

Data Management & Descriptive Statistics SAP

Saleh Abednezhad

2024-2025

```
## Import Data
data <- read.table("C:/UHasselt/Courses/Project Learning from Data/Project Codes/final data.txt",
                  header = TRUE, sep = ";", dec = "\t")
```

```
## Percentage missing values
(colMeans(is.na(data)))*100
```

```
## Region Crop Lutum Sand Silt pH C Cstock Years
## 1.25 0.00 1.25 1.25 1.25 1.25 0.00 0.00 0.00
```

```
## Data preparing
data$Lutum <- as.numeric(as.character(data$Lutum))
data$Sand <- as.numeric(as.character(data$Sand))
data$Silt <- as.numeric(as.character(data$Silt))
data$pH <- as.numeric(as.character(data$pH))
data$C <- as.numeric(as.character(data$C))
data$Cstock <- as.numeric(as.character(data$Cstock))

clean_data <- data[-c(33, 36, 66),]
summary(clean_data)
```

```
##      Region      Crop      Lutum      Sand
## Length:397      Length:397      Min.   :0.0400      Min.   :0.1700
## Class :character Class :character 1st Qu.:0.1700      1st Qu.:0.2400
## Mode  :character Mode  :character Median :0.2200      Median :0.2750
##                                     Mean  :0.2207      Mean   :0.3882
##                                     3rd Qu.:0.2700      3rd Qu.:0.5900
##                                     Max.   :0.4200      Max.   :0.7300
##                                     NA's   :5          NA's   :5
##      Silt      pH      C      Cstock
## Min.   :0.1400      Min.   :4.64      Min.   : 1.550      Min.   : 73.42
## 1st Qu.:0.2300      1st Qu.:5.21      1st Qu.: 1.820      1st Qu.: 89.41
## Median :0.4400      Median :5.83      Median : 2.010      Median : 97.38
## Mean   :0.3909      Mean   :5.65      Mean   : 3.806      Mean   :612.56
## 3rd Qu.:0.5100      3rd Qu.:6.04      3rd Qu.: 2.280      3rd Qu.:104.71
## Max.   :0.6900      Max.   :6.68      Max.   :252.000      Max.   :108650.00
## NA's   :5          NA's   :5
##      Years
## Min.   :10.00
## 1st Qu.:14.00
## Median :17.00
```

```
## Mean :17.34
## 3rd Qu.:21.00
## Max. :25.00
##
```

```
clean_data <- na.omit(clean_data)
summary(clean_data)
```

```
##      Region      Crop      Lutum      Sand
## Length:382      Length:382      Min. :0.0400      Min. :0.1700
## Class :character      Class :character      1st Qu.:0.1700      1st Qu.:0.2400
## Mode :character      Mode :character      Median :0.2200      Median :0.2800
##                                         Mean :0.2204      Mean :0.3899
##                                         3rd Qu.:0.2700      3rd Qu.:0.5900
##                                         Max. :0.4200      Max. :0.7300
##      Silt      pH      C      Cstock
## Min. :0.1400      Min. :4.640      Min. : 1.550      Min. : 73.42
## 1st Qu.:0.2300      1st Qu.:5.210      1st Qu.: 1.820      1st Qu.: 88.90
## Median :0.4400      Median :5.810      Median : 2.010      Median : 97.34
## Mean :0.3895      Mean :5.644      Mean : 3.874      Mean : 632.57
## 3rd Qu.:0.5100      3rd Qu.:6.040      3rd Qu.: 2.250      3rd Qu.: 104.64
## Max. :0.6900      Max. :6.680      Max. :252.000      Max. :108650.00
##      Years
## Min. :10.00
## 1st Qu.:14.00
## Median :17.00
## Mean :17.32
## 3rd Qu.:21.00
## Max. :25.00
```

```
removed_indices <- which(!complete.cases(data))
removed_indices
```

```
## [1] 93 96 145 162 166 185 242 250 251 258 270 346 354 358 385
```

```
rows_greater_than_3 <- which(clean_data$C > 3)
rows_greater_than_3
```

```
## [1] 12 128 254
```

```
clean_data <- clean_data[-c(12, 128, 254),]
clean_data <- clean_data[clean_data$C <= 3, ]
summary(clean_data)
```

```
##      Region      Crop      Lutum      Sand
## Length:379      Length:379      Min. :0.0400      Min. :0.1700
## Class :character      Class :character      1st Qu.:0.1700      1st Qu.:0.2400
## Mode :character      Mode :character      Median :0.2200      Median :0.2800
##                                         Mean :0.2198      Mean :0.3902
##                                         3rd Qu.:0.2700      3rd Qu.:0.5900
##                                         Max. :0.4200      Max. :0.7300
```

```
##      Silt          pH          C          Cstock
## Min.   :0.1400   Min.   :4.640   Min.    :1.550   Min.    : 73.42
## 1st Qu.:0.2300   1st Qu.:5.210   1st Qu.:1.820   1st Qu.: 89.03
## Median :0.4400   Median :5.810   Median :2.010   Median : 97.31
## Mean   :0.3898   Mean    :5.644   Mean    :2.074   Mean    : 636.72
## 3rd Qu.:0.5100   3rd Qu.:6.040   3rd Qu.:2.245   3rd Qu.: 104.48
## Max.   :0.6900   Max.    :6.680   Max.    :2.970   Max.    :108650.00
##      Years
## Min.    :10.00
## 1st Qu. :14.00
## Median  :17.00
## Mean    :17.35
## 3rd Qu. :21.00
## Max.    :25.00
```

```
rows_greater_than_130 <- which(clean_data$Cstock > 130)
rows_greater_than_130
```

```
## [1] 149 349
```

```
clean_data <- clean_data[-c(149, 349),]
summary(data)
```

```
##      Region          Crop          Lutum          Sand
## Length:400   Length:400   Min.    :0.0400   Min.    :0.1700
## Class :character   Class :character   1st Qu.:0.1700   1st Qu.:0.2400
## Mode  :character   Mode  :character   Median :0.2200   Median :0.2800
##                                     Mean   :0.2202   Mean   :0.3881
##                                     3rd Qu.:0.2700   3rd Qu.:0.5900
##                                     Max.    :0.4200   Max.    :0.7300
##                                     NA's    :5        NA's    :5
##      Silt          pH          C          Cstock
## Min.   :0.1400   Min.   :4.64   Min.    : 1.550   Min.    : 73.42
## 1st Qu.:0.2300   1st Qu.:5.21   1st Qu.: 1.820   1st Qu.: 89.38
## Median :0.4400   Median :5.83   Median : 2.010   Median : 97.35
## Mean   :0.3915   Mean    :5.65   Mean    : 3.792   Mean    : 608.68
## 3rd Qu.:0.5100   3rd Qu.:6.04   3rd Qu.: 2.265   3rd Qu.: 104.71
## Max.   :0.6900   Max.    :6.68   Max.    :252.000   Max.    :108650.00
## NA's    :5        NA's    :5
##      Years
## Min.    :10.00
## 1st Qu. :14.00
## Median  :17.00
## Mean    :17.32
## 3rd Qu. :21.00
## Max.    :25.00
##
```

```
summary(clean_data)
```

```
##      Region          Crop          Lutum          Sand
```

```
## Length:377      Length:377      Min.    :0.0400   Min.    :0.170
## Class :character Class :character 1st Qu.:0.1700   1st Qu.:0.240
## Mode  :character Mode  :character Median :0.2200   Median :0.280
##                                     Mean  :0.2194   Mean  :0.391
##                                     3rd Qu.:0.2700   3rd Qu.:0.590
##                                     Max.   :0.4200   Max.   :0.730
##      Silt                pH                C                Cstock
## Min.    :0.1400   Min.    :4.640   Min.    :1.550   Min.    : 73.42
## 1st Qu.:0.2300   1st Qu.:5.210   1st Qu.:1.820   1st Qu.: 88.77
## Median :0.4400   Median :5.800   Median :2.010   Median : 97.29
## Mean   :0.3894   Mean   :5.642   Mean   :2.075   Mean   : 97.97
## 3rd Qu.:0.5100   3rd Qu.:6.040   3rd Qu.:2.250   3rd Qu.:104.37
## Max.   :0.6900   Max.   :6.680   Max.   :2.970   Max.   :126.85
##      Years
## Min.    :10.00
## 1st Qu.:14.00
## Median :17.00
## Mean   :17.34
## 3rd Qu.:21.00
## Max.   :25.00
```

```
clean_data$Region <- as.factor(clean_data$Region)
count(clean_data,Region)
```

```
##      Region    n
## 1    Keempen    1
## 2    Kempeen    1
## 3    Kempen  149
## 4 Leemstreek 224
## 5 Leemstrek    1
## 6 Lemstreek    1
```

```
clean_data <- clean_data %>%
  mutate(Region = case_when(
    Region == "Kempeen" ~ "Kempen",
    Region == "Keempen" ~ "Kempen",
    Region == "Leemstrek" ~ "Leemstreek",
    Region == "Lemstreek" ~ "Leemstreek",
    TRUE ~ Region
  ))
sort(clean_data$C)
```

```
## [1] 1.55 1.58 1.60 1.63 1.64 1.64 1.64 1.64 1.65 1.66 1.66 1.67 1.67 1.67 1.68
## [16] 1.69 1.69 1.69 1.70 1.70 1.70 1.70 1.70 1.71 1.71 1.71 1.71 1.71 1.71 1.71
## [31] 1.72 1.72 1.72 1.72 1.72 1.72 1.73 1.73 1.73 1.73 1.73 1.73 1.73 1.74 1.74
## [46] 1.74 1.75 1.75 1.75 1.75 1.75 1.75 1.76 1.76 1.76 1.76 1.76 1.76 1.76 1.76
## [61] 1.76 1.76 1.76 1.76 1.77 1.77 1.77 1.77 1.77 1.77 1.77 1.77 1.77 1.78 1.78
## [76] 1.78 1.78 1.79 1.79 1.79 1.79 1.79 1.79 1.79 1.79 1.79 1.80 1.80 1.80 1.81
## [91] 1.81 1.81 1.81 1.82 1.82 1.82 1.82 1.82 1.82 1.82 1.82 1.82 1.83 1.83 1.83
## [106] 1.83 1.83 1.83 1.84 1.84 1.84 1.84 1.84 1.85 1.85 1.85 1.85 1.85 1.85 1.85
## [121] 1.85 1.85 1.85 1.86 1.86 1.86 1.86 1.86 1.86 1.86 1.87 1.87 1.87 1.87 1.87
## [136] 1.87 1.88 1.88 1.88 1.88 1.88 1.89 1.89 1.89 1.89 1.89 1.90 1.90 1.90 1.91
```

```
## [151] 1.91 1.91 1.91 1.91 1.92 1.93 1.93 1.93 1.94 1.94 1.94 1.95 1.95 1.95 1.95
## [166] 1.96 1.96 1.96 1.96 1.97 1.97 1.97 1.97 1.97 1.98 1.98 1.98 1.99 1.99 1.99
## [181] 1.99 2.00 2.00 2.00 2.00 2.00 2.00 2.01 2.01 2.01 2.02 2.02 2.02 2.02 2.02
## [196] 2.03 2.03 2.03 2.03 2.03 2.03 2.03 2.03 2.03 2.04 2.04 2.05 2.05 2.05 2.05
## [211] 2.06 2.06 2.06 2.06 2.07 2.07 2.07 2.07 2.08 2.08 2.08 2.08 2.09 2.09 2.09
## [226] 2.09 2.09 2.10 2.10 2.10 2.10 2.10 2.11 2.11 2.11 2.11 2.11 2.12 2.13 2.13
## [241] 2.13 2.13 2.13 2.13 2.13 2.14 2.14 2.14 2.14 2.14 2.14 2.15 2.15 2.15 2.15
## [256] 2.16 2.16 2.16 2.17 2.17 2.17 2.17 2.17 2.18 2.18 2.18 2.18 2.18 2.19 2.19
## [271] 2.19 2.19 2.19 2.19 2.20 2.20 2.20 2.20 2.21 2.21 2.21 2.24 2.25 2.25 2.25
## [286] 2.28 2.28 2.28 2.29 2.30 2.31 2.31 2.31 2.32 2.34 2.34 2.35 2.35 2.35 2.36
## [301] 2.36 2.36 2.38 2.38 2.38 2.39 2.39 2.41 2.41 2.41 2.42 2.42 2.42 2.43 2.43
## [316] 2.44 2.44 2.44 2.44 2.45 2.45 2.46 2.46 2.46 2.46 2.49 2.49 2.51 2.51 2.52
## [331] 2.52 2.53 2.54 2.54 2.54 2.55 2.55 2.56 2.57 2.57 2.57 2.57 2.57 2.58 2.58
## [346] 2.58 2.64 2.66 2.66 2.70 2.72 2.72 2.72 2.72 2.75 2.75 2.76 2.76 2.77 2.77
## [361] 2.77 2.81 2.81 2.82 2.83 2.83 2.83 2.83 2.84 2.86 2.87 2.87 2.87 2.87 2.88
## [376] 2.94 2.97
```

```
sort(clean_data$Cstock)
```

```
## [1] 73.42 75.92 76.28 76.40 77.14 78.11 78.44 78.66 78.70 78.95
## [11] 79.08 79.10 79.22 79.80 80.43 80.47 80.53 80.92 81.12 81.20
## [21] 81.33 81.46 81.71 81.72 81.81 81.92 82.13 82.30 82.39 82.39
## [31] 82.61 82.63 82.63 82.65 82.81 82.91 82.92 83.15 83.17 83.34
## [41] 83.44 83.52 83.58 83.60 83.64 83.77 83.95 84.01 84.15 84.20
## [51] 84.23 84.44 84.44 84.46 84.85 84.86 84.95 84.95 84.96 84.99
## [61] 85.32 85.66 85.80 85.81 86.01 86.11 86.13 86.24 86.28 86.29
## [71] 86.36 86.39 86.39 86.49 86.64 86.66 86.72 86.76 86.88 87.03
## [81] 87.30 87.31 87.42 87.47 87.62 87.82 88.27 88.33 88.40 88.42
## [91] 88.48 88.54 88.65 88.71 88.77 89.29 89.29 89.41 89.41 89.48
## [101] 89.48 89.57 89.57 89.88 89.92 90.10 90.68 90.81 90.81 90.97
## [111] 91.01 91.31 91.34 91.59 91.60 91.66 91.73 91.84 92.02 92.04
## [121] 92.06 92.43 92.58 92.58 92.77 93.04 93.11 93.27 93.36 93.54
## [131] 93.58 93.61 93.88 94.01 94.22 94.30 94.34 94.39 94.53 94.58
## [141] 94.68 94.72 94.81 94.84 94.90 94.93 94.98 95.01 95.05 95.06
## [151] 95.08 95.13 95.26 95.26 95.33 95.37 95.37 95.51 95.56 95.58
## [161] 95.64 96.05 96.06 96.08 96.10 96.13 96.16 96.16 96.20 96.24
## [171] 96.28 96.34 96.34 96.37 96.39 96.49 96.51 96.61 96.64 96.65
## [181] 96.82 97.09 97.15 97.15 97.21 97.22 97.24 97.25 97.29 97.31
## [191] 97.38 97.40 97.50 97.52 97.59 97.66 97.71 97.74 97.76 97.81
## [201] 97.85 97.86 98.01 98.05 98.06 98.07 98.07 98.24 98.31 98.45
## [211] 98.66 98.74 98.84 98.90 99.11 99.20 99.36 99.39 99.53 99.59
## [221] 99.62 99.64 99.67 99.68 99.80 99.98 99.98 100.11 100.12 100.15
## [231] 100.20 100.29 100.42 100.45 100.45 100.46 100.68 100.70 100.83 100.85
## [241] 100.96 101.02 101.14 101.27 101.31 101.38 101.46 101.52 101.65 101.65
## [251] 101.79 101.88 101.88 101.93 102.02 102.04 102.13 102.18 102.19 102.26
## [261] 102.34 102.47 102.47 102.56 102.76 102.84 102.85 103.18 103.26 103.38
## [271] 103.46 103.67 103.69 103.80 103.80 103.83 104.06 104.07 104.08 104.13
## [281] 104.18 104.28 104.37 104.46 104.50 104.68 104.71 104.80 104.83 104.97
## [291] 105.03 105.08 105.19 105.20 105.58 105.64 105.85 105.89 105.97 106.00
## [301] 106.09 106.20 106.81 106.84 106.88 106.89 106.93 107.12 107.19 107.25
## [311] 107.26 107.57 107.62 107.71 107.89 108.14 108.71 109.17 109.26 109.57
## [321] 109.63 109.82 109.99 110.38 110.66 111.06 111.75 111.91 112.14 112.85
## [331] 112.98 115.76 116.03 116.25 116.42 116.80 117.26 117.29 117.58 117.79
## [341] 117.80 118.06 118.08 118.21 118.28 118.32 118.35 118.38 118.42 118.42
```

```
## [351] 118.53 118.60 119.11 119.21 119.58 119.90 119.99 120.01 120.78 120.94
## [361] 121.01 121.06 121.08 121.28 121.58 121.58 121.66 121.78 122.01 122.46
## [371] 122.99 123.40 123.92 124.32 124.95 126.52 126.85
```

```
summary(clean_data$Cstock)
```

```
##      Min. 1st Qu.  Median      Mean 3rd Qu.     Max.
## 73.42   88.77   97.29   97.97  104.37  126.85
```

```
summary(clean_data$C)
```

```
##      Min. 1st Qu.  Median      Mean 3rd Qu.     Max.
## 1.550   1.820   2.010   2.075   2.250   2.970
```

```
clean_data$Crop <- as.factor(clean_data$Crop)
count(clean_data, Crop)
```

```
##      Crop      n
## 1      GM     98
## 2      PG     82
## 3      SM    115
## 4     SMGR     82
```

```
class(clean_data$Region)
```

```
## [1] "character"
```

```
clean_data$Region <- as.factor(clean_data$Region)
clean_data$Crop <- as.factor(clean_data$Crop)
```

```
#Export clean data
```

```
write.csv(clean_data, "C:/UHasselt/Courses/Project Learning from Data/Project Codes/Clean_data.txt",
          row.names = FALSE)
list.files("C:/UHasselt/Courses/Project Learning from Data/Project Codes/")
```

```
## [1] "000010.png"      "000011.png"      "000012.png"      "000013.png"
## [5] "000014.png"      "boxplot.png"     "Casual Graph.png" "Clean_data.txt"
## [9] "final data.txt"  "Project.rmd"
```

```
# Step 4: Exploratory Data Analysis (EDA)
```

```
# Summary statistics by Region and Crop
```

```
clean_data %>%
  group_by(Region, Crop) %>%
  summarize(mean_Cstock = mean(Cstock, na.rm = TRUE),
            sd_Cstock = sd(Cstock, na.rm = TRUE),
            n = n())
```

```
## 'summarise()' has grouped output by 'Region'. You can override using the
## '.groups' argument.
```

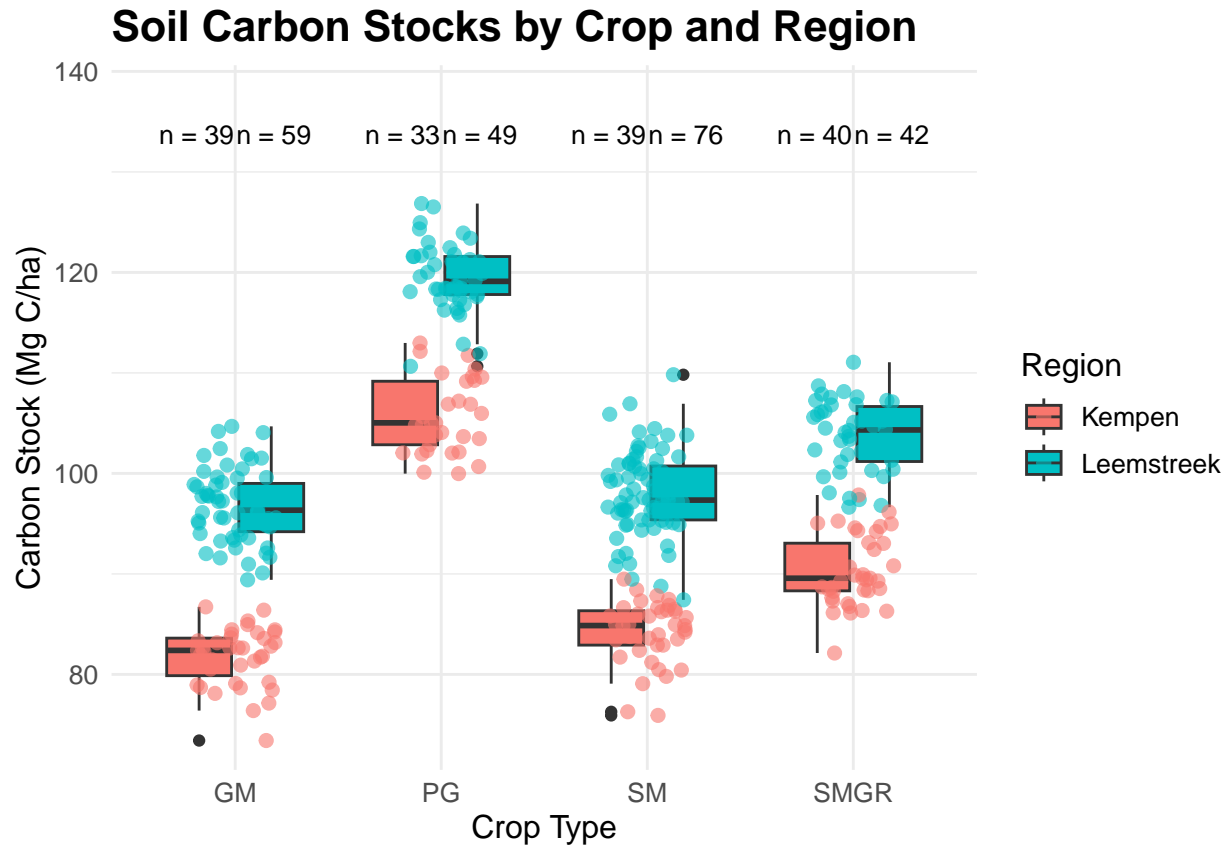
```
## # A tibble: 8 x 5
## # Groups:   Region [2]
##   Region    Crop mean_Cstock sd_Cstock    n
##   <fct>    <fct>      <dbl>      <dbl> <int>
## 1 Kempen    GM          81.8        2.84   39
## 2 Kempen    PG          106.        3.61   33
## 3 Kempen    SM          84.2        3.07   39
## 4 Kempen    SMGR         90.3        3.41   40
## 5 Leemstreek GM          96.8        3.59   59
## 6 Leemstreek PG          119.        3.29   49
## 7 Leemstreek SM          97.9        4.18   76
## 8 Leemstreek SMGR         104.        3.59   42
```

Boxplot of Soil Carbon Stocks by Crop and Region

```
counts <- clean_data %>%
  group_by(Crop, Region) %>%
  summarise(count = n())
```

'summarise()' has grouped output by 'Crop'. You can override using the
'.groups' argument.

```
ggplot(clean_data, aes(x = Crop, y = Cstock, fill = Region)) +
  geom_boxplot(width = 0.7) +
  geom_jitter(aes(color = Region), width = 0.2, size = 2, alpha = 0.6) +
  geom_text(data = counts, aes(x = Crop, y = max(clean_data$Cstock) + 5, label = paste("n =", count)),
    position = position_dodge(width = 0.75),
    size = 3.5, # Adjusted text size
    vjust = -0.5) + # Adjust vertical position of labels, color = "black") +
  theme_minimal() +
  theme(
    plot.title = element_text(size = 16, face = "bold"),
    axis.title = element_text(size = 12),
    axis.text = element_text(size = 10),
    legend.title = element_text(size = 12),
    legend.text = element_text(size = 10)
  ) +
  labs(title = "Soil Carbon Stocks by Crop and Region",
    x = "Crop Type",
    y = "Carbon Stock (Mg C/ha)") +
  # Add some padding to the top of the plot for labels
  scale_y_continuous(expand = expansion(mult = c(0.05, 0.15)))
```



```
# Boxplot of Soil Carbon Stocks by Region and Crop
counts <- clean_data %>%
  group_by(Region, Crop) %>%
  summarise(count = n())
```

```
## 'summarise()' has grouped output by 'Region'. You can override using the
## '.groups' argument.
```

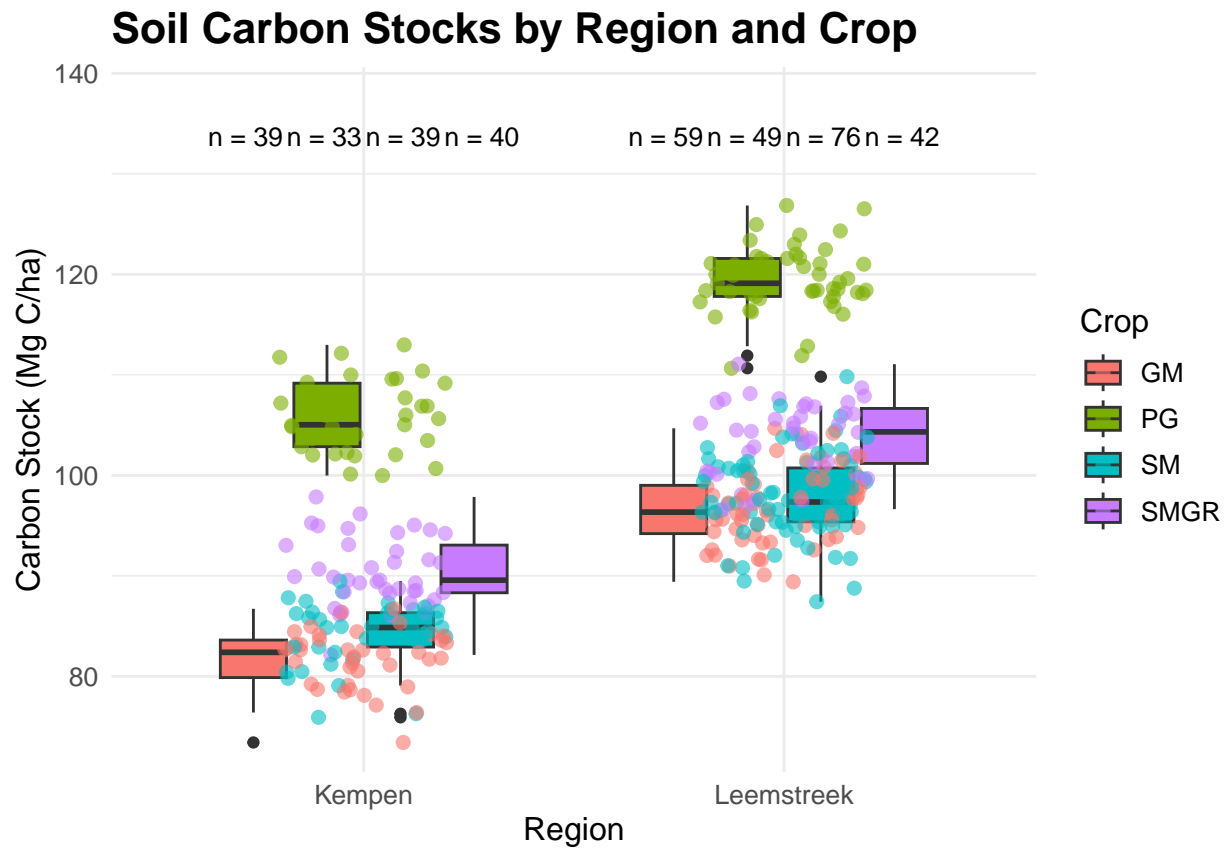
```
ggplot(clean_data, aes(x = Region, y = Cstock, fill = Crop)) +
  geom_boxplot(width = 0.7) +
  geom_jitter(aes(color = Crop), width = 0.2, size = 2, alpha = 0.6) +
  geom_text(
    data = counts,
    aes(x = Region, y = max(clean_data$Cstock) + 5, label = paste("n =", count)),
    position = position_dodge(width = 0.75),
    size = 3.5, # Adjusted text size
    vjust = -0.5
  ) +
  theme_minimal() +
  theme(
    plot.title = element_text(size = 16, face = "bold"),
    axis.title = element_text(size = 12),
    axis.text = element_text(size = 10),
    legend.title = element_text(size = 12),
    legend.text = element_text(size = 10)
```



```

) +
labs(
  title = "Soil Carbon Stocks by Region and Crop",
  x = "Region",
  y = "Carbon Stock (Mg C/ha)"
) +
# Add some padding to the top of the plot for labels
scale_y_continuous(expand = expansion(mult = c(0.05, 0.15)))

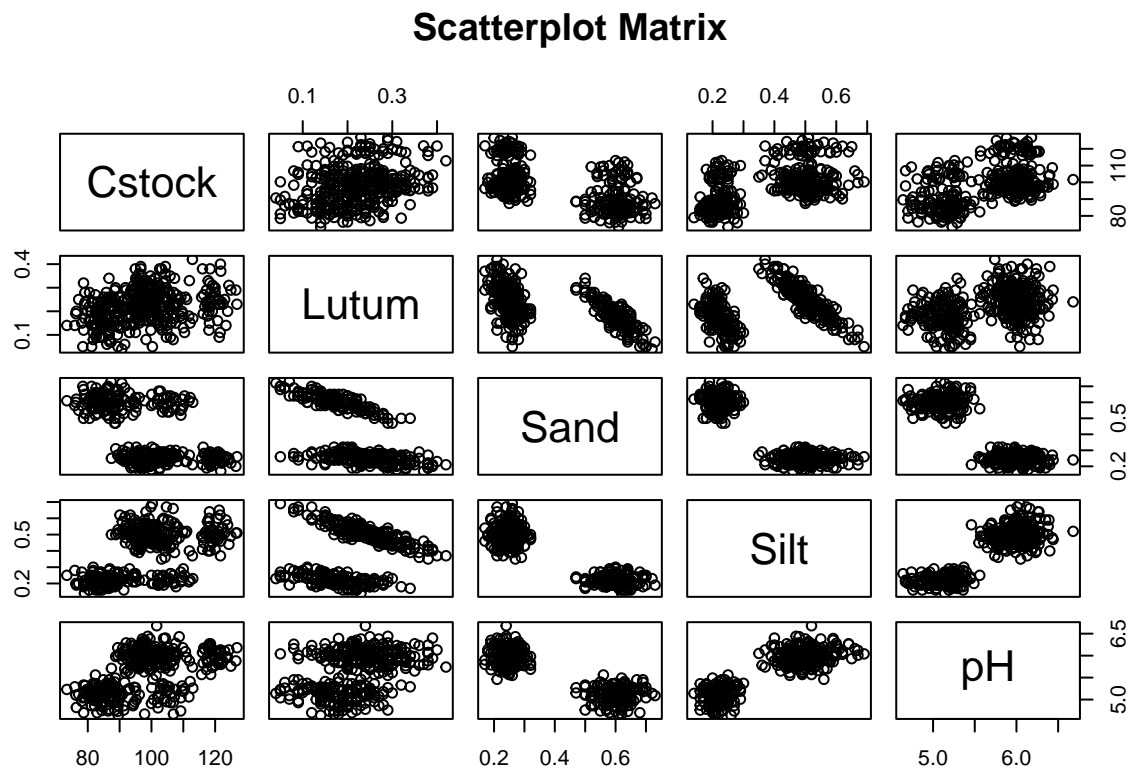
```



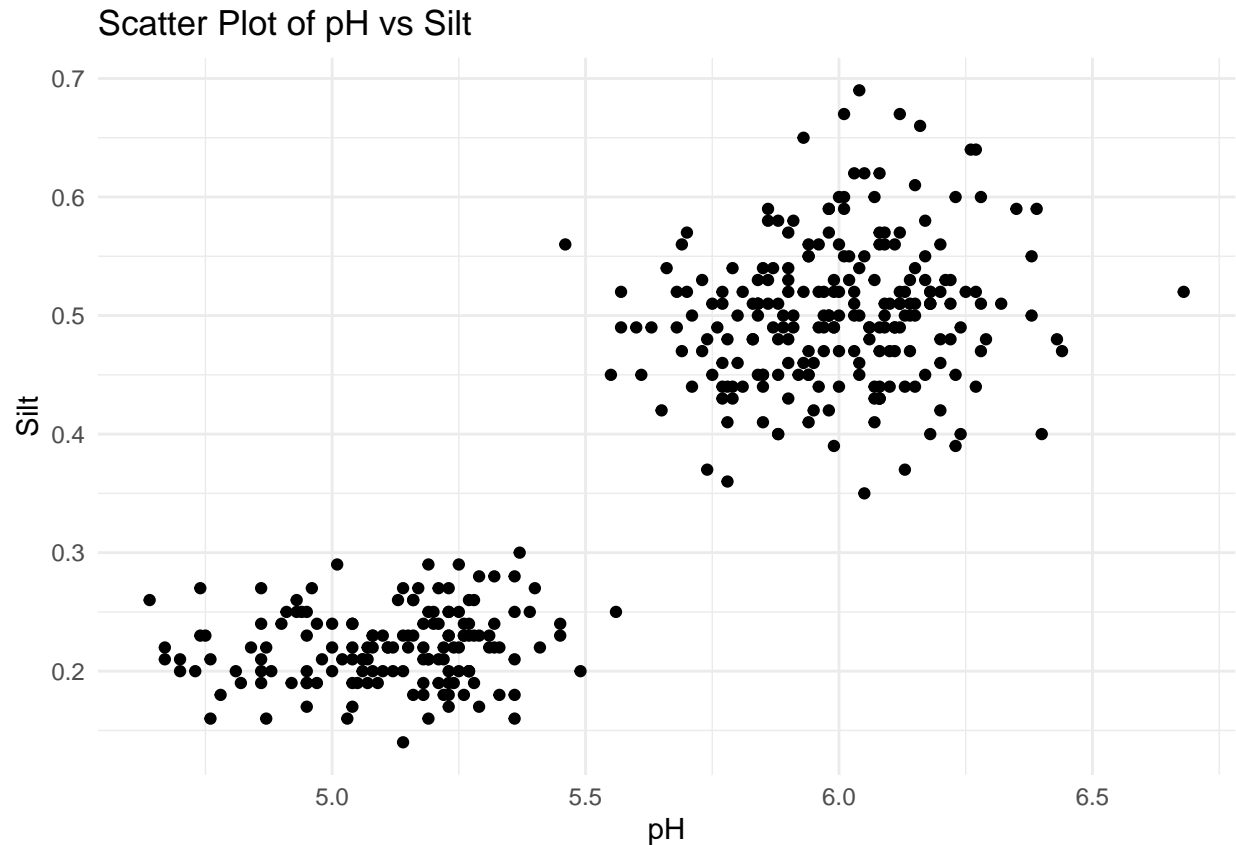
```

# Scatter plot matrix for continuous variables
pairs(clean_data[, c("Cstock", "Lutum", "Sand", "Silt", "pH")],
      main = "Scatterplot Matrix")

```



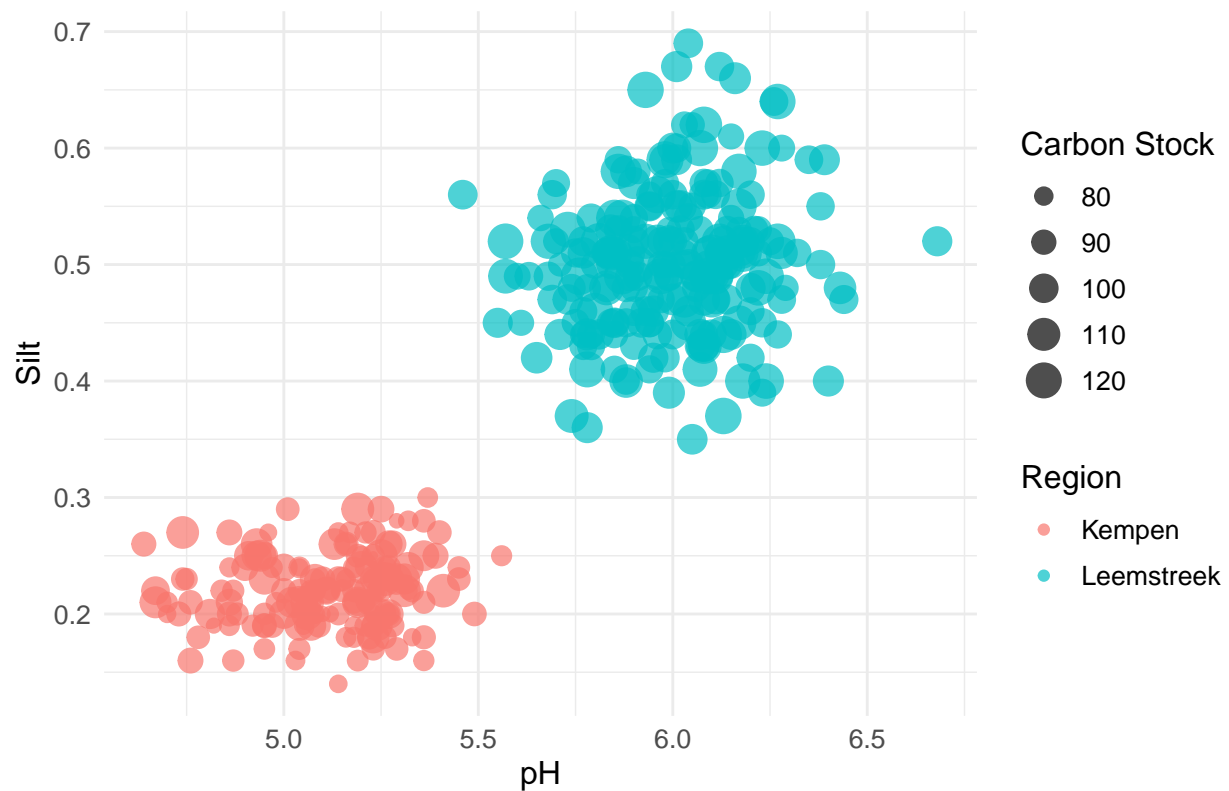
```
ggplot(clean_data, aes(x = pH, y = Silt)) +
  geom_point() +
  labs(title = "Scatter Plot of pH vs Silt",
        x = "pH",
        y = "Silt") +
  theme_minimal()
```



```
library(ggplot2)

ggplot(clean_data, aes(x = pH, y = Silt, color = Region, size = Cstock)) +
  geom_point(alpha = 0.7) + # Adjust transparency with alpha
  labs(
    title = "Scatter Plot of pH vs Silt by Region",
    x = "pH",
    y = "Silt",
    color = "Region", # Legend title for color
    size = "Carbon Stock" # Legend title for size
  ) +
  theme_minimal() +
  theme(
    plot.title = element_text(size = 16, face = "bold"),
    axis.title = element_text(size = 12),
    axis.text = element_text(size = 10),
    legend.title = element_text(size = 12),
    legend.text = element_text(size = 10),
    legend.position = "right" # Place legend on the right side
  )
```

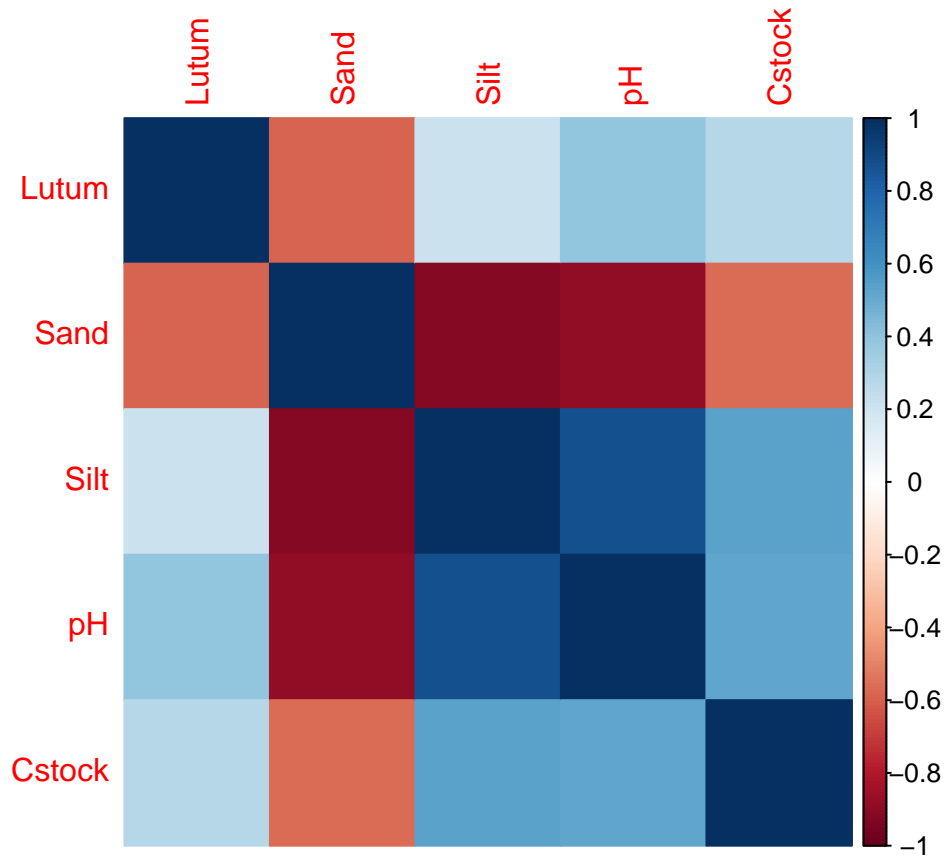
Scatter Plot of pH vs Silt by Region



```
# Correlation matrix
cor_matrix <- cor(clean_data[, c("Lutum", "Sand", "Silt", "pH", "Cstock")])
print(cor_matrix)
```

```
##          Lutum      Sand      Silt      pH      Cstock
## Lutum    1.0000000 -0.5898419  0.2188901  0.3985912  0.2812626
## Sand     -0.5898419  1.0000000 -0.9165763 -0.8888334 -0.5602512
## Silt      0.2188901 -0.9165763  1.0000000  0.8780938  0.5390980
## pH        0.3985912 -0.8888334  0.8780938  1.0000000  0.5219731
## Cstock    0.2812626 -0.5602512  0.5390980  0.5219731  1.0000000
```

```
# Visualize correlations
corrplot(cor_matrix, method = "color")
```



```
install.packages("reshape2")
```

```
## Installing package into 'C:/Users/saleh/AppData/Local/R/win-library/4.4'
## (as 'lib' is unspecified)
```

```
## package 'reshape2' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
## C:\Users\saleh\AppData\Local\Temp\RtmpK0zTsG\downloaded_packages
```

```
library(ggplot2)
library(reshape2)
```

```
## Warning: package 'reshape2' was built under R version 4.4.2
```

```
# Compute the correlation matrix
cor_matrix <- cor(clean_data[, c("Lutum", "Sand", "Silt", "pH", "Cstock")])

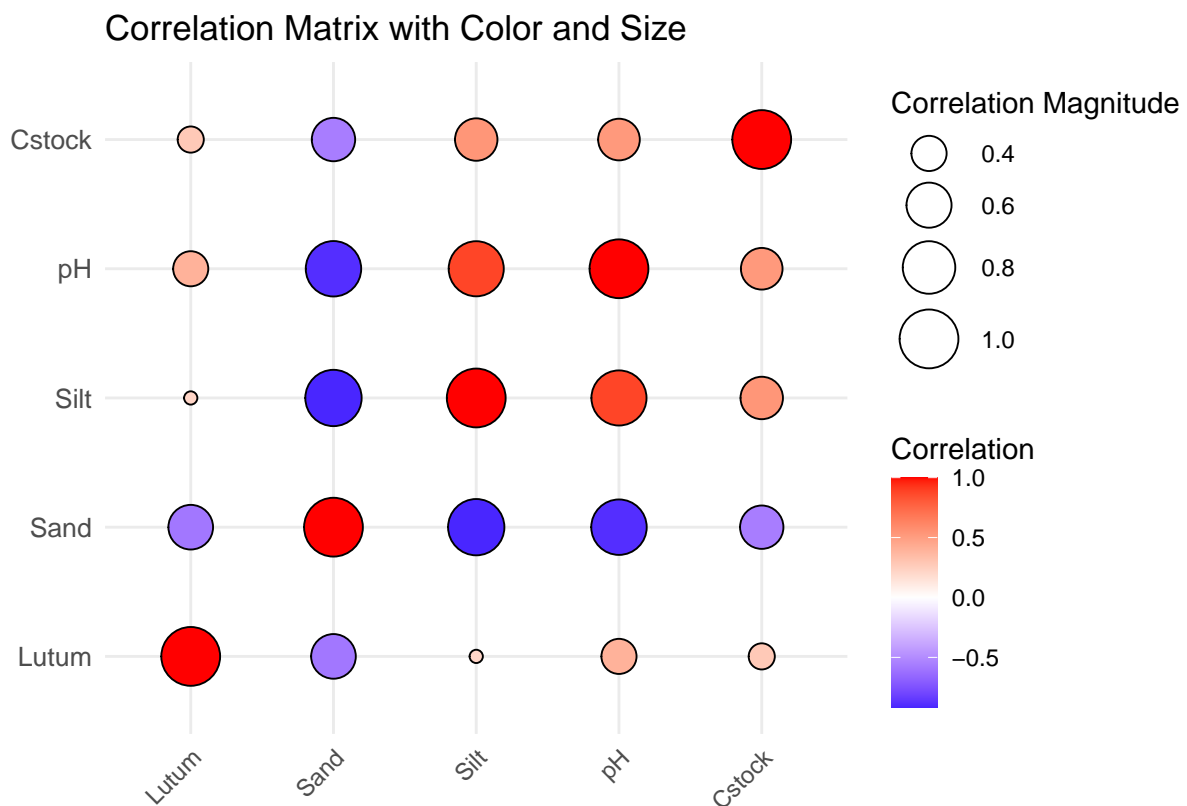
# Melt the correlation matrix into a long format
cor_melted <- melt(cor_matrix)

# Create the heatmap with size and color
ggplot(cor_melted, aes(x = Var1, y = Var2, fill = value, size = abs(value))) +
  geom_point(shape = 21, color = "black") + # Use shape 21 for circles with borders
```

```

scale_fill_gradient2(
  low = "blue", mid = "white", high = "red", midpoint = 0,
  name = "Correlation"
) +
scale_size_continuous(range = c(2, 10), name = "Correlation Magnitude") +
labs(
  title = "Correlation Matrix with Color and Size",
  x = "",
  y = ""
) +
theme_minimal() +
theme(
  axis.text.x = element_text(angle = 45, hjust = 1, vjust = 1),
  axis.text.y = element_text(size = 10),
  legend.position = "right"
)

```



```

# Create interaction plot for Carbon Stock
ggplot_interaction_cstock <- clean_data %>%
  group_by(Region, Crop) %>%
  summarise(
    mean_Cstock = mean(Cstock, na.rm = TRUE),
    se_Cstock = sd(Cstock, na.rm = TRUE) / sqrt(n()),
    .groups = "drop"
  ) %>%

```

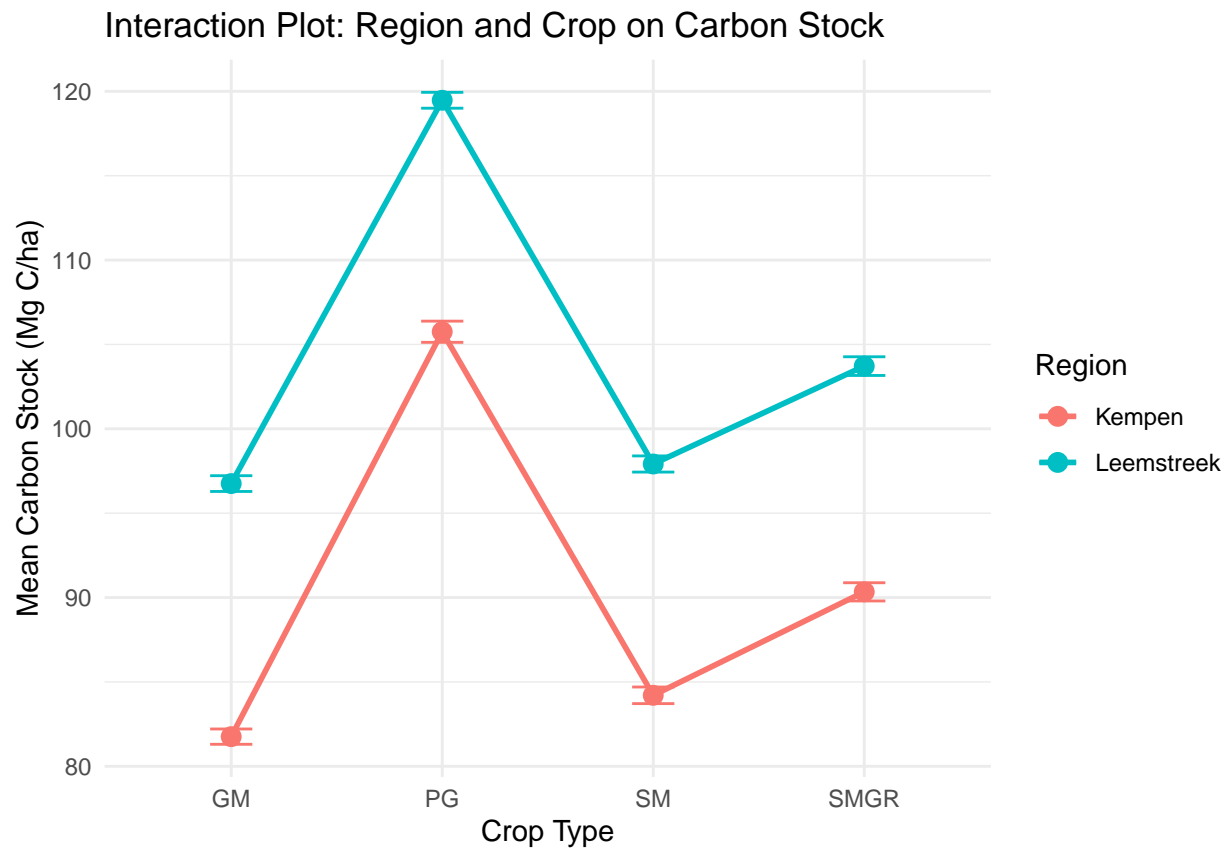
```

ggplot(aes(x = Crop, y = mean_Cstock, color = Region, group = Region)) +
  geom_line(linewidth = 1) +
  geom_point(size = 3) +
  geom_errorbar(aes(ymin = mean_Cstock - se_Cstock,
                    ymax = mean_Cstock + se_Cstock),
                width = 0.2) +
  theme_minimal() +
  labs(
    title = "Interaction Plot: Region and Crop on Carbon Stock",
    x = "Crop Type",
    y = "Mean Carbon Stock (Mg C/ha)"
  )

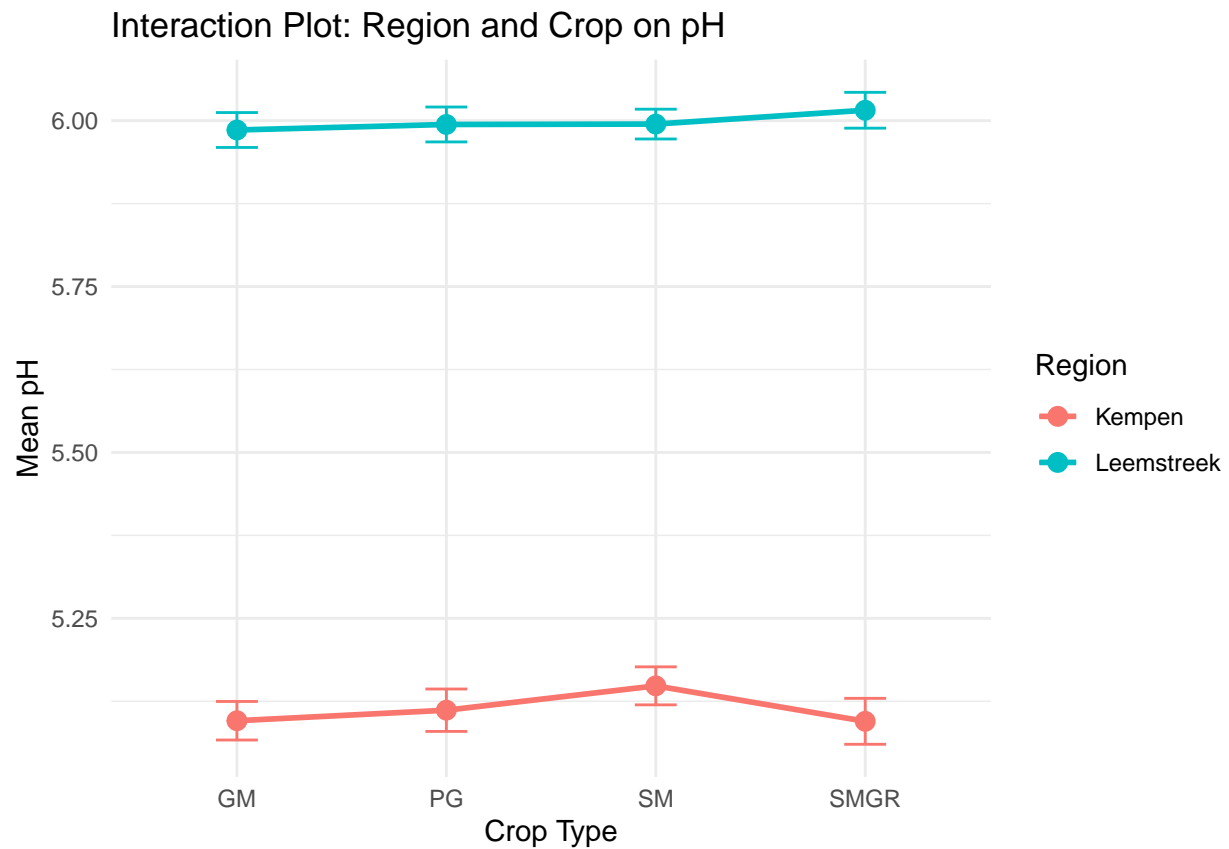
# Create interaction plot for pH
ggplot_interaction_ph <- clean_data %>%
  group_by(Region, Crop) %>%
  summarise(
    mean_pH = mean(pH, na.rm = TRUE),
    se_pH = sd(pH, na.rm = TRUE) / sqrt(n()),
    .groups = "drop"
  ) %>%
  ggplot(aes(x = Crop, y = mean_pH, color = Region, group = Region)) +
  geom_line(linewidth = 1) +
  geom_point(size = 3) +
  geom_errorbar(aes(ymin = mean_pH - se_pH,
                    ymax = mean_pH + se_pH),
                width = 0.2) +
  theme_minimal() +
  labs(
    title = "Interaction Plot: Region and Crop on pH",
    x = "Crop Type",
    y = "Mean pH"
  )

# Print and save plots
print(ggplot_interaction_cstock)

```

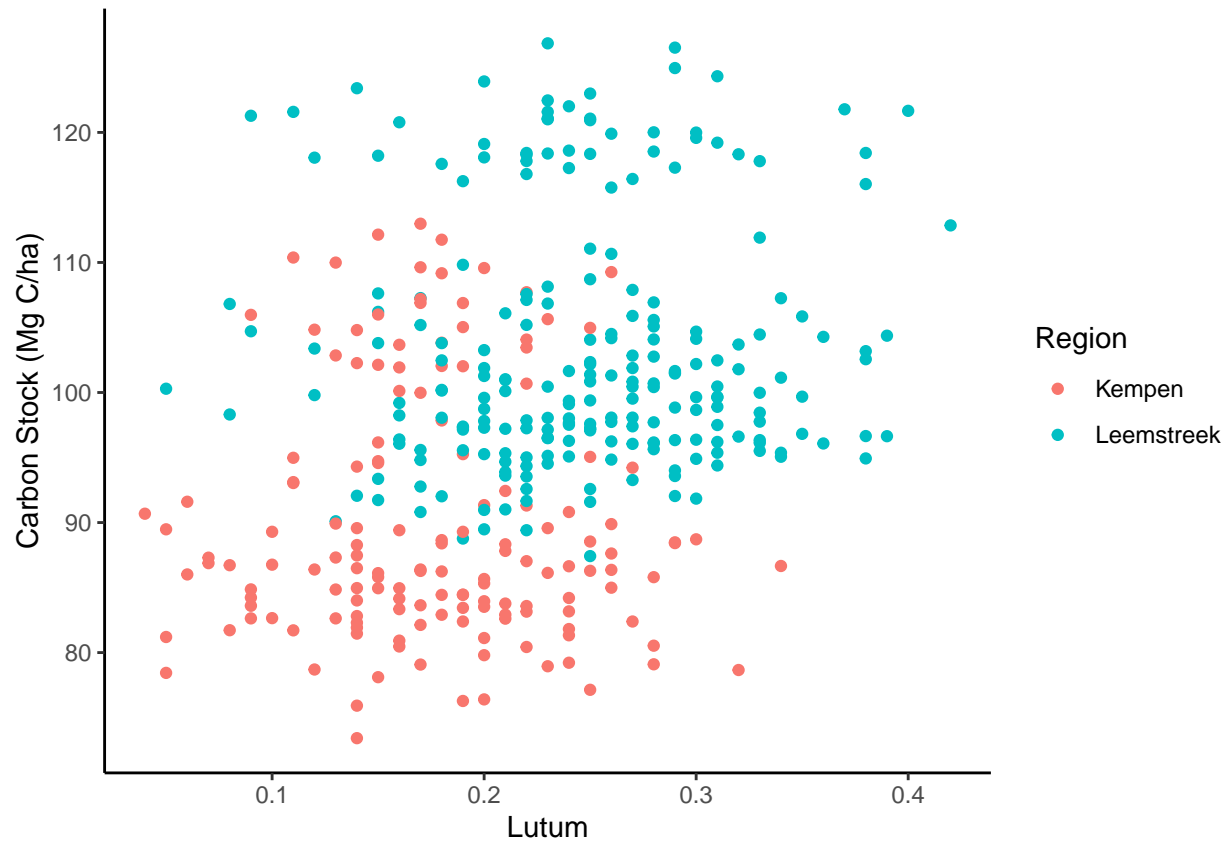


```
print(ggplot_interaction_ph)
```

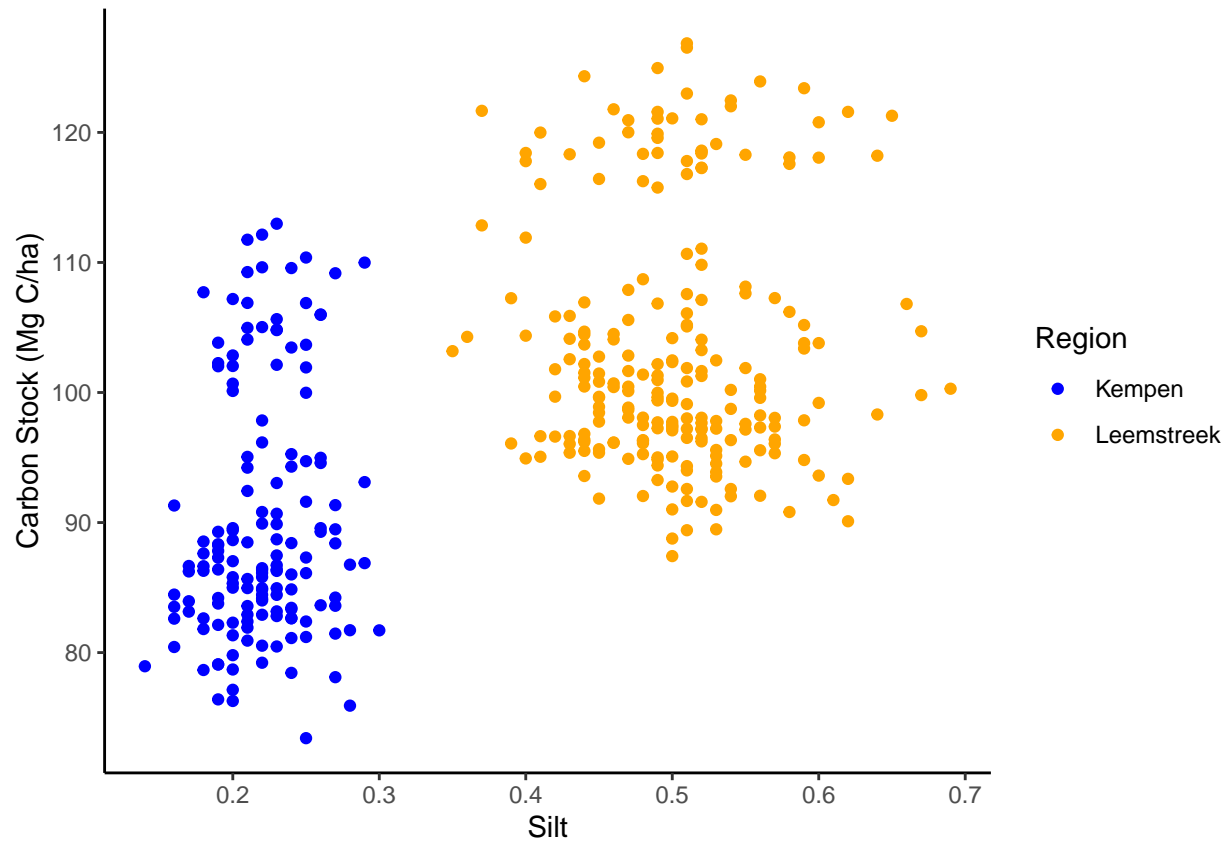



```
# Save ggplot versions
ggsave("interaction_plot_cstock.png", ggplot_interaction_cstock, width = 10, height = 6)
ggsave("interaction_plot_ph.png", ggplot_interaction_ph, width = 10, height = 6)
```

```
ggplot(clean_data, aes(x = Lutum, y = Cstock, color = Region)) +
  geom_point() +
  labs(x = "Lutum", y = "Carbon Stock (Mg C/ha)", color = "Region") +
  theme_classic()
```



```
ggplot(clean_data, aes(x = Silt, y = Cstock, color = Region)) +
  geom_point() +
  scale_color_manual(values = c("Kempen" = "blue", "Leemstreek" = "orange", "other_region" = "green")) +
  labs(x = "Silt", y = "Carbon Stock (Mg C/ha)", color = "Region") +
  theme_classic()
```



```
library(dagitty)
```

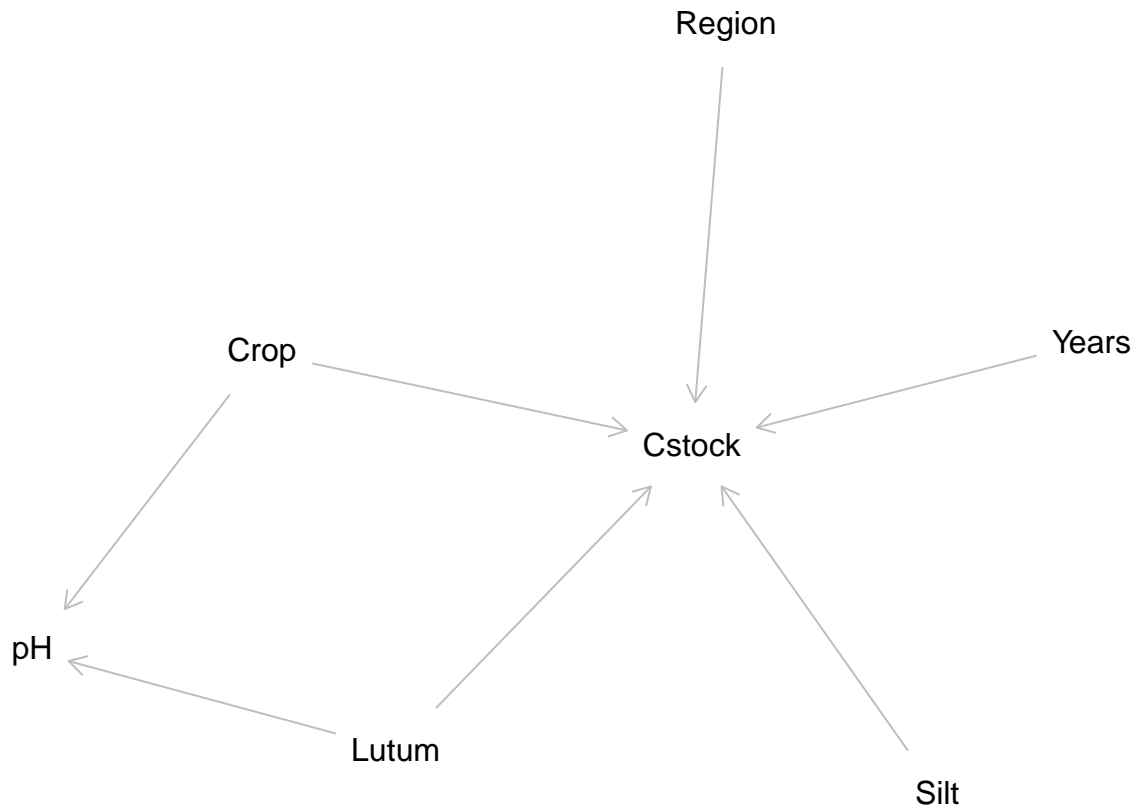
```
## Warning: package 'dagitty' was built under R version 4.4.2
```

```
# Define the causal structure
dag <- dagitty('
dag {
  "Region" -> "Cstock"
  "Years" -> "Cstock"
  "Crop" -> "Cstock"
  "Lutum" -> "Cstock"
  "Silt" -> "Cstock"
  "Crop" -> "pH"
  "Lutum" -> "pH"
}
')
```

Plot the DAG

```
plot(dag)
```

```
## Plot coordinates for graph not supplied! Generating coordinates, see ?coordinates for how to set your
```



```
library(ggdag)
```

```
## Warning: package 'ggdag' was built under R version 4.4.2
```

```
##
```

```
## Attaching package: 'ggdag'
```

```
## The following object is masked from 'package:stats':
```

```
##
```

```
## filter
```

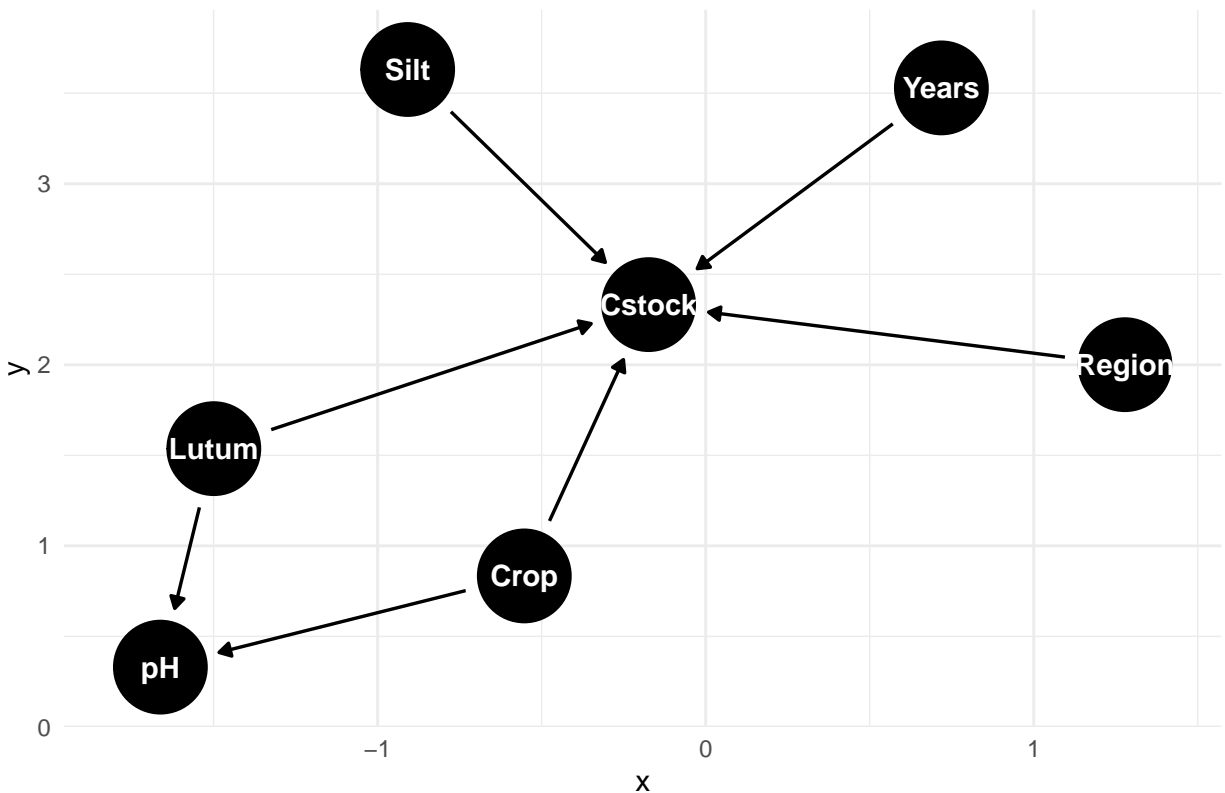
```
# Define the DAG
```

```
dag <- dagify(  
  Cstock ~ Region + Years + Crop + Lutum + Silt,  
  pH ~ Crop + Lutum,  
  exposure = "Crop",  
  outcome = "Cstock"  
)
```

```
# Plot the DAG
```

```
ggdag(dag) +  
  theme_minimal() +  
  ggtitle("Causal Graph for Cstock Analysis")
```

Causal Graph for Cstock Analysis



```
ggsave("causal_graph.png", width = 8, height = 6)
```

```
library(DiagrammeR)
```

```
## Warning: package 'DiagrammeR' was built under R version 4.4.2
```

```
library(DiagrammeRsvg)
```

```
## Warning: package 'DiagrammeRsvg' was built under R version 4.4.2
```

```
library(rsvg)
```

```
## Warning: package 'rsvg' was built under R version 4.4.2
```

```
## Linking to librsvg 2.57.0
```

```

# Define the flow diagram
graph <- grViz("
digraph flowchart {
  graph [layout = dot, rankdir = TB]

  node [shape = rectangle, fontname = Arial]

```

```

# Nodes
Start [label = 'Start']
SelectVars [label = 'Selecting variables\nwith a Causal graph', shape = diamond]
VarType [label = 'Categorical/Continuous Variables']
Explore [label = 'Exploration of\ndata']
MissingValues [label = 'Address Missing\nValues']
FitModel [label = 'Fitting Multiple\nLinear Regression Model']
Assumptions [label = 'Model assumptions\nare violated?', shape = diamond]
Remedial [label = 'Take remedial\nmeasures']
Hypothesis [label = 'Hypothesis Testing']
Conclusion [label = 'Conclusion']
Stop [label = 'Stop']

# Edges
Start -> SelectVars
SelectVars -> Explore
Explore -> VarType
Explore -> FitModel
Explore -> MissingValues
FitModel -> Assumptions
Assumptions -> Remedial [label = 'Yes']
Assumptions -> Hypothesis [label = 'No']
Remedial -> FitModel
Hypothesis -> Conclusion
Conclusion -> Stop

# Subgraphs for horizontal alignment
{ rank = same; VarType; MissingValues }
}
")

# Render the flow diagram
graph

# Render the diagram
svg_code <- export_svg(graph)

# Save as PNG file
rsvg_png(charToRaw(svg_code), "flow_diagram.png")

```