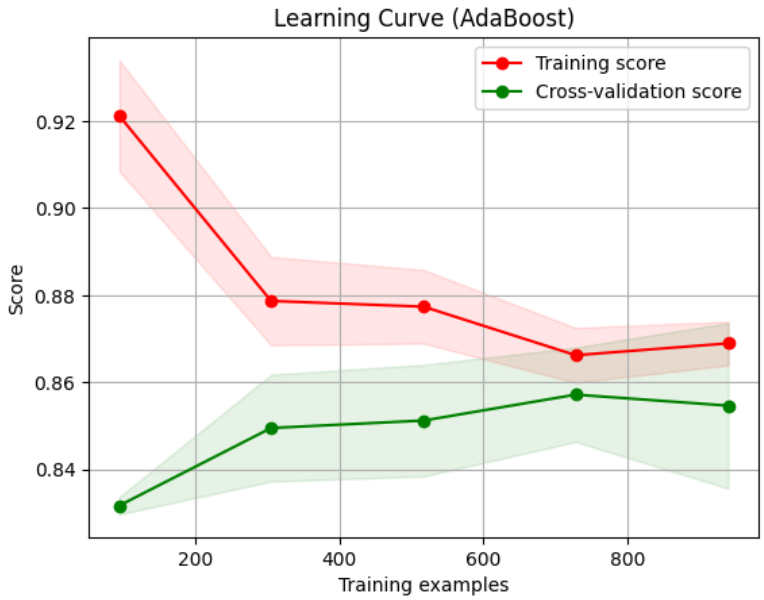
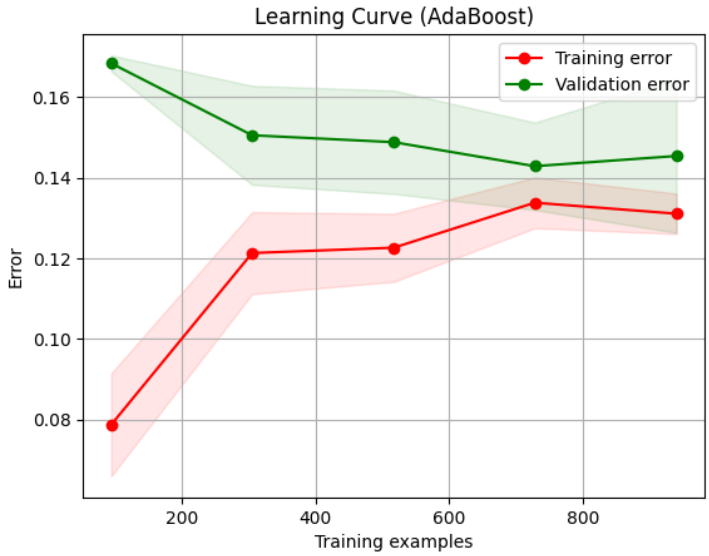
Training Accuracy	0.863829
F1-Score on training	0.8339
Validation Accuracy	0.860169
F1-Score on validation	0.8348

Classification report for validation set:

```
              precision    recall  f1-score   support

     Class 0       0.86      0.98      0.92       191
     Class 1       0.83      0.33      0.48        45

    accuracy                           0.86       236
   macro avg       0.85      0.66      0.70       236
weighted avg       0.86      0.86      0.83       236
```

Confusion Matrix for Validation set	

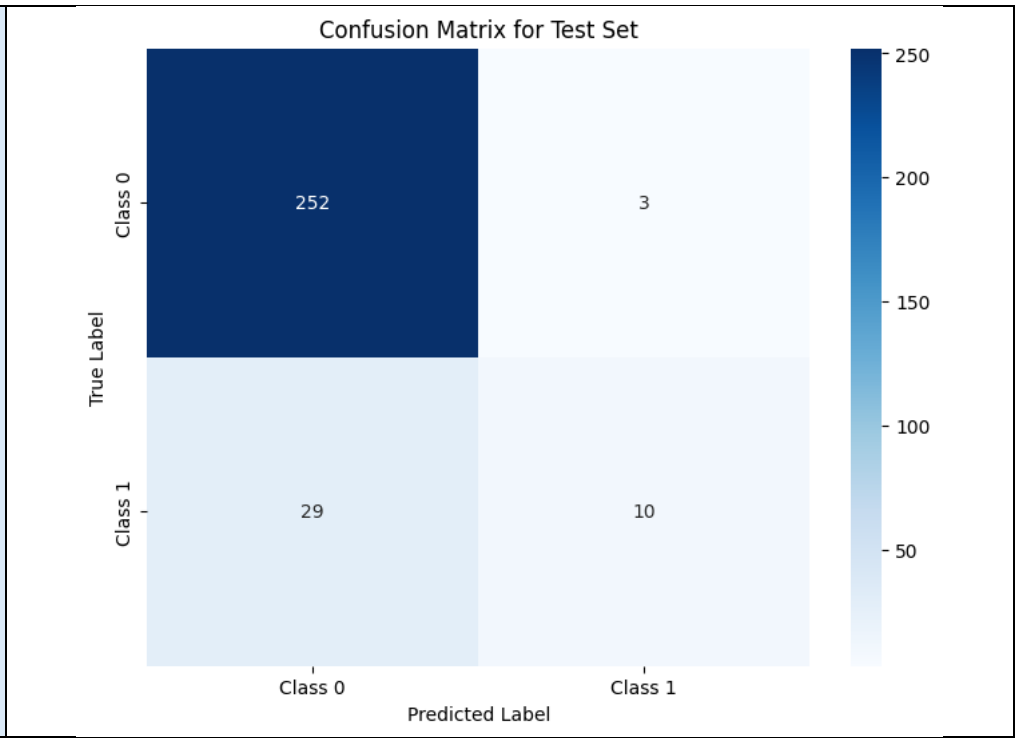
Logistic Regression plots:

Learning Curve (Scores)	 1-**The decreasing training score** suggests that as more training examples are provided, the model is exposed to a wider variety of instances and is learning to generalize better. This is a positive sign as it indicates that the model is not memorizing the training data but rather learning meaningful patterns. 2-**The increasing validation score** indicates that the model's performance on unseen data is improving as more training examples are provided. This suggests that the model is generalizing well to new instances and is not overfitting to the training data. 3-**The small gap between the training and validation** scores suggests that the model is not suffering from significant overfitting. The fact that both scores are increasing with a small gap indicates that the model is learning to generalize well to unseen data without excessively fitting to the training data.
Learning Curve (errors)	

Testing Accuracy	0.891156
F1_Score on Test set	0.866585

Classification Report for Test Set:

	precision	recall	f1-score	support
Class 0	0.90	0.99	0.94	255
Class 1	0.77	0.26	0.38	39
accuracy			0.89	294
macro avg	0.83	0.62	0.66	294
weighted avg	0.88	0.89	0.87	294

Confusion Matrix for Test set	

Models Comparison:

Model	Test Accuracy	Weighted F1_score on Test set	Validation Accuracy	Weighted F1_Score on validation set	Training Accuracy
ZeroR	0.8673	0.8057	0.8093	0.724	0.8322
AdaBoost	0.8707	0.8415	0.8771	0.8522	0.8819
SVM	0.90136	0.88286	0.84118	0.864406	0.8670
LinearSVM	0.90136	0.88514	0.860169	0.83757	0.868085
LogisticRegression	0.8911	0.8665	0.8601	0.8348	0.8638