Machine Learning Classification of Stock Performance

Byrnes Mulumbeni & Laura Tambwe Mwibashiye
DATA 3461 | Final Project

Motivation

The Challenge

- Stock prediction is notoriously difficult
- Even expert investors struggle to consistently beat the market
- Traditional approaches rely on complex models with many data sources

Our Question: Can we predict stock performance using **only fundamental accounting data and specific features**?

Why It Matters

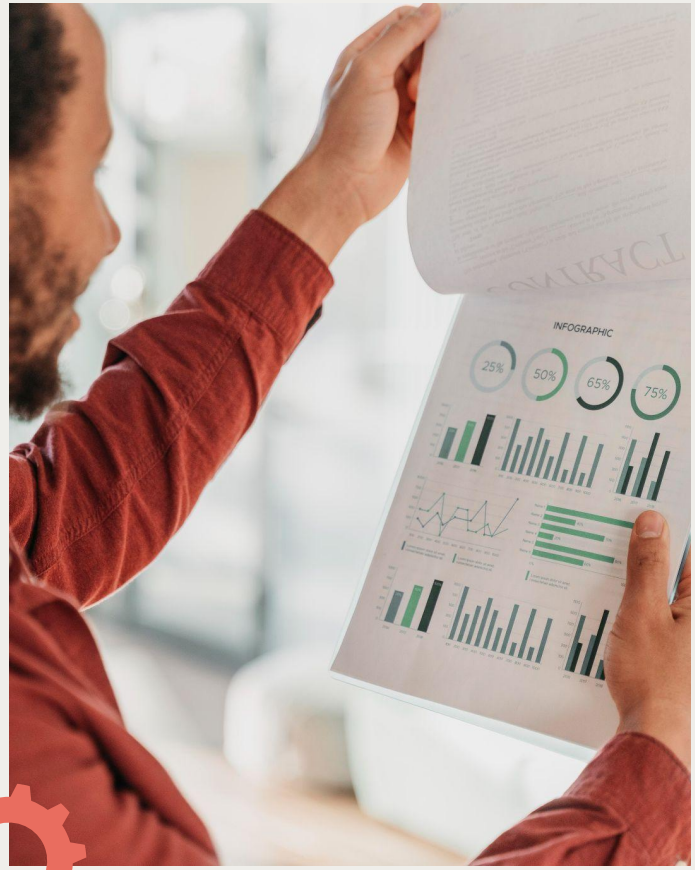
- Fundamental analysis is the foundation of value investing
- Understanding which financial metrics drive returns informs investment strategy
- Machine learning can uncover nonlinear patterns humans miss

Research Questions

1. **Can machine learning models accurately predict whether a US stock will increase in value in 2019 based on financial indicators from 2018?**

2. **Which financial indicators are the most influential in determining whether a stock's price will rise the following year?**

3. **Which algorithm (Logistic Regression, Random Forest, Gradient Boosting) performs best for forecasting stock price movements?**



Data Overview

Source: Kaggle 200+ Financial Indicators of US stocks (2014-2018)

- 4,392 US companies
- 225 financial variables

Target:

- Class 1: Price increased (610)
- Class 0: Price decreased (269)

Dataset shape: (4392, 225)

	Unnamed: 0	Revenue	Revenue Growth	Cost of Revenue	Gross Profit	R&D Expenses	SG&A Expense	Operating Expenses	Operating Income
0	CMCSA	9.450700e+10	0.1115	0.000000e+00	9.450700e+10	0.000000e+00	6.482200e+10	7.549800e+10	1.900900e+10
1	KMI	1.414400e+10	0.0320	7.288000e+09	6.856000e+09	0.000000e+00	6.010000e+08	3.062000e+09	3.794000e+09
2	INTC	7.084800e+10	0.1289	2.711100e+10	4.373700e+10	1.354300e+10	6.750000e+09	2.042100e+10	2.331600e+10
3	MU	3.039100e+10	0.4955	1.250000e+10	1.789100e+10	2.141000e+09	8.130000e+08	2.897000e+09	1.499400e+10
4	GE	1.216150e+11	0.0285	9.546100e+10	2.615400e+10	0.000000e+00	1.811100e+10	4.071100e+10	-1.455700e+10

Interest Expense	...	Receivables growth	Inventory Growth	Asset Growth	Book Value per Share Growth	Debt Growth	R&D Expense Growth	SG&A Expenses Growth	Sector	2019 PRICE VAR [%]	Class
3.542000e+09	...	0.2570	0.0000	0.3426	0.0722	0.7309	0.0000	0.1308	Consumer Cyclical	32.794573	1
1.917000e+09	...	0.0345	-0.0920	-0.0024	0.0076	-0.0137	0.0000	-0.1265	Energy	40.588068	1
-1.260000e+08	...	0.1989	0.0387	0.0382	0.1014	-0.0169	0.0390	-0.0942	Technology	30.295514	1
3.420000e+08	...	0.4573	0.1511	0.2275	0.6395	-0.5841	0.1738	0.0942	Technology	64.213737	1
5.059000e+09	...	-0.2781	-0.2892	-0.1575	-0.4487	-0.2297	0.0000	0.0308	Industrials	44.757840	1

Preprocessing

1. Median imputation for missing values
2. One-hot encode Sector
3. Standardize features

```
Shape of X: (4392, 223)
Shape of y: (4392,)
Shape after encoding Sector: (4392, 232)
```

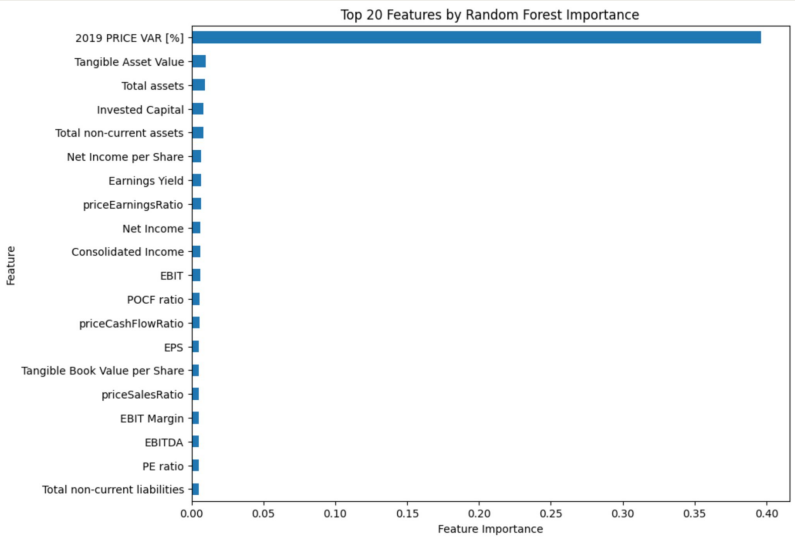
	Revenue	Revenue Growth	Cost of Revenue	Gross Profit	R&D Expenses	SG&A Expense	Operating Expenses	Operating Income	Interest Expense	Earnings before Tax	...	Sector_Communication Services	Sector_Consumer Cyclical
0	4.386163	-0.016826	-0.204355	12.124323	-0.122982	17.793454	13.679531	6.207555	9.301998	5.560956	...	-0.143817	2.771253
1	0.444967	-0.017240	0.288836	0.634284	-0.122982	-0.074849	0.309901	1.063263	4.915309	0.628668	...	-0.143817	-0.360848
2	3.225868	-0.016736	1.630291	5.468959	14.794731	1.635997	3.513880	7.663780	-0.599771	8.694859	...	-0.143817	-0.360848
3	1.241759	-0.014830	0.641540	2.080846	2.235345	-0.015864	0.279447	4.850057	0.663596	5.252378	...	-0.143817	-0.360848
4	5.715604	-0.017258	6.255649	3.164030	-0.122982	4.796984	7.258838	-5.141331	13.397141	-8.524802	...	-0.143817	-0.360848

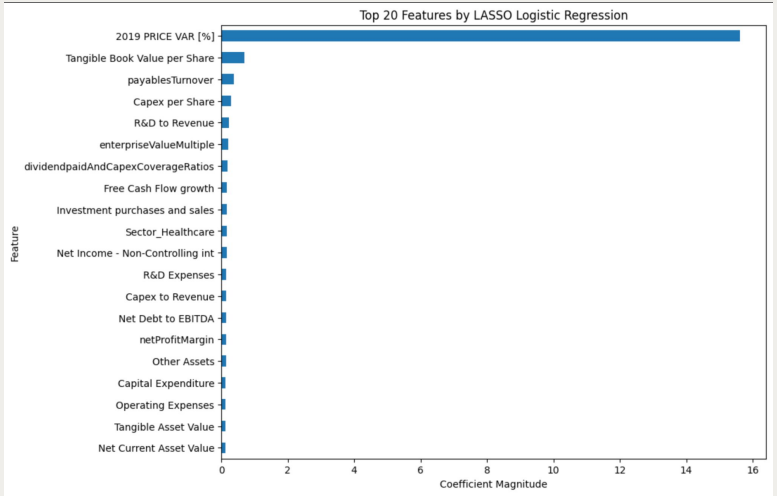
	Sector_Consumer Defensive	Sector_Energy	Sector_Financial Services	Sector_Healthcare	Sector_Industrials	Sector_Real Estate	Sector_Technology	Sector_Utilities
	-0.213226	-0.244634	-0.480564	-0.432095	-0.387738	-0.248272	-0.411496	-0.154195
	-0.213226	4.087747	-0.480564	-0.432095	-0.387738	-0.248272	-0.411496	-0.154195
	-0.213226	-0.244634	-0.480564	-0.432095	-0.387738	-0.248272	2.430156	-0.154195
	-0.213226	-0.244634	-0.480564	-0.432095	-0.387738	-0.248272	2.430156	-0.154195
	-0.213226	-0.244634	-0.480564	-0.432095	2.579063	-0.248272	-0.411496	-0.154195

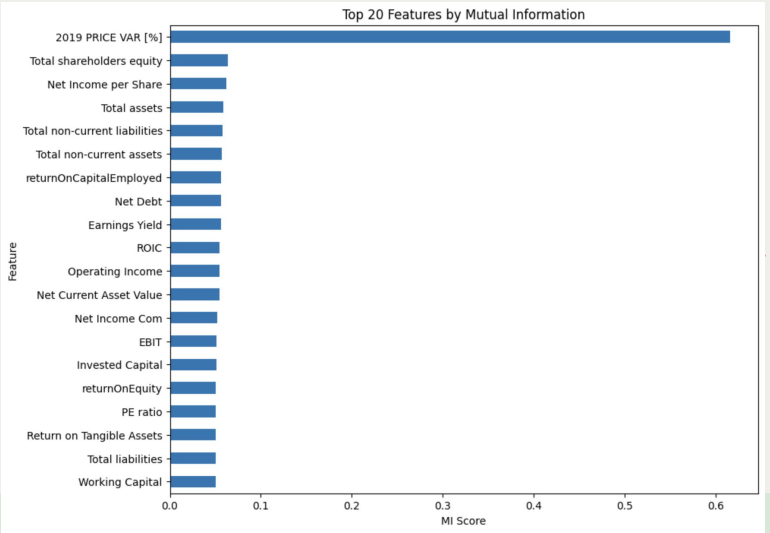
Feature Selection 1

Three methods:

- Mutual Information
- Random Forest Feature Importance
- LASSO Regression

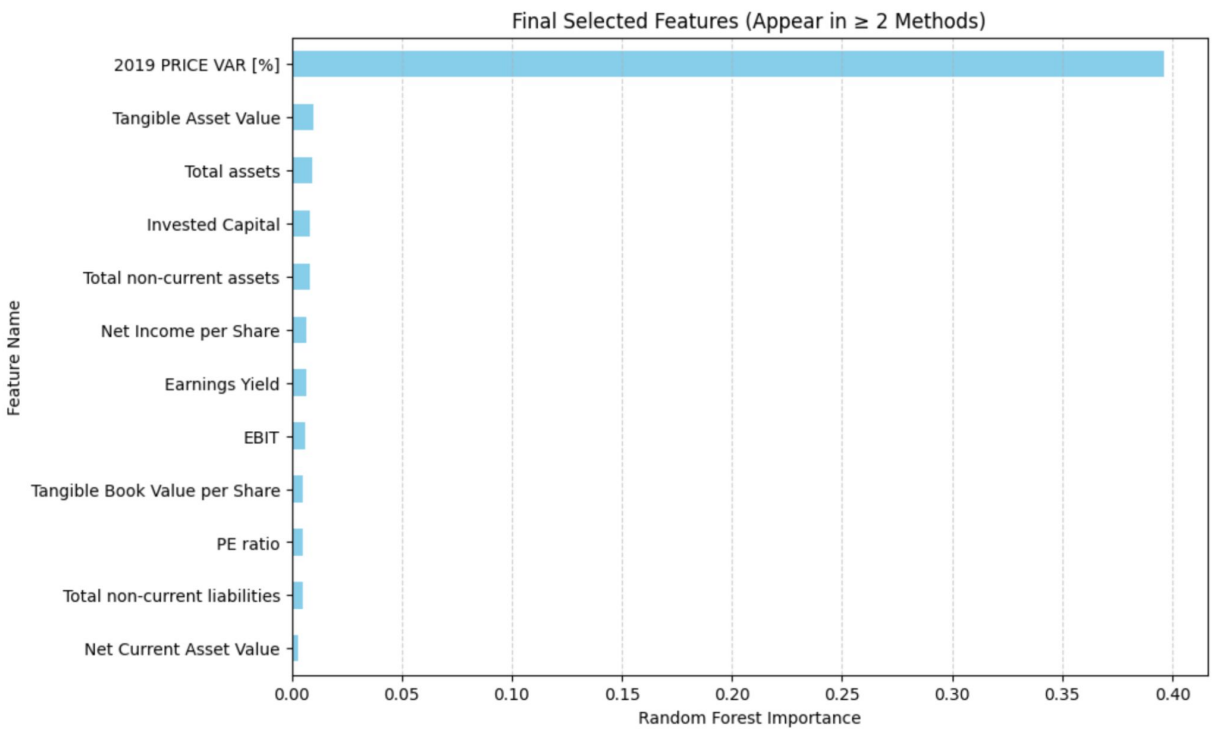






Feature Selection

Criteria: Feature must appear in ≥2 methods
Result: 12 features selected



Problem Discovered: Data Leakage

The feature `2019 PRICE VAR [%]` directly measures the stock's percentage price change in 2019. The target variable `Class` also depends on whether the stock increased in 2019.

Using this feature introduces **data leakage**, because the model has access to the future outcome it is supposed to predict.

This caused unrealistically high performance.

To correct this, we remove `2019 PRICE VAR [%]` from the dataset and redo the feature selection and model training steps.

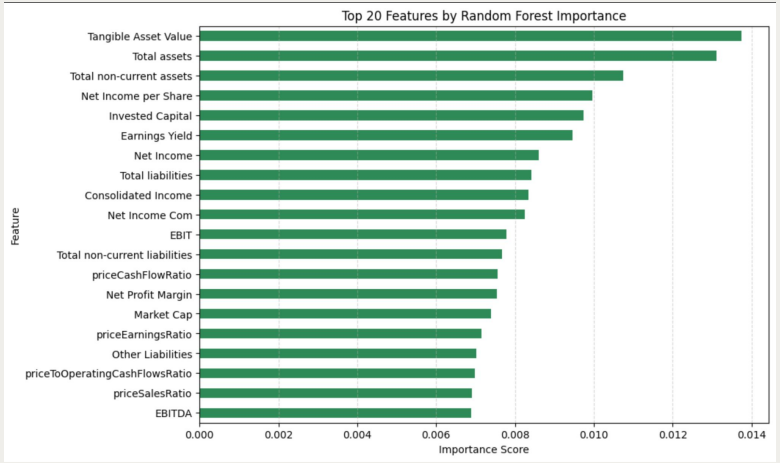
```
=== Logistic Regression Results ===
Accuracy: 0.9840728100113766
Classification Report:
              precision    recall  f1-score   support

           0       1.00      0.95      0.97       269
           1       0.98      1.00      0.99       610

    accuracy                           0.98       879
   macro avg       0.99      0.97      0.98       879
weighted avg       0.98      0.98      0.98       879

AUC: 0.9984581632031202
```

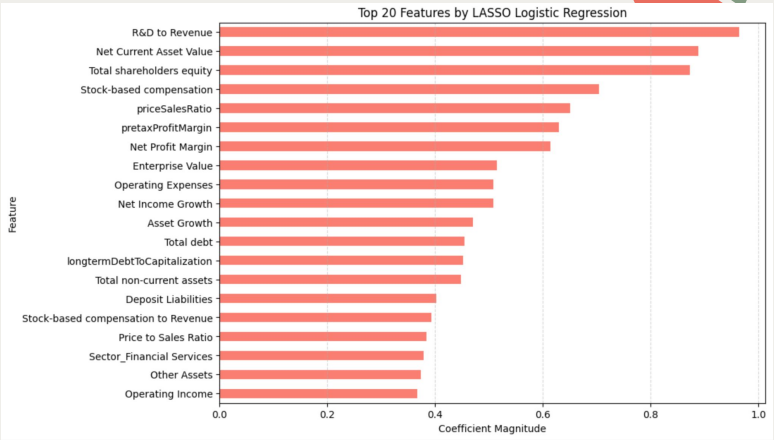
```
=== Random Forest Results ===
Accuracy: 1.0
Classification Report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00       269
           1       1.00      1.00      1.00       610

    accuracy                           1.00       879
   macro avg       1.00      1.00      1.00       879
weighted avg       1.00      1.00      1.00       879

AUC: 1.0
```

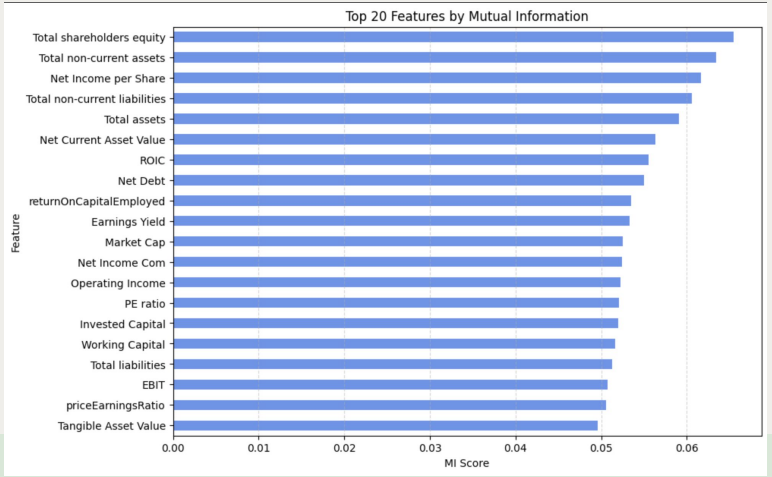
Feature Selection 2

Three methods:

- Mutual Information
- Random Forest
- LASSO Regression

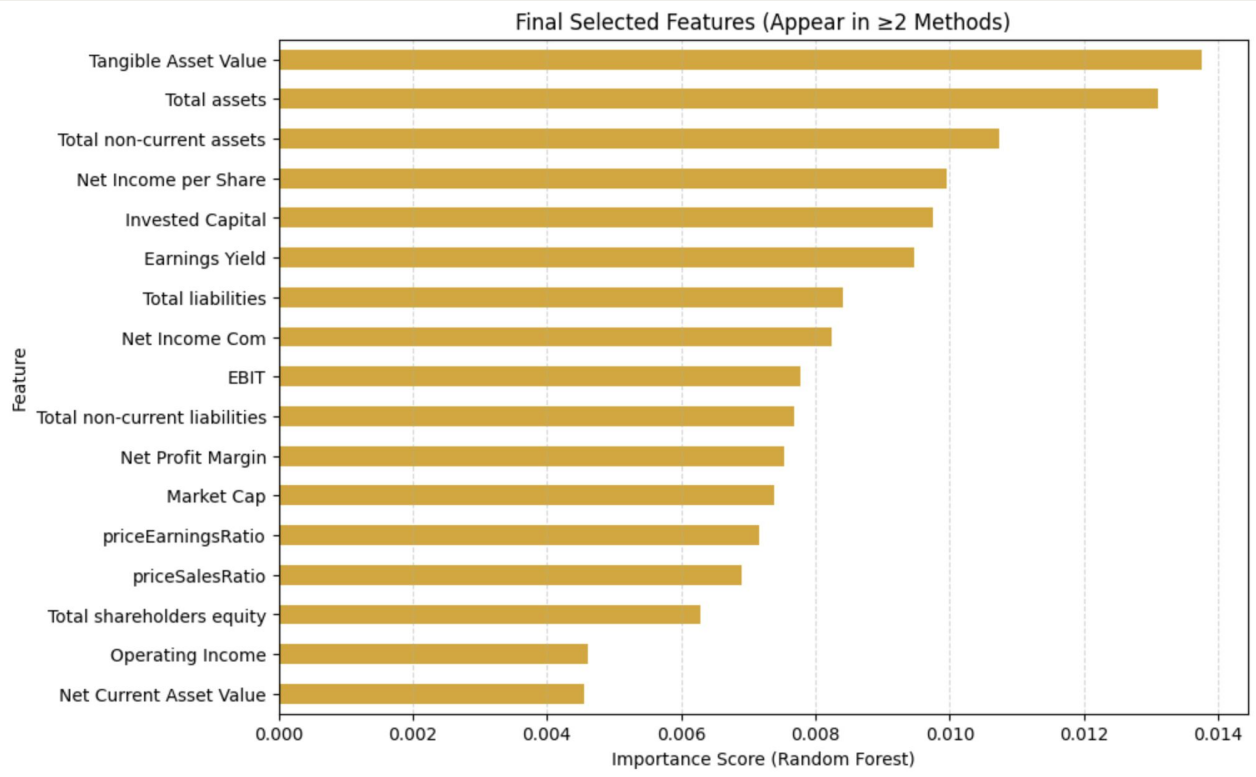






Feature Selection 2

Criteria: Feature must appear in ≥2 methods
Result: 17 features selected



Data Leakage, Again

```
=== Random Forest ===
Accuracy: 1.0

Classification Report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00       269
           1       1.00      1.00      1.00       610

    accuracy                           1.00       879
   macro avg       1.00      1.00      1.00       879
weighted avg       1.00      1.00      1.00       879

AUC: 1.0
```

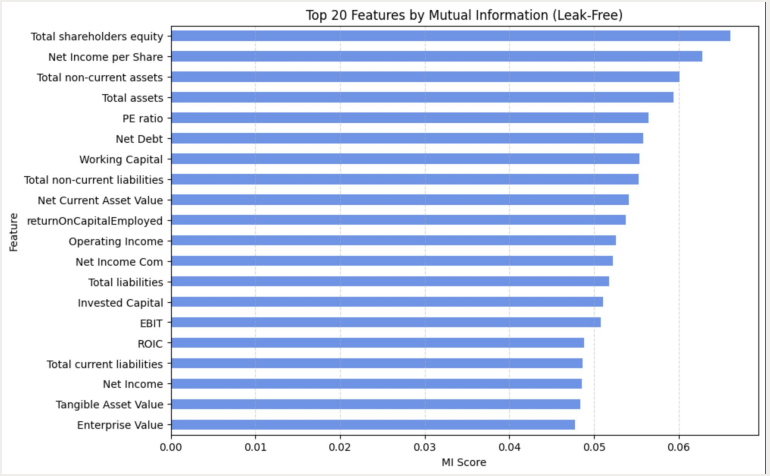
```
=== Gradient Boosting ===
Accuracy: 1.0

Classification Report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00       269
           1       1.00      1.00      1.00       610

    accuracy                           1.00       879
   macro avg       1.00      1.00      1.00       879
weighted avg       1.00      1.00      1.00       879

AUC: 1.0
```

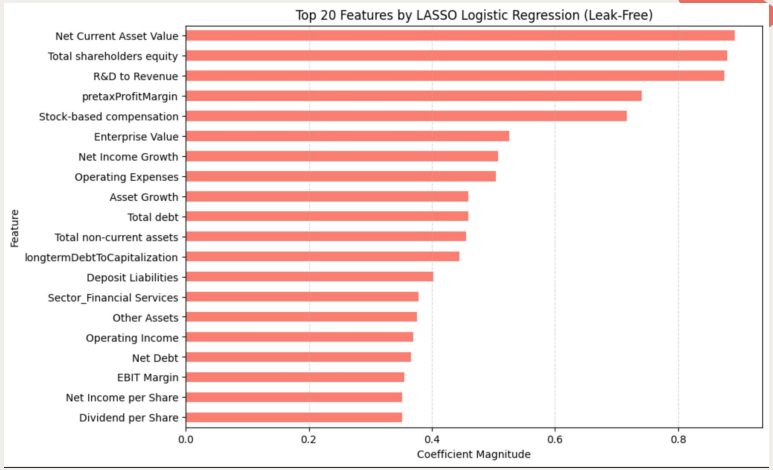
```
=== Logistic Regression ===
Accuracy: 0.9840728100113766

Classification Report:
              precision    recall  f1-score   support

           0       1.00      0.95      0.97       269
           1       0.98      1.00      0.99       610

    accuracy                           0.98       879
   macro avg       0.99      0.97      0.98       879
weighted avg       0.98      0.98      0.98       879

AUC: 0.9984581632031202
```

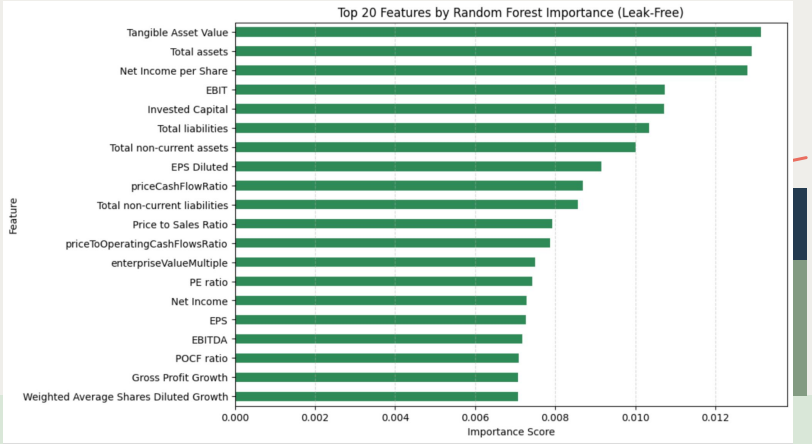
Feature Selection 3

Three methods:

- Mutual Information
- Random Forest
- LASSO Regression

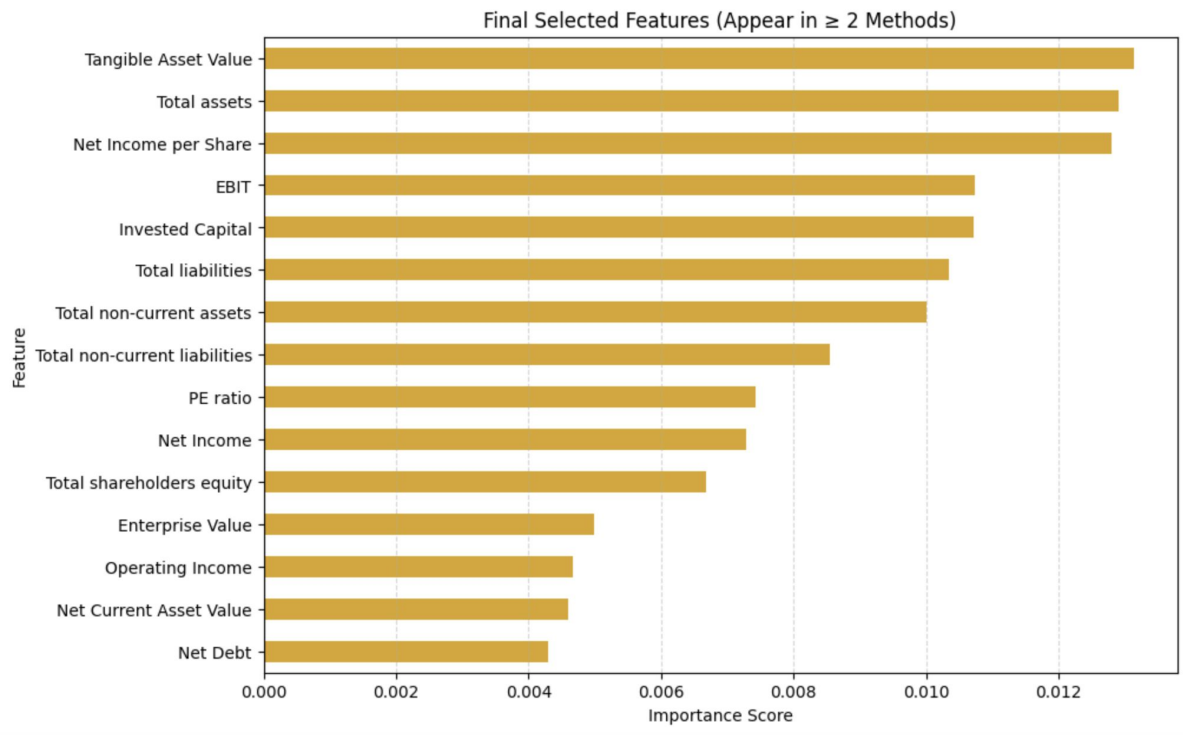






Feature Selection 3

Criteria: Feature must appear in ≥2 methods
Result: 15 features selected



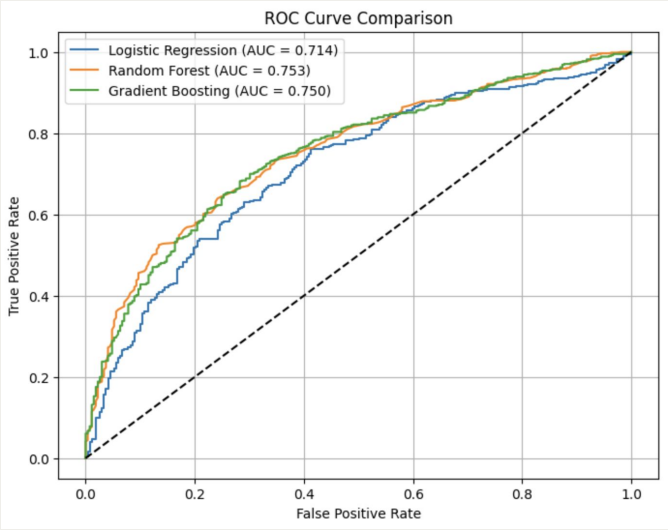
Results

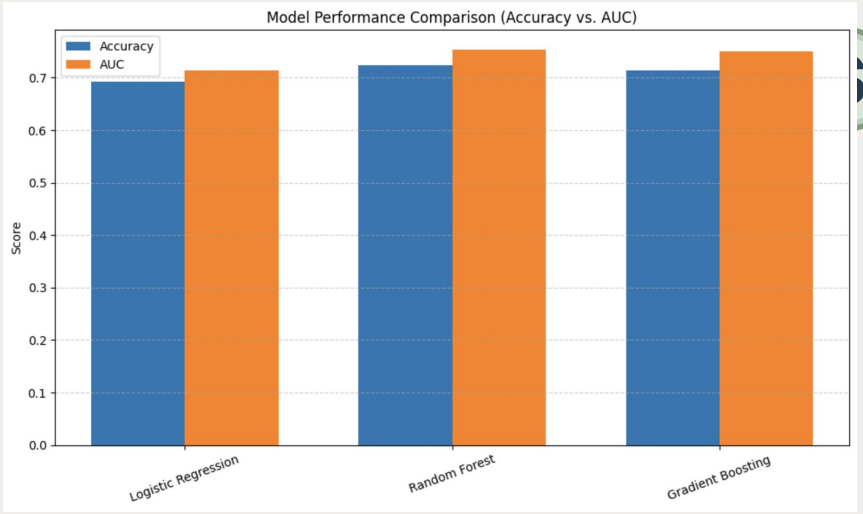
- Logistic Regression struggles with class 0 because the dataset is moderately imbalanced (more class 1 stocks).
- Random Forest and Gradient Boosting significantly outperform Logistic Regression, showing better ability to capture nonlinear relationships in financial data.
- AUC values around 0.75 indicate the models have moderate predictive power, which is expected in real-world stock prediction problems.

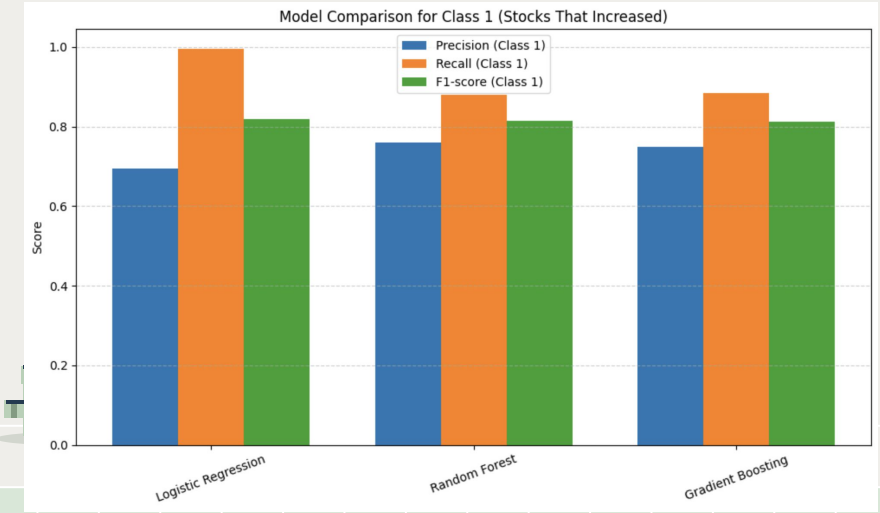
```
=== Logistic Regression ===
Accuracy: 0.6928327645051194

Classification Report:
              precision    recall  f1-score   support

           0       0.40      0.01      0.01       269
           1       0.69      1.00      0.82       610

    accuracy                           0.69       879
   macro avg       0.55      0.50      0.42       879
weighted avg       0.60      0.69      0.57       879

AUC: 0.7144006337985254
```

```
=== Random Forest ===
Accuracy: 0.7235494880546075

Classification Report:
              precision    recall  f1-score   support

           0       0.58      0.37      0.45       269
           1       0.76      0.88      0.82       610

    accuracy                           0.72       879
   macro avg       0.67      0.62      0.63       879
weighted avg       0.70      0.72      0.70       879

AUC: 0.7532908769577671
```

```
=== Gradient Boosting ===
Accuracy: 0.714448236632537

Classification Report:
              precision    recall  f1-score   support

           0       0.56      0.33      0.41       269
           1       0.75      0.89      0.81       610

    accuracy                           0.71       879
   macro avg       0.65      0.61      0.61       879
weighted avg       0.69      0.71      0.69       879

AUC: 0.7504997257602535
```


ROC Curve Comparison
Logistic Regression (AUC = 0.714)
Random Forest (AUC = 0.753)
Gradient Boosting (AUC = 0.750)
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0


Model Performance Comparison (Accuracy vs. AUC)
Accuracy
AUC
Score
Logistic Regression
Random Forest
Gradient Boosting


Model Comparison for Class 1 (Stocks That Increased)
Precision (Class 1)
Recall (Class 1)
F1-score (Class 1)
Score
Logistic Regression
Random Forest
Gradient Boosting


Model Comparison for Class 0 (Stocks That Decreased)
Precision (Class 0)
Recall (Class 0)
F1-score (Class 0)
Score
Logistic Regression
Random Forest
Gradient Boosting

Future Work

Handle class imbalance:

- SMOTE, class weights, threshold tuning

Try advanced models:

- XGBoost, LightGBM

Add external data:

- Market trends, macro indicators

Conclusion

What we built:
A leak-free stock prediction model using only accounting data

What we learned:
Data leakage can hide in plain sight, rigorous validation is critical

What we achieved:

- Random Forest: 72.4% accuracy, 0.753 AUC
- Realistic, reproducible results
- Deep understanding of the methodology

Thank You!

Questions?