Probability and Inference HW 3b

Yaniv Bronshtein(yb262), Vipul Ishwarchandra Gharde(vig4), Animesh Sharma(as3592),Wanying Mo(w

11/8/2021

Import the necessary libraries

```
library(purrr)
library(MASS)
```

Functions to calculate first and second derivatives for Newton Raphson method

```
f1 <- function(a,n,x){
  return(-n*digamma(a)- n*log(mean(x)) + n*log(a) + sum(log(x)))
}
f2 <- function(a,n){
  return(-n*trigamma(a) + n/a)
}
```

This function is used to calculate parameters for different distributions of data using mean and variance

```
get_dist_params <- function(dist, data) {

  mu <- mean(data); var <- var(data)
  # For the beta distribution, run Newton-Raphson 20 times
  if (dist=="beta") {
    x <- data; a2 <- 1; b2 <- 1
    for (i in 1:20){
    # Matrices to hold estimates
      estim <- matrix(c(a2,b2), nrow=2)
    # Compute the partials with respect to alpha and beta
      f<- matrix(c(digamma(a2+b2) - digamma(a2) + mean(log(x)),
                   digamma(a2+b2)- digamma(b2)+ mean(log(1-x))), nrow=2)

      # Compute partial derivatives by solving the Jacobian matrix
      J <- solve(matrix(c(trigamma(a2+b2)-trigamma(a2),
                          trigamma(a2+b2), trigamma(a2+b2),
                          trigamma(a2+b2)-trigamma(b2)), nrow=2,ncol=2))

      f2 <- estim - J%*%f
      a2 <- f2[1,]
      b2 <- f2[2,]
    }
    cat("Beta dist alpha:",a2,'\n')
    cat("Beta dist beta: ",b2,'\n')
  }
  else if (dist == "bernoulli") {
    cat("Bernoulli p:", mu,'\n')
```

```
  }
  else if (dist=="binomial") {
    cat("Binomial p: ", mu/length(data),'\n')
  }
  else if (dist == "exponential") {
    cat("Exponential rate value:", 1/mu,'\n')
  }
  # For gamma distribution, we run Newton-Raphson for 60 iterations
  else if (dist=="gamma") {
    x <- data; n <- length(data); a <- 1

    for(i in 1:60){
      a <- a - f1(a,n,x)/f2(a,n)
    }
    # Beta value
    b <- mean(x)/a
    cat("Gamma alpha: ",a,'\n')
    cat("Gamma beta: ",1/b,'\n')
  }
  else if (dist=="geometric") {
    p <- 1/mu
    cat("Geometric p: ", p,'\n')
  }
  else if (dist=="multinomial") {
    n_row <- nrow(data); prob <- c(0,0,0,0);
    p <- data/length(data)
    prob <- rowSums(p)
    cat('Multinomial prob: ',prob,'\n')
  }
  else if (dist=="multivariate_normal") {
    mu <- colMeans(data)
    x_sub_mean <- data - mu
    cov <- matrix(c(0,0,0,0),2,2)
    for (i in 1:10000) {
      prod <- x_sub_mean[i,] %*% t(x_sub_mean[i,])
      cov <- cov + prod
    }
    covariance <- round(cov / 10000, 3)
    cat(covariance,'\n')
    cat("Multivariate mean: ", mu," Multivariate Sigma: ", covariance,'\n')
  }
  else if (dist=="normal") {
    cat("Normal Mean: value: ", mu,'\n')
    cat("Normal Standard deviation: ", sqrt(var),'\n')
  }
  else if (dist=="poisson") {
    cat("Poisson Lambda: ", mu,'\n')
  }
  else if (dist=="uniform") {
    cat("Uniform a: ",min(data)," Uniform b: ", max(data),'\n')
  }
}
```

Bernoulli Distribution

```
bernoulli_data <- rbinom(1000, 1, 0.75)
dist <- 'Bernoulli'
get_dist_params(dist,bernoulli_data)
```

Beta Distribution

```
beta_data <- rbeta(10000, 2, 8)
dist <- 'beta'
get_dist_params(dist,beta_data)
```

```
## Beta dist alpha: 2.020741
## Beta dist beta:  8.03316
```

Binomial Distribution

```
binom_data <- rbinom(100, 1000, 0.75)
dist <- 'Binomial'
get_dist_params(dist,binom_data)
```

Exponential Distribution

```
exp_data <- rexp(100000, 5)
dist <- 'exponential'
get_dist_params(dist,exp_data)
```

```
## Exponential rate value: 4.996793
```

Gamma Distribution

```
gamma_data <- rgamma(10000, 2, 3)
dist <- 'gamma'
get_dist_params(dist,gamma_data)
```

```
## Gamma alpha:  2.03401
## Gamma beta:  3.080543
```

Geometric Distribution

```
geom_data <- rgeom(100000, 0.25)
dist <- 'geometric'
get_dist_params(dist,geom_data)
```

```
## Geometric p:  0.3352791
```

Multinomial Distribution

```
p = c(0.20,0.40,0.05,0.30)
data <- rmultinom(10000,size=4,p)
dist <- 'multinomial'
get_dist_params(dist, data)
```

```
## Multinomial prob:  0.207475 0.42585 0.05185 0.314825
```

Multivariate Normal Distribution

```
Sum <- matrix(c(9,6,6,16),2,2)
data <- mvrnorm(n = 10000, c(4, 5), Sum)
dist <- 'multivariate_normal'
get_dist_params(dist, data)
```

```
## 9.532 6.107 6.107 16.574
```

```
## Multivariate mean:  3.942471 4.9625  Multivariate Sigma:  9.532 6.107 6.107 16.574
```

Normal Distribution

```r
norm_data <- rnorm(100000, 20, 2)
dist <- 'normal'
get_dist_params(dist,norm_data)
```

```
## Normal Mean: value:  20.00024
## Normal Standard deviation:  2.006343
```

Poisson Distribution

```r
poisson_data <- rpois(40000, lambda = 3)
dist <- 'poisson'
get_dist_params(dist,poisson_data)
```

```
## Poisson Lambda:  3.001175
```

Uniform Distribution

```r
uniform_data <- runif(1000000, 1, 100)
dist <- 'uniform'
get_dist_params(dist,uniform_data)
```

```
## Uniform a:  1.000182  Uniform b:  99.99974
```

Maximum likelihood estimation: (R code separate).

Bernoulli's :

$(x_1, x_2 \ldots x_n) = X$ be the outcomes of n bernoulli's trails, each with probability p

$$L(p, x) = \prod_{i=1}^{n} f(x_i, p)$$

likelihood

$$= \prod_{i=1}^{n} p^{x_i}(1-p)^{1-x_i}$$

$$\log l = \sum_{i=1}^{n} \log (p^{x_i}(1-p))^{1-x_i}$$

$$= \sum_{i=1}^{n} x_i \log p + \left(n - \sum_{i=1}^{n} x_i\right) \log(1-p)$$

Differentiating and equating to 0.

$$\frac{1}{p}\left[\sum_{i=1}^{n} x_i\right] + \left(n - \sum_{i=1}^{n} x_i\right) \cdot \frac{-1}{1-p} = 0$$

Solving for p

$$p = \frac{\sum_{i=1}^{n} x_i}{n}$$ (which is mean)

Exponential distribution :

$$\text{pdf} \Rightarrow f(x) = \frac{1}{\theta} e^{-x/\theta}$$

$$L = \frac{1}{\theta^n} e^{\sum_{i=1}^{n}\left(-\frac{x_i}{\theta}\right)}$$

$$\log L = -n \log \theta - \frac{1}{\theta} \sum_{n=1}^{n} x_i$$

differentiating

$$= -\frac{n}{\theta} + \sum_{i=1}^{n}\left(-\frac{x_i}{\theta^2}\right)$$

$$\hat{\theta} = \sum_{i=1}^{n} \frac{x_i}{n} = \overline{X} \text{ (mean)}$$

Geometric distribution :

$$f(x) = (1-p)^{x-1} \cdot p$$

$$\text{likelihood } (L) = p^n (1-p)^{\sum_{i=1}^{n} x_i - n}$$

$$\log L = n \log p + \sum_{i=1}^{n} x_i - n) \log(1-p)$$

differentiating

$$= \frac{n}{p} - \left(\sum_{i=1}^{n} x_i - n\right) / (1-p$$

$$\hat{p} = \frac{n}{\sum_{i=1}^{n} x_i} = \frac{1}{\bar{x}}$$

Binomial dist:

pdf $f(x) = \frac{n!}{x!\,(n-x)!} p^x (1-p)^{n-x}$

$$L = \prod_{i=1}^{N} f(x) = \prod_{i=1}^{N} \frac{n!}{x!\,(n-x)!} p^{x_i} (1-p)^{n-x_i}$$

$$\log L = \sum_{i=1}^{N} \log(n!) - \sum_{i=1}^{n} \log(x_i!) - \sum_{i=1}^{n} \log(n-x)!$$

$$+ \sum_{i=1}^{N} x_i \cdot \log p + \left(n - \sum_{i=1}^{n} x_i\right) \log(1-p)$$

Differentiating (First 3 terms are 0)

$$= \frac{1}{p} \sum_{i=1}^{n} x_i - \frac{1}{1-p} \sum_{i=1}^{N} (n - x_i) = 0$$

$$\hat{p} = \frac{1}{N} \left(\frac{\sum_{i=1}^{N} x_i}{n} \right) = \frac{\text{mean}}{\text{len(data)}}$$

Poisson:

$$f(x) = \frac{\lambda^x e^{-\lambda}}{x!}$$

$$L(\lambda) = \prod_{i=1}^{n} \frac{\lambda^{x_i} e^{-\lambda}}{x!} = e^{-\lambda n} \left[\frac{\lambda^{\sum_{i=1}^{n} x_i}}{\prod_{i=1}^{n} (x_i)!} \right]$$

$$\log(L) = -\lambda n + \sum_{i=1}^{n} x_i \log(\lambda) - \log\left(\prod_{i=1}^{n} x_i\right)!$$

diff

$$= -n + \sum_{i=1}^{n} \frac{x_i}{\lambda}$$

$$\lambda = \frac{\sum_{i=1}^{n} x_i}{n} \Rightarrow \text{mean of the data.}$$

Uniform distribution:

$$\begin{cases} f(x_i) = 1/\theta & 0 < x_i < \theta \\ f(x) = 0 & \text{else} \end{cases}$$

$$\mathcal{L}(\theta) = \prod_{i=1}^{n} f(x_i) = \prod_{i=1}^{n} \left(\frac{1}{\theta}\right) = \theta^{-n}$$

differentiating

$$= -\frac{n}{\theta}$$

this is < 0 for all $\theta > 0$

$\mathcal{L}(\theta)$ is a decreasing function and is minimized.
at $\theta = X_n$

$\therefore$ MLE $\Rightarrow$ $\theta = X_n$ = max (data)

Normal distribution:

$$f(x) = (2\pi\sigma^2)^{-1/2} e^{\left(-\frac{1}{2}\frac{(x-\mu)^2}{\sigma^2}\right)}$$

$$\mathcal{L} : (2\pi\sigma^2)^{-n/2} e^{\left(\frac{-1}{2\sigma^2}\sum_{i=1}^{n}(x_i-\mu)^2\right)}$$

$$\log(\mathcal{L}) = -\frac{n}{2}\log(2\pi) - \frac{n}{2}\log(\sigma^2) - \frac{1}{2\sigma^2}\sum_{i=1}^{n}(x_i-\mu)^2 \longrightarrow (1)$$

Differentiating w.r.t μ

$$\Rightarrow \frac{1}{\sigma^2}\left(\sum_{j=1}^{n} x_j - \mu\right) = 0$$

$$\sum_{j=1}^{n} x_j - n\mu = 0 \qquad \hat{\mu} = \frac{\sum_{j=1}^{n} x_i}{n} \Rightarrow \text{mean}$$

(1)
differentiating w.r.t σ^2

$$= \frac{-n}{2\sigma^2}\left[\frac{-1}{2}\sum_{j=1}^{n}(x_j-\mu)^2\right] \times \frac{d}{d\sigma^2}\left[\frac{1}{\sigma^2}\right]$$

$$\Rightarrow \frac{-n}{2\sigma^2} + \left[\frac{1}{2}\sum_{j=1}^{n}(x_j-\mu)^2\right]\left(\frac{1}{\sigma^{2^2}}\right) = 0$$

if we rule out $\sigma^2 = 0$ we get

$$\sigma^2 = \sum_{j=1}^{n} \frac{(x_j - \mu)^2}{n} \Rightarrow \text{variance}$$

Multivariate normal distribution:

Joint probability distribution function is

$$f(x) = (2\pi)^{-k/2} / (\det(\gamma_0))^{-1/2} e^{-1/2 (x-\mu) \gamma_0^{-1} (x_j - \mu)}$$

$\mu \rightarrow$ mean $\quad \gamma_0 \rightarrow$ covariance matrix

$$L = (2\pi)^{nk/2} (\det(\gamma))^{-n/2} e^{-1/2 \sum_{i=1}^{n} (x_i - \mu)^T \gamma^{-1} (x_i - \mu)}$$

$$\log(L) = -\frac{n}{2} k \log(2\pi) - \frac{n}{2} \log(\det(\gamma)) - \frac{1}{2} \sum_{j=1}^{n} (x_j - \mu)^T \gamma^{-1} (x_j - \mu)$$

diff w.r.t mean vector

$$\Rightarrow \sum_{j=1}^{n} \gamma^{-1} (x_j - \mu)$$

$$= \frac{1}{\gamma} \sum_{j=1}^{n} x_j - n\mu = 0$$

$$\mu = \frac{1}{n} \sum_{i=1}^{n} x_i \Rightarrow \text{mean of the data.}$$

gradient of log likelihood w.r.t precision matrix

$$\Rightarrow \frac{n}{2} \frac{d}{d\gamma^{-1}} \ln(\det(\gamma^{-1})) - \frac{1}{2} \frac{d}{d\gamma^{-1}} \left(\sum_{i=1}^{n} \text{tr} (x_j - \mu)^T \gamma \times (x_i - \mu) \right)$$

$$= \frac{n}{2}\gamma^{T} - \frac{1}{2}\frac{d}{d\gamma^{1}}\left[\sum_{j=1}^{n} \mathrm{tr}\left[(x_j-\mu)(x_j-\mu)^{T}\gamma^{-1}\right]\right]$$

$$= \frac{n}{2}\gamma^{T} - \frac{1}{2}\left[\sum_{j=1}^{n}(x_j-\mu)(x_j-\mu)^{T}\right)^{T}$$

Transpose on both sides after setting it to 0

we get $\hat{\gamma} = \frac{1}{n}\left(\sum_{i=1}^{n}(x_i-\mu)(x_i-\mu)^{T}\right)$

Multinomial distribution:

$$\text{pdf} = n!\,\prod_{x}\frac{p_x^{n_x}}{n_x!}$$

For some fixed number of observations n, $\mathcal{L}(p) = f_p(n)$

& constraint $C(p) = 1$

$$C(p) = \sum_{x} p_x$$

To maximize $\mathcal{L}$, the gradient of $\mathcal{L}$ and gradient of C are collinear, that is there exists λ such that for every λ

$$\frac{d}{dp_x}\mathcal{L}(p) = \lambda \cdot \frac{\partial}{\partial p_x} C(p)$$

$$\frac{n_x}{p_x}\mathcal{L}(p) = \lambda$$

ie $p(x)$ should be proportional to n_x $\because \sum p_x = 1$, we finally get $\hat{p}_x = \frac{n_x}{n}$ for every x

$\Rightarrow$ In the code probability $n = 10000$ and the size is given & we estimate $\hat{p}$ for give probabilities

Gamma distribution:

Distribution $f(x) = \frac{1}{\beta^{\alpha}\Gamma(\alpha)} x^{\alpha-1} e^{-x/\beta}$, $x > 0$

$$\Pi(f_{x_i} \dots n) = \frac{\left(\prod_{i=1}^{n} x_i^{\alpha-1} e^{-\frac{\Sigma x_i}{\beta}}\right)}{\beta^{n\alpha}(\Gamma(\alpha))^n}$$

Taking log

$$= (\alpha-1)\log\left(\prod_{i=1}^{n} x_i\right) - \frac{1}{\beta}\Sigma x_i - n\alpha\log\beta - n\log(\Gamma(\alpha))$$

$$= (\alpha-1)\sum_{i=1}^{n}\log x - \frac{\Sigma x_i}{\beta} - n\alpha\log\beta - n\log(\Gamma(\alpha))$$

Taking derivative w.r.t β

$$= \frac{\Sigma x_i}{\beta^2} - \frac{n\alpha}{\beta} = 0 \qquad \Rightarrow n\alpha = \frac{\Sigma x_i}{\beta}$$

$$D = \bar{x}$$

$\frac{\quad}{\alpha}$

Using $\beta = \frac{\bar{x}}{\alpha}$ and putting in original function, we get

$f(0)$ in terms of α

Taking derivative w.r.t α

$$\frac{d}{d\alpha} \Rightarrow \sum_{i=1}^{n} \log(x_i) - n \log \beta \cdot n \frac{d}{d\alpha} \log(\Gamma(\alpha))$$

$\downarrow$ digamma

Using $\beta = \frac{\bar{x}}{\alpha}$

$$\Rightarrow \sum_{i=1}^{n} \log x_i - n \log\left(\frac{\bar{x}}{\alpha}\right) - n \,\text{digamma}(\alpha)$$

$$\Rightarrow n \,\text{digamma}(\alpha) - n \log(\bar{x}) + n \log(\alpha) + \sum_{i=1}^{n} \log(x_i)$$

Equate this to 0

so second derivative function, lets call this is $f_1(x)$

is $f_1(x) = -n \,\text{digamma}(\alpha) - n\log(\bar{x}) + n\log(\alpha) + \sum_{i=1}^{n} \log x_i$

Taking second derivative wrt α

$$f_2(x) : -n \,\text{trigamma}(\alpha) + \frac{n}{\alpha} = 0$$

We use these two functions in Newton-Raphson method, until we find a value of α

Beta distribution:

$$f(x) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\,\Gamma(\beta)} x^{\alpha-1} (1-\alpha)^{\beta-1} \quad \alpha,\beta > 0$$

MLE

$$\Pi\,(f(x)) = \prod_{i=1}^{n} \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\;\Gamma(\beta)} x_i^{\alpha-1} (1-x_i)^{\beta-1} \cdots$$

Taking log

$$= n\log(\Gamma(\alpha+\beta)) - n\log(\Gamma(\alpha)) - n\log(\Gamma(\alpha)) + (\alpha-1)\sum_{i=1}^{n}\log(x_i) + (\beta-1)\sum_{i=1}^{n}\log(1-x_i)$$

Partial wrt α

$$\frac{\partial l}{\partial \alpha} = \frac{n\,\Gamma'(\alpha+\beta)}{\Gamma(\alpha+\beta)} - n\frac{\Gamma'(\alpha)}{\Gamma(\alpha)} + \sum_{i=1}^{n}\ln(x_i)$$

w.r.t. β

$$\frac{\partial l}{\partial \beta} = \frac{n\,\Gamma'(\alpha+\beta)}{\Gamma(\alpha+\beta)} - \frac{n\,\Gamma'(\beta)}{\Gamma(\beta)} + \sum_{i=1}^{n}\log(1-x_i) = 0$$

dividing by n

$$\frac{\partial l}{\partial \alpha} \Rightarrow \text{digamma}(\alpha+\beta) - \text{digamma}(\alpha) + \frac{1}{n}\sum_{i=1}^{n}\log(x_i)$$

$$\frac{\partial l}{\partial \beta} \Rightarrow \text{digamma}(\alpha+\beta) - \text{digamma}(\beta) + \frac{1}{n}\sum_{i=1}^{n} \log(1-x_i)$$

To solve this, we need Jacobian matrix

$$J = \begin{vmatrix} \psi'(\alpha+\beta) - \psi'(\alpha) & \psi'(\alpha+\beta) \\ \psi'(\alpha+\beta) & \psi'(\alpha+\beta) - \psi'(\beta) \end{vmatrix}$$

Solving this with a certain initial values gives us values of α, β.