UCL Université catholique de Louvain

Louvain School of Management (LSM)

LOUVAIN School of Management

Predicting Recovery Rates of Defaulted Bonds Using a Neural Network

Research Master's Thesis submitted by
Aymeric Charles de la Brousse

Supervisor
Frédéric Vrins

Co-supervisor
Paolo Gambetti

Academic Year 2017-2018
Master in Business Engineering

Acknowledgements

I would like to express my utmost and sincere gratitude to my thesis supervisor, Prof. Frédéric Vrins and my thesis co-supervisor Paolo Gambetti for their continuous support, enthusiasm, patience and knowledge during all the time of research and writing of this thesis.

Contents

1 Introduction **1**

2 Motivations **3**

3 Literature Review **6**

4 Data **9**
4.1 Predictors 10
4.1.1 Bond-specific variable 10
4.1.2 Macroeconomic variable 12
4.1.3 Company-specific variables 13

5 Models **14**
5.1 Linear Regression 14
5.2 Random Forest 16
5.3 Neural Networks 17
5.3.1 The perceptron 17
5.3.2 Multilayer perceptron with backpropagation 19
5.3.3 Activation function 22
5.3.4 Hidden layer 23
5.3.5 Data transformation 23

6 Results **24**
6.1 In-sample analysis 24
6.1.1 Linear Regression 24
6.1.2 Random Forest 26
6.1.3 Neural Network 28
6.2 Out-sample analysis 29
6.2.1 Cross-validation 29
6.2.2 Performance measures 31
6.2.3 VaR simulation 33

7 Conclusion **34**

References **35**

1 Introduction

In 1974, Bank Herstatt, a German institution, went bankrupt before the settlement of a foreign exchange contract which provoked serious disturbances in the international currency markets.[1] Following the aftermath of this crisis, a group of ten nations decided to form the Basel Committee on Banking Supervision (BCBS). The committee's primary goals were to strengthen financial stability by improving global supervision on the banking sector. The committee's principles were summed in two rules: no foreign banking establishment should escape supervision and the supervision should be adequate (Basel Committee on Banking Supervision, 1983).

In 1988, concerned about the growth in international risk, they released the Basel Capital Accord, a series of policies to ensure minimum capital ratios and to homogenize the regulations among countries, a framework also known as Basel I. Enforced in 1992, Basel I imposed on banks to hold a minimum capital. This minimum capital required by regulators was calculated as a proportion of the risky assets owned by the bank. Each asset fell into a risk category and all assets of the same category were assigned a weight. The riskier the asset, the higher the weight. The minimum capital requirement was a proportion of all these accumulated weights. These measures were taken by the BCBS to minimize three types of risk: market risk, operational risk and credit risk. In this study we focus on credit risk, which the BCBS defines as "the potential that a bank borrower or counter-party will fail to meet its obligations in accordance with agreed terms". The main criticism over Basel I concerned the arbitrary levels of risk assigned to each type of asset. For instance, all commercial loans were given the same risk weight, regardless of the financial health of the counterparty.

Following critics over Basel I and in an effort to improve the regulations and eliminate the loopholes, the BCBS introduced a framework for a new agreement in 1999, which they released in 2004. The new agreement, known as "Basel II" introduced the use of internal models to estimate credit risk. The internal models incorporate three risk factors:

1. Probability of Default (PD): the probability that the bond issuer will default on the repayment of its debt e.g. if the issuer falls into financial trouble or goes bankrupt.

2. Loss Given Default (LGD): the proportion of exposure that the bank might lose if the bond defaults. Loss given default is also referred to as the recovery rate, which equals one minus the LGD.

[1] German regulators forced the bank into liquidation because it had accumulated disproportionate losses compared to its capital. That day, the bank had received a payment in Deutsche Mark to deliver US dollar in New York. The regulators closed the bank before settlement of the contract. The bank was closed at the end of the working day in Germany which was the beginning of a working day in New York. That difference in time zone provoked a chain reaction of unsettled operations, resulting in a loss of $620 million to the banking industry. That cross-currency settlement risk is now called the "Herstatt risk".

3. Exposure at Default (EAD): the outstanding debt pending payment at the time of default.

With the internal model approach, banks would be able to generate better estimations of their risk by taking into account their own counter-parties and exposures.

The regulators offered two options for banks who wished to use internal models:

1. Foundation internal ratings-based (FIRB) approach: Only the probability of default and the exposure at default are determined internally, while the LGD is fixed.

2. Advanced internal ratings-based (AIRB) approach: banks can determine their PD, LGD and EAD internally.

According to Lopez and Saidenberg (2001), financial institutions did not wait for Basel II to start using statistical models as a measurement of risk. In fact, a decade sooner, several factors including the increase of trading activities and the progress of computer technology had already persuaded banks to adopt econometric models. What changed upon the arrival of Basel II is that banks were now encouraged by the regulators to develop their own models. However, the BCBS did not give any particular guidelines on how to model the LGD. This may explain the large heterogeneity among the models of banks who adopted the advanced approach. (Hurlin, Leymarie, & Patin, 2018).

It appears that academics and institutions mainly focused on predicting an appropriate probability of default, neglecting accurate LGD estimates. The LGD has been treated as constant even though their yearly average could vary from 8% to 74% depending on the bond type (Moody's Analytics, 2005). Schuermann (2004) also observed that recovery rates tended to be either very high or low. Given these circumstances, using a constant would be truly unreasonable. There are a few reasons why predictive models for LGD have not attracted the interest they deserve. First of all, unlike PD, the recovery rate calculation with attention to the recovery cash flows is a very complex and time-demanding process in itself. (Baesens & Gestel, 2009). Also, data on LGD is scarce as banks have only started to collect relevant information after the introduction of the AIRB approach (Ruiz, 2015).

This study focuses on the estimation of recovery rates applicable in the framework of the AIRB approach. We evaluate the use of an artificial neural network to predict recovery rates of defaulted corporate bonds. Neural networks are supervised learning algorithms based on the architecture of the brain's neural system. The model learns by adapting to its environment and stores its knowledge in inter-neurons connections. Neural networks have been increasingly used in finance practices were they have demonstrated significant achievements in forecasting T-bills, asset management, portfolio selection and fraud detection (Fadlalla & Lin, 2001).

The remainder of this paper is organized as follows. Section 2 describes the motivations behind the use of an advanced approach to forecast recovery rates

rather than using standardized measure. Section 3 reviews the current literature on recovery rates and the use of neural networks for credit risk. Section 4 discusses the determinants of the recovery rates. Section 5 introduces the neural network and two benchmark models. In Section 6 we examine the results of the in-sample and out-sample analysis. Finally, section 7 provides some concluding observations.

2 Motivations

The main motivations of this paper are to highlight the financial risk associated with the use of a fixed recovery rate for defaulted bonds and to propose an efficient way to estimate them. In this section we demonstrate how a constant recovery rate could underestimate risk measurement. To to this, we introduce the concept of expected loss (EL), unexpected loss (UE) and Value-at-Risk (VaR).

First of all, Lütkebohmert (2009) defines the credit loss of a portfolio of N credits as

$$L = \sum_{n=1}^{N} EAD_n * LGD_n * D_n \tag{1}$$

where D_n is a binary variable and takes a value of 1 if there is a default, and 0 otherwise.

While it is impossible to predict future credit losses, banks can estimate an average loss using historical figures. The average loss is referred to as the expected loss (EL).

The expected loss for a portfolio of N credits is then defined by

$$EL = E[L] = \sum_{n=1}^{N} EAD_n * E[LGD_n] * PD_n \tag{2}$$

where PD_n is the expected value of default $E[D_n]$.

On the other hand, the peaks of credit losses above average that the banks may incur are called the unexpected losses (UL). Expected losses and unexpected losses are shown in Figure 1 and together they constitute the credit risk of a portfolio.

To protect themselves against these peaks of unexpected losses, banks should hold a right amount of capital. Indeed, if the capital is insufficient, there is a risk for the bank to become insolvent in periods of financial distress. On the other hand, higher capital requirements could decrease investments and lending, thus reducing profitability. The adequate amount of capital involves a strategic trade-off between financial security and profitability.

Nonetheless, minimum capital should cover the difference between the expected losses and the maximum loss it might suffer over a certain time horizon and for a certain confidence level. The maximum loss is also referred to as the Value-at-Risk (VaR) and computed as

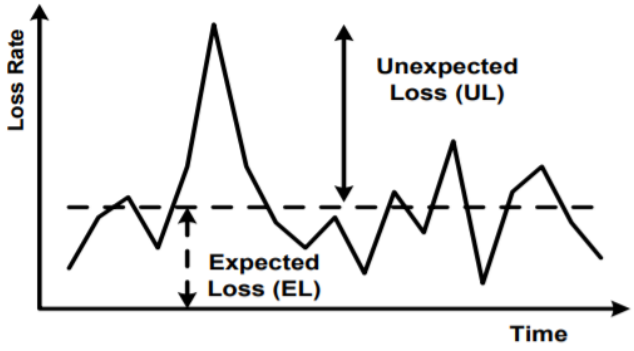

Figure 1
Expected and unexpected losses monitored over time (Basel Committee on Banking Supervision, 2005).

$$VaR_q(L) = inf\{x \epsilon R : P(L > x) \leq 1 - q\} = inf\{x \epsilon R : F_L(x) \geq q\} \quad (3)$$

where q is the confidence level and $F_L(x) = P(L \geq x)$ the distribution function of the loss variable. VaR is essentially a quantile of the loss distribution.

Finally, the capital requirement for a confidence level q is defined by

$$UL_q = VaR_q(L) - EL \quad (4)$$

The idea behind this formula is that the revenues made by the bank's activities over the year should cover the expected losses, while the regulatory capital protects the bank against extremely unfavorable events.

As shown in Figure 2, it is more frequent to experience a level of loss at or below the expected losses than to experience unexpected losses. The grey area represents the probability that the banks won't be able to meet their own obligations with the sum of their profit and stored capital. If the capital required inside the bank is calculated according to the gap between the VaR and the EL, then the chances that a bank will remain solvent through the year is equal to the confidence interval. The higher the confidence interval, the higher the VaR. The minimum confidence level under the Basel II framework is 99.9% (Basel Committee on Banking Supervision, 2005).

Banks who use a fixed recovery rate might underestimate their VaR, and thus their credit risk. To illustrate how the yearly Value-at-Risk differs between using a fixed RR or a stochastic rate, we simulate the maximum losses over a year of 20,000 portfolio, each composed of 500 bonds. To determine which bonds defaults we use a Bernoulli distribution with yearly PD equals 3%. In the first scenario we apply a fixed recovery rate of 0.4 to all the bonds of a portfolio. In the second scenario we assign a different recovery rate to each

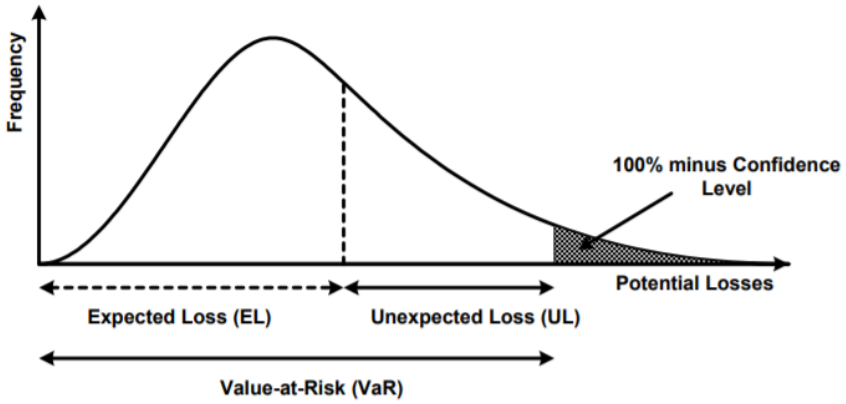

Figure 2
Likelihood of losses at a certain level of loss (Basel Committee on Banking Supervision, 2005).

bond of a portfolios using a beta distribution with a mean of 0.4, alpha of 2 beta of 3. The exposure at default is fixed for all bonds and equals 100. In each simulation, we then pick the loss situated at four different quantiles and the E(L) and compare the outcome. The results are shown in Table 1.

Recovery Rate	$\text{VaR}_{0.99}$	$\text{VaR}_{0.995}$	$\text{VaR}_{0.999}$	VaR_{1}	E(L)
Fixed	1375	1430	1540	1760	829.12
Stochastic	1513.34	1589.47	1753.28	1978.02	905.24

Table 1
Value-at-Risk at quantile $\alpha = 0.99, 0.995, 0.999, 1$ and E(L) of 20,000 portfolios composed of 500 bonds with a PD=3%.

The $VaR_{0.999}$ for a fixed RR equals 1540 while the $VaR_{0.999}$ for a stochastic RR equals 1753,28. The loss distributions of the fixed and stochastic recovery rates are shown in Figure 3.

As seen in Table 1 and Figure 3, using a fixed recovery rate would miscalculate the true maximum loss a bank could suffer in a year. Specifically, the simulation indicates that using a static rate would underestimate the credit risk by up to 12% for the 99.9% confidence interval. Hence, the main motivation behind this research paper is to develop a model capable of accurately estimating recovery rates to encourage the use of internal models and improve financial stability.

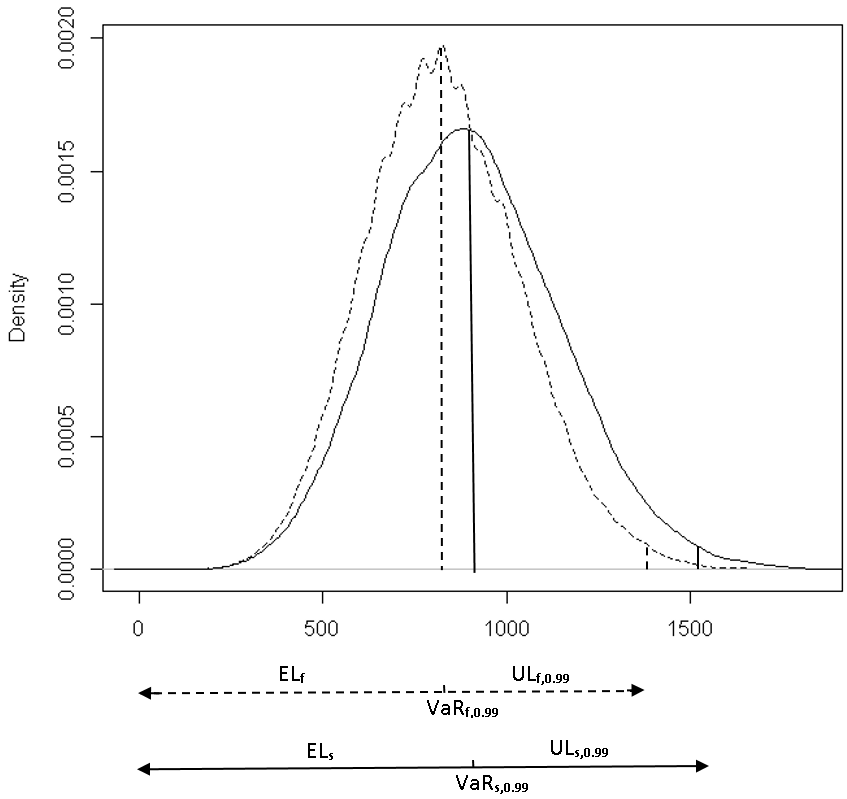

Figure 3
Loss distribution for the fixed and stochastic scenario and their EL, UL and VaR at the 99% confidence interval VaR_f = fixed and VaR_s = stochastic. The dashed curve represents the losses estimated with a static RR fixed at 0.4 and the continuous curve represents the losses estimated with a stochastic RR following a beta regression with $\alpha = 2$ and $\beta = 3$.

3 Literature Review

Before building the forecasting models, we examine the state of the art on recovery rates.

In the past, the literature on credit risk has been dominated by the probability of default, rather than the loss given default. In an attempt to better understand recovery rates, researchers first started to measure the impact of different variables on them. Frye et al. (2000) discovered a strong negative correlation between default rates and recovery rates using Moody's Default Risk Service database for the 1982-1997 period. These findings discredited the previous belief that RR and PD were independent. Indeed, at the time banks were using the same fixed LGD, during periods of economic growth and downturn. Jarrow (2001) arrived at the same conclusions by using both debt and

equity prices, confirming that recovery rates were depending on the state of the economy. He determined that macroeconomic variables did have an effect on recovery rates and that they were fluctuating between economic cycles. Acharya, Bharath, and Srinivasan (2003) reaffirmed the crucial importance of macroeconomic variables. They also highlighted the role of industry-specific conditions in the recovery rates.

Altman, Brady, Resti, and Sironi (2005) discovered a factor that was influencing recovery rates more than macroeconomic variables: supply and demand of defaulted bonds. Their underlying assumption was that during high default years the supply of defaulted bonds was likely to be greater than the demand, which decreased the prices of these bonds on the secondary market and thus affecting the measure of the recovery rates.

Gupton (2005) advanced several factors that prevented institutions to correctly predict the RR. The former lack of observations, the complexity of the recovery process and the lack of relevant predictive factors were all obstacles to recovery predictions. However, Gupton (2005) considered the RR as "fundamental" to predict the potential credit loss. According to him, the RR carried the same importance in risk modelling as PD and a small error could severely mislead the risk estimation. These recent studies coupled with the introduction of the advanced-IRB approach in Basel II fueled the recent appeal for recovery rates predictions. Researchers attempted to predict RR using parametric and non-parametric models. Parametric models facilitate the understanding of the data sample by assuming the sample's distribution to follow a known distribution. Among the scientists who investigated parametric models, X. Huang and Oosterlee (2011) opted for a generalized beta regression. Beta regressions are great tools for predicting variables, like recovery rates, that are bounded in the standard unit interval [0,1]. Sigrist and Stahel (2012) and Yashkir and Yashkir (2013) preferred to use a censored gamma model. Censored variables are often used when a considerable amount of observation are either close to the maximum or close to the minimum. Because recovery rates are either very low or very high, they are said to follow a bimodal distribution. Duan and Hwang (2016) tried to avoid bimodality by including a conditional distribution. Indeed, bimodality make it hard for traditional distributions to fit the model. He then used a beta regression. Calabrese (2014) experimented a mixture model of two Bernouilli random variables and a beta random variable to study the recovery process of Italian banks. Jankowitsch, Nagler, and Subrahmanyam (2014) analyzed the results of a linear regression on the recovery rates of American defaulted bonds from 2002 to 2010. The predictive features chosen were bond characteristics (maturity, coupon, etc.), firm characteristics (equity, intangibility, etc.), macroeconomic variables, liquidity proxies, as well as dummy variables for industry, type of default and seniority of the bond. The authors concluded that the the most important variables were industry and debt seniority of the bond.

Research on parametric models is far more elaborated than on non-parametric models. Nonetheless, the latter are more flexible because unlike parametric

models, they avoid making distribution assumptions. Non-parametric model's flexibility shows great performance in predicting dependent variables but the gain in effectiveness is lost to interpretation as these models are for the most part extremely complicated.

Bastos (2010) compared a non-parametric model, the regression tree, to a parametric model, the fractional response regression to forecast bank loan credit losses. He concluded that the former had better performance at predicting recovery rates shortly after default while the latter worked better on a longer time horizon. A. Bellotti and Crook (2012) also compared a regression tree to another parametric model, the linear regression. He attempted to estimate recovery rates of major UK retail credit cards and surprisingly discovered that the simple linear regression gave the most accurate results.

Yao, Crook, and Andreeva (2015) proposed another non-parametric model, a support vector model, to predict the recovery rates of 6000 observations of defaulted debt securities. Support vector machines are models that can either be used as classifiers or for regression problems by constructing hyperplanes in a high-dimensional space. They can adapt to non-linearity by constructing a hyperplane in a higher dimensional space, which results in a non-linear function separating the data in the lower dimensional space. This algorithm is also called the "kernel trick". To predict the recovery rates Yao et al. (2015) employed a support vector regression. The authors used fourteen explanatory variables allocated in three different categories: "Recovery characteristics", "Accounting variables" and "Macroeconomic variables". They compared the result of their support vector regression to a linear regression and a fractional response regression and noticed that the support vector regression had better results. Support vector machines use supervised learning which Russell and Norvig (2010) defined as an algorithm where "the learning element is given the correct value of the function for particular inputs, and changes its representation of the function to try to match the information provided by the feedback". Another supervised learning method that has become increasingly popular in finance applications is called the "neural network".

In the last years, a good amount of literature has been developed using neural networks to improve the evaluation of credit risk. Boyacioglu, Karab, and Baykanc (2009) tried to classify a database of 65 Turkish banks into two categories: "failed" and "not-failed". A total of 20 financial ratios were used to predict the health status of each bank and the neural network correctly classified 95.5% of the banks in the test set. The network used 5 hidden layers and applied a sigmoid function. Sigmoid functions are useful for recovery rates as they limit the output to a [0,1] range.

Bastos (2010) also used a neural network with a sigmoid function to evaluated the recovery rates of 374 loans granted to small and medium size companies by a Portugal bank. He mostly used bond specific variables like the rating of the bond and the collateral associated with. In addition he added the industry sector and the age of the firm. He demonstrated that a neural network could perform better than a fractional regression. However, the neural network only

beat the fractional regression if the number of observations was sufficiently large.

Pacelli and Azzollini (2011) attempted to classify a database of Italian manufacturing companies according to their credit risk: "safe", "vulnerable" or "risky". To achieve this they used firm-specific variables like short term debt, equity to total asset, cash flow to total asset, the ROA, ROI, ROE, the degree of depreciation and liquidity ratios.Unfortunately, the results were not as expected and the model showed difficulties in predicting the credit risk of the companies.The authors recognized some limits that are proper to NN like local minimum, lack of transparency of the model and the need for a high number of observations for the training set. However, they recognized that neural networks could be an interesting alternative to other methods in certain cases. Their ability to reveal hidden relationships between the features could be convenient to analyze and interpret very complex and obscure processes like financial markets.

This study differentiates itself from the literature by combining elements from previous studies in a unique way. We attempt to predict individual recovery rates of defaulted bonds by using a neural network and a combination of bond-specific, firm-specific and macroeconomic variables. To assess the performance of the neural network, we compare the results to a linear regression and a random forest. We then create a simulation of 20,000 portfolios and evaluate on each model against their computed VaR.

4 Data

In this section we discuss the database of defaulted bonds used in our study. We also define the explanatory variables that we incorporate in the models.

This study uses defaulted bonds from the "Moody's Default & Recovery Database" The sample consists of 415 American corporate bonds that defaulted between 1990 and 2016.

In this database, the recovery rate refers to the average market price of the defaulted security 30 days after default. The market price of expressed as a proportion of the par value of the bond, hence it is scaled from 0 to 1. In the literature, researchers have observed a bimodal distribution for recovery rates. Most of the values were either close to 0 or close to 1. Bruche and González-Aguado (2010) explained that we can either find ourselves in an upward shift of the economy with low probability of default and high recovery rates or in a downward shift of the economy with high PD and low recovery rates. Schuermann (2004) also observed the same results. He added that, while there are actually two modes, lower recovery rates were more common.

The distribution of the recovery rates in our database follows a multimodal beta distribution. Lower recovery rates are more frequent and the density decreases as the recovery rates increase.

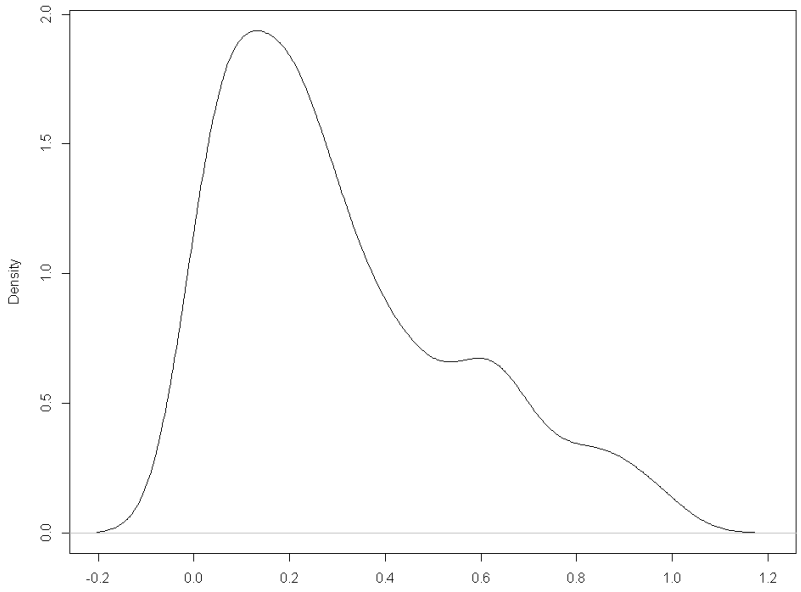

Figure 4
Density plot of the recovery rates taken from our dataset. Recovery rates are bounded by [0,1].In R, by default, the density plot will extend the range so that the density curve approaches 0 at the extreme.

4.1 Predictors

In order to predict the recovery rates, we use three types of explanatory variables: bond-specific, macroeconomic and firm-specific.

4.1.1 Bond-specific variable

Bond-specific variables

1. Debt Seniority

   Six debt seniorities are represented in our database:

   (a) Junior Subordinated
   (b) Subordinated
   (c) Senior Subordinated
   (d) Senior Unsecured
   (e) Senior Secured: Second Lien
   (f) Senior Secured: First Lien

   Seniority is an crucial concept in credit risk as it dictates who has priority on the repayment of the debt in case of business failure . For instance,

debt will always have priority on equity. It means that if the company goes bankrupt, all creditors need to be repaid before any equity holder retrieves his investment. However, not all bonds are created equal and seniority categories between debt holders also exist. A box-plot of the different seniorities found in the database is shown in Figure 5.

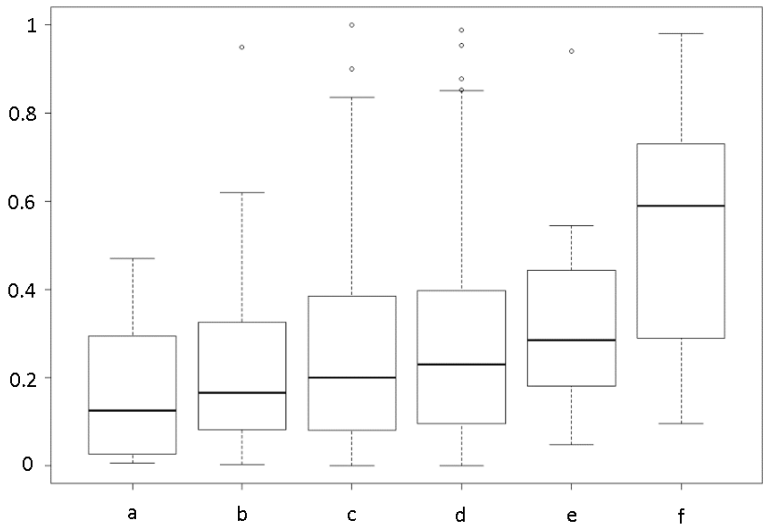

Figure 5
Boxplot of the debt seniority. The boxplot shows the minimum, the first and third quantile, the maximum and the median of each seniority.

First lien debtors clearly have better chances to recover their investment compared to other types of debt holders. The lien is the right from the debt holder to seize the defaulted company's collaterals and to sell them. Collaterals are properties or other assets that the borrower offers as a way for the lender to protect the loan. Nonetheless, even the first lien creditor is not guaranteed to get enough from the sale of the collateral to recover its payment. Therefore, it is essential to pay attention to collaterals before investing in company loans. If the first lien recovers his investment, only then can the second lien reclaim its part and so forth.

2. Sector

   Ten sectors are represented in the database:

   (a) Banking
   (b) Capital Industries
   (c) Consumer industries
   (d) Energy and environment

(e) Finance, insurance and real estate (FIRE)

(f) Media and publishing

(g) Retail and distribution

(h) Technology

(i) Transportation

(j) Utilities

According to the box-plot in Figure 6, it appears that the utilities sector dominates the other industries whereas banking and retail & distribution perform poorly.

Varma and Cantor (2004) studied the high recovery performances of utilities companies. They demonstrated that a high proportion of fixed assets was the reason for these high levels of recovery. Indeed, fixed assets can produce massive amounts of cash flow which is valuable for the debt holders. They also pointed out the steadiness of the cash flow, protected by regulating authorities. Acharya, Bharath, and Srinivasan (2007) added that tangibility could also be a determinant of recovery rates. According their research, because intangible assets are harder to transfer from one firm to the other, acquirers could get little to no value from these assets.

Acharya et al. (2007) also argued that recovery rates were higher for industries in distress. They believed that during downturns these sectors had fewer companies and less liquidity, which would make the redeployment of the defaulted company's assets more challenging. Because it is harder to find an acquirer for the assets, the price of the assets drops and the amounts recovered decrease. This is commonly referred to as the "fire sale effect (Oh, 2018).

3. Coupon:

   The coupon is the annual interest paid to the owner of a bond and is expressed as a percentage of the face value of the bond. The data is composed of 64 zero-coupon bonds and 351 coupon bonds.

4.1.2 Macroeconomic variable

The monthly American default rate is our macroeconomic variable. There is strong evidence in the literature that recovery rates and default rate are dependent. In periods of recession recovery rates are lower than in periods of economic expansion. Altman, Resti, and Sironi (2003) observed about the recovery rates that "they tend to go down just when the number of defaults goes up in economic downturns". Frye (2002) showed that, in times of recession, recoveries were three times lower than during times of growth.

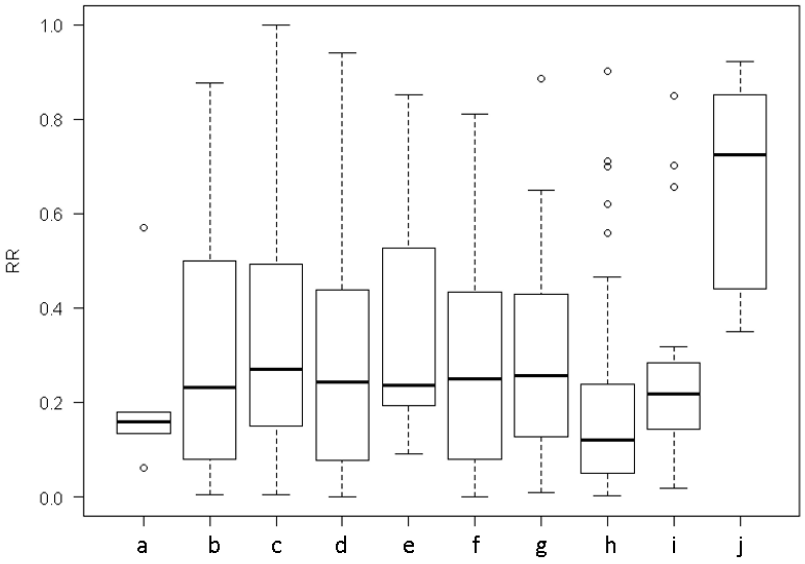

Figure 6
Boxplot of the sector. The boxplot shows the minimum, the first and third quantile, the maximum and the median of each sector.

4.1.3 Company-specific variables

Company-specific variables are accounting measures retrieved from a Bloomberg Terminal, measured on the day of the default.

1. Firm size indicator:

   According to Acharya et al. (2007), we can use the logarithmic transformation of the total assets to evaluate the firm's size. Other studies have also used log(total asset) as a proxy for firm size, including Wald (1999), MacKay and Phillips (2002) and Dalbor, Kim, and Upneja (2004).

2. Tangibility indicator:

   As seen in the utilities sector, the tangibility of the firm could impact the recovery rates. We add a fixed asset to total asset ratio to the list of selected features.

3. Leverage indicator:

   Total debt to total asset has often been used in the literature to predict recovery rates and rightly so. The more assets a company has, the more money it can get from the liquidation to repay its debts. Conversely, the more debts you have, the more competition there is to recover money in case of default because the debtor must share the pie between more debt

holders. It also requires greater coordination between the parties because of a higher dispersed ownership (Bris & Welch, 2005). Lower debt should benefit to all debt holders regardless of seniority. We also add three other leverage ratios to fully comprehend the capital structure of the firms: long term debt to total capital and long term debt to total asset.

A coverage ratio is used by Frye (2002) and Schläfer (2011). It is a measure of the company's capability to meet its debts. The coverage ratio we choose for this study is cash flow to total liabilities.

4. Profitability indicator:

   Varma and Cantor (2004) also argued that a firm's profitability would be a good indicator of debt recovery. Assets of companies that defaulted in a high-growth sector, and whose revenue growth was significantly higher from quarter to quarter, were deemed more profitable and valued higher by debtors. The debtors saw the potential in these assets which explains their higher valuation. Therefore, we incorporate sequential revenue growth to the model.

   Das, Hanouna, and Sarin (2008) studied new ways to evaluate default risk by predicting the spread of credit default swaps (CDS). Credit default swaps are compensations that a CDS seller requires for bearing the risk of default. They observed better results by using accounting based metrics than market based metrics. One of the accounting variables they used to gauge profitability of the firm was the Return on Asset (ROA), measured by the net income divided by the total assets. Therefore, we incorporate the Return on Asset to the model.

   We also add profit margins, which are commonly used in recovery estimates, as seen in Covitz and Han (2005) and Acharya et al. (2003).

The complete list of features can be seen in Table 2.

5 Models

In this section we describe the models we use to predict the recovery rates. We especially do an in-depth explanation of neural networks. In order to evaluate the performance of the network, we build a parametric and a non-parametric model:

1. Linear Regression
2. Random Forest

5.1 Linear Regression

According to Zhang and Thomas (2012), the linear regression is one of the most simple and widely used models in financial areas. It is also the most accessible predictive model for recovery rates modelling.

Macroeconomic	Bond-specific	Firm-specific
American Default Rate	Debt Seniority Sector Coupon	*Firm Size* Logarithm of Total Asset *Tangibility* Fixed Asset to Total Asset *Leverage Ratios* Total Debt to Total Assets Cash Flow to Total Liabilities Long Term Debt to Total Capital Long Term Debt to Total Assets *Profitability Ratios* Return on Assets Sequential Revenue Growth Profit Margin

Table 2
Three different types of features are used: Macroeconomic, bond-specific and company-specific variables.

Each parameter $x_1, x_2, \ldots, x_m$ is assigned a coefficient trough error minimization to predict the response variable y. The coefficients are noted $\beta_1, \beta_2, \ldots, \beta_m$ and the model takes the form:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_m x_m + \epsilon \tag{5}$$

where y is the recovery rate, ϵ is the error term and $x_1, x_2, ..., x_m$ are the explanatory variables. In our case, the bond characteristics, the company fundamentals and the default rate. Linear regressions are non-flexible models with low variance and high bias, therefore it is difficult to describe the non-linearity common to real life events. However, the simplicity of the model allows for a more convenient interpretation of the results. Another risk we encounter with linear regressions is that the predictions are not bounded by [0,1] even though recovery rates are. Concerning the categorical variables, they are implemented by creating dummy variables. The number of dummy variables in each category is equal to the number of possible values minus one. The category that is put aside is called the reference category and can be included in the regression without explicitly stating its value. The reason is that we do not want to give the regression redundant information. The chosen reference category is junior subordinated for debt seniority, and banking for the sector.

Qi and Zhao (2011) and T. Bellotti and Crook (2012) found that linear models could predict recovery rates as well as more complex models even though

there is a risk that the prediction falls out of the 0 to 1 range.

5.2 Random Forest

The random forest creates a multitude of new datasets using a resampling technique and builds a regression tree on each of these samples. Regression trees grew in popularity following the work of Breiman, Friedman, Stone, and Olshen (1984). They are named after the form of their structure, which looks like the root of a tree dividing itself into branches. The basic idea behind regression trees is to split the data into homogeneous subsets. The root of the tree contains all the observations and the branches are constructed by segmenting the data into subsets using the parameters of the model. At each node of the tree (a "parent") there is a logical if-then-else test on a specific explanatory variable, which splits the data into two other nodes (the "children"). For categorical variables, the partitioning is done on the belonging or not to certain groups. On discrete variables, the model splits the data based on a threshold. Every time there is a partitioning, the explanatory variable chosen is the one that minimizes the impurity of the parent node I_p which is given by

$$I_p = p_r I_r + p_l * I_l \tag{6}$$

where p_r, $p - l$ denote the probability of landing in the right or left child node and is calculated by the number of observations in the child node divided by the number of observations in the parent node. I_r and I_l denote the impurity of the child nodes and is measured by the variance inside the child node.

For a child node t totalling n observations x_i and of mean μ, the impurity is given by

$$I_t = \frac{1}{n} \sum_{i=1}^{n} (x_i - \mu_t)^2 \tag{7}$$

The splitting process continues in each node until the variance reduction generated by splitting the node is small enough, usually 1% or when the number of observations in that node reaches a certain minimum threshold. These nodes are called leaves, or terminal nodes. When the algorithm finishes modeling the tree, the prediction is equal to the average value of all the instances present in each leaf of the tree. Because of that, the predictions that the model makes have the same range as the training set. Hence, our recovery rates predictions are bounded by [0,1]. Regression trees are easy to interpret, they can handle non-linearity and they are quite robust to outliers.

Unfortunately, because these trees are able to represent very complex patterns with low bias and high variance, they are often prone to overfitting. In order to avoid overfitting, we construct what is called a random forest. The random forest applies the concept of bootstrap aggregating or bagging to our sample. The approach consists of resampling the original dataset into new random subsets with replacement. Basically, for an original dataset D of N observations, we create a certain number of new datasets D_i of the same lengths N, where we include randomly chosen observations from D. Because we pick

them randomly and "with replacement", a dataset D_i can contain more than once the same observation, or not contain it at all.

When the new datasets D_i are created, we optimize a regression tree on each of them. Once we obtain our models, we run new data through each of the trees and the final predictions equals the average of all the predictions given by each tree. With this technique we are able to build trees that have both high variance and low bias.

5.3 Neural Networks

5.3.1 The perceptron

Artificial neural networks were first discussed in 1943, at a time when computers were still at an early stage. The paper entitled "A logical calculus of the ideas immanent in nervous activity" combined the expertise of neurophysiologist Warren McCulloch and mathematician Walter Pitts in order to model the functioning of the human brain and its decision-making process (McCulloch & Pitts, 1943). The goal was to understand how the brain could process highly sophisticated patterns by using a collection of basic cells and replicate the process mathematically. Neural systems are constituted of billions of neurons, interconnected to each other, and each of them can be stimulated by signals. It starts with an initial stimulus that provokes a chain reaction from neurons to neurons through the firing of neurotransmitters; chemicals which transmit signals across the neurons. Neurons follow the "all-or-none" rule: there is a threshold in each neuron, estimated in millivolts, that the sum of impulses transmitted by connected neurons have to reach in order for the neuron to get activated and in turn pass on the signal to other neurons. If the threshold is not attained, nothing is "fired". The neurotransmitter does not only excite other neurons, it can also inhibit them, which means decreasing the probability that the next neuron will be activated. Rosenblatt (1958), inspired by the works of McCulloch-Pitts, later introduced the concept of perceptron. The perceptron, shown in Figure 7, is the most basic form of artificial neural network and was originally intended to perform binary classifications.

The model is still valid today and according to its creator "demonstrates the feasibility and fruitfulness of a quantitative statistical approach to the organization of cognitive systems".

An obervation is characterized by n features $x_1, x_2, x_3, \ldots, x_n$. These features are the inputs of the perceptron and each of them is represented by a weight $w_1, w_2, w_3, \ldots, w_n$. When the perceptron is activated, each input is multiplied by its weight and summed. The weighted sum, called the pre-activation value and denoted z is computed as

$$z = \sum_{i=1}^{n} W_i x_i = W^T X \tag{8}$$

We also add a neuron of input $x_0 = 1$, and weight w_0 called the "bias". Equivalent to the intercept in a linear regression, the goal of the bias is to shift

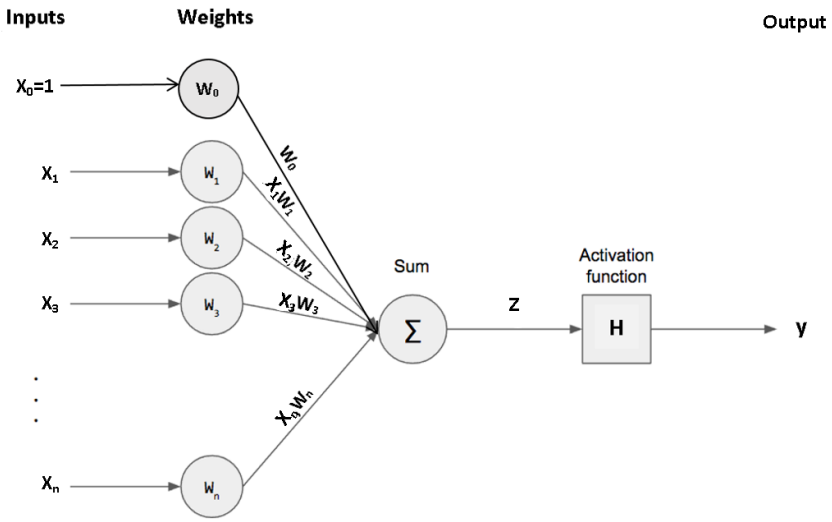

Figure 7
A reproduction of Rosenblatt's original perceptron.

the fitted function vertically. We rewrite the equation as

$$z = \sum_{i=1}^{n} W_i x_i + bias = W^T X + bias \tag{9}$$

It is fundamental to understand that at this stage the perceptron is nothing more than a basic linear regression with an intercept equal to the bias and the weights as the coefficients of the regression. If the weighted sum is equal or greater than a certain threshold q, we consider that the neuron is "activated" and the output y of the perceptron is equal to 1. Otherwise the output is null. This threshold is defined by the activation function. In the original perceptron the activation function is a Heaviside function, denoted H and defined by

$$H(z) = \begin{cases} 1 & if \quad z \geq q \\ 0 & if\ not \end{cases} \tag{10}$$

where $H(z)$ is the outcome of the perceptron, also referred to as y.

A Heaviside function with threshold $q = 0$ is shown in Figure 8 and illustrates the "all-or-none" character of the original perceptron.

One of the features that made perceptrons so groundbreaking at the time resided in its learning mechanism. Inspired by the work of Hebb (1949) who studied the brain's ability to learn by creating and changing the connections between neurons, Rosenblatt allowed the weights of each input to change value. The operation worked as follows:

1. At first, the weights are given random starting values.

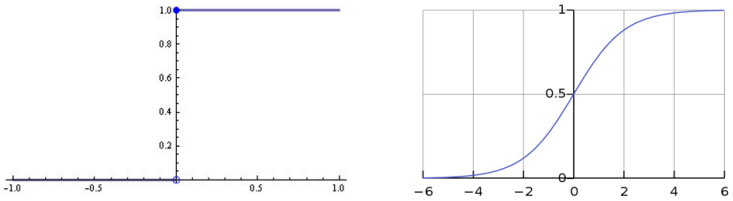

Figure 8
Two activation functions: the Heaviside function (left) and the sigmoid function (right). Source: Imperial College London METRIC online resource.

2. A set of inputs X is passed through the model, resulting in a binary classification, 1 or 0.

3. If the result y matches the true class d, the weights remain the same. If the results differ, the weights are adjusted using Rosenblatt's learning rule.

4. Another set of inputs is then used and the process is repeated until there is no more mistakes.

The learning rule used by Rosenblatt to adjust the weights is given by

$$\Delta w_i = \eta(d - y)x_i \tag{11}$$

where d is the desired output, y is our prediction, η is the learning rate and x_i is the ith input. The direction in which a weight w_i has to be adjusted is given by the difference $(d - y)$ and the sign of the input x_i. The learning rate is the size of the adjustment.

However, the simplicity of the perceptron had limitations. Because of the binary nature of the output, the basic model could only solve classification problems that were linearly dissociable. These constraints were heavily criticized by Minsky and Papert (1969) in their book *"Perceptrons: An Introduction to Computational Geometry"*. As a result of these critics, limited research was accomplished until the 80's as public interest plummeted and funding became insufficient. However, It did not prevent a few researchers to address the existing issues of neural networks and in 1974, Werbos introduced an improved version of the perceptron: the "multilayer perceptron with backpropagation" (Werbos, 1974). Unfortunately, because of the lack of interest in artificial intelligence at the time, it was only rediscovered and made famous by Rumelhart, Hinton, and Williams (1986).

5.3.2 Multilayer perceptron with backpropagation

The multilayer perceptron (MLP) gained the ability to model non-linear events by introducing two major changes. The first one is the addition of one or more layers between the input and output layer, called the hidden layers. A hidden layer is composed of hidden neurons and each of these neurons acts like

a simple perceptron. Their inputs are the outputs of the previous layer, which are then weighted, summed and passed trough the activation function before being transferred to the next layer. The structure of the MLP is shown in Figure 9.

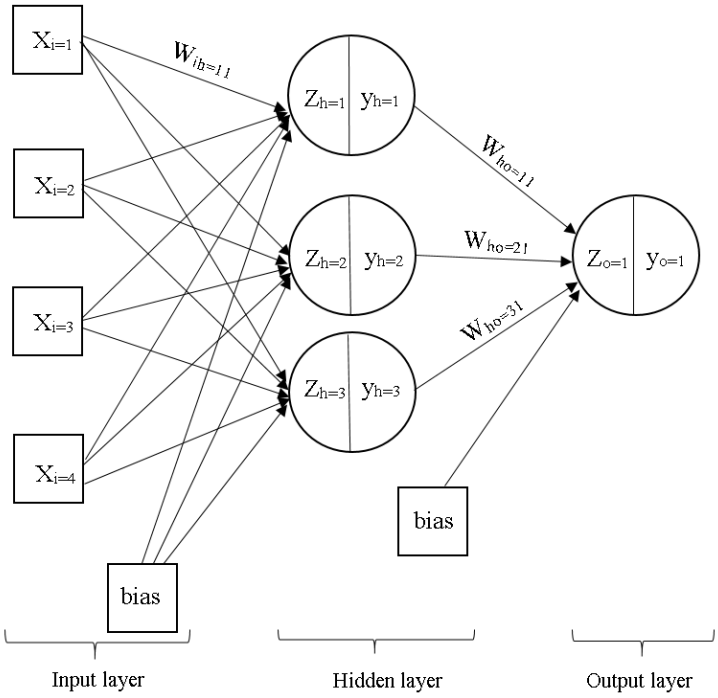

Figure 9
A multilayer perceptron with four inputs, one hidden layer of three hidden neurons and one output.

However, Rosenblatt's learning rule was only meant for perceptrons composed of an input and an output layer. The formula does not work when layers are added. This leads to the second major change which is the introduction of a new learning algorithm. In order for the multilayer perceptron to adjust its weights, Werbos (1974) introduced the concept of backpropagation with gradient descent.

As before, once we feed the network a set of inputs, we compare the prediction with the target value and compute the error. The goal of backpropagation is to assign the error to each of the weights and adjust them accordingly. The method is called "backpropagation" because in order to assign the blame of the error to a neuron, we need to compute the blames on the layer in front of it. In the case of a MLP with one hidden layer (Fig. 9), we first assign the blame of the error to the hidden neurons and use the results to assign the blame to the neurons of the input layer. Hence, the algorithm propagates the error backwards. In order to do so, it uses the gradient descent and the chain rule.

The gradient descent is an optimization algorithm used in multidimensional spaces that computes the partial derivatives of a function in order to find the steepest direction to either its minimum or maximum. In the case of neural networks, the goal is to minimize an error function, the sum of the squared errors, defined as

$$E = SSE = \frac{1}{2}(d - y_o)^2 \tag{12}$$

where d is the desired output, y_o is the prediction at the output layer and $\frac{1}{2}$ is there to offset the effect of the exponent when we differentiate later on.

The adjustment we make to a weight w situated on any of the layers is computed as

$$\Delta w = -\eta \frac{\delta E}{\delta w} \tag{13}$$

where $\frac{\delta E}{\delta w}$ is the partial derivative of E with respect to w and η is the learning rate. Indeed, the gradient indicates the sign of the adjustment and the learning rate is the size of the step we choose to make in that direction. The step size has to be carefully balanced. A smaller step size is safer but takes more time to optimize the network and could result in a local minimum. A bigger step size could lead to overshooting. The network adjusts the weight until it converges. Convergence means that the the prediction error is below a certain threshold.

To compute the gradient of the error function E with respect to the weight w we use a method called the "chain rule". The chain rule states that

$$\frac{\delta c}{\delta a} = \frac{\delta c}{\delta b}\frac{\delta b}{\delta a} \tag{14}$$

if $c = f(b)$ and $b = g(a)$

To have a better understanding on how the gradient is computed and the errors backpropagated, let us assume a neural network composed of one hidden layer and one output neuron. The network is illustrated in Figure 9. First, we adjust the weights w_{ho} situated between the hidden layer and the output layer. Then, we adjust the weights w_{ih} situated between the input and hidden layer.

1. weights between the **hidden** and **output** layer: The partial derivative of the error function E in respect to a weight w_{ho} situated between the hidden and output layer can be computed using the chain rule as follow

$$\frac{\delta E}{\delta w_{ho}} = \frac{\delta E}{\delta y_o}\frac{\delta y_o}{\delta z_o}\frac{\delta z_o}{\delta w_{ho}} \tag{15}$$

where y_o is the prediction of the network at the output layer, z_o is the pre-activation value of the output at the output layer and

$$\begin{aligned}\frac{\delta E}{\delta y_o} &= 2 * \frac{1}{2}(d - y_o) = d - y_o \\ \frac{\delta y_o}{\delta z_o} &= \frac{\delta f(z_o)}{\delta z_o} \\ \frac{\delta z_o}{\delta w_{ho}} &= y_h\end{aligned} \tag{16}$$

2. weights between the **input** and **hidden** layer: The partial derivative of the error function E in respect to a weight w_{ih} situated between the input and hidden layer can be computed using the chain rule as follow

$$\frac{\delta E}{\delta w_{ih}} = \frac{\delta E}{\delta y_h} \frac{\delta y_h}{\delta z_h} \frac{\delta z_h}{\delta w_{ih}} \tag{17}$$

where y_h is the output of the hth hidden neuron, z_h is the pre-activation value at the hth hidden neuron and

$$\frac{\delta E}{\delta y_h} = \frac{\delta E}{\delta z_o} \frac{\delta z_o}{\delta y_h} \tag{18}$$

where

$$\begin{aligned}\frac{\delta E}{\delta z_o} &= \frac{\delta E}{\delta y_o} \frac{\delta y_o}{\delta z_o} \\ \frac{\delta z_o}{\delta y_h} &= w_{ho}\end{aligned} \tag{19}$$

and

$$\begin{aligned}\frac{\delta y_h}{\delta z_h} &= \frac{\delta f(z_h)}{\delta z_h} \\ \frac{\delta z_h}{\delta w_{ih}} &= y_h\end{aligned} \tag{20}$$

To compute the gradient, we need to make sure that the activation function is differentiable, and thus a derivative exists at each point of its domain. The multilayer perceptron has to replace its original Heaviside function by a non-linear differentiable function.

5.3.3 Activation function

The most widely used non-linear functions for neural networks are the tanh and logistic function, also called sigmoid. To predict the recovery rates we use the sigmoid as it bounds the output between 0 and 1. The sigmoid function can be seen in Figure 8 and is computed by

$$\sigma = \frac{1}{1 + e^{-z}} \tag{21}$$

5.3.4 Hidden layer

To maximize the performance of the model we need to determine the right amount of hidden layers and hidden neurons.

Cybenkot (1989) demonstrated that a neural network with a single hidden layer and a sigmoidal activation function could approximate any continuous function. Following his research, Hornik (1991) established that a neural network with only one hidden layer could be a universal approximator regardless of the form of his activation function. This is known as the Universal approximation theorem. For these reasons our neural network is composed of a single hidden layer.

Now that we have fixed the number of hidden layers to a single one, we need adjust the number of hidden neurons. A lack of nodes could lead to underfitting, while a surplus could lead to overfitting. A large amount of hidden neurons will also increase the computational time necessary to train the network. To determine the best amount of neurons we resort to the previous literature.

Following G.-B. Huang (2003), the most adequate number of neurons in the hidden layer should be computed through this equation:

$$\textit{Number of hidden neurons} = 2\sqrt{(m+2)N} \tag{22}$$

where m is the number of outputs and N is the number of inputs.

In our case, we have 27 inputs and one output, thus the formula suggests the use of 18 neurons in the hidden layer.

5.3.5 Data transformation

We transform our continuous variables under the guidance of Sola and Sevilla (1997) who demonstrated that data normalization in neural networks was necessary to obtain good results and to fasten the training speed.

The most common normalization method in the literature is the Z-score normalization using the mean and standard deviation and computed as

$$z_{ij} = \frac{x_{ij} -}{\sigma(x_j)} \tag{23}$$

where z_{ij} is the normalized value of the ith observation of the jth feature and $\sigma(x_j)$ is the standard deviation.

Unfortunately, the z-score normalization is not robust to outliers. An outlier is an observation which seem to be inconsistent with the rest of the data. Hossari, Rahman, and Ratnatunga (2007) demonstrated that companies in financial trouble are expected to show abnormal ratios compared to healthy firms. Further, Guo (2008) highlighted the fact that financial ratios could show extreme values, especially close to the bankruptcy date. Due to the poor financial situation of the companies found in our database, we suspect the presence of outliers.

To handle these outliers, we use a robust normalization technique which replaces the median and the median absolute deviation (MAD) in place of the

mean and standard deviation. The MAD is a frequently used robust measure of the variability of a feature. Rousseeuw and Croux (2012) attributes its success to its simple formula, small computation time and robustness. The MAD of feature 0j can be calculated by

$$MAD_j = median(|x_{ij} - median(x_j)|) \tag{24}$$

The formula is a variation of the mean absolute value formula but it uses the median instead of the mean because it is less affected by outliers. The robust normalization is computed by

$$z_{ij} = \frac{x_{ij} - median(x_j)}{MAD_j} \tag{25}$$

In addition to the robust normalization of our continuous variables, we need to transform our categorical variables. Debt seniority, default type and sector have to be transformed into numerical variables in order to be included in the neural network. A method used by Tino, Schittenkopf, and Dorffner (2001) for his neural network is the *"one-hot"* encoding.

The method consists of creating an input for each category of the categorical variable. Every time an observation falls into a category, its value for that category is 1 and 0 for the others.

6 Results

6.1 In-sample analysis

In this subsection, we analyze how the models handle the data and we compare the relative importance of each parameter between models.

6.1.1 Linear Regression

To analyze the model's performance we use the R-squared and the p-value. We then investigate the relative importance of the variables with a step Akaike information criterion and the standardized coefficients.

1. R-squared:

   R-squared, also known as coefficient of determination, is a widely used and convenient indicator of the goodness of fit of a model. In the regression, R can be expressed as the proportion of variation of the dependent variable y that is explained by the independent variables $x_1, x_2, \ldots, x_n$ of the model. In simpler terms, the higher the R-squared, the better the model.

   $$R^2 = 1 - \frac{Residual\ Sum\ of\ Squares}{Total\ Sum\ of\ Squares} \tag{26}$$

   We obtain a fairly low R^2 of 0.26 which means that only 26% of the variation of the recovery rates are explained by the regression. If we

look at the literature, we can find that these results match the previous research. Yang and Tkachenko (2012) obtained similar results with an R^2 of 0.2. Baesens (2015) also notices that models trying to predict the Loss Given Defaults usually obtain *"awful performances"* with an R^2 typically sitting at less than 0.3.

2. p-value:

   The p-value is the probability of finding the observed, or more extreme results, when the null hypothesis is true.[2] Even though it is often questioned by the scientific community, it is widely accepted that a p-value of less than 0.05 indicates a significant parameter. If the p-value is lower than a certain threshold we can reject the null hypothesis and thus confirm the significance of the variable. If it is higher than that threshold, we cannot reject the null hypothesis, and thus cannot say for sure that the parameter is significant. Most authors denote a p-value lower than 5% as "statistically significant", and a p-value lower than 0.1% as "statistically highly significant".

   The five variables with the highest p-value are:

   (a) Senior secured first lien: high seniority bonds seem to obtain better recovery rates which was expected as the first debtor to be in the line for his debt should be the one that has the most chances to recover his investment.

   (b) Default rate: recovery rates decrease as the American default rate increases. In accordance with the literature which showed a inverted correlation between LGD and PD.

   (c) Cash flow to total liabilities: the regression shows that as the company brings more cash compared to its debt, it increases its ability to repay the debtors. Indeed, assets that are reliable cash producers are valued at a higher price.

   (d) Utilities sector: same as in the literature, utilities show a positive impact on recovery rates in the linear model.

   (e) Coupon: The coupon rate has a positive impact on the recovery rates.

   On the contrary profit margin, firm size and total debt to total asset showed extremely high p-values, meaning that we cannot confirm their significance in the prediction of RR.

   Parameters with a positive effect include the revenue sequential growth, the long term to total asset, the return on asset and the proportion of fixed asset. Long term debt to total capital seems to have affect the recovery rates negatively.

[2] The p-values here are computed under the null hypothesis $H_0 : \beta = 0$

3. Akaike Information Criterion: The Akaike Information Criterion, also known as AIC, is an estimator of the quality of a statistical model and it is meant to compare models with different parameters for a given set of data. The criterion only measures the relative quality of a subset of predictors compared to other subsets and not the overall quality of the model. AIC indicates the best model out of the suggested ones and does not deliver any warning if all subsets are poor estimators. The AIC is computed by

   The AIC is calculated by

$$AIC = 2k - 2ln(\hat{L}) \tag{27}$$

   where k is the number of parameters and indicates the complexity of the model. $\hat{L}$ is the likelihood function and indicates the goodness of fit.

   By using a stepAIC, we compute the AIC of different subsets by adding or removing parameters. The output is the model with the lowest AIC:

   (a) Debt Seniority
   (b) Default Rate
   (c) Cash flow to total liabilities
   (d) Sector
   (e) Coupon

   We consider these variables to be ones with the biggest impact on the recovery rates in the linear model.

6.1.2 Random Forest

Because it would be impossible to visualize all the trees of the random forest, we construct a single regression tree shown in Figure 10. Regression trees are great visual tools for interpretation. By examining the architecture of the tree we can already have a sense of how the tree uses the parameters to split the data. We can have a good comprehension on which parameters have a positive or negative impact on the recovery rates. Indeed, every time a parameter separates data, the lower recovery rates tend to land in the left node and the higher recovery in the right node. The categorical variables used in the tree can be seen in table **??**

The tree tries to assemble groups of homogeneous defaulted bonds with the least amount of intra-group variance. Hence, the first splitting parameters is the one that can best differentiate the data into two subgroups. By looking at the initial splits of the regression tree we can already identify the variables that have the largest influence on the predictions like the debt seniority, the American default rate, the sector and the cash flow to liabilities.

There exists another technique to have a more precise measure of the importance of each variable. For a regression tree, at each split we compute the

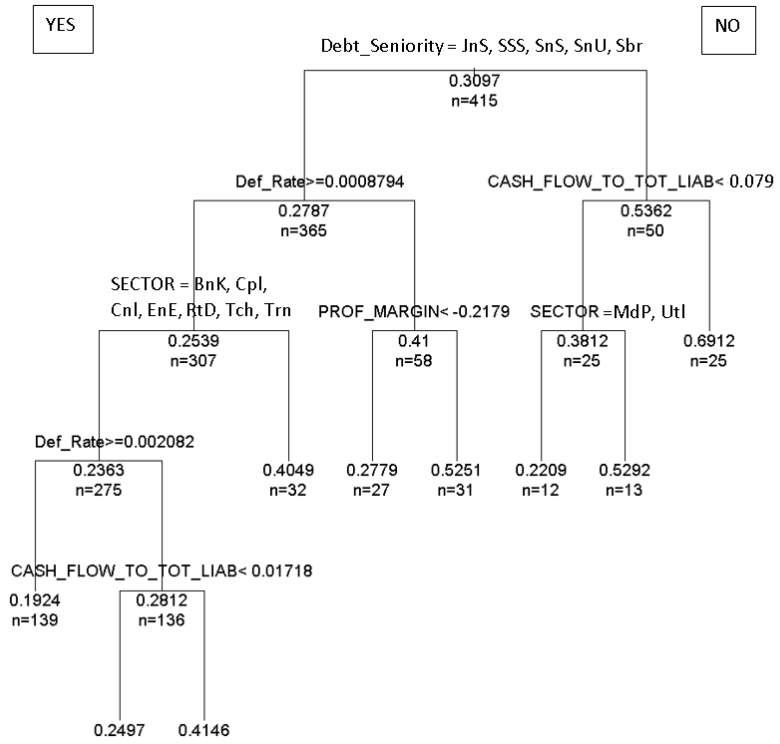

Figure 10
Visual interpretation of the regression tree.

Code	Debt Seniority	Code	Sector
JnS	Junior Subordinated	Bnk	Banking
SSS	Senior Secured Subordinated	Cpl	Capital Industries
SnS	Senior Subordinated	Cnl	Consumer Industries
SnU	Senior Unsecured	EnE	Energy & Environment
Sbr	Subordinated	RtD	Retail & Distribution
		Tch	Technology
		Trn	Transportation
		MdP	Media & Publishig
		Utl	Utilities

Table 3
Codes used by the regression tree in Figure 10.

residual sum of squares (RSS) between the parent node and the child nodes and

attribute the decrease in RSS to the variable that determines the split. The decrease of the error is also called the "increase in purity".

The decrease in RSS is computed as follow:

$$IncNodePurity = RSS_p - RSS_l - RSS_r \tag{28}$$

where p is the parent node, l,r are the child nodes and RSS_t for node t composed of n observations and an intra-group mean μ_t is computed by

$$RSS_i = \sum_{i=1}^{n} (x_i - \mu_t)^2 \tag{29}$$

To compute the purity increase for each variable in the random forest we average the purity increases over all the trees of the random forest and plot it in in Figure 11

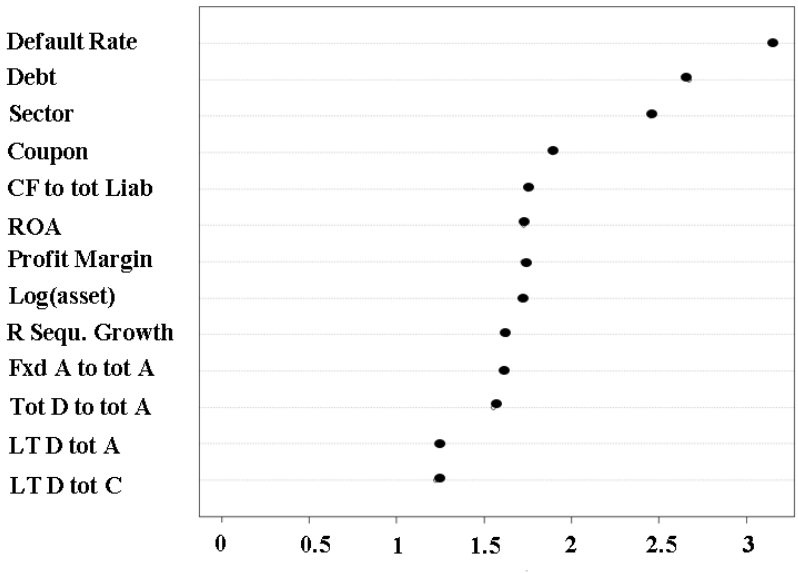

Figure 11
Importance of each variable using the increase in node purity.

6.1.3 Neural Network

Even though neural networks are known to be black-box models, i.e. hard to interpret, a few researchers in the community attempted to evaluate the impact of each input on the predictions.

Garson (1991) suggested an algorithm that evaluates the relative importance of each variable using the absolute value of the weights. He defines the relative

importance of an input of a single hidden layer network as

$$R_i = \frac{\sum_{h=1}^{3} |w_{ih} w_{h1}|}{\sum_{h=1}^{3} |w_{h1}|} \tag{30}$$

where w_{ih} represents the weight between the ith input and the hth hidden unit, and w_{h1} is the weight between the hth hidden unit and the output.

Olden, Joy, and Death (2004) argued that Garson's use of absolute values would fail to account for contrasting influences in the model. He defined contrasting influences as the change in sign that the weights might experience between each layer for the same input. He suggested a simpler algorithm, which summed the product of the weights between the input-hidden and hidden-output layer. This approach, referred to as the "Connection Weight" method, defines the relative importance of an input of a single hidden layer network as

$$R_i = \sum_{h=1}^{3} w_{ih} w_{h1} \tag{31}$$

Before starting the learning process, neural networks are given random weights. As a result, each neural network's weights differ slightly from one network to the other. In order to have homogeneous results, we construct a thousand neural networks with the same criteria and average the resulting weights. Garson's algorithm and Olden's connection weight model can be seen in Figure 12 and Figure 13.

We observe similarities between the neural network and the previous models in terms of variable importance. Debt seniority, sector and default rates are still the most impactful parameters of the model. However, even though cash flow to total liabilities and the coupon seemed to have some effect on the recovery rates in previous models, it does not seem to be the case in the neural network.

Using the connection weight's ability to detect the sign of the variable influence, we notice that the technology sector, the default rate and the size of the firm has a negative impact on recovery rates, which coincides with the previous models.

6.2 Out-sample analysis

In this subsection, we compare each model by predicting unseen data through cross-validation and compute two error measurements. In addition to that, we also assess the impact of each model in the measurement of the VaR by computing the estimated loss of 20,000 simulated portfolios.

6.2.1 Cross-validation

K-fold cross-validation is a resampling technique that allows us to train and test the model on every observation of the dataset. The idea is to train and test the

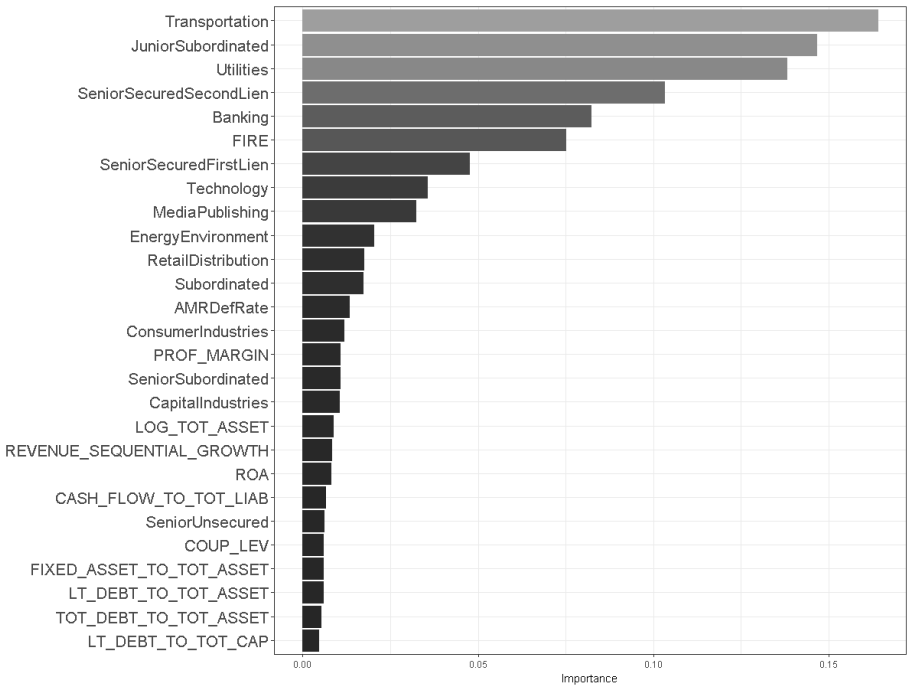

Figure 12
Relative importance using Garson's algorithm.

model on different subsets and average the predictions.

1. Holdout method: The holdout method is the simplest cross-validation. We divide the data into two subsets: the training and the test set. Only the training set is used to train the model, then we ask the model to predict the unseen recovery rates of the test set. The issue with this method is that the results can vary a lot depending on which data has been selected for training or test set.

2. K-fold cross-validation: k-fold cross-validation derives directly from the holdout method. We don't have to divide the data into two samples anymore but we can choose any number k. We divide the dataset into k different subsets and we repeat the holdout method k times. We use all the k samples except one for training and then test the model on that one sample that was not part of the training. We repeat this K times so that all the data has been used for training and testing. This allows us to avoid the randomness of holdout method.

3. Leave-one-out cross-validation : Leave-one-out cross-validation (LOOCV) is an extreme case of the k-fold cross-validation technique which will set K as the number of observations in the model. In our case, we have

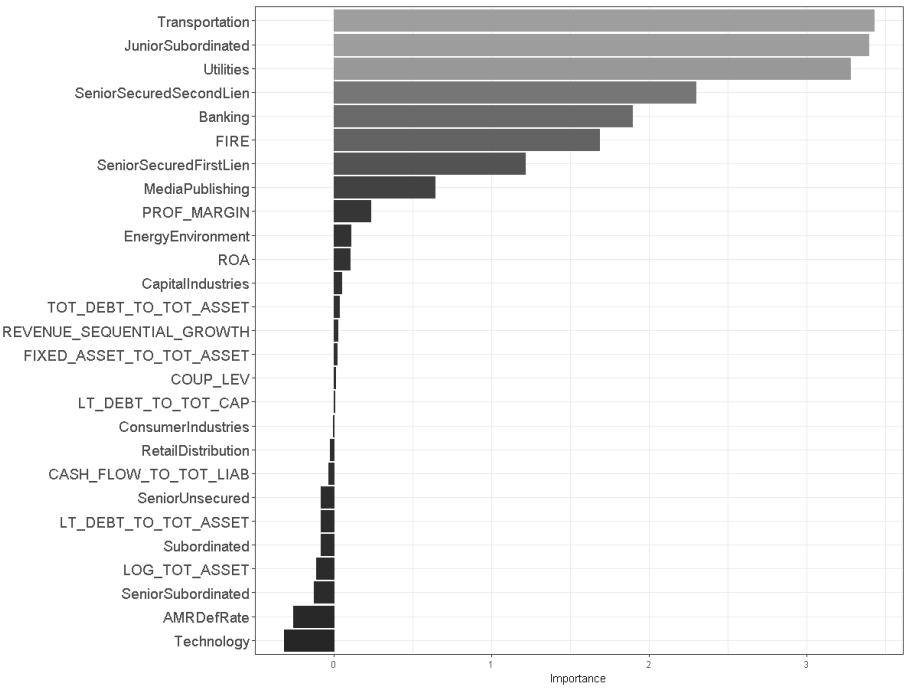

Figure 13
Relative importance using Olden's "Connection Weight" algorithm.

415 observations so we divide the dataset into 415 samples, one for each observation. We then train the model on all of these samples except one that we use for testing and then repeat 415 times until all observations have been used for testing and training. One of the drawbacks of that method is that it is very energy and time consuming as the model has to be trained N times depending on the number of observations.

In this study we compare the results of a 5-fold cross-validation, 10-fold cross-validation and an LOOCV.

6.2.2 Performance measures

We measure the performance of the models using the root-mean-square error (RSME) and the mean absolute error (MAE), computed by

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(RR_i - \hat{RR_i})^2} \tag{32}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|RR_i - \hat{RR_i}| \tag{33}$$

where n is the number of observations, RR_i is the actual value of the observation and $\hat{RR_i}$ is the predicted value. RMSE and MAE are used together because they handle errors differently. In the case of RMSE, the residuals are squared before being averaged, so larger errors have more weight than in the MAE. The results are displayed in Table 4 .

Model	5-fold cross-validation	10-fold cross-validation	LOOCV
Neural Network			
RMSE	0.2790	0.2421	0.1880
MAE	0.2402	0.1952	0.1880
Random Forest			
RMSE	0.2214	0.2197	0.1775
MAE	0.1807	0.1790	0.1775
Linear Regression			
RMSE	0.2346	0.2311	0.1834
MAE	0.1863	0.1843	0.1834

Table 4
Out-sample results of the RMSE and MAE without an overestimation penalty.

The random forest seems to showcase the best performance, followed by the linear regression and then the neural network. An interesting observation is that even though the neural network does not perform as expected, its forecasting abilities improve drastically as the model is trained with more observations.

In addition, we also compute an asymmetric RMSE and MAE. The idea behind asymmetry is to penalize models that overestimate the recovery rates. Overestimations in recovery would lead to underestimating the loss taken by the banks which could, to some extent, weaken the stability of the banking system.

The asymmetric RMSE and MAE are computed as

$$Asymmetric \quad RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(RR_i - \hat{RR_i})^2 * (1 + P_i)} \tag{34}$$

$$Asymmetric \quad MAE = \frac{1}{n}\sum_{i=1}^{n}\left(|RR_i - \hat{RR_i}| * (1 + P_i)\right) \tag{35}$$

where P_i, the penalty for the ith observation, equals 1 when $RR_i < \hat{RR_i}$, and 0 otherwise. The results are shown in Table 5.

The results of Table 5 are consistent with our previous results for RMSE and MAE. We notice that the neural network overestimates more recovery rates than the other models. The linear regression still performs worse than the random forest but NO, the number of overpredicted observations, is lower.

Results from Table 4 and 5 suggest poor performances for the neural network, both in terms of RMSE and MAE. A possible cause could be the fact

Model	5-fold cross-validation	10-fold cross-validation	LOOCV
Neural Network			
RMSE	0.2934	0.2860	0.2281
MAE	0.3026	0.2932	0.2832
NO	250	252	252
Random Forest			
RMSE	0.2849	0.2776	0.2161
MAE	0.2994	0.2891	0.2700
NO	242	241	244
Linear Regression			
RMSE	0.2796	0.2741	0.2231
MAE	0.2825	0.2784	0.2770
NO	241	239	236

Table 5
Out-sample result of the asymmetric RMSE and MAE with an overestimation penalty, where NO is the number of overestimations.

that recovery rates are known to be highly difficult to estimate. This is theory is supported by the low R-squared obtained in the literature and our own observations. Furthermore, neural networks are very efficient when they are trained on extensive sample sizes. Recovery rates typically suffer from scarcity of the data because the probability of observing defaulted bonds is generally low and the fact that banks have only started to collect extensive recovery data after the introduction of the AIRB approach. Finally, the difficulty of training the neural network could be the cause of its poor performance. Indeed, neural networks use many hyperparameters. In machine learning, a hyperparameter is a parameter of the model whose value has to be set before starting the learning process. For neural networks, these parameters include the activation function, the learning rate, the number of hidden layers and the number of neurons in each layer. The neural network for an accurate prediction of the recovery rates could be a single combination of all these parameters.

6.2.3 VaR simulation

However, what do an RMSE of 25% or an MAE of 18% exactly imply in terms of financial stability? These figures give no information whatsoever about the estimation of the risk and the capital requirements. In order to have a better idea on how these models impact the estimation of the yearly Value-at-Risk, we compute the loss of 20,000 simulated portfolios. We remind that the loss of a portfolio of N bonds is computed by

$$L = \sum_{n=1}^{N} EAD_n * (1 - RR_n) * D_n \tag{36}$$

where D_n is a binary variable and takes a value of 1 if there is a default and 0 otherwise. EAD is considered fixed for all bonds and equals 100. We convert

LGD_n to one minus RR_n for convenience.

Each portfolio is composed of 500 fictitious bonds created using the same parameters as the models (debt seniority, coupon, profit margin, etc.). We randomly assign values to the parameters of the bonds by following the density function of each parameter. A good amount of literature has been devoted to observing the dependency between probability of default and recovery rates. We acknowledge the dependency in our calculations by using the same PD to estimate RR and D_n, the binary variable that equals 1 if the bond defaults and 0 otherwise.

Every time we generate a portfolio, we predict the recovery rates of each bond using the previously constructed neural network, random forest and linear regression. We highlight the fact that all models are trained on the exact same portfolios. We then calculate the loss of each portfolio using the loss function above. Once we have computed the loss of each of the portfolios, the value at quantile q represents the VaR_q. We show the outcome of the simulation in Table 6.

Model	$\text{VaR}_{0.99}$	$\text{VaR}_{0.995}$	$\text{VaR}_{0.999}$	VaR_1
Neural Network	1768.871	1851.797	2033.964	2329.154
Random Forest	1557.084	1631.533	1785.715	2048.636
Linear Regression	1816.821	1919.314	2104.344	2435.53

Table 6
Yearly Value-at-Risk estimated by each model on 20,000 portfolios at quantile $\alpha = 0.99, 0.995, 0.999$ and 1.

The results show that the linear regression expects higher maximum losses, while the random forest displays a significantly lower VaR than the other models. The VaR in itself does not help us determine the best model. In fact, it all depends on the objective of the concerned institution. A bank could be more interested in the random forest to lower its capital requirements and increase profitability. On the other hand, regulators could be more inclined to encourage the use of a linear regression to improve the stability of the banking system. Nevertheless, given its prediction performance, its impact on the VaR and its complexity, we do not advice the use of the neural network, as computed in this study, for predicting the recovery rates.

7 Conclusion

This study evaluates the performance of a neural network to forecast the recovery rates of defaulted corporate bonds. We first demonstrate the benefits of estimating the recovery rates instead of a standardized fixed rate by simulat-

ing the maximum loss of 20,000 portfolios in the case of static and stochastic recovery rates.

We then implement a neural network composed of one hidden layer 18 hidden neurons. Each neuron is activated using a sigmoid function in order to ensure a prediction bounded by the unit interval. In the in-sample analysis we evaluate the relative importance of the variables in each model. More specifically we use Garson and Olden's algorithms to estimate the effect of each input on the output. The relative importance seems to coincide between models.

The out-sample analysis indicates the performance of all three models using an RMSE and an MAE, with and without an overestimation penalty. The results show that neural networks perform poorly compared to the other models. The random forest obtains the best predictive performance. We blame the lack of performance of the neural network to the difficulty of predicting recovery rates, the scarcity of the data and the complexity to tune the network's parameters.

Finally, we generate 20,000 portfolios of 500 bonds. Each bond is characterized by the same variables as the inputs of the models. We then compute the yearly Value-at-Risk estimated by each model and determine that the random forest shows the lowest VaR while the linear regression shows the highest. We conclude that the neural networks, as computed in this study, would be a poor choice for estimating the recovery rates due to its poor performance, its VaR and its computation complexity. Nevertheless, we still encourage to experiment with neural networks for recovery rates prediction. A bigger dataset and the right tuning of the network's hyperparameters could lead to acceptable results.

References

Acharya, V. V., Bharath, S. T., & Srinivasan, A. (2003). *Understanding the recovery rates on defaulted securities.*

Acharya, V. V., Bharath, S. T., & Srinivasan, A. (2007). Does industry-wide distress affect defaulted firms? evidence from creditor recoveries. *Journal of Financial Economics*, *85*(3), 787-821.

Altman, E., Brady, B., Resti, A., & Sironi, A. (2005). The link between default and recovery rates: Theory, empirical evidence, and implications. *The Journal of Business*, *78*(6), 2203-2228.

Altman, E., Resti, A., & Sironi, A. (2003). *Default recovery rates in credit risk modeling: A review of the literature and empirical evidence.*

Baesens, B. (2015). *State of the art in credit risk modeling.* Retrieved from `Retrieved from https://www.youtube.com/watch?v=PnKwz`$_j qgzM$ (Webinar)

Baesens, B., & Gestel, T. V. (2009). *Credit risk management: Basic concepts.* Oxford University Press.

Basel Committee on Banking Supervision. (1983). *Principles for the management of credit risk.* Bank for International Settlements.

Basel Committee on Banking Supervision. (2005). *An explanatory note on the basel ii irb risk weight functions.* Bank for International Settlements.

Bastos, J. A. (2010). Predicting bank loan recovery rates with neural networks. *Technical University of Lisbon Working Paper*.

Bellotti, A., & Crook, J. (2012). Loss given default models incorporating macroeconomic variables for credit cards. *International Journal of Forecasting*, *28*(1), 171-182.

Bellotti, T., & Crook, J. (2012). Loss given default models incorporating macroeconomic variables for credit cards. *International Journal of Forecasting*, *28*(1), 171-182.

Boyacioglu, M. A., Karab, Y., & Baykanc, K. (2009). Predicting bank financial failures using neural networks, support vector machines and multivariate statistical methods: A comparative analysis in the sample of savings deposit insurance fund (sdif) transferred banks in turkey. *Expert Systems with Applications*, *36*(2), 3355-3366.

Breiman, L., Friedman, J., Stone, C. J., & Olshen, R. (1984). *Classification and regression trees*. Taylor Francis.

Bris, A., & Welch, I. (2005). The optimal concentration of creditors. *The Journal of Finance*, *60*.

Bruche, M., & González-Aguado, C. (2010). Recovery rates, default probabilities and the credit cycle. *Journal of Banking and Finance*, *34*(4).

Calabrese, R. (2014). Downturn loss given default: Mixture distribution estimation. *European Journal of Operational Research*, *237*(1), 271-277.

Covitz, D. M., & Han, S. (2005). An empirical analysis of bond recovery rates: Exploring a structural view of default. *Finance and Economics Discussion Series Working Paper No 2005-10*.

Cybenkot, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals, and Systems*, *2*, 303-314.

Dalbor, M. C., Kim, A., & Upneja, A. (2004). An initial investigation of firm size and debt use by small restaurant firms. *Journal of Hospitality Financial Management*, *12*(1).

Das, S. R., Hanouna, P., & Sarin, A. (2008). Accounting-based versus market-based cross-sectional models of cds spreads. *Journal of Banking Finance*, *33*(4), 719-730.

Duan, J.-C., & Hwang, R.-C. (2016). Predicting recovery rate at the time of corporate default. *National University of Singapore Working Paper*.

Fadlalla, A., & Lin, C.-H. (2001). An analysis of the applications of neural networks in finance. *Interfaces*, *31*(4), 112–122.

Frye, J. (2002). *Depressing recoveries.* Federal Reserve of Chicago.

Frye, J., et al. (2000). *Collateral damage: A source of systematic credit risk.*

Garson, G. D. (1991). Interpreting neural-network connection weights. *AI Expert*, *6*(4), 46 - 51.

Guo, W. (2008). *Financial ratios as predictors of failure: Evidence from hong kong using logit regression* (Master Thesis). Rotterdam School of Management.

Gupton, G. M. (2005). Advancing loss given default prediction models: How the quiet have quickened. *Economic Notes*, *34*(2), 185-230.

Hebb, D. . (1949). *The organization of behavior.* John What Sons.

Hornik, K. (1991). Approximation capabilities of multilayer feedforward networks. *Neural Networks*, *4*(2), 251-257.

Hossari, G., Rahman, S. F., & Ratnatunga, J. (2007). An empirical evaluation of sampling controversies in ratio-based modelling of corporate collapse. *The Journal Of Applied Research In Accounting And Finance*, *2*(2).

Huang, G.-B. (2003). Learning capability and storage capacity of two-hidden-layer feedforward networks. *IEEE Transactions On Neural Networks*, *14*(2).

Huang, X., & Oosterlee, C. W. (2011). Generalized beta regression models for random loss given default. *The Journal of Credit Risk*, *7*(4).

Hurlin, C., Leymarie, J., & Patin, A. (2018). *Loss functions for lgd model comparison.*

Jankowitsch, R., Nagler, F., & Subrahmanyam, M. G. (2014). The determinants of recovery rates in the us corporate bond market. *Journal of Financial Economics*, *114*(1), 155-177.

Jarrow, R. (2001). Default parameter estimation using market prices. *Financial Analysts Journal*, *57*(5), 75-92.

Lopez, J. A., & Saidenberg, M. R. (2001). *The development of internal models approaches to bank regulation supervision: Lessons from the market risk amendment.*

Lütkebohmert, E. (2009). *Concentration risk in credit portfolios.* Springer-Verlag Berlin Heidelberg.

MacKay, P., & Phillips, G. M. (2002). How firm characteristics affect capital structure: An international comparison. *National Bureau of Economic Research Working Paper No. 9032*.

McCulloch, W. S., & Pitts, W. H. (1943). A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, *5*, 115-133.

Moody's Analytics. (2005). *Default and recovery rates of corporate bond issuers, 1920-2004.*

Oh, S. (2018). Fire-sale acquisitions and intra-industry contagion. *Journal of Corporate Finance*, *50*, 265-293.

Olden, J. D., Joy, M. K., & Death, R. G. (2004). An accurate comparison of methods for quantifying variable importance in artificial neural networks using simulated data. *Ecological Modelling*, *178*, 389-397.

Pacelli, V., & Azzollini, M. (2011). An artificial neural network approach for credit risk management. *Journal of Intelligent Learning Systems and Applications*, 103-112.

Qi, M., & Zhao, X. (2011). A comparison of methods to model loss given default. *Journal of Banking and Finance*, *35*, 2842-2855.

Rosenblatt, F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological Review*, *65*(6).

Rousseeuw, P. J., & Croux, C. (2012). Alternatives to the median absolute deviation. *Journal of the American Statistical Association*, *88*(424), 1273-1283.

Ruiz, I. (2015). Default probability, loss given default, and credit portfolio models. In *Xva desks — a new era for risk management* (p. 104-125). Palgrave Macmillan.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning internal representations by error propagation. In *Parallel distributed processing: Explorations in the microstructure of cognition* (Vol. 1, p. 318-362). MIT Press.

Russell, S. J., & Norvig, P. (2010). *Artificial intelligence: A modern approach*. Prentice Hall.

Schläfer, T. (2011). A new approach to estimating market-implied recovery rates. In *Recovery risk in credit default swap premia* (p. 19-25). Gabler.

Schuermann, T. (2004). What do we know about loss given default? *Wharton Financial Institutions Center Working Paper No. 04-01*.

Sigrist, F., & Stahel, W. A. (2012). *Using the censored gamma distribution for modeling fractional response variables with an application to loss given default.*

Sola, J., & Sevilla, J. (1997). Importance of input data normalization for the application of neural networks to complex industrial problems. *IEEE Transactions On Nuclear Science*, *44*(3), 1464 - 1468.

Tino, P., Schittenkopf, C., & Dorffner, G. (2001). Financial volatility trading using recurrent neural networks. *IEEE Transactions On Neural Networks*, *12*(4), 865 - 874.

Varma, P., & Cantor, R. (2004). *Determinants of recovery rates on defaulted bonds and loans for north american corporate issuers: 1983-2003.* Moody's Investors Service.

Wald, J. K. (1999). How firm characteristics affect capital structure: An international comparison. *The Journal of Financial Research*, *22*(2), 161-187.

Werbos, P. J. (1974). *Beyond regression: New tools for prediction and analysis in the behavioral sciences* (PhD thesis). Harvard University.

Yang, B. H., & Tkachenko, M. (2012). *Modeling of ead and lgd: empirical approaches and technical implementation* (MPRA Paper). University Library of Munich, Germany.

Yao, X., Crook, J., & Andreeva, G. (2015). Support vector regression for loss given default modelling. *European Journal of Operational Research*, *240*(2), 528–538.

Yashkir, O., & Yashkir, Y. (2013). Loss given default modelling: Comparative analysis. *Journal of Risk Model Validation*, *7*, 25-59.

Zhang, J., & Thomas, L. C. (2012). Comparisons of linear regression and survival analysis using single and mixture distributions approaches in modelling lgd. *International Journal of Forecasting*, *28*, 204-215.