Empirical Asset Pricing via Machine Learning

Huang Zhenyu 20744676

Luo Jiahao 20744418

Yang Yannan 20746131

Department of Mathematics, School of Science

The Hong Kong University of Science and Technology

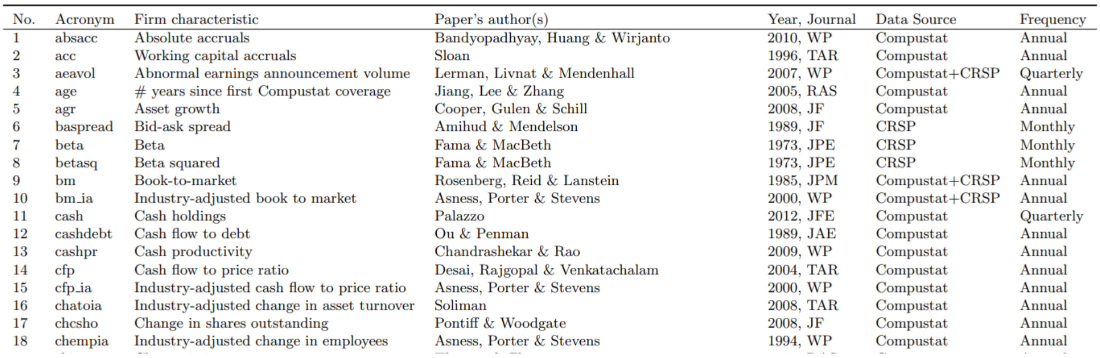
Contents

1 Introduction

Machine learning and deep learning algorithms have been utilized in financial fields like banks, hedge funds and other investment corporations for a long time. Gu, Kelly and Xiu et al. (2018) state in their paper that the deep learning algorithms can reach positive R^2 score in predicting the stock prices, which outperform traditional linear models. Here we are going to replicate what Gu et al. have done in their paper and hoping to see if applying the modern techniques in computer science to a financial context will help us analyze the predictive power of different variables.

2 Data description

The dataset we use comes from Xiu et al., which is available on their homepage. The data is mainly combined with various financial indicators collected by COMPUSTAT and CRSP. The dataset starts from 1926 and ends in 2020, covering the stocks with prices below $5, and so there are over 4,000,000 observations in it. The number of variables is 101, including 95 predictive characteristics related to stocks and 2 characteristics for calculating the market value. The 8 macroeconomic variables in Goyal and Welch are not included in our work.



No.	Acronym	Firm characteristic	Paper's author(s)	Year, Journal	Data Source	Frequency
1	absacc	Absolute accruals	Bandyopadhyay, Huang & Wirjanto	2010, WP	Compustat	Annual
2	acc	Working capital accruals	Sloan	1996, TAR	Compustat	Annual
3	aeavol	Abnormal earnings announcement volume	Lerman, Livnat & Mendenhall	2007, WP	Compustat+CRSP	Quarterly
4	age	# years since first Compustat coverage	Jiang, Lee & Zhang	2005, RAS	Compustat	Annual
5	agr	Asset growth	Cooper, Gulen & Schill	2008, JF	Compustat	Annual
6	baspread	Bid-ask spread	Amihud & Mendelson	1989, JF	CRSP	Monthly
7	beta	Beta	Fama & MacBeth	1973, JPE	CRSP	Monthly
8	betasq	Beta squared	Fama & MacBeth	1973, JPE	CRSP	Monthly
9	bm	Book-to-market	Rosenberg, Reid & Lanstein	1985, JPM	Compustat+CRSP	Annual
10	bm_ia	Industry-adjusted book to market	Asness, Porter & Stevens	2000, WP	Compustat+CRSP	Annual
11	cash	Cash holdings	Palazzo	2012, JFE	Compustat	Quarterly
12	cashdebt	Cash flow to debt	Ou & Penman	1989, JAE	Compustat	Annual
13	cashpr	Cash productivity	Chandrashekar & Rao	2009, WP	Compustat	Annual
14	cfp	Cash flow to price ratio	Desai, Rajgopal & Venkatachalam	2004, TAR	Compustat	Annual
15	cfp_ia	Industry-adjusted cash flow to price ratio	Asness, Porter & Stevens	2000, WP	Compustat	Annual
16	chatoia	Industry-adjusted change in asset turnover	Soliman	2008, TAR	Compustat	Annual
17	chcsho	Change in shares outstanding	Pontiff & Woodgate	2008, JF	Compustat	Annual
18	chempia	Industry-adjusted change in employees	Asness, Porter & Stevens	1994, WP	Compustat	Annual

Figure 1: Description of some variables in the supplementary document

2.1 Data Visualization

Figure 2 displays the distribution of data for each variable. The distribution of most variables is symmetric, and the distribution of a small number of variables is skewed, indicating that there might be some outliers in the data. We will discuss it in later section.

One effective way to understand the data is to calculate correlations among the features. Then, we can observe the correlation between the data of various variables according to the heat map diagram and counted the 10 pairs of variables with the highest absolute value of correlation to help our subsequent research. Among them, quick and currat, pchquick and pchcurrat, betasq and beta, retvol and maxret are highly correlated. Figure 3 shows the overall correlation among variables.

2.2 Data preprocessing

Since there are nearly 100 years' data, we believe there are large amount of missing data existing. We calculate the percentage of missing values of each variable every year in the original dataset. It is not surprising that the percentages of blank values are close to 1 of some variables before 1970s since many descriptive characteristics are not introduced yet in that period. So, we decide not to use the data before 1977 as we don't have much information about the stocks are markets in dataset. Therefore, we greatly reduce the size of dataset. Detailed view of the missing data can be found in figure 4.

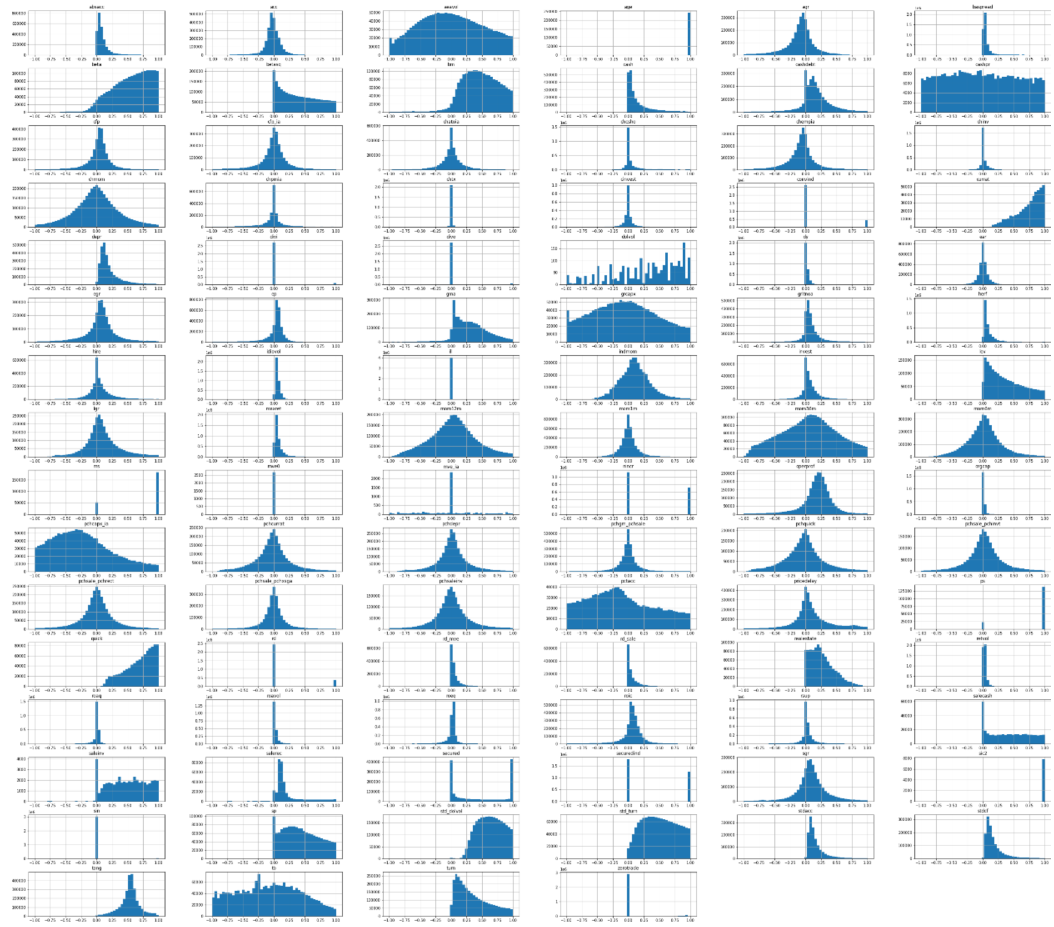


Figure 2: Distribution of all variables

Although we select the years that contain fewer missing values in each variable, we must handle the missing values first before modeling. Xiu et al. suggests we should use the cross-sectional median in each month for each stock in imputation. However, there are still some blanks that cannot be filled in as we fail to find any existing value in one month for certain stocks. As a result, we decide to drop these stocks to avoid introducing noise. Moreover, Xiu et al. in their work give the variable importance of all financial characteristics, according to which we drop 5 more variables since they have over 75% of missing values in one year on average and rank bottom in the importance chart. Now we use the data only from 1977 to 2018, the length of which is exactly 70% of the data period used in the paper to be replicated.

It is recommended that we need to standardize the variables before model training, especially when we use linear models. Distribution of the variables is shown in figure 2 and we mainly use 2 methods to transform all these numerical characteristics. One is standard scaler, which transforms the original variables to a set of numbers with 0 mean and unit standard deviation. The other is robust scaler, which is utilized whenever there are some outliers or extreme values in our data, like the variable zerotrade.

Besides, when we look further into the data, we find some inner connection between variables sic2, the categorial code and permno, representing the stocks. We believe the category of each stock remain stable in our dataset, so we just impute the missing sector by each stock and throw away stocks with unknown sector instead of regarding it as 0 or other number, since it may introduce new information that will influence the model and result.

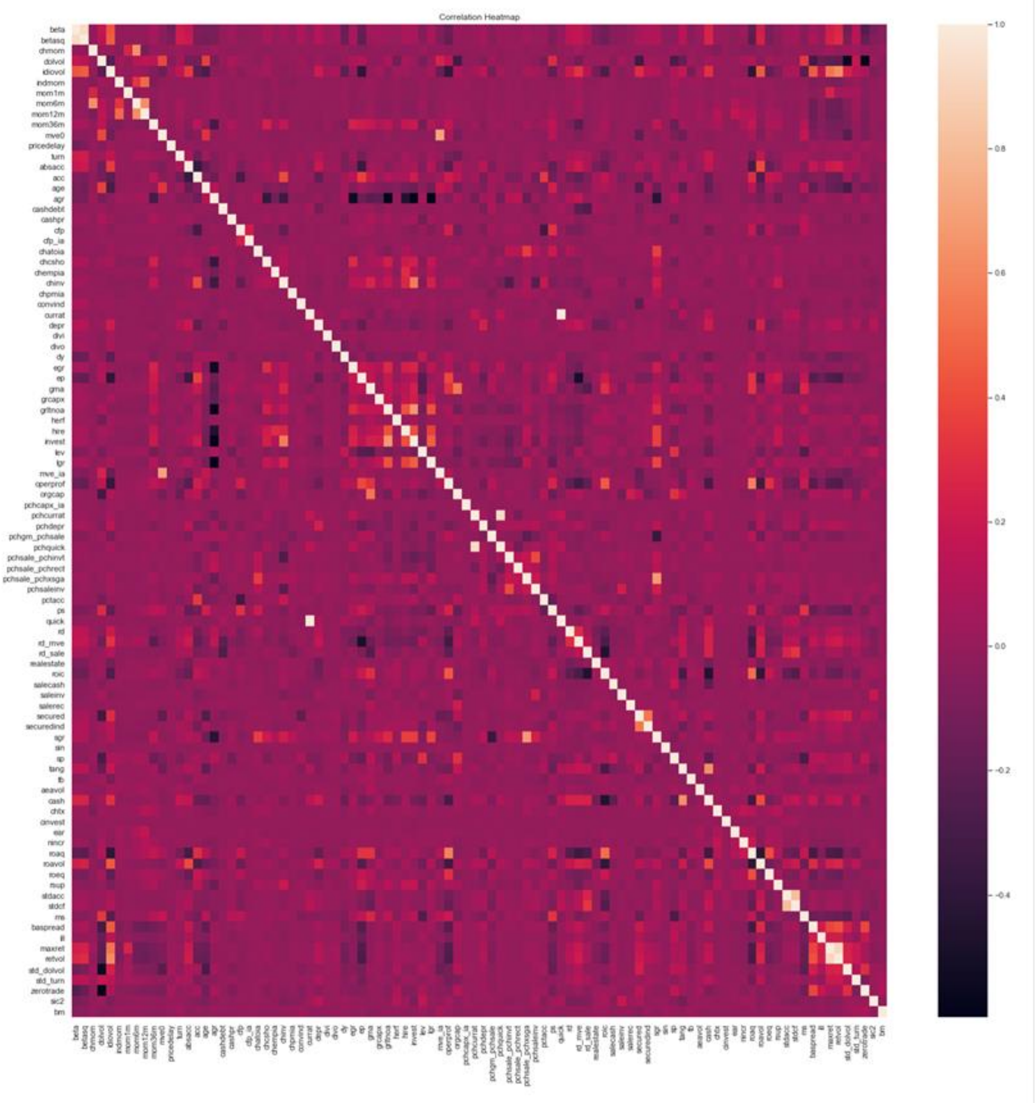


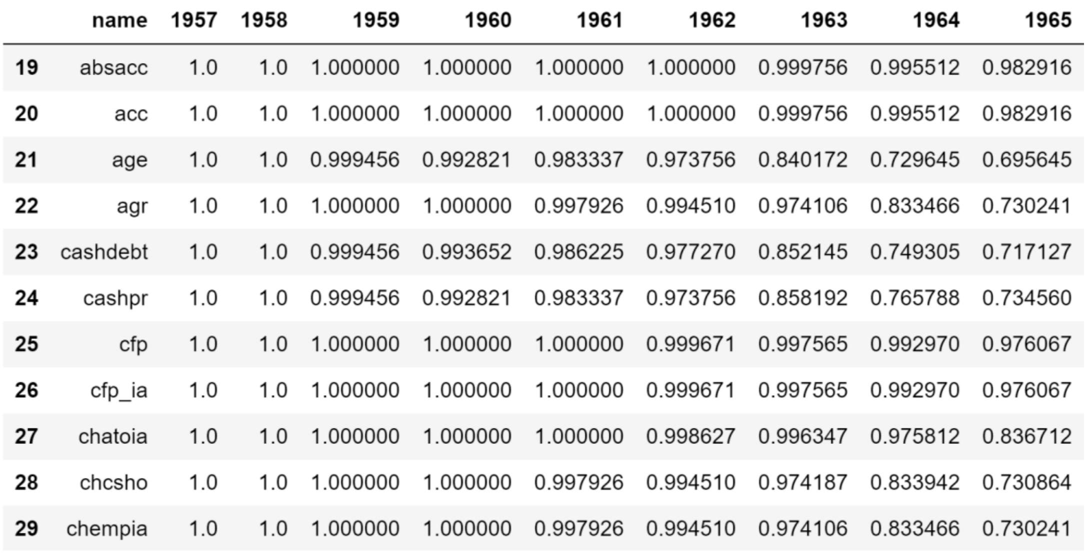
Figure 3: correlation heatmap of all variables

We are now going to work on the remaining dataset with roughly 3,200,000 observations and 89 variables for our analysis. Remember we have already rescaled the variables so that we can train out model in an easier way.

3 Methodology

3.1 Dataset splitting and performance evaluation

We first divide our full dataset into training, validating and testing samples by general machine learning literature. Following the construction in the paper, we first divide 30% of the data into training sample, 20% into validation and the remaining half into testing set. That is, 13 years for training, 8 years for validation and 21 years for testing at the beginning. We adopt the recursive training by Gu et al. and increase the training data by one year while keeping the period of validation years unchanged. In total we train 20 times for each model. Cross validation is not suitable in our analysis since it may take future information into consideration, but it is unknown until we reach the latest time stamp.



	name	1957	1958	1959	1960	1961	1962	1963	1964	1965
19	absacc	1.0	1.0	1.000000	1.000000	1.000000	1.000000	0.999756	0.995512	0.982916
20	acc	1.0	1.0	1.000000	1.000000	1.000000	1.000000	0.999756	0.995512	0.982916
21	age	1.0	1.0	0.999456	0.992821	0.983337	0.973756	0.840172	0.729645	0.695645
22	agr	1.0	1.0	1.000000	1.000000	0.997926	0.994510	0.974106	0.833466	0.730241
23	cashdebt	1.0	1.0	0.999456	0.993652	0.986225	0.977270	0.852145	0.749305	0.717127
24	cashpr	1.0	1.0	0.999456	0.992821	0.983337	0.973756	0.858192	0.765788	0.734560
25	cfp	1.0	1.0	1.000000	1.000000	1.000000	0.999671	0.997565	0.992970	0.976067
26	cfp_ia	1.0	1.0	1.000000	1.000000	1.000000	0.999671	0.997565	0.992970	0.976067
27	chatoia	1.0	1.0	1.000000	1.000000	1.000000	0.998627	0.996347	0.975812	0.836712
28	chcsho	1.0	1.0	1.000000	1.000000	0.997926	0.994510	0.974187	0.833942	0.730864
29	chempia	1.0	1.0	1.000000	1.000000	0.997926	0.994510	0.974106	0.833466	0.730241

Figure 4: missing value percentage of some variables in some years

According to Shihao et al.'s discussion, the general form of an asset's excess return can be defined as

$$r_{i,t+1} = E_t\left(r_{i,t+1}\right) + \epsilon_{i,t+1}$$

where

$$E_t\left(r_{i,t+1}\right) = g^*\left(z_{i,t}\right)$$

The performance of our model is judged by out-of-sample R^2 score defined as

$$R^2_{oos} = 1 - \frac{\sum_{(i,t\in\tau_3)}\left(r_{i,t+1} - \hat{r}_{i,t+1}\right)^2}{\sum_{(i,t\in\tau_3)} {r_{i,t+1}}^2}$$

They further consider that using past average return and information to predict future return will surely underestimate the prediction value since noise play a part in the past data. So, they compute the percentage R^2 by comparing the one given by model prediction and the one obtained via treating the prediction as 0. We adopt the method, but the result seems not that satisfactory.

3.2 Linear models

3.2.1 Simple linear

Linear regression is a simple method that can use a 'straight line' or hyperplane to describe the relationship between response variable and characteristics. It is still popular in real world despite those more complex methods give better results mainly because it can give clear explanation to the relationship between the 2 kinds of variables. The loss function, which is generally least squares, controls the estimation error between the ground true value and prediction. But as we mentioned in section 2.2, there seems to be outliers in our dataset, so we adopt Huber loss function to give better prediction. Huber loss function is a kind of robust method that is less sensitive to outliers in data than the squared error loss. It is defined as

$$H(x;\xi) = \begin{cases} x^2, & if\ |x| \leq \xi \\ 2\xi|x| - \xi^2, & if |x| > \xi \end{cases}$$

In our model, we choose the parameter epsilon = 1.1 to train a more robust model, where the default epsilon = 1.35 reach a quantile of 95%.

Elastic Net is another regularization method that uses the combination of $l1$ and $l2$ penalties to cope with overfitting problems. It adopts the advantages in both Ridge and Lasso regression. We choose the parameter ρ=0.5 to introduce the 2 ways of regularization.

3.2.2 Principle component methods

Despite tuning loss function, we also try principal component regression (PCR) and partial least squares (PLS) which are useful in dimension reduction. PCR combines principal component analysis (PCA) that can best describes the dataset with a smaller number of features and one linear regression. It helps to avoid overfitting, but the dimension reduction and regression are not working simultaneously. Different from PCR, PLS is a supervised learning method whose purpose is to find directions that can explain the independent variables as well as the dependent variables. It is a regression modeling method of transforming multiple dependent variables to multiple independent variables so that it is easy to explain the regression coefficients of independent variables as well as identifying the noise.

3.3 Tree based methods

Regression trees is a popular method in machine learning. In a decision tree, each internal node represents a judgment on an attribute, each branch represents an output of a judgment result, and finally the mean of each leaf node represents a regression result. Randomforest regression contains multiple decision trees, and the output is determined by the mean of each the individual tree. It is capable of handling high-dimensional data and does not need much parameter tuning and feature selection. It also performs well in dealing with outliers and overfitting issues. In regression task, it is suggested that the number of features in each splitting to be around one-third of the total number of features, which is 30 in our work.

Gradient Boosting Regression Tree (GBRT) is adapted from gradient descent method and aims at improving learning methods by combining many weaker learners, which are created sequentially, in attempt to produce a strong learner. In GBRT, each tree learns from the residuals of all previous trees with the calculation of negative gradient to update estimation.

3.4 Neural network

Neural network consists of three parts: input layer, hidden layer and output layer. The number of hidden layers is not fixed. It may be 0 for simple task but may also be hundreds or thousands in complicated methods. The nodes of each layer in the model are called "neurons". The neurons located in the input layer correspond to the characteristics of the training data. The neurons in the hidden layer and the output layer are expressed by the activation function, which is ReLU according to the paper. The model incorporates more flexible predictive associations by adding hidden layers between the inputs and output. As the depth of the network increases and the number of nodes at each layer increases, the expressive ability of the network can be strengthened. The higher the complexity of the network, the stronger the expressive ability and the more complex models can be expressed. We can also see that the learning of the network is the learning of the connection weights and thresholds between each node in the network, that is, to find the optimal connection weights and thresholds so that the model can reach the optimal (generally local) solution.

4 Result analysis

In our replication work, we use linear regression model, linear regression with Huber loss (HuberRegressor), elastic net (without Huber loss), principal component regression (PCR with and without Huber), partial least squares (PLS), random forest, gradient boosting and neural network, a total of 7 different kind of methods to find the specific relationship between the return and the characteristics.

For feature importance, we simply use the absolute value of each coefficient in the linear model to represent the importance of each variable. As for tree-based methods, we simply drop one feature by another to compare the R^2 score so that we can determine the significance, and it has been done in scikit-learn package. We then show the top 25 important features generated by each model in figure5.

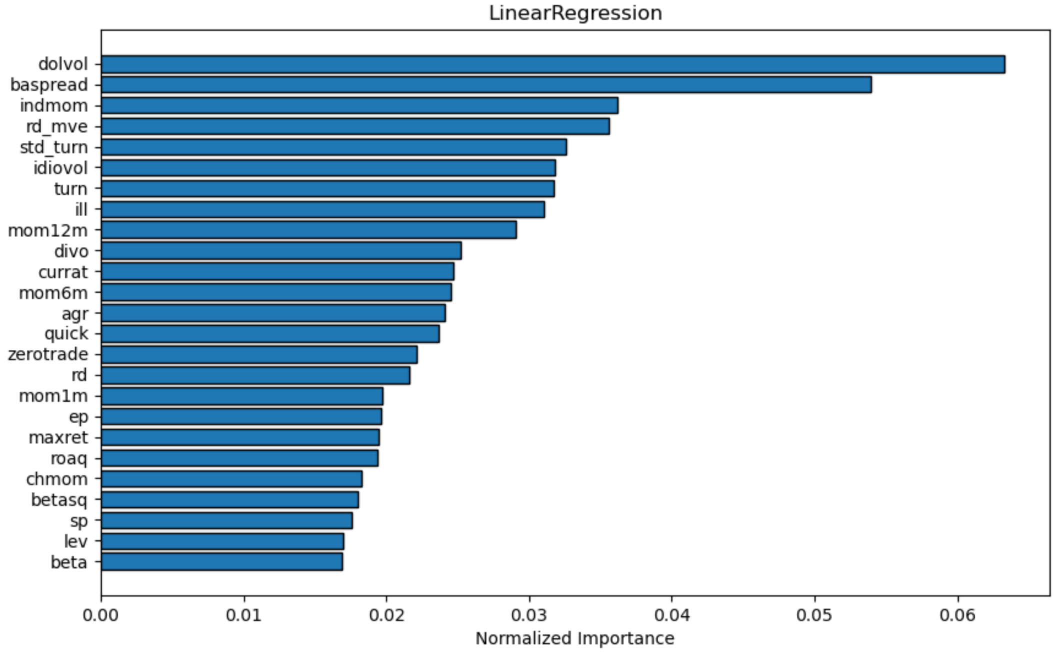


Figure 5: Normalized feature importance given by linear regression

Unfortunately, we fail to get a positive R^2 score, which means the prediction and the ground true value are contradictory. For the percentage R^2, since some models give the score very close to 0, the percentage value is extremely high in some cases. The top and bottom 1000 stocks are selected by market value in the training dataset. Figure 6 gives the average score among different models.

5 Conclusion, future work and acknowledgement

We conduct this simple replication work with various methods and obtain some results given by our machine learning methods. We may consider following steps to improve our model.

First, we can adopt business knowledge and practice into our model so we can understand what is crucial to the final judgment instead of only looking at the information table.

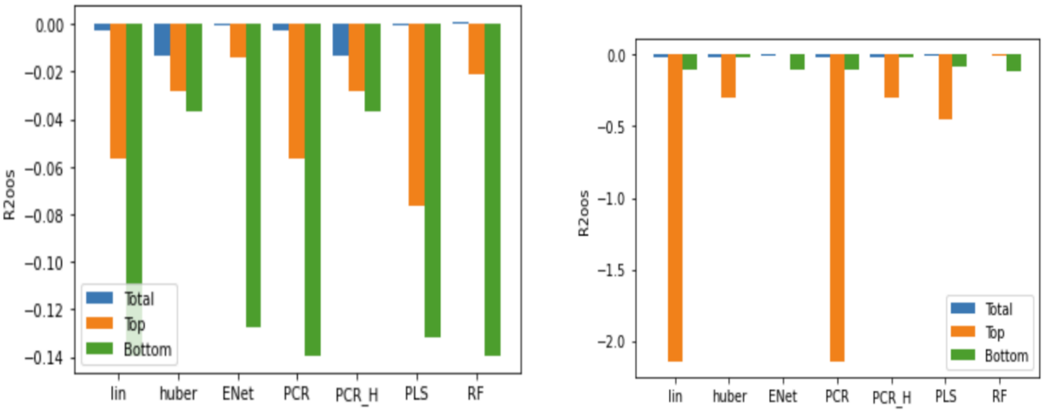
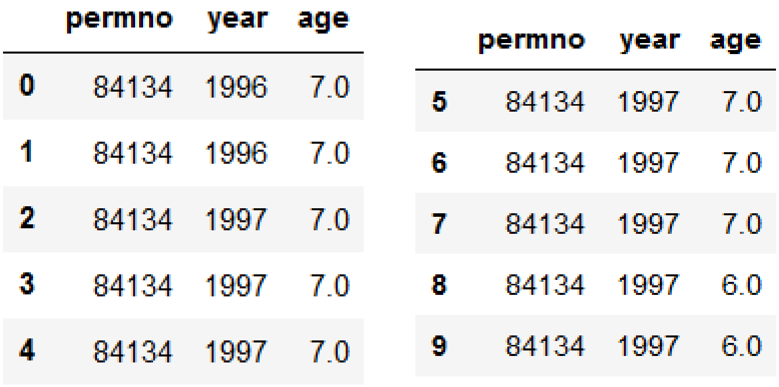


Figure 6: Average R^2 score in validation and testing samples for some models

Second, we may try to find new ways of dealing with data. For the missing data, we may look deeper into the inner relationship like what we have done for category and stock names. For example, the variable age, which is defined as the number of years since first Compustat coverage, meaning that it may be linearly related to year. But in our imputation, there does not exist a linear relationship between the 2 variables, which is quite strange.



	permno	year	age
0	84134	1996	7.0
1	84134	1996	7.0
2	84134	1997	7.0
3	84134	1997	7.0
4	84134	1997	7.0

	permno	year	age
5	84134	1997	7.0
6	84134	1997	7.0
7	84134	1997	7.0
8	84134	1997	6.0
9	84134	1997	6.0

Figure 7: Column ‘age’ after imputation

Third, due to the poor performance of our devices, we just predict the future returns in the whole testing dataset, but the results are analyzed monthly or annually in the paper. We believe the prediction score may rise if we predict with the smaller monthly or annual data, though the computational time will be much higher.

Fourth, the interaction effect is not considered in our model, but discussed in the paper. We may adopt the combinations to see whether it makes a sense in the prediction.

Thanks to those who have shared their experience online!

Reference

[1] Echo Sun, Diebold Mariano Test Package, https://github.com/echosun1996/DieboldMarianoTest

[2] Kaan Yolsever, Empirical Asset Pricing via Deep Learning Algorithms, https://github.com/yolsever/ML-in-equity-prediction