Lab 1

Keara Dreyfuss & Isabella Juara

2024-01-23

1. Data Inspection: • Load the dataset into R and use 2 or more base R functions to inspect the data. • Optional: use the names() function. For example: names(data)=c("City", "Rank", "Sun", "WaterCost",...)

```
df = read.csv("HealthyCities.csv")
names(df)=c("City","Rank","SunHours","Watercost","Obesity","LifeExp","Pollution","Worked","Happiness",
names(df)
```

```
##  [1] "City"      "Rank"      "SunHours"  "Watercost" "Obesity"   "LifeExp"
##  [7] "Pollution" "Worked"    "Happiness" "Outdoor"   "Takeouts"  "GymCost"
```

```
str(df)
```

```
## 'data.frame':    44 obs. of  12 variables:
##  $ City     : chr  "Amsterdam" "Sydney" "Vienna" "Stockholm" ...
##  $ Rank     : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ SunHours : chr  "1858" "2636" "1884" "1821" ...
##  $ Watercost: num  1.92 1.48 1.94 1.72 2.19 1.6 0.78 1.55 1.19 1.08 ...
##  $ Obesity  : chr  "20.40%" "29.00%" "20.10%" "20.60%" ...
##  $ LifeExp  : num  81.2 82.1 81 81.8 79.8 80.4 83.2 80.6 82.2 81.7 ...
##  $ Pollution: chr  "30.93" "26.86" "17.33" "19.63" ...
##  $ Worked   : chr  "1434" "1712" "1501" "1452" ...
##  $ Happiness: num  7.44 7.22 7.29 7.35 7.64 7.8 5.87 7.07 6.4 7.23 ...
##  $ Outdoor  : int  422 406 132 129 154 113 35 254 585 218 ...
##  $ Takeouts : int  1048 1103 1008 598 523 309 539 1729 2344 788 ...
##  $ GymCost  : num  34.9 41.7 25.7 37.3 32.5 ...
```

```
summary(df)
```

```
##      City                Rank         SunHours           Watercost
##  Length:44          Min.   : 1.00   Length:44          Min.   :0.150
##  Class :character   1st Qu.:11.75   Class :character   1st Qu.:0.570
##  Mode  :character   Median :22.50   Mode  :character   Median :1.195
##                     Mean   :22.50                      Mean   :1.173
##                     3rd Qu.:33.25                      3rd Qu.:1.600
##                     Max.   :44.00                      Max.   :3.200
##    Obesity             LifeExp       Pollution           Worked
##  Length:44          Min.   :56.30   Length:44          Length:44
##  Class :character   1st Qu.:75.40   Class :character   Class :character
##  Mode  :character   Median :80.40   Mode  :character   Mode  :character
##                     Mean   :78.17
##                     3rd Qu.:81.80
##                     Max.   :83.20
##    Happiness        Outdoor         Takeouts       GymCost
##  Min.   :3.570   Min.   : 23.0   Min.   : 250   Min.   :16.07
```

```
##  1st Qu.:5.870   1st Qu.:125.2   1st Qu.: 548   1st Qu.:31.31
##  Median :6.900   Median :189.5   Median : 998   Median :37.33
##  Mean   :6.435   Mean   :214.0   Mean   :1443   Mean   :40.42
##  3rd Qu.:7.175   3rd Qu.:288.2   3rd Qu.:1674   3rd Qu.:47.21
##  Max.   :7.800   Max.   :585.0   Max.   :6417   Max.   :73.11
```

2. Descriptive Statistics: • Calculate basic descriptive statistics (mean, median, standard deviation, etc.) for each numeric variable in the dataset.

```
mean(df$Watercost)
```

```
## [1] 1.173409
```

```
median(df$Watercost)
```

```
## [1] 1.195
```

```
sd(df$Watercost)
```

```
## [1] 0.7186419
```

```
mean(df$LifeExp)
```

```
## [1] 78.175
```

```
median(df$LifeExp)
```

```
## [1] 80.4
```

```
sd(df$LifeExp)
```

```
## [1] 5.30437
```

```
mean(df$Happiness)
```

```
## [1] 6.435
```

```
median(df$Happiness)
```

```
## [1] 6.9
```

```
sd(df$Happiness)
```

```
## [1] 0.991202
```

```
mean(df$GymCost)
```

```
## [1] 40.42
```

```
median(df$GymCost)
```

```
## [1] 37.33
```

```
sd(df$GymCost)
```

```
## [1] 15.00646
```

```
mean(df$Outdoor)
```

```
## [1] 213.9773
```

```
median(df$Outdoor)
```

```
## [1] 189.5
```

```r
sd(df$Outdoor)
```

```
## [1] 127.1903
```

```r
mean(df$Takeouts)
```

```
## [1] 1443.114
```

```r
median(df$Takeouts)
```

```
## [1] 998
```

```r
sd(df$Takeouts)
```

```
## [1] 1388.803
```

3. Create a Numeric DataFrame: • Extract all numeric variables from the dataset and create a new DataFrame containing only these variables. • Do this twice, once as we did in class and again using ChatGPT.

```r
numeric_df = df[,c(4,6,9,12)]
```

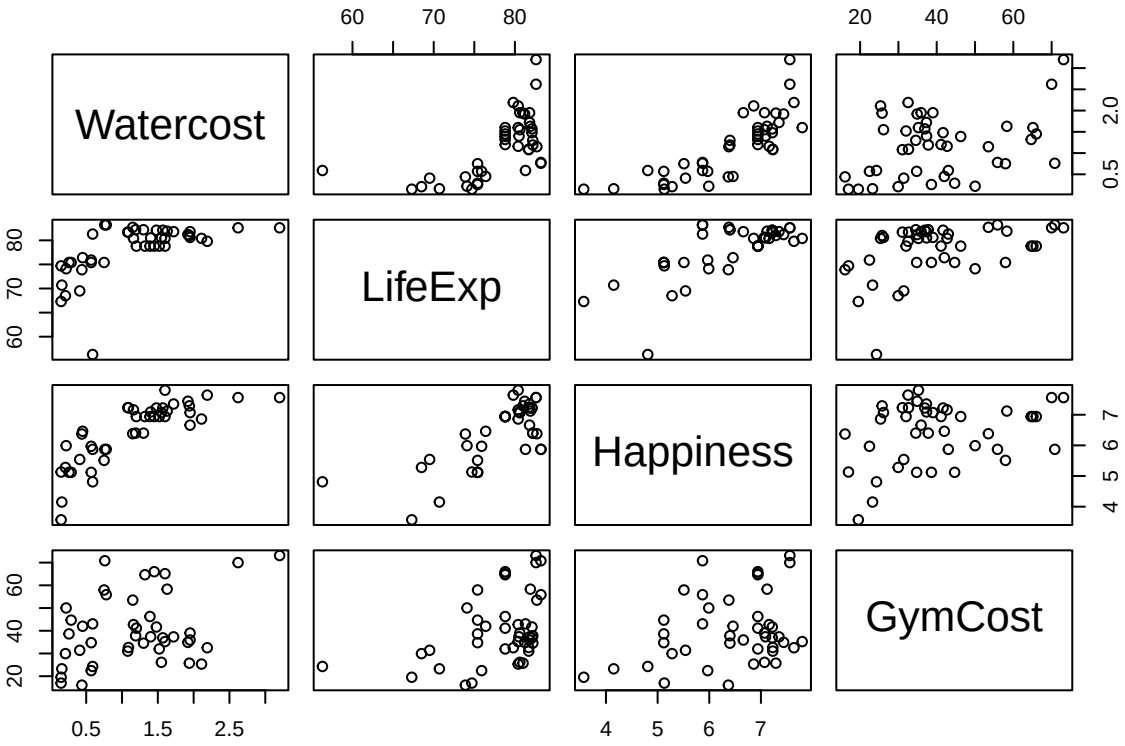
```r
# Extract columns 4, 6, 9, and 12
selected_columns <- df[, c(4, 6, 9, 12)]

# Convert the selected columns to numeric
numeric_df2 <- as.data.frame(sapply(selected_columns, as.numeric))

# Print or use the new numeric data frame
print(numeric_df2)
```

```
##    Watercost LifeExp Happiness GymCost
## 1       1.92    81.2      7.44   34.90
## 2       1.48    82.1      7.22   41.66
## 3       1.94    81.0      7.29   25.74
## 4       1.72    81.8      7.35   37.31
## 5       2.19    79.8      7.64   32.53
## 6       1.60    80.4      7.80   35.23
## 7       0.78    83.2      5.87   55.87
## 8       1.55    80.6      7.07   26.11
## 9       1.19    82.2      6.40   37.80
## 10      1.08    81.7      7.23   31.04
## 11      1.57    82.1      7.22   36.89
## 12      0.26    75.4      5.12   38.62
## 13      0.22    74.1      5.99   50.03
## 14      0.57    75.9      5.97   22.45
## 15      1.09    81.7      7.23   32.64
## 16      1.30    82.2      6.40   34.54
## 17      0.21    68.5      5.28   29.94
## 18      0.59    81.3      5.87   43.03
## 19      1.95    80.6      7.07   39.01
## 20      2.62    82.6      7.56   70.00
## 21      1.63    81.9      7.12   58.31
## 22      0.15    74.7      5.13   16.97
## 23      0.16    70.7      4.15   23.25
## 24      0.57    75.4      5.12   34.76
## 25      1.52    78.8      6.94   32.00
## 26      0.15    67.3      3.57   19.54
```

```
## 27      1.39    78.8      6.94   46.27
## 28      1.40    80.5      7.09   37.35
## 29      0.76    83.2      5.87   70.82
## 30      1.20    78.8      6.94   41.14
## 31      0.75    75.4      5.51   57.95
## 32      0.29    75.4      5.12   44.68
## 33      2.11    80.4      6.86   25.34
## 34      1.60    78.8      6.94   65.13
## 35      1.95    81.8      6.66   35.93
## 36      0.44    73.9      6.37   16.07
## 37      3.20    82.6      7.56   73.11
## 38      1.16    80.4      7.16   42.71
## 39      0.59    56.3      4.81   24.28
## 40      1.15    82.7      6.38   53.49
## 41      1.45    78.8      6.94   65.99
## 42      1.32    78.8      6.94   64.66
## 43      0.41    69.5      5.54   31.40
## 44      0.45    76.4      6.46   41.99
```

4. Scatterplot Matrix: • Create scatterplots for all possible pairs of numeric variables using base R.

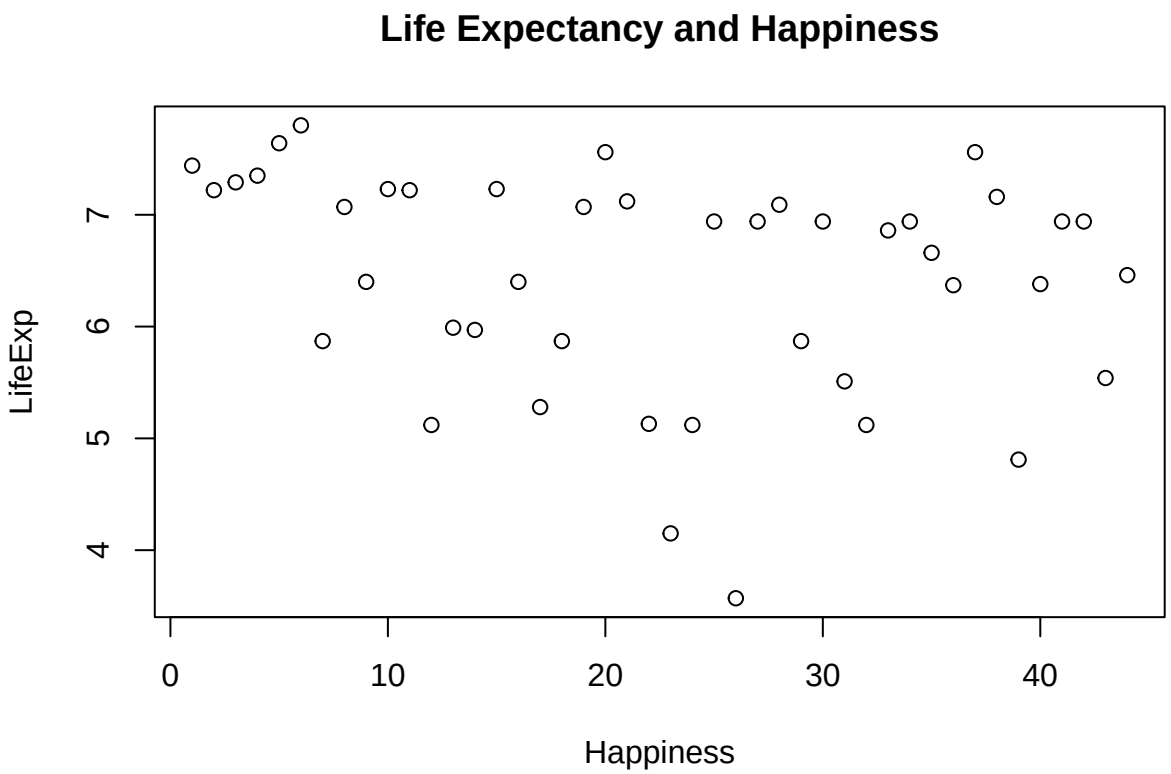
```
plot(numeric_df)
```



5. Scatterplot of Interest: • Which of the scatterplots has the most interesting relationship? Plot the individual scatterplot and carefully label it.

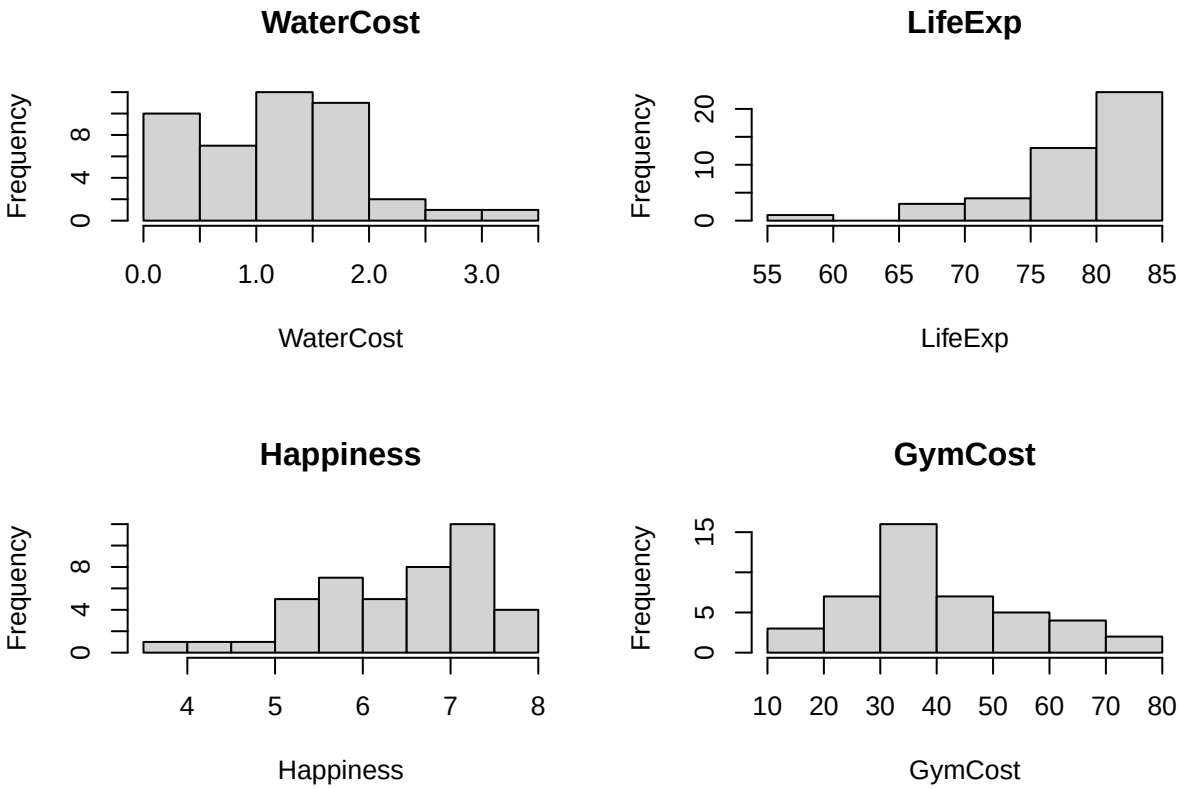
Happiness and Life Expectancy

```
plot(df$Happiness, df$LifeExpectancy, xlab = "Happiness",
     ylab="LifeExp",
     main = "Life Expectancy and Happiness")
```



6. Histograms: • Create a 2 by 2 set of histograms of 4 numeric variables (your choice). Carefully label the histograms. • Do this twice, once as we did in class and again using ChatGPT.

```
par(mfrow=c(2,2))
hist(df$Watercost, xlab = "WaterCost", main = "WaterCost")
hist(df$LifeExp, xlab = "LifeExp", main = "LifeExp")
hist(df$Happiness, xlab = "Happiness", main = "Happiness")
hist(df$GymCost, xlab = "GymCost",
     main = "GymCost")
```



7. Correlation Analysis: • Calculate the correlation matrix for the numeric variables using the cor() function. • Interpret the correlations between different variables. Discuss any strong positive or negative correlations you find.

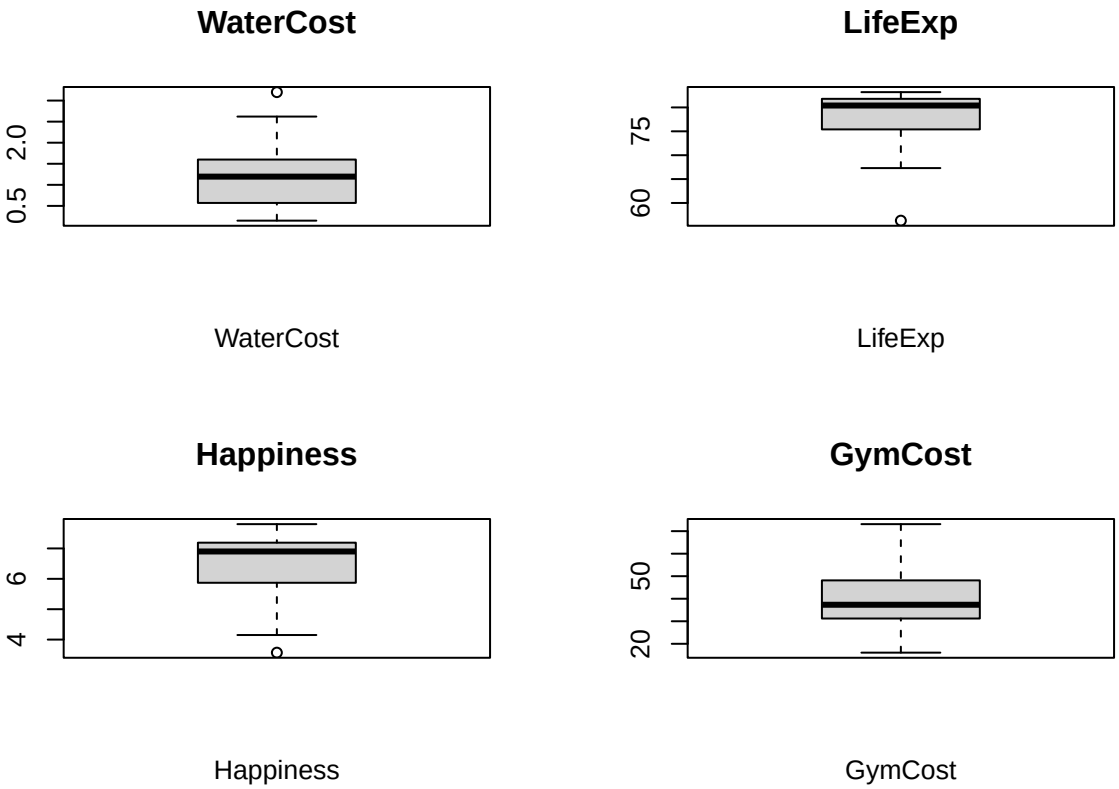
```
cor(numeric_df)
```

```
##           Watercost   LifeExp Happiness   GymCost
## Watercost 1.0000000 0.6123823 0.8131593 0.3564606
## LifeExp   0.6123823 1.0000000 0.7245871 0.4179858
## Happiness 0.8131593 0.7245871 1.0000000 0.2974250
## GymCost   0.3564606 0.4179858 0.2974250 1.0000000
```

Happiness and Water cost have a very high positive correlation, so as Water Cost goes up so does Happiness. Happiness and Gym cost have a low postiive correlation.

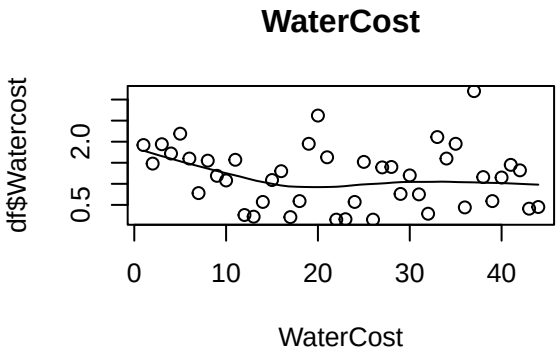
8. Boxplots Creation: • Create a 2 by 2 set of boxplots for 4 numeric variables (your choice) to visually inspect their distribution and identify any outliers.

```
par(mfrow=c(2,2))
boxplot(df$Watercost, xlab = "WaterCost", main = "WaterCost")
boxplot(df$LifeExp, xlab = "LifeExp", main = "LifeExp")
boxplot(df$Happiness, xlab = "Happiness", main = "Happiness")
boxplot(df$GymCost, xlab = "GymCost",
     main = "GymCost")
```



9. Other Graphs: • Create an exceptional plot using a method of your choice.

```
par(mfrow=c(2,2))
scatter.smooth(df$Watercost, xlab = "WaterCost", main = "WaterCost")
```



10. Insights: • List 3 important insights that you learned from this dataset.

Happiness and water cost are highly related. Happiness and Gym cost are not highly correlated. Life expectancy and Happiness are modertely correlated.