Assignment 3:

Data file: quality.csv

FILE Attributes:

i. num_words: number of words in the post.
ii. num_characters: number of character in the post.
iii. num_misspelled: number of misspelled word.
iv. bin_end_qmark: if the post ends with a question mark
v. num_interrogative: number of interrogative word in the post.
vi. bin_start_small: if the answer starts with a small letter. ('1' means yes, otherwise no)
vii. num_sentences: number of sentences per post.
viii. num_punctuations: number of punctuation symbols in the post.
ix. label: the label of the post ('G' for good and 'B' for bad) as determined by the tool.

**Logistic Regression**

Data pre-processing

```
> regQlt <- read.csv(file.choose(), header = TRUE, sep = ",")
> View(regQlt)
>
> # to include G and B samples equally to train and test data
> trainQlt = regQlt[4:26,]
> valQlt1 = regQlt[1:3,]
> valQlt2 = regQlt[27:28,]


> valQlt <- rbind.data.frame(valQlt1,valQlt2)
> names(regQlt)
 [1] "ï..S.No."          "num_words"         "num_characters"    "num_misspel
led"
 [5] "bin_end_qmark"     "num_interrogative" "bin_start_small"   "num_sentenc
es"
 [9] "num_punctuations"  "label"


> #Check if the data is factored
> is.factor(trainQlt$num_words)
[1] FALSE
> is.factor(trainQlt$num_characters)
[1] FALSE
> is.factor(trainQlt$num_misspelled)
[1] FALSE
> is.factor(trainQlt$bin_end_qmark)
[1] FALSE
> is.factor(trainQlt$bin_start_small)
[1] FALSE
> is.factor(trainQlt$num_interrogative)
[1] FALSE
> is.factor(trainQlt$num_sentences)
[1] FALSE
> is.factor(trainQlt$num_punctuations)
[1] FALSE
> is.factor(trainQlt$label)
[1] TRUE
>
```

```
> #bin_end_qmark and bin_start_small have 0 and 1 values thus factor them
>
> trainQlt$bin_end_qmark <- as.factor(trainQlt$bin_end_qmark)
> trainQlt$bin_start_small <- as.factor(trainQlt$bin_start_small)
>
>
> valQlt$bin_end_qmark <- as.factor(valQlt$bin_end_qmark)
> valQlt$bin_start_small <- as.factor(valQlt$bin_start_small)
>
>
> is.factor(trainQlt$bin_end_qmark)
[1] TRUE
> is.factor(trainQlt$bin_start_small)
[1] TRUE
>
> dim(trainQlt)   # 23 sample for training
[1] 23 10
> dim(valQlt)     # 5 for testing
[1]  5 10
```

Creating models using Logistic regression:

# Model 1 using all the variables

```
> # Model 1 using all the variables
>
> mdl = glm(label~num_words+num_characters+num_misspelled+bin_end_qmark+num_i
nterrogative+bin_start_small+num_sentences+num_punctuations, family = binomia
l(link = "logit"),data = trainQlt)
> summary(mdl)

Call:
glm(formula = label ~ num_words + num_characters + num_misspelled +
    bin_end_qmark + num_interrogative + bin_start_small + num_sentences +
    num_punctuations, family = binomial(link = "logit"), data = trainQlt)

Deviance Residuals:
       Min          1Q      Median          3Q         Max
-2.619e-05  -2.110e-08   2.110e-08   2.110e-08   3.148e-05

Coefficients:
                  Estimate Std. Error z value Pr(>|z|)
(Intercept)       -4.919e+01  6.222e+04  -0.001    0.999
num_words          8.441e+00  7.459e+03   0.001    0.999
num_characters    -6.236e-01  1.379e+03   0.000    1.000
num_misspelled    -4.105e+01  2.830e+04  -0.001    0.999
bin_end_qmark1    -6.807e+01  9.377e+05   0.000    1.000
num_interrogative -1.524e+01  4.837e+04   0.000    1.000
bin_start_small1   8.937e+01  7.695e+04   0.001    0.999
num_sentences      5.609e+01  5.028e+04   0.001    0.999
num_punctuations  -1.772e+01  1.522e+04  -0.001    0.999

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 3.1841e+01  on 22  degrees of freedom
Residual deviance: 3.4290e-09  on 14  degrees of freedom
AIC: 18

Number of Fisher Scoring iterations: 25

> rs  = predict(mdl,newdata = valQlt,type="response")
> rs_lbl = ifelse(rs > 0.5,"G","B")
> rs_lbl
```

```
  1   2   3  27  28
"B" "G" "B" "G" "G"
> valQlt$label
[1] B B B G G
Levels: B G
> misClass = mean(rs_lbl != valQlt$label)
> accry = 1 - misClass
> accry
[1] 0.8



> # Model 2 with 6 variables
>
> mdl = glm(label~num_words+num_characters+num_misspelled+num_interrogative+b
in_start_small+num_sentences, family = binomial(link = "logit"),data = trainQ
lt)
> summary(mdl)

Call:
glm(formula = label ~ num_words + num_characters + num_misspelled +
    num_interrogative + bin_start_small + num_sentences, family = binomial(li
nk = "logit"),
    data = trainQlt)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-2.05769  -0.56819   0.00003   0.67368   1.38770

Coefficients:
                  Estimate Std. Error z value Pr(>|z|)
(Intercept)       -2.79708    1.96600  -1.423    0.155
num_words          0.16215    0.19250   0.842    0.400
num_characters     0.01157    0.04836   0.239    0.811
num_misspelled    -1.50816    1.10984  -1.359    0.174
num_interrogative -0.41275    1.44293  -0.286    0.775
bin_start_small1   4.86738    3.58864   1.356    0.175
num_sentences      1.04929    1.59841   0.656    0.512

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 31.841  on 22  degrees of freedom
Residual deviance: 16.715  on 16  degrees of freedom
AIC: 30.715

Number of Fisher Scoring iterations: 8

> rs  = predict(mdl,newdata = valQlt,type="response")
> rs_lbl = ifelse(rs > 0.5,"G","B")
> rs_lbl
  1   2   3  27  28
"B" "G" "B" "B" "G"
> valQlt$label
[1] B B B G G
Levels: B G
> misClass = mean(rs_lbl != valQlt$label)
> accry = 1 - misClass
> accry
[1] 0.6
>
>
> # Model 3 with 5 variables
>
```

```
> mdl = glm(label~num_words+num_characters+num_misspelled+num_interrogative+n
um_sentences, family = binomial(link = "logit"),data = trainQlt)
> summary(mdl)

Call:
glm(formula = label ~ num_words + num_characters + num_misspelled +
    num_interrogative + num_sentences, family = binomial(link = "logit"),
    data = trainQlt)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-1.84008  -0.71723   0.00824   0.80430   1.82266

Coefficients:
                 Estimate Std. Error z value Pr(>|z|)
(Intercept)       -1.67491    1.35663  -1.235    0.217
num_words          0.12056    0.17106   0.705    0.481
num_characters    -0.01501    0.04257  -0.353    0.724
num_misspelled    -0.78981    0.53927  -1.465    0.143
num_interrogative  0.50100    1.07208   0.467    0.640
num_sentences      1.07613    1.26495   0.851    0.395

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 31.841  on 22  degrees of freedom
Residual deviance: 19.255  on 17  degrees of freedom
AIC: 31.255

Number of Fisher Scoring iterations: 7

> rs  = predict(mdl,newdata = valQlt,type="response")
> rs_lbl = ifelse(rs > 0.5,"G","B")
> rs_lbl
  1   2   3  27  28
"B" "B" "G" "G" "G"
> valQlt$label
[1] B B B G G
Levels: B G
> misClass = mean(rs_lbl != valQlt$label)
> accry = 1 - misClass
> accry
[1] 0.8
>
>
> # Model 4 with 4 variables
>
> mdl = glm(label~num_words+num_characters+num_misspelled+num_sentences, fami
ly = binomial(link = "logit"),data = trainQlt)
> summary(mdl)

Call:
glm(formula = label ~ num_words + num_characters + num_misspelled +
    num_sentences, family = binomial(link = "logit"), data = trainQlt)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-1.77925  -0.72063   0.01131   0.75780   1.86969

Coefficients:
               Estimate Std. Error z value Pr(>|z|)
(Intercept)    -1.38767    1.20972  -1.147    0.251
num_words       0.13329    0.16448   0.810    0.418
num_characters -0.01696    0.04283  -0.396    0.692
num_misspelled -0.72077    0.48353  -1.491    0.136
```

```
num_sentences   1.02790    1.25280   0.820    0.412

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 31.841  on 22  degrees of freedom
Residual deviance: 19.477  on 18  degrees of freedom
AIC: 29.477

Number of Fisher Scoring iterations: 7

> rs  = predict(mdl,newdata = valQlt,type="response")
> rs_lbl = ifelse(rs > 0.5,"G","B")
> rs_lbl
  1   2   3  27  28
"B" "G" "G" "G" "G"
> valQlt$label
[1] B B B G G
Levels: B G
> misClass = mean(rs_lbl != valQlt$label)
> accry = 1 - misClass
> accry
[1] 0.6
```

Note : None of the above model is giving significant P value it could be due to less or made up data.

Comparing the accuracy using different models:

| Model | Accuracy |
|---|---|
| Model 1 | 0.8 |
| Model 2 | 0.6 |
| Model 3 | 0.8 |
| Model 4 | 0.6 |