Deep Learning in Financial Time series

10th January 2022

Artificial Intelligence Finance Institute

NYU Courant

Artificial Intelligence Finance Institute

AIFI
ARTIFICIAL INTELLIGENCE
IN FINANCE INSTITUTE

The Artificial Intelligence Finance Institute's (AIFI) mission is to be the world's leading educator in the application of artificial intelligence to investment management, capital markets and risk. We offer one of the industry's most comprehensive and in-depth educational programs, geared towards investment professionals seeking to understand and implement cutting edge AI techniques.

Taught by a diverse staff of world leading academics and practitioners, the AIFI courses teach both the theory and practical implementation of artificial intelligence and machine learning tools in investment management. As part of the program, students will learn the mathematical and statistical theories behind modern quantitative artificial intelligence modeling. Our goal is to train investment professionals in how to use the new wave of computer driven tools and techniques that are rapidly transforming investment management, risk management and capital markets.

Deep Learning in Finance

Machine Learning in Finance

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

Supervised Learning

Predictive or Descriptive

Regression

Learn Regression Function

$f: \mathbb{R}^n \to \mathbb{R}$

Given: Inputs and outputs

(X_i, Y_i)

Classification

Learn Class Function

$f: \mathbb{R}^n \to \{1, \dots, k\}$

Given: Inputs and outputs

(X_i, C_i)

Unsupervised Learning

Descriptive

Clustering

Learn Class Function

$f: \mathbb{R}^n \to \{1, \dots, k\}$

Given: Inputs

(X_i)

Representation Learning

Learn Representer function

$f: \mathbb{R}^n \to \mathbb{R}^k$

Given : Inputs

(X_i)

Reinforcement Learning

Prescriptive

Learn Policy

Learn Policy Function

$f: \mathbb{R}^n \to \mathbb{R}^k$

Given : Tuples

(X_i, a_i, X_{i+1}, r_i)

Inverse Reinforcement Learning

Learn Reward Function

$f: \mathbb{R}^n \to \mathbb{R}$

Given : Tuples

(X_i, a_i, X_{i+1})

Machine Learning in Finance

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

UNSUPERVISED CLUSTERING

- k-Means, FuzzyC-Means
- Hierarchical
- Neural Networks
- Gaussian Mixture
- Hidden Markov Models

SUPERVISED

CLASSIFICATION	REGRESSION
Support Vector Machines	Neural Networks
Discriminant Analysis	Decision Trees
Naïve Bayes	Ensemble Methods
Nearest Neighbors	Non-linearReg. (GLM, Logistic)
CART	Linear Regression

Deep Learning

- Multilayer Perceptron
- Convolutional Neural Networks
- Long Short Term Memory
- Restricted Boltzman Machine
- Auto Encoders

Reinforcement Learning

Deep Architectures in Finance

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

- Classic Theorems on Compression and Model Selection
  - Minimum Description Length principle - The fundamental idea in MDL is to view learning as data compression. By compressing the data, we need to discover regularity or patterns in the data with the high potentiality to generalize to unseen samples. Information bottleneck theory believes that a deep neural network is trained first to represent the data by minimizing the generalization error and then learn to compress this representation by trimming noise.
  - Kolmogorov Complexity – Kolmogorov Complexity relies on the concept of modern computers to define the algorithmic (descriptive) complexity of an object: It is the length of the shortest binary computer program that describes the object. Following MDL, a computer is essentially the most general form of data decompressor.
  - Solomonoff's Inference Theory - Another mathematical formalization of Occam's Razor is Solomonoff's theory of universal inductive inference (Solomonoff, 1964). The principle is to favor models that correspond to the "shortest program" to produce the training data, based on its Kolmogorov complexity

Deep Architectures in Finance

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

- The expressive power of DL models - Deep neural networks have an extremely large number of parameters compared to the traditional statistical models. If we use MDL to measure the complexity of a deep neural network and consider the number of parameters as the model description length, it would look awful. The model description can easily grow out of control. However, having numerous parameters is necessary for a neural network to obtain high expressivity power. Because of its great capability to capture any flexible data representation, deep neural networks have achieved great success in many applications.

    - **Universal Approximation Theorem** - The Universal Approximation Theorem states that a feedforward network with: 1) a linear output layer, 2) at least one hidden layer containing a finite number of neurons and 3) some activation function can approximate any continuous functions on a compact subset of to arbitrary accuracy. The theorem was first proved for sigmoid activation function (Cybenko, 1989). Later it was shown that the universal approximation property is not specific to the choice of activation (Hornik, 1991) but the multilayer feedforward architecture.

    - **Stochastic processes**

Deep Architectures in Finance – Convergence Results

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

Theorem (Universal approximation theorem (Hornik, Stinchcombe and White))

Let $\mathcal{NN}^{\sigma}_{d_0,d_1}$ be the set of neural networks with activation function $\sigma : \mathbb{R} \mapsto \mathbb{R}$, input dimension $d_0 \in \mathbb{N}$ and output dimension $d_1 \in \mathbb{N}$. Then, if σ is continuous and non-constant, $\mathcal{NN}^{\sigma}_{d_0,1}$ is dense in $L^p(\mu)$ for all finite measures μ.

Theorem (Universal approximation theorem for derivatives (Hornik, Stinchcombe and White))

Let $F^* \in \mathcal{C}^n$ and $F : \mathbb{R}^{d_0} \to \mathbb{R}$ and $\mathcal{NN}^{\sigma}_{d_0,1}$ be the set of single-layer neural networks with activation function $\sigma : \mathbb{R} \mapsto \mathbb{R}$, input dimension $d_0 \in \mathbb{N}$ and output dimension 1. Then, if the (non-constant) activation function is $\sigma \in \mathcal{C}^n(\mathbb{R})$, then $\mathcal{NN}^{\sigma}_{d_0,1}$ arbitrarily approximates f and all its derivatives up to order n.

- Note that one should use at least C^1 activation functions to approximate derivatives

Deep Architectures in Finance

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

- Deep Learning and Overfitting (1)
  - **Modern risk curve for Deep Learning**
  - **Regularization and Generalization error -** Regularization is a common way to control overfitting and improve model generalization performance. Interestingly some research (Zhang, et al. 2017) has shown that explicit regularization (i.e. data augmentation, weight decay and dropout) is neither necessary or sufficient for reducing generalization error.
  - **Intrinsic Dimension** (Li et al, 2018). Intrinsic dimension is intuitive, easy to measure, while still revealing many interesting properties of models of different sizes. One intuition behind the measurement of intrinsic dimension is that, since the parameter space has such high dimensionality, it is probably not necessary to exploit all the dimensions to learn efficiently. If we only travel through a slice of objective landscape and still can learn a good solution, the complexity of the resulting model is likely lower than what it appears to be by parameter-counting. This is essentially what intrinsic dimension tries to assess.

Deep Architectures in Finance

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

- Deep Learning and Overfitting (2)

  - **Heterogenous layer robustness** - Zhang et al. (2019) investigated the role of parameters in different layers. The fundamental question raised by the paper is: “are all layers created equal?” The short answer is: No. The model is more sensitive to changes in some layers but not others. Layers can be categorized into two categories with the help of these two operations:
    - Robust Layers: The network has no or only negligible performance degradation after **re-initializing or re-randomizing the layer.**
    - Critical Layers: Otherwise.

  - **Lottery ticket hypothesis** - The lottery ticket hypothesis (Frankle & Carbin, 2019) is another intriguing and inspiring discovery, supporting that only a subset of network parameters have impact on the model performance and thus the network is not overfitted. The lottery ticket hypothesis states that a randomly initialized, dense, feed-forward network contains a pool of subnetworks and among them only a subset are “winning tickets” which can achieve the optimal performance when trained in isolation.

Deep Architectures in Finance

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

- Deep Learning and Optimization
  - Convexity, even locally, cannot be the basis of analysis for over-parameterized systems
  - But what mathematical property encapsulates ability to optimize by gradient descent for landscapes. It turns out that a simple condition proposed in 1963 by Polyak is sufficient for eficient minimization by gradient descent.
  - This PL-condition for Polyak and also Lojasiewicz, who independently analyzed a more general version of the condition in a different context is a simple first order inequality applicable to a broad range of optimization problems.

We say that $\mathcal{L}(\mathbf{w})$ is μ-PL, if the following holds:

$$\frac{1}{2}\|\nabla\mathcal{L}(\mathbf{w})\|^2 \geq \mu(\mathcal{L}(\mathbf{w}) - \mathcal{L}(\mathbf{w}^*)),$$

Here $\mathbf{w}^*$ is a global minimizer and $\mu > 0$ is a fixed real number. The original Polyak's work [74] showed that PL condition within a sufficiently large ball (with radius $O(1/\mu)$) implied convergence of gradient descent.

Bias – Variance Trade - OFF

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

Suppose we have t a model $\hat{f}(x)$ to some training data Tr, and let $(x_0; y_0)$ be a test observation drawn from the population. If the true model is $Y = f(X) + \epsilon\ (with\ f(x) = E(Y|X = x))$,
then

$$E\left(y_0 - \hat{f}(x_0)\right)^2 = Var(\hat{f}(x_0)) + [Bias(\hat{f}(x_0))]^2 + Var(\epsilon)$$

The expectation averages over the variability of y_0 as well as the variability in Tr.

Note that $Bias(\hat{f}(x_0) = E\left[\hat{f}(x_0)\right] - f(x_0)$.

Typically as the flexibility of $\hat{f}$ increases, its variance increases, and its bias decreases. So choosing the flexibility based on average test error amounts to a bias-variance trade-o.

Double Descent Risk Curve

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

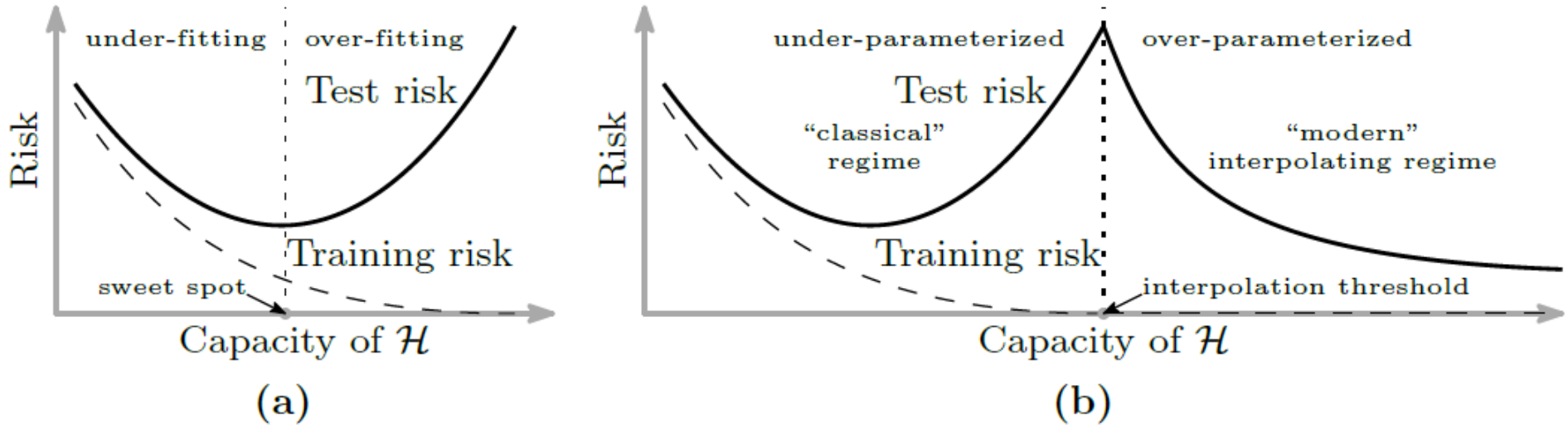


Figure 1: **Curves for training risk (dashed line) and test risk (solid line).** (**a**) The classical *U-shaped risk curve* arising from the bias-variance trade-off. (**b**) The *double descent risk curve*, which incorporates the U-shaped risk curve (i.e., the "classical" regime) together with the observed behavior from using high capacity function classes (i.e., the "modern" interpolating regime), separated by the interpolation threshold. The predictors to the right of the interpolation threshold have zero training risk.

Deep Architectures – Empirical Findings

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

1. Zero training error on random labels: Zero empirical risk can also be achieved for random labels using the same architecture and training scheme with only slightly increased training time.
2. Lack of explicit regularization: The test error depends only mildly on explicit regularization like norm-based penalty terms or dropout.
3. Dependence on the initialization: The same NN trained to zero empirical risk starting from different initializations can exhibit different test errors: This indicates that properties of the local minimum at fs to which gradient descent converges might be correlated with its generalization.
4. Interpolation of noisy training data: One still observes low test error when training up to approximately zero empirical risk using a regression (or surrogate) loss on noisy training data. This is particularly interesting, as the noise is captured by the model but seems not to hurt generalization performance.
5. Further over-parametrization improves generalization performance: Further increasing the NN size can lead to even lower test error:

This suggests that the generalization performance of NNs depends on an interplay of the data distribution PZ , the real distribution and combined with properties of the learning algorithm. A.

Inductive Biases Machine Learning

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

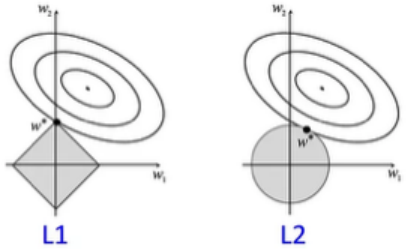
Modelers use rich prior knowledge about the world in order to efficiently learn new concepts. These priors - also known as ”inductive biases” - pertain to the space of internal models considered by a learner, and they help the learner make inferences that go beyond the observed data An inductive bias is a property that allows a learning algorithm to prioritize one solution (or interpretation) over another, independent of the observed data.Inductive biases can express assumptions about either the data-generating process or the space of solutions.

- Maximum conditional independence: if the hypothesis can be cast in a Bayesian framework, try to maximize conditional independence. This is the bias used in the Naive Bayes classifier.
- Minimum cross-validation error: when trying to choose among hypotheses, select the hypothesis with the lowest cross-validation error.
- Maximum margin: when drawing a boundary between two classes, attempt to maximize the width of the boundary. This is the bias used in support vector machines. The assumption is that distinct classes tend to be separated by wide boundaries.
- Minimum description length: when forming a hypothesis, attempt to minimize the length of the description of the hypothesis. Minimum features: unless there is good evidence that a feature is useful, it should be deleted. This is the assumption behind feature selection algorithms.
- Nearest neighbors: assume that most of the cases in a small neighborhood in feature space belong to the same class.

Deep Learning Inductive Biases

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

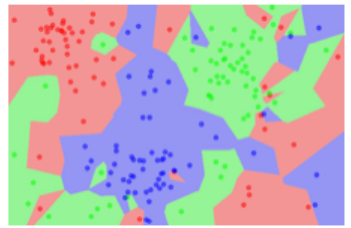
Inductive biases encode our knowledge and assumptions about the world



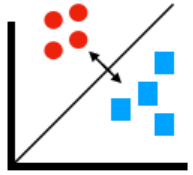
Regularization
Occam's Razor

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Bayesian Models
Prior Belief

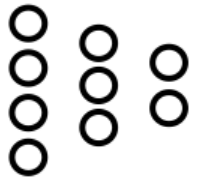


k-Nearest Neighbors
Smoothness



Max-Margin Methods
Inter-class distance

Low-Dimensional Representations
Manifold Hypothesis



Hierarchical Models
Abstraction

Deep Learning Inductive Biases

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

Component	Entities	Relations	Rel. inductive bias	Invariance
Fully connected	Units	All-to-all	Weak	-
Convolutional	Grid elements	Local	Locality	Spatial translation
Recurrent	Timesteps	Sequential	Sequentiality	Time translation
Graph network	Nodes	Edges	Arbitrary	Node, edge permutations

Table 1: Various relational inductive biases in standard deep learning components.

Deep Architectures in Finance – Pros/cons

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

- Pros
  - State of the art results in factor models, time series, classification
  - Deep Reinforcement Learning
  - XGBoost as a competing model
- Cons
  - Non Stationarity – Out of Distribution Events – Causality / Structure
  - Interpretability
  - Overfitting ?
  - Enough data?

Deep Learning in Finance

Time Series

Deep Learning for Equity Time Series Prediction
Joint work with Aymeric Moulin and Gilberto Batres

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

Abstract

We examine the performance of Deep Learning methods applied to equity financial time series. Predicting equity time series is a crucial topic in Finance. To form equity portfolios and do asset allocation, we need to predict returns, compute their risk, and optimize market impact. One of the modeling benefits of Deep Learning architectures is the ability to model non-linear highly dimensional problems. The lack of transparency and a rigorous mathematical theory could be considered less positive sides. The fact that most progress in Deep Learning has been made by trial and error is also cumbersome. Equity financial time series is a challenging domain with some stylized facts: weak stationarity, fat tails in return distributions, small data sets compared to other areas of Artificial Intelligence (AI), slow decay of autocorrelation in returns, and volatility clustering, to name the most important ones. We perform a comparative study between Long ShortTerm Memory Networks (LSTM), Recurrent Neural Networks (RNN), Deep Feed-Forward neural networks (DNN), and Gated Recurrent Unit Networks (GRU). We perform two types of studies. The first focused on a univariate test, and the second a multivariate test. Our tests show that the LSTM performs the best compared to other Deep Learning and classical machine learning models. In terms of performance metrics, the LSTM is better than the baseline model. We also show that the predictions are better than chance. There is enough evidence thatRNN and LSTM can deal with stationary time series and learn the data generating process. Nevertheless, predicting equity non-stationary time series, with market developments like the one caused by the COVID-19 pandemic in 2020, is challenging.

Long Short Term Memory Networks

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

The Long Short-Term Memory Network (LSTM) is a set of subnets with recurrent connections, known as memory blocks. Each block contains one or more self-connected memory cells and three multiplicative units known as the input, output and forget gates which respectively support read, write, and reset operations for the cells [14]. The gating units control the gradient-flow through the memory cell and when closing them allows the gradient pass without alteration for an indefinite amount of time, making the LSTM suitable for learning long time dependencies. Thus overcoming the vanishing gradient problem that RNNs suffer from. We describe in more detail the inner workings of an LSTM cell. The memory cell is composed of an input node, an input gate, an internal state, a forget gate and an output gate. First let bold letters denote vectors, each defined at time step t, then the components in a memory cell are as follows:

- The input node takes the activation from both the input layer, $\boldsymbol{x}_t$, and the hidden state $\boldsymbol{h}_{t-1}$ at time $t-1$. The input is then fed to an activation function, either a tanh or sigmoid.
- The input gate uses a sigmoidal unit that get its input from the current data $\boldsymbol{x}_t$ and the hidden units at time step $t-1$. The input gate multiplies the value of the input node, and because it is a sigmoid unit with range between zero and one, it can control the flow of the signal it multiplies.
- The internal state has a self-recurrent connection with unit weight also called the constant error carousel in [18], and is given by $\boldsymbol{s}_t = \boldsymbol{g}_t \odot \boldsymbol{i}_t + \boldsymbol{f}_t \odot \boldsymbol{s}_{t-1}$. The Hadamard product $\odot$, denotes element-wise product, and $\boldsymbol{f}_t$ is the forget gate, see below.
- The forget gate, $\boldsymbol{f}_t$, was not part of the original model for the LSTM but was introduced by [11]. The forget gate multiplies the internal state at time step $t-1$ and can in that way get

Long Short Term Memory Networks

rid of all the contents in the past as demonstrated by the equation in the list item above.

- The resulting output from a memory cell is produced by multiplying the value of the internal state s_c by the output gate o_c. Often the internal state is run through a tanh activation function.

The equations for the LSTM network can be summarized as follows. As before let $\boldsymbol{g}$ stand for the input to the memory cell, $\boldsymbol{i}$ for the input gate, $\boldsymbol{f}$ for the forget gate, $\boldsymbol{o}$ for the output gate and $\boldsymbol{s}$ for the cell state, then we have, [21]

$$\boldsymbol{g}_t = \phi(W^{gX}\boldsymbol{x}_t + W^{gh}\boldsymbol{h}_{t-1} + \boldsymbol{b}_g) \tag{13}$$
$$\boldsymbol{i}_t = \sigma(W^{iX}\boldsymbol{x}_t + W^{ih}\boldsymbol{h}_{t-1} + \boldsymbol{b}_i) \tag{14}$$
$$\boldsymbol{f}_t = \sigma(W^{fX}\boldsymbol{x}_t + W^{fh}\boldsymbol{h}_{t-1} + \boldsymbol{b}_f) \tag{15}$$
$$\boldsymbol{o}_t = \sigma(W^{oX}\boldsymbol{x}_t + W^{oh}\boldsymbol{h}_{t-1} + \boldsymbol{b}_o) \tag{16}$$
$$\boldsymbol{s}_t = \boldsymbol{g}_t \odot \boldsymbol{i}_t + \boldsymbol{s}_{t-1} \odot \boldsymbol{f}_t \tag{17}$$
$$\boldsymbol{h}_t = \phi(\boldsymbol{s}_t) \odot \boldsymbol{o}_t. \tag{18}$$

where the Hadamard product $\odot$ denotes element-wise multiplication. In the equations $\boldsymbol{h}_t$ is the value of the hidden layer at time t, while $\boldsymbol{h}_{t-1}$ is the output by each memory cell in the hidden layer at time $t-1$. The weights $\{W^{gX}, W^{iX}, W^{fX}, W^{oX}\}$ are the connections between the inputs $\boldsymbol{x}_t$ with the input node, the input gate, the forget gate and the output gate respectively. In the same manner $\{W^{gh}, W^{ih}, W^{fh}, W^{oh}\}$ represent the connections between the hidden layer with the input node, the input gate, the forget gate and the output gate respectively. The bias terms for each of the cell's components is given by $\{\boldsymbol{b}_g, \boldsymbol{b}_i, \boldsymbol{b}_f, \boldsymbol{b}_o\}$.

Long Short Term Memory Networks

AIFI
ARTIFICIAL INTELLIGENCE
IN FINANCE INSTITUTE

These ratios show a good behavior in-sample and out-of-sample. Sharpe ratios of our portfolio experiments are 8 for the long only portfolio and 10 for the long short version, an equally weighted portfolio would have provided a 2.7 Sharpe ratio using the model from 2014 to 2017. Results show consistency when using the same modeling approach in different market regimes. No trading costs have been considered.

We can conclude that LSTM networks are a promising modeling tool in financial time series especially in the multivariate LSTM networks with exogeneous variables. These networks can enable financial engineers to model time dependencies, non-linearity, feature discovery with a very flexible model that might be able to offset the very challenging estimation and non-stationarity in finance and the potential of overfitting. These issues can never be underestimated in finance even more in models with a high number of parameters, non-linearity and difficulty to interpret like LSTM networks.
We think financial engineers should then incorporate deep learning not only to model unstructured but also to structured data. We have very interesting modeling times ahead of us.

Long Short Term Memory Networks

We train one model for each stock. The
data was divided into three portions, 75% train set, 12:5% validation set, and 12:5% test set.
We use data for those stocks where data is available from 1980-01-01 to 2020-07-31.

Stock	Units	Activation	Dropout	Learning-rate	Hit-Ratio Test Set	MSE Test Set	MAE Test Set
XOM	90	elu	0.2	0.010	0.50	0.000303	0.011165
WMT	170	relu	0.4	0.001	0.53	0.000204	0.008870
WFC	240	selu	0.0	0.001	0.49	0.000420	0.012740
VZ	110	relu	0.1	0.001	0.52	0.000150	0.008539
V	100	relu	0	0.001	0.56	0.000510	0.013884
UNH	180	relu	0.0	0.010	0.53	0.000329	0.011101
T	130	relu	0.0	0.010	0.53	0.000218	0.009562
PG	150	relu	0.0	0.010	0.53	0.000164	0.008031
PFE	140	relu	0.4	0.001	0.51	0.000198	0.009459
PEP	80	relu	0.4	0.001	0.54	0.000171	0.007837
PM	190	elu	0	0.001	0.54	0.000496	0.014209
ORCL	130	relu	0.0	0.010	0.55	0.000285	0.009782
NVDA	120	elu	0.0	0.001	0.54	0.001009	0.022151
MSFT	160	relu	0.0	0.010	0.56	0.000299	0.010959
MRK	30	relu	0.0	0.010	0.52	0.000206	0.009721
MO	230	elu	0.0	0.010	0.53	0.000238	0.010355
MMM	190	selu	0.0	0.001	0.54	0.000236	0.009748
MDT	110	relu	0.4	0.001	0.53	0.000234	0.009770
MCD	220	selu	0.0	0.001	0.55	0.000221	0.008648
MA	190	relu	0	0.01	0.58	0.000611	0.015243
KO	120	relu	0.3	0.010	0.52	0.000153	0.007801
JPM	210	selu	0.0	0.001	0.52	0.000351	0.011632
JNJ	140	relu	0.1	0.001	0.52	0.000160	0.007981
INTC	200	relu	0.0	0.010	0.54	0.000427	0.012986
IBM	160	relu	0.0	0.010	0.51	0.000261	0.010298
HON	190	relu	0.4	0.010	0.55	0.000244	0.009587
HD	110	relu	0.0	0.010	0.53	0.000264	0.009960
GSPC	50	elu	0	0.001	0.53	0.000151	0.007160
GOOG	170	selu	0	0.001	0.51	0.000444	0.013833
GILD	180	relu	0.0	0.001	0.51	0.000316	0.009960
GE	210	relu	0.4	0.010	0.46	0.000592	0.009960
FB	180	relu	0	0.001	0.57	0.000712	0.017606
DIS	100	relu	0.1	0.010	0.51	0.000274	0.010371
CVX	30	selu	0.0	0.010	0.52	0.000424	0.012226
CSCO	50	relu	0.3	0.001	0.54	0.000320	0.011435
CMCSA	160	relu	0.0	0.010	0.53	0.000253	0.010970
C	50	relu	0.4	0.010	0.50	0.000539	0.014319
BMY	110	relu	0.2	0.010	0.53	0.000290	0.011392
BAC	160	relu	0.0	0.010	0.51	0.000470	0.014114
BA	80	relu	0.0	0.010	0.52	0.000756	0.015532
BRK-B	80	relu	0.2	0.001	0.52	0.000255	0.009844
AVGO	40	relu	0	0.001	0.53	0.000927	0.019627
AMZN	180	relu	0.0	0.010	0.56	0.000432	0.014260
AMGN	150	relu	0.0	0.010	0.54	0.000262	0.010848
ABT	40	relu	0.0	0.001	0.53	0.000256	0.010783
ABBV	160	relu	0	0.01	0.56	0.000493	0.014903
AAPL	80	relu	0.1	0.010	0.54	0.000334	0.012059

Table I Single stock predictions: Results for LSTM, the performance is measured as the Hit-Ratio (HR), Mean Squared Error (MSE) and Mean Absolute Error (MAE). The results are computed for the independent test set.

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

Long Short Term Memory Networks

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

Start	End	Sharpe Train Set	Sharpe Validation Set	Sharpe Test Set
2015-07-31	2020-07-31	3.8 (L)/5.2 (LS)	2.8 (L)/4.2 (LS)	1.7 (L)/2.3 (LS)
2014-09-01	2019-09-01	3.7 (L)/4.6 (LS)	2.0 (L)/2.2 (LS)	0.6 (L)/1.8 (LS)
2013-09-01	2018-09-01	4.2 (L)/5.0 (LS)	4.0 (L)/5.2 (LS)	0.7 (L)/-0.7 (LS)
2012-09-01	2017-09-01	4.7(L)/5.5(LS)	4.1 (L)/5.9 (LS)	2.0 (L)/1.9 (LS)
2011-09-01	2016-09-01	4.3(L)/5.0(LS)	2.1 (L)/3.2 (LS)	1.0 (L)/1.2 (LS)
2010-09-01	2015-09-01	3.8(L)/4.9(LS)	3.1 (L)/3.2 (LS)	1.1 (L)/0.9 (LS)

Table II Sharpe Ratio (SR) for train, validation and test sets. In each of SR-column the (L) stands for Long portfolio (L) an (LS) for Long-Short portfolio.

To assess the quality of the prediction, we use conventional MAE and MSE. We also use the hit-ratio (HR) for computing the ratio for the number of times the model is right in its predictions.
We build a straightforward strategy using HR prediction accuracy. The long-short strategy consists of updating a portfolio daily by holding a long position on stocks where we predict a positive return and a short position on stocks where we predict a negative return (or long position and at
position for the long-only strategy).

Deep Learning for Equity Time Series Prediction - Conclusion

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

We performed two case studies, one focused on predicting stock returns. Here we consider the prediction task as a univariate problem. The second study focuses on predicting the returns as a multivariate problem. In the univariate tests, we study the performance of Long Short Term Memory Networks (LSTM) compared to some of the most common machine learning algorithms, such as Support Vector Machines (SVM), Recurrent Neural Networks (RNN), and Gated Recurrent Units (GRU). Our study suggests that LSTM shows a better performance compared to the other tested models. By measuring the hit-ratio and the error metrics such as MSE and MAE, we see that LSTM achieves a better performance. The training, validation, and test periods for the univariate case can be found in the Appendix.

In the multivariate stock data experiment, we performed tests to compare LSTMs to GRUs. The GRU experiments showed worse performance than LTSM. Therefore our analysis focuses on training LSTM networks for different periods. Using a simple architecture can produce good Sharpe ratios on the long and long-short portfolios.

There is enough evidence that RNN and LSTM can deal with stationary time series and learn the data generating process. Nevertheless, predicting equity non-stationary time series, with market

Deep Learning for Equity Time Series Prediction - Conclusion

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

developments like the one caused by the Covid-19 pandemic in 2020, is challenging.

We provide an extensive examination of the performance in different periods to assess investment performance for our portfolios. By comparing the performance on the training set and the test set, we can see that overfitting is small and acceptable. Future research will involve adding exogenous variables to the experiments.

The Performance Curve in Financial Time Series

Joint work with Sonam Srivastava

Performance Curve in Financial Time Series

AIFI
ARTIFICIAL INTELLIGENCE
IN FINANCE INSTITUTE

Abstract

We examine in this paper a critical question in finance: the use of large nonlinear over-parametrized models or simpler models to forecast financial time series and the balance between underfitting and overfitting, the bias-variance trade-off, and the absolute performance in the test set. The traditional shape of the performance curve was U-shaped due to a bias-variance trade-off. Still, recently some recent research has pointed out that the performance curve may have a double descent shape in some specific domains.
We discuss some of the recent discoveries in the mathematical theory of machine learning that reduce the gap between theory and practice. We conduct experiments in the financial time series domain using deep neural networks and tree ensembles: random forests and XGBoost from under to over-parametrized.
The performance function doesn't show a U-shape or a double descent shape but a flat profile that means that larger models have the same performance in the test set than smaller models. However, the training error function shows a descent profile consistent with the idea that while training error can be very low when we increase the models' dimensionality, the test error is more stable in the equity financial time series domain. This is consistent with the finance practitioner's theory that backtesting (training data performance) frequently overestimates the test or real-life performance in financial time series prediction. The irreducible error limits the prediction performance.

Keywords: Machine Learning, Time Series, Double Descent, Artificial Intelligence, Bias Variance Tradeoff, Overfitting, Optimization

Financial Engineering Generalization Error

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

FEATURE OF FINANCIAL MODELING	POTENTIAL SOURCE OF GENERALIZATION ERROR
Data generation process	Models are not effective at data generation
	Parameters estimation may have varying level of accuracy
	Regimes, cycles and memory are not captured by the model
	Randomness is not independent and identically distributed
	The stochastic process lacks weak stationarity
Data relevance	Data may not always be relevant to measure the environment state (i.e. nowcasting and forecasting)
Data frequency	Data frequency varies, resulting in irregularity of time intervals (sparse/dense data)
OoD scenarios	Models are not able to detect and generalize OoD scenarios

Deep Architectures in Finance

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

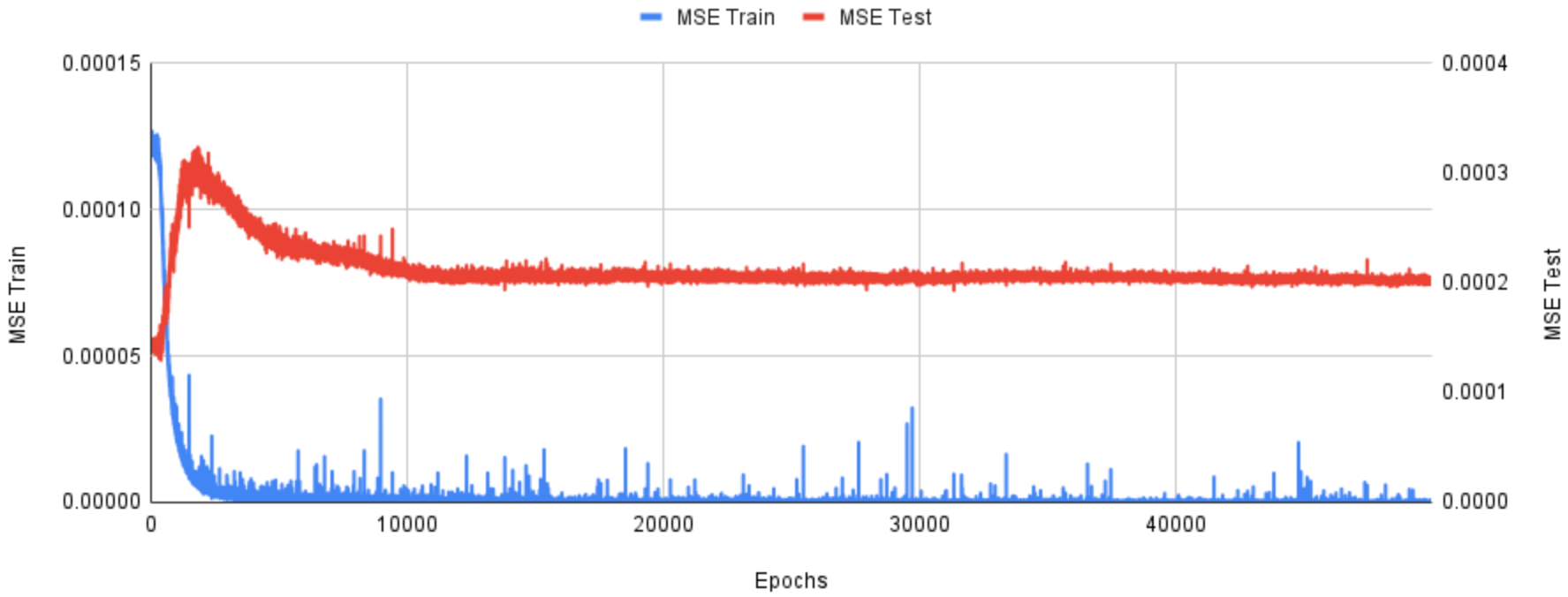
Our approach is to use different financial datasets and algorithms to construct a theory over performance curve in financial time series.
We run this experiment in supervised time series prediction using a set containing three indices:

- S&P 500 Index
- FTSE 100 Index
- MSCI Emerging Market Index

We use daily OHLC data for the experiment.
The data under consideration is from Jan 1st 2010 to Jan 1st 2019. We use 5% of the data at the end of the period as testing set and rest of the data as the training set.
We are trying to forecast the next period return using data with a lookback of 20 days.

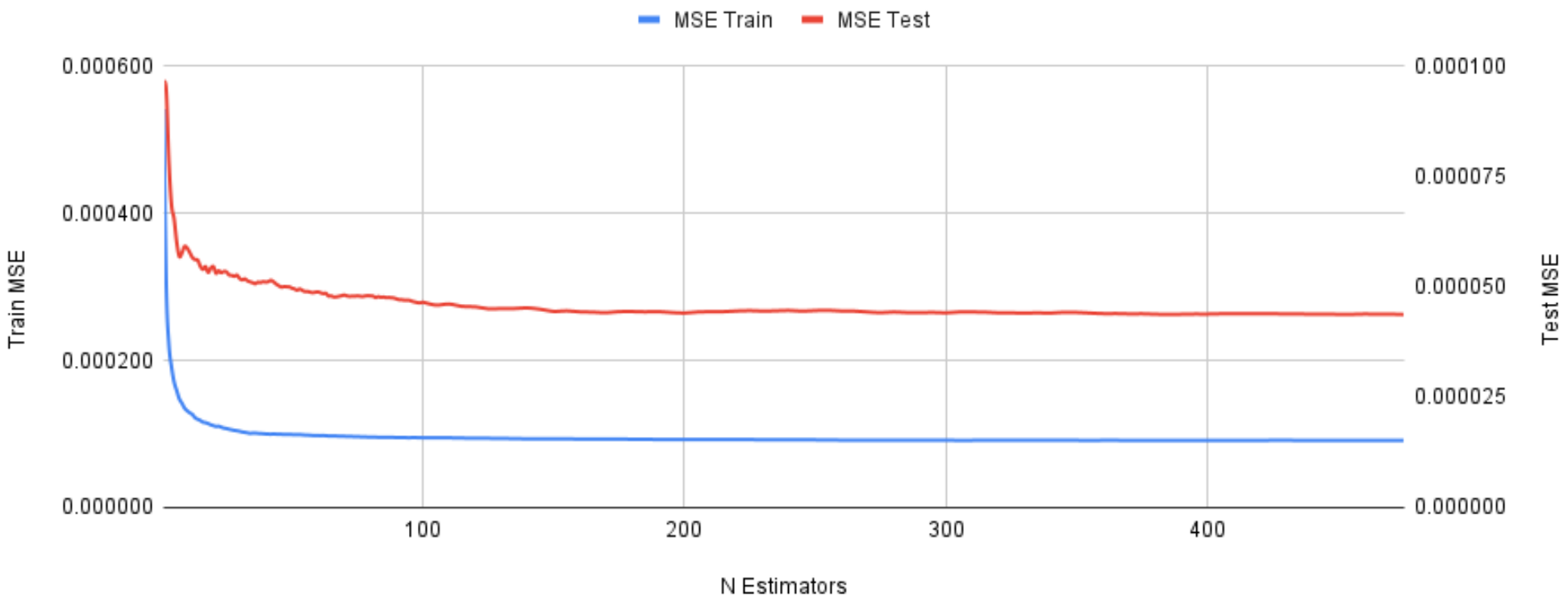
LSTM´s Performance Curve

AIFI
ARTIFICIAL INTELLIGENCE
IN FINANCE INSTITUTE



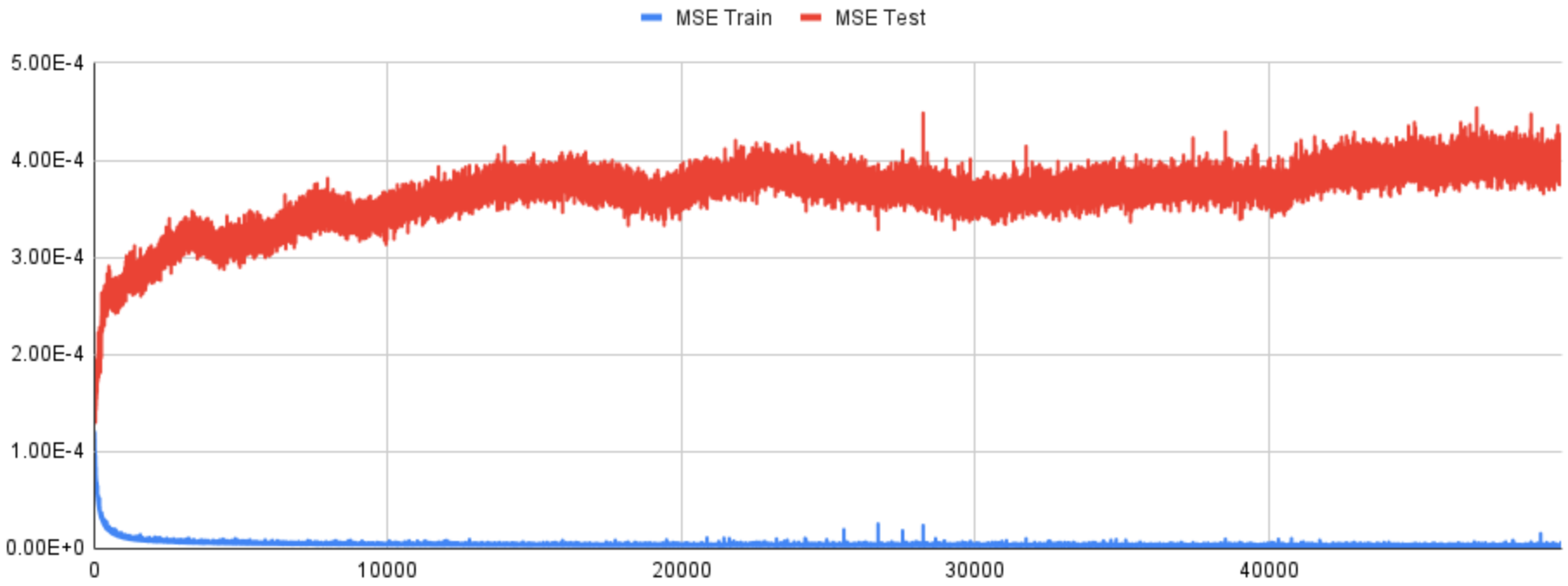
Random Forests Performance Curve

AIFI
ARTIFICIAL INTELLIGENCE
IN FINANCE INSTITUTE



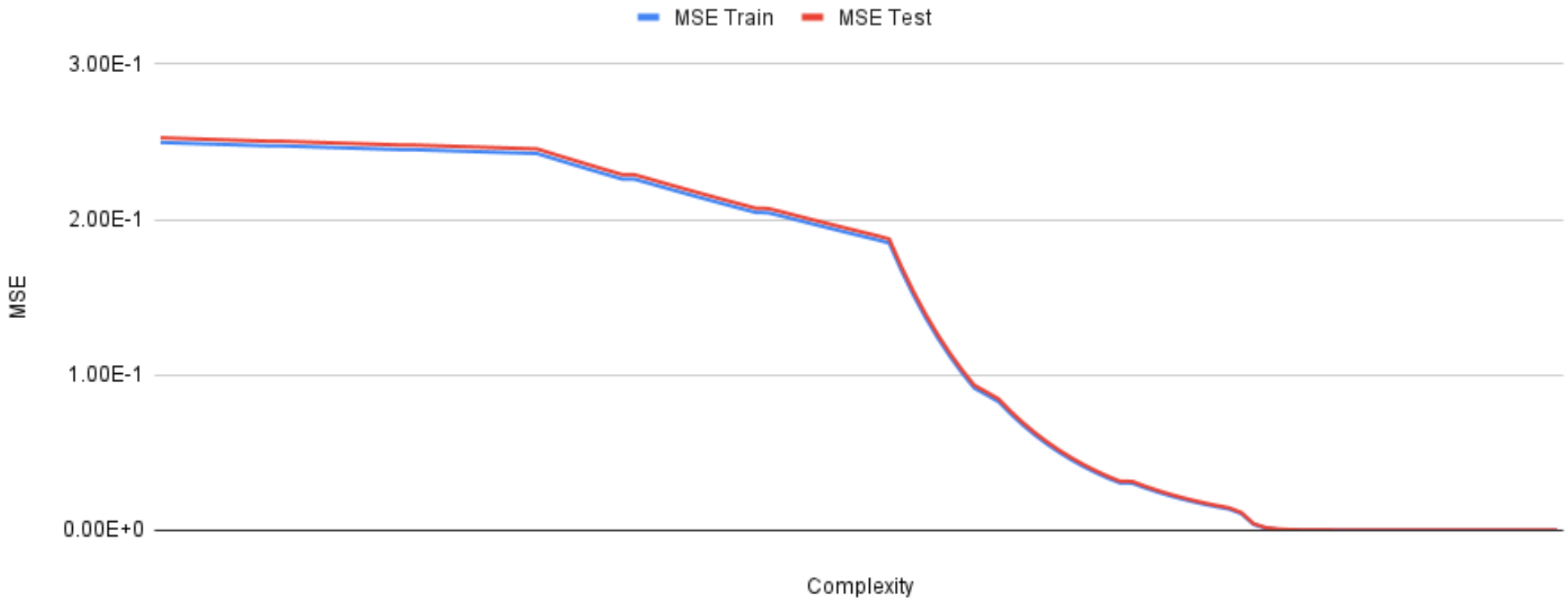
Deep Neural Networks Performance Curve

AIFI
ARTIFICIAL INTELLIGENCE
IN FINANCE INSTITUTE



XgBoost Performance Curve

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE



Performance Curve in Financial Time Series
Conclusion

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

We explored in this paper one of the biggest open questions in finance: the use of large non-linear models to forecast financial time series and the balance between underfitting and overfitting. The traditional shape of bias-variance-trade off is U-shaped. Still, recently some research has pointed out that the performance curve may have a double descent shape when using different dimensionality in the models. We conduct experiments in the financial time series domain using low-dimensional machine learning time series models, comparing them to over-parametrized deep neural networks and trees.
The performance function doesn't show a U-shape or a double descent shape but a flat profile. However, the training error function shows a descent profile consistent with the idea that while training error can be deficient when we increase dimensionality, test error is essentially more stable. This experimental result tells us that overfitting might not be or irreducible error ($Var(\epsilon)$) and the potential non-stationarity nature of financial time series.

Our papers

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

1. **Deep Learning for Equity Time Series Prediction**

Number of pages: 31 • Posted: 24 Nov 2020

Miquel Noguer i Alonso, Gilberto Batres-Estrada and Aymeric Moulin

Artificial Intelligence in Finance Institute, Trell Technologies and Jp Morgan

View PDF | Download | Show Abstract

Downloads
1,714
(13,104)

2. **Deep Reinforcement Learning for Asset Allocation in US Equities**

Number of pages: 29 • Posted: 19 Oct 2020

Miquel Noguer i Alonso and Sonam Srivastava

Artificial Intelligence in Finance Institute and Wright Research

View PDF | Download | Show Abstract

Downloads
1,429
(17,322)

3. **A Meta-Learning approach to Model Uncertainty in Financial Time Series**

Number of pages: 27 • Posted: 30 Mar 2021

Miquel Noguer i Alonso, Gilberto Batres-Estrada and Ghozlane Yahiaoui

Artificial Intelligence in Finance Institute, Trell Technologies and *affiliation not provided to SSRN*

View PDF | Download | Show Abstract

Downloads
447
(84,132)

4. **The Shape of Performance Curve in Financial Time Series**

Number of pages: 12 • Posted: 20 Dec 2021

Miquel Noguer i Alonso and Sonam Srivastava

Artificial Intelligence in Finance Institute and Wright Research

View PDF | Download | Show Abstract

Downloads
428
(88,551)

Our papers

AIFI
ARTIFICIAL INTELLIGENCE IN FINANCE INSTITUTE

1. Sustainable Investment - Exploring the Linkage between Alpha, ESG, and SDG's

Downloads 912 (31,511) | Citation 1

Number of pages: 34 • Posted: 11 Jun 2020 • Last Revised: 12 Oct 2020

Madelyn Antoncic, Geert Bekaert, Richard V Rothenberg and Miquel Noguer

Global AI Corp, Columbia Business School - Finance and Economics, Global AI Co and Global AI Corp

View PDF | Download | Show Abstract

1. arXiv:2111.00526 [pdf, other] cs.CL q-fin.CP q-fin.PM

FinEAS: Financial Embedding Analysis of Sentiment

Authors: Asier Gutiérrez-Fandiño, Miquel Noguer i Alonso, Petter Kolm, Jordi Armengol-Estapé

Abstract: We introduce a new language representation model in finance called Financial Embedding Analysis of Sentiment (FinEAS). In financial markets, news and investor sentiment are significant drivers of security prices. Thus, leveraging the capabilities of modern NLP approaches for financial sentiment analysis is a crucial component in identifying patterns and trends that are useful for market participan... ▽ More

Submitted 19 November, 2021; **v1** submitted 31 October, 2021; **originally announced** November 2021.

2. arXiv:2010.04404 [pdf, other] q-fin.PM q-fin.CP

Deep Reinforcement Learning for Asset Allocation in US Equities

Authors: Miquel Noguer i Alonso, Sonam Srivastava

Abstract: Reinforcement learning is a machine learning approach concerned with solving dynamic optimization problems in an almost model-free way by maximizing a reward function in state and action spaces. This property makes it an exciting area of research for financial problems. Asset allocation, where the goal is to obtain the weights of the assets that maximize the rewards in a given state of the market... ▽ More

Submitted 9 October, 2020; **originally announced** October 2020.

Comments: Submitting to Journal of Machine Learning in Finance

MSC Class: 91-10 **ACM Class:** I.2.m