İSTANBUL TECHNICAL UNIVERSITY ★ INFORMATICS INSTITUTE

GENERAL PAIRS TRADING TECHNIQUES AND APPLICATION OF THE VASICEK MODEL USING GMM ESTIMATION

M.Sc. Thesis by
Sayad R. BARONYAN

Department: **Informatics Institute**

Programme : **Computational Science And Engineering**

SEPTEMBER 2009

<u>İSTANBUL TECHNICAL UNIVERSITY ★ INFORMATICS INSTITUTE</u>

GENERAL PAIRS TRADING TECHNIQUES AND APPLICATION OF THE VASICEK MODEL USING GMM ESTIMATION

M.Sc. Thesis by
Student Sayad R. BARONYAN
(702051010)

Date of submission : **28 August 2009**
Date of defense examination: **11 September 2009**

Supervisor (Chairman) :	**Yard. Doç. Dr. İlkay BODUROĞLU**
Members of the Examining Committee :	**Prof. Dr. Refik GÜLLÜ**
	Doç. Dr. Wolfgang HORMANN

SEPTEMBER 2009

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ BİLİŞİM ENSTİTÜSÜ

EŞLİ ALIM SATIM YÖNTEMLERİ VE GENELLEŞTİRİLMİŞ MOMENT YAKLAŞTIRIMI İLE VASICEK MODELİNİN UYGULANMASI

YÜKSEK LİSANS TEZİ

Student Sayad BARONYAN
(702051010)

Tezin Enstitüye Verildiği Tarih : 28 Ağustos 2009
Tezin Savunulduğu Tarih : 11 Eylül 2009

Tez Danışmanı : Yard. Doç. Dr. İlkay BODUROĞLU
Diğer Jüri Üyeleri : Prof. Dr. Refik GÜLLÜ
Doç. Dr. Wolfgang HORMANN

EYLÜL 2009

FOREWORD

I would like to express my deep appreciation and thanks for my advisor, my family and my friends. This work is supported by ITU Informatics Institute.

September 2009

Sayad Baronyan

Computational Science And Engineering

TABLE OF CONTENTS

Page

ABBREVIATIONS

GMM	**:** Generalized Method of Moments
S&P	**:** Standard and Poors
VIX	**:** Volatility Index
AR	**:** Auto Regression
VAR	**:** Vector Auto Regression
ADF	**:** Augmented Dickey Fuller
ECM	**:** Error Correction Model
B1B2	**:** Beta1, Beta2
MDM	**:** Minimum Distance Method
′	**:** Transpose
2STD	: 2 Standard Deviation Rule
MFR	: Market Factor Ratio
ADF-G	: ADF Test with Granger Causality

LIST OF TABLES

Page

LIST OF FIGURES

Page

GENERAL PAIRS TRADING TECHNIQUES AND APPLICATION OF THE VASICEK MODEL USING GMM ESTIMATION

SUMMARY

The valuation problem of the securities in the marketplace is a very hard process and it is not always accurate. Pairs trading comes with the idea, relative pricing, which if two securities have similar characteristics, they should have the same price. In our work, we firstly find the best pair of assets using the Augmented Dickey Fuller Test and Granger Causality sort method. We then use Vasicek Model, which is a mean reverting model, to construct our trading algorithm. We use GMM optimization to compute the optimal model parameters K*, theta, sigma* of the Vasicek model.

EŞLİ ALIM SATIM YÖNTEMLERİ VE GENELLEŞTİRİLMİŞ MOMENT YAKLAŞTIRIMI İLE VASICEK MODELİNİN UYGULANMASI

ÖZET

Piyasada alınıp satılan varlıkların fiyatlanması zor ve her zaman başarı ile sonuçlanmayan bir süreçtir. Eşli Alım Satım yöntemi; benzer özellikler içeren varlıkların göreceli fiyatlama fikrinden de yola çıkarak, aynı fiyata sahip olması gerektiğini önkoşul olarak kabul eder. Çalışmada en iyi çiftleri Genişletilmiş Dickey Fuller Test, ve Granger Causality Sıralama yöntemlerini kullanarak seçtikten sonra, ortalamaya geri dönüş özelliği içeren Vasicek Modeli'ni kullanarak alım satım yöntemi geliştirilmiştir. Vasicek Model parametreleri Genelleştirilmiş Moment Yaklaştırımı ile bulunmuştur.

1. INTRODUCTION

The idea behind pairs trading is an intuitive equity trading idea which identifies two equity issues that track each other closely and then looks for times when the issues fail to track one another. In its purest form, the pairs trading strategy involves different issues of the same company. The shares should behave identically, adjusted for the proportional difference in value. The idea would involve buying the shares in one market and selling the equivalent amount short in another market, and hedging the currency exposure. Pairs trading may also involve trading two different companies in the same industry. In this context the idea would be to buy one company and sell the other one.

Pairs traders can cross check the performance of a pair of stocks using a variety of statistical methods. The success of a pairs strategy depends on being able to identify which stocks to buy and which stocks to sell and also being able to identify when to buy and when to sell.

The first statistical pair trading was developed by Nunzio Tartaglia, the Morgan Stanley quant, who had lots of mathematicians and physicists in his group. The groups' aim was to develop automated trading systems and one of the techniques they used was pairs trading which involved trading securities in pairs. The process involved identifying pairs of securities that tend to move together. Whenever an anomaly was noticed, the pair would be traded with the idea that the anomaly would correct itself. After a successful implementation, when the group disbanded group members fell apart to other trading firms and diffused the technique: pairs trading.

The general idea in the marketplace from a valuation point is to sell overvalued securities and to buy the undervalued ones. But, calculating the true value of the security is a very difficult process and it is not always accurate. Pairs trading solves this problem by the relative pricing idea which is if two securities have similar characteristics; their price movements should be the same. If one of the securities gains more returns for a while it will get a relatively higher price than the other. Pairs trading involve selling the higher-priced security and buying the lower-priced security with the idea that the mispricing would correct itself.

The mutual mispricing between securities is called spread. The greater spread would trigger greater mispricing and greater potential of profit. Then, a long-short position is constructed. By taking one long and one short position in the market, strategy minimizes beta and therefore minimizes exposure to the market. Hence, the returns of the trade are uncorrelated to market returns, which imply pairs trading to be a market neutral strategy.

Although the idea of pairs trading straightforward, there are crucial points that an analyst should consider. A well constructed pairs trading strategy should have three main properties. 1) A good selection criterion to select the best profitable pairs 2) a good trading structure to maximize profit. 3) A good estimation technique to calibrate the trading strategy.

This thesis, tries to compare different methods for both selection and trading structures while proposing new models for each section and generating an application of GMM estimation.

The rest of this thesis is organized as follows; Chapter 1 introduces the main concept of pairs trading and introduces historical development of the idea, Chapter 2 introduces selection strategies in the literature and introduces new ideas, Chapter 3 introduces Trading Algorithms, Chapter 4 discusses estimation techniques, Chapter 5 introduces the algorithms and methodologies of selection and trading, Chapter 6 reports the empirical results of experiments with interpretations of results and Chapter 7 is a conclusion.

2. SELECTION STRATEGIES

Constructing a profitable trading strategy always starts with a comprehensible selection of investment options. There are mainly three types of analysis used by operators in order to effectively invest and trade on financial markets. These are:

Fundamental analysis: through the analysis of a firm's ratios and financial statements, the investor tries to assess a forecast of the future performance and thus a company's valuation. Even though this type of analysis might be effective, its results are strongly biased and affected by a structural lack of the information available.

Technical analysis: which is made by focusing on a security's past performance data, particularly price and volume.

Quantitative analysis: investment issues are faced trough a quantitative approach, using mathematical and statistical analysis.

In this thesis, four types of quantitative selection techniques will be illustrated: Minimum Distance Method, Augmented Dickey Fuller Test, Augmented Dickey Fuller Test in cooperation with Granger Causality and Market Factor Ratio test.

2.1 Minimum Distance Method

Gatev, Goetzmann and Rouwenhorst test the pairs trading strategy over the daily SP500 data through 1962 to 1997. They match stocks using the minimum distance method (also known as sum of squared deviations) and trade using the 2 standard deviation rule. They use Fama-French Factors to find an evidence of a higher excess return that pairs trading strategy generates comparing to the market return. They also use bootstrapping, to test their selection criteria.

The main idea of the selection criteria is to select pairs which have had the same historical state prices. According to Law of One Price theory, similar securities should have similar prices. To start the process, it is assumed that all the prices are same for a selected starting day. Then, the cumulative returns of the real prices are added to on the starting selected prices. By doing this, a cumulative return index is generated for all stocks. This process is also called normalizing the prices. To select pairs from this data set, sum of squared deviations is used. The formulation of the criteria is given as,

$$\gamma = \sum_{t=1}^{T} \left(\mathrm{P}_{1,\mathrm{t}} - \mathrm{P}_{2,\mathrm{t}}\right)^2 \qquad \textbf{(2.1)}$$

where $P_{1,t}$ and $P_{2,t}$ are two normalized stock prices for a selected time t. The smaller value of γ will give us the information that selected stocks has the same price changes in the selected period of time.

2.2 A New Selection Method using Market Factor Ratio

As explained before, by going one long and one short position in the market, pairs trading strategy is a market neutral strategy. This also means that we should find pairs that have similar market exposures. We propose a new model which checks the market exposures (β) of the securities and selects the best pairs. This can be implemented by regressing securities to their relevant market index.

Linear regression is a model that dependent variable, y_i is a linear combination of the linear *parameters.* Simple Linear Regression offers the model below for *N* data points where one independent variable is: x_i, and two parameters are, β_0 and β_1:

$$y_i = \beta_0 + \beta_1 x_1 + \varepsilon_i, \quad i = 1, \ldots., N \qquad \textbf{(2.2)}$$

As expected, this generates a straight line.

ε_i is the error term for the observation point *i*. From a time series sample, the parameters β_0 and β_1 can be estimated, and following equation can be generated.

$$y_i = \hat{\beta}_0 + \hat{\beta}_1 X_i + e_i \qquad \textbf{(2.3)}$$

e_i is the residual, $\varepsilon_i = y - \hat{y}$. For parameter estimation we use the well known method, ordinary least squares estimation. This method calculates the parameter estimates that minimize the sum of squared residuals, which is given as:

$$SSE = \sum_{i=1}^{N} e_i^2 \tag{2.4}$$

For simple regression, the least squares formula is,

$$\hat{\beta}_1 = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{(x_i - \bar{x})^2} \text{ and } \hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}$$

Where **x** is the mean (average) of the *x* values and y is the mean of the *y* values. In matrix notation, the equation can be written as

$$(X^T X)\hat{\beta} = X^T y \tag{2.5}$$

If we accept that the population error term has a constant variance, the estimate of that variance is given by:

$$\hat{\sigma}_\varepsilon^2 = \frac{SSE}{N-2} \tag{2.6}$$

This is called the root mean square error (RMSE) of the regression. The standard errors of the parameter estimates are given by

$$\hat{\sigma}_{\beta_0} = \hat{\sigma}_\varepsilon \sqrt{\frac{1}{N} + \frac{\bar{x}^2}{\sum(x_i - \bar{x})^2}} \tag{2.7}$$

$$\hat{\sigma}_{\beta_1} = \hat{\sigma}_\varepsilon \sqrt{\frac{1}{\sum(x_i - \bar{x})^2}} \tag{2.8}$$

If we accept that the error term is normally distributed, standard errors can be used to create hypothesis tests about the confidence of the parameters.

Our selection method tries to find the stocks with similar β ‘s. And by doing this we expect more hedge on market risk.

2.3 Augmented Dickey Fuller Test

In order to generate profit in a pairs trading strategy the spread or the ratio of the prices should have a constant mean and a constant volatility. In statistical analysis presence of these two properties are investigated by stationarity.

2.3.1 Stationarity

This property of time series which we will see is very important in the analysis for pairs trading and, more generally, in time series analysis. A time series $\{X_t\}$ is defined as weakly stationary if

$E(X_t^2) < \infty$

$E(X_t) = m$, m being a constant $\forall t$

$$\gamma_x(r,s) = \gamma_x(r+t, s+t) \quad \textbf{(2.9)}$$

The last condition says that, given $r, s \in T$, $CoV(X_r, X_s)$ is independent from t, being only a function of r and s. The intuition behind stationarity is quite simple. If a time series is stationary its probability distribution does not change between observations and this implies that parameters of the distribution (mean and variance for the Normal) do remain constant. On the other hand, a time series is non-stationary when it is affected by a trend or other periodic components.

Stationarity is a desirable property for mainly two reasons. The first is practical and is that many statistical tools and models are based on stationary time series and thus only work with them. The second reason is more theoretical and lies in the fact that stationary series have finite variance. This feature means that the series will never deviate from its mean value for more than a certain distance and hence suggests that the series is mean reverting, which is crucial for the implementation of pairs trading. The speed of mean reverting behavior is captured by the auto covariance function.

Now we will go over the most common types of time series, showing their main features and statistical properties.

A time series $\{X_t\}$ is defined a white noise if

$$E(X_t) = 0$$

$$E(X_t^2) = \sigma^2$$

$$\gamma_x(r,s) = \begin{cases} \sigma^2, r = s \\ 0, r \neq s \end{cases} \quad \textbf{(2.10)}$$

A White Noise is a sequence of drawings from a Normal distribution. The parameters of this Normal distribution are fixed and are not time-varying. Realizations are thus $i.i.d. \int N(0,\sigma^2)$ we also have that $E(X_t X_s) = 0$ $\forall t, s \in T / t \neq s$. This tells that the correlation between random variables is 0 at any time t, which implies that X_t, X_s are independent $\forall t, s \in T / t \neq s$.

Given a $\{\varepsilon_t\}$ series of $i.i.d.(0,\sigma^2)$ random variables, a time series $\{X_t\}$ is defined as a random walk if

$$X_t = \sum_{i=1}^{t} \varepsilon_i \quad \textbf{(2.11)}$$

A random walk is thus the sum of all the past white noise realizations up to the current time t. Each observation can be also thought as the last value plus the current white noise realization. To find out whether a random walk is stationary or not, let's consider this: by hypothesis $\{\varepsilon_t\}$ are $i.i.d.(0,\sigma^2)$, thus $V(\varepsilon_1) = V(\varepsilon_2) = ... = V(\varepsilon_t)$ and

$$V(X_t) = \sum_{i=1}^{t} V(\varepsilon_i) = tV(\varepsilon_i) \quad \textbf{(2.12)}$$

The variance is positively dependent on t and increases with time, this implying that the series might reach extreme values with the course of time. The analysis of the variance tells us that random walk is a non-stationary process which is not likely to be mean reverting.

2.3.2 Dealing with non stationarity

Having considered the most typical pattern followed by time series we now go through some more consideration about time series. Time series of security prices are most of the time non stationary series. Any non-stationary series can be seen as follows

$$X_t = m_t + S_t + Y_t \tag{2.13}$$

Where **X** is the original non stationary time series, **m** is a trend component, **S** is a seasonal component, **Y** represents a stationary time series with zero mean.

This decomposition is based on the fact that non stationary series can be seen as a stationary series plus a trend/seasonal component. See the non stationary series in such a way is very helpful. In fact, it shows an easy way to remove the trend component from the time series and transform it into a stationary one. This technique is known as differentiation and works in a very simple way. It just requires taking the time series realizations and subtracting to each of them the previous one. The differenced time series can be defined as

$$\nabla X_t = X_t - X_{t-1} \tag{2.14}$$

The concept of differentiating leads to the definition of integrated process: a non stationary time series X_t is said to be integrated of order n and is noted I(n) if it becomes stationary after differencing it at least n times.

It is good to note that differentiation allows removing not only constant trend but linear trends as well. Let's consider $X_t = m_t + S_t + Y_t$ with $m_t = \alpha_0 + \alpha_1 t$ and, for sake of simplicity set S=0.

$$\rightarrow X_t = \alpha_0 + \alpha_1 t + Y_t \tag{2.15}$$

By differencing this process we obtain

$$\nabla X_t = \alpha_0 + \alpha_1 t + Y_t - \alpha_0 - \alpha_1 (t-1) - Y_{t-1}$$

$$= \alpha_1 + Y_t - Y_{t-1}$$

Here we have a constant plus a stationary process. The differentiation thus removed the linear trend. It can be shown that differencing twice removes trend of a quadratic form as well. In any case attention must be paid not to difference too much as this will intensify the errors.

Before it has been assumed that the seasonal component is equal to 0. We can now remove this hypothesis thus having

$$X_t = m_t + S_t + Y_t \tag{2.16}$$

with $m_t = \alpha_0 + \alpha_1 t$ and $S_t = S_{t+d}$

$$\rightarrow X_t = \alpha_0 + \alpha_1 t + S_t + Y_t$$

And, differencing for a time lag d.

$$\nabla_d X_t = \alpha_0 + \alpha_1 t + Y_t + S_t - \alpha_0 - \alpha_1 (t-d) - Y_{t-d} - S_{t-d} \tag{2.17}$$

$$= \alpha_1 d + (Y_t - Y_{t-d})$$

∇_d is called lagged differencing operator and is simply defined as $\nabla_d X_t = X_t - X_{t-d}$

We just presented one very common method used to transform a non stationary series into a stationary one. Apart from differentiation there are also other ways to do this. We only cite a general class of transformations proposed by Box and Cox (1964). It looks as follows:

$$f(x) = \begin{cases} \dfrac{(x^{\lambda} - 1)}{\lambda}, \lambda > 0 \\ \log(x), \lambda = 0 \end{cases} \tag{2.18}$$

The log transformation is widely used in finance to obtain the series of the logarithmic prices which shows a number of desirable's properties.

2.3.3 Unit root testing

It has been already said that what we desire that a price time series is stationary. Unit root test serves to check for stationarity. Unit root test is indeed defined as a statistical test of the following form:

H_0: Time series is non stationary

H_1: Time series is stationary

For an autoregressive process AR(1) such as $X_t = \phi_0 + \phi_1 X_{t-1} + \varepsilon_t$ with $\phi_0 = 0$ (for simplicity reasons) the unit root test will be written as follows

H_0: $\phi_1 = 1$

H_1: $\phi_1 < 1$

The same can also be expressed in a slightly different way. Consider

$$\nabla X_t = X_t - X_{t-1}$$

$$\nabla X_t = \underbrace{(\phi_1 - 1)}_{P} X_{t-1} + \varepsilon_t \quad \textbf{(2.19)}$$

Which is the previous model with $\phi_0 = 0$ and the unit root test is now written as

H_0: P=0

H_1: P<0

Let's now try to investigate the logic behind this type of tests; we have to understand how the fact that $\phi_1 < 1$ implies the stationarity of the time series model. Let's consider again the AR(1) model

$$X_t = \phi X_{t-1} + \varepsilon_t$$

It can be shown that it is possible to rewrite it as follows:

$$X_t = \varepsilon_t + \phi\varepsilon_{t-1} + \phi^2\varepsilon_{t-2} + \dots$$

$\rightarrow$

$$X_t = \sum_{i=0}^{\infty} \phi^i \varepsilon_{t-i} \quad \textbf{(2.20)}$$

The notation of this last formula clearly shows that the errors ε (also known as shocks) affect the independent variable and this influence exponentially declines when $|\phi| < 1$. If, otherwise, $\phi = 1$, then there is a unit root and we have that

$$X_t = \sum_{i=0}^{\infty} \varepsilon_{t-i} \qquad \textbf{(2.21)}$$

In which case the shocks have a persistent effect and the dependent variable is fully determined by the sum of past and present shocks. Under the hypothesis that $X_0 = 0$ we have $\sum_{i=t}^{\infty} \varepsilon_{t-i} = 0$ and is thus possible to write X_t as the sum of the shocks from time 1 up to time t this is

$$X_t = \varepsilon_1 + \varepsilon_2 + \ldots + \varepsilon_t \qquad \textbf{(2.22)}$$

And the variance of X is

$$V(X_t) = t\sigma_\varepsilon^2 \qquad \textbf{(2.23)}$$

This is the key point, in fact it is easy to see that the variance of the process is a positive function of t; given that the variance is not constant but time varying, we have shown that the process is non stationary. On the other hand, when $|\phi| < 1$ the variance of the process becomes

$$V(X_t) = \frac{\sigma_\varepsilon^2}{1 - \phi^2} \qquad \textbf{(2.24)}$$

This is a constant, independent from t. We can hence say that the time series is stationary with $|\phi| < 1$.

As an additional argument, consider that the autoregressive model AR(1) under the hypothesis of unit root ($\phi = 1$) and non correlated, homoskedastic errors is actually a random walk, which we know is an example of non stationary process. The term $\sum_{i=1}^{t} \varepsilon_i = ST_t$ is called stochastic trend, as opposed to the linear deterministic trend which takes the form $DT_t = t$ where t=1,2,…,T. Whether the independent variable is dependent on a stochastic or on a deterministic trend is a relevant matter as it takes us back to the concept of integration, which we already introduced above. If a variable depends on a stochastic trend, this means that it is integrated of order one I(1). Otherwise the variable is integrated of order zero I(0) and hence weakly stationary.

The presence of unit root causes some problems in using OLS methodology and hence deserves some detailed consideration. In fact, in presence of unit root, the parameters associated with it show an asymptotic distribution which is very different from the Normal. This distribution is still centered on the correct value (1) but it is skewed on the right with more probability in the right tail. These characteristics cause some difficulties in the assessment of unit root presence. More precisely, if we set a unit root test as described above, to test $H_0 : \phi_1 = 1$ against $H_1 : \phi_1 < 1$, we can still calculate the t statistic as

$$t_{DF} = \frac{\left(\hat{\phi}_{ols} - 1\right)}{\hat{s}.e.\left(\hat{\phi}_{ols}\right)} \qquad \textbf{(2.25)}$$

but we must now be aware of the fact that its asymptotic distribution is not anymore the Student t, rather the t test distribution for unit root case is called Dickey-Fuller and the test is therefore also named Dickey-Fuller test. The t_{DF} above converges in law to

$$t \rightarrow \frac{\frac{1}{2}\left(W_1^2 - 1\right)}{\sqrt{\int_0^1 W_t^2 dt}} \qquad \textbf{(2.26)}$$

Where W is a Brownian Motion defined for $t \in [0;1]$. The Dickey-Fuller density function can be thus be computed only by using Monte Carlo simulation methodology. The critical value for the DF test is lower as it would be if we were using a Student-t distribution. Another peculiarity is that the distribution of the test changes according to the deterministic component of the model. Although Dickey-Fuller test is specific for the AR(1) case, there is also a way to test for unit root an AR(p) process. Such a generic model can be expressed by

$$X_t = \phi_1 X_{t-1} + \ldots + \phi_p X_{t-p} + \varepsilon_t \qquad \textbf{(2.27)}$$

Which is equivalent to $\Delta X_t = \gamma$, $\gamma = -\left(1 - \phi_1 - \phi_2 - \ldots - \phi_p\right)$

For a unit root to be there it is required that

$\gamma = 0$, i.e. $\phi_1 + \ldots + \phi_p = 1$

The test for $\gamma = 0$ is called augmented Dickey-Fuller test (t-ADF) and has the same distribution of the DF test. When executing this king of tests, it must be remembered that their power becomes very low when the relevant parameter ϕ is close to but not exactly one.

As for our case, a testing of the time series stationarity would be working as follows. Consider the differenced AR(1) process as previously described

$$\nabla X_t = \underbrace{(\phi_1 - 1)}_{P} X_{t-1} + \varepsilon_t \qquad \textbf{(2.28)}$$

and the DF test as

H_0: P=0, i.e. time series is non stationary

$H_{1:}$ P<0, i.e. time series is stationary

We now have to regress ∇X_t (that will be the dependent variable) against X_{t-1} (the independent variable) to achieve an estimation $\hat{P}$ of the parameter $P = \phi_1 - 1$. At this point we can calculate the DF t-stat and compare its value with the corresponding DF distribution value, which can be found on proper tables according to the chosen level of significance and the number of observations. If $t_{DF} < t_\alpha(n)$ we reject the null hypothesis, this implying that our time series is non stationary. It must be mentioned that, if the intercept of the regression is not null ($\phi_0 \neq 0$) then this fact advises on the existence of a linear trend distribution and the value $t_\alpha(n)$ becomes different. This could hence lead to a different result of the test. While, as we already said, financial time series are quite often integrated of order one I(1), to run a unit root test on series of higher order of integration the following approach has to be followed.

H_0: time series is I (2) or higher

H_1: time series is I (1)

Then regress $\nabla^2 X_t$ against ∇X_{t-1}, compute the t-stat and check for the result.

Augmented Dickey Fuller Test, which is a unit root test, serves to check for stationarity. Remember that,

$H_{0:}$ Time series is non stationary

$H_{1:}$ Time series is stationary

For an autoregressive process AR (1) such as $X_t = \phi_0 + \phi_1 X_{t-1} + \varepsilon_t$ with $\phi_0 = 0$ (for simplicity reasons) the unit root test will be written as follows

which the previous model with $\phi_0 = 0$ and the unit root test given as,

$H_{0:}$ P=0

$H_{1:}$ P<0

Note that the term augmented comes from the lagged values of the dependent variable. The number of lagged difference terms to include is determined empirically, the idea being to include enough terms so that the error term in tested equation is serially uncorrelated. The tau statistics will be used to determine passing pairs.

2.4 A New Selection Method : ADF Test with Granger Causality

Another important point is in ADF is that, ADF test is a Boolean test which has only two results, Pass or Fail. This means that we cannot sort the tau statistics as we did in the Minimum Distance Method or we will do in the Regression method. For selecting a portfolio of pairs or more generally to quantify the best pair one needs a sorting algorithm supporting ADF test. In this thesis, the pairs will be sorted using the Granger Causality Method.

Granger [9] approach to the question of whether x causes y is to see how much of the current y can be explained by past values of y and then to see whether adding lagged values of x can improve the explanation. y is said to be Granger-caused by x if x helps in the prediction of y, or equivalently if the coefficients on the lagged x's are statistically significant. Two–way causation is frequently the case; x Granger causes y and y Granger causes x.

The statement "x Granger causes y" does not imply that y is the effect or the result of x. Granger causality measures precedence and information content but does not by itself indicate causality in the more common use of the term.

To run a Granger Causality test, bi-variate regressions of the form should be run:

$$y_t = \alpha_0 + a_1 y_{t-1} + \cdots + a_l y_{t-l} + \beta_1 x_{t-1} + \cdots + \beta_l x_{t-l} + \varepsilon_t \quad \textbf{(2.29)}$$

$$x_t = \alpha_0 + a_1 x_{t-1} + \cdots + a_l x_{t-l} + \beta_1 y_{t-1} + \cdots + \beta_l y_{t-l} + \varepsilon_t \quad \textbf{(2.30)}$$

for all possible pairs of (x,y) series in the group. The reported *F*-statistics are the Wald statistics for the joint hypothesis:

$\beta_1 = \beta_2 = \cdots = \beta_l = 0$

for each equation. The null hypothesis is that x does *not* Granger-cause y in the first regression and that y does *not* Granger-cause x in the second regression.

We propose investigating pairs which does Granger cause each other. This is satisfied by checking F-Statistics of the Granger Causality test and taking the sum of the probabilities of the test.

3. TRADING ALGORITHMS

3.1 Two Standard Deviation Rule

As expressed in the Gatev, Goetzmann and Rouwenhorst's paper, traders in the industry generally use as the rule of thumb the two standard deviation rule on pairs trading. Looking from a mathematical finance perspective, this rule actually implements the mean reversion rule using a quantitative and probabilistic way. The main idea of pairs trading is to take open the trade when the spread or the ratio between a selected pair grows and hits a barrier and close it when it comes back to its mean. So one parameter to be found is actually this barrier which will point out an historical high.

As told before, two standard deviation is a kind of industry standard for this kind of historical high observations and not surprisingly it is used in pairs trading also. To get in detail of this rule, we should investigate what standard deviation concept explains.

In probability theory and statistics, standard deviation is a measure of the variability or dispersion of a data set, or a probability distribution. A low standard deviation indicates that the data points tend to be very close to the same value (the mean), while high standard deviation indicates that the data are "spread out" over a large range of values. Two standard deviation is about in the %98 of the confidence interval, which means that with a probability of 0.98 the point will be inside the 2 standard deviation barrier.

When the spread or ratio of pairs goes over %98 confidence interval, traders open their positions on the spread, expecting it will come back to its confidence interval and hopefully will reach back to its mean. It is sure that this rule of thumb may be optimized by using some profit optimization techniques and other quantitative tools.

3.2 Implementation of Vasicek Model to Pairs Trading

Mean reversion is a tendency for a stochastic process to remain near, or tend to return over time to a long-run average value. As a well know examples interest rates and implied volatilities can be given. In general stock prices tend not to have a mean reversion. In pairs trading, the ratio or the spread between pairs tend to have a mean reverting affinity.

Vasicek model [6] is generally used for interest rate modeling but it can be applied on other mean reverting processes as well. The model assumes that a mean reverting process has the stochastic differential equation in the form of:

$$dR_t = \kappa(\theta - R_t)dt + \sigma dW_t \qquad \textbf{(3.1)}$$

where W_t is a Wienner process modeling the random which models the continuous randomness of the system. The standard deviation parameter, σ determines the volatility of the mean reverting process and adjusts the randomness amplitude.

θ , long term mean level. All future trajectories of R will evolve around a mean level in the long run;

κ, speed of reversion. characterizes the velocity at which such trajectories will regroup around θ in time;

σ, instantaneous volatility, measures instant by instant the amplitude of randomness entering the system. Higher σ implies more randomness.

The following derived quantity is also of important to find,

$\frac{\sigma^2}{2\kappa}$, long term variance. All future values of R will come back to the long term mean with this variance after a period of time.

It should be noted that κ and σ tend to oppose each other: increasing σ increases the amount of randomness entering the system, but at the same time increasing κ amounts to increasing the speed at which the system will stabilize statistically around the long term mean θ with a corridor of variance determined also by κ. This is clear when looking at the long term variance,

$$\frac{\sigma^2}{2\kappa} \qquad \textbf{(3.2)}$$

which increases with σ but decreases with κ.

When we solve the stochastic differential equation we come to the result,

$$R(t) = R(0)e^{-\kappa t} + \theta(1 - e^{-\kappa t}) + \sigma e^{-\kappa t} \int_0^t e^{\kappa s}\, dW_s \qquad \textbf{(3.3)}$$

And the expected value or the mean as,

$$E[R_t] = R_0 e^{-\kappa t} + \theta(1 - e^{-\kappa t}) \qquad \textbf{(3.4)}$$

And the variance

$$\frac{\sigma^2}{2\kappa}(1 - e^{-2\kappa t}) \qquad \textbf{(3.5)}$$

In the limit t goes to infinity, we have

$$\lim_{t\to\infty} E[R_t] = \theta \qquad \textbf{(3.6)}$$

And

$$\lim_{t\to\infty} Var[R_t] = \frac{\sigma^2}{2\kappa} \qquad \textbf{(3.7)}$$

In this thesis, we use Vasicek model to calibrate the Pairs trading strategy. Perhaps the most important parameter σ , the conditional volatility is a measure of oscillation magnitudes. Conditional properties of σ help the analyst to calibrate it with the level of the ratios. A very high sigma can lead to a risky trading structure. θ is the long term mean which the ratio will converge and another important parameter κ calibrates the converging speed. A very high convergence parameter can lead to less trading oppurtinities and a very low one can lead to a more risky trading structure.

4. ESTIMATION TECHNIQUES FOR VASICEK MODEL

Calibrating the parameters of the Vasicek model is a hard process and it can be done by several methods. In this thesis, Generalized Method of Moments is selected for estimation of the parameters. This method does have adventage of not having an underlying distribution assumption, which is a very crucial point in dealing with spreads of the pairs.

4.1 Generalized Method of Moments

To explain the dynamic properties of the economic systems, statistical analysis and estimation procedures have crucial importance. Generalized method of moments (GMM) was first introduced into the econometrics literature by Lars Hansen in 1982 [7] . After the invention, the estimation procedure has become a ready to use, flexible tool of application to a large number of econometric and economic models. By relying on gentle and convincing assumptions, GMM has had a big impact on the theory and practice of econometrics. For the theory side, the main earning is that GMM provides a very general framework for considering issues of statistical consequence because it contains many estimators of interest in econometrics. For the practical side, unlike other methods like maximum likelihood process, it generates a computationally appropriate method of estimating nonlinear dynamic models without knowing the probability distribution of the data. Only specified moments derived from an underlying model are enough for GMM estimation. This property of GMM made itself very useful in areas like macroeconomics, finance, agricultural economics, environmental economics, and labor economics.

Consider the single linear equation model below,

$$\mathrm{y_t} = \mathrm{z}_\mathrm{t}'\delta_0 + \varepsilon_t \quad t = 1, \dots n \qquad \textbf{(4.1)}$$

Where ε_t are the error terms, z_t are the explanatory variables, a $L \times 1$ matrix, and may be correlated with ε_t, lastly δ_0 are the unknown coefficients. It is well known that if $E[z_t \varepsilon_t] \neq 0$ than z_t is contains endogenous variables and the estimator δ_0 will be biased and inconsistent. To overcome this, define x_t as a set of instrumental variables, a $K \times 1$ matrix which is orthogonal to set of ε_t, which means that x_t is not correlated with ε_t. Than we can write,

$$E[g_t(w_t, \delta_0)] = E[x_t \varepsilon_t] = E[x_t(\mathrm{y_t} - \mathrm{z_t'}\delta_0)] = 0 \tag{4.2}$$

Where $g_t(w_t, \delta_0) = x_t \varepsilon_t = x_t(\mathrm{y_t} - \mathrm{z_t'}\delta_0)$. From here we can write,

$$E[x_t \mathrm{y_t}] = E[x_t \mathrm{z_t'}]\delta_0 \tag{4.3}$$

Which generates a set of equations, with $E[x_t \mathrm{z_t'}]$ being a $K \times L$ matrix .To solve these equations $E[x_t \mathrm{z_t'}]$ matrix must be a full rank of L. There appears to be three cases between K and L.

The first case, where $K < L$, δ_0 is not identified, and we cannot find a solution to the equations. If $K = L$ then δ_0 is identified and an analytic solution can be found as,

$$\delta_0 = E[x_t \mathrm{z_t'}]^{-1} E[x_t \mathrm{y_t}] \tag{4.4}$$

And lastly where $K > L$, δ_0 is over identified.

In the model, the error terms are allowed to be serially correlated and conditionally heteroskedastic. For the case in which ε_t is conditionally heteroskedastic, it is assumed that $\{g_t\} = \{x_t\varepsilon_t\}$ is a

stationary and ergodic martingale difference sequence (MDS) satisfying

$$E[g_t g_t'] = E[x_t x_t' \varepsilon_t^2] = \mathbf{S} \tag{4.5}$$

Where **S** is a non singular $K \times K$ matrix, also the asymptotic variance-covariance matrix of the sample moments $\bar{\boldsymbol{g}} = n^{-1} \sum_{t=1}^{n} g_t(w_t, \delta_0)$. Using central limit theorem one can show that,

$$\sqrt{n}\,\bar{g} = 1\ \frac{1}{\sqrt{n}}\sum_{t=1}^{n} x_t \varepsilon_t \xrightarrow{d} N(0, \boldsymbol{S}) \quad \textbf{(4.6)}$$

$$\boldsymbol{S} = \sum_{j=-\infty}^{\infty} \Gamma_j = \Gamma_0 + \sum_{j=1}^{\infty} (\Gamma_j + \Gamma_j^{'}) \quad \textbf{(4.7)}$$

Where $\Gamma_j = E\left[g_t g_{t-j}^{'}\right] = E[x_t x_{t-j}^{'} \varepsilon_t \varepsilon_{t-j}]$.

GMM Estimator of δ is constructed using the orthogonality conditions. The idea is to create a set of moment conditions to estimate δ .

$$g_n(\delta) = \frac{1}{n}\sum_{t=1}^{n} x_t (y_t - z_t^{'}\delta) \quad \textbf{(4.8)}$$

$$= \begin{pmatrix} \frac{1}{n}\sum_{t=1}^{n} x_{1t}(y_t - z_t^{'}\delta) \\ \cdot \\ \cdot \\ \cdot \\ \frac{1}{n}\sum_{t=1}^{n} x_{Kt}(y_t - z_t^{'}\delta) \end{pmatrix} \quad \textbf{(4.9)}$$

The system has K linear equations and L unknowns. As told before, as if $K > L$ than there may not be a single unique solution to the system. Then we try to find the most possible solution, the value of δ that makes $S_{xy} - \delta S_{xz} = 0$. To do this we introduce a new matrix W, often called the weighting matrix , which is a $K \times K$ symmetric and positive definite weight matrix, than the GMM estimator is defined as,

$$\hat{\delta}(w) = \arg\min J(\delta, \hat{w}) \quad \textbf{(4.10)}$$

Where

$$J(\delta, \widehat{w}) = ng_n(\widehat{\delta})'\widehat{w}g_n(\delta) \qquad \textbf{(4.11)}$$

$$= n(S_{xy} - \delta S_{xz})'\widehat{w}(S_{xy} - \delta S_{xz}) \qquad \textbf{(4.12)}$$

The analytical solution to this problem can be found by setting $\left(\frac{dJ}{d\delta} = 0\right)$. And the solution appears as;

$$\hat{\delta}(w) = (S_{xz}'\widehat{w}S_{xz})^{-1}(S_{xz}'\widehat{w}S_{xy}) \qquad \textbf{(4.13)}$$

Under standard regularity conditions, it can be shown that,

$$\widehat{\boldsymbol{\delta}}(\widehat{W}) \xrightarrow{d} \delta_0$$

$$\sqrt{\boldsymbol{n}}(\widehat{\boldsymbol{\delta}}(\widehat{W}) - \delta_0) \xrightarrow{d} N(\mathbf{0}, \boldsymbol{cov}(\widehat{\boldsymbol{\delta}}(\widehat{W})) \qquad \textbf{(4.14)}$$

Where $\boldsymbol{cov}\left(\widehat{\boldsymbol{\delta}}(\widehat{W})\right) = (\Sigma_{xz}'W\Sigma_{xz}')^{-1}\Sigma_{xz}'WSW\Sigma_{xz}(\Sigma_{xz}'W\Sigma_{xz})^{-1}$ and the consistent estimate is,

$$\widehat{\boldsymbol{cov}}\left(\widehat{\boldsymbol{\delta}}(\widehat{W})\right) = \left(\Sigma_{xz}'\widehat{\boldsymbol{W}}\Sigma_{xz}'\right)^{-1}\Sigma_{xz}'\widehat{\boldsymbol{W}}\widehat{\boldsymbol{S}}\widehat{\boldsymbol{W}}\Sigma_{xz}\left(\Sigma_{xz}'\widehat{\boldsymbol{W}}\Sigma_{xz}\right)^{-1} \qquad \textbf{(4.15)}$$

Where $\widehat{\boldsymbol{S}}$ is the consistent estimate for $\boldsymbol{S} = \boldsymbol{cov}(\bar{g})$.

δ is defined by the positive semi definite matrix W so that the asymptotic variance of δ depends on W. It is crucial to choose a good W, unless the variance of δ's may be high. So to produce the possible smallest value of $\widehat{W}$, Hansen (1982) showed that $\widehat{W} = S^{-1}$ is a good choice. Here S is the long run variance as expressed earlier. With selecting $\widehat{W} = S^{-1}$ the variance becomes,

$$\boldsymbol{cov}(\bar{g}) = \left(\Sigma_{xz}'\widehat{\boldsymbol{S}}^{-\mathbf{1}}\Sigma_{xz}'\right)^{-1} \qquad \textbf{(4.16)}$$

From here the efficient GMM Estimator is defined as,

$$\delta(\widehat{\boldsymbol{S}}^{-\mathbf{1}}) = \arg\min n g_n(\delta)\widehat{\boldsymbol{S}}^{-\mathbf{1}}\, g_n(\delta) \quad \textbf{(4.17)}$$

As seen, the estimator needs a consistent estimate for **S,** the covariance matrix. However to find a consistent estimate of **S,** a consistent estimate of $\boldsymbol{\delta}$ is needed as explained below:

$$\hat{S} = \frac{1}{n}\sum_{t=1}^{n} x_t x_t^{'} \varepsilon_t^2 = \frac{1}{n}\sum_{t=1}^{n} x_t x_t^{'} (y_t - z_t^{'}\delta)^2 \quad \textbf{(4.18)}$$

The iterated efficient GMM estimation process is repeated until the unknown vector δ , do not change significantly. We know that,

$\delta(\boldsymbol{W}) = \arg\min n g_n(\delta)\boldsymbol{W}\, g_n(\delta)$ and $\boldsymbol{W} = \boldsymbol{S}^{-\mathbf{1}}$

We may choose $\boldsymbol{W} = \boldsymbol{I}$ and than S will become, $\hat{S} = \sum_{t=1}^{n} x_t x_t^{'} (y_t - z_t^{'}\delta(\boldsymbol{I}))^2$ as the start of the iteration. If we solve the equation and find the first **S** and set $\boldsymbol{W} = \boldsymbol{S}^{-\mathbf{1}}$ we find another $\boldsymbol{\delta}$ which will be the second iteration point to start.

If we repeat this process until a certain point that $\boldsymbol{\delta}$ does not change significantly anymore, we will find the optimal solution to our problem.

In some systems, the GMM moment conditions may be dependent on some nonlinear functions. In these cases, the optimal solution to the p model parameters θ, depending on the models moment conditions $g(w_t, \theta)$ should be satisfying the condition for $K \geq p$ nonlinear functions.

$$E[g(w_t, \theta_0)] = 0 \quad \textbf{(4.19)}$$

Adding a response variable y_t, $\boldsymbol{L}$ explanatory variables $\boldsymbol{z_t}$ and $\boldsymbol{K}$ instruments $\boldsymbol{x_t}$, the model may define a nonlinear error term,

$$a(y_t, z_t; \theta_0) = \varepsilon_t \quad \textbf{(4.20)}$$

Such that,

$$E[\varepsilon_t] = E[a(y_t, z_t; \theta_0)] = 0$$

Given that x_t is orthogonal to ε_t, define $g(w_t, \theta_o) = x_t\varepsilon_t = x_t a(y_t, z_t; \theta_0)$ so that,

$$E[g(w_t, \theta_0)] = E[x_t, \varepsilon_t] = E[x_t a(y_t, z_t; \theta_0)] = 0$$

Defines the GMM orthogonality conditions. These equations define also a system of K nonlinear equations in p unknowns. To find θ_0 following are needed:

$$E[g(w_t, \theta_0)] = 0$$

$$E[g(w_t, \theta_0)] \neq 0 \; for \; \theta \neq \theta_0$$

And the $K \times p$ matrix defined as;

$$\boldsymbol{G} = E\left[\frac{\partial g(w_t, \boldsymbol{\theta_0})}{\partial \boldsymbol{\theta}'}\right] \quad \textbf{(4.21)}$$

Has the full column rank p. The sample moment condition for an arbitrary **θ** is

$$g_{n(\theta)} = n^{-1}\sum_{t=1}^{n} g(w_t, \theta) \quad \textbf{(4.22)}$$

If K=p than the system is well identified and the GMM estimator becomes,

$$\hat{\boldsymbol{\theta}} = \boldsymbol{arg\,min\,J}(\boldsymbol{\theta}) \quad \textbf{(4.23)}$$

Where **J** is defined as,

$$\boldsymbol{J}(\boldsymbol{\theta}) = \boldsymbol{n}\boldsymbol{g_n}(\boldsymbol{\theta})'\boldsymbol{g_n}(\boldsymbol{\theta}) \quad \textbf{(4.24)}$$

If K>p, then $\boldsymbol{\theta_0}$ is over identified. As in the fist section, we again define a $K \times K$weighting matrix **W.** Than our GMM estimator becomes,

$$\hat{\boldsymbol{\theta}}(\widehat{\boldsymbol{W}}) = \boldsymbol{arg\,min\,J}(\boldsymbol{\theta}, \widehat{\boldsymbol{W}}) = \boldsymbol{n}\boldsymbol{g_n}(\boldsymbol{\theta})'\,\widehat{\boldsymbol{W}}\boldsymbol{g_n}(\boldsymbol{\theta}) \quad \textbf{(4.25)}$$

Again the efficient GMM estimator uses $\boldsymbol{W} = \boldsymbol{S^{-1}}$ **,** where S can be found by using the iterative techniques in section 1.

Hansen (1982) introduced the **J-statistic** to test behaviors and significance of the models suggested. Hansen refers to GMM objective function evaluated using an efficient GMM estimator:

$$J = J\big(\hat{\boldsymbol{\delta}}(\hat{\boldsymbol{S}}^{-1}), \hat{\boldsymbol{S}}^{-1}\big) = \boldsymbol{n}\boldsymbol{g}_{\boldsymbol{n}}(\hat{\boldsymbol{\delta}}(\hat{\boldsymbol{S}}^{-1})^{'}\hat{\boldsymbol{S}}^{-1}\boldsymbol{g}_{\boldsymbol{n}}(\hat{\boldsymbol{\delta}}(\hat{\boldsymbol{S}}^{-1})) \qquad \textbf{(4.26)}$$

Where $\hat{\boldsymbol{\delta}}(\hat{\boldsymbol{S}}^{-1})$ is an efficient GMM estimator of δ and $\hat{S}$ is a consistent estimate of S. If K=L than J=0, and if K> L than J >0. The larger value of J is an evidence of model misspecification. J-Statistic behaves like a chi-square random variable with degrees of freedom equaling the number of over identifying conditions.

4.2 Vasicek Model Estimation via GMM

To estimate parameters of the Vasicek model explained earlier, GMM estimation will be used. The trading algorithm will include the θ and σ parameters to take the decision on trades. [5]

We assume that pairs trading include mean reverting dynamics; modeling the ratio and the spread with this model and finding the parameters θ and σ will give us dynamic information on the behavior of the pairs.

GMM estimation will be used to estimate parameters dynamically, one reason we choose to use GMM estimation is that it does not assume any probability distributions. As explained earlier, GMM estimation takes the moment conditions into account. While dealing with daily data, the moment conditions can have a bad behavior; to avoid this we use weekly data.

Different from the two standard deviation rule our trading algorithm uses one standard deviation to open the positions and closes the positions when it hits back to its long term mean θ . This can be explained by the Vasicek Model's mean reverting behavior, as we estimate the parameters in a moving window, θ changes by the value κ and the time *t* exponentially. As expected, if we use 2 standard deviation rule in the Vasicek model, less trades will be opened.

5. APPLICATION METHODOLOGY

This section describes the methodology used for the analysis. Firstly, it introduces the training and testing periods used for the experiments, then it introduces the algorithms used for selection and trading.

5.1 Training and Testing Periods

We first define two consecutive periods as Training and Testing. Training period is a selected period of time where the parameters of the experiment are calculated and fixed. It can be seen as a preparation for the testing period. Immediately after the training period, the testing period follows which runs the experiments with tuned parameters.

In our analysis, we first select pairs and then take trading decisions using one step ahead forecasts of the parameters of an underlying model. To generate one step ahead forecast we need to specify a fixed moving window length.

Because of this, our training period needs to answer two questions.

1.1 What are the best pairs for trading?

1.2 What is the optimum window length?

We first select pairs with a selection algorithm and then calibrate the optimum window length. Note that, we scan the same training period two times, once for the selection and once for the window length optimization. Selection of the pairs is made by three different methods as explained earlier, Minimum Distance Method, Market Factor Ratio Selection and ADF Test with Granger Causality.

Window length optimization is actually a profit optimization. With the selected pairs, we trade with 24, 36, 48, 60 and 72 weeks of window length in training period. The most succesfull window length with the highest cumulative profit is selected as the optimum window length.

In our experiment, the dates between 01/01/2007 and 01/01/2008 is selected as training period. Following period, 01/01/2008 to 01/01/2009 is the training period.

5.2 Selection Methods

5.2.1 Minimum Distance Method

The first method for selecting pairs in training period is Minimum Distance Method. This method tries to catch up stocks with similar price movements. For this purpose, it first generates the cumulative price indexes for each series. It is a kind of normalization of the price series to generate a comparable number. For an individual stock, cumulative price index construction starts by setting up the start price of the training period to 100. Then for the next days, the return of the stock is multiplied by the previous cumulative price index member (100 for the second day) and adding this onto the previous cumulative price index member (100 again, for the second day).

After building two cumulative price indexes for the two stocks which are going to be analyzed, the squared difference between two series is summed up. For each possible pair in the selected space this routine is repeated. The sums of each analysis are sorted ascending, and the best pairs are selected from the top of the sorted list.

In our analysis, we select pairs from Dow Jones 30 index, and for $\frac{\binom{30}{2}}{2} = 435$ possibilities, we make the analysis. Note that, we do not start the analysis for the reversed pairs, as the results will be the same.

5.2.2 Market Factor Ratio Method

The second method tries to implement the idea of market risk hedging. It tries to find similiar betas for a selection of pairs. The more the betas are similiar, the more market risk hedging will be done by going one long and one short in the market. For this purpose, this method calculates betas for each pair in the selected space. Then it searches the ratios of betas which are close to one for each possible pair. The criteria below is generated for each possible pair and sorted ascendingly. Again, the best pairs are selected from the top of the sorted list.

$$MFR = Abs\ (\frac{\beta_1}{\beta_2} - 1) \quad \textbf{(5.1)}$$

In our analysis, we select pairs from Dow Jones 30 index, and for $\frac{\binom{30}{2}}{2} = 435$ possibilities, we make the analysis. Note that, we do not start the analysis for the reversed pairs, as the results will be the same.

5.2.3 ADF Test with Granger Causality

ADF Test and Granger Causality Test are explained earlier in Chapter 2. ADF test is a test for stationarity and Granger Causality Test is a test for is a technique for determining whether one time series is useful in forecasting another.

This method firstly searches for the pairs who passes the ADF test, if the ratio of a selected possible pair passes the ADF test on a selected confidence interval, then a Granger Causality test is implemented on this pair, and if in both two tests; $Stock^1$ does not Granger Cause $Stock^2$ and $Stock^2$ does not Granger Cause $Stock^1$ can not be rejected on a selected confidence interval, the pair is selected as a tradeable pair.

In our analysis, we select pairs from Dow Jones 30 index, and for $\frac{\binom{30}{2}}{2} = 435$ possibilities, we make the analysis with the confidence interval selected %90 for each tests (ADF Test and Granger Causality). Note that, we do not start the analysis for the reversed pairs, as the results will be the same.

5.3 Trading Methods

5.3.1 Two Standard Deviation Rule

The first trading algorithm we try is an industry standard for pairs trading. On a selected moving window length, the method calibrates one step ahead forecasts of the mean and the standard deviation parameters related to ratio of the selected pair. Long positions are opened for the ratio when the -2 standard deviation barrier is passed, claiming that the ratio of the pairs will go upward to its mean. Short positions are opened for the ratio when the +2 standard deviation barrier is passed, claiming that the ratio will go down to its mean. Long position are closed when the ratio comes under its mean, and the short positions are closed when the ratio comes below its mean.

The pseudo-code below describes the trading algorithm shortly;

```
Algorithm Two Standard Deviation (Optimum Window Length (OWL))

FOR I = Testing Period Start Date
               To I = Testing Period End Date
  // Step 1: Calibrate the Parameters
   MEAN[I+1] = MEAN(RATIOSERIES[I-OWL to I])
   STD[I+1]  = STD (RATIOSERIES[I-OWL to I])
 // Step 2: Check if any trades are possible
   IF (RATIO[I+1]> 2*STD[I+1])
        TRADE=SHORT;
   ELSE IF (RATIO[I+1]< -2*STD[I+1])
        TRADE=LONG;
   ELSE IF ( TRADE = LONG AND RATIO[I+1]< MEAN[I+1])
        TRADE = CLOSE;
   ELSE IF ( TRADE = SHORT AND RATIO[I+1]> MEAN[I+1])
        TRADE = CLOSE;
   END IF
END FOR
```

5.3.2 Vasicek Model, GMM Estimation and Pairs Trading

The second trading algorithm we try is a strategy where the parameters mean and standard deviation is calibrated using Vasicek Model. On a selected moving window length, the method calibrates one step ahead forecasts of the mean and the standard deviation parameters related to ratio of the selected pair. For parameter optimization of Vasicek Model, Generalized Method of Moments is used.

The trading decision are taken using the calibrated parameters. Same as two standard deviation rule, long positions are opened for the ratio when the -2 standard deviation barrier is passed, claiming that the ratio of the pairs will go upward to its mean. Short positions are opened for the ratio when the +2 standard deviation barrier is passed, claiming that the ratio will go down to its mean. Long position are closed when the ratio comes under its mean, and the short positions are closed when the ratio comes below its mean. The pseudo-code below describes the trading algorithm shortly;

```
Algorithm Vasicek Model Pairs Trading (Optimum Window Length
(OWL))
   FOR I = Testing Period Start Date
                To I = Testing Period End Date
  // Step 1: Calibrate the Parameters
   MEAN[I+1] = VASICEK CALIBRATION VIA GMM(RATIOSERIES[I-OWL to
I])
   STD[I+1]  = VASICEK CALIBRATION VIA GMM (RATIOSERIES[I-OWL to
I])
  // Step 2: Check if any trades are possible
   IF (RATIO[I+1]> 2*STD[I+1])
        TRADE=SHORT;
   ELSE IF (RATIO[I+1]< -2*STD[I+1])
        TRADE=LONG;
   ELSE IF ( TRADE = LONG AND RATIO[I+1]< MEAN[I+1])
        TRADE = CLOSE;
   ELSE IF ( TRADE = SHORT AND RATIO[I+1]> MEAN[I+1])
        TRADE = CLOSE;
   END IF
END FOR
```

5.3.3 GMM Estimation Algorithm

Vasicek model calibration is done using GMM estimation where an iterative GMM method is used. Iterative GMM technique is explained earlier in Chapter 4. The GMM algorithm is explained below:

First Step. Define β as the parameter vector: $\beta = [\mathrm{A\ B\ \sigma}]$. (In our trading implementation, long term mean is $\theta = \frac{-A}{B}$.) Take $W_1 = I$ (the identity matrix), and compute preliminary GMM estimate β_1 by using $\beta_0 = [\mathrm{A_0\ B_0\ \sigma_0}]$ where, $\mathrm{A_0}, \mathrm{B_0}$ and σ_0 are the starting values of estimation.

$$\beta_1 = \arg\min[(\sum_{i=1}^{N}\left(\frac{1}{T}\right)g(Y_i,\beta_0))'\boldsymbol{W_1}\sum_{i=1}^{N}\left(\frac{1}{T}\right)g(Y_i,\beta_0))] \qquad \textbf{(5.2)}$$

and for our Vasicek estimates g is defined as below :

$$g = \begin{pmatrix} e_1 = y - (A + Bx) \\ e_2 = e_1^2 - \sigma^2 \end{pmatrix} \qquad \textbf{(5.3)}$$

Where x is the ratio and y is the one lagged ratio. It can be seen that g takes the first 2 moments into consideration. This estimator is consistent for β_1, although probably not efficient.

Second Step. Take

$$W_2 = \left(\left(\frac{1}{T}\right) \sum_{i=1}^{N} g(Y_i, \beta_1) g(Y_t, \beta_1)' \right)^{-1} \quad \textbf{(5.4)}$$

where we have plugged our first-step preliminary estimate β_1.

This matrix converges in probability to $\Omega - 1$ and therefore if we compute θ with this weighting matrix, such estimator will be asymptotically efficient. And then calculate,

$$\beta_2 = \arg\min[(\sum_{i=1}^{N} \left(\frac{1}{T}\right) g(Y_i, \beta_1))' \boldsymbol{W_2} \sum_{i=1}^{N} \left(\frac{1}{T}\right) g(Y_i, \beta_1))] \quad \textbf{(5.5)}$$

Iterative Steps . Essentially the same procedure as second step GMM, only matrix $\hat{W}_T$ is recalculated several times. That is, estimate obtained in second step is used to calculate weighting matrix for step 3, and so on. For each minimization process we use *fmincon* function of MATLAB, which is a constrained nonlinear optimization routine. We use constrained optimization to lower the computation time and also to find rational parameters between a selected range. The lower constraints for the first and second moment is -10 and 0.001 the upper constraints are 10 for both moments.

Iteration ends these steps when parameter vector β and the corresponding J statistic does not change significantly. As explained earlier, J-stat is a chi-square distributed variable, and the confidence intervals of chi-square distribution can be used. The estimation code can be found in Appendix.

J stat:

$$\boldsymbol{J} = J(\beta(\boldsymbol{W}), \boldsymbol{W}) = \boldsymbol{N g_i}(\beta(\boldsymbol{W}))' \boldsymbol{W g_i}(\beta(\boldsymbol{W})) \quad \textbf{(5.6)}$$

6. EXPERIMENTS AND ANALYSIS

In our work we implemented two main pair's selection algorithms Minimum Distance Method and Augmented Dickey Fuller Test. Moreover, we tried two new methods Market Factor Ratio Selection Method and ADF test with Granger Causality as we explained earlier. After selecting the pairs, we focused on trading side, and using the portfolios selected by our testing algorithms, we tested our trading algorithms, the two standard deviation rule and the Vasicek Model.

6.1 Data and Coding Infrastructure

Using the DJ30 (Dow Jones 30) index components which are listed shortly below, we implement our selection algorithms between the dates 01/01/2007 and 31/12/2007 which we define as training period. The window length optimization is also made in this time frame. The testing period is between 01/01/2008 and 31/12/2008. Note that, after this dataset was constructed, the members of the DJ index was changed. On June 8, 2009, GM and Citigroup were replaced by The Travelers Companies and Cisco Systems, which became the third company traded on the NASDAQ to be part of the Dow.

The data was downloaded from Datastream with weekly (end of week) frequency. The analysis was made on MATLAB 7.1 . We also use MFE Toolbox for Granger Causality Tests. Codes can be found in the Appendix.

Table 6.1: List of Dow Jones 30 Members

Symbol	Industry	Company
MMM	Conglomerate	3M
AA	Aluminum	Alcoa
AXP	Consumer finance	American Express
T	Telecommunication	AT&T
BAC	Banking	Bank of America
BA	Aerospace and defense	Boeing

CAT	Construction and mining equipment	Caterpillar
CVX	Oil & gas	Chevron Corporation
C	Financial services	Citygroup
KO	Beverages	Coca-Cola
DD	Chemical industry	DuPont
XOM	Oil & gas	ExxonMobil
GE	Conglomerate	General Electric
HPQ	Technology	Hewlett-Packard
HD	Home improvement retailer	The Home Depot
INTC	Semiconductors	Intel
IBM	Computers and technology	IBM
JNJ	Pharmaceuticals	Johnson & Johnson
JPM	Banking	JPMorgan Chase
KFT	Food processing	Kraft Foods
MCD	Fast food	McDonald's
MRK	Pharmaceuticals	Merck
MSFT	Software	Microsoft
PFE	Pharmaceuticals	Pfizer
PG	Consumer goods	Procter & Gamble
GM	Automotive	General Motors
UTX	Conglomerate	United Technologies Corporation
VZ	Telecommunication	Verizon Communications
WMT	Retail	Wal-Mart
DIS	Broadcasting and entertainment	Walt Disney

6.2 Result of the Selection Methods

We select our pairs using three different methods as explained earlier. For the Minimum Distance Method (**MDM**) and for the Market Factor Ratio (**MFR**) Selection we select Top 5 pairs from their sorted list outputs and create a equally weighted portfolio. For ADF Test with Granger Causality (**ADF-G**), whole eight passing pairs are added to the portfolio.

Table 6.2: Selected Portfolio Members

Portfolio MDM		Portfolio MFR		Portfolio ADF-G	
MMM	IBM	WMT	JPM	BA	AXP
XOM	UTX	HPQ	DD	CAT	AA
DIS	DD	XOM	CVX	GE	CAT
VZ	HPQ	HD	DIS	MRK	MCD
INTC	HPQ	UTX	CAT	PFE	AA
				T	GE
				T	KFT
				T	MMM

We start analyzing these pairs by reporting their industries. As our goal is to increase the market risk and make profits using the idiosyncratic risk, we expect to find evidence that these pairs are from the same or related industries. Industries of the selected pairs are listed below.

Table 6.3: Industries of Portfolio MDM Members

Portfolio MDM	
Conglomerate	Computers and technology
Oil & gas	Conglomerate
Broadcasting and entertainment	Chemical industry
Telecommunication	Technology
Semiconductors	Technology

Table 6.4: Industries of Portfolio MFR Members

Portfolio MFR	
Retail	Banking
Technology	Chemical industry
Oil & gas	Oil & gas
Home improvement retailer	Broadcasting and entertainment
Conglomerate	Construction and mining equipment

Table 6.5: Industries of Portfolio ADF-G Members

Portfolio ADF-G	
Aerospace and defence	Consumer finance
Construction and mining equipment	Aluminium
Conglomerate	Construction and mining equipment
Pharmaceuticals	Fast food
Pharmaceuticals	Aluminum
Telecommunication	Conglomerate
Telecommunication	Food processing
Telecommunication	Conglomerate

When we examine these results we see that, all three selection criterias are selecting pairs from closely related industries.

For our improvement on ADF test we record that, % 15 of the possible total pairs are passed through ADF test but only %12 of them achieved succesfull results on Granger Causality test. The percentage of the total number of pairs who passed ADF and Granger Causality both was only %2 . From this results we see that in %85 of the selected pairs by ADF, stationarity of the series was reinforced with only one of the component of the pairs. However, in % 15 the stationarity was reinforced by both components of the pairs.

6.3 Result of the Trading Methods

After selecting the pairs in training period we run a profit based window length optimization for each trading algorithm as we discussed earlier in Chapter 5. The Results are listed as below.

Table 6.6: Optimum Window Lengths

Optimized Window Length For each Pair		MMM-IBM	XOM-UTX	DIS-DD	VZ-HPQ	INTC-HPQ			
	Pairs								
MDM Portfolio	2STD	48	72	36	48	24			
	V2STD	24	24	60	24	24			
		WMT-JPM	HPQ-DD	XOM-CVX	HD-DIS	UTX-CAT			
MFR Portfolio	2STD	36	24	24	36	72			
	V2STD	48	36	24	24	48			
		BA-AXP	CAT-AA	GE-CAT	MRK-MCD	PFE-AA	T-GE	T-KFT	T-MMM
ADF-G Portfolio	2STD	36	36	48	48	24	60	60	24
	V2STD	24	24	72	48	72	24	72	36

Then with this optimum window lengths we start trading with each of these portfolios. We use two different trading algorithms 2 Standard Deviation Rule (2STD) and Vasicek Model Calibrated 2 Standard Deviation Rule (V2STD) which are explained in Chapter 5. The cumulative profit of each portfolio for each algorithm is listed below.

Table 6.7: Trade Counts And Cumulative Profits

Selection Method	Trading Method	Trade Counts	Cumulative profits
MDM Portfolio	2STD	8	15,59517
	V2STD	13	21,48928
MFR Portfolio	2STD	6	5,717047
	V2STD	8	17,55449
ADF-G Portfolio	2STD	12	20,52247
	V2STD	**17**	**36,13609**

As we see from Table 6.7 ADF-G Portfolio with the V2STD trading method outperforms the other methods. The graph below shows the values of the equally weighted portfolios with different methods.

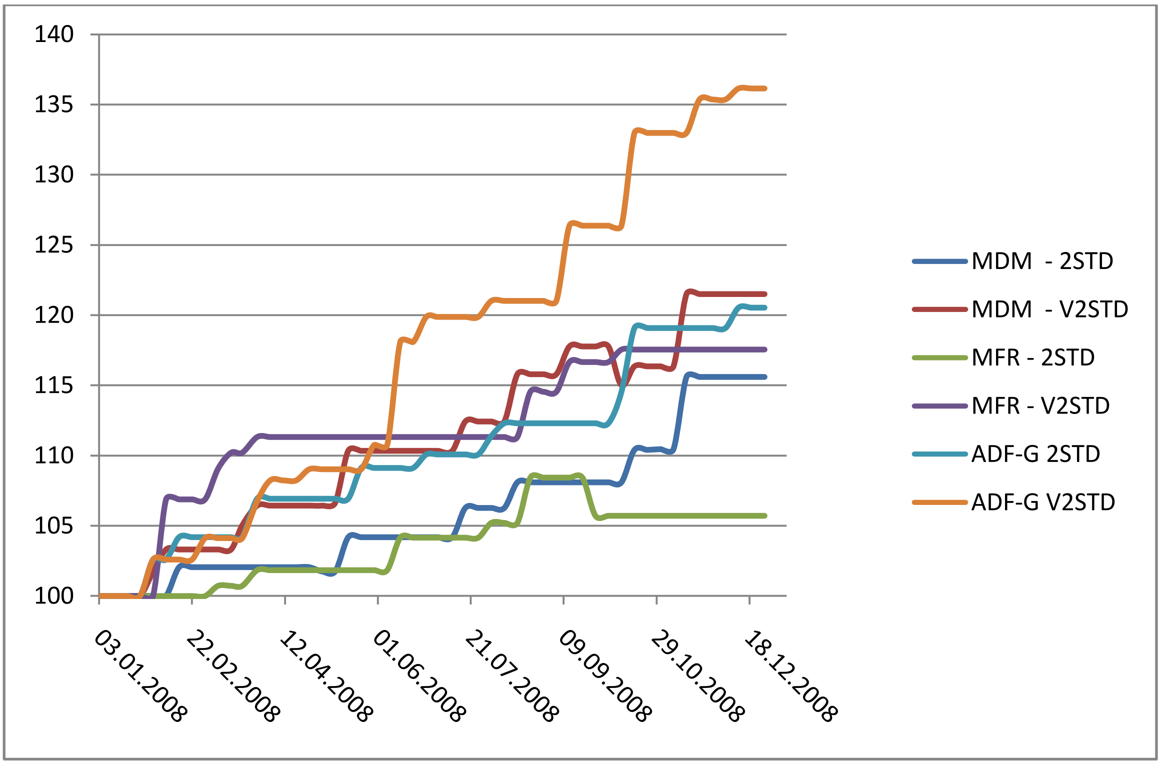

Figure 6.1 : The Cumulative Portfolio Changes over The Testing Period

In our analysis the best profitable selection method was the ADF-Granger Causality method. We see that, Market Factor Ratio Selection method is performing poorly comparing to other methods. Altough pairs trading stragies are hedging the market risk by going one long and one short position in the market, the profits of pairs are related to the idiosyncratic risks of the firms which Market Factor Ratio selection does not take into consideration. Industry standard MDM Selection is a good selection algorithm.It considers both market risk and idiosynratic risks of stocks by looking up directly to price movements of stocks. However, it cannot capture directly the long run relationship as ADF-Granger Causality Method does. In our analysis, we see that, ADF-Granger Causality Method finds out more strong relationships which are more likely to go on being pairs in time.

As a trading algorithm, 2 Standard Deviation rule is a well performing tool. However, Vasicek model calibrated 2 Standard Deviation rule is more attractive to changes in the parameters $\boldsymbol{\theta}$ and $\boldsymbol{\sigma}$. While Vasicek Model takes the mean reverting structure of pairs into consideration, it can generate a better estimate for the parameters. By using the advantages of its nature, Vasicek Model calibrated trading algorithm generates more trades and more profit in both selection methods.

Not suprisingly, the best Selection – Trading couple are the ADF- Granger Causality Selection and Vasicek Model Calibrated Trading algorithms. These two methods can be an alternative for the current industry standards in pairs trading.

Another point we should mention is market neutrality of the general pairs trading algorithms. As pairs trading strategies claim to be market neutral, we report the profit compared to the benchmark index DJ30.

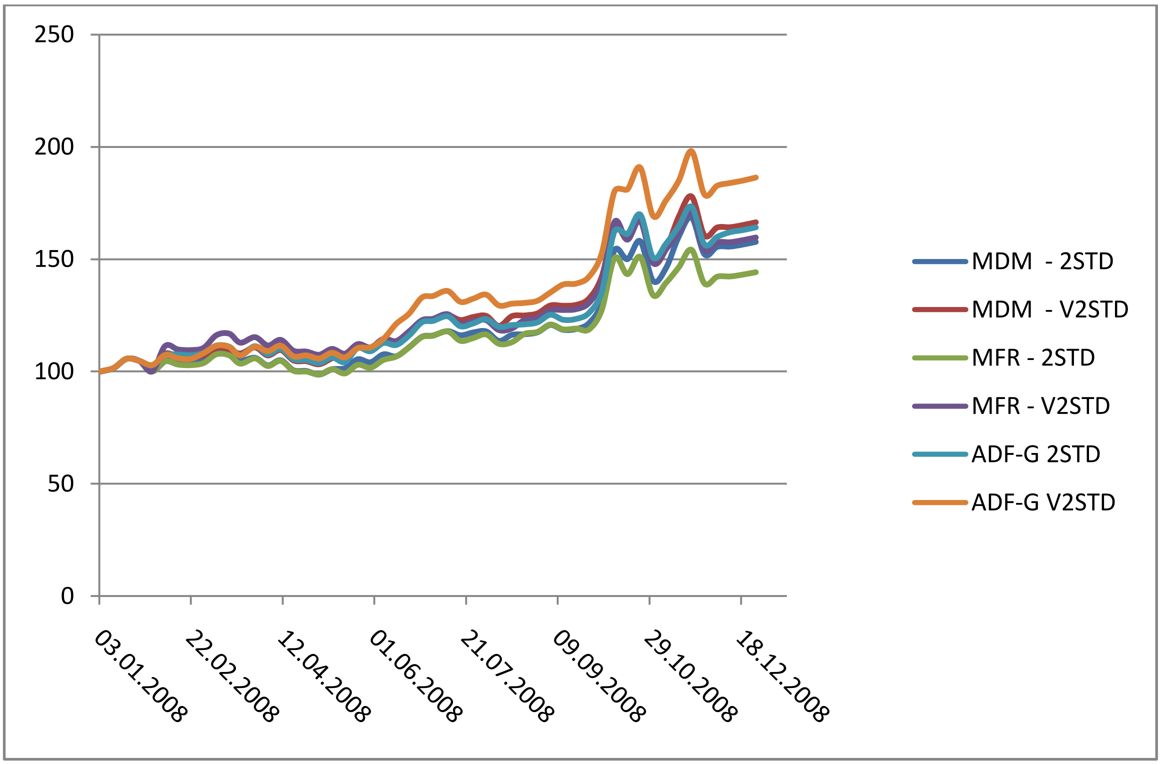

Figure 6.2 : The Cumulative Portfolio Changes Compared to Benchmark Index Dow Jones 30, in Testing Period

As we see from Figure 6.2, the profits compared to the Benchmark Index Dow Jones 30 are remarkably positive . This graph can be regarded as an evidence to the market neutrality of the pairs trading strategies. As the subprime mortgage crisis was still alive in the US markets in 2008, all of the pairs trading strategies kept on generating positive returns.

7. CONCLUSION

This thesis surveyed techniques and quantitative analysis employed in pairs trading. A quality pairs trading strategy should have a good selection criteria, a well defined trading algorithm and a good calibration technique. In this thesis we tried to implement minimum distance method, market factor ratio and ADF-Granger Causality tests as selection criterias. Our results show that minimum distance method is a good measure in selecting pairs, whereas ADF with Granger Causality makes more improvements as a selection criteria. Secondly, we compared 2 standard deviation rule with a mean reverting Vasicek trading model. we calibrated Vasicek using Generalized method of moments. We found GMM method as a succesfull, fast, easy to implement estimator for Vasicek model, and also Vasicek model implemented the trading idea succesfully.

REFERENCES

[1]**E.Gatev, WN Goetzmann, KG Rouwenhorst** Pairs trading: Performance of a relative-value arbitrage rule Review of Financial Studies, 2006 - Soc Financial Studies

[2] **R.J. Elliott, J Van Der Hoek, WP Malcolm** Pairs trading , Quantitative Finance, 2005

[3] **G. Vidyamurthy** Pairs trading: quantitative methods and analysis, 2004

[4] **P Nath** High Frequency Pairs Trading with US Treasury Securities: Risks and

Rewards for Hedge Funds, papers.ssrn.com , 2004

[5] **B. Do, R Faff, K Hamza** A new approach to modeling and estimation for pairs

trading, Proceedings of 2006 Financial Management Association, 2006

[6]**O. Vasicek** An equilibrium characterization of the term structure , Journal of financial economics, 1977

[7] **L.P. Hansen** Large sample properties of generalized method of moments estimators -

Econometrica: Journal of the Econometric Society, 1982

[8] **Stuart A. McCrary** Hedge Fund Course (Wiley Finance) , 2005

[9] **C.W.J Granger,**. "Investigating causal relations by econometric models and cross spectral methods".(Econometrica) ,1969

APPENDICES

APPENDIX A.1 : Matlab Codes for Minimum Distance Method

```
function [nameSet,TopInNumbers]=MinimumDistanceMethod(date1,date2)
close all;
clc;

conn=database('PairsTrading','','');
queryString='SELECT DOWJONES.[TICKER] FROM DOWJONES GROUP BY
DOWJONES.[TICKER]';
curs = exec(conn, queryString);
setdbprefs('DataReturnFormat','cellarray');
curs = fetch(curs);
nameSet = curs.Data;
for i=1:length(nameSet)
    for j=1:length(nameSet)
        time1=cputime;
        if(i>j)
            stock1=char(nameSet(i));
            stock2=char(nameSet(j));
            queryString=['select DATE,VALUE from DOWJONES where
TICKER= ''' stock1 ''' and DATE Between #' date1 '# and #' date2
'# order by DATE'];
            curs = exec(conn, queryString);
            setdbprefs('DataReturnFormat','cellarray');
            curs = fetch(curs);
            dataset = curs.Data;
            datesData= datenum(dataset(:,1),'yyyy-mm-dd
HH:MM:SS');
            stock1Data=cell2mat(dataset(:,2));
            queryString=['select DATE,VALUE from DOWJONES where
TICKER=''' stock2 ''' and DATE Between #' date1 '# and #' date2 '#
order by DATE'];
            curs = exec(conn, queryString);
            setdbprefs('DataReturnFormat','cellarray');
            curs = fetch(curs);
            dataset = curs.Data;
            stock2Data=cell2mat(dataset(:,2));
            returnStock1=(stock1Data(2:end)-stock1Data(1:end-
1))./stock1Data(1:end-1);
            returnStock2=(stock2Data(2:end)-stock2Data(1:end-
1))./stock2Data(1:end-1);

            normalizedStock1(1)=100;
            normalizedStock2(1)=100;
            for k=2:length(stock1Data);
            normalizedStock1(k)=normalizedStock1(k-
1)+returnStock1(k-1)*normalizedStock1(k-1);
```

```
            normalizedStock2(k)=normalizedStock2(k-
1)+returnStock2(k-1)*normalizedStock2(k-1);
            end
            sumSquaredDev(i,j)=sum((normalizedStock1-
normalizedStock2).^2);
            fprintf ([stock1 '\t' stock2 '\t'
num2str(sumSquaredDev(i,j))  '\tNS time:\t' num2str(cputime-time1)
'\n'])

        end
    end
end

close(conn);
u=1;
for i=1:length(nameSet)
    for j=1:length(nameSet)-1
    toBeSorted(u)=sumSquaredDev(i,j);
    sortMap(u)=cellstr(strcat(int2str(i),'-',int2str(j)));
    u=u+1;
    end
end
fprintf('To be Sorted Matrix ready. \n')


[ncol nrow] = size(toBeSorted);
u=1;

for i=1:nrow
    if(toBeSorted(i)~=0)
    nonZeroToBeSorted(u) = toBeSorted(i);
    nonZeroSortMap(u) = sortMap(i);
    u=u+1;
    end
end
fprintf('Non zero elements are found. \n')
[B,IX] = sort(nonZeroToBeSorted);
for i=1:20
Top(i) = nonZeroSortMap(IX(i));
TopInNumbers(i,:)=str2num(char(strrep(Top(i),'-',' ')));
end
fprintf('Sort Complete. \n')


end
```

APPENDIX A.2: Matlab Codes for Augmented Dickey Fuller Test

```
function [nameSet,TopInNumbers,counterTest]=ADFGranTest
(date1,date2)
close all;
clc;
addpath('MFE_Toolbox');
conn=database('PairsTrading','','');
queryString='SELECT DOWJONESTABLE.[TICKER] FROM DOWJONESTABLE
GROUP BY DOWJONESTABLE.[TICKER]';
curs = exec(conn, queryString);
setdbprefs('DataReturnFormat','cellarray');
curs = fetch(curs);
nameSet = curs.Data;
counterTest=0;
counterPassedADF=0;
generalCounter=0;
for i=1:length(nameSet)
    for j=1:length(nameSet)
        time1=cputime;
        if(i>j)
%             date1='01/01/2000';
%             date2='01/01/2009';
            % Open Connection-------------------------------------
---------------------

            %stock1=char(nameSet(i));
            %stock22=char(nameSet(j));
            generalCounter=generalCounter+1;
            stock1=char(nameSet(i));
            stock2=char(nameSet(j));
            queryString=['SELECT DATE,VALUE FROM DOWJONESTABLE
WHERE TICKER= ''' stock1 ''' AND DATE BETWEEN #' date1 '# AND #'
date2 '# ORDER BY DATE'];
            curs = exec(conn, queryString);
            setdbprefs('DataReturnFormat','cellarray');
            curs = fetch(curs);
            dataset = curs.Data;
            datesData1= datenum(dataset(:,1),'yyyy-mm-dd
HH:MM:SS');
            stock1Data=cell2mat(dataset(:,2));
            queryString=['SELECT DATE,VALUE FROM DOWJONESTABLE
WHERE TICKER= ''' stock2 ''' AND DATE BETWEEN #' date1 '# AND #'
date2 '# ORDER BY DATE'];
            curs = exec(conn, queryString);
            setdbprefs('DataReturnFormat','cellarray');
            curs = fetch(curs);
            dataset = curs.Data;
            datesData2= datenum(dataset(:,1),'yyyy-mm-dd
HH:MM:SS');
            stock2Data=cell2mat(dataset(:,2));
            fts11 = fints(datesData1,[stock1Data],{'stock1'});
            fts11=toweekly(fts11);

            fts21 = fints(datesData2,[stock2Data],{'stock2'});
            fts21=toweekly(fts21);
```

```
            ratio=fts2mat(fts11.stock1)./fts2mat(fts21.stock2);
            fts1=fints(fts11.dates,[fts2mat(fts11.stock1)
fts2mat(fts21.stock2) ratio],{'stock1','stock2','Ratio'});



            [adf, adfresid, df,
dfresid]=unitroot(fts2mat(fts1.Ratio));


            if (adf(3,4)==1)
                testadf(i,j)=50000;
                testadfgranger(i,j) = 50000;
                ratioSigmaMu(i,j)=std(ratio)/mean(ratio);
                %              fprintf ([stock1 '\t' stock2 '\t'
num2str(testadf(i,j))  '\tNS time:\t' num2str(cputime-time1)
'\n'])
                fprintf ([stock1 '\t' stock2 '\t'
num2str(testadfgranger(i,j))  '\tNS time:\t' num2str(cputime-
time1) '\n'])
            else
                counterPassedADF=counterPassedADF+1;
                testadf(i,j)=adf(2,4);
                ratioSigmaMu(i,j)=std(ratio)/mean(ratio);
                Y = [fts2mat(fts11.stock1)
fts2mat(fts21.stock2)];
                [STAT,PVAL]=grangercause(Y,0,1);
                if(PVAL(1,2)<0.1 && PVAL(2,1)<0.1)
                    testadfgranger(i,j)= 0.01;
                    counterTest=counterTest+1;
                else
                    testadfgranger(i,j) = 50000;
                end
                %              fprintf ([stock1 '\t' stock2 '\t'
num2str(testadf(i,j))  '\t' num2str(adf(3,4)) ' time:\t'
num2str(cputime-time1) '\n'])
                fprintf ([stock1 '\t' stock2 '\t'
num2str(testadfgranger(i,j))  '\t' num2str(PVAL(2,1)) '\t'
num2str(PVAL(1,2)) ' time:\t' num2str(cputime-time1) '\n'])
            end

        end
    end
end

close(conn);
u=1;
for i=1:length(nameSet)
    for j=1:length(nameSet)-1
%     toBeSorted(u)=testadf(i,j);
    toBeSorted(u)=testadfgranger(i,j);
    sortMap(u)=cellstr(strcat(int2str(i),'-',int2str(j)));
    u=u+1;
    end
end
fprintf('To be Sorted Matrix ready. \n')
```

```
[ncol nrow] = size(toBeSorted);
u=1;

for i=1:nrow
    if(toBeSorted(i)~=0)
    nonZeroToBeSorted(u) = toBeSorted(i);
    nonZeroSortMap(u) = sortMap(i);
    u=u+1;
    end
end
fprintf('Non zero elements are found. \n')
[B,IX] = sort(nonZeroToBeSorted);
if(counterTest==0)
    counterTest=5;
    fprintf('No Pair Could pass the test, providing ADF passers
instead.\n')
end
for i=1:counterTest
Top(i) = nonZeroSortMap(IX(i));
TopInNumbers(i,:)=str2num(char(strrep(Top(i),'-',' ')));
end
fprintf('Sort Complete. \n')


end
```

APPENDIX A.3: Matlab Codes for Market Factor Test

```
function [nameSet,TopInNumbers]=MarketFactorTest (date1,date2)
close all;
clc;

conn=database('PairsTrading','','');
queryString='SELECT StocksTableDowJones.[Ticker] FROM
StocksTableDowJones GROUP BY StocksTableDowJones.[Ticker]';
curs = exec(conn, queryString);
setdbprefs('DataReturnFormat','cellarray');
curs = fetch(curs);
nameSet = curs.Data;
for i=1:length(nameSet)
    for j=1:length(nameSet)
        time1=cputime;
        if(i>j)
            stock1=char(nameSet(i));
            stock2=char(nameSet(j));
            queryString=['select Date,Value from
StocksTableDowJones where Ticker= ''' stock1 ''' and Date Between
#' date1 '# and #' date2 '# order by date'];
            curs = exec(conn, queryString);
            setdbprefs('DataReturnFormat','cellarray');
            curs = fetch(curs);
            dataset = curs.Data;
            datesData= datenum(dataset(:,1),'yyyy-mm-dd
HH:MM:SS');
            stock1Data=cell2mat(dataset(:,2));
            queryString=['select Date,Value from
StocksTableDowJones where Ticker=''' stock2 ''' and Date Between
#' date1 '# and #' date2 '# order by date'];
            curs = exec(conn, queryString);
            setdbprefs('DataReturnFormat','cellarray');
            curs = fetch(curs);
            dataset = curs.Data;
            stock2Data=cell2mat(dataset(:,2));
            x=(1:length(stock1Data))';
            output1 =ols(stock1Data,x);
            output2 =ols(stock2Data,x);
            testols(i,j) = output1.beta/output2.beta;
            fprintf ([stock1 '\t' stock2 '\t'
num2str(testols(i,j))  '\tNS time:\t' num2str(cputime-time1)
'\n'])

        end
    end
end

close(conn);
u=1;
for i=1:length(nameSet)
    for j=1:length(nameSet)-1
    toBeSorted(u)=testols(i,j)-1;
    sortMap(u)=cellstr(strcat(int2str(i),'-',int2str(j)));
```

```
    u=u+1;
    end
end
fprintf('To be Sorted Matrix ready. \n')


[ncol nrow] = size(toBeSorted);
u=1;

for i=1:nrow
    if(toBeSorted(i)~=0)
    nonZeroToBeSorted(u) = toBeSorted(i);
    nonZeroSortMap(u) = sortMap(i);
    u=u+1;
    end
end
fprintf('Non zero elements are found. \n')
[B,IX] = sort(nonZeroToBeSorted);
for i=1:5
Top(i) = nonZeroSortMap(IX(i));
TopInNumbers(i,:)=str2num(char(strrep(Top(i),'-',' ')));
end
fprintf('Sort Complete. \n')


end
```

APPENDIX A.4: Matlab Codes for Vasicek GMM Estimation

```
function [ beta, stderr, covbeta, Qmin, test, ptest ] =
GMMSecond(MomFct,beta0,...
                A,b,Aeq,beq,lb,ub,nonlcon,options,...
                GMMLags,GMMiter,GMMtol1,GMMtol2,varargin);

% start with a call to the core function to get an idea of the
sample size involved
fs=feval(MomFct,beta0,varargin{:});
[T,r]=size(fs); %T=number of observations

k=size(beta0,1); %number of parameters
ddl=T-k; %degree of freedom

b0=beta0;
Omega=eye(r); %start with identity matrix
Qmino=100000; %something big
i=1;
while i <= GMMiter;
      betai=b0; % parameters we started with

      [beta,Qmin,exitflag,output,lambda,grad,hessian] =...

fmincon(@GMM_obj,b0,A,b,Aeq,beq,lb,ub,nonlcon,options,Omega,MomFct
,varargin{:});
      disp('parameters after optimization');
      beta

      if ((max(abs(beta-b0))<GMMtol1) | (abs(Qmin-Qmino)<GMMtol2))
%complete with precision over J
         fprintf('the parameters before and after  loop are');
         [betai beta ]
          break;
      end

      H  = feval(MomFct,beta,varargin{:});
      mH = mean(H);
      H  = H - kron(mH,ones(size(H,1),1));
      Omega = (H'*H)/T;

      for j=1:GMMLags
        % size(H(j+1:end,:)')
        % size(H(1:end-j))
          Gamma = ( H(j+1:end,:)'*H(1:end-j,:)) /T;
          Omega = Omega + (1-j/(GMMLags+1)) * (Gamma + Gamma');
%Parzen type weighting
      end

      b0=beta;
      fprintf('the weighting matrix is \n');
      Omega
      fprintf('done with GMM iteration %5.0f\n',i);
      Qmino=Qmin;
```

```
      i=i+1;
end

D=gradp(@GMM_momgen,beta,MomFct,varargin{:});

if cond(Omega)>100000
    invOmega=pinv(Omega); %Moore-Penrose inverse
else
    invOmega=Omega\eye(size(Omega,1));
end

DiOmegaD=D'*invOmega*D;
covbeta=((DiOmegaD)\eye(size(DiOmegaD,1)))/T;
stderr=sqrt(diag(covbeta));
corbeta=covbeta./kron(stderr,stderr');
tstudent=beta./stderr;
pvalue=2*(1-tcdf(abs(tstudent),ddl));

r=size(Omega,1);
k=size(DiOmegaD,1);
if r>k
    test=T*Qmin;
    ptest=chi2cdf(test,r-k);
else
    test=[];
    ptest=[];
end

name=1:size(beta,1);
name=name';
omat=[ name beta stderr tstudent pvalue];
fprintf('\n');
fprintf('Number of observations :
.....................%12.4f\n',T);
fprintf('Number of parameters :
.....................%12.4f\n',k);
fprintf('Number of degrees of freedom :
...............%12.4f\n',ddl);
fprintf('Number of orthogonalit conditions
:...........%12.4f\n',r);
fprintf('Value of the objective function : ............%12.4f\n',
Qmin);
fprintf('\nTest of overidentification of restrictions :..%12.4f
\n',test);
fprintf('\nCorresponding marginal probability : .........%12.4f
\n',ptest);
fprintf('\n');
fprintf('\n');
fprintf(' Parameter  Estimate      Standard Error   Student-t
Signif.\n');
fprintf('-------------------------------------------------------
-----------\n');
for j=1:size(omat,1);
      fprintf('%7.0f %14.6f %14.6f %14.6f %14.6f \n',omat(j,:));
end
fprintf('\n');

fprintf('correlation matrix of parameters\n');
for j=1:size(corbeta,1);
      fmt1='%8.4f';
```

```
      fmt=kron(ones(1,size(corbeta,1)),fmt1);
      fprintf([fmt '\n'],corbeta(j,:));
end

%-----------------------------------------------------------------

function g=gradp(f,x0,varargin)
% computes the gradient of f evaluated at x
% uses forward gradients. Adjusts for possible differently scaled
x by taking percentage increments
% this function is the equivalent to the gradp function of Gauss
% f should return either a scalar or a column vector
% x0 should be a column vector of parameters
f0=feval(f,x0,varargin{:});
[T,c]=size(f0);

if size(x0,2)>size(x0,1)
    x0=x0';
end

k=size(x0,1); % number of parameters wrt which one should compute
gradient

h=0.000000000001; %some small number

g=zeros(T,k); %will containt the gradient
e=eye(k);
for j=1:k;

    f1=feval(f,(x0.*( ones(k,1) +  e(:,j) *h )),varargin{:});
    g(:,j)=(f1-f0)/(x0(j)*h);

end

%-----------------------------------------------------------------

function GT=GMM_momgen(beta,fnam,varargin);
% computes the average over momentized observations. Corresponds
% to computing gT.
% imports the name of the function, the parameters and the
observations

mt=feval(fnam,beta,varargin{:});
GT=mean(mt)';

%-----------------------------------------------------------------

function J=GMM_obj(beta,Omega,fnam,varargin);
% computes the value of the objective function.
% imports the name of the function, the parameters and the
observations

GT = feval(@GMM_momgen,beta,fnam,varargin{:});
if cond(Omega)>100000
    invOmega=pinv(Omega); %Moore-Penrose inverse
else
    invOmega=Omega\eye(size(Omega,1));
end
J=GT'*invOmega*GT;
```

```
%-----------------------------------------------------------------

function mt=tobeoptimized(beta,rt)
% beta[1)=mean
% beta(2)=standard deviation (not !! variance)
T=size(rt,1);
mt = zeros(T,3);
X=rt(2:end);
Z=[ones(T-1) X];
y = X -rt(1:end-1);
alpha  = beta(1);
betae = beta(2);
sigsq  = beta(3);
gamma=0;
e1 = y - (alpha + betae*X)/52;
e2 = e1.^2 - (sigsq*X.^(2*gamma))/52;
e = [e1 e2];

mt = e;
```

CURRICULUM VITA

Candidate's full name:	**Sayad Reteos Baronyan**
Place and date of birth:	**İstanbul Şişli, 05/11/1983**
Permanent Address:	**Moda Cad. Neşe Sok. Yonca Apt, No:27 D:4 Kadıköy/İstanbul**
Universities and Colleges attended:	**İstanbul Technical University – Physics Engineering (2005)**
Publications:	**S.R.Baronyan, İ.Boduroğlu,E.Şener,** Empirical Evidence that Market-Neutral Trading Performs Best Under Most Severe Market Conditions, 2009 – The Manchester Review