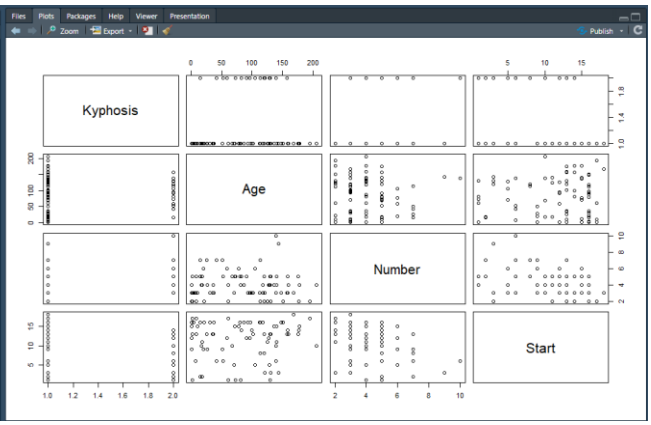
Michael Dang – 16257750

COMP-SCI 5565

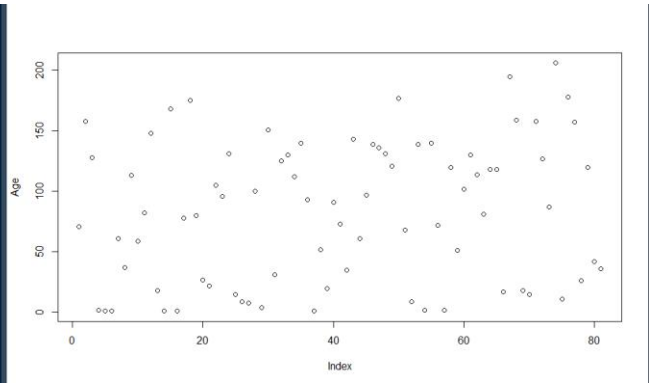
Classification Lab

10/09/2023

```
> library(rpart)
> names(kyphosis)
[1] "Kyphosis" "Age"      "Number"   "Start"
> summary(kyphosis)
    Kyphosis        Age             Number          Start
 absent :64   Min.   :  1.00   Min.   : 2.000   Min.   : 1.00
 present:17   1st Qu.: 26.00   1st Qu.: 3.000   1st Qu.: 9.00
              Median : 87.00   Median : 4.000   Median :13.00
              Mean   : 83.65   Mean   : 4.049   Mean   :11.49
              3rd Qu.:130.00   3rd Qu.: 5.000   3rd Qu.:16.00
              Max.   :206.00   Max.   :10.000   Max.   :18.00
> pairs(kyphosis)
> cor(kyphosis)
Error in cor(kyphosis) : 'x' must be numeric
> cor(kyphosis[, -1])
                Age     Number       Start
Age      1.00000000 -0.0166875  0.05782789
Number  -0.01668750  1.0000000 -0.42509875
Start    0.05782789 -0.4250988  1.00000000
>
```




```
> attach(kyphosis)
> plot(Age)
>
```



1. Logistics Regression

```
> glm.fits <- glm(Kyphosis ~., data = kyphosis, family = binomial)
> summary(glm.fits)

Call:
glm(formula = Kyphosis ~ ., family = binomial, data = kyphosis)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.3124  -0.5484  -0.3632  -0.1659   2.1613

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -2.036934   1.449575  -1.405  0.15996
Age          0.010930   0.006446   1.696  0.08996 .
Number       0.410601   0.224861   1.826  0.06785 .
Start       -0.206510   0.067699  -3.050  0.00229 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 83.234  on 80  degrees of freedom
Residual deviance: 61.380  on 77  degrees of freedom
AIC: 69.38

Number of Fisher Scoring iterations: 5

> coef(glm.fits)
(Intercept)         Age      Number       Start
-2.03693352  0.01093048  0.41060119 -0.20651005
>
```

```
> summary(glm.fits)$coef
              Estimate  Std. Error   z value    Pr(>|z|)
(Intercept) -2.03693352 1.449574526 -1.405194 0.159963578
Age          0.01093048 0.006446256  1.695633 0.089955471
Number       0.41060119 0.224860819  1.826024 0.067846690
Start       -0.20651005 0.067698863 -3.050421 0.002285206
> summary(glm.fits)$coef[, 4]
(Intercept)         Age      Number       Start
0.159963578 0.089955471 0.067846690 0.002285206
> glm.probs <- predict(glm.fits, type = 'response')
> glm.probs[1:10]
        1         2         3         4         5         6
0.25700076 0.12246899 0.49300613 0.45795535 0.02985049 0.01088999
        7         8         9        10
0.01696249 0.02401279 0.03609816 0.19677901
> contrasts(Kyphosis)
        present
absent        0
present       1
> glm.pred <- rep('absent', 81)
> glm.pred[glm.probs > .5] = 'present'
> table(glm.pred, Kyphosis)
         Kyphosis
glm.pred  absent present
  absent      61      10
  present      3       7
> (7 + 61)/81
[1] 0.8395062
> mean(glm.pred == Kyphosis)
[1] 0.8395062
> |
```

```
> train <- (Age< 100)
> kyphosis.100 <- kyphosis[!train]
Error in `[.data.frame`(kyphosis, !train) : undefined columns selected
> kyphosis.100 <- kyphosis[!train, ]
> kyphosis.100
   Kyphosis Age Number Start
2    absent 158      3    14
3   present 128      4     5
9    absent 113      2    16
12   absent 148      3    16
15   absent 168      3    18
18   absent 175      5    13
22  present 105      6     5
24   absent 131      2     3
28   absent 100      3    14
30   absent 151      2    16
32   absent 125      2    11
33   absent 130      5    13
34   absent 112      3    16
35   absent 140      5    11
43   absent 143      9     3
46  present 139      3    10
47   absent 136      4    15
48   absent 131      5    13
49  present 121      3     3
50   absent 177      2    14
53  present 139     10     6
55   absent 140      4    15
58  present 120      5     8
60   absent 102      3    13
61  present 130      4     1
62  present 114      7     8
64   absent 118      3    16
65   absent 118      4    16
```

```
65   absent 118      4    16
67   absent 195      2    17
68   absent 159      4    13
71   absent 158      5    14
72   absent 127      4    12
74   absent 206      4    10
76   absent 178      4    15
77  present 157      3    13
79   absent 120      2    13
> Kyphosis.100 <- Kyphosis[!train]
> Kyphosis.100
 [1] absent  present absent  absent  absent  absent  present absent  absent  absent  absent  absent  absent
[14] absent  absent  present absent  absent  present absent  present absent  present absent  present present
[27] absent  absent  absent  absent  absent  absent  absent  absent  present absent
Levels: absent present
> dim(Kyphosis.100)
NULL
> nrow(Kyphosis.100)
NULL
> nrow(kyphosis.100)
[1] 36
> Kyphosis.100 <- data.frame(Kyphosis.100)
> nrow(Kyphosis.100)
[1] 36
> glm.fits <- glm(Kyphosis ~ . ,data = kyphosis, family = binomial(link = 'logit'), subset = train)
> glm.probs <- predict(glm.fits, Kyphosis.100)
Warning message:
'newdata' had 36 rows but variables found have 81 rows
> glm.probs <- predict(glm.fits, Kyphosis.100, type = 'response')
Warning message:
'newdata' had 36 rows but variables found have 81 rows
> glm.probs <- predict(glm.fits, kyphosis.100, type = 'response')
> dim(kyphosis.100)
[1] 36  4
```

```
> glm.pred <- rep('absent', 36)
> glm.pred[glm.probs > .5] <- 'Present'
> table(glm.pred, Kyphosis.100)
Error in table(glm.pred, Kyphosis.100) :
  all arguments must have the same length
> glm.pred
 [1] "Present" "Present" "absent"  "Present" "Present" "Present" "Present" "Present" "absent"  "Present"
[11] "absent"  "Present" "absent"  "Present" "Present" "Present" "Present" "Present" "Present" "Present"
[21] "Present" "Present" "Present" "absent"  "Present" "Present" "absent"  "Present" "Present" "Present"
[31] "Present" "Present" "Present" "Present" "Present" "absent"
> dim(glm.pred)
NULL
> glm.pred <- data.frame(glm.pred)
> table(glm.pred, Kyphosis.100)
Error in xtfrm.data.frame(x) :
  (converted from warning) cannot xtfrm data frames
> dim(glm.pred)
[1] 36  1
> dim(Kyphosis.100)
[1] 36  1
> table(glm.pred, Kyphosis.100)
Error in xtfrm.data.frame(x) :
  (converted from warning) cannot xtfrm data frames
> table(glm.pred, kyphosis.100)
Error in xtfrm.data.frame(x) :
  (converted from warning) cannot xtfrm data frames
> order(kyphosis.100[, 'column'])
Error in `[.data.frame`(kyphosis.100, , "column") :
  undefined columns selected
> Kyphosis.100
```

```
> Kyphosis.100
   Kyphosis.100
1        absent
2       present
3        absent
4        absent
5        absent
6        absent
7       present
8        absent
9        absent
10       absent
11       absent
12       absent
13       absent
14       absent
15       absent
16      present
17       absent
18       absent
19      present
20       absent
21      present
22       absent
23      present
24       absent
25      present
26      present
27       absent
28       absent
29       absent
30       absent
31       absent
32       absent
```

```
33       absent
34       absent
35      present
36       absent
> glm.pred
   glm.pred
1   Present
2   Present
3    absent
4   Present
5   Present
6   Present
7   Present
8   Present
9    absent
10  Present
11   absent
12  Present
13   absent
14  Present
15  Present
16  Present
17  Present
18  Present
19  Present
20  Present
21  Present
22  Present
23  Present
24   absent
```

```
25  Present
26  Present
27   absent
28  Present
29  Present
30  Present
31  Present
32  Present
33  Present
34  Present
35  Present
36   absent
> Kyphosis.100 <- as.matrix(Kyphosis.100)
> glm.pred <- as.matrix(glm.pred)
> table(glm.pred, kyphosis.100)
Error in table(glm.pred, kyphosis.100) :
  all arguments must have the same length
> dim(glm.pred)
[1] 36  1
> table(glm.pred, Kyphosis.100)
         Kyphosis.100
glm.pred  absent present
  absent       7       0
  Present     20       9
> mean(glm.pred == Kyphosis.100)
[1] 0.1944444
> mean(glm.pred != Kyphosis.100)
[1] 0.8055556
>
```

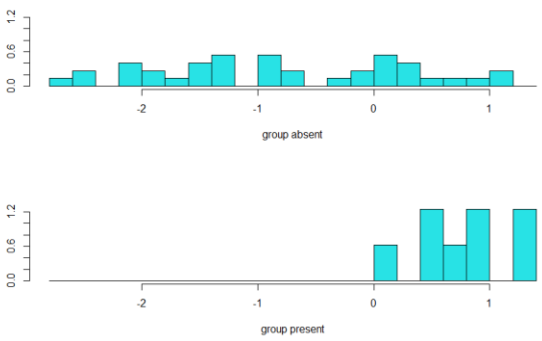
2. Linear Discriminant Analysis

```
> lda.fit <- lda(Kyphosis ~ ., data = kyphosis, subset = train)
> lda.fit
Call:
lda(Kyphosis ~ ., data = kyphosis, subset = train)

Prior probabilities of groups:
   absent   present
0.8222222 0.1777778

Group means:
              Age   Number    Start
absent  33.89189 3.864865 12.08108
present 63.75000 5.375000  8.12500

Coefficients of linear discriminants:
              LD1
Age     0.02038026
Number  0.43491883
Start  -0.07412279
> plot(lda.fit)
>
```




```
> lda.pred <- predict(lda.fit, Kyphosis.100)
Error in model.frame.default(Terms, newdata, na.action = na.pass, xlev = object$xlevels) :
  'data' must be a data.frame, not a matrix or an array
> Kyphosis.100 <- data.frame(Kyphosis.100)
> lda.pred <- predict(lda.fit, Kyphosis.100)
Error: (converted from warning) 'newdata' had 36 rows but variables found have 81 rows
> lda.pred <- predict(lda.fit, kyphosis.100)
> lda.class <- lda.pred$class
> table(lda.class, Kyphosis.100)
Error in table(lda.class, Kyphosis.100) :
  all arguments must have the same length
> lda.class
 [1] present present absent  absent  present present present present absent  absent  absent  present
[13] absent  present present present present present present present present present present absent
[25] present present absent  absent  present present present present present present present absent
Levels: absent present
> lda.class <- as.matrix(lda.class)
> Kyphosis.100 <- as.matrix(Kyphosis.100)
> table(lda.class, Kyphosis.100)
          Kyphosis.100
lda.class absent present
  absent      10       0
  present     17       9
> mean(lda.class == Kyphosis.100)
[1] 0.5277778
> sum(lda.pred$posterior[, 1] >= .5)
[1] 10
> sum(lda.pred$posterior[, 1] < .5)
[1] 26
>
```

```
> lda.pred$posterior[1:20, 1]
        2         3         9        12        15        18        22        24
0.404154668 0.239973046 0.875452578 0.540038374 0.439368621 0.083221512 0.144557218 0.469166980
       28        30        32        33        34        35        43        46
0.810627844 0.677661238 0.729340155 0.274879287 0.786504093 0.179654990 0.005222655 0.438618350
       47        48        49        50
0.437404471 0.268593655 0.381380655 0.422175065
> lda.class[1:20]
 [1] "present" "present" "absent"  "absent"  "present" "present" "present" "present" "absent"  "absent"
[11] "absent"  "present" "absent"  "present" "present" "present" "present" "present" "present" "present"
> sum(lda.pred$posterior[, 1] > .9)
[1] 0
>
```

3. Quadratic Discriminant Analysis

```
> qda.fit <- qda(Kyphosis ~ Age + Number, data = kyphosis, subset = train)
> qda.fit
Call:
qda(Kyphosis ~ Age + Number, data = kyphosis, subset = train)

Prior probabilities of groups:
   absent   present
0.8222222 0.1777778

Group means:
              Age   Number
absent  33.89189 3.864865
present 63.75000 5.375000
> qda.class <- predict(qda.fit, kyphosis.100)$class
> table(qda.class, Kyphosis.100)
         Kyphosis.100
qda.class absent present
  absent      15       6
  present     12       3
> mean(qda.class == Kyphosis.100)
[1] 0.5
>
```

4. Naïve Bayes

```
> install.packages('e1071', type = 'source')
Installing package into 'C:/Users/hoang/AppData/Local/R/win-library/4.2'
(as 'lib' is unspecified)
trying URL 'https://cran.rstudio.com/src/contrib/e1071_1.7-13.tar.gz'
Content type 'application/x-gzip' length 314205 bytes (306 KB)
downloaded 306 KB

* installing *source* package 'e1071' ...
** package 'e1071' successfully unpacked and MD5 sums checked
** using staged installation
** libs
gcc  -I"C:/PROGRA~1/R/R-42~1.1/include" -DNDEBUG     -I"C:/Program Files/R/rtools42/x86_64-w64-mingw32.stati
c.posix/include"     -O2 -Wall  -std=gnu99 -mfpmath=sse -msse2 -mstackrealign  -c Rsvm.c -o Rsvm.o
gcc  -I"C:/PROGRA~1/R/R-42~1.1/include" -DNDEBUG     -I"C:/Program Files/R/rtools42/x86_64-w64-mingw32.stati
c.posix/include"     -O2 -Wall  -std=gnu99 -mfpmath=sse -msse2 -mstackrealign  -c cmeans.c -o cmeans.o
gcc  -I"C:/PROGRA~1/R/R-42~1.1/include" -DNDEBUG     -I"C:/Program Files/R/rtools42/x86_64-w64-mingw32.stati
c.posix/include"     -O2 -Wall  -std=gnu99 -mfpmath=sse -msse2 -mstackrealign  -c cshell.c -o cshell.o
gcc  -I"C:/PROGRA~1/R/R-42~1.1/include" -DNDEBUG     -I"C:/Program Files/R/rtools42/x86_64-w64-mingw32.stati
c.posix/include"     -O2 -Wall  -std=gnu99 -mfpmath=sse -msse2 -mstackrealign  -c floyd.c -o floyd.o
gcc  -I"C:/PROGRA~1/R/R-42~1.1/include" -DNDEBUG     -I"C:/Program Files/R/rtools42/x86_64-w64-mingw32.stati
c.posix/include"     -O2 -Wall  -std=gnu99 -mfpmath=sse -msse2 -mstackrealign  -c init.c -o init.o
g++ -std=gnu++11  -I"C:/PROGRA~1/R/R-42~1.1/include" -DNDEBUG     -I"C:/Program Files/R/rtools42/x86_64-w64-
mingw32.static.posix/include"     -O2 -Wall  -mfpmath=sse -msse2 -mstackrealign  -c svm.cpp -o svm.o
g++ -std=gnu++11 -shared -s -static-libgcc -o e1071.dll tmp.def Rsvm.o cmeans.o cshell.o floyd.o init.o svm.
o -LC:/Program Files/R/rtools42/x86_64-w64-mingw32.static.posix/lib/x64 -LC:/Program Files/R/rtools42/x86_64
-w64-mingw32.static.posix/lib -LC:/PROGRA~1/R/R-42~1.1/bin/x64 -lR
installing to C:/Users/hoang/AppData/Local/R/win-library/4.2/00LOCK-e1071/00new/e1071/libs/x64
** R
** inst
** byte-compile and prepare package for lazy loading
** help
*** installing help indices
** building package indices
** installing vignettes
```

```
> library(e1071)
> nb.fit <- naiveBayes(Kyphosis ~ ., data = kyphosis, subset = train)
> nb.fit

Naive Bayes Classifier for Discrete Predictors

Call:
naiveBayes.default(x = X, y = Y, laplace = laplace)

A-priori probabilities:
Y
   absent   present
0.8222222 0.1777778

Conditional probabilities:
         Age
Y             [,1]     [,2]
  absent  33.89189 31.51083
  present 63.75000 27.28553

         Number
Y             [,1]     [,2]
  absent  3.864865 1.336707
  present 5.375000 1.302470

         Start
Y             [,1]     [,2]
  absent  12.08108 4.957477
  present  8.12500 5.026714

> |
```

```
> mean(Age[train][Kyphosis[train] == 'absent'])
[1] 33.89189
> sd(Age[train][Kyphosis[train] == 'absent'])
[1] 31.51083
> nb.class <- predict(nb.fit, kyphosis.100)
> table(nb.class, Kyphosis.100)
          Kyphosis.100
nb.class  absent present
  absent      21       5
  present      6       4
> mean(nb.class == Kyphosis.100)
[1] 0.6944444
> nb.preds <- predict(nb.fit, kyphosis.100, type = 'raw')
> nb.preds[1:5, ]
        absent    present
[1,] 0.7358102 0.26418975
[2,] 0.5563512 0.44364878
[3,] 0.9023209 0.09767905
[4,] 0.7359595 0.26404050
[5,] 0.7421443 0.25785575
> |
```

A.

- Yes, naiveBayes performs better than Logistics even though the dataset in naiveBayes is smaller. Because the naiveBayes work better with 2 more classes.

B.

- Compared to LDA, I feel the method is more interpretable.

C.

- This dataset is hard to interpret, I think because the data of each feature is random (eg, in Age, data are distributed randomly) and there are many classes of Number features.
- I believe Random Forest will outperform the other model for this particular dataset.

5. K-Nearest Neighbor

```
> library(class)
> train.X <- cbind(Age, Start)[train, ]
> test.X <- cbind(Age, Start)[!train, ]
> train.Kyphosis <- Kyphosis[train]
> set.seed(1)
> knn.pred <- knn(train.X, test.X, train.Kyphosis, k = 1)
> table(knn.pred, Kyphosis.100)
         Kyphosis.100
knn.pred  absent present
  absent      26       3
  present      1       6
> (26 + 6)/36
[1] 0.8888889
> knn.pred <- knn(train.X, test.X, train.Kyphosis, k = 3)
> table(knn.pred, Kyphosis.100)
         Kyphosis.100
knn.pred  absent present
  absent      27       8
  present      0       1
> mean(knn.pred == Kyphosis.100)
[1] 0.7777778
> |
```

```
> dim(Caravan)
[1] 5822   86
> attach(Caravan)
> summary(Purchase)
  No  Yes
5474  348
> 348/5822
[1] 0.05977327
> standardized.X <- scale(Caravan[, -86])
> var(Caravan[, 1])
[1] 165.0378
> var(Caravan[, 2])
[1] 0.1647078
> var(standardized.X[, 1])
[1] 1
> var(standardized.X[, 2])
[1] 1
> |
```

```
> test <- 1:1000
> train.X <- standardized.X[-test, ]
> test.X <- standardized.X[test, ]
> train.Y <- Purchase[-test]
> test.Y <- Purchase[test]
> set.seed(1)
> knn.pred <- knn(train.X, test.X, train.Y, k = 1)
> mean(test.Y != knn.pred)
[1] 0.118
> mean(test.Y != 'No')
[1] 0.059
> |
```

```
> table(knn.pred, test.Y)
        test.Y
knn.pred  No Yes
     No  873  50
     Yes  68   9
> 9 / (68 + 9)
[1] 0.1168831
> knn.pred <- knn(train.X, test.X, train.Y, k = 3)
> table(knn.pred, test.Y)
        test.Y
knn.pred  No Yes
     No  920  54
     Yes  21   5
> 5 / 26
[1] 0.1923077
> knn.pred <- knn(train.X, test.X, train.Y, k = 5)
> table(knn.pred, test.Y)
        test.Y
knn.pred  No Yes
     No  930  55
     Yes  11   4
> 4/15
[1] 0.2666667
> |
```

```
> glm.fits <- glm(Purchase ~ ., data = Caravan, family = binomial, subset = -test)
Error: (converted from warning) glm.fit: fitted probabilities numerically 0 or 1 occurred
> glm.probs <- predict(glm.fits, Caravan[test, ], type = 'response')
Error: (converted from warning) 'newdata' had 1000 rows but variables found have 81 rows
> glm.pred <- rep('No', 1000)
> glm.pred[glm.probs > .5] <- 'Yes'
> table(glm.pred, test.Y)
        test.Y
glm.pred  No Yes
     No  184  11
     Yes 757  48
> glm.pred <- rep('No', 1000)
> glm.pred[glm.probs > .25] <- 'Yes'
> table(glm.pred, test.Y)
        test.Y
glm.pred  No Yes
     No  130   9
     Yes 811  50
> 50 / (50 + 811)
[1] 0.05807201
> |
```