Thanks again for your time and your notes. I will sumarize the story briefly and show what is bothering me.

Important result (btw. how do you write letter d above an arrow?):

$$\sqrt{n}\left(\hat{\theta}-\theta_0\right) \to N\left(0, I\left(\theta_0\right)^{-1}\right)$$

Asimptotically variance equals (is this OK, since I got quite lost around $I(\theta)$ and $I_1(\theta)$ in your PDF)

$$\begin{aligned}
Var\left(\sqrt{n}\left(\hat{\theta}-\theta\right)\right) &= I(\theta)^{-1} \\
Var\left(\left(\hat{\theta}-\theta\right)\right) &= Var\left(\hat{\theta}\right) = nI(\theta)^{-1} \\
&= n\left(nI_1(\theta)^{-1}\right) \\
&= I_1(\theta)^{-1}
\end{aligned}$$

I came to all this after exchange with a friend. We were looking in R, how standard errors are computed and this example shows this. Say we have 1000 iid values from gamma distribution with parameters shape a and scale θ

$$f(x) = \frac{1}{a^{\theta}\Gamma(a)} x^{(a-1)} e^{-\left(\frac{x}{\theta}\right)}.$$

Or in terms of parameter rate $\lambda = 1/\theta$

$$f(x) = \frac{\lambda^a}{\Gamma(a)} x^{(a-1)} e^{-\lambda x}.$$

```
## Simulate from Gamma(3, 1.8)
n <- 1000
y <- rgamma(n = n, shape = 3, rate = 1.8)
```

Now we can estimate parameters via method of moments since we know $E(X) = a/\lambda$ and $Var(X) = a/\lambda^2$ and therefore $\lambda = E(X)/Var(X)$ and $a = E(X)^2/Var(X)$.

```
## Method of moments
(rate <- mean(y) / var(y))
## [1] 1.6314
(shape <- mean(y)^2 / var(y))
## [1] 2.7711
```

What about maximum likelihood estimates? Loglikelihood is

$$l\left(a,\lambda|\mathbf{x}\right)=n\times a\times log\left(\lambda\right)-n\times log\left(\Gamma\left(a\right)\right)+\left(a-1\right)\sum_{i=1}^{n}log\left(x_i\right)-\lambda\sum_{i=1}^{n}x_i.$$

After taking partial derivatives and setting them to 0 we do not get closed form equations and numerical optimization must be used.

In R we can use fitdistr function, which helps in producing the call for optimization via optim function

```
## Maximum likelihood
library(MASS)
(tmp <- fitdistr(x = y, densfun = “gamma”))
##    shape       rate
## 2.841056   1.672556
##(0.120345) (0.077486)
```

Let us figure out how standard errors are computed. Using above result

$$\sqrt{n}\left(\left(\hat{a},\hat{\lambda}\right)-\left(a,\lambda\right)\right)\to N\left(\left(0,0\right),I\left(a,\lambda\right)^{-1}\right)$$

Fisher's expected information equals

$$I\left(a,\lambda\right)=\begin{pmatrix}-\frac{\Gamma''(a)\Gamma(a)-\left(\Gamma'(a)\right)^2}{\Gamma^2(a)} & -1/\lambda\\ -1/\lambda & a/\lambda^2\end{pmatrix}.$$

I can do this in R with the following code

```
## Take maximum likelihood estimates of parameters
a <- tmp$estimate[[1]]
lambda <- tmp$estimate[[2]]
## Compute Fishers’s expected information about a and λ
gammaD2 <- (gamma(a) * psigamma(a)^2) + (gamma(a) * psigamma(a, 1))
gammaD <- gamma(a) * psigamma(a)
I11 <- (gammaD2 * gamma(a) - (gammaD)^2) / gamma(a)^2
I12 <- -1 / lambda
I21 <- I12
I22 <- a / lambda^2
(I <- matrix(c(I11, I12, I21, I22), nrow = 2))
##         [,1]     [,2]
##[1,]  0.42103 -0.59789
##[2,] -0.59789  1.01559
```

```
## Taking inverse of I(a,λ) and square root of diagonal elements yields
invI <- solve(I)
sqrt(diag(invI))
## [1] 3.8056 2.4503
```

Hmm, this is not the same as standard errors from fitdistr call, but this is

```
sqrt(diag(invI)) / sqrt(n)
## [1] 0.120345 0.077486
```

How, does fitdistr come up with standard errors? It calls optim function and this function returns Hessian matrix. As far as I know this is a matrix of second derivatives and therefore equals to Fisher's observed information. For gamma example Hessian equals

```
tmp$hessian
##        shape    rate
##shape  421.03 -597.89
##rate  -597.89 1015.59
```

Comparing to $I(a, \lambda)$ Hessian seems to be $I(a, \lambda)\, n$. Square root of Hessian inverse and taking only diagonal elements yields

```
diag(sqrt(solve(tmp$hessian)))
##   shape     rate
##0.120345 0.077486
```

I am a bit lost here? Is this just R stuff or related to expected/observed Fisher's information?