

BANKRUPTCY PREDICTION SYSTEM

a solution to better asses financial risk

Presented by :

- Aya Ghanouchi
- Eya Derbeli
- Yosr Jaouadi
- Zaineb Belaid
- Sarra Azaiez

INTRODUCTION

This project tackles the challenge of detecting early signs of financial instability in banks, which often lack tools to analyze complex financial data and predict bankruptcy risks.

By utilizing advanced data analysis and Object-Oriented Programming (oOP), the system offers a reliable solution for early risk detection, enabling banks to anticipate bankruptcy and enhance financial stability.

DATA FILTERING & PROCESSING

Data Source: Taiwanese banks financial records and bancrptcy 2016

Classes Involved:

- DataLoader: Filters relevant columns and loads it to the FinancialData after handling correlations and missing values.
- FinancialData: encapsulates the financial attributes related to each company (such as roaC, debtRatio, operatingMargin, etc.)

```
public class FinancialData {  
    private double roaC;  
    public double getRoaC() { return roaC; }  
    public void setRoaC(double roaC) { this.roaC = roaC; }  
}
```

the encapsulation ensures that the data is protected from unauthorized modification and maintains data integrity.

ANALYSIS

Class: FinancialAnalyzer

Key Methods:

- `calculateAltmanZScore()`: Predicts bankruptcy risk based on financial ratios
- `calculateAdvancedZScore()`: Uses additional financial parameters for deeper analysis and adds them to the `FinancialData`

```
public class AdvancedFinancialAnalyzer extends  
FinancialAnalyzer {  
    @Override public void analyzeAll(List<FinancialData> dataList)  
    { ... }  
}
```

AdvancedFinancialAnalyzer extends FinancialAnalyzer, inheriting its core analysis methods while introducing more advanced financial risk evaluation techniques.

BANKRUPTCYMODEL

- Classifier: Uses a J48 decision tree with tuned parameters for improved performance and probability estimation.
- Data Preprocessing: Converts FinancialData to a Weka dataset, normalizes, and prepares it for training.
- Class Imbalance Handling
- Model Training: Builds and evaluates the model using 10-fold cross-validation.

```
public class BankruptcyModel extends Classifier {  
}
```

The model inherits from Weka's Classifier class.

BANKRUPTCYMODEL

Feature Selection: Applies Information Gain to retain the most relevant features.

Predictions:

- `predictProbability` gives the bankruptcy probability for one instance.
- `predictAllProbabilities` returns probabilities for multiple instances.
- `predict` classifies a single instance.

Model Evaluation: Evaluates accuracy on a test dataset.



`predict(Instance)` and `predict(List<Instance>)` methods are overloaded, allowing different parameter types.

VISUALIZATION

Class: ChartGenerator & ReportGenerator

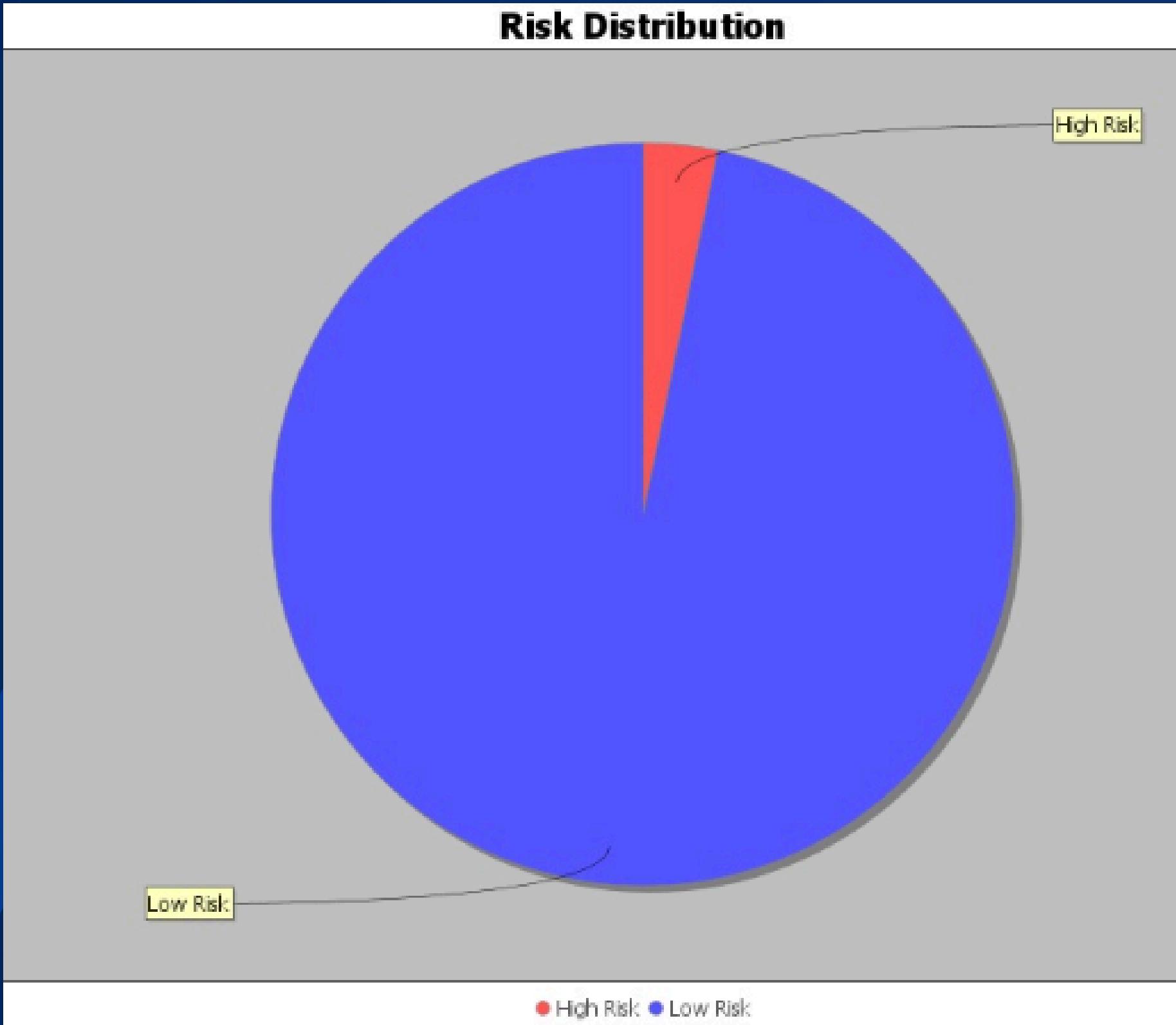
- Purpose: Generate different types of risk reports and visualizations to help stakeholders easily interpret complex financial data and analysis results.

VISUALIZATION

The ReportGenerator class is designed to generate a detailed report that combines financial data and bankruptcy predictions. It writes the results to a CSV file with financial metrics and analysis, along with the risk predictions generated by the BankruptcyModel and financial ratios calculated using the FinancialAnalyzer.

Net Income to Total	Gross Profit to Sales	Predicted Risk Level	Predicted Probability	Debt-to-Equity Ratio	Z-Score
0.7168453432	0.6014532901	High Risk	0.9375	0.3390770068	3.167081743
0.795297136	0.6102365259	High Risk	0.9375	0.3297401479	3.210769194
0.774669697	0.6014493405	High Risk	0.9375	0.3347768513	3.167166406
0.7395545252	0.5835376122	Low Risk	0.01764705882	0.3315089787	3.234255545
0.7950158726	0.5987815099	Low Risk	0.01764705882	0.3307263194	3.288347208
0.7104204951	0.5901723267	High Risk	0.9375	0.335534372	3.199660071

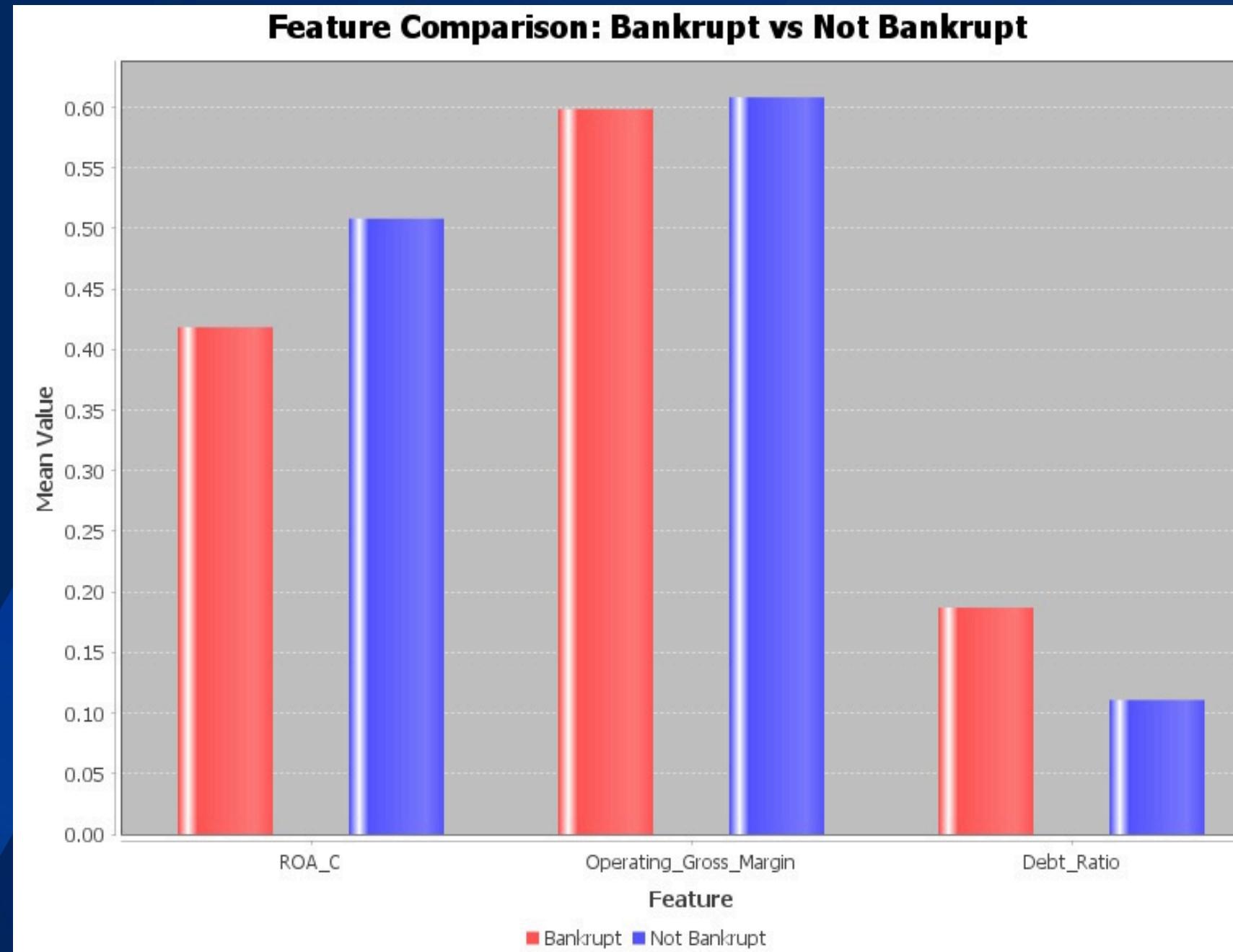
VISUALIZATION



RiskPieChart:

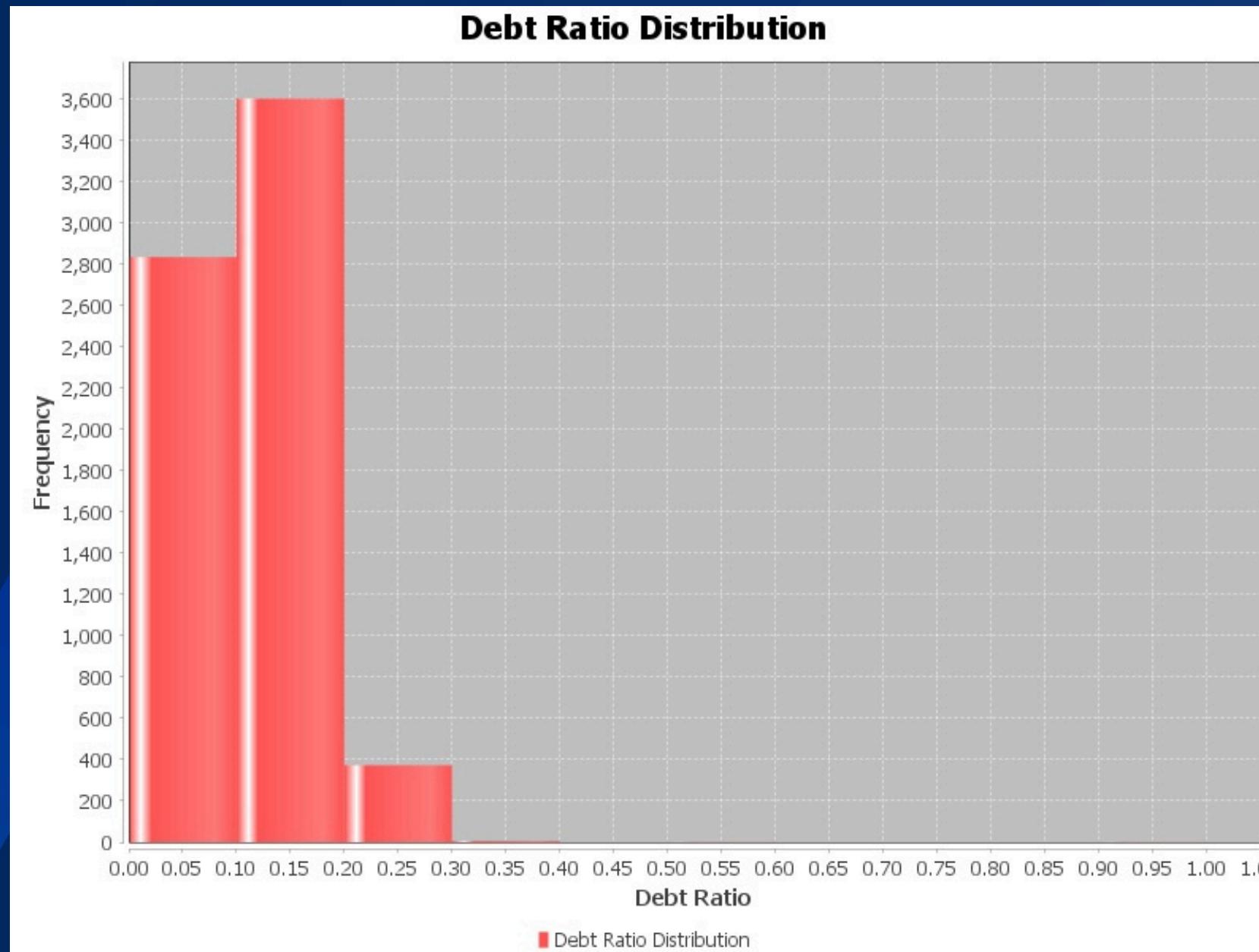
Generates a pie chart showing the distribution of "High Risk" and "Low Risk" banks based on bankruptcy predictions.

VISUALIZATION



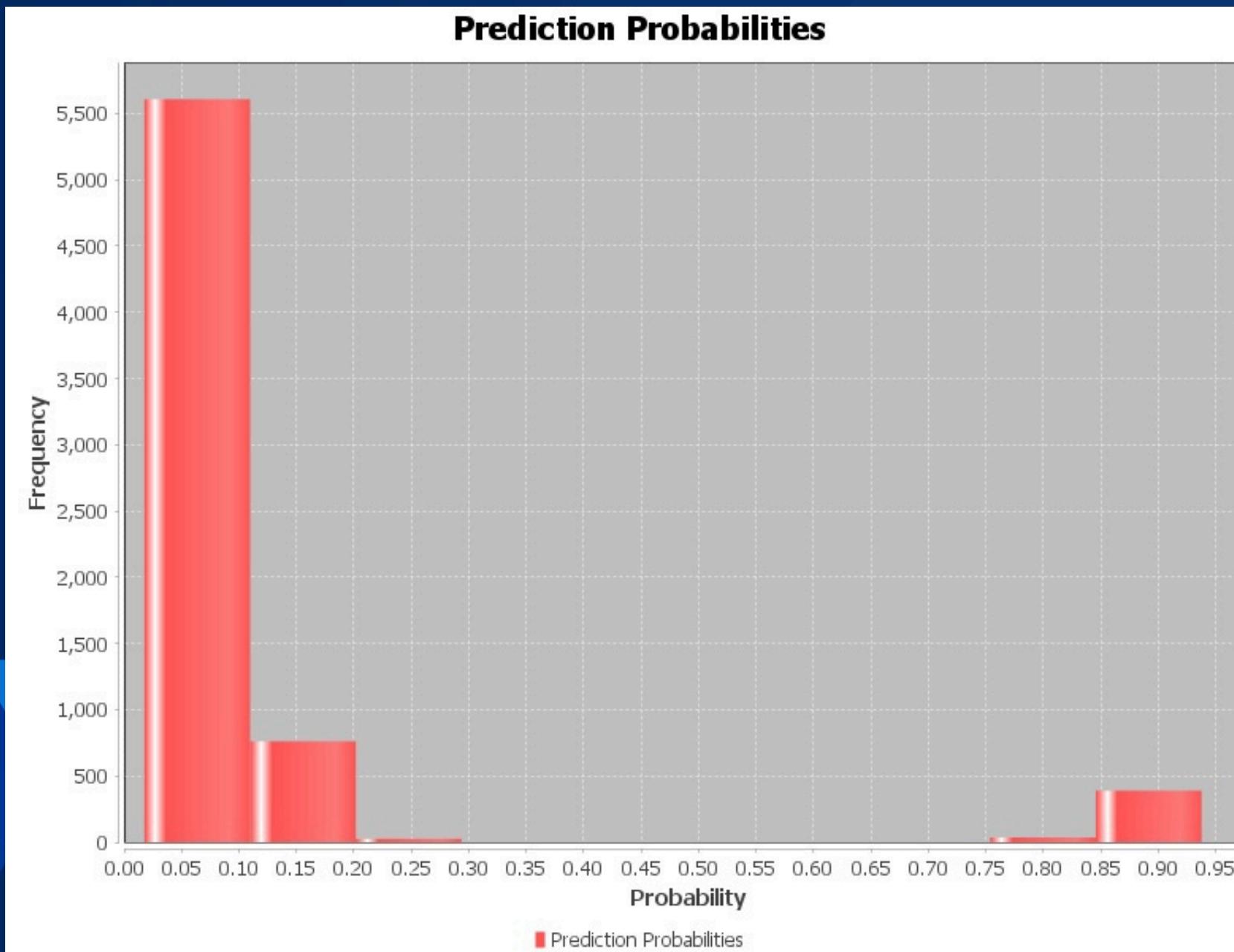
FeatureComparisonBarChart:
Generates a bar chart comparing the mean values of different financial features (e.g., ROA, debt ratio) between bankrupt and non-bankrupt banks.

VISUALIZATION



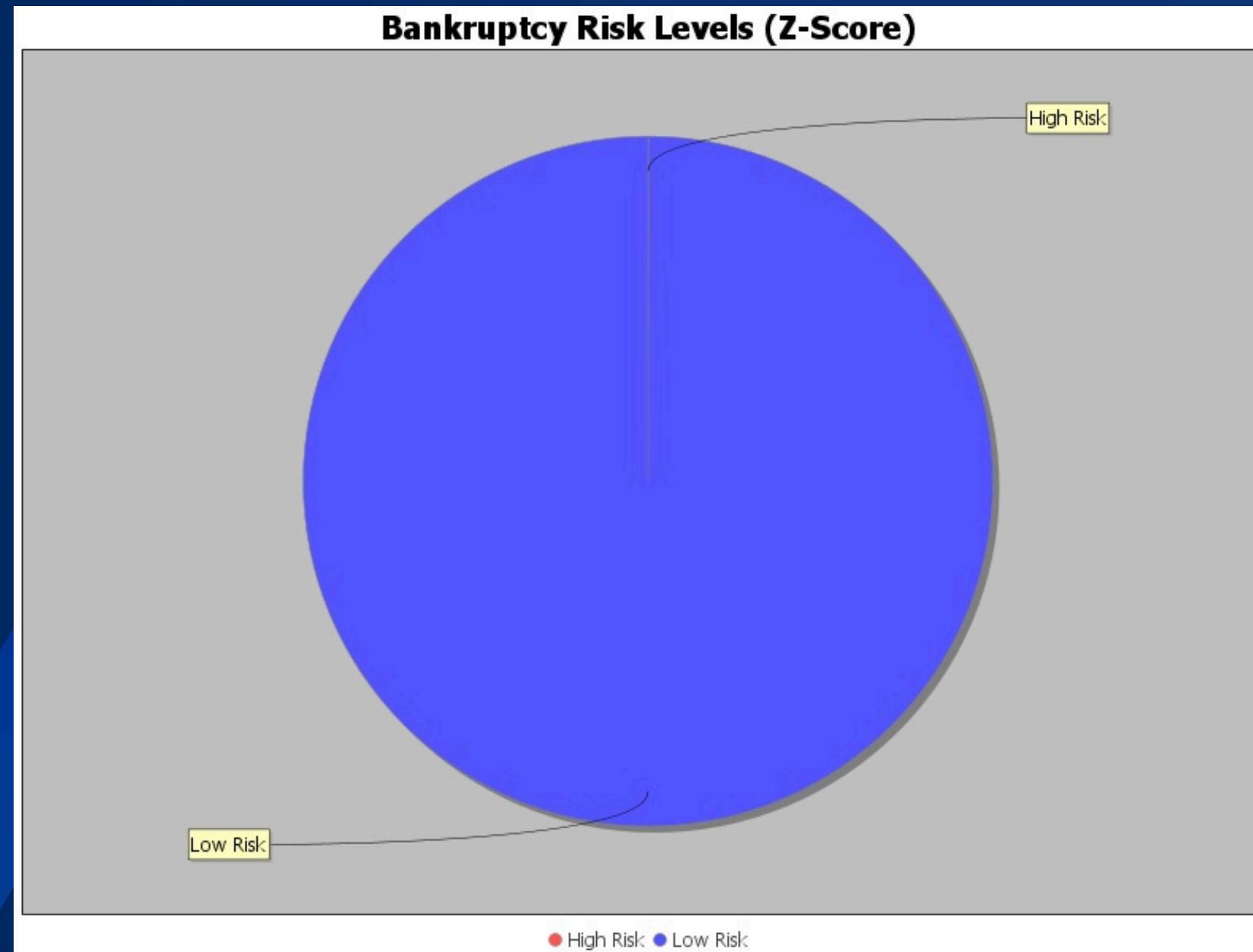
DebtRatioHistogram:
Creates a histogram displaying the distribution of the debt ratio across banks.

VISUALIZATION



Prediction Probabilities Histogram:
A histogram that shows the distribution of prediction probabilities for bankruptcy classification.

VISUALIZATION



Bankruptcy Risk Level Pie Chart (z-Score):

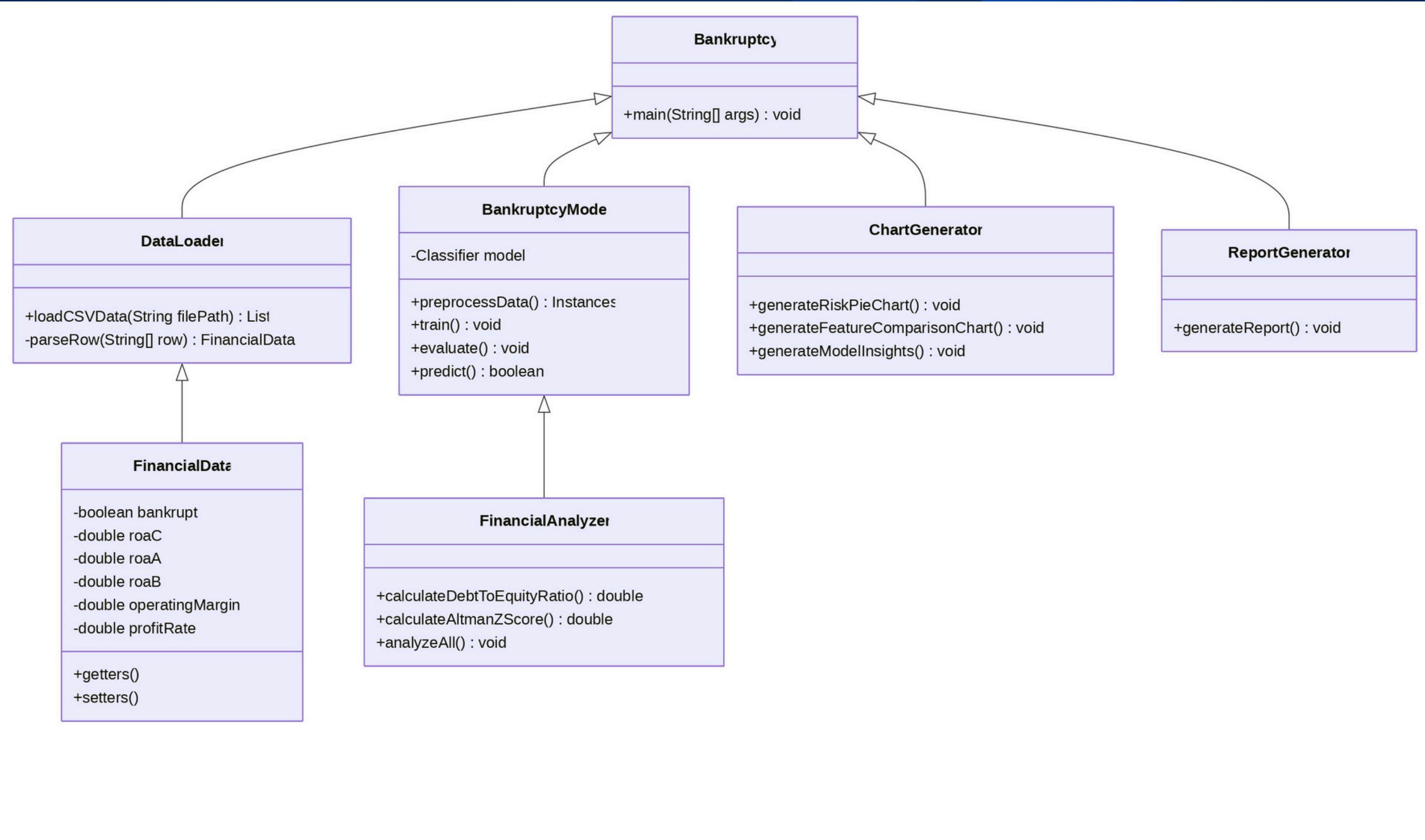
- Generates a pie chart showing bankruptcy risk levels based on the Altman Z-Score calculated using the FinancialAnalyzer.
- Companies with a Z-Score below 1.8 are considered high risk.

OBJECT-ORIENTED PROGRAMMING CONCEPTS

Aiming to create a flexible, maintainable, and extendable solution for predicting bankruptcy risk based on financial data. We included :

- **Encapsulation** to protect financial data by using getter and setter methods for controlled access.
- **Inheritance** to allow specialized classes, like AdvancedFinancialAnalyzer, to reuse and extend functionality from FinancialAnalyzer.
- **Polymorphism** to enable methods like analyzeAll() to behave differently based on the class calling them, making the system adaptable to future changes.
- **Abstraction** ensures that we don't need to modify other parts of the codebase, but simply extend the system by adding new subclasses.

ARCHITECTURE DIAGRAM



CONCLUSION

The Bankruptcy Prediction System uses Object-Oriented Programming to deliver a scalable solution for assessing bankruptcy risks. It enhances decision-making with automated, accurate risk assessments and can be expanded to include real-time data analysis for ongoing evaluations.

THANK YOU