(1) College students completing a preliminary year

The data for college students was very small but it had a lot of columns which made it very rich. There were 6 character columns and 28 numerical columns. Most of the numerical columns had the value of either 1 or 0

As always, I printed the head of the data and then the summary and immediately noticed the missing values. Because we had character variables, there was a possibility of missing values in these columns too so I checked all the character variables separately. In this process, I found that 4 out of the 6 character variables had 2 rows when all of them were missing
There were several missing values in the remaining 2 columns. To solve the missing values in character columns, we delete the two rows with missing values in 4 columns and the 2 columns with a lot of missing values
The next step was to solve the missing values in numerical columns which is easy. As a simple step, they have been imputed with mean of the data
Then we set a seed so that the results can be replicated. We have divided our data into 85%-15% train test split and then fitted out logistic regression model on the train data only.
The model summary shows us that there were no anomalies in model fitting and the model looks good at the onset. To verify this, we make predictions on the test data and check how many of them were correct. The model accuracy on test data is as follows:

```
predictions  0  1
          0  5  2
          1  3 11
```

This shows that we were able to make 5 out of 8 correct predictions as 0 and 11 out of 13 correct predictions as 1. The total accuracy is 5+11 = 16 correct predictions out of 21. This is equal to 16/21 or 76.19% accuracy and is a great number

We then try to find out the features which decide the model predictions. These features in their order of importance are:

Receptivity.to.Academic.Assistance..percentile.score.before.start.of.semester.	1.18E-04
Completed.Connect...1.yes..0.no.1	6.41E-05
Number.of.Peer.Mentor.Meetings.Attended	6.37E-05
Peer.MentorFrancess	5.55E-05
Peer.MentorCorrinn	5.36E-05
Completed.Connect...1.yes..0.no.0, contract for fall	5.27E-05
Peer.MentorJoli	4.93E-05
Peer.MentorGeorge	4.89E-05
Attended.Orientation...1.yes..0.no.	3.87E-05
Dropout.Proneness..percentile.score.before.start.of.semester.	3.77E-05
Receptivity.to.Academic.Assistance..percentile.score.before.start.of.semester.	1.18E-04

CODE:

```
#Import libraries
library(dplyr)
library(caret)

#Read data
data = read.csv("232408682483120_File.csv")

#View data
head(data)
summary(data) #missing value alert

#Check character columns
unique(data$Gender) #So there are missing values here too

#Check how many missing in gender
length(which(data$Gender=="")) #Only 2. Delete
data = data %>% filter(!Gender=="")

#Check how many missing in Federal Ethnic Group
length(which(data$Federal.Ethnic.Group=="")) #No missing values now
#Check how many missing in Peer Mentor
length(which(data$Peer.Mentor=="")) #No missing values now
#Check how many missing in Completed.Connect...1.yes..0.no.
length(which(data$Completed.Connect...1.yes..0.no.=="")) #No missing values now
#Check how many missing in Reason.for.not.Completing.Connect
length(which(data$Reason.for.not.Completing.Connect=="")) #Should be removed
#Check how many missing in Reason.not.Retained
length(which(data$Reason.not.Retained=="")) #Should be removed

data$Reason.for.not.Completing.Connect=NULL
data$Reason.not.Retained=NULL

#Let's fill all the remaining missing values by their mean
for(i in 1:ncol(data)){
  data[is.na(data[,i]), i] <- mean(data[,i], na.rm = TRUE)
}

#Divide data in train and test
set.seed(5)

random_rows = sample(c(TRUE, FALSE), nrow(data), replace=TRUE,
prob=c(0.85,0.15))
college_train = data[random_rows, ]
college_test = data[!random_rows, ]

#Train the model
model.fit = glm(Retained.F17.F18...1.yes..0.no. ~ .,  data =
college_train,  family = "binomial")
#Check how the model performed
```

```
model.fit #Some variables have NA values. They don't have any value

#Model accuracy check
predictions = predict(model.fit, college_test, type = "response")
predictions = round(predictions,0)
table(predictions,college_test$Retained.F17.F18...1.yes..0.no.) #16/21 = 76.19%

#Check for important features
list_of_feature_importance = as.data.frame(varImp(model.fit))
list_of_feature_importance = list_of_feature_importance %>%
arrange(desc(Overall) )
head(list_of_feature_importance,10)
```

(2) Heart Health Data

Logistic model to predict whether a person seeks medical treatment in 2 days or less

first of all, I created a separate data frame by making a copy of the heart health data. Next, I created a new column named “delay_day_2” where values are 1 if value in delaydays is less than or equals to 2 otherwise 0. Before fitting the logistic regression model to the dataset, I dropped all the unnecessary columns (ID, delaydays) from the data frame.

Next, I fitted the logistic model to the prepared data by declaring delay_day_2 column as dependent variable and others as independent variables. The summary of the fitted logistic model is shown below:

```
Call:
glm(formula = delay_day_2 ~ ., family = "binomial", data = df1)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.1462  -1.1038   0.6974   1.0924   1.9241

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)   0.497772   1.188270   0.419  0.67529
Age           0.013161   0.009351   1.407  0.15929
Gender       -0.093782   0.217231  -0.432  0.66595
Ethnicity    -0.053748   0.187852  -0.286  0.77479
Marital       0.074639   0.177241   0.421  0.67367
Livewith     -0.208708   0.264574  -0.789  0.43020
Education     0.009505   0.077872   0.122  0.90285
palpitations  0.140329   0.126346   1.111  0.26671
orthopnea    -0.041271   0.116699  -0.354  0.72360
chestpain     0.126625   0.126599   1.000  0.31721
nausea       -0.092194   0.135773  -0.679  0.49712
cough        -0.314724   0.112493  -2.798  0.00515 **
fatigue      -0.187713   0.139209  -1.348  0.17752
dyspnea       0.093580   0.134549   0.696  0.48674
edema        -0.223813   0.123652  -1.810  0.07029 .
PND          -0.171321   0.111453  -1.537  0.12425
tightshoes    0.129220   0.131284   0.984  0.32498
weightgain    0.204976   0.113305   1.809  0.07044 .
DOE          -0.209921   0.124091  -1.692  0.09071 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 555.84  on 400  degrees of freedom
Residual deviance: 522.49  on 382  degrees of freedom
  (5 observations deleted due to missingness)
AIC: 560.49
```

So, we can write the fitted model as follows:

Delay_day_2= 0.498 + 0.013*(Age) – 0.094*(Gender) – 0.054*(Ethnicity) + 0.075*(Marital) – 0.209*(Livewith) + 0.010*(Education) + 0.140*(palpitations) - 0.041*(orthopnea) + 0.127*(chestpain) -0.092*(nausea) - 0.315*(cough) - 0.188*(fatigue) + 0.094*(dyspnea) - 0.224*(edema) - 0.171*(PND) + 0.129*(tightshoes) + 0.205*(weightgain) - 0.210*(DOE).

We can notice in the model output summary that there is only one attribute for which corresponding p-value is less than 0.05. Thus, we can conclude that all the variables except “cough” is useful in predicting the outcome.

Logistic model to predict whether a person seeks medical treatment on or less than the cohort average delay days

Again, I created a separate data frame by making a copy of the heart health data. Next, I created a new column named “delay_day_avg” where values are 1 if value in delaydays is less than or equals to the mean value of the “delaydays” column otherwise 0. Before fitting the logistic regression model to the dataset, I dropped all the unnecessary columns (ID, delaydays) from the data frame.

Next, I fitted the logistic model to the prepared data by declaring delay_day_avg column as dependent variable and others as independent variables. The summary of the fitted logistic model is shown below:

```
Call:
glm(formula = delay_day_avg ~ ., family = "binomial", data = df2)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.9479  -1.3037   0.7038   0.8517   1.4634

Coefficients:
               Estimate Std. Error z value Pr(>|z|)
(Intercept)   1.2585031  1.2763221   0.986   0.3241
Age           0.0032097  0.0098841   0.325   0.7454
Gender       -0.0690729  0.2332990  -0.296   0.7672
Ethnicity     0.0095675  0.1973446   0.048   0.9613
Marital       0.0003959  0.1906928   0.002   0.9983
Livewith      0.0647389  0.2849460   0.227   0.8203
Education     0.0022595  0.0834440   0.027   0.9784
palpitations -0.0466646  0.1355530  -0.344   0.7307
orthopnea     0.0203143  0.1261581   0.161   0.8721
chestpain     0.0873590  0.1349603   0.647   0.5174
nausea       -0.3109419  0.1387087  -2.242   0.0250 *
cough        -0.0686571  0.1210645  -0.567   0.5706
fatigue       0.1390914  0.1478189   0.941   0.3467
dyspnea       0.0498990  0.1467811   0.340   0.7339
edema        -0.3101735  0.1292778  -2.399   0.0164 *
PND          -0.1798364  0.1194612  -1.505   0.1322
tightshoes    0.0805700  0.1371233   0.588   0.5568
weightgain    0.1455938  0.1207061   1.206   0.2277
DOE          -0.2215595  0.1353602  -1.637   0.1017
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 487.69  on 400  degrees of freedom
Residual deviance: 467.75  on 382  degrees of freedom
  (5 observations deleted due to missingness)
AIC: 505.75
```

So, we can write the fitted model as follows:

Delay_day_avg = 1.259 + 0.003*(Age) – 0.069*(Gender) – 0.010*(Ethnicity) + 0.000*(Marital) – 0.065*(Livewith) + 0.002*(Education) - 0.047 *(palpitations) + 0.020*(orthopnea) + 0.087*(chestpain) -0.311*(nausea) - 0.069*(cough) + 0.139*(fatigue) + 0.050*(dyspnea) - 0.310*(edema) - 0.180*(PND) + 0.081*(tightshoes) + 0.146*(weightgain) - 0.222*(DOE).

We can notice in the model output summary that there is only two attribute for which corresponding p-value is less than 0.05. Thus, we can conclude that all the variables except “nausea” & “edema” is useful in predicting the outcome.

Logistic model to predict whether a person seeks medical treatment in 1 days or less

first of all, I created a separate data frame by making a copy of the heart health data. Next, I created a new column named “delay_day_1” where values are 1 if value in delaydays is less than or equals to 1 otherwise 0. Before fitting the logistic regression model to the dataset, I dropped all the unnecessary columns (ID, delaydays) from the data frame.

Next, I fitted the logistic model to the prepared data by declaring delay_day_1 column as dependent variable and others as independent variables. The summary of the logistic model is shown below:

```
Call:
glm(formula = delay_day_1 ~ ., family = "binomial", data = df3)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.5801  -0.8993  -0.6710   1.1516   2.0683

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  -1.23266    1.26685  -0.973  0.33055
Age           0.01799    0.01021   1.763  0.07796 .
Gender        0.05989    0.23177   0.258  0.79611
Ethnicity    -0.08135    0.21068  -0.386  0.69942
Marital       0.16490    0.19145   0.861  0.38908
Livewith     -0.22360    0.27889  -0.802  0.42270
Education     0.06223    0.08167   0.762  0.44604
palpitations  0.09533    0.13260   0.719  0.47222
orthopnea    -0.34312    0.12488  -2.748  0.00600 **
chestpain    -0.11724    0.13883  -0.844  0.39841
nausea        0.01827    0.14710   0.124  0.90115
cough        -0.32566    0.12176  -2.675  0.00748 **
fatigue       0.12620    0.15026   0.840  0.40098
dyspnea       0.17020    0.14257   1.194  0.23256
edema        -0.35512    0.13662  -2.599  0.00934 **
PND           0.11054    0.11988   0.922  0.35651
tightshoes    0.13545    0.14437   0.938  0.34811
weightgain    0.13873    0.12125   1.144  0.25259
DOE          -0.27123    0.13079  -2.074  0.03810 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 514.98  on 400  degrees of freedom
Residual deviance: 473.58  on 382  degrees of freedom
  (5 observations deleted due to missingness)
AIC: 511.58
```

So, we can write the fitted model as follows:

Delay_day_avg = -1.233 + 0.018*(Age) – 0.060*(Gender) – 0.081*(Ethnicity) + 0.165*(Marital) – 0.224*(Livewith) + 0.062*(Education) + 0.095 *(palpitations) - 0.343*(orthopnea) + 0.117*(chestpain) + 0.018*(nausea) - 0.326*(cough) + 0.126*(fatigue) + 0.170*(dyspnea) - 0.355*(edema) + 0.111*(PND) + 0.135*(tightshoes) + 0.139*(weightgain) - 0.271*(DOE).

We can notice in the model output summary that there is four different attributes for which corresponding p-value is less than 0.05. Thus, we can conclude that all the variables except “orthopnea”, “cough”, “edema” & “DOE” are useful in predicting the outcome.

CODE:

```
library(readxl)
heart_health= read_excel("heart health data.xls")

df1= heart_health
df1$delay_day_2= ifelse(df1$delaydays<=2,1,0)
df1= df1[,c(-1,-20)]
colnames(df1)
logistic_model_1= glm(delay_day_2~., data = df1, family = "binomial")
summary(logistic_model_1)

df2= heart_health
avg_delay= mean(df2$delaydays, na.rm= TRUE)
avg_delay
df2$delay_day_avg= ifelse(df2$delaydays<= avg_delay, 1,0)
df2= df2[,c(-1,-20)]
colnames(df2)
logistic_model_2= glm(delay_day_avg~., data = df2, family = "binomial")
summary(logistic_model_2)

df3= heart_health
df3$delay_day_1= ifelse(df3$delaydays<=1, 1, 0)
df3= df3[,c(-1,-20)]
colnames(df3)
logistic_model_3= glm(delay_day_1~., data = df3, family= "binomial")
summary(logistic_model_3)
```