A4 part 1

2025-10-04

```
# download and save files locally
download.file("https://raw.githubusercontent.com/ghazkha/Assessment4/main/gene_expression.tsv",
              destfile = "gene_expression.tsv")

download.file("https://raw.githubusercontent.com/ghazkha/Assessment4/main/growth_data.csv",
              destfile = "growth_data.csv")

# now you can read them
gene_data <- read.table("gene_expression.tsv", header = TRUE, sep = "\t", row.names = 1)
growth_data <- read.csv("growth_data.csv")
```

```
# Import the tab-separated file
gene_data <- read.table("gene_expression.tsv",
                        header = TRUE,
                        sep = "\t",
                        row.names = 1)

# View the first 6 genes (rows)
head(gene_data)
```

```
##                                GTEX.1117F.0226.SM.5GZZ7 GTEX.1117F.0426.SM.5EGHI
## ENSG00000223972.5_DDX11L1                             0                        0
## ENSG00000227232.5_WASH7P                            187                      109
## ENSG00000278267.1_MIR6859-1                           0                        0
## ENSG00000243485.5_MIR1302-2HG                         1                        0
## ENSG00000237613.2_FAM138A                             0                        0
## ENSG00000268020.3_OR4G4P                              0                        1
##                                GTEX.1117F.0526.SM.5EGHJ
## ENSG00000223972.5_DDX11L1                             0
## ENSG00000227232.5_WASH7P                            143
## ENSG00000278267.1_MIR6859-1                           1
## ENSG00000243485.5_MIR1302-2HG                         0
## ENSG00000237613.2_FAM138A                             0
## ENSG00000268020.3_OR4G4P                              0
```

```
# Add a column for the mean expression value across all samples
gene_data$mean_expression <- rowMeans(gene_data)

# Display the first six genes again
head(gene_data)
```

```
##                                GTEX.1117F.0226.SM.5GZZ7 GTEX.1117F.0426.SM.5EGHI
## ENSG00000223972.5_DDX11L1                             0                        0
## ENSG00000227232.5_WASH7P                            187                      109
## ENSG00000278267.1_MIR6859-1                           0                        0
## ENSG00000243485.5_MIR1302-2HG                         1                        0
## ENSG00000237613.2_FAM138A                            0                        0
## ENSG00000268020.3_OR4G4P                             0                        1
##                               GTEX.1117F.0526.SM.5EGHJ mean_expression
## ENSG00000223972.5_DDX11L1                            0       0.0000000
## ENSG00000227232.5_WASH7P                           143     146.3333333
## ENSG00000278267.1_MIR6859-1                          1       0.3333333
## ENSG00000243485.5_MIR1302-2HG                        0       0.3333333
## ENSG00000237613.2_FAM138A                            0       0.0000000
## ENSG00000268020.3_OR4G4P                             0       0.3333333
```

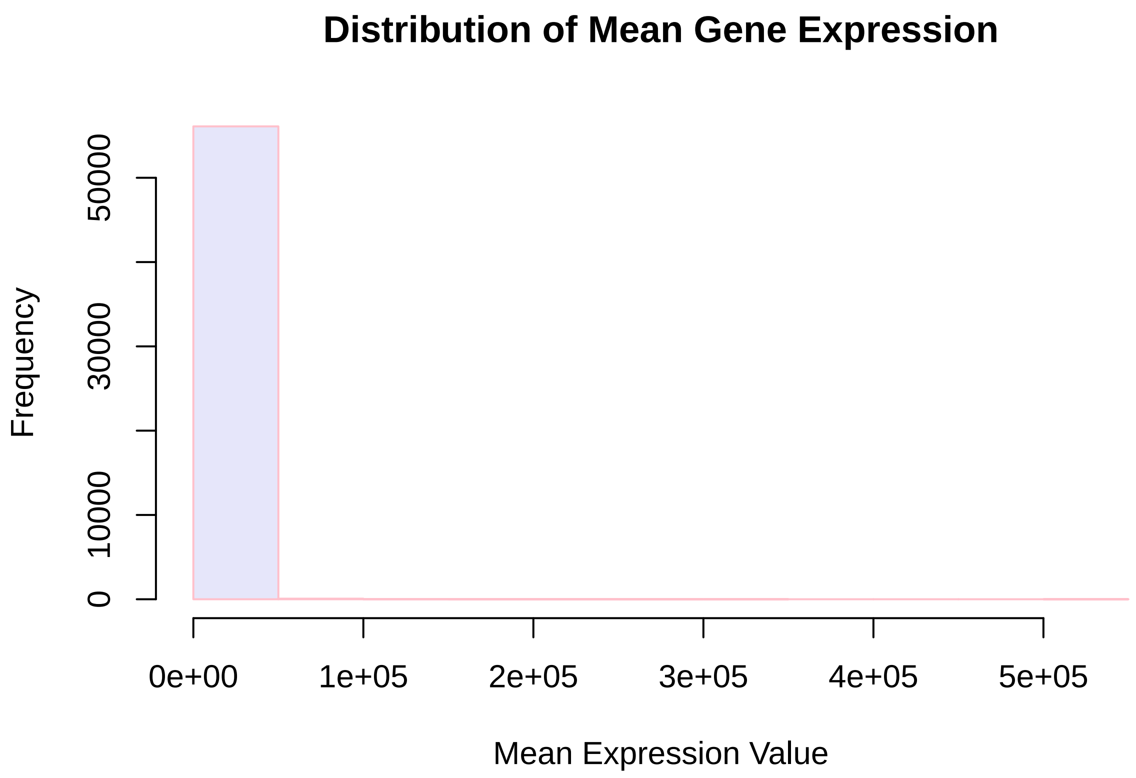
```
# Sort by mean expression and display the top 10
top10 <- head(gene_data[order(-gene_data$mean_expression), ], 10)
top10
```

```
##                           GTEX.1117F.0226.SM.5GZZ7 GTEX.1117F.0426.SM.5EGHI
## ENSG00000198804.2_MT-CO1                    267250                  1101779
## ENSG00000198886.2_MT-ND4                    273188                   991891
## ENSG00000198938.2_MT-CO3                    250277                  1041376
## ENSG00000198888.2_MT-ND1                    243853                   772966
## ENSG00000198899.2_MT-ATP6                   141374                   696715
## ENSG00000198727.2_MT-CYB                    127194                   638209
## ENSG00000198763.3_MT-ND2                    159303                   543786
## ENSG00000211445.11_GPX3                     464959                    39396
## ENSG00000198712.1_MT-CO2                    128858                   545360
## ENSG00000156508.17_EEF1A1                   317642                    39573
##                           GTEX.1117F.0526.SM.5EGHJ mean_expression
## ENSG00000198804.2_MT-CO1                    218923        529317.3
## ENSG00000198886.2_MT-ND4                    277628        514235.7
## ENSG00000198938.2_MT-CO3                    223178        504943.7
## ENSG00000198888.2_MT-ND1                    194032        403617.0
## ENSG00000198899.2_MT-ATP6                   151166        329751.7
## ENSG00000198727.2_MT-CYB                    141359        302254.0
## ENSG00000198763.3_MT-ND2                    149564        284217.7
## ENSG00000211445.11_GPX3                     306070        270141.7
## ENSG00000198712.1_MT-CO2                    122816        265678.0
## ENSG00000156508.17_EEF1A1                   339347        232187.3
```

```
# Count how many genes have mean expression below 10
low_genes <- sum(gene_data$mean_expression < 10)
low_genes
```

```
## [1] 35988
```

```
# Plot histogram of mean expression values
hist(gene_data$mean_expression,
     main = "Distribution of Mean Gene Expression",
     xlab = "Mean Expression Value",
     ylab = "Frequency",
     col = "lavender",
     border = "pink")
```



```
# Read the CSV data
growth_data <- read.csv("growth_data.csv", header = TRUE)

# Display column names
colnames(growth_data)
```

```
## [1] "Site"             "TreeID"           "Circumf_2005_cm" "Circumf_2010_cm"
## [5] "Circumf_2015_cm" "Circumf_2020_cm"
```

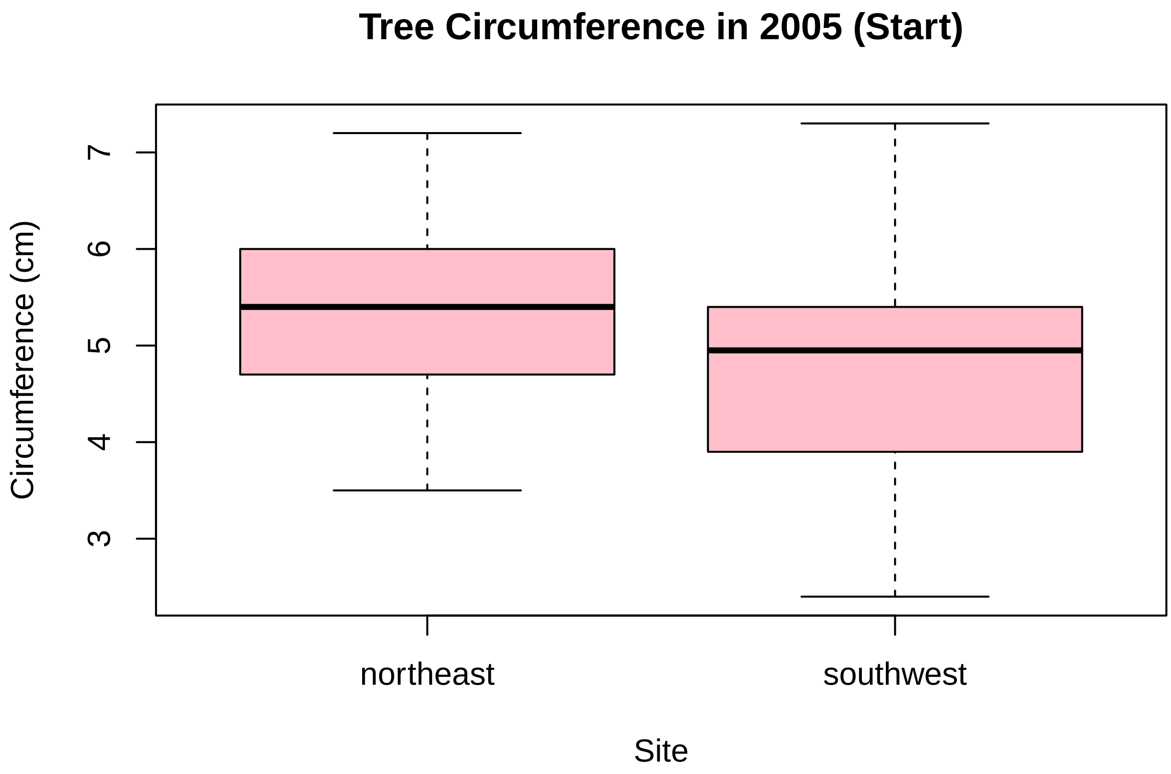
```
colnames(growth_data)
```

```
## [1] "Site"             "TreeID"           "Circumf_2005_cm" "Circumf_2010_cm"
## [5] "Circumf_2015_cm" "Circumf_2020_cm"
```

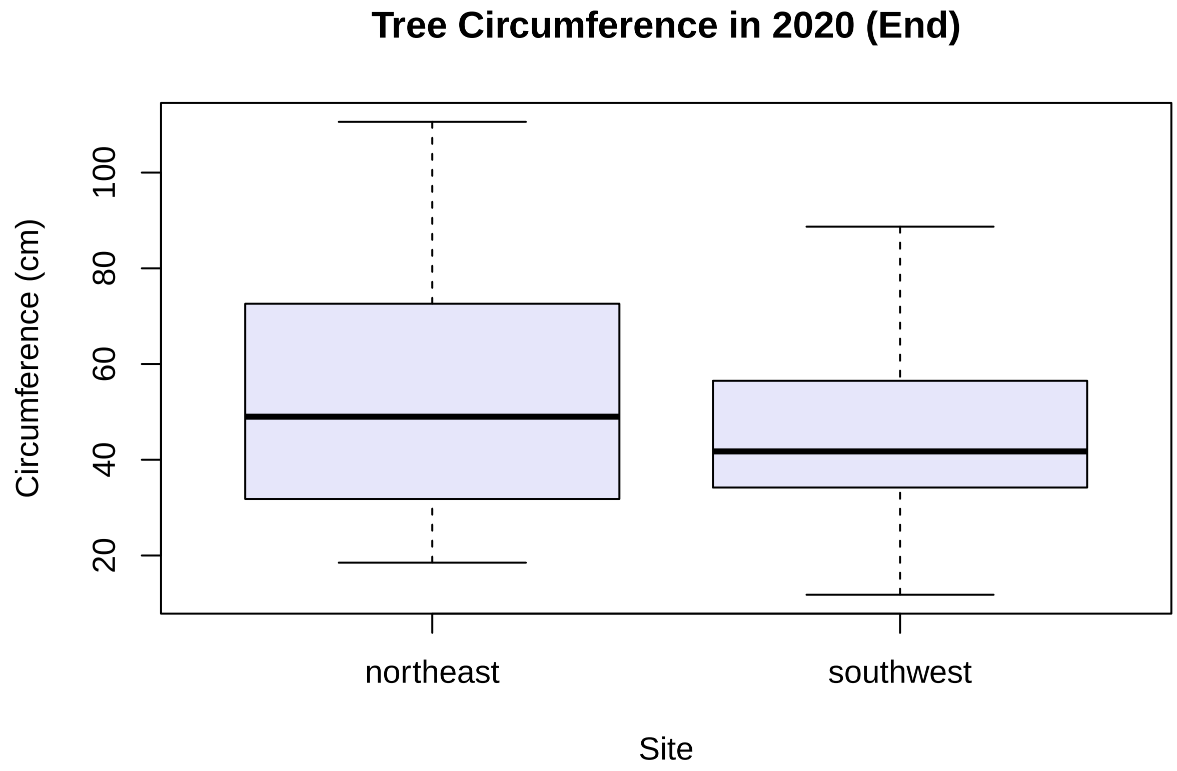
```
aggregate(cbind(Circumf_2005_cm, Circumf_2020_cm) ~ Site,
          data = growth_data,
          FUN = function(x) c(mean = mean(x, na.rm = TRUE),
                              sd = sd(x, na.rm = TRUE)))
```

```
##        Site Circumf_2005_cm.mean Circumf_2005_cm.sd Circumf_2020_cm.mean
## 1 northeast            5.2920000          0.9140267             54.22800
## 2 southwest            4.8620000          1.1474710             45.59600
##   Circumf_2020_cm.sd
## 1           25.22795
## 2           17.87345
```

```
boxplot(Circumf_2005_cm ~ Site, data = growth_data,
        main = "Tree Circumference in 2005 (Start)",
        ylab = "Circumference (cm)", col = "pink")
```



```
boxplot(Circumf_2020_cm ~ Site, data = growth_data,
        main = "Tree Circumference in 2020 (End)",
        ylab = "Circumference (cm)", col = "lavender")
```



```r
growth_data$growth_2010_2020 <- growth_data$Circumf_2020_cm - growth_data$Circumf_2010_cm
aggregate(growth_2010_2020 ~ Site, data = growth_data, mean)
```

```
##        Site growth_2010_2020
## 1 northeast            42.94
## 2 southwest            35.49
```

```r
t.test(growth_2010_2020 ~ Site, data = growth_data)
```

```
##
##  Welch Two Sample t-test
##
## data:  growth_2010_2020 by Site
## t = 1.8882, df = 87.978, p-value = 0.06229
## alternative hypothesis: true difference in means between group northeast and group southwest is not
## 95 percent confidence interval:
##  -0.3909251 15.2909251
## sample estimates:
## mean in group northeast mean in group southwest
##                   42.94                   35.49
```

```r
aggregate(cbind(Circumf_2005_cm, Circumf_2020_cm) ~ Site,
          data = growth_data,
          FUN = function(x) c(mean = mean(x), sd = sd(x)))
```

```
##        Site Circumf_2005_cm.mean Circumf_2005_cm.sd Circumf_2020_cm.mean
## 1 northeast            5.2920000          0.9140267             54.22800
## 2 southwest            4.8620000          1.1474710             45.59600
##   Circumf_2020_cm.sd
## 1           25.22795
## 2           17.87345
```

```r
## Make sure columns are numeric (safe if they already are)
growth_data$Circumf_2005_cm <- as.numeric(growth_data$Circumf_2005_cm)
growth_data$Circumf_2020_cm <- as.numeric(growth_data$Circumf_2020_cm)

## Compute mean & sd by Site
agg <- aggregate(cbind(Circumf_2005_cm, Circumf_2020_cm) ~ Site,
                 data = growth_data,
                 FUN = function(x) c(mean = mean(x, na.rm = TRUE),
                                     sd   = sd(x,   na.rm = TRUE)))

## Unnest the matrix columns into a clean data frame
out <- data.frame(
  Site       = agg$Site,
  Start_mean = agg$Circumf_2005_cm[, "mean"],
  Start_sd   = agg$Circumf_2005_cm[, "sd"],
  End_mean   = agg$Circumf_2020_cm[, "mean"],
  End_sd     = agg$Circumf_2020_cm[, "sd"],
  row.names = NULL
)

## FORCE the display (works in scripts, Rmd, and notebooks)
print(out)
```

```
##        Site Start_mean  Start_sd End_mean   End_sd
## 1 northeast      5.292 0.9140267   54.228 25.22795
## 2 southwest      4.862 1.1474710   45.596 17.87345
```