Bayesian Ensembled Knowledge Extraction Strategy for Online Portfolio Selection

Abhishek Kumar
Center for Ubiquitous Computing
University of Oulu
Oulu, Finland
Department of Computer Science
University of Helsinki
Helsinki, Finland
abhishek.kumar@oulu.fi

Aviv Segev
Department of Computer Science
University of South Alabama
Mobile, USA
segev@southalabama.edu

***Abstract*—Online portfolio selection, one of the major fundamental problems in finance, has been explored quite extensively in recent years by machine learning and artificial intelligence communities. Recent state-of-the-art methods have focused on *Mean Reversion* significantly and have demonstrated outstanding performance. Another version of the same phenomenon, *Median Reversion*, has also performed well and demonstrated its ability to be robust against noises and outliers. Another important characteristic is *Momentum*. In this paper, a Bayesian ensembling approach to extract knowledge from both *Mean Reversion* and *Median Reversion* simultaneously based on the momentum associated with each one is proposed for the online portfolio selection task. The proposed method demonstrates its effectiveness by outperforming current state-of-the-art algorithms on several datasets.**

I. Introduction

Portfolio selection, one of the major fundamental problems in finance, deals with the allocation of wealth to a set of assets, the portfolio, so that the investor can attain certain financial objectives over a given period of time. This problem is studied in finance and also in mathematics, machine learning, and artificial intelligence [1]. It is generally studied in light of two mathematical models: mean-variance theory [2] and Kelly investment strategy [3]–[5]. Mean-variance theory deals with the trade-off between return (mean) and risk (variance) and is mainly suitable for single-period portfolio selection. Kelly investment is suitable for multi-period portfolio selection. Unlike any kind of trade-off, Kelly investment simply focuses on maximizing the expected log return of the portfolio. Recent work has exploited the concept of Kelly investment, especially the online portfolio selection.

One of the most significant characteristics in finance is *reversion*. It has been extensively studied in recent years in light of online portfolio selection. These algorithms based on *mean reversion* [6]–[9] have performed quite well on real datasets. Another variant of this property is *median reversion* [10]. Using *median reversion* is an ideal choice in the presence of noise and outliers. But if the dataset is relatively clean, using *mean reversion* is a better choice. Another important characteristic is *momentum*. Thus, we have the following problems: 1) How to exploit *reversion* and *momentum* and 2) When to exploit *mean reversion* and when to exploit *median reversion*.

To address these problems, we propose an approach which extracts knowledge from both *mean reversion* and *median reversion*, named "Bayesian Ensembled Mean-Median Reversion (BMMR) Based Strategy for Online Portfolio Selection". The basic idea is to ensemble both properties using Bayesian probability, then to learn from mistakes made in the previous prediction followed by explicit estimation of the portfolio using the online machine learning approach [11]. The proposed method outperforms current state-of-the-art methods on many real datasets in terms of the final *cumulative wealth*. Also, on other datasets it makes sure that anyone who invests based on this algorithm will have greater final cumulative wealth than anyone who follows a market-oriented approach, invests in index funds, and uses other algorithms [6], [12]–[14]. Also, the run time of BMMR is efficient and suitable for high frequency trading applications [15], [16].

The remainder of the paper is organized as follows. Section 2 describes the on-line portfolio selection setting. Section 3 reviews some recent works done in on-line portfolio selection. Section 4 describes the proposed BMMR approach in detail. Section 5 presents the experiments and comparative studies along with metrics used to measure the performance and the datasets used to test the performance. Finally, Section 6 offers some concluding remarks.

II. Problem Settings

A portfolio of *m* assets has been considered for the purpose of investment in financial markets. This investment lasts for *n* trading periods. On the t^{th} trading period, the price-relative vector for the *m* assets of the portfolio is represented by x_t = ($x_{t,1}$, $x_{t,2}$, ..., $x_{t,m}$) where element $x_{t,i}$ represents price-relative of asset *i* on period *t* and $x_{t,i}$ is formulated as ratio of the closing price of asset *i* on period *t* to the closing price of asset *i* on period *t*-1 i.e. $\frac{p_{t,i}}{p_{t-1,i}}$. Let x_1^n = (x_1, x_2, ..., x_n) be the sequence of the *m* dimensional price-relative vector for *n* periods starting from the beginning period 1.

The investment in the portfolio on the t^{th} trading period is represented by *portfolio weight vector* b_t = ($b_{t,1}$, $b_{t,2}$, ..., $b_{t,m}$). Each element $b_{t,i}$ represents the proportion of the wealth invested in asset *i* on the t^{th} trading period. The portfolio is assumed to be self financed, i.e. it requires financing only at the beginning of the investment period. It also assumes no margin or short selling. Therefore, each element of the portfolio vector is non-negative and all elements sum up to one i.e. $b_t \in \Delta_m$, where $\Delta_m = \{b_t : b_t \in R_+^m, \sum_{j=1}^{m} b_{t,j} = 1\}$. The investment procedure is represented by the *portfolio selection strategy*. This *portfolio selection strategy* is given as: $b_1 = \{\frac{1}{m}, ..., \frac{1}{m}\}$ and a sequence of mappings $R_+^{m(t-1)} \rightarrow \Delta_m$, t = 2, 3, ... where $b_t = b_t(x_1, x_2, ..., x_{t-1})$ is the portfolio invested on the t^{th} trading period having knowledge of past price-relatives of assets of the portfolio $x^{t-1} = \{x_1, x_2, ..., x_{t-1}\}$. The *portfolio selection strategy* for *n* trading period is given by b^n = (b_1, ..., b_n).

For the t^{th} trading period, an investment formulated by portfolio b_t leads to portfolio daily return m_t, i.e., on the t^{th} trading period, the wealth increases by the factor of c_t which is given by $b_t^T x_t = \sum_{i=1}^{m} b_{t,i} x_{t,i}$. Every next trading period, we reinvest the entire wealth accumulated so far and use price-relatives. This leads to multiplicative cumulative return of investments. Therefore, after *n* trading periods, the investment formulated by portfolio selection strategy b^n results in cumulative portfolio wealth C_n. By n^{th} trading period, the initial wealth increases by a factor of $\prod_{t=1}^{n} b_t^T x_t$. Therefore, the cumulative wealth by the n^{th} trading period is given by $C_n(b_1^n, x_1^n) = C_0 \prod_{t=1}^{n}(b_t^T x_t)$ where C_0 is the initial wealth, and the amount of wealth invested to create portfolio in the beginning is 1.

Now the problem, i.e. online Portfolio Selection, is dealt with as sequential decision making. The goal of the investor is to formulate investment strategy b_1^n such that the final portfolio cumulative wealth C_n is maximized. In the beginning of each t^{th} trading period, the investor has access to historical daily price-relatives till the last trading period, $t-1$. The investor does not know price relative information of portfolio assets because the market has not yet revealed the actual price. We assume that the market reveals the actual price of assets at the end of the period in the form of the closing price. So the investor uses the available information up to the $(t-1)^{th}$ trading period and formulates strategy b_t for the next daily price-relative x_t. The investor keeps repeating this procedure until the end of the trading period. Finally, the investment strategy is judged based on final cumulative wealth C_n. Algorithm 1 summarizes the problem setting.

To define the problem as in Algorithm 1, apart from no margin/short selling, certain assumptions have been made. No Transaction Cost: Although every time any change like buy/sell to any asset of the portfolio is made in the market it incurs a certain transaction fee, the model does not factor in transaction fees. No Tax: The model assumes no tax. High Market Liquidity: The model allows the investor to buy and sell desired quantities of portfolio assets at the last closing price. Impact Cost: The model assumes that the portfolio assets selection strategy suggested by it does not cause major disruptions in the market.

ALGORITHM 1: Online Portfolio Selection Setting

Input: Initialize Wealth $C_0 = 1$, Investment Strategy b_1 $=\{\frac{1}{m}, ..., \frac{1}{m}\}$, N trading periods
Output: Final Cumulative Wealth C_n
Initialization: $t = 1$
while $t \leq N$ **do**
  Investor formulates portfolio b_t based on historical price information (b^{t-1}, x^{t-1})
  Actual price-relative on trading period t is generated by the market i.e. x_t
  Portfolio daily return s_t = $b_t^T x_t$ and update cumulative wealth $C_t = C_{t-1} \times b_t^T x_t$
  Based on score of investment on t^{th} trading period, the investor updates its portfolio selection
  $t = t + 1$
end

III. Bayesian Ensembling

Financial data is highly complex in nature and contains multiple characteristics. The three major long-term characteristics are *reversion*, *momentum*, and *repetition of history*. Empirically, exploiting *reversion* leads to superior performance in terms of final cumulative of wealth. Most state-of-the-art methods focus on exploiting only one characteristic. The main hypothesis for the proposed method is that exploiting more characteristics may lead to more accurate prediction of price information and better performance. Therefore, the proposed method tries to exploit *reversion* and *momentum*.

These two characteristics are opposite in nature. The *reversion* characteristic is known as the *Follow the Loser* approach. The *momentum* characteristic is known as the *Follow the Winner* approach. Using these two characteristics means following the winning stocks as well as the losing stocks at the same time, which is a dilemma in itself. So the proposed method tries to exploit the *reversion* characteristic more efficiently. *Reversion* is exploited by either *mean* or *median*. *Mean* is a better choice on clean datasets. *Median* is a better choice on datasets containing noises or outliers. But the real-world dataset is mixed, i.e. not very clean, not very noisy. Hence the decision problem is when to use *mean reversion* and when to use *median reversion*. This decision problem is tackled by the *momentum* characteristic. If high momentum is associated with *mean reversion*, then *mean reversion* plays a significant role in predicting price information. If high momentum is associated with *median reversion*, then *median reversion* plays a significant role in predicting price information. This way, the proposed method intelligently tackles the decision problem of *mean* vs. *median* and combines both characteristics.

A. General Framework for Estimating Price Relatives

The first step in portfolio investment requires estimation of price information of portfolio assets on the next trading period $t+1$ as accurately as possible. With this information, the investor would be able to put more capital on profitable assets. Several methods involve prediction of price information, followed by an optimization step. Two recent methods *Online Moving Average Mean Reversion* (OLMAR) [9] and *Robust Median Reversion* (RMR) [10] have shown that stock prices possess a multi-period reversion property. This reversion property is exploited by two different approaches: 1) Mean Reversion [9] and 2) Median Reversion [10]. Empirically these two approaches show promising results. The Median Reversion approach is more robust because it is not affected by noises and outliers severely unlike the Mean Reversion approach.

We develop a generalized approach for estimating price information for the next trading period which exploits mean reversion and median reversion simultaneously. The approach proposes Bayesian Ensembling, which effectively exploits the reversion property from financial data by combining *OLMAR* [9] and *RMR* [10] to make a more accurate prediction of price-relatives for the next trading period. The Bayesian Ensembling approach for online portfolio selection involves sequential Bayesian updating. The general framework for prediction using *BMMR* is given as follows:

$$\tilde{x}_{t+1} = W_{t+1}^{OLMAR} * \tilde{x}_{t+1}^{OLMAR} + W_{t+1}^{RMR} * \tilde{x}_{t+1}^{RMR}$$

such that $W_{t+1}^{OLMAR} + W_{t+1}^{RMR} = 1$

where, $\tilde{x}_{t+1}$ is the predicted price-relative on the $(t+1)^{th}$ trading period, W_{t+1}^{OLMAR} is the Bayesian weight of *OLMAR* on $(t+1)^{th}$ trading period and is also quantification of momentum associated with the *mean reversion* characteristic, W_{t+1}^{RMR} is the Bayesian weight of *RMR* on $(t+1)^{th}$ trading period and is also the quantification of momentum associated with the *median reversion* characteristic, $\tilde{x}_{t+1}^{OLMAR}$ is predicted price-relative on $(t+1)^{th}$ trading period using the *OLMAR* approach and $\tilde{x}_{t+1}^{RMR}$ is predicted price-relative on $(t+1)^{th}$ trading period using the *RMR* approach.

B. Sequential Bayesian Updating

Sequential Bayesian Updating provides a way to combine two different methods. It does so by calculating weights (or associated momentum) to be allocated to each individual prediction. The Bayesian weights for each approach are calculated:

$$W_{t+1}^{OLMAR} = W_{t}^{OLMAR} * L_{t+1}^{OLMAR};$$

Where W_{t+1}^{OLMAR} is weight (posterior probability) for the *OLMAR* approach on the $(t+1)^{th}$ trading period, W_{t}^{OLMAR} is weight (prior probability) on the t^{th} trading period, and L_{t+1}^{OLMAR} is the likelihood function for *OLMAR* approach on the $(t+1)^{th}$ trading period. As mentioned, this weight (posterior probability) is the quantification of momentum associated with the *OLMAR* approach.

According to Officer [17], stock returns tend to follow normal distribution which is an exponential function. So the Likelihood function for *OLMAR* on any $(t+1)^{th}$ trading period is given as:

$$L_{t+1}^{OLMAR} = \frac{e^{-\sum_{i=1}^{m}(\tilde{x}_{t,i}^{OLMAR}-x_{t,i})^2}}{e^{-\sum_{i=1}^{m}(\tilde{x}_{t,i}^{OLMAR}-x_{t,i})^2}+e^{-\sum_{i=1}^{m}(\tilde{x}_{t,i}^{RMR}-x_{t,i})^2}}$$

where, $\tilde{x}_{t,i}^{OLMAR}$ is predicted price-relative for asset i for t^{th} trading period by the *OLMAR* prediction approach, $\tilde{x}_{t,i}^{RMR}$ is predicted price-relative for asset i for t^{th} trading period by the *RMR* prediction approach, $x_{t,i}$ is the actual price-relative for asset i for t^{th} trading period as revealed by the market, and m is total number of assets in the portfolio. Similarly, the likelihood function and weight function for *RMR* can be also calculated.

C. Portfolio Optimization

The next step requires deciding the proportion of the wealth being allocated to each asset of the stock based on the expected price-relatives of each asset received from the previous step. We adopted the idea of Passive Aggressive (PA) Online Learning [11] to exploit mean reversion and median reversion for getting the maximized final cumulative wealth. It is similar to the one used in PAMR [7], OLMAR [18], and RMR [10]. The equation aims at finding the optimal portfolio by minimizing the deviation from the difference from the last portfolio, while satisfying $b \cdot \hat{x}_{t+1} \geq \epsilon$. The formulation effectively exploits reversion in price-relative information.

$$b_{t+1} = \operatorname*{argmin}_{b\in\Delta_m} \tfrac{1}{2}\|b-b_t\|^2 \text{ such that } b\cdot\tilde{x}_{t+1} \geq \epsilon \quad (1)$$

If the constraint of this formulation is satisfied, that is, if the expected return is higher than the threshold ϵ, then the resulting portfolio on the $(t+1)^{th}$ trading period is the same as the previous portfolio on the t^{th} trading period. If the constraint is not satisfied, then it tries to calculate a new portfolio for the $(t+1)^{th}$ trading period such that the expected return is greater than the threshold ϵ, minimizing the distance of this new portfolio from the previous portfolio, the one on the t^{th} trading period. The algorithm *BMMR* explicitly involves the reversion idea as it requires determination of the price-relative $\hat{x}_{t+1}$ for the next trading period $(t+1)^{th}$. Following the work of [11], [18], [19], the solution of the optimization problem (1) in terms of Lagrangian multiplier is

$$b_{t+1} = b_t + \lambda_{t+1}(\tilde{x}_{t+1} - \bar{x}_{t+1}\cdot 1)$$

where $\bar{x}_{t+1} = \frac{1\cdot\tilde{x}_{t+1}}{m}$ and $\lambda_{t+1} = \max\{0, \frac{\epsilon - b_t\cdot\tilde{x}_{t+1}}{\|\tilde{x}_{t+1}-\bar{x}_{t+1}1\|^2}\}$.

Finally, to ensure that the portfolio b_{t+1} is non-negative, it is projected to the simplex domain [20].

IV. Algorithm

The first step of the model involves calculation of the expected relative price on the next trading period. Using the *RMR* approach, L_1 median is determined as the expected price for the next trading and price-relative for the next period is determined as the ratio to L_1 price to price on the $(t-1)^{th}$ trading period [10]. Algorithm 2 shows steps to determine L_1 median. It uses the Modified Weiszfeld Algorithm [21] for it.

Algorithm 3 shows steps to estimate price-relatives of portfolio assets on the $(t+1)^{th}$ trading period based on *OLMAR* [9]. This algorithm is responsible for exploiting *mean reversion* characteristics in the financial market.

Algorithm 4 shows steps to determine final price-relatives through Bayesian ensembling of *mean reversion* and *median reversion* characteristics of the financial market as given by Algorithm 2 and Algorithm 3 respectively.

ALGORITHM 2: L_1 Median: Determining expected price-relative on next trading period

Input: Price p_t, p_{t-1}, ..., p_{t-w+1}; Maximum number of iterations m; Tolerance Level λ
Output: Expected price-relative $\tilde{x}_{t+1}$
Initialization: $j = 2$;
$\mu_1 = median(p_t, p_{t-1}, ..., p_{t-w+1})$;
$\|.\|$ represents Euclidean norm
while $j \leq m$ **do**
$\eta(\mu) = \begin{cases} 1, & \text{if } \mu = p_{t-j} \\ 0, & \text{otherwise} \end{cases}$
$\tilde{R}(\mu) = \sum_{p_{t-j} \neq \mu} \frac{p_{t-j} - \mu}{\|p_{t-j} - \mu\|}$
$\gamma(\mu) = \|\tilde{R}(\mu)\|$
$\tilde{\mu} = \frac{\sum_{p_{t-j} \neq \mu} \frac{p_{t-j}}{\|p_{t-j} - \mu\|}}{\sum_{p_{t-j} \neq \mu} \frac{1}{\|p_{t-j} - \mu\|}}$
$\mu = min(1, \frac{\eta(\mu)}{\gamma(\mu)}) + \tilde{\mu}(1 - \frac{\eta(\mu)}{\gamma(\mu)})^+$
if $(\|\mu_{j-1} - \mu_j\| \leq \lambda \|\mu_j\|)$ **then**
break
end
end
$\tilde{x}_{t+1} = \frac{\mu}{p_t}$

ALGORITHM 3: OLMAR: Determining expected price-relative on next trading period

Input: window size w, Historical price-relative information up to t^{th} trading period: x_1^{t-1}
Output: Expected price-relative $\tilde{x}_{t+1}$
$\tilde{x}_{t+1} = \frac{MovingAverage_t(w)}{p_t} = \frac{1}{w}(1 + \frac{1}{x_t} + ... + \frac{1}{\odot_{i=0}^{w-2} x_{t-i}})$
$\odot$ is element wise product

Algorithm 5 determines the investment strategy on the $(t+1)^{th}$ trading period using price-relative information predicted by Algorithm 4, Algorithm 3, and Algorithm 2. It is responsible for deciding how much capital should be invested in any particular asset of the portfolio on the $(t+1)^{th}$ trading period. It uses an online passive aggressive machine learning algorithm to decide allocation [9], [11].

Algorithm 6 shows the overall framework to determine investment strategy on the $(t+1)^{th}$ trading period which exploits both *mean reversion* and *median reversion* characteristics of the financial market and combines them using the Bayesian Ensembling approach.

The time complexity of trading algorithms plays a significant role in a high frequency trading environment where thousands of transactions take place in fractions of seconds [22]. BMMR is linear with respect to number of stocks in the portfolio ***m*** and number of trading periods ***n***. Algorithm 2 is implemented in *O(l)* as it involves a maximum number of ***l*** iterations. Algorithm 3 takes *O(m)* per period. In addition, Algorithm 4 takes *O(m)* per period. The total time complexity of algorithm BMMR is *O(mn) + O(mn) + O(ln)*, or *O(mn)+ O(ln)*.

ALGORITHM 4: Bayesian Ensembling: Ensembling price-relatives predicted by Algorithm 2 and Algorithm 3

Input: Historical Price Information: p_1^t, Historical Price-Relative Information: x_1^t, Maximum Number of Iterations: m, Tolerance Level: λ
Output: Expected price-relative $\tilde{x}_{t+1}$ on the $(t+1)^{th}$ trading period
Initialization: $W_1^{OLMAR} = W_1^{RMR} = 0.5$,
$\mu_1 = median(p_t, p_{t-1}, ..., p_{t-w+1})$;
Expected Price-Relative given by Algorithm 2
$\tilde{x}_{t+1}^{RMR}$ = Algorithm 2: $L_M edian(p_1^t, m, \lambda)$
Expected Price-Relative given by Algorithm 3
$\tilde{x}_{t+1}^{OLMAR}$ = Algorithm 3: $OLMAR(x_1^t, w)$
Calculate final expected price on $(t+1)^{th}$ trading period using Bayesian ensembling;
$\tilde{x}_{t+1} = W_{t+1}^{RMR} * \tilde{x}_{t+1}^{RMR} + W_{t+1}^{OLMAR} * \tilde{x}_{t+1}^{OLMAR}$
Update weights;
$L_{t+1}^{RMR} = \frac{e^{-\sum_{i=1}^{m}(\tilde{x}_{t,i}^{RMR} - x_{t,i})^2}}{e^{-\sum_{i=1}^{m}(\tilde{x}_{t,i}^{RMR} - x_{t,i})^2} + e^{-\sum_{i=1}^{m}(\tilde{x}_{t,i}^{RMR} - x_{t,i})^2}}$
$W_{t+1}^{RMR} = W_t^{RMR} * L_{t+1}^{RMR}$,
$L_{t+1}^{OLMAR} = \frac{e^{-\sum_{i=1}^{m}(\tilde{x}_{t,i}^{OLMAR} - x_{t,i})^2}}{e^{-\sum_{i=1}^{m}(\tilde{x}_{t,i}^{OLMAR} - x_{t,i})^2} + e^{-\sum_{i=1}^{m}(\tilde{x}_{t,i}^{RMR} - x_{t,i})^2}}$
$W_{t+1}^{OLMAR} = W_t^{OLMAR} * L_{t+1}^{OLMAR}$

ALGORITHM 5: BMMR: Determining allocation of the wealth to be invested in each portfolio asset on the $(t+1)^{th}$ trading period

Input: Threshold Reversion ϵ, Predicted next price-relative $\tilde{x}_{t+1}$, Current Portfolio b_t
Output: Next Portfolio b_{t+1}
Calculate the predicted Market Return:
$\bar{x}_{t+1} = \frac{1^T \tilde{x}_{t+1}}{m}$
Calculate hinge loss function:
$loss = \epsilon - b_t \cdot \tilde{x}_{t+1}$
Calculate the Lagrangian multiplier:
$\lambda_{t+1} = \max\{0, \frac{loss}{\|\tilde{x}_{t+1} - \bar{x}_{t+1} \cdot 1\|^2}\}$
Update the portfolio:
$b_{t+1} = b_t + \lambda_{t+1}(\tilde{x}_{t+1} - \bar{x}_{t+1} \cdot 1)$
Normalize the portfolio b_{t+1}:
$b_{t+1} = \underset{b \in \Delta_m}{\operatorname{argmin}} \|b - b_{t+1}\|^2$

ALGORITHM 6: Online Portfolio Selection using BMMR

Input: Window w ≥ 2; price-relative x_1^n: From 1^{st} trading period to t^{th} trading period; Maximum number of iterations m; Tolerance Level λ; Threshold Reversion ϵ; Initial Wealth C_0; Initial Investment Strategy b_0 =$\{\frac{1}{m}$, ..., $\frac{1}{m}\}$
Output: C_n: Final cumulative wealth after n^{th} trading period
Initialization: $i = 1$; $t = 1$; Price Information $p_0 = 1$;
Convert price-relative information into actual price information on any period t
while $t \leq n$ **do**
    Extract price information from price-relative information $p_t = p_{t-1} \cdot x_i$
    Investor formulates portfolio b_t based on historical price information (b^{t-1}, x^{t-1}, p^{t-1})
    Actual price-relative on trading period t is generated by the market i.e. x_t
    Portfolio daily return is given by $b_t^T x_t$ and the cumulative wealth is updated to $C_t = C_{t-1} \times b_t^T x_t$
    Investor predicts price-relatives of portfolio assets for the next trading period t + 1:
        $\tilde{x}_{t+1}$= Algorithm 4: Bayesian Ensembling(p_1^t,x_1^t,m,λ)
    Investor updates its portfolio selection strategy according to:
        b_{t+1}= Algorithm 5: BMMR(ϵ, $\tilde{x}_{t+1}$, b_t)
    $t = t + 1$
end

V. Experiments

A. Datasets

In the experiment, 6 different datasets were used. Each dataset contains historical relative daily prices of stocks. Each dataset is publicly available. Moreover, the datasets can extracted from the public domain like Google Finance and Yahoo Finance etc. All these 6 datasets represent several financial markets.

1) **NYSE (O)**: It is one of the oldest and standard benchmark datasets for testing any kind of financial optimization problem involving stock prices. It is named as NYSE (Old) or simply NYSE (O). It was first used by [23] and later on by [24], [18], [6], [6], [25], [14], and [13] to devise various types of investment strategies. This dataset contains 5651 daily price-relatives of 36 stocks listed on New York Stock Exchange Market for a period of 22 years starting from 3 July 1962 to 31 December 1984.
2) **NYSE (N)**: This dataset is the extended version of the previous NYSE dataset. It is called NYSE (New) or simply NYSE (N). It contains historical daily price-relatives of 23 stocks listed on New York Exchange Market for period of 6431 trading periods starting from 1 January 1985 to 30 June 2010. It contains fewer stocks than NYSE (O) contains because some companies were taken over by other companies or went bankrupt.
3) **TSE**: This dataset was collected by [6]. It contains daily price-relatives of 88 stocks from Toronto Stock Exchange for a period of 1259 trading periods starting from 4 January 1994 to 31 December 1998.
4) **SP500**: This dataset was also collected by [6]. It contains daily price-relatives of 25 companies which have the largest market capitalization among 500 SP500 companies for a period of 1276 trading periods starting from 2 January 1998 to 31 January 2003.
5) **MSCI**: This dataset consists of global equity indices. These indices are used to make the MSCI World Index. This dataset is a collection of 24 indices which represent the equity markets of 24 countries around the globe. The dataset contains daily price-relatives for a period of 1043 trading periods starting from 1 April 2006 to 31 March 2010. It is maintained by MSCI Inc., previously known as Morgan Stanley Capital International.
6) **DJIA**: This dataset is a collection of 30 Dow Jones composite stocks. It contains daily price-relatives for a period of 507 trading periods starting from 14 January 2001 to 14 January 2003.

In Table I, these 6 datasets have been summarized.

B. Metrics

Next, we describe various metrics which we used to measure the performance of our algorithm.

1) **Cumulative Wealth**: Cumulative Wealth is the aggregate amount that the given investment has generated over a fixed period of time. The higher its value, the better the investment strategy.
2) **Standard Deviation**: Standard Deviation is a measure of the dispersion of a set of data from its mean. In finance, it is used to measure volatility of a given investment strategy. It is widely used as a metric to estimate the amount of expected volatility. This metric is quite important for risk averse investors. The lower the value, the better the investment strategy.
3) **Sharpe Ratio**: Sharpe Ratio is a measure for calculating risk-adjusted return, and this ratio has become the industry standard for such calculations. The Sharpe Ratio is the average return earned in excess of the risk-free rate per unit of volatility or total risk. The performance associated with risk-taking activities is determined by subtracting the risk-free rate from the mean return. The higher the value of the Sharpe Ratio, the better the investment strategy.
4) **Calmar Ratio**: It is a comparison of the average annual compounded rate of return and the maximum drawdown risk of commodity trading advisors and hedge funds. The lower its value, the worse the performance of investment on a risk-adjusted basis over the specified time period; the higher the Calmar Ratio, the better it performed. In general, the time period used is three years, but this

TABLE I
SUMMARY OF ALL SIX DATASETS FROM REAL FINANCIAL MARKETS.

Dataset	Market	Region	Time Frame	#Trading periods	#Assets
NYSE (O)	Stock	USA	3 July 1962 - 31 December 1994	5651	36
NYSE (N)	Stock	USA	1 January 1985 - 30 June 2010	6431	23
TSE	Stock	Canada	4 January 1994 - 31 December 1998	1259	88
SP500	Stock	USA	2 January 1998 - 31 January 2003	1276	25
MSCI	Index	Global	1 April 2006 - 31 March 2010	1043	24
DJIA	Stock	USA	14 January 2001 - 14 January 2003	507	30

can be higher or lower based on the investment under consideration.

5) **Sortino Ratio**: It is a modification of the Sharpe Ratio that differentiates harmful volatility from general volatility by taking into account the standard deviation of negative asset returns, called downside deviation. The Sortino Ratio subtracts the risk-free rate of return from the portfolios return, and then divides that by the downside deviation. A large Sortino Ratio indicates there is a low probability of a large loss. So, the higher the value of the Sortino Ratio, the better the investment strategy. It is calculated as follows:
   Sortino Ratio = $\frac{<R>-R_f}{\sigma_d}$
   Where $< R >$ is the Expected Return, R_f is the Risk free Rate of Return, and σ_d is Standard Deviation of Negative Asset Returns.
6) **Maximum Drawdown**: A Maximum Drawdown (MDD) is the maximum loss from a peak to a trough of a portfolio, before a new peak is attained. Maximum Drawdown is an indicator of downside risk over a specified time period. It can be used both as a standalone measure or as an input into other metrics such as "Return over Maximum Drawdown" and Calmar Ratio. The lower its value, the better the investment strategy. Maximum Drawdown is expressed in percentage terms and computed as:

   (Trough Value - Peak Value)/Peak Value

The performance of the proposed algorithm has been measured and presented using all six metrics, but the primary focus is on Cumulative Wealth.

C. Comparison Approaches

The proposed algorithm has been compared with a number of existing benchmark algorithms and other non benchmark algorithms having better empirical performance, introduced by the Computer Science community. Here, a list of these algorithms is provided with their parameters as introduced in their original studies.

1) **Market**: Simply once buy-then-hold-uniformly. No parameters.
2) **Best Stock**: Follow the best performing stock in the hindsight. Knowledge of the best stock is known in hindsight. No parameters.
3) **BCRP**: Follow the Best Constant Rebalanced Portfolio strategy in hindsight. No parameters.
4) **UP**: This algorithm was introduced by [23]. We focus on its implementation by [26]. Its parameters are: $\delta_0 = 0.004$, $\delta = 0.005$, $m = 100$, and $S = 500$.
5) **EG**: Exponential Gradient Algorithm with learning rate parameter: $\eta = 0.05$ [25].
6) **ONS**: On-line Newton Step introduced with parameters: $\eta = 0$, $\beta = 1$, $\gamma = 0.125$ [24].
7) **Anticor**: Exploiting *Mean Reversion* characteristics using cross-correlation and auto-correlation [6]. No parameters.
8) **B**^{K}: Nonparametric Kernel based moving window strategy with parameters: $w = 5$, $L = 10$, and correlation coefficient threshold $\epsilon = 0.1$ [14].
9) **B**^{NN}: Non Parametric Nearest Neighbour based strategy with parameters: $w = 5$, $L = 10$, and $p_l = 0.02 + 0.5(l-1)/(L-1)$ [13].
10) **CORN**: Correlation Driven Non-Parametric based strategy with parameters: $w = 5$, $p = 1$, and $\rho = 0.1$ [12].
11) **PAMR**: Passive Aggressive Mean Reversion with parameter: $\epsilon = 0.5$ [7].
12) **CWMR**: Confidence Weighted Mean Reversion with parameter: $\epsilon = 0.5$ [8].
13) **OLMAR**: On-Line Moving Average Reversion with parameters: $\epsilon = 10$ and $w = 5$ [18].
14) **RMR**: Robust Median Reversion with parameters: $\epsilon = 10$ and $w = 5$ [10].

D. Results

1) Cumulative Wealth: Table II presents the Cumulative Wealth achieved by all algorithms. BMMR outperforms all state-of-the-art methods and benchmark methods in datasets NYSE(N), SP500, and DJIA. In datasets NYSE(O), TSE, and MSCI, it outperforms all benchmark algorithms and most non benchmark algorithms.

2) Sharpe Ratio: Figure 1 shows the Sharpe Ratio achieved by BMMR and benchmarks as well as other state-of-the-art methods on all six different datasets. Higher Sharpe Ratio is always preferred. From Figure 1 it can be observed that the Sharpe Ratio of BMMR is more or less the same as achieved by other methods. So it can be concluded that for a given amount of risk, the return produced by BMMR is high, thus making it a safe investment strategy.

3) Standard Deviation: Figure 2 shows the Standard Deviation achieved by BMMR and benchmarks as well as other state-of-the-art methods on all six datasets. Standard

TABLE II
CUMULATIVE WEALTH ACHIEVED BY VARIOUS STRATEGIES ON THE SIX DATASETS. BEST RESULTS IN EACH DATASET ARE HIGHLIGHTED IN **BOLD**.

Method	NYSE (O)	NYSE (N)	TSE	SP500	MSCI	DJIA
Market	14.50	18.06	1.61	1.34	0.91	0.76
Best - Stock	54.14	83.51	6.28	3.78	1.50	1.19
BCRP	250.60	120.32	6.78	4.07	1.51	1.24
UP	26.68	31.49	1.60	1.62	0.92	0.81
EG	27.09	31.00	1.59	1.63	0.93	0.81
ONS	109.19	21.59	1.62	3.34	0.86	1.53
B^K	1.08E+09	4.64E+03	1.62	2.24	2.64	0.68
B^{NN}	3.35E+11	6.80E+04	2.27	3.07	13.47	0.88
CORN	1.48E+13	5.37E+05	3.56	6.35	**26.10**	0.84
Anticor	2.41E+08	6.21E+06	39.36	5.89	3.22	2.29
PAMR	5.14E+15	1.25E+06	264.86	5.09	15.23	0.68
CWMR	6.49E+15	1.41E+06	332.62	5.90	17.28	0.68
OLMAR	4.04E+16	2.24E+08	**424.80**	5.83	16.33	2.12
RMR	**1.64E+17**	3.25E+08	181.34	8.28	16.76	2.67
BMMR	9.43E+16	**4.74E+08**	60.97	**12.79**	14.02	**2.77**

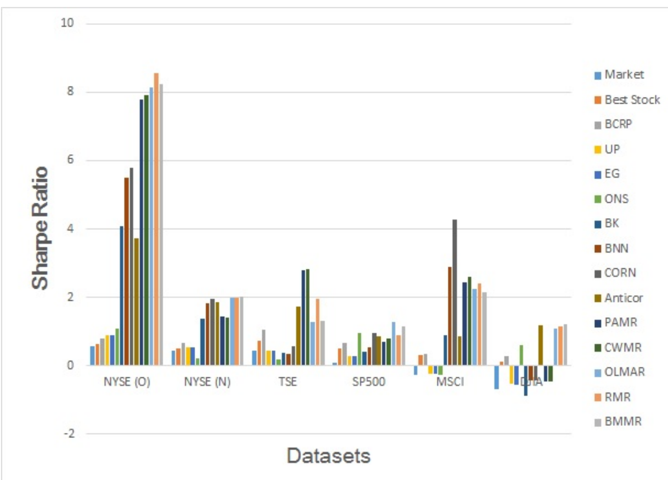

Fig. 1. Sharpe Ratio

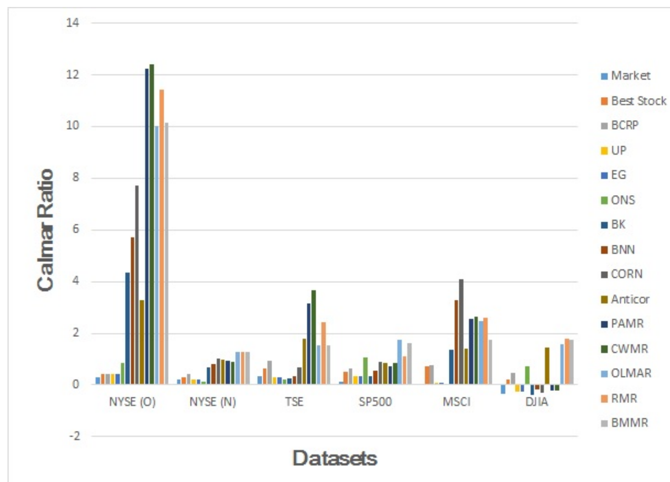

Fig. 3. Calmar Ratio

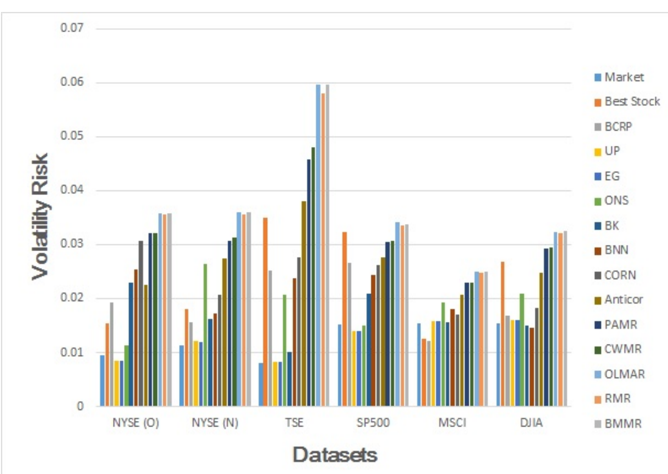

Fig. 2. Volatility Risk - Standard Deviation

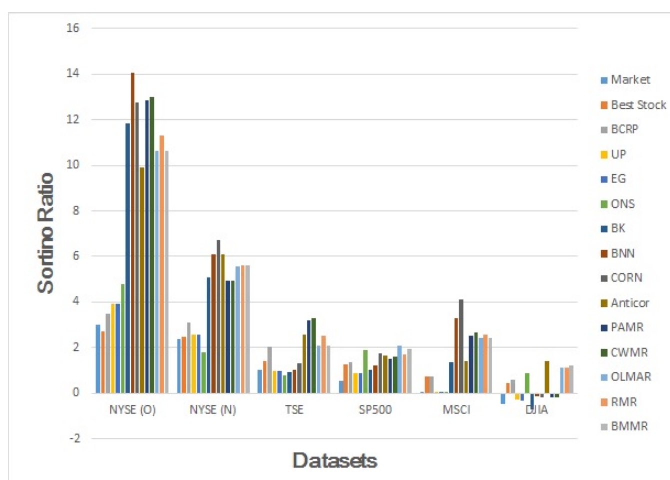

Fig. 4. Sortino Ratio

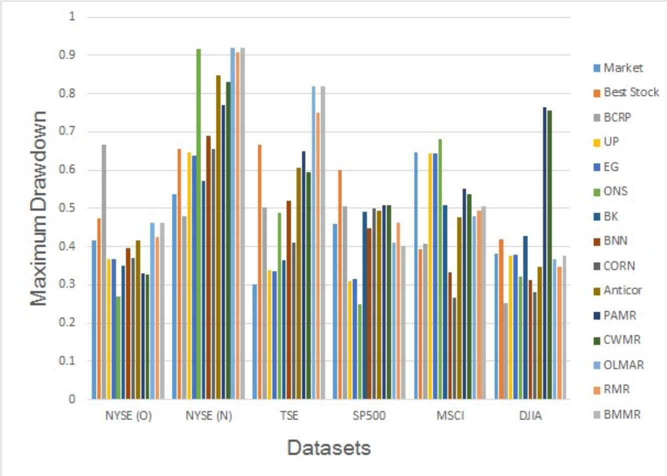

Fig. 5. Maximum Drawdown

Deviation is considered a metric to measure **Volatility Risk**. In the financial market, higher return always come with higher risk. The volatility risk of BMMR is comparable to the other algorithms which also achieved nearly the same return. In light of the cumulative wealth achieved by BMMR, the Volatility Risk is justified.

4) Calmar Ratio: Figure 3 shows the Calmar Ratio achieved by all methods. The Calmar Ratio of BMMR is not drastically low: it is more or less the same as other state-of-the-art algorithms. Generally a higher Calmar Ratio is preferred. It shows that BMMR performs as well or almost the same as other state-of-the-art methods on a risk-adjusted basis over the specified time period.

5) Sortino Ratio: Figure 4 shows the Sortino Ratio achieved by BMMR and other methods on all datasets. The Sortino Ratio achieved by BMMR is quite high. It indicates that the chance of suffering a large loss is quite low while following this BMMR based strategy. It confirms that BMMR is relatively stable in terms of its wealth generation capacity.

6) Maximum Drawdown: Figure 5 shows performance of all methods based on *Maximum Drawdown*. Results in several datasets show that Maximum Drawdown of BMMR is not significantly high in light of its wealth and is nearly the same as other state-of-the-art algorithms. The chance of BMMR declining from its historical peak is relatively low. So BMMR is not a high risk strategy.

E. Performance Analysis Dataset-Wise

Table II shows the proposed method achieves superior performance in 3 out of 6 datasets. Performance in each dataset depends on characteristics embedded in it. We discuss each dataset individually to find conditions or characteristics under which the proposed algorithm achieves the superior performance.

NYSE (O): The oldest benchmark dataset was recorded from 1962 to 1994 from the New York Stock Exchange. During this time, sophisticated hardware was not used well to record data and to process transactions. As a result, any algorithm using this dataset must be robust to noises and outliers. *Median reversion* offers the most ideal choice due to its robust nature. RMR is entirely based on *median reversion* and achieves the most superior performance. The proposed method sometimes uses *mean reversion* and sometimes uses *median reversion*, but not only *median reversion* like RMR.

NYSE (N): This dataset was recorded from 1985 to 2010 from the New York Stock Exchange. During this time, computers were used to record data and to process transactions. The hardware used at the exchange was relatively sophisticated. As a result, this dataset is relatively clean, i.e. cleaner than NYSE (O). Empirical experiments have already shown that reversion characteristics are quite significant for this dataset. The proposed method uses both *mean* and *median* to exploit reversion characteristics and achieves the most superior performance.

TSE: This dataset was recorded between 1994 and 1998 from the Toronto Stock Exchange (Canada). During this time, the Canadian stock market was stable, unlike the financial crisis in the East Asian stock market. Since there were no major ups and downs, noises, or outliers, there is no need to use a robust metric like median. In the absence of noises or outliers, the reversion characteristic is best exploited by *mean*. So *mean reversion* based OLMAR achieves the best performance.

SP500: This dataset of 25 companies with the largest market capitalization among all SP500 companies was recorded between 1998 and 2003. This dataset of 25 companies include Internet-based companies like Microsoft, Apple, Verizon, and Amazon, which suffered from the Dot-Com bubble, while other big companies suffered from the East Asian financial crisis. Due to these crises, stock prices of these companies suffered from major ups and downs and outliers. The dataset is relatively clean due to the sophisticated hardware used to store and process transactions. Empirical experiments have shown that the reversion characteristic is quite significant for this type of dataset. The proposed method uses both *mean* and *median* to exploit reversion characteristics and achieves the most superior performance.

MSCI: This dataset was recorded between 2006 and 2010, when the worst global financial crisis since the 1929 financial crisis was experienced. During a major global crisis, prices of all stocks of the portfolio become correlated. This characteristic can be exploited via learning through history, and less from *reversion* and *momentum* characteristics during the period of a major crisis. So the method which learns from historic correlation, CORN, achieves the best performance on this dataset. The proposed method does not exploit this characteristic, resulting in lower performance.

DJIA: This dataset, recorded between 1998 and 2003, contains 30 Dow Jones composite stock prices, including companies like Microsoft, Intel, and AT&T which suffered from the Dot-Com bubble that caused major ups and downs and outliers in stock prices. The dataset is relatively clean due to the sophisticated hardware used to store and process transactions. As mentioned, the reversion characteristic is quite significant for this dataset. The proposed method, which uses both *mean* and *median* to exploit the reversion characteristic, again achieves the most superior performance.

VI. Conclusion

This paper proposes a novel multi-period online portfolio selection strategy BMMR, which exploits the *reversion* and the *momentum* characteristics and tackles the decision problem of *mean reversion* vs. *median reversion* using a Bayesian ensembling approach. The results show that the proposed method achieves superior performance when the dataset contains the reversion property and a certain amount of noises/outliers. The proposed method works well because it exploits *reversion* efficiently and also is robust to noises and outliers. The algorithm is efficient computationally, making it quite suitable for large scale and high frequency trading. Future research includes different ensemble learning models like Bagging, Boosting, and Stacking to tackle the decision problem.

References

[1] B. Li and S. Hoi, "Online portfolio selection: A survey," *ACM Computing Surveys (CSUR)*, vol. 46, no. 3, p. 35, 2014.

[2] H. Markowitz, "Portfolio selection," *Journal of Finance*, vol. 7, no. 1, pp. 77–91, 1952.

[3] J. J. Kelly, "A new interpretation of information rate," *IRE Transactions on Information Theory*, vol. 2, no. 3, pp. 185–189, 1956.

[4] L. Breiman, "Optimal gambling systems for favorable games," *The Kelly Capital Growth Investment Criterion*, p. 47, 1961.

[5] M. Finkelstein and R. Whitley, "Optimal strategies for repeated games," *Advances in Applied Probability*, 1981.

[6] A. Borodin, R. El-Yaniv, and V. Gogan, "Can we learn to beat the best stock," *Journal of Artificial Intelligence Research*, pp. 579–594, 2004.

[7] B. Li, S. Hoi, P. Zhao, and V. Gopalkrishnan, "Confidence weighted mean reversion strategy for online portfolio selection," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 7, no. 1, p. 4, 2013.

[8] B. Li, C. Hoi, P. Zhao, and V. Gopalkrishnan, "Confidence weighted mean reversion strategy for on-line portfolio selection," in *Proceedings of the JMLR: Workshop and Conference*, vol. 15, 2011, p. 434.

[9] B. Li and S. Hoi, "On-line portfolio selection with moving average reversion," in *Proceedings of the 29th ACM International Conference on Machine learning*, 2012.

[10] D. Huang, J. Zhou, B. Li, S. Hoi, and S. Zhou, "Robust median reversion strategy for on-line portfolio selection," in *Proceedings of the 23th International Joint Conference on Artificial Intelligence*. AAAI Press, 2013, pp. 2006–2012.

[11] K. Crammer, O. Dekel, J. Keshet, S. Shalev-Shwartz, and Y. Singer, "Online passive-aggressive algorithms," *Journal of Machine Learning Research*, vol. 7, pp. 551–585, 2006.

[12] B. Li, S. Hoi, and V. Gopalkrishnan, "Corn: Correlation-driven nonparametric learning approach for portfolio selection," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 2, no. 3, p. 21, 2011.

[13] L. Györfi, F. Udina, and H. Walk, "Nonparametric nearest neighbor based empirical portfolio selection strategies," *Statistics & Decisions International Mathematical Journal for Stochastic Methods and Models*, vol. 26, no. 2, pp. 145–157, 2008.

[14] L. Györfi, G. Lugosi, and F. Udina, "Nonparametric kernel-based sequential investment strategies," *Mathematical Finance*, vol. 16, no. 2, pp. 337–357, 2006.

[15] I. Domowitz and H. Yegerman, "The cost of algorithmic trading: a first look at comparative performance," *Trading*, vol. 1, no. 1, pp. 33–42, 2006.

[16] R. Almgren and N. Chriss, "Optimal execution of portfolio transactions," *Journal of Risk*, vol. 3, pp. 5–40, 2001.

[17] R. Officer, "The distribution of stock returns," *Journal of the American Statistical Association*, vol. 67, no. 340, pp. 807–812, 1972.

[18] B. Li, S. Hoi, D. Sahoo, and Z. Liu, "Moving average reversion strategy for on-line portfolio selection," *Artificial Intelligence*, vol. 222, pp. 104–123, 2015.

[19] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[20] J. Duchi, S. Shalev-Shwartz, Y. Singer, and T. Chandra, "Efficient projections onto the l1 ball for learning in high dimensions," in *Proceedings of the 25th ACM International Conference on Machine Learning*, 2008, pp. 272–279.

[21] Y. Vardi and C. Zhang, "The multivariate l1-median and associated data depth," *Proceedings of the National Academy of Sciences*, vol. 97, no. 4, pp. 1423–1426, 2000.

[22] I. Aldridge, *High-frequency trading: a practical guide to algorithmic strategies and trading systems*. John Wiley and Sons, 2009, vol. 459.

[23] T. M. Cover, "Universal portfolios," *Mathematical Finance*, vol. 1, no. 1, pp. 1–29, 1991.

[24] A. Agarwal, E. Hazan, S. Kale, and R. E. Schapire, "Algorithms for portfolio management based on the newton method," in *Proceedings of the 23rd ACM International Conference on Machine Learning*, 2006, pp. 9–16.

[25] D. P. Helmbold, R. E. Schapire, Y. Singer, and M. K. Warmuth, "On-line portfolio selection using multiplicative updates," *Mathematical Finance*, vol. 8, no. 4, pp. 325–347, 1998.

[26] A. Kalai and S. Vempala, "Efficient algorithms for universal portfolios," *Journal of Machine Learning Research*, vol. 3, pp. 423–440, 2003.