International Journal of Psychosocial Rehabilitation, Vol. 24, Issue 08, 2020
ISSN: 1475-7192

Financial Asset Valuation Using Online Learning Techniques in Semi-Streaming Data

[1]J.V Vidhya, [2]Proma Mukherjee, [3]Harinipriya B

Abstract--*Online learning is a process toward an- swering an arrangement of inquiries when the knowledge of the geuinine result is given. Online learning majorly constitutes the algorithms Weighted Ma- jority as well as Randomized Weighted Majority. These are popularly called the mistake bound models, which means that the algorithms can specify the upper bound of its number of mistakes made in the prediction. This project introduces Financial asset valuation using online learning techniques combined with streaming data input, by means of the Apache Spark and Python collaboration (PySpark), Resilient Distributed Datasets (RDD) and Web scraping for processing the semi streaming data and Anaconda, Python3 in Jupyter Notebook to process the batch database. Our aim is to considerably reduce the regret bound for prediction of the values as well as to introduce a novel approach to streaming data applied on online-learning methods.*

Key words---*Algorithms weighted Ma-jority*

I. INTRODUCTION

The streaming data is derived from the live data source using pipelining through Apache Kafka, Apache Spark is used to process the streaming data and convert into a form used to store temporarily in small chunks of data. PySpark (the Apache Spark and Python collaboration) and Web scraping using Python are used to transfer and receive live data. Once stored, the data is derived in order to apply online learning algorithms. This is an ensemble approach. [9] Let n be the number of companies being processed. The available data about the stock values of these n companies over a period of T days is processed using an online learning algorithm as we develop below. We therefore have a table of data with T rows and n columns where each row gives the stock values for each of the n companies on T distinct days. These data for T days are used as T “experts”. Another set of S rows of stock values of the same n stocks is used as “training data”. We may view the increase or decrease in stock values over one specific day, say Day 1, as some indication of how the stock values move up or down or remain the same. Such changes when observed for several days may very well give an idea how the stocks would behave on a different day in future, say Day 16. Similarly, the change in the stock value for a company on Day 2, is also an indicator of how the stock value would possibly alter on the same future Day 16.

Let us assume that the T experts predict transitions (i.e., ‘up’ / ‘down’ / ‘no change’) for all the n stocks for a number S of days. The “expertise”, i.e., the transition information of n stocks from day to day, for T days, will be

[1]*Assistant Professor (Sr.G), SRM IST, vidhyachandran8890@gmail.com*
[2]*Student(final year), BTech CSE, SRM IST, mailtoproma@gmail.com*
[3]*Student(final year), BTech CSE, SRM IST, harinipriya.b@gmail.com*

DOI: 10.37200/IJPR/V24I8/PR280266
Received: 21 Jan 2020 | Revised: 08 Feb 2020 | Accepted: 14 Mar 2020

pitted against the transition data, for S days, which will act as the "training set" of size S.

The expertise of the expert is to be viewed as his ability to predict transitions for the n stocks, where for which each wrong prediction, the expert would suffer a penalty[1]. Also, naturally, all these T daily experts would be predicting the stock value base on the actual value of stock transitions in these T previous days.

Observe that the transition on an arbitrary day may or may not correlate with some training data set transitions. Therefore the predic- tions can be safely said to be arbitrary, and for each wrong prediction, the expert would suffer a penalty. It is the job of the *online learning algorithm* to produce *final prediction answers*, and to also decide as to which of the experts (forecasters) could be labeled as the "best expert".

The dataset originally derived from the various sources of streaming data[11] such as heteroge- neous, dynamic and distributed souces, followed by reducing their weights to one-thid. [12].

Each day's transition can be thought of as a *trend*, which can be used to predict the transition for a future date. So, each day's trend is an expert, covering all stocks. Therefore, there are as many experts as the number of rows. And each expert has n tries across companies, where n is the number of stocks. The number of experts T is the number of data rows. In general, any day's stock transition can be predicted by the rest of the transitions. The predicted transition could very well be for an intermediate day as well.

Organization of the report the material presented in this report is organized into five different sections. Literature Survey A survey across the different research papers that have been referenced through- out the scope of this project, and have been cited appropriately, which [2] as the base paper for the project.

The online learning method gives a brief introduc- tion about Avrim Blum's[1] online survey model and the algorithms applied to solve online learning problems, namely the Weighted Majority and the Randomized Weighted Majority algorithms. This section consists of three subsections. Predicting from "expert advice" deals with predictions of the outcome as done by randomly chosen experts. The second subsection deals with the weighted majority algorithm, in which the weight of each mistaken expert is duly reduced, and the final outcome is derived from the majority weight. The third and fi- nal subsection is the randomized weighted majority algorithm, which adds the element of randomization to the weighted majority algorithm, and the final outcome is selected based on random probabilistic values.

Enhancement Strategy gives an insight into an additional concept introduced in this paper over Blum's basic model, it is called the "enhancement method".[15]

Cumulative probabilities from weights for ran- domized decision for multiple outcomes This section defines the testing and training datasets in detail and the concepts that have been used to build the developments performed.

Proposed expertise model based on actual stock value changes from live data The final outcome and results of the experiments performed are con- tained in this section, along with references to the tables and graphs from the appendix.

II. LITERATURE SURVEY

Online Learning Applied to Autonomous Valuation of Financial Assets, Marcel Soares Ribeiro and Paulo Andre Lima de Castro, IEEE 2018 [2] This is, in some ways, the base paper of the project in which comparative error-adjusted RWMA (EARWMA) is done over Weka, a data-mining technology tool used for comparative study. The drawbacks of this paper are that it lacks the appropriate streaming data sources required in order to make the predictions, or, in other words, it uses directly batch-processed data. This drawback is handled in this paper. In addition, it lack enhancement strategy to further enhance the performance of RWMA algorithm, which makes the algorithm less efficient.

Ensemble learning methods for decision making: Status and future prospects, Shahid Ali, Sreenivas Sremath Tirumala and Abdolhossein Sarrafzadeh, *IEEE 2015*[8] This paper essentially deals with ensemble learning methodologies, that are used to combine several machine learning algorithms and incorporate into a single scope. This method is used to diversify the data sources as well as for a more efficient processing of the data thus procured. The algorithm uses minor Machine Learning tasks to achieve the goal.

Time-varying stochastic multi-armed bandit prob- lems by Sattar Vakili, Qing Zhao and Yuan Zhou, IEEE 2014[9] In this paper, the stochastic multi- armed bandit problem is defined, in which, the problem statement is distributed across several arms, the “arm” or the facet of the algorithm that produces the maximum gain, is considered as the best arm. The algorithm has to make an informed decision about choosing the arm that yields maximum benefit. It is used for maximum profit estimation. The drawback is that there is arbitrary distribution of weight (benefit) in each arm, leading to incoherence in the final outcome. Another major drawback which is not addressed in this paper is the loose regret bounds. One of the major tasks of online learning techniques is to tighten the mistake/regret bound of the algorithm.

Cascading Randomized Weighted Majority: A New Online Ensemble Learning Algorithm, Mohammadzaman Zamani, Hamid Beigy, and Amirreza Shaban, IEEE 2014[10] The cascading version of Randomized Weighted Majority Algorithm (CR- WMA) is found to significantly increase the ef- ficiency of the algorithm since it introduces an improvement over the already existing RWMA al- gorithm. Since the algorithm uses classes to identify best outcome rather than using expert systems, the identification of ‘true positive’ and ‘false positive’ classes becomes a real challenge. The deciding weight factor is complicated, and in most cases difficult to understand.

Ensemble of distributed learners for online clas- sification of dynamic data streams, Luca Canzian, Yu Zhang, Mihaela van *der Schaar, IEEE 2015*[11] This paper tends to eliminate the need to define a specific source of data, and its approach allows that the data can be derived from a multitude of sources: heterogeneous, distributed as well as dynamic, hence eliminating the need for a single data-source. The data derived from different sources is cleaned separately before combining into a comprehensive set, which is worked on using the algorithms. The only drawback is that the performance gains varies between 34% 71%, which is not acceptable as per market standards.

Ensemble based classification using small training sets : A novel approach, C.V Krishna Veni, T Sobha

Rani, IEEE 2014[12] This project focuses on reducing the dataset by one-third, it is useful when processing very large amounts of data in a limited space or processing power. The annotated dataset is reduced to one-third of its original size without any significant change in its distribution of features. The drawback is that there is significant chance of erroneous or incorrect modeling, as well as imbalanced datasets.

SVM Multi-classifier and Web-document classifica- tion, Jiu-Zheng Liang, IEEE 2004[13] The Support Vector Machine (SVM) Support-vector machines are used to analyze data for classification and regression in supervised models. Given, a set of training data, Standard Vector Machine is used to mark the classes as belonging to one category or another, hence it can be called a non-probabilistic linear classifier in binary, the examples in space are categorized as having a division by a gap as large as possible. Using SVM for web page classification has been found to be more effective than the already existing methods to do the same. But due to high data complexity, it does not find any practical usage in regular classification problems.

A Weighted Majority Vote Strategy Using Bayesian Networks, Luigi P. Cordella, Claudio De Stefano, Francesco Fontanella, Alessandra Scotto di Freca, Springer 2013[14] It uses the approach of joint- probabilities using Byesian classification technique. The algorithm combines different classifiers to function as one, the efficiencies of all these classi- fication methods get combined, whereas the draw- backs get divided.

A Reinforcement Learning and Recurrent Neu- ral Network Based Dynamic User Modeling Sys- tem, Abhishek Trupathi, Ashwin T S, Ram Mo- hana Reddy Guddeti, IEEE 2018[15] The model is trained to learn emotional intelligence by a context- aware approach. The concept is based on novel and collaborative-based approach which is used under reinforcement technology of Markov model. Initial errors are very high due to a condition known as "cold start". It means that at the beginning of training the model, where there is no training done on the model yet, it is not able to make informed decisions. Only after facing some generations of training, it is able to pick up the pace and perform better.

III. THE ONLINE LEARNING METHOD

Online learning [1], [3] is the process of answering a sequence of questions when the knowledge of the actual zz`outcome is given. The questions are answered rationally in order to make intelligent decisions daily. E.g., *Will it rain today? Do I fight or flee from wild animals?* Online learning algorithms is therefore an important domain in machine learning, and it has several practical applications.

Online learning is performed in a series of rounds, where in each round the learner is expected to give an answer to a question. After the prediction is done by the best expert, the right answer is shown and the learner suffers a loss (in this context, called *regret*) on the off chance that there is an inconsistency between his answer and the right one.x`

The areas of Online Algorithms [1], [3] and Ma- chine Learning manage issues about settling on The areas of Online Algorithms [1], [3] and Machine Learning manage issues about settling on decisions in the present,

DOI: 10.37200/IJPR/V24I8/PR280266
Received: 21 Jan 2020 | Revised: 08 Feb 2020 | Accepted: 14 Mar 2020

in the light of knowledge about the past. The goals and objectives of online learning algorithms may vary from problem to problem even though in general they can be said to fit into the "Computational Learning Theory" framework []. We begin with the concepts of "Prediction from Expert Advice" and then elaborate on "The Weighted Majority and Randomized models", followed by a suitable combination of these two techniques.

Predicting from "Expert" Advice

In this technique, the algorithm is given a set of "expert" advice to be pitted against the original outcome, each expert being penalized by a general factor each time it predicts wrongly. It is to be noted that in this context, "expert" is anyone who is willing to give his opinion, and it does not, in general, have much to do with the literary sense of the term. Let us consider the classic example of whether or not it will rain on a particular day. Say, we have three experts, namely, Expert A, Expert B and Neighbour's Dog.

Table 1: Rain prediction table

Expert A	Expert B	Neighbour's Dog	Rained
Yes	No	No	Yes
No	Yes	Yes	Yes
No	Yes	No	No

In the above table, we find that the expertise of Expert A and that of neighbour's dog match, and is greater than that of Expert B, since Expert A and the neighbour's dog predict correctly twice, whereas Expert B is found to predict correctly only once.

The sample stock prediction data-set used in the predictions and analysis for 10 companies are tab- ulated below in two tables. See Tables II and III in the appendix.

The Weighted Majority Algorithm

In this learning technique [1], each expert opinion is assigned a particular weight. Initially, the expert weights w_1, w_2......w_n are assigned the value 1. We are given a set of predictions x_1, x_2......x_n by the respective experts. For the sake of simplicity we shall consider the prediction outcomes to be $x_i \in \{0, 1\}$. The opinion with the *majority* is selected as the final prediction answer of the algorithm. The outcome is 1 $\sum_{i:x_i=1}^{N} w_i \geq \sum_{i:x_i=0}^{N} w_i$, and 0, otherwise.

After the algorithm predicts an answer, the right answer say l is shown. The experts who went wrong are penalized by halving their respective weights, that is, if xi 6= l, then wi wi=2. We may use a more generalized penalization technique whereby the mistaken experts' weights are reduced by a factor of βfor every mistake, i.e., if xi 6= l, then $w_i \leftarrow$ (1- β _)w_i.

Theorem 1. *[1] The number of mistakes that the Weighted Majority Algorithm makes cannot exceed* 2.41(*m + log n*), *where the number of mistakes made by the most efficient expert is given by m*

Proof. We know that initially all the weights in W are assigned 1. Therefore, initially, $W = n$. If the prediction by the algorithm turns out to be incorrect, it means at least half of the experts must have made the wrong

DOI: 10.37200/IJPR/V24I8/PR280266
Received: 21 Jan 2020 | Revised: 08 Feb 2020 | Accepted: 14 Mar 2020

prediction, and therefore W is reduced by at least $\frac{W}{4}$. So, after the algorithm commits M mistakes, we have

$$W \leq n(\frac{3}{4})^M \qquad (1)$$

If the best expert makes m mistakes, then

$$W \geq (\frac{1}{2})^m \qquad (2)$$

Combining inequalities 1 and 2, we have

$$n(\frac{3}{4})^M \geq \frac{1}{2}^m$$

$$(4/3)^M \leq n2^m \implies M \log \frac{4}{3} \leq \log n2^m$$

$$\implies M \log \tfrac{4}{3} \leq \log n + m \log 2$$
$$\implies M \leq \frac{\log n + m \log 2}{\log \frac{4}{3}}$$

$$\implies M \leq 2.409441778(m + \log n)$$

For problems where there are three or more out- comes, we have a slightly different bound as fol- lows. If there are $c > 2$ outcomes then we must choose the outcome where the weights of the ex- perts predicting the outcome is at least $\frac{W}{}$. So, The inequality 3 would become

$$W \leq n(\frac{5}{6})^M \qquad (3)$$

Consequently, we would arrive at the bound M≤
$\frac{m+\log n}{\log \frac{6}{5}} \leq 3.801784017(m + \log n)$.

Randomized Weighted Majority Algorithm

The "randomized weighted $\in \{0,1\}$ majority algorithm" tightens the error bound of the weighted majority algorithm. In this technique, we use the final weight of each expert to determine the probability for its prediction being chosen as the outcome of the algorithm. Let the elements of the vector of weights $W = w_1, w_2$ be initialized to 1 each. Let, once again, x_1, x_2 be the set of opinions given by the experts. Let, for the sake of simplicity x_i After predictions are made, the correct answer l is revealed. We shall then update the weights of the mistaken experts by deducting a penalization.

$$\forall (x_i \neq l), w_i \leftarrow (1 - \beta)w$$

The probability of choosing expert opinion x_i is given by

$$P(x_i) = \frac{w_i}{W}$$

DOI: 10.37200/IJPR/V24I8/PR280266
Received: 21 Jan 2020 | Revised: 08 Feb 2020 | Accepted: 14 Mar 2020

Where, $W = \sum_{i=1}^{N} w_i$ denoting the sum of the entire set of weights.

IV. THE ENHANCEMENT STRATEGY

Theorem 2. The mistake bound under the condition that the weight of each of the n experts predicting correctly is rewarded by a factor of and those predicting wrongly are penalized by a factor of β , is given by

$$M \leq \frac{m\,ln(\frac{1}{\beta}) + (n-m)ln\,(\frac{1}{\alpha}) + ln\,n + (\alpha+2)}{(2-\beta+\alpha)}$$

where, the number of mistakes made by the best expert is given by m.

Proof. Let, fractions F_i of experts are mistaken at time instance $_i$. The final weight after time t is given by

$$W \leftarrow W(1-(1-\beta)F_i) + W(1+(1+\alpha))(1-F_i)$$

$$\implies W = n\prod_{i=1}^{t} W(2-(1-\beta)F_i+(1-\alpha)(1-F_i))$$

$$= n\prod_{i=1}^{t}((3+\alpha)-(2-\beta+\alpha)F_i)$$

$$= n\prod_{i=1}^{t}(1-((-\alpha-2)+(2-\beta+\alpha)F_i) \geq \beta^m\alpha^{n-m}$$

Assuming the best expert makes m mistakes, the final weight is in any case greater than the final weight of the best expert. Now, taking log on both sides gives us

$$ln\,n + \sum_{i=1}^{t} ln(1-((-\alpha-2)+(2-\beta+\alpha)F_i) \geq m\,ln(\beta) +$$

$$(n-m)$$

ln(α)Taking negation on both sides gives us,

$$-ln\,n - \sum_{i=1}^{t} ln(1-((-\alpha-2)+(2-\beta+\alpha)F_i) \leq$$

$$m\,ln(\frac{1}{\beta}) + (n-m)\,ln(\frac{1}{\alpha})$$

$$\implies ln\,n + \sum_{i=1}^{t}(-\alpha-2)+(2-\beta+\alpha)F_i \leq m\,ln(\frac{1}{\beta})$$

$$+(n-m)\,ln(\frac{1}{\alpha})$$

Since we know that, ln (1 - (x + c)) ≤ - (x + c) as shown in the Figure1 below. ln (1 - (x + c)) is shown in red and (1 - (x + c)) in blue line.

$$\implies M \log \tfrac{4}{3} \leq \log n + m \log 2$$
$$\implies M \leq \frac{\log n + m \log 2}{\log \frac{4}{3}}$$
$$\implies M \leq 2.409441778(m + \log n)$$

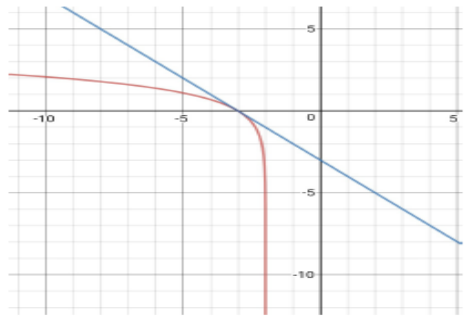

Figure 1: Example Graph

$$\implies -ln\, n + (-\alpha - 2) + \sum_{i=1}^{\iota} (2 - \beta + \alpha) F_i \leq m\, ln(\frac{1}{\beta}) +$$
$$(n - m)\, ln(\frac{1}{\alpha})$$

$$or, -ln\, n + (-\alpha - 2) + (2 - \beta + \alpha) M \leq m\, ln(\frac{1}{\beta}) +$$
$$(n - m)\, ln(\frac{1}{\alpha})$$

Where M is the total number of mistakes made by the algorithm, and M Pt i=1 Fi. Rearranging, we have

$$M \leq \frac{m\, ln(\frac{1}{\beta}) + (n - m) ln\,(\frac{1}{\alpha}) + ln\, n + (\alpha + 2)}{(2 - \beta + \alpha)}$$

It is to be noted that this bound value is significantly lesser than the conventional method, since in the denominator we receive a factor of (2 +) which reduces the overall bound value by at least 50%, since value in usual cases does not exceed 0:5. However, the exact percentage of tightening is simply an assumption. Please refer to Appendix II for the programming code and results.

DOI: 10.37200/IJPR/V24I8/PR280266
Received: 21 Jan 2020 | Revised: 08 Feb 2020 | Accepted: 14 Mar 2020

Table 2: Stock data Table 1

Date	(DIS)	(MSFT)	(NKE)	(JNJ)	
12/28/2017	107.77	85.72	62.95	296.35	140.56
12/27/2017	107.64	85.71	62.95	295.62	140.57
12/26/2017	108.12	85.4	63.65	295.36	140.09
12/22/2017	108.67	85.5	63.29	295.1	140.12
12/21/2017	109.57	85.51	64.77	295.03	141.06
20/12/2017	109.69	85.52	63.59	297.9	141.16
19/12/2017	111.81	85.83	63.59	297.25	141.78
18/12/2017	111.03	85.83	64.81	296.14	141.8
15/12/2017	111.27	85.83	64.79	293.94	142.46
14/12/2017	110.57	84.69	64.53	293.88	141.65

Table 3: Stock data Table 2

Date	(DIS)	(MSFT)	(NKE)	(JNJ)	
12/28/2017	173.1	46.22	84.02	99.7	154.04
12/27/2017	172.67	46.11	83.9	99.13	153.13
12/26/2017	171.29	46.08	83.98	98.57	152.83
12/22/2017	171.42	46.7	83.97	98.74	152.5
12/21/2017	171.85	46.76	83.85	98.5	151.5
20/12/2017	172.17	47.56	82.87	98.51	152.95
19/12/2017	173.39	47.04	82.44	99.15	153.23
18/12/2017	174.2	46.26	82.94	99.68	153.33
15/12/2017	174.06	44.56	83.03	98.52	152.5
14/12/2017	173.14	43.26	82.9	97.15	154

V. PROPOSED EXPERTISE MODEL BASED ON DATA SET T OF ACTUAL STOCK VALUE CHANGES DERIVED FROM LIVE STOCK DATA

The stock values are directly derived from the live data streaming web-site by the use of a special APIKEY that is generated especially for the purpose. The streaming uses Apache Spark, Web Scraping, Resilient Distributed Datasets and PySpark in order to process and capture the streaming data into batch-processed data. An example dataset is shown in the figure of the table below, which shows some exemplary conversion of streaming to batc data, also the same data that we have performed the coding and results on. The daily changes in stock values is given in the "expert" set T. The values here would be used to determine the predictions of experts for the daily changes in stock values– increase, decrease or no change.

The stock values are directly derived from the live S We combine the technique of weight penalties for experts in the training phase and use these weights to define probabilities based on weights for multiple outcomes or decisions as follows. Stocks may either go up or go down or remain at the sam value. Such three outcomes can be predicte by each expert. For each type of outcome we add the weights of experts making that prediction. So we get one weight for each type of outcome. In the stocks problem we thus have three sets of experts, one set for each

DOI: 10.37200/IJPR/V24I8/PR280266
Received: 21 Jan 2020 | Revised: 08 Feb 2020 | Accepted: 14 Mar 2020

outcome. Of course, we penalize experts individually, for making wrong predictions for each training example. If we are dealing with *n* stocks and jSj-sized training set S, then we can penalize for wrong predictions across stocks as well as across training examples, therefore.

The test data is on a future date that is disjoint from the experts set T as well as the training set S. The testing phase with test data from the set V The test data is in the set V . Please see appendix I for details of programming code 1. In the above figure, we show the variation of the weights of the elements of T of experts. Weights over the first training set (shown in blue) is followed by the performance on the second training se (shown in red). We can see the gradual drop in values of weights. After the training set S was processed, the algorithm was tested on a Test set V.

The test data is on a future date that is disjoint from Set (training set). The following table illustrates set jTj of expert

opinion.

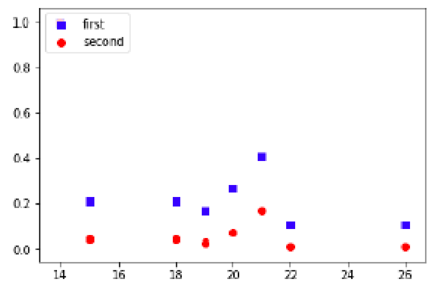

Figure 2: The above figure shows how the weights change across the training and test sets

The set of weights W after first training set is given as:
[0.10737418240000003, 0.10737418240000003, 0.4096, 0.26214400000000004, 0.16777216000000003, 0.20971520000000005, 0.20971520000000005, 1.0]
Similarly, set of W weights after second training is given as:
[0.011529215046068, 0.011529215046068, 0.16777216000000003, 0.06871947673600001, 0.028147497671065603, 0.04398046511104001, 0.04398046511104001, 1.0]

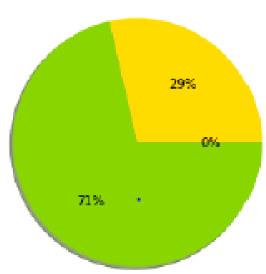

Figure 3: The above figure shows the weights across three major expert opinions, “increase”, “decrease” or “no change”

DOI: 10.37200/IJPR/V24I8/PR280266
Received: 21 Jan 2020 | Revised: 08 Feb 2020 | Accepted: 14 Mar 2020

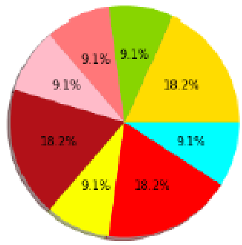

Figure 4: The above figure shows the distribution of weights across several predicting experts.

As can be seen from the above two graphs, when compared, the optimized method 5 yields a more comprehensive way of examining the best expert opinion, rather than the randomized approach, 4 which yields, in the later training stages, a less comprehensible and chaotic distribution of expert opinions, which conduces to a more complex and less understandable model.

VI. OBSERVATIONS AND CONCLUSIONS FROM THE STUDY

We have studied online-learning strategies suggested by Avrim Blum in his paper [1]. We have suggested some improvement in the algorithms used in this paper, by adding the concept of "Enhancement" technique, which considers the reward parameter in addition to the Penalization factor in conventional online learning theory. In addition, we have paved a way for streaming data to be used instead of batch-processed or already available data, for stock prediction.

The final dataset considered for evaluation of the models is derived from across various online sources, derived, as suggested earlier, through several streaming data platforms such as Apache Spark, Python and PySpark, followed by their effective storage and use as batch-data. The data consisting of stock data across several renowned companies over a period of 1000 days, that is approximately 4 years. In drawing the above observation we have used training set jSj, table o expert opinion, set jTj and test set. It may be noted that sets jTj; jSj and jV j are disjoint sets.

We can note from the above graph in Figure 2 that the change in weights is significant from one training subset to another, whereas the result of penalizing weights on the test set does no significantly change the weights across the subsets.

One possible reason for this is that the values of the subsets in jV j are significantly close to one another. However, this result may change over larger test sets. The expertise of the expert is to be viewed as his ability to predict transitions for the n stocks, where for which each wrong prediction, the expert would suffer a penalty[1]. Also, naturally, all these T daily experts would be predicting the stock value based on the actual value of stock transitions in these T previous days. Observe that the transition on an arbitrary day may or may not correlate with some training data set transitions.

DOI: 10.37200/IJPR/V24I8/PR280266
Received: 21 Jan 2020 | Revised: 08 Feb 2020 | Accepted: 14 Mar 2020

International Journal of Psychosocial Rehabilitation, Vol. 24, Issue 08, 2020
ISSN: 1475-7192

Results of the program code

The outcome (without using enhancement strategy in RWMA)

Expected: 1 | 1 | 1 | 1 | -1 | 1 | -1 | 1 | 1 | 1

Predicted: 1 | -1 | 1 | 1 | -1 | 1 | 1 | 1 | -1 | 1

The algorithm predicts correctly 60% of the time, and erroneously 40% of the time.

```
plt.scatter(y, x1, marker='x', color='blue')
plt.scatter(y, x2, marker='o', color='red')
plt.show()
```

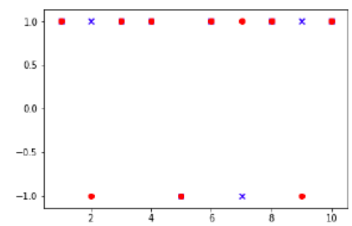

Figure 5: The above figure shows the correlation between the expected and predicted quantities is 60%

The outcome (using reinforcement strategy in RWMA):

Expected: -1 | -1 | -1 | -1 | -1 | -1 | 1 | -1 | -1 | -1

Predicted: -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1

In this case, it yields 90% accuracy, though the accuracy is experimentally found to vary from 80-100% depending on the test cases

```
plt.scatter(y, x4, marker='o', color='red')
plt.show()
```

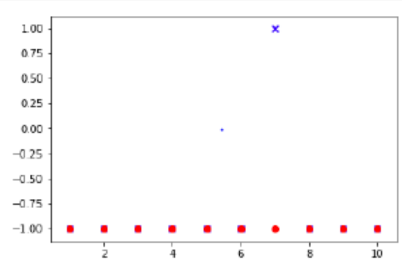

Figure 6: The above figure shows the correlation between the expected and predicted quantities is 90 %

Only one outlying point around the 7th observation is noted, all the other points collineate with the expected points.

Explanation

The group -1 is rewarded more across the test set, because there are more test cases that match the value -1, than there are cases that match 1 or 0, i.e., stock values have decreased a lot more often across the

DOI: 10.37200/IJPR/V24I8/PR280266
Received: 21 Jan 2020 | Revised: 08 Feb 2020 | Accepted: 14 Mar 2020

days, than it has increased or remained constant. Hence we get the group containing -1 as having the highest probability of being among selected from among the three.

We have a given set of experts whose expertise depends on the number of trends they can predict for the future of n stocks. Following is the tabl that shws the set of experts, let jTj The experts are trained on a training set, say jSj which is disjoint from jTj. In the training process, individual weights of the experts are initially assigned value of 1. As the experiment proceeds, for incorrect prediction on each training subset, a factor ofα is multiplied to the incorrect expert weights is subtracted from the total weight of that expert, similarly a factor of multiplied to the correctly predicting experts' weight is added to their total weight. After the training operation is completed, we move on to the test phase. In this, the experts encounter test data and are expected to apply their learning from training, to predict the data correctly.

However, on being mistaken, the cumulative weight of a group (0/ - 1/1) is penalized, or awarded on being correct. For example, if, across the test set, a particular stock value increases twice and falls once, the cumulative weight of group 1 will be rewarded twice and penalized once, whereas the group of -1 will be rewarded only once when the stock value falls, at the other two instances when the value rises, it will be penalized.

REFERENCES

1. Avrim Blum On-Line Algorithms in Machine Learning Carnegie Mellon University, Pittsburg PA 15213, 1998 Marcel Soares Ribeiro and Paulo Andre Lima de Castro: Online Learning Applied to Autonomous Valuation of Financial Assets IEEE, 2018 Shai Shalev-Shwartz: Online Learning: Theory, Algo- rithms, and Applications Senate of the Hebrew Univer- sity, July 2007Learning and Games Universita` degli Studi Milano,
2. Universit at Pompeu Fabra, Barcelona, July 2006 Peter Auer, Nicolò Ces A-Bianchi, Yaov Freund and Robert E. Schapire: The Nonsto chastic Multiarmed Bandit Problem Society for Industrial and Applied Math- ematics, 2002
3. Gidon Eshel The Yule Walker Equations for the AR Coefficients ShahidAli, Sreenivas SremathTirumala and Abdolhos- seinSarrafzadeh: Ensemble learning methods for de- cision making: Status and future prospects IEEE, July 2007
4. Sattar Vakili Qing Zhao and Yuan Zhou: Time-varying stochastic multi-armed bandit problems IEEE, July 2014
5. Mohammadzaman Zamani, Hamid Beigy, and Amirreza- Shaban: Cascading Randomized Weighted Majority: A New Online Ensemble Learning Algorithm IEEE, July 2014
6. Luca Canzian, Yu Zhang, Mihaela van der Schaar: En- semble of distributed learners for online classification of dynamic data streams IEEE, July 2015
7. C.V Krishna Veni, T SobhaRani: Ensemble based clas- sification using small training sets : A novel approach IEEE, 2014
8. Jiu-Zheng Liang: SVM Multi-classifier and Web- document classification IEEE, 2004
9. LuigiP. Cordella, ClaudioDeStefano, Francesco-
10. Fontanella, AlessandraScotto di Freca: A Weighted Majority Vote Strategy Using Bayesian Networks Springer, 2013
11. Abhishek Trupathi, Ashwin T S, Ram Mohana Reddy Guddeti: A Reinforcement Learning and Recurrent Neural Network Based Dynamic User Modeling Sys- tem IEEE, 2018
12. Nicolo-CesaBianchi, Gabor Lugosi, Cambridge Univer- sity Press: Prediction, Learning and Games

DOI: 10.37200/IJPR/V24I8/PR280266
Received: 21 Jan 2020 | Revised: 08 Feb 2020 | Accepted: 14 Mar 2020