Национальный исследовательский университет -

Высшая школа экономики

Международный Институт Экономики и Финансов

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

на тему: Application of Collaborative Filtering to Financial Time Series: Stock price prediction

Студент 4 курса

Лобойко Арсений Александрович

Научный руководитель

Чугай Роман Николаевич

МОСКВА, 2013 год.

Contents

Abstract

Generally, financial time series were found to be deterministically chaotic, non-stationary and complex. Since the invention of CAPM and APT researchers tried to find factors which influence stock prices. For this purpose they used historical market data, accounting variables, macroeconomic indicators etc. The later papers in this field tried to apply dimensionality reduction techniques in the form of Principal Component Analysis (PCA) or Independent Component Analysis (ICA) to identify the most influential parameters in the context of forecasting model. This paper investigates the application of Collaborative Filtering (CF) to stock market in attempt to reveal the latent drivers of stocks price movements and construct a model of stock price prediction using only information on past returns.

We apply CF to the one year data of daily returns for 406 companies from S&P500 Index and compare the results with those obtained using Principal Component Analysis. Our research indicates that Collaborative Filtering possesses the same dimensionality reduction power as PCA. However, being model specific and not restricted by orthogonality of its principal components CF outperforms PCA in stock price prediction. All in all, we show that CF may be used to forecast stock returns on the basis of sole historical price information and outperform the market.

1 Introduction

The question of stock price prediction has been in the limelight for many decades. It usually falls into the discussion of weak, semi-strong and strong forms of market efficiency introduced by Roberts (1967). According to his definition a market is said to be efficient with respect to information set Ω_t if it is impossible to make economic profits by trading on the basis of the information set Ω_t which includes historical prices, publicly available information and private information for weak, semi-strong and strong forms of efficiency respectively. A little broader definition was later given by Timmermann and Granger (2004) who added additional restrictions on efficiency requiring the market to be efficient also with respect to search technologies, S_t, and forecasting models, M_t, so that it is impossible to make economic profits by trading on the basis of signals produced from a forecasting model in M_t defined over a predictor variables in the information set Ω_t and selected using a search technology in S_t.

In our paper we are interested in the possibility for predicting stock prices. The wide review on the weak-form efficiency literature with a special focus on the stock markets is provided by Lim and Brooks (2011). The whole work in this filed may be divided into two large groups. The first group tests deviations of financial series from random walk model using unit-root test, linear serial correlations, non-linear serial dependence, long memory etc. The second group studies profitability of trading strategies based on past returns such as technical trading rules (see Park and Irwin (2007) for the review of latest studies in this sphere).

In recent years researchers started using machine learning to reduce the number of components in predicting stock returns. Lim and Ngerng (2012) apply Principal Component Analysis (PCA) with Varimax rotation to decrease the number of explanatory variables to only 4 in regression of Returns-on-Equity (proxy for stock return) on more than 70 input variables. They found excess of equity returns over risk-free rate, current year profitability, future profit (proxy for an analysts' predictions), leverage and standard deviation of Returns-on-Equity to be the drivers of stock return which explain 39.6% of variation in returns. Thus, they extended the work of Fama and French (1993) who suggested a three-factor model for stock returns which included market risk premium (long position in value-weighted index portfolio and short position in T-Bills), Book-to-Market ratio and a size of the firm. The model was later extended

by Carhart (1997) to include the fourth factor - Momentum which controls the return of stocks in terms of previous year's stock market performance.

There are also some papers which attempt to use PCA for testing technical trading rules prediction power. E.g. Ince and Trafalis (2007) apply Kernel Principal Component Analysis – a non-linear version of PCA, which maps data from the input space to a feature space where linear PCA is consequently used (see Appendix 6). They later use kPCA components in Support Vector Regression (SVR) (see Appendix 3) and MLP neural network for stock price prediction. Despite using non-linear techniques they still refer to the initial set of explanatory variabels (more than 100) and technical indicators such as Exponential Moving Average (EMA), Relative Strength Index (RSI), Bollinger Band (BB), Moving Average Convergence Divergence (MACD), Chaikin Money Flow (CMF). Lu, Lee and Chiu (2009) perform similar technique trying to predict Nikkei 225 Index using its futures prices traded on different bourses (SGX-DT, OSE, CME), technical indicators and previous day cash market stock index. They use Independent Component Analysis (ICA) to reduce variation in input data before using it for SVR model.

As we see there are little studies in the field of stock price prediction using PCA or similar techniques. Recent papers use dimensionality reduction techniques mainly to get rid of redundancy in the input data. Some papers used the technique to reveal the number of latent factors influencing stock price movements. The most famous is the work of Roll and Ross (1980) where they use Factor Analysis (see Appendix 2 for FA description) to find the number of explanatory factors for APT model. Later Back and Weigend (1997) used Independent Component Analysis (ICA) to show existence of some underlying structure in stock price data although they didn't test the prediction power of ICA.

In this paper we apply Collaborative Filtering to financial time series data which has never been done before. Collaborative Filtering was mainly used in the recommender systems by online giants such as Google or eBay to reveal customers preferences and make in-sample forecasting of the products they will most likely buy. We use Collaborative Filtering technique to reveal latent dependencies among stock price movements. We expand the usual CF model to include lagged factor variables and use it to make out-of-sample forecasting of stock returns. The resulting model appeared to forecast stock price movements hundreds of times better than if they followed a simple Random Walk process while ICA-SVR model of Lu, Lee and Chiu increased accuracy only in 3-5 times. Despite a significant increase in accuracy the model

provides only a slight increase in the probability of determining the market direction (51.13%) because of the structure of stock returns which are distributed around almost zero mean. Still a simple investment strategy of full diversification on the basis of Collaborating Filtering allows making significantly higher returns (8.29%) than the naïve strategy of taking diversified long position in all stocks (-0.29%). Ignoring investment into stocks which move less than 0.1% in a day eliminates the problem of zero-mean distribution and significantly enhances the probability of predicting the direction of the market (52.11%) which peaks at the level of 54.66% for the investment in stocks with less than 0.8% of daily returns providing return of more than 30% in half a year.

The rest of the paper is organized in the following way. Section 2 provides a background to Collaborative Filtering as a recommender system, its application to financial time series data and a guide to some algorithms of CF assessment. Section 3 examines possibility of CF to eliminate redundancy in data dimensions and compares it with other known models of dimensionality reduction (PCA and MLE Factor Analysis). Our specific results on the prediction properties of CF to the stocks of S&P500 Index are given in Section 4 together with its robustness evaluation and significance tests. Section 5 summarizes our findings and concludes.

2 Collaborative Filtering in General

2.1 Collaborative Filtering

Collaborative Filtering (CF) is a "class of methods that recommend items to users based on preferences other users have expressed for those items" [1]. This type of models appeared in 1990s as a solution for dealing with information overload primarily in online industry and was later integrated into online systems and e-commerce. Amazon.com, Netflix, eBay, Last.fm and other e-commerce companies integrated Collaborative Filtering to recommend items for users to buy on the basis of browsing and purchase history (see Ekstrand, Riedl and Konstan (2010) or Lathia (2010) for more information on CF).

Despite the wide use in e-commerce the method found little application to financial time series. The reason for this is that CF is based on the preferences of agents revealed through the process of trading. However, the information on the trading activity of each market participant is not public. Hence, Collaborative Filtering lacks input information for assessment nevertheless, researchers find different proxies for it. E.g. Avery, Chevalier and Zeckhauser (2009) use data on the future stock price kept by the Motley Fool Company which can be found on the CAPS website[2]. They divide all stocks into 4 groups on the basis of their star-rating on CAPS site and apply a standard four factor model of Fama and French (1993) and Carhart for each of the group. They found that different stock groups had different factor loading indicating that CAPS participants may have a payoff-relevant information for stock price determination. Hence, this paper pioneers the use of Collaborative Filtering to the financial time series data.

2.2 Collaborative Filtering in Recommender Systems

Collaborative Filtering helps to explain variation in observed returns by a smaller set of unobserved (latent) factors. It is easier to show how CF is used in Recommender Systems before explaining its application to time series data. Consider a movie example taken from online lectures on Machine Learning of Andrew Ng – a professor in Stanford University[3].

Assume we have a sample of films and a sample of moviegoers. When a moviegoer watches film he gives it a rating from 0 to 5. Knowing who and how rated films we would like to estimate

[1] Source: Michael D. Ekstrand, John T. Riedl and Joseph A. Konstan "Collaborative Filtering Recommender Systems"
[2] www.caps.fool.com
[3] https://class.coursera.org/ml/lecture/preview

the ratings of the remaining films not yet watched by moviegoers. Consider Table 1 which shows how each of 5 films was rated by 4 different moviegoers. We assume that each film possess latent features X (degree of romantic and action in our case) which are rated by the viewers. Each moviegoer has got its own preference regarding romantic and action nature of the film represented by a vector θ where $\theta \in M(2,1)$ in our case. Therefore, if we know the latent feature X of each film we can easily estimate the preferences of users θ by minimizing the squared error of our prediction adjusted for regularization term $\frac{\lambda}{2}\sum_{j=1}^{n_u}\sum_{k=1}^{n}(\theta_k^{(j)})^2$ which allows to prevent model overfitting:

$$\min_{\theta^{(1)},\dots\theta^{(n_u)}} \frac{1}{2}\sum_{j=1}^{n_u}\sum_{i:r(i,j)=1}(\theta^{(j)})^T x^{(i)} - y^{(i,j)})^2 + \frac{\lambda}{2}\sum_{j=1}^{n_u}\sum_{k=1}^{n}(\theta_k^{(j)})^2 ,$$

where n_u - is the number of users

k – is the number of latent factors

y – is the rating of film i by user j

λ – regularization coefficient

Table 1

Movie	Alice	Bob	Carol	Dave	X1 (romance)	X2 (action)
Love at last	**5**	**5**	**0**	**0**	**0.9**	**0**
Romance forever	**5**	**?**	**?**	**0**	**1**	**0.01**
Cute puppies of love	**?**	**4**	**0**	**?**	**0.99**	**0**
Nonstop car chases	**0**	**0**	**5**	**4**	**0.1**	**1**
Swords vs. karate	**0**	**0**	**5**	**?**	**0**	**0.9**

On the other hand, if we know preferences of each individual θ together we the rating matrix Y we can estimate the value of latent factors X by minimizing the following cost function:

$$\min_{x^{(1)},\dots x^{(n_m)}} \frac{1}{2}\sum_{i=1}^{n_m}\sum_{j:r(i,j)=1}(\theta^{(j)})^T x^{(i)} - y^{(i,j)})^2 + \frac{\lambda}{2}\sum_{j=1}^{n_m}\sum_{k=1}^{n}(x_k^{(i)})^2 ,$$

where n_m – is the number of films under assessment.

However, in reality we do not have either the matrix of preferences θ or the matrix of latent factors X . This problem may be solved via Collaborative Filtering by randomly generating the

matrix of preferences θ and the matrix of latent factors X, next minimizing our cost function by θ and X alternatively, so that we keep θ constant and find optimal X values. Then X is kept constant and minimization proceeds by θ argument. Further θ are again fixed and we seek for optimal X. This iterative process (collaboration) is repeated until the cost function converges. Another way is to minimize augmented cost function by both θ and X using Gradient Descent technique (see section 2.4):

$$\min_{\substack{x^{(1)},\dots x^{(n_m)},\\ \theta^{(1)},\dots \theta^{(n_u)}}} \frac{1}{2}\sum_{i=1}^{n_m}\sum_{j:r(i,j)=1}(\theta^{(j)})^T x^{(i)} - y^{(i,j)})^2 + \frac{\lambda}{2}\sum_{j=1}^{n_m}\sum_{k=1}^{n}(x_k^{(i)})^2 + \frac{\lambda}{2}\sum_{j=1}^{n_u}\sum_{k=1}^{n}(\theta_k^{(j)})^2$$

2.3 Collaborative Filtering for time series data

Let's consider now our implementation of Collaborative Filtering to time series data. According to the Arbitrage Pricing Theory proposed by Ross (1976) stock price movements are caused by a number of latent factors θ. All stocks are subject to the same number of factors although have different exposure to them. Hence, the stock return may be described by the following model:

$$R_i = E_i + \beta_{i1}\theta_1 + \dots + \beta_{ik}\theta_k + \varepsilon_i,$$
$$= E_i + \beta_i\theta + \varepsilon_i,$$

Where ε_i are mutually stochastically uncorrelated disturbance terms, $E(\varepsilon_i) = 0$.

$E(\theta_i) = 0$ and E_i is a mean stock return.

Let's change the notation and call β's the loading factors and θs the states or state variables. We assume that loading factors β differ from stock to stock but are kept constant in time, while states variables θ are similar for all stocks but differ in time. Hence, we can write that

$$R_{it} = E_i + \beta_{i1}\theta_{1t} + \dots + \beta_{ik}\theta_{kt} + \varepsilon_{it}$$

Then we can represent the returns of the stocks in a matrix form $Y \in M(n_m, n_u)$ which looks similar to the one in Table 1. The elements of the matrix are stock returns R_{ij} with stocks being in the rows and time periods in columns (see Table 2).

Table 2

	Time 1	Time 2	Time 3	Factor 1
Share 1	**R11**	**R12**	**R13**	**β_1**
Share 2	**R21**	**R22**	**R23**	**β_2**
Share 3	**R31**	**R32**	**R33**	**β_3**
Share 4	**R41**	**R42**	**R43**	**β_4**
…	**…**	**…**	**…**	

We can estimate both loadings and state variables using Collaborative Filtering in a similar way as it was done for movies. We simply minimize the cost function yet without regularization term as we are interested in the number of significant factors for stock return determination and not in the forecasting abilities of the model:

$$\min_{\substack{x^{(1)},\dots x^{(n_m)},\\ \theta^{(1)},\dots \theta^{(n_u)}}} J(x^{(1)},\dots,x^{(n_m)},\theta^{(1)},\dots,\theta^{(n_u)}) = \frac{1}{2}\sum_{i=1}^{n_m}\sum_{j=1}^{n_u}(\theta^{(j)})^T x^{(i)} - y^{(i,j)})^2$$

Where $y^{(i,j)}$ - return of stock i at time j

Let k be the number of significant factors, then

$x^{(i)}$ – is a vector of factor loadings for stock i, $x^{(i)} \in M(1,k)$

$\theta^{(j)}$ – is a vector of state variables at time j, $\theta^{j} \in M(k,1)$

n_m – overall number of stocks

n_u – overall number of time periods

2.4 Gradient Descent

The data we are using is the stock returns of the companies from S&P500 Index for the period of 244 trading days for the year 2009. The stocks with insufficient number of observations were not included into our sample so a total of 406 stocks were analyzed. Hence, matrix of factor loadings is $x \in M(406,k)$ the matrix of state variables is $\theta \in M(k,244)$ and the matrix of returns is $Y \in M(406,244)$.

To evaluate the loadings and state variables we apply Gradient Descent in MatLab software. As gradient shows the direction of the fastest increase in function, then conducting small steps in the opposite direction allows finding the local minima of the function or the global minima in

case the function is convex. Let's denote the size of the step (learning rate) by α. To determine the gradient we need to calculate the following First Order Conditions:

$$FOC_{x_k^{(i)}} = \sum_j (\theta^{(j)^T} x^{(i)} - y^{(i,j)})\theta_k^{(j)}$$

$$FOC_{\theta_k^{(j)}} = \sum_j (\theta^{(j)^T} x^{(i)} - y^{(i,j)})x_k^{(i)}$$

Then the cost function $J(\vec{x},\vec{\theta})$ will decrease with each iteration $J(\vec{x}_{m+1},\vec{\theta}_{m+1}) \le J(\vec{x}_m,\vec{\theta}_m)$ for $x_{k,m+1}^{(i)} = x_{k,m}^{(i)} - \alpha\, FOC_{x_k^{(i)}}$ and $\theta_{k,m+1}^{(j)} = \theta_{k,m}^{(j)} - \alpha\, FOC_{\theta_k^{(j)}}$ until it converges after

$\frac{J(\vec{x}_{m+1},\vec{\theta}_{m+1}) - J(\vec{x}_m,\vec{\theta}_m)}{J(\vec{x}_m,\vec{\theta}_m)} \le \text{minimum gain}$, where minimum gain was set at the level of 10^{-10} for our analysis.

We choose α to be 0.01 as it is the maximum learning rate at which our Loss function does not diverge. The closer we are to the optimum the smaller is the convergence rate as the value of the gradient falls. To accelerate the process we increase the learning rate 0.01 for each 500 iterations. This allows decreasing the number of iterations needed for the series to converge in 3 times.

3 Collaborative Filtering and Dimensionality Reduction

3.1 Comparison with Principal Component Analysis and Factor Analysis

After the invention of an Arbitrage Pricing Theory by Ross (1976) a lot of studies were conducted to determine the number of factors which influence stock price movements. Most of them used observed explanatory variables while only a few referred to the dimensionality reduction technique. The most prominent work in this field was done by Roll and Ross. They applied maximum likelihood factor analysis to estimate the number of factors and the matrix of loadings for 1260 stocks for the period from 3 July 1962 to 31 December 1972. They divided the stocks into groups of 30 securities (42 groups as a whole) because of the processor limitation of their computer. They found out that in 88.1% of the groups there was at least one factor with non-zero risk premium, in 57.1% at least two factors, in 33% at least three factors. Hence, they concluded that at least three factors are important for APT but probably no more than four. This finding is still in line with most of the papers. Fama and French (1993) revealed 3 factors that drive stock return, later expanded to 4 by Cahart. Chen, Roll and Ross (1986) tried to find macroeconomic model for stock return determination. They revealed such important factors in explaining stock price movements as changes in GDP growth rates, changes in default risk premium represented by the spread between Yield-to-Maturity on AAA and BBB rated bonds, changes in the slope of Yield Curve represented by the spread between Long Term and Short Term bond rates – a total of 3 factors and a few less significant ones. The latest study of Lim and Ngerng found 4 significant factors by applying PCA to the number of accounting indicators. All in all, most paper findings have been consistent with conclusions drawn by Roll and Ross that there are 3 to 5 drivers of stock returns.

3.2 Comparison of Collaborative Filtering with Principal Component Analysis and Factor Analysis: number of significant factors determination.

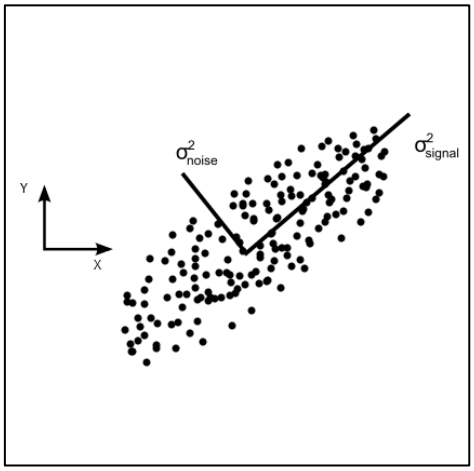
Besides Collaborative Filtering there exist a number of other dimensionality reduction techniques such as Principal Component Analysis (PCA) and Factor Analysis (FA). All of them are quite similar but CF may give superior results in some cases described below.

PCA is simply a data reduction technique which does not assume the existence of underlying variable (e.g. Shlens (2009) and Lathia). Its main aim is finding a linear combination of original basis vectors to re-express the data set in a way to decorrelate it. This is done by choosing the

new basis vectors (principal components) in the direction of the largest variance. The consequent principal components are constructed to be orthogonal to the initial ones. So, the method assumes linearity and orthogonality of principal components as well as importance of large variance structure. Consider Figure 1 where data points represented in a two-dimensional space. PCA will choose the first principal component to explain the largest variation in data (σ^2_{signal}). The second component will explain the largest variation in the remaining directions orthogonal to the principal component already chosen σ^2_{noise}.

Figure 1

PCA algorithm of principal components determination



The main goal of PCA is to reduce redundancy in data X which is done by eliminating any covariance between matrix elements and making the matrix diagonal. Hence, PCA finds an orthonormal transformation matrix P that transforms matrix X into the matrix Y ($Y = PX$) such that covariance matrix of Y is diagonal - $\frac{1}{n}YY^T = C_Y$, where C_Y is diagonal. The estimation of principal components is usually conducted via Singular Value Decomposition technique (see Appendix 1).

Let Y be the matrix of returns where rows represent different shares and columns different time period - $Y \in M(406,244)$. For the PCA to be consistent with CF with need to apply SVD to transposed matrix of returns Y^T so that it will be decomposed into an orthogonal matrix, a diagonal matrix, and another orthogonal matrix:

$$Y^T = U\Sigma V^T$$

Where U a column orthogonal matrix $U \in M(244,406)$, S is a diagonal matrix consisting of singular values $S \in M(244,406)$ and $V \in M(406,406)$. The principal components are given by the vectors of orthonormal columns in U weighted by singular values from S:

$Y^T V = US$.

In regard to time series data the rows of matrix V^T will correspond to the factor loadings and columns to shares while columns of the matrix product US will contain state variables and rows of US will show the evolution of the state variable through time.

We believe that CF may give superior results to PCA in some cases regarding both dimensionality reduction and stock price prediction. The main reason for this is that PCA assumes orthogonality of principal components so if the data is located as in Figure 2 (3-d space), the PCA will most likely find principal components reproduced by red arrows, although it is clearly seen that variation will be lower if we try to explain data on the basis of principal components represented by blue arrows. Moreover, the prediction power of Collaborative Filtering should also be higher which is clearly seen on Figure 3 where two intersecting clouds represent data points for IT and Oil industries. PCA will find principal components represented by red arrows while CF will find a basis corresponding to blue arrows. So if the basis represents securities' factor loadings the results of prediction for two models may be quite different. This is also confirmed by Ekstrand: "The resulting model will not be a true SVD of a rating matrix, as the component matrices are no longer orthogonal, but tends to be more accurate at predicting unseen preferences than the unregularized SVD"[4]

[4] Michael D. Ekstrand et al., "Collaborative Filtering Recommender Systems", Vol. 4, No. 2 (2010), page 104

Figure 2
Failure of PCA in dimensionality reduction

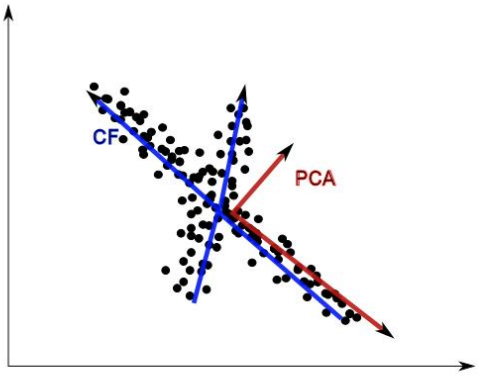
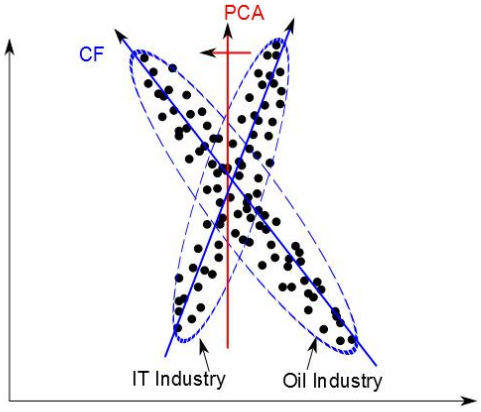


Figure 3
Failure of PCA in forecasting ability



Another technique for revealing latent structure of the data is Factor Analysis (FA). While Principal Component Analysis is just a dimensionality reduction technique and considers all the variance, Factor Analysis looks only at the common variance among indicators by explicitly modeling measurement error. It assumes that a set of observed variables Y depends on a set of unobserved (latent) factors f, so the model may be represented as follows:

$$y_i = \sum_j \lambda_{ij} f_j + \varepsilon_i, \ \forall i$$

Where λ_{ij} is the j-th factor loading, f_j is the j-th common factor, ε_i is the error term for variable i.

In the matrix form

$$X = \Lambda F + \varepsilon$$

Where $F \sim N(0, I)$ and $\varepsilon \sim N(0, \Psi)$, both independent of each other.

The equation is then estimated using Maximum Likelihood technique (see Appendix 2).

So, CF and FA are almost the same as both take into account measurement error. However, CF allows for inclusion of different number of lags and non-linearity of the model while FA does not.

The procedure of defining the number of components is quite subjective (e.g. Abdi and Williams (2010)). We use the scree or elbow test. So, we choose the number of components at

which the slope of the graph changes sharply from steep to flat ("elbow"), accounting for the components which provide more than 2% explanation in variation to our model. We also consider the overall level of explained variation treating 50 percent as a threshold value.

Consider the scree plot for Collaborative Filtering Analysis (see Figure 4 and 5). The plot shows how much variation is explained by each additional component measured by R – squared. While Collaborative Filtering and PCA are calculated using MatLab, Factor Analysis was performed using SPSS application. Comparing all three methods (CF with PCA and FA) we can see that they provide almost similar results (see Figure 6). On the basis of our graphs we see that CF may not be treated as a technique superior to either PCA or MLE FA in the dimensionality reduction exercise as the difference in explaining power between models for all components does not exceed 10^{-8}. However, CF at least performs not worse than the other two models.

On the basis of the chosen criteria (scree plot, additional variation explained, threshold value of 50% explained variation) we find 3 factors to be significant which explain 51% of variation in stock price movements. It is consistent with the findings of other papers of Lim and Ngerng, Fama and French, Roll and Ross, Chen, Roll and Ross which generally find from 3 to 5 factors to be significant in explaining stock returns.

Figure 4

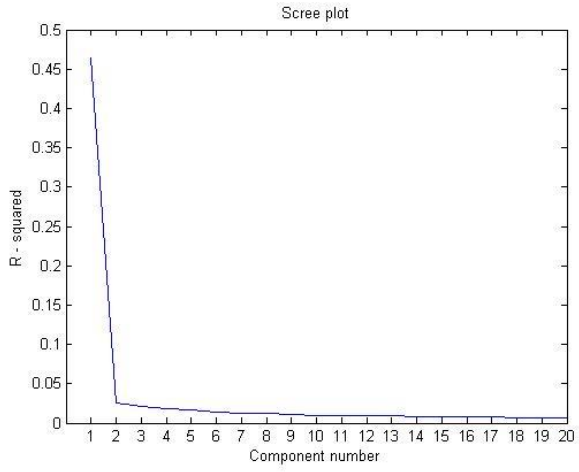
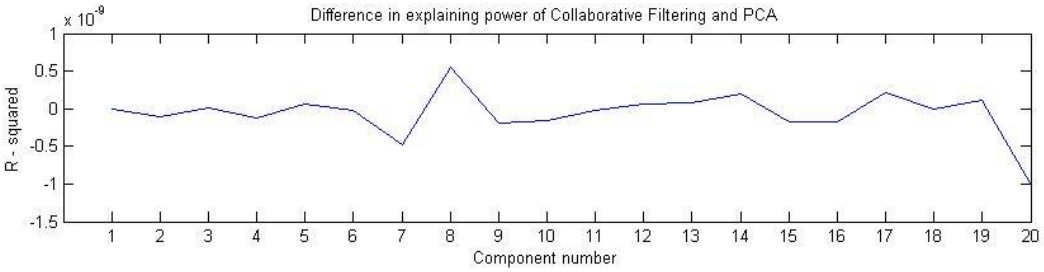
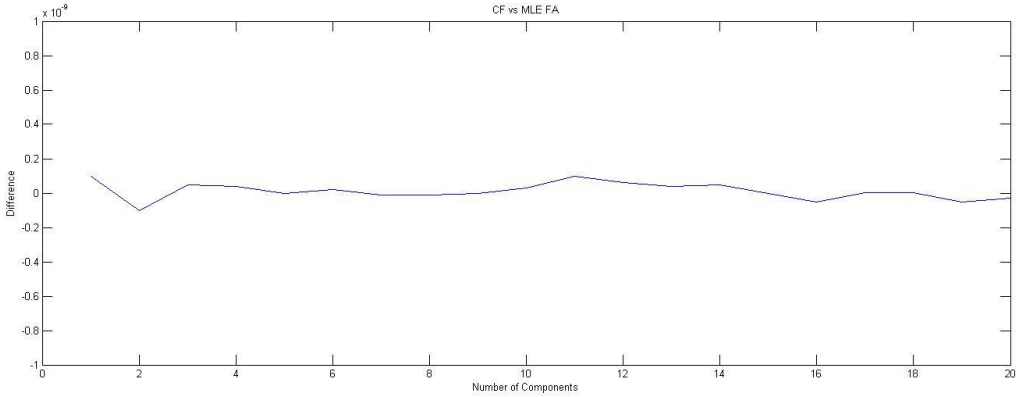


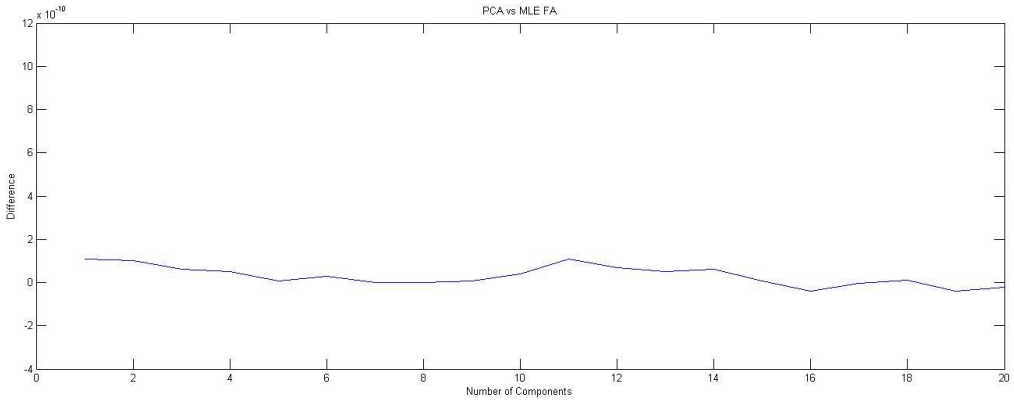
Figure 5

Number of Components	Additional Variation Explained	Overall Variation Explained
1	46.37%	46.37%
2	2.54%	48.91%
3	2.09%	51.00%
4	1.78%	52.78%
5	1.63%	54.41%
6	1.44%	55.85%
7	1.30%	57.15%
8	1.26%	58.41%
9	1.08%	59.49%
10	1.02%	60.51%
11	0.98%	61.49%
12	0.95%	62.44%
13	0.91%	63.35%
14	0.87%	64.22%
15	0.82%	65.04%
16	0.78%	65.82%
17	0.75%	66.57%
18	0.71%	67.28%
19	0.68%	67.96%
20	0.62%	68.58%

Figure 6


Difference in explaining power of Collaborative Filtering and PCA
x 10^-9
R - squared
Component number


CF vs MLE FA
x 10^-9
Difference
Number of Components


PCA vs MLE FA
x 10^-10
Difference
Number of Components

4 Collaborative Filtering and Stock Price Prediction

4.1 Weak form of Market Efficiency

The main goal of this paper is to construct a model on the basis of historical prices to predict future returns. Hence, it is mainly concerned with a weak form of market efficiency. The review of previous works in this field may be found in Lim (2011). They can be divided into a two broad groups. The first group tests predictability of stock returns on the basis of historical prices analyzing mainly deviations of stock returns from the random walk model. This is done in a number of ways by applying unit root tests, long memory tests, analyzing serial correlation of returns of both linear and non-linear types etc. The second group tests profitability of trading strategies on the basis of past returns such as momentum and contrarian strategies, technical trading rules etc. Usually the market is found to be efficient (e.g. Dai and Lee (2011); Chen, Huang and Lai (2011) or Roll and Ross). However, some studies reveal positive autocorrelation of stock returns for the short time intervals (e.g. Kung and Carverhill (2012) or Lo and MacKinlay (1998)) although they can be explained by infrequent trading of the securities under study. At the same time researches find negative autocorrelation of returns on long-term horizon. This can be caused by a mean reverting process which is not picked up by return generating model providing issues for joint hypothesis problem (e.g. Fama and French (1988)). However, recent study of Boudoukh, Richardson and Whitelaw (2008) shows that long-term returns cannot be predicted. Overall, there is mixed evidence on the market efficiency of all types.

The current studies in this field use dimensionality reduction technique mainly to reduce redundancy in input data for the non-linear forecasting models of SVR or neural network type (e.g. Ince and Trafalis or Lu, Lee and Chiu). Collaborative Filtering does not require any input data except prices although additional explanatory variables may enhance the predictive power of the model. In our paper we apply a linear version of Collaborative Filtering with one lag but still there is exist a possibility of increasing the number of lags and introducing cross-terms to add non-linearity to the model. The comparison of forecasting power of SVR and neural network models with extended versions of CF is left for future work.

4.2 Proposed forecasting model using Collaborative Filtering

Investigating the relevance of three factors in stock price movements we attempt to construct a forecasting model for stock returns by including lagged state variables. So, we use a gradient

descent technique to estimate both state variables and factor loadings of the following type of model:

$$r_{i,t} = \beta_{i,1}\theta_{1,t} + \beta_{i,2}\theta_{2,t} + \beta_{i,3}\theta_{3,t} + \beta_{i,4}\theta_{1,t-1} + \beta_{i,5}\theta_{2,t-1} + \beta_{i,6}\theta_{3,t-1}$$

We use a rolling window of size 10, 50 and 100 to estimate parameters of the model and predict values of future returns $r_{i,t+1}$ based on the current state variables so that predicted returns equal

$$\hat{r}_{i,t+1} = \beta_{i,4}\theta_{1,t} + \beta_{i,5}\theta_{2,t} + \beta_{i,6}\theta_{3,t}$$

Where $\hat{r}_{i,t+1}$ is a predicted return.

The R^2 coefficient is not a proper metrics to compare models as we do not use pure regressions, so R^2 no longer shows the percentage of explained variation. The models for different rolling windows are compared on the basis of Root Mean Squared Error (RMSE), Mean Absolute Difference (MAD), Directional Symmetry (DS), Correct Up Trend (CP) and Correct Down Trend (CD) where each metrics was calculated as follows:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n_m}\sum_{t=W+1}^{n_u}(r_{i,t} - \hat{r}_{i,t})^2}{(n_u - W)n_m}}$$

$$MAD = \frac{\sum_{i=1}^{n_m}\sum_{t=W+1}^{n_u}| r_{i,t} - \hat{r}_{i,t} |}{(n_u - W)n_m}$$

$$DS = \frac{100}{(n_u - W)n_m}\sum_{i=1}^{n_m}\sum_{t=W+1}^{n_u} d_{i,t} \text{ , where } d_i = \begin{cases} 1, & r_{i,t}\hat{r}_{i,t} \geq 0 \\ 0, & \text{otherwise} \end{cases}$$

$$CP = \sum_t \frac{100}{n_{UP,t}}\sum_i d_{i,t} \text{ , where } d_{i,t} = \begin{cases} 1, r_{i,t}\hat{r}_{i,t} \geq 0 \text{ and } r_i > 0 \\ 0, otherwise \end{cases}$$

$$CD = \sum_t \frac{100}{n_{DOWN,t}}\sum_i d_{i,t} \text{ , where } d_{i,t} = \begin{cases} 1, r_{i,t}\hat{r}_{i,t} \geq 0 \text{ and } r_i < 0 \\ 0, otherwise \end{cases}$$

W is the estimation window which equals 100 in our case as we compare all models for the same prediction time interval (101 – 244).

n_u – total number of time periods

n_m – total number of stocks.

n_{DOWN} – number of observations when stock price goes down

n_{UP} – number of observations when stock price goes up

Thus, the process of making predictions for the model with rolling window of 50 looked as follows. As we need to make prediction for the time period from 101 to 244, we estimate parameters beta and theta for the time interval 51 – 100. We randomly generate truncated matrices of state variables ($\theta_{tr} \in M(3,50)$) and factor loadings($\beta_{tr} \in M(50,3)$, where both were chosen to be distributed uniformly from -1 to 1.

We run Gradient Descent for the estimation period where we simultaneously change values of both truncated states and betas matrices for each iteration according to the following formulas:

$$x_k^{(i)} = x_k^{(i)} - \alpha\left[\sum_j (\theta^{(j)^T} x^{(i)} + \theta_{-1}^{(j)^T} x_{-1}^{(i)} - y^{(i,j)})\theta_k^{(j)}\right]$$

$$\theta_k^{(j)} = \theta_k^{(j)} - \alpha\left[\sum_i (\theta^{(j)^T} x^{(i)} + \theta^{(j-1)^T} x_{-1}^{(i)} - y^{(i,j)})x_k^{(i)}\right] - \alpha\left[\sum_i \theta^{(j+1)^T} x_{+1}^{(i)} + \theta^{(j)^T} x^{(i)} - y^{(i,j)})x_{k,-1}^{(i)}\right]$$

Where $x_{-1}^{(i)}$ stand for the vector of factor loadings for lagged state variables.

We didn't include regularization term $\lambda x_k^{(i)}$ for the first equation and $\lambda\theta_k^{(j)}$ for the second because the capacity of computer didn't allow finding optimal value of regularization coefficient, although we believe that its inclusion may strengthen the prediction power of our model by eliminating overfitting on the estimation data set.

We repeated the iterative process until the loss function converged $\frac{Loss_t - Loss_{t-1}}{Loss_{t-1}} \le 10^{-8}$. The prediction of future returns was made in a way $\hat{r}_{i,t+1} = \beta_{i,4}\theta_{1,t} + \beta_{i,5}\theta_{2,t} + \beta_{i,6}\theta_{3,t}$ for each stock I where $\beta_{i,4}, \beta_{i,5}, \beta_{i,6}$ are equivalent to $\beta_{1,-1}^{(i)}, \beta_{2,-1}^{(i)}, \beta_{3,-1}^{(i)}$ in the above notations. Then the rolling window was moved one step ahead and the process repeated starting from generation of random states and betas.

4.3 Robustness Evaluation

To make our evaluation robust we compared the performance of our model for different rolling windows with PCA and a benchmark case of random walk prediction. For the latter purpose a matrix of random returns RR was generated where $RR \in M(406,244)$ with each element distributed randomly $r_{i,t} \sim N(0,1)$. We apply Collaborative Filtering and PCA for prediction of random walk for the time interval 101 – 244 with prediction of actual returns so that the prediction intervals for each model are the same. The outcome is reflected in Tables 3 and 4 for actual returns predicted and Tables 5 and 6 for random walk series.

To make prediction on the basis of PCA model, we find principal components coefficients V^T and their representation in the principal component space $U\Sigma$, so that $V^T \in M(406,406)$ where column vectors represent loading for each share and $U\Sigma \in M(244,406)$ where each raw represents value of state variables for a certain time period. Then we run a regression within the rolling window of current returns on past state variables to assess factor loading for lagged state variables and make prediction for one period ahead where $\hat{r}_{i,t+1} = \beta_{i,4}\theta_{1,t} + \beta_{i,5}\theta_{2,t} + \beta_{i,6}\theta_{3,t}$ is a predicted future return.

Table 3

Performance metrics for actual returns, Collaborative Filtering

Actual returns	Metrics				
Rolling window size	RMSE*1000	MAD*1000	DS	CP	CD
10	45.3992	27.051	50.83%	52.38%	50.22%
50	23.0422	15.6928	51.13%	50.69%	52.57%
100	22.2784	15.3774	49.34%	49.89%	49.71%

Table 4

Performance metrics for actual returns, Principal Component Analysis

Actual returns	Metrics				
Rolling window size	RMSE*1000	MAD*1000	DS	CP	CD
10	28.4588	20.1475	50.40%	49.27%	52.53%
50	27.9196	19.8129	49.28%	49.45%	50.04%
100	27.6882	19.7168	50.25%	50.37%	51.08%

Table 5

Performance metrics for random walk, Collaborative Filtering

Random Walk	Metrics				
Rolling window size	RMSE*1000	MAD*1000	DS	CP	CD
10	1839.2	1124.3	49.40%	50.03%	50.22%
50	280.167	202.2774	49.20%	50.21%	53.46%
100	184.5219	136.2676	49.43%	49.74%	50.06%

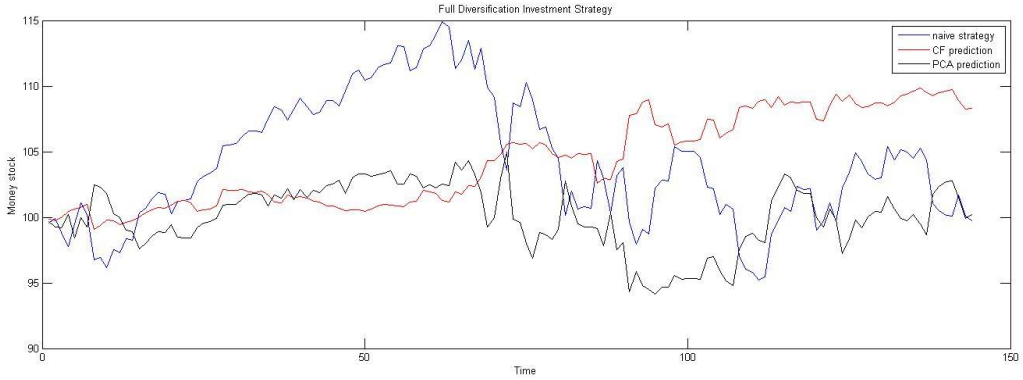
Table 6

Performance metrics for random walk, Principal Component Analysis

Random Walk	Metrics				
Rolling window size	RMSE*1000	MAD*1000	DS	CP	CD
10	1165.8	926.9145	50.22%	50.24%	50.21%
50	1045.8	833.8936	50.02%	49.94%	50.09%
100	1031.3	821.4674	50.18%	50.06%	50.30%

It is clearly seen that the model for actual returns is superior over the model for random walk series for both Collaborative Filtering and PCA as RMSE and MAD indicators are better for the actual returns in hundred times. The DS indicator equals approximately 50% for all of the models indicating inability to predict the direction of the market. The only exception is a CF model with rolling window of 50 where DS equals 51.13 %. Still there exists a pattern in the stock price movements since both CF and PCA significantly increase accuracy of prediction. The small DS value may be caused by a mean value of returns close to zero (4.6785e-04) so that even a slight deviation from the actual return may fail to predict the direction of stock movement. We see that a simple investment strategy of taking long position in stocks which are expected to rise and short position in stocks which are expected to fall against the naïve strategy of going long into all stocks provides return of 8.2749% for CF against 0.2355% for PCA and -0.2604% for naïve strategy where CF investment strategy never give large down movements in the value of portfolio (see Figure 7).

Figure 7



The reason why modeling returns for rolling window of 50 provides better results than for 10 or 100 rolling windows may lie in the stationarity of factor loadings for each stock. Obviously, the exposure of a stock to each of the factor changes in time because of the change in company's strategy such as business expansion, change in corporate policy, amount of debt, management etc. Hence, estimation period of 100 days may be too long for the loadings to be kept at the similar level. At the same time a period of 10 days may be too short for estimation because too many observations are lost and estimation of parameters is poor due to the low number of degrees of freedom.

So we provide DS Metrics for the investment strategy of going long in stocks that are predicted to rise by more than k percent and go short in stocks which are predicted to fall by more than k percent (see Figure 8). We call the k value a threshold return of investment. It is clearly seen that the low value of DS indicator was caused by a large number of stocks traded near zero returns as their elimination enhances Directional Symmetry to 54.5% at the level of threshold return of 0.8%. For higher levels of threshold DS indicator falls because of the loosening diversification effect of our portfolio. At the same time the prediction power of our model falls proving that high accuracy and low DS were caused by most returns fluctuating around zero mean (see Table 7). Figure 9 shows the return from investment strategies with different threshold returns. Obviously, the graph indicates that elimination of non-profitable stocks raises the prediction of the model providing higher profit.

Figure 8

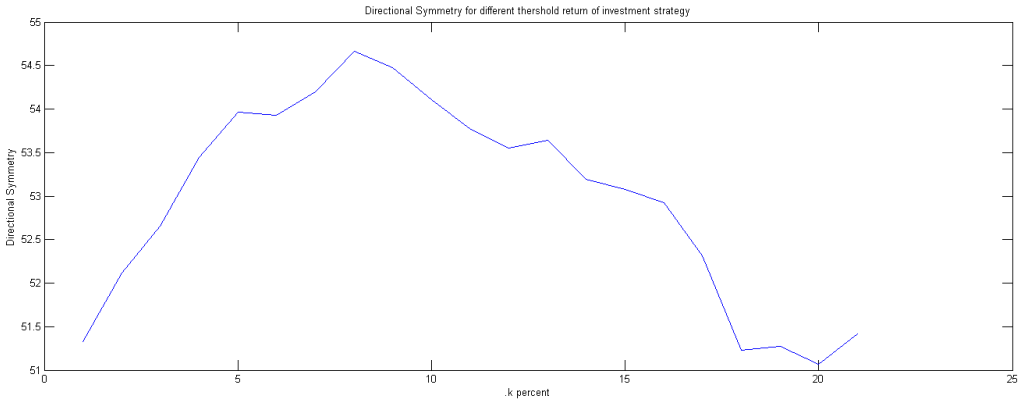
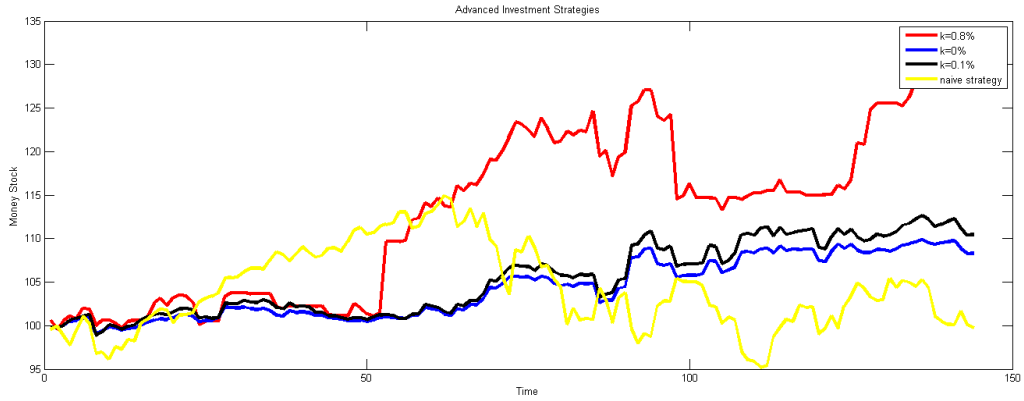


Table 7

Model performance for different threshold returns

Actual returns	Metrics		
Threshold .k	RMSE*1000	MAD*1000	DS
0.10%	24.868	16.8684	52.1%
0.50%	31.0555	21.467	53.9%
1.00%	41.8815	29.616	53.8%
1.50%	51.8567	38.1123	52.9%
2.00%	60.1004	46.0102	51.4%

Figure 9



4.4 Significance test

To formally test our model we use Wilcoxon signed-rank test. The test is parametric thus, it does not require any information on the distribution of the data. It is a widely used test for statistical comparison of the predictive power of two different models for a time series data and serves as an alternative for the paired Student's t-test and other numerous tests for time series (e.g. Diebold and Mariano (1995) or Pollock, Macaulay, Thomson and Onkal (2005)). Wilcoxon signed-rank test allows forecast errors to be non-zero mean, non-Gaussian, and

contemporaneously correlated. The null hypothesis of the test is that predictions produced by two different models are drawn from the continuous distributions with equal medians $H_0 : med(L^A(e_{i,t}) - L^B(e_{i,t})) = 0$ which is equivalent to testing that two models are identical.

The test statistics is calculated on the basis of differences of paired observations the absolute value of which are ranked from smallest to largest where the sign of each difference is assigned to the corresponding rank. Consider a loss function of the type $L(y_{i,t}, \hat{y}_{i,t}) = \mid y_{i,t} - \hat{y}_{i,t} \mid$. Denote it simply by $L(e_{i,t})$. For the two different prediction models A and B let the value of loss functions be $L^A(e_{i,t})$ and $L^B(e_{i,t})$ respectively. Denote $d_n \equiv L^A(e_{i,t}) - L^B(e_{i,t})$.

Let $I_+(d_n) = \begin{cases} 1, & \text{if } d_n > 0 \\ 0, & \text{Otherwise} \end{cases}$, where n is the length of our time series

then statistics $J = \sum_n rank(\mid d_n \mid) I_+(d_n)$ will be asymptotically (for the number of observations greater than 25) distributed as normal with mean $\mu_J = \frac{n(n+1)}{4}$ and $\sigma_J = \sqrt{\frac{n(n+1)(2n+1)}{24}}$

Hence, the test statistics may be calculated as follows:

$$Z = \frac{J - \mu_J}{\sigma_J} \overset{asympt}{\sim} N(0,1)$$

See Diebold and Mariano for a more detailed description of the test.

Application of the one-sided Wilcoxon test for each stock separately results in rejection of null hypothesis for all of the stocks. Hence, we claim that our model indeed reveals the latent patterns in stock price movements. However, since the fluctuation occurs near zero mean it is still difficult to construct the profitable trading strategy without filtering for low return stocks which is indicated by Directional Symmetry only slightly exceeding 50% (51.13%).

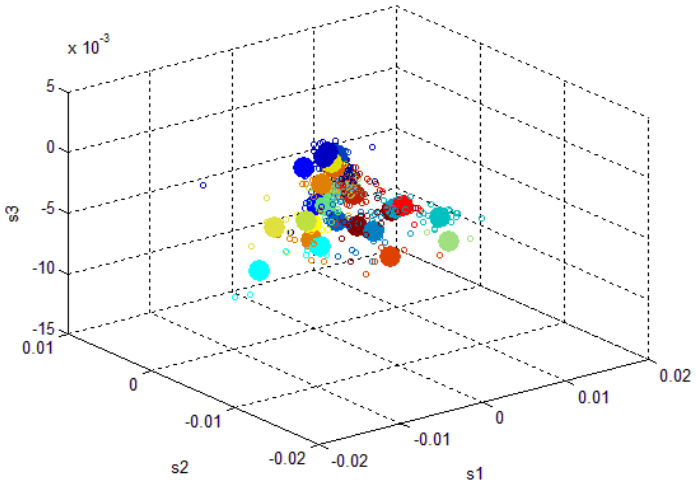
The same result of a test is obtained for the Principal Component Analysis model of stock price prediction. The p-values are around zero for all stock series. However, comparison of PCA and CF forecasts according to a one-sided Wilcoxon test shows that Collaborative Filtering outperforms PCA for 313 stocks out of 406 at the 5% significance level proving our idea that CF should be superior to PCA.

4.5 Additional evidence for Collaborative Filtering. Stock Clustering

It is a well-known fact that stocks from the same industry are subject to similar risks. An additional demonstration of the ability of Collaborative Filtering model to account for the latent factors in stock price determination is the clustering of stock loadings. Indeed, using Collaborative Filtering for the whole period of 244 trading days allows combining stocks into clusters on the basis of proximity among their factor loadings.

Figure 10

Clustering of stocks for CF with 3 state variables



Consider Figure 10.The figure shows clustering of stocks for Collaborative Filtering with three state variables. The clusters are chosen in a way to minimize the sum over all clusters of the within cluster distance. The distance is measured as the sum of absolute differences between clustering data and cluster centroid. It is clearly seen from Figure 10 that the loadings tend to cluster for some of the stocks and it happens that each cluster contains stocks which come mainly from the same industry (see Appendix 5 for the structure of each cluster). Therefore, Collaborative Filtering reveals that stocks from one industry have similar exposure to latent drivers of the market which is consistent with empirical findings. This may serve as additional evidence for the ability of CF to explain the market.

5. Conclusion

In our paper we investigated Collaborative Filtering technique in relation to time series data represented by the stocks of S&P500 Index for the year 2009. We considered the dimensionality reduction properties of CF with respect to other well-known techniques such as Principal Component Analysis and Maximum Likelihood Factor Analysis as well as its forecasting properties. We found out that Collaborative Filtering cannot outperform either of the models in dimensionality reduction exercise as it explains almost the same share of variation.

The forecasting properties of Collaborative Filtering appeared to be significantly better than that of PCA. The model allows revealing latent drivers of the stock market increasing prediction hundreds of times relative to the model applied to a simple random walk process. Still Collaborative Filtering with three state variables and one lag correctly predicts the movement of the market in only 51.13% cases. The reason for this is due to the distribution of returns around zero mean. The problem is solved by filtering near zero-return stocks. Even the slightest filtration of 0.1% enhances the prediction probability to 52.11% peaking at the level of 54.66% for 0.8% filtration level. We believe that the model will provide much better results for the other data sets as the period under consideration was subject to excess market volatility.

This paper is only a first step in the implementation process of Collaborative Filtering to stock market. The CF model may be further extended to include greater number of lags as well as cross-terms to account for non-linearity. Observed explanatory variables may be included into the model in addition to historical price information which should also increase in predictive power of Collaborative Filtering.

References

Abdi, H., Williams, L. J. (2010) Principal Component Analysis, Wiley Interdisciplinary Reviews: Computational Statistics 2

Avery, C., Chevalier, J., Zeckhauser, R. (2009). The "CAPS" Prediction System and Stock Market Returns, Faculty Research Working Paper Series

Back, A. D., Weigend, A. S. (1997). A First application of Independent Component Analysis to Extracting Structure from Stock Returns, International Journal of Neural Systems Vol. 8

Boudoukh, J., Richardson, M, Whitelaw, R. F. (2008). The Myth of Long-Horizon Predictability, Review of Financial Studies, 1577 – 1605

Carhart, M. M. (1997). On Persistence in Mutual Fund Performance, Journal of Finance Vol. 52, 57 – 82.

Chen, C. W., Huang, C. S., Lai, H. W. (2011). Data Snooping on Technical Analysis: Evidence from the Taiwan Stock Market, Review of Pacific Basin Financial Markets and Policies

Chen, N. F., Roll, R., Ross, S. A. (1986). Economic Forces and the Stock Market, The Journal of Business 59(3), 383 – 403.

Dai, Y. S., Lee W. M. (2011) The profitability of technical analysis in the Taiwan-U.S. forward foreign exchange market, Economics Bulletin 31(2)

Diebold, F.X., Mariano, R.S. (1995), Comparing Predictive Accuracy, Journal of Business and Economic Statistics 13, 253 – 256

Eisenhauer, J. G. (2003). Regression through Origin, Teaching Statistics 25

Ekstrand, M. D., Riedl, J. T., & Konstan, J. A. (2010). Collaborative filtering recommender systems. Foundations and Trends in Human-Computer Interaction, 4(2), 81-173

Fama E., French K., (1993). The Cross-section of Expected Stock Returns, Journal of Finance 33, 3 - 56

Fama, E. F., French, K. R. (1988). Dividend Yields and Expected Stock Returns, Journal of Financial Economics, 3 -25.

Ince, H., Trafalis, T. B. (2007). Kernel Principal Component Analysis and Support Vector Machines for Stock Price Prediction, IIE Transactions 39, 629 – 637

Jaditz, T., Riddick, L. A., Sayers, C.L. (1998). Multivariate nonlinear forecasting using financial information to forecast the real sector, Macroeconomic Dynamics 2, 369 – 382

Kung, J. J., Carverhill, A. P. (2012). A Bootstrap Analysis of the Nikkei 225, Journal of Economics integration 27(3), 487 - 504

Lathia, N. K. (2010). Evaluating Collaborative Filtering Over Time, Working Paper

Lauritzen, S. (2007). Latent Variable Models and Factor Analysis, online lectures, http://www.stats.ox.ac.uk/~steffen/teaching/fsmHT07/fsm607c.pdf

Lim, K. P., Brooks, R. (2011) The Evolution of Stock Market Efficiency over Time: a Survey of Empirical Literature, Journal of Economic Surveys 25(1), 69 – 108

Lim, T. J., Ngerng, M. H., (2012). Cross-section of Equity Returns Motivated by Fama and French, Procedia Economics and Finance 2, 284 – 291

Lo, A. W., MacKinlay, A. C. (1988). Stock market prices do not follow random walks: evidence from a simple specification test, The Review of Financial Studies, 1(1) , 41 – 66

Lu, C. J., Lee, T. S., Chiu, C. C. (2009). Financial time series forecasting using independent component analysis and support vector regression, Decision Support Systems 47, 115 – 125

Park, C. H., Irwin, S. H. (2007). What do we know about the profitability of technical analysis, Journal of Economic Surveys 21, 786 – 826.

Pollock, A. C., Macaulay, A., Thomson, M. E., Onkal, D. (2005), Performance evaluation of judgemental directional exchange rate predictions, International Journal of Forecasting 21, 473 – 489

Roberts, H. (1967). Statistical versus clinical prediction of the stock market, Unpublished manuscript

Roll, R., Ross, S. A. (1980). An empirical investigation of the Arbitrage Pricing Theory, The Journal of Finance 35(5), 1073 – 1103

Ross, S. A. (1976). The arbitrage theory of capital asset pricing, Journal of Economic Theory, 13(3), 341-360

Shlens, J. (2009). A Tutorial on Principal Component Analysis, Working Paper.

Timmermann, A., Granger, Clive W.J. (2004). Efficient Market Hypothesis and forecasting, International Journal of Forecasting 20, 15 – 27

Appendix 1 – Principal Component Analysis

Consider the method of estimating principal components using Singular Value Decomposition (SVD). Let Y be the matrix of returns where rows represent different shares and columns different time period - $Y \in M(406,244)$. Let $X = Y^T$ and $\{\hat{v}_1, \hat{v}_2, ..., \hat{v}_r\}$ be the set of orthonormal eigenvectors ($\hat{v}_i \in M(m,1)$) with associated eigenvalues $\{\lambda_1, \lambda_2, ..., \lambda_r\}$ for symmetric matrix XX^T:

$$(XX^T)\hat{v}_i = \lambda_i \hat{v}_i$$

Define $\sigma_i \equiv \sqrt{\lambda_i}$ and let $\{\hat{u}_1, \hat{u}_2, ..., \hat{u}_r\}$ be a set of vectors such that $\hat{u}_i \equiv \frac{1}{\sigma_i} X\hat{v}_i$ where $\hat{u}_i \in M(n,1)$. It happens that

$$\hat{u}_i \hat{u}_j = \begin{cases} 1 \text{if i} = \text{j} \\ 0 \text{otherwise} \end{cases}$$ and $|| X\hat{v}_i || = \sigma_i$.(see Appendix 4)

So, the scalar version of SVD is

$$X\hat{v}_i = \sigma_i \hat{u}_i$$

Which can be written in a scalar form

$$XV = U\Sigma$$

Where $V = [\hat{v}_1, \hat{v}_2, ..., \hat{v}_{\tilde{m}}]$

$U = [\hat{u}_1, \hat{u}_2, ..., \hat{u}_{\tilde{n}}]$

The matrices V and U are filled with additional m-r and n-r vectors respectively to deal with degeneracy issues.

$$\Sigma = \begin{bmatrix} \sigma_1 & 0 & ... & ... & ... & 0 \\ 0 & ... & 0 & ... & ... & ... \\ ... & 0 & \sigma_r & 0 & ... & ... \\ ... & ... & 0 & 0 & 0 & ... \\ ... & ... & ... & 0 & ... & 0 \\ 0 & ... & ... & ... & 0 & 0 \end{bmatrix}$$

As V is orthogonal, then $V^{-1} \equiv V^T$ and

$$X = U\Sigma V^T$$

So, any matrix X can be decomposed into an orthogonal matrix, a diagonal matrix, and another orthogonal matrix.

If we choose $Y = \frac{1}{\sqrt{n}} X^T$, then

$YY^T = (\frac{1}{\sqrt{n}} X^T)^T (\frac{1}{\sqrt{n}} X^T) = \frac{1}{n} XX^T = C_X$, where C_X is covariance matrix of X.

The eigenvectors of C_X are the principal components of X. Indeed, let $C_X \equiv \frac{1}{n} XX^T$ and

$C_Y \equiv \frac{1}{n} YY^T$. Then as $PX = Y$

$$C_Y = \frac{1}{n}(PX)(PX)^T$$
$$= \frac{1}{n} PXX^T P^T$$
$$= P(\frac{1}{n} XX^T)P^T$$
$$= PC_X P^T$$

As C_X is a square symmetric matrix, then it can be diagonalized by an orthogonal matrix of its eigenvectors. Therefore, $C_X = E^T DE$. Selecting P to be a matrix where each row p_i is an eigenvector of $\frac{1}{n} XX^T$ so that $P \equiv E^T$ and noting that in this case $P^{-1} = P^T$ we can write the following:

$$C_Y = PC_X P^T$$
$$= P(E^T DE)P^T$$
$$= P(P^T DP)P^T$$
$$= (PP^T)D(PP^T)$$
$$= (PP^{-1})D(PP^{-1})$$
$$= D$$

Indeed, when principal components correspond to eigenvectors of the covariance matrix for X $C_X = \frac{1}{n} XX^T$, covariance matrix of Y appear to be diagonal where each entry corresponds to

the variance of X along the newly chosen basis vector. Hence, the columns of matrix V are the principal components of X as V contains eigenvectors of YY^T.

Appendix 2 – Factor Analysis

Factor Analysis model (see Steffen Lauritzen (2007)).

Assume we have a set of observed factors X which may depend on a set of unobserved (latent) factors f. Then the model for factor analysis may be described as follows:

$$x_i = \sum_j \lambda_{ij} f_j + \varepsilon_i, \ \forall i$$

Where λ_{ij} is the j-th factor loading, f_j is the j-th common factor, ε_i is the error term for variable i. Then in matrix form

$$X = \Lambda F + \varepsilon,$$

Where $F \sim N(0, I)$ and $\varepsilon \sim N(0, \Psi)$, both independent of each other.

The idea of Linear Factor Analysis is to describe variation in observed variables by variation in the smaller number of latent variables. Therefore, the distribution of X can be considered as follows:

$X \sim N(0, \Sigma)$, where $\Sigma = \Lambda\Lambda^T + \Psi$.

It is then estimated by the maximum likelihood method. Let $S = \frac{1}{N}\sum_{n=1}^{N}(X_n - \bar{X})(X_n - \bar{X})^T$,

then the log-likelihood function for MLE is as follows:

$$\max_{\Sigma} \left[\log L(\Sigma) = -\frac{np}{2}\log(2\pi) - \frac{n}{2}\log\det(\Sigma) - \frac{n}{2}tr(\Sigma^{-1}S) \right]$$
$$\text{s.t. } \Sigma = \Lambda\Lambda^{\mathrm{T}} + \Psi$$

Solution to it is as follows:

$$\Psi = diag(S - \Lambda\Lambda^T)$$
$$S^* = \Psi^{-1/2} S \Psi^{-1/2}, \Lambda^* = \Psi^{-1/2}\Lambda$$

Thus, the columns of $\Lambda^* = (\lambda_1^*,...,\lambda_q^*)$ are eigenvectors of the q largest eigenvalues of S^*.

Denoting by Γ the diagonal matrix for which Γ_{ii} equals λ_i^*, then the following should hold:

$$\text{if } \Gamma_{ii} > 1, \text{then} S^*\Lambda^* = \Lambda^*\Gamma.$$

The algorithm proceeds as follows. It begins with an initial value of Ψ, finds q largest eigenvalues of matrix S^* - e_i^*. Let $\lambda_i^* = \theta_i e_i^*$ then we can solve $S^*\Lambda^* = \Lambda^*\Gamma$ for Λ^* and Λ. Using the matrix for factor loadings we can find new Ψ from $\Psi = diag(S - \Lambda\Lambda^T)$.

Appendix 3 - Support Vector Regression.

Support Vector Regression uses non-linearly mapping of inputs into a higher dimensional feature space (F) consequently correlating them linearly with outputs. If the usual regression model looks as $y_i = f(\vec{x}_i) + \delta_i$, where $f(x)$ is a linear function, SVR extends the function to the form of $f(x) = (v * \Phi(x)) + b$, where $\Phi(x)$ is a mapping function, v is a vector of weights so that $v * \Phi(x)$ is a dot product in space F and b is constant.

Appendix 4

For any arbitrary matrix X, $X \in M(m,n)$ the symmetric matrix $X^T X$ has a set of orthonormal eigenvectors $\{\hat{v}_1,...,\hat{v}_n\}$ and a set of associated eigenvalues $\{\hat{\lambda}_1,...,\hat{\lambda}_n\}$. The set of vectors $\{X\hat{v}_1,...,X\hat{v}_n\}$ then form an orthonormal basis, where $|| X\hat{v}_i ||= \sqrt{\lambda_i}$.

As $u_i u_j = \frac{1}{\sigma_i \sigma_j}(X\hat{v}_i)^T X\hat{v}_j = \frac{1}{\sigma_i \sigma_j}\hat{v}_i^T (X^T X)\hat{v}_j = \frac{1}{\sigma_i \sigma_j}\lambda_j \hat{v}_i^T \hat{v}_j$

As the set of eigenvectors of X is orthogonal, then $\hat{v}_i^T \hat{v}_j = \begin{cases} 1, i = j \\ 0, otherwise \end{cases}$

Hence, $u_i u_j = \begin{cases} 1, i = j \\ 0, otherwise \end{cases}$

Appendix 5 – Stock Clusters

Examples of Stock Clustering

Cluster num: 1 Num of shares: 16	Cluster num: 2 Num of shares: 10	Cluster num: 4 Num of shares: 15
AVY -- Avery Dennison Corp. -- Industrials	AMD -- Advanced Micro Devices -- Information Technology	AEE -- Ameren Corporation -- Utilities
CCL -- Carnival Corp. -- Consumer Discretionary	DOW -- Dow Chemical -- Materials	AYE -- Allegheny Energy -- Utilities
COH -- Coach, Inc. -- Consumer Discretionary	GCI -- Gannett Co. -- Consumer Discretionary	BDX -- Becton, Dickinson -- Health Care
DD -- Du Pont (E.I.) -- Materials	GT -- Goodyear Tire & Rubber -- Consumer Discretionary	D -- Dominion Resources -- Utilities
F -- Ford Motor -- Consumer Discretionary	HOG -- Harley-Davidson -- Consumer Discretionary	DGX -- Quest Diagnostics -- Health Care
FDX -- FedEx Corporation -- Industrials	HOT -- Starwood Hotels & Resorts -- Consumer Discretionary	EIX -- Edison Int'l -- Utilities
FLR -- Fluor Corp. (New) -- Industrials	JBL -- Jabil Circuit -- Information Technology	ETR -- Entergy Corp. -- Utilities
IGT -- International Game Technology -- Consumer Discretionary	JWN -- Nordstrom -- Consumer Discretionary	FE -- FirstEnergy Corp. -- Utilities
JCI -- Johnson Controls -- Consumer Discretionary	SNDK -- SanDisk Corporation -- Information Technology	FIS -- Fidelity National Information Services -- Information Technology
LTD -- Limited Brands, Inc. -- Consumer Discretionary	WYN -- Wyndham Worldwide -- Consumer Discretionary	LMT -- Lockheed Martin Corp. -- Industrials
R -- Ryder System -- Industrials		MRK -- Merck & Co. -- Health Care
ROK -- Rockwell Automation, Inc. -- Industrials	Cluster num: 3 Num of shares: 5	Q -- Qwest Communications Int -- Telecommunication Services
RRD -- Donnelley (R.R.) & Sons -- Industrials	KEY -- KeyCorp -- Financials	SLE -- Sara Lee Corp. -- Consumer Staples
SII -- Smith International -- Energy	MI -- Marshall & Ilsley Corp. -- Financials	SWY -- Safeway Inc. -- Consumer Staples
TIF -- Tiffany & Co. -- Consumer Discretionary	RF -- Regions Financial Corp. -- Financials	XOM -- Exxon Mobil Corp. -- Energy
WHR -- Whirlpool Corp. -- Consumer Discretionary	STI -- SunTrust Banks -- Financials	
	ZION -- Zions Bancorp -- Financials	
Cluster num: 5 Num of shares: 11	Cluster num: 6 Num of shares: 10	Cluster num: 7 Num of shares: 8
ADM -- Archer-Daniels-Midland -- Consumer Staples	AIZ -- Assurant Inc -- Financials	APOL -- Apollo Group -- Consumer Discretionary
AET -- Aetna Inc. -- Health Care	ALL -- Allstate Corp. -- Financials	AZO -- AutoZone Inc. -- Consumer Discretionary
AGN -- Allergan, Inc. -- Health Care	BBY -- Best Buy Co., Inc. -- Consumer Discretionary	CB -- Chubb Corp. -- Financials
CCE -- Coca-Cola Enterprises -- Consumer Staples	BIG -- Big Lots, Inc. -- Consumer Discretionary	HCBK -- Hudson City Bancorp -- Financials
CI -- CIGNA Corp. -- Health Care	CME -- Chicago Mercantile Exchange -- Financials	MHP -- McGraw-Hill -- Consumer Discretionary
DHR -- Danaher Corp. -- Industrials	FII -- Federated Investors Inc. -- Financials	TAP -- Molson Coors Brewing Company -- Consumer Staples
MOT -- Motorola Inc. -- Information Technology	LOW -- Lowe's Cos. -- Consumer Discretionary	TJX -- TJX Companies Inc. -- Consumer Discretionary
NI -- NiSource Inc. -- Utilities	MCO -- Moody's Corp -- Financials	TRV -- The Travelers Companies, Inc. -- Financials
SVU -- Supervalu Inc. -- Consumer Staples	MMC -- Marsh & McLennan -- Financials	
TE -- TECO Energy -- Utilities	PGR -- Progressive Corp. -- Financials	
WAT -- Waters Corporation -- Health Care		

Appendix 6 – Kernel Principal Component Analysis

kPCA is a non-linear extraction technique which maps data from input space to feature space where linear PCA is consequently used.

So, let $\Phi : \Re^n \to F$ be a non-linear mapping. Then for the input dataset $x = \{x_1,...,x_n\}$, we have a corresponding mapped dataset $\Phi = \{\Phi(x_1),...,\Phi(x_n)\}$. Consequent application of PCA diagonalizes an n-sample covariance matrix for the mapped dataset C.

$$C = \frac{1}{n}\sum_{i=1}^{n}\Phi(x_i)\Phi(x_i)^T$$

Hence, finding solution of the following eigenvalue problem:

$$\lambda V = \hat{C}V,$$
$$V \in F, \lambda \geq 0$$

Through the Singular Value Decomposition technique allows computing principal components for the projection matrix $\Phi(x)$ as was described in Appendix 1.