CAPM v. Dynamic CAPM Using the Kalman Filter

Code

Casey Tirshfield

The file `m_sp500ret_3mtcm.txt` contains three columns. The second column gives the monthly returns of the S&P500 index from January 1994 to December 2006. The third column gives the monthly rates of the 3-month U.S. Treasury bill in the secondary market, which are obtained from the Federal Reserve Bank of St. Louis and used as the risk-free rate here. Consider the ten monthly log returns in the file `m_logret_10stocks.txt`.

Hide

```
# data frames
df <- read.table('m_sp500ret_3mtcm.txt', skip=1, header=TRUE)
logret <- read.table('m_logret_10stocks.txt', header=TRUE)
# make dates in date columns a date type
df$Date <- as.Date(paste('01', df$Date, sep='-'), '%d-%b-%y')
logret$Date <- as.Date(logret$Date, '%m/%d/%Y')
# splits data frames according to date
df_split <- split(df, df$Date<as.Date('1998-07-01'))
logret_split <- split(logret, logret$Date<as.Date('1998-07-01'))
# create new data frame for time period
df_per1 <- as.data.frame(df_split[2])
df_per2 <- as.data.frame(df_split[1])
logret_per1 <- as.data.frame(logret_split[2])
logret_per2 <- as.data.frame(logret_split[1])
# risk free rate columns
rfr <- df[,3] / (12 * 100)
rfr_per1 <- df_per1[,3] / (12 * 100)
rfr_per2 <- df_per2[,3] / (12 * 100)
# excessive returns matrices
exc_logret <- apply(logret[,-1], 2, function(x){x - rfr})
exc_logret_per1 <- apply(logret_per1[,-1], 2, function(x){x - rfr_per1})
exc_logret_per2 <- apply(logret_per2[,-1], 2, function(x){x - rfr_per2})
# excessive return columns for the S&P 500
exc_sp <- df[,2] - rfr
exc_sp_per1 <- df_per1[,2] - rfr_per1
exc_sp_per2 <- df_per2[,2] - rfr_per2
```

(a) For each stock, fit CAPM for the period from January 1994 to June 1998 and for the subsequent period from July 1998 to December 2006. Are your estimated betas significantly different for the two periods?

Hide

```
model <- lm(exc_logret ~ exc_sp)
model_per1 <- lm(exc_logret_per1 ~ exc_sp_per1)
model_per2 <- lm(exc_logret_per2 ~ exc_sp_per2)
alpha <- coef(model)[1,]
beta <- coef(model)[2,]
alpha_per1 <- coef(model_per1)[1,]
beta_per1 <- coef(model_per1)[2,]
alpha_per2 <- coef(model_per2)[1,]
beta_per2 <- coef(model_per2)[2,]
# formatting for printing to consol
alphabeta <- matrix(c(alpha, beta), nrow=10, ncol=2)
rownames(alphabeta) <- c(names(logret[,-1]))
colnames(alphabeta) <- c('alphas', 'betas')
print('The alphas and betas for the entire period are:')
```

```
[1] "The alphas and betas for the entire period are:"
```

Hide

```
print(alphabeta)
```

```
            alphas     betas
AAPL  0.0038473784 1.3846398
ADBE  0.0046581721 1.5313505
ADP   0.0008070075 0.8476769
AMD  -0.0006186328 2.3238266
DELL  0.0088838584 1.6749919
GTW  -0.0054253352 2.2328015
HP    0.0019004141 0.8752343
IBM   0.0026256227 1.3479235
MSFT  0.0042555493 1.4585386
ORCL  0.0039104996 1.5676042
```

Hide

```
alphabeta_per1 <- matrix(c(alpha_per1, beta_per1), nrow=10, ncol=2)
rownames(alphabeta_per1) <- c(names(logret[,-1]))
colnames(alphabeta_per1) <- c('alphas', 'betas')
print('The alphas and betas for the period from January 1994 to June 1998 are:')
```

```
[1] "The alphas and betas for the period from January 1994 to June 1998 are:"
```

Hide

```
print(alphabeta_per1)
```

```
           alphas     betas
AAPL -0.005887987 0.5980913
ADBE -0.002317855 1.2538262
ADP   0.002182129 0.6155901
AMD  -0.007045352 0.8600476
DELL  0.024414993 1.7210087
GTW   0.005033032 1.3032995
HP   -0.001619345 0.7814124
IBM   0.004102624 1.1388893
MSFT  0.011237136 1.2412122
ORCL  0.001573735 0.9305611
```

Hide

```
alphabeta_per2 <- matrix(c(alpha_per2, beta_per2), nrow=10, ncol=2)
rownames(alphabeta_per2) <- c(names(logret[,-1]))
colnames(alphabeta_per2) <- c('alphas', 'betas')
print('The alphas and betas for the period from July 1998 to December 2006 are:')
```

```
[1] "The alphas and betas for the period from July 1998 to December 2006 are:"
```

Hide

```
print(alphabeta_per2)
```

```
            alphas     betas
AAPL  0.0107874665 1.6819206
ADBE  0.0090243112 1.6601368
ADP   0.0005547309 0.9067323
AMD   0.0059664982 2.7981819
DELL  0.0003763540 1.5566408
GTW  -0.0091166136 2.4359083
HP    0.0040053405 0.9266097
IBM   0.0022692773 1.3995039
MSFT  0.0009361403 1.4754164
ORCL  0.0065272008 1.7709404
```

Hide

```
# calculate the difference of the estimated betas for the two periods
diff_beta <- beta_per1 - beta_per2
# formatting for printing to consol
difference <- matrix(c(beta_per1, beta_per2, diff_beta), nrow=10, ncol=3)
rownames(difference) <- c(names(logret[,-1]))
colnames(difference) <- c('period I', 'period II', 'difference')
print('The difference between the estimated betas of period one and period two are:')
```

```
[1] "The difference between the estimated betas of period one and period two are:"
```

Hide

```
print(difference)
```

```
      period I period II difference
AAPL 0.5980913 1.6819206 -1.0838292
ADBE 1.2538262 1.6601368 -0.4063107
ADP  0.6155901 0.9067323 -0.2911422
AMD  0.8600476 2.7981819 -1.9381342
DELL 1.7210087 1.5566408  0.1643679
GTW  1.3032995 2.4359083 -1.1326088
HP   0.7814124 0.9266097 -0.1451973
IBM  1.1388893 1.3995039 -0.2606146
MSFT 1.2412122 1.4754164 -0.2342042
ORCL 0.9305611 1.7709404 -0.8403793
```

(b) Consider the dynamic linear model $r_t - f_{f,t} = \beta_t(r_{M,t} - r_{f,t}) + \epsilon_t, \quad \beta_{t+1} = \beta_t + \omega_{t+1}$ with independent $\epsilon_t \sim \mathcal{N}(0, \sigma^2)$ and $\omega_t \sim \mathcal{N}(0, \sigma_\omega^2)$, for CAPM with timevarying betas. Use the Kalman filter with $\sigma_\omega = 0.2$ to estimate β_t sequentially during the period July 1998-December 2006. The estimated beta $\hat{\beta}$ and error variance $\hat{\sigma}^2$ obtained in (a) for the period from January 1994 to June 1998 can be used to initialize $\hat{\beta}_0$ and to substitute for σ^2 in the Kalman Filter. Plot, compare, and discuss your sequential estimates with the estimate of beta in (a) for the period July 1998 to December 2006.

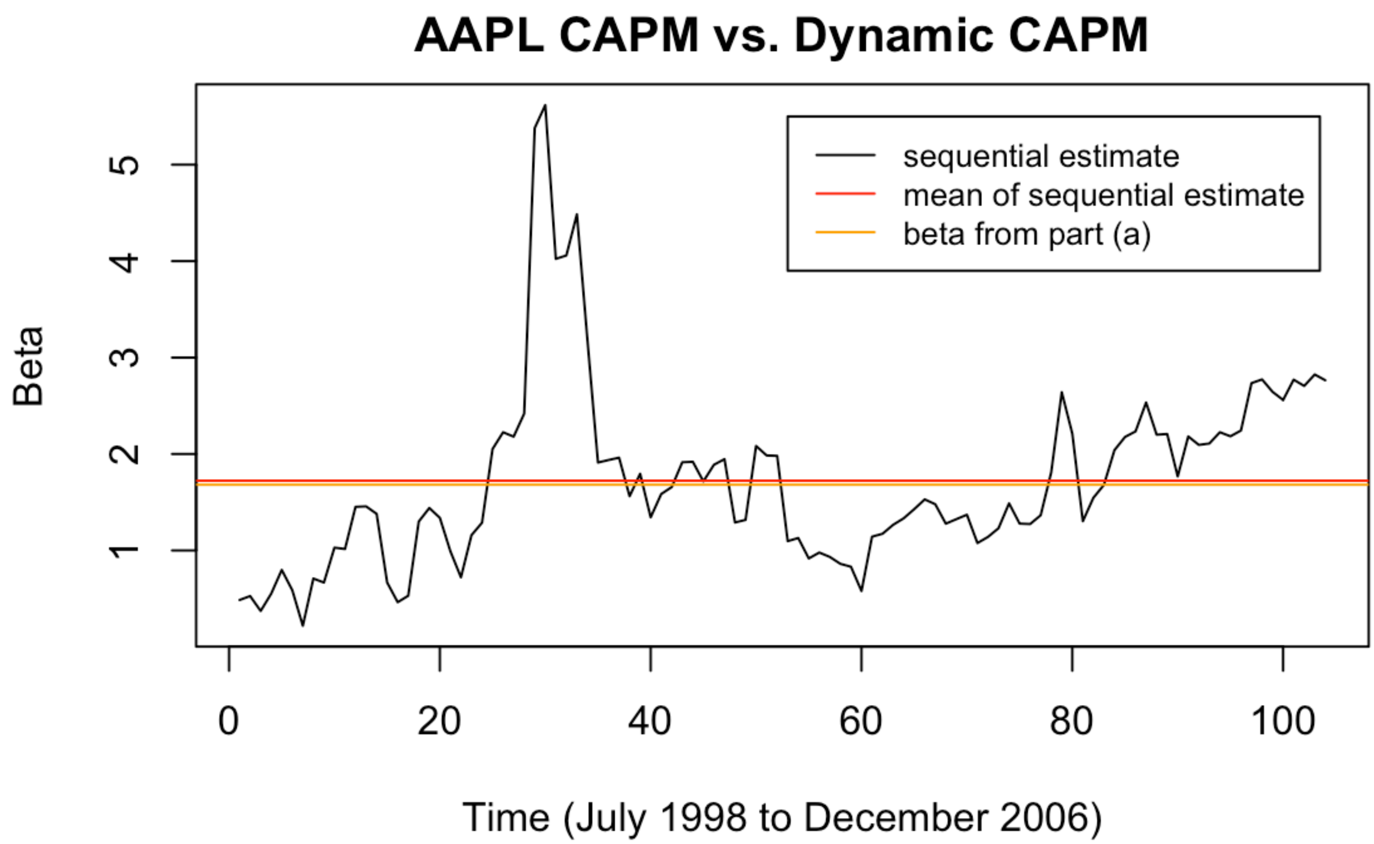
Hide

```
# this code is inspired by the book's code found at the following link: https://web.s
tanford.edu/~xing/statfinbook/_BookFun/ex5.3.2_kalman_capm.txt
kalmanf.update <- function(y, g, xt.t, Pt.t, F, sig2W, sig2v){
  Pta.t <- F * F * Pt.t + sig2w
  Pta.ta <- Pta.t - (Pta.t * g) ^ 2 / (g * g * Pta.t + sig2v)
  xta.t <- F * xt.t
  xta.ta <- xta.t + Pta.t * g / (g * g * Pta.t + sig2v) * (y - g * xta.t)
  list(xt.t=xta.ta, Pt.t=Pta.ta)
}
kalmanf.estx <- function(y, G, G0, fit0, sig2w) {
```

```
  est.x <- est.P <- rep(0, length(y) + 1)
  sig2v <- sum(fit0$resid ^ 2) / length(fit0$resid)
  est.x[1] <- xt.t <- as.numeric(fit0$coeff)
  est.P[1] <- Pt.t <- sum(fit0$resid ^ 2) / (length(fit0$resid) - 1) / sum(G0 ^ 2)
  F <- 1
  for (i in 1:length(y)) {
    kalmanf <- kalmanf.update(y[i], G[i], xt.t, Pt.t, F, sig2w, sig2v)
    est.x[i + 1] <- xt.t <- kalmanf$xt.t
    est.P[i + 1] <- Pt.t <- kalmanf$Pt.t
  }
  list(est.x = est.x, est.P = est.P)
}
ts.beta<-function(y, G, seg0, sig2w) {
  fit0 <- lm(y[seg0] ~ G[seg0] - 1)
  est <- kalmanf.estx(y[-seg0], G[-seg0], G[seg0], fit0, sig2w)
  est
}
#   learning period = 53 month
seg0 <- seq(1,53)
sig2w <- 0.2
est.AAPL <- ts.beta(exc_logret[,1], exc_sp, seg0, sig2w)
est.ADBE <- ts.beta(exc_logret[,2], exc_sp, seg0, sig2w)
est.ADP <- ts.beta(exc_logret[,3], exc_sp, seg0, sig2w)
est.AMD <- ts.beta(exc_logret[,4], exc_sp, seg0, sig2w)
est.DELL <- ts.beta(exc_logret[,5], exc_sp, seg0, sig2w)
est.GTW <- ts.beta(exc_logret[,6], exc_sp, seg0, sig2w)
est.HP <- ts.beta(exc_logret[,7], exc_sp, seg0, sig2w)
est.IBM <- ts.beta(exc_logret[,8], exc_sp, seg0, sig2w)
est.MSFT <- ts.beta(exc_logret[,9], exc_sp, seg0, sig2w)
est.ORCL <- ts.beta(exc_logret[,10], exc_sp, seg0, sig2w)
est.AAPL_df <- as.data.frame(est.AAPL)
est.ADBE_df <- as.data.frame(est.ADBE)
est.ADP_df <- as.data.frame(est.ADP)
est.AMD_df <- as.data.frame(est.AMD)
est.DELL_df <- as.data.frame(est.DELL)
est.GTW_df <- as.data.frame(est.GTW)
est.HP_df <- as.data.frame(est.HP)
est.IBM_df <- as.data.frame(est.IBM)
est.MSFT_df <- as.data.frame(est.MSFT)
est.ORCL_df <- as.data.frame(est.ORCL)
ts.plot(est.AAPL_df$est.x, xlab='Time (July 1998 to December 2006)', ylab='Beta', mai
n='AAPL CAPM vs. Dynamic CAPM')
abline(h=1.6819206, col='orange')
```

Hide

```
abline(h=mean(est.AAPL_df$est.x), col='red')
legend(53, 5.5, legend=c('sequential estimate', 'mean of sequential estimate', 'beta
from part (a)'), col=c('black', 'red', 'orange'), lty=1:1, cex=0.8)
```

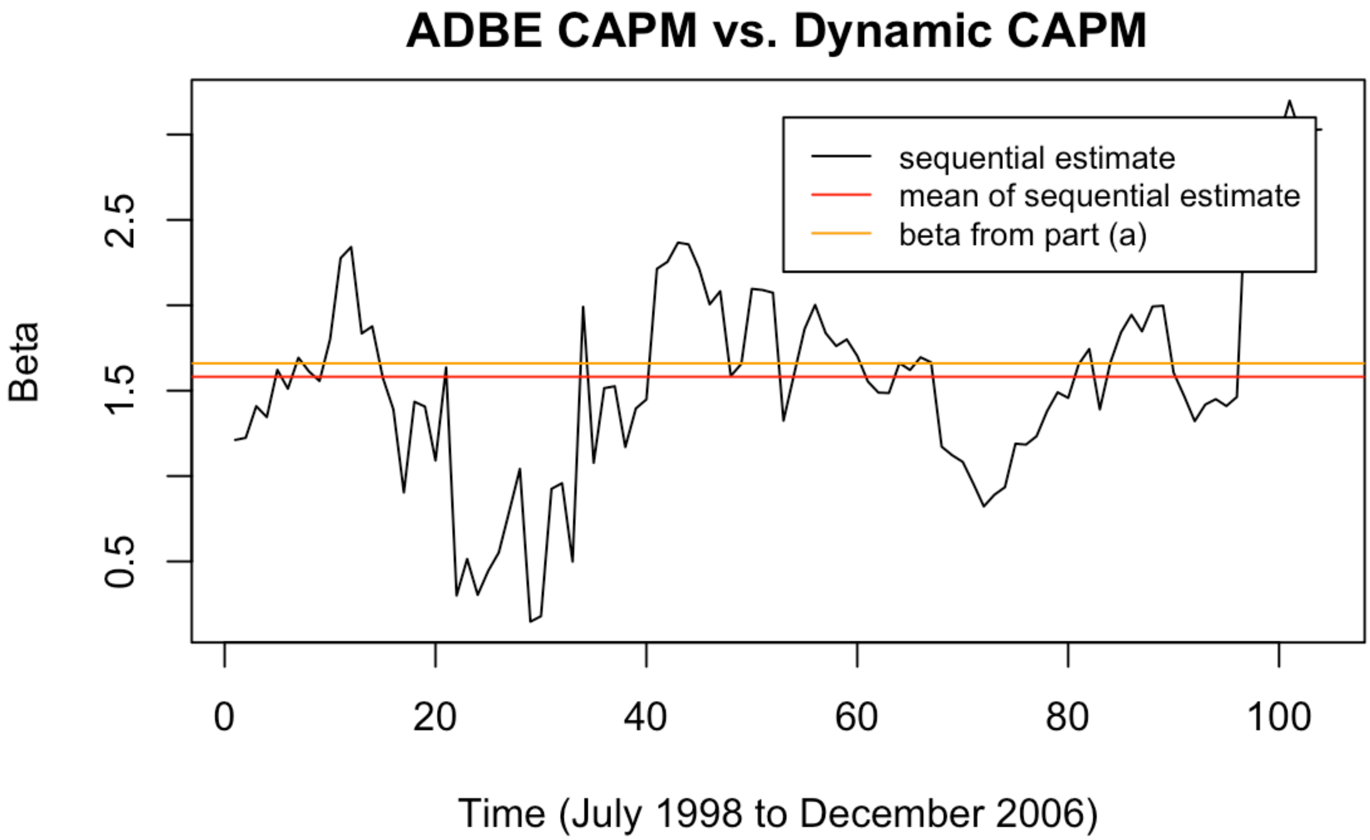


Hide

```
ts.plot(est.ADBE_df$est.x, xlab='Time (July 1998 to December 2006)', ylab='Beta', mai
n='ADBE CAPM vs. Dynamic CAPM')
abline(h=1.6601368, col='orange')
```

Hide

```
abline(h=mean(est.ADBE_df$est.x), col='red')
legend(53, 3.1, legend=c('sequential estimate', 'mean of sequential estimate', 'beta
from part (a)'), col=c('black', 'red', 'orange'), lty=1:1, cex=0.8)
```

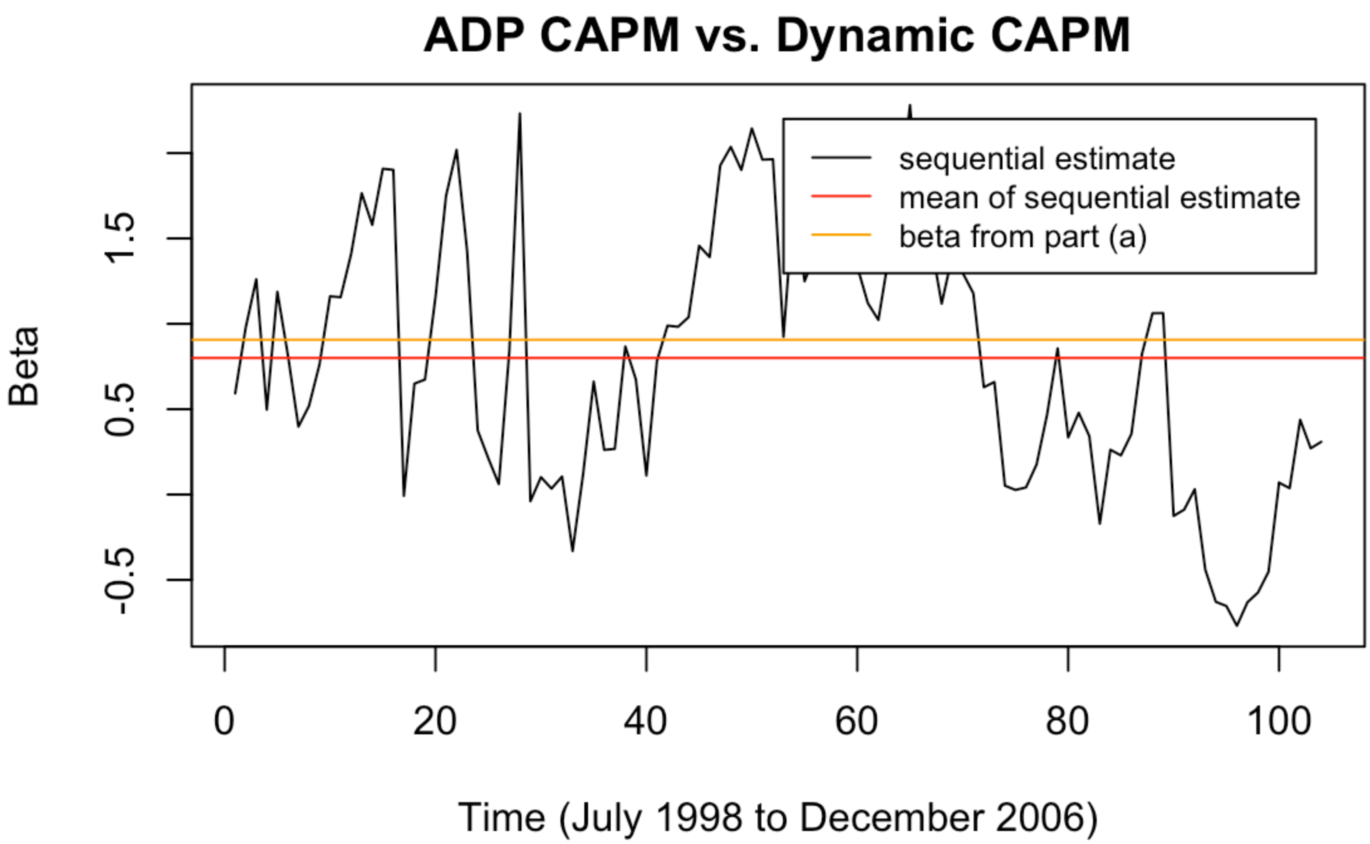


Hide

```
ts.plot(est.ADP_df$est.x, xlab='Time (July 1998 to December 2006)', ylab='Beta', main
='ADP CAPM vs. Dynamic CAPM')
abline(h=0.9067323, col='orange')
```

Hide

```
abline(h=mean(est.ADP_df$est.x), col='red')
legend(53, 2.2, legend=c('sequential estimate', 'mean of sequential estimate', 'beta
from part (a)'), col=c('black', 'red', 'orange'), lty=1:1, cex=0.8)
```

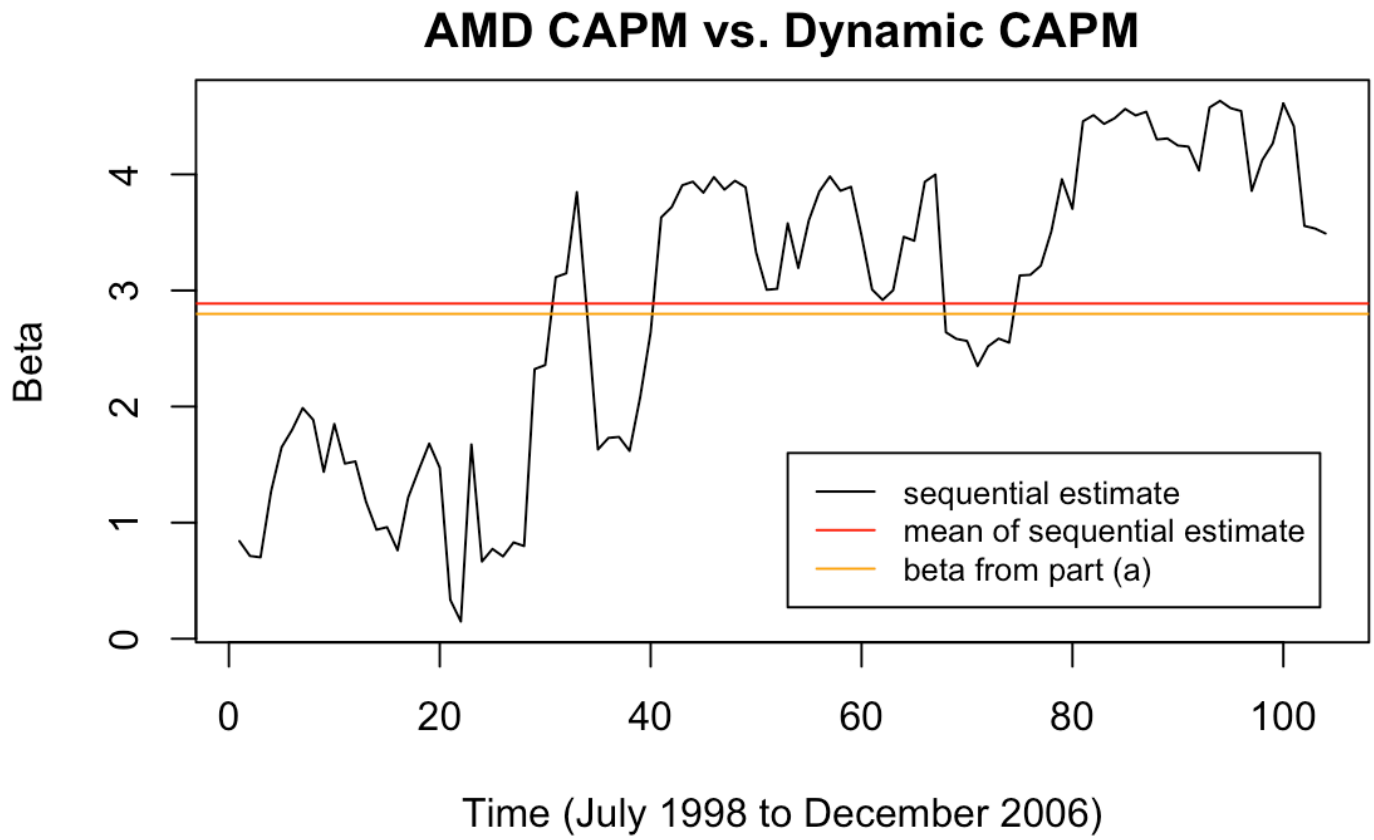


Hide

```
ts.plot(est.AMD_df$est.x, xlab='Time (July 1998 to December 2006)', ylab='Beta', main
='AMD CAPM vs. Dynamic CAPM')
abline(h=2.7981819, col='orange')
```

Hide

```
abline(h=mean(est.AMD_df$est.x), col='red')
legend(53, 1.6, legend=c('sequential estimate', 'mean of sequential estimate', 'beta
from part (a)'), col=c('black', 'red', 'orange'), lty=1:1, cex=0.8)
```

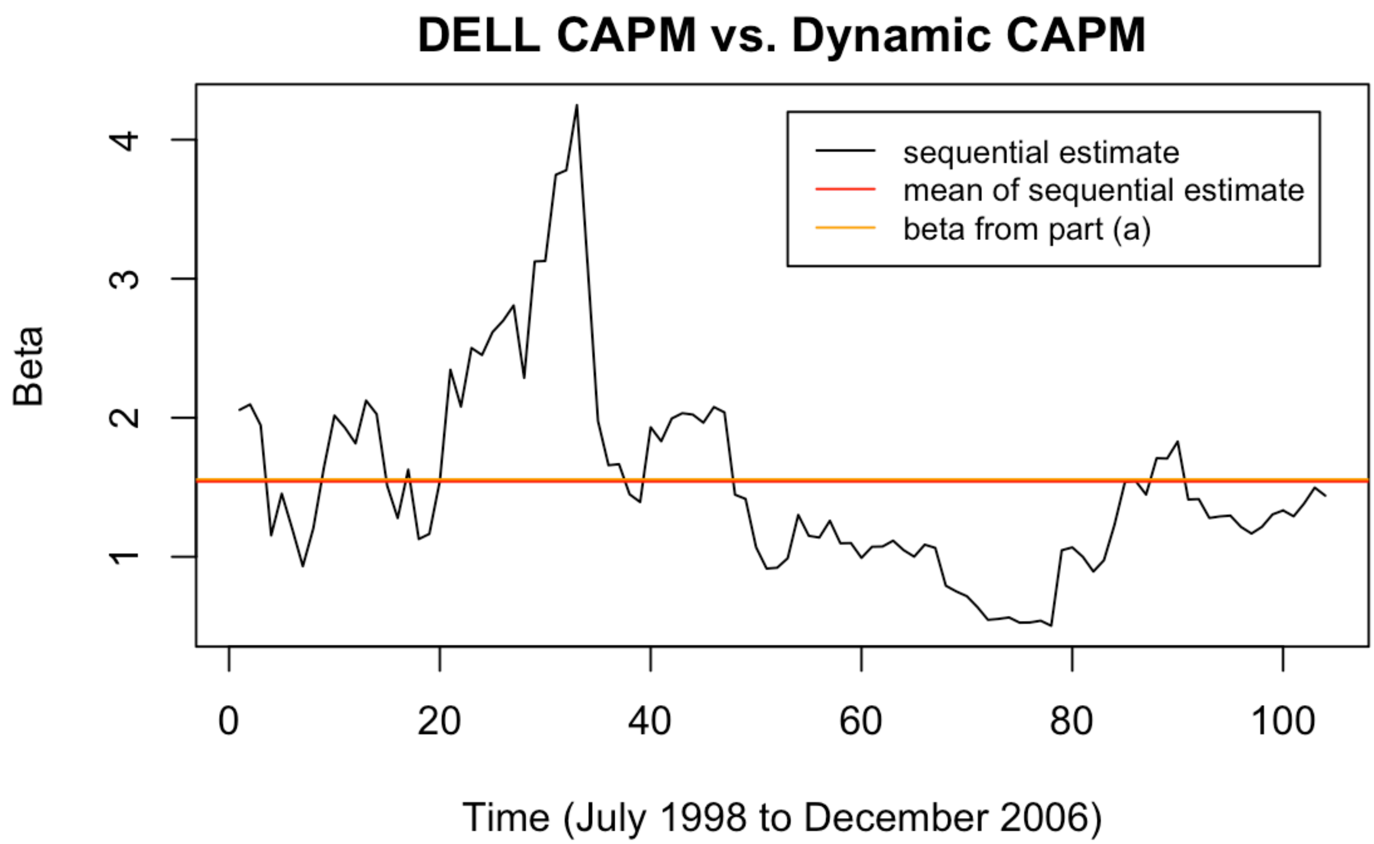


Hide

```
ts.plot(est.DELL_df$est.x, xlab='Time (July 1998 to December 2006)', ylab='Beta', mai
n='DELL CAPM vs. Dynamic CAPM')
abline(h=1.5566408, col='orange')
```

Hide

```
abline(h=mean(est.DELL_df$est.x), col='red')
legend(53, 4.2, legend=c('sequential estimate', 'mean of sequential estimate', 'beta
from part (a)'), col=c('black', 'red', 'orange'), lty=1:1, cex=0.8)
```

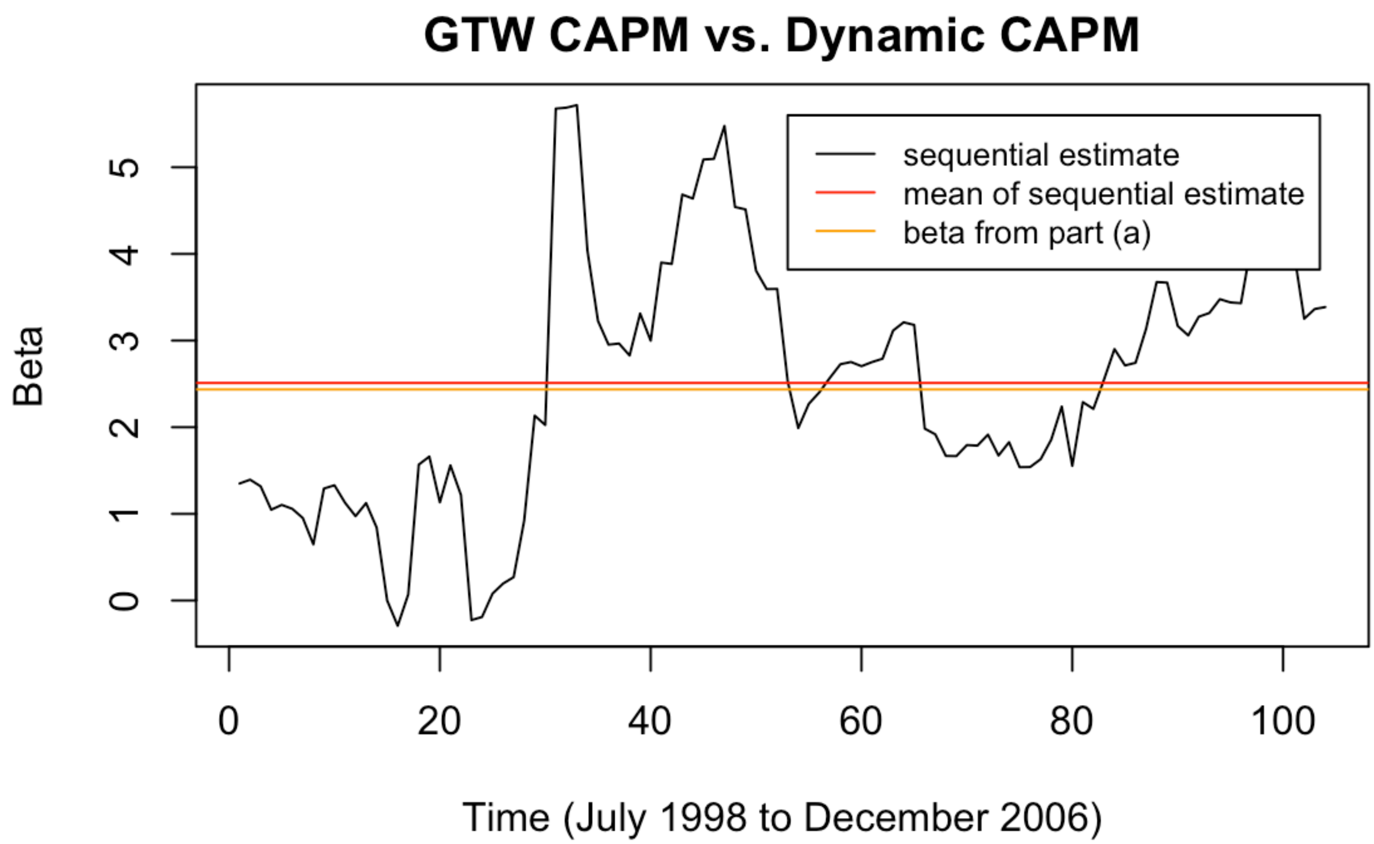


Hide

```
ts.plot(est.GTW_df$est.x, xlab='Time (July 1998 to December 2006)', ylab='Beta', main
='GTW CAPM vs. Dynamic CAPM')
abline(h=2.4359083, col='orange')
```

Hide

```
abline(h=mean(est.GTW_df$est.x), col='red')
legend(53, 5.6, legend=c('sequential estimate', 'mean of sequential estimate', 'beta
from part (a)'), col=c('black', 'red', 'orange'), lty=1:1, cex=0.8)
```

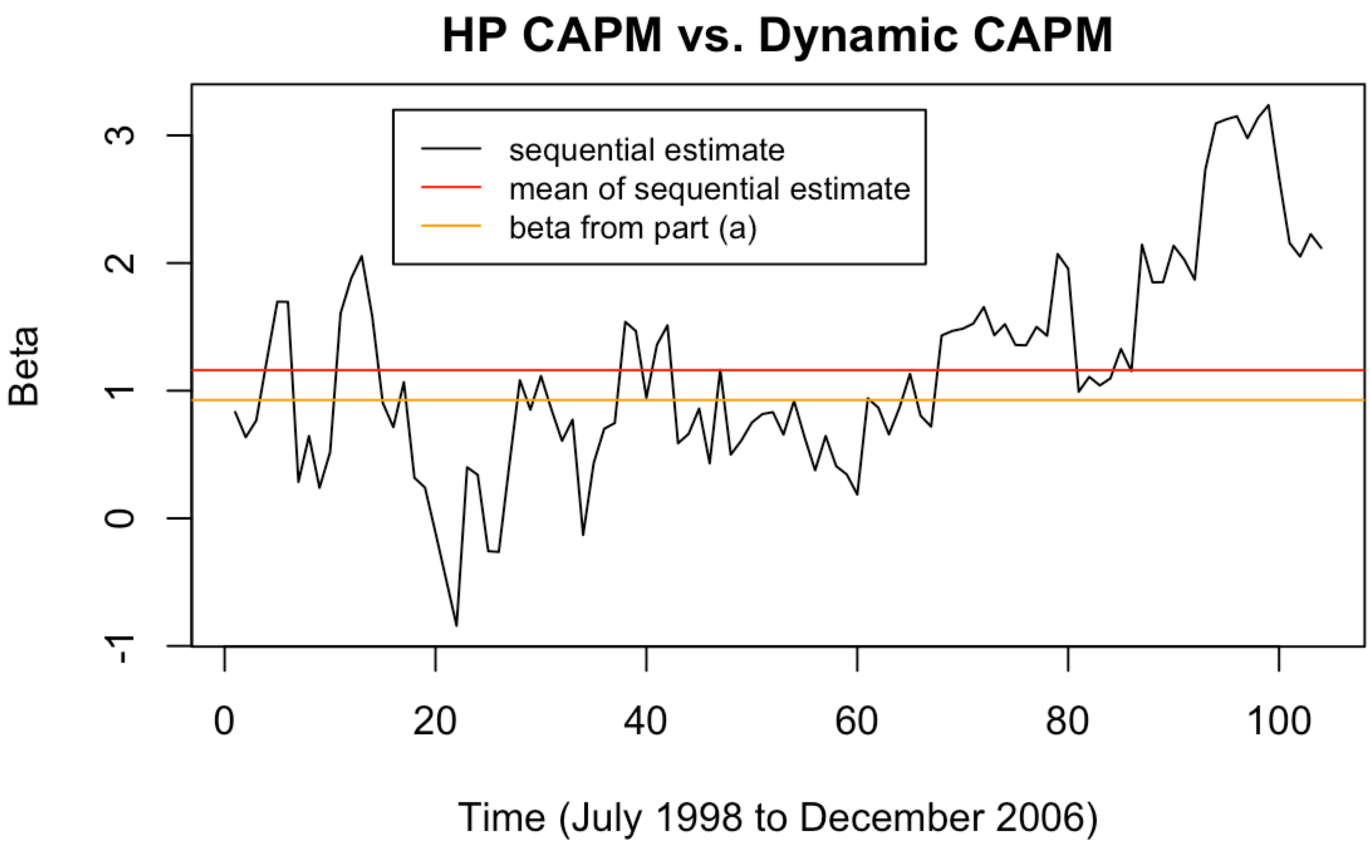


Hide

```
ts.plot(est.HP_df$est.x, xlab='Time (July 1998 to December 2006)', ylab='Beta', main=
'HP CAPM vs. Dynamic CAPM')
abline(h=0.9266097, col='orange')
```

Hide

```
abline(h=mean(est.HP_df$est.x), col='red')
legend(16, 3.2, legend=c('sequential estimate', 'mean of sequential estimate', 'beta
from part (a)'), col=c('black', 'red', 'orange'), lty=1:1, cex=0.8)
```

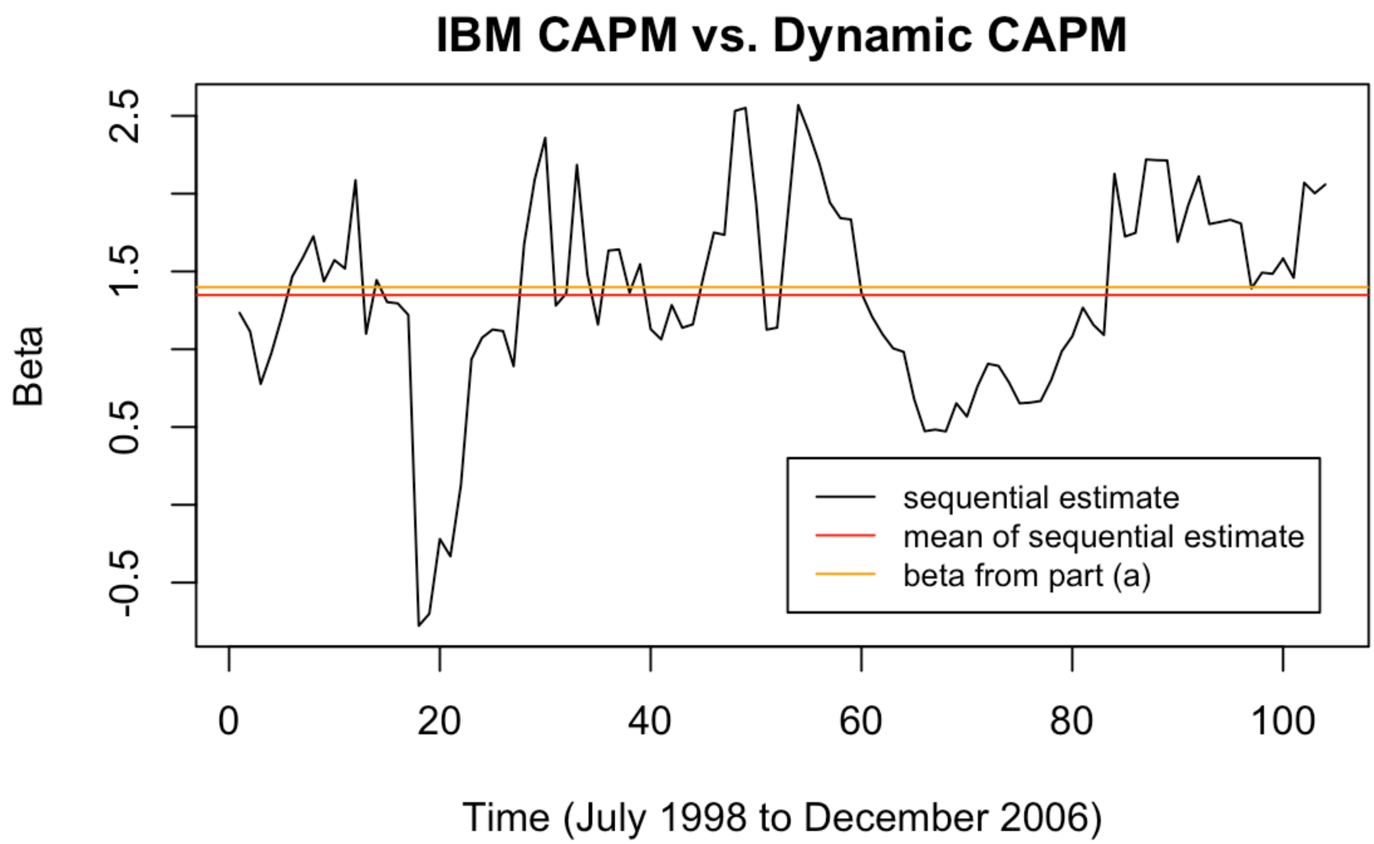


Hide

```
ts.plot(est.IBM_df$est.x, xlab='Time (July 1998 to December 2006)', ylab='Beta', main
='IBM CAPM vs. Dynamic CAPM')
abline(h=1.3995039, col='orange')
```

Hide

```
abline(h=mean(est.IBM_df$est.x), col='red')
legend(53, .3, legend=c('sequential estimate', 'mean of sequential estimate', 'beta f
rom part (a)'), col=c('black', 'red', 'orange'), lty=1:1, cex=0.8)
```

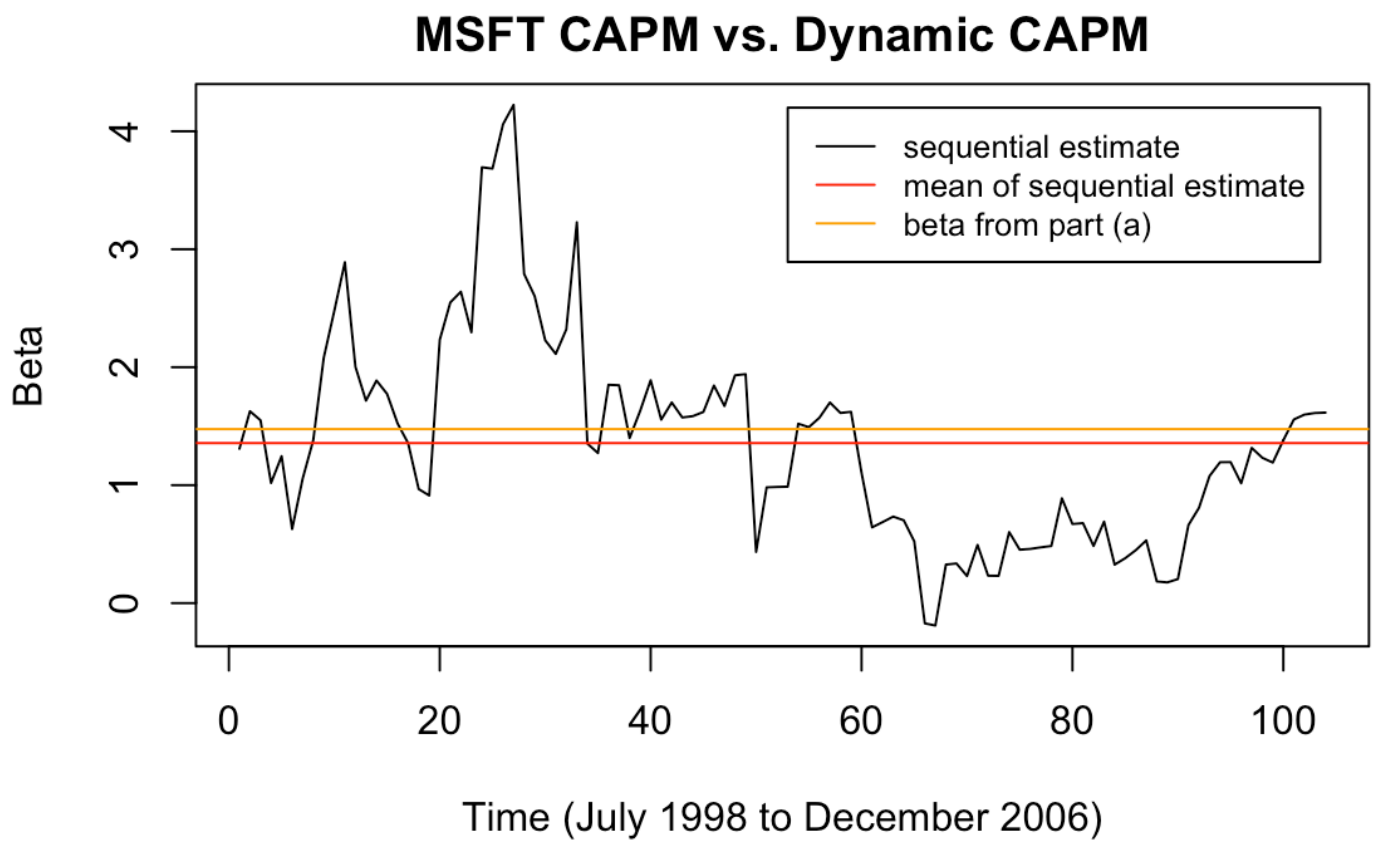


Hide

```
ts.plot(est.MSFT_df$est.x, xlab='Time (July 1998 to December 2006)', ylab='Beta', mai
n='MSFT CAPM vs. Dynamic CAPM')
abline(h=1.4754164, col='orange')
```

Hide

```
abline(h=mean(est.MSFT_df$est.x), col='red')
legend(53, 4.2, legend=c('sequential estimate', 'mean of sequential estimate', 'beta
from part (a)'), col=c('black', 'red', 'orange'), lty=1:1, cex=0.8)
```

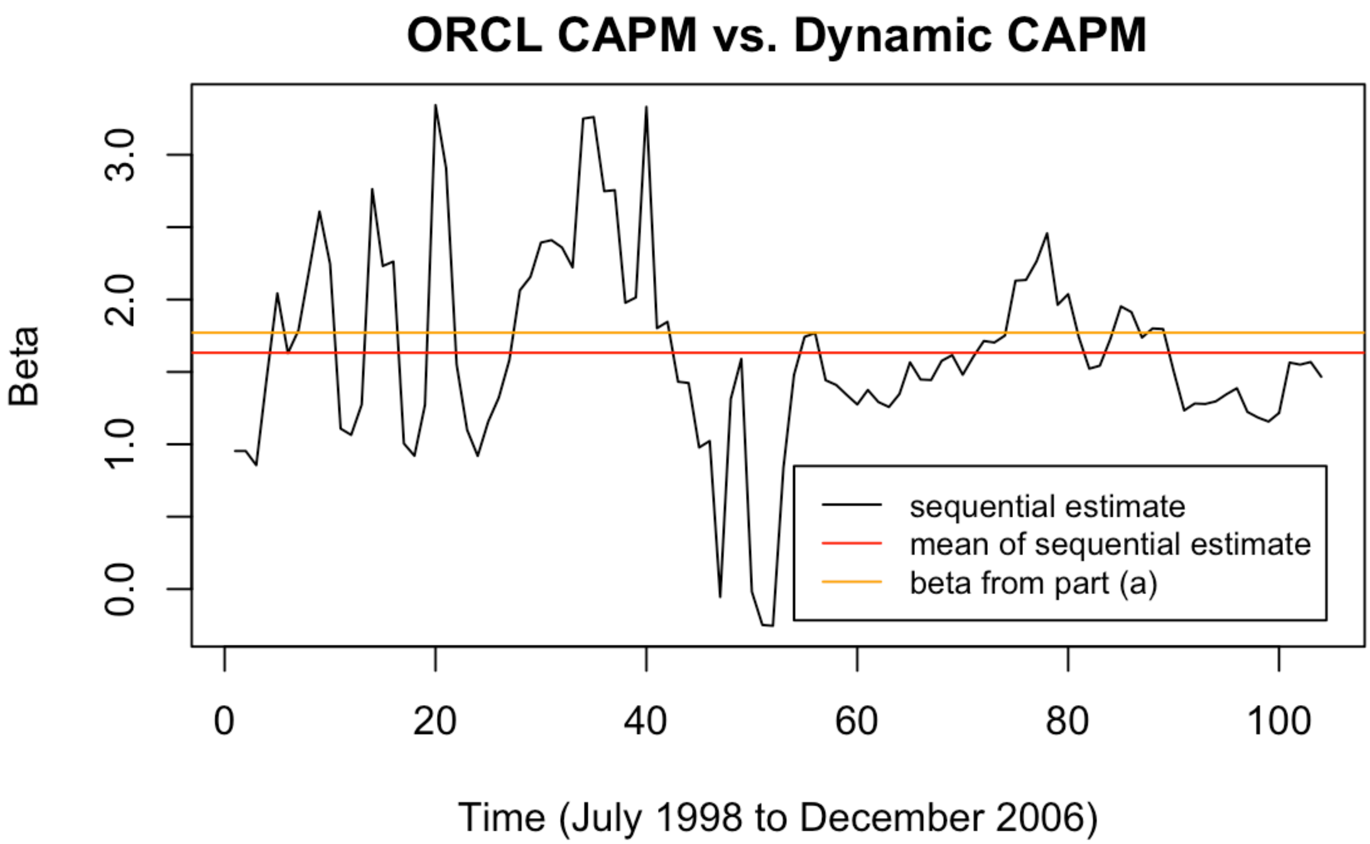


Hide

```
ts.plot(est.ORCL_df$est.x, xlab='Time (July 1998 to December 2006)', ylab='Beta', mai
n='ORCL CAPM vs. Dynamic CAPM')
abline(h=1.7709404, col='orange')
```

Hide

```
abline(h=mean(est.ORCL_df$est.x), col='red')
legend(54, .85, legend=c('sequential estimate', 'mean of sequential estimate', 'beta
from part (a)'), col=c('black', 'red', 'orange'), lty=1:1, cex=0.8)
```



Hide

```
print('From the above plots, we can see that the mean of the timevarying betas is qui
te close to the betas we obtained in part (a).')
```

```
[1] "From the above plots, we can see that the mean of the timevarying betas is quite
close to the betas we obtained in part (a)."
```