A Simple Application of Pairs Trading Strategy

Qingmin Shi

May 10, 2016

1. Introduction

The basic idea of pairs trading is to find two similar financial assets which trade at some spread. If there exists equilibrium between two assets and an anomaly is observed in the relationship, one can seek to profit from the comparative mispricing by selling the relative overvalued asset and simultaneously buying the undervalued asset.

The main theoretical foundation of my project is cointegration. Specifically, suppose we have asset X and Y, we need to find weights a and b, such that the linear combination of X and Y is a stationary noise process. That is to say, we need to make $aX + bY = \epsilon$, where ϵ is a stationary noise process.

This paper explored several different approaches for cointegration. It is natural to think about linear regression, i.e., $Y = aX + \epsilon$. However, the diagnostic plot indicate that the residuals are not stationary. We also tried AR models and got better performance. Finally, we decide on the Johansen procedure, which uses VAR (vector autoregressive) models instead of regression. Using this method, we can obtain a stationary noise process.

After getting the stationary noise process, we can begin to trade the spread. Firstly, we can normalize the process $aX + bY$. Then, we observe the process. When its value is below 1.5 standard error, we build a long position. Once its value is within 0.3 standard error, we cover the long position we built. When its value is above 1.5 standard error, we build a short position. Once its value is within 0.3 standard error, we cover the short position we built.

The above is the main framework of the project. We can see all the details in the following contents.

2. Data Description

Our choice of the stock pair is AT&T and Verizon, starting from May 1st in 2008, to April 1st in 2011. We use the daily adjusted closing price, so we have totally 737 prices for each stock. Denote log price of AT&T as X, log price of Verizon as Y, we can see that X is a 737×1 vector, Y is also a 737×1 vector. We can see from the plots that these two stocks do have similar trend of prices. The correlation coefficient is 0.82.

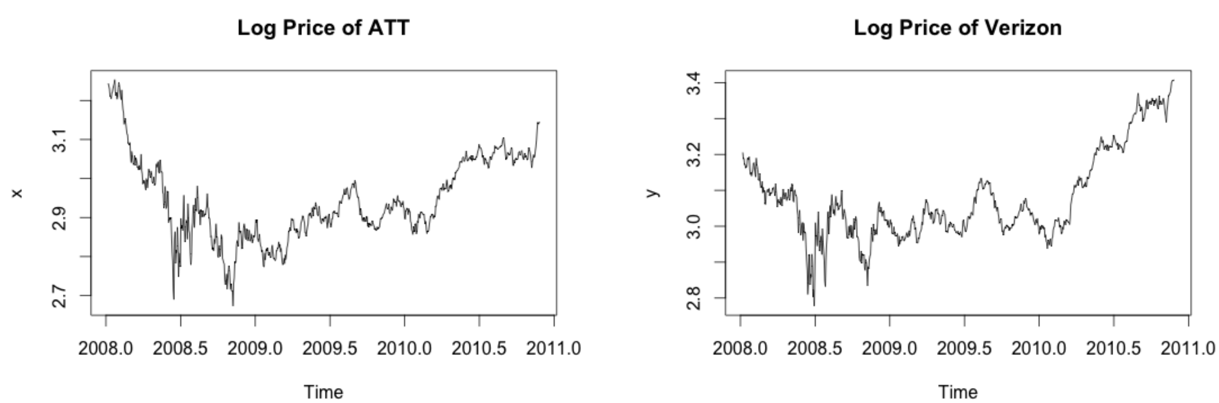

Figure 1: Price trend of AT&T and Verizon

3. Cointegration

3.1 The linear regression approach and AR(3) approach

First, we consider the simple linear regression model. The residual plot shows the residual series has certain patterns and it is not stationary. To further confirm it, we plot the ACF, which shows that it decays very slowly. So we fit the AR(3) model to the residual series. The coefficients are not significant. Otherwise, although the ADF test shows that the residuals of the AR(3) model is stationary, the ACF and PACF plot are not clean. After we plot the ACF and PACF of the squared of residual, it is obvious to tell that there is ARCH effect. Thus, the simple linear regression model with the AR(3) model is not satisfying. So we decide to use the Johansen procedure to fit the series.

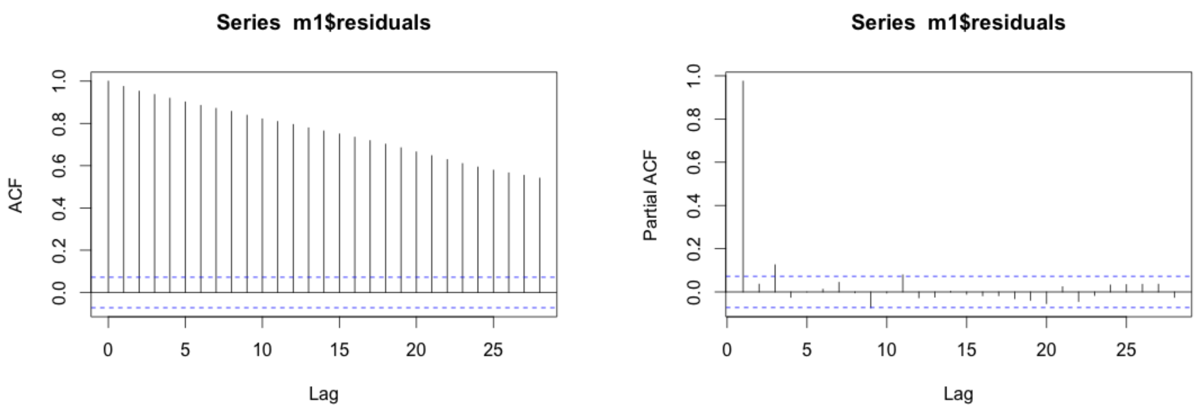

Figure 2: ACF and PACF of linear residuals indicating it is not a good model

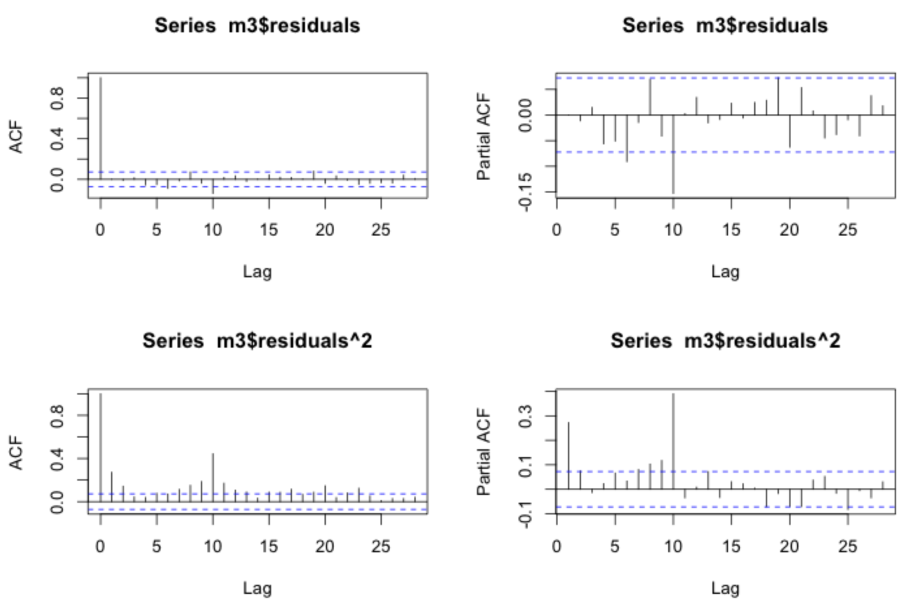

Figure 3: ACF and PACF of AR(3) squared residuals indicating it is not a good model

3.2 The Johansen procedure

```
######################
# Johansen-Procedure #
######################

Test type: trace statistic , with linear trend

Eigenvalues (lambda):
[1] 0.4902120 0.1794881

Values of teststatistic and critical values of test:

          test 10pct  5pct  1pct
r <= 1 |  5.54  6.50  8.18 11.65
r = 0  | 24.40 15.66 17.95 23.52
```

```
Eigenvectors, normalised to first column:
(These are the cointegration relations)

           ATTad.l3 Verad.l3
ATTad.l3  1.0000000  1.00000
Verad.l3 -0.3394999 12.10922

Weights W:
(This is the loading matrix)

           ATTad.l3    Verad.l3
ATTad.d -0.9852927 -0.01696224
Verad.d  0.4193060 -0.02548269
```

Figure 4: Results of the Johansen procedure

We can see from results of Johansen procedure that the estimated cointegration coefficient γ is -0.34.

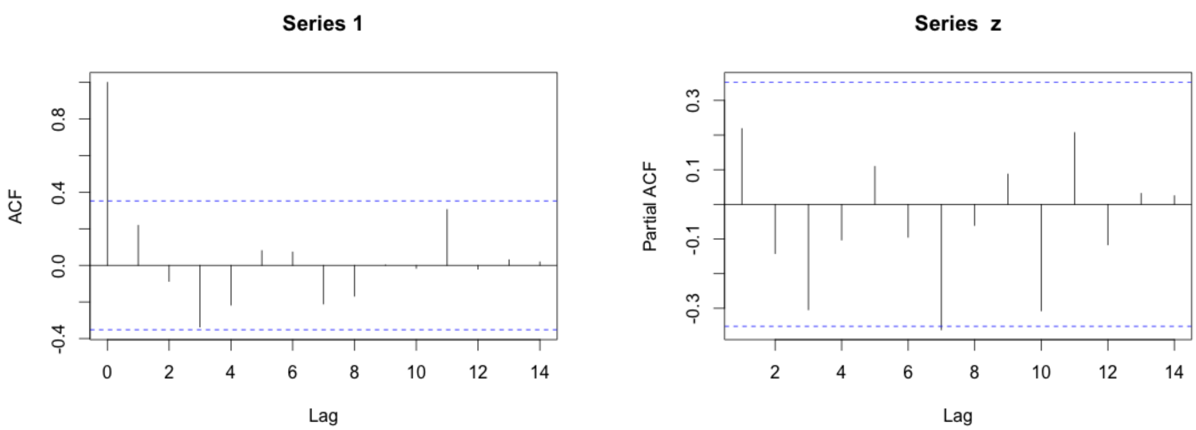

Figure 5: ACF and PACF of the residuals generated by Johansen procedure

We can see that this model is satisfactory. We can treat the residuals as a stationary noise process.

4. Trading Strategy and Application

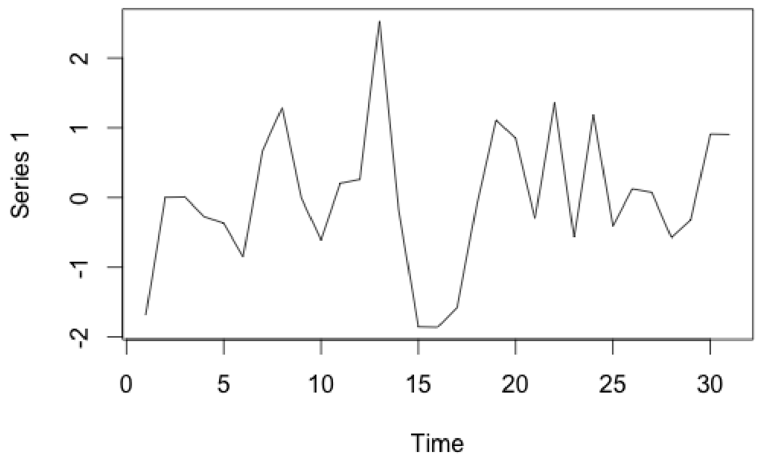

Figure 6: A glance of the normalized residuals

4.1 The fixed gamma approach

Using fixed gamma approach, our trading strategy can be expressed as follows. Firstly, we run Johansen procedure from day 1 to day 30 and obtain gamma. Then, in the next 30 trading days, i.e., day 31 to day 60, we use this fixed gamma to build the stationary noise process.

4.2 The floating gamma approach

Using floating gamma approach, we use gamma computed from the most recent 30 days and update our gamma every 10 days or 5 days. For example, if we update gamma every 10 days, the gamma used in days 31-40 is computed from days 1-30, the gamma used in days 41-50 is computed from days 11 to 40, and the gamma used in days 51-60 is computed from days 21-50.

4.3 Trading Strategy and Performance Comparation

Our trading process is as follows: Go over the residual process day by day. When the value of the residual is below 1.5 standard error, we build a long position. Once its value is within 0.3 standard error, we cover the long position we built. When its value is above 1.5 standard error, we build a short position. Once its value is within 0.3 standard error, we cover the short position we built.

Backtesting performances of our trading strategies are as follows:

**Apply the Strategy to the next 30 Days
Using Fixed Gamma**

	31-60 Days
Gamma	-0.3394999
Gain	0.06663566

**Apply the Strategy to the next 30 Days
Using Floating Gamma
(Changing Every 10 Days)**

	31-40 days	41-50 days	51-60 days
Gamma	-0.3394999	-1.420094	-1.558607
Gain	0.3372988	-0.3185177	0.401776
Total Gain	0.4205571		

**Apply the Strategy to the next 30 Days
Using Floating Gamma
(Changing Every 5 Days)**

	31-35 days	36-40 days	41-45 days	46-50 days	51-55 days	56-60 days
Gamma	0.3395	0.6975	1.4201	1.3287	1.5586	1.8208
Gain	0.3728	0.0367	0.0445	0.0579	0.4987	0.5359
Total Gain	1.268230477					

We can see clearly that the floating gamma approach has a better performance than fixed gamma approach. We can conclude that updating gamma every 5 trading days, and each gamma generating from the previous 30 days of price data is a good choice.

Code

```
library(tseries)
library(forecast)
library(urca)
###Data Selection
ATT=ATT <- read.csv("~/Desktop/ts project/ATT.csv", sep=";")
Ver=Verizon <- read.csv("~/Desktop/ts project/Verizon.csv", sep="")
par(mfrow=c(1,1))
x=ts(log(ATT[85:821,]),start=c(2008,5,1),freq=255)
plot(x,main='Log Price of ATT')
y=ts(log(Ver[85:821,]),start=c(2008,5,1),freq=255)
plot(y,main='Log Price of Verizon')
cor(log(ATT[85:821,]),log(Ver[85:821,]))

###Simple Linear Regression with AR(3) Model
m1=lm(log(ATT[85:821,]) ~ log(Ver[85:821,]))
summary(m1)
wt=m1$residuals
plot(ts(wt),start=c(2008,5,1),freq=255)
acf(wt)
pacf(wt)
m3=arima(wt,order=c(3,0,0),include.mean = F)
m3
summary(m3)
adf.test(m3$residuals)
par(mfrow=c(2,2))
acf(m3$residuals)
acf(m3$residuals^2)
pacf(m3$residuals)
pacf(m3$residuals^2)
par(mfrow=c(1,1))


###Cointegration
ATTad=log(ATT[684:714,])
Verad=log(Ver[684:714,])
xx=ts(ATTad)
plot(xx,main='Selected Log Price of ATT')
yy=ts(Verad)
plot(yy,main='Selected Log Price of Verizon')
xxy=cbind(ATTad,Verad)
mx=ar(xxy)
mx$order
cot=ca.jo(xxy, ecdet='none', type='trace', K=3, spec='longrun')
summary(cot)
cot@V[2,1] #gamma

###Strategy Application
zz=cbind(xx,yy)
weight=c(1,cot@V[2,1])
z=ts(zz%*%weight)
plot(z)
acf(z)
```

```
pacf(z)
sd(z)
mean(z)
abs((z-mean(z))/sd(z))
plot((z-mean(z))/sd(z))

####Fixed Gamma
spread = (z-mean(z))/sd(z)
longpos = rep(0,31)
i = 1
while(i != 31)
{
  if(spread[i]<-1.5)
  {
    longpos[i] = 1
    for(j in (i+1):31)
    {
      if((spread[j]>-0.3)|| j == 31)
        break
    }
    longpos[j] = -1
    i = j + 1
  }
  else
  {
    i = i + 1
  }
}
shortpos = rep(0,31)
i = 1
while(i != 31)
{
  if(spread[i] > 1.5)
  {
    shortpos[i] = 1
    for(j in (i+1):31)
    {
      if((spread[j]<0.3)|| j == 31)
        break
    }
    shortpos[j] = -1
    i = j + 1
  }
  else
  {
    i = i + 1
  }
}

gain1 = 0
buy=c(0,0)
sell=c(0,0)
for(i in 1:31)
{
```

```
  if(longpos[i]==1)
  {
    buy = c(ATT[714+i,],Ver[714+i,])
  }
  if(longpos[i] == -1)
  {
    sell = c(ATT[714+i,],Ver[714+i,])
    gain1=gain1+c(1,-Ver[714,]^(cot@V[2,1]-1))%*%(sell-buy)
  }
}

gain2 = 0
buy=c(0,0)
sell=c(0,0)
for(i in 1:31)
{
  if(shortpos[i]==1)
  {
    sell = c(ATT[714+i,],Ver[714+i,])
  }
  if(shortpos[i] == -1)
  {
    buy = c(ATT[714+i,],Ver[714+i,])
    gain2=gain2+c(1,-Ver[714,]^(cot@V[2,1]-1))%*%(sell-buy)
  }
}

gain = gain1 + gain2
gain

####Floating Gamma(Changing Every 5 Days)
####For simplification, I only post the calculation of the first two gains.
####The method of using floating gamma changing every 10 days to calculate the gain is the same.
####First Gamma
ATTad=log(ATT[684:714,])
Verad=log(Ver[684:714,])
xx=ts(ATTad)
plot(xx,main='Selected Log Price of ATT')
yy=ts(Verad)
plot(yy,main='Selected Log Price of Verizon')
xxy=cbind(ATTad,Verad)
mx=ar(xxy)
mx$order
cot=ca.jo(xxy, ecdet='none', type='trace', K=3, spec='longrun')
summary(cot)
cot@V[2,1] #gamma
zz=cbind(xx,yy)
weight=c(1,cot@V[2,1])
z=ts(zz%*%weight)
plot(z)
acf(z)
pacf(z)
sd(z)
mean(z)
```

```
abs((z-mean(z))/sd(z))
plot((z-mean(z))/sd(z))

spread = (z-mean(z))/sd(z)
longpos = rep(0,6)
i = 1
while(i != 6)
{
  if(spread[i]<-1.5)
  {
    longpos[i] = 1
    for(j in (i+1):6)
    {
      if((spread[j]>-0.3)|| j == 6)
        break
    }
    longpos[j] = -1
    i = j + 1
  }
  else
  {
    i = i + 1
  }
}
shortpos = rep(0,6)
i = 1
while(i != 6)
{
  if(spread[i] > 1.5)
  {
    shortpos[i] = 1
    for(j in (i+1):6)
    {
      if((spread[j]<0.3)|| j == 6)
        break
    }
    shortpos[j] = -1
    i = j + 1
  }
  else
  {
    i = i + 1
  }
}

gain1 = 0
buy=c(0,0)
sell=c(0,0)
for(i in 1:6)
{
  if(longpos[i]==1)
  {
    buy = c(ATT[714+i,],Ver[714+i,])
  }
```

```
  if(longpos[i] == -1)
  {
    sell = c(ATT[714+i,],Ver[714+i,])
    gain1=gain1+c(1,-Ver[714,]^(cot@V[2,1]-1))%*%(sell-buy)
  }
}

gain2 = 0
buy=c(0,0)
sell=c(0,0)
for(i in 1:6)
{
  if(shortpos[i]==1)
  {
    sell = c(ATT[714+i,],Ver[714+i,])
  }
  if(shortpos[i] == -1)
  {
    buy = c(ATT[714+i,],Ver[714+i,])
    gain2=gain2+c(1,-Ver[714,]^(cot@V[2,1]-1))%*%(sell-buy)
  }
}

gain = gain1 + gain2
gain

####Second Gamma
ATTad=log(ATT[689:719,])
Verad=log(Ver[689:719,])
xx=ts(ATTad)
plot(xx,main='Selected Log Price of ATT')
yy=ts(Verad)
plot(yy,main='Selected Log Price of Verizon')
xxy=cbind(ATTad,Verad)
mx=ar(xxy)
mx$order
cot=ca.jo(xxy, ecdet='none', type='trace', K=3, spec='longrun')
summary(cot)
cot@V[2,1] #gamma

spread = (z-mean(z))/sd(z)
longpos = rep(0,6)
i = 1
while(i != 6)
{
  if(spread[i]<-1.5)
  {
    longpos[i] = 1
    for(j in (i+1):6)
    {
      if((spread[j]>-0.3)|| j == 6)
        break
    }
    longpos[j] = -1
```

```
    i = j + 1
  }
  else
  {
    i = i + 1
  }
}
shortpos = rep(0,6)
i = 1
while(i != 6)
{
  if(spread[i] > 1.5)
  {
    shortpos[i] = 1
    for(j in (i+1):6)
    {
      if((spread[j]<0.3)|| j == 6)
        break
    }
    shortpos[j] = -1
    i = j + 1
  }
  else
  {
    i = i + 1
  }
}

gain1 = 0
buy=c(0,0)
sell=c(0,0)
for(i in 1:6)
{
  if(longpos[i]==1)
  {
    buy = c(ATT[719+i,],Ver[719+i,])
  }
  if(longpos[i] == -1)
  {
    sell = c(ATT[719+i,],Ver[719+i,])
    gain1=gain1+c(1,-Ver[719,]^(cot@V[2,1]-1))%*%(sell-buy)
  }
}

gain2 = 0
buy=c(0,0)
sell=c(0,0)
for(i in 1:6)
{
  if(shortpos[i]==1)
  {
    sell = c(ATT[719+i,],Ver[719+i,])
  }
  if(shortpos[i] == -1)
```

```
  {
    buy = c(ATT[719+i,],Ver[719+i,])
    gain2=gain2+c(1,-Ver[719,]^(cot@V[2,1]-1))%*%(sell-buy)
  }
}

gain = gain1 + gain2
gain
```