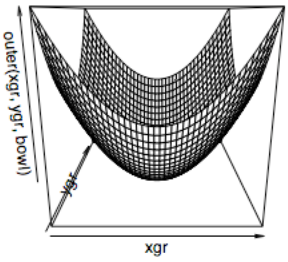
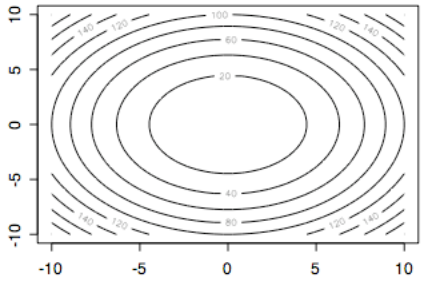
Plotting Gamma 2-parameter Confidence Regions

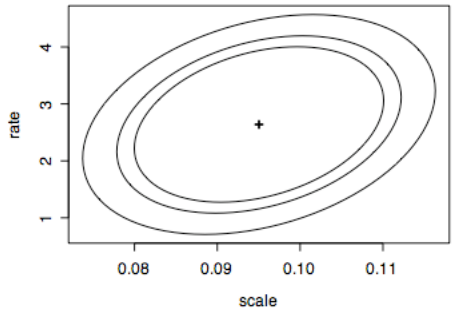
```
> bowl
function (x, y)
{
    x^2 + y^2
}

> xgr <- seq(-10,10,length=40)
> ygr <- seq(-10,10,length=40)
> persp(xgr, ygr, outer(xgr,ygr,bowl))
> contour(xgr, ygr, outer(xgr,ygr,bowl))
```



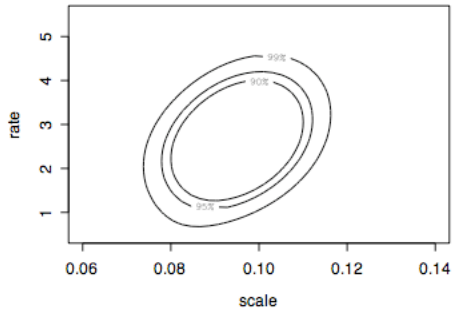


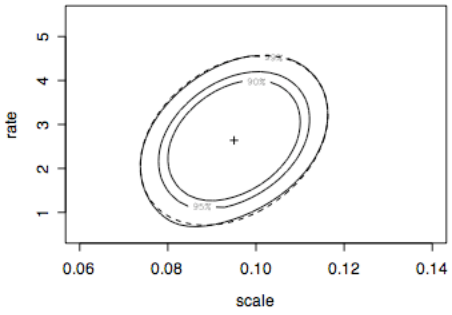

```
> ellipsem
function (mu, amat, c2, npoints = 100, showcentre = T, ...)
{
    if (all(dim(amat) == c(2, 2))) {
        eamat <- eigen(amat)
        hlen <- sqrt(c2/eamat$val)
        theta <- angle(eamat$vec[1, 1], eamat$vec[2, 1])
        ellipse(hlen[1], hlen[2], theta, mu[1], mu[2], npoints = npoints,
            ...)
        if (showcentre)
            points(mu[1], mu[2], pch = 3)
    }
    invisible()
}

> ellipse
function (hlaxa = 1, hlaxb = 1, theta = 0, xc = 0, yc = 0, newplot = F,
    npoints = 100, ...)
{
    a <- seq(0, 2 * pi, length = npoints + 1)
    x <- hlaxa * cos(a)
    y <- hlaxb * sin(a)
    alpha <- angle(x, y)
    rad <- sqrt(x^2 + y^2)
    xp <- rad * cos(alpha + theta) + xc
    yp <- rad * sin(alpha + theta) + yc
    if (newplot)
        plot(xp, yp, type = "l", ...)
    else lines(xp, yp, ...)
    invisible()
}

> fitxx <- gamma.mles1(xx, 1, 1)
> fitxx
$estimates
           [,1]
[1,] 0.09503273
```

```
[2,] 2.63908190

$infmat
            [,1]       [,2]
[1,] 22433.92392 -75.783931
[2,]   -75.78393   2.728962

> ellipsem(fitxx$est,fitxx$inf,qchisq(.99,2),newplot=T,xlab="scale",ylab="rate")
> ellipsem(fitxx$est,fitxx$inf,qchisq(.95,2))
> ellipsem(fitxx$est,fitxx$inf,qchisq(.90,2))
```




```
> gammaconfscore
function (xx, sgr, rgr, ...)
{
    contour(sgr, rgr, outer(sgr, rgr, scorepivot, n = length(xx),
        xbar = mean(xx), logxbar = mean(log(xx))), levels = qchisq(c(0.9,
        0.95, 0.99), 2), labels = c("90%", "95%", "99%"), xlab = "scale",
        ylab = "rate", ...)
}

> scorepivot
function (shape, rate, n, xbar, logxbar)
{
    stiinvs <- function(s1, s2, i11, i12, i22) {
        (s1 * s1 * i22 - 2 * s1 * s2 * i12 + s2 * s2 * i11)/(i11 *
            i22 - i12 * i12)
    }
    n * stiinvs(log(rate) - digamma(shape) + logxbar, shape/rate -
        xbar, trigamma(shape), -1/rate, shape/rate^2)
}

> sgr <- seq(.06,.14,len=40)
> rgr <- seq(.5,5.5,len=40)
> gammaconfscore(xx, sgr, rgr)
> ellipsem(fitxx$est,fitxx$inf,qchisq(.90,2),lty=2)
```





```
> gammaconfLLR
function (xx, sgr, rgr, ...)
{
    fitxxn <- gamma.mlen(xx, mean(sgr), mean(rgr))
    contour(sgr, rgr, outer(sgr, rgr, neg2LLR, n = length(xx),
        xbar = mean(xx), logxbar = mean(log(xx)), negLLmin = fitxxn$minimum),
        levels = qchisq(c(0.9, 0.95, 0.99), 2), labels = c("90%",
            "95%", "99%"), xlab = "scale", ylab = "rate", ...)
}

> neg2LLR
function (shape, rate, n, xbar, logxbar, negLLmin)
{
    2 * (-n * (shape * log(rate) - lgamma(shape) + (shape - 1) *
        logxbar - rate * xbar) - negLLmin)
}

> gammaconfLLR(xx, sgr, rgr)
> ellipsem(fitxx$est,fitxx$inf,qchisq(.99,2),lty=2)
```