*Spatial Statistics Lab 2*

**Onyedikachi J Okeke**

2024-01-24

## **0.0 Load and summarize the data set**

# Specify the file path  
file\_path <- "C:/Users/GIS/Desktop/UNM Doc/STATISTICS PROGRAM/MY COURSES/Spring 2024 Courses/Spatial Statistics/Lab work/Lab 2/Lab2data.csv"  
  
# Load the data into R as Lab2data  
Lab2data <- read.csv(file\_path)  
  
# Print the loaded data  
summary(Lab2data)

#### Result

## Day Month Temp2 preci   
## Length:153 Length:153 Min. : 0.40 Min. : 0.1001   
## Class :character Class :character 1st Qu.:12.00 1st Qu.: 0.1022   
## Mode :character Mode :character Median :18.00 Median : 0.3000   
## Mean :18.11 Mean : 1.4066   
## 3rd Qu.:25.10 3rd Qu.: 1.2345   
## Max. :31.90 Max. :14.5000   
## pollen EVI   
## Min. : 0.000 Min. :2058   
## 1st Qu.: 9.002 1st Qu.:3511   
## Median : 21.605 Median :5401   
## Mean : 192.158 Mean :4577   
## 3rd Qu.: 99.432 3rd Qu.:5541   
## Max. :5048.354 Max. :5569

## **1.0 Find the dimensions of the loaded data**

# Print the dimensions of the loaded data  
cat("Dimensions of Lab2data:", dim(Lab2data), "\n")

#### Result

## Dimensions of Lab2data: 153 6

## **2.0 What are the pollen count values for the first 10 days?**

# Count values for the first 10 days  
pollen\_count <- Lab2data$pollen[1:10]  
print(pollen\_count)

#### Result

## [1] 156.881313 15.467172 205.492424 26.515152 124.228395 45.010288  
## [7] 46.810700 3.600823 3.600823 9.002058

## **3.0 Calculate the following statistics for pollen count**

# Assuming "pollen\_count" is the column name in Lab2data  
pollen\_count <- Lab2data$pollen  
  
# Calculate mean  
mean\_pollen <- mean(pollen\_count)  
  
# Calculate variance  
var\_pollen <- var(pollen\_count)  
  
# Calculate standard deviation  
std\_dev\_pollen <- sd(pollen\_count)  
  
# Calculate quantiles (e.g., 25th, 50th, and 75th percentiles)  
quantiles\_pollen <- quantile(pollen\_count, c(0.25, 0.50, 0.75))  
  
# Print the results  
cat("Mean Pollen Count:", mean\_pollen, "\n")  
cat("Variance of Pollen Count:", var\_pollen, "\n")  
cat("Standard Deviation of Pollen Count:", std\_dev\_pollen, "\n")  
cat("Quantiles of Pollen Count (25%, 50%, 75%):", quantiles\_pollen, "\n")

#### Result

## Mean Pollen Count: 192.1584

## Variance of Pollen Count: 344436.2

## Standard Deviation of Pollen Count: 586.8868

## Quantiles of Pollen Count (25%, 50%, 75%): 9.002058 21.60494 99.43182

## **4.0 Statistics for pollen count in April?**

# Assuming "Date" is the column name in Lab2data for dates, and "pollen\_count" for pollen counts  
  
# Filter data for April  
april\_data <- subset(Lab2data,Month == "April")  
  
# Assuming "pollen\_count" is the column name for pollen counts  
pollen\_count\_april <- april\_data$pollen  
  
#print data  
print(pollen\_count\_april)  
  
# Calculate mean  
mean\_pollen\_april <- mean(pollen\_count\_april)  
  
# Calculate variance  
var\_pollen\_april <- var(pollen\_count\_april)  
  
# Calculate standard deviation  
std\_dev\_pollen\_april <- sd(pollen\_count\_april)  
  
# Calculate quantiles (e.g., 25th, 50th, and 75th percentiles)  
quantiles\_pollen\_april <- quantile(pollen\_count\_april, c(0.25, 0.50, 0.75))  
  
# Print the results  
cat("Mean Pollen Count in April:", mean\_pollen\_april, "\n")  
cat("Variance of Pollen Count in April:", var\_pollen\_april, "\n")  
cat("Standard Deviation of Pollen Count in April:", std\_dev\_pollen\_april, "\n")  
cat("Quantiles of Pollen Count in April (25%, 50%, 75%):", quantiles\_pollen\_april, "\n")

#### Result

## [1] 156.881313 15.467172 205.492424 26.515152 124.228395 45.010288  
## [7] 46.810700 3.600823 3.600823 9.002058 28.806584 48.611111  
## [13] 17.676768 14.362374 22.095960 10.802469 10.802469 10.802469  
## [19] 14.403292 48.611111 99.431818 1586.489899 1884.785354 3834.876543  
## [25] 5048.353909 873.199589 806.502525 1889.204545 1416.351010 487.911523

## Mean Pollen Count in April: 626.3563

## Variance of Pollen Count in April: 1442156

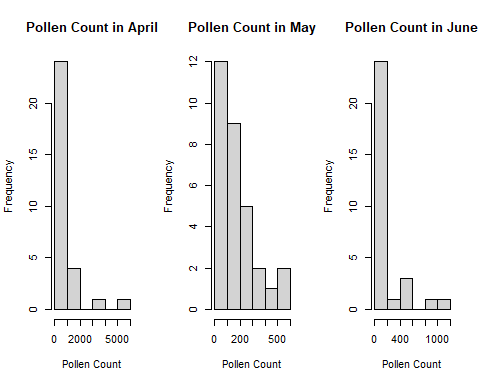
## Standard Deviation of Pollen Count in April: 1200.898

## Quantiles of Pollen Count in April (25%, 50%, 75%): 14.66926 47.71091 726.8548

## **5.0 Histograms (in frequencies) of pollen count in April, May, and June**

# Assuming "Month" and "pollen\_count" are the column names in Lab2data  
# Assuming "Date" is already in Date format  
  
# Subset data for April, May, and June  
may\_data <- subset(Lab2data,Month == "May")  
pollen\_count\_may <- may\_data$pollen  
  
june\_data <- subset(Lab2data,Month == "June")  
pollen\_count\_june <- june\_data$pollen  
  
# Create a 1x3 layout for the histograms  
par(mfrow = c(1, 3))  
  
# Plot histogram for April  
hist(pollen\_count\_april, main = "Pollen Count in April", xlab = "Pollen Count", ylab = "Frequency")  
  
# Plot histogram for May  
hist(pollen\_count\_may, main = "Pollen Count in May", xlab = "Pollen Count", ylab = "Frequency")  
  
# Plot histogram for June  
hist(pollen\_count\_june, main = "Pollen Count in June", xlab = "Pollen Count", ylab = "Frequency")  
  
# Reset the layout to default  
par(mfrow = c(1, 1))

#### Result



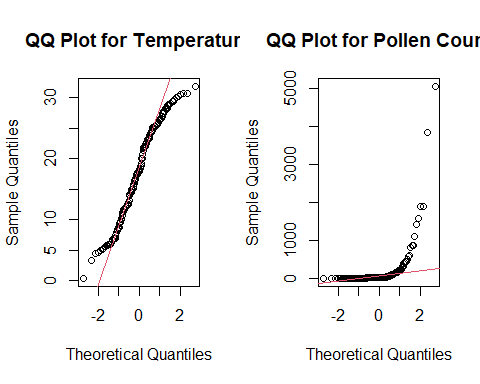
## **6.0 Use qqnorm and qqline to construct plots for temperature and pollen count**

par(mfrow=c(1,2))  
  
# QQ plot for temperature  
qqnorm(Lab2data$Temp2, main = "QQ Plot for Temperature")  
qqline(Lab2data$Temp2, col = 2)  
  
  
# QQ plot for pollen count  
qqnorm(Lab2data$pollen, main = "QQ Plot for Pollen Count")  
qqline(Lab2data$pollen, col = 2)  
  
  
# Shapiro-Wilk Normality Test for temperature  
shapiro\_temp <- shapiro.test(Lab2data$Temp2)  
cat("Shapiro-Wilk Normality Test for Temperature:\n", "p-value =", shapiro\_temp$p.value, "\n")  
  
# Shapiro-Wilk Normality Test for pollen count  
shapiro\_pollen <- shapiro.test(Lab2data$pollen)  
cat("Shapiro-Wilk Normality Test for Pollen Count:\n", "p-value =", shapiro\_pollen$p.value, "\n"

#### Result

## Shapiro-Wilk Normality Test for Temperature:  
## p-value = 0.0002311045

## Shapiro-Wilk Normality Test for Pollen Count:  
## p-value = 2.086788e-23



The Shapiro-Wilk Normality Test is used to assess whether a sample comes from a normally distributed population. In this case, the p-values obtained for both the temperature and pollen count are extremely small:

Temperature:

Shapiro-Wilk p-value: 0.0002311045 Pollen Count:

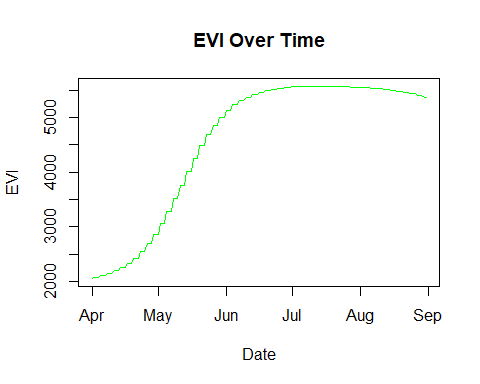
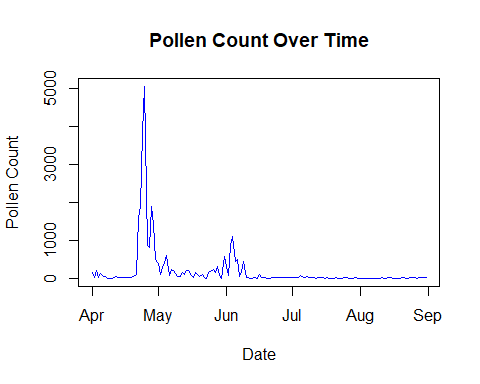
Shapiro-Wilk p-value: 2.086788e-23 (very close to zero)

The small p-values in both cases suggest that we have evidence to reject the null hypothesis that the data follows a normal distribution. In other words, the data for both temperature and pollen count are likely not normally distributed.

## **7.0 Plot to construct two graphs for pollen count and EVI for the whole dataset**

# Convert the "Day" column to a Date type  
Lab2data$Day <- as.Date(Lab2data$Day, format="%m/%d/%Y")  
  
# Plot 1: Pollen Count  
plot(Lab2data$Day, Lab2data$pollen, type="l", col="blue", xlab="Date", ylab="Pollen Count", main="Pollen Count Over Time")  
  
# Plot 2: EVI  
plot(Lab2data$Day, Lab2data$EVI, type="l", col="green", xlab="Date", ylab="EVI", main="EVI Over Time")

#### Result



## **8.0 two-sample t-test to test**

# Perform a two-sample t-test  
t\_test\_result <- t.test(pollen\_count\_april, pollen\_count\_may)  
print(t\_test\_result)

#### Result

##   
## Welch Two Sample t-test  
##   
## data: pollen\_count\_april and pollen\_count\_may  
## t = 2.0501, df = 29.91, p-value = 0.04921  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## 1.681686 904.343558  
## sample estimates:  
## mean of x mean of y   
## 626.3563 173.3437

## **9.0 two-sample t-test to test**

# Fit a linear regression model  
regression\_model <- lm(pollen ~ EVI, data = Lab2data)  
  
# Get the formula and coefficients  
formula <- as.formula(regression\_model)  
coefficients <- coef(regression\_model)  
  
# Print the formula and coefficients  
cat("Regression Formula:", format(formula), "\n")  
cat("Coefficients:\n")  
print(coefficients)  
  
# Check the significance of the predictor (EVI)  
summary(regression\_model)  
  
par(mfrow=c(1,2))  
  
# Plot the data and add the fitted regression line  
plot(Lab2data$EVI, Lab2data$Pollen, xlab = "EVI", ylab = "Pollen Count", main = "Pollen Count Vs EVI")  
abline(regression\_model, col = "red")  
  
# Plot residuals in a separate graph  
residuals <- residuals(regression\_model)  
plot(Lab2data$EVI, residuals, xlab = "EVI", ylab = "Residuals", main = "Residuals Vs EVI

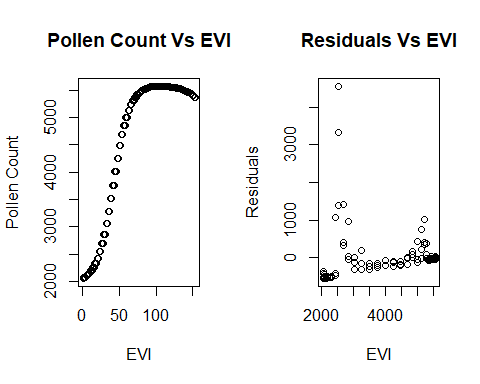
#### Result

## Regression Formula: pollen ~ EVI

## Coefficients:

## (Intercept) EVI   
## 874.7703750 -0.1491512

##   
## Call:  
## lm(formula = pollen ~ EVI, data = Lab2data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -550.8 -112.5 -43.7 -23.3 4553.6   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 874.77038 166.88763 5.242 5.28e-07 \*\*\*  
## EVI -0.14915 0.03511 -4.248 3.76e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 556.5 on 151 degrees of freedom  
## Multiple R-squared: 0.1067, Adjusted R-squared: 0.1008   
## F-statistic: 18.04 on 1 and 151 DF, p-value: 3.764e-05



## **10.0 two-sample t-test to test**

# Assuming "Pollen," "EVI," "Precipitation," and "Temperature" are the correct column names in Lab2data  
# Fit a multiple linear regression model  
regression\_model <- lm(pollen ~ EVI + preci + Temp2, data = Lab2data)  
  
# Get the formula and coefficients  
formula <- as.formula(regression\_model)  
coefficients <- coef(regression\_model)  
  
# Print the formula and coefficients  
cat("Regression Formula:", format(formula), "\n")  
cat("Coefficients:\n")  
print(coefficients)  
  
# Check the significance of the predictors  
summary(regression\_model)  
  
# Plot residuals  
residuals <- residuals(regression\_model)  
plot(residuals, xlab = "Observation", ylab = "Residuals", main = "Residuals Plot")  
abline(h = 0, col = "red", lty = 2) # Add a horizontal line at y = 0

#### Result

## Regression Formula: pollen ~ EVI + preci + Temp2

## Coefficients:

## (Intercept) EVI preci Temp2   
## 1084.1682948 -0.3411214 -14.1307435 38.0543237

##   
## Call:  
## lm(formula = pollen ~ EVI + preci + Temp2, data = Lab2data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -680.4 -217.1 -78.5 119.2 4157.5   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1084.1683 169.1924 6.408 1.83e-09 \*\*\*  
## EVI -0.3411 0.0619 -5.511 1.53e-07 \*\*\*  
## preci -14.1307 17.2757 -0.818 0.414689   
## Temp2 38.0543 10.2008 3.731 0.000271 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 534.2 on 149 degrees of freedom  
## Multiple R-squared: 0.1877, Adjusted R-squared: 0.1714   
## F-statistic: 11.48 on 3 and 149 DF, p-value: 8.161e-07

