4 | Multivariate Statistical Models

Suppose we are interested in daily returns on two or more assets, then the joint behavior is described by multivariate distributions.

Covariance and Correlation Matrices

Let $\boldsymbol{Y} = (Y_1, ..., Y_d)^T$ be a d-dimensional random vector, then

$$\boldsymbol{\mu} = E(\boldsymbol{Y}) = \{E(Y_1), \ldots, E(Y_d)\}^T$$

and the variance covariance matrix of $\boldsymbol{Y}$ is

$$\mathrm{Cov}(\boldsymbol{Y}) = E\left\{(\boldsymbol{Y} - \boldsymbol{\mu})(\boldsymbol{Y} - \boldsymbol{\mu})^T\right\}.$$

Let σ_i, $i = 1, \ldots, d$ be the standard deviation of Y_i, the correlation matrix of $\boldsymbol{Y}$ is

$$\mathrm{Corr}(\boldsymbol{Y}) = \mathrm{diag}^{-1}\{\sigma_1, \ldots, \sigma_d\}\,\mathrm{Cov}(\boldsymbol{Y})\,\mathrm{diag}^{-1}\{\sigma_1, \ldots, \sigma_d\}.$$

The (i, j)th entry of $\mathrm{Cov}(\boldsymbol{Y})$ is $\mathrm{Cov}(Y_i, Y_j) = E[(Y_i - \mu_i)(Y_j - \mu_j)]$ and the (i, j)th correlation matrix is $\rho_{ij} = \mathrm{Cov}(Y_i, Y_j)/\sigma_i\sigma_j$.

Suppose that $d_1 \times 1$ dimensional $\boldsymbol{Y}_1$ and $d_2 \times 1$ dimensional $\boldsymbol{Y}_2$ are two random vectors withe mean vectors $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_2$, then the covariance matrix of $\boldsymbol{Y}_1$ and $\boldsymbol{Y}_2$ is a $d_1 \times d_2$ matrix

$$\mathrm{Cov}(\boldsymbol{Y}_1, \boldsymbol{Y}_2) = E\left\{(\boldsymbol{Y}_1 - \boldsymbol{\mu}_1)(\boldsymbol{Y}_2 - \boldsymbol{\mu}_2)^T\right\}.$$

Linear functions of random variables Suppose that $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$ are two constant vectors having the same dimensions of $\boldsymbol{Y}_1$ and $\boldsymbol{Y}_2$, then

$$\mathrm{Cov}(\boldsymbol{w}_1^T\boldsymbol{Y}_1, \boldsymbol{w}_2^T\boldsymbol{Y}_2) = \boldsymbol{w}_1^T\mathrm{Cov}(\boldsymbol{Y}_1, \boldsymbol{Y}_2)\boldsymbol{w}_2 .$$

The special cases of $\boldsymbol{Y}_1 = \boldsymbol{Y}_2 = \boldsymbol{Y}$ or/and $\boldsymbol{w}_1 = \boldsymbol{w}_2 = \boldsymbol{w}$ give

$$\begin{aligned}\mathrm{Cov}(\boldsymbol{w}_1^T\boldsymbol{Y}, \boldsymbol{w}_2^T\boldsymbol{Y}) &= \boldsymbol{w}_1^T\mathrm{Cov}(\boldsymbol{Y})\boldsymbol{w}_2 ,\\ \mathrm{var}(\boldsymbol{w}^T\boldsymbol{Y}) &= \boldsymbol{w}^T\mathrm{Cov}(\boldsymbol{Y})\boldsymbol{w} .\end{aligned} \tag{4.1}$$

Similar covariance expression can be obtained when $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$ are replaced by $d_1 \times q_1$ and $d_2 \times q_2$ matrices $\boldsymbol{W}_1$ and $\boldsymbol{W}_2$.

$$\mathrm{Cov}(\boldsymbol{W}_1^T\boldsymbol{Y}_1, \boldsymbol{W}_2^T\boldsymbol{Y}_2) = \boldsymbol{W}_1^T\mathrm{Cov}(\boldsymbol{Y}_1, \boldsymbol{Y}_2)\boldsymbol{W}_2 ,$$

which is of dimension $q_1 \times q_2$. For the cases that $\boldsymbol{Y}_1 = \boldsymbol{Y}_2 = \boldsymbol{Y}$ or/and $\boldsymbol{W}_1 = \boldsymbol{W}_2 = \boldsymbol{W}$,

$$\mathrm{Cov}(\boldsymbol{W}_1^T\boldsymbol{Y}\ \boldsymbol{W}_2^T\boldsymbol{Y}) = \boldsymbol{W}_1^T\mathrm{Cov}(\boldsymbol{Y})\,\boldsymbol{W}_2 , \quad \mathrm{Cov}(\boldsymbol{W}^T\boldsymbol{Y}) = \boldsymbol{W}^T\mathrm{Cov}(\boldsymbol{Y})\,\boldsymbol{W} . \tag{4.2}$$

This is a square variance covariance matrix.

The Multivariate Normal Distribution

The random variable $\boldsymbol{Y} = (Y_1, \ldots, Y_d)^T$ has a d-dimensional multivariate normal distribution with mean $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_d)^T$ and variance covariance matrix $\boldsymbol{\Sigma}$, denoted by $\boldsymbol{Y} \sim N_d(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, if its probability density function is

$$\phi_d(\boldsymbol{y}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{d/2}|\boldsymbol{\Sigma}|^{1/2}} \exp\left\{-\frac{1}{2}(\boldsymbol{y}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{y}-\boldsymbol{\mu})\right\} \tag{4.3}$$

where $|\boldsymbol{\Sigma}|$ is the determinant of $\boldsymbol{\Sigma}$. If the variance covariance matrix $\boldsymbol{\Sigma}$ is diagonal, $\boldsymbol{\Sigma} = \mathrm{diag}\{\sigma_1^2, \ldots, \sigma_d^2\}$, then it can be shown algebraically that

$$\phi_d(\boldsymbol{y}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \prod_{i=1}^{d} \phi_1(y_i|\mu_i, \sigma_i^2) ,$$

implying that y_i's, $i = 1, \ldots, d$, the components of $\boldsymbol{y}$, are independent.

The density (4.3) depends on $\boldsymbol{y}$ only through the quadratic form $(\boldsymbol{y}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{y}-\boldsymbol{\mu})$, the density is constant on each ellipse

$$\left\{\boldsymbol{y} : (\boldsymbol{y}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{y}-\boldsymbol{\mu}) = c\right\}, \qquad c > 0 .$$

The constant determines the size of the ellipse, with larger values of c gives larger ellipses, each entered at $\boldsymbol{\mu}$. Such density are called elliptically contoured.

If $\boldsymbol{Y} \sim N_d(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, then following from (4.1) and (4.2), for any $d \times 1$ constant vector $\boldsymbol{b} \neq \boldsymbol{0}$ and constant a, $\boldsymbol{b}^T\boldsymbol{Y} + a$ is a univariate normal distribution

$$\boldsymbol{b}^T\boldsymbol{Y} + a \sim N(\boldsymbol{b}^T\boldsymbol{\mu} + a, \boldsymbol{b}^T\boldsymbol{\Sigma}\boldsymbol{b}) . \tag{4.4}$$

Furthermore, for any $d \times q$ constant matrix $\boldsymbol{B}$ and $q \times 1$ constant $\boldsymbol{a}$, $q \leq d$, the random vector $\boldsymbol{B}^T\boldsymbol{Y} + \boldsymbol{a}$ has a q dimensional normal distribution,

$$\boldsymbol{B}^T\boldsymbol{Y} + \boldsymbol{a} \sim N_q(\boldsymbol{B}^T\boldsymbol{\mu} + \boldsymbol{a}, \boldsymbol{B}^T\boldsymbol{\Sigma}\boldsymbol{B}) .$$

These facts will be used to find linear transform of a multivariate-t distribution, which will be defined next.

The Multivariate t-Distribution

The random variable $\boldsymbol{Y} = (Y_1, \ldots, Y_d)^T$ has a d-dimensional multivariate $t_\nu(\boldsymbol{\mu}, \boldsymbol{\Lambda})$ distribution if

$$\boldsymbol{Y} = \boldsymbol{\mu} + \frac{1}{\sqrt{W/\nu}} \boldsymbol{Z} ,$$

where $W \sim \chi^2_\nu$ and $\boldsymbol{Z} \sim N_d(\boldsymbol{0}, \boldsymbol{\Lambda})$. For $\nu > 1$, the vector $\boldsymbol{\mu}$ is the mean of $\boldsymbol{Y}$. For any value of ν, $\boldsymbol{\mu}$ contains the medians of the components of $\boldsymbol{Y}$ and the contours of the density of $\boldsymbol{Y}$ are ellipses centered at $\boldsymbol{\mu}$. For $\nu > 2$, the covariance matrix of $\boldsymbol{Y}$ exists and is

$$\boldsymbol{\Sigma} = \frac{\nu}{\nu - 2} \boldsymbol{\Lambda} . \tag{4.5}$$

The matrix $\boldsymbol{\Lambda}$ is called scale matrix. If $\Sigma_{i,j} = 0$ then Y_i and Y_j are uncorrelated, but they are dependent because of the tail dependence, outliers in one component tend to occur with outliers in other components.

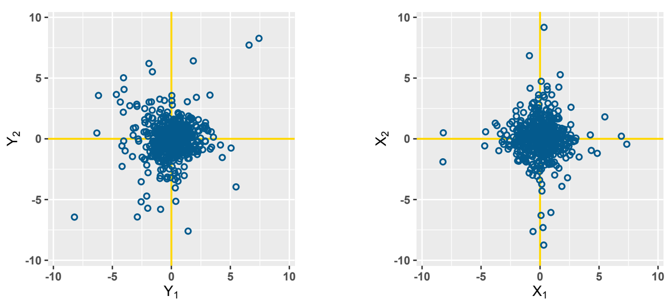


Figure 4.1: The left penal is the plot of a bivariate $t_3(\boldsymbol{0}, \mathbf{I}_2)$. The right panel is the plot of two independent t_3 random variables. Both sample sizes are 500.

The right panel in Figure 4.1 are scatter plot of 500 pairs independent t_3 random variables, outliers are concentrated near the x- and

$\boldsymbol{y}$- axes, outliers in Y_1 and outliers in Y_2 are not associated. While in the left panel, outliers are uniformly in all directions.

Tail dependence is common among stock returns and the multivariate t-distribution can be a model for them, see the scatterplots of weekly returns on Adobe, Microsoft, Oracle and Qualcomm from Jan. 1, 2007 to Aug 31, 2024 in Figure 4.2.

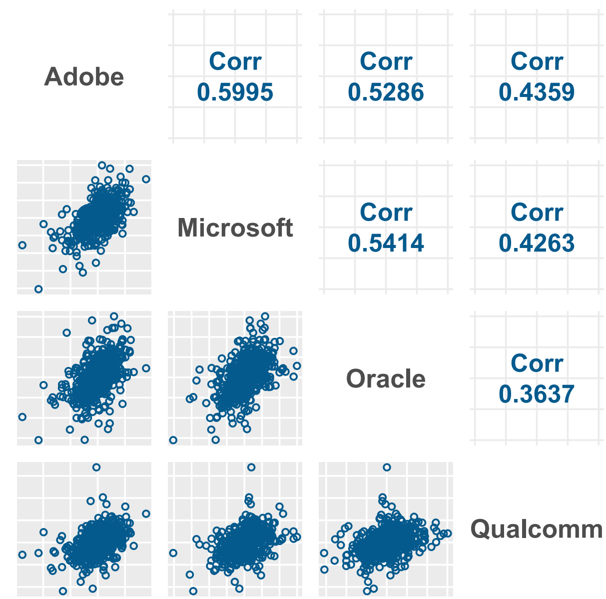


Figure 4.2: Scatter plot matrix of weekly returns on Adobe, Microsoft, Oracle and Qualcomm from Jan. 1, 2007 to Aug 31, 2024. The sample size $n = 922$.

If $\boldsymbol{Y} \sim t_\nu(\boldsymbol{\mu}, \boldsymbol{\Lambda})$ distribution and $\boldsymbol{w}$ is a vector of weights, then following from (4.4), $\boldsymbol{w}^T\boldsymbol{Y}$ has a univariate t-distribution

$$\boldsymbol{w}^T\boldsymbol{Y} \sim t_\nu(\boldsymbol{w}^T\boldsymbol{\mu}, \boldsymbol{w}^T\boldsymbol{\Lambda}\boldsymbol{w}) .$$

If $\nu > 2$, then $\text{var}(\boldsymbol{w}^T\boldsymbol{Y}) = \boldsymbol{w}^T\boldsymbol{\Sigma}\boldsymbol{w}$. This fact is useful to compute risk measures for a portfolio. If returns on the assets have a multivariate t-distribution, then the return on the portfolio will have a univariate t-distribution. We will use this result later.

The Multivariate Skewed t-Distributions

Azzalini and Capitanio (2003) have proposed a skewed extension of the multivariate t-distribution. The univariate special case was discussed in Handout 4. In the multivariate case, the shape parameter $\boldsymbol{\alpha}$ is a vector parameter determining the amounts of skewness in the components of the distribution. If $\boldsymbol{Y}$ has a skewed-t distribution, then each component Y_i is left- or right-skewed depending on $\alpha_i < 0$ or $\alpha_i > 0$ and is symmetric if $\alpha_i = 0$.

Fitting Multivariate Distributions by Maximum Likelihood

Fitting a multivariate distribution by maximum likelihood can be challenging computationally. A d-dimensional model can have d parameters for the location vector and $d(d+1)/2$ parameters for the scale matrix without further restrictions. The data we will use is the 4 weekly returns shown in Figure 4.2.

```
head(yt,2)

##                 ADBE       MSFT      ORCL     QCOM
## 2007-01-05 -1.223407 -0.7394916  2.875422 2.353678
## 2007-01-12 -1.638160  5.1613376 -0.796784 2.375241

tail(yt,2)
```



```
##                 ADBE        MSFT     ORCL     QCOM
## 2024-08-23 0.8706909 -0.40226874 1.243420 0.763718
## 2024-08-30 2.8446950  0.08394137 1.497454 1.032125

n = dim(yt)[1]; d = dim(yt)[2]; c(n = n, d = d)
##   n   d
## 922   4
```

Fitting the Multivariate Normal Distribution

Just like the univariate case, the maximum of the multivariate normal likelihood can be solved analytically. Suppose $\boldsymbol{Y}$ is a d-dimensional normal random vector and $\mathbf{Y}$ an $n\times d$ observed data of sample size n, the MLE of the mean $\boldsymbol{\mu}$ and variance covariance matrix $\boldsymbol{\Sigma}$ are simply

$$\overline{\boldsymbol{Y}} = (\overline{Y}_1, \ldots, \overline{Y}_d)^T \quad \text{and} \quad \boldsymbol{S} = \frac{1}{n}\mathbf{Y}^T\mathbf{Y}.$$

```
apply(yt, 2, mean) ## compute column means

##      ADBE      MSFT      ORCL      QCOM
## 0.2859923 0.3225330 0.2514464 0.2113622

var(yt) ## sample variance; same as cov(yt)

##           ADBE      MSFT      ORCL      QCOM
## ADBE 18.680195  8.983414  8.098641  8.459433
## MSFT  8.983414 12.019660  6.653186  6.636909
## ORCL  8.098641  6.653186 12.586587  5.802162
## QCOM  8.459433  6.636909  5.802162 20.189255

var(yt[,1], yt[,2:4]) ## covariance of (y1,y2) and (y3,y4)

##          MSFT     ORCL     QCOM
## ADBE 8.983414 8.098641 8.459433
```

R computes sample variance or covariance, they are normalized by $n-1$ instead of n for MLE. Correlation can be computed with `cor()`.

Fitting the Multivariate t-distribution

There is no analytical solution for the MLE of a multivariate t-distribution, however, the computation can be simplified by the profile likelihood of the distribution. See appendix for the profile likelihood.

Preliminary analysis before fitting By definition, the marginal distributions of a multivariate t are all t-distributions with the same degrees of freedom, ν. Before deciding to model our data with a multivariate t-distribution, we should explore the marginal distributions of the data. We use `fitdistr()` to fit t distribution to each return series.

```
library(MASS)
nu = c(); se = c(); skew = c(); kurt = c()
for(i in 1:d){
        skew[i] = Sk.fun(yt[,i]);kurt[i] = Kur.fun(yt[,i])
        start = list(m = mean(yt[,i]), s = sd(yt[,i]), df = 4)
        fit = fitdistr(yt[,i], "t", start, lower = c(-100,0.0001,0.1))
        nu[i] = fit$est["df"]
        se[i] = fit$sd["df"]
}
stat = cbind(nu,se, nu-qnorm(.975)*se, nu + qnorm(.975)*se, skew, kurt)
rownames(stat) = syb
colnames(stat) = c("nu","std err", "lower 95%", "upper 95%",
                   "Skewness", "Kurtosis")
stat

##            nu   std err lower 95% upper 95%    Skewness Kurtosis
## ADBE 4.165771 0.6134780  2.963377  5.368166 -0.78361063 6.963727
## MSFT 4.024008 0.5718403  2.903221  5.144794 -0.31262881 6.302570
## ORCL 4.057068 0.6089788  2.863491  5.250644 -0.09577001 5.248066
## QCOM 3.865381 0.5416682  2.803731  4.927031  0.17023851 8.510995
```

The 95% CIs for ν indicate the DF among the 4 distributions are roughly the same. We also calculate the sample skewness and kur-

tosis as we will also consider the Multivariate skewed-t distribution. The estimates also show the degrees of freedom ν are significantly greater than 2, which is required for using the R functions for the profile likelihood method.

Profile likelihood method To estimate the parameters of a multivariate t-distribution, one can use the R function `cov.trob()` in R's MASS package. This function computes the MLE of $\boldsymbol{\mu}$ and $\boldsymbol{\Lambda}$ with a fixed value of ν, $\nu > 2$. That is, the profile likelihood of ν.

```
library(MASS)
args(cov.trob)

## function (x, wt = rep(1, n), cor = FALSE, center = TRUE, nu = 5,
##     maxit = 25, tol = 0.01)
```

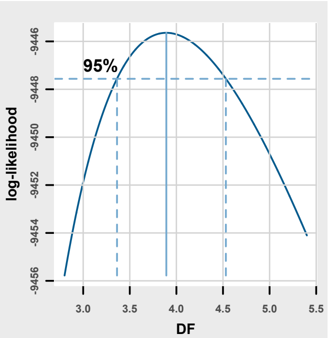
For example `cov.trob(x, nu = 5)` computes the MLE of mean and scale matrix of fitting data x to a multivariate-t with DF $= 5$. To find the MLE of ν, we compute the profile log-likelihood for a sequence of candidate values of ν along with their corresponding MLE for mean and scale using the R function `dmt()` in R's `mnormt` package.

```
library(mnormt)
args(dmt)

## function (x, mean = rep(0, d), S, df = Inf, log = FALSE)
```

This R function `dmt()` computes the joint density of a multivariate t for input data x, with specified `mean`, `S` and `df`. The argument S is the scale matrix $\boldsymbol{\Lambda}$.

Eg 4.1. Profile likelihood method for fitting multivariate t distribution.

```
library(MASS)  # needed for cov.trob
library(mnormt)  # needed for dmt
df = seq(2.8,5.4, 0.01) ## candidate values of DF
loglik_p = c()
for(i in 1:length(df)){
  fit = cov.trob(yt,nu=df[i])   ## MLE of mu and Lambda for a fixed df
  loglik_p[i] = sum(dmt(yt,mean = fit$center, S = fit$cov, df=df[i],
                       log = T))
}
nu = df[which.max(loglik_p)];  ## The df gives the max of loglik_p
cat("The MLE of degrees of freedom:", paste(nu) )

## The MLE of degrees of freedom: 3.89
```



The maximum profile likelihood estimates of the degrees of freedom is $\hat{\nu} = 3.89$. Setting `nu = 3.89` in `cov.trob()`, we will obtain the MLE of mean `$center` and scale matrix `$cov`. The correlation estimate is optional. We stress again, despite the name `$cov`, it is not the estimate of the covariance matrix.

The 95% CI for ν is $(3.37, 4.53)$, it is calculated by using the formula (4.6) in Appendix.

```
ci.df<-df[loglik_p > (max(loglik_p)-qchisq(.95,1)/2)]
ci = c(min(ci.df), max(ci.df,1))
cat("95% CI for nu:", ci)

## 95% CI for nu: 3.37 4.53
```



```
est = cov.trob(yt, nu = nu, cor = T)
names(est)

## [1] "cov"    "center" "n.obs"  "cor"    "call"   "iter"

est$center  ## MLE of mean vector

##      ADBE      MSFT      ORCL      QCOM
## 0.4311421 0.4276957 0.2755423 0.3637282

est$cov     ## MLE of scale matrix Lambda

##           ADBE     MSFT     ORCL      QCOM
## ADBE 10.196218 4.946401 4.818932  5.003304
## MSFT  4.946401 6.546973 3.877660  3.854161
## ORCL  4.818932 3.877660 7.094651  3.615896
## QCOM  5.003304 3.854161 3.615896 10.800534

cat("The MLE of Cov:\n"); est$cov*nu/(nu-2);    ## Compute covariance

## The MLE of Cov:
##          ADBE      MSFT      ORCL      QCOM
## ADBE 20.98587 10.180688  9.918330 10.297806
## MSFT 10.18069 13.474986  7.981005  7.932638
## ORCL  9.91833  7.981005 14.602218  7.442241
## QCOM 10.29781  7.932638  7.442241 22.229671
```

The returned values `$cov` is the estimate of $\boldsymbol{\Lambda}$ in (4.3). The last computation is the covariance matrix estimate of the fitted multivariate t distribution, see (4.5). The number of parameters with a dimension $d = 4$ multivariate t is $p = 15$ which is required for computing the AIC and BIC.

```
cat("Fitting Multivariate t  ##\n")
## Fitting Multivariate t  ##
d = dim(yt)[2]; n = dim(yt)[1]  ## d: dimension of rt, n: sample size
p = d*(d+1)/2 + d + 1 ## p: number of parameters
cat(paste(c("d","n", "p"), c(d,n,p), sep = ": "), sep = ", ")
## d: 4, n: 922, p: 15
aic_mt = -2*max(loglik_p) + 2*p
bic_mt = -2*max(loglik_p) + log(n)*p
cat(paste(c("aic","bic"), round(c(aic_mt,bic_mt),2), sep = " = "), sep = ", ")
## aic = 18921.28, bic = 18993.68
```

Fitting the Multivariate Skewed t-Distributions

The previous univariate analysis for checking equal degrees of freedom is sufficient for the skewed case. We still show the fits of A-C univariate skewered t, which is defined different from those of F-S. The R package sn is provided by the authors including both univariate and multivariate skewed t distribution functions and fit functions which however do not give standard error or Hessian. We use `fitdistr()` to fit the A-C univariate skewed t and show only the skew (α) and DF (ν) parameter estimates.

```
library(sn)  ## A-C skewed t package
args(dst)  ## density of univariate A-C skewed t

## function (x, xi = 0, omega = 1, alpha = 0, nu = Inf, dp = NULL,
##     log = FALSE)

## fit universite skewed t to each series
start = list(xi = 0, omega = 1, alpha = 0, nu = 4)
lower =  c(-100, .001, -3, 0.1)
fits = vector("list", length = d)
for(i in 1:d) fits[[i]] = fitdistr(yt[,i], dst, start, lower = lower)

## Omit commands of getting estimates together
cat("alpha estimates:\n"); alpha; cat("nu estimates:\n");nu

## alpha estimates:
##             est        se  lower 95%    upper 95%
## ADBE -0.3897164 0.2014761 -0.7846022  0.005169443
## MSFT -0.1292434 0.1891762 -0.5000220  0.241535218
## ORCL -0.1396378 0.1861046 -0.5043961  0.225120590
## QCOM -0.4357137 0.1825270 -0.7934601 -0.077967280
## nu estimates:
##           est        se lower 95% upper 95%
## ADBE 4.357600 0.6782240  3.028306  5.686895
## MSFT 4.028297 0.5738813  2.903510  5.153083
## ORCL 4.046896 0.6050767  2.860967  5.232824
## QCOM 3.808659 0.5234839  2.782649  4.834669
```


The estimates of ν are similar to those of symmetric t, only QCOM's skew estimate is significant. Recall that an A-C distribution is skewed only if $\alpha = 0$. We now fit the multivariate skewed t-model to the weekly returns of the 4 stocks using the function `mst.mple()` in R's sn package.

```
args(mst.mple)

## function (x, y, start = NULL, w, fixed.nu = NULL, symmetr = FALSE,
##     penalty = NULL, trace = FALSE, opt.method = c("nlminb", "Nelder-Mead",
##         "BFGS", "CG", "SANN"), control = list())

fit_st = mst.mple(y = yt)
names(fit_st)

## [1] "call"        "dp"          "dp.complete" "logL"        "boundary"
## [6] "aux"         "opt.method"

fit_st$dp

## $beta
##          ADBE      MSFT      ORCL     QCOM
## [1,] 1.039871 0.8979921 0.3159133 1.202567
##
## $Omega
##           ADBE     MSFT     ORCL      QCOM
## ADBE 10.457776 5.160666 4.830463  5.404994
## MSFT  5.160666 6.724456 3.888806  4.162432
## ORCL  4.830463 3.888806 7.075192  3.636138
## QCOM  5.404994 4.162432 3.636138 11.336213
##
## $alpha
##       ADBE       MSFT       ORCL       QCOM
## -0.1643118 -0.1562824  0.2756865 -0.2799110
##
## $nu
## [1] 3.87623
```

The argument x is for the designed matrix, the data to be fit is specified to the second argument y. The direct parameter estimates in the retuned value do not reveal information about the location or scale. If the ν estimate is greater than 4 so that the distribution has finite fourth moment, the `dp2cp()` function can be used to obtain the mean vector, covariance matrix and skewness of each component. Our ν estimate is 3.876 which is not satisfied the finite fourth moment condition for applying `dp2cp()`. The estimate of ν is very close to that of the symmetric multivariate-t model, 3.89. The log-likelihood evaluated at the MLE is stored in `logL` which is required for computing the AIC and BIC.

```
cat("Fitting Multivariate skew t  ##\n")

## Fitting Multivariate skew t  ##

p.st = d*(d+1)/2 + d + 1 + d  ## number of parameters of a multivariate skew t
cat("Number of parameters:", paste(p.st))

## Number of parameters: 19

aic_st = -2*fit_st$logL + 2*p.st
bic_st = -2*fit_st$logL + log(n)*p.st
cat("skewed multivariate t:\n"); c(aic = aic_st, bic = bic_st)

## skewed multivariate t:
##      aic      bic
## 18922.26 19013.96
```

The number of parameters is 19 in this model. The AIC $= 18922.26$ and BIC $= 119013.96$. Both are higher than those of the multivariate-t model having AIC $= 18921.28$ and BIC $= 18933.68$. Adding four additional skewness parameters does not sufficiently improve the fit. This result suggests that the symmetric t-model is the preferable model for this data set.

Appendix

Profile likelihood Consider a univariate model with parameter $\boldsymbol{\theta} = (\theta_1, \boldsymbol{\theta}_2^T)^T$, where θ_1 is the parameter of interest. The profile likelihood function is given by

$$L_p(\theta_1) = L\{\hat{\boldsymbol{\theta}}_2(\theta_1)\} = \max_{\boldsymbol{\theta}_2} L(\theta_1, \boldsymbol{\theta}_2).$$

The RHS means the $L(\theta_1, \boldsymbol{\theta}_2)$ is maximized over $\hat{\boldsymbol{\theta}}_2$ with θ_1 fixed to create a function of θ_1 only. The MLE of θ_1 is the value $\hat{\theta}_1$ that maximizes $L_p(\theta_1)$ and the MLE of $\hat{\boldsymbol{\theta}}_2$ is $\boldsymbol{\theta}_2(\hat{\theta}_1)$. All the likelihoods $L(\cdot)$ can be replaced by $\log L(\cdot)$ in computation.

Let $\theta_{0,1}$ be hypothesized value of θ_1. By the theory of likelihood ratio tests, one accepts the null hypothesis $H_0 : \theta_1 = \theta_{0,1}$ if

$$L_p(\theta_{0,1}) > L_p(\hat{\theta}_1) - \frac{1}{2}\chi^2_{\alpha,1},$$

where $\chi^2_{\alpha,1}$ is the α-upper quantile of the chi-squared distribution with DF $= 1$. The profile likelihood confidence interval for θ_1 is the set of all null values that would be accepted,

$$\left\{\theta_1 : L_p(\theta_1) > L_p(\hat{\theta}_1) - \frac{1}{2}\chi^2_{\alpha,1}\right\} \tag{4.6}$$

The profile likelihood can be defined for a subset of the parameters, rather than for just a single parameter.The profile likelihood can also be defined for a subset of the parameters, rather than for just a single parameter.

Eg 4.2. Revisit Eg. 3.10. Suppose only μ of the normal is of interest.

$$\frac{\partial}{\partial\sigma^2}\log L(\sigma^2) = -\frac{n}{2}\frac{1}{\sigma^2}+\frac{1}{2\sigma^4}\sum_{i=1}^{n}(Y_i-\mu)^2 \overset{\text{set}}{=} 0$$

$$\Longrightarrow \hat\sigma^2(\mu)=\frac{1}{n}\sum_{i=1}^{n}(Y_i-\mu)^2$$

$$\log L_p(\mu) = -\frac{n}{2}\log\left\{\frac{1}{n}\sum_{i=1}^{n}(Y_i-\mu)^2\right\}+\text{constant} \qquad (4.7)$$

The MLE $\hat\mu$ is obtained by maximizing $\log L_p(\mu)$, we get $\hat\mu=\bar Y$ the same as before. The MLE of σ^2 is then

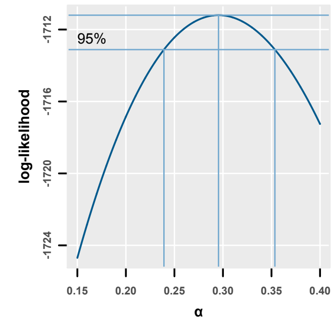
$$\hat\sigma^2(\hat\mu)=\frac{1}{n}\sum_{i=1}^{n}(Y_i-\bar Y)^2.$$

Eg 4.3. Estimating a Box-Cox transformation. Assume that for some values of α, μ and σ^2, the transformed data $Y_1^{(\alpha)}, Y_2^{(\alpha)}, \dots Y_n^{(\alpha)}$ are $i.i.d.$ $N(\mu,\sigma^2)$ distributed. All 3 parameters can be estimated by maximum likelihood. For a fixed value of α the MLE of μ and σ^2,

$$\hat\mu(\alpha)=\frac{1}{n}\sum_{i=1}^{n}Y_i^{(\alpha)} \quad\text{and}\quad \hat\sigma^2(\alpha)=\frac{1}{n}\sum_{i=1}^{n}\{Y_i^{(\alpha)}-\hat\mu(\alpha)\}^2$$

as shown in Eg. 4.2. These values can be plugged into the log-likelihood to obtained the profile log-likelihood for α.

This can be easily done with the function `boxcox()` in R's `MASS` package. We illustrate the use of `boxcox()` with a simulated sample of 512 χ^2_2 random variables $Y_1,\dots,Y_{512}$. The estimate $\hat\alpha = 0.2954$.




```
library(MASS)
set.seed(996052752)
n = 512; y = rchisq(n,2)  ## simulated data
a =  seq(0.15, 0.4, 0.0001)  ## a sequence of candidates for alpha
bc = boxcox(y~1,  lambda = a) ## Compute profile lik for each a
names(bc) ## output $x is a's; $y is profile lik at each a

## [1] "x" "y"

alpha = bc$x[which.max(bc$y)];  ## $x with max $y is the MLE
cat("MLE of alpha:", alpha)

## MLE of alpha: 0.2954
```

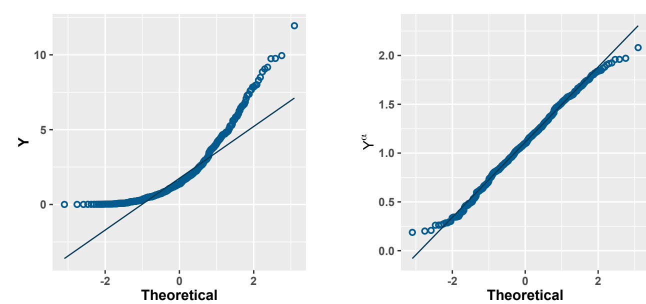


Figure 4.3: Normal probability plots of 512 simulated χ^2_2 random data before, the left panel, and after, the right panel Box-Cox transformation.