Single Index Model

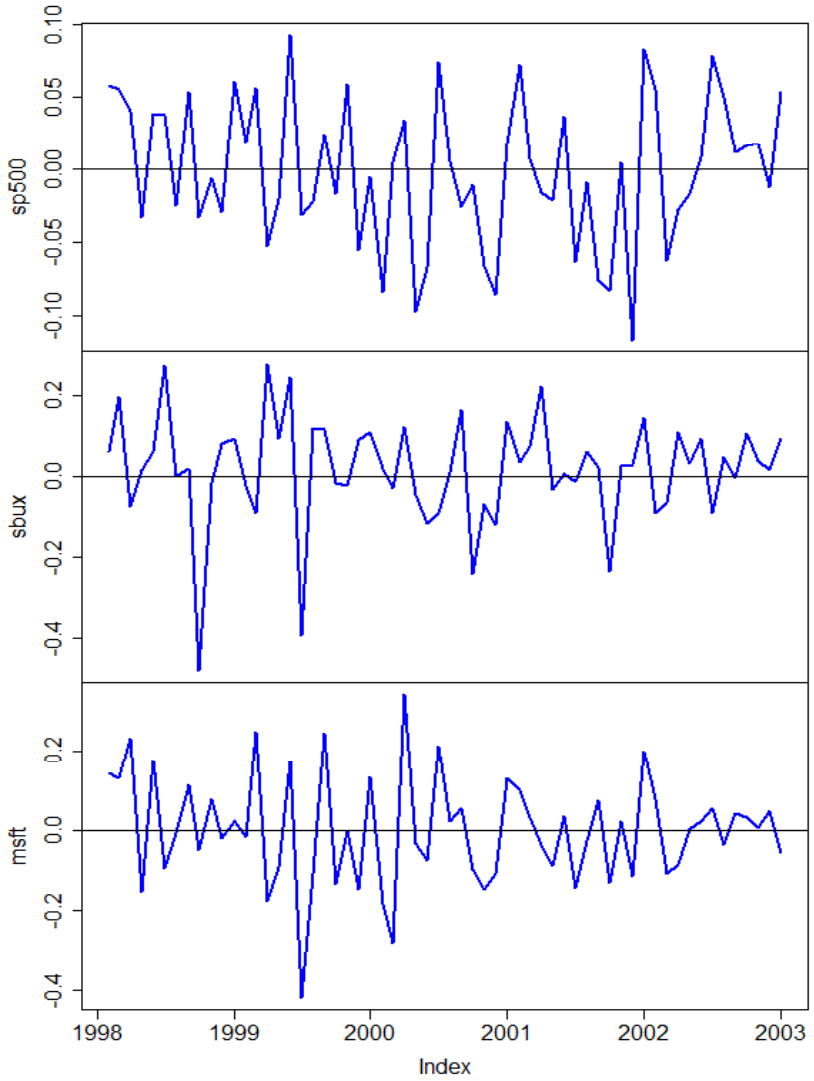
Amath 462/Econ 424
Eric Zivot
Summer 2013
Updated: August 13, 2013

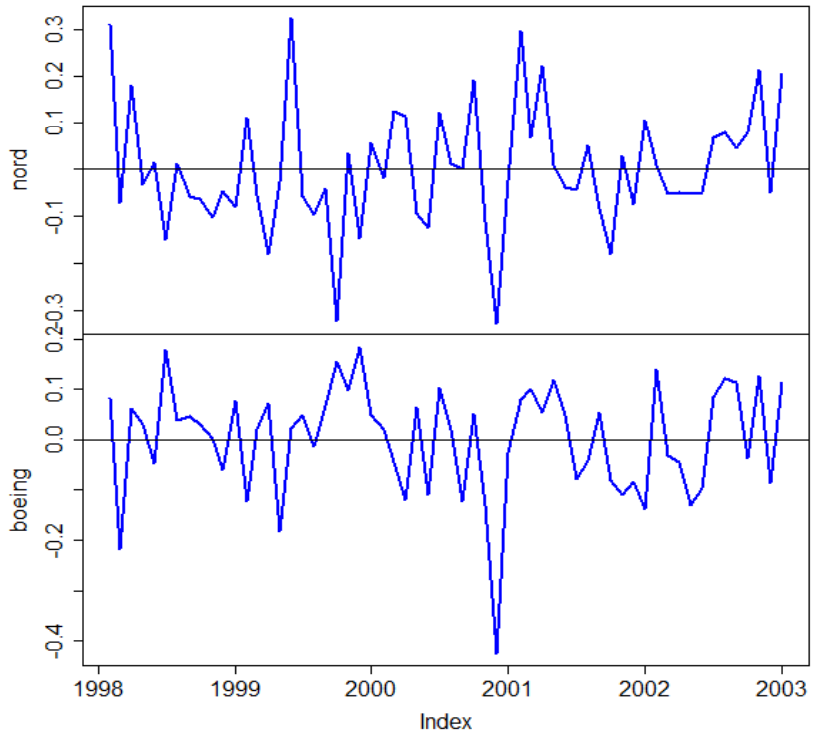
Example Data

Monthly continuously compounded returns on S&P 500, Starbucks, Microsoft, Nordstrom and Boeing from 1/1/98 – 1/1/2003 (5 years of monthly data)

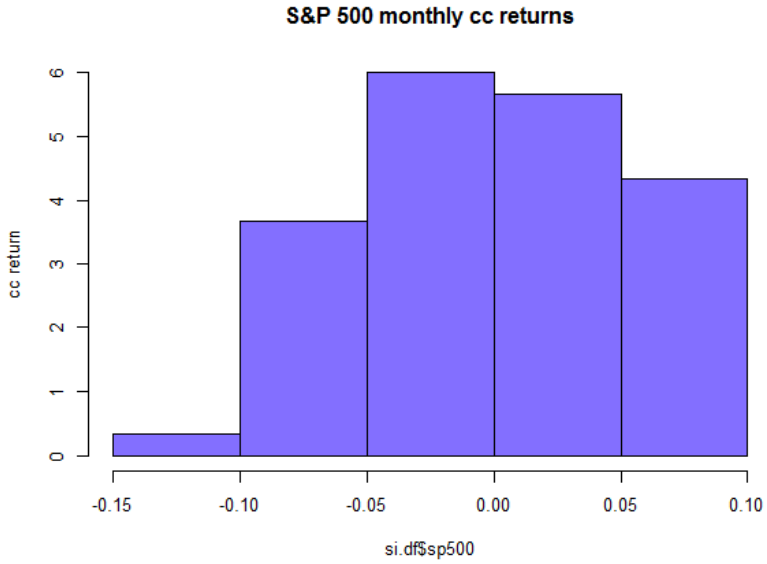
```
> head(si.df)
              sp500     sbux     msft     nord   boeing
1998-02-01  0.05744  0.06079  0.14155  0.31280  0.08202
1998-03-01  0.05484  0.19625  0.12835 -0.07138 -0.21923
1998-04-01  0.04019 -0.07471  0.23258  0.18243  0.06103
1998-05-01 -0.03282  0.01524 -0.15346 -0.03172  0.03069
1998-06-01  0.03806  0.05947  0.17738  0.01545 -0.04702
1998-07-01  0.03724  0.27495 -0.09734 -0.14975  0.17825
```

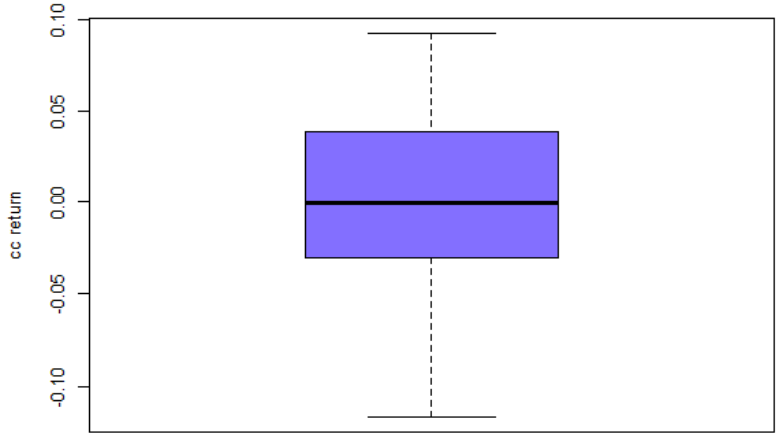
Example Data: Returns

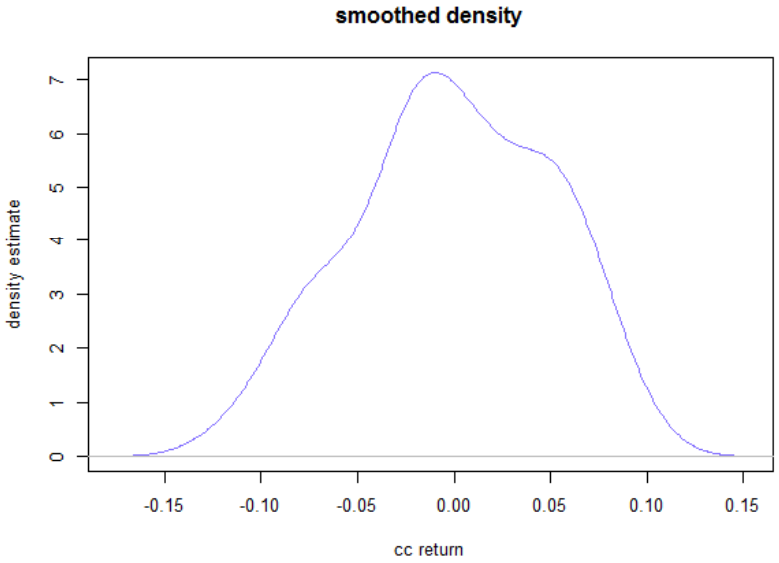


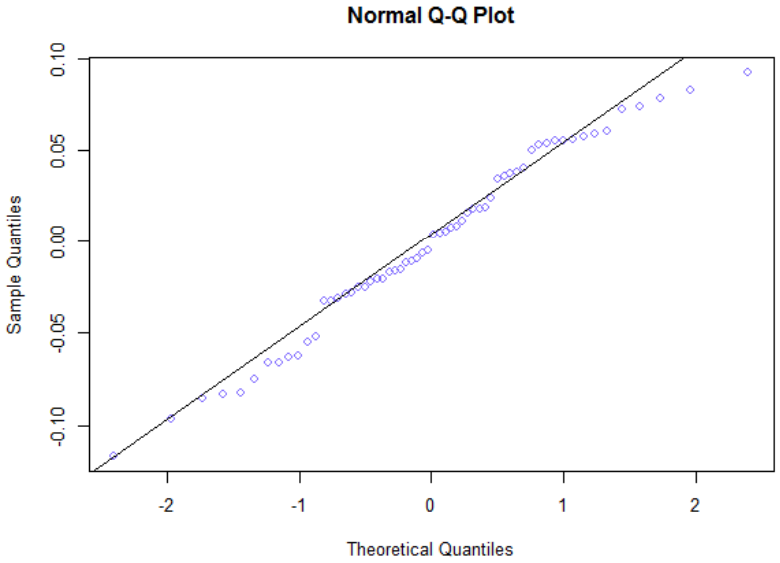


Distribution of Market Returns

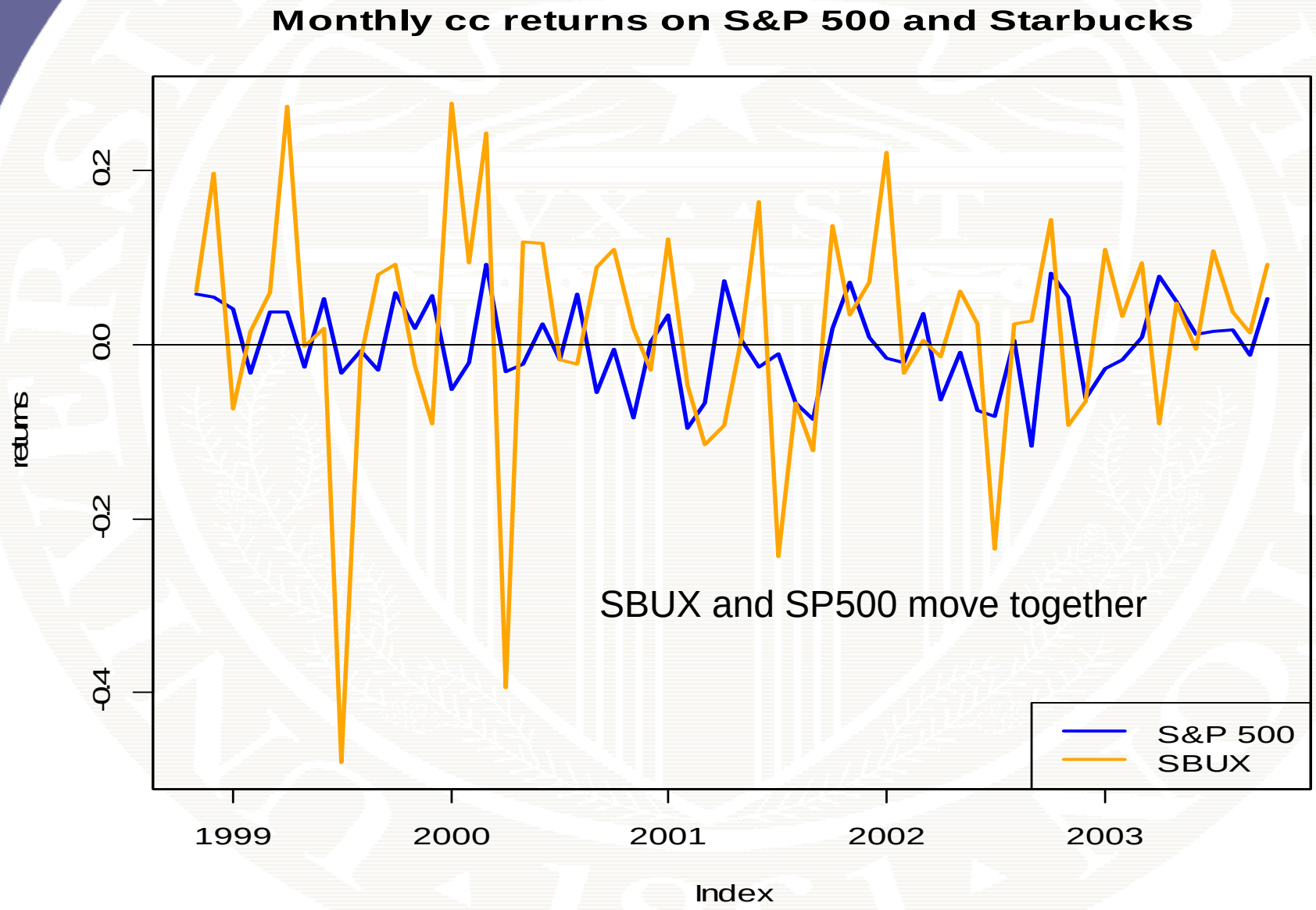




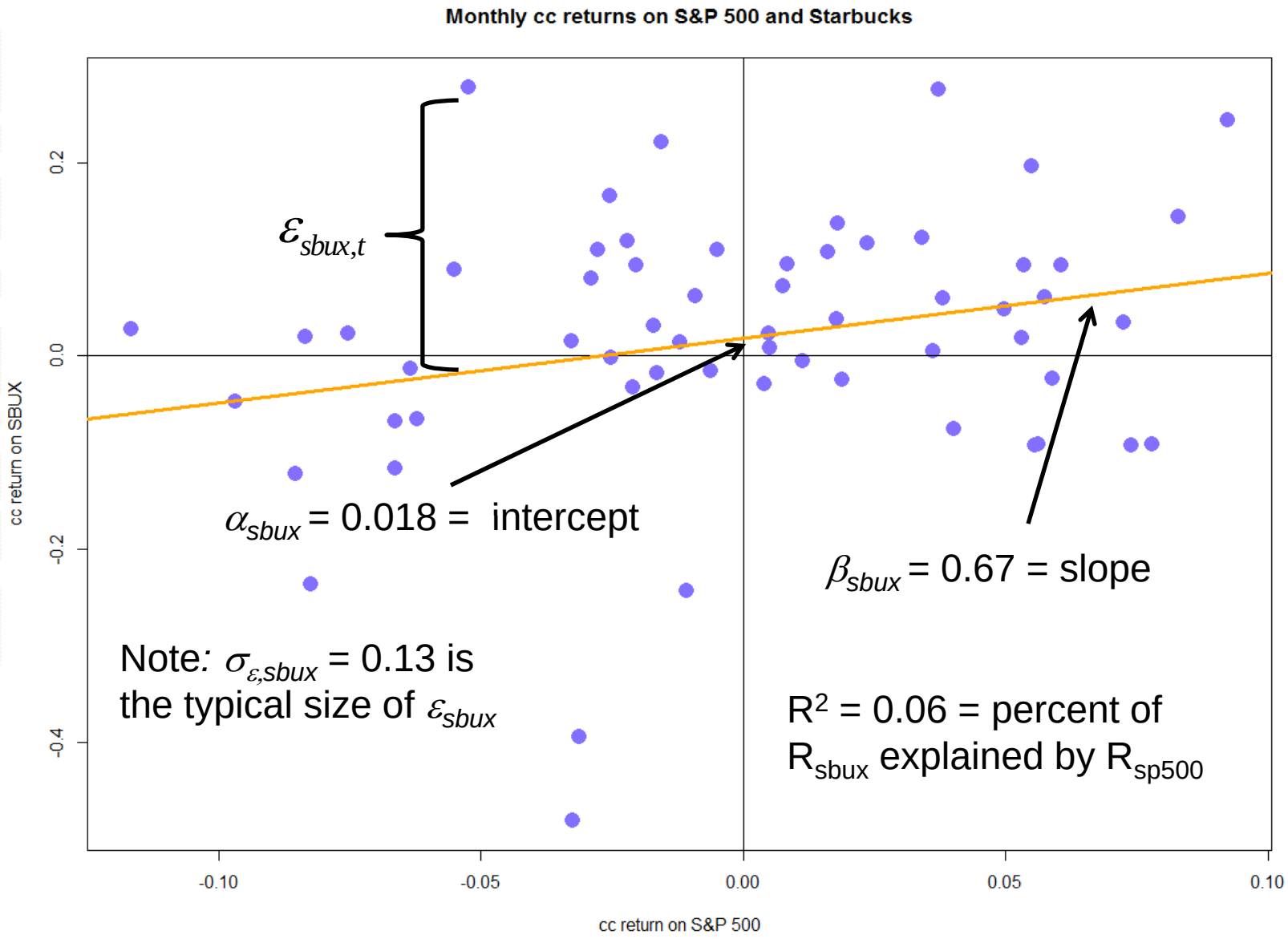




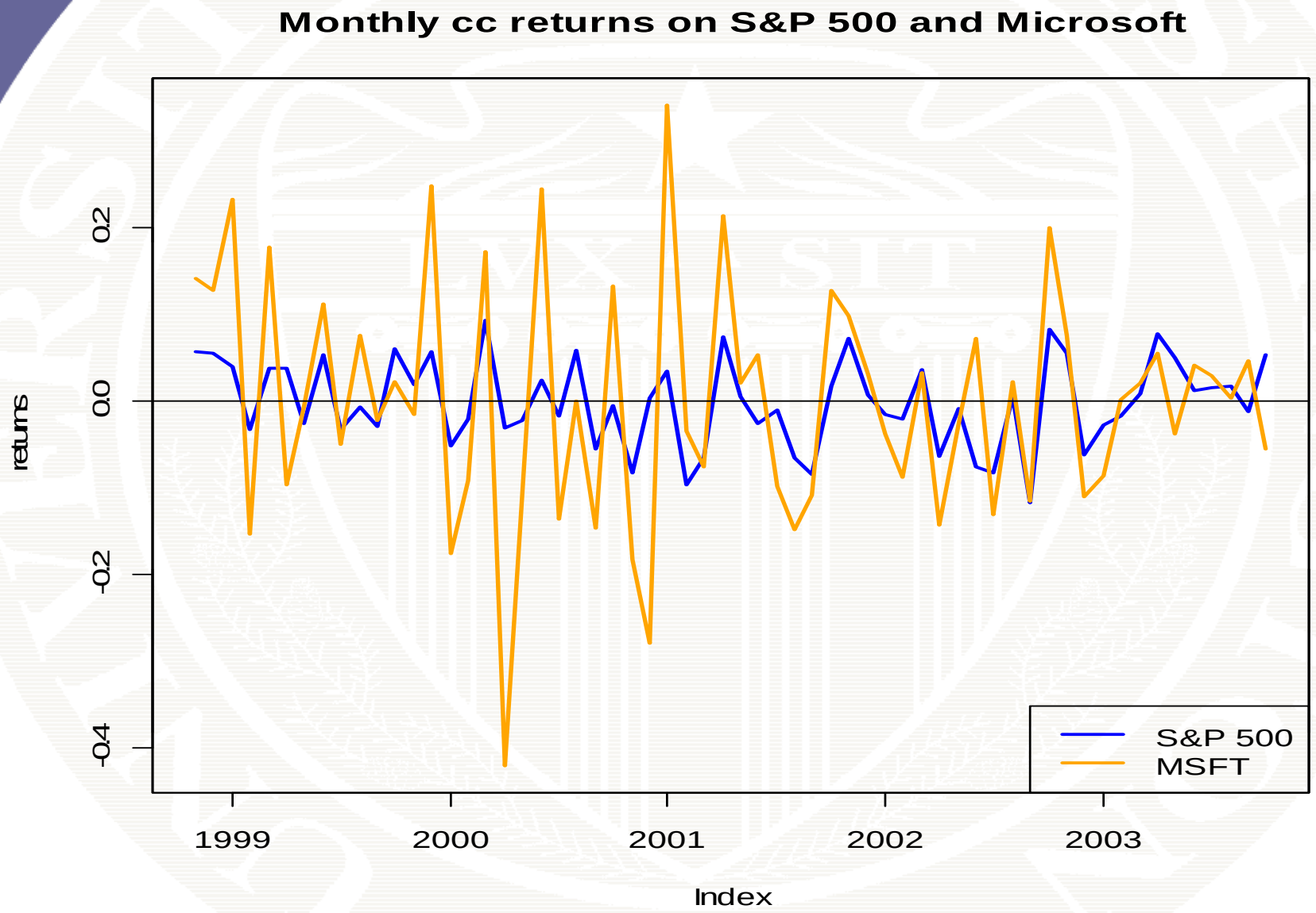
$$R_{sbux} = \alpha_{sbux} + \beta_{sbux} R_M + \varepsilon_{sbux} \qquad \beta_{sbux} = 0.67$$



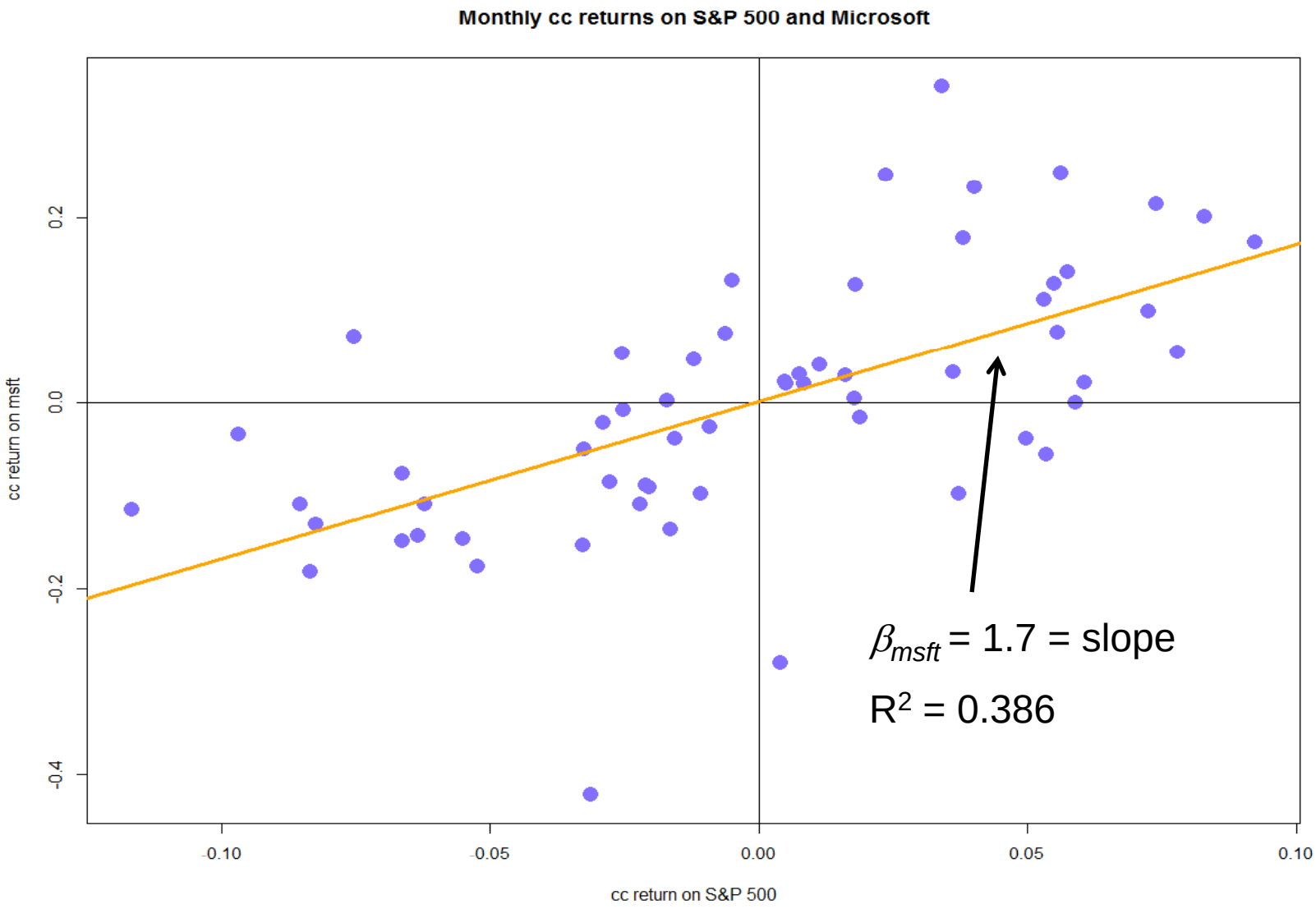
$$R_{sbux} = \alpha_{sbux} + \beta_{sbux} R_M + \varepsilon_{sbux},\ \varepsilon_{sbux} \sim N(0, \sigma^2_{\varepsilon,sbux})$$



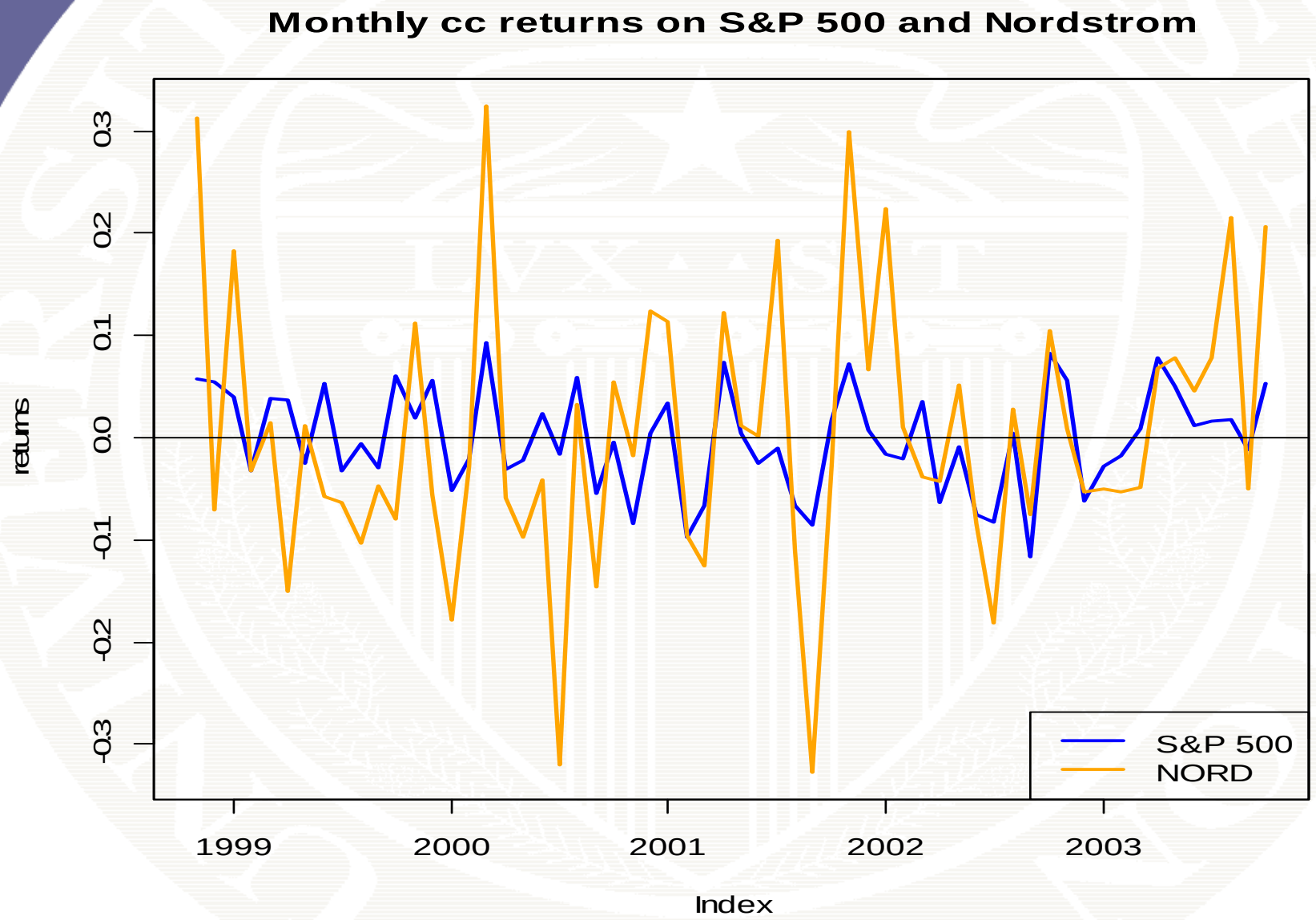
Beta = 1.7



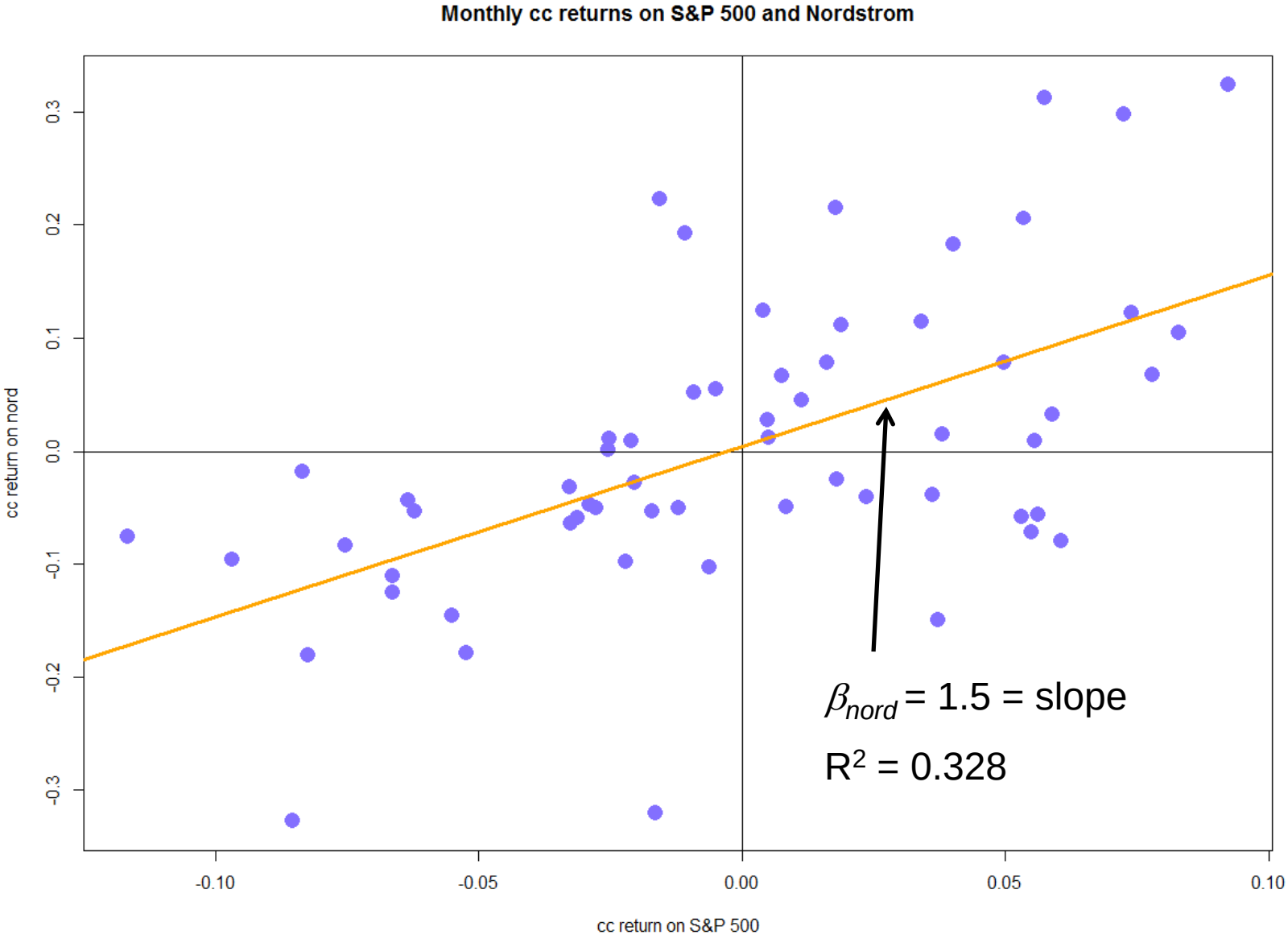
$$R_{msft} = \alpha_{msft} + \beta_{msft} R_M + \varepsilon_{msft}$$



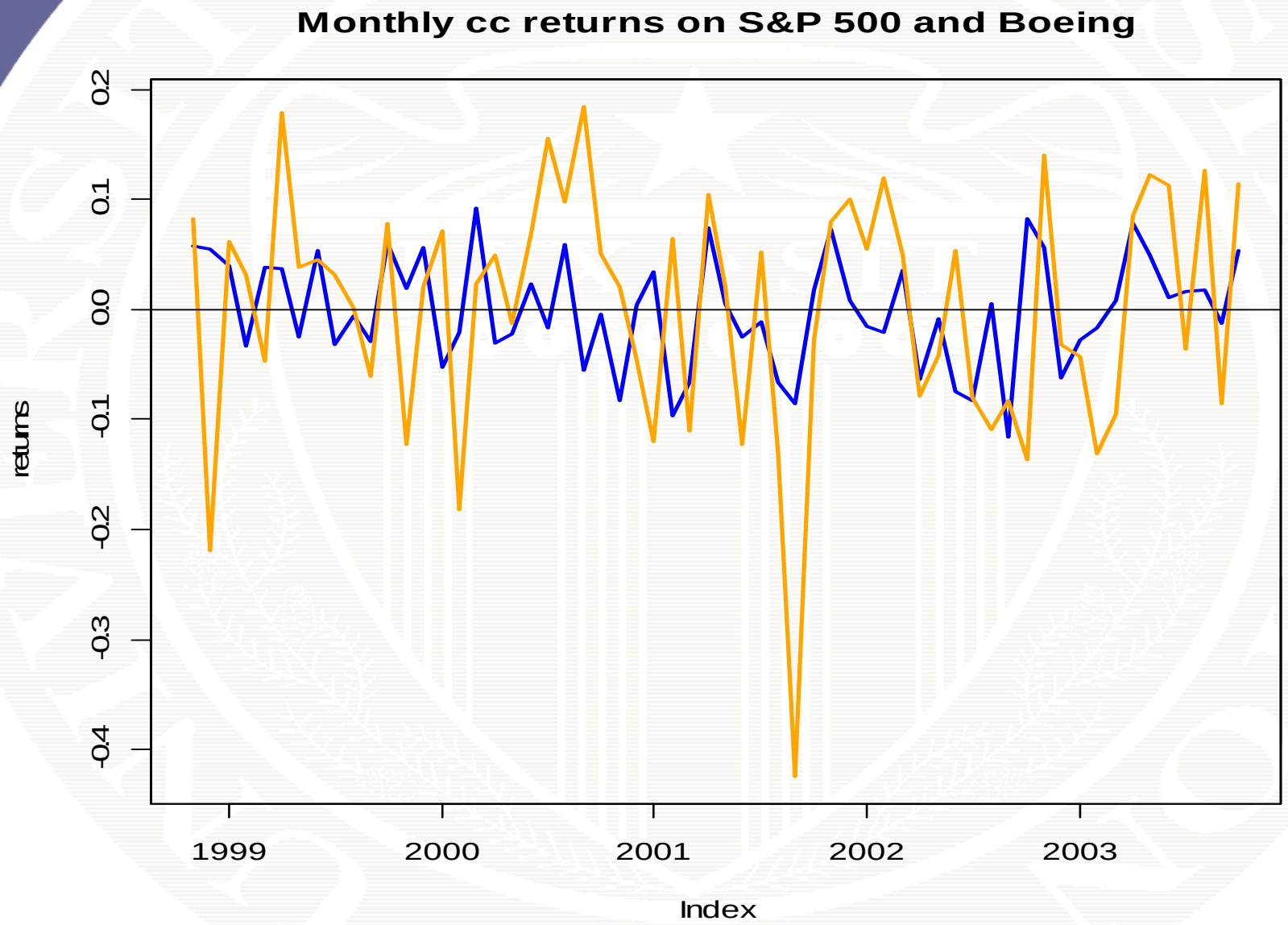
Beta = 1.5



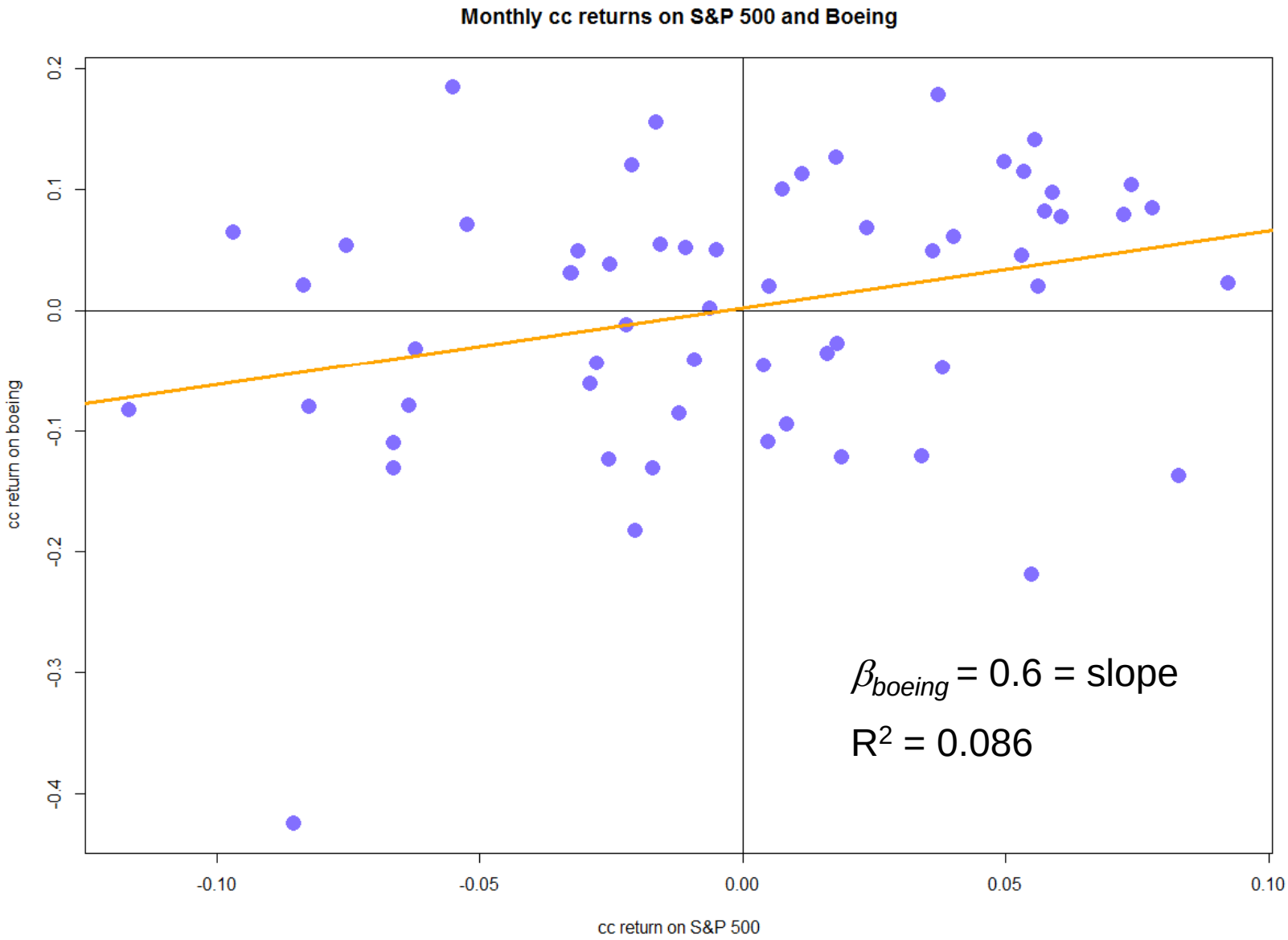
$$R_{nord} = \alpha_{nord} + \beta_{nord} R_M + \varepsilon_{nord}$$



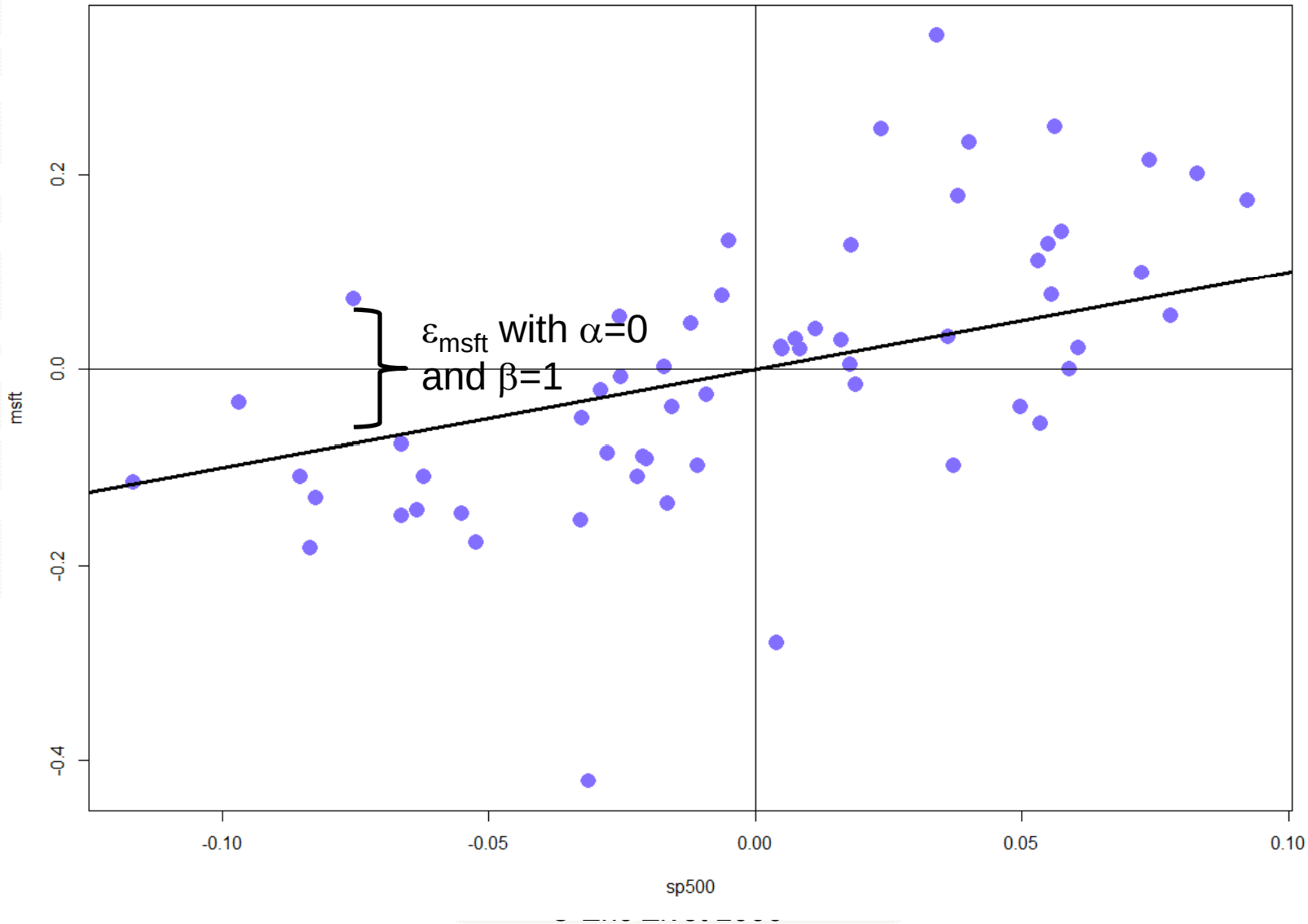
Beta = 0.64



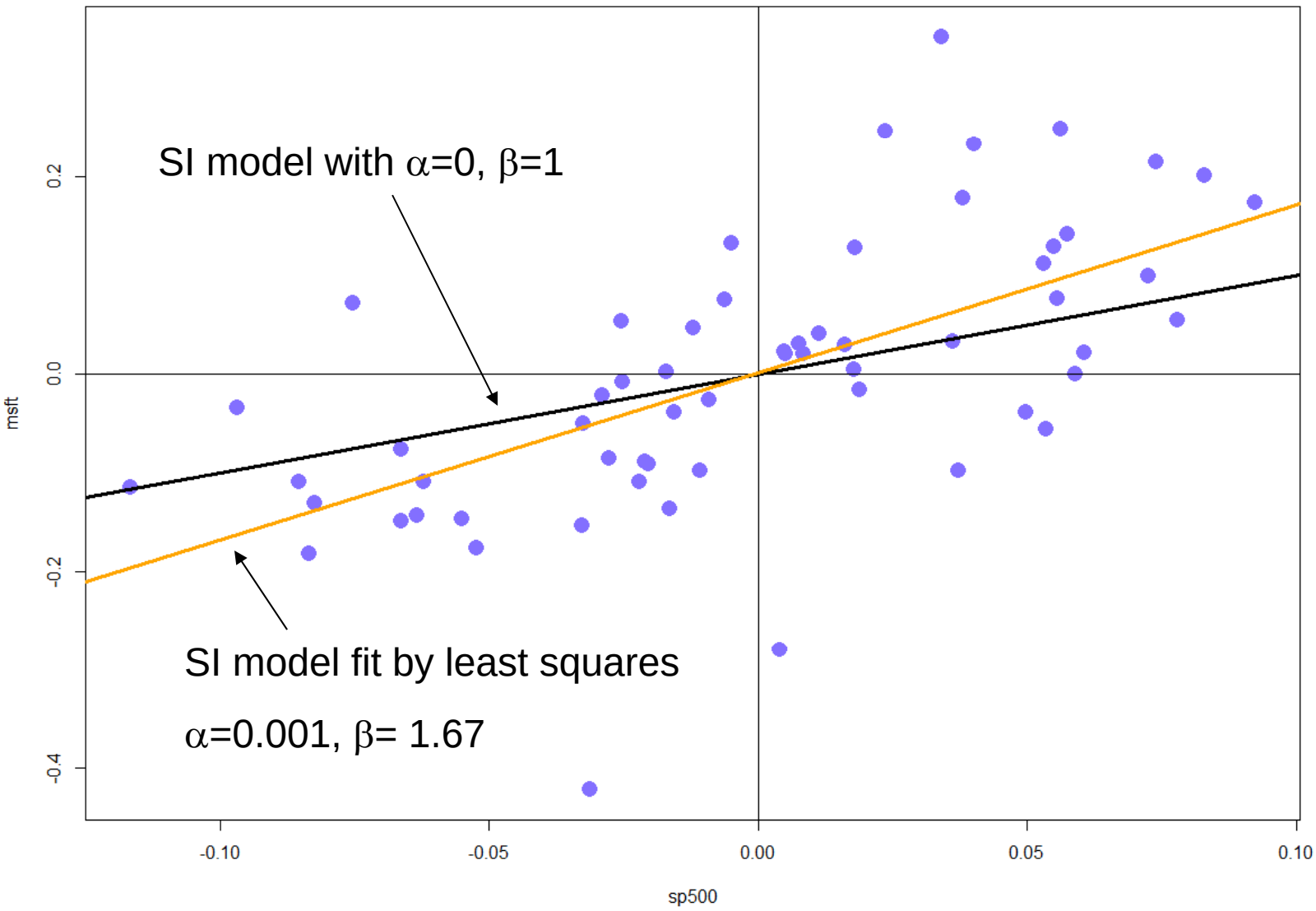
$$R_{boeing} = \alpha_{boeing} + \beta_{boeing} R_M + \varepsilon_{boeing}$$



SI Model for MSFT with α=0 and β=1



SI Model Fit by Least Squares



Example Data

Monthly continuously compounded returns on S&P 500, Starbucks, Microsoft, Nordstrom and Boeing

```
> head(si.df)
     sp500     sbux     msft     nord   boeing
1  0.05744  0.06079  0.14155  0.31280  0.08202
2  0.05484  0.19625  0.12835 -0.07138 -0.21923
3  0.04019 -0.07471  0.23258  0.18243  0.06103
4 -0.03282  0.01524 -0.15346 -0.03172  0.03069
5  0.03806  0.05947  0.17738  0.01545 -0.04702
6  0.03724  0.27495 -0.09734 -0.14975  0.17825

> colnames(si.df)
[1] "sp500"  "sbux"   "msft"   "nord"   "boeing"
```

Least Squares in R

```
> msft.fit = lm(msft~sp500,data=si.df)
> class(msft.fit)
[1] "lm“
> names(msft.fit)
 [1] "coefficients"  "residuals"     "effects“  "rank"
 [5] "fitted.values" "assign"        "qr"
"df.residual"
 [9] "xlevels"       "call"          "terms”    "model"
> msft.fit$coef
 (Intercept)        sp500
   0.001199    1.697067
```

$\hat{\alpha}$ $\hat{\beta}$

LM Print Method

```
> msft.fit
Call:
lm(formula = msft ~ sp500, data = si.df)

Coefficients:
(Intercept)        sp500
     0.0012       1.6971
```

LM Summary Method

```
> summary(msft.fit)
Call:
lm(formula = msft ~ sp500, data = si.df)

Residuals:
    Min      1Q  Median      3Q     Max
-0.3690 -0.0540  0.0050  0.0469  0.2828

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   0.0012     0.0140    0.09     0.93
sp500         1.6971     0.2808    6.04  1.2e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.'

Residual standard error: 0.109 on 58 degrees of freedom
Multiple R-squared: 0.386, Adjusted R-squared: 0.376
F-statistic: 36.5 on 1 and 58 DF,  p-value: 1.16e-07
```

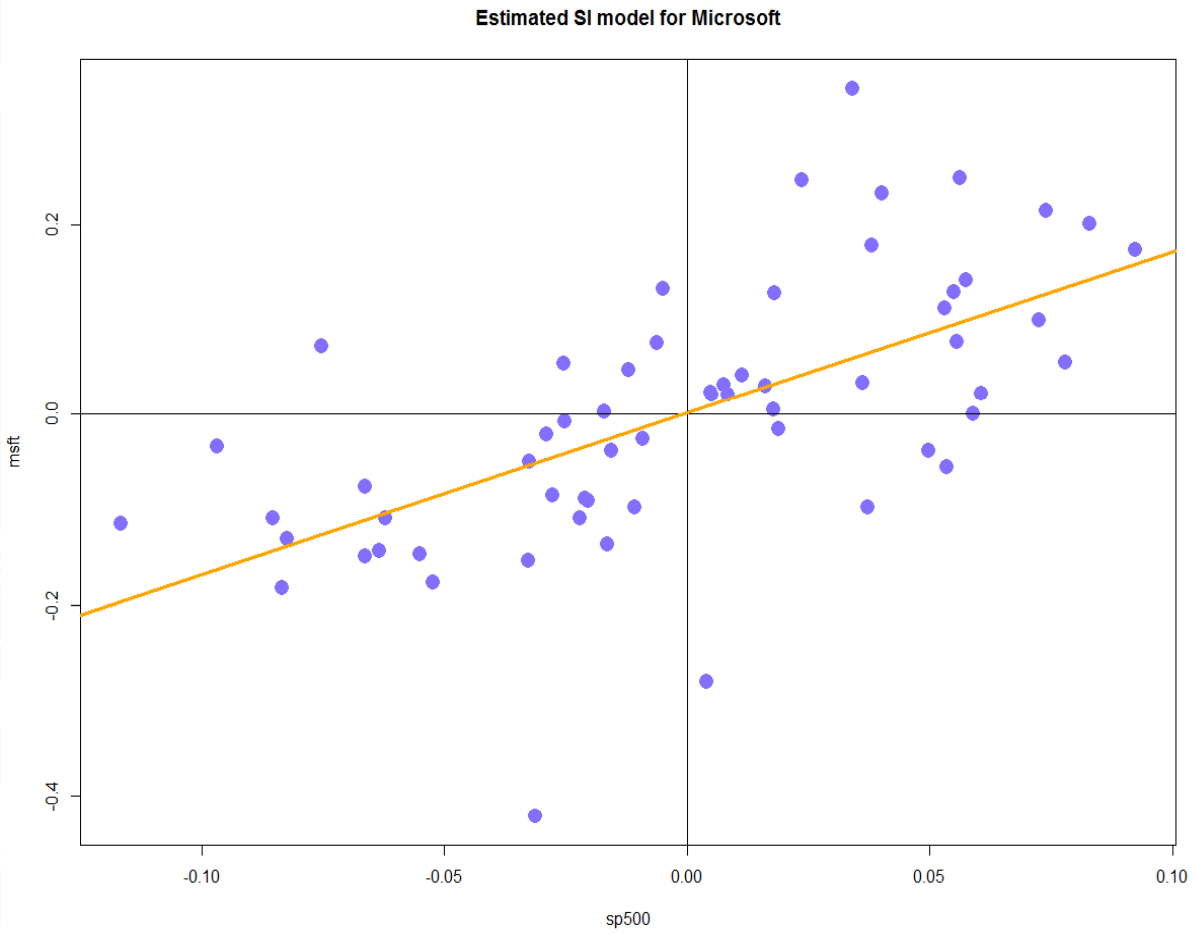
$SE(\hat{\alpha})$

$SE(\hat{\beta})$

$\hat{\sigma}_{\varepsilon}$

R^2

Scatterplot with Regression Line




```
> plot(si.df$sp500, si.df$msft, pch=16, lwd=2, col="blue")
> abline(msft.fit, col="orange", lwd=2)
> abline(h=0, v=0)
```

95% Confidence Intervals

$$\hat{\beta} \pm 2 \times SE(\hat{\beta})$$

$$1.697 \pm 2 \times (0.2808)$$

$$= [1.135,\ 2.259]$$

Note: 95% confidence interval is pretty big!

=> β is not very precisely estimated for individual stocks

Confidence Intervals In R

```
> confint(msft.fit, level=0.95)
                2.5 %   97.5 %
(Intercept) -0.02688 0.02928
sp500        1.13489 2.25925
```

LM Extractor Functions

```
> coef(msft.fit)
(Intercept)       sp500
   0.001199    1.697067
```

$$\hat{\varepsilon}_t = R_t - \hat{\alpha} - \hat{\beta} R_{Mt}$$

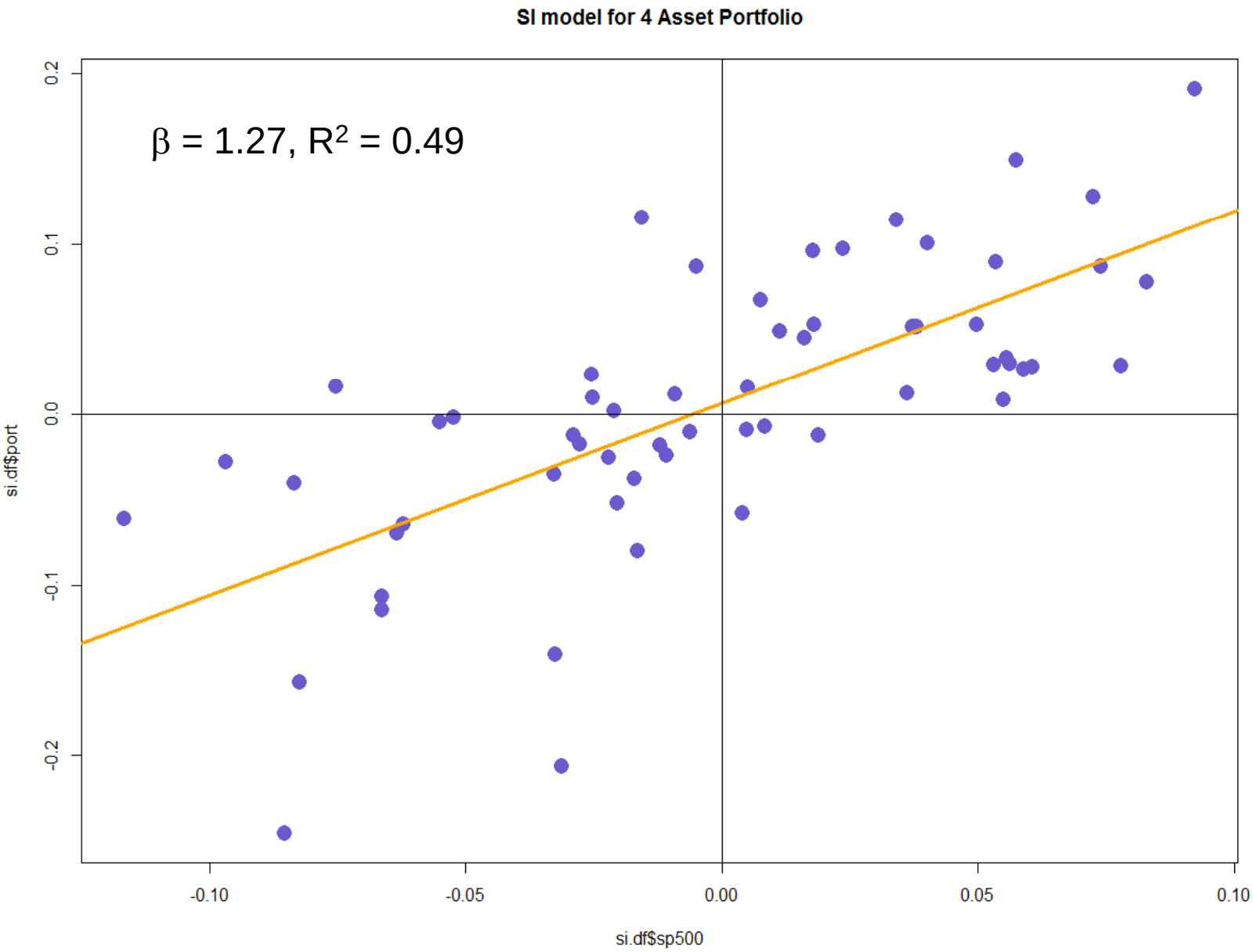
```
> residuals(msft.fit)[1:5]
1998-02-01 1998-03-01 1998-04-01 1998-05-01 1998-06-01
   0.04286    0.03408    0.16318   -0.09897    0.11159
```

$$\hat{R}_t = \hat{\alpha} + \hat{\beta} R_{Mt}$$

```
> fitted(msft.fit)[1:5]
1998-02-01 1998-03-01 1998-04-01 1998-05-01 1998-06-01
   0.09869    0.09427    0.06941   -0.05449    0.06579
```

SI Model for 4 Asset Portfolio

```
> port = (si.df$sbux + si.df$msft + si.df$nord +
+         si.df$boeing)/4
> new.data = data.frame(si.df,port)
> port.fit = lm(port~sp500,data=new.data)

> summary(port.fit)

Call: lm(formula = port ~ sp500, data = new.data)
Residuals:
     Min       1Q   Median       3Q    Max
 -0.1776 -0.03609 -0.002005 0.04635 0.1264

Coefficients:
             Value Std. Error t value Pr(>|t|)
(Intercept) 0.0065 0.0075     0.8616  0.3924
      sp500 1.1276 0.1510     7.4668  0.0000

Residual standard error: 0.05842 on 58 degrees of freedom
Multiple R-Squared: 0.4901
```



Portfolio Beta

```
# show beta of portfolio = weighted avg of individual betas

> beta.sbux = coef(lm(sbux~sp500,data=si.df))[2]
> beta.msft = coef(lm(msft~sp500,data=si.df))[2]
> beta.nord = coef(lm(nord~sp500,data=si.df))[2]
> beta.boeing = coef(lm(boeing~sp500,data=si.df))[2]

> (beta.sbux + beta.msft + beta.nord + beta.boeing)/4
    sp500
 1.127567

> coef(port.fit)[2]
    sp500
 1.127567
```

Single Index Model Fit

Asset	β	SE(β)	σ_ε	R^2
Boeing	0.638	0.273	0.106	0.09
Msft	1.697	0.281	0.109	0.39
Nord	1.508	0.283	0.110	0.33
Sbux	0.667	0.342	0.132	0.06
port	**1.128**	**0.151**	**0.058**	**0.49**

Single stocks vs. portfolio

- Portfolio β is closer to 1
- Portfolio β is estimated more precisely (SE is smaller)
- Portfolio σ_ε is smaller (diversification effect)
- Portfolio R^2 is higher (diversification effect)

Estimating Covariances

```
> beta.vec
  SBUX   MSFT   NORD BOEING
0.6666 1.6971 1.5080 0.6386

> sig2.sp500
[1] 0.002537

> cov.market = sig2.sp500*(beta.vec%*%t(beta.vec))
> cov.market
         SBUX     MSFT     NORD   BOEING
[1,] 0.001127 0.002870 0.002550 0.001080
[2,] 0.002870 0.007307 0.006493 0.002750
[3,] 0.002550 0.006493 0.005769 0.002443
[4,] 0.001080 0.002750 0.002443 0.001035
```

Estimating Covariances

```
> D.mat =
diag(c(sig2e.sbux,sig2e.msft,sig2e.nord,sig2e.boeing))
> D.mat
        [,1]    [,2]    [,3]    [,4]
[1,] 0.01719 0.00000 0.00000 0.00000
[2,] 0.00000 0.01161 0.00000 0.00000
[3,] 0.00000 0.00000 0.01179 0.00000
[4,] 0.00000 0.00000 0.00000 0.01101

> cov.si = cov.market + D.mat
> cov.si
         SBUX     MSFT     NORD   BOEING
[1,] 0.01832 0.002870 0.002550 0.001080
[2,] 0.00287 0.018913 0.006493 0.002750
[3,] 0.00255 0.006493 0.017564 0.002443
[4,] 0.00108 0.002750 0.002443 0.012045
```

Estimating Covariances

```
# compare with sample covariance matrix
> print(cov.hat,digits=4)
             sbux      msft     nord    boeing
  sbux 0.0183176 0.0055003 0.002735 0.0001221
  msft 0.0055003 0.0189132 0.006987 0.0001189
  nord 0.0027354 0.0069870 0.017564 0.0037662
boeing 0.0001221 0.0001189 0.003766 0.0120446

> print(cov.si,digits=4)
        SBUX     MSFT     NORD   BOEING
[1,] 0.01832 0.002870 0.002550 0.001080
[2,] 0.00287 0.018913 0.006493 0.002750
[3,] 0.00255 0.006493 0.017564 0.002443
[4,] 0.00108 0.002750 0.002443 0.012045
```

Estimating Correlations

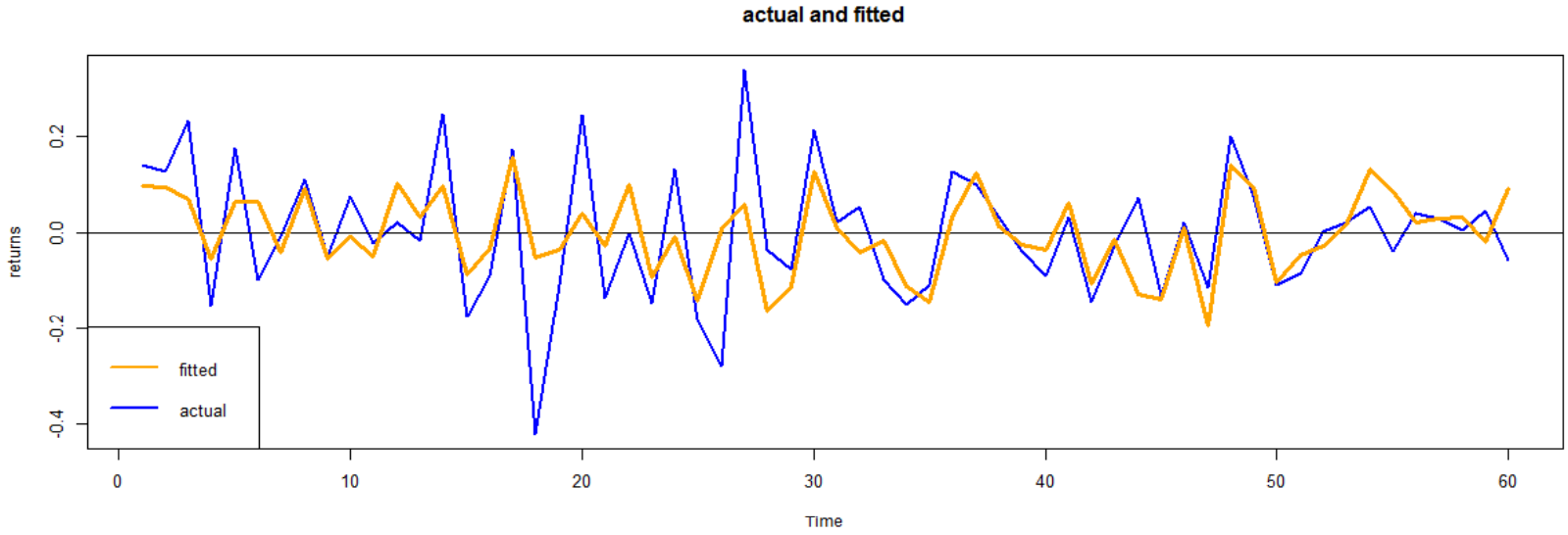
```
> print(cor.hat,digits=4)
          sbux     msft   nord   boeing
  sbux 1.000000 0.295506 0.1525 0.008218
  msft 0.295506 1.000000 0.3833 0.007876
  nord 0.152500 0.383348 1.0000 0.258940
boeing 0.008218 0.007876 0.2589 1.000000

> print(cor.si,digits=4)
          sbux   msft   nord  boeing
  sbux 1.00000 0.1542 0.1422 0.07271
  msft 0.15419 1.0000 0.3562 0.18218
  nord 0.14218 0.3562 1.0000 0.16798
boeing 0.07271 0.1822 0.1680 1.00000
```

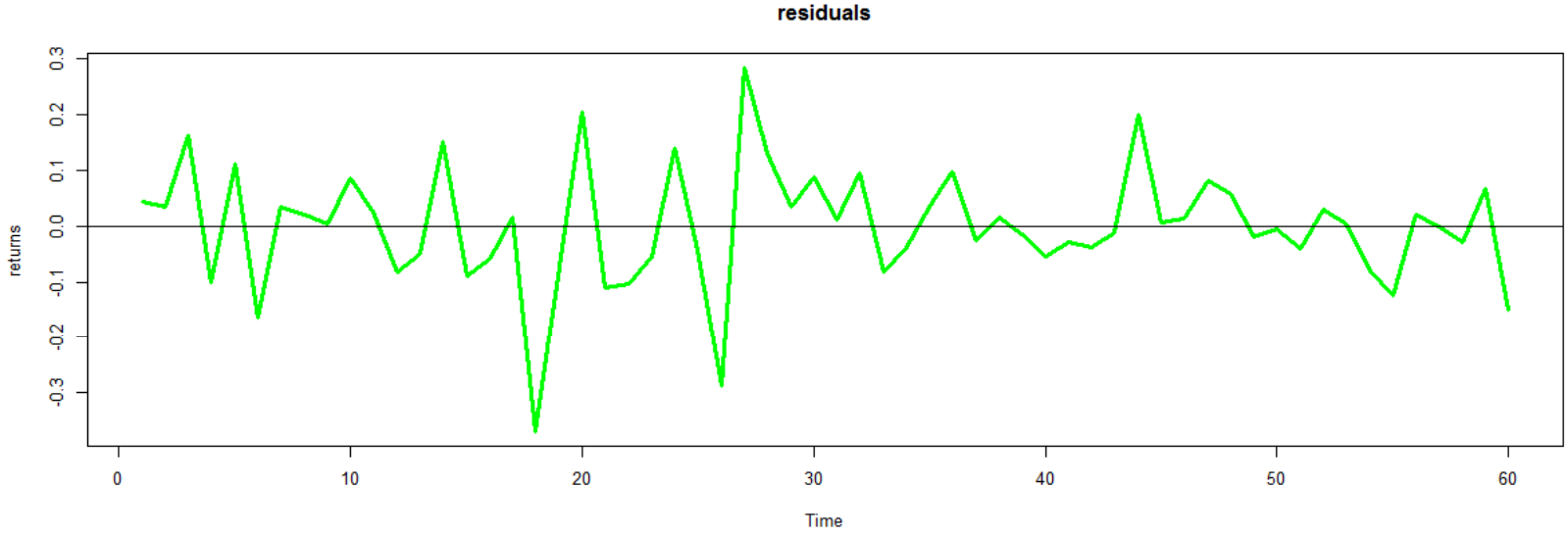
t-Values

In the R summary output, the t values are t-statistics for testing the hypothesis that the true coefficient is equal to zero

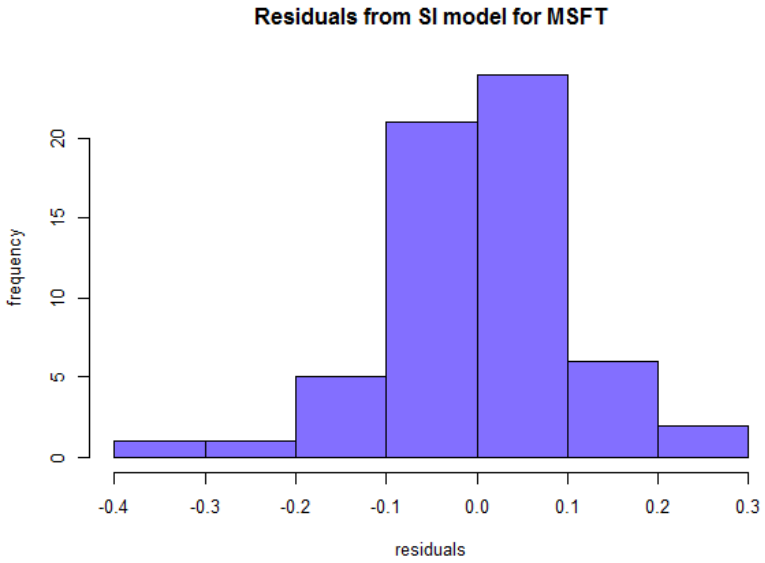
$$t_{\alpha=0} = \frac{\hat{\alpha}}{SE(\hat{\alpha})} = \frac{.0012}{.0140} = .0855$$

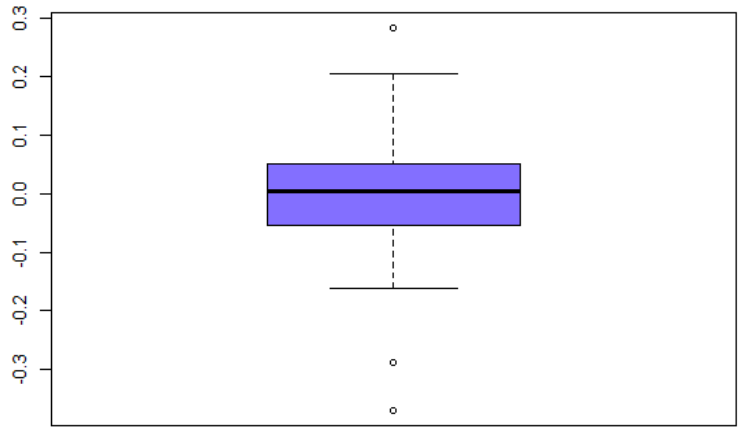
$$t_{\beta=0} = \frac{\hat{\beta}}{SE(\hat{\beta})} = \frac{1.6971}{.2808} = 6.0426$$

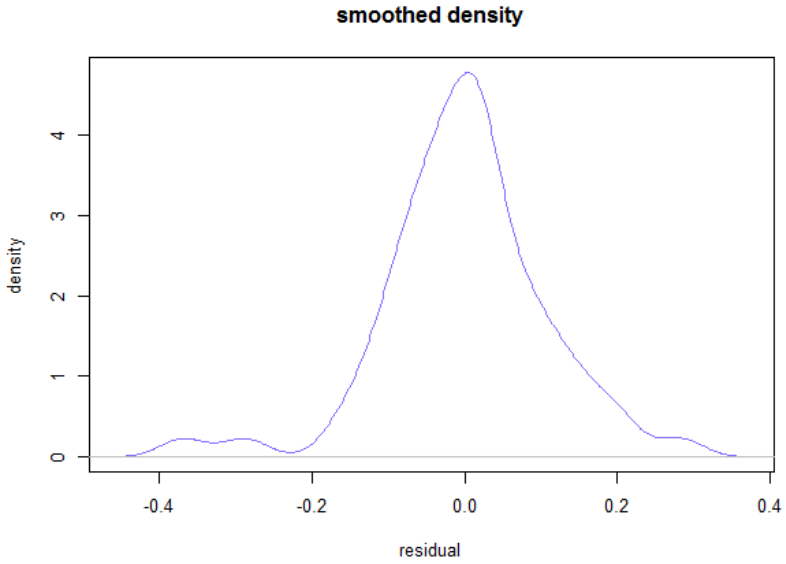


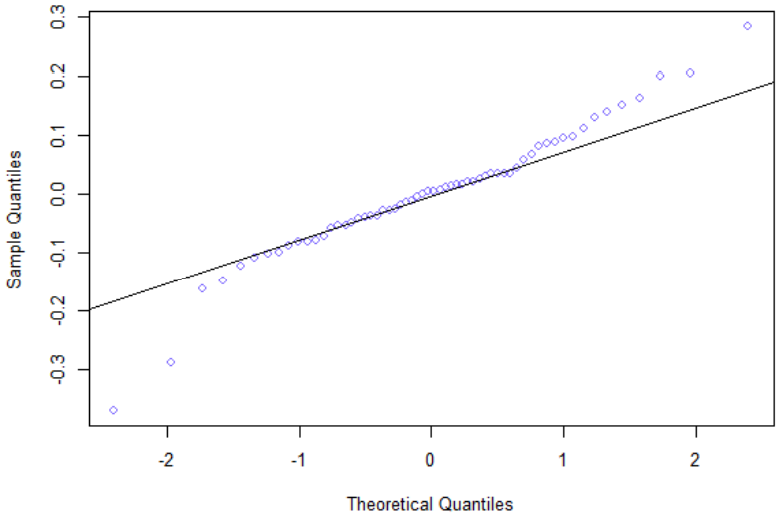


In the Single Index model the errors should behave like White Noise

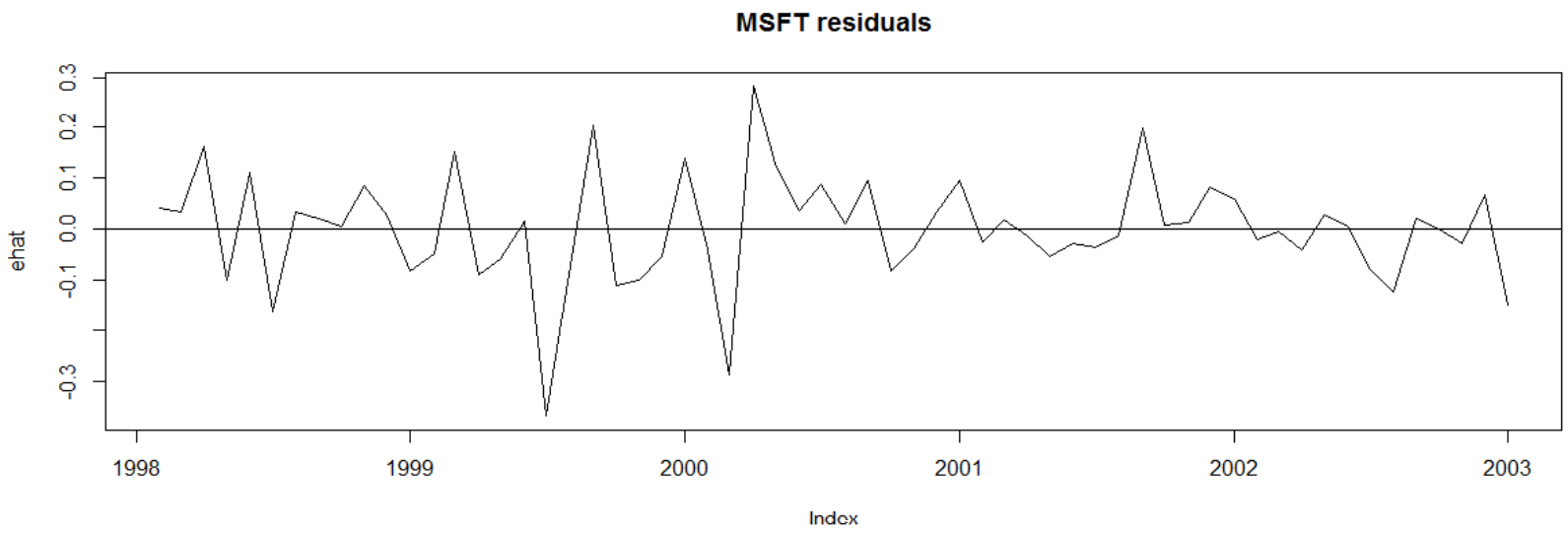


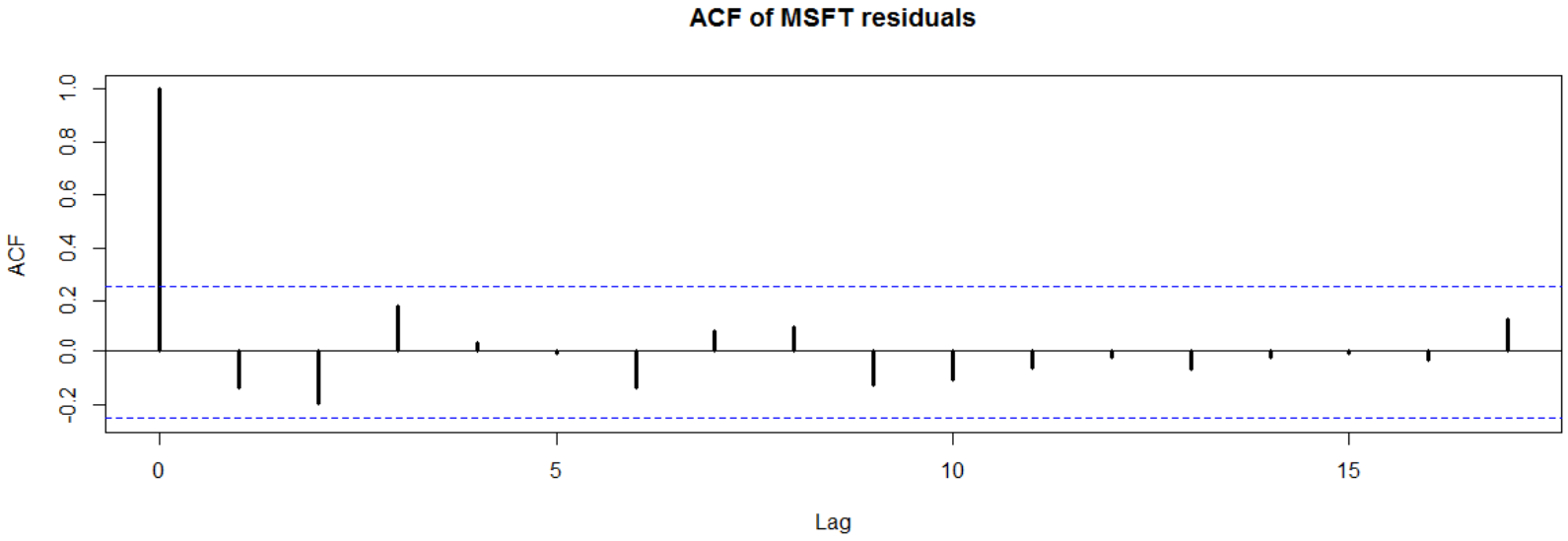






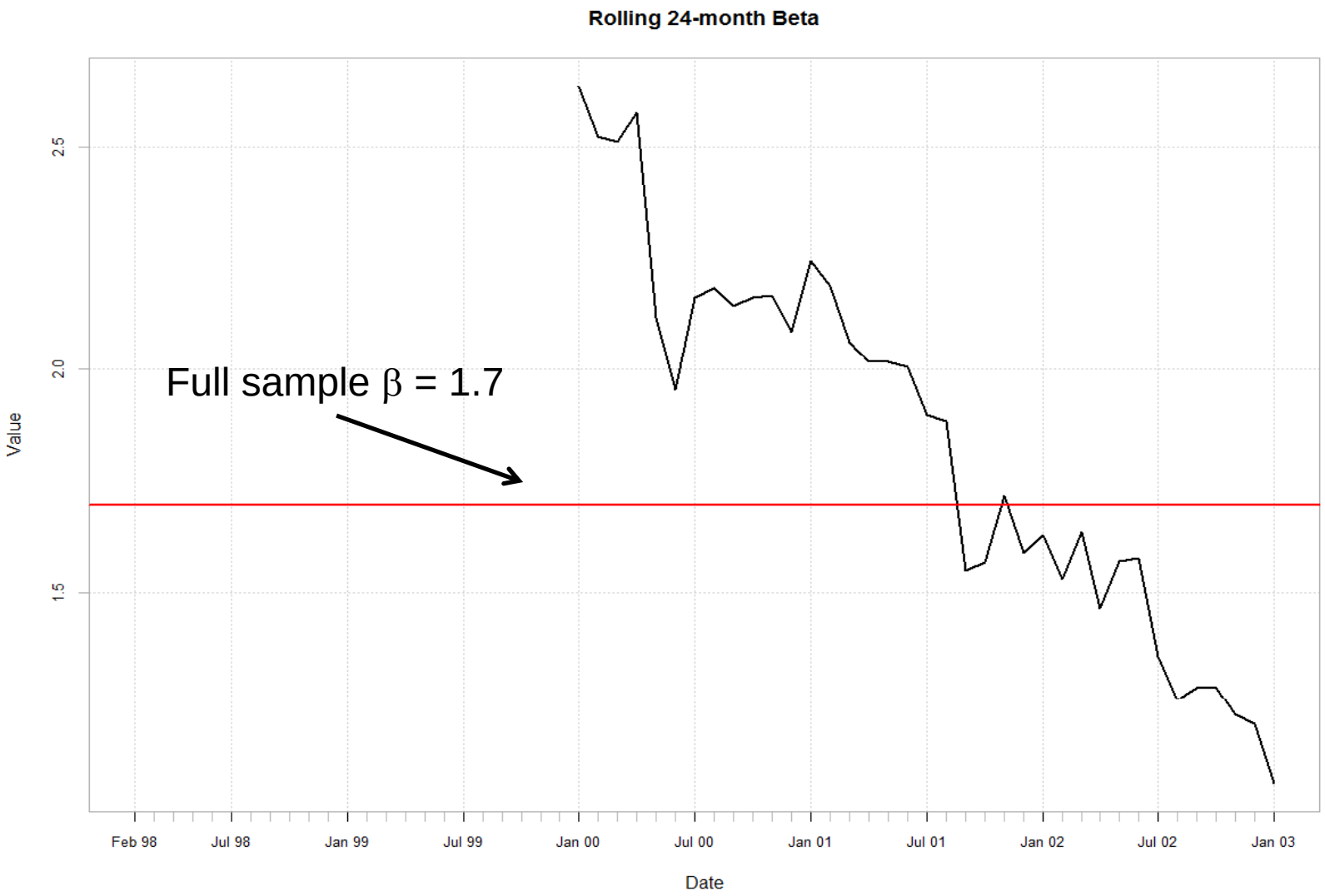
In the Single Index model the errors should be Normally Distributed



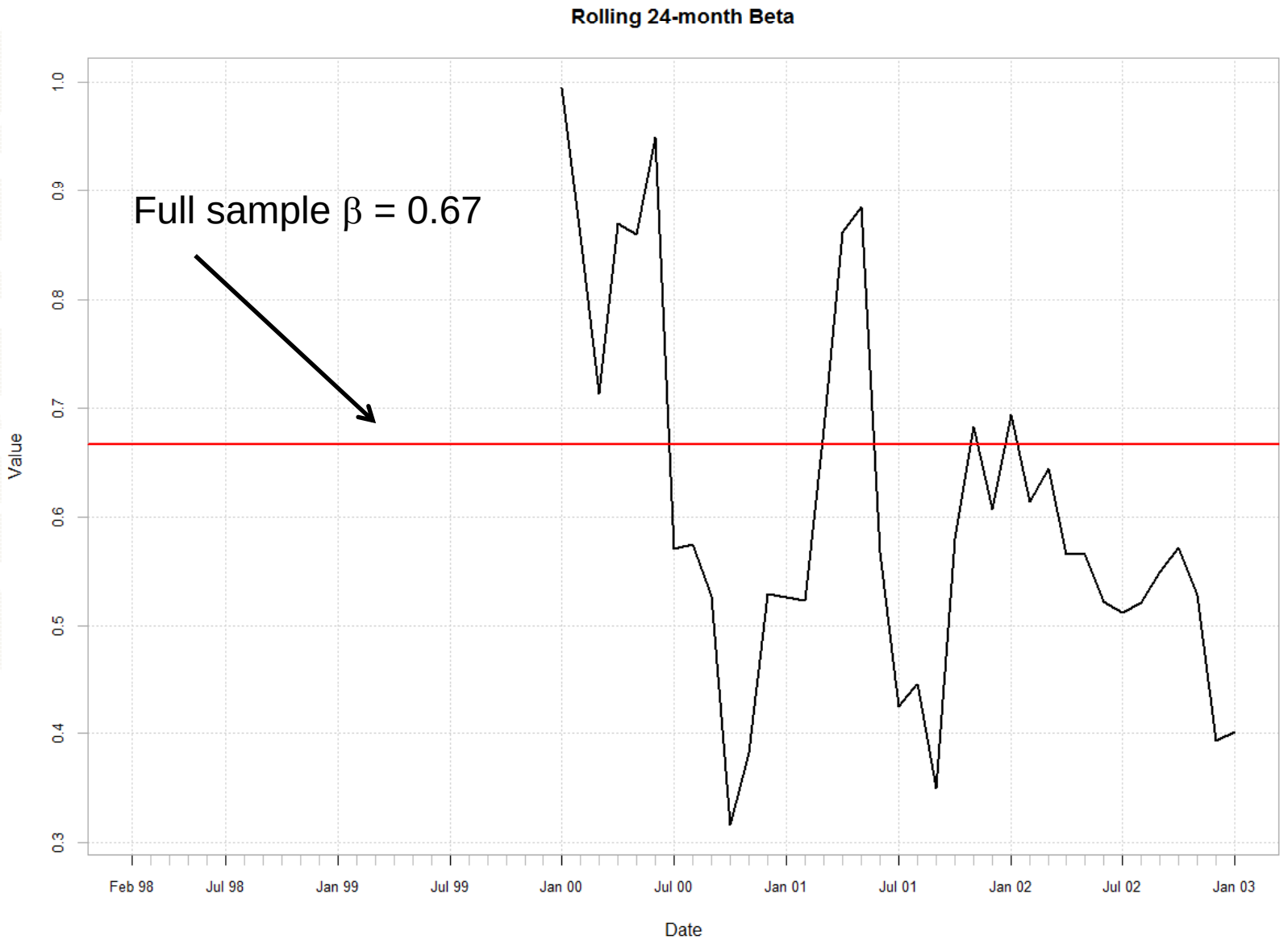


In the Single Index model the errors should behave like White Noise

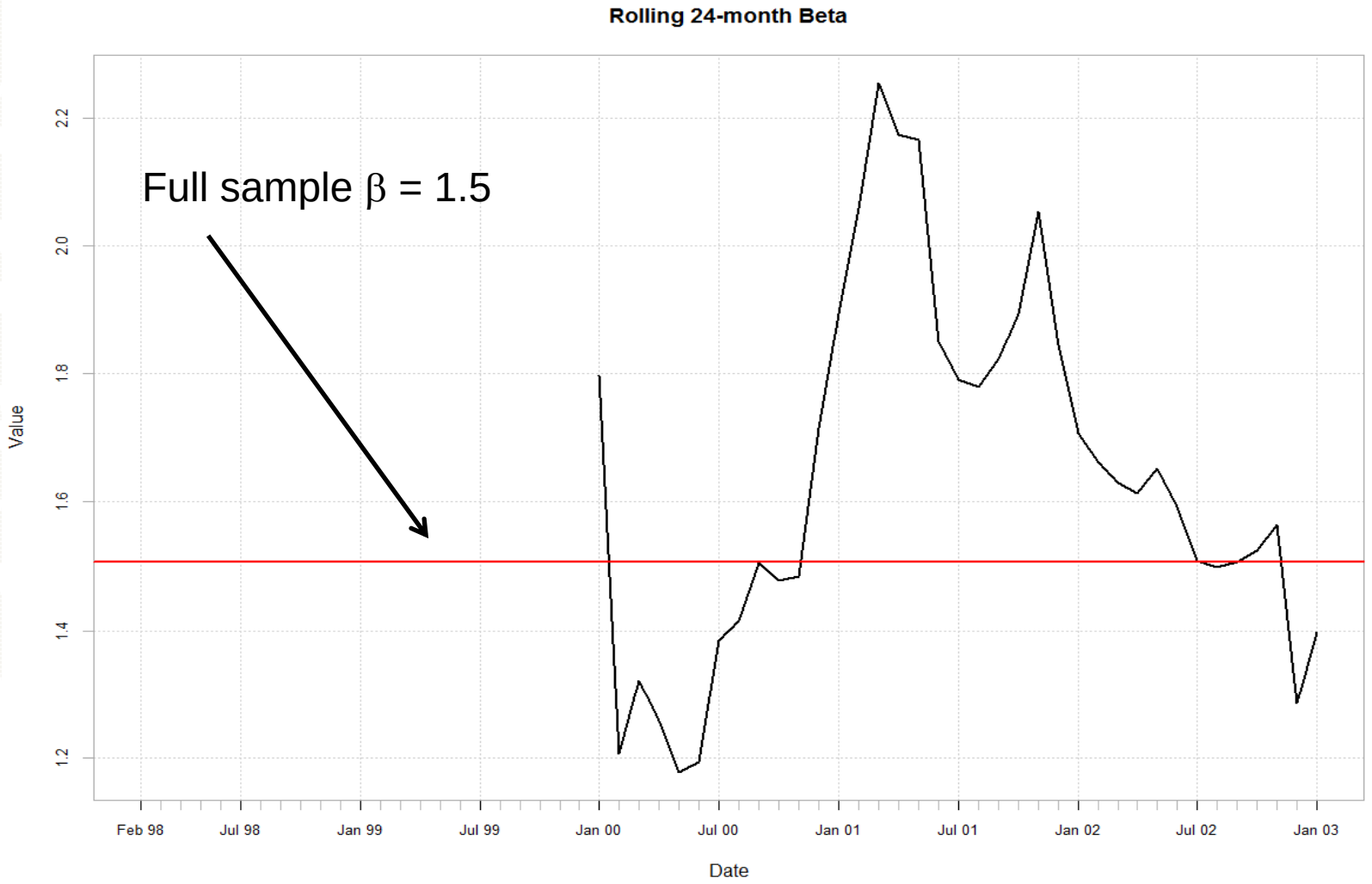
24 month rolling β's for MSFT



24 month rolling β's for SBUX



24 month rolling β's for NORD



24 month rolling β's for Boeing