EXECUTIVE SUMMARY

Corporate failure prediction is not new and has been the pursuit of many since the dawn of business intelligence in the 60's and 70's of the 20th century. Throughout the past few decades, as humanity went through multiple technological revolutions and financial turmoils, the ability to foresee financial failures always boiled down to two things – availability of *quality* data and *timeliness* of insights.

With the recent advances in computer science, enhanced collaboration, and knowledge sharing as well as the availability of big data, we believe that the inherent tradeoff between the quality and the timeliness of financial insights is becoming less significant. Analysts are now well-equipped to easily combine massive volumes of structured and unstructured data, and use various data engineering techniques to cure data for consumption analytics. We are now able to take advantage of Machine Learning and AI algorithms to predict potential liquidity crises with incredible precision.

The benefits of such solutions are numerous and include the following to name a few:

1. Data-driven AI techniques provide early warning signs of looming liquidity crises thereby empowering executive management to make informed decisions in time to change course before irreversible damage to the company financial health is incurred.
2. Creditors and investors (institutional or otherwise) will gain access to more reliable and higher quality data in time to make well-informed resource allocation decisions. AI-powered real-time analytical models that predict solvency, liquidity and profitability of investee organizations will influence lending and investment decisions.
3. Credit rating agencies can use AI models to adjust their ratings of debtholder organizations in a more timely manner. Here, the ratings evaluate debtors' ability to pay back debt by making timely principal and interest payments.
4. For large publicly-held companies, availability of timely insights is invaluable as any disclosure usually causes a wave of stock trading amongst investors. The investor relations department must be aware of current and upcoming issues that their organization may face, particularly those that relate to fiduciary duty and organizational impact. It must be able to assess the various patterns of stock-trading that a public company may experience, often as the result of a public disclosure or any research reports issued by financial analysts. The impact of financial disclosures on stock prices can be severe enough to justify an "AI health check".

The early signs of corporate failure can be evaluated through a number of accounting ratios and metrics, which are widely used in Finance:

1. **Liquidity ratios**[E1]: Liquidity ratios are used to determine a debtor's ability to pay off current debt obligations without raising external capital. Liquidity ratios measure a company's margin of safety and include current ratio, quick ratio, and operating cash flow ratio. Liquid assets include things like cash, money market instruments, and marketable securities.
2. **Leverage ratios**[E2]**:** A leverage ratio is financial measurements that look at how much capital comes in the form of debt (loans) or assesses the ability of a company to meet its

financial obligations. A leverage ratio may also be used to measure a company's mix of operating expenses to get an idea of how changes in output will affect operating income. Common leverage ratios include the debt-equity ratio, equity multiplier, degree of financial leverage, and consumer leverage ratio. Banks have regulatory oversight on the level of leverage they are able to have, as measured by leverage ratio.

A leverage ratio provides insights that too much debt can be dangerous for a company and its investors. However, if a company's operations can generate a higher rate of return than the interest rate on its loans, then the debt is helping to fuel growth in profits. Nonetheless, uncontrolled debt levels can lead to credit downgrades or worse. On the other hand, too few debts can also raise questions. A reluctance or inability to borrow may be a sign that operating margins are simply too tight.

3. **Efficiency ratios**[E3]**:** Efficiency ratios measure the ability of a business to use its assets and liabilities to generate sales. A highly efficient organization has minimized its net investment in assets, and so requires less capital and debt in order to remain in operation. In the case of assets, efficiency ratios compare an aggregated set of assets to sales or the cost of goods sold. In the case of liabilities, the main efficiency ratio compares payables to total purchases from suppliers. Efficiency ratios are used to judge the management of a business. If an asset-related ratio is high, this implies that the management team is effective in using the minimum amount of assets in relation to a given amount of sales. Conversely, a low liability-related ratio implies management effectiveness, since payables are being stretched. However, a low rate of liability turnover could be related to deliberate payment delays past terms, which could result in a company being denied further credit by its suppliers. Also, the desire to achieve a high asset ratio could drive management to cut back on necessary investments in fixed assets, or to stock finished goods in such low volumes that deliveries to customers are delayed. Thus, undue attention to efficiency ratios may not be in the long-term interests of a business.

4. **Profitability ratios**[E4]: Profitability ratios are another type of financial metrics used by analysts and investors to evaluate the ability of a company to generate income relative to revenue, balance sheet assets, operating costs, and shareholders' equity. They show how well a company utilizes its assets to produce value for shareholders. A higher ratio is commonly sought-after by most companies, as this usually means the business is performing well. The ratios are most useful when they are analyzed in comparison to similar companies or compared to previous periods (i.e., trends).

5. **Market value ratios**[E5]: Market value ratios are used to evaluate the current share price of a publicly-held company's stock. These ratios are employed by current and potential investors to determine whether a company's shares are overpriced or under-priced. The most common market value ratios are as follows:

    a. Book value per share: this measure is used as a benchmark to see if the market value per share is higher or lower, which can be used as the basis for buy or sell decisions.

    b. Dividend yield: This is the return on investment to investors if they were to buy the shares at the current market price.

    c. Earnings per share: This measurement does not reflect the market price of a company's shares in any way, but can be used by investors to derive the price they think the shares are worth.

    d. Market value per share: This reveals the value that the market currently assigns to each share of a company's stock.

    e. Price per earnings ratio: The resulting multiple is used to evaluate whether the shares are overpriced or under-priced in comparison to the same ratio results for competing companies.

    These ratios are not closely watched by the managers of a business, since these individuals are more concerned with operational issues. The main exception is the investor relations officer, who must be able to see the company's performance from the perspective of investors, and so is much more likely to track these measurements closely. Market value ratios are not applied to the shares of privately-held entities, since there is no accurate way to assign a market value to their shares.

The backbone of analysis of financial health risks is the following formula that shows relationship of key accounting variables to the main metric used for resource allocation decisions, the ROE, or Return on Equity capital:

$$ROE = \frac{Net\ Income}{Revenue} \times \frac{Revenue}{Total\ Assets} \times \frac{Total\ Assets}{Shareholder's\ Equity}$$

or, in other words:

$$ROE = Net\ profit\ margin \times Asset\ turnover \times Leverage\ ratio$$

These metrics and the five types of ratios mentioned above are the ones that we have found to be amongst the top 30 ratios with the highest predictive power as filtered by our AI algorithm utilizing recursive feature elimination technique.

All of these inputs are available on the financial statements of companies and are routinely included in the Annual Report book disclosed to all stakeholders. Each ratio and metric can have various degrees of impact and velocity on the health of the organization. It is often difficult to understand the overall effect of all the ratios and metrics on the company's health.

For example, high amounts of debt may not necessarily predicate bankruptcy if they are coupled with even higher profitability and liquidity. It may be that the organization has recognized a lucrative opportunity and has decided to borrow funds to go after it due to even higher returns. This is the kind of business leverage that is frequently used by companies to get to the next level of product development and maturity. If, however, the trend is such that higher degrees of debt do not translate into high profitability and liquidity then the business is in financial trouble and should act to recover by either improving efficiencies or decreasing leverage and seeking equity investment instead.

In summary, there is certain interrelationship between the metrics and ratios and sometimes the negative and the positive movements offset creating no meaningful changes in the direction of the company. Other times, the ratios interact to exacerbate negative trends, which drive financial performance of the company down over time. These can go unnoticed without a thorough analysis.

The above ratios and metrics are all important indicators of value proposition and lasting competitive advantage. Preparing financial statements and narratives to communicate this information to stakeholders takes weeks, particularly in larger organizations, and organizations with high degrees of social influence, where disclosures go through several rounds of revisions before being published. Review of financial statements, management analyses and narratives takes time, too. This time gap between when events take place and when they are being acted upon often results in poor decisions and a race after inside information to gain some advantage.

Team Humphrey's approach:

The Humphrey team is offering a solution to this problem by introducing the latest AI algorithms to allow identification and explanation of the nonlinear interactions between a wide range of accounting metrics and variables to get a sense of the most important combinations as well as their power to predict bankruptcy. We are confident that our approach will provide better and timely predictions compared to organizations who have not adopted the latest AI technologies and, hence, are unable to reap the benefits AI provides.

INTRODUCE THE DATASET

The dataset[D1] chosen for this project is acquired from Emerging Markets Information Service, EMIS[D2]. EMIS (formerly known as ISI Emerging Markets) was founded in 1994 by Harvard Business School graduate Gary Mueller. Inspired by the lack of information available to him while researching first as a Fulbright Fellow and later as a KPMG analyst in Eastern Europe, Gary started the company with the purpose of providing easy access to critical business information and research on emerging markets. EMIS was acquired by Euromoney Institutional Investor PLC in 1999. In 2018, EMIS and its sister company CEIC were acquired by CITIC Capital Partners and Caixin Global. As a result of this acquisition EMIS and CEIC now form ISI

Emerging Markets Group. Today, EMIS is a company that employs nearly three hundred people in 13 countries around the world, catering to nearly 2,000 clients[D3]. EMIS now operates in and reports where high reward goes hand-in-hand with high risk. Using the relationships it has forged with local providers of news, analysis and data, EMIS brings time-sensitive, hard-to-get, relevant news, research and analytical data, peer comparisons and more for over 145 emerging markets.

The dataset is about bankruptcy prediction of Polish companies. The data contains companies that went bankrupt between 2000 and 2012. These companies were analyzed from 2007 to 2013 while still fully operational. The data came in five separate datasets that were downloaded separately and named “1stYear”, “2ndYear”, “3rdYear”, “4thYear” and “5thYear”. Based on the collected data, five classes were identified based on the forecasting period:

1stYear: The “First Year” dataset contains financial ratings for the companies that are anticipated to go bankrupt after five years from the stated date. The data contains 7027 observations (financial statements), 271 represents bankrupted companies, 6756 firms that did not go bankrupt in the forecasting period.

2ndYear: The “Second Year” dataset contains financial ratings for the companies that are anticipated to go bankrupt after four years from the stated date. The data contains 10173 instances (financial statements), 400 represents bankrupted companies, 9773 firms that did not go bankrupt in the forecasting period.

3rdYear: The “Third Year” dataset contains financial ratings for the companies that are anticipated to go bankrupt after three years from the stated date. The data contains 10503 instances (financial statements), 495 represents bankrupted companies, 10008 firms that did not go bankrupt in the forecasting period.

4thYear: The “Fourth Year” dataset contains financial ratings for the companies that are anticipated to go bankrupt after two years from the stated date. The data contains 9792 instances (financial statements), 515 represents bankrupted companies, 9277 firms that did not go bankrupt in the forecasting period.

5thYear: The “Fifth Year” dataset contains financial ratings for the companies that are anticipated to go bankrupt after one year from the stated date. The data contains 5910 instances (financial statements), 410 represents bankrupted companies, 5500 firms that did not go bankrupt in the forecasting period.

The dataset is real, multivariate with 64 features and the target is the Bankruptcy boolean class, which clearly makes the dataset imbalanced(please refer to Table 1 below). The dataset is imbalanced because the ratio of bankrupt to operating companies is very low. It also contains many missing entries resulting in data loss of more than 50% across the five datasets (please refer to Table 2 below):

Year	Operating	Bankrupt
1	6756	271
2	9773	400
3	10008	495
4	9277	515
5	5500	410

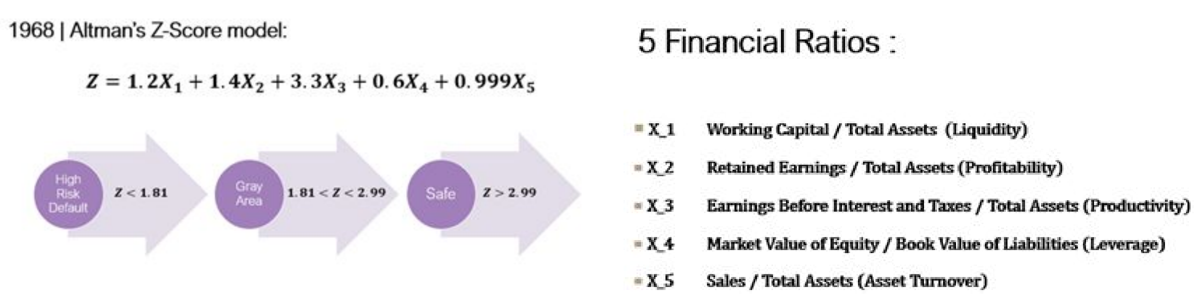
Table 1: Imbalanced data: operating vs. bankruptcy observations

	Total Instances			
Year	Observations	Missing Values	With Values	Data loss %
1	7027	3833	3194	54.55
2	10173	6085	4088	59.82
3	10503	5618	4885	53.49
4	9792	5023	4769	51.30
5	5910	2879	3031	48.71

Table 2: Data incompleteness, year-based datasets

EXISTING METHODS

Much research work was published over the years focusing on bankruptcy prediction of public firms. The area of research is indeed important (as described in the Executive Summary) and is well suited for testing and refining of increasingly sophisticated, data-driven predictive techniques. Bankruptcy prediction has been the subject of research for over 50 years. The common technique to assess likelihood of corporate failure is called Altman's z-score, and was created by Edward I. Altman in 1968. It is still used today. The score is based on five financial ratios that can be calculated from data found on firms' financial statements and Annual Reports. It predicts whether a company has a high probability of becoming insolvent within the next few years and is based on statistical technique called multiple discriminant analysis (MDA).



This formula represents logistic regression with five variables weighted according to their predictive power in assessing future bankruptcy. One of the known issues with the formula is that it does not consider adjustments based on industry, maturity or size of the organization whereas an acceptable z-score metric ranges widely from industry to industry and across organizations of various sizes.

In 1980, James Ohlson applied logistic regression[M1] to a much larger dataset. However, the resulting models have consistently been predicting bankruptcies with lower than 80% success rate. To improve prediction power organizations would need to run different models by industry, size and maturity of the organization. This is tedious and time-consuming.

Neural network models have also been tested on bankruptcy prediction. Modern techniques applied by business information firms surpass the traditional ones, which are based on financial statement data, and take into account current events such as age, judgements, bad press, payment incidents and payment experiences from creditors.

In the paper by Barboza, Kimura and Altman from 2017 called “Machine learning models and bankruptcy prediction”[M2] the research team compared 8 different AI algorithms on 13,300 firms’ datasets. These algorithms included:

- SVM-Linear
- SVM-RBF
- Boosting
- Bagging
- Random Forest
- Neural Networks
- Logit
- MDA

The research team has appropriately selected areas under the ROC curve (AUC) and total estimated accuracy (ACC) as evaluation metrics. The most common issues with these approaches as well as traditional techniques are: **unscaled inputs, missing data, imbalanced datasets, inadequately tuned models, and no collaboration between models**.

Taking into account the fact that most investors and creditors are more concerned with Type I error(companies that went bankrupt but were not predicted to), testing results (Figure 1 below) show an interesting dynamic. Because Neural Nets and SVM were heavily over-classifying, these algorithms managed to catch almost all of the future bankruptcies keeping Type I error low as compared to Boosting, Bagging and Random Forest algorithms (at 6.77% and 7.52% compared to 18.80%, 17.29% and 16.54%, respectively). The overall accuracy was lowest for Random Forest technique, which, to an extent, is due to an imbalanced dataset. Neural Nets handled this problem the best and correctly predicted most of the future bankruptcies.

Referring to Figure 1 below, it is clear that Random Forest has the lowest Type II error (12.9%) on the test set. Although minimizing Type I error is very important, minimizing Type II error (companies that didn’t go bankrupt but were predicted to go bankrupt) is also important because we want to minimize unfair treatment towards healthy firms aiming to secure funding and obtain lower credit rates. Investors and creditors might not be willing to lend to them based on the misclassifications.

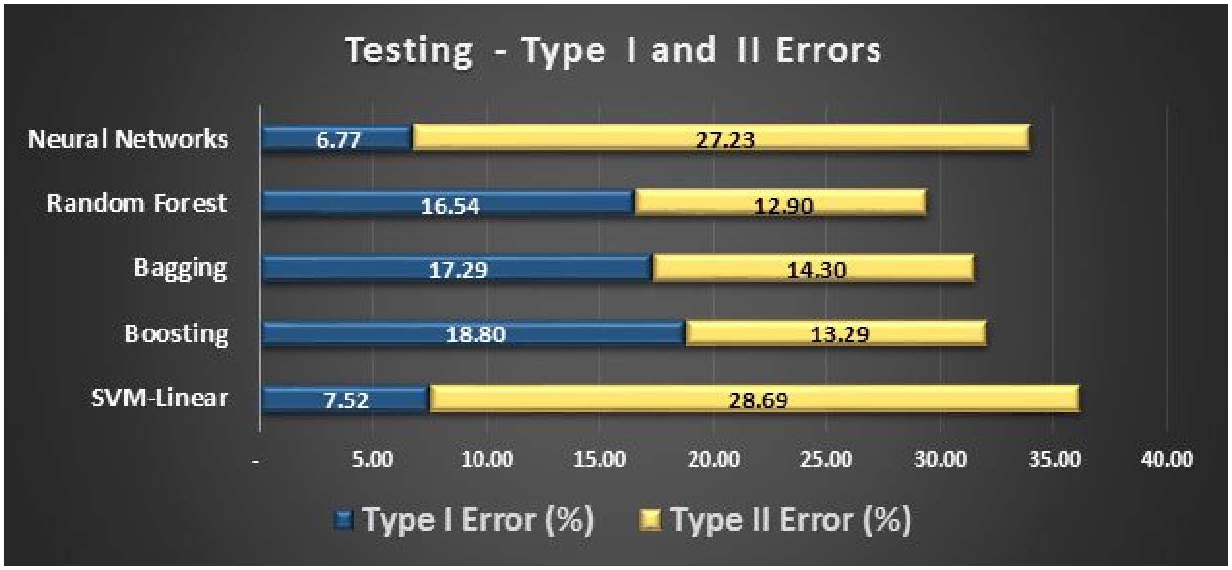


Figure 1: Type I and II errors on the testing set of Altman's 2017 paper

The outcome of this research work shows that traditional models (MDA and logistic regressions) have lower predictive capacity (52% to 77%) compared to machine learning models (71% to 87%) suggesting that machine learning can be used to significantly improve corporate credit risk assessment and bankruptcy prediction.

OUR APPROACH

In this section we will first define the problem we aim to solve, state our hypothesis prior to the experiment, any assumptions we have made, and describe in detail our novel pipeline from start to finish.

Problem Statement

Building on top of the paper "Machine learning models and bankruptcy prediction" [1] published in 2017. Our team aims to more accurately predict company bankruptcies by addressing a few potential issues the paper failed to tackle:

- Unscaled inputs
- Observations with missing values
- Class imbalance
- Models weren't fine tuned
- No collaboration between models
- Lack of feature engineering

Hypothesis

1. If the 6 issues mentioned above are properly addressed, we expect to significantly outperform the results achieved by the paper.
2. After training on 1 of the 5 datasets (the dataset that predicts company bankruptcy 5 years in the future) we expect our model to be robust enough to perform at least equally well across each of the remaining 4 datasets without any additional training.

Assumptions

- Companies in each of the 5 datasets follow a similar distribution with each other for bankrupt and non-bankrupt companies
- All missing values within each dataset are missing due to human error
- All existing values within each dataset are all correctly entered

Our Novel Pipeline

1. Select dataset "1year.arff" for modeling
   We believe predicting company bankruptcies 5 years in the future is the most challenging when compared to the other 4 datasets. "1year.arff" was selected to build models on.
2. Label each of the 64 features
   Instead of feature names such as "X1", "X2", etc. Each feature is relabeled to the name they actually represent, such as "net profit / total assets", "EBIT / total assets", etc.
3. Removing any completely duplicated rows
   If more than one observation (i.e. data row) has completely the same value for every feature, only one of these observations is kept.
4. Standardize every feature
   In order to transform data in every feature to a similar mean and variance while maintaining their relative distances within each feature, StandardScaler was applied towards each of the 64 features. StandardScaler automatically adjusts every feature to have a mean of 0 and a variance of 1. StandardScaler was chosen over MinMaxScaler due to MinMaxScaler compressing all inliers in a very narrow range while StandardScaler spreads them better in comparison. [2]
5. Impute missing values with KNN Imputer
   As 3,794 rows (54.6%) out of 6,945 rows contain some features with null values, instead of dropping all rows with missing values, each observation's missing values are imputed using the mean value from 7 of their nearest non-null neighbors. [3] Since KNN Imputer is based on euclidean distances, step 4 was mandatory before this step.
6. Remove features with low variances
   Since features with extremely low variance seldom help machine learning models to perform better, every feature with less than 10% variance was targeted for removal. For this dataset, however, no feature was required to be removed at this step.
7. Correlation Thresholding with Maximal Information Coefficient(MIC) [4]
   Correlation thresholding removes features that are highly correlated with others because these features provide redundant information. First, we calculate correlations between every pair of features. Then, if the correlation between a pair of features is above 90%, we only keep one of these features. Usually Pearson's Correlation is used to measure the correlation between features; however, since we want to also look into features that are correlated non-linearly, MIC (refer to Figure 2) was chosen as the correlation measure because it can identify linearly and non-linearly correlated features. MIC rates the correlation from 0 to 1 where 1 means completely correlated. 18 features were removed during this step, with 46 features remaining.
8. Un-standardize every feature
   These 46 remaining features are unstandardized and restored to their original unscaled values.
9. Automated Feature Engineering [5]

Since we wanted to explore different combinations between existing features. We have used an open-source package called 'Featuretools' to automatically generate every possible addition, subtraction, multiplication, and division combination between every pair from the remaining 46 features. As a result, we generated 5,175 new features. From these new features, we removed any feature that contains null or infinity values. After combining the remaining generated features with the 46 original features, we now have a total of 3,601 features. Since addition, subtraction, multiplication, and division only make sense when performed on unscaled values, step 8 was required prior to this step.

10. Re-standardize every feature
    Similar to step 4, StandardScaler was re-applied towards each of the 3,601 features. This evened out the potential weight differences for each feature, allowing for a less biased model to be built in the steps to follow. [6]
11. Train-Test Split
    The entire dataset gets separated into the training set and the testing set such that the testing set (refer to Figure 3):
    - contains no imputed values in order to avoid leakage
    - consists of 20% of the entire dataset, which is sufficient for testing
    - maintains the same ratio of class imbalance as the entire dataset in order to better represent real-world data
12. Tackle Class Imbalance in the Training Set [7]
    Since our classes are extremely imbalanced (with 261 observations labeled as bankrupt and 5,295 observations labeled as non-bankrupt in the training set) we need to address imbalance issues before trying to build a model, otherwise performance will be negatively impacted. As we only have 5,295 in the majority class, oversampling was selected instead of undersampling. Adaptive Synthetic Sampling (ADASYN) was chosen as our oversampling technique due to it being state-of-the-art and popular amongst the data science community. We now have 5,256 observations labeled as bankrupt and 5,295 observations labeled as non-bankrupt in the training set.
13. Model Selection
    Based on the current popularity amongst the data science online community, we have chosen 5 machine learning supervised classifiers to predict bankruptcies:
    RandomForest (rfc), AdaBoost (adb), GradientBoost (gdb), ExtraTrees (etc), and SVM (svc).
14. Recursive Feature Elimination [8] (this is performed for each of the 5 models in step 13)
    To cut computational cost during the training phase, we select only the top 30 features by recursively considering smaller and smaller sets of features from the 3,601 features. Every recursion removes only 10% of the remaining features. Please note that the resulting 5 lists of 30 features are slightly different because we used different models to reduce features from 3,601 to 30.
15. Parameter Tuning with Cross-Validation (this is done for each of the 5 models in step 13)
    30 minutes (on a laptop with i5 7th Gen CPU) of Randomized Search with 20-fold cross-validation [9] was done for each of the 5 models based on their unique top 30 features. The model with the highest average ROC-AUC score was selected. The reason behind using cross-validation was to minimize overfitting. ROC-AUC was selected as the performance metric due to our training data being quite well balanced after oversampling [10].
16. 2nd level stacking trained with XGBoost

With 5 tuned models, we now turn every prediction probability into a feature of a new dataset. By predicting every observation inside training and testing set with each of the 5 models, we have essentially generated a new training and testing set each with 5 features. Target columns for these new training and testing sets remain the same as before (refer to Figure 4). We then train XGBoost Classifier (stacked) for 30 minutes with 20-fold cross-validation on this new training set. Now we are ready for prediction!

OUR RESULTS

In this section, we review all the results in-depth and compare them to our hypothesis.

Test Set Performances from year1.arff
Referring to Table 3 and Figure 5, when comparing type I and type II errors, we can see that GradientBoost performed the best on the test set with RandomForest and Stacked XGBoost tied in close second place. One interesting find is that the stacked model doesn't always perform better than every one of its children models. Another interesting find is that the two False Negatives missed by RandomForest, GradientBoost, and Stacked XGBoost are identical. This either means:

- these 2 observations behaves differently than every bankrupt observations seen in the training set, therefore by training on the entire dataset, type I error will continue to decrease below 20%
- these 2 observations are actually non-bankrupt and mislabeled, therefore the type I error is actually 0%

Comparing this finding with our first hypothesis, we have definitely managed to reduce the type II error significantly down to less than 1%.

Model	TP	TN	FP	FN	Type I %	Type II %	ACC %
rfc	**8**	1365	14	**2**	**20**	1.02	98.85
adb	7	1363	16	3	30	1.16	98.63
gdb	**8**	**1367**	**12**	**2**	**20**	**0.87**	**98.99**
etc	3	1349	30	7	70	2.18	97.34
svc	4	945	434	6	60	31.47	68.32
stacked	**8**	1365	14	**2**	**20**	1.02	98.85

Table 3: Test Set performances on year1.arff over different models with threshold = 0.5

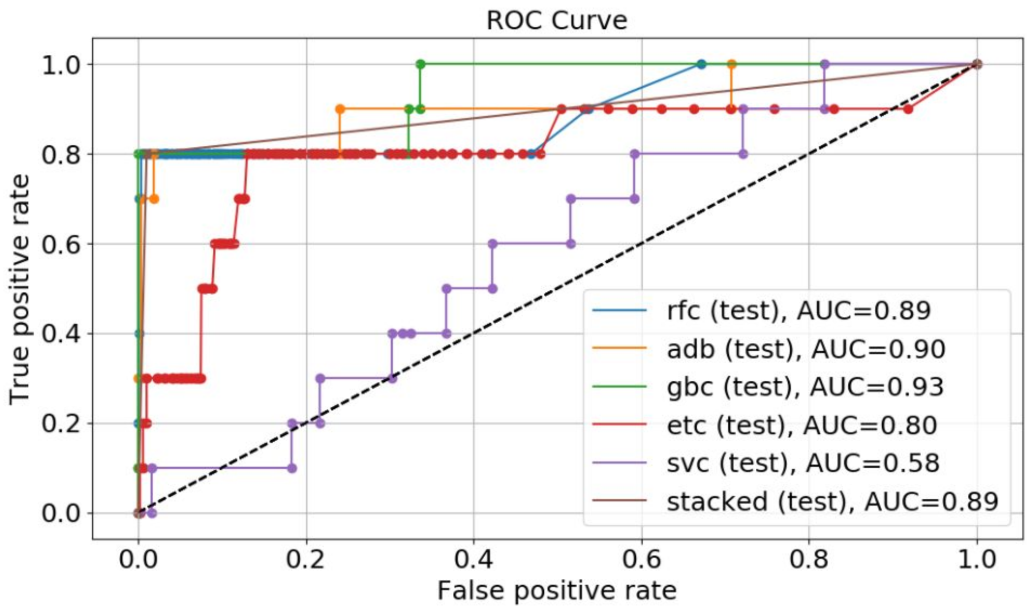


Figure 5: Test Set ROC curves on year1.arff over different models

Performances from year2.arff, year3.arff, year4.arff, year5.arff
We then proceeded to use these 6 trained and tuned models to try to predict bankruptcies on the remaining 4 datasets, treating all non-null observations as potential test sets. From Figure 6, we have observed that as we use the same models to predict bankruptcy of companies from 5 year in the future to 1 year in the future, the models on average gradually performed worse. This contradicts with our second hypothesis likely because the behaviors of companies about to go bankrupt next year is very different than the behaviors of companies about to go bankrupt in the next 5 years.

RECOMMENDATIONS

We believe that this model would be of great interest to banks and other financial institutions that lend money, provide OTC derivatives, or engage in long-term contracts with large corporations. Any situation where predicting a bankruptcy ahead of time could give the institution time to react and mitigate the impact will benefit from our model.

The complexity of this model is such that deploying it would require an established IT infrastructure and governance teams to support it. It is not meant as a quick metric calculation such as Altman's Z-score, nor as a lightweight Excel add-in. We see two primary options:

1. deployed as a new model, or enhancement to an existing model, into the in-house risk infrastructure (e.g. to the tools already utilized by the Credit Risk and Credit Adjudication teams), or
2. as a cloud-based Software-as-a-Service (SaaS) model with a suitable web-based interface or API.

Under either of the options, a robust maintenance, retraining/retuning and backtesting processes must be established, well-documented and properly maintained on an ongoing basis to ensure the results remain as accurate as possible. This will require a small number of data scientists to periodically:

1. Perform ongoing QA on the results of the model
    a. Evaluate the performance on a weekly basis
    b. Analyze quality of data inputs
    c. Analyze false positives and false negatives
    d. Provide a quarterly report with conclusions on performance
2. Retrain/retune the model
    a. Retrain/retune on a monthly basis
    b. Analyze updated model's new performance
    c. Deploy the model to the production environment
3. Governance & Control
    a. Upkeep all documentation
    b. Comply with all internal and external regulations, both existing and upcoming
    c. Handle any issues/exceptions
4. Enhancements
    a. Based on 3 points above, create requirement documents for new enhancements
    b. Perform testing on enhancements
    c. Provide recommendation to deploy to production

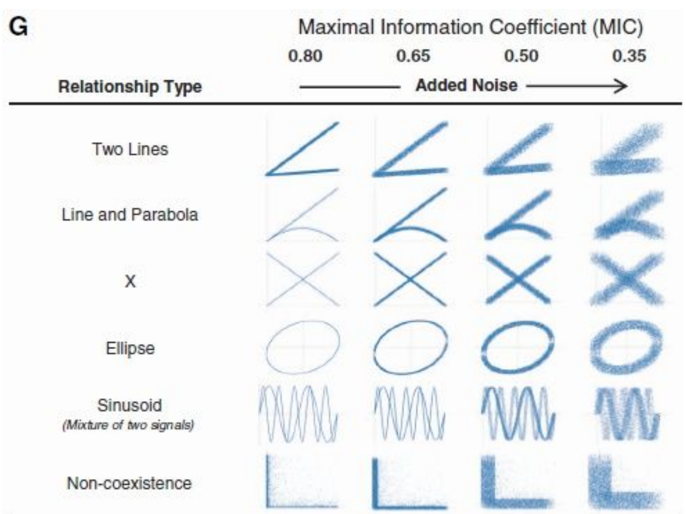
Establishing a solid maintenance and upgrade framework as per recommendations listed above will ensure the ongoing performance of this model remains as high as possible.

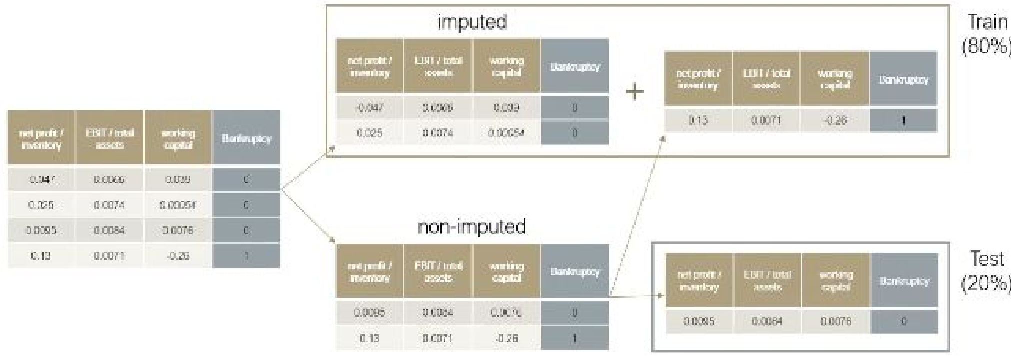
NEXT STEPS

There are a few areas worth exploring as it applies to modeling:

1. In addition to classic supervised Machine Learning, there have been recently developed and published techniques that show great promise. One of them is TabNet by Google (TabNet: Attentive Interpretable Tabular Learning -- https://arxiv.org/abs/1908.07442) which shows excellent results in tabular datasets such as the one used for this project.
2. Another idea to explore is reshaping this problem into a semi-supervised "one-class" anomaly detection task. For bankruptcies, the detection algorithm will try to identify companies with a combination of features (i.e. metrics) that are outside of the normal range of non-bankrupt companies. This way, any company with unique metrics could be flagged for potential review. The downside could be an increase in false positives rate as not all companies with unique characteristics are bankrupt.
3. The modeling used for this project was influenced and geared towards the dataset available, which comprised metrics of companies at a point in time. Given more data, especially across the temporal dimension, we can either explore additional types of models (RNNs such as LSTMs) or engineer better features that take into account the direction and the change of metrics over time.

APPENDIX

[E1]	https://www.investopedia.com/terms/l/liquidityratios.asp
[E2]	https://www.investopedia.com/terms/l/leverageratio.asp
[E3]	https://www.accountingtools.com/articles/efficiency-ratios.html
[E4]	https://corporatefinanceinstitute.com/resources/knowledge/finance/profitability-ratios/
[E5]	https://www.accountingtools.com/articles/market-value-ratios.html
[D1]	https://archive.ics.uci.edu/ml/datasets/Polish+companies+bankruptcy+data
[D2]	https://www.emis.com/
[D3]	https://www.emis.com/about
[M1]	https://www.jstor.org/stable/2490395?seq=1
[M2]	https://dl.acm.org/doi/10.1016/j.eswa.2017.04.006
Figure 2	 A demonstration of MIC towards non-linear relationships https://www.kdnuggets.com/2011/12/reshef-mine-mic-maximal-information-coefficient-software-exploredata.html

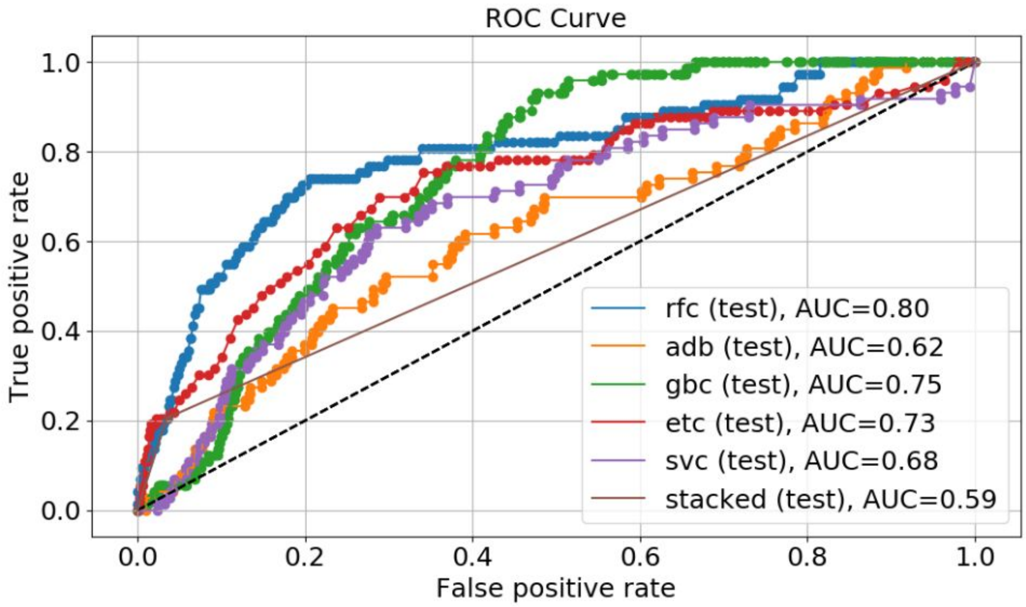
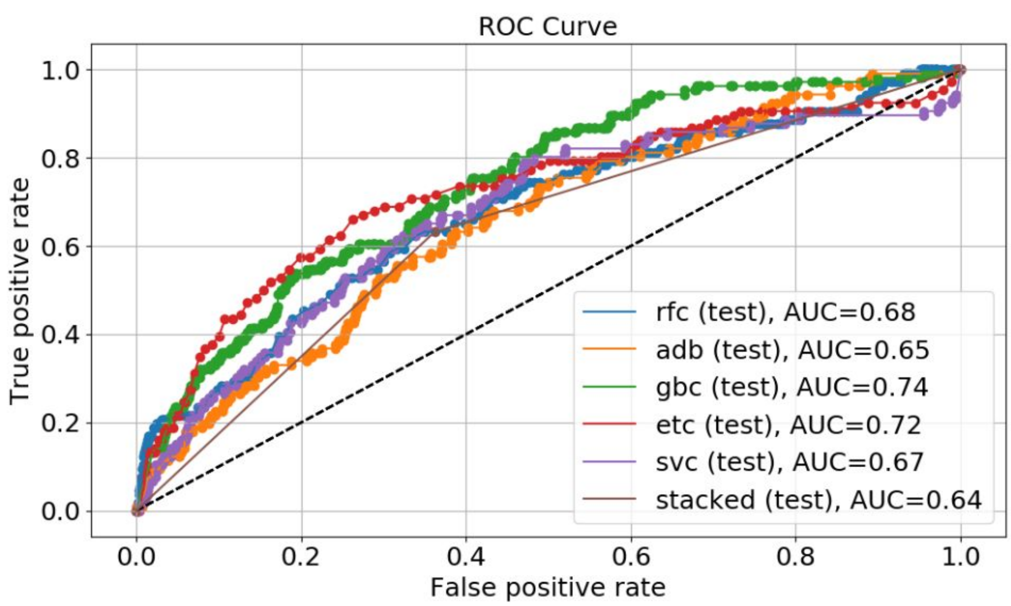
<table>
<tr><td>Figure 3</td><td>


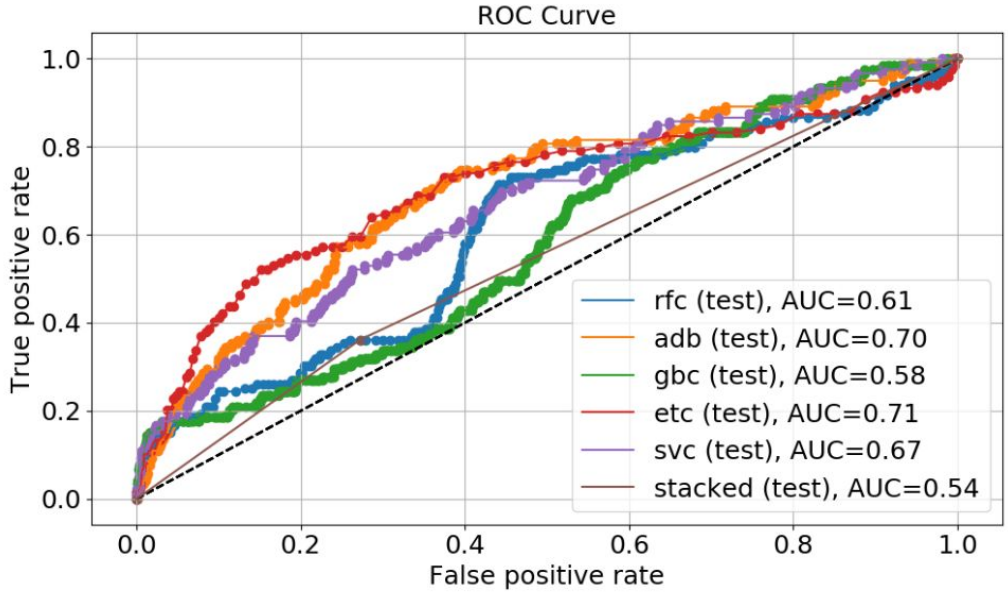
A demonstration of how the data got split into train and test</td></tr>
<tr><td>Figure 4</td><td>
<table>
<tr><th>Random Forest Prediction</th><th>AdaBoost Prediction</th><th>GradientBoost Prediction</th><th>ExtraTree Prediction</th><th>SVM Prediction</th><th>Target</th></tr>
<tr><td>0.2</td><td>0.1</td><td>0.1</td><td>0.5</td><td>0.3</td><td>0</td></tr>
<tr><td>0.1</td><td>0.3</td><td>0.1</td><td>0.4</td><td>0.3</td><td>0</td></tr>
<tr><td>0.9</td><td>0.8</td><td>0.9</td><td>0.7</td><td>0.6</td><td>1</td></tr>
<tr><td>0.8</td><td>0.9</td><td>0.9</td><td>0.7</td><td>0.2</td><td>1</td></tr>
</table>
A demonstration of the new dataset for training and testing</td></tr>
</table>

[1]	https://www.sciencedirect.com/science/article/abs/pii/S0957417417302415?via%3Dihub
[2]	https://scikit-learn.org/stable/auto_examples/preprocessing/plot_all_scaling.html
[3]	https://scikit-learn.org/stable/modules/generated/sklearn.impute.KNNImputer.html
[4]	https://medium.com/analytics-vidhya/feature-selection-methods-with-code-examples-a78439477cd4
[5]	https://towardsdatascience.com/why-automated-feature-engineering-will-change-the-way-you-do-machine-learning-5c15bf188b96
[6]	https://medium.com/@swethalakshmanan14/how-when-and-why-should-you-normalize-standardize-rescale-your-data-3f083def38ff
[7]	https://en.wikipedia.org/wiki/Oversampling_and_undersampling_in_data_analysis
[8]	https://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.RFE.html
[9]	https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.RandomizedSearchCV.html

[10]	https://towardsdatascience.com/the-ultimate-guide-to-binary-classification-metrics-c25c3627dd0a#a1a2

Figure 6: ROC Curves from models trained on 1year.arff evaluated on 4 unseen datasets

Dataset	ROC Curve from the model trained on 1year.arff
2year.arff	
3year.arff	

4year.arff	
5year.arff