Using the **MarkowitzR** package

Steven E. Pav *

January 7, 2020

Abstract

The asymptotic distribution of the Markowitz portfolio can be found via the central limit theorem and delta method. This allows one to construct Wald statistics on the elements of the Markowitz portfolio or perform shrinkage on those elements. This technique allows the use of robust standard errors; can be extended to deal with hedged portfolios, conditional hetereskedasticity and expectation; allows estimation of the proportion of error due to mis-estimation of the covariance matrix. Example computations via the **MarkowitzR** package are illustrated.

1 Introduction

Given p assets with expected return $\boldsymbol{\mu}$ and covariance of return Σ, define the 'Markowitz portfolio' as

$$\boldsymbol{\nu}_* =_{\mathrm{df}} \Sigma^{-1}\boldsymbol{\mu}. \tag{1}$$

In practice, the population parameters $\boldsymbol{\mu}$ and Σ are not known and must be estimated from samples. Use of the central limit theorem and the delta method gives asymptotic normality of the vector $\boldsymbol{\nu}_*$, with a covariance that may be estimated from the data. [12] For essentially no extra computational work, one also gets an estimate of the covariance of $\boldsymbol{\nu}_*$ with Σ^{-1}.

2 Example usage

2.1 Fake data

The variance-covariance matrix of 'inverse theta' is computed by `itheta_vcov`; a fancier, more general version is given by `mp_vcov`. The primary use case of the variance-covariance matrix is to compute the marginal Wald test statistics under the null of zero weight in the Markowitz portfolio. The basic procedure is illustrated here:

```
require(gtools)
# set the colnames of X appropriately
set.coln <- defmacro(X, expr = {
    if (is.null(colnames(X))) {
```

* shabbychef@gmail.com

```r
        colnames(X) <- paste0(deparse(substitute(X),
            nlines = 1), 1:(dim(X)[2]))
    }
})
# compute Wald scores via this trick
wrap.itheta <- function(rets, ret.what = c("wald",
    "mp"), ...) {
    set.coln(rets)
    ret.what <- match.arg(ret.what)
    # 'flipping the sign on returns is idiomatic'
    asymv <- itheta_vcov(-as.matrix(rets), ...)
    qidx <- 2:asymv$pp
    mp <- asymv$mu[qidx]
    stat <- switch(ret.what, mp = mp, wald = mp/sqrt(diag(asymv$Ohat[2:asymv$pp,
        2:asymv$pp])))
    names(stat) <- colnames(rets)
    return(stat)
}
wrap.mp <- function(rets, ret.what = c("wald", "mp"),
    ...) {
    set.coln(rets)
    ret.what <- match.arg(ret.what)
    asymv <- mp_vcov(as.matrix(rets), ...)
    mp <- t(asymv$W)
    stat <- switch(ret.what, mp = mp, wald = mp/sqrt(diag(asymv$What)))
    return(stat)
}
# t-stat via Britten-Jones procedure
bjones.ts <- function(rets) {
    set.coln(rets)
    ones.vec <- matrix(1, nrow = dim(rets)[1], ncol = 1)
    rets <- as.matrix(rets)
    bjones.mod <- lm(ones.vec ~ rets - 1)
    bjones.sum <- summary(bjones.mod)
    retval <- bjones.sum$coefficients[, 3]
    names(retval) <- colnames(rets)
    return(retval)
}
# compare the procedures
do.both <- function(rets, ...) {
    set.coln(rets)
    retval <- rbind(bjones.ts(rets), wrap.itheta(rets,
        ...))
    rownames(retval) <- c("Britten Jones t-stat", "via itheta_vcov")
    return(retval)
}
do.all <- function(rets, ...) {
    set.coln(rets)
    retval <- do.both(rets, ...)
    retval <- rbind(retval, wrap.mp(rets, ...))
    rownames(retval)[dim(retval)[1]] <- "via mp_vcov"
    return(retval)
}
```

```r
n.day <- 1000
n.stock <- 5
```

First some example usage under randomly generated data to compare against the t-statistics produced by the Britten-Jones procedure. [2] The tests are performed on 1000 observations of 5 assets, first under the null of zero mean returns, then under the alternative. The results of these two tests are very close, as is typically the case under Gaussian returns when the ratio of days to assets is large. This is a way of "sanity checking" the implementation of itheta_vcov, mp_vcov and the Wald tests.

```r
# under the null: all returns are zero mean;
set.seed(as.integer(charToRaw("7af85b0b-e521-4059-bebe-55ad9a9a0456")))
rets <- matrix(rnorm(n.day * n.stock), nrow = n.day)
# compare them:
print(do.all(rets))

##                         rets1 rets2 rets3 rets4 rets5
## Britten Jones t-stat     0.47 0.048   1.4 -0.47  -1.4
## via itheta_vcov          0.47 0.048   1.4 -0.48  -1.4
## via mp_vcov              0.47 0.048   1.4 -0.48  -1.4

# returns under the alternative
set.seed(as.integer(charToRaw("464dcbea-375b-49d0-8f38-b48b5a33c7ea")))
rets <- matrix(rnorm(n.day * n.stock, mean = 0.1),
    nrow = n.day)
print(do.all(rets))

##                         rets1 rets2 rets3 rets4 rets5
## Britten Jones t-stat        2   2.3   3.2   3.0   4.1
## via itheta_vcov             2   2.3   3.2   2.9   4.0
## via mp_vcov                 2   2.3   3.2   2.9   4.0
```

We should, and do, see that the two procedures, itheta_vcov and mp_vcov return the same values. The latter is simply a more elaborate version of the former. The functions itheta_vcov and mp_vcov also compute the Markowitz portfolio itself, as below. Note that for this example, the mean return of each stock is 0.1, and the covariance matrix is I, thus the Markowitz portfolio is uniformly 0.1.

```r
# returns under the alternative
set.seed(as.integer(charToRaw("464dcbea-375b-49d0-8f38-b48b5a33c7ea")))
rets <- matrix(rnorm(n.day * n.stock, mean = 0.1),
    nrow = n.day)
print(wrap.itheta(rets, ret.what = "mp"))

## rets1 rets2 rets3 rets4 rets5
## 0.064 0.077 0.104 0.095 0.133
```

2.1.1 Covariance with the precision matrix

Since this estimation procedure computes the covariance jointly of the Markowitz portfolio and the precision matrix, we can estimate the amount of error in the Markowitz portfolio which is attributable to mis-estimation of the covariance. The remainder we can attribute to mis-estimation of the mean vector, which, is typically implicated as the leading effect. [3] Here, for each of the 5 members of the Markowitz portfolio, I estimate the squared coefficient of multiple correlation, expressed as percents.

```r
# multiple correlation coefficients of portfolio
# error to precision errors.
mult.cor <- function(rets, ...) {
    set.coln(rets)
    # 'flipping the sign on returns is idiomatic'
    asymv <- itheta_vcov(-as.matrix(rets), ...)
    Ro <- cov2cor(asymv$Ohat)
    prec.idx <- (asymv$p + 1):(dim(asymv$Ohat)[1])
    prec.Ro <- Ro[prec.idx, prec.idx]
    xcor <- Ro[2:asymv$p, prec.idx]
    R.sq <- diag(xcor %*% (solve(prec.Ro, t(xcor))))
}
set.seed(as.integer(charToRaw("464dcbea-375b-49d0-8f38-b48b5a33c7ea")))
rets <- matrix(rnorm(n.day * n.stock, mean = 0.08),
    nrow = n.day)
print(signif(100 * mult.cor(rets), digits = 2))

##  x rets1  x rets2  x rets3  x rets4  x rets5
##      4.4      4.6      4.2      6.1      5.8
```

Thus, in the case, misestimation of the covariance matrix is contributing around 5 percent of the error in the elements of the Markowitz portfolio. The remainder we attribute to misestimation of the mean vector. Note that when the population maximal Sharpe ratio is larger, the *proportion* of error in the Markowitz portfolio attributed to misestimation of the precision matrix increases, even though the total error in the elements of the Markowitz portfolio is decreasing:

```r
set.seed(as.integer(charToRaw("464dcbea-375b-49d0-8f38-b48b5a33c7ea")))
rets <- matrix(rnorm(n.day * n.stock, mean = 0.16),
    nrow = n.day)
print(signif(100 * mult.cor(rets), digits = 2))

##  x rets1  x rets2  x rets3  x rets4  x rets5
##       14       12       12       16       16
```

2.1.2 Conditional expectations

Now consider the model where the mean return is conditional on some features observable prior to the period where the investment decision is required. The Markowitz portfolio is replaced by the 'Markowitz coefficient' which linearly scales the features into a portfolio. [12]

Here we generate such a population, then check the Markowitz coefficient by scattering the fit coefficients against the population values in Figure 1.

```
# generate data with given W, Sigma
Xgen <- function(W, Sigma, Feat) {
    Btrue <- Sigma %*% W
    Xmean <- Feat %*% t(Btrue)
    Shalf <- chol(Sigma)
    X <- Xmean + matrix(rnorm(prod(dim(Xmean))), ncol = dim(Xmean)[2]) %*%
        Shalf
}
n.feat <- 4
n.ret <- 8
n.obs <- 2000
set.seed(12321)

Feat <- matrix(rnorm(n.obs * n.feat), ncol = n.feat)
Wtrue <- 10 * matrix(rnorm(n.feat * n.ret), ncol = n.feat)
Sigma <- cov(matrix(rnorm(100 * n.ret), ncol = n.ret))
Sigma <- Sigma + diag(seq(from = 1, to = 3, length.out = n.ret))
X <- Xgen(Wtrue, Sigma, Feat)
ism <- mp_vcov(X, feat = Feat, fit.intercept = TRUE)

# a bit of legerdemain b/c there's an intercept
# term fit
Wcomp <- cbind(0, Wtrue)
```

```
# scatter them against each other
w.true <- Wcomp
w.fit <- ism$W
dim(w.true) <- c(length(w.true), 1)
dim(w.fit) <- c(length(w.fit), 1)
plot(w.true, w.fit, main = "Markowitz coefficient",
    xlab = "True Value ", ylab = "Fit Value ", pch = 1)
# abline(lm(w.fit ~ w.true), col='red')
abline(a = 0, b = 1, col = "green")
```

2.1.3 Asymptotic normality

The techniques employed here give not just the asymptotic covariance of the Markowitz portfolio, but also claim asymptotic normality. Here I compute the vector of errors in the Markowitz coefficient, and transform them to have approximate unit covariance, then Q-Q plot them against the normal in Figure 2.

```
n.feat <- 4
n.ret <- 8
n.obs <- 2000
set.seed(12321)
Feat <- matrix(rnorm(n.obs * n.feat), ncol = n.feat)
Wtrue <- 10 * matrix(rnorm(n.feat * n.ret), ncol = n.feat)
Sigma <- cov(matrix(rnorm(100 * n.ret), ncol = n.ret))
Sigma <- Sigma + diag(seq(from = 1, to = 3, length.out = n.ret))
X <- Xgen(Wtrue, Sigma, Feat)
ism <- mp_vcov(X, feat = Feat, fit.intercept = TRUE)
```

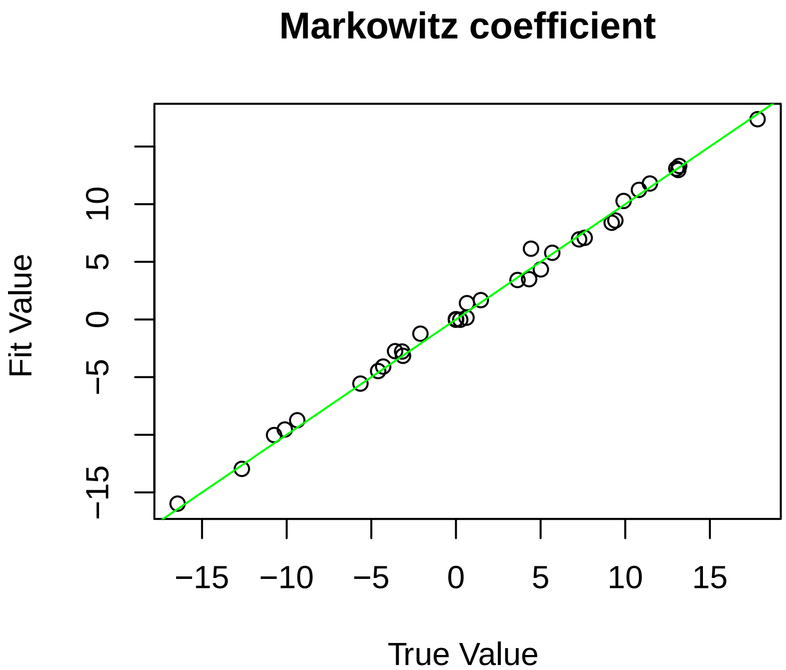

Figure 1: Scatter of the fit value against the true value of the Markowitz Coefficien is shown, for 8 assets, and 4 predictive features, given 2000 days of observations of Gaussian returns. The $y = x$ line is plotted in green.

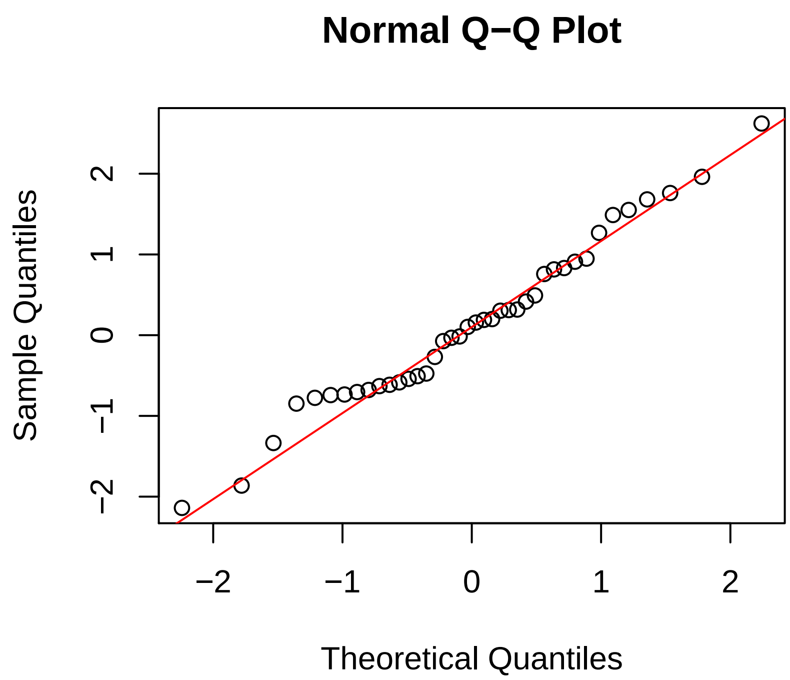

Figure 2: Sample quantiles of the error of the Markowitz coefficient, transformed to approximate unit covariance using the estimated covariance, are plotted against those of the normal.

```
Wcomp <- cbind(0, Wtrue)
# compute the errors
errs <- ism$W - Wcomp
dim(errs) <- c(length(errs), 1)
# transform them to approximately identity
# covariance
Zerr <- solve(t(chol(ism$What)), errs)
# did it work?
qqnorm(Zerr)
qqline(Zerr, col = 2)
```

2.1.4 Portfolio subspace constraints

Now consider the *constrained* portfolio which must be (for whatever reason) the linear combination of two ‘baskets’ of the underlying assets: one the ‘market’, the other having some exposure to some fake factor. The portfolio and Wald statistics are computed as follows:

```
# first for the case where the real Markowitz
# Portfolio is actually just 'the market': equal
```

```
# dollar long in each stock.
set.seed(as.integer(charToRaw("dbeebe3f-da7e-4d11-b014-feac88a1d6cb")))
rets <- matrix(rnorm(n.day * n.stock, mean = 0.1),
    nrow = n.day)
Jmat <- matrix(c(1, 1, 1, -1), nrow = 2, ncol = 2 *
    n.stock)  # overbuilt
Jmat <- Jmat[, 1:n.stock]  # fixed.
print(Jmat)

##      [,1] [,2] [,3] [,4] [,5]
## [1,]    1    1    1    1    1
## [2,]    1   -1    1   -1    1

# first, unconstrained:
print(wrap.mp(rets))

##           rets1 rets2 rets3 rets4 rets5
## Intercept   3.7   3.8   2.8   3.9   2.9

# now with subspace constraint:
print(wrap.mp(rets, Jmat = Jmat))

##           rets1 rets2 rets3 rets4 rets5
## Intercept   5.5   5.1   5.5   5.1   5.5

# and print the portfolio too:
print(wrap.mp(rets, ret.what = "mp", Jmat = Jmat))

##           rets1 rets2 rets3 rets4 rets5
## Intercept   0.1  0.13   0.1  0.13   0.1

# now for the case where the real Markowitz
# portfolio is not just the market.
set.seed(as.integer(charToRaw("420f1dfd-b19b-4175-83b3-b96548264bf8")))
rets <- matrix(rnorm(n.day * n.stock, mean = 0), nrow = n.day)
mu.off <- t(matrix(seq(from = -0.15, to = 0.15, length.out = n.stock),
    nrow = n.stock, ncol = n.day))
rets <- rets + mu.off
print(wrap.mp(rets, Jmat = Jmat))

##           rets1 rets2 rets3 rets4 rets5
## Intercept -0.12 -0.67 -0.12 -0.67 -0.12

# and print the portfolio too:
print(wrap.mp(rets, ret.what = "mp", Jmat = Jmat))

##             rets1  rets2   rets3  rets4   rets5
## Intercept -0.0023 -0.015 -0.0023 -0.015 -0.0023
```

2.1.5 Portfolio hedging constraints

Now consider the *constrained* portfolio which has no covariance with the portfolio equal dollar long in each of the fake assets, *i.e.*, 'the market.' The Wald statistics are computed as follows:

```r
# first for the case where the real Markowitz
# Portfolio is actually equal dollar long in each
# stock.
set.seed(as.integer(charToRaw("0bda3ab6-53a7-4f5a-aa6a-0fe21edbaa20")))
rets <- matrix(rnorm(n.day * n.stock, mean = 0.1),
    nrow = n.day)
Gmat <- matrix(1, nrow = 1, ncol = n.stock)
print(wrap.mp(rets, Gmat = Gmat))

##           rets1 rets2 rets3 rets4 rets5
## Intercept  -0.7     1   1.4  0.79  -2.4

# and print the portfolio too:
print(wrap.mp(rets, ret.what = "mp", Gmat = Gmat))

##            rets1 rets2 rets3 rets4  rets5
## Intercept -0.021 0.031  0.04 0.024 -0.064

# and in the case where it is not.
set.seed(as.integer(charToRaw("420f1dfd-b19b-4175-83b3-b96548264bf8")))
rets <- matrix(rnorm(n.day * n.stock, mean = 0), nrow = n.day)
mu.off <- t(matrix(seq(from = -0.8, to = 0.8, length.out = n.stock),
    nrow = n.stock, ncol = n.day))
rets <- rets + mu.off
Gmat <- matrix(1, nrow = 1, ncol = n.stock)
print(wrap.mp(rets, Gmat = Gmat))

##           rets1 rets2 rets3 rets4 rets5
## Intercept   -15    -8 -0.96   7.1    15

# and print the portfolio too:
print(wrap.mp(rets, ret.what = "mp", Gmat = Gmat))

##           rets1 rets2  rets3 rets4 rets5
## Intercept  -0.8 -0.44 -0.049  0.36  0.85
```

The hedging code also computes the Rao-Giri statistic for portfolio spanning [13, 5, 6, 7], with a variance estimate of the same, as below. In this example, the true Markowitz portfolio is $0.1\mathbf{1}$. Thus when we hedge out $\mathbf{1}$ in the first example, the population value of the maximal Sharpe ratio is zero. In the second example, when we hedge out a random vector, the population maximal Sharpe ratio is non-zero.

```r
rao.giri <- function(rets, Gmat, ...) {
    set.coln(rets)
    asymv <- mp_vcov(as.matrix(rets), Gmat = Gmat,
        ...)
    stat <- asymv$mu[1]
    a.var <- asymv$Ohat[1, 1]
    return(list(stat = stat, a.var = a.var, wald = stat/sqrt(a.var)))
}
# here we hedge out G, the rows of which span the
# true Markowitz portfolio
set.seed(as.integer(charToRaw("4618fc2e-9c58-4ea7-81f7-6ed771ace500")))
```

```r
rets <- matrix(rnorm(n.day * n.stock, mean = 0.1),
    nrow = n.day)
Gmat <- matrix(1, nrow = 1, ncol = n.stock)
print(rao.giri(rets, Gmat)$wald)

## [1] 0.81

# here we hedge out a random G, the rows of which
# do not span the true Markowitz portfolio
set.seed(as.integer(charToRaw("4abaccd9-8cac-4149-b30d-f3b4d32b44df")))
rets <- matrix(rnorm(n.day * n.stock, mean = 0.1),
    nrow = n.day)
Gmat <- matrix(rnorm(n.stock), nrow = 1, ncol = n.stock)
print(rao.giri(rets, Gmat)$wald)

## [1] 3
```

2.2 Fama French three portfolios

I download the monthly Fama French 3-factor portfolio returns from **Quandl** [11], compute the returns excess the risk free rate, then call both procedures on the returns. I also use robust standard errors [14] via the **sandwich** package.

```r
ff.data <- read.csv(paste0("http://www.quandl.com/api/v1/datasets/",
    "KFRENCH/FACTORS_M.csv?&trim_start=1926-07-31&trim_end=2013-10-31",
    "&sort_order=asc"), colClasses = c(Month = "Date"))

rownames(ff.data) <- ff.data$Month
ff.data <- ff.data[, !(colnames(ff.data) %in% c("Month"))]
# will not matter, but convert pcts:
ff.data <- 0.01 * ff.data
```

```r
rfr <- ff.data[, "RF"]

ff.ret <- cbind(ff.data[, "Mkt.RF"], ff.data[, c("HML",
    "SMB")] - rep(rfr, 2))
colnames(ff.ret)[1] <- "MKT"
# try both procedures:
print(do.both(ff.ret))

##                       MKT  HML  SMB
## Britten Jones t-stat 4.1 0.25 -1.9
## via itheta_vcov      3.9 0.26 -1.9

# try with robust s.e.
if (require(sandwich)) {
    print(do.both(ff.ret, vcov.func = sandwich::vcovHAC))
}

##                       MKT  HML  SMB
## Britten Jones t-stat 4.1 0.25 -1.9
## via itheta_vcov      3.5 0.23 -1.8
```

The Wald statistics are slightly less optimistic than the Britten-Jones t-statistics for the long MKT and short SMB positions. This is amplified when the HAC estimator is used.

References

[1] Michael W Brandt. Portfolio choice problems. *Handbook of financial econometrics*, 1:269–336, 2009. URL http://shr.receptidocs.ru/docs/5/4748/conv_1/file1.pdf#page=298.

[2] Mark Britten-Jones. The sampling error in estimates of mean-variance efficient portfolio weights. *The Journal of Finance*, 54(2):655–671, 1999. URL http://www.jstor.org/stable/2697722.

[3] Vijay Kumar Chopra and William T. Ziemba. The effect of errors in means, variances, and covariances on optimal portfolio choice. *The Journal of Portfolio Management*, 19(2):6–11, 1993. URL http://faculty.fuqua.duke.edu/~charvey/Teaching/BA453_2006/Chopra_The_effect_of_1993.pdf.

[4] John Howland Cochrane. *Asset pricing*. Princeton Univ. Press, Princeton [u.a.], 2001. ISBN 0691074984. URL http://gso.gbv.de/DB=2.1/CMD?ACT=SRCHA&SRT=YOP&IKT=1016&TRM=ppn+322224764&sourceid=fbw_bibsonomy.

[5] Narayan C. Giri. On the likelihood ratio test of a normal multivariate testing problem. *The Annals of Mathematical Statistics*, 35(1):181–189, 1964. doi: 10.1214/aoms/1177703740. URL http://projecteuclid.org/euclid.aoms/1177703740.

[6] Gur Huberman and Shmuel Kandel. Mean-variance spanning. *The Journal of Finance*, 42(4):pp. 873–888, 1987. ISSN 00221082. URL http://www.jstor.org/stable/2328296.

[7] Raymond Kan and GuoFu Zhou. Tests of mean-variance spanning. *Annals of Economics and Finance*, 13(1), 2012. URL http://www.aeconf.net/Articles/May2012/aef130105.pdf.

[8] Harry Markowitz. Portfolio selection. *The Journal of Finance*, 7(1):pp. 77–91, 1952. ISSN 00221082. URL http://www.jstor.org/stable/2975974.

[9] Harry Markowitz. The early history of portfolio theory: 1600-1960. *Financial Analysts Journal*, pages 5–16, 1999. URL http://www.jstor.org/stable/10.2307/4480178.

[10] Harry Markowitz. Foundations of portfolio theory. *The Journal of Finance*, 46(2):469–477, 2012. URL http://onlinelibrary.wiley.com/doi/10.1111/j.1540-6261.1991.tb02669.x/abstract.

[11] Raymond McTaggart and Gergely Daroczi. *Quandl: Quandl Data Connection*, 2013. URL http://cran.r-project.org/package=Quandl. R package version 2.1.2.

[12] Steven E. Pav. Asymptotic distribution of the Markowitz portfolio. Privately Published, 2013. URL http://arxiv.org/abs/1312.0557.

[13] C. Radhakrishna Rao. *Advanced Statistical Methods in Biometric Research*. John Wiley and Sons, 1952. URL http://books.google.com/books?id=HvFLAAAAMAAJ.

[14] Achim Zeileis. Econometric computing with HC and HAC covariance matrix estimators. *Journal of Statistical Software*, 11(10):1–17, 11 2004. ISSN 1548-7660. URL http://www.jstatsoft.org/v11/i10.