Prediction of Credit Defaults based on Weight Dimensionality Reduction Neural Network and M-Band Discrete Wavelet Transform

Alejandro Antonio Mayorga, Letian Wang

Abstract

Credit Defaults of Companies are of utmost importance for many people such as investors that may include the general public, entrepreneurs looking to fund the next Apple or companies looking to support small companies. These Defaults can be modeled using financial information ,however there are too many companies in the world to perform a careful analysis on each one. A system which is already trained and takes financial information and returns a credit Default is the best for this task.Firstly, We implemented M-band discrete wavelet transform (MDWT) to decompose our dataset into M different frequency components to discover some hidden information we would not get otherwise In this paper we propose a novel weight dimensionality reduction neural networks (WDR-NN) that uses dimensionality reduction techniques such as UMAP, Wavelets, PCA, convolutions and Max Pooling to generate a new neural network and then pass relevant information that is in a reduced dimension but preserves the overall structure of the networks weights.. In our research, two Datasets were used: England Companies Binary Classification of whether the company went bankrupt at some point and Moody's and Fitch Credit Defaults using binary classification to see if their rating by the agency was higher than a given threshold. The results have shown that our WDR-NN model outperformed a regular neural network by yielding a 13% accuracy increase in predicting Company fraud as binary classification. We utilize Shapley Values on our WDR-NN and find that Operating Cash Flow Share and Days of Sales outstanding are the two most important features in determining a company's default. It beats traditional methods such as Least Discriminant Analysis, Logistic Regression, Decision Trees and Support Vector Machine in various metrics on both datasets.

1.Introduction

In today's day and age, it is crucial for investors and anyone in general to have a numerical way to measure their risk of financial decisions. As the 21st century has evolved, new technology has entered its way into the world in many fields. One of these fields that was most changed from not only the new technology but also academic achievements in fields such as mathematics and Machine Learning is finances. From the stock market to banks to company analysis and public information, the horizon to make money in the world of finance is a lot broader now. This comes at a price however as decisions have to be made a lot more calculated as one bad financial decision could result in the loss of a monumental amount of money. For investors,companies and even just the common person it is crucial to determine the risk of their financial decision. When it comes to companies, the risk is even greater as the amount invested in companies is a vast amount on average. A way to numerically measure these risks which are provided by the government are credit Defaults of a company. These Defaults are of utmost importance because even one misrating could lead to losing a monumental amount of money. The leading companies for Credit Ratings are Moodys, S&P 500 and also ESG Scores (Provided by the government). All of these scores by law must be made public. When it comes to credit Defaults, these rating agencies must be efficient as they cannot spend a lot of resources on one company as they also have to rate small companies. A system simple enough which can be explained to the general public but also complex enough to capture deep relationships between features of a company and which can generate predictions quickly is a very ideal system for these rating agencies. A framework which is perfect for this is machine learning. As mentioned above, technology has rapidly evolved. 2 benefits of this revolution have really changed the field of machine learning.

1. A substantial increase in high quality data that is accessible to the general public and easy to understand.

2. Substantial Processing Power as the invention of the GPU and CPU and microprocessors has decreased the time mathematical operations can be done by a computer.

The field of deep learning has boomed under this revolution as methods from this field rely heavily on data and with more data, comes the better understanding of the fundamental relationships underlying the data. Neural networks are a type of Deep Learning Algorithm which comes to mind when solving problems like these. As neural networks can be optimized through hyperparameter tuning and feature engineering and many other features which can be optimized for the best result, these types of machine learning will learn complex relationships between data which humans cannot learn. Many will claim neural networks are a black box but the mathematics behind these networks are simple to understand once you have mathematical understanding, many other machine learning algorithms become simple to understand. Their decisions are difficult to explain but using methods such as Shapley Values, we can assess the Networks decision and explain them. Linear Algebra and Calculus along with Statistics are the main mathematical frameworks which are needed to understand these machine learning concepts. To explain the relationships which an algorithm has converged, we utilize statistical methods such as Shapley values which will make our network results more explainable. Obviously if you want to learn about more complex networks such as Convolutional Neural Networks or Liquid Neural Networks, a deeper mathematical basis will be needed. Going back to the field of credit Defaults , employing a neural network to determine what rating should be given to a company based on its financials is a method that is highly appealing as not much will need to be done from our side once the network is deployed. However, we propose a new type of neural networks which utilizes a wavelet transform along with a dimensionality reduction technique to optimize our network's weights in order to help the network find more complex relationships within the data. We call this the Weight Dimensionality Reduction Neural Network(WDR-NN). We utilize this model as traditional Neural Networks perform poorly on the datasets as much of the datasets used are very small. Our innovation Neural Network(WDR-NN) fixes this as we can train a smaller network that is more generalized to learn complex relationships it would otherwise be unable to learn as it is too small. We compare our newly proposed method with other models such as Random Forest Classification,Support Vector Machines. We build each model and compare results with and without our new innovation. We

also propose to see how we can use a wavelet transform and UMAP/PCA/Autoencoders on the data and see how that changes our results with different models.

1.1 Credit Defaults

Credit Defaults are when a company fails to comply with certain financial regulations that were imposed on them. For many companies, their rating can be an indication of whether they will Default. When a company defaults, the bank loses the most money so accurate Credit Defaults are important. For our Moody's Dataset, we set a Threshold of Below "BBB" as the company to default and anything above or equal to not default.

1.2 Literature Review

Credit evaluation analysis is a preventive pre-control method used by commercial banks to evaluate and analyze the credit rating of lending enterprises (projects) in response to loan risks, with the aim of preventing and controlling loan risks, and quantitatively managing loan risk levels.

With the continuous progress of science, machine learning has been applied in more and more fields. In the financial field, ML (machine learning) has been applied to estimate credit rating classification models, and XAI (interpretable artificial intelligence) has been used to study the relationship between corporate financial indicators and credit rating. At the same time, XAI has been used to improve the interpretability of ML.[1]

Among many popular credit evaluation models, random forests, artificial neural networks, and support vector machines have high accuracy.[2] After comparing the accuracy of many common models in credit evaluation, it can be found that convolutional neural networks perform better than some more complex models in credit evaluation.[3]

In recent years, artificial neural networks have been continuously deepening and have made great progress. They have successfully solved many practical problems that are difficult to solve by modern computers in fields such as pattern recognition, intelligent robots, automatic control, predictive estimation, biology, medicine, economics, etc., demonstrating good intelligent characteristics.

Using convolutional neural networks and Transformers to model short-term and long-term dependencies within a time series can successfully predict the stock prices of S&P 500 constituent stocks, with better

performance than most models including the state of the art deep learning based autoregressive model DeepAR. [4]

Wavelet transform is a popular transformation analysis method that can fully highlight the characteristics of certain aspects of the problem, perform localization analysis on time and frequency, and ultimately achieve time subdivision at high frequencies and frequency subdivision at low frequencies.[5] Discrete wavelet transform is also used to analyze the relationship between microbiome and human traits.[6] Wavelet transform can also be used in the financial field. The data obtained from wavelet transform can improve the prediction results in a short period of time, but for long-term predictions, the results are not significant.[7]

As Found by [10], it was found that wavelet activation neural networks outperform the default neural network. We use [10] the Mexican Hat function as a simple way to express the wavelet Transform. Also found [10], more data does not always beat quality data in financial applications so we use this to our advantage as we use an API that has financial ratios. It was also interesting that the Wavelet Activation function underperformed compared to a regular Neural Network[10] . It was also found that ‘‘more data beats clever algorithms but better data beats more data’’[10].We wanted to put this claim to the test to see whether the Moody’s Data which was quality data but not very big in size had a better performance as compared to the England Dataset which had less quality than the Moody’s Dataset but more data.

In the field of finance, we always need a way to explain decisions whether it be for Credit Rating, buying a stock,investing into a company etc. As many of the machine learning methods used today are very complex, their decisions as to why they got to a particular value cannot be explained as shown in[11]. We attempt to use Shapley values to explain our models conclusion as suggested in [11].

As suggested by[12], non-parametric methods often work the best in Credit Defaults as many relationships in Credit Defaults are non-linear and models that find mass amounts of non-linear relationships would be useful. We put this claim to the test as we use 2 parametric methods(Logistic Regression and Linear Support Vector Machine) and 2 non-parametric methods(Neural Networks and Decision Trees) and compare their success with various metrics.

From[13], Tensor Decomposition can be used for model reduction. Although our idea is different from the Tensor Decomposition, we can also apply our UMAP/PCA/AUTOENCODERS to other forms of neural networks such as Transformers,Recurrent Neural Networks,etc.

Autoencoders can be used as a means of dimensionality reduction as suggested by[14].They can learn relationships that other machine learning methods cannot so they are our desired choice of dimensionality reduction for our bias terms.

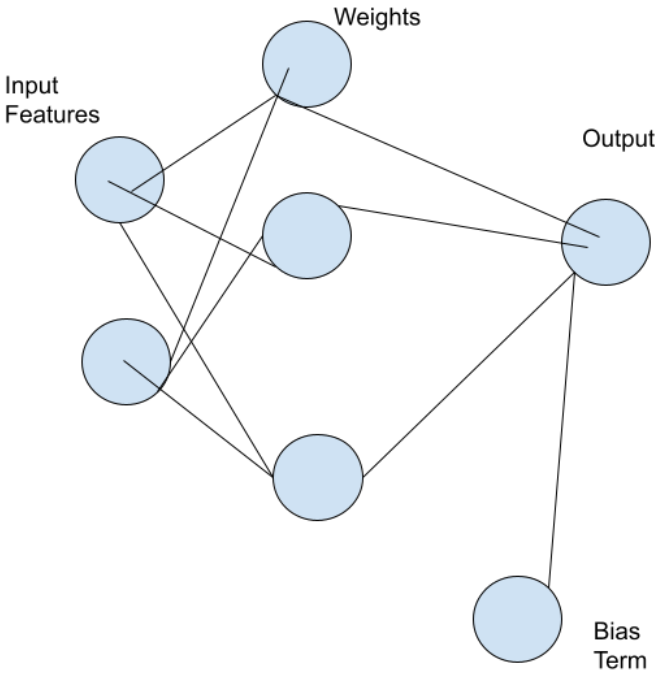
Background to Methods

2.1 Neural Networks

Neural Networks are part of the field of Deep Learning and Artificial Neural Networks form the basis for Deep Learning as most Deep Learning Algorithms are derived from Artificial Neural Networks. They work as the following

1. Take an input part of training and must be numerical values. Can be vector or matrix or even tensor.

We now introduce the term forward propagate. In a neural network we have multiple hidden layers. These layers serve as a way for the model to learn linear patterns in the data. We have so called nodes in each hidden layer which are values.



As seen in the image above, have nodes which contain numerical values. The basic idea is that we have a weight matrix which we multiply the previous layer by to get to the next layer. However, like most problems in life, our data does not have linear patterns and we mostly face nonlinear patterns.Here we apply a so-called activation function. The purpose for this activation function is so that our model will learn non-linear patterns in the data. As the name suggests we take a value,matrix,tensor,etc and apply a transformation to it and return another value.Common activation functions include Tanh,ReLu,Sigmoid,Swish,etc. These are very good choices as they are not very complex but help the model learn very well. As seen later, we employ a wavelet based activation function.

The final process in a neural network is to backpropagate the prediction. We use a loss function such as Mean-Squared-Error for Regression to determine what the loss is for the network with the given parameters. We take the partial derivatives of the weights and biases with respect to the loss function in an attempt to minimize the loss function as the weights and biases all serve as

parameters for the prediction and therefore parameters for the loss function. Once we take the partial derivatives, we update our parameters for the models using the following

New = Old - Learning Rate * (Partial Derivative Old with respect to Loss function)
Where Old is any given parameter of the network and New will be the new parameter which will help the network decrease the loss function and increase performance.We repeat the process for a given amount of times until we have reached a performance that is satisfactory or the network stops learning.

2.2 Wavelet Transform

The wavelet transform is a form of decomposing a signal into multiple frequencies that are relevant. The basic idea is we take our data matrix and multiply it by a so-called "Wavelet transform matrix" which has the properties that it is orthogonal and orthonormal. They have vectors which we call filters which we slide through the matrix. As they are orthogonal, we can use the property that the transpose is the inverse. When we multiply the data matrix by the wavelet matrix, we now have a matrix which is in the wavelet domain.
In mathematical view, an orthogonal M-Band Discrete Wavelet Transform (DWT) uses a set of M filters in a filter bank with certain properties. In any such filter bank, there are one low pass filter α and $M-1$ high pass filters $\beta^{(j)}$ for $j = 1, \ldots, M-1$ with N vanishing moments. These filters satisfy the following conditions:

$$\sum_{i=1}^{n} \alpha_i = \sqrt{M},$$

$$\sum_{i=1}^{n} i^k \beta_i^{(j)} = 0 \; for \; k = 0, 1, \ldots, N-1, j = 1, \ldots M-1,$$

$$\|\alpha\| = \|\beta^{(j)}\| = 1 \; for \; j = 1, \ldots, M-1,$$

$$\langle \alpha, \beta^{(j)} \rangle = 0 \quad for \; j = 1, \ldots, M-1,$$

$$\langle \beta^{(i)}, \beta^{(j)} \rangle = 0 \quad for \; i, j = 1, \ldots, M-1 \text{ and } i \neq j.$$

M-band DWT can be used to decompose a signal $S \in R^{Mk}(k \in N, k \geq M)$ into the sum of M different frequency components. To do so, we create a corresponding $Mk \times Mk$ DWT matrix W_1 by shifting and wrapping around the filters α and $\beta^{(j)}$ for $j = 1, \ldots, M-1$. Generally, M-band L-regular DWT matrix has M filters with each length ML, and the non-zero elements in each row shifts by M columns. To be concise, here is a 16 × 16 4-band 2-regular DWT matrix:

$$\begin{bmatrix}
\alpha_1 & \alpha_2 & \alpha_3 & \alpha_4 & \alpha_5 & \alpha_6 & \alpha_7 & \alpha_8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \alpha_1 & \alpha_2 & \alpha_3 & \alpha_4 & \alpha_5 & \alpha_6 & \alpha_7 & \alpha_8 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \alpha_1 & \alpha_2 & \alpha_3 & \alpha_4 & \alpha_5 & \alpha_6 & \alpha_7 & \alpha_8 \\
\alpha_5 & \alpha_6 & \alpha_7 & \alpha_8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \alpha_1 & \alpha_2 & \alpha_3 & \alpha_4 \\
\beta_1 & \beta_2 & \beta_3 & \beta_4 & \beta_5 & \beta_6 & \beta_7 & \beta_8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \beta_1 & \beta_2 & \beta_3 & \beta_4 & \beta_5 & \beta_6 & \beta_7 & \beta_8 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \beta_1 & \beta_2 & \beta_3 & \beta_4 & \beta_5 & \beta_6 & \beta_7 & \beta_8 \\
\beta_5 & \beta_6 & \beta_7 & \beta_8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \beta_1 & \beta_2 & \beta_3 & \beta_4 \\
\gamma_1 & \gamma_2 & \gamma_3 & \gamma_4 & \gamma_5 & \gamma_6 & \gamma_7 & \gamma_8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \gamma_1 & \gamma_2 & \gamma_3 & \gamma_4 & \gamma_5 & \gamma_6 & \gamma_7 & \gamma_8 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \gamma_1 & \gamma_2 & \gamma_3 & \gamma_4 & \gamma_5 & \gamma_6 & \gamma_7 & \gamma_8 \\
\gamma_5 & \gamma_6 & \gamma_7 & \gamma_8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \gamma_1 & \gamma_2 & \gamma_3 & \gamma_4 \\
\delta_1 & \delta_2 & \delta_3 & \delta_4 & \delta_5 & \delta_6 & \delta_7 & \delta_8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \delta_1 & \delta_2 & \delta_3 & \delta_4 & \delta_5 & \delta_6 & \delta_7 & \delta_8 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \delta_1 & \delta_2 & \delta_3 & \delta_4 & \delta_5 & \delta_6 & \delta_7 & \delta_8 \\
\delta_5 & \delta_6 & \delta_7 & \delta_8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \delta_1 & \delta_2 & \delta_3 & \delta_4
\end{bmatrix},$$

where

α= [-0.06737176, 0.09419511, 0.40580489, 0.56737176, 0.56737176, 0.40580489, 0.09419511, -0.06737176],

β= [-0.09419511, 0.06737176, 0.56737176, 0.40580489, -0.40580489, -0.56737176, -0.06737176, 0.0941951],

γ= [-0.09419511, -0.06737176, 0.56737176, -0.40580489, -0.40580489,-0.56737176, -0.06737176, -0.0941951],

δ= [-0.06737176, -0.09419511, 0.40580489, -0.56737176, 0.56737176, -0.40580489, 0.09419511, 0.06737176]

are corresponding filters.

Given the wavelet matrix W_1, the first level DWT of S can be done by:

$$W_1 S = [a^1\ d_1^1\ d_2^1\ \ldots\ d_{M-1}^1]^T \triangleq \widetilde{S}_1,$$

where $a^1 = \left[a_1^1, a_2^1, \ldots, a_k^1\right]^T$, and $d_i^1 = \left[d_{i,1}^1, d_{i,2,}^1 \ldots, d_{i,k}^1\right]^T$, $i = 1, 2, \ldots, M-1$.

Intuitively, $W_1 S = \widetilde{S}_1$ means that we transform S into the corresponding wavelet domain, or coordinates of S under the wavelet basis.

Denote $C_1,\ C_2, \ldots, C_{Mk}$ as the column vectors of W_1^T, then $\{C_1,\ C_2, \ldots, C_{Mk}\}$ forms an orthonormal basis of R^{Mk}, and we have

$$S = s_1 C_1 + s_2 C_2 + \ldots + s_n C_n,$$

$where\ n = Mk\ and\ s_i = C_i^T S = \langle C_i, S\rangle$, the inner product of $C_i\ and\ S$ for $i = 1, 2, \ldots, n$. Therefore

$$s_i = \{a_i^1\ for\ i = 1, 2, \ldots, k\ d_{1,i}^1\ for\ i = (k+1), k+2, \ldots, 2k\ \vdots\ d_{(M-1),i}^1\ for\ i = (M-1)k+1, \ldots, Mk$$

.

Let:

$$\{A^1 = a_1^1 C_1 + \ldots + a_k^1 C_k,\ D_i^1 = d_{i,1}^1 C_{ik+1} + \ldots + d_{i,(i+1)k}^1 C_{(i+1)k}$$

for $i = 1, \ldots, M-1$. Then A^1 is corresponding to a^1 and D_i^1 is corresponding to $d_i^1\ for\ i = 1, \ldots, M-1$, and $S = A^{(1)} + D_1^{(1)} + \ldots + D_{M-1}^{(1)}$.

Furthermore, such decomposition is unique, since the following M subspaces

$$V_1 = span\{C_1, \ldots, C_k\}, \qquad and\ W_1^{(i)} = span\{C_{ik+1}, \ldots, C_{(i+1)k}\},\ for\ i = 1, \ldots, M-1,$$

are orthogonal to each other, and $R^{Mk} = V_1 \oplus W_1^{(1)} \oplus \ldots \oplus W_1^{(M-1)}$ is the direct sum of those M subspaces.

Moreover, the M-Band DWT of S decomposes S into M different frequency components with $a^1\ being$ the lowest frequency component (or trend) and $d_1^1, \ldots, d_{M-1}^1$ being the higher frequency

components (or fluctuations) of S. If necessary, and k is divisible by M, we can apply DWT to a^1 using a $k \times k$ DWT matrix W_2 such that:

$$W_2 a^1 = \left[a^2\, d_1^2 \ldots\, d_{M-1}^2\right]^T \triangleq \widetilde{S}_2,$$

where $a^2 = \left[a_1^2, a_{2,}^2 a_3^2, \ldots, a_{\frac{k}{M}}^2\right]^T$, $d_i^2 = \left[d_{i,1}^2, d_{i,2}^2, d_{i,3}^2, \ldots, d_{i,\frac{k}{M}}^2\right]^T$, $i = 1, \ldots M - 1$.

The M-Band DWT of a^1 decomposes a^1 into M different frequency components with a^2 being lowest frequency and d_i^2 $(i = 1,\ldots, M - 1)$ being higher frequency components of a^1.

For certain applications in which we use time series data, it may be of importance to us to access the particular date that is inside the transformed matrix so we simply multiply by the inverse to go back to the original time series domain. The low-frequency component contains the most information out of the original matrix. The other parts, which correspond to the high frequency, contain the detailed information. For this study we employ the wavelet transform as a means of data-processing method and a dimensionality reduction for our neural network weights.

2.3 Other Classification methods

2.3.1 Uniform Manifold Approximation and Projection(UMAP)

UMAP is a dimensionality reduction technique which is groundbreaking as it utilizes the abstract field of topology to reduce the dimension of the data. The basic idea is that we represent the data in a lower dimensional space that preserves the original data's structure in its original higher dimensional space. We do this by maintaining relationships between 2 points such that the probability that 2 points randomly select each other as neighbors is maintained in the reduction process.

2.3.2 Principal Component Analysis(PCA)

PCA is another dimensionality reduction technique that maintains the overall structure of the data. We do this by the following

1. Take mean matrix of data

2. Centered Matrix = Data matrix - Mean Matrix
3. Take the covariance matrix of Centered Matrix.
4. Sort the eigenvectors of the covariance matrix by the eigenvalues in a Greatest to Least
5. The eigenvalues represent what value they hold of the entire variance of the total data.
6. Choose which combination of eigenvectors maintain overall variance of data matrix to whatever degree you want.
7. Put into one matrix with eigenvectors highest variance as beginning columns
8. Take the dot product of the eigenvector matrix and the centered and that should be your data in a reduced dimension.

Properties from Linear Algebra such as Singular Value Decomposition form the mathematical basis for PCA.

2.3.3 Autoencoders

Autoencoders are from the field of Natural Language Processing and work similar to neural networks however there is a difference. We have an encoding function that takes the original data and puts it to a lower dimension. We then have a decoding function that has a lower dimensionality and our goal is to train both functions so that that we optimize the encoding so that it can perform an accurate dimensionality reduction and train the decoding so that it can accurately put a lower dimensional vector, matrix or tensor into whatever dimension we want serving as an inverse transformation. When these are trained, we can use them for feature extraction models.

3.Methodology, Experimental Design and data explanation

3.1 Wavelet-based Data Processing

We apply different machine learning methods on the dataset of credit data from some UK companies. There are 62923 rows in the original data, and each row represents one company, while each column represents one variable. The dataset contains 58 indexes with each containing data of 19 or 20 years, while other 8 variables that are not dependent on time are also attached.
First we clean the data by deleting those columns with too few valid entries and filling the missing value with the median of the corresponding column. Then we apply 4-band discrete

wavelet transform (DWT) on the time-dependent variables and then we can obtain one low-frequency component (approximation part) and three high-frequency components (details parts), with each component containing five entries.

In our method, let S be the cleaned data before DWT with

$$S = [S_0, S_1, \ldots, S_k],$$

Where S_0 is the variables not dependent on time, and $S_i, i = 1, 2, \ldots, k$ are matrices for k indexes, with each row representing one company, and each column representing one time. We apply 4-band DWT on each S_i^T to obtain

$$WS_i^T = [\, a_i^T,\ d_{i,1}^T,\ d_{i,2}^T,\ d_{i,3}^T \,]^T,$$

where a_i is the low-frequency component and $d_{i,1}$, $d_{i,2}$, $d_{i,3}$ are three high-frequency components.

We use each component of the transformed data, as well as other time independent variables to construct the new data matrix, that is

$$T_0 = [S_0, a_1, \ldots, a_k], \text{ and } T_j = [S_0, d_{1,j}, \ldots, d_{k,j}] \text{ for j=1,2,3,}$$

each has 298 columns. Also, we can combine all the four components and the time independent variables to construct a large data matrix with 1168 columns, that is,

$$T = [S_0, a_1, \ldots, a_k, d_{1,1}, \ldots, d_{k,1}, d_{1,2}, \ldots, d_{k,2}, d_{1,3}, \ldots, d_{k,3}].$$

To train the credit score model, we randomly divide the data into five parts, with four of them as training set and the rest as test set.

3.2 Weight Dimensionality Reduction Neural Network(WDR-NN)

In this study we propose 3 new types of neural networks.

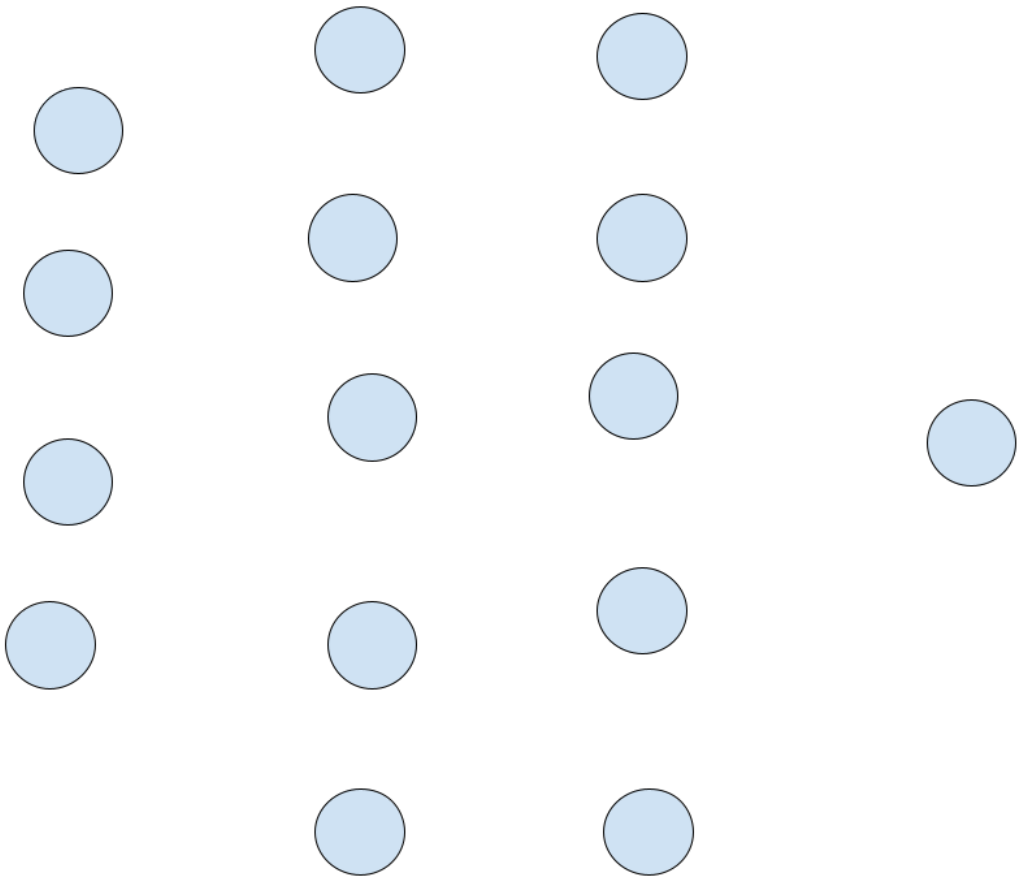
1. UMAP with PCA with autoencoders

In a traditional neural network the weights are stored after training for model prediction. The innovation idea here was to perform a dimensionality reduction on the weights so they could be given to another network which would have most of the information which the other network learned but was smaller. This prevented overfitting but also helped a smaller network arrive at a solution quicker that has relationships that it could have not learned by itself. For our original network we had an input layer then a hidden layer with 256 neurons,another hidden layer with

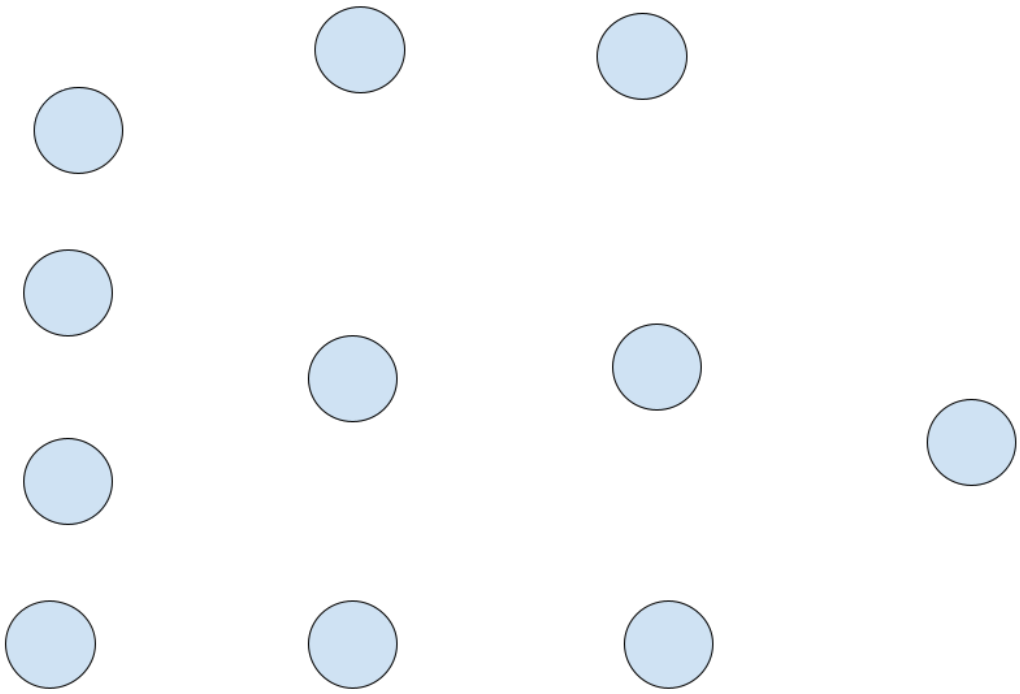
256 neurons,another hidden layer with 256 neurons, and finally an output neuron. We then reduce the weights in hidden layer 2 and 3.The weight matrix is in the shape of a 256 by 256 matrix. We perform UMAP on data so that only 128 components are left and we are left with a 256 by 128 matrix. We transpose and reduce to 128 components. We now have 128 by 128 weight matrices which contain most of the information from the original weights. For weights in the first hidden layer, it depends on the amount of features you have but we will use 57 features as an example. The matrix was size 57 by 256. We perform PCA so that the weight matrix is a 57 by 128. For all of our biases and hidden layer 4 , they were all in a shape of 1 by 256(hidden layer 4 was in 256 by 1 but we took transpose). We then created an autoencoder that would just encode the vectors so they were all 128 by 1. They were all in a reduced dimension which could then be passed to the following network

1. Input layer with varying amount of features.
2. Hidden layer 1 has 128 neurons
3. Hidden layer 2 has 128 neurons
4. Hidden Layer 3 has 128 neurons
5. 1 Output neuron

We then train the network on new data so that it learns more relationships along with the ones it had learned from the previous network.



Original Neural Network No Weight Reduction (5 Neurons on Each Hidden Layer)



This right here is the WDR-NN which has less parameters than the Neural Network above(3 Neurons each hidden Layer)

2. Wavelet Transform

We repeat the same process above in terms of dimensionality reduction. For the 256 by 256 matrix we apply Daubechies 2 Band Wavelet to the weight matrix. We then take the lower frequency part as it contains most of the information.This matrix is size 128 by 128 . We repeat on hidden layer weights 3 and 2. For Hidden layer 1, we apply the wavelet transform ,transpose and take the lower frequency part. For all the biases, we apply the wavelet transform and take the lower frequency part. For the last hidden layer weights we apply the wavelet transform and take the lower frequency part

3. Wavelet Activation Function

For neural networks, we have activation functions that help the network learn non-linear relationships within the data. We can utilize wavelet activation functions which help the network

learn about the different frequencies about the data. We utilize the Morlet Wavelet and The Mexican Hat Function.

3.3 Datasets

We have the following 2 sets of data.

1. A dataset that contains financials from England Companies whose features are not given as the data is about the public.The data is also transformed so there is no way we can find what the features represent. We have a class label.(1167 Features 12000 entries)
2. A dataset with companies and their credit Rating(Fitch,Moodys,etc) along with 25 financial features about the company at the time of the rating.(Time Series Data)(25 Features 3000 entries.

For all datasets we set the class label as Binary Classification.

For the 1st Dataset, the class label is already binary.

For the Credit Rating Dataset, we set anything higher than a BB as a 0 and Lower as 1.

4. Accuracy Metrics

To compare the performance of the model, we use the following measures:

(1) AUC: the area under the ROC curve;

(2) Accuracy: the total accuracy on the whole data set;

For the other four measures, denote

TP = the number of positive samples with a true prediction;

FP = the number of positive samples with a false prediction;

TN = the number of negative samples with a true prediction;

FN = the number of negative samples with a false prediction.

Then we have

(2) Accuracy = (TP + TN) / (TP + FN + FP + TN)

(3) $Gmean = \sqrt{\frac{TP}{TP+FN} \bullet \frac{TN}{TN+FP}}$;

(4) $F - measure = \frac{2TP}{2\,TP+FN+FP}$;

(5) $Type\ 1\ error = \frac{FP}{TN+FP}$;

(6) $Type\ 2\ error = \frac{FN}{TP+FN}$.

4.1 Empirical Results

Table 1: Performance among different data pre-processing, with and without Wavelet transform, on England data

		AUC	accuracy	Gmean	F Measure	Type1 error	Type2 error
LDA	no dwt	0.851	0.846	0.685	0.408	0.101	0.327
LDA	low	**0.85**9	**0.947**	**0.688**	0.585	0.013	0.520
LDA	high1	0.841	0.941	0.652	0.512	0.020	0.560
LDA	high2	0.853	0.944	0.671	0.550	0.016	0.541
LDA	high3	0.820	0.940	0.657	0.536	0.016	0.560
LDA	all	0.686	0.884	0.722	0.483	0.082	0.431
DT	no dwt	0.714	0.905	0.824	0.563	0.079	0.259
DT	low	**0.722**	**0.91**5	0.825	0.565	0.077	0.258
DT	high1	0.718	0.906	0.825	0.575	0.077	0.261
DT	high2	0.715	0.907	0.827	0.575	0.077	0.263
DT	high3	0.721	0.908	0.826	0.578	0.077	0.264
DT	all	0.717	0.910	**0.847**	0.582	0.075	0.264
NN	no dwt	0.742	0.809	0.74	0.601	0.109	0.384
NN	low	**0.772**	0.842	**0.77**	0.653	0.108	0.315
NN	high1	0.768	0.844	0.77	0.640	0.112	0.328
NN	high2	0.765	**0.846**	0.76	0.577	0.119	0.352
NN	high3	0.767	0.845	0.77	0.651	0.114	0.327
NN	all	0.724	0.840	0.76	0.590	0.132	0.320

LG	no dwt	0.628	0.842	0.777	0.385	0.104	0.327
LG	low	0.632	0.891	0.777	0.415	0.098	0.321
LG	high1	0.636	0.886	0.778	0.417	0.098	0.318
LG	high2	0.635	0.892	0.764	0.407	0.099	0.339
LG	high3	0.642	**0.89**4	0.771	0.408	0.093	0.341
LG	all	**0.676**	0.885	0.780	0.413	0.039	0.335

From this table, it can be seen that for the Least Discriminant Analysis model, the use of wavelet transform can indeed improve the accuracy of the data, but at the same time, the Type II error also increases. For Decision Tree and logistic regression, whether or not to do wavelet transform has little impact on the results. For Base Neural Network, the accuracy has been improved, and other indicators have not changed much.

Table 2(England Dataset no Wavelet Transform Preprocessing)

	England Data	England Data	England Data	England Data	England Data	England Data	England Data
	Accuracy	Precision	F-Measure	Type 2 Error	G-Mean	Type 1 Error	AUC
NN + UMAP/PCA	**0.976888**	0.955621	0.977307	0	0.976723	0.046012	0.98
Base Neural Network	0.858243	0.783784	0.873973	0.012384	0.849129	0.269939	0.539
Decision Tree	0.919877	0.888252	0.922619	0.040248	0.919204	0.119632	0.92
Wavelet Activation NN	0.427562	0.392904	0.551661	0.074303	0.335607	0.878327	0.539
Wavelet Dimensionality Reduction NN	0.927581	0.877049	0.931785	0.006192	0.925541	0.138037	0.948
Least Discriminant Analysis	0.909091	0.84555	0.916312	0	0.904996	0.180982	0.91
Support Vector Machine	0.950693	0.909859	0.952802	0	0.949653	0.09816	0.951
Logistic Regression	0.96302	0.930836	0.964179	0	0.962487	0.07362	0.963

Table 3(Moody's Credit Rating Dataset no Wavelet Transform Preprocessing)

	Moody's Data	Moody's Data	Moody's Data	Moody's Data	Moody's Data	Moody's Data	Moody's Data
	Accuracy	Precision	F-Measure	Type 2 Error	G-Mean	Type 1 Error	AUC
NN UMAP/PCA	0.942078	0.97037	0.939068	0.090278	0.940947	0.026756	0.957
Base Neural Network	0.860307	0.915323	0.847015	0.211806	0.856059	0.070234	0.928
Decision Tree	0.741056	0.666667	0.781609	0.055556	0.71754	0.454849	0.647
Wavelet Activation NN	0.708688	0.670554	0.729002	0.201389	0.704837	0.377926	0.739
Wavelet Dimensionality Reduction NN	0.930153	0.924399	0.929188	0.065972	0.930217	0.073579	0.93
Least Discriminant Analysis	0.901193	0.942308	0.894161	0.149306	0.898898	0.050167	0.966
Support Vector Machine	0.679727	0.703252	0.64794	0.399306	0.673822	0.244147	0.735
Logistic Regression	0.674617	0.614118	0.732118	0.09375	0.639669	0.548495	0.679

As seen in our results, our new UMAP neural network outperforms traditional methods consistently. Something to note is that our Network has improved accuracy on smaller datasets(2000 samples) compared to a bigger dataset(720000 samples) where there exists non-class imbalances. As seen in the England Data the, the UMAP Neural Network Yields a 12 percent increase in accuracy as compared to the original Neural Network. The Wavelet Activation Neural Network performs very poorly on the England Data and we attribute this to the lack of data as a transformation as the Wavelet Transform may delay the learning rate of the network yielding a poor final accuracy. The Wavelet Dimensionality Network also yields a 7 percent increase in accuracy as compared to the original Neural Network. Our new innovation point of using UMAP is the clear leader in all metrics for the England Data. Because the data is small, the smaller model can inherit relationships from the original network that would otherwise be almost impossible for the network to learn on its own. The UMAP Neural Network is computationally complex when the previous network is very big. UMAP in particular is very computationally complex but PCA and Autoencoders are not computationally complex. The total training time for an original network with over 1 million parameters for UMAP took approximately 9 minutes.

Although the Moody’s dataset has more samples than the England dataset, there are less features leading to less data in general. This is seen as the accuracy rate in this dataset is lower than the England Dataset. Our UMAP network outperforms all models except in 2 metrics. We see an increase of 20 percent in accuracy in our UMAP Neural Network. The model is very consistent in many metrics as compared to other methods such as Least Discriminant Analysis.

We also compare the results of applying a discrete wavelet transform to the England Data as a preprocessing technique. The results are below for taking only the high frequency component of the data after the transform(4 Band transform was used)

Table 4(High Frequency Component of England Data after Wavelet Transform)

	EnglandData	EnglandData	England Data	EnglandData	EnglandData	EnglandData	EnglandData
	Accuracy	Precision	F-Measure	Type 2 Error	G-Mean	Type 1 Error	AUC
Neural Network UMAP/PCA	0.965699	0.950495	0.967254	0.015385	0.964937	0.054348	0.985
Base Neural Network	0.886544	0.883838	0.890585	0.102564	0.886147	0.125	0.953
Decision Tree	0.939314	0.934343	0.941476	0.051282	0.938983	0.070652	0.939
Wavelet Activation Neural Network	0.686016	0.715909	0.679245	0.353846	0.68598	0.271739	0.769
Wavelet Dimensionality Reduction Neural Network	0.854881	0.911765	0.849315	0.205128	0.854443	0.081522	0.94
Least Discriminant Analysis	0.915567	0.886256	0.921182	0.041026	0.913176	0.130435	0.914
Support Vector Machine	0.920844	0.894737	0.925743	0.041026	0.918866	0.119565	0.92
Logistic Regression	0.912929	0.897059	0.917293	0.061538	0.911786	0.11413	0.912

As seen in the table above, our UMAP Neural Network has a decrease in accuracy but still is the leader of all other methods in all other metrics. It is worthy to note that the Wavelet Dimensionality Reduction Neural Network has a decrease in accuracy and we attribute this to the fact we perform a double wavelet transform that is not of the same kind so this repetitive behavior leads to a decrease in accuracy.

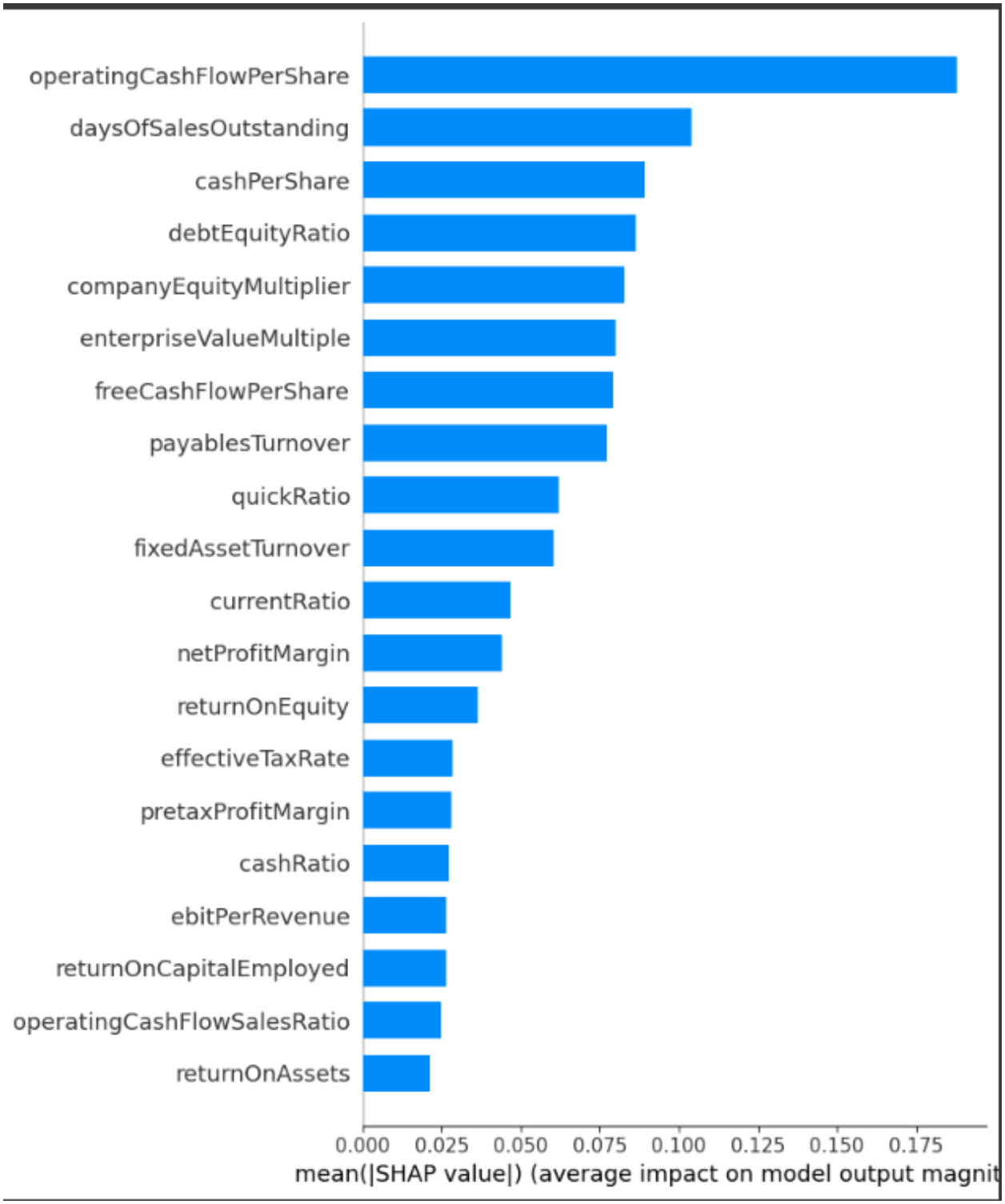
Table 5(England Data Wavelet Transform Preprocessing Technique Lower Frequency)

	England Data	England Data	England Data	England Data	England Data	England Data	England Data

	Accuracy	Precision	F-Measure	Type 2 Error	G-Mean	Type 1 Error	AUC
Neural Network UMAP/PCA	0.960422	0.932692	0.962779	0.005128	0.958736	0.076087	0.977
Base Neural Network	0.683377	0.751678	0.651163	0.425641	0.677394	0.201087	0.796
Decision Tree	0.957333	0.948718	0.958549	0.031414	0.957051	0.054348	0.947
Wavelet Activation Neural Network	0.765172	0.791209	0.763926	0.261538	0.76547	0.206522	0.837
Wavelet Dimensionality Reduction Neural Network	0.899736	0.906736	0.902062	0.102564	0.899802	0.097826	0.951
Least Discriminant Analysis	0.891821	0.840708	0.902613	0.025641	0.885282	0.195652	0.889
Support Vector Machine	0.905013	0.863014	0.913043	0.030769	0.900669	0.163043	0.903
Logistic Regression	0.899736	0.858447	0.908213	0.035897	0.895362	0.168478	0.898

Our results have shown that our Hybrid Neural Network clearly outperforms traditional methods in the realm of Credit Defaults . We accredit this to the fact our method can learn complex relationships that would otherwise be difficult to learn. It is worthy to note that if the original network does not learn much information neither will the UMAP network as the relationships it will learn will not add much value to the final prediction.

4.2 Explainability of Our Empirical Results



These are the Shapley Values of our UMAP model which was the most accurate model out of all of the models. It is clear that Operating Cash Flow Share is the most significant feature when it comes to determining the credit Rating of the company. It is worthy to note that the Shapley values are computationally complex to compute for the specific package we used. It is relevant

to know that features that contain the most relevance to the model output all have to do with how much money a company makes.

5.Discussion and Conclusion

5.1 Conclusion

We model credit Defaults in this paper using various Deep Learning Techniques and Machine Learning Techniques and our new innovation of WDR-NN. The models were trained with 2 datasets

1 .A dataset that contains financials from England Companies whose features are not given as the data is about the public.The data is also transformed so there is no way we can find what the features represent. We have a class label.(1167 Features 12000 entries)

2. A dataset with companies and their credit Rating(Fitch,Moodys,etc) along with 25 financial features about the company at the time of the rating.(Time Series Data)(25 Features 3000 entries.)

We conclude that our newly proposed innovation of WDR-NN outperforms traditional methods in the problem of Credit Defaults. Our Shapley values indicate that any ratio regarding how much money a company has is crucial to whether the company credit rating will default or not.

The 2 most important features of a company regarding whether the company will default are Operating Cash Flow Share and Days of Sales outstanding. We also notice that the Wavelet Dimensionality Reduction works well but not as well as our UMAP/PCA/Autoencoders and we attribute this to the fact the model can save relevant information in the reduction process but also loses more information some of which is just noise which has no real value to the relationships the network learns. Our top two models are our WDR-NN and Decision Tree.

5.2 Future Work

For a future reference it would be interesting to look at the impact of our “Model Reduction” on other types of networks such as Convolutional Neural Networks or Transformers. It would also be worthy to apply this on a Support Vector Machine where we apply a dimensionality reduction on the weights. We would also apply these methods to other financial problems such as the stock market or futures market. We would attempt to solve the class imbalance with Generative

Adversarial Networks and then do the Weight Reduction on different Neural Networks such as Bidirectional Networks.

References

[1] Hashimoto, R., Miura, K., Yoshizaki, Y. (2023). Application of Machine Learning to a Credit Rating Classification Model: Techniques for Improving the Explainability of Machine Learning. Bank of Japan Working Paper Series.

[2]Wallis, M. , Kumar, K. , & Gepp, A. . (2019). Credit Rating Forecasting Using Machine Learning Techniques.

[3]Tavakoli, M., Chandra, R., Tian, F., Bravo, C. (2023). Multi-Modal Deep Learning for Credit Rating Prediction Using Text and Numerical Data Streams. arxiv: 2304.10740.

[4]Zeng, Z., Kaur, R., Siddagangappa, S., Rahimi, S., Balch, T., Veloso, M. (2023) Financial Time Series Forecasting using CNN and Transformer. arxiv: 2304.04912.

[5]Daubechies, I. (1993). Ten lectures on wavelets. Journal of the Acoustical Society of America.

[6]Shankar, A., Chang, S., Zhao, Y., Wang, X., Liu, T. (2022). Wavelet-based Microbiome Correlations of Host Traits. Proceedings of the 2022 6th International Conference on Computational Biology and Bioinformatics.

[7]Rahimyar, A. H. , Nguyen, H. , & Wang, X. (2017). Stock Forecasting Using M-band Wavelet Based Machine Learning Methods. International Journal of Advances in Electronics and Computer Science.

[8] Saia, R. , Carta, S. , & Fenu, G. . (2018). A Wavelet-based Data Analysis to Credit Scoring. 2nd International Conference on Digital Signal Processing (ICDSP 2018).

[9] Noriszura, I. , & Jaber, J. J. . (2020). Assessment of credit losses based on arima-wavelet method. Journal of Theoretical and Applied Information Technology, 98(9), 1379-1392.

[10] A, M. V. , Peter Gordon Rtzel LL.M b a, & A, S. H. . (2022). Forecasting performance of wavelet neural networks and other neural network topologies: a comparative study based on financial market data sets. Machine Learning with Applications.

[11] Bussmann, N. , Giudici, P. , Marinelli, D. , & Papenbrock, J. . Explainable machine learning in credit risk management. Social Science Electronic Publishing.

[12]Wallis, Mark. Credit Rating Forecasting Using Machine Learning Techniques.

[13]Liu, Xingyi, and Keshab K Parhi. “Tensor Decomposition for Model Reduction in Neural Networks: A Review.” IEEE Circuits and Systems Magazine, 26 Apr. 2023.

[14] Wang, Yasi, et al. “Auto-encoder based dimensionality reduction.” *Neurocomputing*, vol. 184, 2016, https://www.sciencedirect.com/science/article/abs/pii/S0925231215017671.

Declaration of Academic Integrity

The participating team declares that the paper submitted is comprised of original research and results obtained under the guidance of the instructor. To the team's best knowledge, the paper does not contain research results, published or not, from a person who is not a team member, except for the content listed in the references and the acknowledgment. If there is any misinformation, we are willing to take all the related responsibilities.

Names of team members Alejandro Mayorga Wang Letian

Signatures of team members Alejandro Mayorga Wang Letian

Name of the instructor Xiaodi Wang

Signature of the instructor

Date 8/20/2023

Commitments on Academic Honesty and Integrity

We hereby declare that we

1. are fully committed to the principle of honesty, integrity and fair play throughout the competition.
2. actually perform the research work ourselves and thus truly understand the content of the work.
3. observe the common standard of academic integrity adopted by most journals and degree theses.
4. have declared all the assistance and contribution we have received from any personnel, agency, institution, etc. for the research work.
5. undertake to avoid getting in touch with assessment panel members in a way that may lead to direct or indirect conflict of interest.
6. undertake to avoid any interaction with assessment panel members that would undermine the neutrality of the panel member and fairness of the assessment process.
7. observe the safety regulations of the laboratory(ies) where we conduct the experiment(s), if applicable.
8. observe all rules and regulations of the competition.
9. agree that the decision of YHSA is final in all matters related to the competition.

We understand and agree that failure to honour the above commitments may lead to disqualification from the competition and/or removal of reward, if applicable; that any unethical deeds, if found, will be disclosed to the school principal of team member(s) and relevant parties if deemed necessary; and that the decision of YHSA is final and no appeal will be accepted.

(Signatures of full team below)

X Alejandro Mayorga
Name of team member:

X Wang Letian
Name of team member:

X
Name of team member:

X
Name of supervising teacher: Xiaodi Wang