**Intelligent Crude Oil Price Forecaster**

Ardalan Tebyanian

Dept. Computer Science

Baha’i Institute for Higher Education

Tehran, Iran

[ArdalanTebyanian@hotmail.com](mailto:ArdalanTebyanian@hotmail.com)

***Abstract* —We propose two ensemble regression algorithms for forecasting the daily price of crude oil from features extracted from the U.S. Energy Administration and some international news agencies. An ensemble regression model consists of a group of homogeneous regressors with varying parameters, e.g. linear regression models with different ridge regularization parameters. The first ensemble method called “recent leader” picks the individual regressor with least mean square error over recent data. The second model called “exponentially weighted ensemble” combines individual regressors in a linear fashion with weights of constituent models decaying exponentially with the mean square error over past data. These two methods were tested with linear regression, support vector regression, decision trees and Gaussian processes. Exponentially weighted ensemble with support vector regression had the best performance.**

Keywords— Ensemble Learning, Online Learning, Crude oil price, forecasting, linear regression, support vector machine, decision tree, Gaussian process

I. Introduction

Our system predicts the price of crude oil (West Texas Intermediate) from two groups of features: news-related features and quantitative features. Quantitative features are obtained from U.S. Energy Information Administration. They contains information such as ??? as prices o, production, and costs [2]. Quantitative data is not enough to be applied for forecasting, so the system needs to take news-related features as well. News-related features are crucial information latent in the news that could possibly determine the price of the crude oil. Our predictive model extracts features from some of the most reliable international news agencies such as Reuters, Euro News, Bloomberg, and CNN. Features are extracted using the Tfidf technique for transformation of texts.

We explore 4 classes of regression models: support vector regression, Gaussian processes, linear regression and decision trees. Each of these classes needs parameter tuning. For instance the amount of regularization in linear regression with L2 regularization is an important parameter that could lead to over-fitting or under-fitting if not set properly. We propose two ensemble methods for parameter tuning. The first method picks the parameter of a class of regression models (e.g. linear regression models with L2 regularization) that minimizes the mean squared error of recent predictions. The other method combines regressors of varying parameters in a linear fashion. Each regressor has a non-negative weight that depends on its performance in the past. The higher the mean squared error of the regressor over past predictions the lower the weight. Initially all the weighted are the same. The weights add up to one and are updated daily. The ensemble’s prediction is the weighted sum of all predictions.

This paper is organized in the following fashion: section 1.1-1.3 talks about extracting features from news agencies. Section 2 covers the parameters that needs tuning and section 3 explains the above-mentioned ensemble methods of parameter tuning and section 4 contains the results.

II. News Parsing

Webpage parsing is an essential part of the system as news-related features are mined from news agencies’ websites. Contents of a webpage are included in its html code and the system parses webpage’s html code in order to get links, topic and text of news. Every html code is made of html tags; each representing a component in a webpage. The system opens the URL of a website and gets contents of the webpage in the html format and parses them.

1. Validation

There are many links in every webpage that are not usable and if the system planned to open all links, it would waste a lot of time. So, there is a mechanism that prevents unnecessary links. If any of the following happens then the link is rejected.

1. A link without any topic
2. A link that contains “video” or “pictures”
3. The number of words in a topic is less than a certain threshold ???
4. The number of characters in a link is less than a certain threshold ???
5. The topic or link is repeated several time (links that are repeated frequently and are not usable specially in archive sites)
6. Text Analyzing with Tfidf

Tfidf is a technique applied on top of bag of words for determining the importance of a word in a document. It consists of two parts: the term frequency (*tf*) and the inverse document frequency (*idf*). Tfidf’s formula is:

*t* is the term or the keyword, *d* is a link for a specific date and *D* is the collection of all links for a specific date.

*Tf (t,d)* is the frequency of keyword *t* in a link *d* divided by the length of *d*.

And *idf* is computed in the following way:

*idf* makes sure that the words that are common in a lot of documents do not get high weights. *Tf-idf* is calculated for every keyword in every link and at the end of the day, the system calculates the average of the *tf* for each keyword in that day. This procedure is repeated for every day and the system returns numeric values that show the importance of the keywords in that day. In this step, news-related variables change into numeric values that are meaningful for the forecasting part.

1. Parameters

As we mentioned earlier we explore 4 groups of regression models, linear regression, support vector regression, decision trees, and Gaussian processes. The parameters that need tuning are: The “Ridge” in Linear regression, “C” or the multiplier of slack variables in support vector regression which controls the complexity of the model, minimum number of samples in each node and the depth of the tree in decision tree, and “Noise” in Gaussian. We use “Recent Leader” in Section 3.2 and “Exponentially Weighted Voting” in Section 3.2 for tuning these parameters

1. Recent Leader

In this technique multiple regressors are trained with different parameters. For example for support vector regression, multiple regressors are trained each with a different “C” parameter, where “C” is the coefficient multiplied to the sum of the squares of the slack variables. The prediction happens on a daily basis. The best parameter can be not be picked by cross validation. Our experiments show that the best parameters can change rapidly over time even within a few days. “Recent Leader” takes the parameter with minimum least square error over the last few days. Figure 1 shows that the best number of days to look back for support vector regression is one, meaning the best model of yesterday leads to best performance.

B. Exponentially Weighted Voting

In thie

[3]

---------------------------------------------------

Energy Information Administration

1. Results
   1. Finding the best parameter from the previous days.

|  |  |
| --- | --- |
| Technique | Average Error |
| Linear regression | $0.986965409 |
| Support Vector | $0.980913208 |
| Decision Tree | $2.018549665 |
| Gaussian Process | $3.596292472 |

|  |  |
| --- | --- |
| Technique | Average Error |
| Linear regression | $1.839471798 |
| Support vector | $1.042434537 |
| Decision tree | $1.454572516 |
| Gaussian process | $3.951076423 |

|  |  |
| --- | --- |
| Technique | Total Average Error on Dec/ Jan 13/14 |
| Linear regression | $1.4132186035 |
| Support vector | $1.0116738725 |
| Decision tree | $1.7365610905 |
| Gaussian process | $3.7736844475 |

|  |  |
| --- | --- |
| Technique | Average Error Without Optimization, Dec 13 |
| Linear regression | $1.9255316 |
| Support vector | $1.38803963 |
| Decision tree | $5.399696585 |
| Gaussian process | $2.456163716 |

|  |  |
| --- | --- |
| Kernel | Error Value |
| RBF (in Support Vector) | $2.355278246 |
| Poly (in Support Vector) | $3.687331899 |
| RBF (in Gaussian Process) | $2.02712625 |
| Poly (in Gaussian Process) | $1.046930641 |

* 1. Results Based on WMA

Dec 2013:

|  |  |  |
| --- | --- | --- |
| Technique | Average Error | η |
| Linear regression | $0.982452342 | 90 |
| Support vector | $0.884684483 | 5 |
| Decision tree | $2.024790489 | 40 |

|  |  |  |
| --- | --- | --- |
| Technique | Average Error | η |
| Linear regression | $1.848572915 | 30 |
| Support Vector | $1.025054963 | 5 |
| Decision tree | $0.93964427 | 90 |

1. Future Works

In the next phase of this project, the system should be able to analyze news’ texts and detect more reliable data for forecasting. It should be able to give feedback to trader about the commodities market with intelligent text analyzing and make better decision for forecasting. The realization of Algorithmic Trading concept that is trading without human intervention is the final destination of this project.

1. Conclusion

This system detects texts of news from news agencies’ websites and mines importance of predefined news-related variables in them. After that it forecasts the price of crude oil (WTI) with use of four machine learning techniques that are: 1) Linear regression 2) Support Vector for regression 3) Decision Tree 4) Gaussian Process. It can finds optimum parameters for these techniques and forecast crude oil price. As, it is shown in Results part, Support Vector technique is the best technique for forecasting the price of crude oil. As, it is was shown in this project and other researches [4], Support Vector technique has the best revenue and as it was shown , the forecasted curve fits the real curve. After Support vector, Linear regression is the best and after that Decision tree and finally Gaussian process. Decision Tree and Gaussian process are not reliable techniques in this project since the can face user with more risks.

Acknowledgement

This research paper is made possible through the help and support from Dr. Fares Hedayati, the supervisor of this project. I would like to thank Dr. Fares Hedayati for his support and encouragement.

References

[1] Xiongpai Qin, “Making Use of the Big Data: Next Generation of Algorithmic Trading”, School of Information, Renmin University of China, Beijing.

[2] U.S. Energy Information Administration, Available: <http://www.eia.gov/> , Accessed: June 2, 2014.

[3] Shivani Agarwal, “Online Learning from Experts: Weighted Majority and Hedge”, Lecture 20, 17 Nov 2011.

[4] S.K Shevade, S. S. Keerthi, C. Bhattachayya, and K. R. K. Murthy, “Improvements of the SMO Algorithm for SVM regression”, IEEE Transactions on Neural Networks, Vol. 11, No. 5, September, 2000.