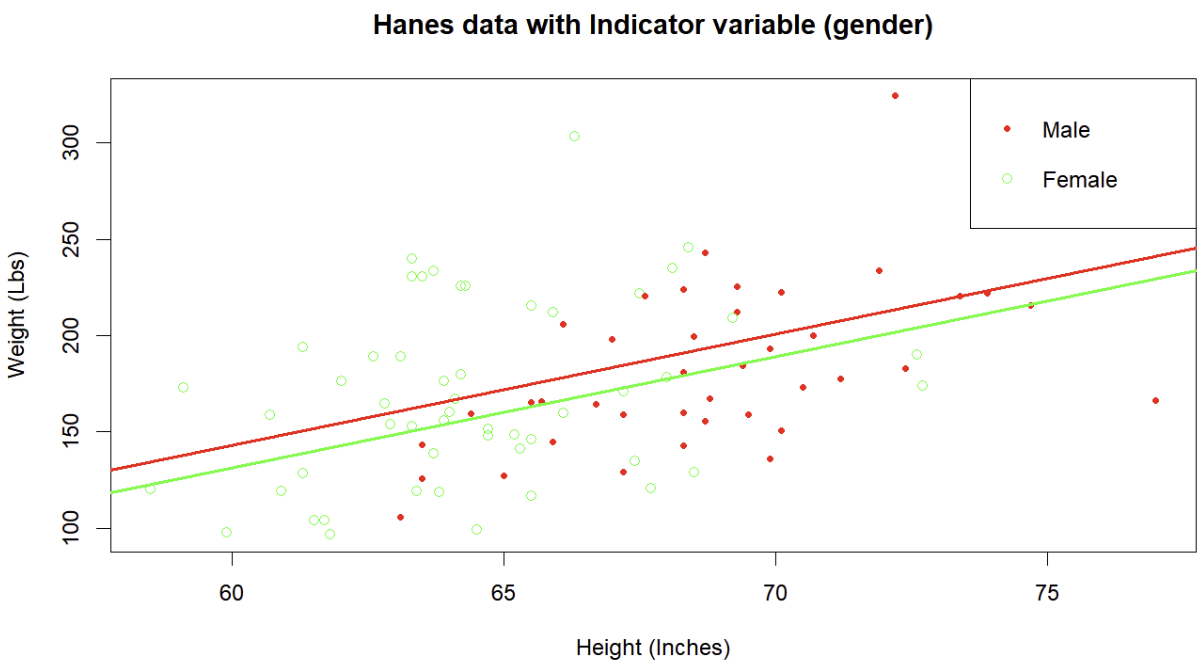
Advanced Statistics mini project 2: Scatter plot – Hanes Data

Submitted : Vishruth Acharya – vxa220058@utdallas.edu

1] The regression equation and plots from the R file on the Hanes data is illustrated bellow:




The regression equation for estimating the weight with the indicator variables height and gender is as follows:

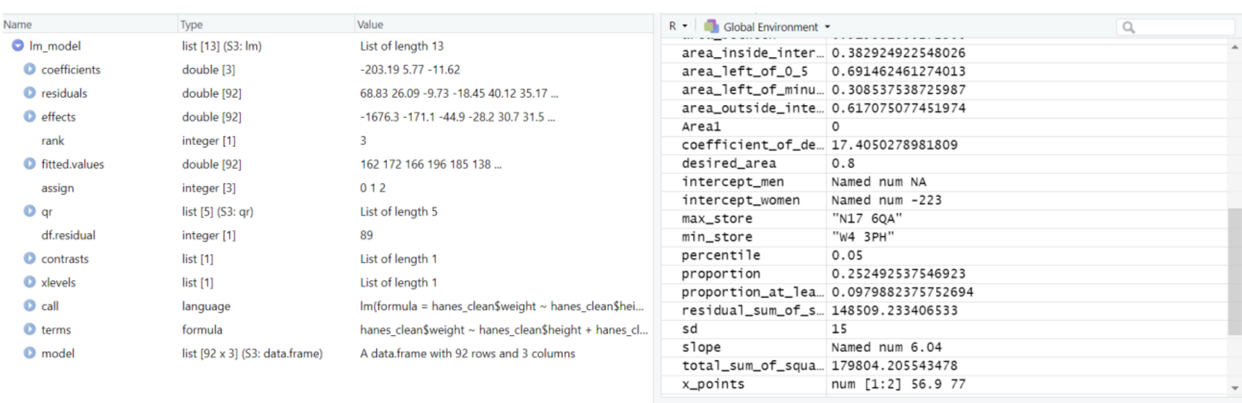
$Weight = -203.190 + 5.77 * (\boldsymbol{height}) - 11.65 * (\boldsymbol{gender})$

$Weight = -214.805 + 5.77 * (\boldsymbol{height}) \rightarrow (\boldsymbol{Male})$

$Weight = -203.190 + 5.77 * (\boldsymbol{height}) \rightarrow (\boldsymbol{Female})$

$Coefficient\ of\ Determination = 17.4\ \%$

Listed below are the values obtained in R.

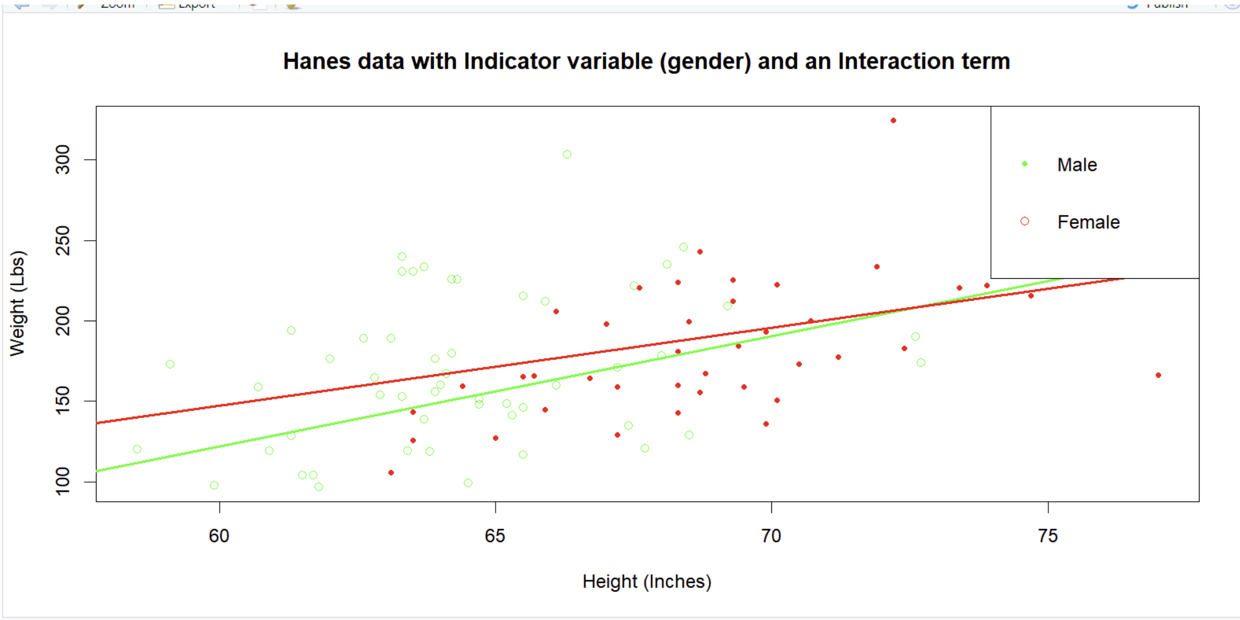


| Name | Type | Value |
|---|---|---|
| lm_model | list [13] (S3: lm) | List of length 13 |
| coefficients | double [3] | -203.19 5.77 -11.62 |
| residuals | double [92] | 68.83 26.09 -9.73 -18.45 40.12 35.17 ... |
| effects | double [92] | -1676.3 -171.1 -44.9 -28.2 30.7 31.5 ... |
| rank | integer [1] | 3 |
| fitted.values | double [92] | 162 172 166 196 185 138 ... |
| assign | integer [3] | 0 1 2 |
| qr | list [5] (S3: qr) | List of length 5 |
| df.residual | integer [1] | 89 |
| contrasts | list [1] | List of length 1 |
| xlevels | list [1] | List of length 1 |
| call | language | lm(formula = hanes_clean$weight ~ hanes_clean$hei... |
| terms | formula | hanes_clean$weight ~ hanes_clean$height + hanes_cl... |
| model | list [92 x 3] (S3: data.frame) | A data.frame with 92 rows and 3 columns |

R Global Environment

| | |
|---|---|
| area_inside_inter... | 0.382924922548026 |
| area_left_of_0_5 | 0.691462461274013 |
| area_left_of_minu... | 0.308537538725987 |
| area_outside_inte... | 0.617075077451974 |
| Area1 | 0 |
| coefficient_of_de... | 17.4050278981809 |
| desired_area | 0.8 |
| intercept_men | Named num NA |
| intercept_women | Named num -223 |
| max_store | "N17 6QA" |
| min_store | "W4 3PH" |
| percentile | 0.05 |
| proportion | 0.252492537546923 |
| proportion_at_lea... | 0.0979882375752694 |
| residual_sum_of_s... | 148509.233406533 |
| sd | 15 |
| slope | Named num 6.04 |
| total_sum_of_squa... | 179804.205543478 |
| x_points | num [1:2] 56.9 77 |

2] The regression equation and plots from the R file on the Hanes data is illustrated bellow:




Here is the regression equation for Target variable Weight and input variables height and gender.

$Weight = -143.998 + 4.854 * (\boldsymbol{height}) - 145.143 * (\boldsymbol{gender}) + 1.998 * (\boldsymbol{height}) * (\boldsymbol{gender})$

$Weight = -289.141 + 6.852 * (\boldsymbol{height}) \rightarrow (\boldsymbol{Male})$

$Weight = -143.998 + 4.854 * (\boldsymbol{height}) \rightarrow (\boldsymbol{Female})$

$Coefficient\ of\ Determination$ = 17.85 %.

Here are the values from the R code:

| | | |
|---|---|---|
| coefficients | double [3] | -203.19 5.77 -11.62 |
| residuals | double [92] | 68.83 26.09 -9.73 -18.45 40.12 35.17 ... |
| effects | double [92] | -1676.3 -171.1 -44.9 -28.2 30.7 31.5 ... |
| rank | integer [1] | 3 |
| fitted.values | double [92] | 162 172 166 196 185 138 ... |
| assign | integer [3] | 0 1 2 |

(No selection)

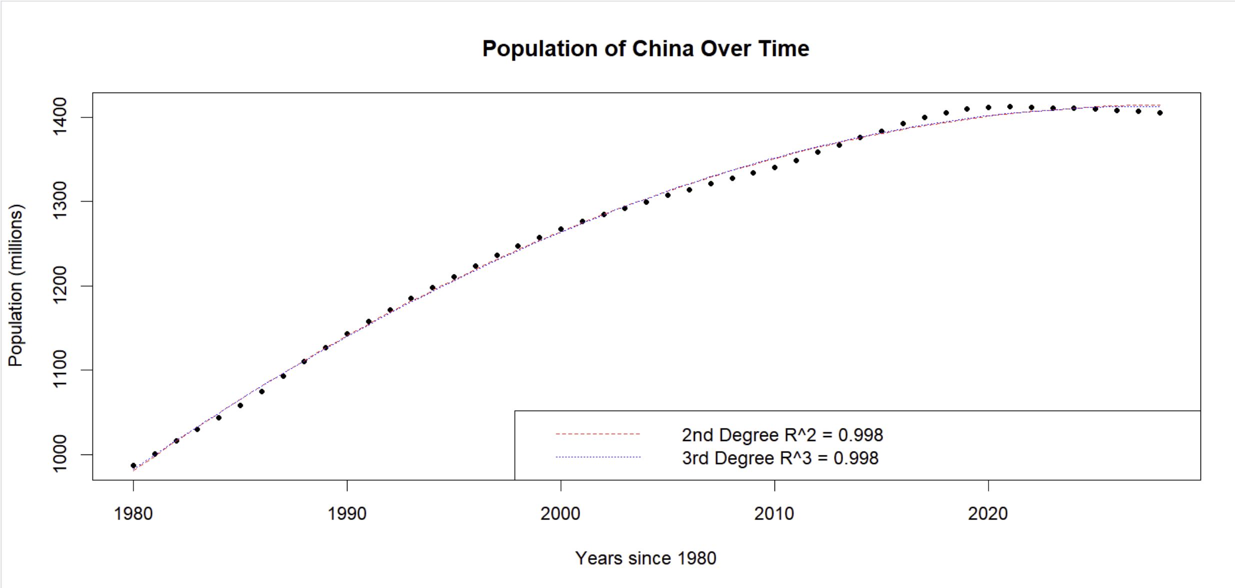
Console Background Jobs

R 4.3.1 · C:/Users/vishu/Downloads/

```
Call:
lm(formula = hanes_clean$weight ~ hanes_clean$height + hanes_clean$gender +
    hanes_clean$height:hanes_clean$gender)

Coefficients:
                          (Intercept)
                             -143.998
                   hanes_clean$height
                                4.854
                  hanes_clean$genderM
                             -145.143
hanes_clean$height:hanes_clean$genderM
                                1.998
```

3] The dataset that I selected from [Daily Data | Statista] is - China's Population:




To determine the population of China over the years:

The second-degree polynomial model (model2) for your China population data has the following coefficients:

Intercept: -7.55e+05

Coefficient for year: 7.46e+02

Coefficient for (year^2): -1.84e-01

The regression equation for the second-degree polynomial model is:

For second Degree polynomial:

Population=−7.55e+05+7.46e+02·year−1.84e−01·year^2

The third-degree polynomial model (model3) for your China population data has the following coefficients:

Intercept: 2.02e+06

Coefficient for year: -3.41e+03

Coefficient for (year^2): 1.89e+00

Coefficient for (year^3): -3.45e-04

The regression equation for the third-degree polynomial model is:

$Population=2.02e+06-3.41e+03 \cdot year+1.89e+00 \cdot year^2-3.45e-04 \cdot year^3$

We need the co-efficient of correlation, slopes  of each model 2 and 3, and here are the values obtained in R:

| | | |
|---|---|---|
| model2 | list [12] (S3: lm) | List of length 12 |
| coefficients | double [3] | -7.55e+05 7.46e+02 -1.84e-01 |
| residuals | double [49] | 6.498 2.481 0.981 -2.430 -5.525 -6.801 ... |
| effects | double [49] | -8867.86 895.84 229.98 -3.48 -6.57 -7.83 ... |
| rank | integer [1] | 3 |
| fitted.values | double [49] | 981 998 1016 1033 1049 1065 ... |
| assign | integer [3] | 0 1 2 |
| qr | list [5] (S3: qr) | List of length 5 |
| df.residual | integer [1] | 46 |
| xlevels | list [0] | List of length 0 |
| call | language | lm(formula = Population ~ year + I(year^2), data = china) |
| terms | formula | Population ~ year + I(year^2) |
| model | list [49 x 3] (S3: data.frame) | A data.frame with 49 rows and 3 columns |

| | | |
|---|---|---|
| model3 | list [12] (S3: lm) | List of length 12 |
| coefficients | double [4] | 2.02e+06 -3.41e+03 1.89e+00 -3.45e-04 |
| residuals | double [49] | 4.7073 1.1379 0.0386 -3.0188 -5.8021 -6.8092 ... |
| effects | double [49] | -8867.86 895.84 229.98 -5.35 -6.52 -7.70 ... |
| rank | integer [1] | 4 |
| fitted.values | double [49] | 982 1000 1017 1033 1049 1065 ... |
| assign | integer [4] | 0 1 2 3 |
| qr | list [5] (S3: qr) | List of length 5 |
| df.residual | integer [1] | 45 |
| xlevels | list [0] | List of length 0 |
| call | language | lm(formula = Population ~ year + I(year^2) + I(year^3), data = china) |
| terms | formula | Population ~ year + I(year^2) + I(year^3) |
| model | list [49 x 4] (S3: data.frame) | A data.frame with 49 rows and 4 columns |

Reference webpages:

1. [R CHARTS | A collection of charts and graphs made with the R programming language (r-charts.com)]()
2. [R Tutorial (w3schools.com)]()

R code that I wrote :

```
1] avg <- function(x) {
  sum(x) / length(x)
}

# Question 1
hanes <- readRDS("hanes.rds") # Reading hanes data
hanes_clean <- na.omit(hanes) # Removing NA values

# Creating scatter plot
plot(
  x = hanes_clean$height,
  y = hanes_clean$weight,
  xlab = "Height (Inches)",
  ylab = "Weight (Lbs)",
  main = "Hanes data with Indicator variable (gender)",
  pch = ifelse(hanes_clean$gender == "M", 20, 1),
  col = ifelse(hanes_clean$gender == "M", "red", "green")
)

# Getting intercept and slope
lm_model <- lm(hanes_clean$weight ~ hanes_clean$height + hanes_clean$gender)

# Adding regression line for male (yellow)
abline(
  a = coef(lm_model)[1],
  b = coef(lm_model)[2],
  lwd = 2,
```

```
  col = "red"
)

# Adding regression line for female (green)
abline(
  a = coef(lm_model)[1] + coef(lm_model)[3],
  b = coef(lm_model)[2],
  lwd = 2,
  col = "green"
)

# Adding legends
legend("topright", c("Male", "Female"), pch = c(20, 1), col = c("red", "green"))

# Getting residual sum of squares
residual_sum_of_squares <- sum(lm_model$residuals^2)

# Getting Total sum of squares
total_sum_of_squares <- sum((hanes_clean$weight - avg(hanes_clean$weight))^2)

# Calculating coefficient of determination
coefficient_of_determination <- ((total_sum_of_squares - residual_sum_of_squares) / total_sum_of_squares) * 100
```

2] plot(x=hanes_clean$height, y=hanes_clean$weight,xlab="Height (Inches)",ylab="Weight (Lbs)",main="Hanes data with Indicator variable (gender) and an Interaction term",pch= ifelse(hanes_clean$gender == "M",20,1),col =ifelse(hanes_clean$gender == "M","red","green")

```
#creating scatter plot
lm(hanes_clean$weight ~ hanes_clean$height + hanes_clean$gender + hanes_clean$height:hanes_clean$gender)
```

```
# getting intercept and slop

abline(a=-289.141, b=6.852, lwd=2, col="green")

# adding legends

sum(lm(hanes_clean$weight ~ hanes_clean$height + hanes_clean$gender+
hanes_clean$height:hanes_clean$gender)$residuals^2)

# adding regression line for male

abline(a=-143.998, b=4.854, lwd=2, col="red")

# adding regression line for female

legend("topright", c("Male", "Female"), pch = c(20,1),col =c("green","red"))

# calculating coefficient of determination

residual_sum_of_squares <- (179804.2 -147695.6)*100/179804.2
```

3]

```
 # Load the required packages if not already loaded

install.packages("ggplot2")

library(ggplot2)


# Define avg function

avg <- function(x) {

  return(mean(x))

}


# Data

china <- data.frame(

  year = c(

    1980, 1981, 1982, 1983, 1984, 1985, 1986, 1987, 1988, 1989, 1990,

    1991, 1992, 1993, 1994, 1995, 1996, 1997, 1998, 1999, 2000, 2001,

    2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012,

    2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020, 2021, 2022, 2023,
```

```r
    2024, 2025, 2026, 2027, 2028
  ),
  Population = c(
    987.05, 1000.72, 1016.54, 1030.08, 1043.57, 1058.51, 1075.07, 1093,
    1110.26, 1127.04, 1143.33, 1158.23, 1171.71, 1185.17, 1198.50, 1211.21,
    1223.89, 1236.26, 1247.61, 1257.86, 1267.43, 1276.27, 1284.53, 1292.27,
    1299.88, 1307.56, 1314.48, 1321.29, 1328.02, 1334.50, 1340.91, 1349.16,
    1359.22, 1367.26, 1376.46, 1383.26, 1392.32, 1400.11, 1405.41, 1410.08,
    1412.12, 1412.60, 1411.75, 1411.40, 1410.78, 1409.82, 1408.53, 1406.94, 1405.04
  )
)

# Set x-axis label
xlab <- paste("Years since", min(china$year))

# Set y-axis label
ylab <- "Population (millions)"

# Set main title
main_title <- "Population of China Over Time"

# Fit a second-degree polynomial model
model2 <- lm(Population ~ year + I(year^2), data = china)

# Fit a third-degree polynomial model
model3 <- lm(Population ~ year + I(year^2) + I(year^3), data = china)

# Plot the population data
plot(x = china$year, y = china$Population, xlab = xlab, ylab = ylab, main = main_title, pch = 20)
```

```
# Define a function for the second-degree polynomial
p2_function <- function(x) {
  return(predict(model2, newdata = data.frame(year = x)))
}

# Plot the second-degree polynomial
curve(p2_function, from = min(china$year), to = max(china$year), col = "red", lty = 2, add = TRUE)

# Define a function for the third-degree polynomial
p3_function <- function(x) {
  return(predict(model3, newdata = data.frame(year = x)))
}

# Plot the third-degree polynomial
curve(p3_function, from = min(china$year), to = max(china$year), col = "blue", lty = 3, add = TRUE)

# Calculate R-squared values
r_squared2 <- summary(model2)$r.squared
r_squared3 <- summary(model3)$r.squared

# Add coefficients of determination to the plot
legend(
  "bottomright",
  legend = c(
    paste("2nd Degree R^2 =", round(r_squared2, 4)),
    paste("3rd Degree R^3 =", round(r_squared3, 4))
  ),
  col = c("red", "blue"),
```

```
  lty = c(2, 3)
)
```