qbs124_hw3_gibran

Gibran Erlangga

4/16/2022

Question 1

(10 points). Referring to the R function cdf.dyn the cost of overlooking a patient with high blood pressure that may lead to a stroke is $50K and the cost of wrong decision on the high risk health situations is $25K. Use the binormal curve to find the optimal decision on the threshold that minimizes the cost. Display the binormal ROC and the vertical and horizontal lines for the implied optimal false positive and sensitivity rates. Display axis(side=3) to show the threshold values.

```
# plot binormal ROC
# do log-transformation, show the actual value at the optimal value on the plot
# import data and set-up

n=100
X=exp(rnorm(n,mean=1,sd=.1))*75
Y=exp(rnorm(1.5*n,mean=.9,sd=.08))*75
nY=rep(0,1.5*n)
nX = rep(1, n)

X_data = data.frame(cbind(nX, X))
names(X_data) = c('n', 'val')
Y_data = data.frame(cbind(nY, Y))
names(Y_data) = c('n', 'val')

data <- rbind(X_data, Y_data)
a = data$n
b = data$val

par(mfrow = c(1, 1), mar = c(4.5, 4.5, 3, 1), cex.main = 1.5, cex.lab = 1.5)
n = length(a)
ind = rep(0, n)
ind[a == 0] = 1
XY = data$val
ind = ind[order(XY)]
XY = XY[order(XY)]
sens = comp.spec = rep(0, n)
AUC = 0

for (i in 1:n) {
sens[i] = sum(XY < XY[i] & ind == 1)/sum(ind == 1)
comp.spec[i] = sum(XY < XY[i] & ind == 0)/sum(ind == 0)
if (i > 1)
```

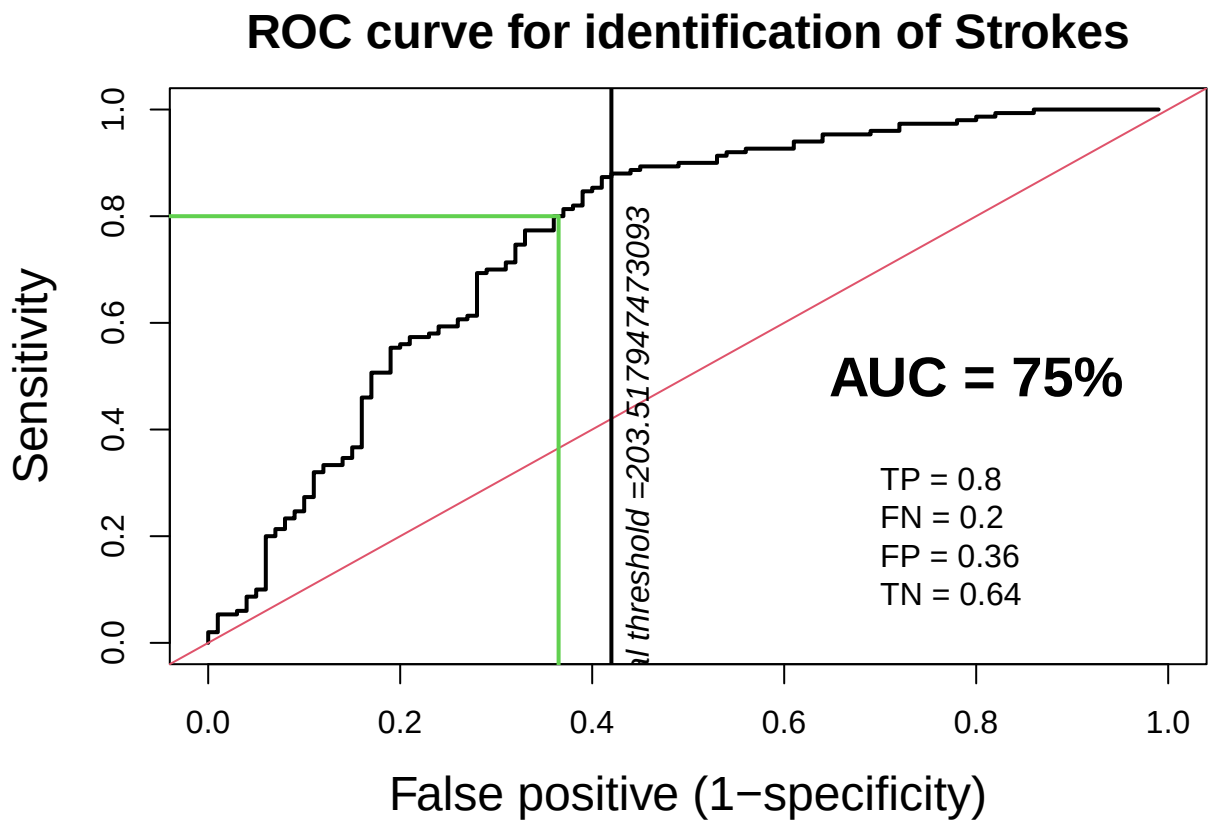
```
AUC = AUC + sens[i] * (comp.spec[i] - comp.spec[i - 1])
}
plot(comp.spec, sens, type = "s", xlim = c(0, 1), ylim = c(0, 1), xlab = "False positive (1-specificity)
ylab = "Sensitivity", lwd = 2, main = "ROC curve for identification of Strokes")
segments(-1, -1, 2, 2, col = 2)
text(0.8, 0.5, paste("AUC = ", round(AUC * 100), "%", sep = ""), cex = 1.75,
font = 2)
a = abs(sens - 0.8)

c.8 = mean(comp.spec[a == min(a)])
lines(x = c(-1, c.8, c.8), y = c(0.8, 0.8, -2), col = 3, lwd = 2)
text(0.7, 0.2, paste("TP =", 0.8, "\nFN =", 1 - 0.8, "\nFP =", round(c.8, 2), "\nTN =", 1 - round(c.8, 2
th.8 = 10**mean(XY[a == min(a)])
print(paste("FP.8=", c.8, " Th.8=", th.8, "thousand dollars"))
```

```
## [1] "FP.8= 0.365  Th.8= 2.3969368998047e+199 thousand dollars"
```

```
tot.cost = rep(NA, n)
for (i in 1:n) {
sens[i] = sum(XY < XY[i] & ind == 1)/sum(ind == 1)
comp.spec[i] = sum(XY < XY[i] & ind == 0)/sum(ind == 0)
tot.cost[i] = 50 * (1 - sens[i]) + 25 * comp.spec[i]
}
iopt = which(tot.cost == min(tot.cost))
opt.inc = XY[iopt]
print(paste("Optimal threshold =", opt.inc))
```

```
## [1] "Optimal threshold = 203.517947473093"
```

```
segments(comp.spec[iopt], -1, comp.spec[iopt], 2, lwd = 2)
text(comp.spec[iopt] + 0.03, 0.3, paste("Optimal threshold =", opt.inc, "",
sep = ""), srt = 90, cex = 1, font = 3)
```



Question 2

2. (10 points). Using the data set HeightWeight.csv for 25,000 Korean teenagers estimate the weight of an individual with height 70 inches. Display the scatterplot and the prediction along with the ś1.96$_{y|x}$ interval using the sample estimates from the Normal regression section (no lm function).

```
# import data
hw_data <- read.csv('HeightWeight.csv')
head(hw_data)
```

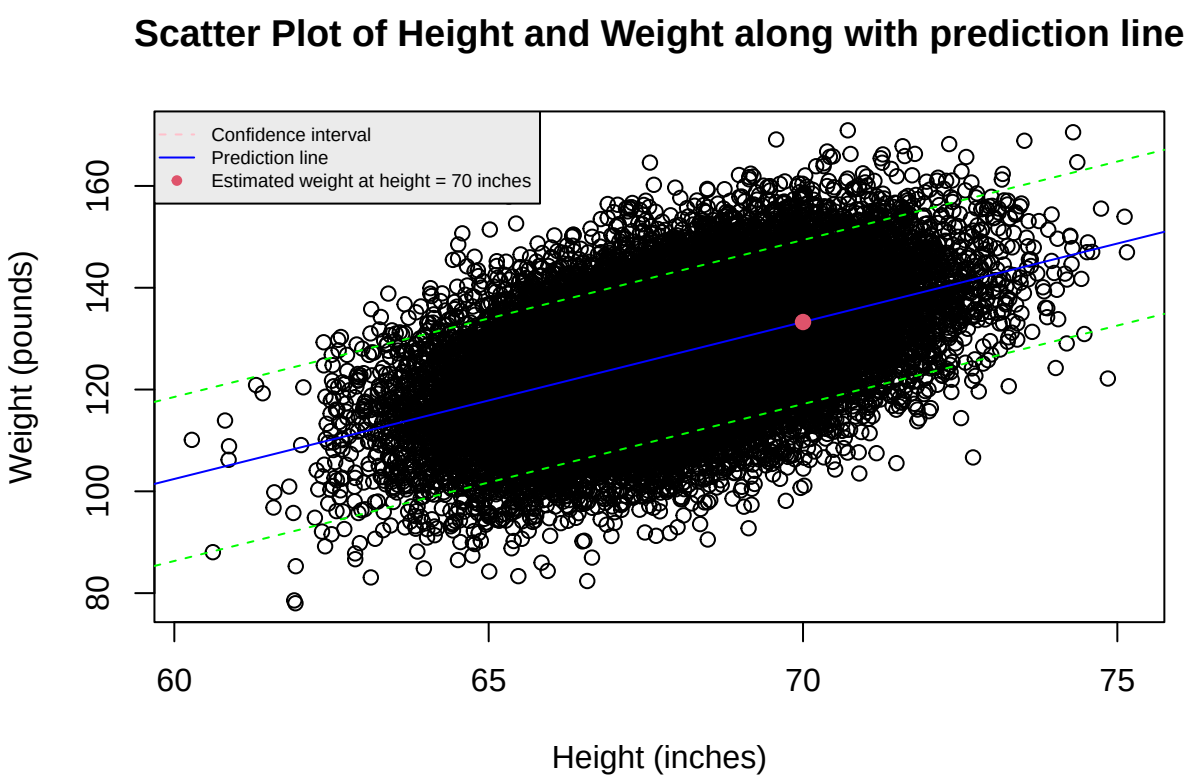
```
##   Height.Inches. Weight.Pounds.
## 1       65.78331       112.9925
## 2       71.51521       136.4873
## 3       69.39874       153.0269
## 4       68.21660       142.3354
## 5       67.78781       144.2971
## 6       68.69784       123.3024
```

```
# in this case, Y = weight, X = height, with x = 70 inches
cond_mean <- function(y, X, x) {
  mean_X = mean(X)
  std_X = sqrt(var(X))
  mean_y = mean(y)
  std_y = sqrt(var(y))
```

```
  corr_coef = cor(y, X)
  return (mean_y + corr_coef*(std_y/std_X)*(x-mean_X))
}

estimated_weight = cond_mean(hw_data$Weight.Pounds., hw_data$Height.Inches., 70)
estimated_weight_1above = cond_mean(hw_data$Weight.Pounds., hw_data$Height.Inches., 71)
slope=(estimated_weight_1above-estimated_weight)/1
estimated_inter = cond_mean(hw_data$Weight.Pounds., hw_data$Height.Inches., 0)
RO=cor(hw_data$Weight.Pounds., hw_data$Height.Inches.)
a<-sqrt((1-RO)*(sd(hw_data$Weight.Pounds.)^2))
b<-a*1.96
upper<-(estimated_inter+b)
lower<-(estimated_inter-b)

paste('Estimated weight of Korean individual with height of 70 inches is', round(estimated_weight, 3),
```

```
## [1] "Estimated weight of Korean individual with height of 70 inches is 133.268 pounds."
```

```
plot(hw_data$Height.Inches., hw_data$Weight.Pounds.,
     xlab = 'Height (inches)',
     ylab = 'Weight (pounds)',
     main = 'Scatter Plot of Height and Weight along with prediction line')

# linear prediction line
abline(estimated_inter,slope, col = "blue")

# confidence interval
abline(upper,slope, col = "green", lty=2)
abline(lower,slope, col = "green", lty=2)
points(70, estimated_weight, pch=19, col=2)
legend('topleft',
       c('Confidence interval', 'Prediction line',
         'Estimated weight at height = 70 inches'),
       bg='gray92',
       lty=c(2, 1, NA),
       pch=c(NA, NA, 19),
       col=c('pink', 'blue', 2),
       cex=0.6
       )
```



Question 3

(5 points). Correct the wrong conclusion on improving the revenue by hiring more truck drives by running a linear regression of revenue on time and the number of truck drivers. Explain why it helps.

Answer: The i.i.d assumption does not get valid as we have value that grows over time. As we have value growing time we use the formula coefficient of termination in regression $R^2 = 1 - \frac{\sum x_i^2}{\sum (y_i - \overline{y})^2}$ and $\sum x_i^2$ as numerator, is the vertical distance square, while in the denominator, which is so initally there was a huge variance which is not the variance of y as y grows in time, so here the denominator increases and R^2 approaching 1.

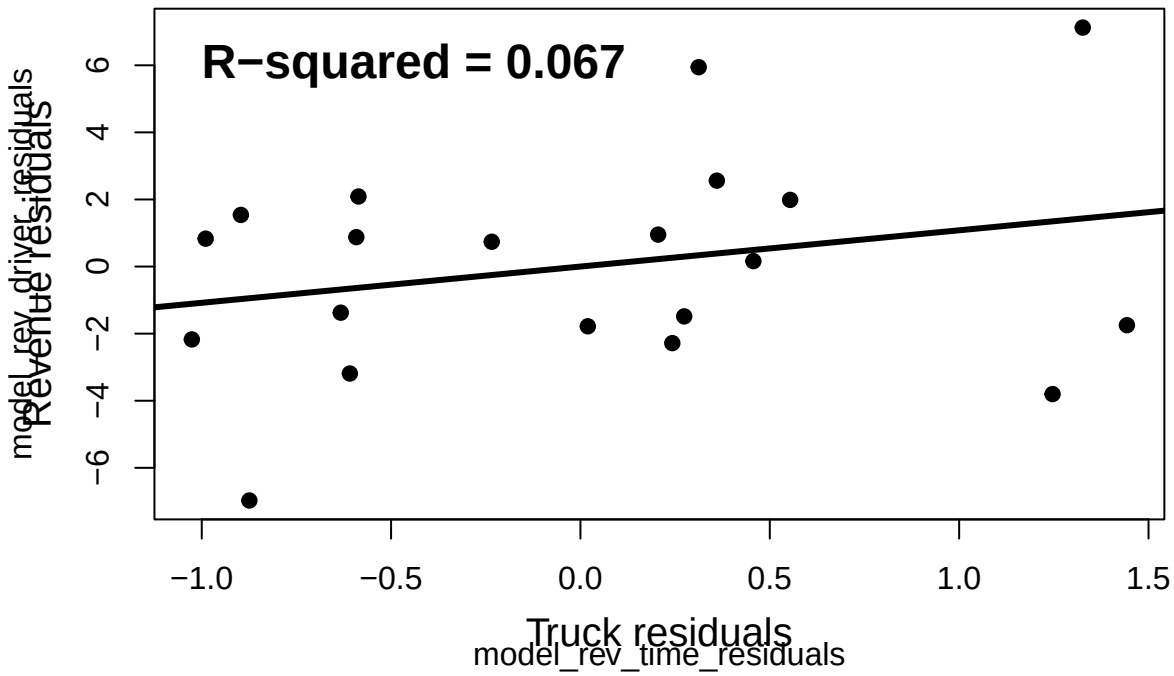
To fix this problem, x and y has to be i.i.d or fluctuating from the mean of value. Therefore, we can exclude time from the analysis. I computed the trend and revenue as the function of time and take residuals, take difference between revenue and trend, and same for truck drivers and then correlate both the residuals. The correlation coefficient is positive, and also the coefficient of termination is 0.067.

```
# import data
truck_data <- read.csv('truckR.data.csv')
truck_data$driver_id = 1:nrow(truck_data)
head(truck_data)
```

```
##     truc.dr  revenue driver_id
## 1 0.4380666 199.5622         1
## 2 1.5074743 201.2457         2
## 3 2.4587882 207.1027         3
```

```
## 4 1.4478681 200.2298         4
## 5 3.1957828 209.4537         5
## 6 3.4301239 209.8960         6
```

```
# linear regression on revenue given time
model_rev_time_residuals = lm(truc.dr ~ driver_id,data=truck_data)$residuals
# linear regression on revenue given number of truck drivers
model_rev_driver_residuals = lm(revenue ~ driver_id,data=truck_data)$residuals

plot(model_rev_time_residuals, model_rev_driver_residuals, cex=1, pch=19)
abline(lsfit(x=model_rev_time_residuals,y=model_rev_driver_residuals),lwd=3)
mtext(side=1,"Truck residuals",cex=1.25,line=2.5)
mtext(side=2,"Revenue residuals",cex=1.25,line=2.5)
r2=cor(model_rev_time_residuals,model_rev_driver_residuals)**2
text(-1,6,paste("R-squared =",round(r2,3)),adj=0,font=2,cex=1.5)
```



Question 4

(5 points). Provide possible explanation for the negative sign at Nose despite its positive correlation with Height.

```
da=read.csv("HeightFootNose.csv")
par(mfrow=c(1,3),mar=c(3.5,3.5,3,1))
```

```r
# foot vs height
plot(da$Foot,da$Height,xlab="",ylab="")
title(paste("Height versus length of foot, R =",round(cor(da$Foot,da$Height),2)))
mtext(side=1,"Foot, inches",cex=1.25,line=2.5)
mtext(side=2,"Height, inches",cex=1.25,line=2.25)
print(summary(lm(Height~Foot,data=da)))
```

```
##
## Call:
## lm(formula = Height ~ Foot, data = da)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -8.236 -2.001 -0.026  2.118  8.399
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 54.71283    0.96820   56.51   <2e-16 ***
## Foot         1.52460    0.09531   16.00   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.063 on 495 degrees of freedom
## Multiple R-squared:  0.3408, Adjusted R-squared:  0.3394
## F-statistic: 255.9 on 1 and 495 DF,  p-value: < 2.2e-16
```

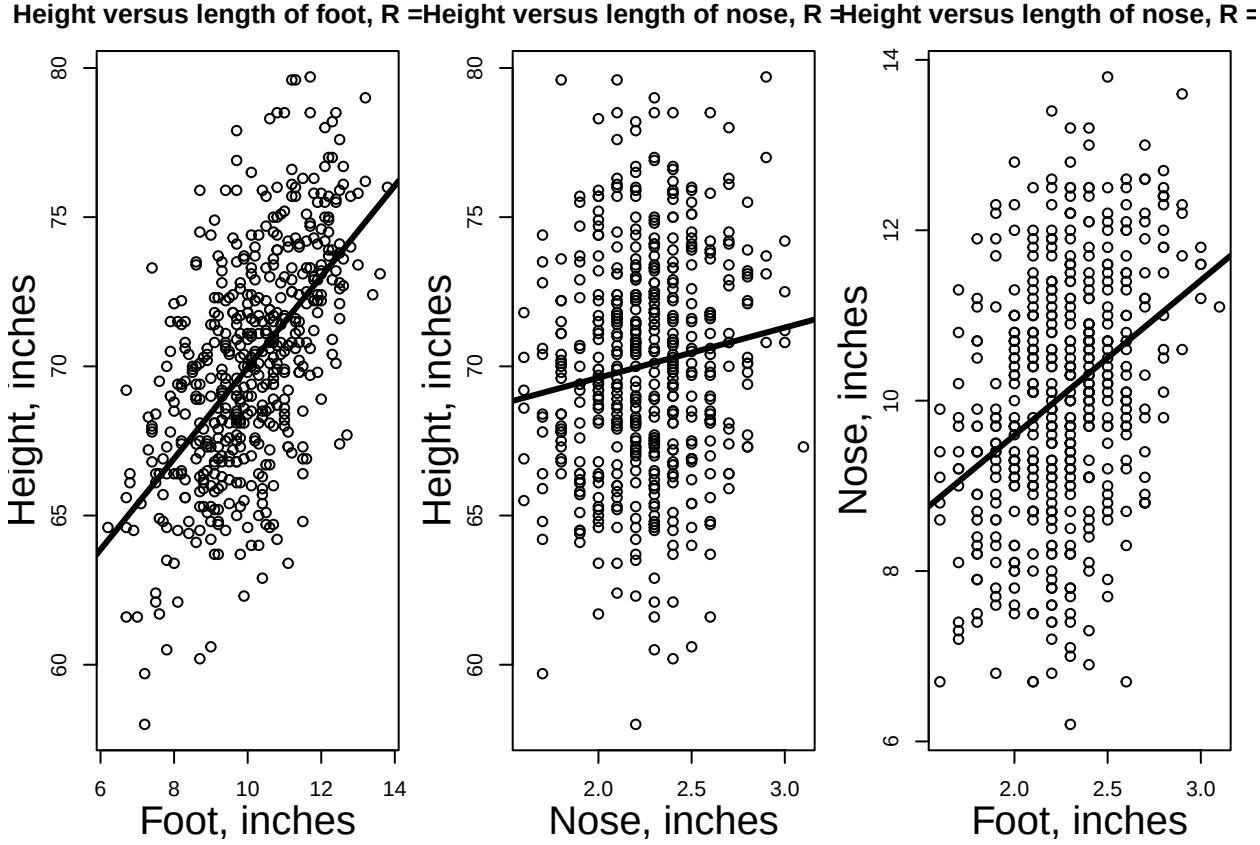
```r
abline(lsfit(x=da$Foot,y=da$Height),lwd=3)

# nose vs height
plot(da$Nose,da$Height,xlab="",ylab="")
title(paste("Height versus length of nose, R =",round(cor(da$Nose,da$Height),2)))
mtext(side=1,"Nose, inches",cex=1.25,line=2.5)
mtext(side=2,"Height, inches",cex=1.25,line=2.25)
print(summary(lm(Height~Nose,data=da)))
```

```
##
## Call:
## lm(formula = Height ~ Nose, data = da)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.9534  -2.7217  -0.0217   2.5687  10.3197
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  66.2514     1.3331  49.696  < 2e-16 ***
## Nose          1.6827     0.5869   2.867  0.00432 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.742 on 495 degrees of freedom
## Multiple R-squared:  0.01634,    Adjusted R-squared:  0.01435
## F-statistic: 8.221 on 1 and 495 DF,  p-value: 0.004316
```

```r
abline(lsfit(x=da$Nose,y=da$Height),lwd=3)

# nose vs foot
plot(da$Nose,da$Foot,xlab="",ylab="")
title(paste("Height versus length of nose, R =",round(cor(da$Nose,da$Foot),2)))
mtext(side=1,"Foot, inches",cex=1.25,line=2.5)
mtext(side=2,"Nose, inches",cex=1.25,line=2.25)
print(summary(lm(Nose~Foot,data=da)))
```

```
##
## Call:
## lm(formula = Nose ~ Foot, data = da)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.64251 -0.19246  0.00051  0.17894  0.77179
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.534454   0.084489  18.162   <2e-16 ***
## Foot        0.071510   0.008317   8.598   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2673 on 495 degrees of freedom
## Multiple R-squared:  0.1299, Adjusted R-squared:  0.1282
## F-statistic: 73.92 on 1 and 495 DF,  p-value: < 2.2e-16
```

```r
abline(lsfit(x=da$Nose,y=da$Foot),lwd=3)
```




```
# set foot and nose as predictors and height as dependent variable
print(summary(lm(Height~Foot+Nose,data=da)))
```

```
##
## Call:
## lm(formula = Height ~ Foot + Nose, data = da)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.5211 -1.9269  0.0138  2.0121  8.3635
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  56.6307     1.2436  45.537   <2e-16 ***
## Foot          1.6140     0.1017  15.874   <2e-16 ***
## Nose         -1.2499     0.5125  -2.439   0.0151 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.048 on 494 degrees of freedom
## Multiple R-squared:  0.3486, Adjusted R-squared:  0.346
## F-statistic: 132.2 on 2 and 494 DF,  p-value: < 2.2e-16
```

```
print(c(sd(da$Height),sd(da$Foot),sd(da$Nose)))
```

```
## [1] 3.7691664 1.4431759 0.2863013
```

```
f=da$Foot-mean(da$Foot)
n=da$Nose-mean(da$Nose)
h=da$Height
print(summary(lm(h~f+n,data=da)))
```

```
##
## Call:
## lm(formula = h ~ f + n, data = da)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.5211 -1.9269  0.0138  2.0121  8.3635
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  70.0435     0.1367 512.274   <2e-16 ***
## f             1.6140     0.1017  15.874   <2e-16 ***
## n            -1.2499     0.5125  -2.439   0.0151 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.048 on 494 degrees of freedom
## Multiple R-squared:  0.3486, Adjusted R-squared:  0.346
## F-statistic: 132.2 on 2 and 494 DF,  p-value: < 2.2e-16
```

Answer: The graphics above show the relationship between Foot vs Height, Nose vs Height and Foot vs Nose (as independent vs dependent variable). We can observe that when we set foot as constant, tall people have shorter nose or taller, like with respect of the foot size of them all being same, that is constant.

Question 5

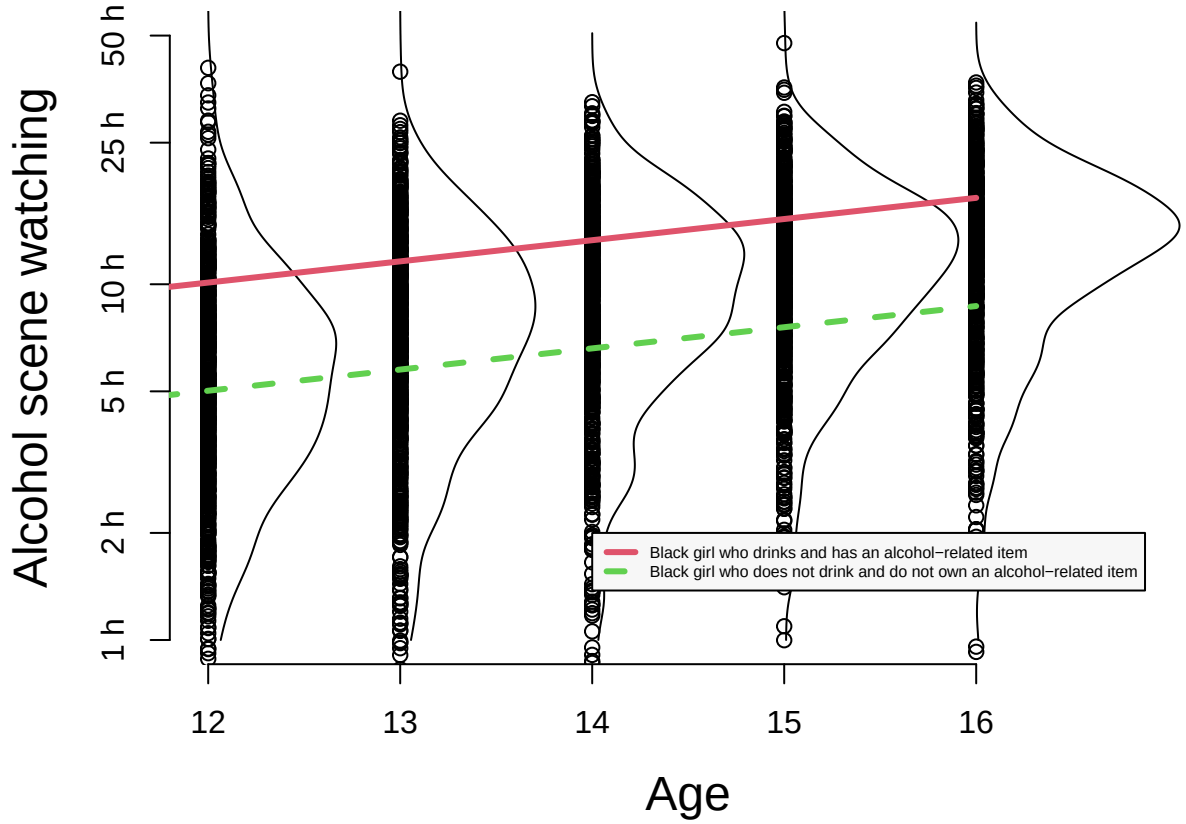
(10 points). Display the time watching of alcohol scenes as a function of age for a black girl who drinks, has an alcohol related item, with high income and high parents' education, and has good grades as in function kidsdrink(job=2). To contrast, display the same girl but who does not drink and does not have an alcohol related item. Compute and display the the effect of drinking and having an alcohol related item.

```
kidsdrink_data <-read.csv("kidsdrink.csv")

d<-kidsdrink_data
d=cbind(d,log(1/60^2+d$alcm))
    names(d)[ncol(d)]="logalcm"
    o=lm(logalcm~drink+age+boy+race+alcbr+pared+inc+grade,data=d)
    print(summary(o))
```

```
##
## Call:
## lm(formula = logalcm ~ drink + age + boy + race + alcbr + pared +
##     inc + grade, data = d)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
```

```
## -10.1467  -0.3481   0.0987   0.4663   2.1806
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.093682   0.133021  -0.704 0.481314
## drink        0.432670   0.028335  15.270  < 2e-16 ***
## age          0.137065   0.009459  14.490  < 2e-16 ***
## boy          0.048303   0.024298   1.988 0.046895 *
## race         0.266762   0.045818   5.822 6.29e-09 ***
## alcbr        0.267472   0.040170   6.658 3.16e-11 ***
## pared       -0.022162   0.027022  -0.820 0.412198
## inc          0.084440   0.037931   2.226 0.026063 *
## grade       -0.090740   0.025978  -3.493 0.000483 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7416 on 3796 degrees of freedom
## Multiple R-squared:  0.1969, Adjusted R-squared:  0.1952
## F-statistic: 116.3 on 8 and 3796 DF,  p-value: < 2.2e-16
```

```
    par(mfrow=c(1,1),mar=c(4.5,4.5,1,1),cex.lab=1.5)
    alab=c(1,2,5,10,25,50);lalab=log(alab)
    plot(d$age,d$logalcm,xlim=c(12,17),ylim=range(lalab),type="n",
         axes=F,xlab="Age",ylab="Alcohol scene watching")
    axis(side=1,12:16)
    axis(side=2,at=lalab,labels=paste(alab,"h"),srt=90)
    for(a in 12:16)
    {
        da=d$logalcm[d$age==a];n=length(da)
        points(rep(a,n),da)
        den=density(da,from=0)
        lines(a+1.25*den$y,den$x)
    }
    x=11:16
    a=coef(o)

    lines(x,a[1]+a[2]+a[3]*x+a[6]+a[7]+a[8],col=2,lwd=3)
    lines(x,a[1]+a[3]*x+a[7]+a[8],col=3,lwd=3,lty=2)
    legend(14,log(2),c("Black girl who drinks and has an alcohol-related item",
                       "Black girl who does not drink and do not own an alcohol-related item"),col=2:3
```


Alcohol scene watching
1 h
2 h
5 h
10 h
25 h
50 h
12
13
14
15
16
Age
Black girl who drinks and has an alcohol−related item
Black girl who does not drink and do not own an alcohol−related item