VALUATION OF FIXED INCOME SECURITIES AND ESTIMATION OF TERM STRUCTURE ON INTERNATIONAL BOND MARKET USING MACHINE LEARNING TECHNIQUES

A Thesis

by

Ali Dartanel

Submitted to the
Graduate School of Sciences and Engineering
In Partial Fulfillment of the Requirements for
the Degree of

Master of Science

in the
Department of Financial Engineering

Özyeğin University
May 2022

VALUATION OF FIXED INCOME SECURITIES AND ESTIMATION OF TERM STRUCTURE ON INTERNATIONAL BOND MARKET USING MACHINE LEARNING TECHNIQUES

Approved by:

Assistant Prof. Emrah Ahi,
Advisor,
Department of International Finance
Özyeğin University

Assistant Prof. Levent Güntay,
Department of International Finance
Özyeğin University

Assistant Prof. Erdinç Akyıldırım,
Department of Management
Boğaziçi University

Date Approved: 27 May 2022

ABSTRACT

In this study, I focus on predicting bond risk premia in Turkish Eurobonds market using machine learning methods. Machine learning uses statistical learning techniques to gather useful structures of a data set without being explicitly programmed. In recent years machine learning has become a very popular topic and shown very good results in a wide variety of fields, but there is a lack of research in the field of term structure modeling. In order to predict Turkish Eurobond returns, I implemented several machine learning models such as OLS, PCA, Ridge, Lasso, Elastic net and neural networks. The raw data set I used comprises of Turkey Government Eurobond yields between 2005 and 2020, inclusive. Both monthly and yearly returns are estimated separately. Zero-coupon rates and forward rates are calculated from the raw data and used as left-hand site elements for machine learning predictions. Macroeconomic variables are also added to forward rates as factors. I compared the out-of-sample performance of the models and I found that Penalized linear regression yields the best results for excess bond return prediction, providing nearly 10% out-of-sample R^2. Neural networks are the second-best performer yielding around 3-4% out-of-sample R^2. Plus, adding macroeconomic variables to the models slightly improved the results by 2-3%. Also, yearly returns estimation performed better than monthly returns for OLS, Ridge, Lasso and Elastic net regressions, but not for neural networks.

ÖZETÇE

Bu tez çalışması, Türkiye Eurobond piyasası getiri eğrisini ve farklı vadelerdeki bono getirilerini makine öğrenmesi yöntemleri kullanılarak modellenmesine odaklanacaktır. Makine öğrenimi, detaylıca programlanmadan bir veri kümesinin yararlı yapılarını toplamak için istatistiksel öğrenme tekniklerini kullanır. Son yıllarda makine öğrenimi çok daha popüler bir konu haline geldi ve çeşitli alanlarda çok iyi sonuçlar gösterdi, ancak getiri eğrisini modelleme alanında araştırma eksikliği olduğu dikkatimizi çekmekte. Faiz oranlarının vade yapısını modellemek için yapay sinir ağları, derin öğrenme yöntemleri, doğrusal modeller, cezalı doğrusal regresyonlar ve boyutluluk azaltma teknikleri gibi makine öğrenmesi tekniklerini kullandık. Ham data, 2005 ve 2020 yılları arasındaki Turkiye Eurobondları faizlerinden oluşturulmuştur. Aylık ve yıllık getiriler ayrı ayrı tahmin edilmiştir. Ham data kullanılarak önce kuponsuz tahvil getiri eğrisi, sonraısında da ileri vadeli getiriler hesaplanmış ve modellerin bağımlı değişkenleri olarak dahil edilmiştir. Bu değişkenlere makro ekonomik veriler de eklenerek ayrıca tahmin yapılmıştır. Modellerin örneklem dışı performanslarını kıyaslarak incelendiğinde cezalı doğrusal regresyonların yaklaşık 10% R^2_{OOS} sağlayarak en başarılı modeller olduğu görülmüştür. Sinir ağları modelleri 3-4% R^2_{OOS} ile en başarılı ikinci performansı sergilemiştir. Makro verilerin modellere değişken olarak eklenmesi sonuçları 2-3% iyileştirmiştir. Ayıca, doğrusal ve cezalı doğrusal modeller için yıllık getiri tahminleri aylık getiri tahminlerine göre daha iyi sonuç vermiştir.

ACKNOWLEDGMENTS

I would like to thank my advisor Assistant Prof. Emrah Ahi, at the Department of International Finance under the Graduate School of Business at Özyeğin University. He has put a lot of work in this paper, providing consistent help with data analysis as well as being a prosperous guide. Also, I would like to thank Assistant Prof. Levent Güntay, at the Department of International Finance under the Graduate School of Business, for his invariable support especially with technical details regarding neural network models.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

The fixed-income market, whose total market value outstanding of corporate and government bonds is in fact significantly larger than that of equities, is a key source of financing for businesses and governments. It also represents a significant investing opportunity for institutions such as pension funds, mutual funds, insurance companies, sovereign wealth funds, as well as individuals. Understanding the valuation of fixed-income securities is crucial for investors, issuers, and financial analysts. Plus, interest rates play a vital role in valuation of financial assets and for making macroeconomic policy. However, being not always simple and stable, interest rates are calculated and usually estimated from the market prices of government securities with different maturities. Such alternative assets as real estate, private equity, hedge fund investments are illiquid with long-term cash flows, making it more difficult to provide a source for market prices. Thus, a properly estimated term structure of interest rates is essential for obtaining the intrinsic values of these assets.

Basic bond valuation methods include pricing a bond using a market discount rate for each of the future cash flows and pricing a bond using a series of spot rates. Valuation using spot rates allows for each future cash flow to be discounted at a rate with its associated maturity. The term structure interest rate estimation requires making important decisions. For example, assumptions related to taxes and liquidity premiums should be considered. Specific functional form to approximate the interest rates as well as an empirical method for estimating the parameters must be chosen. Besides, the yield curve

is a key representation of the state of the bond market. The slope of the yield curve is an important indicator of expectation for future interest rates, growth and inflation. So, this has been a subject of considerable research. Many statistical techniques and tools commonly used in econometrics and finance have been applied to model the yield curve. In addition, some methods to estimate the default-free term structure of interest rates include the bootstrapping method, the McMulloch cubic-spline method, the Nelson-Siegel-Svensson method.

Bootstrapping is vital for calculating zero-coupon yields from the raw yield data obtained from the market. Then, calculating forward rates from the zero-yields is of paramount importance since the market, especially an emerging market like Turkey, does not usually present a yield for every *n*-year of maturity particularly for the long-end of the curve. Therefore, I used Nelson-Siegel-Svensson method, which is the best option to generate the forward rates for the scope of the estimation horizon. After acquiring dependent factors, I used the following machine learning techniques to predict 3, 5 and 10-year excess bond returns: Principal component analysis, Ordinary least squares, Penalized linear regressions, and neural networks. While doing so, I followed some previous work for calculating zero-coupon yields and forward rates, and for machine learning techniques decisions. Previous works regarding term structure estimation, excess bond return estimation and use of machine learning methods for bond markets are explained in detail in Section 2.

CHAPTER II

LITERATURE REVIEW

Calculating the zero-coupon yields of a bond market is vital in many aspects. Zero-coupon and forward rates together are the rudimentary part of predicting future yields and returns. Bolder and Streliski's [1] work is a decent example of forward rate and zero-coupon bond yield database. They create historical data of zero-coupon yields and forward rates for Canada treasury bond market. Not as historical as theirs, since Turkey treasury bond market not going back that much in time, we also created zero-coupon yields and forward rates for Turkish Government Eurobond market between the years 2005 and 2020. The secondary aim of Bolder and Streliski [1] is to apply the data on various models and examine the behavior of the models, especially Nelson-Siegel and Svensson method. They also compare their results with the model the Bank of Canada had been using. The ultimate objective is to choose the best performing yield curve model, but they have two main problems, one of which is the optimization of parameters. I face the same issue as well in my work especially using the neural network models. The second problem is picking appropriate data. I also share the data problem because the lack of extensiveness of the bond market in an emerging market like Turkey might cause some scarcity in terms of number of different maturity bonds. This drawback is investigated later in the paper.

Similar to Bolder and Streliski [1], Diebold and Li [2] also use Nelson-Siegel model to predict the yield curve. They disregard the arbitrage-free and equilibrium approaches. What they do differently is their intention to focus on the Nelson-Siegel

model by using autoregressive models. They interpret their three-dimensional parameter as factors corresponding to level, slope and curvature. They use autoregressive models for predicting the factors, and then using the predicting factors they are able to estimate the yield curve. They work their model for both short and long-end of the curve separately and come up with better results in the long-end. My work shows similarities in this aspect, predicting relatively better out-of-sample R^2 values with long-end excess returns.

Svensson [3] also works on Nelson-Siegel model, but unlike aforementioned papers, he uses it to forecast monetary policy rate. His main motivation roots to the radical change of foreign exchange policy to floating state, which make monetary policy expectations even more crucial. Forward interest rates are among the most significant factors to forecast monetary policy rate, and it has been in use for a long time not only forecast the monetary policy rate but also for expectation of future inflation rates, short-term interest rates as well as pricing financial instruments and arbitrage possibilities. The difficult part is the lack of medium and long maturity zero-coupon bonds in the treasury markets. Calculating forward interest rates is a two-step process and will be explained in Section 3.1.

There are many studies focusing on treasury yields, especially trying to forecast excess returns of relatively long duration bonds as well as trying to figure out whether forward or spot yields could help forecasting future rates. Culbertson [4], Shiller *et al*. [5] and Fama [6] are among the ones focusing on this but not being able to have satisfying enough results. One of the reasons is most searched for a single factor -or a few at best- to explain the returns. In this sense, Cox *et al*. [7] tell the relevance between the returns of discount instruments and forward rates by insinuating the excess return of different maturity bonds are linear functions of forward rates. Their regressions do not favor for a single factor but find at least several latent factors to explain the variation in treasury bills

instead.

Fama and Bliss [8] improve this study by confirming prices of two-to-five-year bonds, by calculating forward rates first, can significantly explain yearly changes of spot rates of these bonds. They manage to have an R^2 of 18% by extending the forecast horizon to five years. In short, they find that one can predict the excess return of five year-bond by using the spread between the five-year forward rate and one-year spot rate.

In addition to Fama and Bliss [8], Ilmanen [9] also studies to predict excess bond returns in long-maturity of the curve. He chooses to focus on six countries and achieves to explain the variations of the excess bond returns at the long end of the curve by 4 to 12 percent. He investigates how correlated the excess bond returns among different countries. He finds out that they are highly correlated, and excess bond returns are explicable with the help of one global risk factor -determined by average bond returns- and constant conditional betas.

Cochraine and Piazessi (2005) examine this by regressing one-year excess return of a long duration bond by using five different forward rates. They calculate the excess return by subtracting one-year borrowing rate from the return they gain by the long duration bond after holding it for one year and then selling it. The one advantage of calculating the risk premia through excess returns is eliminating the effect of inflation as well as interest rates, thereby attaining real risk premia. In addition to obtaining R^2 values around 44%, they have another significant observation is getting similar pattern for regression coefficients even when performing different maturities.

From another aspect, Cochrane and Piazessi [10] improve Fama and Bliss's [8] work by predicting better excess returns with same linear combination of forward rates for different maturities. In addition to achieving two-and-a-half times better R^2 value, they manage to advance previous works, which are able to explain different maturity

returns with various forward rate combinations, by obtaining smaller *p*-values.

Their work also objects the expectation hypothesis, which advocates predicting the future short-term interest rates by the level of spot long-term rates, meaning one must have the same return by holding one-year bond by consecutive *n* years versus holding *n*-year bond for the same period. For example, while predicting the excess return of three-year bond, Cochrane and Piazessi [10] use five-year forward rate. I do not use this method. I predict the excess returns of three, five and ten-year bonds, and I use forward rates of one-to-*n* year bonds on the left-hand side of the regression.

Machine learning techniques are relatively new in finance and economics; therefore, those techniques are less used in predicting bond returns as well as yield curve itself. Sambasivan and Das [11] make use of machine learning to forecast the US treasury yield curve. Their primary focus is to use Gaussian Processes together with a dynamic modeling strategy like Kalman Filter. They continuously update the hyper-parameters as long as the algorithm receives novel yield curve data, which is a time-series frequency of one day. My work has some similarities with Sambasivan and Das [11] in terms of data frequency and process of updating the model. I also use daily frequency of forward rates, and continuously update the model by expanding the new data the algorithm receives. However, the methods they use differ from mine. Details will be enlarged upon later.

One of the main findings of Sambasivan and Das [11] is about the accuracy of the forecasts of various models they use on different maturity of the term structure. For example, they find out that multivariate time series method succeeds better for predicting the short-term of the yield curve, whereas Gaussian processes explain well in medium and long-end of the curve. One potential drawback of this study might be the data range because the study uses a span of only ten years of US treasury yields. Although studying an emerging market instead of a developed market, I take into account a span of sixteen

years of Turkey Eurobond yields.

Recent improvements in statistics and computer science have an impact on finance in a sense that economists' choices of models for forecasting and prediction. The prevailing methods are limited to data compression techniques in the last decades, however, with the recent improvements new model preferences and dimensionality reduction techniques begin playing more important role. That these methods not being comfortable with structural analysis and parameter interferences might be the reason for the limitation. Indeed, machine learning prioritizes out-of-sample prediction by using a large set of sample data. This makes machine learning techniques convenient for predicting asset returns and calculating risk premium.

In this sense, my work gained inspiration from Bianchi et al. [12], whose main focus is to predict excess US treasury bond returns by using various machine learning techniques. They have two main approaches, one of which is forecasting excess returns by using only forward rates and this is similar to Cochrane and Piazessi [10] method. The second approach is using macroeconomic data in addition to yield curve data, which is similar to Ludvigson and Ng [13]. My work has the same two approaches. First, I use only forward rates obtained from the spot yield curve data, then I conduct the same steps adding macroeconomic data to forward rates. The details are explained in Section 4. I use linear regressions, dimensionality reduction techniques, penalized linear regressions, and neural networks. Linear regressions include linear model and partial least squares; dimensionality reduction techniques include principal component analysis; penalized linear regressions include Ridge, Lasso and ten different elastic net models; and for neural networks we used one, two and three layers with different number of nodes (usually 3,5 and 7). In addition to this, Bianchi et al. [12] use more diversified techniques, including boosted regression trees, extremely randomized regression trees and random forests. But

methodically, both Bianchi et al. [12] and my work show parallelism with Gu, Kelly and Xiu's (2020) structure, meaning different machine learning techniques are compared according to their out-of-sample performances.

Bianchi et al. [12] have three different aspects of their service to asset return literature, one of which is the out-of-sample performance of extreme trees and neural networks advances other techniques for both forward rates only and macroeconomics data included approaches and have significantly larger R^2 values than Cochrane and Piazessi [10] which use single linear combination of forward rates and Ludvigson and Ng [13] which use yields and macro data together. Second, using macroeconomics data in addition to forward rates improves the results, for example in neural network methods, Bianchi et al. [12] obtain almost 10% larger R^2 value with macro data. Third, deep neural network has decent results for three factors of the yield curve structure. Only forward rates approach forecasts the level better, taking macro data into account improves it and it begins forecasting both the level and slope better. This makes sense since the slope has more things to do with macroeconomic situation. Moreover, their work supports risk premium modelling because they show that neural network forecasts are not cyclical, but more in line with macroeconomic uncertainty and risk averseness.

Plus, about the macroeconomic data use in forecasting, Joslin, Prieblisch, Singleton (2014) advocate that macro data which are not spanned by bond yields produce better results, whereas Bauer and Rudebusch (2017) tell otherwise. Bianchi et al. [12] findings also support this discussion. Bianchi et al. [12] says un-spanned theory is backed even though they consider the potential non-linearity in interest rates. Considering the non-linearity is also vital for predicting excess bond returns. They also find a heterogeneity among macro data; for example, orders, inventory, and output data play more significant role for the long-end of the curve whereas stock and labor related data

for the short-end of the curve. In short, they say macroeconomics data not spanned by bond yields may vary among the term structure.

Ludvigson and Ng [13] dig into the use of macroeconomic data for predicting bond risk premia. Their main questions are whether cyclical macroeconomic data have first-hand impact on excess bond returns and whether macroeconomic data carry any new information about treasury market, meaning the information is not already priced in bond yields. They claim most economic theories suggest that risk acceptance would pay off especially in recession periods, and indeed Cochrane and Campbell [15] assert that market risk premia increase with contracting or quietly expanding periods.

However, Ludvigson and Ng [13] tell it is difficult to find a direct relation between macroeconomic activity and bond market risk premia. They say such macro data as gross domestic product or inflation contribute little to forecast bond excess returns, whereas forward rates and yield spreads are more powerful factors for predicting excess returns. They say one reason to this must be that most macroeconomic data are less frequently and impeccably observed. Plus, some macro data variables might be latent, and it is more difficult for macro data to be measured in line with theoretically accepted concepts. However, they try to overcome these drawbacks by a method of dynamic factor analysis. Considering there may be more than hundreds of macroeconomic data to work with, one can presume that finding direct relation between a myriad of macro data and excess bond returns would be complicated. Dynamic factor analysis helps summarizing the large data in macroeconomic variables into fewer comprehensive factors. Ludvigson and Ng [13] come up with that using dynamic factor analysis improves model and macro data is able to forecast excess bond returns. Having the best prediction result for two-year bond return, Ludvigson and Ng [13] find out that macroeconomic variables can explain around 25% the variation in two-to-five-year bond excess returns.

Looking into results of Bianchi et al. [12] presents three outcomes. First, extreme trees and neural networks give the best out-of-sample results among all methods, for both yields only and macroeconomics data approaches. Second, architecture of the network is vital while performing with macro data; namely, while including macroeconomics data, increasing the layer of neural networks also increases out-of-sample results. Deeper networks usually work better than more shallow ones. However, the depth solely does not explain the performance. Bianchi et al. [12] make different sub-groups from the large macro data and perform shallow networks to each sub-group. They find out that some sub-groups' shallow network performs similar to deep neural network model with yields only. Grouping macro data and the depth of the network is novel in the literature. Third, the aforementioned group ensembling technique is vital for deciding what types of non-linearity are significant, whether interaction of many variables or higher polynomial within category. Bianchi et al. [12] come up with non-linearity within group is more relevant to performance, since their work does not include interaction between groups.

CHAPTER III

METHODOLOGY

In this section, I go over the data preparation process, calculations, and reasoning behind the methods; then, elaborate on machine learning techniques utilized in predictions.

I work on a similar way with Bianchi et al. [12]. I choose two main approaches to predict excess bond returns. First, I predict excess bond returns with using only yield data on Turkey Eurobond market. This method follows what Cochrane and Piazessi [10] propose. Bianchi et al. [12] use the dataset on US Treasury yields provided by Liu and Wu (2019). This way they are able to extend the dataset beyond that of Fama and Bliss [8], in other words, they work including bonds with maturities up to ten years whereas Fama and Bliss [8] do so up to five. I also choose to work with one-to-ten-year bonds, however, I have to prepare the dataset from scratch since there is no previous work on Turkish Eurobond market data. Therefore, I first calculate zero-coupon yields of Turkish Eurobonds with a daily frequency. Then, I calculate forward rates using Nelson-Siegel-Svensson method for ten different maturities for the daily data. Please see Section 3.1 for details.

3.1 Construction of the Yield Curve

Implied forward rates can easily be calculated from zero-coupon bond yields, however, bonds with longer maturity than one year are usually coupon bonds, which makes calculation of forward interest rates more intricate. Coupon bonds prevailingly

have coupon payments every six months until maturity and these coupon payments can be considered as single zero-coupon bonds each. We can calculate the yield to maturity of a coupon bond as a weighted average of every coupon payment plus the nominal payment at the maturity. This significantly differs coupon bonds from zero-coupon bonds with the same maturity. Since bond markets usually do not have zero-coupon bonds longer than maturity of one year, calculating forward interest rates becomes a two-step process. First, implied spot rates must be calculated from coupon bond especially for medium and long terms, then implied forward rates must be calculated using implied forward rates.

Let us start with calculating the price of a zero-coupon bond. Let $\boldsymbol{s(t, M)}$ be the spot interest yield of a zero-coupon bond, where $\boldsymbol{t}$ is trading date and $\boldsymbol{M}$ is the maturity of the bond, $\boldsymbol{M > t}$. Also, let $\boldsymbol{m}$ be the time to maturity of the bond, $\boldsymbol{m = M - t}$. The spot yield curve is derived from $\boldsymbol{s(t, M)}$ and $\boldsymbol{m}$ of every bond in the market with different times of maturity. A discount function is needed to price the zero-coupon bond and can be denoted by $\boldsymbol{d(t, M)}$, where *exp* is the exponential function.

$$d(t, M) = \exp\left(-\frac{s(t, M)}{100}(M - t)\right) \qquad (1)$$

For a coupon bond, there are coupon payments $\boldsymbol{C}$ every $\boldsymbol{k}$ period of time, usually six months, until $\boldsymbol{M}$, maturity date, where the bond also pays a face value of 100. Assuming $\boldsymbol{C}$ is yearly coupon payment, keep in mind that if coupon payment period is six months, $\boldsymbol{C/2}$ will be paid in every $\boldsymbol{k}$. But, for the sake of simplicity, let us assume $\boldsymbol{C}$ is paid every $\boldsymbol{k}$ period, where $\boldsymbol{k = 1, 2, \ldots, m}$ representing every coupon payment of the bond. The price of a coupon bond therefore is calculated by adding up today's price of every coupon payment and the face value. Price of coupon payments can be denoted as

$\boldsymbol{d(t, t+k)}$, and price of the face value as $\boldsymbol{100d(t, t+m)}$. The formula representing the price of a coupon bond at trade date is as follows:

$$P(t, M) = \sum_{k=1}^{m} c\, d(t, t+k) + 100 d(t, t+m) \qquad (2)$$

Taking into consideration the internal rate of return, $\boldsymbol{y(t, M)}$, and exponential function for the price calculation, the price of a coupon bond can be rewritten as follows:

$$P(t, t+m) = \sum_{k=1}^{m} c \exp\left(-\frac{y(t, t+m)}{100} k\right) + 100 \exp\left(-\frac{y(t, t+m)}{100} m\right) \qquad (3)$$

Calculating the price **of** coupon bonds with (2) corresponds to spot rates, namely the rate of zero-coupon bonds which is calculated in (1).

On the other hand, instead of zero rate, calculating the price with (3) uses internal rate of return, which can be seen as a complicated average of spot yields. Plus, the coupon rate of bonds makes difference in a way that coupon bonds with the same maturity can have different yield to maturity depending on their coupon rates. Also, yield to maturity corresponds to the yearly interest rate an investor earns if the investor holds the bond until maturity including the coupon and nominal payments. Unlike yield to maturity, spot rate shows the rate of return an investor earns by buying and selling the bond on secondary market, neither having to hold the bond until maturity nor to receive coupon payments. Therefore, using spot rate instead of yield to maturity is crucial for estimating the term structure or excess bond returns.

Calculating forward rates using spot rates is easier. Let us assume, at trade date $\boldsymbol{t}$, one sells a zero-coupon bond maturing at forward contract's start date, $\boldsymbol{t'}$, and buys a zero-coupon bond with a maturity same as forward contract's, $\boldsymbol{T}$. The return is calculated as the implied forward rate. For the sake of the calculation, let us assume $\boldsymbol{T > t' > t}$. The

relation is as follows:

$$f(t,t',T) = \frac{(T-t)s(t,T) - (t'-t)s(t,t')}{T-t'} \tag{4}$$

In this study, I use implied forward rates for $\boldsymbol{t = 1, 2, \ldots, 10}$. Since there are not always zero yields with maturities corresponding to whole numbers of years, I use Nelson-Siegel-Svensson's forward rate function to calculate forward rates after obtaining spot yields. The formula follows:

$$\begin{aligned} f(m,b) = {} & \beta 0 + \beta 1 \exp\left(-\frac{m}{r1}\right) + \beta 2 \frac{m}{r1} \exp\left(-\frac{m}{r1}\right) \\ & + \beta 3 \frac{m}{r2} \exp\left(-\frac{m}{r2}\right) \end{aligned} \tag{5}$$

Where $\boldsymbol{b = \beta 0, \beta 1, \beta 2, \beta 3, r1, r2}$ is a vector and recall that $\boldsymbol{m = T - t}$ denotes time to maturity. Please also recall that there are four components in Nelson-Siegel-Svensson's forward rate function. The constant $\boldsymbol{\beta 0}$, the exponential term $\boldsymbol{\beta 1 \exp\left(-\frac{m}{r1}\right)}$ monotonically increasing or decreasing towards zero depending on the sign of $\boldsymbol{\beta 1}$, the component $\boldsymbol{\beta 2 \frac{m}{r1} \exp\left(-\frac{m}{r1}\right)}$ generating a hump-shape or u-shape depending on the sign of $\boldsymbol{\beta 2}$, and the fourth term $\boldsymbol{\beta 3 \frac{m}{r2} \exp\left(-\frac{m}{r2}\right)}$ a second hump shape since $\boldsymbol{\beta 3}$ is always positive.

After calculating forward rates, I follow Campbell and Shiller (1991) for determining bond returns. Suppose a zero-coupon bond pays a dollar with a maturity $\boldsymbol{t + m}$, and its logarithmic price is denoted by $\boldsymbol{p_t^{(m)}}$. The compounded yield of the zero-coupon bond is also calculated via the following equation: $\boldsymbol{y_t^{(m)} = -\frac{1}{m} p_t^{(m)}}$. The subscript and the superscript correspond to trading time and the maturity of the bond, respectively. The prevailing formula for the logarithmic excess return of $\boldsymbol{m}$-year bond for

holding it for one year is shown by $\boldsymbol{r_{t+1}^{(m)} = p_{t+1}^{(m-1)} - p_t^{(m)} - y_t^{(1)}}$. If I were to elaborate on this formula, the logarithmic excess return of $\boldsymbol{m}$-year bond after holding it for one year time is calculated by subtracting both the price of $\boldsymbol{m}$ -year bond at trading time and the yield of one-year maturity from the price of $\boldsymbol{m-1}$ year bond after the holding period. A clearer formula for the excess return of a $\boldsymbol{m}$-year bond is as follows:

$$r_{t+1}^{(m)} = -(m-1)\left(y_{t+1}^{(m-1)} - y_t^{(m)}\right) + \left(y_t^{(m)} - y_t^{(1)}\right) \quad (6)$$

Bianchi et al. [12] use the same formula for their excess return forecasts. As mentioned above, the prevailing calculation is conducted for holding period of one year. However, my data does not go back much in time, in fact, it comprises sixteen years from 2005 to 2020, including. Therefore, I mostly had to estimate monthly returns. Assume I try to estimate excess return of a ten-year Eurobond, according to the formula I need a zero-yield of nine years and eleven months maturity. This would be very difficult and time-consuming, plus, it would not be very efficient to use McCuloch's cubic spline or any kind of interpolation since it has a disadvantage of being unstable especially at the longest maturities. Hereby, I modified (6) into following:

$$r_{t+1}^{(m)} = -(m)\left(y_{t+1}^{(m)} - y_t^{(m)}\right) + \left(y_t^{(m)} - y_t^{(1)}\right) \quad (7)$$

That Turkish Eurobond market database is not extensive enough forces me to have a trade-off between accuracies of return formula and forward rate calculations.

3.2 Forecasting Methods

3.2.1. Principal Component Regression

Principal component regression is a liner regression technique that involves

principal component analysis. Principal component regression is especially used to overcome the problem with collinearity in linear regression by combining explanatory variables to a smaller set of uncorrelated variables. There might be multiple factors with very high correlation, which would decrease the efficiency of the model. Solely eliminating one of the correlated variables is not a good solution to collinearity problem, since that probably means a loss of information for predicting the dependent variable. PCA is a method to find the linear combination that accounts for as much variability as possible. Therefore, instead of eliminating likely independent factors, combining them is a better solution. One major problem with principal component regression is that the explanatory variables have been combined in a way that has little to do with the dependent variable. Weights of the variables can be optimized to make the first principal component have the maximum variance. So, one or more variables can be unnecessarily given higher weights and those variables contributes most to the variance of the first principal component. However, this might not necessarily mean that those variables are among the best in explaining the variance of the dependent variable.

3.2.2. Partial Least Square Regression

Partial Least Square regression can be seen as an extension of the principal component regression. Similar to principal component regression, the method of partial least squares, also called projection to latent structures, is a technique that reduces the number of explanatory variables to a smaller set of uncorrelated variables. Both methods are used to overcome the problem of collinearity in linear regression and in cases where there are more explanatory variables than observations. Recall the problem with PCA is that it combines the variables without involving the dependent variable. It would make more sense to find weights that also maximize the prediction of the dependent variable.

3.2.3. Penalized Linear Regressions

Ordinary Least Squares method is improved by Penalized Linear Regression. The least squares method might create a model that overfits the training data and has high variance with the test data. The main idea behind Penalized linear regression is to find a new model that does not fit the training data as well. In other words, some bias term is introduced to the model. In return for the bias term, there will be a significant drop in variance. By starting with a slightly worse fit, penalized linear regression can provide better long-term predictions. Least squares regression determines values for the parameters in a model and it minimizes the sum of the squared residuals. In addition to minimizing residuals, penalized regression also minimizes a bias term which can be summarized as (λ * slope). Slope depends on the number of factors in the model so for example, in my prediction for ten-year bond excess return with yields-only approach, lambda is multiplied by ten different slopes of the ten different forward rate factors. But in short, slope adds the penalty to the model, whereas lambda decides how severe the penalty will be. Also, as lambda increases the slope decreases and gets closer to zero. So, the larger lambda gets, prediction for the model output becomes less sensitive to factors.

The most popular penalized linear regressions are Ridge and Lasso regressions. They are very similar but have very important minor differences at the same time. For the penalty term, Ridge regression calculates the minimized residual errors by multiplying lambda with the sum of squares of slopes of the factors, whereas Lasso regression does the same by multiplying lambda with sum of absolute values of the slopes. Plus, another difference between Ridge and Lasso regression is that Ridge regression can only shrink the slope asymptotically close to zero, while Lasso regression can shrink the slope completely to zero. This feature of Lasso regression makes it able to remove unnecessary

factors from the model and work with less but more useful estimators. This way, Lasso regression might be better than Ridge regression at reducing the variance in models that contain a lot of useless variables. In contrast, Ridge regression tends to work better when most variables are useful.

In cases where there is a myriad of factors, it is not always clear to choose between Ridge and Lasso since we are not able to tell whether most factors will be useful. Lasso tends to minimize the effect of relatively less useful parameters and Ridge tends to shrink the effect of all factors. Therefore, it makes more sense to use Elastic-net regression which combines Lasso and Ridge regressions. While adding the penalty terms, Elastic-net regression gives different weights to lambda values of both Ridge and Lasso penalty terms and combines them in one model.

3.2.4. Neural Networks

Neural networks form the base of deep learning which is a sub-field of machine learning. In short, neural networks take in data, train themselves to recognize the patterns in this data and then predict the outputs for a new set of similar data. Neural networks have the input layer where the data is introduced to the network, and the output layer where the network presents its predictions, and the hidden layers in between where the deep learning materializes. Figure 1 depicts an example of a 2-layer neural network with an input layer with ten inputs, two hidden layers with seven nodes each, and an output layer. The following summarizes the working principle of neural networks. For example, in my work, forward rates are received in the input layer, then every forward rate is connected to the first hidden layer, which are assigned with various weights. The inputs are multiplied with weights and added with biases the nodes have. This value is passed to a threshold function called the activation function. The result of the function determines

how the particular node is significant, and passes to the next hidden layer as inputs, accordingly. Similar process takes place in the second layer and the output is determined this way. The network also compares the real result with its prediction, and computes how much it deviates. Then it feeds the error back to the network and modifies the weights of the nodes. This is called backpropagation. Neural networks iteratively performs forward and backpropagation to determine the optimum weights of the nodes.

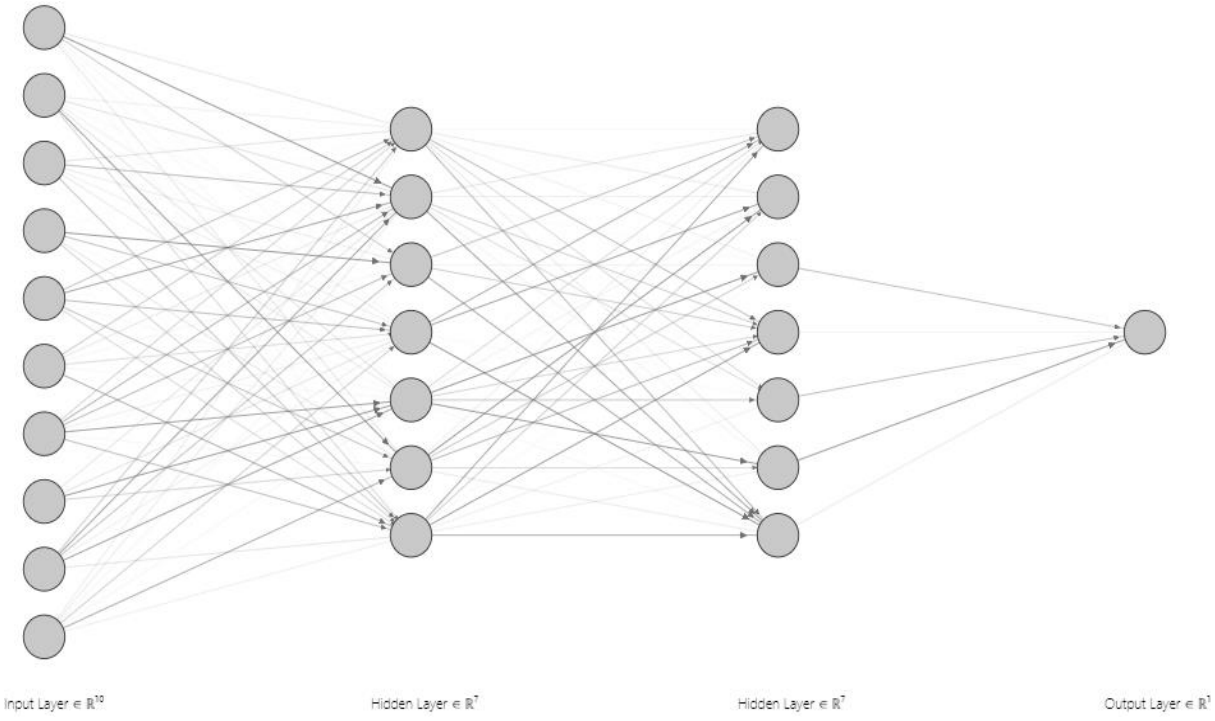

Figure 1: Example of a 2-layer neural network architecture with forward rates only

CHAPTER IV

DATA

The main predictors for machine learning techniques are forward rates of Turkish Eurobond market. Since I did not have forward rates in hand, I must calculate them using zero-coupon yields. To calculate zero-coupon yields, I need all Turkish Eurobonds yields, except the callable ones. Therefore, my first choice of data source is Bloomberg Terminal to import raw yields data. For my second way to predict excess bond returns, I use macro data in addition to forward rates. I search for macro data that Bianchi et al. [12] use in their work; however, macro data related to Turkey is not that frequently and historically recorded as the US data. Therefore, I had to eliminate some macro data that Bianchi et al. [12] use. The ones I include are Turkey Industrial Production, Employment Rate, Labor Force, Hours Worked Manufacturing, Industrial Production of Durable Goods, Monetary bases 1, 2 and 3, Gross Reserves, Net Reserves, Stock exchange index return, Fed Funds Rate, US Generic two-yield, US Generic five-yield, US Generic ten-yield, US Corporate High Yield Spreads, Bbb rated US Corporate Spreads, Real effective exchange rate, Consumer price index, Producer Price index. We also used Central Bank of the Republic of Turkey database for CPI and PPI.

Table 1: Summary of Statistics

		Mean	St. Dev.	Minimum	Maximum
Forward Rates	1-year	-15.88%	43.10%	3.38%	1.93%
	2-year	-3.23%	12.07%	3.87%	1.65%
	3-year	1.60%	13.02%	4.42%	1.55%
	4-year	1.99%	13.43%	4.81%	1.49%
	5-year	2.23%	13.51%	5.11%	1.44%
	6-year	2.45%	13.46%	5.34%	1.40%
	7-year	2.63%	13.37%	5.53%	1.36%
	8-year	2.79%	13.28%	5.68%	1.32%
	9-year	2.97%	13.20%	5.81%	1.30%
	10-year	3.15%	13.13%	5.92%	1.28%
Macroeconomic variables	Industrial Production	-31.40%	35.67%	1.05%	10.23%
	Employment	-4.69%	2.74%	0.04%	0.89%
	Labor Force	-3.67%	3.05%	0.21%	0.77%
	Hours Worked Manuf.	-31.39%	40.70%	0.27%	4.33%
	Ind.Prod Dur.Goods	-41.83%	61.60%	1.47%	13.84%
	M1	-10.48%	17.50%	1.72%	4.25%
	M2	-4.71%	12.04%	1.55%	2.08%
	M3	-4.64%	11.59%	1.51%	2.06%
	Gross Reserves	-17.82%	10.28%	0.19%	4.29%
	Net Reserves	-11.18%	10.10%	0.47%	2.77%
	Bist 100	-23.12%	22.85%	1.11%	7.61%
	Fed Funds Rate	-0.96%	0.25%	-0.01%	0.16%
	US 2Y	-0.95%	0.67%	-0.02%	0.20%
	US 5Y	-0.91%	0.68%	-0.02%	0.23%
	US 10Y	-1.03%	0.64%	-0.02%	0.24%
	US Corp. HY Spread	-3.45%	4.63%	0.01%	0.86%
	BBB Corp. Spread	-0.72%	1.47%	0.00%	0.22%
	ReeR	-21.00%	13.99%	-0.21%	3.78%
	CPI	-1.44%	6.30%	0.78%	0.89%
	PPI	-3.54%	10.88%	0.84%	1.47%

My work includes linear models, dimensionality reduction techniques, penalized linear models and neural networks as machine learning techniques to predict excess bond returns. The main resulting parameter I follow as the model's success is the out-of-sample R^2. Simply, the idea behind it is comparing real excess returns with the returns machine learning methods predict. As Campbell and Thompson (2007) suggest, I use the following formula for calculating the out-of- sample R^2:

$$R^2_{OOS} = 1 - \frac{\sum_{t_0=0}^{M-1}(r_{t+1}^{(m)} - \hat{r}_{t+1}^{(m)}(Md))^2}{\sum_{t_0=0}^{M-1}(r_{t+1}^{(m)} - \bar{r}_{t+1}^{(m)}(Md))^2} \quad (8)$$

where $\hat{r}_{t+1}^{(m)}(M)$ and $\bar{r}_{t+1}^{(m)}(Md)$ denote the forecast of the excess bond returns for maturity $\boldsymbol{M}$ using model $\boldsymbol{Md}$ and the prediction error obtained based on the historical mean, respectively.

I would like to elaborate on how I collect test data results. Recall that yield data consists of 2005 to 2020, including. As mentioned previously, this range of data might be considered relatively limited for having strong predicting results. Therefore, I conduct two different ways when utilizing yields data to overcome this obstacle. First, I predict monthly returns instead of yearly returns to better exploit the limited available data. I also use two different frequencies of forward rates data when predicting monthly returns. I had daily frequency of Eurobond yields data at the beginning. Therefore, I was able to use Nelson-Siegel-Svensson method to calculate zero yields on a daily frequency in time series. Having calculated zero yields on a daily frequency at the beginning of the work allow me to have both daily and monthly forward rates data to work with. In other words, the second method that helped me to overcome limited data drawback is using daily frequency of forward rates as model factors. One can assume that there are twenty

business days in a month on average, so I predicted excess bond returns of twenty days later when using daily data. In short, both being able to use daily data and to predict monthly instead of yearly returns give me a relatively larger database than I already have. On the other hand, using daily frequency data is a negative factor for our prediction results, because it involves much more noisy data than monthly frequency data.

Turkish Eurobond market had around fifteen different maturity bonds in 2005, the period the data begins. For couple of years the number of bonds stays around fifteen. After the year of 2010, with new issuances the number increases to 18 bonds and then between 2012-2014 the number of different maturity bonds goes beyond 20. In the last years of time period of my scope, the number of bonds reaches beyond 25. Obviously increasing number of different maturity bonds works better for interpolating zero-coupon yield curve. Therefore, especially in the early years of the database, I probably have relatively less strong results for zero-coupon yields, thereby producing questionable forward rates data, which are the main factors for our predictions. Considering that the majority of Eurobonds have relatively short-term maturities, I could assert that zero-coupon yields with short maturities are robust. However, especially for the earlier years of the database, the longer zero-coupon yields might be more questionable.

I would like to elaborate on the model design in terms of deciding on training, validation, and test data. For all techniques, I take the first three years of the forward rates data as the training (+validation) data, as default. For validation data, I use the last 20% of the training data. For the test data, I use the next month starting from the end of the validation data, namely the first three years. From this point, I use similar technique as Bianchi et al. [12] do, meaning that I expand the test and validation data step-by-step. The magnitude of each step is a month long. For the daily data, it is twenty days of forward

rates, and for the monthly data it is just another row of the next month's forward rates. Increasing the training data range automatically pushes the test data one month further. In each step, the model takes training data and produces an output model. Then the produced model takes validation data as input and produces another model. This model takes the test data and produces the prediction. This process is repeated for every month until the data reaches the end of the year 2020. The predictions for all months are recorded separately, and the list of those predictions with the real returns at the corresponding months are used to calculate the R^2_{OOS}.

The validation data cannot be used for the Principal component analysis, Linear models, and Partial least squares by definition. For these techniques, I only used training and test data, again with expending the data set month-by-month. Penalized linear regressions and Neural networks are able to utilize the validation data, thereby working more extensiveley and accurately.

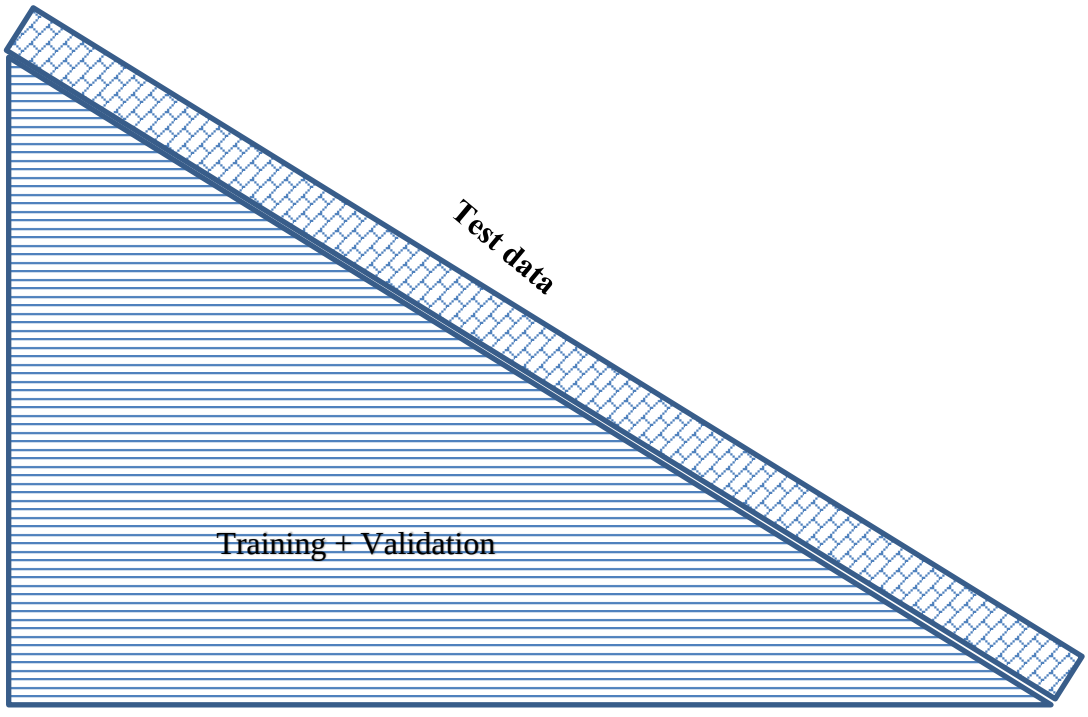

Figure 2: Data splitting

I focus on excess bond returns of only three different maturities, 3-year, 5-year and 10-year bonds. I also use different models for each maturity. For example, I use ten, five and three factors for estimating 10-year bond return. Ten factors include all forward rates from one year to nine years. Five factors include one, three, five, seven and nine-year forward rates. Three factors include one, five and nine-year forward rates. This is the yields-only approach. For the macro variables approach, I conduct the same models including twenty macro data to the corresponding yields data.

CHAPTER V

PERFORMANCE EVALUATION OF ALTERNATIVE MODELS

I categorize the models in four different headings: Dimensionality Reduction techniques, Linear models, Penalized Linear regressions, and Neural Networks. Also, I predict bond excess returns in three different ways: Monthly excess returns using daily frequency data, monthly excess returns using monthly frequency data, and yearly excess returns using monthly frequency data. Therefore, I prepare the results according to these categorizations.

Table 2 shows the results for linear models and principal component analysis for both monthly and yearly returns predictions, including both daily and monthly frequency data for monthly predictions. With the daily frequency data, linear models produced R^2_{OOS} around -40% for 10-year return, -50% for 5-year return, and -24% for 3-year return. PCA yields similar results, only improving 3-year return by 3%. Using monthly frequency data instead of daily improves 10-year returns by 3% average to around -38% R^2_{OOS}. On the other hand, 5-year return is not improved whereas 3-year return is worse of by monthly data.

Plus, yearly return predictions of linear models significantly improves the R^2_{OOS} for 10-year return to around -7% for LM and -5% for PLS. 5-year return is also significantly improves from around -50% to -25% for both LM and PLS. However, R^2_{OOS} for 3-year return deteriorates.

Table 2: R^2_{OOS} for PCA, LM, PLS (Yields only)

Models	**R^2_{OOS} – Yeilds Only**								
	Daily Frequency Monthly return			Monthly Frequency Monthly return			Monthly Frequency Yearly return		
	$r^{(10)}_{t+1}$	$r^{(5)}_{t+1}$	$r^{(3)}_{t+1}$	$r^{(10)}_{t+1}$	$r^{(5)}_{t+1}$	$r^{(3)}_{t+1}$	$r^{(10)}_{t+1}$	$r^{(5)}_{t+1}$	$r^{(3)}_{t+1}$
A. Dimensionality Reduction									
PCA (10 factors)	-41.2%								
PCA (5 factors)	-41.3%	-49.3%							
PCA (3 factors)	-41.5%	-49.8%	-21.2%						
B. Linear models									
LM (10 factors)	-40.5%			-40.5%			-7.02%		
LM (5 factors)	-42.4%	-50.7%		-40.7%	-52.8%		-9.57%	-26.9%	
LM (3 factors)	-42.3%	-50.9%	-24.7%	-39.7%	-52.9%	-39.7%	-7.19%	-26.9%	-45.1%
PLS (10 factors)	-41.2%			-38.5%			-6.01%		
PLS (5 factors)	-40.9%	-50.0%		-38.3%	-51.1%		-5.37%	-25.2%	
PLS (3 factors)	-40.6%	-49.6%	-24.5%	-38.0%	-51.1%	-38.8%	-4.39%	-24.8%	-44.3%

For penalized linear regressions, Ridge, Lasso and nine different Elastic regressions with corresponding alpha values are used. Table 3 depicts the results for both monthly and yearly returns predictions, including both daily and monthly frequency data for monthly predictions. Penalized linear regressions produce various outcomes. Forecasting monthly returns using daily data, the model does better for shorter maturity returns as opposed to Bianchi *et al*. [12] results. R^2_{OOS} values for 10-year return are around 2-3%, whereas R^2_{OOS} values for 5-year and 3-year returns are around 4-5% and 10-11%, respectively. The lesser the left-hand side factors yielded better predictions.

However, forecasting monthly returns using monthly data decreases R^2_{OOS} values for all maturities. Penalizing the model works better with more frequency data. Here, I would like to elaborate on the process. Section 3.3 explains the training, validation and test data splitting. Penalized regression uses excatly as told in that section. Using daily freqeuncy data, the model extends month-by-month, meaning twenty days-by-twenty days. Using twenty different data in each step, the model therefore predicts twenty different returns for each step. Instead of using predictions for everyday, the model records every twentieth prediction sperately. On the other hand, in monthly frequency data, the model uses one data for one month to predict the monthly return. More data creates noise and irrelevance is not valid for penalized linear regression results. This might be the reason for better R^2_{OOS} values with daily frequency.

I also observe that forecasting yearly returns produce much better results for 10-year excess bond returns, and slightly better results for 5-year excess bond returns, whereas similar results for 3-year excess bond returns. Plus, specific to forecasting yearly

returns of 10-year bond, I observe that using five independent parameters instead of ten (or three) yields slightly better R^2_{OOS} values.

Table 3: R^2_{OOS} for Ridge, Lasso and Elasitc regressions (Yields only)

Models		**R^2_{OOS} – Yeilds Only**								
		Daily Frequency Monthly return			Monthly Frequency Monthly return			Monthly Frequency Yearly return		
	#factors	$r^{(10)}_{t+1}$	$r^{(5)}_{t+1}$	$r^{(3)}_{t+1}$	$r^{(10)}_{t+1}$	$r^{(5)}_{t+1}$	$r^{(3)}_{t+1}$	$r^{(10)}_{t+1}$	$r^{(5)}_{t+1}$	$r^{(3)}_{t+1}$
C. **Penalized Linear models**										
Ridge (α = 0)	10	1.13%			0.07%			9.01%		
Ridge (α = 0)	5	3.82%	5.68%		0.39%	1.47%		9.65%	2.73%	
Ridge (α = 0)	3	4.40%	0.94%	0.88%	1.07%	2.46%	7.48%	7.77%	1.68%	1.14%
Elastic (α = 0.1)	10	2.26%			0.06%			8.99%		
Elastic (α = 0.1)	5	3.75%	4.57%		0.95%	0.88%		9.64%	2.73%	
Elastic (α = 0.1)	3	4.36%	0.19%	1.88%	0.25%	0.65%	1.83%	7.76%	2.03%	6.65%
Elastic (α = 0.2)	10	0.03%			0.01%			8.96%		
Elastic (α = 0.2)	5	2.89%	4.38%		0.62%	0.33%		9.61%	2.72%	
Elastic (α = 0.2)	3	2.45%	0.13%	5.96%	0.05%	0.36%	1.30%	7.75%	2.24%	1.34%
Elastic (α = 0.3)	10	0.00%			0.00%			8.90%		
Elastic (α = 0.3)	5	2.74%	4.41%		0.49%	0.30%		9.57%	2.71%	
Elastic (α = 0.3)	3	1.89%	0.13%	9.71%	0.00%	0.33%	1.21%	7.74%	2.52%	0.40%
Elastic (α = 0.4)	10	0.01%			0.01%			8.84%		
Elastic (α = 0.4)	5	2.88%	3.59%		0.38%	0.31%		9.54%	2.70%	
Elastic (α = 0.4)	3	1.84%	0.10%	5.43%	0.00%	0.32%	1.17%	7.73%	3.20%	0.68%
Elastic (α = 0.5)	10	3.60%			0.15%			8.79%		
Elastic (α = 0.5)	5	2.89%	3.58%		0.31%	0.22%		9.51%	2.68%	
Elastic (α = 0.5)	3	1.67%	0.11%	5.24%	0.01%	0.21%	0.95%	7.72%	3.17%	1.28%
Elastic (α = 0.6)	10	0.00%			0.16%			8.73%		
Elastic (α = 0.6)	5	2.70%	3.21%		0.32%	0.24%		9.47%	2.66%	
Elastic (α = 0.6)	3	1.61%	0.14%	13.8%	0.01%	0.65%	0.92%	7.71%	3.14%	0.03%
Elastic (α = 0.7)	10	0.00%			0.14%			8.70%		
Elastic (α = 0.7)	5	2.52%	3.10%		0.31%	1.60%		9.42%	2.63%	
Elastic (α = 0.7)	3	1.66%	0.11%	11.3%	0.02%	1.60%	1.00%	7.70%	3.12%	0.03%
Elastic (α = 0.8)	10	0.01%			0.13%			8.67%		
Elastic (α = 0.8)	5	2.36%	3.16%		0.22%	1.10%		9.37%	2.60%	
Elastic (α = 0.8)	3	1.85%	0.13%	14.3%	0.01%	1.10%	1.63%	7.69%	3.09%	0.01%
Elastic (α = 0.9)	10	0.00%			0.14%			8.64%		
Elastic (α = 0.9)	5	2.15%	3.43%		0.32%	1.29%		9.32%	2.56%	
Elastic(α = 0.9)	3	1.52%	0.13%	14.5%	0.05%	1.29%	0.99%	7.68%	3.07%	0.05%
Lasso (α = 1)	10	0.00%			0.16%			8.58%		
Lasso (α = 1)	5	1.92%	2.69%		0.60%	0.82%		9.27%	2.53%	
Lasso (α = 1)	3	1.19%	0.12%	10.8%	0.06%	0.82%	2.66%	7.67%	3.04%	0.01%

Table 4 presents the results for neural networks for both monthly and yearly returns predictions, including both daily and monthly frequency data for monthly predictions. In general, neural network models generated worse results than penalized linear regressions. As the depth of the structure increases, the predictive power also slightly increases. For 10-year return, two and three layers work better than single layer structures. The best R^2_{OOS} value that monthly returns yielded is around 2%. However, yearly return estimations increased the predictive power especially by three-layer structure with five nodes each, to close to 4%.

Table 4: R^2_{oos} for Neural Networks (Yields only)

Models		**R^2oos – Yeilds Only**								
		Daily Frequency			Monthly Frequency			Monthly Frequency		
		Monthly return			Monthly return			Yearly return		
	#factors	$r^{(10)}_{t+1}$	$r^{(5)}_{t+1}$	$r^{(3)}_{t+1}$	$r^{(10)}_{t+1}$	$r^{(5)}_{t+1}$	$r^{(3)}_{t+1}$	$r^{(10)}_{t+1}$	$r^{(5)}_{t+1}$	$r^{(3)}_{t+1}$
D. Neural Networks										
NN-1 layer (3 nodes)	10	0.22%			0.44%			0.03%		
	5	0.20%	1.68%		0.07%	0.12%		0.09%	0.00%	
	3	0.06%	0.25%	0.04%	0.01%	0.16%	0.05%	2.24%	0.53%	0.12%
NN-1 layer (5 nodes)	10	0.12%			0.08%			0.52%		
	5	0.60%	0.64%		0.07%	0.01%		0.00%	0.03%	
	3	0.65%	0.43%	0.02%	0.05%	0.02%	0.01%	0.04%	0.01%	0.02%
NN-1 layer (7 nodes)	10	0.01%			0.06%			0.19%		
	5	0.01%	0.01%		0.03%	0.03%		0.08%	0.19%	
	3	0.04%	0.04%	0.09%	0.02%	0.07%	0.06%	0.06%	1.02%	0.09%
NN-2 layers (3+3 nodes)	10	1.16%			0.16%			0.01%		
	5	0.38%	0.26%		0.01%	0.02%		0.10%	1.32%	
	3	0.24%	0.12%	0.26%	0.04%	0.09%	0.13%	0.03%	1.12%	0.24%
NN-2 layers (5+5 nodes)	10	0.05%			0.02%			0.07%		
	5	2.19%	0.14%		0.00%	0.02%		0.06%	0.47%	
	3	0.95%	0.06%	0.06%	0.00%	0.01%	0.04%	0.05%	0.22%	0.02%
NN-2 layers (7+7 nodes)	10	1.06%			0.06%			0.03%		
	5	0.00%	0.73%		0.35%	0.11%		0.70%	0.03%	
	3	0.05%	0.26%	0.17%	0.19%	0.04%	0.15%	0.22%	0.22%	0.25%
NN-2 layers (8+4 nodes)	10	0.19%			0.10%			0.26%		
	5	1.12%	0.23%		0.01%	0.21%		0.74%	0.28%	
	3	0.76%	0.07%	0.36%	0.00%	0.08%	0.17%	0.11%	0.32%	1.04%
NN-3 layers (3+3+3 nodes)	10	1.64%			0.31%			0.27%		
	5	1.03%	0.21%		0.00%	0.06%		0.33%	0.11%	
	3	0.87%	0.14%	0.09%	0.05%	0.05%	0.02%	0.23%	0.13%	0.03%
NN-3 layers (5+5+5 nodes)	10	0.01%			0.03%			3.63%		
	5	0.76%	0.09%		0.01%	0.00%		0.09%	0.00%	

	3	0.45%	0.06%	0.17%	0.05%	0.01%	0.09%	1.08%	0.04%	0.56%
	10	0.15%			0.91%			1.94%		
NN-3 layers (7+7+7 nodes)	5	0.20%	0.26%		0.01%	0.20%		0.61%	0.07%	
	3	0.11%	0.10%	0.11%	0.23%	0.07%	0.06%	1.28%	0.06%	0.54%
	10	0.06%			0.08%			0.00%		
NN-3 layers (5+4+3 nodes)	5	0.73%	0.21%		0.07%	0.17%		0.08%	0.77%	
	3	0.18%	0.07%	0.05%	0.11%	0.08%	0.03%	0.16%	0.21%	0.01%

Table 5 and 6 show the results for linear models and principal component analysis for both monthly and yearly returns predictions, including both daily and monthly frequency data for monthly predictions. Similar to yields-only approach, yearly return predictions significantly improve when compared to monthly predictions for linear models, Ridge, Lasso and Elastic net regressions, and slightly improve for neural network models. On the other hand, a significant effect of macro variables with the linear models in monthly predictions is not observed; R^2_{OOS} values are improved by only 2-3% for all maturity bonds, compared to yields-only approach. However, a negative effect of macro data can be seen when estimating yearly returns. Although there is no significant difference in the results of the OLS and PLS models, it is seen that the predictive power of the penalized models is adversely affected. There are twenty different macro variables included in the model. Such variables as Hours Worked in Manufacturing and Industrial Durable Consumer Goods data occasionally becomes stable. The returns of these indices might be falsifying. Plus, Turkey's stock market index is recorded in Turkish liras, whereas the Eurobonds are in US dollars. Stock exchange index return might have been modified in terms of currency.

Table 7 presents the results for neural networks for both monthly and yearly returns predictions, including both daily and monthly frequency data for monthly predictions. For monthly returns, it is observed that yields and macro data approach results almost similar and slightly worse predictive power with yields-only approach, with R^2_{OOS} values around 0-1%. On the other hand, for yearly return predictions, macroeconomic variables play a minor role and slightly improve predictive power for especially ten and five-year returns. There are some interesting results in Table 7. For example, although two-layer network structures work best on average around 2%, the

highest R^2_{OOS} value comes from a single-layer 10-year return just above 3%. Also, for 10-year return, less forward rates data slightly improves predictive power especially as the structure gets more complex. It is seen that three or five factor models with two and three layers produce higher R^2_{OOS} values for 10-year returns. For 5-year returns, predictive power of more factors is better when the neural network structure is smaller; however, predictive power with fewer factors improves with increasing number of layers

Table 5: R^2_{oos} for PCA, LM, PLS (Yields + Macrodata)

Models	**R^2_{OOS} – Yields + Macrodata**					
	Monthly Frequency			Monthly Frequency		
	Monthly return			Yearly return		
	$r^{(10)}_{t+1}$	$r^{(5)}_{t+1}$	$r^{(3)}_{t+1}$	$r^{(10)}_{t+1}$	$r^{(5)}_{t+1}$	$r^{(3)}_{t+1}$
A. Dimensionality Reduction						
PCA						
PCA (5 factors)						
PCA (3 factors)						
B. Linear models						
LM	-38.7%			-7.83%		
LM (5 factors)	-39.7%	-50.0%		-10.8%	-25.9%	
LM (3 factors)	-40.1%	-50.1%	-37.7%	-9.18%	-26.1%	-43.5%
PLS	-38.7%			-20.67%		
PLS (5 factors)	-38.9%	-50.9%		-19.6%	-31.5%	
PLS (3 factors)	-38.9%	-51.0%	-38.6%	-19.3%	-31.4%	-47.7%

Table 6: R^2_{oos} for Ridge, Lasso and Elasitc regressions (Yields + Macrodata)

Models		**R^2oos – Yields + Macrodata**					
		Monthly Frequency Monthly return			Monthly Frequency Yearly return		
	#factors	$r^{(10)}_{t+1}$	$r^{(5)}_{t+1}$	$r^{(3)}_{t+1}$	$r^{(10)}_{t+1}$	$r^{(5)}_{t+1}$	$r^{(3)}_{t+1}$
C. Penalized Linear models							
Ridge (α = 0)	10	2.54%			1.83%		
Ridge (α = 0)	5	2.82%	4.25%		0.34%	2.09%	
Ridge (α = 0)	3	2.82%	4.08%	6.49%	0.16%	0.03%	1.43%
Elastic (α = 0.1)	10	2.67%			0.01%		
Elastic (α = 0.1)	5	2.90%	4.47%		0.26%	0.24%	
Elastic (α = 0.1)	3	3.62%	5.00%	3.80%	2.07%	0.17%	6.63%
Elastic (α = 0.2)	10	2.80%			0.42%		
Elastic (α = 0.2)	5	2.98%	4.58%		0.12%	0.56%	
Elastic (α = 0.2)	3	3.73%	5.15%	3.19%	0.14%	0.04%	1.28%
Elastic (α = 0.3)	10	2.97%			0.84%		
Elastic (α = 0.3)	5	3.12%	4.63%		0.60%	0.73%	
Elastic (α = 0.3)	3	3.85%	5.19%	9.26%	0.02%	0.19%	0.34%
Elastic (α = 0.4)	10	3.17%			1.19%		
Elastic (α = 0.4)	5	3.28%	4.65%		0.98%	0.86%	
Elastic (α = 0.4)	3	3.73%	5.19%	9.13%	0.18%	0.35%	0.67%
Elastic (α = 0.5)	10	3.32%			1.45%		
Elastic (α = 0.5)	5	3.41%	4.67%		1.28%	0.88%	
Elastic (α = 0.5)	3	3.84%	5.16%	8.94%	0.39%	0.47%	1.27%
Elastic (α = 0.6)	10	3.40%			1.64%		
Elastic (α = 0.6)	5	3.52%	4.69%		1.61%	1.00%	
Elastic (α = 0.6)	3	3.91%	5.21%	8.70%	0.57%	0.64%	0.01%
Elastic (α = 0.7)	10	3.45%			1.77%		
Elastic (α = 0.7)	5	3.57%	4.68%		1.82%	1.09%	
Elastic (α = 0.7)	3	3.99%	5.23%	8.45%	0.77%	0.64%	0.01%
Elastic (α = 0.8)	10	3.49%			1.92%		
Elastic (α = 0.8)	5	3.60%	4.69%		1.96%	1.10%	
Elastic (α = 0.8)	3	4.03%	5.24%	8.17%	0.82%	0.75%	0.00%
Elastic (α = 0.9)	10	3.51%			2.09%		
Elastic (α = 0.9)	5	3.63%	4.65%		2.18%	1.14%	
Elastic (α = 0.9)	3	4.05%	5.22%	7.90%	0.98%	0.69%	0.04%
Lasso (α = 1)	10	3.52%			2.29%		
Lasso (α = 1)	5	3.70%	4.62%		2.23%	1.03%	
Lasso (α = 1)	3	4.07%	5.20%	7.58%	1.06%	0.78%	0.03%

Table 7: R^2_{oos} for Neural Networks (Yields + Macrodata)

Models		**R^2oos – Yields + Macrodata**					
		Monthly Frequency			Monthly Frequency		
		Monthly return			Yearly return		
	#factors	$r^{(10)}_{t+1}$	$r^{(5)}_{t+1}$	$r^{(3)}_{t+1}$	$r^{(10)}_{t+1}$	$r^{(5)}_{t+1}$	$r^{(3)}_{t+1}$
E. Neural Networks							
NN-1 layer (3 nodes)	10	0.02%			0.08%		
	5	0.05%	0.02%		0.91%	0.07%	
	3	0.00%	0.00%	0.25%	3.23%	0.12%	0.05%
NN-1 layer (5 nodes)	10	0.16%			0.08%		
	5	0.01%	0.01%		0.29%	0.17%	
	3	0.12%	0.51%	0.03%	0.00%	0.01%	0.03%
NN-1 layer (7 nodes)	10	0.07%			0.09%		
	5	0.00%	0.23%		0.60%	0.27%	
	3	0.08%	0.08%	0.10%	0.03%	0.06%	0.53%
NN-2 layers (3+3 nodes)	10	0.09%			0.01%		
	5	0.03%	0.13%		0.10%	0.15%	
	3	0.03%	0.05%	0.10%	0.76%	0.11%	0.01%
NN-2 layers (5+5 nodes)	10	0.10%			0.12%		
	5	0.06%	0.09%		0.14%	0.68%	
	3	0.05%	0.03%	0.02%	1.14%	0.01%	0.00%
NN-2 layers (7+7 nodes)	10	0.01%			0.20%		
	5	0.04%	0.00%		0.65%	2.04%	
	3	0.02%	0.10%	0.01%	0.03%	0.00%	0.12%
NN-2 layers (8+4 nodes)	10	0.17%			0.27%		
	5	0.01%	0.01%		0.04%	0.24%	
	3	0.01%	0.02%	0.00%	1.43%	1.52%	0.03%
NN-3 layers (3+3+3 nodes)	10	0.03%			0.25%		
	5	0.16%	0.00%		0.01%	0.03%	
	3	0.05%	0.02%	0.00%	0.40%	2.09%	0.08%
NN-3 layers (5+5+5 nodes)	10	0.03%			0.20%		
	5	0.06%	0.04%		0.01%	0.65%	

	3	0.14%	0.01%	0.05%	0.58%	0.96%	1.65%
	10	0.09%			0.00%		
NN-3 layers (7+7+7 nodes)	5	0.00%	0.00%		0.46%	0.28%	
	3	0.29%	0.00%	0.03%	1.88%	1.79%	0.01%
	10	0.01%			0.14%		
NN-3 layers (5+4+3 nodes)	5	0.01%	0.19%		0.31%	1.28%	
	3	0.00%	0.02%	0.10%	0.01%	0.05%	0.23%

CHAPTER VI

CONCLUSION

Machine learning techniques are relatively new in finance and economics; therefore, those techniques are less used in predicting bond returns and term structure. With recent improvements in statistics and computer science, economists' preferences of models for forecasting and prediction have been altering. The prevailing methods have been mostly limited to data compression techniques, but with the recent developments new model preferences such as dimensionality reduction techniques are more visible. That these methods not being comfortable with structural analysis and parameter interferences might be the reason for the limitation. Indeed, machine learning sets priority to out-of-sample prediction by utilizing a large set of sample data. This makes machine learning techniques more suitable for estimating asset returns.

The focal point of this work is to predict excess bond returns with different maturities by using various factors. To summarize, I predict 3,5, 10-year Turkish Eurobonds monthly and yearly returns by incorporating in forward rates only and macroeconomics variable included. The data set ranges from 2005 to 2020, inclusive. For prediction, principal component analysis, linear models, partial least squares, penalized linear regressions, and neural network methods are used. The models' success is evaluated according to their out-of-sample R^2 values. In line with expectations, yields-only approach with linear models produced negative R^2_{OOS} values for monthly returns, while predicting yearly returns improved significantly. This improvement is valid for most of the models used. However, I face some perplexing results, as well. For example,

instead of neural networks, penalized linear regressions yield the most powerful predictions. With that, in some models, shorter maturity returns are more successfully estimated than longer maturity bond returns. Plus, adding macro variables usually improve R^2_{OOS} values rather slightly than considerably.

For emerging markets, and especially for Turkey treasury market, this work can be considered an important inception for the use of machine learning techniques for predicting term structure and excess bond returns. My work produces some striking results, and with the help of macro variables, it can be significantly improved building on it.

REFERENCES

[1] D. J. Bolder and D. Streliski, “Yield curve modelling at the Bank of Canada,” SSRN Electronic Journal, 1999.

[2] F. X. Diebold and C. Li, “Forecasting the term structure of government bond yields,” Journal of Econometrics, vol. 130, no. 2, pp. 337–364, 2006.

[3] L. E. O. Svensson, “Estimating and interpreting forward interest rates: Sweden 1992 - 1994,” Sep. 1994.

[4] J. M. Culbertson, “The term structure of interest rates,” The Quarterly Journal of Economics, vol. 71, no. 4, pp. 485–517, Aug. 1957.

[5] R. J. Shiller, J. Y. Campbell, K. L. Schoenholtz, and L. Weiss, “Forward rates and future policy: Interpreting the term structure of interest rates,” Brookings Papers on Economic Activity, vol. 1983, no. 1, pp. 173–223, 1983.

[6] E. F. Fama, “Forward and spot exchange rates,” Journal of Monetary Economics, vol. 14, no. 3, pp. 319–338, Nov. 1984.

[7] J. C. Cox, J. E. Ingersoll, and S. A. Ross, “An intertemporal general equilibrium model of asset prices,” Econometrica, vol. 53, no. 2, pp. 363–384, Mar. 1985.

[8] E. Fama and R. Bliss, “The information on long-maturity forward rates,” The American Economic Review, vol. 77, no. 4, pp. 680–692, Sep. 1987.

[9] A. N. T. T. I. Ilmanen, “Time-varying expected returns in international bond markets,” The Journal of Finance, vol. 50, no. 2, pp. 481–506, Jun. 1995.

[10] J. H. Cochrane and M. Piazzesi, “Bond risk premia,” American Economic Review, vol. 95, no. 1, pp. 138–160, Mar. 2005.

[11] R. Sambasivan and S. Das, “A statistical machine learning approach to yield curve forecasting,” 2017 International Conference on Computational Intelligence in Data Science (ICCIDS), 2017.

[12] D. Bianchi, M. Büchner, and A. Tamoni, “Bond risk premia with machine learning,” SSRN Electronic Journal, May 2020.

[13] S. C. Ludvigson and S. Ng, “Macro factors in Bond Risk Premia,” Review of Financial Studies, vol. 22, no. 12, pp. 5027–5067, Dec. 2009.

[14] G. S. Shea, “Pitfalls in smoothing interest rate term structure data: Equilibrium Models and spline approximations,” The Journal of Financial and Quantitative Analysis, vol. 19, no. 3, pp. 253–269, Sep. 1984.

[15] J. Campbell and J. Cochrane, “By force of habit: A consumption-based explanation of aggregate stock market behavior,” Journal of Political Economy, vol. 107, no. 2, Apr. 1999.

VITA

Ali Dartanel

Education:

- Ozyegin University, Istanbul, Turkey — Sep 2019 - May 2022

  Graduate School of Sciences and Engineering,

  Dept. of Financial Engineering

- McGill University, Montreal, Canada — Sep 2010 – Jun 2015

  Bachelor of Engineering, Electrical Engineering

- The Koc School, Istanbul, Turkey — Sep 2005 – Jun 2010

  High School, International Baccalaureate Program

Professional:

- Garanti BBVA Asset Management, Istanbul, Turkey — Jan 2021 – present
- HSBC Asset Management, Istanbul, Turkey — Jul 2017 – Jan 2021
- MWH Turkey, Project Management, Istanbul, Turkey — Jul 2015 – Jan 2017