Copula and Joint Probability Analysis

Hassan Mir

2024-06-15

In this part, I fit copulas to find out joint probability from a built-in financial dataset in R.

```
warning = FALSE
library('gmm')
```

```
## Warning: package 'gmm' was built under R version 4.3.3
```

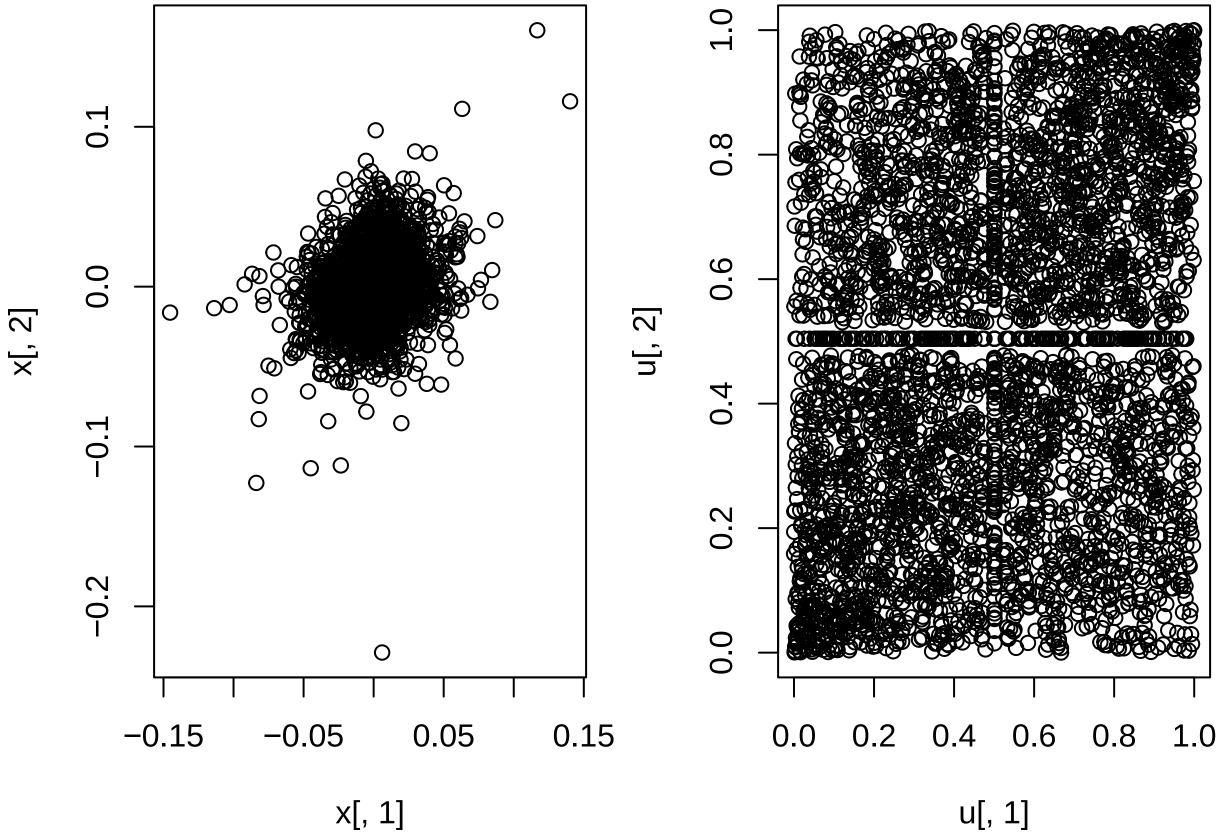
```
## Loading required package: sandwich
```

```
library('copula')
```

```
## Warning: package 'copula' was built under R version 4.3.3
```

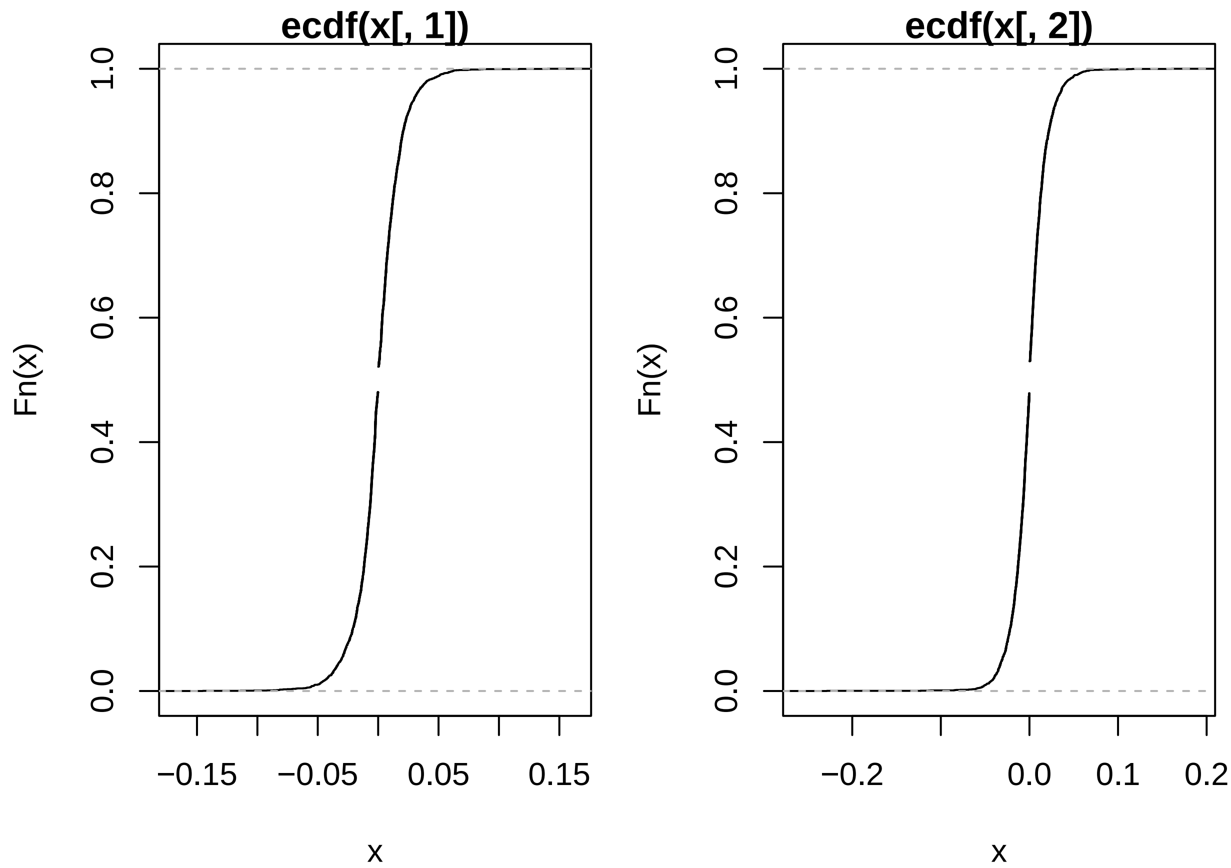
Loadig data from the Finance dataset and calculating the returns of companies with tickers T and PC. Then I convert returns to log-returns and bind the x's column-wise. Then I convert the observations to pseudo-observations abd obtain empirical cumulative distribution functions. Finally I plot the results.

```
data(Finance)
x1<-Finance[,'T']/100
x2<-Finance[,'PC']/100

x1<-log(x1+1)
x2<-log(x2+1)

n<-dim(Finance)[1]

x<-cbind(x1,x2)

u <- pobs(x)

par(mfrow=c(1,2),mar=c(4, 4, 1, 1))
plot(x=x[,1],y=x[,2])
plot(x=u[,1],y=u[,2])
```



Computing empirical cumulative distribution function, ECDF.

```
F1<-ecdf(x[,1])
F2<-ecdf(x[,2])

par(mfrow=c(1,2),mar=c(4, 4, 1, 1))
plot(F1);plot(F2)
```



Now I fit Clayton copula to the pseudo-observations and estimate the joint probability.

```
myfit.clayton<-fitCopula(copula=claytonCopula(param=0.1,dim=2),
                         data=u,method='itau')
myfit.clayton
```

```
## Call: fitCopula(claytonCopula(param = 0.1, dim = 2), data = u, ... = pairlist(method = "itau"))
## Fit based on "inversion of Kendall's tau" and 4012 2-dimensional observations.
## Copula: claytonCopula
##  alpha
## 0.3461
```

```
attributes(myfit.clayton)
```

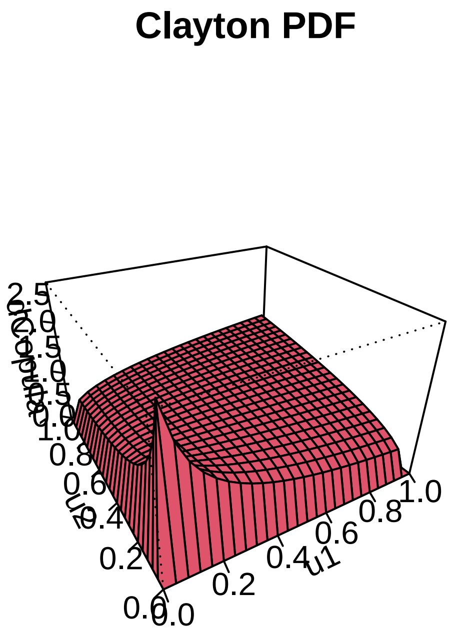
```
## $copula
## Clayton copula, dim. d = 2
## Dimension:  2
## Parameters:
##   alpha    = 0.3461085
##
## $estimate
## [1] 0.3461085
##
## $var.est
##             [,1]
## [1,] 0.000903225
##
## $loglik
```

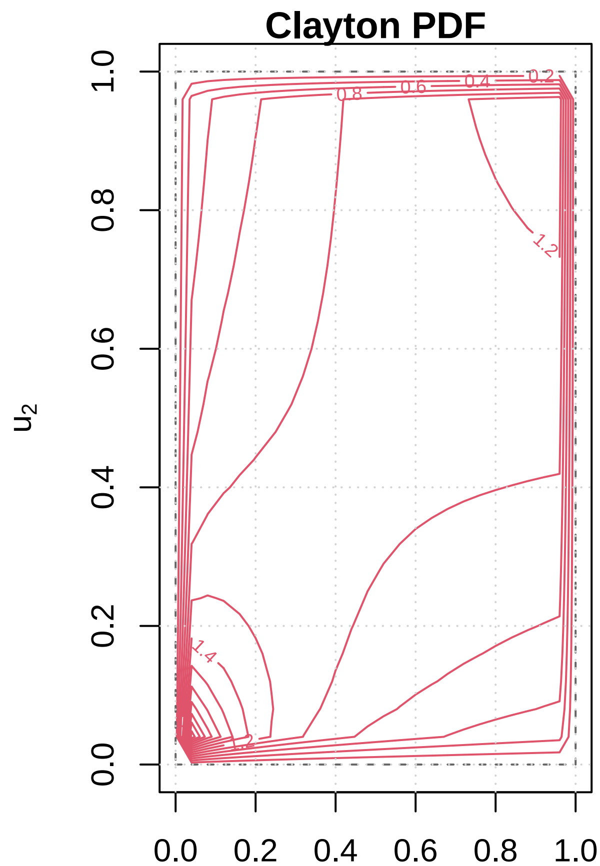
```
## [1] NA
##
## $nsample
## [1] 4012
##
## $method
## [1] "inversion of Kendall's tau"
##
## $call
## fitCopula(copula = claytonCopula(param = 0.1, dim = 2), data = u,
##     ... = pairlist(method = "itau"))
##
## $fitting.stats
## $fitting.stats$convergence
## [1] NA
##
##
## $class
## [1] "fitCopula"
## attr(,"package")
## [1] "copula"
```

```
myfit.clayton@estimate
```

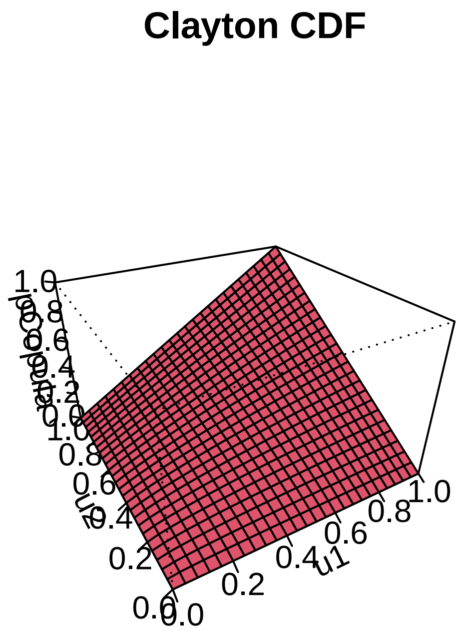
```
## [1] 0.3461085
```

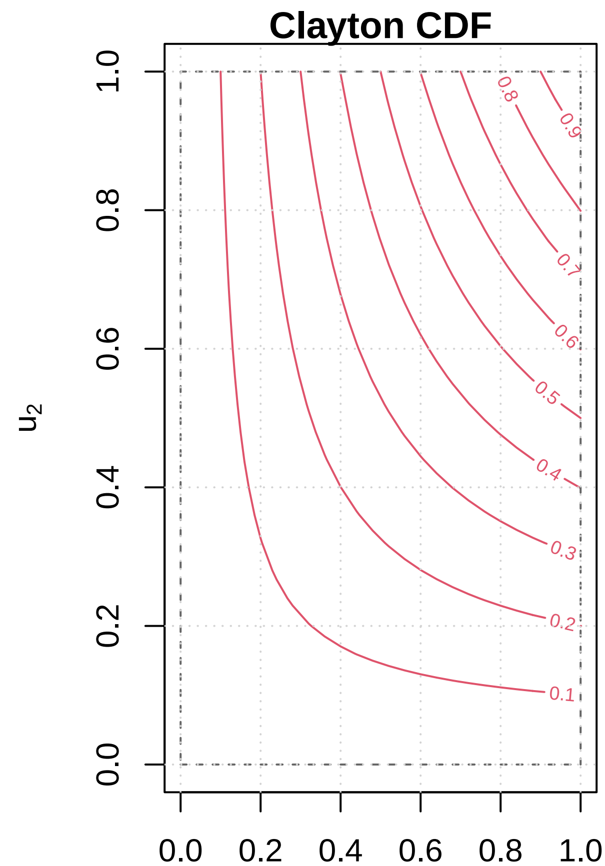
```
mycopula.clayton <- claytonCopula(param=myfit.clayton@estimate,dim=2)

par(mfrow=c(1,2),mar=c(2, 4, 1, 1))
persp(mycopula.clayton,dCopula,col=2,main='Clayton PDF')
contour(mycopula.clayton,dCopula,col=2,main='Clayton PDF');grid()
```






```r
par(mfrow=c(1,2),mar=c(2, 4, 1, 1))
persp(mycopula.clayton,pCopula,col=2,main='Clayton CDF')
contour(mycopula.clayton,pCopula,col=2,main='Clayton CDF');grid()
```






```
d.clayton<-dCopula(u=u,copula=mycopula.clayton)
p.clayton<-pCopula(u=u,copula=mycopula.clayton)

x1<- -0.1
x2<- -0.15
u0<-matrix(c(F1(x1),F2(x2)),nrow=1)
#convert x1 and x2 to pobs (using ECDFs) and save as a 1x2 matrix
p0.clayton<-pCopula(u=u0,copula=mycopula.clayton) #Prob(X1<=x1,X2<=x2)
p0.clayton
```

```
## [1] 6.106767e-05
```