Ejercicio 5 - Estadística Computacional

Jaime Riquelme

2023-11-24

Método de Máxima Verosimilitud

Ejercicio 1

Paso 0

Definimos los datos a utilizar:

```r
# Establecimiento de una semilla para garantizar la reproducibilidad
set.seed(10)

# Definición del tamaño de la muestra
tamano_muestra = 5000

# Asignación del valor del parámetro lambda para la distribución de Poisson
valor_lambda = 6

# Generación de una muestra de datos siguiendo una distribución de Poisson
datos_poisson = rpois(n = tamano_muestra, lambda = valor_lambda)
```

Paso 1: Distribución de Poisson

La función de masa de probabilidad para una distribución de Poisson se define como $f(X, \lambda) = \frac{e^{-\lambda}\lambda^X}{X!}$. Para varias variables, la función conjunta es el producto de estas funciones individuales.

Paso 2: Aplicación del Logaritmo Natural

Aplicando el logaritmo natural a la función conjunta, se simplifica a $\ln L(X_1, X_2, ..., X_n; \lambda) = -n\lambda + (\sum_{i=1}^{n} X_i)\ln(\lambda) - \sum_{i=1}^{n}\ln(X_i!)$.

Paso 3: Derivación para Estimación de Máxima Verosimilitud

Derivando esta función respecto a λ y estableciendo la derivada igual a cero, se busca el estimador de máxima verosimilitud.

Paso 4: Maximización de la Log-Verosimilitud

El estimador de máxima verosimilitud para λ se obtiene como el promedio de las observaciones $\lambda = \frac{\sum_{i=1}^{n} X_i}{n}$.

Primero creamos la funcion log:

```
neg_log_verosimilitud = function(param_lambda=valor_lambda, datos_observados=datos_poisson, tam_muestra=
  suma_datos = sum(datos_observados)
  # Cálculo de la log-verosimilitud negativa para la distribución de Poisson
  log_verosimilitud = -tam_muestra * param_lambda + suma_datos * log(param_lambda) - sum(log(factorial(d
  return(-log_verosimilitud) # Inversión del signo ya que 'optim' realiza minimización
}
```

Definimos las variables a utilizar para calcular el maximo verisimilud:

fn: La función que se va a optimizar, en este caso, neg__log__verosimilitud.

par: La estimación inicial del parámetro, en este caso, c(1). Este valor representa el punto de partida para el parámetro param__lambda.

lower: El límite inferior para los parámetros. Para param__lambda, el límite inferior es 0, ya que no puede ser negativo.

upper: El límite superior para los parámetros. Para param__lambda, se utiliza Inf, indicando que no hay un límite superior específico.

hessian: Indica si se debe calcular el hessiano o no. Se establece en TRUE para obtener información sobre la varianza de la estimación.

method: El método de optimización utilizado. En este caso, se emplea "L-BFGS-B", un método adecuado para optimización con límites en los parámetros.

datos__observados: Los datos observados, en este caso, datos__poisson. Esta es la muestra de datos generada por una distribución de Poisson.

```
#Realizamos el cálculo.

resultado_optimizacion = optim(
  par = c(1), # Valor inicial para param_lambda
  fn = neg_log_verosimilitud, # Función a minimizar
  lower = 0, # Límite inferior para param_lambda (debe ser no negativo)
  upper = Inf, # Límite superior para param_lambda
  hessian = TRUE, # Cálculo del hessiano para análisis de varianza
  method = "L-BFGS-B", # Método de optimización utilizado
  datos_observados = datos_poisson, # Datos de la muestra observados
  tam_muestra = tamano_muestra # Tamaño de la muestra
)

# Estimación del valor máximo verosímil para param_lambda
valor_estimado_lambda = resultado_optimizacion$par
cat("El valor original de param_lambda es", valor_lambda, "y el estimado por máxima verosimilitud es", v
```

tam_muestra: El tamaño de la muestra, representado por la variable tamano_muestra.

```
## El valor original de param_lambda es 6 y el estimado por máxima verosimilitud es 5.9994
```

Método de Máxima Verosimilitud

Ejercicio 2

Paso 0

Definimos los datos a utilizar:

```r
# Establecimiento de una semilla para garantizar la consistencia de los resultados
set.seed(10)

# Definición de parámetros para la generación de datos exponenciales
lambda = 10
Muestra = 5000

# Creación de una serie de datos con distribución exponencial
datos = rexp(Muestra, rate = lambda)
```

Paso 1: Verosimilitud Conjunta

La función de verosimilitud conjunta $L(\mathbf{X}, \alpha)$ es $\prod_{i=1}^{n} \alpha e^{-\alpha X_i}$, donde $\mathbf{X}$ es el vector de observaciones.

Paso 2: Log-Verosimilitud

El logaritmo de la función de verosimilitud se simplifica a $\log L(\mathbf{X}, \alpha) = n \log(\alpha) - \alpha \sum_{i=1}^{n} X_i$.

Paso 3: Derivación

La derivada de la función log-verosimilitud con respecto a α es $\frac{d\ell(\alpha)}{d\alpha} = \frac{n}{\alpha} - \sum_{i=1}^{n} X_i$.

Paso 4: Maximización

Igualando la derivada a cero, se encuentra que $\alpha = \frac{n}{\sum_{i=1}^{n} X_i}$, el estimador de máxima verosimilitud para α.

A continuación, encontramos el valor que maximiza la verosimilitud:

fn: Especificamos la función a optimizar, en nuestro caso, log_verosimilitud_exponencial.

par: Señalamos una estimación inicial para el parámetro, aquí utilizamos c(1).

lower: Determinamos el límite inferior para los parámetros.

upper: Establecemos el límite superior para los parámetros.

hessian: Indicamos si se debe calcular el hessiano durante el proceso.

method: Seleccionamos el método de optimización utilizado, que es "L-BFGS-B".

n: Definimos el tamaño total de la muestra.

```
# Función log-verosimilitud para la distribución exponencial
log_verosimilitud_exponencial <- function(alpha = lambda, x = datos, n = Muestra) {
  log_verosim = (n * log(alpha)) - (alpha * sum(x))
  return(-log_verosim) # Se usa el signo negativo porque optim() minimiza por defecto
}

# Uso de optim para encontrar el estimador de máxima verosimilitud de alpha
resultado_optim <- optim(fn = log_verosimilitud_exponencial,
                         par = c(1),
                         lower = c(-Inf),
                         upper = c(Inf),
                         hessian = TRUE,
                         method = "L-BFGS-B",
                         n = Muestra,
                         x = datos
)

# Resultados del estimador de máxima verosimilitud
valor_estimado = resultado_optim$par
cat("El valor original de lambda es", lambda, "y el estimador de máxima verosimilitud es", valor_estima
```

x: Representamos los datos de entrada, que en este caso son los datos.

```
## El valor original de lambda es 10 y el estimador de máxima verosimilitud es 9.946715
```