STAT0017 ICA2

Hongwei Peng

25/04/2019

(a)

```
load("ICA2_data.RData")
library("fUnitRoots")
```

```
## Loading required package: timeDate
```

```
## Loading required package: timeSeries
```

```
## Loading required package: fBasics
```

```
library(CDVine)
```

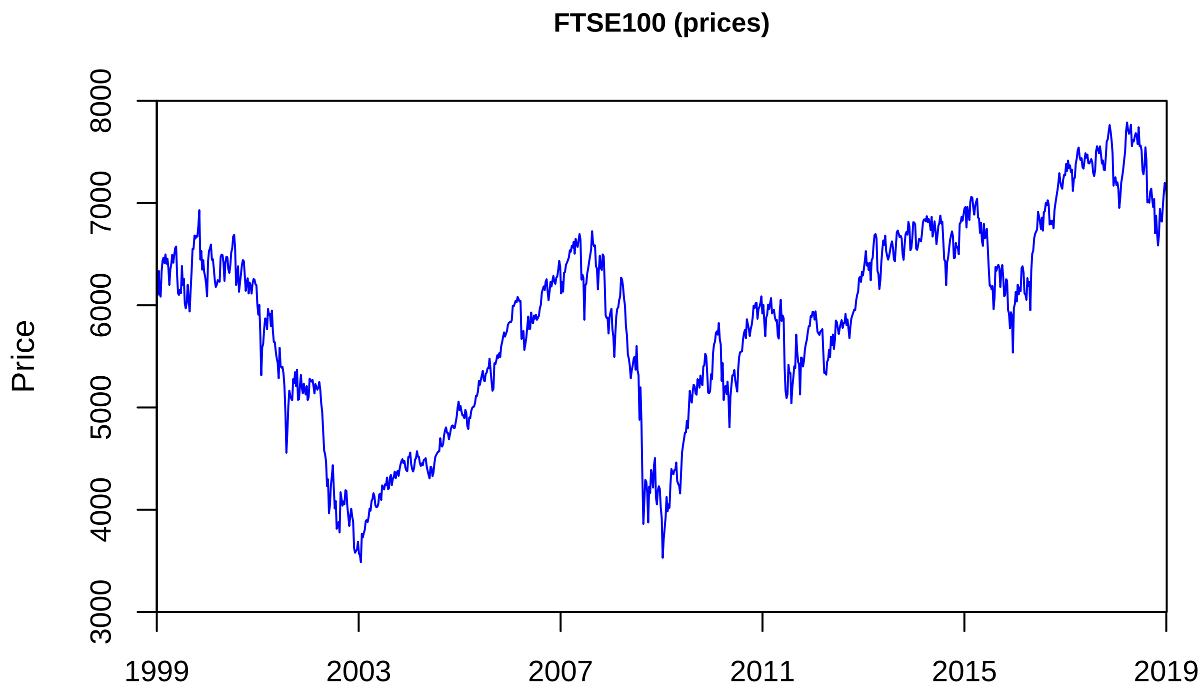
```
## The CDVine package is no longer developed actively.
## Please consider using the more general VineCopula package
## (see https://CRAN.R-project.org/package=VineCopula),
## which extends and improves the functionality of CDVine.
```

```
library(fGarch)
```

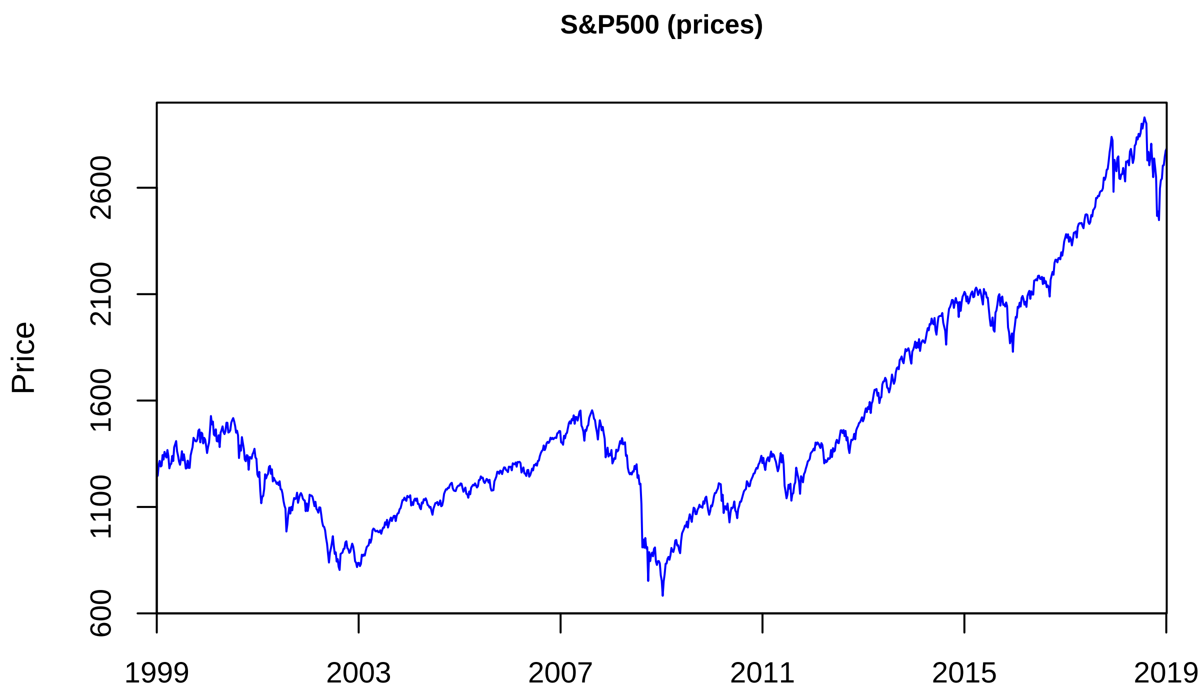
```
## Warning: package 'fGarch' was built under R version 3.5.2
```

```
library(goftest)
library(KScorrect)
library(stats)
library(ggplot2)
library("nloptr")
```

```
opar <- par("mfrow","mar")
plot(data$ftse100~as.Date(data$date,"%d/%m/%y"),type="l",xaxt='n',yaxt='n',
     xlab="",ylab="Price",col="blue",main="FTSE100 (prices)",xaxs="i",
     yaxs="i",ylim=c(3000,8000),cex.main=0.8,cex.lab=1)
axis(2, at = seq(3000,8000,1000), tick=TRUE,cex.axis=0.9)
axis.Date(1, cex.axis=0.9,at=seq(as.Date("1999/02/25"), as.Date("2019/02/28"), "4 years"))
```




```r
plot(data$sp500~as.Date(data$date,"%d/%m/%y"),type="l",xaxt='n',yaxt='n',
     xlab="",ylab="Price",col="blue",main="S&P500 (prices)",xaxs="i",
     yaxs="i",ylim=c(600,3000),cex.main=0.8,cex.lab=1)
axis(2, at = seq(600,3000,500), tick=TRUE,cex.axis=0.9)
axis.Date(1, cex.axis=0.9,at=seq(as.Date("1999/02/25"), as.Date("2019/02/28"), "4 years"))
```




```r
plot(data$sse~as.Date(data$date,"%d/%m/%y"),type="l",xaxt='n',yaxt='n',
     xlab="",ylab="Price",col="blue",main="SSE (prices)",xaxs="i",
     yaxs="i",ylim=c(1000,6500),cex.main=0.8,cex.lab=1)
axis(2, at = seq(1000,6500,500), tick=TRUE,cex.axis=0.9)
axis.Date(1, cex.axis=0.9,at=seq(as.Date("1999/02/25"), as.Date("2019/02/28"), "4 years"))
```


```
unitrootTest(data$ftse100)
```

```
##
## Title:
##  Augmented Dickey-Fuller Test
##
## Test Results:
##   PARAMETER:
##     Lag Order: 1
##   STATISTIC:
##     DF: -0.0358
##   P VALUE:
##     t: 0.6707
##     n: 0.6738
##
## Description:
##  Mon Apr 29 19:51:00 2019 by user:
```

```
unitrootTest(data$sp500)
```

```
##
## Title:
##  Augmented Dickey-Fuller Test
##
## Test Results:
##   PARAMETER:
##     Lag Order: 1
##   STATISTIC:
##     DF: 1.7922
##   P VALUE:
##     t: 0.9828
##     n: 0.983
```

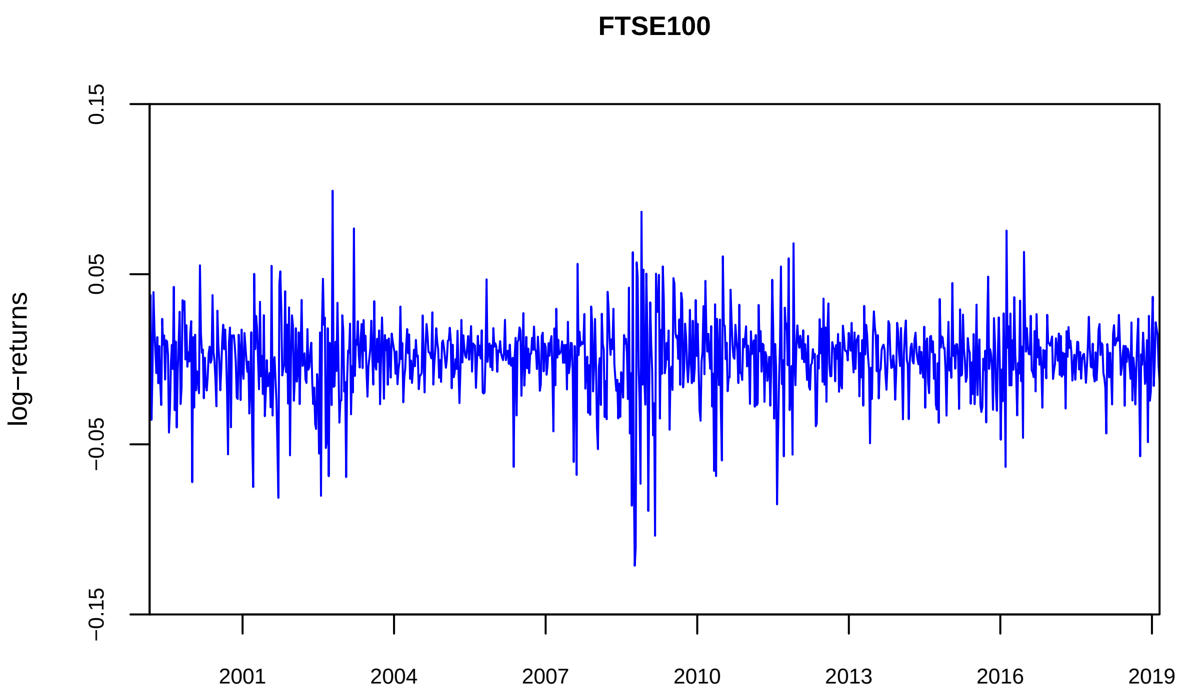
```
##
## Description:
##  Mon Apr 29 19:51:00 2019 by user:
```

```
unitrootTest(data$sse)
```

```
##
## Title:
##  Augmented Dickey-Fuller Test
##
## Test Results:
##   PARAMETER:
##     Lag Order: 1
##   STATISTIC:
##     DF: -0.2252
##   P VALUE:
##     t: 0.6051
##     n: 0.6296
##
## Description:
##  Mon Apr 29 19:51:00 2019 by user:
```

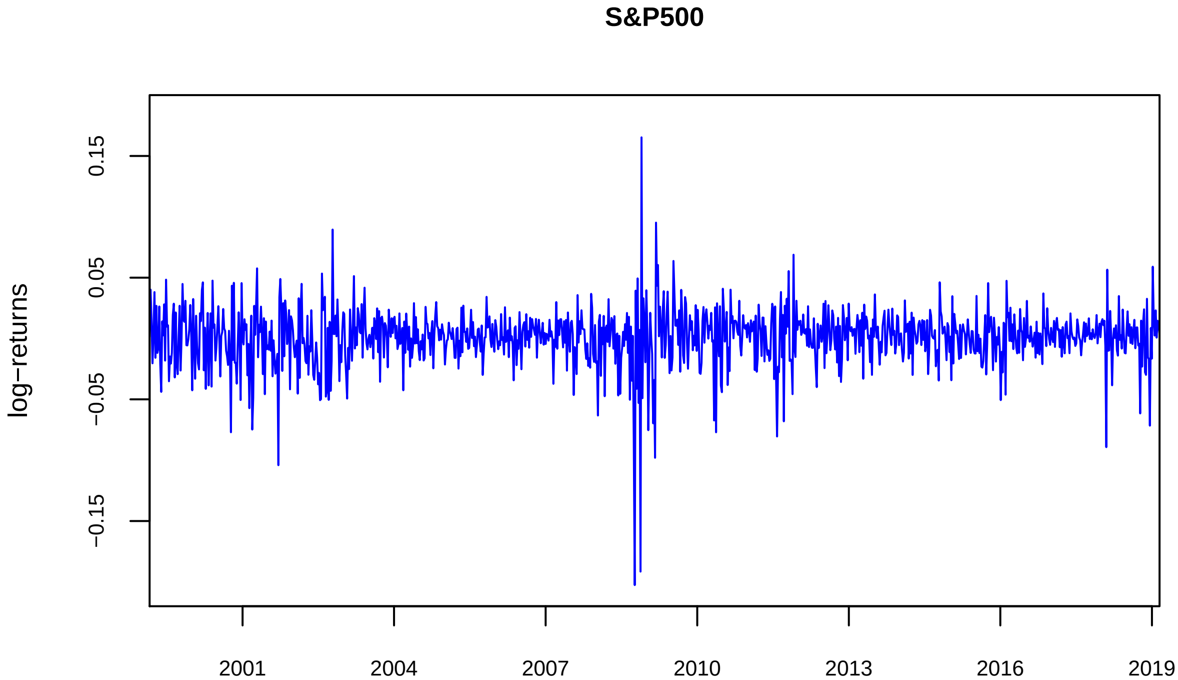
```
ret1<-diff(log(data$ftse100), lag=1,na=remove)
ret2<-diff(log(data$sp500), lag=1,na=remove)
ret3<-diff(log(data$sse), lag=1,na=remove)

plot(ret1~as.Date(data$date[2:length(data$date)],"%d/%m/%y"),type="l",yaxt='n',xaxt='n',
     xlab="",ylab="log-returns",main="FTSE100",xaxs="i",
     yaxs="i", col="blue",ylim=c(-0.15,0.15),cex.main=0.8,cex.lab=0.8)
axis(2, at = seq(-0.15,0.15,0.1), tick=TRUE,cex.axis=0.7)
axis.Date(1, cex.axis=0.7, at=seq(as.Date("1998/01/04"), as.Date("2019/02/28"), "3 years"))
```

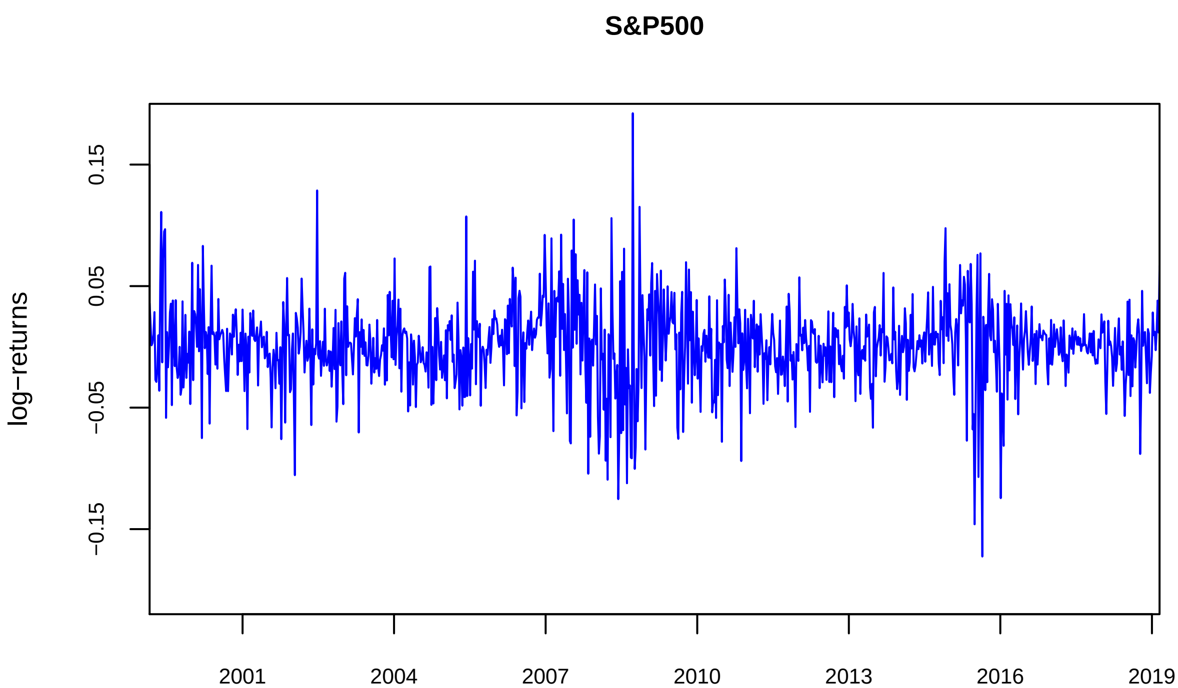



```
plot(ret2~as.Date(data$date[2:length(data$date)],"%d/%m/%y"),type="l",yaxt='n',xaxt='n',
     xlab="",ylab="log-returns",main="S&P500",xaxs="i",
```

```
     yaxs="i", col="blue",ylim=c(-0.22,0.2),cex.main=0.8,cex.lab=0.8)
axis(2, at = seq(-0.15,0.15,0.1), tick=TRUE,cex.axis=0.7)
axis.Date(1, cex.axis=0.7, at=seq(as.Date("1998/01/04"), as.Date("2019/02/28"), "3 years"))
```




```
plot(ret3~as.Date(data$date[2:length(data$date)],"%d/%m/%y"),type="l",yaxt='n',xaxt='n',
     xlab="",ylab="log-returns",main="S&P500",xaxs="i",
     yaxs="i", col="blue",ylim=c(-0.22,0.2),cex.main=0.8,cex.lab=0.8)
axis(2, at = seq(-0.15,0.15,0.1), tick=TRUE,cex.axis=0.7)
axis.Date(1, cex.axis=0.7, at=seq(as.Date("1998/01/04"), as.Date("2019/02/28"), "3 years"))
```



```r
unitrootTest(ret1)
```

```
##
## Title:
##  Augmented Dickey-Fuller Test
##
## Test Results:
##   PARAMETER:
##     Lag Order: 1
##   STATISTIC:
##     DF: -23.8913
##   P VALUE:
##     t: < 2.2e-16
##     n: 0.0005934
##
## Description:
##  Mon Apr 29 19:51:00 2019 by user:
```

```r
unitrootTest(ret2)
```

```
##
## Title:
##  Augmented Dickey-Fuller Test
##
## Test Results:
##   PARAMETER:
##     Lag Order: 1
##   STATISTIC:
##     DF: -22.442
##   P VALUE:
##     t: < 2.2e-16
##     n: 0.0008784
##
## Description:
##  Mon Apr 29 19:51:00 2019 by user:
```

```r
unitrootTest(ret3)
```

```
##
## Title:
##  Augmented Dickey-Fuller Test
##
## Test Results:
##   PARAMETER:
##     Lag Order: 1
##   STATISTIC:
##     DF: -21.4706
##   P VALUE:
##     t: < 2.2e-16
##     n: 0.001144
##
## Description:
##  Mon Apr 29 19:51:00 2019 by user:
```

```r
#acf(ret1)
#acf(ret1^2)
```

```r
#acf(ret2)
#acf(ret2^2)

#acf(ret3)
#acf(ret3^2)

model1=garchFit(formula=~arma(3,0)+garch(1,1),data=ret1,trace=F,cond.dist="sstd")
res1 <- residuals(model1, standardize=TRUE)
Box.test(res1, lag = 10, type = c("Ljung-Box"), fitdf = 0)$p.value
```

```
## [1] 0.2759107
```

```r
Box.test(res1^2, lag = 10, type = c("Ljung-Box"), fitdf = 0)$p.value
```

```
## [1] 0.3307542
```

```r
shape1<-coef(model1)[9]
skew1<-coef(model1)[8]
u1<-psstd(res1, mean=0, sd=1, nu=shape1, xi=skew1)
#hist(u1)

#Kolmogorov-Smirnov test
KStest1<-LcKS(u1, cdf = "punif")
KStest1$p.value
```

```
## [1] 0.7518
```

```r
#Anderson-Darling test
ADtest1<-ad.test(u1, null="punif")
ADtest1$p.value
```

```
## [1] 0.4132992
```

```r
model2=garchFit(formula=~arma(7,0)+garch(1,1),data=ret2,trace=F,cond.dist="sstd")
res2 <- residuals(model2, standardize=TRUE)
Box.test(res2, lag = 10, type = c("Ljung-Box"), fitdf = 0)$p.value
```

```
## [1] 0.02378138
```

```r
Box.test(res2^2, lag = 10, type = c("Ljung-Box"), fitdf = 0)$p.value
```

```
## [1] 0.3601997
```

```r
shape2<-coef(model2)[13]
skew2<-coef(model2)[12]
u2<-psstd(res2, mean=0, sd=1, nu=shape2, xi=skew2)
#hist(u2)

#Kolmogorov-Smirnov test
KStest2<-LcKS(u2, cdf = "punif")
KStest2$p.value
```

```
## [1] 0.2576
```

```r
#Anderson-Darling test
ADtest2<-ad.test(u2, null="punif")
ADtest2$p.value
```

```
## [1] 0.1472603
```

```r
model3=garchFit(formula=~arma(3,0)+garch(1,1),data=ret3,trace=F,cond.dist="sstd")
res3 <- residuals(model3, standardize=TRUE)
Box.test(res3, lag = 10, type = c("Ljung-Box"), fitdf = 0)$p.value
```

```
## [1] 0.7480733
```

```r
Box.test(res3^2, lag = 10, type = c("Ljung-Box"), fitdf = 0)$p.value
```

```
## [1] 0.9878187
```

```r
shape3<-coef(model3)[9]
skew3<-coef(model3)[8]
u3<-psstd(res3, mean=0, sd=1, nu=shape3, xi=skew3)
#hist(u3)

#Kolmogorov-Smirnov test
KStest3<-LcKS(u3, cdf = "punif")
KStest3$p.value
```

```
## [1] 0.8602
```

```r
#Anderson-Darling test
ADtest3<-ad.test(u3, null="punif")
ADtest3$p.value
```

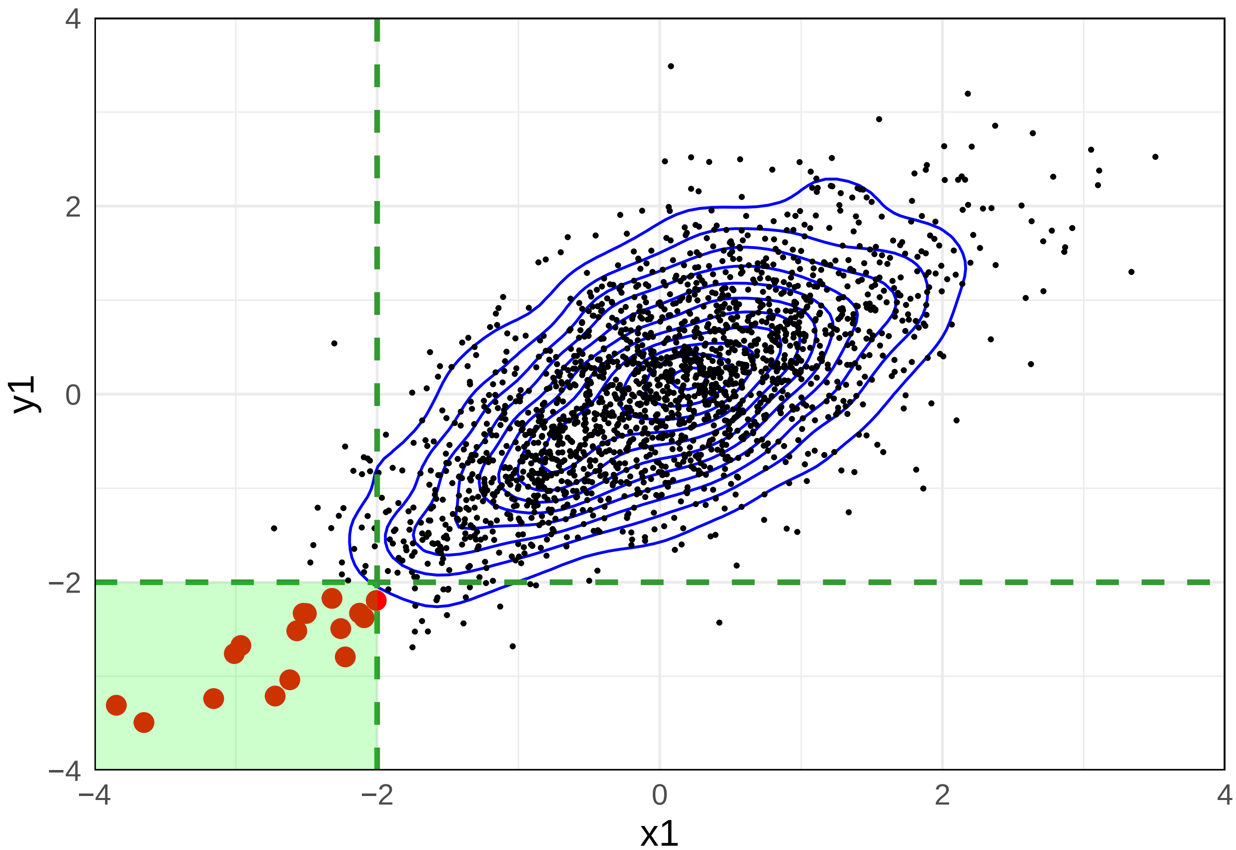
```
## [1] 0.8637571
```

```r
u=cbind(u1,u2,u3)
# Vine selection "manually"
model_1 = BiCopSelect(u[,2],u[,1],familyset=c(1:10,13,14,16,23,24,26))
model_2 = BiCopSelect(u[,2],u[,3],familyset=c(1:10,13,14,16,23,24,26))
h1 = BiCopHfunc(u[,2],u[,3],model_1$family,model_1$par,model_1$par2)
h2 = BiCopHfunc(u[,1],u[,3],model_2$family,model_2$par,model_2$par2)
model_3 = BiCopSelect(h1$hfunc2,h2$hfunc2,familyset=c(1:10,13,14,16,23,24,26))


### Compute Value-at-Risk of ftse100,sp500,sse using MC method based on copulas
u1=BiCopSim(2000, model_1$family, model_1$par, par2=model_1$par2)
x <- qnorm(u1)
u2=BiCopSim(2000, model_2$family, model_2$par, par2=model_2$par2)
y <- qnorm(u2)
u3=BiCopSim(2000, model_3$family, model_3$par, par2=model_3$par2)
z <- qnorm(u3)

zz=data.frame(x)
rownames(zz)<-NULL
colnames(zz)<-c("x1", "y1")
p1=ggplot(zz, aes(x1,  y1)) + geom_density2d(colour="blue")+
  geom_point(size=0.4)+guides(alpha=FALSE)+
  geom_point(data=subset(zz, !x1>-2 & !y1>-2), colour="red", size=3)+
  theme_minimal()+
  theme(panel.border = element_rect(colour = "black", fill=NA, size=0.5),text=element_text(size=14))+
  geom_vline(xintercept = -2, linetype="dashed", color = "#339933", size=1)+
  geom_hline(yintercept = -2, linetype="dashed", color = "#339933", size=1)+
  scale_x_continuous(limits = c(-4,4), expand = c(0, 0)) +
```

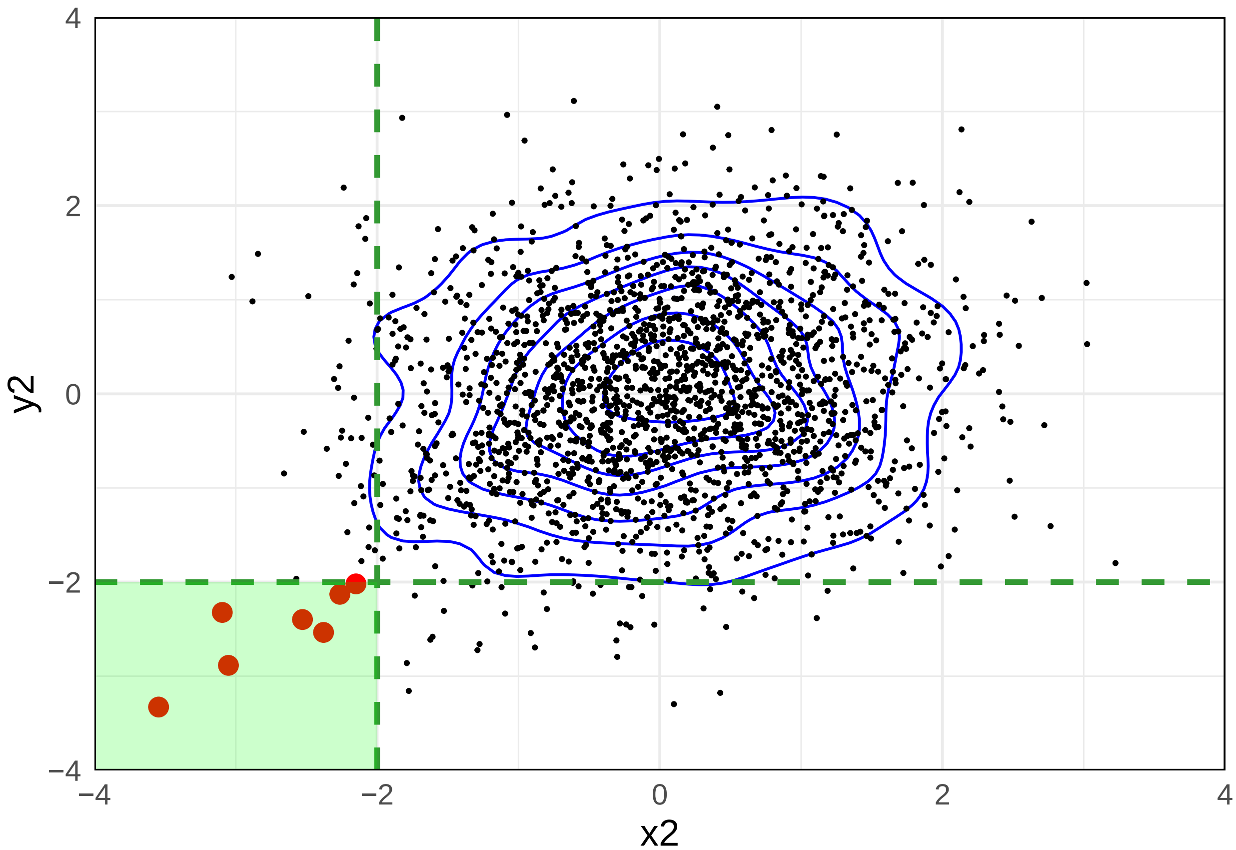
```
  scale_y_continuous(limits = c(-4,4), expand = c(0, 0)) +
  annotate("rect", xmin = c(-4), xmax = c(-2),
           ymin = -4, ymax = -2,
           alpha = 0.2, fill = c("green"))
p1#MC method based on ftse100
```



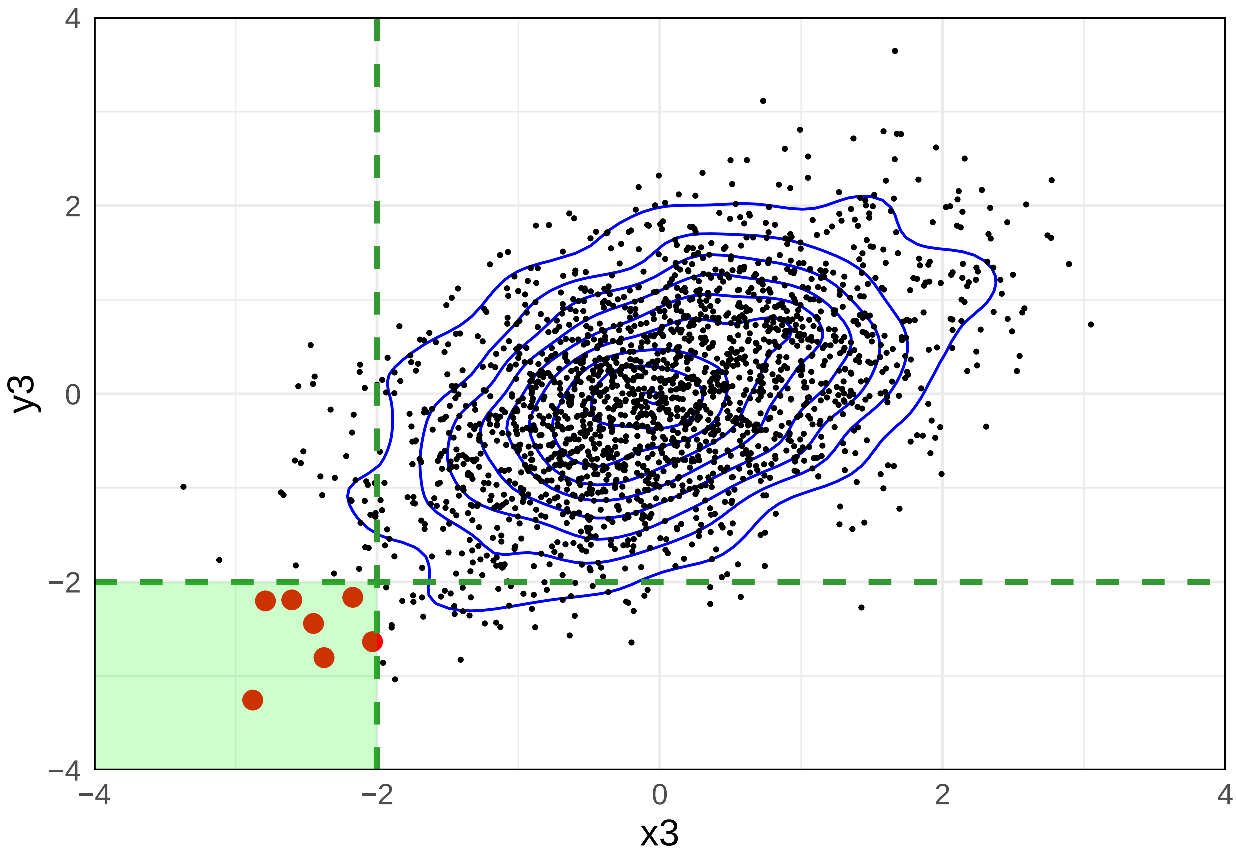

```
zz=data.frame(y)
rownames(zz)<-NULL
colnames(zz)<-c("x2", "y2")
p2=ggplot(zz, aes(x2,  y2)) + geom_density2d(colour="blue")+
  geom_point(size=0.4)+guides(alpha=FALSE)+
  geom_point(data=subset(zz, !x2>-2 & !y2>-2), colour="red", size=3)+
  theme_minimal()+
  theme(panel.border = element_rect(colour = "black", fill=NA, size=0.5),text=element_text(size=14))+
  geom_vline(xintercept = -2, linetype="dashed", color = "#339933", size=1)+
  geom_hline(yintercept = -2, linetype="dashed", color = "#339933", size=1)+
  scale_x_continuous(limits = c(-4,4), expand = c(0, 0)) +
  scale_y_continuous(limits = c(-4,4), expand = c(0, 0)) +
  annotate("rect", xmin = c(-4), xmax = c(-2),
           ymin = -4, ymax = -2,
           alpha = 0.2, fill = c("green"))
p2##MC method based on sp500
```

```
## Warning: Removed 1 rows containing non-finite values (stat_density2d).

## Warning: Removed 1 rows containing missing values (geom_point).
```




```
zz=data.frame(z)
rownames(zz)<-NULL
colnames(zz)<-c("x3", "y3")
p3=ggplot(zz, aes(x3,  y3)) + geom_density2d(colour="blue")+
  geom_point(size=0.4)+guides(alpha=FALSE)+
  geom_point(data=subset(zz, !x3>-2 & !y3>-2), colour="red", size=3)+
  theme_minimal()+
  theme(panel.border = element_rect(colour = "black", fill=NA, size=0.5),text=element_text(size=14))+
  geom_vline(xintercept = -2, linetype="dashed", color = "#339933", size=1)+
  geom_hline(yintercept = -2, linetype="dashed", color = "#339933", size=1)+
  scale_x_continuous(limits = c(-4,4), expand = c(0, 0)) +
  scale_y_continuous(limits = c(-4,4), expand = c(0, 0)) +
  annotate("rect", xmin = c(-4), xmax = c(-2),
           ymin = -4, ymax = -2,
           alpha = 0.2, fill = c("green"))
p3##MC method based on sse
```




```
var=matrix(0,3,2)

retport1=log(1+((exp(x[,1])-1)*0.5+(exp(x[,2])-1)*0.5))
var[1,]=quantile(retport1,c(0.01,0.05))

retport2=log(1+((exp(y[,1])-1)*0.5+(exp(y[,2])-1)*0.5))
var[2,]=quantile(retport2,c(0.01,0.05))

retport3=log(1+((exp(z[,1])-1)*0.5+(exp(z[,2])-1)*0.5))
var[3,]=quantile(retport3,c(0.01,0.05))

### VaR for portfolio returns generated using different coplas
var  #the log diff VaR under 0.01 and 0.05
```

```
##            [,1]      [,2]
## [1,] -2.052375 -1.478215
## [2,] -1.744600 -1.151467
## [3,] -1.894482 -1.346672
```

(b)

```
#u=cbind(ret1,ret2,ret3)
u=cbind(u1,u2,u3)
cor(u[,1:3],method = c("kendall"))
```

```
##              [,1]          [,2]          [,3]
## [1,] 1.000000000  0.4706059206  0.0095317659
## [2,] 0.470605921  1.0000000000 -0.0007788896
## [3,] 0.009531766 -0.0007788896  1.0000000000
```

```r
u1u2u3=cbind(u[,1],u[,2],u[,3])
vinemodel=CDVineCopSelect(u1u2u3,type=2,familyset=c(1:10,13,14,23,24))
vinemodel
```

```
## $family
## [1] 7 1 6
##
## $par
## [1] 0.80283454 0.00555147 1.01224599
##
## $par2
## [1] 1.340331 0.000000 0.000000
```

```r
N=2000
u1u2u3_sim=CDVineSim(N, family=vinemodel$family, vinemodel$par,  vinemodel$par2, type=2)
cor(u1u2u3_sim,method = c("kendall"))
```

```
##              [,1]        [,2]        [,3]
## [1,]  1.00000000  0.45495393 -0.01667434
## [2,]  0.45495393  1.00000000 -0.01703954
## [3,] -0.01667434 -0.01703954  1.00000000
```

```r
cor(u1u2u3,method = c("kendall"))
```

```
##              [,1]          [,2]          [,3]
## [1,] 1.000000000  0.4706059206  0.0095317659
## [2,] 0.470605921  1.0000000000 -0.0007788896
## [3,] 0.009531766 -0.0007788896  1.0000000000
```

```r
vinemodel_sim=CDVineCopSelect(u1u2u3_sim,type=2,familyset=c(1:10,13,14,23,24))
vinemodel_sim
```

```
## $family
## [1]  9 23  5
##
## $par
## [1]  1.44631876 -0.06256853 -0.03291068
##
## $par2
## [1] 1.143613 0.000000 0.000000
```

(c)

```r
loglike_fun <- function(x1, x2, mu1, mu2, sigma1, sigma2, rho)
{
  par1<-par[1]
  par2<-par[2]
  rho <-par[3]
  n=length(u1)
  sumlik=-sum(dmvnorm(cbind(u[,1],u[,2]), c(par1,par2), matrix(c(1, rho, rho, 1), ncol=2), log=T))
```

```
  # Output
  cat("log-likelihood ->",sprintf("%4.4f",- sumlik),"\n")

  return(sumlik)
}
```

```
library("CDVine")
fit <- BiCopEst(u[,1],u[,2],family=1,method="mle",se=TRUE)
#BiCopPDF(u1, u2, family=1,par = fit$par,par2 = fit$par2)
#the resulting estimate with the parameter value estimated using function BiCopEst.
```