Loan Status Prediction

Upload CSV dataset (loan dataset)

Drag and drop file here
Limit 200MB per file • CSV

Browse files

loan_approval_dataset.csv 375.3KB

Data shape: `(4269, 13)`

Introduction

This Loan Status Prediction dashboard is built using a structured loan dataset that contains key financial, demographic, and credit-related information of customers who applied for a loan.
The dataset typically includes attributes such as:

- Applicant Income
- Co-applicant Income
- Loan Amount
- Credit History
- Education Level
- Employment Type
- Property Area
- Gender and Marital Status

Using these features, the dashboard aims to analyze patterns that influence loan approval and apply machine-learning models to predict whether an application is likely to be **Approved** or **Rejected**.

This dashboard helps users to:

- Understand the dataset with quick visual insights
- Explore important customer attributes
- Apply feature engineering and SelectKBest filtering
- Train multiple machine-learning models
- Compare model accuracy using visual charts

By transforming raw loan data into meaningful insights and predictions, this project demonstrates how machine learning can support risk assessment, automate decision-making, and improve efficiency in the loan approval process.

Dataset Information

- **Rows:** 4269
- **Columns:** 13
- **First 12 Columns:** loan_id, no_of_dependents, education, self_employed, income_annum, loan_amount, loan_term, cibil_score, residential_assets_value, commercial_assets_value, luxury_assets_value, bank_asset_value...

Step 1 — Quick EDA

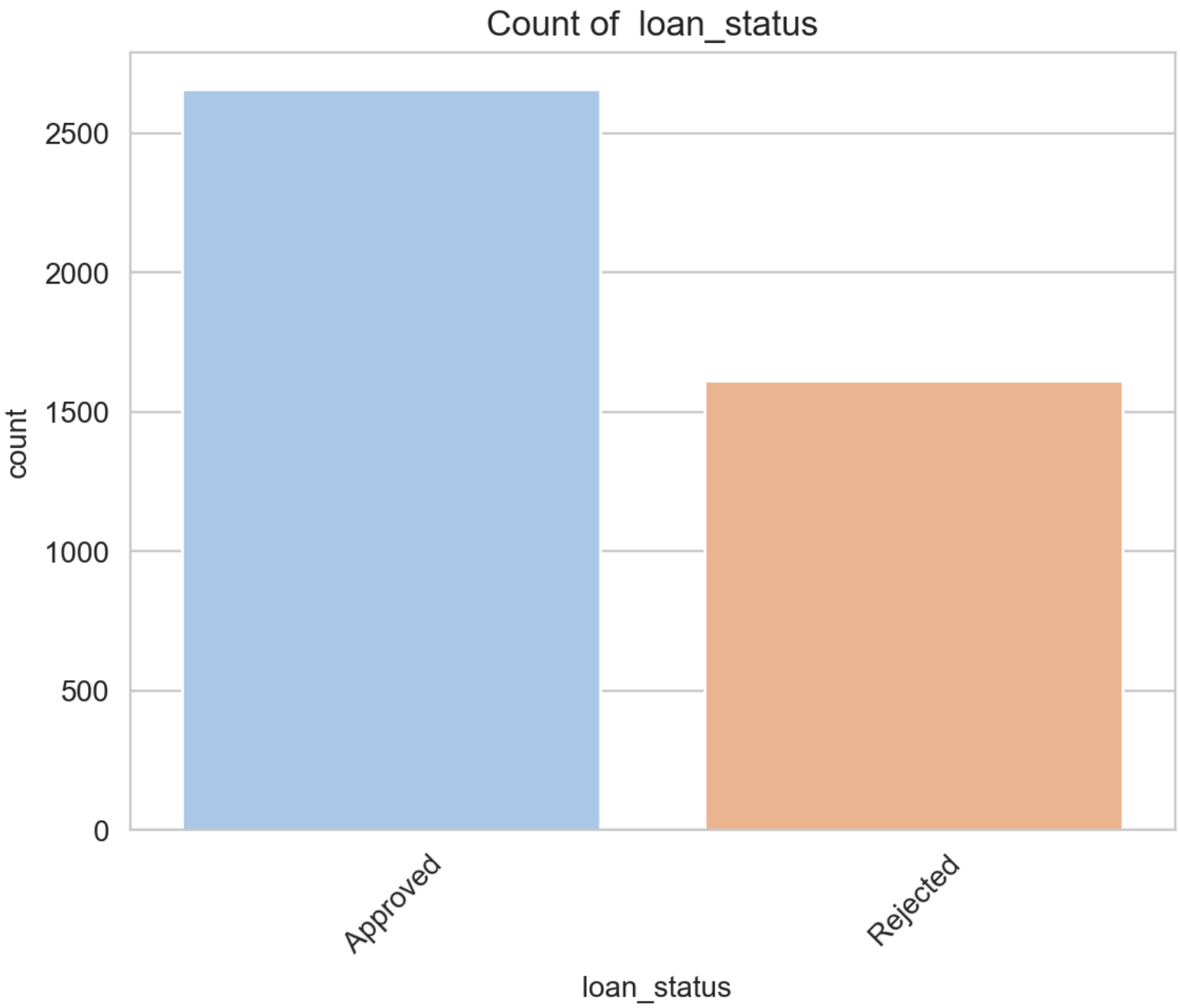
Show dataset head

	loan_id	no_of_dependents	education	self_employed	income_annum	loan_amount	loan_term	cibil_score	residential_assets_value	comme
0	1	2	Graduate	No	9600000	29900000	12	778	2400000	
1	2	0	Not Graduate	Yes	4100000	12200000	8	417	2700000	
2	3	3	Graduate	No	9100000	29700000	20	506	7100000	
3	4	3	Graduate	No	8200000	30700000	8	467	18200000	
4	5	5	Not Graduate	Yes	9800000	24200000	20	382	12400000	

📊 Visualization

Choose a categorical column for bar chart

loan_status



Step 2 — Features & Target

Columns to drop (irrelevant):

loan_id × residential_asse… ×

Dropped columns. New shape: `(4269, 11)`

Target column (loan_status)

loan_status

Input features (choose at least 1)

no_of_dependents × education × self_employed × income_annum × loan_amount × loan_term × commercial_ass… ×
luxury_assets_v… × bank_asset_value × cibil_score ×

Step 3 — Preprocessing & Feature Engineering

3.1 Label-encoding categorical columns...

Encoded categorical columns:


```
[
  0 : " education"
  1 : " self_employed"
]
```


Numeric columns available: `10`

3.5 Assembling candidate features...

Candidate feature matrix shape (train): `(3201, 10)`

Step 4 — Feature Selection (SelectKBest)

Selected top 10 features.

Step 5 — Train (3 models)

🚀 Start training

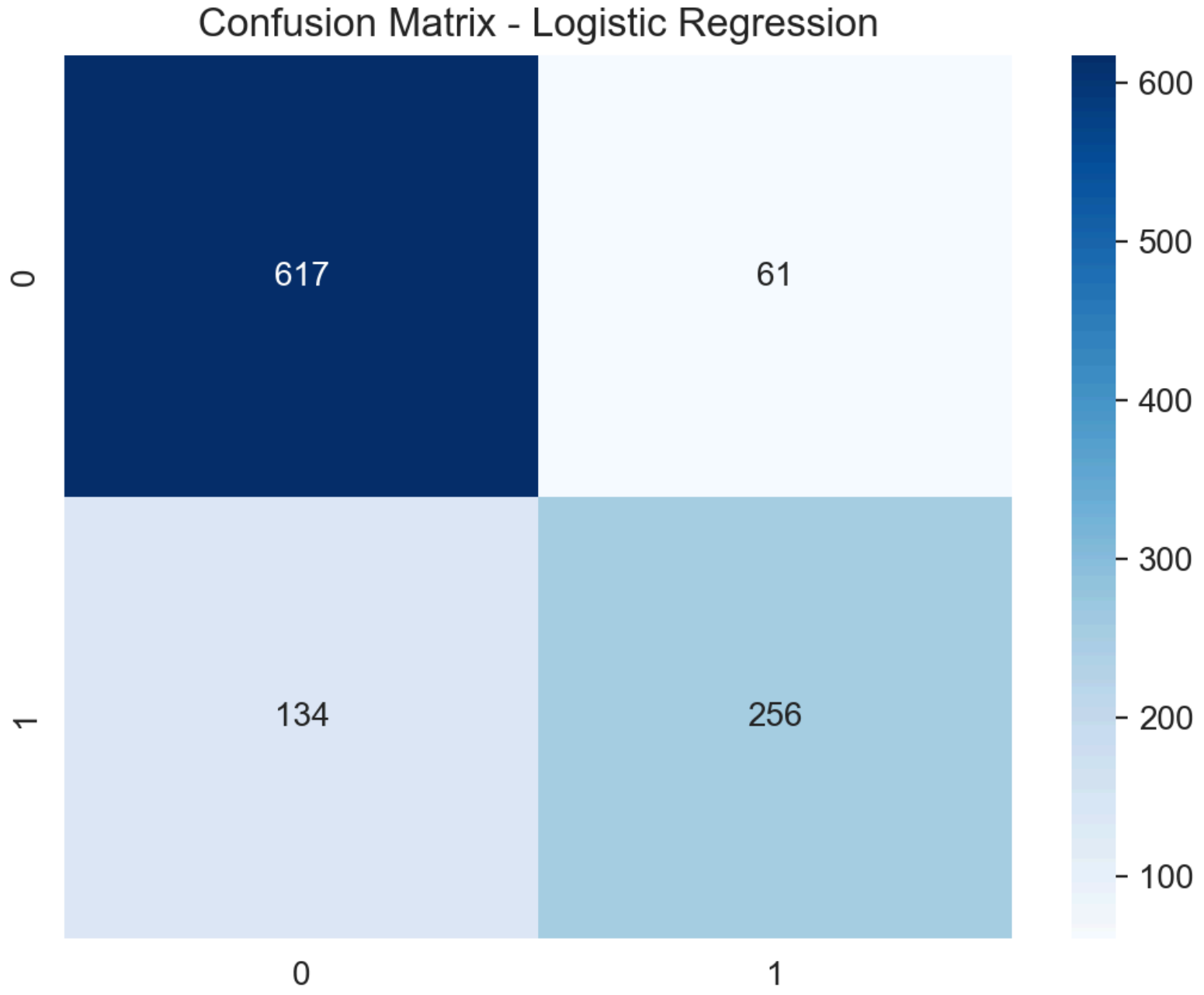
Training model: **Logistic Regression** (1/3)

Logistic Regression finished — Accuracy: 0.817 Time: 0.2s

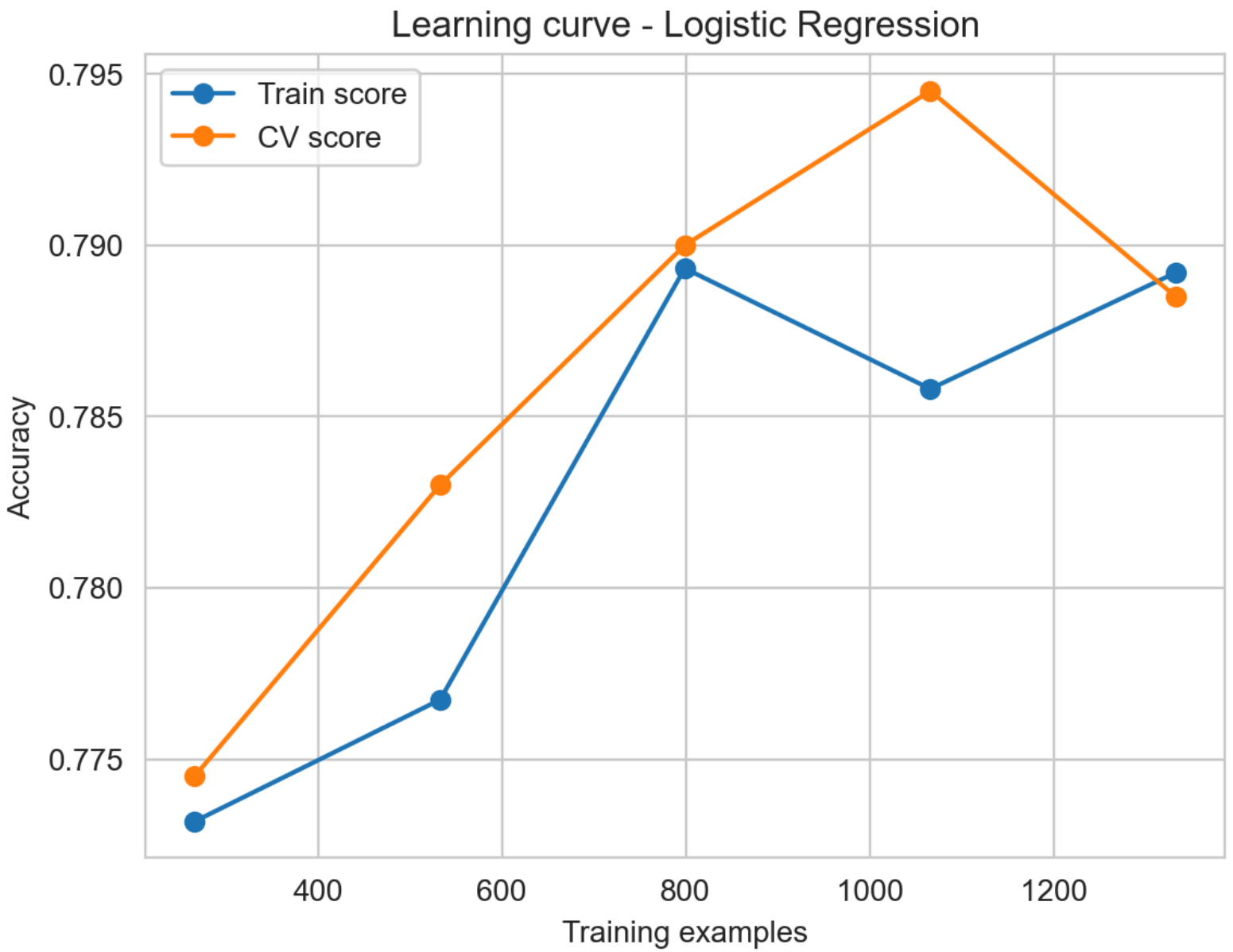
Accuracy	Precision	Recall	F1-Score
0.817	0.816	0.817	0.813

Classification report (per class):

	precision	recall	f1-score	support
0	0.8216	0.91	0.8635	678
1	0.8076	0.6564	0.7242	390
accuracy	0.8174	0.8174	0.8174	0.8174
macro avg	0.8146	0.7832	0.7939	1068
weighted avg	0.8165	0.8174	0.8127	1068



Learning curve:



Download model: Logistic Regression.joblib

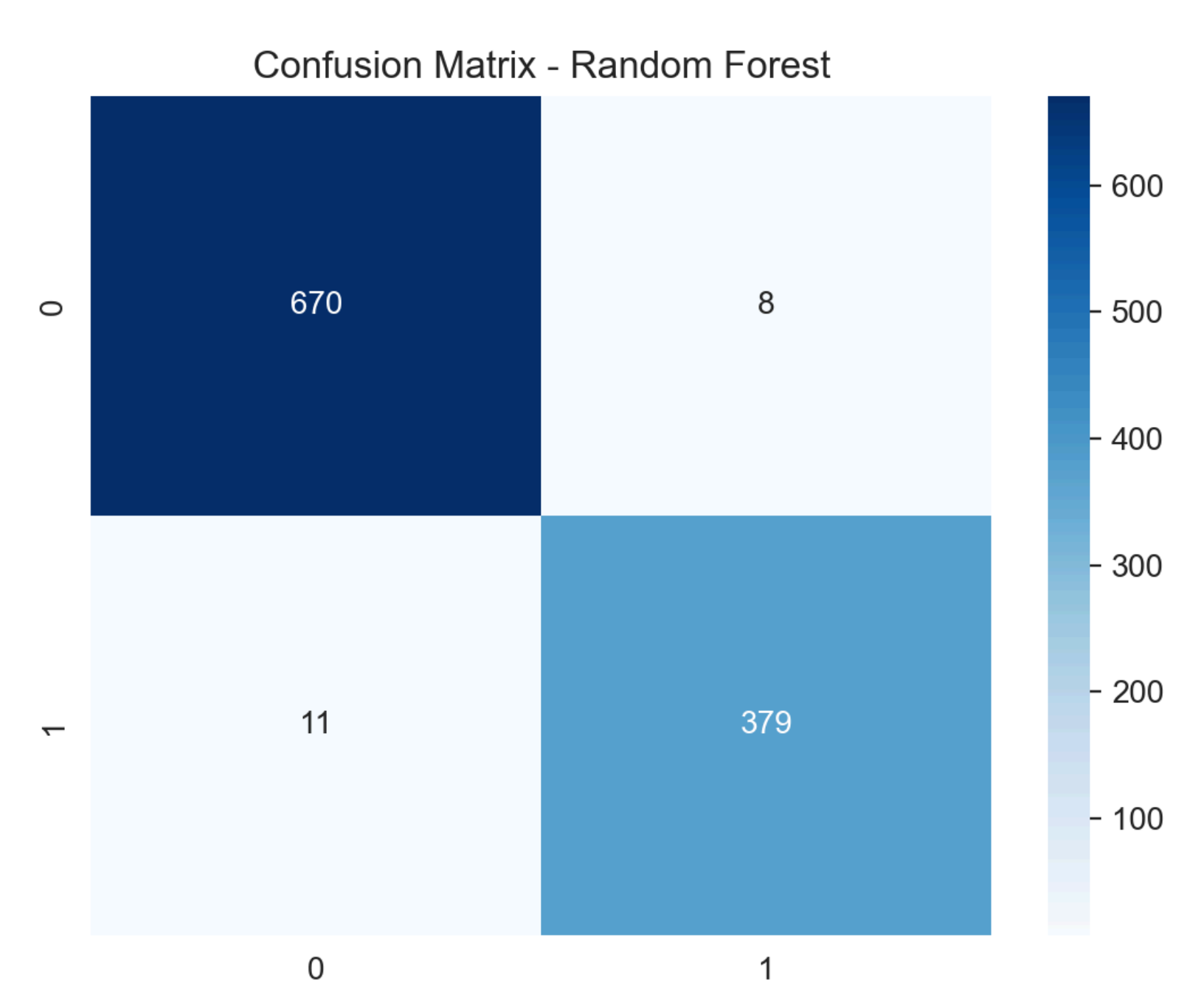
Training model: **Random Forest** (2/3)

Random Forest finished — Accuracy: 0.982 Time: 1.2s

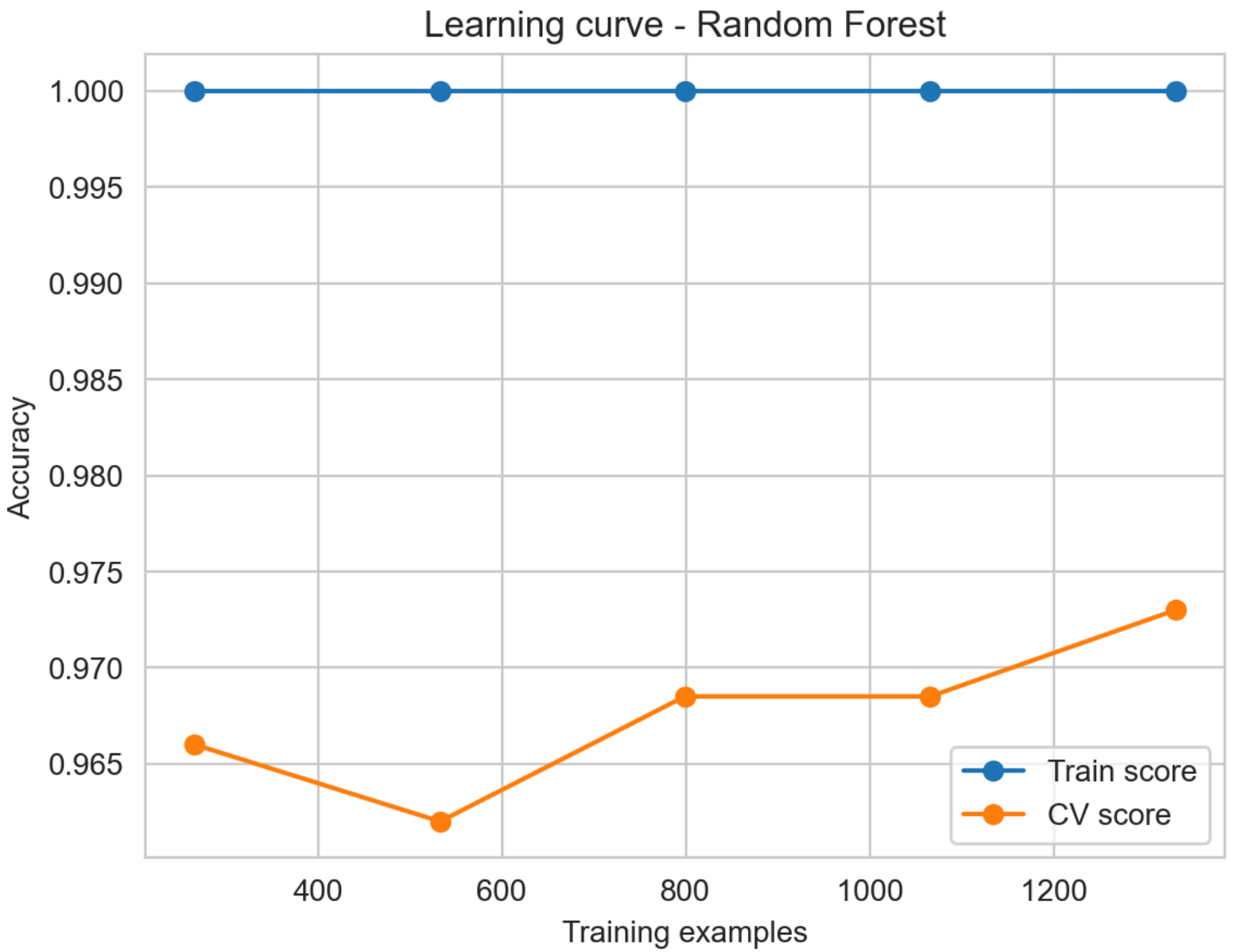
Accuracy	Precision	Recall	F1-Score
0.982	0.982	0.982	0.982

Classification report (per class):

	precision	recall	f1-score	support
0	0.9838	0.9882	0.986	678
1	0.9793	0.9718	0.9755	390
accuracy	0.9822	0.9822	0.9822	0.9822
macro avg	0.9816	0.98	0.9808	1068
weighted avg	0.9822	0.9822	0.9822	1068



Learning curve:



Download model: Random Forest.joblib

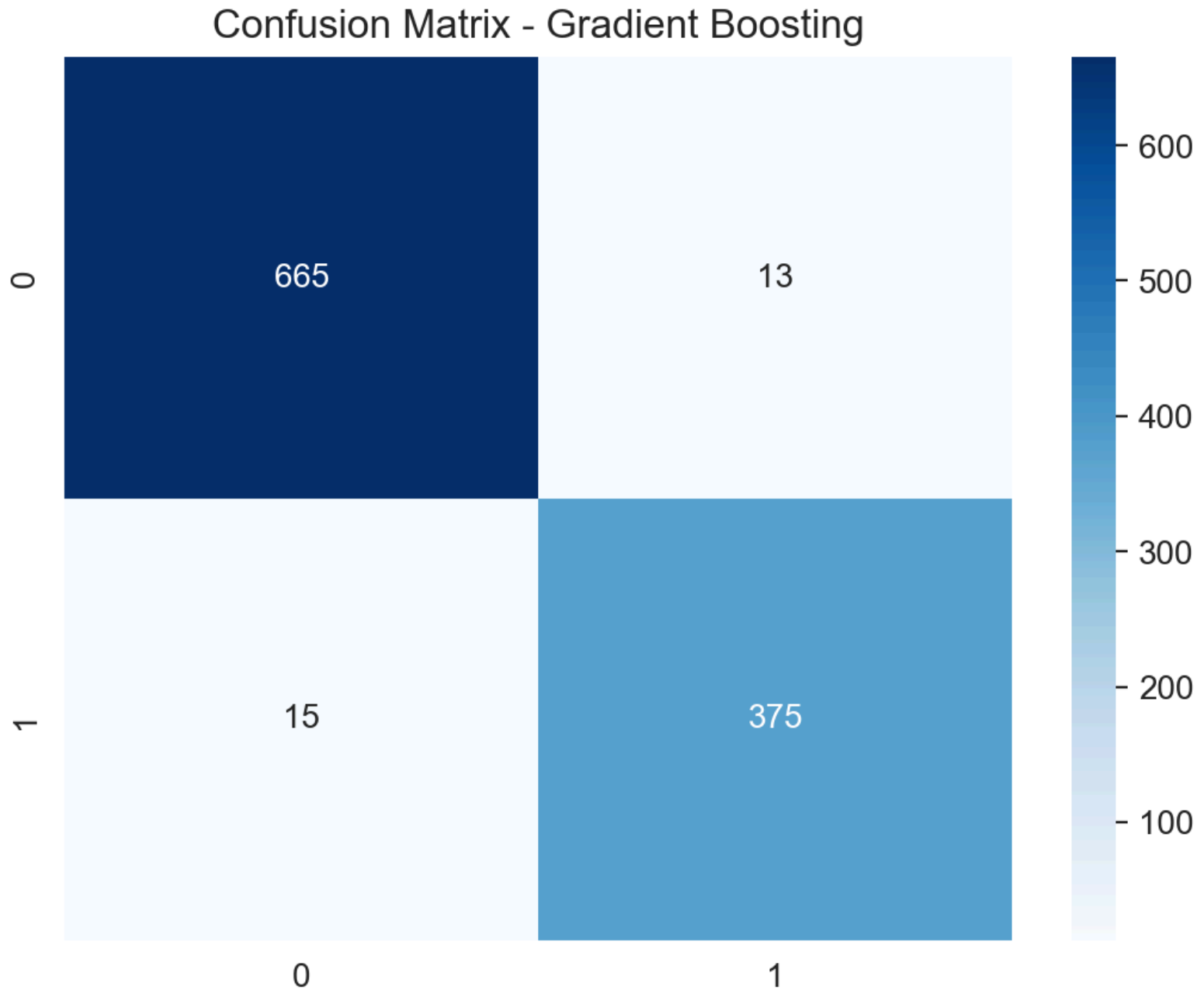
Training model: **Gradient Boosting** (3/3)

Gradient Boosting finished — Accuracy: 0.974 Time: 1.1s

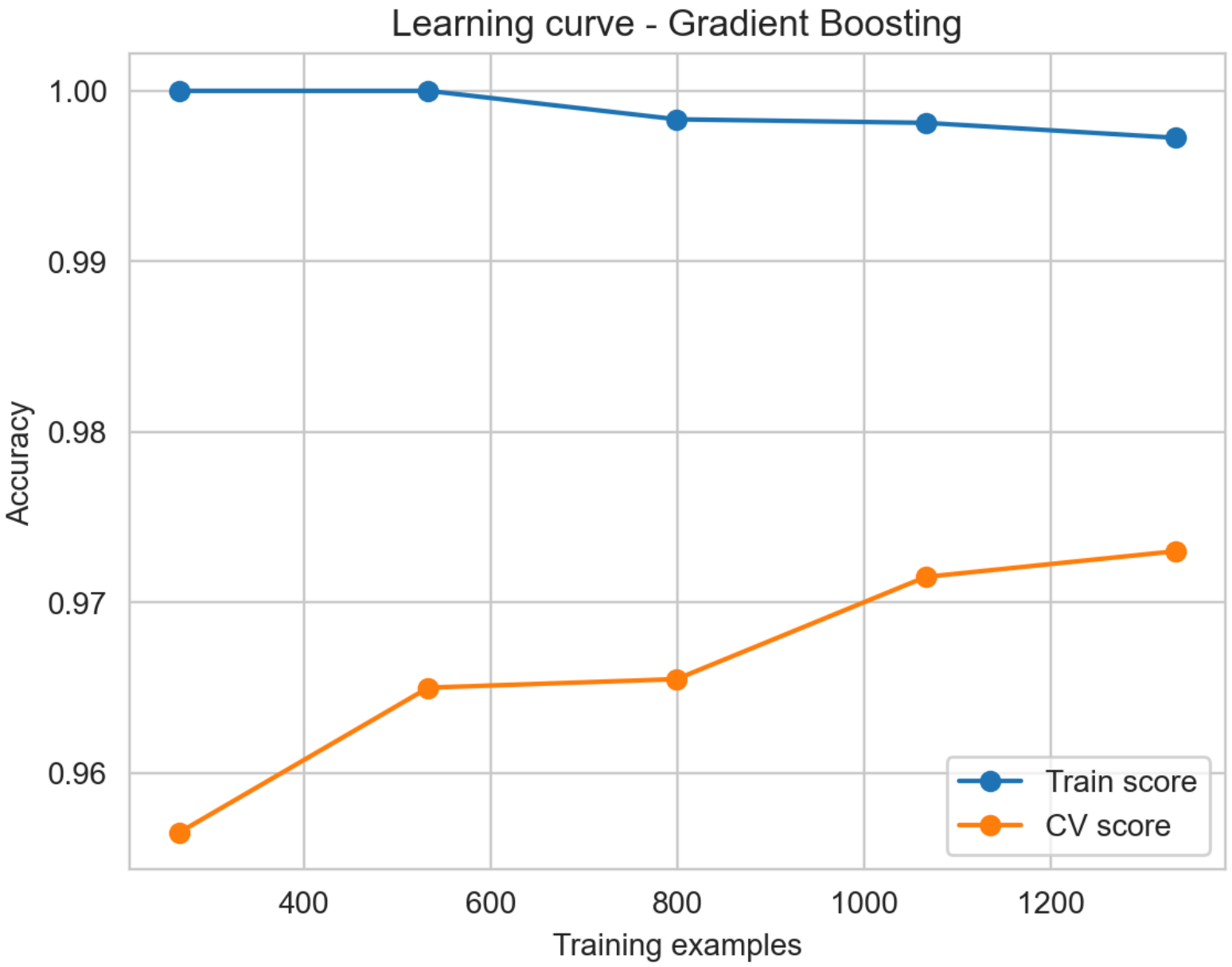
Accuracy	Precision	Recall	F1-Score
0.974	0.974	0.974	0.974

Classification report (per class):

	precision	recall	f1-score	support
0	0.9779	0.9808	0.9794	678
1	0.9665	0.9615	0.964	390
accuracy	0.9738	0.9738	0.9738	0.9738
macro avg	0.9722	0.9712	0.9717	1068
weighted avg	0.9738	0.9738	0.9738	1068



Learning curve:



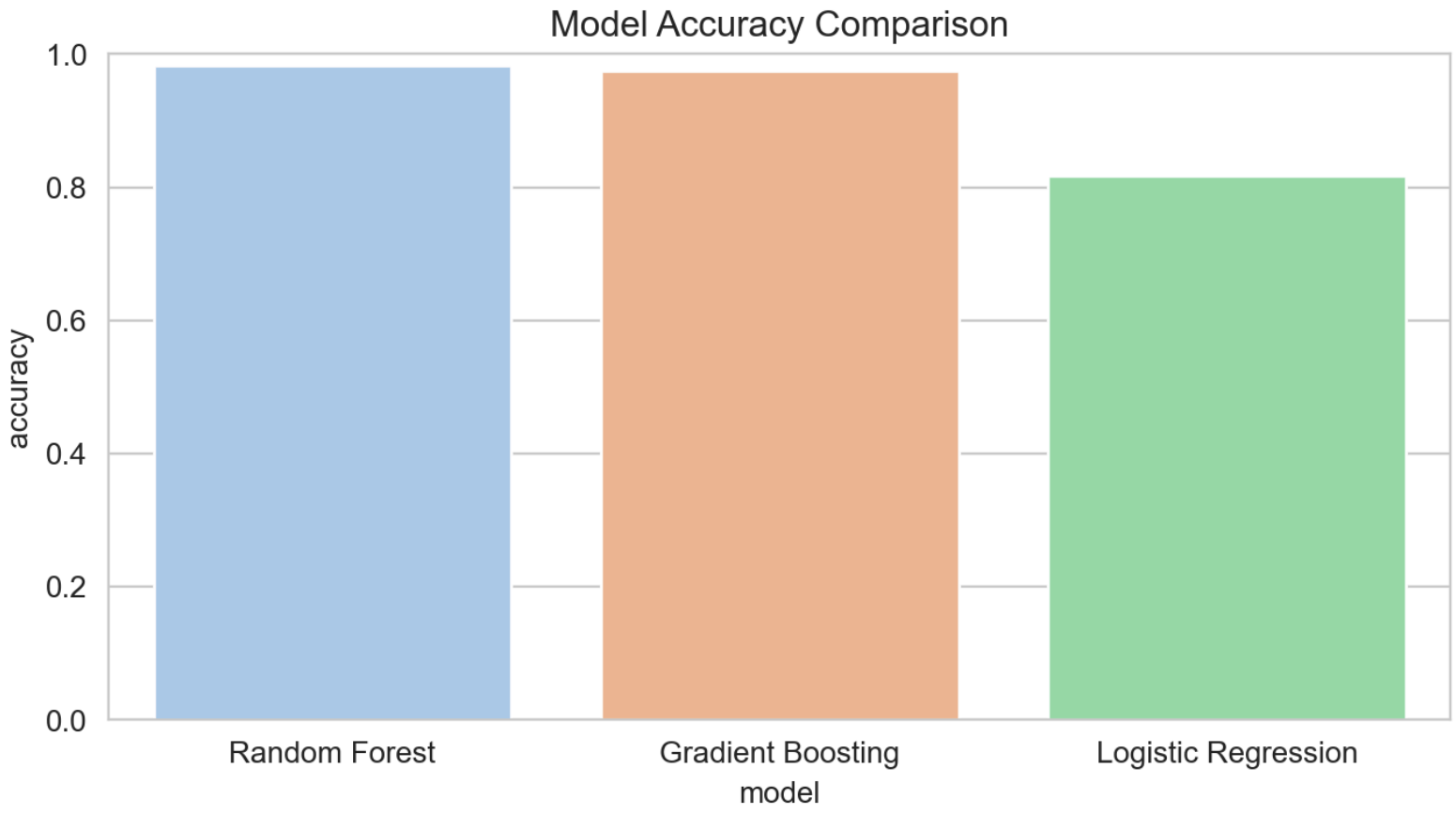
Download model: Gradient Boosting.joblib

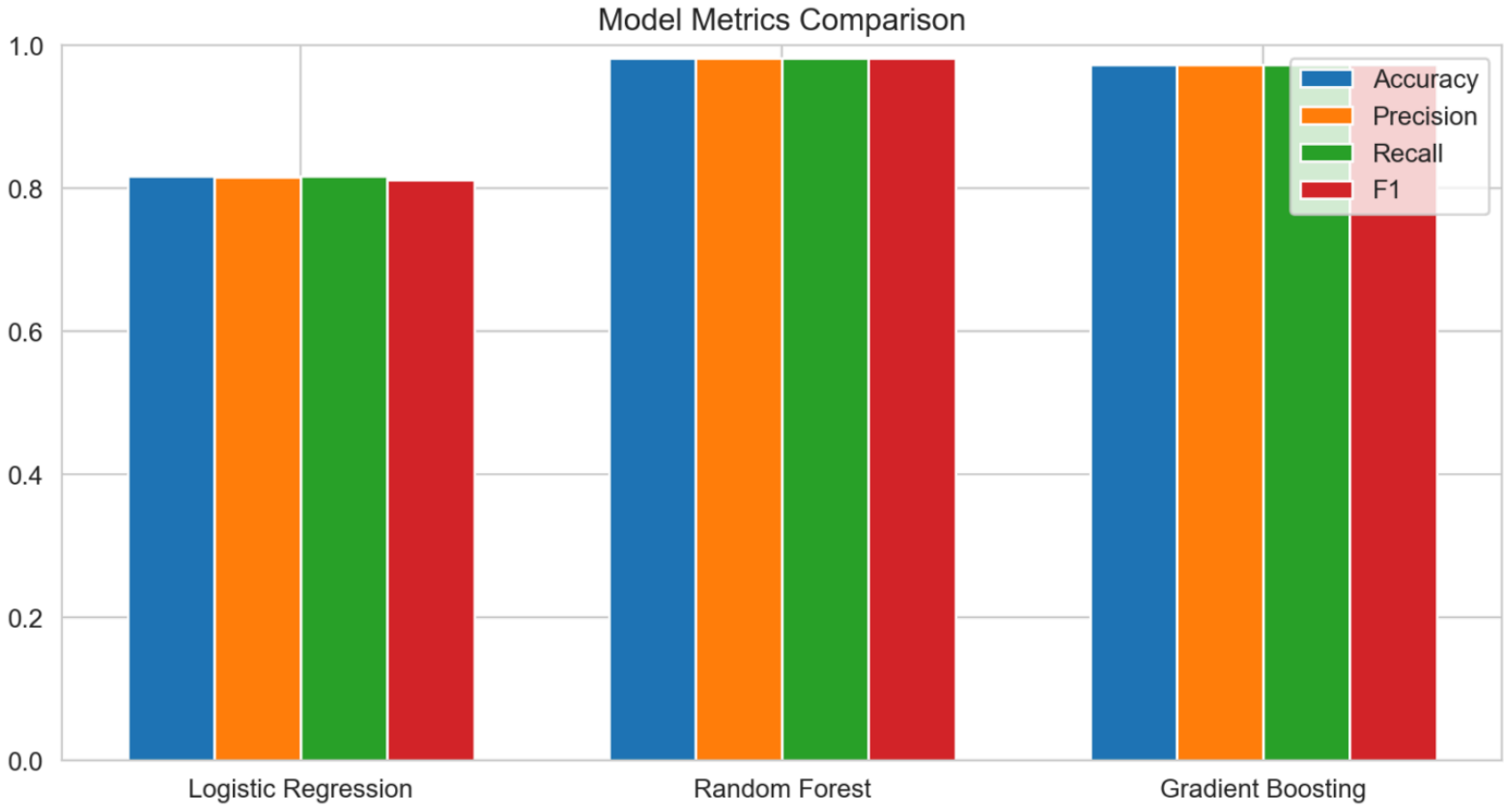
All requested models finished training.

Model Comparison

Summary table:

	model	accuracy	precision	recall	f1	time_s
1	Random Forest	0.9822	0.9822	0.9822	0.9822	1.182
2	Gradient Boosting	0.9738	0.9738	0.9738	0.9738	1.0697
0	Logistic Regression	0.8174	0.8165	0.8174	0.8127	0.1899





Conclusion

We trained **3** ML models on your dataset.

- **Best Model:** Random Forest
- **Accuracy:** 0.982
- **Precision:** 0.982
- **Recall:** 0.982
- **F1 Score:** 0.982