

# LAB 12

## Question 1

```
# Data
set.seed(123)
data <- rnorm(100, mean = 5, sd = 2)

# Log-likelihood
loglik <- function(params, x) {
  mu <- params[1]
  sigma2 <- params[2]
  n <- length(x)
  ll <- -0.5 * n * log(2 * pi * sigma2) - sum((x - mu)^2) / (2 * sigma2)
  return(ll)
}

# Gradient ascent algorithm
gradient_ascent <- function(x, init, lr = 0.001, tol = 1e-6, max_iter = 1000) {
  params <- init
  for (i in 1:max_iter) {
    mu <- params[1]
    sigma2 <- params[2]
    n <- length(x)
    grad_mu <- sum(x - mu) / sigma2
    grad_sigma2 <- -0.5 * n / sigma2 + sum((x - mu)^2) / (2 * sigma2^2)
    grads <- c(grad_mu, grad_sigma2)
    params_new <- params + lr * grads
    if (sqrt(sum((params_new - params)^2)) < tol) break
    params <- params_new
  }
  return(params)
}
```

```
# Initial guess
init <- c(mean(data), var(data))
# Run gradient ascent
mle_params <- gradient_ascent(data, init)
cat("Gradient Ascent MLE: mu =", mle_params[1], ", variance =", mle_params[2], "\n")
```

Gradient Ascent MLE: mu = 5.180812 , variance = 3.299943

```
# Compare with analytical MLE
cat("Analytical MLE: mu =", mean(data), ", variance =", var(data), "\n")
```

Analytical MLE: mu = 5.180812 , variance = 3.332931

## Question 2

```
# Data
income <- c(45000, 40000, 60000, 50000, 55000, 50000, 35000, 65000, 53000, 48000,
            37000, 31000, 40000, 75000, 43000, 49000, 37500, 71000, 34000, 27000)
age <- c(2, 4, 3, 2, 2, 5, 7, 2, 2, 1,
        5, 7, 4, 2, 9, 2, 4, 1, 5, 6)
y <- c(0, 0, 1, 1, 0, 1, 1, 1, 0, 0,
       1, 1, 1, 0, 1, 0, 1, 0, 0, 0)

# Standardize predictors for better convergence
x1 <- scale(income)
x2 <- scale(age)

X <- cbind(1, x1, x2)

# Sigmoid function
sigmoid <- function(z) 1 / (1 + exp(-z))

# Log-likelihood
loglik_logreg <- function(beta, X, y) {
  p <- sigmoid(X %*% beta)
  sum(y * log(p) + (1 - y) * log(1 - p))
}
```

```
# Gradient ascent for logistic regression
gradient_ascent_logreg <- function(X, y, lr = 0.1, tol = 1e-6, max_iter = 10000) {
  beta <- rep(0, ncol(X))
  for (i in 1:max_iter) {
    p <- sigmoid(X %*% beta)
    grad <- t(X) %*% (y - p)
    beta_new <- beta + lr * grad
    if (sqrt(sum((beta_new - beta)^2)) < tol) break
    beta <- beta_new
  }
  return(beta)
}

# Run gradient ascent
beta_mle <- gradient_ascent_logreg(X, y)
cat("Gradient Ascent MLE coefficients(Intercept, Income, Age):\n")
```

Gradient Ascent MLE coefficients(Intercept, Income, Age):

```
print(beta_mle)
```

```
      [,1]
[1,] 0.1471983
[2,] 0.9656761
[3,] 2.2176669
```

```
# Compare with glm
glm_fit <- glm(y ~ x1 + x2, family = binomial)
cat("GLM coefficients:\n")
```

GLM coefficients:

```
print(coef(glm_fit))
```

```
(Intercept)      x1      x2
  0.1472001  0.9656863  2.2176829
```