16 November 2015
کنفرانس بین المللی
پژوهش های نوین در مدیریت
و مهندسی صنایع
International Conference on Modern Research in Management and Industrial Engineering
International Conference on Modern Research in Management
ISC
ISI

Index tracking by means of fuzzy clustering using time series features

Seyed Milad Rezvanizaniani
Master of Science in Financial engineering, Amirkabir University (Tehran Polytechnique), Iran
m.rezvani@aut.ac.ir

Bahareh Dadgar
Master of Science in Financial engineering, Amirkabir University (Tehran Polytechnique), Iran
baharedadgar@aut.ac.ir

Pejman Mehran
Assistant Professor, Amirkabir University (Tehran Polytechnique), Iran
p.mehran@aut.ac.ir

Abstract

Index tracking is an investment strategy aimed to replicate the performance of a particular financial market which is usually shown by a specific index. The strategy become more popular in recent years, as a robust and sustainable one with a lower management and transaction cost in comparison to speculations and active investments. This paper extracted some basic features of the financial time series in order to cluster stocks and make an index fund (tracking portfolio) to track NASDAQ100 index. This is the first time of using fuzzy clustering methods for index tracking, which trigger us to fill this gap of literature. In this paper we proposed a new way of weighting to selected stocks for index fund based on each stock membership degree in each cluster. At last, the new fuzzy model was compared with an integer OR model, our model highly reduced the running time of the model compared to OR model, although its error is more than the OR model. The advantage of reducing time cost shows itself in a larger markets which consists of thousands of stocks.

Keywords: index tracking, fuzzy c-means, entropy

Introduction

In recent years investing on stock markets have become very popular, especially amongst mutual fund managers and pension funds. In general, there are two major strategy for investing in stock market: Active strategy and passive one.

In active strategy managers expected to outperform the market return based on their experience and knowledge, they usually take more risk to achieve higher return than the market by frequent trading and speculation. They hope to beat the market and gain more benefit, although the management team is expensive due to high fixed payment to experts and high transaction cost of frequent trading. In this situation investor face both systematic risk (market risk) and unsystematic risk (Company- or industry-specific hazard that is inherent in each investment).

In passive strategy managers have less degree of flexibility in comparison to active managers. Their goal is to gain specific return which is usually market index. In this strategy investors avoid further risks, while it is cheaper than active investments because there is no need to number of experts analysts for management team and passive managers usually invest on stocks for long term and avoid frequent trading which is highly reduced the transaction cost (Beasley, 2003).

Passive investments become more popular in recent years, due to disadvantages of active investments. Most of active funds -a fund that invest actively- failed to beat market return, although best active funds achieve mush higher return than the market. Based on statistical data 80% of active funds could not reach their goal even (He Ni, 2013). On the other hand if an active fund can satisfy its objective and beat the market return in a year, still there is no guarantee to outperform the market for next year too (Bogle, 1992). But those who support passive investments believe that stock markets value gradually grow in long term and by avoiding extra risks they prefer to take a small profit of the market (Beasley, 2003).

Based on the Sharp’s Capital Asset Pricing model in the market risk level no other portfolios can beat the market portfolio. He believes that if such a portfolio exist the law of supply and demand put its security prices back in the line (Sharpe, 1964). That is why most of the passive fund managers choose market index as their goal and the index tracking problem become a more popular issue for researchers.

The easiest way to follow the market index is full replication which means invest on all stocks of an index relating to their market capital. But in reality it is not useful because by following this strategy the portfolio become so large which so many stocks have very small proportions. The index stocks may change over time and the index fund portfolio need to be changed repeatedly, and it force a lot of transaction cost to the index fund (Beasley, 2003).

In order to avoid mentioned problems we proposed a model that assign all the market stocks in a limited clusters, and that make investors not to invest on stocks with the same characteristics. The model select just one stock from each cluster and invest on them instead of investing on all stocks. In fact, each selected stock is representing other index stocks.

Literature review

Looking at previous works of index tracking, we have noticed that there exist several models for reproducing the performance of an index using only a subset of stocks. Some models ware based on Markowitz model their aim were minimize variance of the difference between index return and tracking portfolio return. Hodges was the first one who used the Markowitz model and compare index tradeoff curve with tracking portfolio trade off curve (Hodges., 1976).

Some other researchers used factor based models, in these models each stock is related to one or more economic factors and the model shows the relationship between them. (Beasley and Meade, 2004) and (Resnick and Larsen, 1998) used a single factor model in order to minimize tracking error given

a subset of shares. (Salkin and Meade, 1990) using quadratic programing to solve the problem They also considered the effect of industry stratification within a tracking portfolio. (Andrew Rudd, 1980) present a single factor model to track S&P500 with heuristic. After him some other researchers like (Francesco Coriellia, 2006) expand the model to multiple factor model.

There is some independent models, (Roll, 1992) used a quadratic function to minimize tracking error. (Alex Frino and D R Ghallagher, 2001) proposed a model and consider transaction cost and stock dividend too. (Beasley et al, 2003) use an evolutionary algorithm for index tracking problem.

(Nesrn Okaya & Uğur Akmanb, 2003) use constraint aggregation (CA) technique for the first time in index tracking problem and compare it with the normal solution however both solution leads to a same result but using CA reduce the time cost much more. (N.A. Canakgoz, 2009) present a more advanced model based on regression, he used a mixed integer linear programming for index tracking and enhanced index tracking which means obtaining more profit than market return. (Diana Barro and Elio Canestreli, 2009) present a multistage tracking error model and solve it in stochastic programming framework. They also consider about transaction cost and liquidity components in their model.

(Christian Dose and Silvano Cincotti, 2005) describe a stochastic optimization technique based on time series clustering for index tracking problem. First they select a subset of stocks from the market to construct the tracking portfolio and in the second stage they use stochastic optimization to weight the selected stocks. Finally they show that using clustering enhanced the results.

Portfolio selection model

As presented by (Cornuejols G and Tutuncu, 2007) the model selects stocks for a tracking portfolio. After solving the model, the selections are weighted based on their market value. There are numerous measures of similarity between assets. We use their work as a benchmark for validation of our model (Chen Chen, 2012).

$$\rho_{ij} = \text{similarity measure between stock i and stock j} \tag{1}$$

Suppose we construct a portfolio of q assets from a target index of n assets. Let ρ_{ij} represent the similarity between asset I and asset j.

Let y_j represent if asset j is selected to be in the portfolio (1 if true, 0 otherwise). Let x_{ij} represent whether asset j is a representative of stock I, x_{ij} is 1 if j is the most similar asset in the portfolio to i, 0 otherwise.

$$Z = \max \qquad \sum_{i=1}^{n}\sum_{j=1}^{n}\rho_{ij}x_{ij}$$

Subject to : $\sum_{j=1}^{n} y_j = q$ (portfolio size constraint)

$$\sum_{j=1}^{n} x_{ij} = 1 \qquad \text{for } 1 = 1, \dots, n$$

(each stock has exactly one representative in the portfolio)

$$x_{ij} \le y_j \qquad \text{for } 1 = 1, \dots, n; \; j = 1, \dots, n$$

(stock must be in the portfolio to be a representative)

$$x_{ij}, y_j = 0 \text{ or } 1 \qquad \text{for } 1 = 1, \dots, n; \; j = 1, \dots, n$$

Having solved this model, a weight ω_j is calculated for each selected asset j using the sum of the market value, V_i, of each stock from the index it is representing

$$\mathrm{\omega_j = \sum_{i=1}^{n} V_i\, x_{ij}} \qquad (2)$$

$$\mathrm{\frac{\omega_j}{\sum_{f=1}^{n} \omega_f}} \qquad (3)$$

Fuzzy C-means

Fuzzy c-means (FCM) is a method of clustering which allows one piece of data to belong to two or more clusters. This method -developed by (Dunn, 1973) and improved by (Bezdek, 1981)- is frequently used in pattern recognition. It is based on minimization of the objective function showed in equation (4).

$$J_m = \sum_{i=1}^{N} \sum_{j=1}^{C} u_{ij}^{m} \, ||x_i - c_j||^2 \qquad 1 \le m < \infty \qquad (4)$$

Where m is any real number greater than 1, u_{ij} is the degree of membership of x_i in the cluster j, x_i is the ith of d-dimensional measured data, c_j is the d-dimension center of the cluster, and ||*|| is any norm expressing the similarity between any measured data and the center.

Fuzzy partitioning is carried out through an iterative optimization of the objective function shown above, with the update of membership u_{ij} and the cluster centers c_j, equation (5) shows the formula of updating u_{ij}, and in equation (6) shows the formula of c_j.

$$u_{ij} = \frac{1}{\sum_{k=1}^{c} \left(\frac{||x_i - c_j||}{||x_i - c_k||} \right)^{\frac{2}{m-1}}} \qquad (5)$$

$$c_j = \frac{\sum_{i=1}^{N} u_{ij}^{m} . x_i}{\sum_{i=1}^{N} u_{ij}^{m}} \qquad (6)$$

This iteration will stop when, the model reach the maximum number of iteration or minimum amount of improvement, whereas k are the iteration steps. This procedure converges to a local minimum or a saddle point of J_m. The algorithm is described in appendix 1 too.

In our model we used Euclidian distance function, and defined maximum number of iteration 1000, and minimum amount of improvement 10^{-9}. Another crucial parameter of this model is the amount of m, which we use hill climbing to find the best value of it for our model, we start from m=1 and increase it 0.1 each time to find the best value of it and finally find out that the best value of m is 2.2 for our model. The procedure of running model is shown as a flow chart in figure 1.

After running the FCM model we choose stocks with highest membership degree from each cluster as representatives of other stocks and invest on them. The suitable weight of investing on each stock is calculate as follow:

$$w_j = \frac{\sum_{i=1}^{N} u_{ij} * MC_i}{\sum_{i=1}^{N} MC_i} \qquad (7)$$

In which, w_j is the weight that must be invest on stock j that is the selected stock from cluster j because it had the highest membership degree. And MC_i is the market capital of stock i.

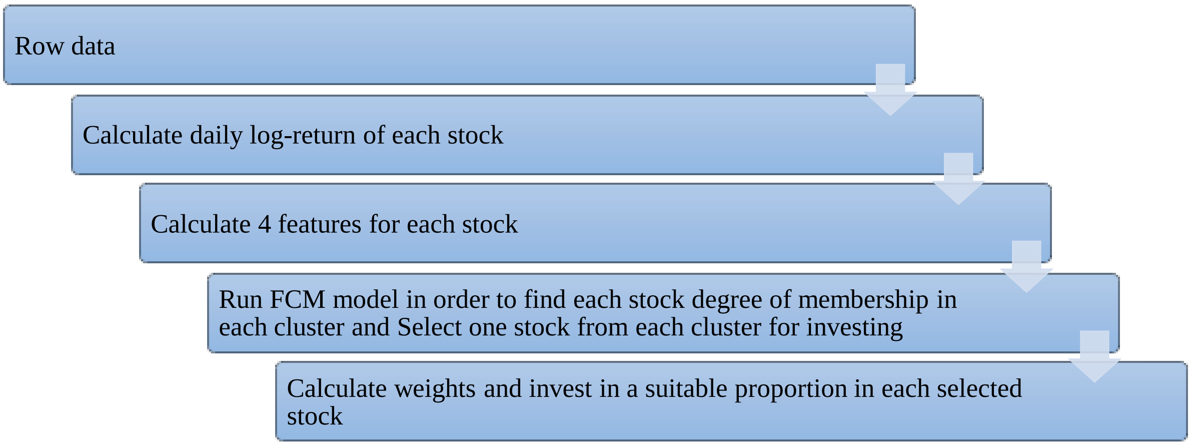
Row data
Calculate daily log-return of each stock
Calculate 4 features for each stock
Run FCM model in order to find each stock degree of membership in each cluster and Select one stock from each cluster for investing
Calculate weights and invest in a suitable proportion in each selected stock

Figure 1- Flow chart of running model procedure

Data

In order to implement the model we used NASDAQ market data set to track NASDAQ100 index which consist of 100 different stocks. We used log-return of these stock for 1008 days. 708 records are used to run the model and the other 300 records are used for testing the model. We omit 6 stocks out of 100 stocks due to lack of records (less than 1008days).

We extract 4 features from all stocks time series in order to implement fuzzy c-means algorithms that is shown in table 1.

We defined 4 features for each stocks which is clustered by them. First of all daily log-return of each stock is calculated and some features extracted from that. Feature 1 is the standard deviation of daily returns for each stock. Feature 2 is the average of daily return of each stock. Feature 3 is the correlation between each stock daily returns and the index daily returns. And feature 4 is Entropy which is going to be discuss in next section. Using entropy as a feature for index tracking is another contribution of this paper.

Table 1- stock features

feature 1	**feature 2**	**feature 3**	**feature 4**
Standard deviation	Mean	Correlation	Entropy

Entropy

Quantifying the amount of regularity for a time series is an essential task in understanding the behavior of a system. One of the most popular regularity measurements for a time series is the sample entropy (SampEn) (Richman & Moorman, 2000) which is an unbiased estimator of the conditional probability that two similar sequences of m consecutive data points (m is the embedded dimension) will remain similar when one more consecutive point is included (Costa, et al., 2003). The SampEn characterizes complexity strictly on a time scale defined by the sampling procedure which is used to obtain the time series under evaluation. However, the long-term structures in the time series cannot be captured by SampEn. In regard to this disadvantage, (Costa, et al., 2002) proposed the multiscale entropy (MSE) algorithm , which uses sample entropies (SampEns) of a time series at multiple scales to tackle this problem (Shuen-De Wu, et al., 2013).

Essentially, the MSE is used to compute the corresponding SampEn over a sequence of scale factors. For an one-dimensional time series, $x = \{x_1, x_2, x_3, \dots, x_n\}$, the coarse-grained time series, $y^{(\tau)}$ can be constructed at a scale factor of τ, according to equation (8).

$$y_j^{(\tau)} = \frac{1}{\tau}\sum_{i=(j-1)\tau+1}^{j\tau} x_i \qquad 1 \le j \le \frac{N}{\tau} \qquad (8)$$

As shown in Figure 2, the coarse-grained time series is divided into non-overlapping windows of length τ, and the data points inside each window are averaged. We then define the entropy measurement of each coarse-grained time series as the MSE value. In this paper, the SampEn is used as the entropy measurement. The algorithm proposed in (Pan, et al., 2011) is repeated here, and we refer to the algorithm as follow.

```
For i = 1:N
{
For j = i+1:N
{
if( |$x_i$ - $x_j$ | < r & |$x_{i+1}$ - $x_{j+1}$ | < r )
{
$n_n$ = $n_n$ +1
if( |$x_{i+2}$ - $x_{j+2}$| < r )
$n_d$ = $n_d$+1
}
}
}
SampEn = -log ($n_d$ / $n_n$ )
```

In the whole study of this paper, the sample entropy of each coarse grained time series is calculated with m = 2 and r = 0.15σ [2], where σ denotes the standard deviation (SD) of the original time series (Shuen-De Wu, et al., 2013).

Entropy measurements are highly dependent on the length of time-series. As the length of each coarse-grained time series is equal to that of the original time series divided by the scale factor, τ, the variance of entropy measurements grows as the length of coarse-grained time series is reduced. The estimation error of a conventional MSE algorithm would be very large at large scale factors. In the following section, composite multiscale entropy (CMSE), is proposed to overcome this problem (Shuen-De Wu, et al., 2013).

Composite Multiscale Entropy

As shown in Figure 3, there are two and three coarse-grained time series divided from the original time series for scale factors of 2 and 3 respectively. The kth coarse-grained time series for a scale factor of τ, $y_k^{(\tau)} = \left\{y_{k,1}^{(\tau)}, y_{k,2}^{(\tau)}, \dots, y_{k,p}^{(\tau)}\right\}$ is defined as in equation (9).

$$y_{k,j}^{(\tau)} = \frac{1}{\tau} \sum_{i=(j-1)\tau+1}^{j\tau+k-1} x_i \qquad 1 \le j \le \frac{N}{\tau}, \quad 1 \le k \le \tau \tag{9}$$

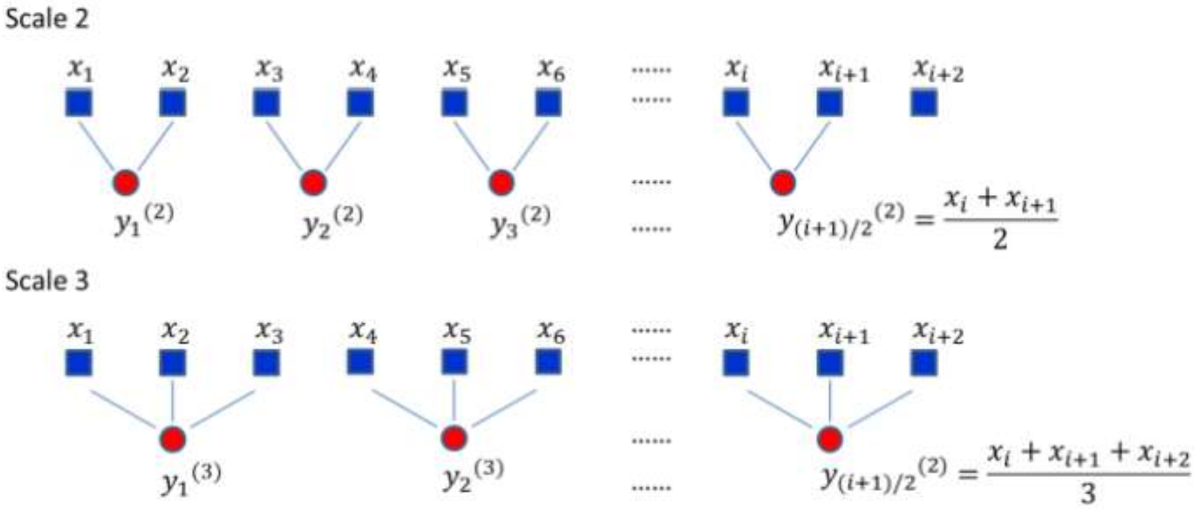
Scale 2
x_1 x_2 x_3 x_4 x_5 x_6 ...... x_i x_{i+1} x_{i+2}
$y_1{}^{(2)}$ $y_2{}^{(2)}$ $y_3{}^{(2)}$ ......
$y_{(i+1)/2}{}^{(2)} = \frac{x_i + x_{i+1}}{2}$
Scale 3
x_1 x_2 x_3 x_4 x_5 x_6 ...... x_i x_{i+1} x_{i+2}
$y_1{}^{(3)}$ $y_2{}^{(3)}$ ......
$y_{(i+1)/2}{}^{(2)} = \frac{x_i + x_{i+1} + x_{i+2}}{3}$

Figure 2- Schematic illustration of the coarse-grained procedure (Shuen-De Wu, et al., 2013)

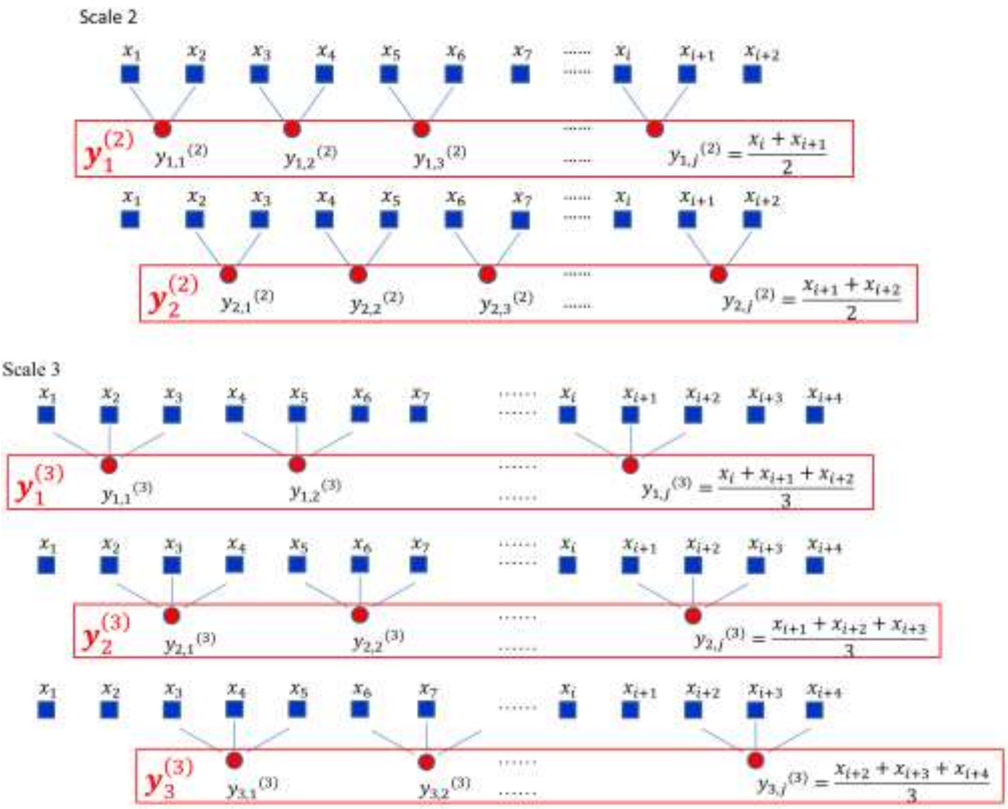
Scale 2
$y_1^{(2)}$ $y_{1,1}{}^{(2)}$ $y_{1,2}{}^{(2)}$ $y_{1,3}{}^{(2)}$ $y_{1,j}{}^{(2)} = \frac{x_i + x_{i+1}}{2}$
$y_2^{(2)}$ $y_{2,1}{}^{(2)}$ $y_{2,2}{}^{(2)}$ $y_{2,3}{}^{(2)}$ $y_{2,j}{}^{(2)} = \frac{x_{i+1} + x_{i+2}}{2}$
Scale 3
$y_1^{(3)}$ $y_{1,1}{}^{(3)}$ $y_{1,2}{}^{(3)}$ $y_{1,j}{}^{(3)} = \frac{x_i + x_{i+1} + x_{i+2}}{3}$
$y_2^{(3)}$ $y_{2,1}{}^{(3)}$ $y_{2,2}{}^{(3)}$ $y_{2,j}{}^{(3)} = \frac{x_{i+1} + x_{i+2} + x_{i+3}}{3}$
$y_3^{(3)}$ $y_{3,1}{}^{(3)}$ $y_{3,2}{}^{(3)}$ $y_{3,j}{}^{(3)} = \frac{x_{i+2} + x_{i+3} + x_{i+4}}{3}$

Figure 3- Schematic illustration of the CMSE procedure (Shuen-De Wu, et al., 2013)

In the conventional MSE algorithm, for each scale, the MSE is computed by only using the first coarse-grained time series, $y_1^{(\tau)}$:

$$\text{MSE}(\text{x}, \tau, m, r) = \text{SamEn}(y_1^{(\tau)}, m, r) \tag{10}$$

In the CMSE algorithm, at a scale factor of τ, the sample entropies of all coarse-grained time series are calculated and the CMSE value is defined as the means of τ entropy values that is shown in equation (11):

$$\text{CMSE}(\text{x}, \tau, m, r) = \frac{1}{\tau}\sum_{k=1}^{\tau} SampEn(y_1^{(\tau)}, m, r) \tag{11}$$

The MATLAB code of CMSE is proposed in appendix 2 (Shuen-De Wu, et al., 2013).

Conclusion and results

We run both FCM and OR model to separate the market into 5 clusters and construct the index fund –tracking portfolio- we used the first 708 records to run both models and the last 300 records to test them and calculate the error. Error is calculated as follow for both model just like (Beasley, et al., 2003):

$$\text{Error} = \text{STD}(\text{tracking portfolio return} - \text{index return}) \tag{12}$$

Our model highly reduced the running time of the model compared to OR model, although its error is more than the OR model. The advantage of reducing time cost shows itself in a larger markets which consists of thousands of stocks. Table 2, shows the results of both model.

Table 2-results of FCM and OR model

model	Error	%change	Time (s)	%change
OR	0.014		554.735	
FCM	0.0167	19.2857	24.536	-0.95577

For future works, using other features of time series or using another method of fuzzy clustering may reduce the error while keep the time cost of the model in an extremely low level.

References

Alex Frino, David R, D. & Gallagher, 2001. Tracking S&P 500 Index Funds,. *The Journal of Portfolio Management , Vol. 28, No. 1,* pp. 44-55.

Andrew Rudd, 1980. Optimal Selection of Passive Portfolios,. *Financial Management, Vol. 9, No. 1 ,,* pp. 57-66.

Beasley, J. E., M. N. & Chang, T. J., 2003. An evolutionary heuristic for the index tracking problem.. *European Journal of Operations Research ,* 148(3), p. 621–643..

Beasley & Meade, N., 2004. An evaluation of passive strategies to beat the index. *The Tanaka Business School,.*

Bezdek, J. C., 1981. Pattern Recognition with Fuzzy Objective Function Algoritms. *Plenum Press, New York.*

Bogle, J. C., 1992. Selecting equity mutual funds.. *The Journal of Portfolio Management ,* pp. 94-100..

Chen Chen & Kwon, R. H., 2012. Robust portfolio selection for index tracking,. *Computers & Operations Research,* p. 829–837..

Christian Dose & Cincotti, S., 2005. Clustering of financial time series with application to index and enhanced index tracking portfolio. *Physica A,* p. 145–151.

Cornuejols G & R, T., 2007. *Optimization methods in finance.* s.l.:Cambridge University Press..

Costa, M., Goldberger, A. & Peng, C., 2002. Multiscale entropy analysis of complex physiologic time series. *Phys. Rev. Lett.*

Costa, M., Peng, C., Goldberger, A. & Hausdorff, J., 2003. Multiscale entropy analysis of human gait dynamics. *Physica A,* Volume 330, p. 53–60..

Diana Barro & Canestrelli, E., 2009. Tracking error: a multistage portfolio model. *Annals of Operations Research,* pp. Volume 165, Issue 1, pp 47-66.

Dunn, J. C., 1973. A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters. *Journal of Cybernetics,* pp. 32-57.

Francesco Coriellia & Marcellino, M., 2006. Factor based index tracking. *Journal of Banking & Finance,* Volume 30(Issue 8), p. 2215–2233.

G Salkin & Meade, N., 1989. Index funds-construction and performance measurement. *the Journal of Operational Research Society,* 40(10), p. 871–879.

He Ni, Y. W., 2013. Stock Index tracking by Pareto efficient genetic algorithm.. *Journal of Applied Soft Computing 13, ,* p. 4519–4535..

Hodges. & D, S., 1976. Problems in the application of portfolio selection models.. *Omega, Volume 4, Issue 6,* p. 699–709.

N.A. Canakgoz, 2009. Mixed-integer programming approaches for index tracking and enhanced indexation. *European Journal of Operational Research 196 (2118),* p. 384–399.

Okaya, N. & Uğur Akmanb, ,., 2003. Index tracking with constraint aggregation,. *Applied Economics Letters,* pp. Volume 10, Issue 14.

Pan, Y., Lin, W., Wang, Y. & Lee, K., 2011. Computing multiscale entropy with orthogonal range search. *J. Mar. Sci. Technol,* Volume 19, p. 107–113.

Resnick, G. & Larsen-Jr., .., 1998. Empirical insights on indexing. *The Journal of Portfolio Management,* 25(1), p. 51–60.

Richman, J. & Moorman, J., 2000. Physiological time-series analysis using approximate entropy and sample entropy.. *Physiol. Heart Circul. Physiol,* Volume 278, pp. 2039-2049.

Roll, R., 1992 . A mean/variance analysis of tracking error.. *The Journal of Portfolio Management 18 (Summer),* , pp. 13 - 22 ..

Salkin & Meade, N., 1990. Developing and maintaining an equity index fund. *The Journal of Operational Research Society,* 41(7), p. 599–607.

Sharpe, W. F. .., September 1964. CAPITAL ASSET PRICES: A THEORY OF MARKET EQUILIBRIUM UNDER CONDITIONS OF RISK.. *The Journal of Finance,Volume 19, Issue 3,,* p. 425–442.

Shuen-De Wu, et al., 2013. Time Series Analysis Using Composite Multiscale Entropy. *entropy,* Volume 15, pp. 1069-1084.

Takeda, Y. T. a. E., 1995. Bicriteria optimization problem of designing an index fund. *The Journal of the Operational Research Society,* 46(8), p. 1023–1032.

Appendix 1

The FCM algorithm is composed of the following steps:

1. *Initialize U=[u_{ij}] matrix, $U^{(0)}$*
2. *At k-step: calculate the centers vectors $C^{(k)}$=[c_j] with $U^{(k)}$*

$$c_j = \frac{\sum_{i=1}^{N} u_{ij}^{m} . x_i}{\sum_{i=1}^{N} u_{ij}^{m}}$$

3. *Update $U^{(k)}$, $U^{(k+1)}$*

$$u_{ij} = \frac{1}{\sum_{k=1}^{c} \left(\frac{||x_i - c_j||}{||x_i - c_k||} \right)^{\frac{2}{m-1}}}$$

4. *If || $U^{(k+1)}$ - $U^{(k)}$||< ε then STOP; otherwise return to step 2.*

Appendix 2

The Matlab Code for the Composite Multiscale Entropy Algorithm

```
function E = CMSE(data,scale)
r = 0.15*std(data);

for i = 1:scale   % i:scale index
    for j = 1:i   % j:croasegrain series index
        buf = CoarseGrain(data(j:end),i);
        P(i) =SampEn(buf,r);
    end
end
 SP=sum(P);
 E=SP/scale;
```

```
%Coarse Grain Procedure
% iSig: input signal ; s : scale numbers ; oSig: output signal

function oSig=CoarseGrain(iSig,s)
N=length(iSig); %length of input signal
for i=1:1:N/s
    oSig(i)=mean(iSig((i-1)*s+1:i*s));
end

%function to calculate sample entropy. See Algorithm 1
function entropy = SampEn(data,r)
l = length(data);
Nn = 0;
Nd = 0;
for i = 1:l-2
    for j = i+1:l-2
        if abs(data(i)-data(j))<r && abs(data(i+1)-data(j+1))<r
            Nn = Nn+1;
            if abs(data(i+2)-data(j+2))<r
                Nd = Nd+1;
            end
        end
    end
end
entropy = -log(Nd/Nn);
```