Bond Risk Premia with Machine Learning

Daniele Bianchi
University of Warwick

Matthias Büchner
University of Warwick

Andrea Tamoni
London School of Economics

Modelling with Big Data and Machine Learning

November 2018

This Paper

What we do: We propose, compare and evaluate a variety of machine learning methods for bond returns predictability in a regression-based context.

This Paper

What we do: We propose, compare and evaluate a variety of machine learning methods for bond returns predictability in a regression-based context.

Research questions:

— Can machine learning (ML) methods capture predictable variation in bond excess returns better than benchmark principal component regressions (PCR)?

This Paper

What we do: We propose, compare and evaluate a variety of machine learning methods for bond returns predictability in a regression-based context.

Research questions:

— Can machine learning (ML) methods capture predictable variation in bond excess returns better than benchmark principal component regressions (PCR)?

— Why is it so? In particular, what features really matter?

This Paper

What we do: We propose, compare and evaluate a variety of machine learning methods for bond returns predictability in a regression-based context.

Research questions:

— Can machine learning (ML) methods capture predictable variation in bond excess returns better than benchmark principal component regressions (PCR)?

— Why is it so? In particular, what features really matter?

— How to disentangle the incremental contribution of ML for the measurement of bond risk premia? which variables matter?

This Paper

What we do: We propose, compare and evaluate a variety of machine learning methods for bond returns predictability in a regression-based context.

Research questions:

— Can machine learning (ML) methods capture predictable variation in bond excess returns better than benchmark principal component regressions (PCR)?

— Why is it so? In particular, what features really matter?

— How to disentangle the incremental contribution of ML for the measurement of bond risk premia? which variables matter?

This is relevant for any application that requires estimates of expected bond returns: (a) selecting portfolios, (b) evaluating portfolio performance, (c) risk management, (d) understanding risk premia, etc...

Main Findings

Supervised learning vs. data compression:

We show that Principal Component Analysis (PCA) is outperformed by both sparse linear models, e.g., ElasticNet, and dense non-linear models, e.g., shallow and deep neural networks (NN).

$$\text{PCA} \ll \text{Penalized regressions} \ll \text{NNs}$$

Main Findings

Supervised learning vs. data compression:

We show that Principal Component Analysis (PCA) is outperformed by both sparse linear models, e.g., ElasticNet, and dense non-linear models, e.g., shallow and deep neural networks (NN).

$$\text{PCA} \ll \text{Penalized regressions} \ll \text{NNs}$$

Macro and financial variables through the lens of NNs:

Substantial heterogeneity in the relative importance of predictors; broadly speaking financial variables (output growth/inflation) matter for the short-end (long-end) of the yield curve.

Main Findings

Supervised learning vs. data compression:

We show that Principal Component Analysis (PCA) is outperformed by both sparse linear models, e.g., ElasticNet, and dense non-linear models, e.g., shallow and deep neural networks (NN).

$$\text{PCA} \ll \text{Penalized regressions} \ll \text{NNs}$$

Macro and financial variables through the lens of NNs:

Substantial heterogeneity in the relative importance of predictors; broadly speaking financial variables (output growth/inflation) matter for the short-end (long-end) of the yield curve.

Why neural networks work:

The factors extracted from NNs have incremental forecasting power for the level and slope of the term structure, as well as aggregate economy activity and nominal variables.

Related Literature

Recent contributions:

Bond returns predictability:

(e.g., Fama and Bliss (1987), Campbell and Shiller (1991), Cochrane and Piazzesi (2005), Ludvigson and Ng (2009), Thornton and Valente (2012), Joslin, Priebsch and Singleton (2014), Greenwood and Vayanos (2014), Cieslak and Povala (2015), Gargano, Pettenuzzo and Timmermann (2017),...)

Machine learning in empirical asset pricing:

(e.g., Kuan and White (1994), Lo (1994), Hutchinson, Lo and Poggio (1994), Kelly and Pruitt (2013,2015), Feng, Giglio and Xiu (2017), Freyberger, Neuhierl and Weber (2017), Giglio and Xiu (2017), Gu, Kelly and Xiu (2018),...)

Deep learning in empirical finance:

(e.g., Jones (2006), Heaton, Polson and Witte (2017), Messmer (2017), Polson and Sokolov (2017), Sirignano, Sadhwani and Giesecke (2018),...)

Outline

- Motivation.
- Research design.
- An empirical study of US treasury bonds.
- Understand the performance of neural networks
- Conclusions.

Motivation

Identity links current yield to the sum of one-period yields and excess returns

$$y_t^{(n)} = \frac{1}{n} E_t \left(\sum_{j=0}^{\infty} y_{t+j}^{(1)} \right) + \frac{1}{n} E_t \left(\sum_{j=0}^{\infty} xr_{t+j,t+j+1}^{(n-j)} \right)$$

(see Duffee 2013)

Motivation

Identity links current yield to the sum of one-period yields and excess returns

$$y_t^{(n)} = \frac{1}{n} E_t \left(\sum_{j=0}^{\infty} y_{t+j}^{(1)} \mid \boldsymbol{x}_t \right) + \frac{1}{n} E_t \left(\sum_{j=0}^{\infty} xr_{t+j,t+j+1}^{(n-j)} \mid \boldsymbol{x}_t \right)$$

(see Duffee 2013)

Motivation

Identity links current yield to the sum of one-period yields and excess returns

$$y_t^{(n)} = \frac{1}{n} E_t \left(\sum_{j=0}^{\infty} y_{t+j}^{(1)} \mid \boldsymbol{x}_t \right) + \frac{1}{n} E_t \left(\sum_{j=0}^{\infty} xr_{t+j,t+j+1}^{(n-j)} \mid \boldsymbol{x}_t \right)$$

(see Duffee 2013)

State variables $\boldsymbol{x}_t$ are assumed to have a unique effect on the yield curve $\boldsymbol{y}_t$, i.e.,

$$\boldsymbol{y}_t = f(\boldsymbol{x}_t; N) \ ,$$

Motivation

Identity links current yield to the sum of one-period yields and excess returns

$$y_t^{(n)} = \frac{1}{n} E_t \left(\sum_{j=0}^{\infty} y_{t+j}^{(1)} \mid \boldsymbol{x}_t \right) + \frac{1}{n} E_t \left(\sum_{j=0}^{\infty} xr_{t+j,t+j+1}^{(n-j)} \mid \boldsymbol{x}_t \right)$$

(see Duffee 2013)

State variables $\boldsymbol{x}_t$ are assumed to have a unique effect on the yield curve $\boldsymbol{y}_t$, i.e.,

$$\boldsymbol{x}_t = f^{-1}(\boldsymbol{y}_t; N)$$

e.g., $\boldsymbol{x}_t$: level, slope and curvature.

Motivation

Identity links current yield to the sum of one-period yields and excess returns

$$y_t^{(n)} = \frac{1}{n} E_t \left(\sum_{j=0}^{\infty} y_{t+j}^{(1)} \mid \boldsymbol{x}_t \right) + \frac{1}{n} E_t \left(\sum_{j=0}^{\infty} xr_{t+j,t+j+1}^{(n-j)} \mid \boldsymbol{x}_t \right)$$

(see Duffee 2013)

State variables $\boldsymbol{x}_t$ are assumed to have a unique effect on the yield curve $\boldsymbol{y}_t$, i.e.,

$$\boldsymbol{x}_t = f^{-1}(\boldsymbol{y}_t; N)$$

e.g., $\boldsymbol{x}_t$: level, slope and curvature.

$$\downarrow$$

$$E_t \left(xr_{t+1}^{(n)} \right) = g(\boldsymbol{y}_t; N)$$

for some $g(\cdot)$ unknown a priori.

Motivation

Identity links current yield to the sum of one-period yields and excess returns

$$y_t^{(n)} = \frac{1}{n} E_t \left(\sum_{j=0}^{\infty} y_{t+j}^{(1)} \mid \boldsymbol{x}_t \right) + \frac{1}{n} E_t \left(\sum_{j=0}^{\infty} xr_{t+j,t+j+1}^{(n-j)} \mid \boldsymbol{x}_t \right)$$

(see Duffee 2013)

State variables $\boldsymbol{x}_t$ are assumed to have a unique effect on the yield curve $\boldsymbol{y}_t$, i.e.,

$$\boldsymbol{x}_t = f^{-1}(\boldsymbol{y}_t; N)$$

e.g., $\boldsymbol{x}_t$: level, slope and curvature.

$$\downarrow$$

$$E_t \left(xr_{t+1}^{(n)} \right) = g(\boldsymbol{y}_t; N)$$

for some $g(\cdot)$ unknown a priori.

A canonical approach is to assume $g(\cdot)$ is linear, s.t.

$$E_t \left(xr_{t+1}^{(n)} \right) = \hat{\alpha} + \hat{\boldsymbol{\beta}}^\top \boldsymbol{x}_t$$

Ex 1: Cochrane and Piazzesi (2005), $\boldsymbol{x}_t$ is a linear combination of the forward rates.

Motivation

Identity links current yield to the sum of one-period yields and excess returns

$$y_t^{(n)} = \frac{1}{n} E_t \left(\sum_{j=0}^{\infty} y_{t+j}^{(1)} \mid \boldsymbol{x}_t \right) + \frac{1}{n} E_t \left(\sum_{j=0}^{\infty} xr_{t+j,t+j+1}^{(n-j)} \mid \boldsymbol{x}_t \right)$$

(see Duffee 2013)

State variables $\boldsymbol{x}_t$ are assumed to have a unique effect on the yield curve $\boldsymbol{y}_t$, i.e.,

$$\boldsymbol{x}_t = f^{-1}(\boldsymbol{y}_t; N)$$

e.g., $\boldsymbol{x}_t$: level, slope and curvature.

$$\downarrow$$

$$E_t \left(xr_{t+1}^{(n)} \right) = g(\boldsymbol{y}_t; N)$$

for some $g(\cdot)$ unknown a priori.

A canonical approach is to assume $g(\cdot)$ is linear, s.t.

$$E_t \left(xr_{t+1}^{(n)} \right) = \hat{\alpha} + \hat{\boldsymbol{\beta}}^\top \boldsymbol{x}_t + \hat{\boldsymbol{\gamma}}^\top \boldsymbol{F}_t$$

Ex 1: Cochrane and Piazzesi (2005), $\boldsymbol{x}_t$ is a linear combination of the forward rates.

Ex 2: Ludvigson and Ng (2009), $\boldsymbol{F}_t$ is a linear combination of a large panel of macro variables.

Motivation: Why do we need Machine Learning?

POLICY FORUM | TECHNOLOGY AND THE ECONOMY

What can machine learning do?

Erik Brynjolfsson[1,2], Tom Mitchell[3]
+ See all authors and affiliations

Science 22 Dec 2017:
Vol. 358, Issue 6370, pp. 1530-1534
DOI: 10.1126/science.aap8062

Key criteria for suitable machine learning tasks:

— Learning a function that maps well-defined inputs to outputs.

learning $g(\cdot)$

— Clear feedback with clearly definable goals and metrics outputs.

predictive R^2

— No need for detailed explanation of how "decisions" are taken.

focus is on prediction and not inference.

Research Design

Research Design

Goal: Recursive forecasting of the one-year holding period excess returns.

Research Design

Goal: Recursive forecasting of the one-year holding period excess returns.

Two benchmark applications of regression-based forecasting:

- Forecasting based on forward rates (Cochrane and Piazzesi 2005).
- Forecasting based on a large panel of macro variables and forward rates (Ludvigson and Ng 2009).

Research Design

Goal: Recursive forecasting of the one-year holding period excess returns.

Two benchmark applications of regression-based forecasting:

- Forecasting based on forward rates (Cochrane and Piazzesi 2005).
- Forecasting based on a large panel of macro variables and forward rates (Ludvigson and Ng 2009).

Competing Methodologies:

- Data compression methods (e.g., PCA and PLS)
- Simple and penalized linear regressions (e.g., Ridge, Lasso, ElasticNet)
- Non-linear models (regression trees, random forests, shallow and deep neural networks)

Research Design

Goal: Recursive forecasting of the one-year holding period excess returns.

Two benchmark applications of regression-based forecasting:

- Forecasting based on forward rates (Cochrane and Piazzesi 2005).
- Forecasting based on a large panel of macro variables and forward rates (Ludvigson and Ng 2009).

Competing Methodologies:

- Data compression methods (e.g., PCA and PLS)
- Simple and penalized linear regressions (e.g., Ridge, Lasso, ElasticNet)
- Non-linear models (regression trees, random forests, shallow and deep neural networks)

Out-of-sample performance metrics:

- Mean Squared Prediction Error (MSPE) and predictive R^2 (see Campbell and Thompson 2007).
- Diebold and Mariano (2002) test for pairwise comparison across models.

Research Design: Digression on Neural Networks

Recall, for e.g.,

$$E_t\left(xr_{t+1}^{(n)}\right) = \hat{\alpha} + \hat{\boldsymbol{\beta}}^\top \boldsymbol{x}_t$$

Research Design: Digression on Neural Networks

Recall, for e.g.,

$$E_t\left(xr_{t+1}^{(n)}\right) = \hat{\alpha} + \hat{\boldsymbol{\beta}}^\top \boldsymbol{x}_t$$

the neural network is just a hierarchical model of the form

$$\begin{aligned}
\boldsymbol{x}_t &= h_L\left(\boldsymbol{W}_L \boldsymbol{z}_{L-1,t} + b_L\right) && \text{(Hidden layer } L\text{)} \\
\boldsymbol{z}_{L-1,t} &= h_{L-1}\left(\boldsymbol{W}_{L-1} \boldsymbol{z}_{L-2,t} + b_{L-1}\right) && \text{(Hidden layer } L-1\text{)} \\
&\vdots \\
\boldsymbol{z}_{1,t} &= h_1\left(\boldsymbol{W}_1 \boldsymbol{y}_t + b_1\right) && \text{(Hidden layer 1)}
\end{aligned}$$

Activation function:

Rectified Linear Unit (ReLU) activation function, i.e.,

$$h(z) = \max(z, 0)$$

Research Design: Digression on Neural Networks

Recall, for e.g.,

$$E_t\left(xr_{t+1}^{(n)}\right) = \hat{\alpha} + \hat{\boldsymbol{\beta}}^\top \boldsymbol{x}_t$$

the neural network is just a hierarchical model of the form

$$\begin{aligned}
\boldsymbol{x}_t &= h_L\left(\boldsymbol{W}_L \boldsymbol{z}_{L-1,t} + b_L\right) && \text{(Hidden layer } L\text{)} \\
\boldsymbol{z}_{L-1,t} &= h_{L-1}\left(\boldsymbol{W}_{L-1} \boldsymbol{z}_{L-2,t} + b_{L-1}\right) && \text{(Hidden layer } L-1\text{)} \\
&\vdots \\
\boldsymbol{z}_{1,t} &= h_1\left(\boldsymbol{W}_1 \boldsymbol{y}_t + b_1\right) && \text{(Hidden layer 1)}
\end{aligned}$$

Activation function:

Rectified Linear Unit (ReLU) activation function, i.e.,

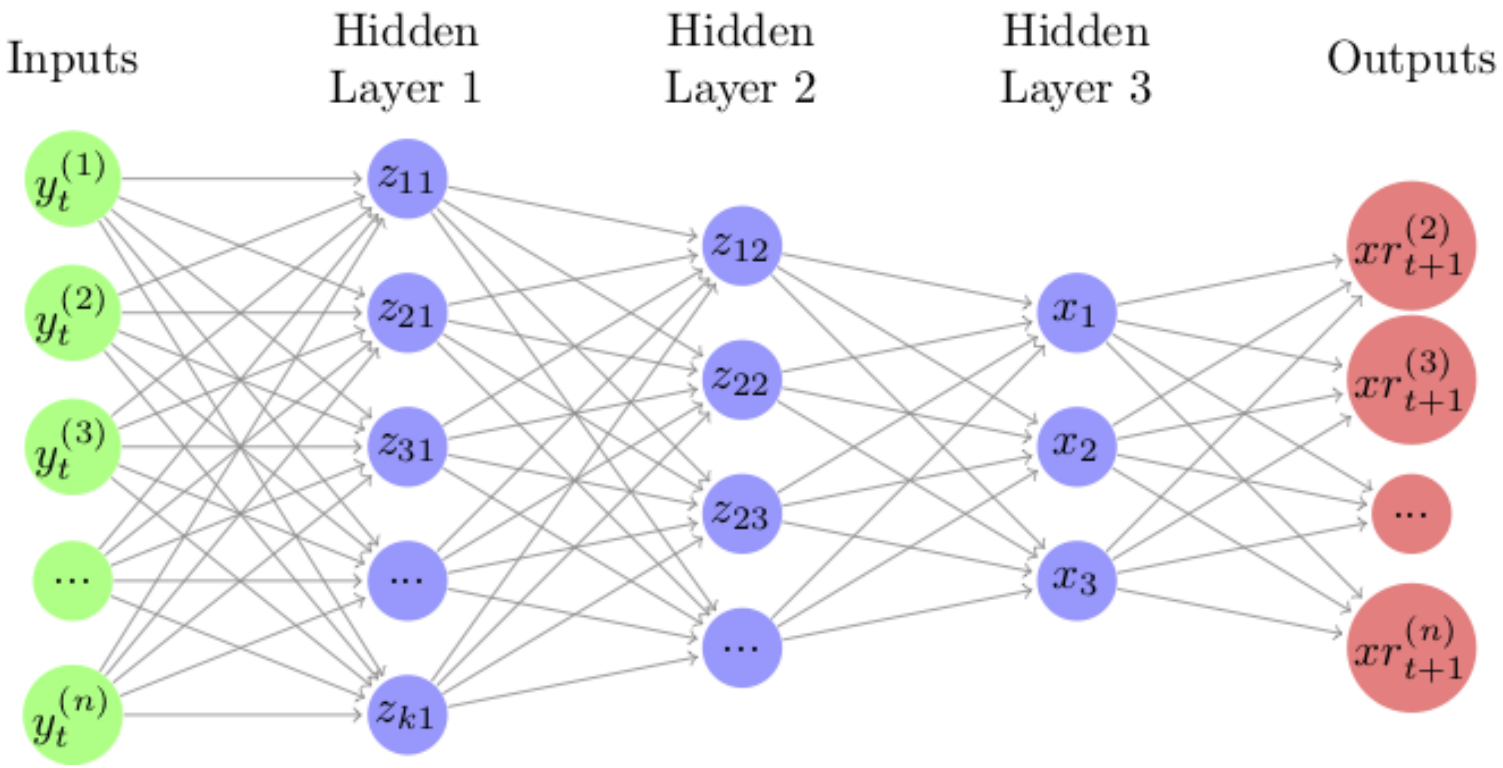
$$h(z) = \max(z, 0)$$

Optimization and regularization:

Stochastic Gradient Descent (SGD)
+
Early stopping, batch normalization, model averaging, drop-out.

Research Design: Digression on Neural Networks

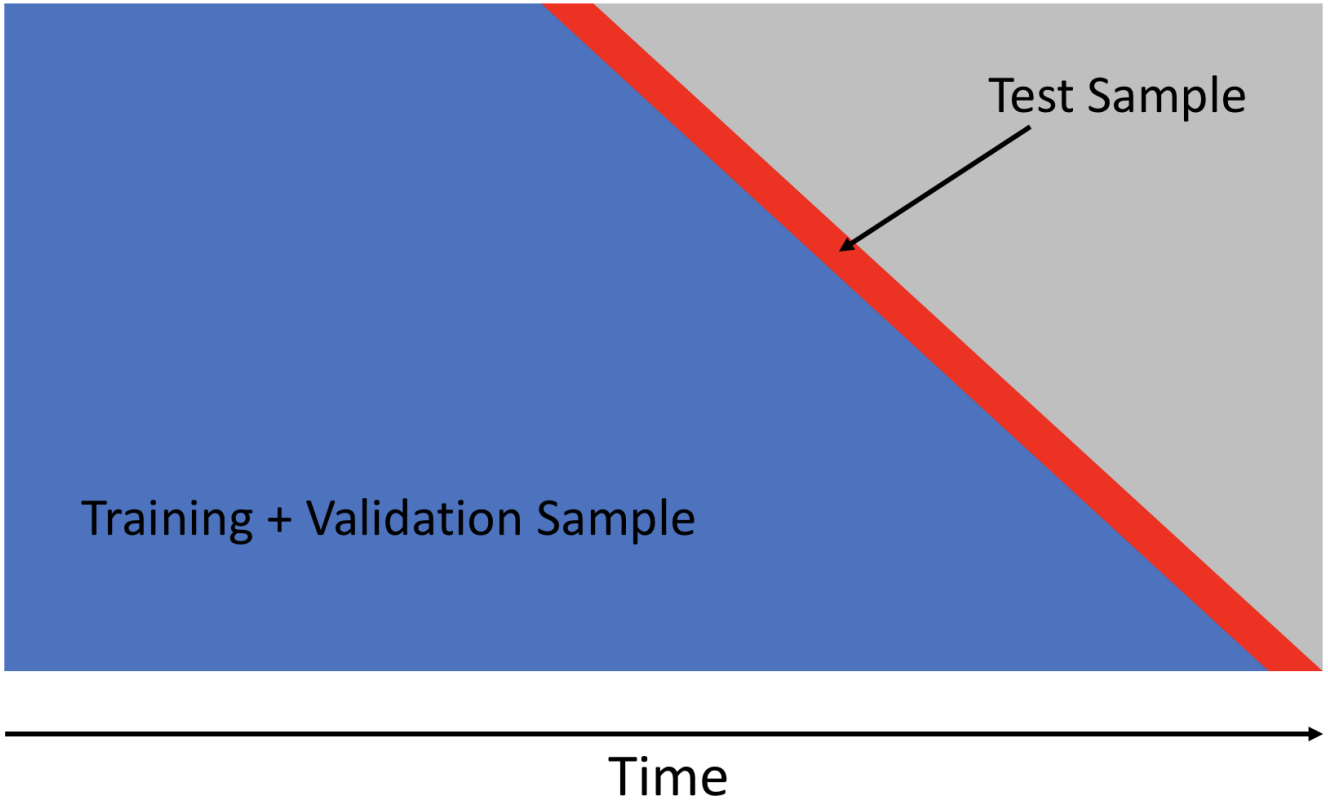
Figure: Example of a Neural Network



This figure shows a "deep" neural network which consists of an ouput layer and three hidden layers. The green circles represent the input variables, that is the cross-section of yields, macroeconomic variables of both. The purple circles represent the fully connected hidden nodes. The red circles represent the output variables, that is the bond excess returns across maturities.

Research Design: Sample Splitting

- We initially divide the sample in three parts: training (85%) + validation (15%) and testing sample (one-month ahead).
- We recursively refit each machine learning method at each time t.



An Empirical Study of US Treasury Bonds

Bond Return Predictability and the Yield Curve

Table: Out-of-Sample Results with Forward Rates, Short Sample

	Mean Squared Prediction Error				$R^2_{oos}(\%)$			
	$rx^{(2)}_{t+1}$	$rx^{(3)}_{t+1}$	$rx^{(4)}_{t+1}$	$rx^{(5)}_{t+1}$	$rx^{(2)}_{t+1}$	$rx^{(3)}_{t+1}$	$rx^{(4)}_{t+1}$	$rx^{(5)}_{t+1}$
Panel A: PCA, PLS and Autoencoder								
PCA (5 components)	2.69	9.21	17.25	26.17	-41.5	-31.3	-28.9	-27.2
PCA (3 components)	2.64	9.09	17.11	25.97	-38.4	-29.5	-27.9	-26.2
PLS (5 components)	2.69	9.21	17.25	26.17	-41.5	-31.3	-28.9	-27.2
PLS (3 components)	2.64	9.09	17.09	25.98	-38.7	-29.5	-27.8	-26.2
Autoencoder	2.56	8.82	16.51	25.05	-34.7	-25.7	-23.4	-21.7
Panel B: Simple and Penalized Linear Regressions								
OLS	2.69	9.21	17.25	26.17	-41.5	-31.3	-28.9	-27.2
Ridge	2.59	8.92	16.80	25.35	-35.8	-27.2	-25.6	-23.2
Lasso	1.94	6.95	13.43	20.46	-1.8	1.0	-0.4	0.6
Elastic Net	1.95	7.01	13.52	20.91	-2.5	0.0	-1.1	-1.6
Panel C: Regression Trees and Neural Networks								
Boosted Regression Tree	2.16	7.37	14.17	21.15	-13.3	-5.0	-5.9	-2.8
Random Forests	2.13	6.99	12.61	20.58	-12.1	0.4	5.7	0.0
NN - 2 layer (5 nodes)	1.94	6.45	11.90	17.80	-1.9	8.1	11.0	13.5
NN - 2 layer (3 nodes)	1.82	6.41	11.79	17.53	4.3	8.7	11.8	14.8
NN - 3 layer (5 nodes each)	1.75	5.92	10.72	15.95	7.9	15.6	19.9	22.5
NN - 3 layer (3 nodes each)	1.82	6.10	11.36	16.56	4.6	13.0	15.1	19.6
NN - 4 Layer (3 nodes each)	1.76	5.97	11.00	16.20	7.4	15.0	17.8	21.3
NN - 4 Layer (4,3,2 nodes each)	1.69	5.91	10.83	16.07	11.0	15.7	19.0	21.9

Bond Return Predictability and the Yield Curve

Table: Out-of-Sample Results with Forward Rates, Short Sample

	Mean Squared Prediction Error				$R^2_{oos}(\%)$			
	$rx^{(2)}_{t+1}$	$rx^{(3)}_{t+1}$	$rx^{(4)}_{t+1}$	$rx^{(5)}_{t+1}$	$rx^{(2)}_{t+1}$	$rx^{(3)}_{t+1}$	$rx^{(4)}_{t+1}$	$rx^{(5)}_{t+1}$
Panel A: PCA, PLS and Autoencoder								
PCA (5 components)	2.69	9.21	17.25	26.17	-41.5	-31.3	-28.9	-27.2
PCA (3 components)	2.64	9.09	17.11	25.97	-38.4	-29.5	-27.9	-26.2
PLS (5 components)	2.69	9.21	17.25	26.17	-41.5	-31.3	-28.9	-27.2
PLS (3 components)	2.64	9.09	17.09	25.98	-38.7	-29.5	-27.8	-26.2
Autoencoder	2.56	8.82	16.51	25.05	-34.7	-25.7	-23.4	-21.7
Panel B: Simple and Penalized Linear Regressions								
OLS	2.69	9.21	17.25	26.17	-41.5	-31.3	-28.9	-27.2
Ridge	2.59	8.92	16.80	25.35	-35.8	-27.2	-25.6	-23.2
Lasso	1.94	6.95	13.43	20.46	-1.8	1.0	-0.4	0.6
Elastic Net	1.95	7.01	13.52	20.91	-2.5	0.0	-1.1	-1.6
Panel C: Regression Trees and Neural Networks								
Boosted Regression Tree	2.16	7.37	14.17	21.15	-13.3	-5.0	-5.9	-2.8
Random Forests	2.13	6.99	12.61	20.58	-12.1	0.4	5.7	0.0
NN - 2 layer (5 nodes)	1.94	6.45	11.90	17.80	-1.9	8.1	11.0	13.5
NN - 2 layer (3 nodes)	1.82	6.41	11.79	17.53	4.3	8.7	11.8	14.8
NN - 3 layer (5 nodes each)	1.75	5.92	10.72	15.95	7.9	15.6	19.9	22.5
NN - 3 layer (3 nodes each)	1.82	6.10	11.36	16.56	4.6	13.0	15.1	19.6
NN - 4 Layer (3 nodes each)	1.76	5.97	11.00	16.20	7.4	15.0	17.8	21.3
NN - 4 Layer (4,3,2 nodes each)	1.69	5.91	10.83	16.07	11.0	15.7	19.0	21.9

Bond Return Predictability and the Yield Curve

Table: Out-of-Sample Results with Forward Rates, Short Sample

	Mean Squared Prediction Error				$R^2_{oos}(\%)$			
	$rx^{(2)}_{t+1}$	$rx^{(3)}_{t+1}$	$rx^{(4)}_{t+1}$	$rx^{(5)}_{t+1}$	$rx^{(2)}_{t+1}$	$rx^{(3)}_{t+1}$	$rx^{(4)}_{t+1}$	$rx^{(5)}_{t+1}$
Panel A: PCA, PLS and Autoencoder								
PCA (5 components)	2.69	9.21	17.25	26.17	-41.5	-31.3	-28.9	-27.2
PCA (3 components)	2.64	9.09	17.11	25.97	-38.4	-29.5	-27.9	-26.2
PLS (5 components)	2.69	9.21	17.25	26.17	-41.5	-31.3	-28.9	-27.2
PLS (3 components)	2.64	9.09	17.09	25.98	-38.7	-29.5	-27.8	-26.2
Autoencoder	2.56	8.82	16.51	25.05	-34.7	-25.7	-23.4	-21.7
Panel B: Simple and Penalized Linear Regressions								
OLS	2.69	9.21	17.25	26.17	-41.5	-31.3	-28.9	-27.2
Ridge	2.59	8.92	16.80	25.35	-35.8	-27.2	-25.6	-23.2
Lasso	1.94	6.95	13.43	20.46	-1.8	1.0	-0.4	0.6
Elastic Net	1.95	7.01	13.52	20.91	-2.5	0.0	-1.1	-1.6
Panel C: Regression Trees and Neural Networks								
Boosted Regression Tree	2.16	7.37	14.17	21.15	-13.3	-5.0	-5.9	-2.8
Random Forests	2.13	6.99	12.61	20.58	-12.1	0.4	5.7	0.0
NN - 2 layer (5 nodes)	1.94	6.45	11.90	17.80	-1.9	8.1	11.0	13.5
NN - 2 layer (3 nodes)	1.82	6.41	11.79	17.53	4.3	8.7	11.8	14.8
NN - 3 layer (5 nodes each)	1.75	5.92	10.72	15.95	7.9	15.6	19.9	22.5
NN - 3 layer (3 nodes each)	1.82	6.10	11.36	16.56	4.6	13.0	15.1	19.6
NN - 4 Layer (3 nodes each)	1.76	5.97	11.00	16.20	7.4	15.0	17.8	21.3
NN - 4 Layer (4,3,2 nodes each)	1.69	5.91	10.83	16.07	11.0	15.7	19.0	21.9

Bond Return Predictability and the Yield Curve

Table: Out-of-Sample Results with Forward Rates, Long Sample

	Mean Squared Prediction Error				$R^2_{oos}(\%)$			
	$rx^{(2)}_{t+1}$	$rx^{(3)}_{t+1}$	$rx^{(4)}_{t+1}$	$rx^{(5)}_{t+1}$	$rx^{(2)}_{t+1}$	$rx^{(3)}_{t+1}$	$rx^{(4)}_{t+1}$	$rx^{(5)}_{t+1}$
Panel A: PCA, PLS and Autoencoder								
PCA (5 components)	2.39	8.06	14.93	22.49	-78.7	-62.3	-48.2	-38.8
PCA (3 components)	2.06	6.92	12.94	19.26	-54.0	-39.2	-28.4	-18.8
PLS (5 components)	2.39	8.06	14.93	22.49	-78.7	-62.3	-48.2	-38.8
PLS (3 components)	2.21	7.54	13.87	20.85	-64.7	-51.9	-37.6	-28.6
Autoencoder	2.01	6.68	12.29	18.37	-50.4	-34.5	-22.0	-13.3
Panel B: Simple and Penalized Linear Regressions								
OLS	2.39	8.06	14.93	22.49	-78.7	-62.3	-48.2	-38.8
Ridge	2.15	7.21	13.47	20.28	-60.4	-45.1	-33.6	-25.1
Lasso	1.34	5.00	10.10	15.97	-0.5	-0.6	-0.2	1.4
Elastic Net	1.36	5.04	10.17	16.35	-1.5	-1.5	-0.9	-0.9
Panel C: Regression Trees and Neural Networks								
Boosted Regression Tree	1.49	5.28	10.59	16.67	-11.4	-6.4	-5.0	-2.8
Random Forests	1.44	4.71	9.12	13.42	-7.5	5.2	9.5	17.2
NN - 2 layer (5 nodes)	1.49	4.80	8.99	13.59	-11.3	3.4	10.8	16.1
NN - 2 layer (3 nodes)	1.47	4.82	9.03	13.56	-10.1	2.9	10.4	16.3
NN - 3 layer (5 nodes each)	1.37	4.58	8.31	12.50	-2.2	7.8	17.5	22.9
NN - 3 layer (3 nodes each)	1.48	4.77	8.75	13.03	-10.7	4.0	13.2	19.6
NN - 4 Layer (3 nodes each)	1.46	4.67	8.55	12.84	-8.9	6.0	15.2	20.8
NN - 4 Layer (4,3,2 nodes each)	1.28	4.37	8.21	13.07	4.6	11.9	18.6	19.3

Bond Return Predictability and the Yield Curve

Table: Out-of-Sample Results with Forward Rates, Long Sample

	Mean Squared Prediction Error				$R^2_{oos}(\%)$			
	$rx^{(2)}_{t+1}$	$rx^{(3)}_{t+1}$	$rx^{(4)}_{t+1}$	$rx^{(5)}_{t+1}$	$rx^{(2)}_{t+1}$	$rx^{(3)}_{t+1}$	$rx^{(4)}_{t+1}$	$rx^{(5)}_{t+1}$
Panel A: PCA, PLS and Autoencoder								
PCA (5 components)	2.39	8.06	14.93	22.49	-78.7	-62.3	-48.2	-38.8
PCA (3 components)	2.06	6.92	12.94	19.26	-54.0	-39.2	-28.4	-18.8
PLS (5 components)	2.39	8.06	14.93	22.49	-78.7	-62.3	-48.2	-38.8
PLS (3 components)	2.21	7.54	13.87	20.85	-64.7	-51.9	-37.6	-28.6
Autoencoder	2.01	6.68	12.29	18.37	-50.4	-34.5	-22.0	-13.3
Panel B: Simple and Penalized Linear Regressions								
OLS	2.39	8.06	14.93	22.49	-78.7	-62.3	-48.2	-38.8
Ridge	2.15	7.21	13.47	20.28	-60.4	-45.1	-33.6	-25.1
Lasso	1.34	5.00	10.10	15.97	-0.5	-0.6	-0.2	1.4
Elastic Net	1.36	5.04	10.17	16.35	-1.5	-1.5	-0.9	-0.9
Panel C: Regression Trees and Neural Networks								
Boosted Regression Tree	1.49	5.28	10.59	16.67	-11.4	-6.4	-5.0	-2.8
Random Forests	1.44	4.71	9.12	13.42	-7.5	5.2	9.5	17.2
NN - 2 layer (5 nodes)	1.49	4.80	8.99	13.59	-11.3	3.4	10.8	16.1
NN - 2 layer (3 nodes)	1.47	4.82	9.03	13.56	-10.1	2.9	10.4	16.3
NN - 3 layer (5 nodes each)	1.37	4.58	8.31	12.50	-2.2	7.8	17.5	22.9
NN - 3 layer (3 nodes each)	1.48	4.77	8.75	13.03	-10.7	4.0	13.2	19.6
NN - 4 Layer (3 nodes each)	1.46	4.67	8.55	12.84	-8.9	6.0	15.2	20.8
NN - 4 Layer (4,3,2 nodes each)	1.28	4.37	8.21	13.07	4.6	11.9	18.6	19.3

Bond Return Predictability and the Yield Curve

Summary of the results:

1. Data compression methodologies show similar performances.
2. Sparse linear regressions, e.g., elastic-net, may help improve the forecasts of bond excess returns.
3. Non-linearity matters: Deep neural networks exploit non-linearities which are overlooked by using a simple linear combination of yields.
   - Deep networks tend to outpeform shallow networks, especially at the short/medium term of the term structure and more so when the ZLB binds.

Bond Return Predictability and Macro Variables

Table: OOS Results with Macro Variables and Fwd Rates, Short Sample

	Mean Squared Prediction Error				R^2_{oos}(%)			
	$rx^{(2)}_{t+1}$	$rx^{(3)}_{t+1}$	$rx^{(4)}_{t+1}$	$rx^{(5)}_{t+1}$	$rx^{(2)}_{t+1}$	$rx^{(3)}_{t+1}$	$rx^{(4)}_{t+1}$	$rx^{(5)}_{t+1}$
Panel A: PCA, PLS and Autoencoder								
PCA as in LN	1.98	7.47	15.02	22.91	-3.8	-6.5	-12.3	-11.3
PLS (macro + fwd rates)	2.38	8.81	17.23	26.12	-25.0	-25.6	-28.8	-26.9
Autoencoder + CP factor	2.11	7.88	15.50	23.17	-10.6	-12.3	-15.8	-12.6
Panel B: Simple and Penalized Linear Regressions								
OLS	2.38	8.77	17.47	27.16	-25.2	-25.1	-30.6	-31.9
Ridge (macro + fwd rates)	3.04	10.95	21.80	34.62	-59.7	-56.0	-62.9	-68.2
Lasso (macro + fwd rates)	1.96	7.23	13.91	20.84	-2.7	-3.0	-4.0	-1.2
Elastic Net (macro + fwd rates)	1.96	7.16	13.28	20.11	-2.8	-2.0	0.7	2.3
Panel C: Regression Trees and Neural Networks								
Boosted Regression Trees	1.63	5.54	11.24	16.35	14.4	21.0	15.9	20.6
Random Forests	1.42	4.85	9.52	14.61	25.7	28.9	28.8	29.0
NN 2 Layer, ensemble macro and fwd	1.41	5.04	9.41	14.32	25.7	28.2	29.7	30.4
NN 3 Layer, ensemble macro and fwd	1.40	4.99	9.39	14.31	26.6	28.9	29.8	30.5
NN 4 Layer, ensemble macro and fwd	1.39	5.02	9.48	14.41	27.1	28.4	29.2	30.0
NN 2 Layer, groups ensembling and fwd	1.37	5.00	9.52	14.67	27.8	28.7	28.8	28.7

Bond Return Predictability and Macro Variables

Table: OOS Results with Macro Variables and Fwd Rates, Long Sample

	Mean Squared Prediction Error				$R^2_{oos}(\%)$			
	$rx^{(2)}_{t+1}$	$rx^{(3)}_{t+1}$	$rx^{(4)}_{t+1}$	$rx^{(5)}_{t+1}$	$rx^{(2)}_{t+1}$	$rx^{(3)}_{t+1}$	$rx^{(4)}_{t+1}$	$rx^{(5)}_{t+1}$
Panel A: PCA, PLS and Autoencoder								
PCA as in LN	1.46	5.00	9.78	14.79	-9.3	-0.7	2.9	8.7
PLS (macro + fwd rates)	2.17	6.70	12.50	19.14	-62.5	-34.8	-24.0	-18.1
Autoencoder + CP factor	1.51	5.37	11.05	17.02	-12.6	-8.1	-9.6	-5.0
Panel B: Simple and Penalized Linear Regressions								
OLS	3.04	10.19	19.76	29.87	-127.2	-105.2	-96.0	-84.3
Ridge (macro + fwd rates)	2.49	8.58	16.84	26.67	-85.9	-72.7	-67.1	-64.6
Lasso (macro + fwd rates)	1.39	5.08	10.01	15.78	-3.8	-2.2	0.7	2.6
Elastic Net (macro + fwd rates)	1.39	5.00	10.09	16.40	-3.8	-0.7	-0.1	-1.2
Panel C: Regression Trees and Neural Networks								
Boosted Regression Trees	1.51	4.28	8.36	13.26	-12.8	13.8	17.1	18.2
Random Forests	1.26	3.87	7.24	13.17	5.7	22.1	28.1	18.7
NN 2 Layer, ensemble macro and fwd	1.88	4.94	8.10	11.96	-40.7	0.6	19.6	26.2
NN 3 Layer, ensemble macro and fwd	1.17	4.17	7.33	10.94	12.3	13.8	27.2	32.5
NN 4 Layer, ensemble macro and fwd	1.04	3.76	6.81	10.28	22.3	22.1	32.5	36.6
NN 2 Layer, groups ensembling and fwd	1.19	3.61	6.98	11.05	11.0	27.3	30.8	31.8

Bond Return Predictability and Macro Variables

Table: OOS Results with Macro Variables and Fwd Rates, Long Sample

	Mean Squared Prediction Error				$R^2_{oos}(\%)$			
	$rx^{(2)}_{t+1}$	$rx^{(3)}_{t+1}$	$rx^{(4)}_{t+1}$	$rx^{(5)}_{t+1}$	$rx^{(2)}_{t+1}$	$rx^{(3)}_{t+1}$	$rx^{(4)}_{t+1}$	$rx^{(5)}_{t+1}$
Panel A: PCA, PLS and Autoencoder								
PCA as in LN	1.46	5.00	9.78	14.79	-9.3	-0.7	2.9	8.7
PLS (macro + fwd rates)	2.17	6.70	12.50	19.14	-62.5	-34.8	-24.0	-18.1
Autoencoder + CP factor	1.51	5.37	11.05	17.02	-12.6	-8.1	-9.6	-5.0
Panel B: Simple and Penalized Linear Regressions								
OLS	3.04	10.19	19.76	29.87	-127.2	-105.2	-96.0	-84.3
Ridge (macro + fwd rates)	2.49	8.58	16.84	26.67	-85.9	-72.7	-67.1	-64.6
Lasso (macro + fwd rates)	1.39	5.08	10.01	15.78	-3.8	-2.2	0.7	2.6
Elastic Net (macro + fwd rates)	1.39	5.00	10.09	16.40	-3.8	-0.7	-0.1	-1.2
Panel C: Regression Trees and Neural Networks								
Boosted Regression Trees	1.51	4.28	8.36	13.26	-12.8	13.8	17.1	18.2
Random Forests	1.26	3.87	7.24	13.17	5.7	22.1	28.1	18.7
NN 2 Layer, ensemble macro and fwd	1.88	4.94	8.10	11.96	-40.7	0.6	19.6	26.2
NN 3 Layer, ensemble macro and fwd	1.17	4.17	7.33	10.94	12.3	13.8	27.2	32.5
NN 4 Layer, ensemble macro and fwd	1.04	3.76	6.81	10.28	22.3	22.1	32.5	36.6
NN 2 Layer, groups ensembling and fwd	1.19	3.61	6.98	11.05	11.0	27.3	30.8	31.8

Bond Return Predictability and Macro Variables

Summary of the results:

1. We confirm that macroeconomic variables carry information that is not contained in the yield curve (Duffee 2011, Joslin et al. 2014).

2. Elastic-net substantially outperforms alternative penalized regressions such as Lasso and Ridge.

3. Non-linearity matters: both shallow and deep neural networks outperforms competing strategies, especially for the longer sample.

   - For the longer sample the depth of the network matters, whereas for the short sample shallow and deep networks are almost equivalent.
   - Groups ensembling could compensate the depth of the network.
   - Separate networks for macro and fwd rates substantially improves the predictive performance.

Relative Importance of Macroeconomic Variables

It is instructive to understand what variables might drive the results within the context of neural networks.

Relative Importance of Macroeconomic Variables

It is instructive to understand what variables might drive the results within the context of neural networks.

$$\left.\frac{\partial xr_{t+1}^{(n)}}{\partial y_{it}}\right|_{y_{it}=\overline{y}_i}$$

We investigate at each time t the marginal relevance of single variables based on the partial derivative of the target variable with respect to the sample average of each input.

where $\overline{y}_i$ represents the in-sample mean of the input variable i.

Relative Importance of Macroeconomic Variables

It is instructive to understand what variables might drive the results within the context of neural networks.

$$\left.\frac{\partial xr_{t+1}^{(n)}}{\partial y_{it}}\right|_{y_{it}=\overline{y}_i}$$

We investigate at each time t the marginal relevance of single variables based on the partial derivative of the target variable with respect to the sample average of each input.

where $\overline{y}_i$ represents the in-sample mean of the input variable i.

This allows to compute the sensitivity conditional on the network structure and the average value of the other input variables.

Relative Importance of Macroeconomic Variables

It is instructive to understand what variables might drive the results within the context of neural networks.

$$\left.\frac{\partial xr_{t+1}^{(n)}}{\partial y_{it}}\right|_{y_{it}=\overline{y}_i}$$

We investigate at each time t the marginal relevance of single variables based on the partial derivative of the target variable with respect to the sample average of each input.

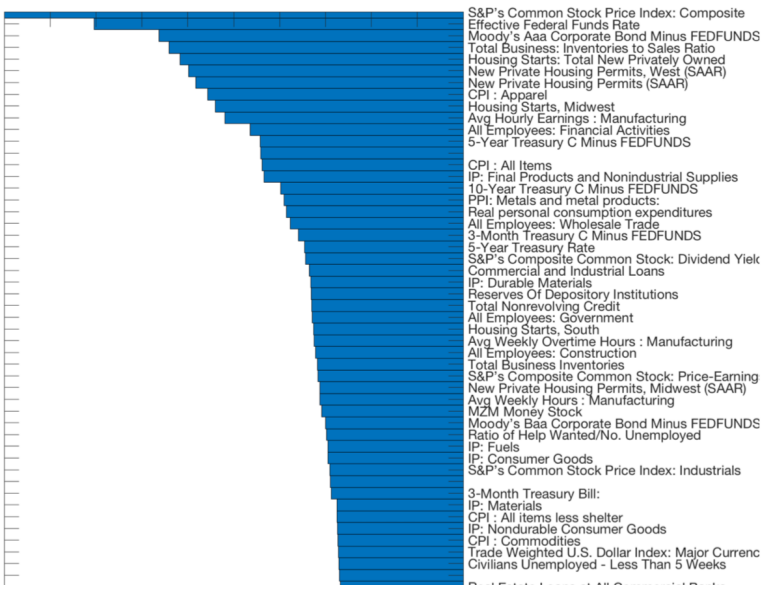
where $\overline{y}_i$ represents the in-sample mean of the input variable i.

This allows to compute the sensitivity conditional on the network structure and the average value of the other input variables.

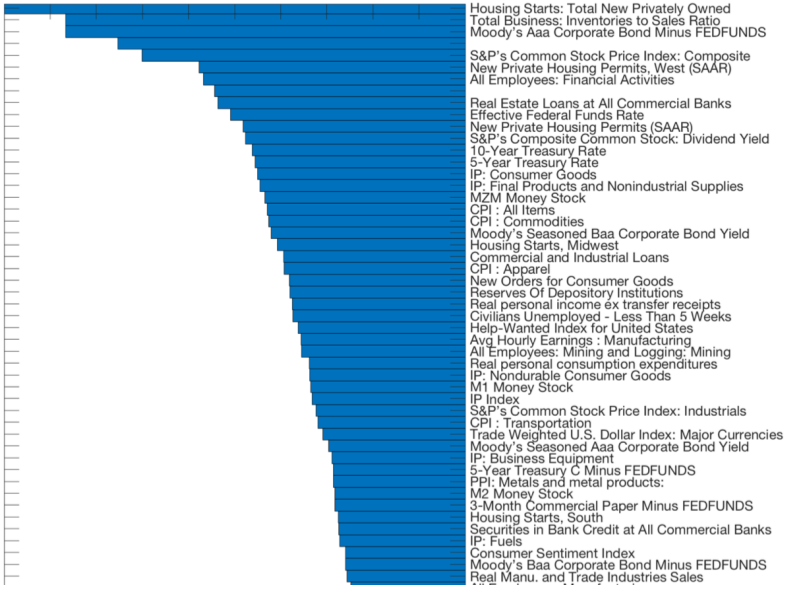
N.B: Results are averaged throughout the OOS period; benchmark specification is a deep neural network with four layers.

Relative Importance of Macroeconomic Variables

Figure: Relative Importance Averaged Across Maturities



(a) Short Sample



(b) Long Sample

This figure shows the relative importance of each input variable obtained from a four-layer neural network. Variables are labeled according to McCracken and Ng (2016). The results are further averaged across the bond maturities. The left panel shows the results for the short sample period from 1964:01 to 2008:12, whereas the right panel shows the results for the long sample period from 1964:01 to 2016:12.

Relative Importance of Macroeconomic Variables

We further investigate:

1. The maturity-specific effect of macro and financial variables.
2. Both the average and the maturity-specific effect of groups of predictors.

Relative Importance of Macroeconomic Variables

We further investigate:

1. The maturity-specific effect of macro and financial variables.
2. Both the average and the maturity-specific effect of groups of predictors.

Maturity-specific effect:

- Short-end of the curve: S&P price, term and credit spread matter more.
- Long-end of the curve: housing starts and prices matter more.

Relative Importance of Macroeconomic Variables

We further investigate:

1. The maturity-specific effect of macro and financial variables.
2. Both the average and the maturity-specific effect of groups of predictors.

Maturity-specific effect:

- Short-end of the curve: S&P price, term and credit spread matter more.
- Long-end of the curve: housing starts and prices matter more.

Groups of variables:

- Short-end of the curve: financial variables matter more.
- Long-end of the curve: output&income, consumption matter more.

Such heterogeneity is confirmed for the short and the long sample.

The Value of Neural Networks

Forecasting PCs of Yields and Macroeconomic Variables

$$PC_{i,t+12} - PC_{i,t} = b_0 + \boldsymbol{b}_1^\top \mathcal{P}_t + \boldsymbol{b}_2^\top \boldsymbol{x}_t + \epsilon_{i,t+1} \qquad \text{for} \quad i = 1, 2, 3$$

Forecasting PCs of Yields and Macroeconomic Variables

$$PC_{i,t+12} - PC_{i,t} = b_0 + \boldsymbol{b}_1^\top \mathcal{P}_t + \boldsymbol{b}_2^\top \boldsymbol{x}_t + \epsilon_{i,t+1} \qquad \text{for} \quad i = 1, 2, 3$$

$PC_{i,t}$ is the ith principal component extracted from the yield curve or macroeconomic variables.

Forecasting PCs of Yields and Macroeconomic Variables

$$PC_{i,t+12} - PC_{i,t} = b_0 + \boldsymbol{b}_1^\top \mathcal{P}_t + \boldsymbol{b}_2^\top \boldsymbol{x}_t + \epsilon_{i,t+1} \qquad \text{for} \quad i = 1, 2, 3$$

$PC_{i,t}$ is the ith principal component extracted from the yield curve or macroeconomic variables.

$\mathcal{P}_t = (PC_{1,t}, PC_{2,t}, PC_{3,t})'$

Level, slope and curvature if from yields.

Economic activity, interest rate spreads, inflation if from macro variables (see Ludvigson and Ng 2009).

Forecasting PCs of Yields and Macroeconomic Variables

$$PC_{i,t+12} - PC_{i,t} = b_0 + \boldsymbol{b}_1^\top \mathcal{P}_t + \boldsymbol{b}_2^\top \boldsymbol{x}_t + \epsilon_{i,t+1} \qquad \text{for} \quad i = 1, 2, 3$$

$PC_{i,t}$ is the ith principal component extracted from the yield curve or macroeconomic variables.

$\mathcal{P}_t = (PC_{1,t}, PC_{2,t}, PC_{3,t})'$

Level, slope and curvature if from yields.

Economic activity, interest rate spreads, inflation if from macro variables (see Ludvigson and Ng 2009).

$\boldsymbol{x}_t$ are the factors extracted from the last hidden layer of a neural network (either using yields or macro variables).

Forecasting PCs of Yields and Macroeconomic Variables

$$PC_{i,t+12} - PC_{i,t} = b_0 + \boldsymbol{b}_1^\top \mathcal{P}_t + \boldsymbol{b}_2^\top \boldsymbol{x}_t + \epsilon_{i,t+1} \qquad \text{for} \quad i = 1, 2, 3$$

$PC_{i,t}$ is the ith principal component extracted from the yield curve or macroeconomic variables.

$\mathcal{P}_t = (PC_{1,t}, PC_{2,t}, PC_{3,t})'$

Level, slope and curvature if from yields.

Economic activity, interest rate spreads, inflation if from macro variables (see Ludvigson and Ng 2009).

$\boldsymbol{x}_t$ are the factors extracted from the last hidden layer of a neural network (either using yields or macro variables).

Question: What is the incremental explanatory power (w.r.t. PCs) of the hidden factors extracted from a neural network?

Forecasting PCs of Yields

Panel A: Predicting changes in PCs of yields from yields.

	Short Sample			Long Sample		
	PC # 1	PC # 2	PC # 3	PC # 1	PC # 2	PC # 3
PCA	16.83	28.08	58.78	11.38	26.38	55.79
NN 2 Layer + PCA	22.22	29.21	59.23	17.47	30.30	56.57
NN 3 Layer + PCA	23.06	28.78	59.46	19.39	29.26	56.90
NN 4 Layer + PCA	24.82	28.88	59.41	19.54	29.16	56.97

This table reports the in-sample R^2 (%) of a predictive regression where the dependent variable is the year-on-year growth rate of the first three principal components extracted from the cross-section of forward rates. The independent variables are the first three principal components extracted from the term structure. In addition to the PCA we also compute the in-sample R^2 obtained by adding the factors extracted from the same information sets obtained by a set of shallow and deep neural networks (3 nodes for each layer).

Forecasting PCs of Yields

Panel B: Predicting changes in PCs of yields from macro variables.

	Short Sample			Long Sample		
	PC # 1	PC # 2	PC # 3	PC # 1	PC # 2	PC # 3
PCA	15.47	39.61	3.26	6.75	35.84	2.92
NN 2 Layer + PCA	52.00	58.98	16.16	22.94	48.40	11.19
NN 3 Layer + PCA	62.03	57.18	14.05	20.07	44.43	7.34
NN 4 Layer + PCA	55.07	53.19	10.86	22.91	42.72	5.58

This table reports the in-sample R^2 (%) of a predictive regression where the dependent variable is the year-on-year growth rate of the first three principal components extracted from the cross-section of forward rates. The independent variables are the first three principal components extracted from macroeconomic variables. In addition to the PCA we also compute the in-sample R^2 obtained by adding the factors extracted from the same information sets obtained by a set of shallow and deep neural networks.

Forecasting PCs of Macroeconomic Variables

Panel C: Predicting changes in PCs of macroeconomic variables.

	Short Sample			Long Sample		
	PC # 1	PC # 2	PC # 3	PC # 1	PC # 2	PC # 3
PCA	38.97	37.73	52.28	39.20	39.67	50.06
NN 2 Layer + PCA	42.20	47.16	62.63	42.95	45.74	57.46
NN 3 Layer + PCA	40.59	42.87	61.95	40.54	43.56	55.37
NN 4 Layer + PCA	40.18	42.39	60.86	40.42	43.48	56.98

This table reports the in-sample R^2 (%) of a predictive regression where the dependent variable is the year-on-year growth rate of the first three principal components extracted from a large panel of macroeconomic variables. The independent variables are the first three principal components extracted from macroeconomic variables. In addition to the PCA we also compute the in-sample R^2 obtained by adding the factors extracted from the same information sets obtained by a set of shallow and deep neural networks.

Conclusion

We show that machine learning techniques can lead to a more accurate out-of-sample forecast of bond excess returns.

In particular, we provide evidence that the functional form of the mapping between current yields, macroeconomic information and future bond excess returns might not be linear.

Open issues/next steps:

- Monthly returns in addition to one-year holding period returns.
- Additional methodologies, e.g., group Lasso, predictive Lasso, slice-inverse regressions, etc.
- ...