**classify that a person had an affair or not**

Data description :

A data frame containing 601 observations on 9 variables.

affairs : numeric. How often engaged in extramarital sexual intercourse during the past year?

gender : factor indicating gender.

age : numeric variable coding age in years: 17.5 = under 20, 22 = 20–24, 27 = 25–29, 32 = 30–34, 37 = 35–39, 42 = 40–44, 47 = 45–49, 52 = 50–54, 57 = 55 or over.

yearsmarried : numeric variable coding number of years married: 0.125 = 3 months or less, 0.417 = 4–6 months, 0.75 = 6 months–1 year, 1.5 = 1–2 years, 4 = 3–5 years, 7 = 6–8 years, 10 = 9–11 years, 15 = 12 or more years.

children : factor. Are there children in the marriage?

religiousness : numeric variable coding religiousness: 1 = anti, 2 = not at all, 3 = slightly, 4 = somewhat, 5 = very.

education : numeric variable coding level of education: 9 = grade school, 12 = high school graduate, 14 = some college, 16 = college graduate, 17 = some graduate work, 18 = master's degree, 20 = Ph.D., M.D., or other advanced degree.

occupation : numeric variable coding occupation according to Hollingshead classification (reverse numbering).

rating : numeric variable coding self rating of marriage: 1 = very unhappy, 2 = somewhat unhappy, 3 = average, 4 = happier than average, 5 = very happy.

**Ans:**

> affairs <- read\_csv("affairs.csv")

> summary(affairs)

affairs gender age yearsmarried

Min. : 0.000 Length:601 Min. :17.50 Min. : 0.125

1st Qu.: 0.000 Class :character 1st Qu.:27.00 1st Qu.: 4.000

Median : 0.000 Mode :character Median :32.00 Median : 7.000

Mean : 1.456 Mean :32.49 Mean : 8.178

3rd Qu.: 0.000 3rd Qu.:37.00 3rd Qu.:15.000

Max. :12.000 Max. :57.00 Max. :15.000

children religiousness education occupation

Length:601 Min. :1.000 Min. : 9.00 Min. :1.000

Class :character 1st Qu.:2.000 1st Qu.:14.00 1st Qu.:3.000

Mode :character Median :3.000 Median :16.00 Median :5.000

Mean :3.116 Mean :16.17 Mean :4.195

3rd Qu.:4.000 3rd Qu.:18.00 3rd Qu.:6.000

Max. :5.000 Max. :20.00 Max. :7.000

rating

Min. :1.000

1st Qu.:3.000

Median :4.000

Mean :3.932

3rd Qu.:5.000

Max. :5.000

> sum(is.na(affairs))

[1] 0

**No NA values so no need of Imputation.**

> str(affairs)

Classes ‘spec\_tbl\_df’, ‘tbl\_df’, ‘tbl’ and 'data.frame': 601 obs. of 9 variables:

$ affairs : num 0 0 0 0 0 0 0 0 0 0 ...

$ gender : chr "male" "female" "female" "male" ...

$ age : num 37 27 32 57 22 32 22 57 32 22 ...

$ yearsmarried : num 10 4 15 15 0.75 1.5 0.75 15 15 1.5 ...

$ children : chr "no" "no" "yes" "yes" ...

$ religiousness: num 3 4 1 5 2 2 2 2 4 4 ...

$ education : num 18 14 12 18 17 17 12 14 16 14 ...

$ occupation : num 7 6 1 6 6 5 1 4 1 4 ...

$ rating : num 4 4 4 5 3 5 3 4 2 5 ...

**Column gender and children are char strings so need to convert into 0 and 1.**

> affairs$gender <- ifelse(affairs$gender == "male",0,1)

> affairs$children <- ifelse(affairs$children == "no",0,1)

> str(affairs)

Classes ‘spec\_tbl\_df’, ‘tbl\_df’, ‘tbl’ and 'data.frame': 601 obs. of 9 variables:

$ affairs : num 0 0 0 0 0 0 0 0 0 0 ...

$ gender : num 0 1 1 0 0 1 1 0 1 0 ...

$ age : num 37 27 32 57 22 32 22 57 32 22 ...

$ yearsmarried : num 10 4 15 15 0.75 1.5 0.75 15 15 1.5 ...

$ children : num 0 0 1 1 0 0 0 1 1 0 ...

$ religiousness: num 3 4 1 5 2 2 2 2 4 4 ...

$ education : num 18 14 12 18 17 17 12 14 16 14 ...

$ occupation : num 7 6 1 6 6 5 1 4 1 4 ...

$ rating : num 4 4 4 5 3 5 3 4 2 5 ...

**Columns gender and children need to be factors variables.**

**So convert into factor type.**

> affairs$gender <- factor(affairs$gender)

> affairs$children <- factor(affairs$children)

> str(affairs)

Classes ‘spec\_tbl\_df’, ‘tbl\_df’, ‘tbl’ and 'data.frame': 601 obs. of 9 variables:

$ affairs : num 0 0 0 0 0 0 0 0 0 0 ...

$ gender : Factor w/ 2 levels "0","1": 1 2 2 1 1 2 2 1 2 1 ...

$ age : num 37 27 32 57 22 32 22 57 32 22 ...

$ yearsmarried : num 10 4 15 15 0.75 1.5 0.75 15 15 1.5 ...

$ children : Factor w/ 2 levels "0","1": 1 1 2 2 1 1 1 2 2 1 ...

$ religiousness: num 3 4 1 5 2 2 2 2 4 4 ...

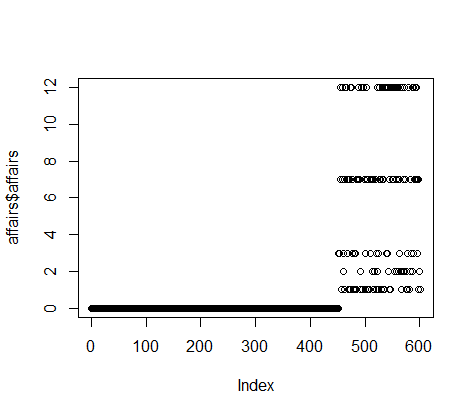
$ education : num 18 14 12 18 17 17 12 14 16 14 ...

$ occupation : num 7 6 1 6 6 5 1 4 1 4 ...

$ rating : num 4 4 4 5 3 5 3 4 2 5 ...

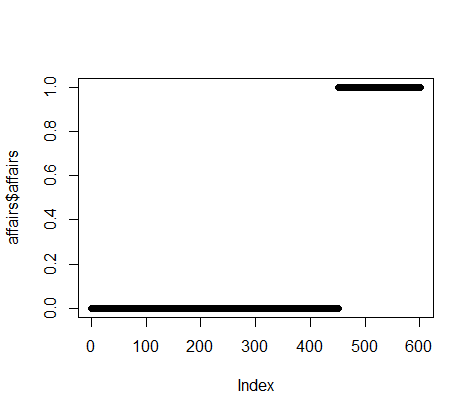
**As the column affairs is containing multiple values other than just 0 and 1,we need to narrow it down to whether he or she having affair or not?**

> plot(affairs$affairs)



**Converting all non zero values to 1.**

> affairs$affairs <- ifelse(affairs$affairs == "0",yes = 0,no = 1)



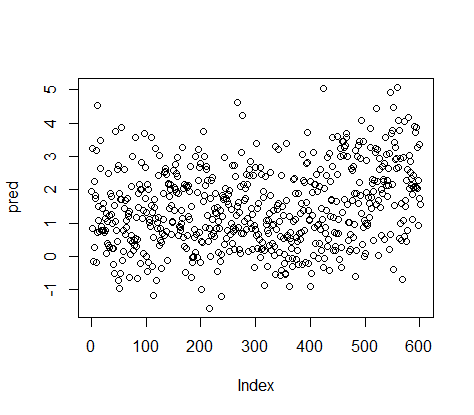
**Now as EDA part is completed, moving forward to model building.**

1. **Model 1**

> model.affairs <- lm(affairs ~ ., data = affairs)

> pred <- predict(model.affairs, affairs)

> plot(pred)



> gmodel <- glm(affairs ~ ., data = affairs, family = binomial)

> summary(gmodel)

Call:

glm(formula = affairs ~ ., family = binomial, data = affairs)

Deviance Residuals:

Min 1Q Median 3Q Max

-1.5713 -0.7499 -0.5690 -0.2539 2.5191

Coefficients:

Estimate Std. Error z value Pr(>|z|)

(Intercept) 1.65754 0.97170 1.706 0.088042 .

gender1 -0.28029 0.23909 -1.172 0.241083

age -0.04426 0.01825 -2.425 0.015301 \*

yearsmarried 0.09477 0.03221 2.942 0.003262 \*\*

children1 0.39767 0.29151 1.364 0.172508

religiousness -0.32472 0.08975 -3.618 0.000297 \*\*\*

education 0.02105 0.05051 0.417 0.676851

occupation 0.03092 0.07178 0.431 0.666630

rating -0.46845 0.09091 -5.153 2.56e-07 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 675.38 on 600 degrees of freedom

Residual deviance: 609.51 on 592 degrees of freedom

AIC: 627.51

Number of Fisher Scoring iterations: 4

**Confusion matrix table:**

> prob <- predict(gmodel,affairs,type="response")

> conf\_matrix <- table(prob>0.5, affairs$affairs)

> conf\_matrix

0 1

FALSE 435 125

TRUE 16 25

> Accuracy <- sum(diag(conf\_matrix)/sum(conf\_matrix))

> Accuracy

[1] 0.765391

**Need to create new vector to store the prob values.**

> pred\_values <- NULL

> yes\_no <- NULL

> pred\_values <- ifelse(prob>=0.5,1,0)

> yes\_no <- ifelse(prob>=0.5,"yes","no")

> affairs[,"prob"] <- prob

> affairs[,"pred\_values"] <- pred\_values

> affairs[,"yes\_no"] <- yes\_no

> head(affairs)

affairs gender age yearsmarried children religiousness education occupation rating prob pred\_values yes\_no

1 0 0 37 10.00 0 3 18 7 4 0.21673660 0 no

2 0 1 27 4.00 0 4 14 6 4 0.10611966 0 no

3 0 1 32 15.00 1 1 12 1 4 0.46639008 0 no

4 0 0 57 15.00 1 5 18 6 5 0.07964614 0 no

5 0 0 22 0.75 0 2 17 6 3 0.31942684 0 no

6 0 1 32 1.50 0 2 17 5 5 0.08502181 0 no

> table(affairs$affairs, affairs$pred\_values)

0 1

0 435 16

1 125 25

**To plot ROC curve:**

> library(ROCR)

> rocrpred<-prediction(prob,affairs$affairs)

> rocrperf<-performance(rocrpred,'tpr','fpr')

> str(rocrperf)

Formal class 'performance' [package "ROCR"] with 6 slots

..@ x.name : chr "False positive rate"

..@ y.name : chr "True positive rate"

..@ alpha.name : chr "Cutoff"

..@ x.values :List of 1

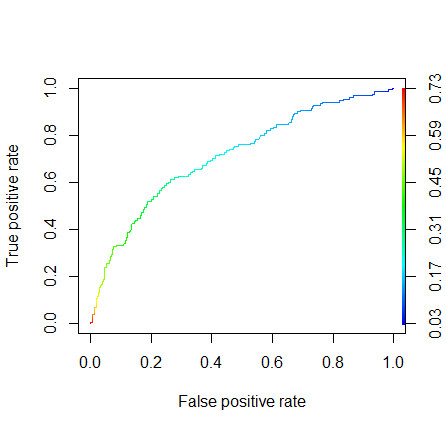
.. ..$ : num [1:570] 0 0 0.00222 0.00443 0.00443 ...

..@ y.values :List of 1

.. ..$ : num [1:570] 0 0.00667 0.00667 0.00667 0.01333 ...

..@ alpha.values:List of 1

.. ..$ : num [1:570] Inf 0.725 0.709 0.709 0.696 ...

> plot(rocrperf,colorize=T,text.adj=c(-0.2,1.7))

Adjusting Cut off Values:

> rocr\_cutoff <- data.frame(cut\_off = rocrperf@alpha.values[[1]],fpr=rocrperf@x.values,tpr=rocrperf@y.values)

> colnames(rocr\_cutoff) <- c("cut\_off","FPR","TPR")

> View(rocr\_cutoff)

> head(rocr\_cutoff)

cut\_off FPR TPR

1 Inf 0.000000000 0.000000000

2 0.7247299 0.000000000 0.006666667

3 0.7089944 0.002217295 0.006666667

4 0.7089896 0.004434590 0.006666667

5 0.6956243 0.004434590 0.013333333

6 0.6828339 0.006651885 0.013333333

> library(dplyr)

> rocr\_cutoff$cut\_off <- round(rocr\_cutoff$cut\_off,6)

> rocr\_cutoff <- arrange(rocr\_cutoff,desc(TPR))

> View(rocr\_cutoff)

> head(rocr\_cutoff,n = 15)

cut\_off FPR TPR

1 0.041885 0.9955654 1.0000000

2 0.034586 0.9977827 1.0000000

3 0.031727 1.0000000 1.0000000

4 0.053474 0.9844789 0.9933333

5 0.052337 0.9866962 0.9933333

6 0.050755 0.9889135 0.9933333

7 0.043885 0.9911308 0.9933333

8 0.042956 0.9933481 0.9933333

9 0.042930 0.9955654 0.9933333

10 0.078211 0.9356984 0.9866667

11 0.077703 0.9401330 0.9866667

12 0.077312 0.9423503 0.9866667

13 0.076953 0.9445676 0.9866667

14 0.074017 0.9467849 0.9866667

15 0.073642 0.9490022 0.9866667