Homework 5

Rolando Santos

2023-11-17

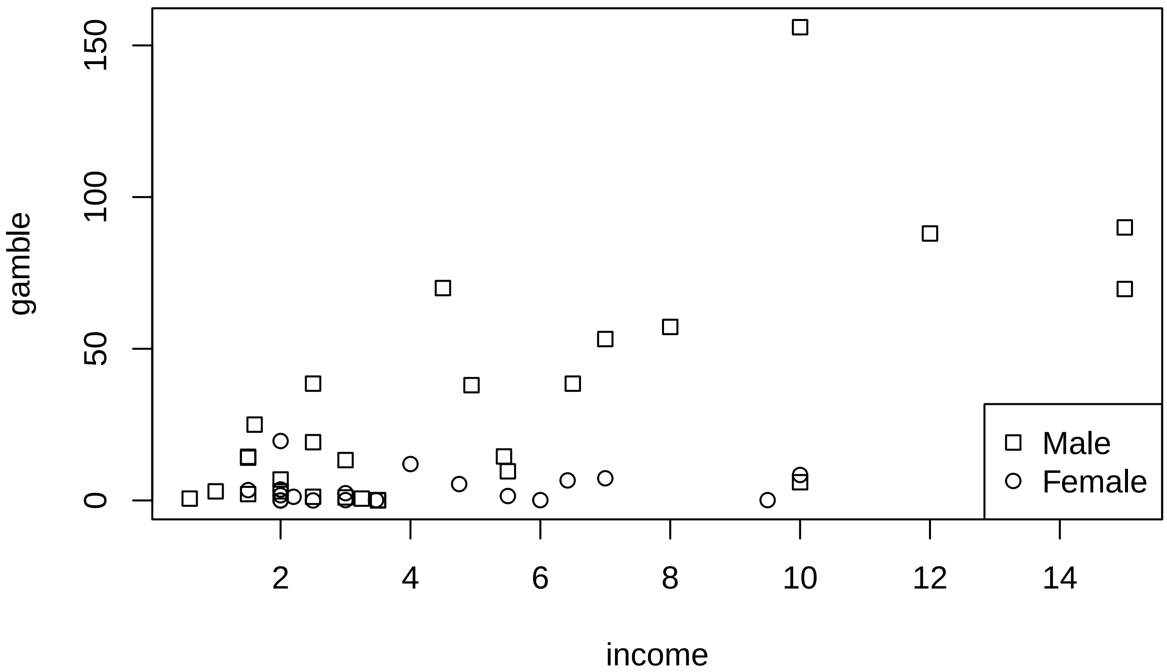
Question 1: Let's revisit the teengamb dataset in this question

```
teengamb <- read.csv("teengamb.csv")
head(teengamb)
```

```
##   sex status income verbal gamble
## 1   1     51   2.00      8    0.0
## 2   1     28   2.50      8    0.0
## 3   1     37   2.00      6    0.0
## 4   1     28   7.00      4    7.3
## 5   1     65   2.00      8   19.6
## 6   1     61   3.47      6    0.1
```

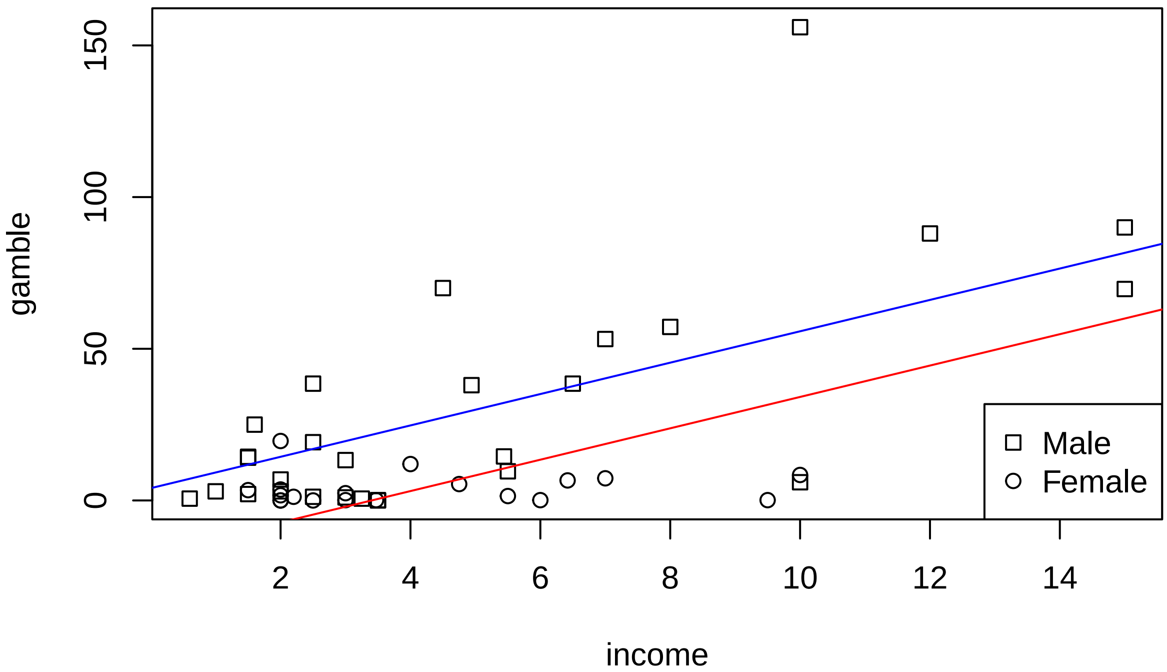
a. Make a plot of gamble on income using a different plotting symbol depending on the sex.

```
plot(
  gamble ~ income,
  pch = c(0, 1)[as.factor(sex)],
  data = teengamb
)
legend(
  "bottomright",
  legend = c("Male", "Female"),
  pch = c(0, 1, 2)
)
```



b. Fit a regression model with gamble as the response and income and sex as predictors. Display the regression fit with sex = 0 and sex = 1 separately on the plot. (Hint: use abline function)

```
lm <- lm(gamble ~ income  + sex, data = teengamb)
plot(
  gamble ~ income,
  pch = c(0, 1)[as.factor(sex)],
  data = teengamb)
abline(lm$coefficients[1], lm$coefficients[2], col = 'blue')
abline(lm$coefficients[1] + lm$coefficients[3], lm$coefficients[2], col = 'red')
legend(
  "bottomright",
  legend = c("Male", "Female"),
  pch = c(0, 1, 2)
)
```



c. Use the Matching package to find matches on sex by treating income as the confounder. Use the same parameters as in the lecture slides. How many matched pairs were found? How many cases were not matched?

```r
library(Matching)
```

```
## Loading required package: MASS
```

```
## ##
## ##  Matching (Version 4.10-14, Build Date: 2023-09-13)
## ##  See https://www.jsekhon.com for additional documentation.
## ##  Please cite software as:
## ##   Jasjeet S. Sekhon. 2011. ``Multivariate and Propensity Score Matching
## ##   Software with Automated Balance Optimization: The Matching package for R.''
## ##   Journal of Statistical Software, 42(7): 1-52.
## ##
```

```r
set.seed(2022)
mm <- GenMatch(teengamb$sex, teengamb$income, ties = FALSE)
```

```
## Loading required namespace: rgenoud
```

```
## Warning in GenMatch(teengamb$sex, teengamb$income, ties = FALSE): The key
## tuning parameters for optimization were are all left at their default values.
## The 'pop.size' option in particular should probably be increased for optimal
## results.  For details please see the help page and https://www.jsekhon.com
```

```
##
```

```
##
## Sun Dec  3 16:24:30 2023
## Domains:
##  0.000000e+00   <=  X1   <=    1.000000e+03
##
## Data Type: Floating Point
## Operators (code number, name, population)
##  (1) Cloning..........................  15
##  (2) Uniform Mutation.................  12
##  (3) Boundary Mutation................  12
##  (4) Non-Uniform Mutation.............  12
##  (5) Polytope Crossover...............  12
##  (6) Simple Crossover.................  12
##  (7) Whole Non-Uniform Mutation........  12
##  (8) Heuristic Crossover..............  12
##  (9) Local-Minimum Crossover..........  0
##
## SOFT Maximum Number of Generations: 100
## Maximum Nonchanging Generations: 4
## Population size       : 100
## Convergence Tolerance: 1.000000e-03
##
## Not Using the BFGS Derivative Based Optimizer on the Best Individual Each Generation.
## Not Checking Gradients before Stopping.
## Using Out of Bounds Individuals.
##
## Maximization Problem.
## GENERATION: 0 (initializing the population)
## Lexical Fit..... 6.648968e-01  1.000000e+00
## #unique......... 100, #Total UniqueCount: 100
## var 1:
## best............ 1.000000e+00
## mean............ 5.088976e+02
## variance........ 8.392732e+04
##
## GENERATION: 1
## Lexical Fit..... 6.648968e-01  1.000000e+00
## #unique......... 57, #Total UniqueCount: 157
## var 1:
## best............ 1.000000e+00
## mean............ 3.964992e+02
## variance........ 9.965589e+04
##
## GENERATION: 2
## Lexical Fit..... 6.648968e-01  1.000000e+00
## #unique......... 57, #Total UniqueCount: 214
## var 1:
## best............ 1.000000e+00
## mean............ 4.452137e+02
## variance........ 9.704986e+04
##
## GENERATION: 3
## Lexical Fit..... 6.648968e-01  1.000000e+00
## #unique......... 59, #Total UniqueCount: 273
```

```
## var 1:
## best............ 1.000000e+00
## mean............ 3.586202e+02
## variance........ 1.052395e+05
##
## GENERATION: 4
## Lexical Fit..... 6.648968e-01  1.000000e+00
## #unique......... 51, #Total UniqueCount: 324
## var 1:
## best............ 1.000000e+00
## mean............ 3.883162e+02
## variance........ 1.018758e+05
##
## GENERATION: 5
## Lexical Fit..... 6.648968e-01  1.000000e+00
## #unique......... 55, #Total UniqueCount: 379
## var 1:
## best............ 1.000000e+00
## mean............ 3.584421e+02
## variance........ 9.711598e+04
##
## 'wait.generations' limit reached.
## No significant improvement in 4 generations.
##
## Solution Lexical Fitness Value:
## 6.648968e-01  1.000000e+00
##
## Parameters at the Solution:
##
##  X[ 1] : 1.000000e+00
##
## Solution Found Generation 1
## Number of Generations Run 5
##
## Sun Dec  3 16:24:31 2023
## Total run time : 0 hours 0 minutes and 1 seconds
```

```
match <- mm$matches[, 1:2]
match
```

```
##       [,1] [,2]
##  [1,]    1   34
##  [2,]    2   47
##  [3,]    3   41
##  [4,]    4   32
##  [5,]    5   34
##  [6,]    6   23
##  [7,]    7   30
##  [8,]    8   25
##  [9,]    9   34
## [10,]   10   25
## [11,]   11   43
## [12,]   12   45
## [13,]   13   41
```

```
## [14,]   14   41
## [15,]   15   21
## [16,]   16   46
## [17,]   17   39
## [18,]   18   39
## [19,]   19   36
```

```r
nrow(match)
```

```
## [1] 19
```

```r
nrow(teengamb[-c(match[, 1], match[, 2]), ])
```

```
## [1] 15
```

There are 19 total pairs found. There are 15 cases in the dataset where no matching pair was found.

d. Compute the differences in gamble for the matched pairs. Is there a significant non-zero difference using one-sample t-test?

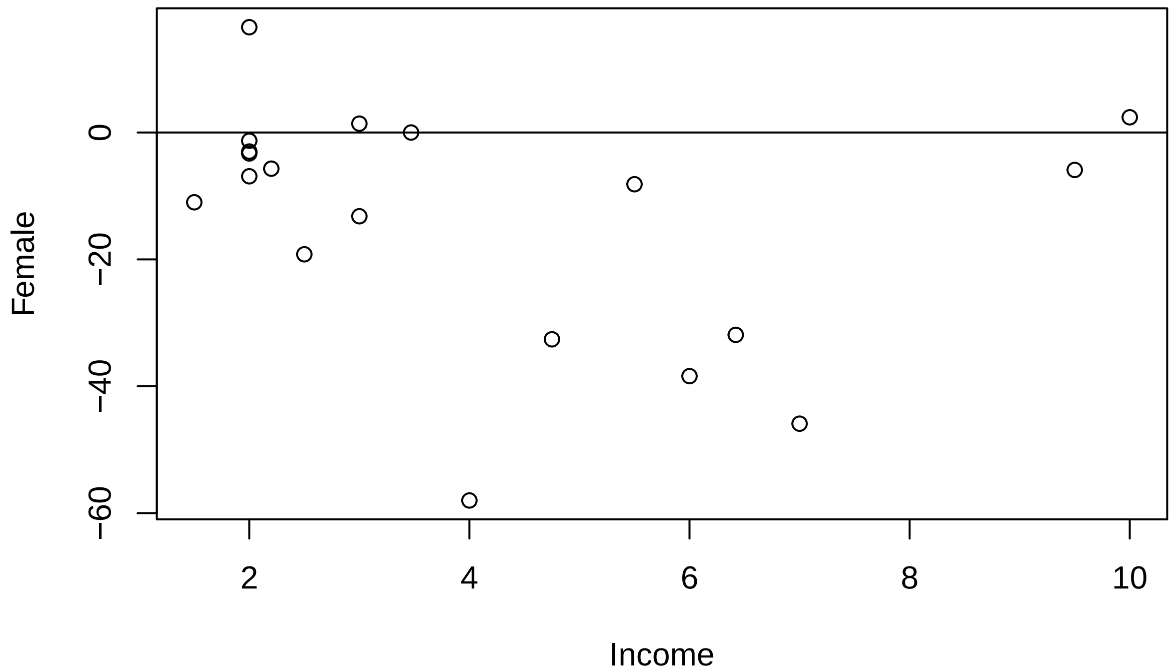
```r
pdiff <- teengamb$gamble[match[,1]] - teengamb$gamble[match[,2]]
t.test(pdiff)
```

```
##
##  One Sample t-test
##
## data:  pdiff
## t = -3.1863, df = 18, p-value = 0.005115
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  -23.060860  -4.733876
## sample estimates:
## mean of x
## -13.89737
```

Based on the t-test p-value, there is a significant difference between gambling values among matched pairs.

e. Plot the difference in gamble against income. In what proportion of pairs did the female gamble more than the male?

```r
plot(pdiff ~ teengamb$income[match[,1]],
     xlab="Income", ylab="Female")
abline(h=0)
```




```r
mean(pdiff > 0)
```

```
## [1] 0.1578947
```

We can see that in ~15.7% of our matched pairs, females gambled more than their male counterparts.

f. Do the conclusions from the linear model and the matched pair approach agree? Give you interpretation and insight.

They do appear to agree, in our linear model we see on our regression lines that males were likelier to gamble more compared to females. In our matches we found that only ~15.7% of females gambled more than males. Our plot also showed that most points that were were below the 0 line (signifying that the difference female.gambling - male.gambling was mostly male favored).

Question 2: The infmort dataset records the infant mortality of 105 countries with their income, region, and oil export information. The infant mortality in regions of the world may be related to per capita income and whether oil is exported.

```r
infmort <- read.csv("infmort.csv")
head(infmort)
```

```
##                      X   region income mortality            oil
## 1 Australia               Asia   3426      26.7 no oil exports
## 2 Austria               Europe   3350      23.7 no oil exports
## 3 Belgium               Europe   3346      17.0 no oil exports
## 4 Canada              Americas   4751      16.8 no oil exports
## 5 Denmark               Europe   5029      13.5 no oil exports
## 6 Finland               Europe   3312      10.1 no oil exports
```

a. Which variables are continuous? Which are categorical variables? How many levels the categorical variable have?

```r
infmort$X <- as.factor(infmort$X)
infmort$region <- as.factor(infmort$region)
infmort$oil <- as.factor(infmort$oil)

levels(infmort$X)
```

```
##   [1] "Afganistan          " "Algeria             " "Argentina           "
##   [4] "Australia           " "Austria             " "Bangladesh          "
##   [7] "Belgium             " "Bolivia             " "Brazil              "
##  [10] "Britain             " "Burma               " "Burundi             "
##  [13] "Cambodia            " "Cameroon            " "Canada              "
##  [16] "Central_African_Rep" "Chad                " "Chile               "
##  [19] "Colombia            " "Congo               " "Costa_Rica          "
##  [22] "Dahomey             " "Denmark             " "Dominican_Republic  "
##  [25] "Ecuador             " "Egypt               " "El_Salvador         "
##  [28] "Ethiopia            " "Finland             " "France              "
##  [31] "Ghana               " "Greece              " "Guatemala           "
##  [34] "Guinea              " "Haiti               " "Honduras            "
##  [37] "India               " "Indonesia           " "Iran                "
##  [40] "Iraq                " "Ireland             " "Israel              "
##  [43] "Italy               " "Ivory_Coast         " "Jamaica             "
##  [46] "Japan               " "Jordan              " "Kenya               "
##  [49] "Laos                " "Lebanon             " "Liberia             "
##  [52] "Libya               " "Madagascar          " "Malawi              "
##  [55] "Malaysia            " "Mali                " "Mauritania          "
##  [58] "Mexico              " "Moroco              " "Nepal               "
##  [61] "Netherlands         " "New_Zealand         " "Nicaragua           "
##  [64] "Niger               " "Nigeria             " "Norway              "
##  [67] "Pakistan            " "Panama              " "Papua_New_Guinea    "
##  [70] "Paraguay            " "Peru                " "Philippines         "
##  [73] "Portugal            " "Rwanda              " "Saudi_Arabia        "
##  [76] "Sierra_Leone        " "Singapore           " "Somalia             "
##  [79] "South_Africa        " "South_Korea         " "South_Vietnam       "
##  [82] "Southern_Yemen      " "Spain               " "Sri_Lanka           "
##  [85] "Sudan               " "Sweden              " "Switzerland         "
##  [88] "Syria               " "Taiwan              " "Tanzania            "
##  [91] "Thailand            " "Togo                " "Trinidad_and_Tobago"
##  [94] "Tunisia             " "Turkey              " "Uganda              "
##  [97] "United_States       " "Upper_Volta         " "Uruguay             "
## [100] "Venezuela           " "West_Germany        " "Yemen               "
## [103] "Yugoslavia          " "Zaire               " "Zambia              "
```

```r
levels(infmort$region)
```

```
## [1] "Africa"   "Americas" "Asia"     "Europe"
```

```r
levels(infmort$oil)
```

```
## [1] "no oil exports" "oil exports"
```

From first glance at the dataset, the country(X), region and oil variables are categorical, and the income and mortality variables are numeric.

The country(X) variable has 105 levels (every entry is a unique country), region has 4 levels and oil has 2 levels.

b. Regress mortality on all other variables. Interpret the model output and the meaning of estimated parameters.

```
lm.model <- lm(mortality ~ X, data = infmort)
summary(lm.model)
```

```
##
## Call:
## lm(formula = mortality ~ X, data = infmort)
##
## Residuals:
## ALL 101 residuals are 0: no residual degrees of freedom!
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)               400.0        NaN     NaN      NaN
## XAlgeria                 -313.7        NaN     NaN      NaN
## XArgentina               -340.4        NaN     NaN      NaN
## XAustralia               -373.3        NaN     NaN      NaN
## XAustria                 -376.3        NaN     NaN      NaN
## XBangladesh              -275.7        NaN     NaN      NaN
## XBelgium                 -383.0        NaN     NaN      NaN
## XBolivia                 -339.6        NaN     NaN      NaN
## XBrazil                  -230.0        NaN     NaN      NaN
## XBritain                 -382.5        NaN     NaN      NaN
## XBurma                   -200.0        NaN     NaN      NaN
## XBurundi                 -250.0        NaN     NaN      NaN
## XCambodia                -300.0        NaN     NaN      NaN
## XCameroon                -263.0        NaN     NaN      NaN
## XCanada                  -383.2        NaN     NaN      NaN
## XCentral_African_Rep     -210.0        NaN     NaN      NaN
## XChad                    -240.0        NaN     NaN      NaN
## XChile                   -322.0        NaN     NaN      NaN
## XColombia                -337.2        NaN     NaN      NaN
## XCongo                   -220.0        NaN     NaN      NaN
## XCosta_Rica              -345.6        NaN     NaN      NaN
## XDahomey                 -290.4        NaN     NaN      NaN
## XDenmark                 -386.5        NaN     NaN      NaN
## XDominican_Republic      -351.2        NaN     NaN      NaN
## XEcuador                 -321.5        NaN     NaN      NaN
## XEgypt                   -286.0        NaN     NaN      NaN
## XEl_Salvador             -341.8        NaN     NaN      NaN
## XEthiopia                -315.8        NaN     NaN      NaN
## XFinland                 -389.9        NaN     NaN      NaN
## XFrance                  -387.1        NaN     NaN      NaN
## XGhana                   -336.3        NaN     NaN      NaN
```

```
## XGreece                -372.2        NaN     NaN      NaN
## XGuatemala             -320.9        NaN     NaN      NaN
## XGuinea                -184.0        NaN     NaN      NaN
## XHonduras              -360.7        NaN     NaN      NaN
## XIndia                 -339.4        NaN     NaN      NaN
## XIndonesia             -275.0        NaN     NaN      NaN
## XIraq                  -371.9        NaN     NaN      NaN
## XIreland               -382.2        NaN     NaN      NaN
## XIsrael                -377.9        NaN     NaN      NaN
## XItaly                 -374.3        NaN     NaN      NaN
## XIvory_Coast           -262.0        NaN     NaN      NaN
## XJamaica               -373.8        NaN     NaN      NaN
## XJapan                 -388.3        NaN     NaN      NaN
## XJordan                -378.7        NaN     NaN      NaN
## XKenya                 -345.0        NaN     NaN      NaN
## XLebanon               -386.4        NaN     NaN      NaN
## XLiberia               -240.8        NaN     NaN      NaN
## XLibya                 -100.0        NaN     NaN      NaN
## XMadagascar            -298.0        NaN     NaN      NaN
## XMalawi                -251.7        NaN     NaN      NaN
## XMalaysia              -368.0        NaN     NaN      NaN
## XMali                  -280.0        NaN     NaN      NaN
## XMauritania            -213.0        NaN     NaN      NaN
## XMexico                -339.1        NaN     NaN      NaN
## XMoroco                -251.0        NaN     NaN      NaN
## XNetherlands           -388.4        NaN     NaN      NaN
## XNew_Zealand           -383.8        NaN     NaN      NaN
## XNicaragua             -354.0        NaN     NaN      NaN
## XNiger                 -200.0        NaN     NaN      NaN
## XNigeria               -342.0        NaN     NaN      NaN
## XNorway                -388.7        NaN     NaN      NaN
## XPakistan              -275.7        NaN     NaN      NaN
## XPanama                -365.9        NaN     NaN      NaN
## XPapua_New_Guinea      -389.8        NaN     NaN      NaN
## XParaguay              -361.4        NaN     NaN      NaN
## XPeru                  -334.9        NaN     NaN      NaN
## XPhilippines           -332.1        NaN     NaN      NaN
## XPortugal              -355.2        NaN     NaN      NaN
## XRwanda                -267.1        NaN     NaN      NaN
## XSaudi_Arabia           250.0        NaN     NaN      NaN
## XSierra_Leone          -230.0        NaN     NaN      NaN
## XSingapore             -379.6        NaN     NaN      NaN
## XSomalia               -242.0        NaN     NaN      NaN
## XSouth_Africa          -328.5        NaN     NaN      NaN
## XSouth_Korea           -342.0        NaN     NaN      NaN
## XSouth_Vietnam         -300.0        NaN     NaN      NaN
## XSouthern_Yemen        -320.0        NaN     NaN      NaN
## XSpain                 -384.9        NaN     NaN      NaN
## XSri_Lanka             -354.9        NaN     NaN      NaN
## XSudan                 -270.6        NaN     NaN      NaN
## XSweden                -390.4        NaN     NaN      NaN
## XSwitzerland           -387.2        NaN     NaN      NaN
## XSyria                 -378.3        NaN     NaN      NaN
## XTaiwan                -380.9        NaN     NaN      NaN
```

```
## XTanzania              -237.5        NaN     NaN       NaN
## XThailand              -373.0        NaN     NaN       NaN
## XTogo                  -273.0        NaN     NaN       NaN
## XTrinidad_and_Tobago   -373.8        NaN     NaN       NaN
## XTunisia               -323.7        NaN     NaN       NaN
## XTurkey                -247.0        NaN     NaN       NaN
## XUganda                -240.0        NaN     NaN       NaN
## XUnited_States         -382.4        NaN     NaN       NaN
## XUpper_Volta           -220.0        NaN     NaN       NaN
## XUruguay               -359.6        NaN     NaN       NaN
## XVenezuela             -348.3        NaN     NaN       NaN
## XWest_Germany          -379.6        NaN     NaN       NaN
## XYemen                 -350.0        NaN     NaN       NaN
## XYugoslavia            -356.7        NaN     NaN       NaN
## XZaire                 -296.0        NaN     NaN       NaN
## XZambia                -141.0        NaN     NaN       NaN
##
## Residual standard error: NaN on 0 degrees of freedom
##   (4 observations deleted due to missingness)
## Multiple R-squared:      1,  Adjusted R-squared:    NaN
## F-statistic:   NaN on 100 and 0 DF,  p-value: NA
```

Since X is a categorical value of 105 levels, the model is trying to create dummy variable coefficients for every level, however since every level is an entry and unique, all other variables are irrelevant and we directly calculate the infant mortality based on the coefficient for the dummy variable. An example is that the intercept is 400, if we want to find the infant mortality for Algeria, we subtract -313.7 and get 86.3 which is the exact value from the dataset. This is problematic because we are depending solely on the country value and this model cannot work if we have a country outside of the dataset if we want to predict other unique data entries.

c. Regress mortality on income, region, oil, the interaction between income and region, and the interaction between income and oil. Compare this model with the one in (b). Interpret the estimated parameters.

```r
lm.model <- lm(mortality ~ income + region + oil, data = infmort)
summary(lm.model)
```

```
##
## Call:
## lm(formula = mortality ~ income + region + oil, data = infmort)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -156.00  -32.20   -4.44   13.65  488.82
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)     1.368e+02  1.363e+01  10.042  < 2e-16 ***
## income         -5.290e-03  7.404e-03  -0.714 0.476685
## regionAmericas -8.365e+01  2.180e+01  -3.837 0.000224 ***
## regionAsia     -4.589e+01  2.014e+01  -2.278 0.024977 *
## regionEurope   -1.015e+02  3.073e+01  -3.303 0.001351 **
```

```
## oiloil exports  7.834e+01  2.891e+01   2.710 0.007992 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 77.36 on 95 degrees of freedom
##   (4 observations deleted due to missingness)
## Multiple R-squared:  0.3105, Adjusted R-squared:  0.2742
## F-statistic: 8.556 on 5 and 95 DF,  p-value: 1.015e-06
```

```r
lm.model <- lm(income ~ region, data = infmort)
summary(lm.model)
```

```
##
## Call:
## lm(formula = income ~ region, data = infmort)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2634.2  -515.9  -192.2     7.8  4583.1
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)        273.2      180.5   1.514   0.1332
## regionAmericas     666.6      284.1   2.346   0.0209 *
## regionAsia         365.6      263.6   1.387   0.1685
## regionEurope      2767.0      306.8   9.020 1.29e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1052 on 101 degrees of freedom
## Multiple R-squared:  0.4641, Adjusted R-squared:  0.4482
## F-statistic: 29.16 on 3 and 101 DF,  p-value: 1.157e-13
```

```r
predict(lm.model, infmort)
```

```
##         1         2         3         4         5         6         7         8
##  638.8667 3040.2222 3040.2222  939.8696 3040.2222 3040.2222 3040.2222 3040.2222
##         9        10        11        12        13        14        15        16
## 3040.2222 3040.2222 3040.2222 3040.2222  638.8667 3040.2222 3040.2222  273.2353
##        17        18        19        20        21        22        23        24
## 3040.2222 3040.2222 3040.2222  939.8696  273.2353  939.8696  638.8667  638.8667
##        25        26        27        28        29        30        31        32
##  638.8667  273.2353  273.2353  638.8667  939.8696  939.8696  939.8696  939.8696
##        33        34        35        36        37        38        39        40
##  939.8696  939.8696  939.8696 3040.2222  939.8696  638.8667  939.8696  638.8667
##        41        42        43        44        45        46        47        48
##  638.8667  939.8696  939.8696  939.8696  939.8696  638.8667 3040.2222  638.8667
##        49        50        51        52        53        54        55        56
##  939.8696  273.2353  939.8696 3040.2222  273.2353  939.8696  273.2353  273.2353
##        57        58        59        60        61        62        63        64
##  273.2353  939.8696  273.2353  939.8696  273.2353  638.8667  638.8667  273.2353
##        65        66        67        68        69        70        71        72
##  273.2353  638.8667  939.8696  638.8667  638.8667  638.8667  638.8667  638.8667
```

```
##        73        74        75        76        77        78        79        80
##  638.8667  638.8667  638.8667  273.2353  638.8667  273.2353  273.2353  273.2353
##        81        82        83        84        85        86        87        88
##  273.2353  273.2353  939.8696  638.8667  273.2353  638.8667  273.2353  273.2353
##        89        90        91        92        93        94        95        96
##  273.2353  273.2353  638.8667  273.2353  638.8667  273.2353  273.2353  273.2353
##        97        98        99       100       101       102       103       104
##  638.8667  273.2353  273.2353  273.2353  273.2353  273.2353  638.8667  638.8667
##       105
##  273.2353
```

```
lm.model <- lm(income ~ oil, data = infmort)
summary(lm.model)
```

```
##
## Call:
## lm(formula = income ~ oil, data = infmort)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -952.1 -877.1 -668.1  188.9 4593.9
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)      1002.0      145.3   6.897  4.4e-10 ***
## oiloil exports    -46.5      496.2  -0.094    0.926
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1424 on 103 degrees of freedom
## Multiple R-squared:  8.523e-05,  Adjusted R-squared:  -0.009623
## F-statistic: 0.008779 on 1 and 103 DF,  p-value: 0.9255
```

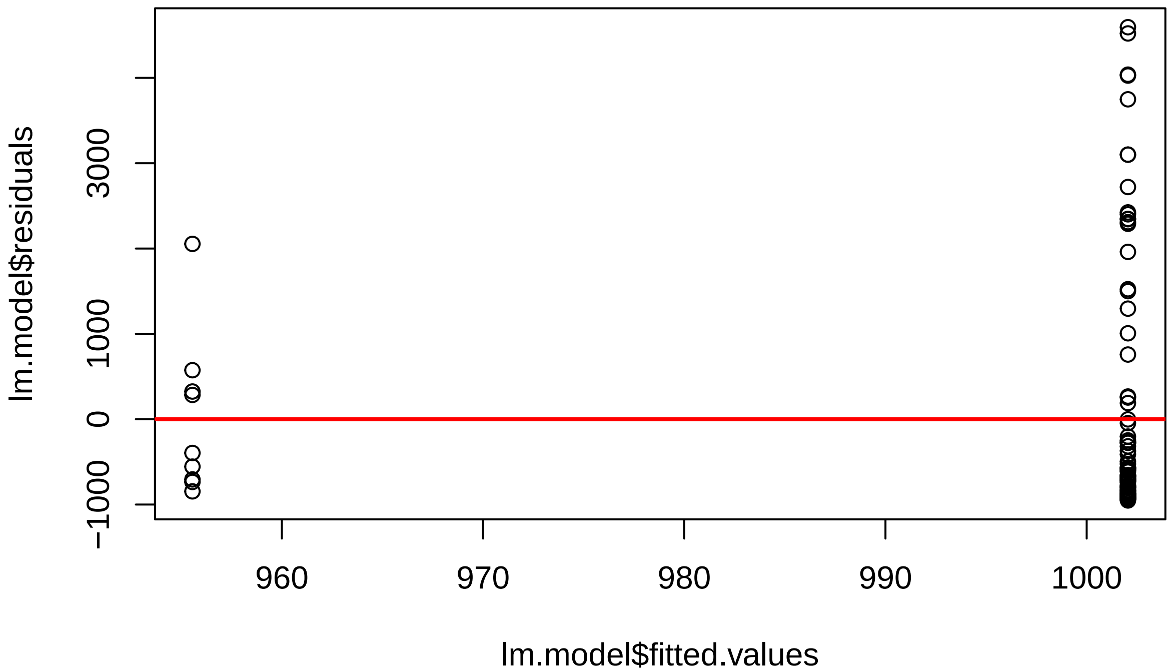
We can see when we remove country(X), we see that we have coefficients that are significant aside from income. In this model the intercept also appears to be significant. We can interpret the model as, if the region is America, Asia or Europe and depending on the income and if the country does not export oil, the rate of infant mortality decreases.

In the income and region model, we see that the Europe region is significant, and that there is a large increase of income when associated with Europe. The intercept represents Africa. When the region is also America the coefficient is also significant.

In the income and oil model, we see that countries that export oil, make less (-46.5) than countries that do not export oil.

d. Does the model in (c) satisfy the constant variance assumption? If not, give a transformation and refit the model. Check if the transformation solves the issue.

```
plot(lm.model$fitted.values, lm.model$residuals)
abline(h=0 ,col='red', lwd=2)
```

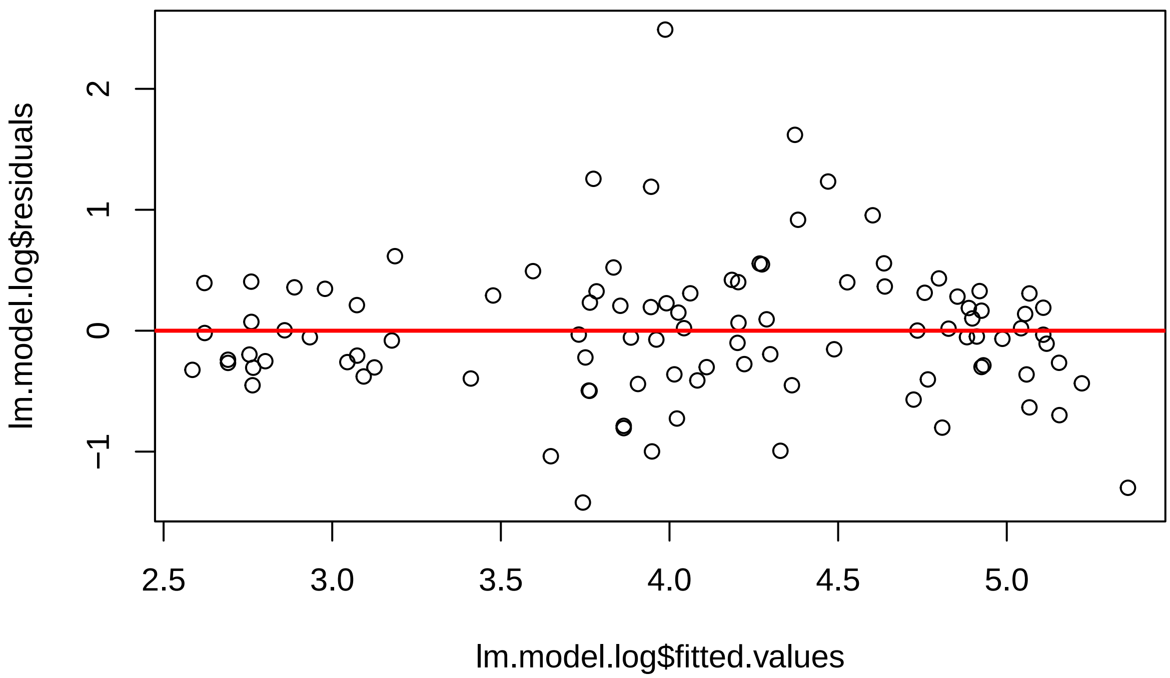


We can see that the model is violating constant variance, most of the points are grouped together at certain points in the model.

```
lm.model.log <- lm(log(mortality) ~ log(income) + region + oil, data = infmort)
summary(lm.model.log)
```

```
##
## Call:
## lm(formula = log(mortality) ~ log(income) + region + oil, data = infmort)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.4208 -0.3062 -0.0331  0.3091  2.4897
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      6.55210    0.34969  18.737  < 2e-16 ***
## log(income)     -0.33985    0.06658  -5.104 1.70e-06 ***
## regionAmericas -0.54984    0.18449  -2.980 0.003657 **
## regionAsia     -0.71292    0.15757  -4.524 1.75e-05 ***
## regionEurope   -1.03383    0.25672  -4.027 0.000114 ***
## oiloil exports  0.64021    0.22505   2.845 0.005444 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5908 on 95 degrees of freedom
##   (4 observations deleted due to missingness)
## Multiple R-squared:  0.6464, Adjusted R-squared:  0.6278
## F-statistic: 34.73 on 5 and 95 DF,  p-value: < 2.2e-16
```

```
plot(lm.model.log$fitted.values, lm.model.log$residuals)
abline(h=0 ,col='red', lwd=2)
```



After taking the log of the predictor and the log of income, we see that our fitted vs. residuals graph is showing a bit more randomness, this model is not violating constant variance.

e. Interpret the estimated parameters in (d) for region and oil variables.

We can see that similarly to our non-transformed model, with all other variables constant, the infant mortality is lower in the America, Asia and Europe regions. Likewise, infant mortality increases in countries that export oil vs countries that do not.

Question 3: In this question, you will manually implement part of the maximum likelihood estimation for logistic regression. No coding is needed. Suppose we have a dataset with one predictor X and one binary response Y. The dataset (x_i, y_i) is

$$(4,1)/(3,1)/(2,0)/(1,0)$$

So it only contains 4 observations. We use a logistic regression to model the relationship between X and Y

$$P(Y=1) = \frac{1}{1+e^{-(\beta_0+\beta_1 X)}}$$

a. Write down the likelihood function for this dataset.

$$l_1 = p^{y_1} * (1-p)^{1-y_1} = p$$

$$l_1 = \frac{1}{1+e^{-(\beta_0+4\beta_1)}}$$

$$l_2 = p^{y_2} * (1-p)^{1-y_2} = p$$

$$l_2 = \frac{1}{1+e^{-(\beta_0+3\beta_1)}}$$

$$l_3 = p^{y_3} * (1-p)^{1-y_3} = 1-p$$

$$l_3 = 1 - \frac{1}{1+e^{-(\beta_0+2\beta_1)}}$$
$$l_3 = \frac{e^{-(\beta_0+2\beta_1)}}{1+e^{-(\beta_0+2\beta_1)}}$$

$$l_4 = p^{y_4} * (1-p)^{1-y_4} = 1-p$$
$$l_4 = 1 - \frac{1}{1+e^{-(\beta_0+\beta_1)}}$$

$$l_4 = \frac{e^{-(\beta_0+\beta_1)}}{1+e^{-(\beta_0+\beta_1)}}$$

$$L = \prod_{i=1}^{n} l_i = \frac{(e^{-(\beta_0+2\beta_1)})(e^{-(\beta_0+\beta_1)})}{(1+e^{-(\beta_0+4\beta_1)})(1+e^{-(\beta_0+3\beta_1)})(1+e^{-(\beta_0+2\beta_1)})(1+e^{-(\beta_0+\beta_1)})} = L(\beta_0, \beta_1)$$

b. Write down the log-likihood function for this dataset.

$$logL(\beta) = log[L = \prod_{i=1}^{n} p^{y_i}(1-p)^{1-y_i}] = \sum_{i=1}^{n}[y_i log(p) + (1-y_i)log(1-p)]$$

$$logL(\beta) = log(\frac{1}{1+e^{-(\beta_0+4\beta_1)}}) + log(\frac{1}{1+e^{-(\beta_0+3\beta_1)}}) + log(\frac{e^{-(\beta_0+2\beta_1)}}{1+e^{-(\beta_0+2\beta_1)}}) + log(\frac{e^{-(\beta_0+\beta_1)}}{1+e^{-(\beta_0+\beta_1)}})$$

Question 4: In this question, you will use all predictors in births dataset to predict the baby's birth weight.

```
births <- read.csv("births.csv")
head(births)
```

```
##   Gender Premie weight Apgar1 Fage Mage Feduc Meduc TotPreg Visits   Marital
## 1   Male     No    124      8   31   25    13    14       1     13   Married
## 2 Female     No    177      8   36   26     9    12       2     11 Unmarried
## 3   Male     No    107      3   30   16    12     8       2     10 Unmarried
## 4 Female     No    144      6   33   37    12    14       2     12 Unmarried
## 5   Male     No    117      9   36   33    10    16       2     19   Married
## 6 Female     No     98      4   31   29    14    16       3     20   Married
##   Racemom Racedad Hispmom Hispdad Gained     Habit MomPriorCond BirthDef
## 1   White   White NotHisp NotHisp     40 NonSmoker         None     None
## 2   White   White Mexican Mexican     20 NonSmoker         None     None
## 3   White Unknown Mexican Unknown     70 NonSmoker At Least One     None
## 4   White   White NotHisp NotHisp     50 NonSmoker         None     None
## 5   White   Black NotHisp NotHisp     40 NonSmoker At Least One     None
## 6   White   White NotHisp NotHisp     21 NonSmoker         None     None
```

```
##      DelivComp BirthComp
## 1 At Least One      None
## 2 At Least One      None
## 3 At Least One      None
## 4 At Least One      None
## 5         None      None
## 6         None      None
```

a. Randomly split the whole dataset into 80% training and 20% test set. Train a linear model with all predictors using training set. Use this model to predict the weight in the test set. Calculate the prediction MSE, RMSE, and NRMSE on the test set. Use random seed 2022 before you split the data. Interpret the meaning of NRMSE.

```r
set.seed(2022)
index.train <- sample(1:dim(births)[1], 0.8 * dim(births)[1])
data.train <- births[index.train,]
data.test <- births[-index.train,]

lm.model <- lm(weight ~ ., data = data.train)
yhat.test <- predict(lm.model, data.test)

y.test <- data.test$weight
MSE.test <- mean((y.test - yhat.test)^2)
MSE.test
```

```
## [1] 278.7375
```

```r
RMSE.test <- sqrt(MSE.test)
RMSE.test
```

```
## [1] 16.69543
```

```r
NRMSE.test <- RMSE.test / mean(y.test)
NRMSE.test
```

```
## [1] 0.1421661
```

Looking at the NRMSE, we have a ~14.2% error for our birth weight prediction.

b. Repeat the data split and model training in (a), but this time predict on the training set. Calculate the MSE, RMSE, and NRMSE on the training set. Compare with test MSE, RMSE, and RMSE. What did you find? What do you think why you have a such result?

```r
set.seed(2022)
index.train <- sample(1:dim(births)[1], 0.8 * dim(births)[1])
data.train <- births[index.train,]
data.test <- births[-index.train,]
```

```r
lm.model <- lm(weight ~ ., data = data.train)
yhat.train <- predict(lm.model, data.train)

y.train <- data.train$weight
MSE.train <- mean((y.train - yhat.train)^2)
MSE.train
```

```
## [1] 250.2563
```

```r
RMSE.train <- sqrt(MSE.train)
RMSE.train
```

```
## [1] 15.81949
```

```r
NRMSE.train <- RMSE.train / mean(y.train)
NRMSE.train
```

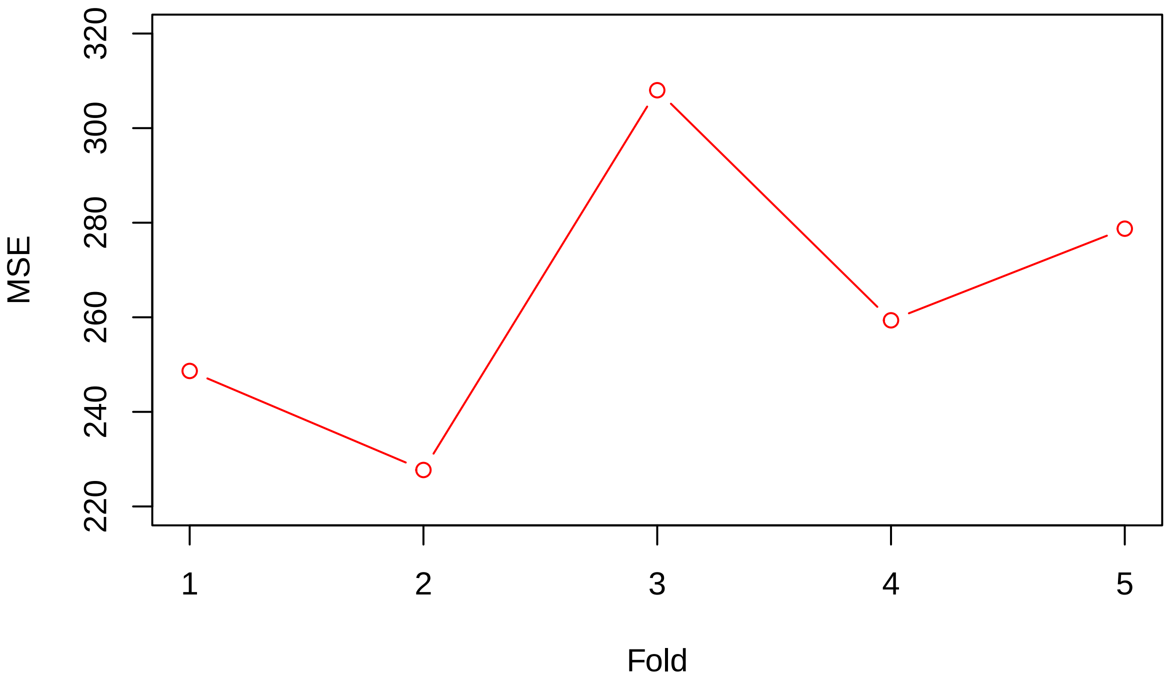
```
## [1] 0.1367234
```

Fitting our data on our training data, we have a ~13.7% error, the reason this is lower is because our data was trained for with this data, so its best fit for this data. Using the training data for testing model accuracy and error rate could potentially result with us having an overfitted model.

c. Conduct a 5-fold cross-validation to predict weight. Plot the test MSE for each fold. Show the average test MSE obtained from the cross-validation. Again, use 2022 as the random seed.

```r
set.seed(2022)

index.random <- sample(1:dim(births)[1])

groups <- cut(1:1992, 5, labels = FALSE)
index.fold <- split(index.random, groups)

MSEs <- c()

# 5-fold cross-validation
for(index.test in index.fold){
  data.test <- births[index.test,]
  data.train <- births[-index.test,]

  # fit a linear model on the training set
  lm.model <- lm(weight ~ ., data = data.train)

  # predict on the test set
  yhat.test <- predict(lm.model, data.test)

  # calculate test MSE
  y.test <- data.test$weight
  MSE.test <- mean((y.test - yhat.test)^2)
  MSEs <- c(MSEs, MSE.test)
```

```r
}
# plot 5 MSEs
plot(1:5, MSEs, type='b', col='red', xlab='Fold', ylab='MSE', ylim=c(220,320))
```




```r
# Average 5 MSEs
mean(MSEs)
```

```
## [1] 264.4943
```

The resulting average MSE that we end up with is 264.4943.