Problem Set 1

Onno Steenweg

Due: February 11, 2026

Question 1

The Kolmogorov-Smirnov test uses cumulative distribution statistics test the similarity of the empirical distribution of some observed data and a specified PDF, and serves as a goodness of fit test. The test statistic is created by:

$$D = \max_{i=1:n} \left\{ \frac{i}{n} - F_{(i)}, F_{(i)} - \frac{i-1}{n} \right\}$$

where F is the theoretical cumulative distribution of the distribution being tested and $F_{(i)}$ is the ith ordered value. Intuitively, the statistic takes the largest absolute difference between the two distribution functions across all x values. Large values indicate dissimilarity and the rejection of the hypothesis that the empirical distribution matches the queried theoretical distribution. The p-value is calculated from the Kolmogorov- Smirnoff CDF:

$$p(D \leq d) = \frac{\sqrt{2\pi}}{d} \sum_{k=1}^{\infty} e^{-(2k-1)^2\pi^2/(8d^2)}$$

which generally requires approximation methods (see Marsaglia, Tsang, and Wang 2003). This so-called non-parametric test (this label comes from the fact that the distribution of the test statistic does not depend on the distribution of the data being tested) performs poorly in small samples, but works well in a simulation environment. Write an `R` function that implements this test where the reference distribution is normal. Using `R` generate 1,000 Cauchy random variables (`rcauchy(1000, location = 0, scale = 1)`) and perform the test (remember, use the same seed, something like `set.seed(123)`, whenever you're generating your own data).

As a hint, you can create the empirical distribution and theoretical CDF using this code:

```r
# create empirical distribution of observed data
ECDF <- ecdf(data)
empiricalCDF <- ECDF(data)
# generate test statistic
D <- max(abs(empiricalCDF - pnorm(data)))
```

Answers Question 1

The task in Question 1 asks to manually implement a one-sample Kolmogorov-Smirnov goodness-of-fit test in R for a normal reference distribution, rather than relying on a built-in function. Specifically, it is asks to write an R function that constructs the empirical cumulative distribution function from observed data, compares it to the theoretical normal CDF, and computes the KS test statistic as the maximum absolute difference between the two. After this, the task asks to use this function in a simulation setting by generating 1,000 observations from a Cauchy distribution with a fixed random seed and applying the KS implementation to this data to assess whether it is consistent with normality, demonstrating the behavior of the test when the null hypothesis is false.

To solve this task, the R code below first sets a fixed random seed to ensure that the simulated results are reproducible. It then generates 1,000 random observations from a Cauchy distribution, which serves as the observed data to be tested. Next, a custom function is defined to implement the one-sample Kolmogorov–Smirnov test for a normal reference distribution: within this function, the data are sorted, the empirical cumulative distribution function is constructed from the ordered observations, and the theoretical normal cumulative distribution function is evaluated at the same data points. The code then computes the Kolmogorov–Smirnov test statistic as the maximum absolute difference between the empirical and theoretical CDFs. Finally, this function is applied to the simulated Cauchy data to obtain the KS test statistic, which is expected to be large because the data do not follow a normal distribution.

```r
set.seed(123)
data <- rcauchy(1000, location = 0, scale = 1)

ks_normal_test <- function(data) {
    data_sorted <- sort(data)
    n <- length(data_sorted)
    ECDF <- ecdf(data_sorted)
    empiricalCDF <- ECDF(data_sorted)
    theoreticalCDF <- pnorm(data_sorted, mean = 0, sd = 1)
    D <- max(abs(empiricalCDF - theoreticalCDF))
    return(D)}

D_value <- ks_normal_test(data)
D_value
```

The result of applying the custom Kolmogorov–Smirnov test to the simulated Cauchy dataset produced a test statistic of D=0.1347. This value represents the maximum absolute difference between the empirical cumulative distribution function of the Cauchy data and the theoretical CDF of a standard normal distribution. Since the Cauchy distribution is heavy-tailed and fundamentally different from a normal distribution, a relatively large KS statistic is expected. This indicates that the empirical data are not consistent with normality, and if

we were to calculate the corresponding p-value, it would be small, leading to rejection of the null hypothesis that the data come from a normal distribution. The result demonstrates that the KS test successfully detects deviations from the assumed reference distribution, even in a simulation setting.

As an extra, I calculated the p-value for the KS test. The R code shown below first computed the KS statistic by comparing the empirical cumulative distribution function of the Cauchy data to the theoretical normal CDF, then applied the asymptotic formula to approximate the p-value. The resulting p-value of approximately 2.58×10^{-16} is extremely small, providing strong evidence against the null hypothesis and confirming that the Cauchy data are not consistent with a normal distribution.

```r
    ks_pvalue <- function(D, n, tol = 1e-6) {
        lambda <- (sqrt(n) + 0.12 + 0.11 / sqrt(n)) * D
        k <- 1
        sum <- 0
        term <- 1
        while(term > tol) {
            term <- 2 * (-1)^(k-1) * exp(-2 * k^2 * lambda^2)
            sum <- sum + term
            k <- k + 1}
        return(sum)}

    p_value <- ks_pvalue(D_value, length(data))
    p_value
```

Question 2

Estimate an OLS regression in `R` that uses the Newton-Raphson algorithm (specifically `BFGS`, which is a quasi-Newton method), and show that you get the equivalent results to using `lm`. Use the code below to create your data.

Answers Question 2

The task asks to estimate an ordinary least squares (OLS) regression in R using a maximum likelihood approach with the Newton-Raphson algorithm, specifically the BFGS quasi-Newton method, rather than relying on the built-in lm() function. It requires writing code that maximizes the log-likelihood of a linear regression model with normally distributed errors to obtain estimates of the intercept, slope, and residual standard deviation. Finally, the task asks to demonstrate that these estimates are equivalent to those obtained from lm(), showing that the Newton-Raphson optimization reproduces standard OLS results.

To do so, the R code shown below first generates a dataset with 200 observations, where the independent variable x is drawn from a uniform distribution and the dependent variable

y is constructed using a linear model with slope 2.75 and normally distributed noise. It then computes standard OLS estimates using the standard lm() for comparison. Next, a negative log-likelihood function for linear regression is defined, with a log-transformation of the residual standard deviation to ensure positivity during optimization. The design matrix including the intercept is created, and optim() is used with the BFGS method to maximize the log-likelihood. Finally, the estimated coefficients and residual standard deviation are extracted from the optimized parameters, which are found to match the results from lm().

```r
set.seed(123)
data <- data.frame(x = runif(200, 1, 10))
data$y <- 0 + 2.75 * data$x + rnorm(200, 0, 1.5)

ols_lm <- lm(y ~ x, data = data)
coef(ols_lm)

neg_loglik <- function(params, y, X) {
    beta <- params[1:ncol(X)]
    log_sigma <- params[ncol(X) + 1]
    sigma <- exp(log_sigma)
    n <- length(y)
    residuals <- y - X %*% beta
    -(-n/2*log(2*pi*sigma^2) - sum(residuals^2)/(2*sigma^2))}

X <- cbind(1, data$x)
y <- data$y

start <- c(0, 0, log(1.5))

fit <- optim(start, neg_loglik, y=y, X=X, method="BFGS", hessian=TRUE)

beta_hat <- fit$par[1:2]
sigma_hat <- exp(fit$par[3])

beta_hat
sigma_hat
```