

```

# I. Terveer: Formelsammlung MaSta für Wirtschaftswissenschaften
# 3. Auflage 2019, UVK
# R-Skripte zum Abschnitt "Tabellen von Stichprobenverteilungen"

# Die jeweils erzeugten Tabellen können z.B. mit dem xtable-Paket in LaTeX-
Tabellen konvertiert werden.

# 1. Verteilungsfunktion der Standardnormalverteilung

decs=seq(0,3.2,0.1)
mils=seq(0,0.095,0.005)

# Normal table
cdfnormal<-function (x,y){pnorm(x+y)}
cdfnormaltable=outer(decs,mils,cdfnormal)
rownames(cdfnormaltable)=decs
colnames(cdfnormaltable)=mils

# 2. Quantile von Standardnormal- und t(n)-Verteilung für n<=2000
alphalist=rev(1-c(0.0005,0.001,0.005,0.01,0.025,0.05,0.1))
nlist=c(Inf,seq(1,2000))
qtttable=outer(nlist,alphalist,function(n,alpha)
{round(qt(df=n,p=alpha),digits=2)})
# Dies ist die vollständige Tabelle, in der allerdings viele
aufeinanderfolgende Zeilen identisch sind.
# Von solchen Zeilenkolonnen wird jeweils nur die erste behalten
selectlist=c(TRUE,! (sapply(2:(length(nlist)),function(i)
{as.logical(min(qtttable[i,]==qtttable[i-1,]))}))) # die Zeilennummern, die
übrigbleiben.
shortqtttable=qtttable[selectlist,] # die verkürzte Quantiltabelle
rownames(shortqtttable)=nlist[selectlist]
colnames(shortqtttable)=alphalist

# 3. Quantile der chi^2(n)-Verteilung für n<=100
alphalistlinks=c(0.0005,0.001,0.0025,0.005,0.01,0.025,0.05,0.1)
alphalistrechts=rev(1-alphalistlinks)
alphalist=c(alphalistlinks,rev(1-alphalistlinks))
nlist=c(seq(1,80))
nlist=c(seq(81,100))
nlist
length(alphalist)
qchitable=outer(nlist,alphalist,function(n,alpha){qchisq(df=n,p=alpha)})
rownames(qchitable)=c(nlist)
colnames(qchitable)=alphalist

# 4. Quantile der F(m,n)-Verteilung

mlist=1:20 # für die Teiltabellen im Buch mlist=1:10 und mlist=11:20
nlist=1:500
alpha=0.9 # bzw. alpha=0.95 alpha=0.99
qftable=outer(nlist,mlist,function(n,m)
{round(qf(alpha,df1=m,df2=n),digits=2)})
# Diese Tabelle enthält wie die t-Quantiltabelle Kolonnen identischer
Zeilen, von denen die oberste jetzt behalten wird und die übrigen
gestrichen werden
selectlist=c(TRUE,! (sapply(2:(length(nlist)),function(i)
{as.logical(min(qftable[i,]==qftable[i-1,]))})))

```

```

shortqftable=qftable[selectlist,]
rownames(shortqftable)=nlist[selectlist]
colnames(shortqftable)=mlist

# 5. Quantile der Wilcoxon-w(n1,n2)-Verteilung
alpha=0.005 #bzw. alpha=0.01 bzw. alpha=0.025 bzw. alpha=0.05 bzw.
alpha=0.1
n1list=2:25
n2list=2:25
qwilcoxtable=outer(n1list,n2list,function(x,y){qwilcox(alpha,x,y)+x*(x+1)/
2})
# R verwendet eine translatierte Version der Wilcoxon-Verteilung
rownames(qwilcoxtable)=n1list
colnames(qwilcoxtable)=n2list

# 6. Quantile der Kolmogoroff-Verteilung, einfache Hypothese
ksf<-function(x,nsumm=100){if (x==0) 0 else
{1+2*sum((-1)^(1:nsumm)*exp((-2*x^2)*(1:nsumm)^2))}} # die Approximation
der Reihe in der VF ist mit n=100 zufriedenstellend
vksf<-Vectorize(ksf,vectorize.args = c("x"))
ksf(1.627)
curve(vksf,from=0,to=2)
ksf(0.5)
ksfa<-function(x,a,nsumm=100){ksf(x,nsumm)-a}
uniroot(ksfa,c(0,2),0.9,100)
alphalist=c(seq(0.1,0.9,0.1),0.95,0.99)
ksqlist=c()
for (i in (1:length(alphalist))) {
  ksqlist<-c(ksqlist,round(uniroot(ksfa,c(0,3),alphalist[i],100)
$root,digits=2))
}

ksqtable=t(as.matrix(ksqlist))
colnames(ksqtable)<-alphalist
xtable(ksqtable)

```