Assignment 2 - MA5755 (Data analytics and Visualization) Roll no: ME21M038

15/04/2021

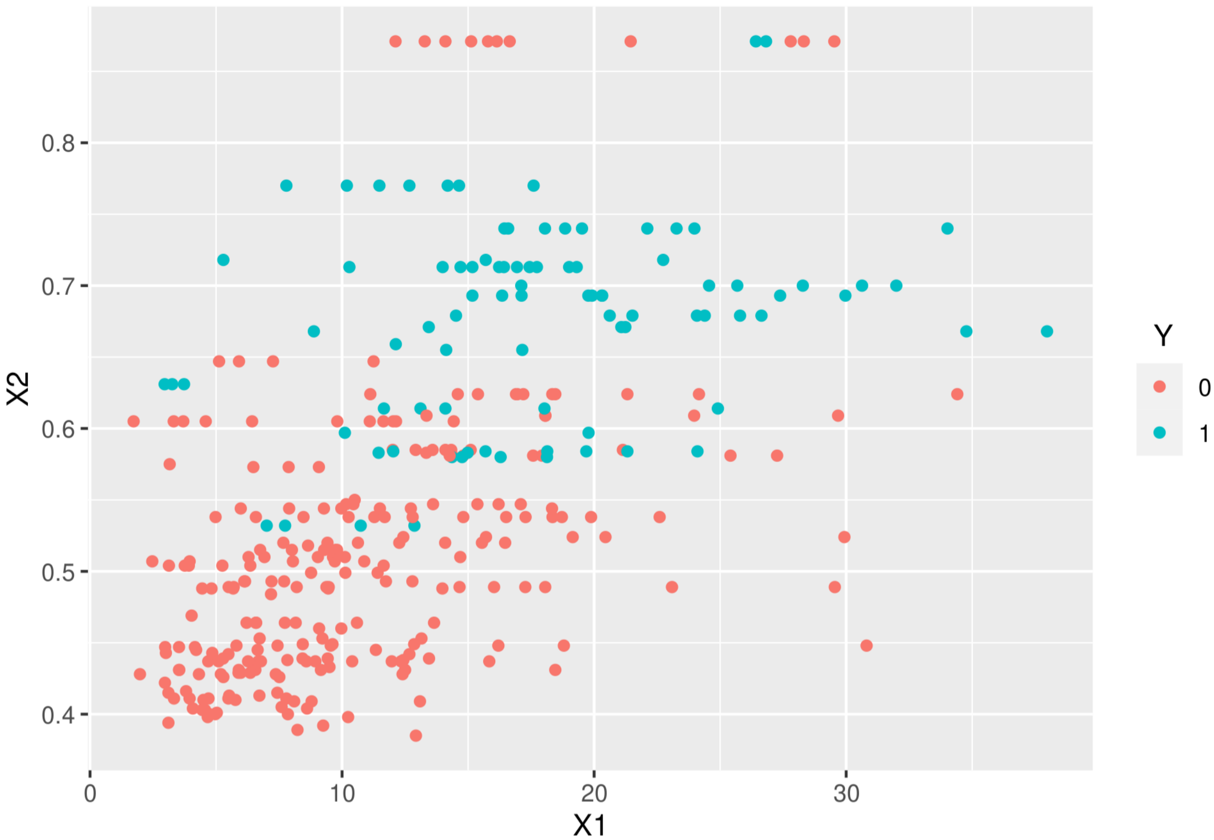
Problem 1

```
df = read.csv("datacl.csv")

set.seed(2021)
x = data.frame(df$x1,df$x2)
y = data.frame(df$y)
train = sample(1:nrow(x),2/3*nrow(x))
test = (-train)
y.test = y[test]
df2  = data.frame(x[train,1],x[train,2],y[train,1])

colnames(df2)[1] = "X1"
colnames(df2)[2] = "X2"
colnames(df2)[3] = "Y"

# plot
ggplot(df2,aes(x=X1,y=X2))+geom_point(aes(col=factor(Y)))+labs(color="  Y")
```



Logistic regression model

```
sample = sample(c(TRUE,FALSE),nrow(df),replace=TRUE,prob=c(2/3,1/3))
train = df[sample,]
test = df[!sample,]

# logistic regression model
lr.model = glm(formula = y~x1+x2,family = "binomial",data = train)
summary(lr.model)$coefficients
```

```
##                 Estimate Std. Error   z value     Pr(>|z|)
## (Intercept) -12.03286668 1.35848390 -8.857570 8.177723e-19
## x1            0.08267754 0.02756686  2.999165 2.707210e-03
## x2           16.16350642 2.26517012  7.135670 9.631686e-13
```

Hypothesis testing

```
tx1=t.test(train$x1~train$y,alt="two.sided",var.eq=FALSE,paired=FALSE)
tx1
```

Hypothesis testing for X1

```
##
##  Welch Two Sample t-test
##
## data:  train$x1 by train$y
```

```
## t = -10.186, df = 124.35, p-value < 2.2e-16
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
##  -10.221144  -6.895186
## sample estimates:
## mean in group 0 mean in group 1
##        10.77525        19.33342
```

```
tx2=t.test(train$x2~train$y)
tx2
```

Hypothesis testing for X2

```
##
##  Welch Two Sample t-test
##
## data:  train$x2 by train$y
## t = -18.675, df = 184.97, p-value < 2.2e-16
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
##  -0.1935863 -0.1565933
## sample estimates:
## mean in group 0 mean in group 1
##       0.5106444       0.6857342
```

Leave one out cross validation

```
# y~x1
glm.fit = glm(data = df,y~x1)
cv.error1 = cv.glm(df,glm.fit)$delta[1]

# y~x2
glm.fit = glm(data = df,y~x2)
cv.error2 = cv.glm(df,glm.fit)$delta[1]

# y~x1+x2
glm.fit = glm(data = df,y~x1+x2)
cv.error12 = cv.glm(df,glm.fit)$delta[1]
```

Pediction error

for y = x1 model 14.3777583 %

for y = x2 model 11.607406 %

for y = x1+x2 model 11.1079124 %

y=x1+x2 is the best model in terms of accuracy. Model accuracy = 88.8920876 %

Problem 2

```
# linear discriminant analysis model

train$X = NULL
test$X = NULL
lda.model = lda(y~x1+x2,data = train)
pred = predict(lda.model,test)

lda = confusionMatrix(pred$class,test$y)
lda_err = (lda$`0`[2]+lda$`1`[1])/(lda$`0`[1]+lda$`0`[2]+lda$`1`[1]+lda$`1`[2])
```

Linear Discriminant analasis Confusion Matrix

```
##      0  1
## 0 111  5
## 1  18 36
```

Missclassification Error is equal to 13.5294118 %

```
# quadratic discriminant analysis

qda.model = qda(y~x1+x2,data = train)
pred1 = predict(qda.model,test)

qda = confusionMatrix(pred1$class,test$y)
qda_err = (qda$`0`[2]+qda$`1`[1])/(qda$`0`[1]+qda$`0`[2]+qda$`1`[1]+qda$`1`[2])
```

Quadratic Discriminant analasis Confusion Matrix

```
##      0  1
## 0 111  5
## 1  16 38
```

Missclassification Error is equal to 13.5294118 %

Two obeservation of LDA False negative is converted into True Positive in QDA so QDA is better in terms of accuracy

prediction accuracy LDA 86.4705882 % and QDA 87.6470588 %

Problem 3

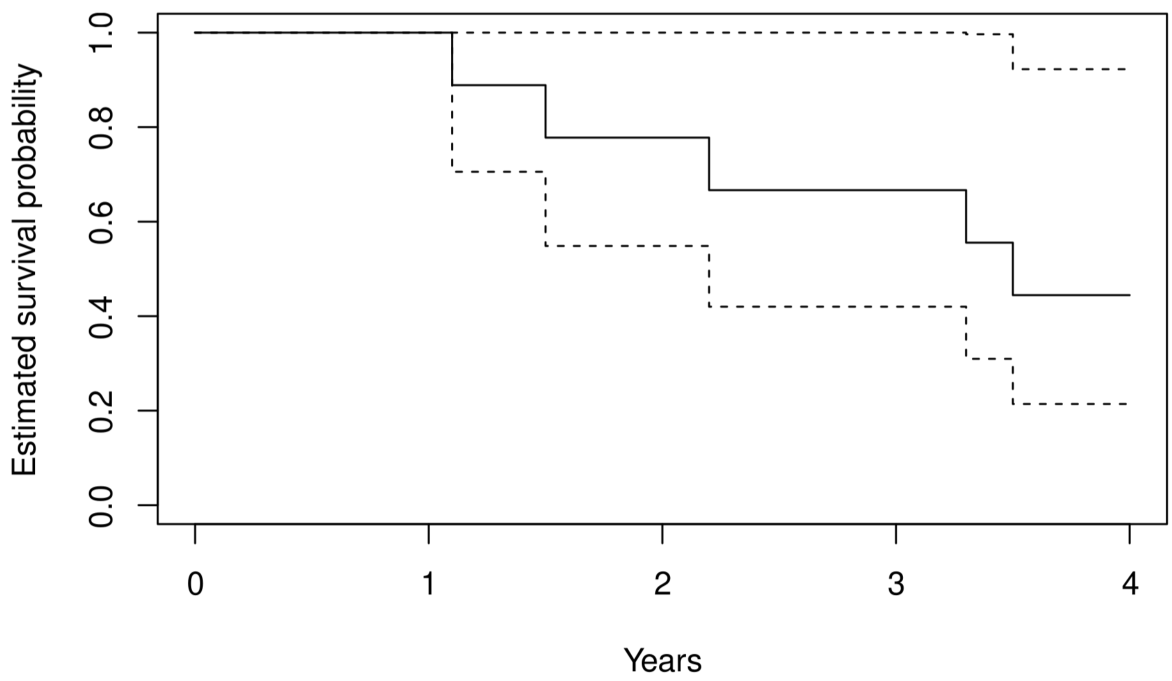
A)

```
company = c("A","A","A","A","A","B","B","B","B","B")
phone = c("A1","A2","A3","A4","A5","B1","B2","B3","B4","B5")
t = c(1.1,2.2,3.3,4,4,1,1.5,3.5,4,4)
eve = c(1,1,1,0,0,0,1,1,0,0)
Surv(t,eve)
```

```
##  [1] 1.1  2.2  3.3  4.0+ 4.0+ 1.0+ 1.5  3.5  4.0+ 4.0+
```

```
sobj = data.frame(company,phone,t,eve)

# survfit object builds
km = survfit(Surv(t,eve)~1,data=sobj)
plot(km,xlab="Years",ylab="Estimated survival probability")
```




```
fit.surv = surv_fit(Surv(t,eve)~company,data=sobj)
median = surv_median(fit.surv)
```

Survival time median of company B = 3.5 years is more than company A = 3.3 years

B)

```
company = c("A","A","A","A","A","C","C","C","C","C")
phone = c("A1","A2","A3","A4","A5","C1","C2","C3","C4","C5")
t = c(1.1,2.2,3.3,4,4,1,1.5,3.5,2,5)
eve = c(1,1,1,0,0,0,1,1,0,0)

sobj_new = data.frame(company,phone,t,eve)
fit.surv = surv_fit(Surv(t,eve)~1,data = sobj_new)
median = surv_median(fit.surv)
```

Yes we can compare company A and C

Survival time median of company C = NA years is more than company A = 3.5 years

C)

Survival Probability at 2.25 years 64.8 %

Survival Probability at 3.9 years 38.9 %

Problem 4

A) This kind of censoring can cause bias because of loss of data so that censoring mechanism affect survival time in this case.-> Denpendent

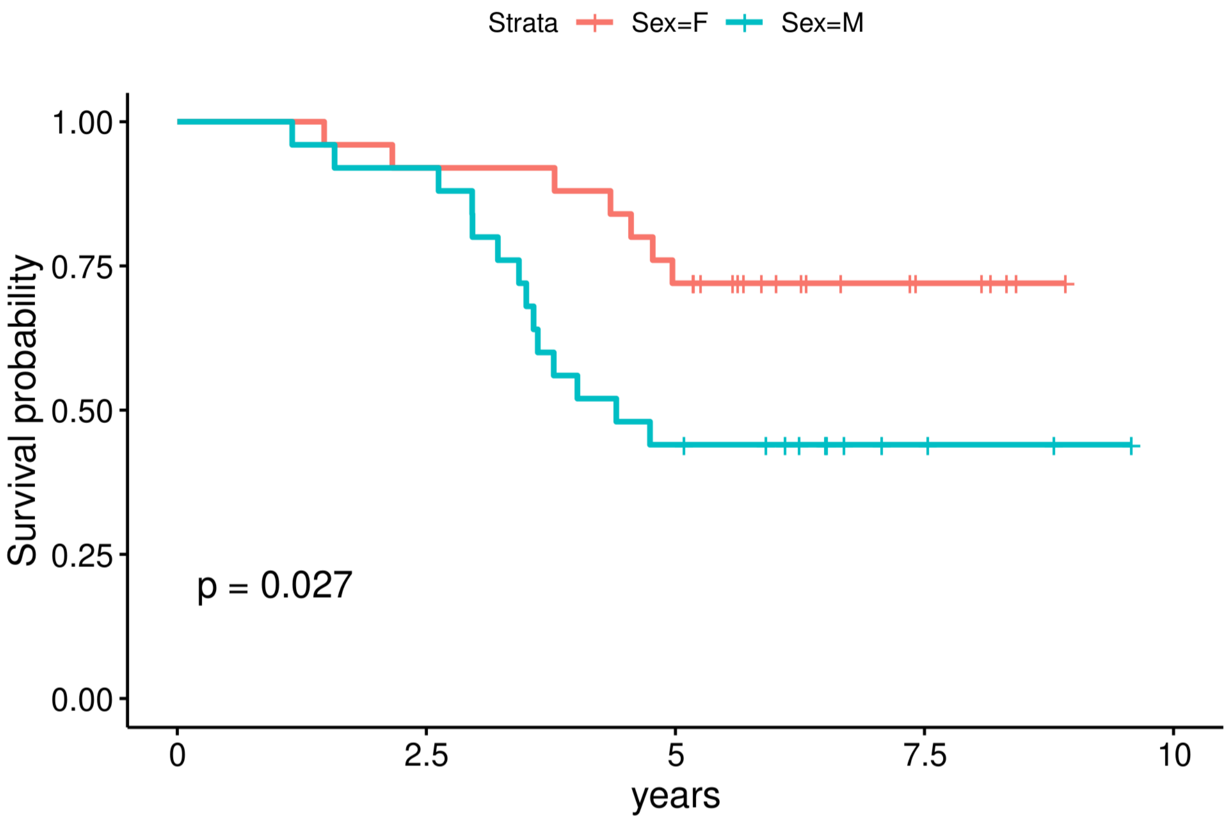
B) Above 99 years patients are at high risk of death even though they are treated well so survival time for censored and uncensored wont affect esimate of time to event. -> Independent

C) This condition is same as previous patient is at high risk so censoring wont affect estimate of time to event. -> Independent

Problem 5

```
df = read.csv("datasur.csv")
df$X = NULL
x = model.matrix(survival.status~.,df)[,-1]
y = df$survival.status

# Kaplan Meir curve
fit.sex = survfit(Surv(time,event = survival.status)~Sex,data=df)
ggsurvplot(fit.sex,xlab="years",pval=TRUE)
```




```
# logrank test
test.logrank = survdiff(Surv(df$time,event = df$survival.status)~df$Sex)
test.logrank
```

```
## Call:
## survdiff(formula = Surv(df$time, event = df$survival.status) ~
##     df$Sex)
##
##             N Observed Expected (O-E)^2/E (O-E)^2/V
## df$Sex=F 25        7    11.99      2.07      4.89
## df$Sex=M 25       14     9.01      2.76      4.89
##
##  Chisq= 4.9  on 1 degrees of freedom, p= 0.03
```

pvlaue = 0.03 which is less than 0.05 there is significant difference

```
# Cox proportional hazard model
fit.cox = coxph(Surv(df$time,event=df$survival.status)~df$Sex+df$Age)
summary(fit.cox)
```

```
## Call:
## coxph(formula = Surv(df$time, event = df$survival.status) ~ df$Sex +
##     df$Age)
##
##   n= 50, number of events= 21
##
##               coef  exp(coef) se(coef)      z Pr(>|z|)
## df$SexM   6.0227  412.7062   1.8852  3.195 0.001399 **
```

```
## df$Age   -2.4546    0.0859   0.6990 -3.512 0.000445 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##         exp(coef) exp(-coef) lower .95 upper .95
## df$SexM  412.7062   0.002423  10.25621 16607.155
## df$Age     0.0859  11.641919   0.02183     0.338
##
## Concordance= 0.993  (se = 0.005 )
## Likelihood ratio test= 131.9  on 2 df,   p=<2e-16
## Wald test            = 12.45  on 2 df,   p=0.002
## Score (logrank) test = 58.91  on 2 df,   p=2e-13
```

```
fit.cox$coefficients
```

```
##   df$SexM    df$Age
##  6.022736 -2.454612
```

Probability $= 6.022736 \times 10^{-6}$