Week 2 - AYU - Pod

Contents

In this practice, we will study a term life insurance dataset collected by the Survey of Consumer Finances (SCF) which includes information about face values of life insurance demographic characteristics the customers such as age, income and martial status. we will use multiple linear model to predict the face value of term life insurance. We will demonstrate how categorical work in the model and perform F-test to compare a reduced model and a full model. We also discuss different ways to improve the model performance such as variable transformation, adding new variables and interaction terms. We then discuss the issue of collinearity and how to overcome it. You will have an opportunity to practice what you learn in a

1. MLR in R

```r
library(tidyverse)
d <- read_csv("data/TermLife.csv")
d1 <- d[d$FACE>0, ]
modelMLR <- lm(FACE ~ EDUCATION+NUMHH+INCOME, data=d1)
summary(modelMLR)
```

```
##
## Call:
## lm(formula = FACE ~ EDUCATION + NUMHH + INCOME, data = d1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2655152  -651472  -339712   -31468 13039540
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.773e+06  6.107e+05  -2.904 0.003987 **
## EDUCATION    1.463e+05  3.849e+04   3.801 0.000178 ***
## NUMHH        1.098e+05  6.552e+04   1.675 0.095001 .
## INCOME       3.392e-01  1.201e-01   2.825 0.005077 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1596000 on 271 degrees of freedom
## Multiple R-squared:  0.1011, Adjusted R-squared:  0.09115
## F-statistic: 10.16 on 3 and 271 DF,  p-value: 2.31e-06
```

2. Prediction

```r
predict(modelMLR, list(EDUCATION = 14, NUMHH =2, INCOME = 54000))
```

```
##        1
## 512999.9
```

```r
predict(modelMLR, list(EDUCATION = c(14, 20), NUMHH = c(2, 4), INCOME = c(54000, 32000)))
```

```
##         1         2
##  512999.9 1603060.4
```

```r
predict(modelMLR, list(EDUCATION = 14, NUMHH =2, INCOME = 54000), interval = 'confidence', level = 0.95)
```

```
##        fit      lwr      upr
## 1 512999.9 282512.2 743487.6
```

3. Categorical Variables

```r
modelMLR <- lm(FACE ~ EDUCATION+NUMHH+INCOME + factor(ETHNICITY), data=d1)
summary(modelMLR)
```

```
##
## Call:
## lm(formula = FACE ~ EDUCATION + NUMHH + INCOME + factor(ETHNICITY),
##     data = d1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2676482  -665029  -344030    60930 13009121
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)        -1.505e+06  6.498e+05  -2.316  0.02130 *
## EDUCATION           1.313e+05  4.108e+04   3.195  0.00157 **
## NUMHH               1.139e+05  6.754e+04   1.686  0.09302 .
## INCOME              3.381e-01  1.204e-01   2.808  0.00536 **
## factor(ETHNICITY)2 -3.565e+05  3.033e+05  -1.175  0.24097
## factor(ETHNICITY)3 -3.256e+05  4.350e+05  -0.749  0.45474
## factor(ETHNICITY)7  2.672e+04  4.187e+05   0.064  0.94917
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1600000 on 268 degrees of freedom
## Multiple R-squared:  0.107,  Adjusted R-squared:  0.08702
## F-statistic: 5.353 on 6 and 268 DF,  p-value: 3.078e-05
```

```r
predict(modelMLR, list(EDUCATION = 14, NUMHH =2, INCOME = 54000, ETHNICITY=1))
```

```
##      1
## 578506
```

4. Reduced vs Full Model (Lack of fit Test)

```r
fullmodel <- lm(FACE ~ EDUCATION+NUMHH+INCOME, data=d1)
redmodel <- lm(FACE ~ EDUCATION+NUMHH, data=d1)

anova(redmodel,fullmodel)
```

```
## Analysis of Variance Table
##
## Model 1: FACE ~ EDUCATION + NUMHH
## Model 2: FACE ~ EDUCATION + NUMHH + INCOME
##   Res.Df        RSS Df  Sum of Sq      F   Pr(>F)
## 1    272 7.1083e+14
## 2    271 6.9050e+14  1 2.0336e+13 7.9814 0.005077 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

5. Improving Models

- Transformation

```
modelMLR <- lm(log(FACE) ~ NUMHH+log(INCOME) + factor(ETHNICITY), data=d1)
summary(modelMLR)
```

```
##
## Call:
## lm(formula = log(FACE) ~ NUMHH + log(INCOME) + factor(ETHNICITY),
##     data = d1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.9312 -0.9144  0.0474  0.9701  5.6674
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)          4.32749    0.86111   5.025 9.17e-07 ***
## NUMHH                0.27973    0.06793   4.118 5.09e-05 ***
## log(INCOME)          0.62183    0.07774   7.999 3.76e-14 ***
## factor(ETHNICITY)2 -0.41161    0.30143  -1.366    0.173
## factor(ETHNICITY)3 -0.56233    0.41799  -1.345    0.180
## factor(ETHNICITY)7 -0.23870    0.41893  -0.570    0.569
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.598 on 269 degrees of freedom
## Multiple R-squared:  0.2833, Adjusted R-squared:   0.27
## F-statistic: 21.26 on 5 and 269 DF,  p-value: < 2.2e-16
```

- Add More variables

```
modelMLR <- lm(log(FACE) ~ EDUCATION + NUMHH+log(INCOME) + factor(ETHNICITY), data=d1)
summary(modelMLR)
```

```
##
## Call:
## lm(formula = log(FACE) ~ EDUCATION + NUMHH + log(INCOME) + factor(ETHNICITY),
##     data = d1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.7967 -0.8938  0.1020  0.8963  4.6705
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)          2.58087    0.89287   2.891  0.00416 **
## EDUCATION            0.20751    0.04086   5.079 7.13e-07 ***
## NUMHH                0.30534    0.06520   4.683 4.49e-06 ***
## log(INCOME)          0.49698    0.07835   6.343 9.52e-10 ***
## factor(ETHNICITY)2 -0.23353    0.29055  -0.804  0.42226
```

```
## factor(ETHNICITY)3  0.02170    0.41617   0.052  0.95846
## factor(ETHNICITY)7 -0.38440    0.40188  -0.956  0.33969
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.529 on 268 degrees of freedom
## Multiple R-squared:  0.3462, Adjusted R-squared:  0.3316
## F-statistic: 23.65 on 6 and 268 DF,  p-value: < 2.2e-16
```

- Add Interactions

```
modelMLR <- lm(log(FACE) ~ EDUCATION + NUMHH+log(INCOME) + factor(ETHNICITY) + EDUCATION*NUMHH*ETHNICIT
summary(modelMLR)
```

```
##
## Call:
## lm(formula = log(FACE) ~ EDUCATION + NUMHH + log(INCOME) + factor(ETHNICITY) +
##     EDUCATION * NUMHH * ETHNICITY, data = d1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.7578 -0.8443  0.0489  0.8815  4.6860
##
## Coefficients: (1 not defined because of singularities)
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 3.101863   1.694526   1.831   0.0683 .
## EDUCATION                   0.202323   0.147482   1.372   0.1713
## NUMHH                       0.260602   0.698221   0.373   0.7093
## log(INCOME)                 0.488257   0.080077   6.097 3.81e-09 ***
## factor(ETHNICITY)2          0.280079   1.078129   0.260   0.7952
## factor(ETHNICITY)3          1.128634   2.106627   0.536   0.5926
## factor(ETHNICITY)7          2.873956   6.465350   0.445   0.6570
## ETHNICITY                         NA         NA      NA       NA
## EDUCATION:NUMHH             0.007050   0.046388   0.152   0.8793
## EDUCATION:ETHNICITY        -0.029237   0.070578  -0.414   0.6790
## NUMHH:ETHNICITY            -0.120071   0.286895  -0.419   0.6759
## EDUCATION:NUMHH:ETHNICITY   0.006102   0.019298   0.316   0.7521
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.537 on 264 degrees of freedom
## Multiple R-squared:  0.3492, Adjusted R-squared:  0.3246
## F-statistic: 14.17 on 10 and 264 DF,  p-value: < 2.2e-16
```

```
modelMLR <- lm(log(FACE) ~ poly(log(INCOME), 2), data=d1)
summary(modelMLR)
```

```
##
## Call:
## lm(formula = log(FACE) ~ poly(log(INCOME), 2), data = d1)
##
## Residuals:
```

```
##     Min      1Q  Median      3Q     Max
## -6.2283 -0.8105  0.0274  0.8956  5.1888
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)           11.99029    0.09694 123.689  < 2e-16 ***
## poly(log(INCOME), 2)1 14.92077    1.60756   9.282  < 2e-16 ***
## poly(log(INCOME), 2)2  5.77530    1.60756   3.593 0.000388 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.608 on 272 degrees of freedom
## Multiple R-squared:  0.267,  Adjusted R-squared:  0.2616
## F-statistic: 49.53 on 2 and 272 DF,  p-value: < 2.2e-16
```

6. Collinearity

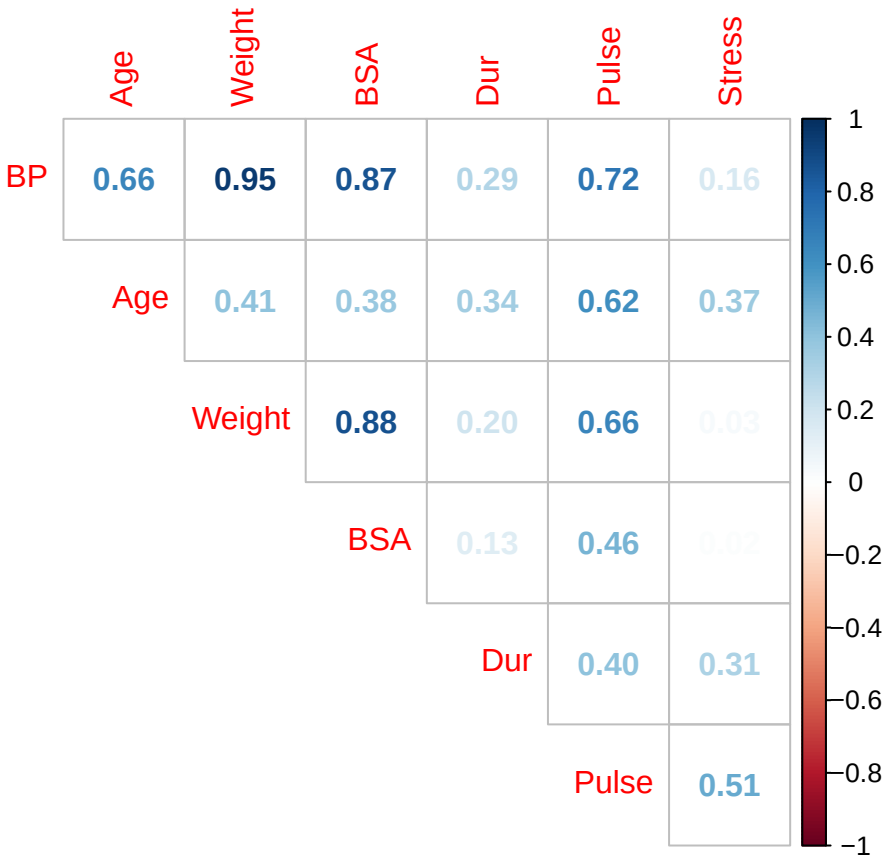
```
library(tidyverse)
library(corrplot)
library(car)

bloodpress <- read_csv("data/bloodpress.csv")

model <- lm(BP ~ Age +  Weight + BSA  +    Dur +  Pulse +  Stress, data=bloodpress)

summary(model)
```

```
##
## Call:
## lm(formula = BP ~ Age + Weight + BSA + Dur + Pulse + Stress,
##     data = bloodpress)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.93213 -0.11314  0.03064  0.21834  0.48454
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -12.870476   2.556650  -5.034 0.000229 ***
## Age           0.703259   0.049606  14.177 2.76e-09 ***
## Weight        0.969920   0.063108  15.369 1.02e-09 ***
## BSA           3.776491   1.580151   2.390 0.032694 *
## Dur           0.068383   0.048441   1.412 0.181534
## Pulse        -0.084485   0.051609  -1.637 0.125594
## Stress        0.005572   0.003412   1.633 0.126491
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4072 on 13 degrees of freedom
## Multiple R-squared:  0.9962, Adjusted R-squared:  0.9944
## F-statistic: 560.6 on 6 and 13 DF,  p-value: 6.395e-15
```

```r
# Check for collinearity
vif(model)
```

```
##      Age   Weight      BSA      Dur    Pulse   Stress
## 1.762807 8.417035 5.328751 1.237309 4.413575 1.834845
```

```r
# Check the correlation
corrplot(cor(bloodpress[, -1]), method = "number", type = "upper", diag = FALSE)
```



We notice that BSA and Pulse has high correlation with other variables. We will remove one of these variables and recheck the vif

```r
model <- lm(BP ~ Age +  Weight + BSA +    Dur  +  Stress, data=bloodpress)

summary(model)
```

```
##
## Call:
## lm(formula = BP ~ Age + Weight + BSA + Dur + Stress, data = bloodpress)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.9484 -0.1951  0.0895  0.2137  0.4930
##
```

```
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -12.587398   2.699492  -4.663 0.000366 ***
## Age           0.670659   0.048081  13.948 1.33e-09 ***
## Weight        0.899467   0.048847  18.414 3.29e-11 ***
## BSA           4.887143   1.510274   3.236 0.005978 **
## Dur           0.057500   0.050780   1.132 0.276520
## Stress        0.002396   0.002971   0.806 0.433510
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.431 on 14 degrees of freedom
## Multiple R-squared:  0.9954, Adjusted R-squared:  0.9937
## F-statistic: 600.2 on 5 and 14 DF,  p-value: 8.236e-16
```

```
# Check for collinearity
vif(model)
```

```
##      Age   Weight      BSA      Dur   Stress
## 1.478699 4.502496 4.346372 1.214004 1.241717
```

```
model <- lm(BP ~ Age +  Weight + Dur +  Pulse +  Stress, data=bloodpress)

summary(model)
```

```
##
## Call:
## lm(formula = BP ~ Age + Weight + Dur + Pulse + Stress, data = bloodpress)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.02600 -0.18526 -0.00077  0.21934  0.72533
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -15.116781   2.748758  -5.499 7.83e-05 ***
## Age           0.731940   0.055646  13.154 2.85e-09 ***
## Weight        1.098958   0.037773  29.093 6.37e-14 ***
## Dur           0.064105   0.055965   1.145   0.2712
## Pulse        -0.137444   0.053885  -2.551   0.0231 *
## Stress        0.007429   0.003841   1.934   0.0736 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4708 on 14 degrees of freedom
## Multiple R-squared:  0.9945, Adjusted R-squared:  0.9925
## F-statistic: 502.5 on 5 and 14 DF,  p-value: 2.835e-15
```

```
# Check for collinearity
vif(model)
```

```
##      Age   Weight      Dur    Pulse   Stress
## 1.659637 2.256150 1.235620 3.599913 1.739641
```

From the new vif, we will decide to remove BSA.

7. Questions

In this AYU, we will examine the hospital cost of patients in Wisconsin in 2003 using patients' information such as their age, gender, and their length of stay at the hospital.

1. Regress the total charge (`TOTCHG`) on age and length of stay. What is the R-squared of the regression? With significant level of 5%, is there any variable not significant based on coefficient p-values of the model?
2. Use the model to predict the total charge of a 13-year-old that stays a week at a hospital.
3. Adding variable `GENDER` to the regression. Write the equations of this new linear model. Use the model to predict the total charge of a 13-year-old female that stays a week at a hospital.
4. Are both AGE and APRDRG not significant and should not be added to the model? Use the F-test to address the question.
5. Could you improve the model in 3 using the methods discussed for the term life dataset?
6. Make a correlation plot. Calculate the vif. Should one pursue multilinearity analysis on the model?
7. In this question, we study the the seatpos dataset to study the car seat position of the driver at their comfort. This dataset is collected by the University of Michigan collected data on 38 drivers.

https://search.r-project.org/CRAN/refmans/faraway/html/seatpos.html

We would like to regress `hipcenter` on all other variables. Check the VIF of this regression and handle the multilinearity issue if it occurs.

8. Submission

We will use rmarkdown document for AYU submission. Rmarkdown allows us to include r codes and the output of r codes in a same document. It also serves as a text editor where we can write our analysis. The document that you are looking at now is an example of an Rmarkdown document. It also has a pdf version and aMicrosoft Word version. The Rmarkdown version of this AYU is here

Follow these steps to create your Rmarkdown document.

- Step 1. Download a template Rmarkdown at this link
- Step 2. Open the download file with Rstudio
- Step 3. For each question, there is an R section where you will include the R codes to answer the questions.
- Step 4. Once you finish answering all the question. Click to `Knit` then choose `Knit to PDF` to generate the PDF version of the answers.
- Step 5. Submit the PDF to Canvas.