Bernoulli: $Y \sim Bin(1, \pi)$, $\pi \in (0,1)$

$P_{Y_i}(y_i; \pi) = \pi^{y_i}(1-\pi)^{1-y_i}$ $\quad \left[\begin{array}{l} E(Y_i) = \pi \\ V(Y_i) = \pi(1-\pi) \end{array} \right.$

Binomial: $Y \sim Bin(n, \pi)$, $\pi \in (0,1)$

$P_{Y_i}(y_i; \pi) = \binom{n}{y_i} \pi^{y_i}(1-\pi)^{n-y_i}$ $\quad \left[\begin{array}{l} E(Y_i) = n\pi \\ V(Y_i) = n\pi(1-\pi) \end{array} \right.$

Normal distribution: $Y \sim N(\mu, \sigma^2)$, $\mu \in \mathbb{R}$, $\sigma \in \mathbb{R}^+$

$P_{Y_i}(y_i; \mu, \sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{y_i - \mu}{\sigma}\right)^2}$ $\quad \left[\begin{array}{l} E(Y) = \mu \\ V(Y) = \sigma^2 \end{array} \right.$

Poisson: $Y \sim Poisson(\lambda)$, $\lambda > 0$

$P_{Y_i}(y_i; \pi) = \frac{\lambda^y}{y!} e^{-\lambda}$, $y_i \geq 0$ $\quad \left[E(Y) = V(Y) = \lambda \right.$

Exponential distribution: (time)

$Y \sim Exp(\lambda)$, $\lambda > 0$

$P_Y(y; \lambda) = \lambda e^{-\lambda y}$ $\quad \left[\begin{array}{l} E(Y) = \frac{1}{\lambda} \\ V(Y) = \frac{1}{\lambda^2} \end{array} \right.$

Gamma: $Y \sim Gamma(\alpha, \lambda)$; $\alpha, \lambda > 0$

$P_Y(y; \alpha, \lambda) = \frac{\lambda^\alpha y^{\alpha-1}}{\Gamma(\alpha)} \alpha^{-\lambda y}$

---

likelihood: $\boxed{L(\theta) = P_Y(y, \theta)}$, y is fixed and θ varies

log-likelihood: $\boxed{\ell(\theta) = \log L(\theta)}$ for mathematical covenience

score function: $\boxed{\ell_*(\theta) = \frac{\partial \ell(\theta)}{\partial \theta}}$

$\hat{\theta}$ = **maximum likelihood estimate** (mle) if $L(\hat{\theta}) > L(\theta)$

↳ **likelihood equation:** $\boxed{\ell_*(\theta) = 0}$ $\Rightarrow$ find value $\hat{\theta}$

normalized log-likelihood = $\boxed{\ell(\theta) - \ell(\hat{\theta})}$

observed information of θ: $\boxed{j(\theta) = -\ell_{**}(\theta) = -\frac{\partial^2 \ell(\theta)}{\partial \theta^2}}$

parabolic approximation of normalized log-likelihood: $\boxed{\ell(\theta) - \ell(\hat{\theta}) = -\frac{1}{2}(\theta - \hat{\theta})^2 j(\hat{\theta})}$

approximation for distribution of the mle: $\boxed{\begin{array}{l} \hat{\theta}_n \\ \hat{\theta}_n - \theta \end{array} \ \dot\sim \ \begin{array}{l} N(\theta, j(\hat{\theta})^{-1}) \\ N_2\left(\begin{smallmatrix}0\\0\end{smallmatrix}\right) \end{array}}$

(for multivariates parameters)

---

alt code

{	123	~	126
}	125	@	64
[	91	*	42
]	93	<	60
		^	94

Newton Raphson steps

0) $\ell_*(\alpha, \beta) = 0 \iff \begin{cases} \frac{\partial \ell(\alpha,\beta)}{\partial \alpha} = 0 \\ \frac{\partial \ell(\alpha,\beta)}{\partial \beta} = 0 \end{cases}$

1) substitute the parameters from one eq. to the other

2) implement with:

parameters · obs · 1/

```
nr.muon <- function(x,alpha=0.6,eps=0.000001)
{
    n = length(x)
    diff = 1
    while(diff>eps)
    {
    alpha.old = alpha
    s = sum(x/(1+alpha*x))
    j = sum((x/(1+alpha*x))^2)
    alpha = alpha+s/j
    diff = abs(alpha-alpha.old)
    }
```

second derivative

3) alpha.hat.nr = nr.muon(x = x)
lambda.hat.nr = ...
round(c('alpha.hat' = alpha.hat.nr, 'lambda.hat' = lambda.hat.nr), 3)

Observed info-matrix

```
j <- function(y, γ, β){
j11 <-
j12 <-
j22 <-
out = matrix(c(j11, j12, j12, j22), ncol=2)
return(out)
}
```

Inversed matrix

```
solve(j.hat)
```

Plots: normalized likelihood / parabolic approx.

1)
```
loglik <- function(){
    out = ...
    return(out)
}
```
```
parApprox <- function(){
    out = ... (with j(μ̂) substituted)
    return(out)
}
```

2) `vloglik = Vectorize(loglik, "mu")`

3) to split screen : `par(mfrow=c(1,2))`

```
curve(vloglik(x, y^obs) - loglik(muhat, y^obs),   | vparApprox(x, y^obs)
      from = 5, to = 5.5, lwd = 2, col = 2
      ylab = '', xlab = 'mu', main = 'name'
```

(*) to put them togheter
```
curve(...)
curve(..., add=T)
```

Confidence interval (γ, α)

```
parabola <- function(theta, theta.hat, n){
  # theta = c(mu, sigma2)
  # theta.hat = c(mu.hat, sigma2.hat)
  sigma2.hat <- theta.hat[2]
  j <- matrix(c(n/sigma2.hat, 0, 0, n/(2*sigma2.hat^2)),2,2
  diff <- matrix(theta - theta.hat,2, 1)
  return(-0.5*t(diff)%*%j%*%diff)
}
```

0.01 3
```
m <-seq(3.2,3.8,length=100)
v <-seq(0.7,1.7,length=100)
```
0.01 3
```
p <- matrix(rep(NA, 100*100), 100,100)

theta.hat <- c(3.5,1.2)
n <- 100

for(i in 1:100){
for(j in 1:100){
p[i,j] <- parabola(c(m[i],v[j]), theta.hat, n) }
}
```

→ values from mle (for `theta.hat <- c(3.5,1.2)`)