HOMEWORK 2

Group 12

Statistical Analysis of Delhi Temperature Data and Iris Dataset

Introduction

This report presents a statistical analysis of daily maximum temperatures in Delhi and the petal length of Iris flowers. We'll use custom implementations of the Kolmogorov-Smirnov test to examine the normality of the data and the distribution of sample statistics.

Understanding the Kolmogorov-Smirnov (K-S) Test

The Kolmogorov-Smirnov (K-S) test is a powerful method for checking if our data follows a specific distribution, such as the normal distribution. It works by comparing the behavior of our actual data (represented by its empirical cumulative distribution function, ECDF) with how we expect it to behave under the assumed distribution (represented by the theoretical cumulative distribution function, CDF).

How the K-S Test Works

1. We create the ECDF, which tracks how many values are below a given number in our dataset.
2. We compare this ECDF to the perfect theoretical model (e.g., the normal distribution's CDF).
3. The K-S statistic is the largest difference between these two functions.

If this difference is too large, it suggests our data probably doesn't follow the assumed distribution. The p-value from the test tells us whether this difference is small enough to accept the null hypothesis (meaning the data follows the distribution) or if we should reject it.

```
# This function takes a sample and the hypothesized distribution(F_0) as
# its arguments and returns the value of the Kolmogorv-Smirnov test statistic
ks.stat <- function(data, F_0 = pnorm)
{
  ecdf.fn <- ecdf(data)   # this function calculates empirical cdf
  rtn <- max(abs(ecdf.fn(data) - F_0(data)))    # K-S test statistic
  return(rtn)
}

# This function takes a sample and the hypothesized distribution(F_0) as
# its arguments and returns the p-value of Kolmogorov-Smirnov test
ks.test.12 <- function(data, F_0 = pnorm)
{
  obsd.ks.stat <- ks.stat(data, F_0)
  m <- 1000    # number of resamples
```

```
  sum <- 0
  for(i in 1:m)
  {
    resample <- sample(data, length(data), replace = TRUE)
    if(ks.stat(resample, F_0) > obsd.ks.stat)
    {
      sum <- sum + 1
    }
  }

  p.val <- sum/m
  return(p.val)
}
```

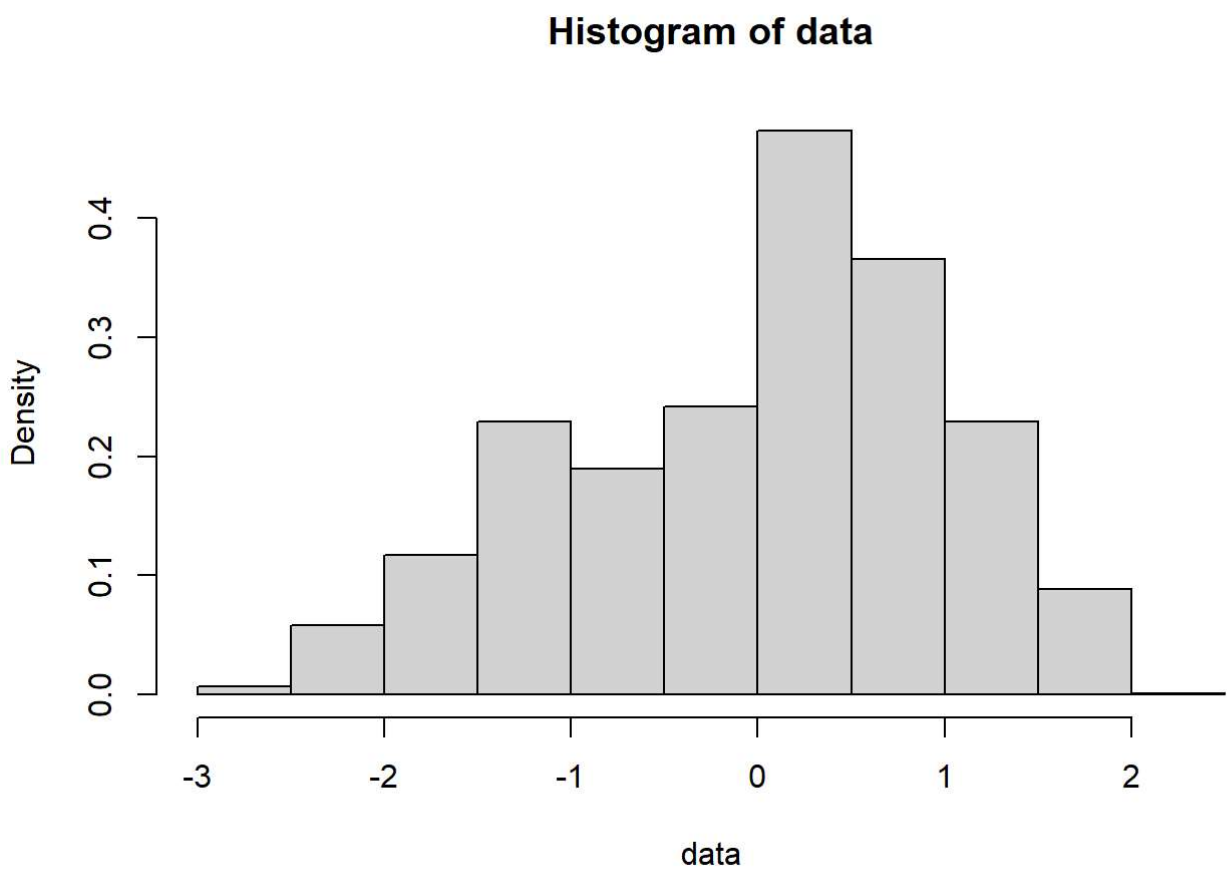
Part 1: Analysis of Delhi Temperature Data

normalizing the data

data <- (data - mean(data))/sd(data)

This code snippet loads the Delhi temperature data, converts the date column to Date format, and extracts the maximum temperature data for the period from January 1, 2015, to December 31, 2019. The data is then normalized by subtracting the mean and dividing by the standard deviation.

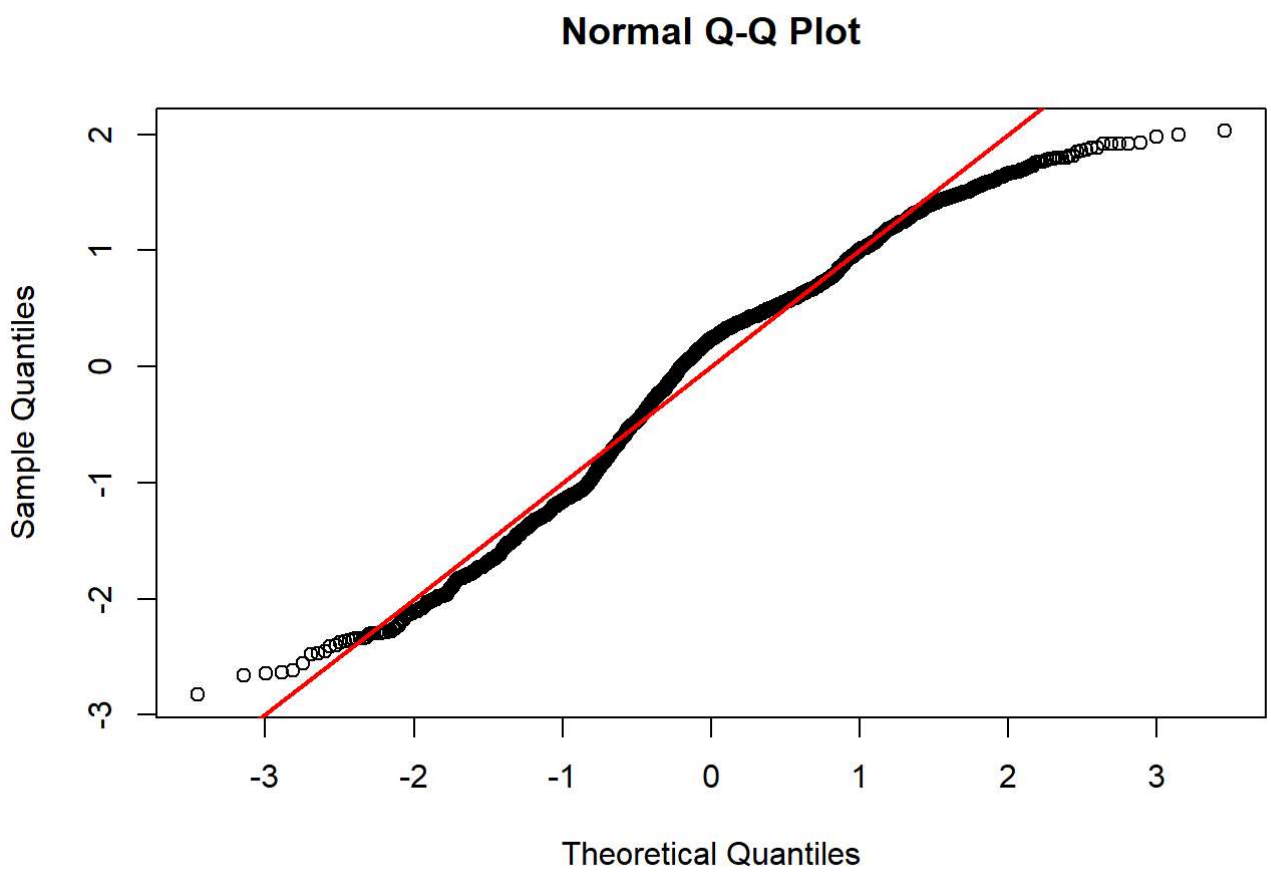
```
delhi_temp <- read.csv("delhi_temp.csv")
delhi_temp$Date <- as.Date(delhi_temp$Date)
# accessing daily maximum temperature from 01.01.2015 to 31.12.2019
indx1 <- which(delhi_temp$Date == "2015-01-01")
indx2 <- which(delhi_temp$Date == "2019-12-31")
data <- delhi_temp$Temp.Max[indx1:indx2]

# normalizing the data
data <- (data - mean(data))/sd(data)
hist(data, prob = TRUE)
```




```
#It seems that the data is slightly -vely skeweed
```

The histogram suggests that the data might be slightly negatively skewed. To further investigate the normality of the data, we create a Q-Q plot:

```
qqnorm(data)
lines(-5:5, -5:5, col = "red", lwd = 2)
```




```
#Here also it seems that the data is not normal
```

The Q-Q plot also indicates that the data may not be perfectly normal, as there are deviations from the red line (which represents perfect normality) at the tails.

To quantitatively assess the normality of the data, we use a custom implementation of the Kolmogorov-Smirnov test:

```
ks.test.12(data, F_0 = rnorm)
```

```
[1] 0.523
```

The p-value from this test is not very small, which suggests that the data does not provide strong evidence against the null hypothesis of normality. However, it's important to note that failing to reject the null hypothesis is not the same as proving that the data is normally distributed.

Part 2: Analysis of Sample Statistics

In this section, we analyze the distribution of sample statistics (mean, median, and mode) using the Iris dataset.

First, we load the Iris dataset and prepare it for analysis:

```
data(iris)
data <- iris$Petal.Length
n <- length(data)
```

We then define a function to calculate the sample mode:

```
sample.mode <- function(x) {
  freq.table <- table(x)  # Frequency table
  mode.val <- as.numeric(names(freq.table)[freq.table == max(freq.table)])  # Extract mode
  return(mode.val[1])  # Return first mode in case of ties
}
```

This function finds the mode of a given sample by identifying the value(s) with the highest frequency.

Next, we generate multiple resamples and calculate the mean, median, and mode for each:

```
set.seed(124)
# Generate multiple resamples and calculate mean, median, and mode
m <- 1000  # Number of resamples
means <- numeric(m)
medians <- numeric(m)
modes <- numeric(m)

for(i in 1:m) {
  resample <- sample(data, size = n, replace = TRUE)
  means[i] <- mean(resample)
  medians[i] <- median(resample)
  modes[i] <- sample.mode(resample)
}
```

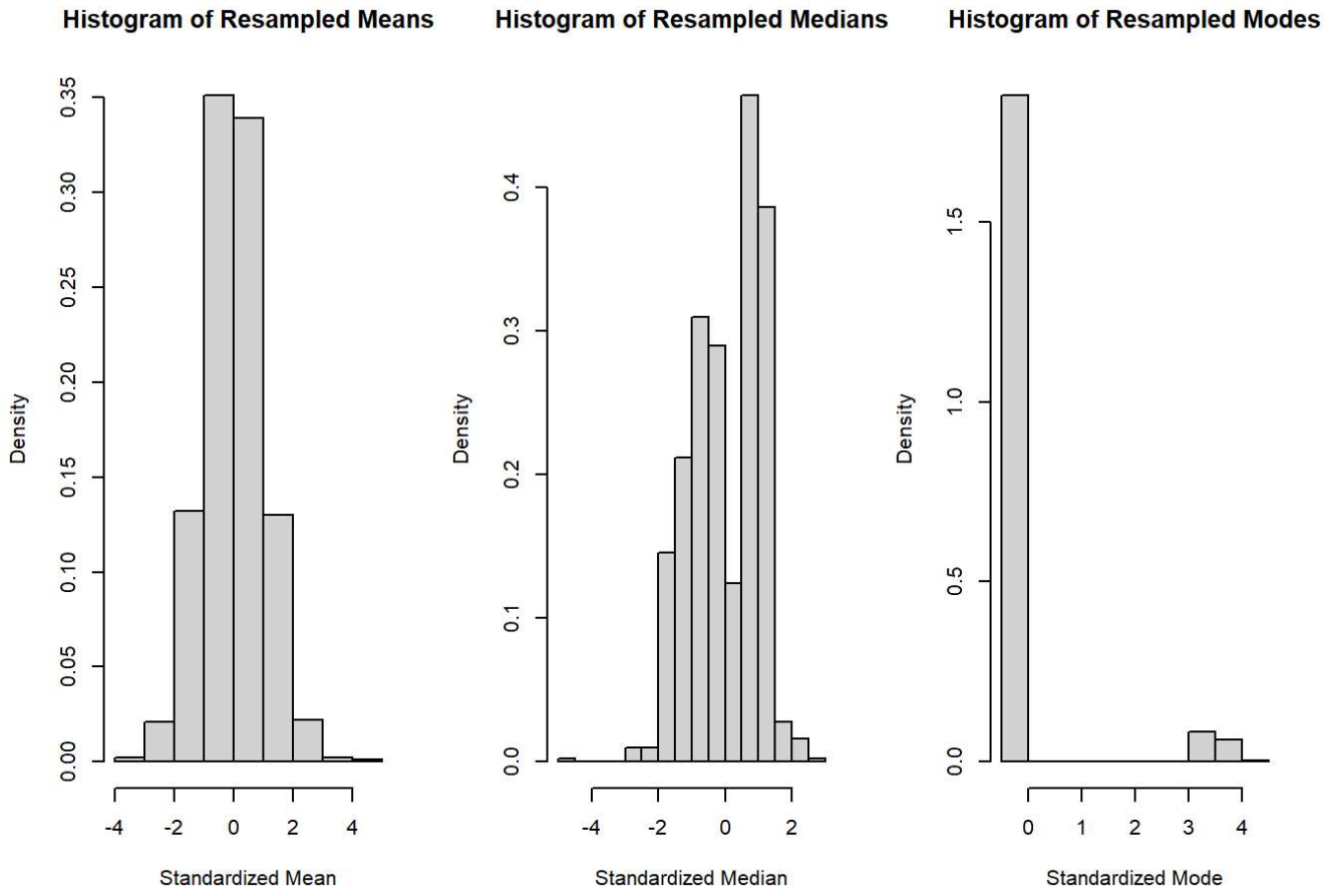
This code performs bootstrap resampling, creating 1000 resamples of the original data and calculating the mean, median, and mode for each resample.

We then normalize these sample statistics:

```
# Normalizing the vectors
means <- (means - mean(means)) / sd(means)
medians <- (medians - mean(medians)) / sd(medians)
modes <- (modes - mean(modes)) / sd(modes)
```

To visualize the distributions of these sample statistics, we create histograms:

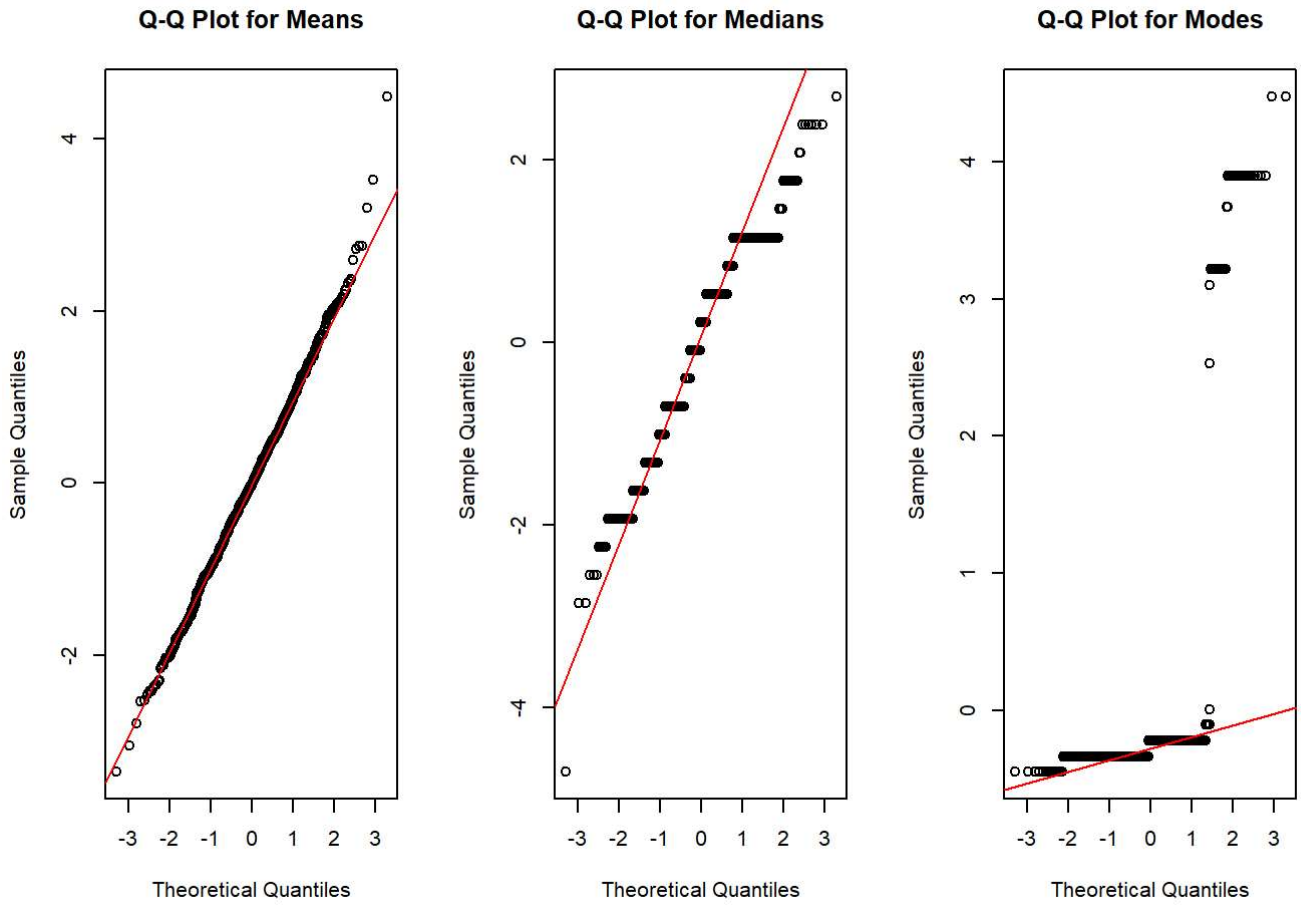
```
par(mfrow=c(1,3))

# Plot histograms
hist(means, prob = TRUE, main = "Histogram of Resampled Means", xlab = "Standardized Mean")
hist(medians, prob = TRUE, main = "Histogram of Resampled Medians", xlab = "Standardized Media
hist(modes, prob = TRUE, main = "Histogram of Resampled Modes", xlab = "Standardized Mode")
```



Histogram suggests that the data follows normal distribution.Now,lets move on to Q-Q plot for additional clarity.

Next, let's create Q-Q plots for each sample statistic:

```
par(mfrow=c(1,3))
# Q-Q Plots
qqnorm(means, main = "Q-Q Plot for Means")
qqline(means, col = "red")

qqnorm(medians, main = "Q-Q Plot for Medians")
qqline(medians, col = "red")

qqnorm(modes, main = "Q-Q Plot for Modes")
qqline(modes, col = "red")
```



- **Means:** The points lie close to the diagonal, indicating that the resampled means follow a **normal distribution**.
- **Medians:** The points show minor deviations but still mostly align with the normal distribution.
- **Modes:** The points deviate heavily, clustering at certain values, which suggests that the mode is highly discrete and non-normal(seems like).

Finally, we use our custom Kolmogorov-Smirnov test to quantitatively assess the normality of each sample statistic:

```
# Perform Kolmogorov-Smirnov test
ks.test.12(means, F_0 = rnorm)
```

```
[1] 0.498
```

```
ks.test.12(medians, F_0 = rnorm)
```

```
[1] 0.298
```

```
ks.test.12(modes, F_0 = rnorm)
```

```
[1] 0.591
```

These tests provide p-values that indicate whether each sample statistic (mean, median, and mode) follows a normal distribution. A high p-value (typically above 0.05) suggests we fail to reject the null hypothesis, indicating that the data is consistent with a normal distribution.

Interpreting the Results

The combination of visual methods (histograms and Q-Q plots) with the quantitative K-S test provides a comprehensive assessment of the normality of our sample statistics. This approach allows us to understand both the overall shape of the distributions and any specific deviations from normality.

It's important to note that while the Central Limit Theorem suggests that sample means should approach a normal distribution for large sample sizes, this may not always hold true for medians and modes, especially with smaller sample sizes or highly skewed underlying distributions.

Conclusion

This analysis demonstrates the application of custom statistical tests and visualization techniques to assess the normality of data and sample statistics. The Delhi temperature data showed slight deviations from normality, while the analysis of sample statistics from the Iris dataset provided insights into the distributional properties of means, medians, and modes in repeated sampling.

The custom implementation of the Kolmogorov-Smirnov test allowed for a flexible approach to assessing normality, complementing visual methods like histograms and Q-Q plots. This combination of graphical and numerical methods provides a comprehensive approach to exploring the distributional characteristics of data and derived statistics.