Comparing Sparse Index Tracking Algorithms for Passive Investing

Mode: Application

Aditya Singh
Department of Mechanical Engineering
Indian Institute of Technology Kanpur
adisingh@iitk.ac.in

Varun Rajesh Gadre
Department of Mechanical Engineering
Indian Institute of Technology Kanpur
vgadre@iitk.ac.in

***Abstract*—Index tracking strategy enjoys unrivaled popularity within the passive investors circle. Sparse index tracking, a regression problem suited for portfolio management without high capital requirement constraints, is a well studied strategy providing a good reflection of the market performance. We review three papers demonstrating algorithms to handle the problem, comparing their performance and scalability.**

I. Introduction

Index investing is the most popular passive investment strategy to *not* beat the market in the long run. Since passive investors believe that none can outperform the markets eventually, a portfolio which can replicate the market's performance are of paramount interest to them. Because index are seen as a measure of the market, tracking the index is the major goal.

An index generally comprises of a large set of stocks in varying proportions which are revised over time. Creating a portfolio comprising of all the stocks comprising the index in their exact proportions is not feasible because: (1) a lot of money will be spent in the transaction costs due to large scale buying and selling, (2) since stocks cannot be bought in fractions, buying them to get the exact proportion to the index will require a lot of capital investment. A strategy employed to address these issues involve creating a portfolio with a small subset of the stocks of an index in a derived proportion such that they replicate the market behaviour. The problem of determining this proportion of selected stocks is formulated in a way that it comes under the class of l_2 norm regression subject to portfolio constraints.

Several papers have been published in this direction. We review three such papers with different approach to the problem of sparse index tracking [1]–[3]. All these three papers begin by defining a similar objective function, called as the empirical tracking error (ETE) given as:

$$\begin{aligned} \text{ETE}(\mathbf{w}) = & \frac{1}{T}\left\|\mathbf{X}\mathbf{w}-\mathbf{r}_b\right\|_2^2 \\ \text{s.t.} & \sum_{i=1}^{N}\mathbf{w}=1, \\ & \mathbf{w}\geq \mathbf{0} \end{aligned} \tag{1}$$

where $\mathbf{X}\in\mathbb{R}^{D\times N}$ is the log-return[1] of each of the N assets of the index for D days, $\mathbf{w}\in[0,1]^N$ is the weight fraction associated with each stock in the index, and $\mathbf{r}\in\mathbb{R}^D$ is the log-return[1] of the index for D days. Because of sparsity requirement of $\mathbf{w}$, a regularization is commonly used to penalize non-zero w_i [4]. The most common sparsity inducing norm, l_1 norm is trivially one as seen in one of its constraints. So, the new choice of norm is the critical part to solving this problem. Choice of the norm for regularization is one of the factors of difference amongst the papers [1]–[3]. Section II details the approach and algorithm of the three papers, Section III reports simulation results over various indexes. Section IV suggests future work for this project.

II. Literature Review

A. Sparse Portfolios for High-Dimensional Financial Index Tracking

Benidis et al. in their paper [1] introduce the index tracking problem given by Eq.1, and motivate the need of regularization to ensure sparsity of vector $\mathbf{w}$. The authors use the l_0 'norm' for regularization. In addition to the regular portfolio constraints on $\mathbf{w}$ in Eq.1, they add a holding constraint $0\leq l\leq w_i\leq u$ to avoid extreme positions or very small orders from their solution. Because l_0 norm is neither convex nor concave and is difficult to handle during optimization, they use an approximation function to the l_0 norm given by $\rho_{p,\gamma}(x)=\frac{\log\left(1+\frac{|x|}{p}\right)}{\log\left(1+\frac{\gamma}{p}\right)}$ for a scalar x. The optimization problem in their paper transforms to:

$$\begin{aligned} \min_{\mathbf{w}} \quad & \frac{1}{T}\left\|\mathbf{X}\mathbf{w}-\mathbf{r}_b\right\|_2^2+\lambda\mathbf{1}^\top\boldsymbol{\rho}_{p,u}(\mathbf{w}) \\ \text{s.t.} \quad & \mathbf{w}^\top\mathbf{1}=1, \\ & \mathbf{l}\odot\mathbf{s}\leq\mathbf{w}\leq\mathbf{u}\odot\mathbf{s}, \\ & \mathbf{s}\in\{0,1\}^N \end{aligned} \tag{2}$$

Here, $\boldsymbol{\rho_{p,u}}(\mathbf{w})=[\rho_{p,u}(w_1),\ldots,\rho_{p,u}(w_N)]$, $\mathbf{l}$ and $\mathbf{u}$ are vectors of size N with all values l and u respectively. $s_i=\mathcal{I}_{\{w_i>0\}}$ is an indicator function for positive weights. The authors employ Majorization-Minimization (MM) algorithm to handle the problem. Referring to the nature of the curve

[1] $\mathbf{x}_i=\left[\log\left(\frac{p_1^i}{p_1^{i-1}}\right),\log\left(\frac{p_2^i}{p_2^{i-1}}\right),\ldots,\log\left(\frac{p_N^i}{p_N^{i-1}}\right)\right]$, $i\in\{2,\ldots,D\}$ where p_j^i is the price of j-th stock on i-th day. $\mathbf{x}_1=\mathbf{0}$.

in Fig.1, it is easy to see that a linear function with slope equal to the slope of concave approximation function $\rho_{p,u}(w)$ majorizes it at that point for any $w \geq 0$.

The slope and the constant of the majorizing line is determined to be-

$$d_{\{p,\gamma\}(w^{(k)}} = \frac{1}{\kappa\left(p + w^{(k)}\right)} \tag{3}$$

$$c_{\{p,\gamma\}}(w^{(k)}) = \frac{\log\left(1 + w^{(k)}/p\right)}{\kappa} - \frac{w^{(k)}}{\kappa\left(p + w^{(k)}\right)} \tag{4}$$

$$\mathbf{d}_{\{p,\gamma\}} = \left[d_{\{p,\gamma\}}(w_1^{(k)}), \ldots, d_{\{p,\gamma\}}(w_N^{(k)})\right] \tag{5}$$

where $\kappa = \log(1 + \gamma/p)$. Using this majorizing function in place of $\rho_{p,u}$ in Eq.2 and after ignoring the constant, we reach the first algorithmic solution which the authors refer to as LAIT. For the simpler case of $l = 0$, the optimization problem becomes-

$$\begin{aligned} \min_{\mathbf{w}} \quad & \frac{1}{T}\|\mathbf{X}\mathbf{w} - \mathbf{r}_b\|_2^2 + \lambda \mathbf{d}_{\{p,u\}}^{(k)\top}\mathbf{w} \\ \text{s.t.} \quad & \mathbf{w}^\top \mathbf{1} = 1, \\ & \mathbf{0} \leq \mathbf{w} \leq \mathbf{u} \end{aligned} \tag{6}$$

Algorithm 1: Linear Approximation for Index Tracking problem (LAIT)

Result: Optimized sparse vector $\mathbf{w}$
1 Set k = 0, initialize $\mathbf{w}^{(0)} \in \mathcal{W}$
2 **repeat**
3   Compute $\mathbf{d}_{\{p,u\}}^{(k)}$ using Eq.5
4   Solve Eq.6 using a quadratic program solver and set the solution as $\mathbf{w}^{(k+1)}$
5     $k \leftarrow k + 1$
6 **until** *covergence*;
7 **return** $\mathbf{w}^{(k)}$

Because of computational cost associated with the MM strategy, the authors next attempt to develop a closed form solution to Step 3 of Algorithm 1. They make use of the dual of the the objective function along with its constraints. Before solving the dual problem, they again majorize the objective function of Eq.6 to make the coefficient of the highest order term independent of $\mathbf{X}$.

Defining $\mathbf{L} = \frac{1}{T}\mathbf{X}^\top\mathbf{X}$, $\mathbf{M} = \lambda_{\max}^{(\mathbf{L})}\mathbf{I}$ such that $\mathbf{M} \succeq \mathbf{L}$, and $\mathcal{W}_u = \left\{\mathbf{w} | \mathbf{w}^\top\mathbf{1} = 1, \mathbf{0} \leq \mathbf{w} \leq u\mathbf{1}\right\}$, $\mathbf{w}^\top\mathbf{L}\mathbf{w}$ is majorized at $\mathbf{w}^{(k)}$ by:

$$\begin{aligned} \mathbf{w}^\top\mathbf{L}\mathbf{w} &\leq \mathbf{w}^\top\mathbf{M}\mathbf{w} + 2\mathbf{w}^\top\left(\mathbf{L} - \mathbf{M}\right)\mathbf{w}^{(k)} + const \\ &= \lambda_{\max}^{(\mathbf{L})}\mathbf{w}^\top\mathbf{w} + 2\mathbf{w}^\top(\mathbf{L} - \lambda_{\max}^{(L)}\mathbf{I})\mathbf{w}^{(k)} + const \end{aligned} \tag{7}$$

Dropping the constant terms, Eq.6 at $(k+1)$-th iteration becomes-

$$\begin{aligned} \min_{\mathbf{w}} \quad & \mathbf{w}^\top\mathbf{w} + \mathbf{q}^{(k)\top}\mathbf{w} \\ \text{s.t.} \quad & \mathbf{w} \in \mathcal{W}_u \end{aligned} \tag{8}$$

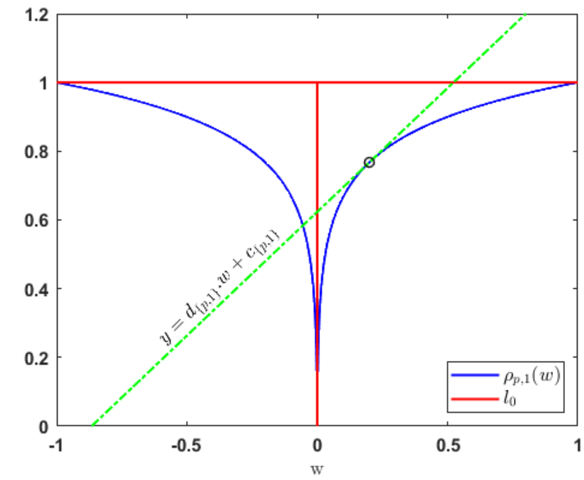

Fig. 1. Approximation function to the l_0 'norm' $\rho_{\{p,u\}}$ for $p = 1e-3$ and $u = 1$. Its majorizing function, $y = d_{\{p,u\}}.w + c_{\{p,u\}}$ plotted for $w = 0.2$.

where

$$\mathbf{q}^{(k)} = \frac{1}{\lambda_{\max}^{(L)}}\left(2\left(\mathbf{L} - \lambda_{\max}^{(L)}\mathbf{I}\right)\mathbf{w}^{(k)} + \lambda\mathbf{d}_{\{p,u\}}^{(k)} - \frac{2}{T}\mathbf{X}^\top\mathbf{r}_b\right) \tag{9}$$

After the problem has reduced to the form of Eq.8, we formulate its dual and get the dual solution as-

(M1): For $u = 1$

$$\begin{aligned} \mathbf{w}^* &= \left(-\frac{1}{2}\left(\mu\mathbf{1} + \mathbf{q}\right)\right)^+ \\ \mu &= -\frac{\sum_{i\in\mathcal{A}} q_i + 2}{\text{card}\left(\mathcal{A}\right)} \\ \mathcal{A} &= \{i | \mu + q_i < 0\} \end{aligned} \tag{10}$$

(M2): For $u < 1$

$$\begin{aligned} \mathbf{w}^* &= \left(\min\left(\left(-\frac{1}{2}\left(\mu\mathbf{1} + \mathbf{q}\right)\right), u\mathbf{1}\right)\right)^+ \\ \mu &= -\frac{\sum_{i\in\mathcal{B}_2} q_i + 2 - \text{card}\left(\mathcal{B}_1\right)2u}{\text{card}\left(\mathcal{B}_2\right)} \\ \mathcal{B}_1 &= \{i | \mu + q_i \leq -2u\} \\ \mathcal{B}_2 &= \{i | -2u < \mu + q_i < 0\} \end{aligned} \tag{11}$$

For detailed derivation, the reader is referred to the Appendex of [1]. After these definitions, we formally define this closed form algorithm in Algorithm 2. The authors refer to this algorithm as SLAIT.

B. Sparse Index Tracking based on $L_{1/2}$ Model and Algorithm

Xu et al. in [5] proposed $l_{1/2}$ regularization to ensure weights sparsity while minimizing the ETE. The $l_{1/2}$ norm of $\mathbf{x}$ is defined as-

$$\|\mathbf{x}\|_{1/2} = \left(\sum_{i=1}^{n}|x_i|^{1/2}\right)^2 \tag{12}$$

Algorithm 2: Specialized LAIT with $\mathcal{W} = \mathcal{W}_u$ (SLAIT)

Result: Optimized sparse vector $\mathbf{w}$

Set k = 0, initialize $\mathbf{w}^{(0)} \in \mathcal{W}_u$

Compute $\lambda_{\max}^{(\mathbf{L})}$

repeat

  Compute $\mathbf{q}^{(k)}$ using Eq.9

  Use M1/M2 to solve Eq.8

  Set the solution to $\mathbf{w}^{(k+1)}$

  $k \leftarrow k+1$

until *covergence*;

return $\mathbf{w}^{(k)}$

The $l_{1/2}$ regularization is a non-convex and non-smooth problem, so the authors derived a half thresholding algorithm [5] for obtaining the solution. The regularizing parameter used is automatically corrected by the algorithm to an appropriate value irrespective of the initial condition.

Let $\mathcal{S}(\mathbf{w})$ be the support set of w, i.e. $\mathcal{S}(\mathbf{w}) = \{i|w_i > 0\}$, and l and u are the lower and upper bounds of w_i respectively. For λ as the regularization parameter, the optimization equation and the constraints are defined as

$$\begin{aligned} \min \|\mathbf{X}\mathbf{w} - \mathbf{r}_b\|_2^2 &+ \lambda \|\mathbf{w}\|_{1/2}^{1/2} \\ \text{s.t.} \qquad \mathbf{1}^\top \mathbf{w} &= 1 \\ l \le w_i \le u, &\, i \in \mathcal{S}(\mathbf{w}) \\ w_i = 0, &\, i \notin \mathcal{S}(\mathbf{w}) \end{aligned} \tag{13}$$

The authors proposed an iterative algorithm for solving the above optimization equation. If we first consider the unconstrained equation, with $\mathbf{w}^{(k)}$ as the weights of the k-th iteration, its next iteration can be defined as-

$$\mathbf{w}^{(k+1)} = \mathbf{H}_{\lambda_k \mu_k, 1/2}(\mathbf{w}^{(k)} + \mu_k \mathbf{X}^\top (\mathbf{r}_b - \mathbf{X}\mathbf{w}^{(k)})) \tag{14}$$

The authors call this algorithm as the hybrid half thresholding algorithm. Here, $\mathbf{H}$ is the half-thresholding operator given by

$$\begin{gathered} \mathbf{H}_{\lambda\mu,1/2}(\mathbf{x}) = (h_{\lambda\mu,1/2}(x_1), h_{\lambda\mu,1/2}(x_2), ..., h_{\lambda\mu,1/2}(x_n)) \\ h_{\lambda\mu,1/2}(x_i) = \begin{cases} \frac{2}{3}|x_i|\left(1 + \cos\left(\frac{2\pi}{3} - \frac{2\varphi_\lambda(x_i)}{3}\right)\right), & \\ \qquad\qquad |x_i| > \frac{\sqrt[3]{54}}{4}(\lambda\mu)^{\frac{2}{3}} & \\ 0, \qquad \text{otherwise} & \end{cases} \\ \cos\varphi_\lambda(x_i) = \frac{\lambda}{8}\left(\frac{|x_i|}{3}\right)^{-3/2} \end{gathered} \tag{15}$$

Now, for the sparsity problem to determine exactly K stocks to invest in, parameters for the n-th iteration are adopted as-

$$\begin{gathered} \mu_k = \mu_0, \\ \lambda_k = \min\left\{\lambda_{k-1}, \frac{\sqrt{96}}{9}\|\mathbf{X}\|^2 \left|[\mathbf{B}_{\mu_k}(\mathbf{w}_k)]_{K+1}\right|^{3/2}\right\}, \\ \text{where} \qquad \mu_0 = \frac{1-\epsilon}{\|\mathbf{X}\|^2}, \quad \epsilon \ll 1 \\ \mathbf{B}_{\mu_k}(\mathbf{w}^{(k)}) = \mathbf{w}^{(k)} + \mu_k \mathbf{X}^T(\mathbf{r}_b - \mathbf{X}\mathbf{w}^{(k)}) \end{gathered} \tag{16}$$

Here, $[\mathbf{B}_\mu(\mathbf{x}_k)]_{K+1}$ is the (K + 1)-th largest component of $\mathbf{B}_\mu(\mathbf{x}_k)$ by magnitude. As a result, the final outcome will not depend on the initial choice of regularization parameter and will adapt itself. After obtaining the weights for the unconstrained case using Eq. 15, we express $\mathcal{S}(\mathbf{w})$ as

$$\mathcal{S}(\mathbf{w}) = \{i|w_i > 0\} \tag{17}$$

Then the non zero optimal asset weights can be solved by solving the following problem with a quadratic solver-

$$\begin{aligned} \min \quad & \|\mathbf{X}\mathbf{w} - \mathbf{r}_b\|_2^2 \\ \text{s.t.} \quad & \mathbf{1}^\top \mathbf{w} = 1 \\ & l \le w_i \le u, \qquad i \in \mathcal{S}(\mathbf{w}) \end{aligned} \tag{18}$$

. Defining $\mathcal{W}_u = \{\mathbf{w}|\mathbf{w}^\top \mathbf{1} = 1, \mathbf{0} \le \mathbf{w} \le u\mathbf{1}\}$, Algorithm 3 states the strategy proposed in [2]. We call it as L12.

Algorithm 3: Half thresholding index tracking (L12)

Result: Optimized sparse vector $\mathbf{w}$

Set k = 0, initialize $\mathbf{w}^{(0)} \in \mathcal{W}_u$

repeat

  Compute λ_k and μ_k using Eq. 16

  Use λ_k and μ_k in Eq. 15 to get the unconstrained case weights

  Calculate $\mathcal{S}(\mathbf{w})$ using 17

  Solve 18 using $\mathcal{S}(\mathbf{w})$ with a quadratic solver

  $k \leftarrow k+1$

until *covergence*;

return $\mathbf{w}^{(k)}$

C. Diversity and Sparsity: A new perspective on Index Tracking

Zheng et al. [3] approaches the problem by giving equal importance to both the sparsity of the weight vector, and the diversity of the selected stocks. More diversity in the portfolio leads to lesser risk because of different sectors responding differently to an event. They regularize the ETE defined in Eq.1 with two different functions, a weighted l_2 norms and a reweighted l_1 norm. They begin by defining an similarity matrix $\mathbf{A}$ where $A_{ij} = 1$ if the i-th and j-th stock belong to the same sector (eg, banking, technology, automobile, etc.), and 0 otherwise. As an example, we can expect $A_{ij} = 1$ if i-th stock is National Instruments and j-th stock is Texas Instruments. A_{ii} is always 1. An alternate representation of $\mathbf{A}$ could be

$$\mathbf{A} = \mathbf{Z}^\top \mathbf{Z} \tag{19}$$

where $\mathbf{Z} \in \{0,1\}^{K \times N}$ for K unique industry sectors and N number of stocks. The i-th column of Z is a one-hot vector of industry labels for the i-th stock. Therefore, $\mathbf{1}^\top \mathbf{Z} = \mathbf{1}$. To induce diversity in the optimized weight vector $\mathbf{w}$, a weighted l_2 norm known as Tikhonov regularization [6] and given by $\mathbf{w}^\top \mathbf{A}\mathbf{w}$ is added. Using Eq. 19, the l_2 norm can be written as-

$$l_2(\mathbf{w}) = \mathbf{w}^\top \mathbf{A}\mathbf{w} = \mathbf{w}^\top \mathbf{Z}^\top \mathbf{Z}\mathbf{w} = \|\mathbf{Z}\mathbf{x}\|_2^2 \tag{20}$$

For sparsity, a new l_1 norm is defined because of the constant value issue of the l_1 norm on $\mathbf{w}$. The authors begin by creating an affinity matrix $\mathbf{S}$ of the log-returns of each of the N stocks. $S_{ij} = \exp\left(-\frac{d^2(x_i,x_j)}{\sigma^2}\right)$ for $i \neq j$, and 0 otherwise. Column vector $x_k \in \mathbb{R}^D$ contains log-return of k-th stock for D days. Distance $d(x_i, x_j)$ can be the Euclidean distance $d(x_i, x_j) = \|x_i - x_j\|_2$, or the Spearman's rank correlation coefficient [7], $d(x_i, x_j) = \sqrt{2\left(1 - \rho(x_i, x_j)\right)}$ where $\rho(x_i, x_j)$ is the rank correlation coefficient. The authors suggest σ to be the median of all non diagonal terms of $\mathbf{S}$.

Following this, a symmetric normalized Laplacian matrix defined as $\mathbf{L} = \Lambda^{-\frac{1}{2}}\mathbf{S}\Lambda^{\frac{1}{2}}$ is calculated. Here $\Lambda = \operatorname{diag}\left(\sum_j S_{ij}\right)$. Next, the authors extract K largest eigenvectors of $\mathbf{L}$ denoted as $v_1, \ldots, v_K$, and stack them in a matrix $\mathbf{H} = [v_1^\top; \ldots; v_K^\top]$. The columns of matrix $\mathbf{H}$ are l_2 normalized using $\mathbf{H}_{ij} \leftarrow \frac{\mathbf{H}_{ij}}{\left(\sum_i \mathbf{H}_{ij}^2\right)^{\frac{1}{2}}}$. Then k-means clustering is performed to get K-clusters denoting the new industry sectors in this latent space $\mathbf{H}$. This complete step is known as Spectral Clustering [8]. We then use these cluster assignments for each stock to create the new one-hot vector matrix $\mathbf{Z}$, and thus get $\mathbf{A}$ using Eq. 19.

The reweighted l_1 norm for sparsity is defined as-

$$l_1(\mathbf{w}) = \sum_{i=k}^{K} \frac{1}{|\mathcal{C}_i|} \sum_{j \in \mathcal{C}_i} |w_j| \tag{21}$$

where $\mathcal{C}_i$ denotes the set of indices in the i-th cluster. $|\mathcal{C}_i|$ is the i-th cluster's size. In matrix notation, the expression can be shown to be-

$$l_1(\mathbf{w}) = \mathbf{1}^\top(\mathbf{Z}\mathbf{Z}^\top)^{-1}\mathbf{Z}\mathbf{w} \tag{22}$$

Thus the optimization problem becomes-

$$\begin{aligned} \min_{\mathbf{w}} \quad & \|\mathbf{X}\mathbf{w} - \mathbf{r}_b\|_2^2 + \lambda_1 \|\mathbf{Z}\mathbf{w}\|_2^2 + \lambda_2 \mathbf{1}^\top \left(\mathbf{Z}\mathbf{Z}^\top\right)^{-1} \mathbf{Z}\mathbf{w} \\ & \text{s.t.} \quad \mathbf{1}^\top\mathbf{w} = 1, \mathbf{w} \geq \mathbf{0} \end{aligned} \tag{23}$$

Expressing it in quadratic form-

$$\begin{aligned} \min_{\mathbf{w}} \quad & \mathbf{w}^\top(\mathbf{X}^\top\mathbf{X} + \lambda_1\mathbf{Z}^\top\mathbf{Z})\mathbf{w} \quad + \\ & (\lambda_2\mathbf{1}^\top(\mathbf{Z}\mathbf{Z}^\top)^{-1}\mathbf{Z} - 2\mathbf{X}^\top\mathbf{r}_b)^\top\mathbf{w} \\ \text{s.t.} \quad & \mathbf{1}^\top\mathbf{w} = 1, -\mathbf{w} \leq \mathbf{0} \end{aligned} \tag{24}$$

The final algorithm is composed as Algorithm 4. We will refer to it as SADA. The constraints on $\mathbf{w}$ in Eq. 24 are written as $\mathcal{W}$ in the algorithm.

Algorithm 4: Sparse and Diverse Algorithm (SADA)

Result: Optimized sparse vector $\mathbf{w}$
Set k = 0, initialize $\mathbf{w}^{(0)} \in \mathcal{W}$
Use $\mathbf{X}$ for spectral clustering to get $\mathbf{Z}$
repeat
    Solve Eq. 24 using a quadratic program solver
    Set the solution as $\mathbf{w}^{(k+1)}$
    $k \leftarrow k + 1$
until *covergence*;
return $\mathbf{w}^{(k)}$

III. Experiments and Results

A. Dataset

We test these algorithms on the National Stock Exchange (NSE) NIFTY50 index. It consists the top 50 NSE listed stocks based on market capital. To test scalability of the algorithms, we also use US S&P500. Data is extracted for a period of 5 years ranging from 23rd March 2016 to 22nd March 2021. To calculate the weight vector $\mathbf{w}$ at the beginning of our investment period, we use the past 1 year data from the date of investment. Hence, for a 5 year dataset, maximum investment simulation window can be 4 years. Market Indexes are revised regularly by the index provider and, therefore, few stocks were added/removed during this 5 year time frame of our dataset. However, historic index constituents data aren't available for free. So we have assumed the stocks part of the index on 22nd March to be the constituents for all the 5 years. Stocks that were first listed on the market in between our time frame and are part of the index currently were priced at 0 before the date they were listed. These are inaccurate assumption which may lead to anomaly right at the beginning. However, such assumptions are also a good test of robustness for these algorithms.

Two methods of investment are considered- Static and Dynamic. For Static investment, we calculate the weights at the beginning of our investment period and do not revise them throughout the timeline. For Dynamic investment, the weights are revised at fixed intervals based on past 1 year data from the revision date. For example, if we invest on Jan 1, 2020 based on weights calculated from Jan 1, 2019 to Dec 31, 2019, and re-adjust on April 1, 2020, weights are calculated based on stocks/index performance from April 1, 2019 to March 31, 2020.

B. Simulation

The following simulations were performed using the algorithms described in Section II-

1) Fig. 2 shows simulation result of Static investment returns based on NIFTY 50 for four years using Algorithms 1, 2, 3, and 4. Assuming an initial investment of Re.1 for normalized visualization, we plot the returns of the index and our portfolio based on the return of the index and its constituents.
2) Fig. 3 shows Dynamic investment returns based on NIFTY 50 for same parameters as the Static investment, but with a readjustment period of 63 days (3 trading months). The total capital remains constant at the boundary of each period. That is, the total capital at the end of period i is the total capital invested at the beginning of period $i + 1$.
3) We study the effect of different readjustment periods on our result in Fig. 4. While larger gap in readjustment

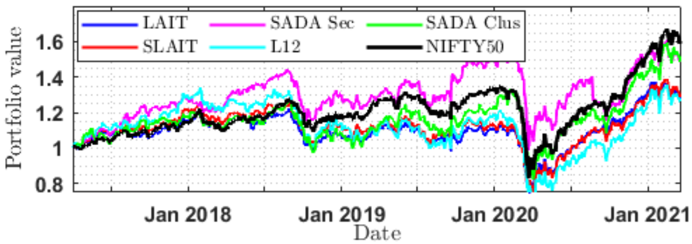

Fig. 2. Static investment returns using strategies from Section II on NIFTY 50 for a period of 4 years. SADA Sec is the implementation of Algorithm 4 using sector labels provided from the index provider. SADA Clus uses spectral clustering to find improved sector label as discussed in subsection II-C. Number of labels for SADA Clus (K) is set equal to number of labels provided by the index provider.

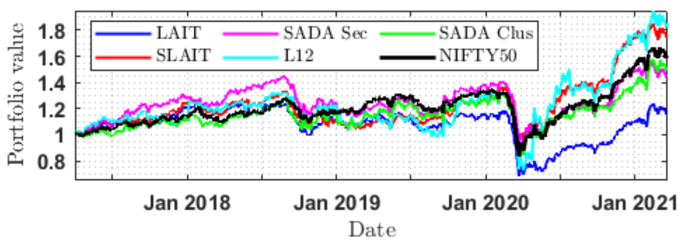

Fig. 3. Dynamic investment returns using strategies from Section II on NIFTY 50 for a period of 4 years. Readjustment period is 63 days (3 trading months).

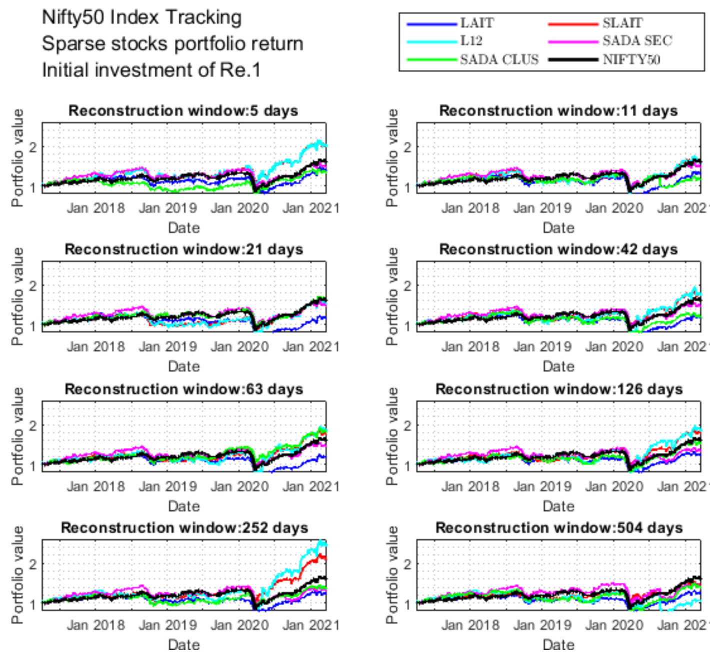

Fig. 4. Effect of changing readjustment period for Dynamic investment on NIFTY 50 over a period of 4 years. Periods compared- 1 trading week, 2 trading weeks, 1 trading month, 2 trading months, 3 trading months, 6 trading months, 1 trading year, 2 trading years.

leads to poor performance for $L_{1/2}$ and SLAIT, lesser readjustment period does not guarantee a better result.

4) The run-time for the five algorithms are tested on normally distributed randomly generated matrices $\mathbf{X}$ and $\mathbf{r}_b$. Change in run-time with increase in investment period is shown in Fig. 5. $L_{1/2}$ is the slowest algorithm with a single iteration 3 orders slower to SLAIT for large periods of investment.
5) We investigate the performance of the algorithms on larger datasets by simulating it for S&P 500 dataset for the same time period. Fig. 6 shows the result. Major deflection is observed for result of all the algorithms except the cluster based Algorithm 4. A major reason of this anomaly could be that S&P 500 is revised quarterly. So for a dataset 5 years longs, the stock constituents were changed 15 times. Two stocks were listed for the first time during this interval. Since we assumed the stock composition at on March 22, 2021 to constant for the past years, the recommendations made by the algorithms failed to track the index. However, the result of 'SADA Clus' algorithm established its robustness to anomalous data as well.

Table I lists the mean squared error between the index and portfolio's return. 'SADA Clus' algorithm produces the best result for both the datasets.

IV. Future Work

The simulations can be improved with accurate historic index and constituents closing prices.

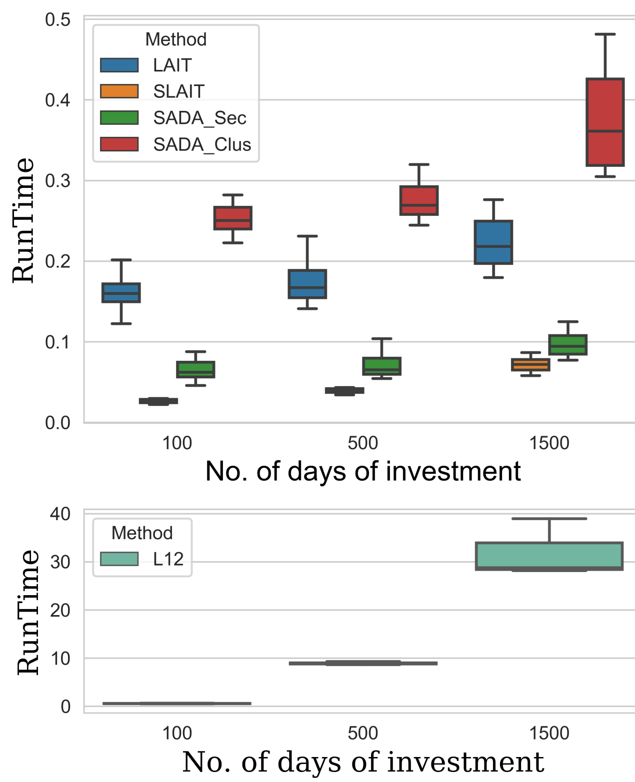

Fig. 5. Comparison of run time for all the algorithms listed in Section II with increasing investment period. Run time is in seconds.

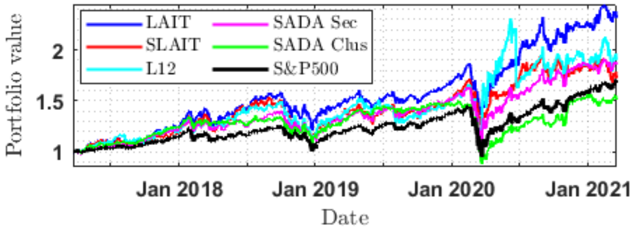

Fig. 6. Dynamic investment return using strategies from Section II on S&P 500 for a period of 4 years. Readjustment period is 63 days (3 trading months).

TABLE I
MEAN SQUARED ERROR BETWEEN INDEX AND PORTFOLIO RETURNS

	LAIT	SLAIT	$L_{1/2}$	SADA Sec	SADA Clus
NSE50 (WR)[1]	0.0045	0.0042	0.0051	0.0038	0.0023
NSE50 (R63)[2]	0.0063	0.0033	0.0037	0.0029	**0.0020**
S&P500 (R63)[2]	0.0119	0.0065	0.0084	0.0048	**0.0035**

[1] WR: Withour re-adjustment [2] R63: Re-adjusment period of 63 days

We expect faster convergence for LAIT and SLAIT in Algorithms 1, 2 with an improved approximation of l_0 norm- $1-\exp\left(-\frac{x^2}{\gamma p}\right)$. See Fig. 7. However, this exponential function converges quickly to 1, after which its slope becomes zero. A Huber penalty inspired expression that becomes linear after a point instead of converging to one may work better that the $\rho_{p,\gamma}(x)$ approximation for l_0 'norm'.

The SADA algorithm is very sensitive to initialization, and requires few iterations to give the best results. This was found to be due to overlapping clusters which were difficult which changed assigments for different initialization in k-means. An iterative Deep Embedding Clustering [9] can be employed which projects data in a latent space where are clusters are easily seperable. In this method, the cluster centres as well as the weights of the data projection networks are updated simultaneously to tailor the latent space according to the data to be clustered. This, along with the eigen gap method commonly used in spectral clustering [8], can be used to find better sector assignment of the stocks.

ACKNOWLEDGMENT

We sincerely thank Dr. Ketan Rajawat for teaching us the technical background required to undertake this project and for giving us the opportunity to work on this project as part of the course EE609A.

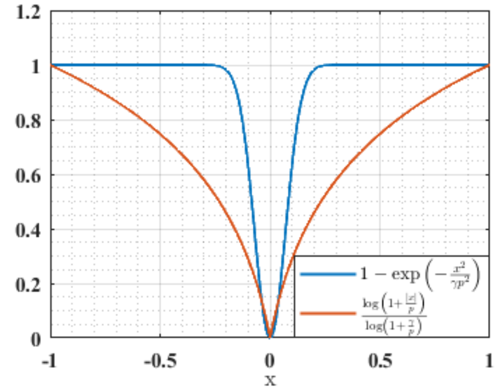

Fig. 7. Comparison between current approximation and improved approximation for LAIT and SLAIT. $p = 0.1$, $\gamma = 1$.

REFERENCES

[1] K. Benidis, Y. Feng, and D. P. Palomar, "Sparse portfolios for high-dimensional financial index tracking," *IEEE Transactions on Signal Processing*, vol. 66, no. 1, pp. 155–170, 2018.

[2] F. Xu, Z. Xu, and H. Xue, "Sparse index tracking based on $l_{1/2}$ model and algorithm," 2015.

[3] Y. Zheng, T. M. Hospedales, and Y. Yang, "Diversity and sparsity: A new perspective on index tracking," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 1768–1772.

[4] F. Bach, R. Jenatton, J. Mairal, and G. Obozinski, "Optimization with sparsity-inducing penalties," *Found. Trends Mach. Learn.*, vol. 4, no. 1, p. 1–106, Jan. 2012. [Online]. Available: https://doi.org/10.1561/2200000015

[5] Z. Xu, X. Chang, F. Xu, and H. Zhang, "$l_{1/2}$ regularization: A thresholding representation theory and a fast solver," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 7, pp. 1013–1027, 2012.

[6] A. N. Tikhonov and V. I. Arsenin, *Solutions of ill-posed problems*. New York NY: Winston, 1977.

[7] C. Spearman, "The proof and measurement of association between two things," *The American Journal of Psychology*, vol. 100, no. 3/4, pp. 441–471, 1987. [Online]. Available: http://www.jstor.org/stable/1422689

[8] U. von Luxburg, "A tutorial on spectral clustering," 2007, arXiv: 0771.0189.

[9] J. Xie, R. Girshick, and A. Farhadi, "Unsupervised deep embedding for clustering analysis," in *International conference on machine learning*. PMLR, 2016, pp. 478–487.