Using Machine Learning to Predict Stock Returns in the Energy Sector

In this paper, I conduct comprehensive evaluations on frequently used machine learning models to find out which top twenty stocks can form a portfolio that outperforms an out-of-sample benchmark in the energy sector. Utilizing historical data and cutting edge machine learning algorithms to predict future market trends in a well-established industry sector can help us to discover new industry trends. In the evaluation, four different datasets from S&P 500 are fetched with Python built-in functions and used for detailed data analysis. Additionally, macroeconomic and political factors are considered. These data are found across five different countries. After obtaining the datasets, I apply linear regression, lasso penalized regression, random forests, gradient boosting regression, and neural networks to train and test models to see which machine learning methodology gives the best performance.

The datasets I used for this project are directly related to the stock performance in the energy sector over the ten years from yahoo finance and I also added exchange rate data from five other foreign countries since I observe that political stability can influence the stock returns. The machine learning performance for each regression is slightly below 1, which on average is lower than the benchmark performance. Hence, even the best machine learning algorithm performance suboptimal comparing with the benchmark performance, and this is could be due to many external factors, which will be described in details in the paper.

Charles Yu  
[yrtcha@umich.edu](mailto:yrtcha@umich.edu)

347-502-5488

Dec 4th, 2022

Contents

[Introduction 1](#_Toc120745162)

[Data Preparation 2](#_Toc120745163)

[Method 2](#_Toc120745164)

[Linear Regression 2](#_Toc120745165)

[Lasso Penalized Regression 3](#_Toc120745166)

[Random Forests 3](#_Toc120745167)

[Gradient Boosting Regression 4](#_Toc120745168)

[Neural Networks 4](#_Toc120745169)

[Results 4](#_Toc120745170)

[Overall machine learning result 4](#_Toc120745171)

[Random Forest Best Predictors 5](#_Toc120745172)

[Gradient Boosting Regression Best Predictors 5](#_Toc120745173)

[Conclusion 5](#_Toc120745174)

Tables

**Table 1. Projected Returns for top twenty energy stocks in the portfolio 1**

**Table 2. Benchmark vs. Multiple financial metrics on energy stock 5**

Figures

**Figure 1. Cumulative strategy returns of energy stocks in the past ten years using linear regression 3**

**Figure 2. Cumulative strategy returns of energy stocks in the past ten years using lasso regression 3**

**Figure 3. The random forest and gradient boosting regression importance plot for the returns of stocks in the energy stock 5**

# Introduction

In the era of big data, machine learning and artificial intelligence provide ideal avenues through which to predict stock markets prices and trends. The energy sector continues to be an industry of particular interest, because of its interconnectedness with current events and far-reaching effects on the global economy due to rapid and frequent commodity price fluctuations. Subsequently, utilizing historical data to predict future stock returns in the energy sector may be useful in foreseeing industry trends.

In this report, I conducted comprehensive evaluations on five frequently used machine learning models and created a proposed portfolio of twenty stocks that outperforms an out-of-sample benchmark. In my evaluation of these predictive models, four different datasets were used: the previously used signals dataset, a signals dataset with smoothed return data without null values, a signals dataset with smoothed return data with null values, and a signals dataset with currency volatility data. All of these data are referenced from New York Stock Exchange. For all models, I employed rolling and expanding windows, tested many valid windows, and determined that an expanding window produced the best results. The table below shows the projected returns of top twenty energy stocks using the expanding window technique in the past ten years. As we can observe, all the projected returns are positive, which means total returns exceeds total costs for each company. This shows that the top twenty energy stocks are gaining profits and the energy market as a whole is booming.

Table 1. Projected Returns for top twenty energy stocks in the portfolio

|  |  |
| --- | --- |
| Company (Ticker) | Projected Return |
| PPL Corp. (PPL) | 5.2% |
| Adams Resources & Energy Inc. (AE) | 0.4% |
| Matrix Service Co. (MTRX) | 5.5% |
| NOV Inc. (NOV) | 4.2% |
| Helix Energy Solutions Group Inc. (HLX) | 3.2% |
| Southern Company (SO) | 3.9% |
| Entergy Corp. (ETR) | 7.5% |
| Pinnacle West Capital Corp. (PNW) | 1.6% |
| PG&E Corp. (PCG) | 5.4% |
| Avista Corp. (AVA) | 2.0% |
| NiSource Inc. (NI) | 10.1% |
| Spire Inc. (SR) | 8.4% |
| Atmos Energy Corp. (ATO) | 6.4% |
| Northwest Natural Holding Co. (NWN) | 6.1% |
| Southwest Gas Holdings Inc. (SWX) | 8.9% |
| Southwestern Energy Co. (SWN) | 14.4% |
| Coterra Energy Inc. (CTRA) | 8.8% |
| Williams Co. (WMB) | 8.5% |
| SJW Group (SJW) | 7.7% |
| Marathon Oil Corp. (MRO) | 14.6% |

# Data Preparation

In addition to the data I found on yahoo finance, I also added exchange rate data, relative to the strength of the U.S. Dollar, from five countries: Canada, Russia, Saudi Arabia, Iraq, and the United Arab Emirates. During my preliminary research, I discovered that political instability in oil-producing nations plays a critical role in stock price movements. The previously listed countries were chosen because they constitute the countries that produce the most oil annually apart from the United States. I also decided to remove outliers from the stock returns. Not doing so would mean inputting noisy data into my models which would make it harder for the models to learn a certain trend on my data. I passed the returns of each stock through a Tukey smoother function and used these returns in all my models. Overall, I saw better performance on validation and test sets.

# Method

Model evaluation is one of the most vital parts of data science since it helps us to understand the accuracy and performance of the model and makes it easy to present to others. There’re three main metrics used for model evaluation: Adjusted R-squared, Mean Square Error/Root Mean Square Error, and Mean Absolute Error. Each statistic has its own benefits and limitations, but all give an approximation of how close a prediction is to the real value.

R-Squared measures how much variability can be explained in the model. The R-Squared value ranges from 0 to 1, and larger values indicate a better fit between the prediction and actual value. However, it does not take into consideration the overfitting problem, which is why Adjusted R-Squared is introduced. Adjusted R-Squared penalizes additional independent variables added to the model and adjusts to prevent these overfitting issues.[[1]](#footnote-1)

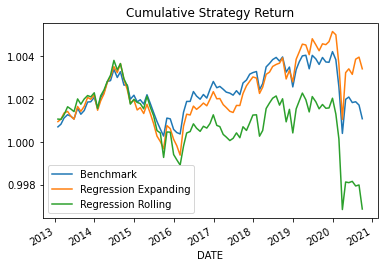
Mean Square Error (MSE)/Root Mean Square Error (RMSE) is an absolute measure of the fit, contrasting how R-Squared is a relative measure of the fit.

Mean Absolute Error (MAE) is similar to MSE, but MAE takes the sum of the absolute value of error. An important consideration between the two metrics is that MSE penalizes prediction errors more while MAE treats all of them the same.[[2]](#footnote-2)

I considered all three metrics and concluded that Adjusted R-Squared would produce the best results because it is the only metric that considers the overfitting problem, while the other statistics are generally better suited to compare performance between different models. Subsequently, Adjusted R-Squared would provide better insights, especially when my dataset contains more than 500 different stocks.

## Linear Regression

The linear regression model was the simplest model in my analysis of predicted stock returns in the energy sector. A linear regression “attempts to model the relationship between two variables by fitting a linear equation to observed data.” Stock returns is considered the dependent variable, and other variables are considered the explanatory variables, such as return on equity, return on assets, the book to market value, etc.[[3]](#footnote-3) The main objective of linear regression is to understand the relationship between the variables of interest. Tuning parameters for the linear regression model involved choosing the appropriate training, validation, and testing sets, and whether to use a rolling or expanding window. I determined that the linear regression produced the highest out-of-sample validation R-Squared with a training set before 2016, a validation set from 2017 to 2019, and a test set of 2020. The expanding window with the large training set was suitable for this application because of the inclusion of currency volatility data. In order to capture the total variation in this data, which accounts for movements in global politics, a larger training set was necessary.

Figure 1. Cumulative strategy returns of energy stocks in the past ten years using linear regression

The graph on the right shows the performance comparison among benchmark, expanding window regression, and rolling window regression. The benchmark here is a default portfolio of all energy stocks with equal weights.

## Lasso Penalized Regression

Lasso Penalized Regression is used to select relevant features by setting coefficients to zero. It is an extension of linear regression that a regularization parameter multiplied by the summation of the absolute value of weights and the loss function of linear regression.

The main objective of Lasso Regression is to shrink the coefficients to avoid overfitting and find the optimal variables that best predict stock returns in the energy sector. The best approach to predict the stock returns is maximizing R-squared in the out-of-sample datasets. Out-of-sample testing refers to using “new” data which is not found in the dataset used to build the model.

In terms of tuning hyperparameters, the portfolio dataset is split into training, validation, and testing sets. For the training set, I subset the dataset that is before 2016. The validation set is a subset from 2017 to 2019. The test set is in the year 2020. The goal of using the validation set is to evaluate the model from the training model. It helps us to tune parameters and find the optimal model before confirming results with the test set. An important financial metric is called alpha, which measures the returns on investment. After trying different values for initialized alpha and the objective, I found that the best R-squared value is approximately 0.3%. This result implies that the model fails to explain the variations of energy stock returns around its mean.

**Figure 2. Cumulative strategy returns of energy stocks in the past ten years using lasso regression**

After finding the R-squared value, I plot the benchmark return value with the Lasso Regression return value, which shows that the benchmark return outperformed lasso regression returns. I set return on asset and return on equity to zero since they are the relevant features. Based on the statistics table, only two variables are significant are tickers symbol and aggregate growth, which does not make sense. Furthermore, plot in figure 2 shows that the regression score gradually decreases as the year goes by and it’s way below the benchmark metric. This again proves that Lasso Regression is not the best to predict the best stock returns.

## 

## Random Forests

Essentially, a random forest works similar to a tree regression except it aggregates the result of multiple trees. Tree models in general are used as opposed to linear regression when the data appears to be non-linear. Essentially, in the tree, every node is an if-else statement that determines the final value predicted. Every node gets split dependent on different variable criteria, such as if book to market ratio is greater than 0.4, go left, else go right.

For random forests, I looked at the best information gain being which would give the highest validation r squared. To tune the hyperparameters I looked at different minimum tree depths (how far the branches of the tree can go), n\_estimators(the number of trees to be used in the forest), and finally max features. Originally, I experimented with values of n\_estimators greater than 100. However, upon some testing, there were few improvements in the model and it did not get much better after 100, even though computationally it took significantly longer. The depths I looked at included 1,2,3 as I believe any deeper and the tree would be overly complex as there is a bias-variance tradeoff with models. I also looked at only 3 or 5 features to be careful with not overfitting the model.

## Gradient Boosting Regression

Gradient Boosting Regression generates a randomly generated number of trees. However, each subsequent tree is built off of the residuals from the tree before it. As a regressor, the gradient boosting algorithm can be used for predicting the continuous target variable, and the cost function is Mean Square Error. All of the hyperparameters that were used are the same as the ones used in the random forest, as the models are similar with the only difference being that new trees are built off of the residuals, not randomness.

## Neural Networks

Neural networks are the ultimate “black box” of ML algorithms. We do not know exactly what is happening under the hood of complex neural networks. Essentially, neural networks have a forward run, in which a loss is calculated, and then this loss is tuned through backpropagation, where weights are changed in the direction that is expected to reduce the loss.

After attempting multiple initial architectures, I found the model drastically overfitting at 5 dense layers and slightly underfitting at 3 layers or less, hence I begin my hyperparameter tuning at 4 dense layers with activation functions. The number of neurons per hidden layer I start with is a standard 16, 32, and 64 respectively. When I noticed that the model was overfitting, I added dropouts of 0.3 and batch normalization after each hidden layer. Afterward, the model seemed to be slightly underfitting this time, where I increased the neurons per hidden layer to 32, 64, and 128 respectively. At this point, I also decided to try ReLU activation functions and add a learning rate decay function, with an initial learning rate of 0.02 and a decay rate of 0.7. Both of these implementations led to my largest validation r-squared of around 0.7%, which is not the largest my other machine learning models have provided. Only 0.7% of the variability of smoothed returns is explained by the variables given.

Overall, the usage of only dense neural networks is limited to predicting stock returns in the energy sector. Tuning hyperparameters hardly increases the validation r-squared, thus they have limited influence on the predictive power of dense neural networks in this context. I believe that recurrent neural networks, which have some sense of time series memory in the construction, would have a better chance of predicting. Given the correct variables, this machine learning method has the potential to beat the benchmark, however, this would come at the sacrifice of understanding which of these variables are most significant and which are mostly irrelevant to the energy market and stock prices in general.

# Results

## Overall machine learning result

**Best Hyperparameters for Random Forest:** {'max\_depth': 3, 'max\_features': 5, 'n\_estimators': 100, 'random\_state': 42}, **Best Test-sample R-squared:** -0.095, **Best Validation-sample R-squared:** 0.055

The test r squared is much worse than the validation r squared. This may be due to overfitting the validation hyperparameter and the training data. This also could be because my test set was 2020 which was an odd year due to covid and all stocks plummeted. I should have not included 2020 in my analysis to get better returns.

Table 2. Benchmark vs. Multiple financial metrics on energy stock

Compared to the benchmark, even the best portfolio predicted from this algorithm performs worse in every regard. The returns are expected to be negative as you can expect a 90-cent return on a dollar. Therefore, the benchmark performance wins!

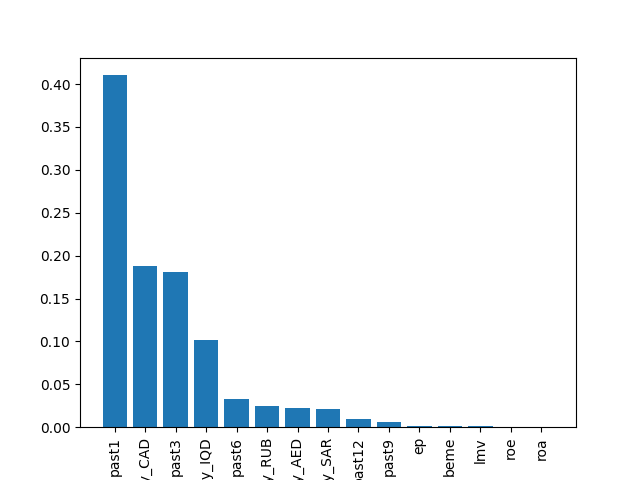
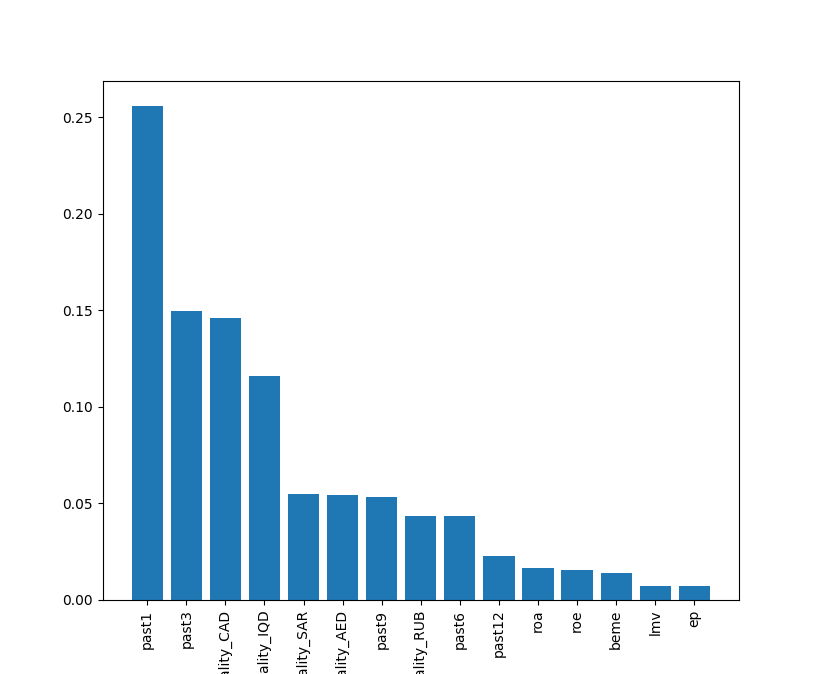
|  |  |  |
| --- | --- | --- |
|  | Top 1 | Benchmark |
| $1 return | 0.90 | 0.99 |
| Geometric return | -0.02 | -1.05 |
| Alpha | 1.4% | 1.85% |
| Beta | 0.94 | 1.25 |
| Sharpe Ratio | 0.34 | 0.52 |

## Random Forest Best Predictors

The random forest importance plot shows that the most important variables out of my data set for the prediction of returns in the energy sector are recent past returns and currency volatilities for the five largest oil exporters against the US dollar. The plot shows that excluding the ‘past1’ variable from the model would result in around a 40% loss in its accuracy.

## Gradient Boosting Regression Best Predictors

The top 10 most important variables are the same. Once again, ‘past1’ appears to be the most important variable: if it is excluded, then the model will lose around 25% of its accuracy. This reinforces the correlation between recent past data and major oil exporters’ currencies’ volatilities with the stock returns of the energy sector.

**Figure 3. The random forest and gradient boosting regression importance plot for the returns of stocks in the energy stock**

# Conclusion

The companies that I recommend be held onto if one must hold onto one are the ones that appeared in decile 0 most frequently for all dates of the portfolios created by the random forest model. The overall return of investing in these stocks is 0.9 compared to a benchmark of 0.99. However, this is a worse performance than the benchmark, so even my best ML algorithm still gave a worse performance than the benchmark. This also may be due to the time series graph I looked at for returns vs time for the data showing a decreasing return in this sector for the past few years which makes it hard to predict using a machine learning algorithm if the stocks are dipping. Additionally, since I am attempting to predict returns on an industry largely affected by the political climate, there is not a sufficient number of political events that occurred during given training time frame.

1. https://towardsdatascience.com/what-are-the-best-metrics-to-evaluate-your-regression-model-418ca481755b [↑](#footnote-ref-1)
2. Ibid. [↑](#footnote-ref-2)
3. http://www.stat.yale.edu/Courses/1997-98/101/linreg.htm [↑](#footnote-ref-3)