".First" = function()

{

module(finmetrics)

}

"Call1" = function()

{

my.list <- Call2()

my.list

}

"Call2" = function()

{

xm <- matrix(1:12, 3, 4)

xv <- vector("double", 3)

xv <- xm[, 3]

local.list <- list(xm, xv)

}

"my.plots" = function()

{

par(mfrow = c(1, 2))

plot(play.df$V1, play.df$V2)

plot(play.df$V2, play.df$V3)

}

"pickClose" = function(x)

{

# Taken From Zivot and Wang page 32

# return closing values of vector

if(length(dim(x))) x = as.vector(as.matrix(x))

len = length(x)

if(!len)

as(NA, class(x))

else x[len]

}

"sim.reg" = function(n = 50, k = 5, reps = 1000, seed = 30)

{

set.seed(seed)

x <- matrix(runif(n \* k), n, k)

betas <- 20 + 15 \* runif(k + 1)

x <- cbind(1, x)

y <- x %\*% betas + rnorm(n)

#

#

print(betas)

ols.fit <- lm(y ~ (-1) + x)

print(summary(ols.fit))

#

#single run

#

cat("\n\n Single Run \n")

cat("\n Population Estimate \n")

for(i in 1:(k + 1)) {

cat("\n Beta[", i, "]: ", ols.fit$coefficients[i], " ", betas[i])

}

#

# Estimated Beta distributions - fixed betas - vary X and e

#

Betas <- matrix(0, reps, (k + 1))

index = 1

while(index <= reps) {

x <- matrix(runif(n \* k), n, k)

x <- cbind(1, x)

y <- x %\*% betas + rnorm(n)

ols.fit <- lm(y ~ (-1) + x)

Betas[index, ] <- ols.fit$coefficients

index <- index + 1

}

par(mfrow = c(3, 3))

cat("\n\n\n Shapiro Wilks Tests of Normaality \n")

for(i in 1:(k + 1)) {

hist(Betas[, i])

qqnorm(Betas[, i])

cat("\n beta[",(i-1),"] =" ,shapiro.test(Betas[,i])$p.value)

}

cat("\n")

}

"training" = function(local.df = play.df)

{

# THIS IS NOT A WORKING FUNCTION

# IT IS JUST A HOLDING PLACE FOR ISSUES

# TO BE COVERED IN TRAINING CLASS

x <- 10

y <- seq(1, 10)

z <- rep(5, 10)

a <- x + y

b <- x \* y

c <- y - z

#y %\*% t(z)

#t(y) %\*% z

X <- rep(1, nrow(local.df))

X <- cbind(X, local.df$X)

ols.fit <- solve(t(X) %\*% X) %\*% t(X) %\*% local.df$Y

subSmpl <- play.df[local.df$Y > 0, ]

subSmpl2 <- play.df[3:10, 2:4]

par(mfrow = c(1, 1))

plot(local.df$V1, local.df$V2)

{

my.plots <- function(toplot.df = local.df)

{

par(mfrow = c(1, 2))

plot(toplot.df$V2, toplot.df$V3)

plot(toplot.df$V3, toplot.df$V4)

}

}

my.plots(local.df)

play.ts <- timeSeries(local.df)

play.ts@positions <- 1:118

nr.play.ts <- numRows(play.ts)

play.ts@positions <- rep(NA, nr.play.ts)

j.date.start <- timeDate("1/1/1990")

j.date.start + nr.play.ts - 1

play.ts@positions <- timeSeq(from = j.date.start, to = j.date.end, by = "days")

play.month.ts <- aggregateSeries(play.ts, FUN = pickClose, by = "months")

#play.month.ts <- aggregateSeries(play.ts, FUN = pickClose, by = "weeks", adj = 0.99)

summary(OLS(Y ~ tslag(X, 1:3), data = play.ts, na.rm = T))

}

"scope"<-

function()

{

a = 3

for(i in 1:2){

b = i

print(c)

}

a = b

print(a)

print(b)

#print(i)

print(c)

print(w)

}

"scope2"<-

function()

{

w = 10

scope()

}