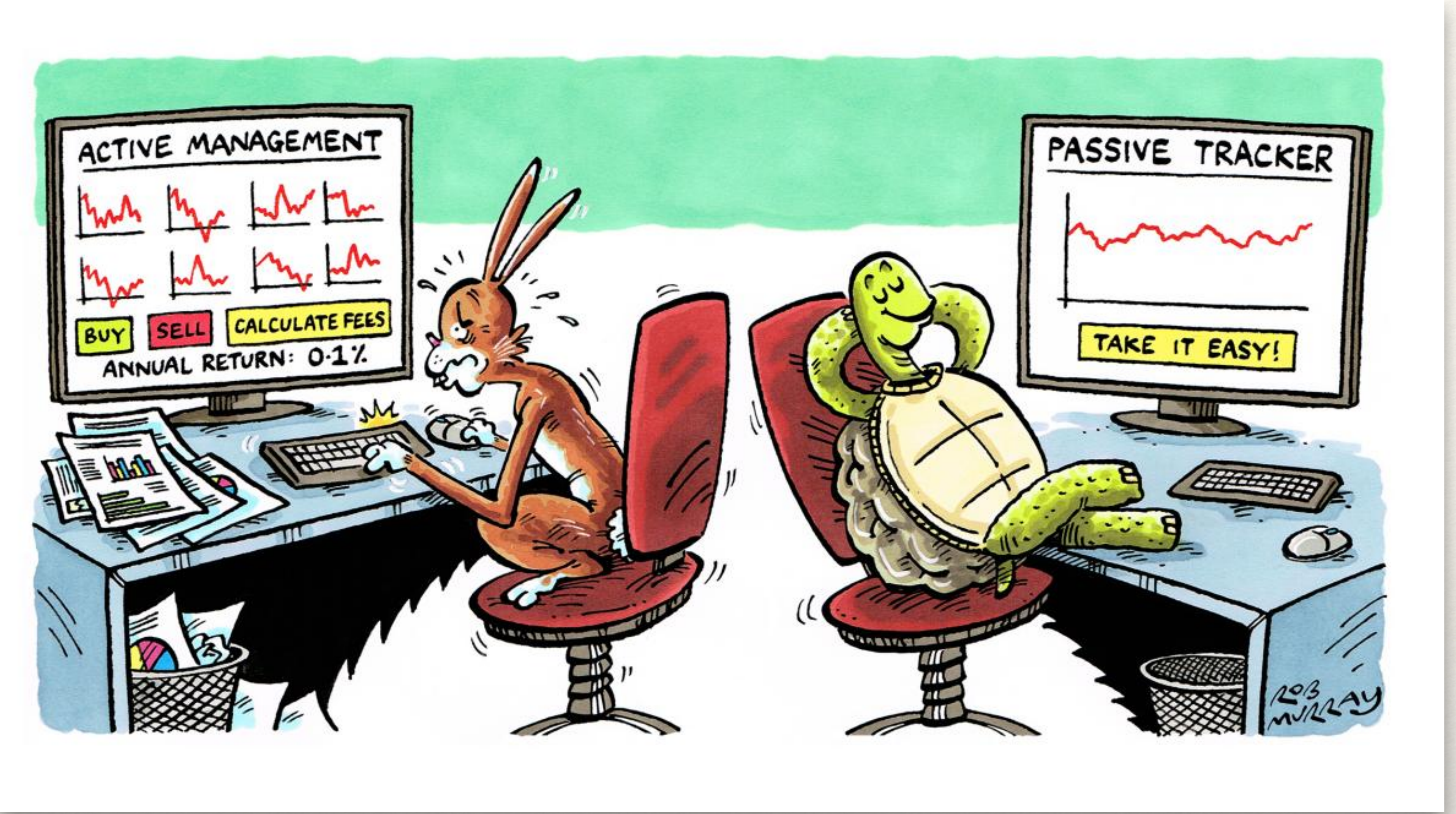
Predicting US Equity Mutual Fund Outperformance/Underperformance vs. S&P 500

Mike Choi



"Picking the best-performing funds is like trying to predict the dice before you roll them down the craps table. I can't do it. The public can't do it."

–Paul Samuelson, Nobel laureate in Economics, 1970

Key Questions to Answer

- Can I build a classification model that can predict a US equity mutual fund outperformance/underperformance vs. the S&P 500 with good accuracy?

- Based on the model results, should one put his/her money in the hands of stock-pickers (mutual funds) or index mimickers (index funds)?

Data - Scope and Assumptions

Data	❖ Morningstar API
Observations	❖ US equity mutual funds • Non-US/non-equity funds have different performance benchmarks ❖ **~ 6000 funds in total**
Features	❖ 16 features - stock, portfolio, and fund-level statistics ❖ No features on investment styles or sector weights ❖ No features that are directly derived from or indicative of past performance
Target	❖ Outperform or Underperform (based on 3-year *annualized* return vs. S&P 500)

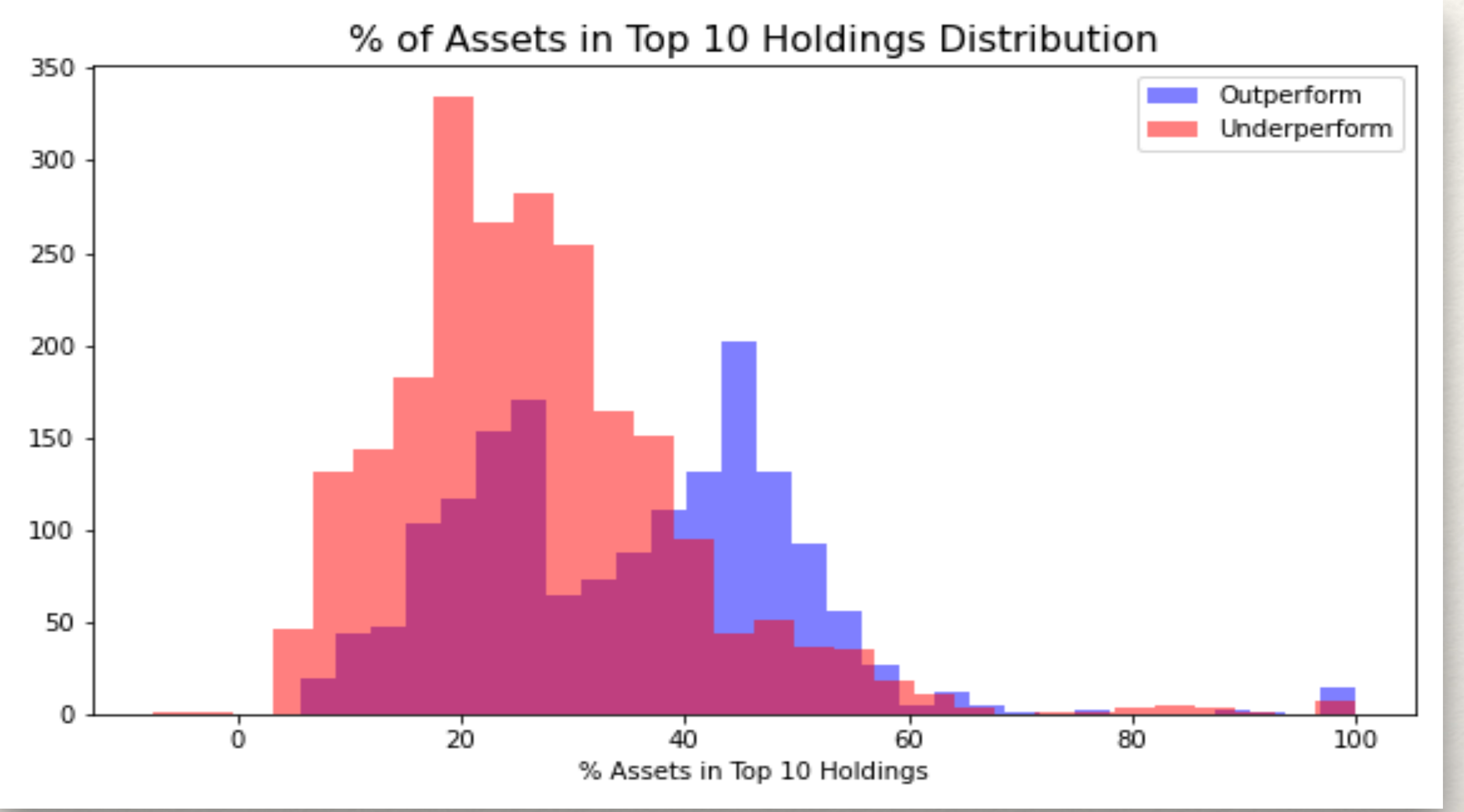
Feature Selection through Statistical Tests and Feature Importances

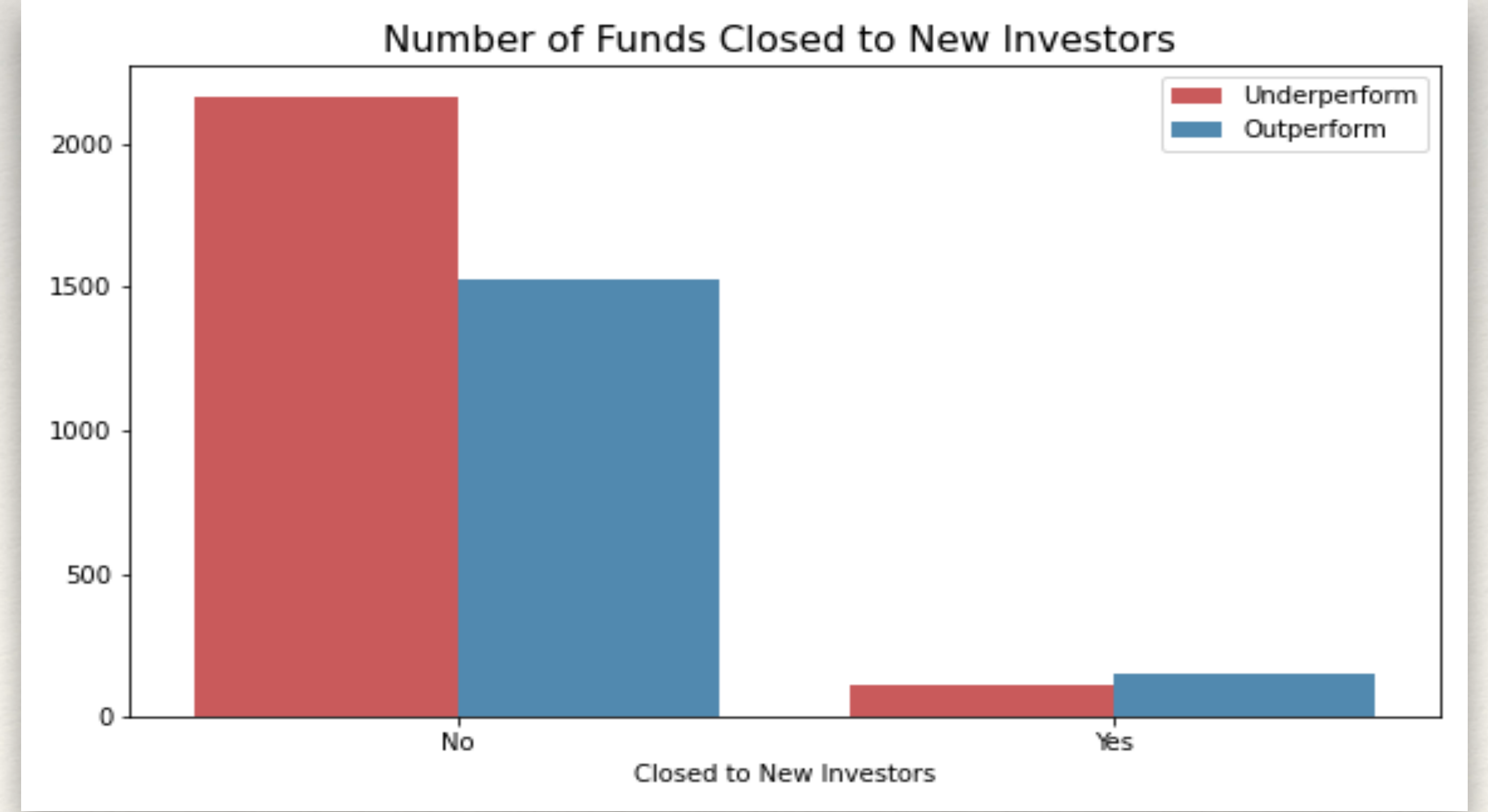
- Used Scikit-learn's feature selection methods
  - **Set number of features = 5**

	Morningstar Sust. Rating	**ROE Last Year (%)**	**Debt / Capital Last Year (%)**	**No. Holdings in Portfolio**	**% Assets in Top 10 Holdings**	**Expense Ratio**	**Closed to New Investors**	**No-load Fund**
Chi-square test	✓		✓		✓		✓	✓
F test	✓		✓		✓	✓		✓
Mutual_info_ classif test		✓	✓	✓	✓		✓	
Recursive feature elimination		✓	✓	✓	✓	✓		
SelectFromModel		✓	✓	✓	✓		✓	
Total	2	3	5	3	5	2	3	2

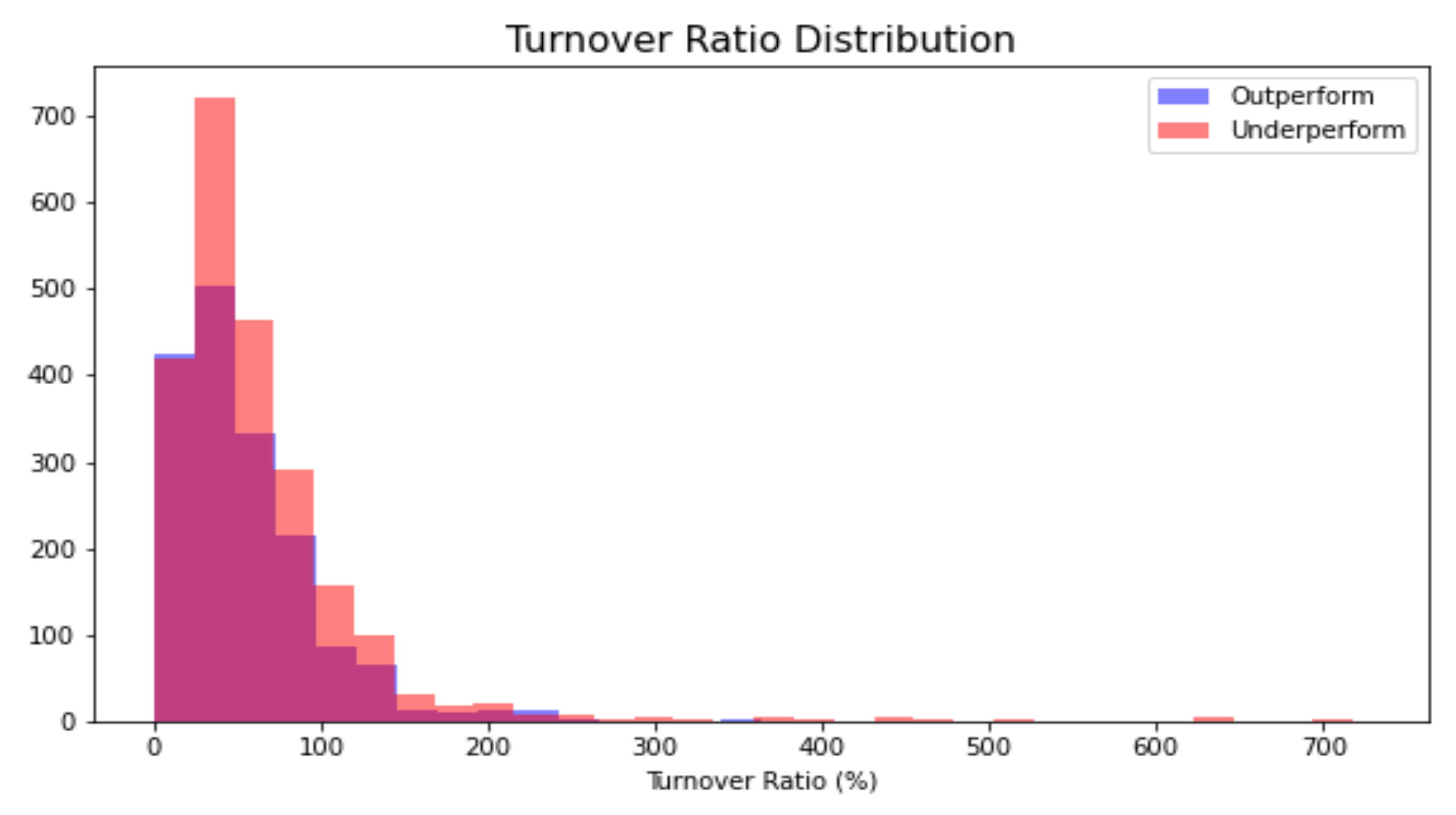
Class Separation Observed in Selected Features

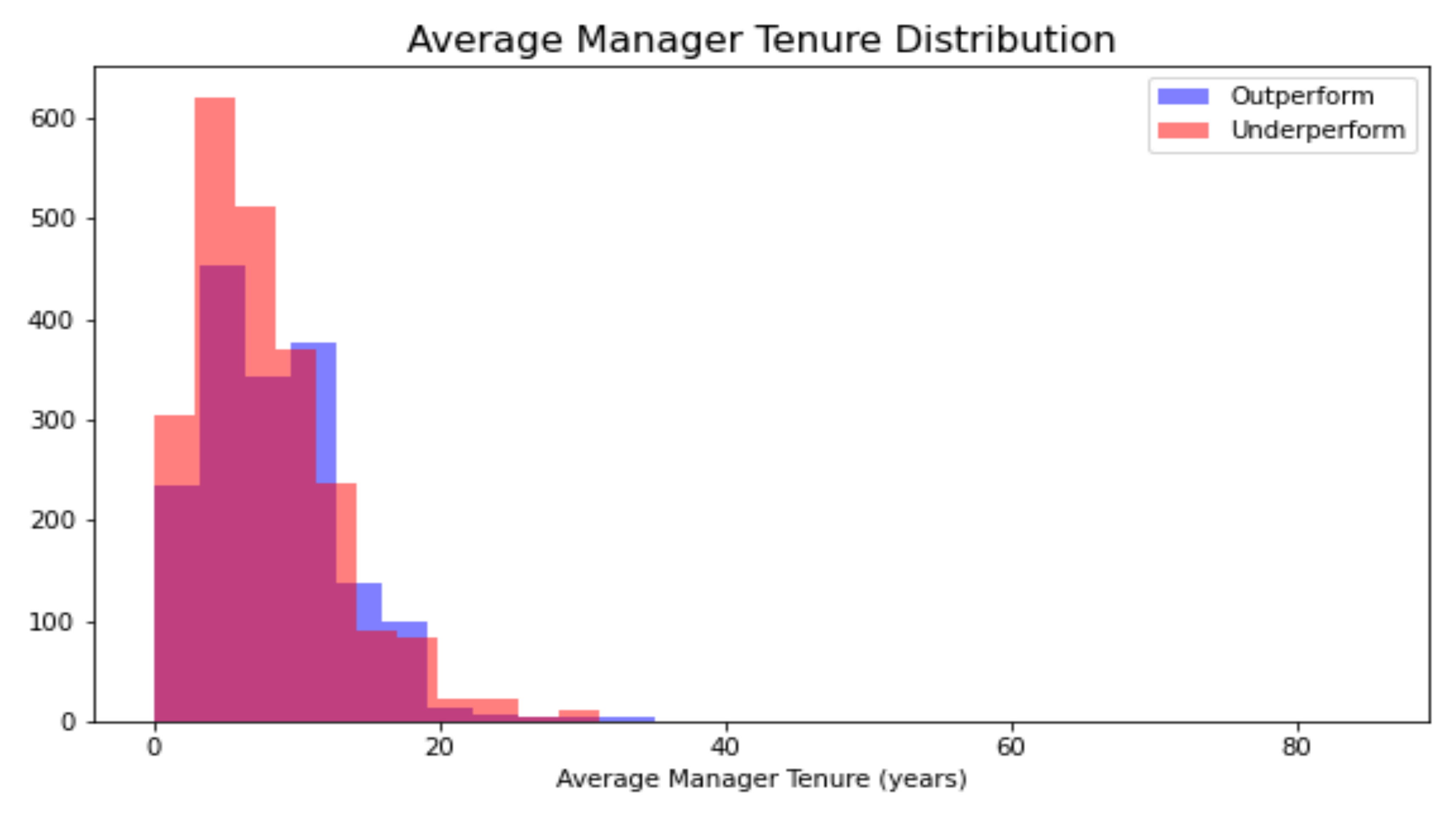
Selected Features:





Non-selected Features:



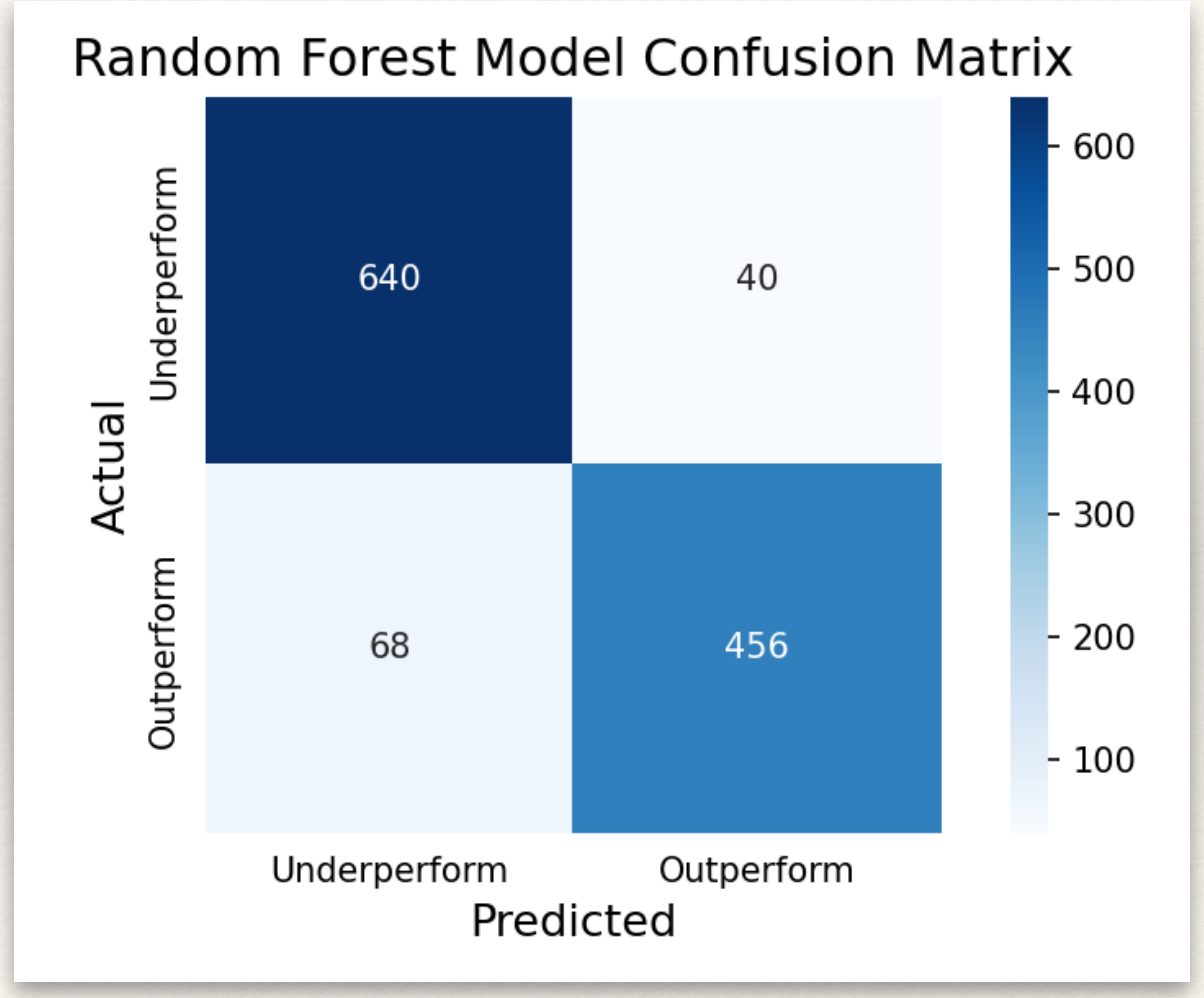


Choosing the Best Model

- Used GridSearchCV to cross-validate models and tune hyper-parameters

	Accuracy	Precision	Recall	F1 Score	ROC AUC Score	Log-loss
Logistic Regression	0.75	0.75	0.64	0.69	0.82	0.55
K-nearest Neighbors	0.95	0.94	0.94	0.94	0.96	1.40
Decision Tree	0.87	0.89	0.79	0.84	0.93	0.79
Random Forest	**0.96**	**0.96**	**0.95**	**0.95**	**0.99**	**0.12**
Support Vector Machine	0.93	0.92	0.90	0.91	0.94	NaN
Naive Bayes	0.72	0.80	0.45	0.58	0.83	0.66
XGBoost	0.95	0.94	0.95	0.94	0.98	0.16

Final Model Testing - Random Forest

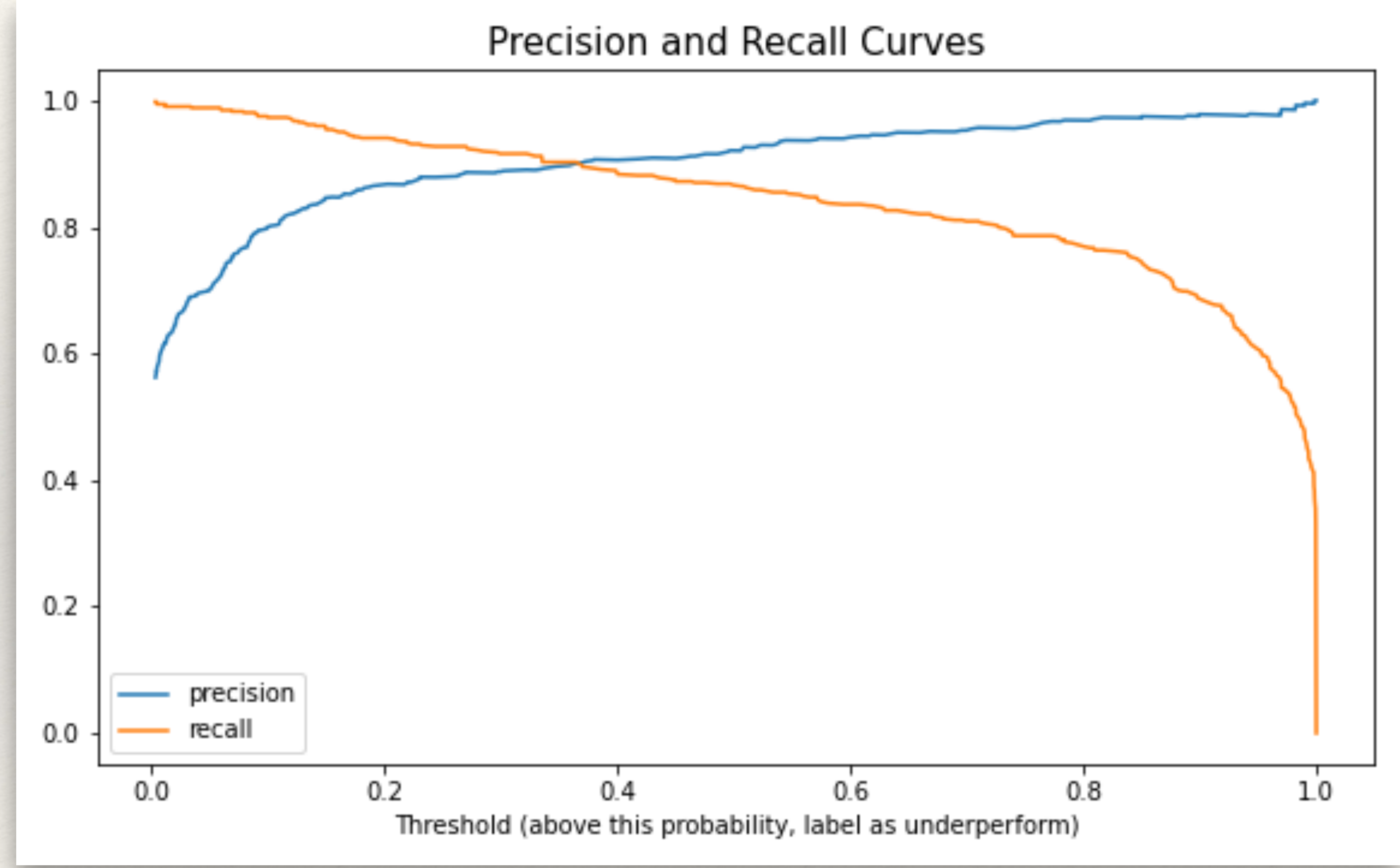


Model Test Results	
Accuracy	0.91
Precision	0.92
Recall	0.88
F1 Score	0.90
ROC AUC Score	0.97
Log-loss	0.21

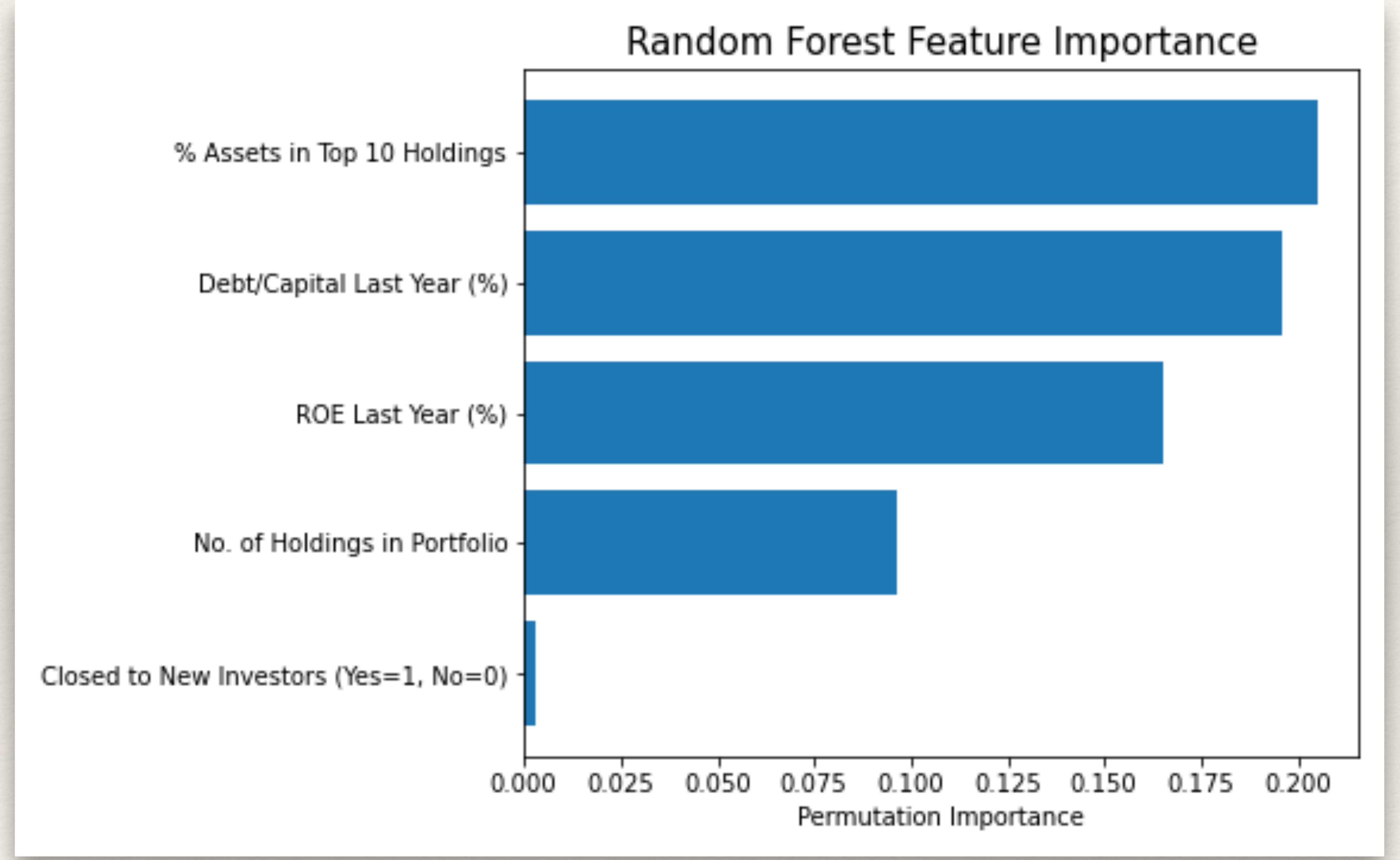
Precision vs. Recall - Investor Risk Appetite

- Increasing recall = minimizing the risk of missing out on outperforming funds at the expense of increasing the risk of selecting underperforming funds

- Precision/recall trade-off (i.e. threshold) depends on the **risk appetite of the investor**



Stock-pickers or Index Mimickers?



Increase in:	P(Outperform)	Favors Stock-picking?
% Assets in Top 10 Holdings	↑	Yes
Debt/Capital (%)	↓	Yes
ROE (%)	↑	Yes
# of Holdings in Portfolio	↓	Yes
Closed to New Investors = 1	↑	No

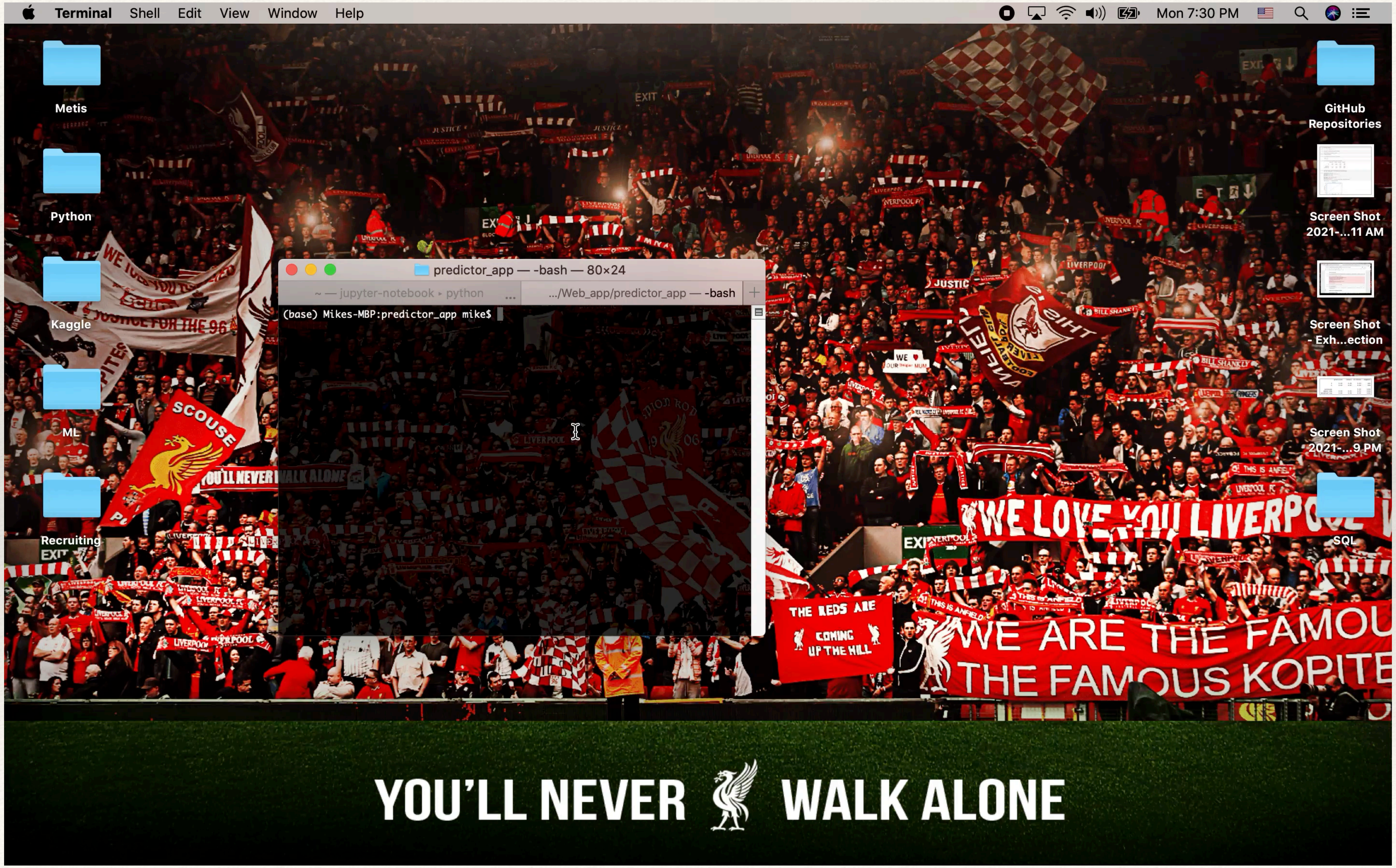
Limitations of Data/Model and Next Steps

Limitations	Next Steps
Most financial metrics and portfolio statistics are not static	Gather data on features from three years ago
Classification algorithms don’t capture the *degree* of outperformance/underperformance	Perform a linear regression analysis

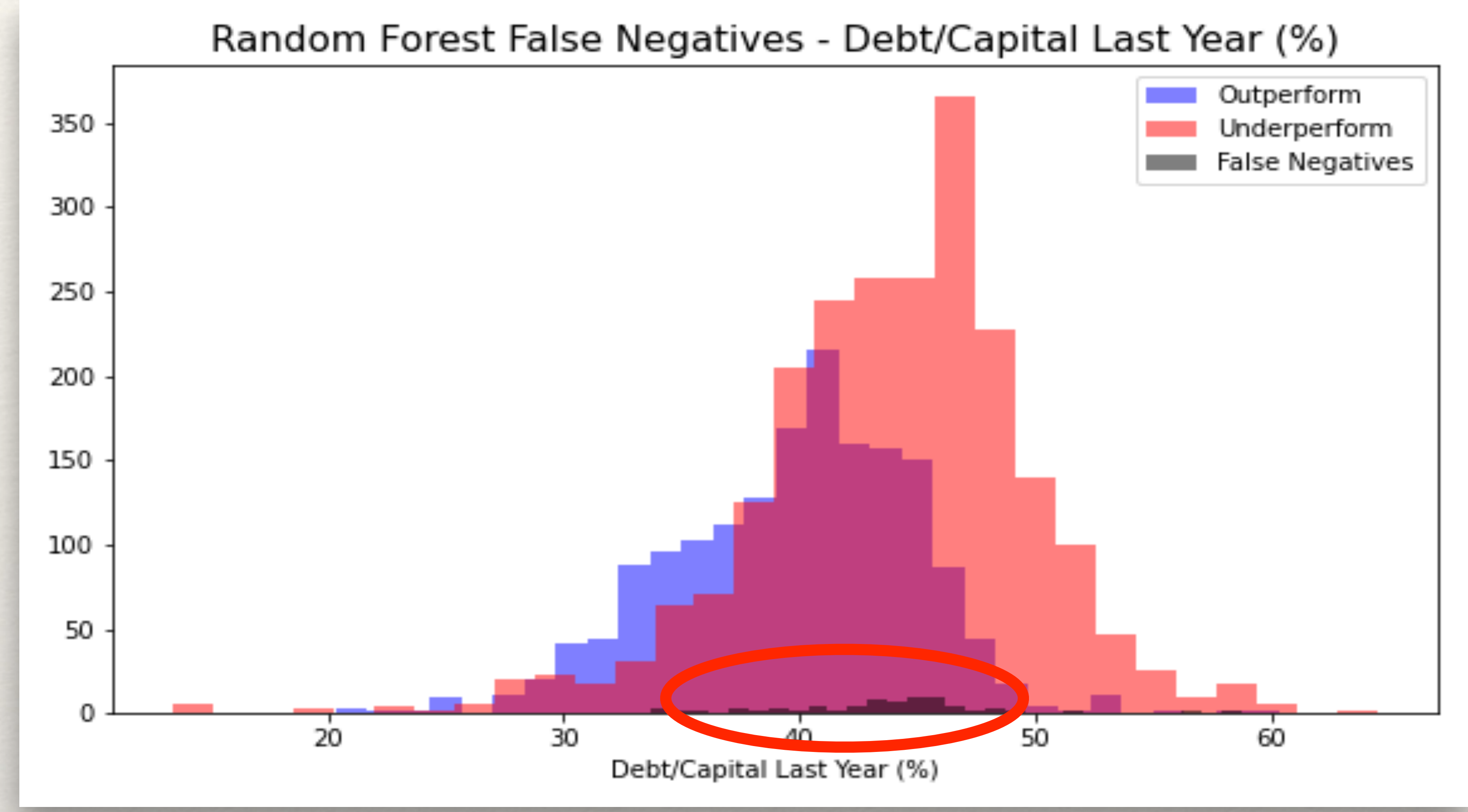
Thank you!

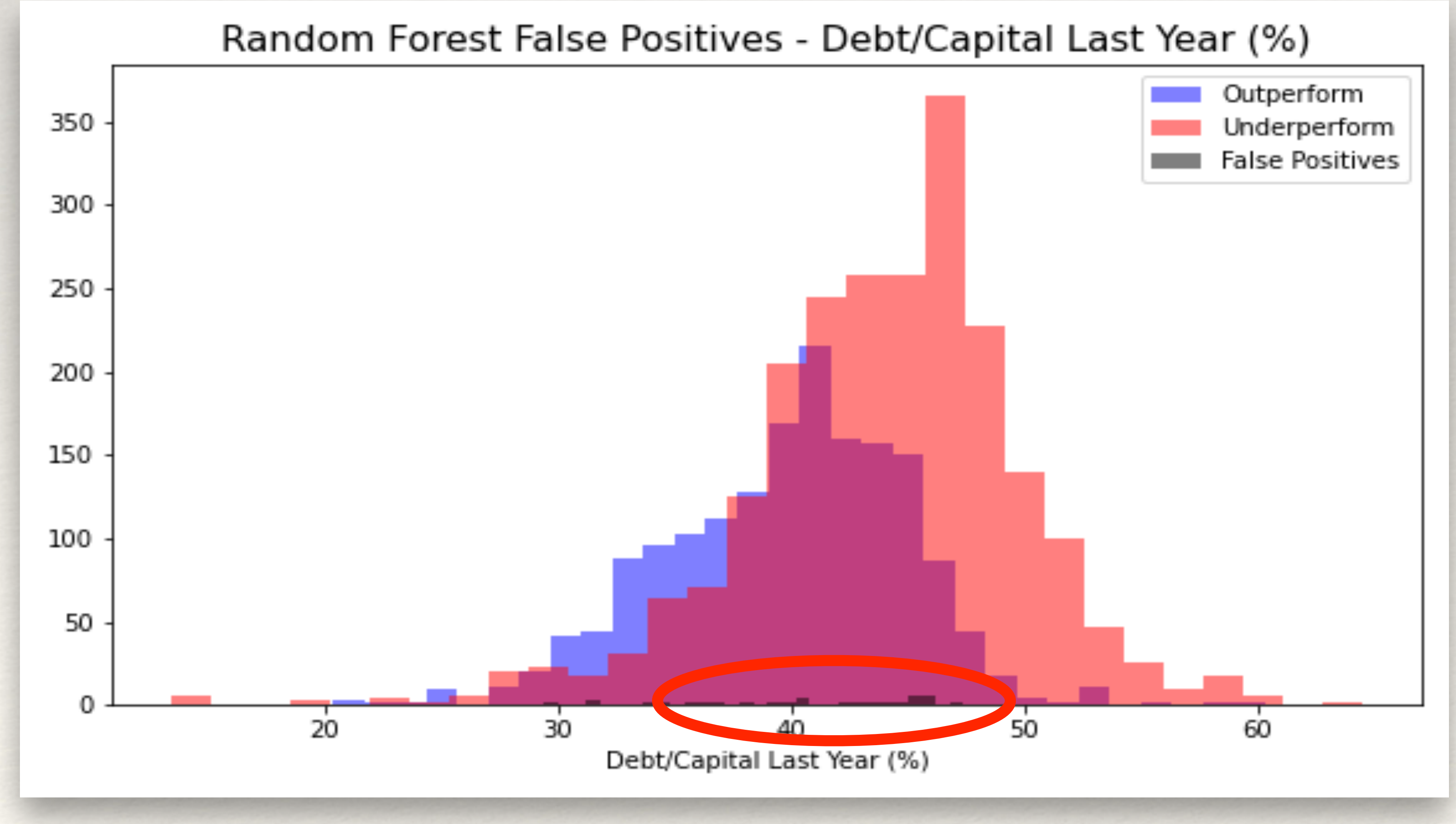
Questions?

Appendix I - Web App Demo



Appendix II - Error Analysis





- The false positives and negatives both tend to be distributed around the ranges where Outperform and Underperform overlap the most

- Many of the false positives had long manager tenure and above average expense ratio - further investigation needed

Appendix III - Sklearn Feature Selection Glossary

- **VarianceThreshold**: A simple baseline approach to feature selection. It removes all features whose variance doesn't meet some threshold (default = 0).
- **Chi-square test**: Measures dependence between stochastic variables, so using this function "weeds out" the features that are the most likely to be independent of class and therefore irrelevant for classification.
- **F test**: Computes the ANOVA f-value for each feature-target combination to look for any statistically significant relationship
- **Mutual_info_classif**: Measures the dependency between two random variables based on entropy estimate from k-nearest neighbors distances
- **Recursive feature elimination (RFE)**: Given an external estimator that assigns weights to features (e.g., through a feature_importances_ attribute), RFE elects features by recursively considering smaller and smaller sets of features
- **SelectFromModel**: A meta-transformer that removes the features considered unimportant if the corresponding coef_ or feature_importances_ values are below the provided threshold parameter. Compared to univariate feature selection (Chi-squared test, f-test, etc.), model-based feature selection consider all feature at once, thus can capture interactions.