Analysis of eight daily European bank asset values using vine copulas between 2013-2017(Application to Lecture 2)

Claudia Czado, Technical University Munich,

10 August, 2018

Load packages

```
library(VineCopula) # primary software for vine copulas
library(rafalib) # allows for nicer mulltiple plots
library(ggplot2) # allows for nice plots
library(reshape2) # allows to merge data
library(fGarch) # fits GARCH times series models
library(TSP) # needed to fit D-Vines
```

Setup

Read in data

```
load("stock_data1317.Rdata")
lret<-data.frame(stock_data1317$lret)
data.org<-data.frame(stock_data1317$data_orinigal)
colnames(lret)
```

```
##  [1] "ACA.PA"  "BBVA.MC" "BNP.PA"  "CBK.DE"  "DBK.DE"  "GLE.PA"  "ISP.MI"
##  [8] "SAN.MC"  "BAS.DE"  "BAYN.DE" "FME.DE"  "FRE.DE"  "LIN.DE"  "MRK.DE"
## [15] "SDF.DE"
```

Create bank data with create daily dates

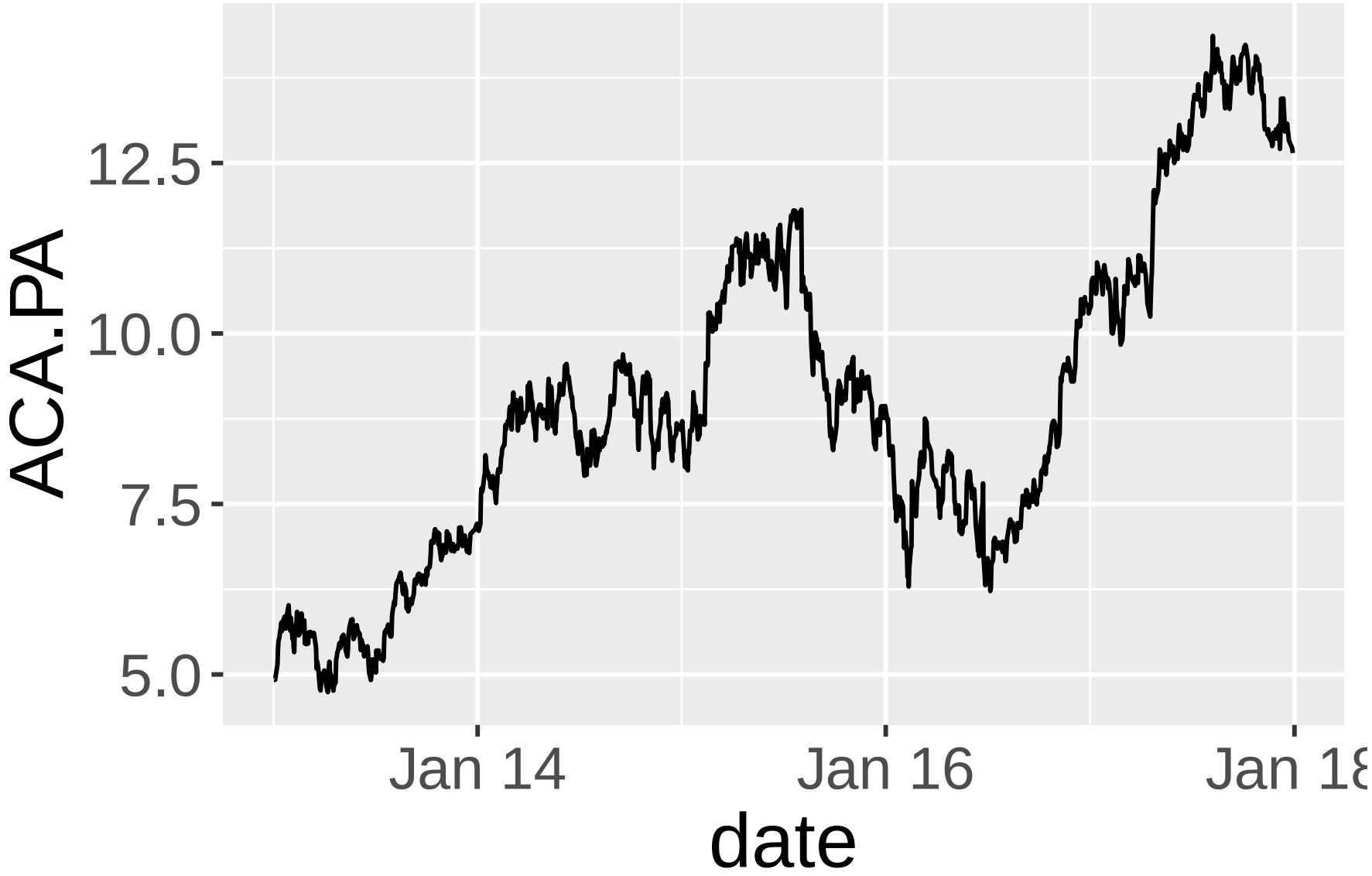
```
bank.names<-colnames(lret)[1:8]
dates<-as.Date(rownames(data.org))
data.bank<-data.frame(dates=dates, data.org[,bank.names])
dim(data.bank)
```

```
## [1] 1280    9
```

```
lret.bank<-data.frame(dates=as.Date(rownames(lret)),
                      lret[,bank.names])
dim(lret.bank)
```

```
## [1] 1279    9
```

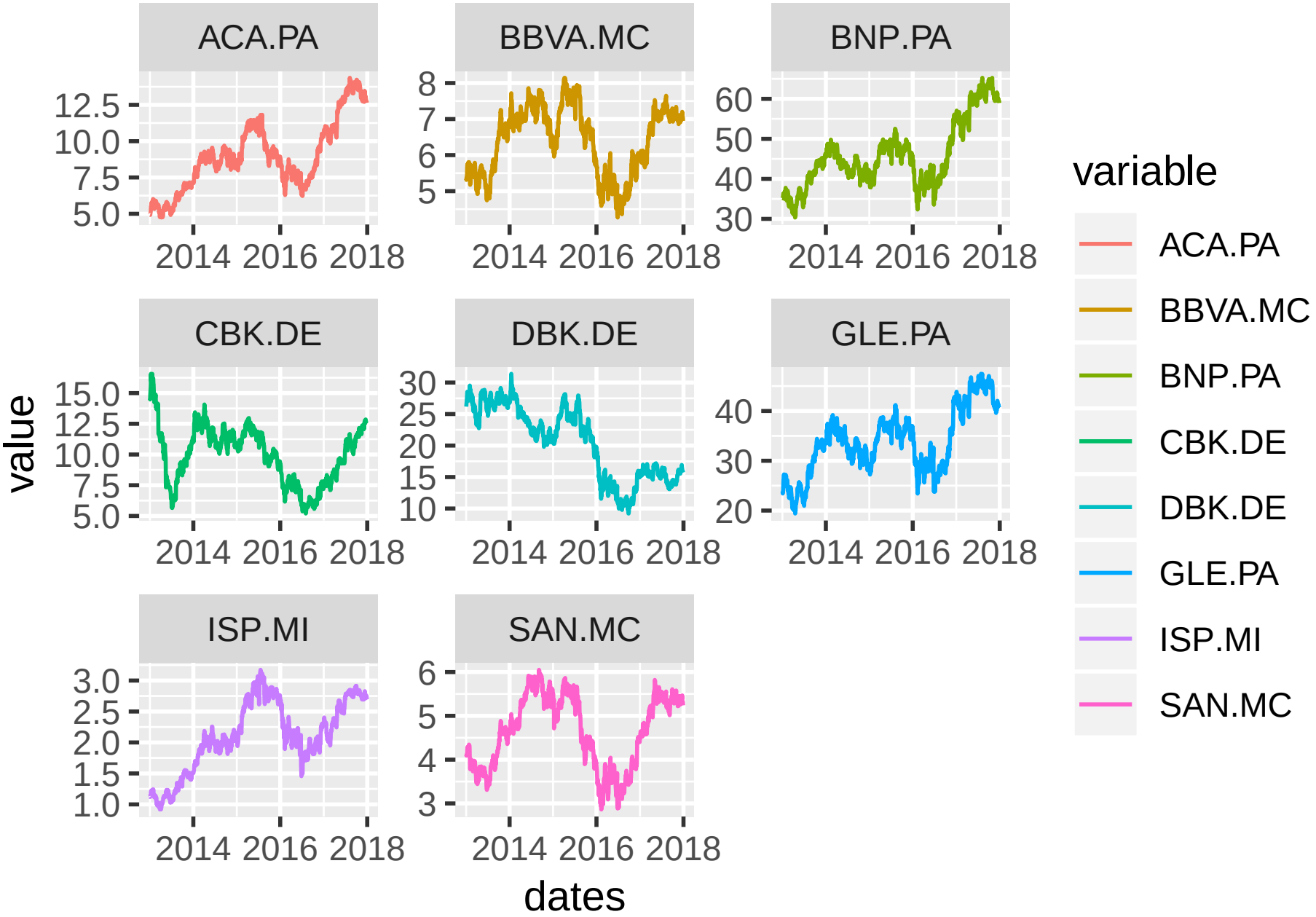
Show time trend of single stock “ACA.PA”



R-code: Show time trend of single stock "ACA.PA"

```
ggplot(data.bank, aes(dates, ACA.PA)) +
    geom_line() +
    scale_x_date(labels = scales::date_format("%b %y")) +
    xlab("date") +
    ylab("ACA.PA") +
     theme(plot.title = element_text(lineheight = 0.8,
                        face = "bold", size = 20)) +
    theme(text = element_text(size = 18))
```

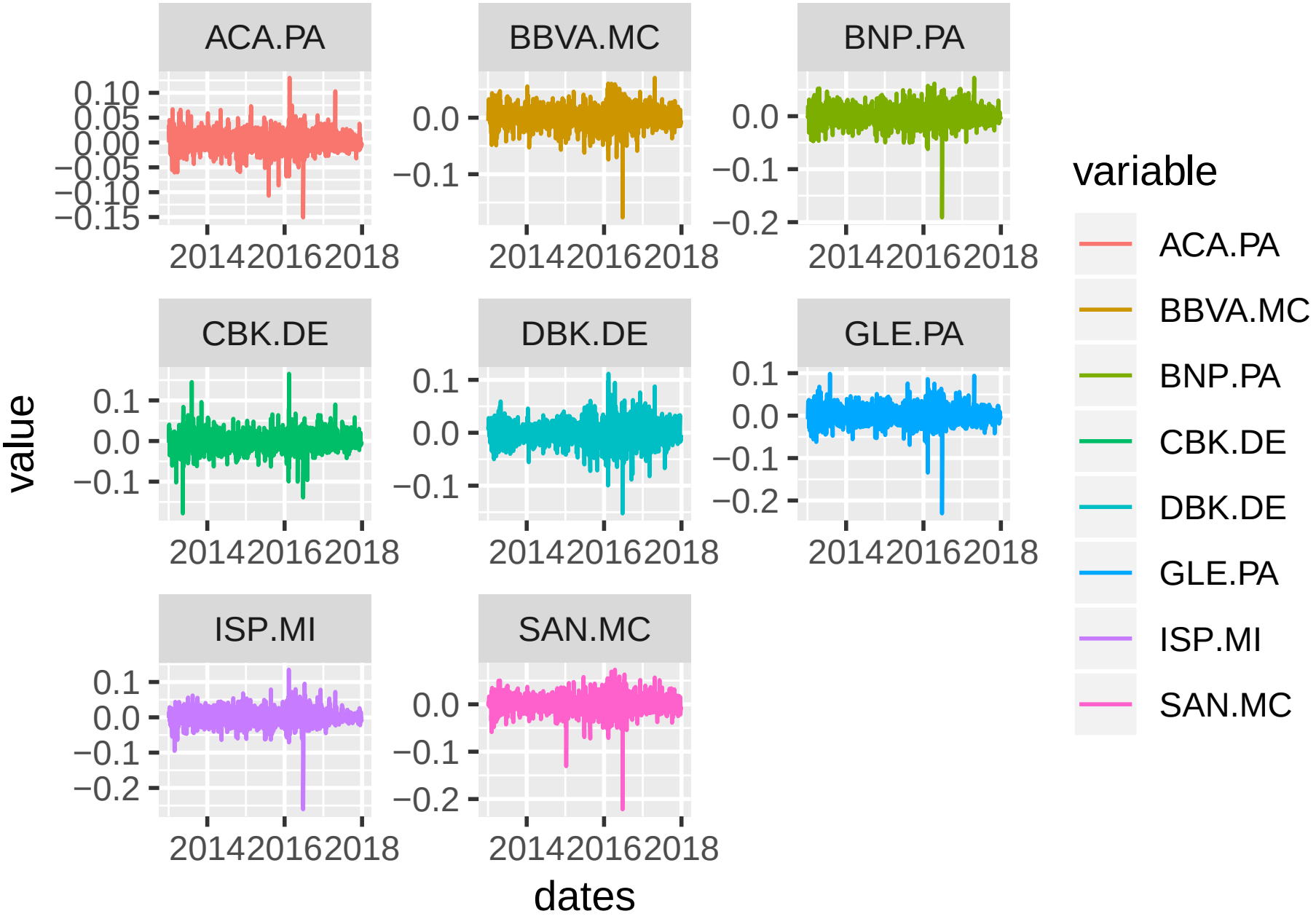
Time plot of bank stocks



R:code: Time plot of bank stocks

```
df <- melt(data.bank[, 1:9], id="dates")
ggplot(df) + geom_line(aes(x=dates, y=value, color=variable))
+ facet_wrap( ~ variable, scales="free")
```

Time plot of log returns for banks



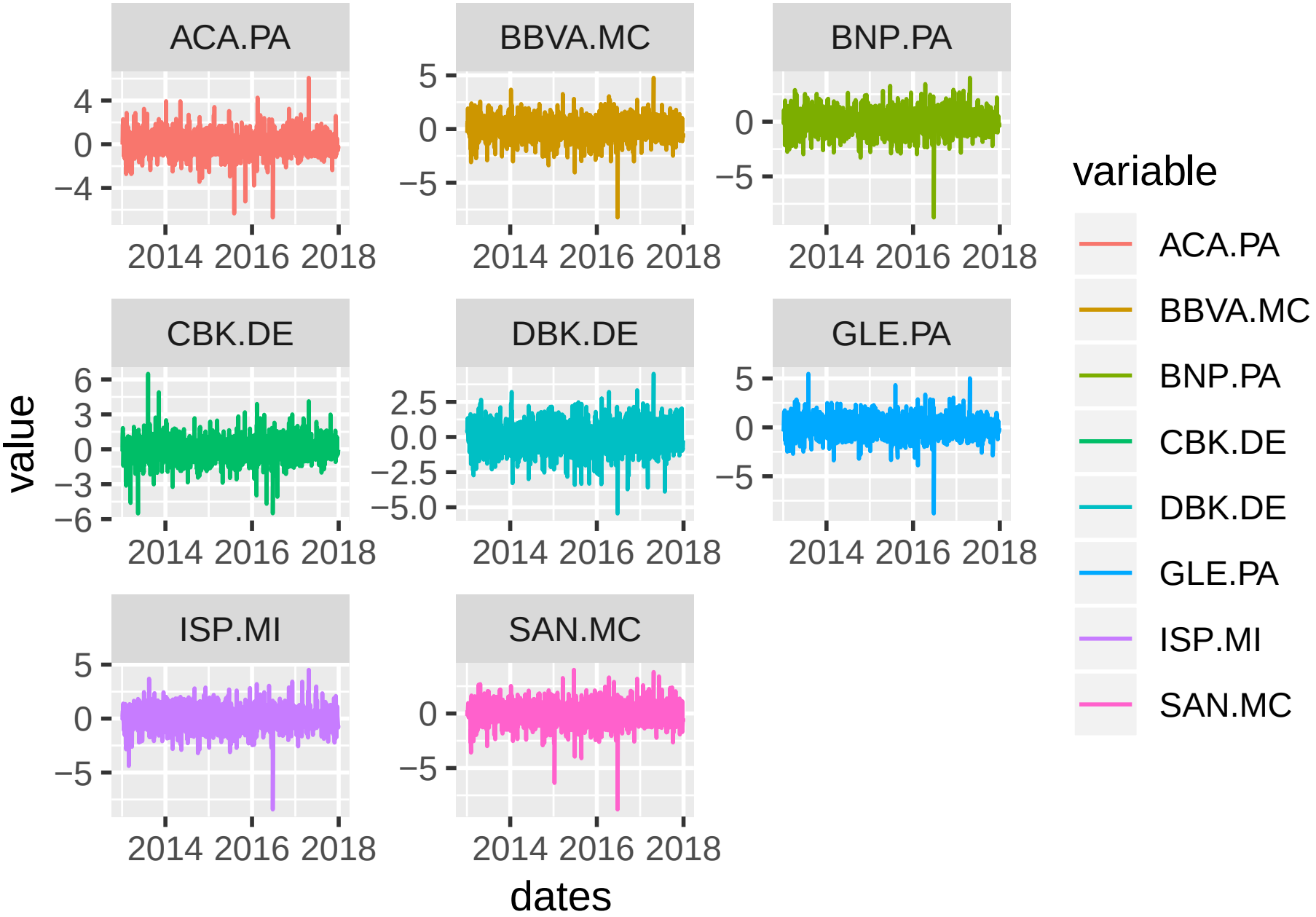
R code:Time plot of log returns for banks

```
df <- melt(lret.bank[, 1:9], id="dates")
ggplot(df) + geom_line(aes(x=dates, y=value, color=variable))
+ facet_wrap( ~ variable, scales="free")
```

Fit marginal GARCH models with Student t errors

```
fits = list() # Garch(1,1) fits
res = list() # standardized residuals
pvalues = rep(0,8) # p-values of KS test for t distribution
resdata= matrix(0,dim(lret.bank)[1],8)
udata = matrix(0,dim(lret.bank)[1],8)
for(i in 1:8){
fits[[i]] = garchFit(formula=~garch(1,1),
            data=lret.bank[,i+1], trace=FALSE,
            include.mean=TRUE, cond.dist="std")
res[[i]] = residuals(fits[[i]], standardize=TRUE)
           pvalues[i] = ks.test(res[[i]], "pstd",
          0, 1, coef(fits[[i]])[5])$p.value
  #transform standardized residuals with Student t cdf
  udata[,i] = pstd(res[[i]], nu=coef(fits[[i]])[5])
  resdata[,i]= res[[i]]
}
colnames(udata)<-colnames(lret.bank[,2:9])
colnames(udata)<-c("ACA","BBVA","BNP","CBK","DBK","GLE", "ISP","SAN")
rownames(udata)<-rownames(lret.bank)
colnames(resdata)<-colnames(lret.bank[,2:9])
rownames(resdata)<-rownames(lret.bank)
```

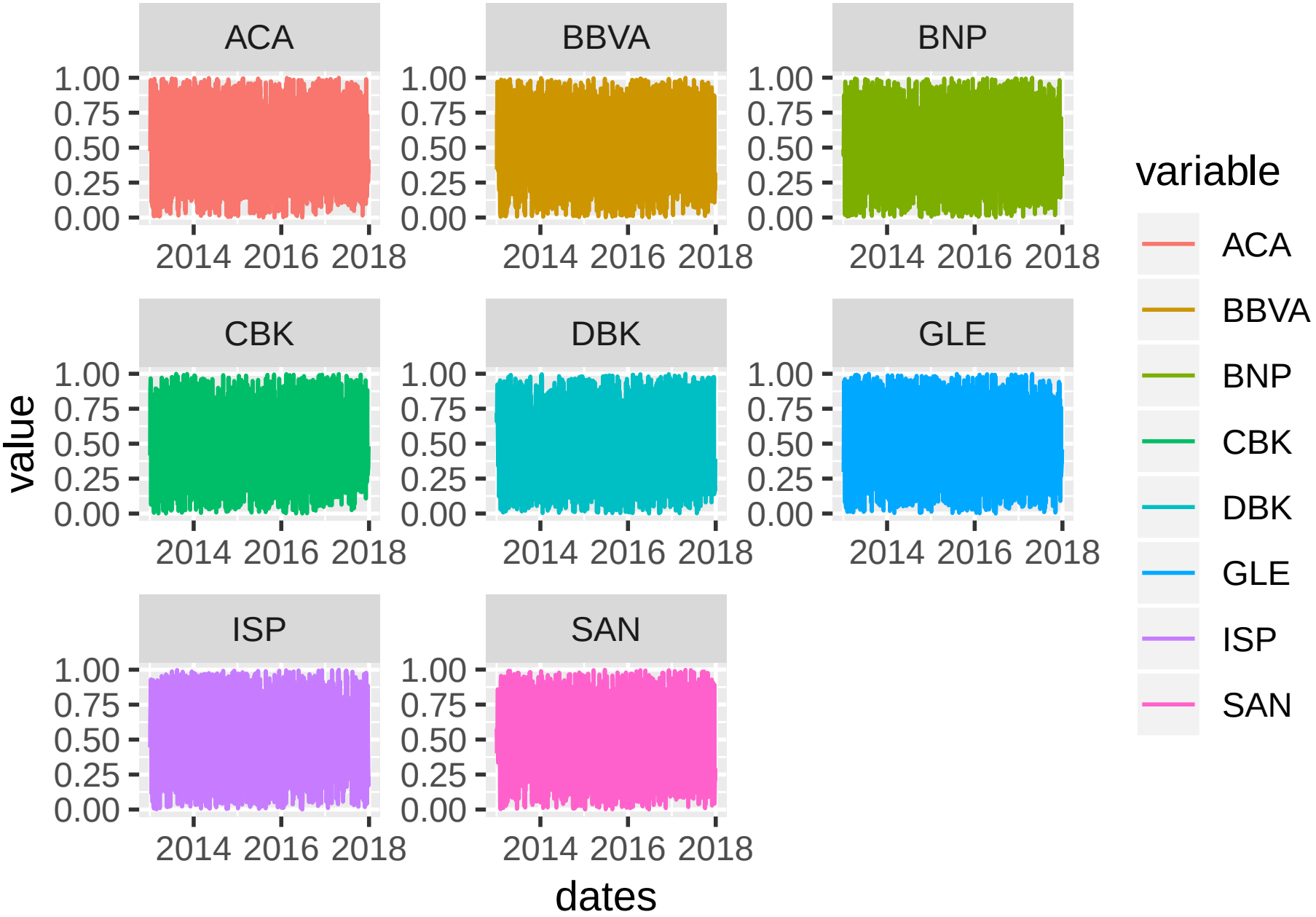
time plot of standardized residuals for banks



R code: time plot of standardized residuals for banks

```
res.dates<-data.frame(dates=as.Date(rownames(lret)),resdata)
df <- melt(res.dates[, 1:9], id="dates")
ggplot(df) + geom_line(aes(x=dates, y=value, color=variable))+
 facet_wrap( ~ variable, scales="free")
```

time plot of copula data for banks



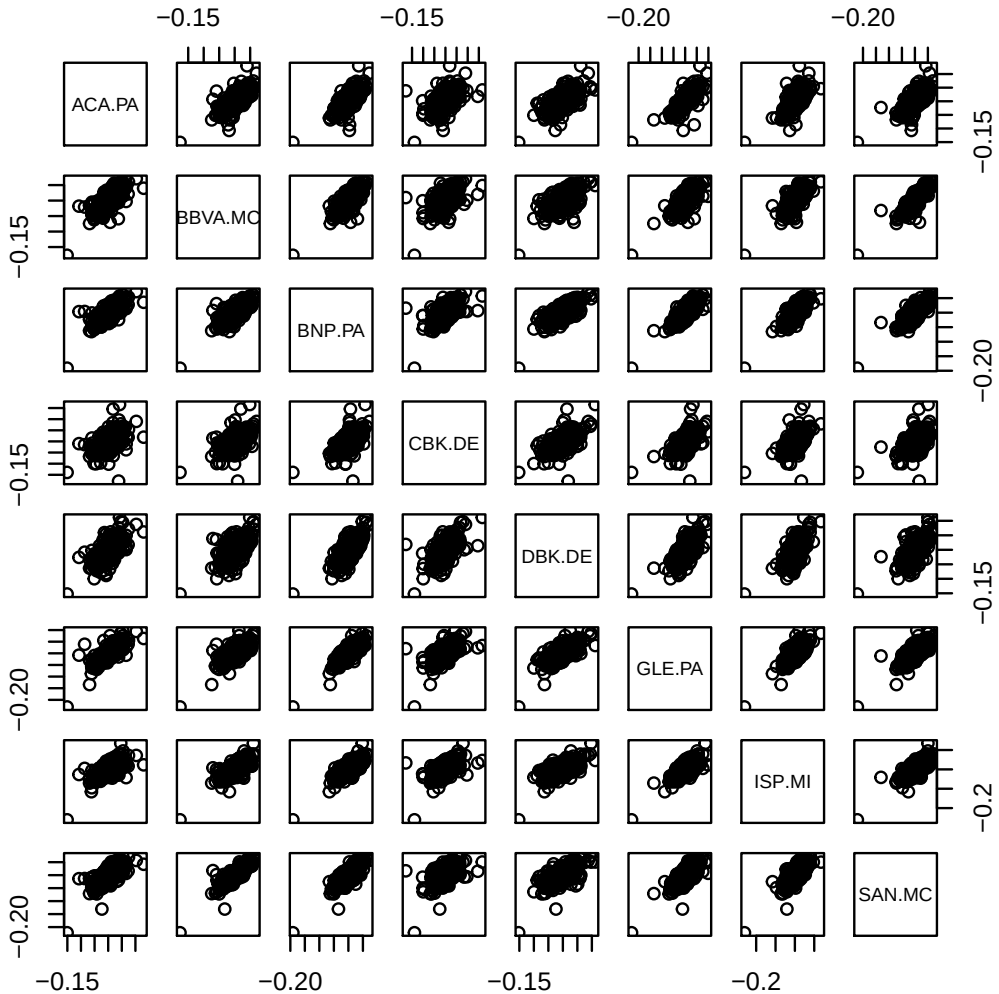
R code: time plot of copula data for banks

```
udata.dates<-data.frame(dates=as.Date(rownames(lret)),udata)
df <- melt(udata.dates[, 1:9], id="dates")
ggplot(df) + geom_line(aes(x=dates, y=value, color=variable))
+ facet_wrap( ~ variable, scales="free")
```

Exploring pairwise dependence

pairs plot ignoring serial dependence

```
pairs(lret.bank[,2:9])
```



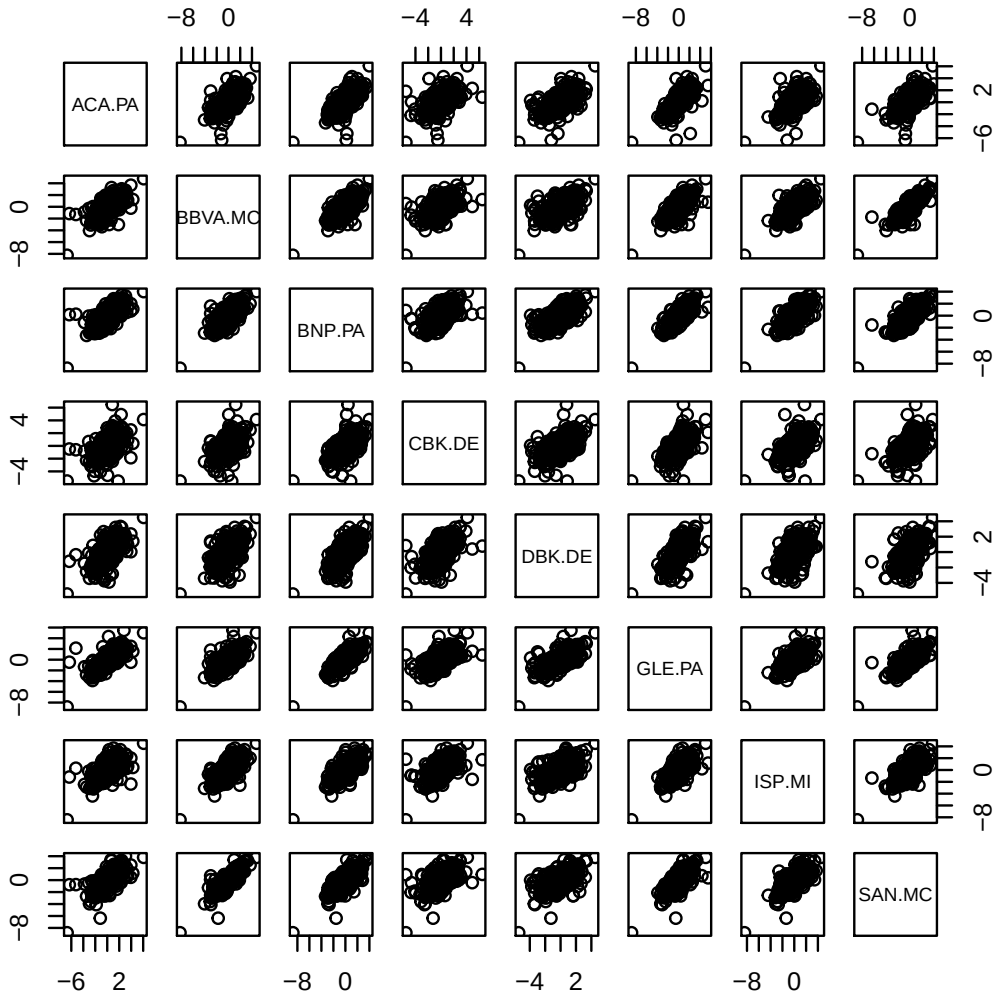
Kendall's tau ignoring serial dependence

```
round(cor(lret.bank[,2:9],method="kendall"), digits=2)
```

```
##          ACA.PA BBVA.MC BNP.PA CBK.DE DBK.DE GLE.PA ISP.MI SAN.MC
## ACA.PA     1.00    0.56   0.61   0.50   0.53   0.65   0.52   0.56
## BBVA.MC    0.56    1.00   0.59   0.46   0.52   0.60   0.52   0.72
## BNP.PA     0.61    0.59   1.00   0.47   0.56   0.68   0.55   0.60
## CBK.DE     0.50    0.46   0.47   1.00   0.53   0.53   0.45   0.47
## DBK.DE     0.53    0.52   0.56   0.53   1.00   0.58   0.47   0.53
## GLE.PA     0.65    0.60   0.68   0.53   0.58   1.00   0.56   0.59
## ISP.MI     0.52    0.52   0.55   0.45   0.47   0.56   1.00   0.53
## SAN.MC     0.56    0.72   0.60   0.47   0.53   0.59   0.53   1.00
```

Dependence among standardized residuals (x-scale)

```
pairs(resdata)
```



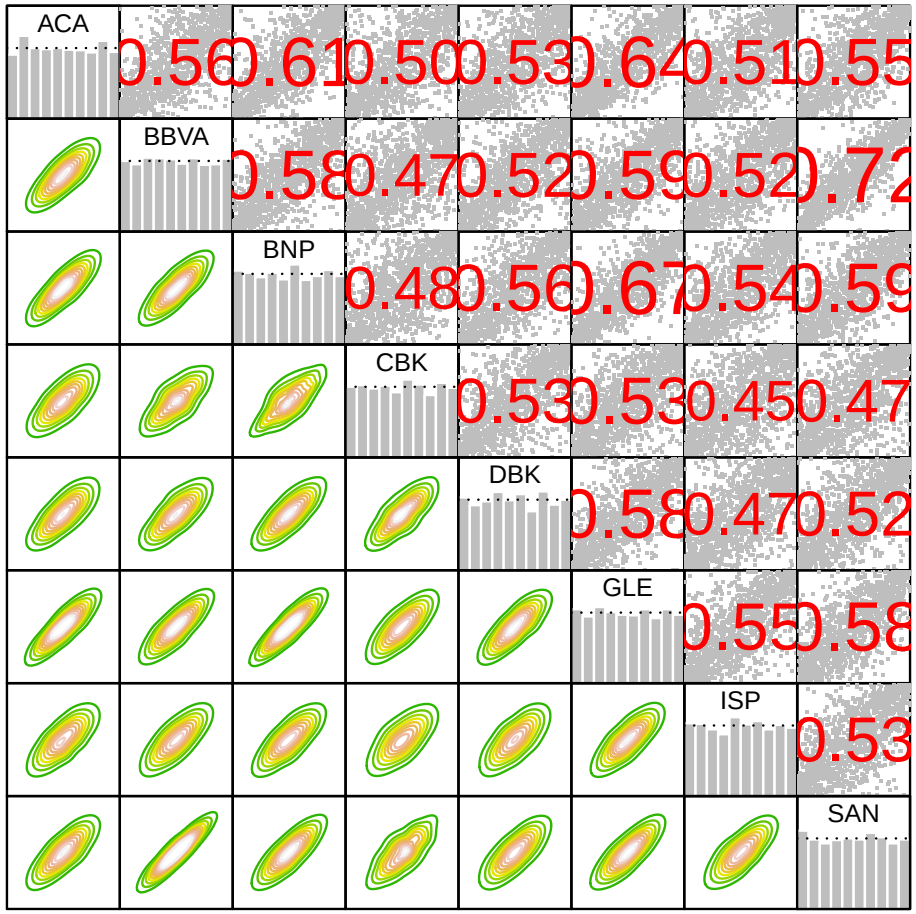
Kendall's tau among standardized residuals

```
round(cor(resdata,method="kendall"), digits=2)
```

```
##          ACA.PA BBVA.MC BNP.PA CBK.DE DBK.DE GLE.PA ISP.MI SAN.MC
## ACA.PA     1.00    0.56   0.61   0.50   0.53   0.64   0.51   0.55
## BBVA.MC    0.56    1.00   0.58   0.47   0.52   0.59   0.52   0.72
## BNP.PA     0.61    0.58   1.00   0.48   0.56   0.67   0.54   0.59
## CBK.DE     0.50    0.47   0.48   1.00   0.53   0.53   0.45   0.47
## DBK.DE     0.53    0.52   0.56   0.53   1.00   0.58   0.47   0.52
## GLE.PA     0.64    0.59   0.67   0.53   0.58   1.00   0.55   0.58
## ISP.MI     0.51    0.52   0.54   0.45   0.47   0.55   1.00   0.53
## SAN.MC     0.55    0.72   0.59   0.47   0.52   0.58   0.53   1.00
```

Normalized contour plots (z-scale)

```
udata.cop<-as.copuladata(udata)
pairs(udata.cop)
```



Parametric vine copula analysis using 4 candidate copula families

Fit R-vine using only 4 pair copula families with and without independence test

```
rv.fsub<-RVineStructureSelect(udata.cop, family=c(1,2,3,4),
                  method = "mle")
rv.fsub.ind<-RVineStructureSelect(udata.cop, family=c(1,2,3,4),
                  method = "mle",indeptest = TRUE)
# suppress console output
dummy<-capture.output(sum.fsub<- summary(rv.fsub))
dummy<-capture.output(sum.fsub.ind<- summary(rv.fsub.ind))
```

Results of first two trees of rv.fsub

1 <-> ACA, 2 <-> BBVA, 3 <-> BNP, 4 <-> CBK,

5 <-> DBK, 6 <-> GLE, 7 <-> ISP, 8 <-> SAN

```
print(sum.fsub[1:14, ], digits = 3)
```

```
##    tree    edge family cop   par  par2   tau    utd    ltd
## 1     1     8,3      2   t 0.797  5.51 0.587 0.4212 0.4212
## 2     1     8,2      2   t 0.902  3.85 0.715 0.6384 0.6384
## 3     1     3,6      2   t 0.864  4.37 0.665 0.5576 0.5576
## 4     1     6,1      2   t 0.839  4.40 0.634 0.5208 0.5208
## 5     1     6,7      2   t 0.752  5.05 0.541 0.3896 0.3896
## 6     1     6,5      2   t 0.785  4.08 0.575 0.4690 0.4690
## 7     1     6,4      2   t 0.733  5.04 0.524 0.3721 0.3721
## 8     2   3,2;8      2   t 0.275  6.07 0.177 0.0846 0.0846
## 9     2   8,6;3      2   t 0.326  8.61 0.212 0.0526 0.0526
## 10    2   3,1;6      2   t 0.302  7.07 0.195 0.0708 0.0708
## 11    2   3,7;6      2   t 0.290  9.53 0.187 0.0356 0.0356
## 12    2   3,5;6      2   t 0.270 10.44 0.174 0.0257 0.0257
## 13    2   5,4;6      2   t 0.387  9.81 0.253 0.0517 0.0517
## 14    3 2,6;8,3      4   G 1.125  0.00 0.111 0.1484 0.0000
```

Results of trees 2 to 5 of rv.fsub

1 <-> ACA, 2 <-> BBVA, 3 <-> BNP, 4 <-> CBK,

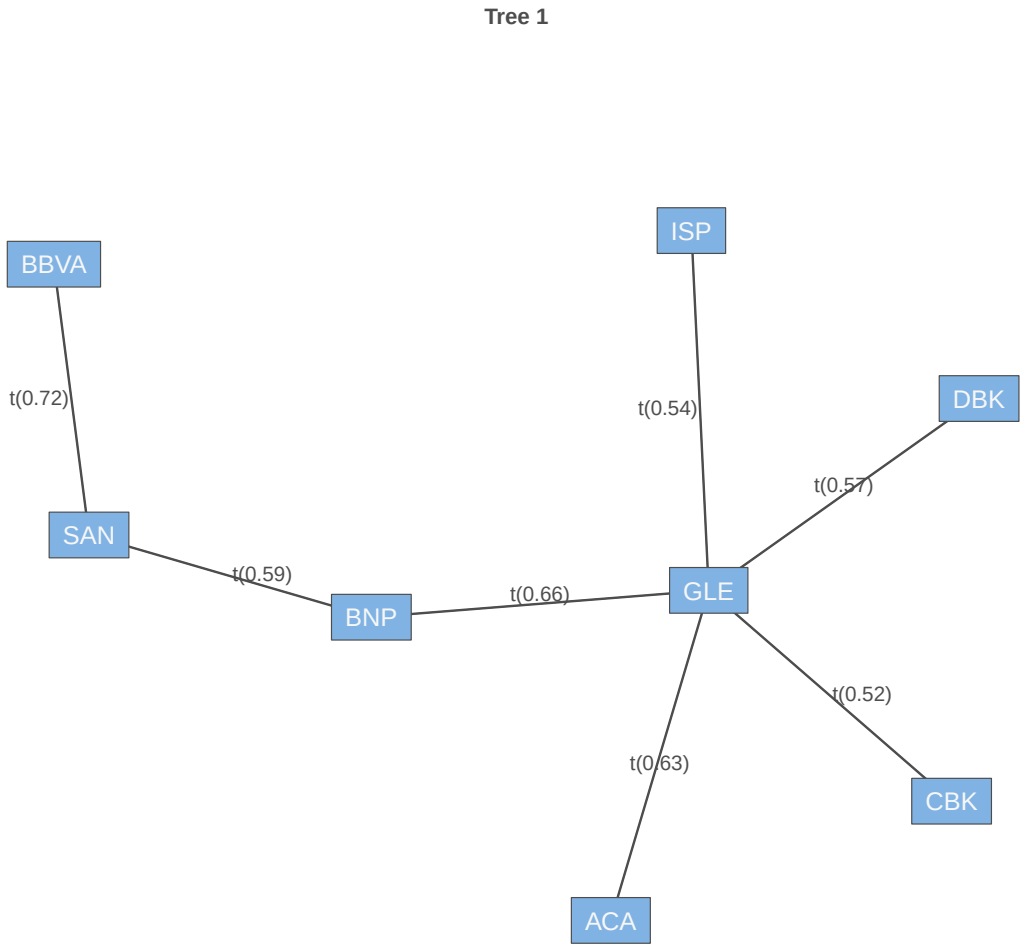
5 <-> DBK, 6 <-> GLE, 7 <-> ISP, 8 <-> SAN

```
print(sum.fsub[15:28, ], digits = 3)
```

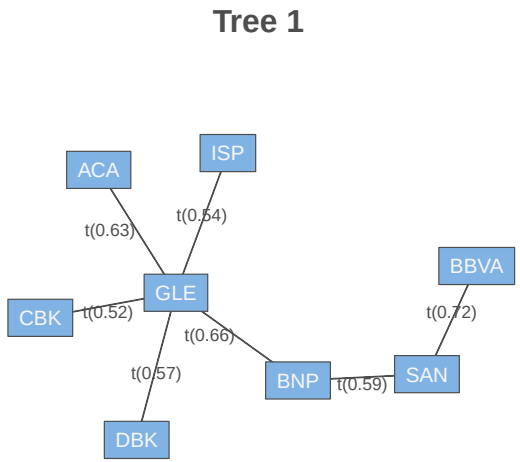
```
##    tree             edge family cop    par par2    tau      utd      ltd
## 15    3          8,1;3,6      1   N 0.1836  0.0 0.1175 0.00e+00 0.00e+00
## 16    3          8,7;3,6      2   t 0.2539 17.5 0.1634 3.72e-03 3.72e-03
## 17    3          8,5;3,6      2   t 0.1810 10.5 0.1159 1.59e-02 1.59e-02
## 18    3          3,4;5,6      1   N 0.0354  0.0 0.0226 0.00e+00 0.00e+00
## 19    4        2,1;8,3,6      2   t 0.1141 28.8 0.0728 3.45e-05 3.45e-05
## 20    4        1,7;8,3,6      2   t 0.1132 30.0 0.0722 2.34e-05 2.34e-05
## 21    4        1,5;8,3,6      1   N 0.0921  0.0 0.0587 0.00e+00 0.00e+00
## 22    4        8,4;3,5,6      1   N 0.1258  0.0 0.0803 0.00e+00 0.00e+00
## 23    5      2,7;1,8,3,6      1   N 0.0830  0.0 0.0529 0.00e+00 0.00e+00
## 24    5      2,5;1,8,3,6      2   t 0.0645 18.3 0.0411 5.63e-04 5.63e-04
## 25    5      1,4;8,3,5,6      2   t 0.1309 18.9 0.0836 8.63e-04 8.63e-04
## 26    6    7,5;2,1,8,3,6      1   N 0.0493  0.0 0.0314 0.00e+00 0.00e+00
## 27    6    2,4;1,8,3,5,6      1   N 0.0249  0.0 0.0158 0.00e+00 0.00e+00
## 28    7 7,4;2,1,8,3,5,6      1   N 0.1017  0.0 0.0649 0.00e+00 0.00e+00
```

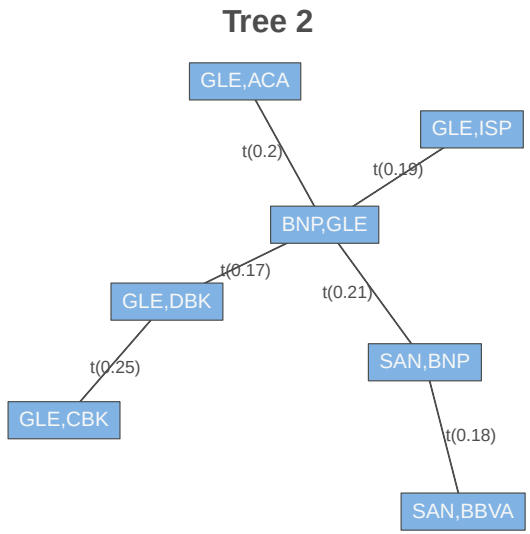
Tree 1 of rv.fsub

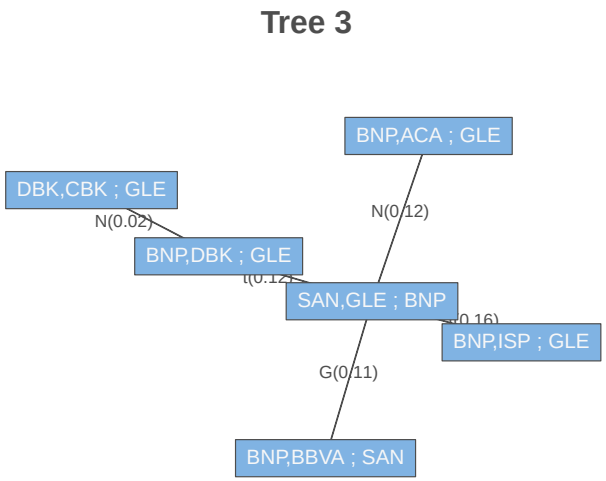
```
plot(rv.fsub,tree=1,edge.labels="family-tau",type=1)
```

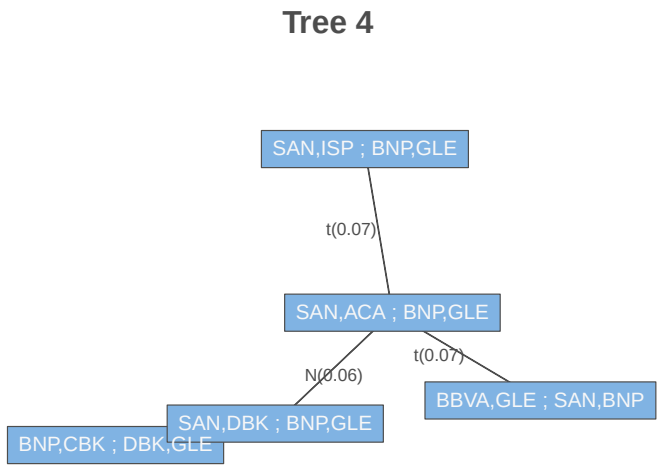


Trees 1 to 4 of rv.sub

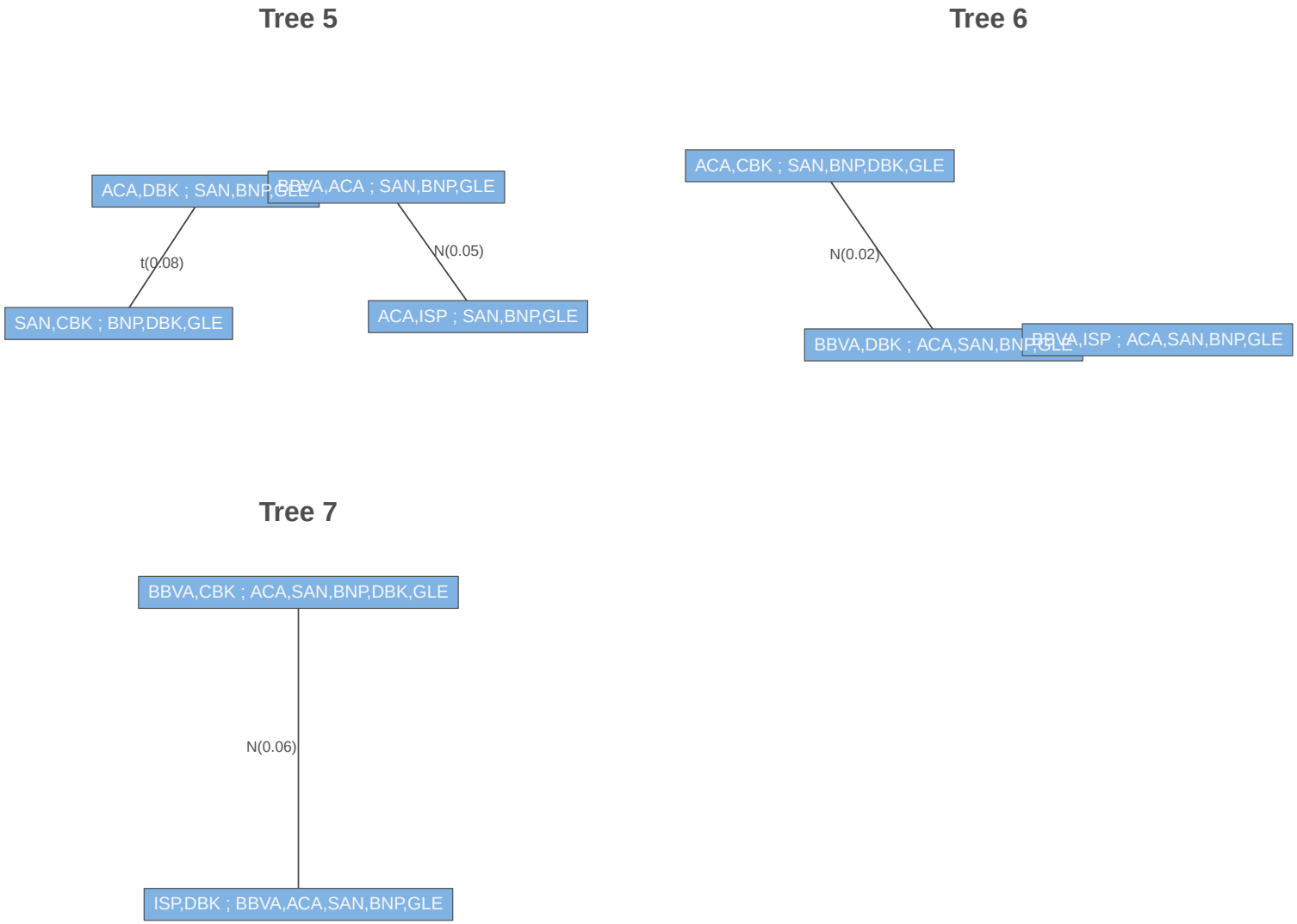








Trees 5 to 7 of rv.sub



Parametric vine copula analysis using all candidate copula families

Fit R-vine using all pair copula families without and with independence test

```
rv<-RVineStructureSelect(udata.cop, family=NA,
                  method = "mle")
rv.ind<-RVineStructureSelect(udata.cop, family=NA,
                  method = "mle", indeptest=TRUE)
# suppress console output
dummy<-capture.output(sum.rv<- summary(rv))
dummy<-capture.output(sum.rv.ind<- summary(rv.ind))
```

Results of first two trees of rv

1 <-> ACA, 2 <-> BBVA, 3 <-> BNP, 4 <-> CBK,

5 <-> DBK, 6 <-> GLE, 7 <-> ISP, 8 <-> SAN

```
print(sum.rv[1:14, ], digits = 3)
```

```
##    tree    edge family cop   par   par2   tau    utd    ltd
## 1     1     8,3      2   t 0.797  5.505 0.587 0.4212 0.4212
## 2     1     8,2      2   t 0.902  3.853 0.715 0.6384 0.6384
## 3     1     3,6      2   t 0.864  4.369 0.665 0.5576 0.5576
## 4     1     6,1      2   t 0.839  4.400 0.634 0.5208 0.5208
## 5     1     6,7      2   t 0.752  5.054 0.541 0.3896 0.3896
## 6     1     6,5      2   t 0.785  4.082 0.575 0.4690 0.4690
## 7     1     6,4      2   t 0.733  5.038 0.524 0.3721 0.3721
## 8     2   3,2;8      2   t 0.275  6.068 0.177 0.0846 0.0846
## 9     2   8,6;3      2   t 0.326  8.613 0.212 0.0526 0.0526
## 10    2   3,1;6      2   t 0.302  7.072 0.195 0.0708 0.0708
## 11    2   3,7;6      7 BB1 0.203  1.114 0.185 0.1367 0.0464
## 12    2   3,5;6      2   t 0.270 10.436 0.174 0.0257 0.0257
## 13    2   5,4;6      5   F 2.737  0.000 0.284 0.0000 0.0000
## 14    3 2,6;8,3     10 BB8 1.693  0.805 0.145 0.0000 0.0000
```

Results of trees 2 to 5 of rv

1 <-> ACA, 2 <-> BBVA, 3 <-> BNP, 4 <-> CBK,

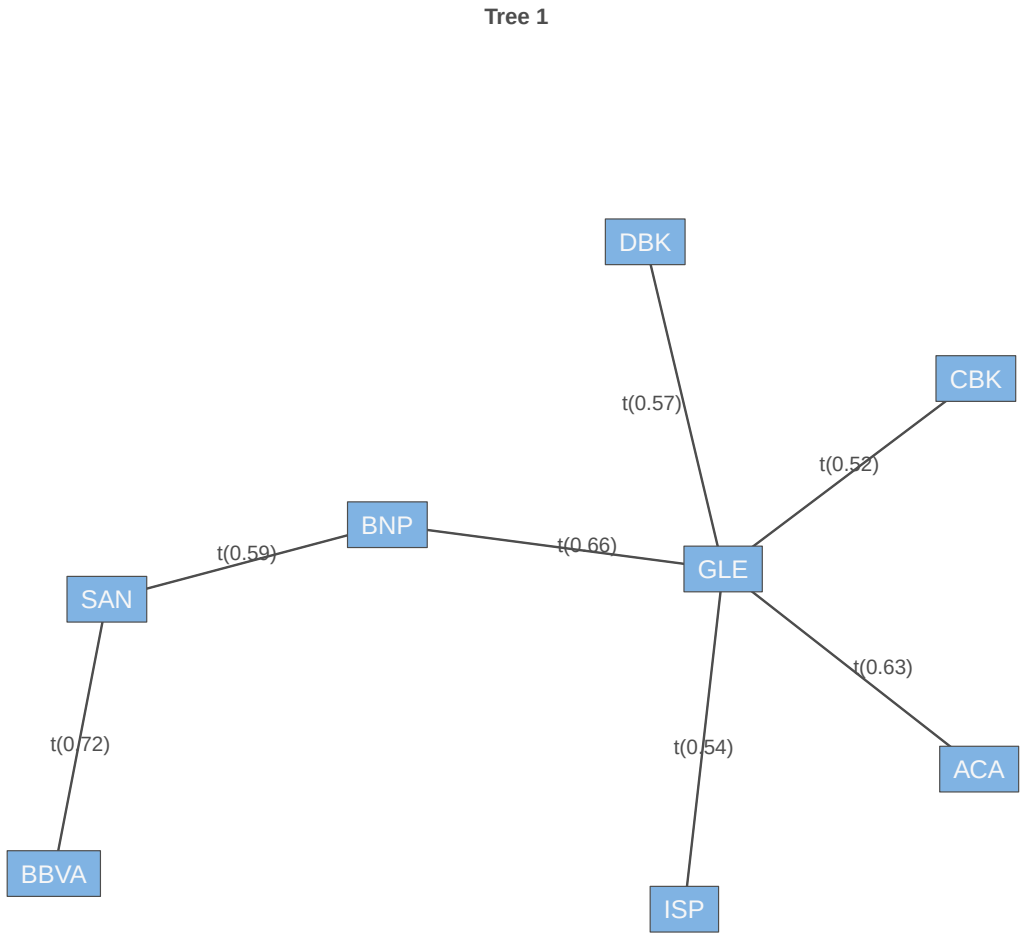
5 <-> DBK, 6 <-> GLE, 7 <-> ISP, 8 <-> SAN

```
print(sum.rv[15:28, ], digits = 2)
```

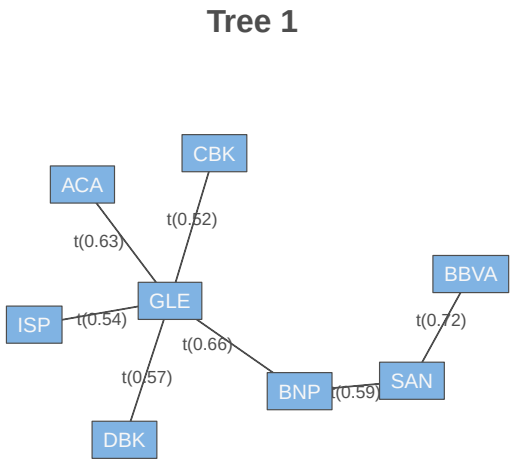
```
##    tree           edge family  cop   par    par2     tau     utd     ltd
## 15    3        8,1;3,6     20 SBB8 1.950  0.6826 1.4e-01 0.0e+00 0.0e+00
## 16    3        8,7;3,6      2    t 0.256 17.8509 1.6e-01 3.5e-03 3.5e-03
## 17    3        8,5;3,6      5    F 1.290  0.0000 1.4e-01 0.0e+00 0.0e+00
## 18    3        3,4;5,6     10  BB8 1.111  0.9201 3.8e-02 0.0e+00 0.0e+00
## 19    4      2,1;8,3,6      7  BB1 0.059  1.0364 6.3e-02 4.8e-02 1.3e-05
## 20    4      1,7;8,3,6      5    F 0.783  0.0000 8.7e-02 0.0e+00 0.0e+00
## 21    4      1,5;8,3,6      5    F 0.671  0.0000 7.4e-02 0.0e+00 0.0e+00
## 22    4      8,4;3,5,6      5    F 0.949  0.0000 1.0e-01 0.0e+00 0.0e+00
## 23    5    2,7;1,8,3,6      1    N 0.085  0.0000 5.4e-02 0.0e+00 0.0e+00
## 24    5    2,5;1,8,3,6      2    t 0.061 16.0869 3.9e-02 1.2e-03 1.2e-03
## 25    5    1,4;8,3,5,6      5    F 1.030  0.0000 1.1e-01 0.0e+00 0.0e+00
## 26    6  7,5;2,1,8,3,6      5    F 0.410  0.0000 4.6e-02 0.0e+00 0.0e+00
## 27    6  2,4;1,8,3,5,6    104 Tawn 9.254  0.0001 6.2e-12 1.0e-04 0.0e+00
## 28    7 7,4;2,1,8,3,5,6      2    t 0.098 28.6734 6.2e-02 2.8e-05 2.8e-05
```

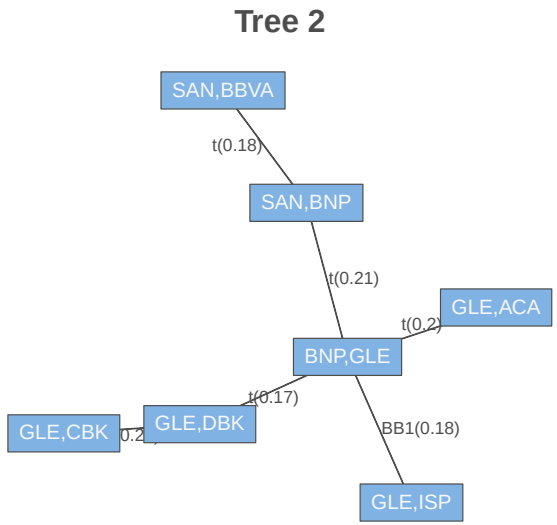
Tree 1 of rv

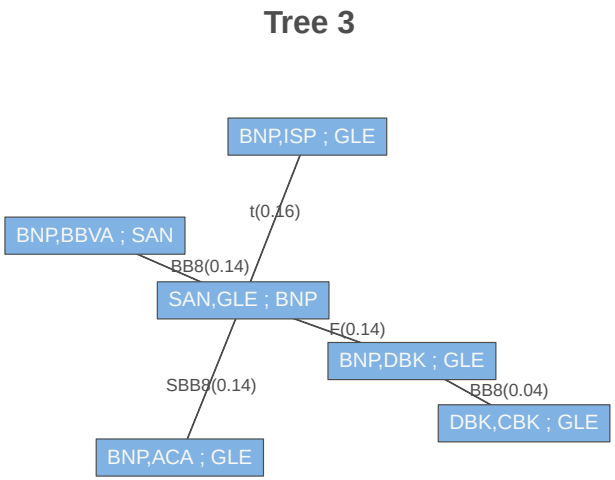
```
plot(rv,tree=1,edge.labels="family-tau",type=1)
```

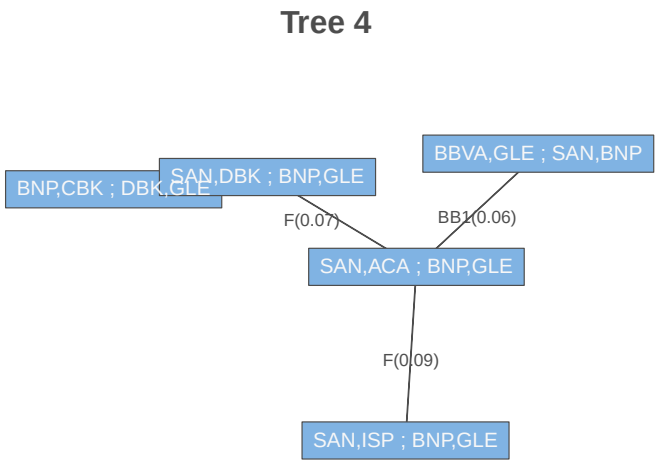


Trees 1 to 4 of rv

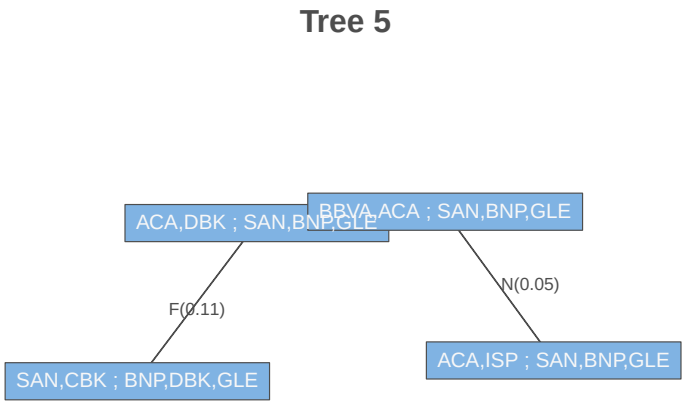


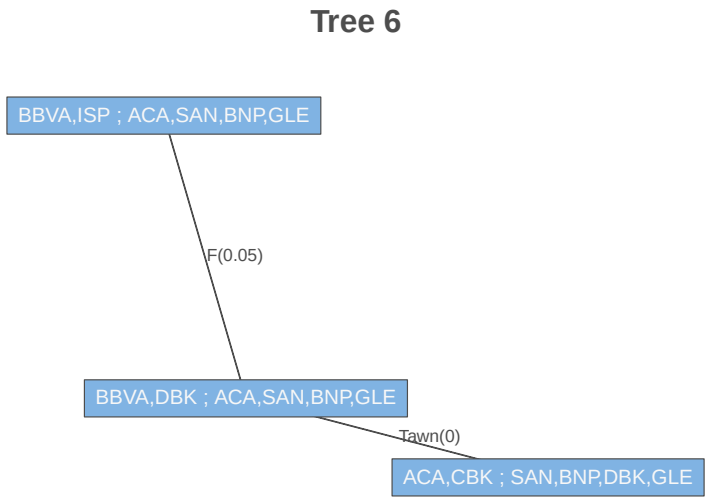




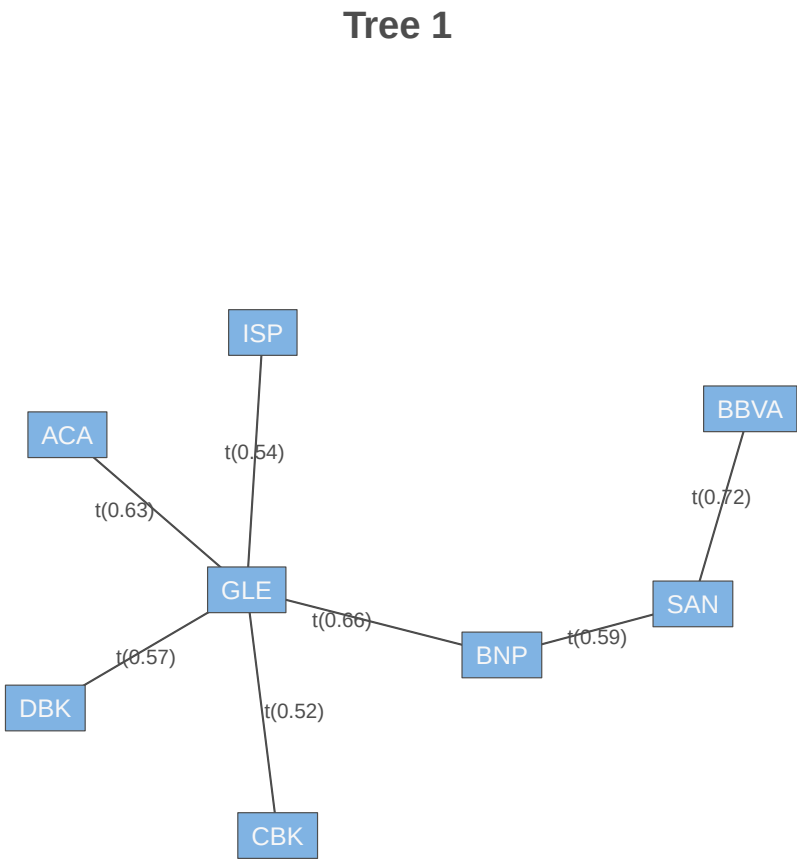


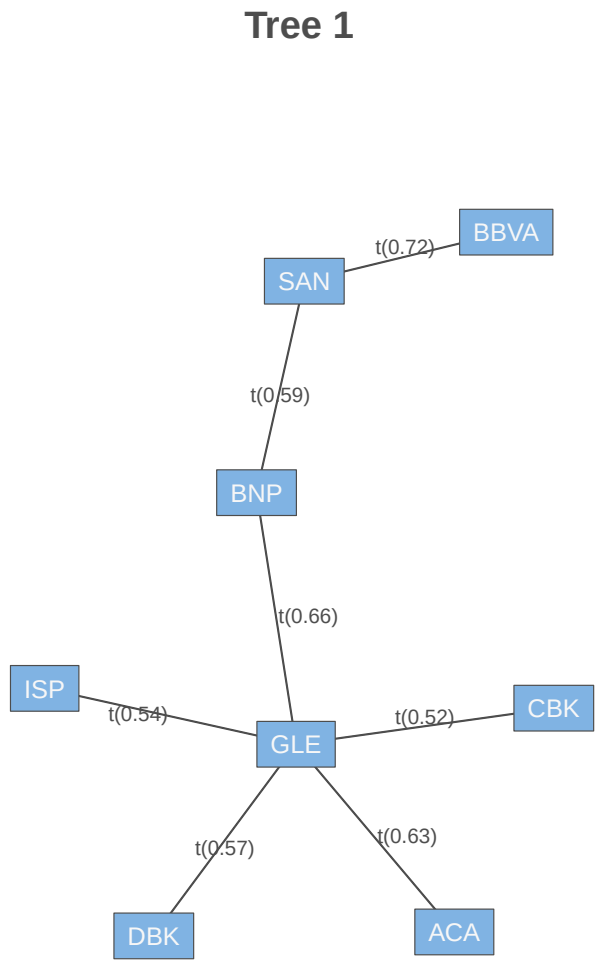
Trees 5 to 7 of rv





Comparing first tree of R-vine with 4 and unrestricted candidate pair copulas





Fitting C-vines

Fitting C-vines using RVineStructureSelect

```
cv.fsub<-RVineStructureSelect(udata.cop, family=1:4,
                 method = "mle",type="CVine")
cv.fsub.ind<-RVineStructureSelect(udata.cop, family=1:4,
                 method = "mle",type="CVine",indeptest = TRUE)
cv<-RVineStructureSelect(udata.cop, family=NA,
                 method = "mle",type="CVine")
cv.ind<-RVineStructureSelect(udata.cop, family=NA,
                 method = "mle",type="CVine",indeptest = TRUE)
```

Fitting D-vines

Fitting D-vines using the TSP package

```
d = dim(udata.cop)[2]
M = 1 - abs(TauMatrix(udata.cop))
hamilton = insert_dummy(TSP(M),label="cut")
sol = solve_TSP(hamilton,method="repetitive_nn")
order = cut_tour(sol,"cut")
DVM= D2RVine(order,family=rep(0,d*(d-1)/2),par=rep(0,d*(d-1)/2))
dv.fsub=RVineCopSelect(data=udata.cop,
        familyset=1:4,Matrix=DVM$Matrix)
dv.fsub.ind=RVineCopSelect(data=udata.cop,familyset=1:4,
                        indeptest=TRUE, Matrix=DVM$Matrix)
dv=RVineCopSelect(data=udata.cop,familyset=NA,Matrix=DVM$Matrix)
dv.ind=RVineCopSelect(data=udata.cop,familyset=NA,
                        indeptest=TRUE, Matrix=DVM$Matrix)
```

Comparison of R-vines, C-vines and D-vines

Output function

```
vine.out<-function(fit=out12.3fixed,data=udata.f3,digits=2){
df<-sum(abs(fit$par)>0)+sum(fit$par2>0)
out<-round(c(RVineLogLik(data, fit)$loglik,df,
             RVineAIC(data,fit)$AIC,
             RVineBIC(data,fit)$BIC),digits)
names(out)<-c("loglik","par","AIC","BIC")
out
}
```

Apply output function to all fitted vine models

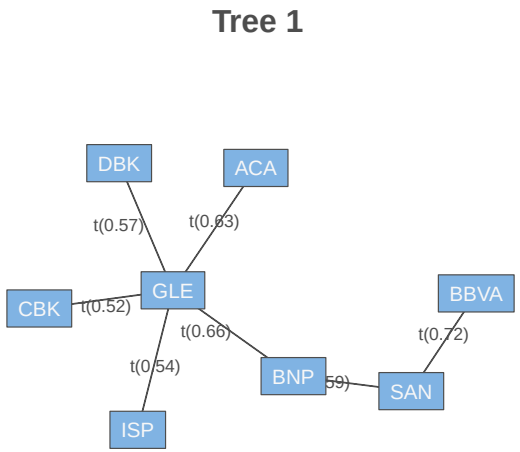
```
out.table<-rbind(
vine.out(fit=rv.fsub,data=udata.cop),
vine.out(fit=rv.fsub.ind,data=udata.cop),
vine.out(fit=rv,data=udata.cop),
vine.out(fit=rv.ind,data=udata.cop),
vine.out(fit=cv.fsub,data=udata.cop),
vine.out(fit=cv.fsub.ind,data=udata.cop),
vine.out(fit=cv,data=udata.cop),
vine.out(fit=cv.ind,data=udata.cop),
vine.out(fit=dv.fsub,data=udata.cop),
vine.out(fit=dv.fsub.ind,data=udata.cop),
vine.out(fit=dv,data=udata.cop),
vine.out(fit=dv.ind,data=udata.cop)
)
row.names(out.table)<-c("rv.fsub","rv.fsub.ind","rv","rv.ind",
  "cv.fsub","cv.fsub.ind","cv","cv.ind",
  "dv.fsub","dv.fsub.ind","dv","dv.ind")
```

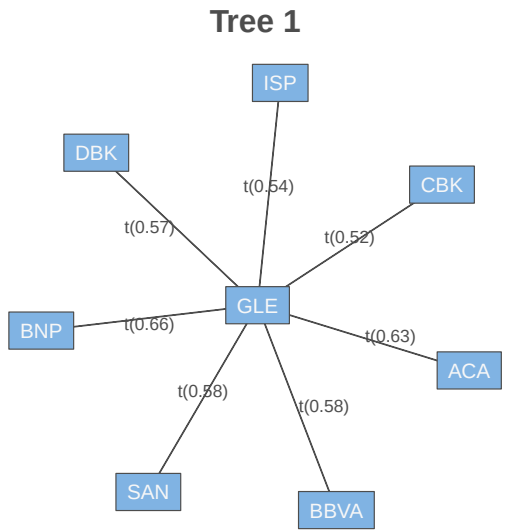
Comparison of all fitted vine models

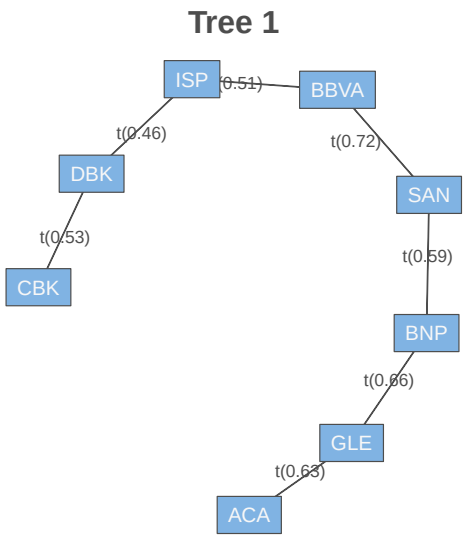
```
round(out.table,digits=2)
```

```
##                loglik par       AIC       BIC
## rv.fsub      5688.25  47 -11282.51 -11040.28
## rv.fsub.ind 5686.90  45 -11283.80 -11051.88
## rv           5729.65  48 -11363.29 -11115.91
## rv.ind       5724.21  44 -11360.41 -11133.64
## cv.fsub      5684.91  47 -11275.82 -11033.59
## cv.fsub.ind 5677.10  43 -11268.19 -11046.58
## cv           5710.42  47 -11326.83 -11084.60
## cv.ind       5703.89  43 -11321.79 -11100.17
## dv.fsub      5675.02  49 -11252.03 -10999.49
## dv.fsub.ind 5675.02  49 -11252.03 -10999.49
## dv           5712.61  47 -11331.21 -11088.98
## dv.ind       5712.61  47 -11331.21 -11088.98
```

Comparing first tree of R-vine, C-vine and D-vine pair







Storing vine specifications

R-vine matrix of rv

1 <-> ACA, 2 <-> BBVA, 3 <-> BNP, 4 <-> CBK,

5 <-> DBK, 6 <-> GLE, 7 <-> ISP, 8 <-> SAN

Column 1 dentifies the following edges:

47;2183556, 42;18356, 41;8365; 48;356, 43;56 45;6 46

```
rv$Matrix
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8]
## [1,]    4    0    0    0    0    0    0    0
## [2,]    7    5    0    0    0    0    0    0
## [3,]    2    7    7    0    0    0    0    0
## [4,]    1    2    2    1    0    0    0    0
## [5,]    8    1    1    2    6    0    0    0
## [6,]    3    8    8    8    2    2    0    0
## [7,]    5    3    3    3    8    3    3    0
## [8,]    6    6    6    6    3    8    8    8
```

C-vine matrix of rv

1 <-> ACA, 2 <-> BBVA, 3 <-> BNP, 4 <-> CBK,

5 <-> DBK, 6 <-> GLE, 7 <-> ISP, 8 <-> SAN

```
cv$Matrix
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8]
## [1,]    2    0    0    0    0    0    0    0
## [2,]    4    3    0    0    0    0    0    0
## [3,]    3    4    4    0    0    0    0    0
## [4,]    7    7    7    1    0    0    0    0
## [5,]    1    1    1    7    5    0    0    0
## [6,]    5    5    5    5    7    7    0    0
## [7,]    8    8    8    8    8    8    6    0
## [8,]    6    6    6    6    6    6    8    8
```

D-vine matrix of rv

1 <-> ACA, 2 <-> BBVA, 3 <-> BNP, 4 <-> CBK,

5 <-> DBK, 6 <-> GLE, 7 <-> ISP, 8 <-> SAN

```
dv$Matrix
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8]
## [1,]    4    0    0    0    0    0    0    0
## [2,]    1    5    0    0    0    0    0    0
## [3,]    6    1    7    0    0    0    0    0
## [4,]    3    6    1    2    0    0    0    0
## [5,]    8    3    6    1    8    0    0    0
## [6,]    2    8    3    6    1    3    0    0
## [7,]    7    2    8    3    6    1    6    0
## [8,]    5    7    2    8    3    6    1    1
```