Logistic Regression

Introduction:

Logistic Regression is a supervised machine learning algorithm widely used for binary classification tasks. It's used to predict the probability that an instance belongs to a given class or not. For example, it can be used to identify whether an email is spam or no, diagnose diseases based on patient test results, or predict voting behavior based on patient test results, or predict voting behavior based on a given set of independent variables.

DATASET – 1

(CO2 – Carbon Dioxide Uptake in Grass Plants)

About the dataset :

The CO2 data frame has 84 rows and 5 columns of data from an experiment on the cold tolerance of the grass species *Echinochloa crus-galli*.

It has 5 attributes:

- **Plant :** An ordered factor which is also a unique identifier for each plant.
- **Type :** It contains the origin of the plant (Mississippi or Quebec)
- **Treatment :** Factors with levels chilled or nonchilled.
- **conc :** a numeric vector of ambient carbon dioxide concentrations
- **uptake :** a numeric vector of carbon dioxide uptake rates

Objective:

We'll try to fit a regression model to predict Treatment type based on the conc (CO2 concentration) and uptake (CO2 uptake).

Methodology:

We'll try to fit a logistic regression model as the attribute **Treatment** is having only 2 values i.e. chilled and nonchilled.

1. **Installing and Loading Library**

```
> install.packages("ISLR")
> library(ISLR)
```

2. Attaching dataset and initial processing

```
> attach(CO2)
```

As we know that our target attribute (i.e. Treatment) is having 2 categorical variables so, we'll first be converting it to binary values (i.e. 0 and 1).

```
> CO2$Treatment<-ifelse(CO2$Treatment=="nonchilled",0,1)
```

3. Model Fitting

We try to fit the model for dependent variable (i.e. Treatment) on independent variables (conc and uptake).

```
> model=glm(Treatment~ conc + uptake,data = CO2,family="binomial")
```

4. Model Summary

```
> summary(model)
```

```
Call:
glm(formula = Treatment ~ conc + uptake, family = "binomial",
    data = CO2)

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  1.6506858  0.6551051   2.520  0.01174 *
conc         0.0015974  0.0009655   1.655  0.09802 .
uptake      -0.0859501  0.0270428  -3.178  0.00148 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 116.45  on 83  degrees of freedom
Residual deviance: 104.70  on 81  degrees of freedom
AIC: 110.7

Number of Fisher Scoring iterations: 4
```

From summary, we can see the estimated values of regression coefficients and intercept.Also, we can see that uptake is significant attribute in predicting our target variable whereas conc is not having enough significance.

5. Predicting Values and Calculating Accuracy

```
> CO2$prob<-predict(model,type="response")
> CO2$cutoff<-ifelse(CO2$prob<0.5,0,1)
> ct=table(CO2$Treatment,CO2$cutoff)
> accuracy=sum(diag(ct))/sum(ct)
> accuracy
```

```
> CO2$prob<-predict(model,type="response")
> CO2$cutoff<-ifelse(CO2$prob<0.5,0,1)
> ct=table(CO2$Treatment,CO2$cutoff)
> accuracy=sum(diag(ct))/sum(ct)
> accuracy
[1] 0.6309524
```

We are getting an accuracy score of 0.6309524 (i.e. we can say our model is predicting target variable with almost 63% accuracy)

```
> ct

     0  1
0 28 14
1 17 25
```

We can see the contingency table also. Correct predictions for 0 (i.e nonchiled) is 28 whereas for 1 (i.e. chilled) is 25. The number of correct prediction is more than wrong predictions.

Results:

We used CO2 dataset and tried to fit a logistic regression model for predicting whether the grass went under chilled treatment or nonchilled ones on the basis of carbon dioxide concentration and CO2 uptake. We got an accuracy of 63% with our model and found that CO2 concentration isn't singnificantly affecting the target variable during the prediction.

DATASET – 2

(Default – Credit Card Default Data)

About the dataset :

A simulated data set containing information in ten thousand customers. The aim here is to predict which customers will default on their credit card debt.

It has 4 attributes:

- **default :** A factor with levels No and Yes indicating whether the customer defaulted on their debt.
- **student :** A factor with levels No and Yes indicating whether the customer is a student.
- **balance :** The average balance that the customer has remaining on their credit card after making their monthly payment.
- **Income :** Income of customer

Objective:

We'll try to fit a Logistic Regression Model to predict whether customer will default on their credit card payment or not based on their average balance, income and whether they are student or not.

Methodology:

We'll try to fit a Logistic regression model for default on student, balace, income

1. Attaching Dataset and Initial Processing

```
> attach(Default)
```

As we know that our target attribute (i.e. default) is having 2 categorical values so, we'll first be converting it to binary values (i.e. 0 and 1).

```
> Default$default<-ifelse(Default$default=="No",0,1)
```

2 . Model Fitting

We try to fit the model for predicting default based on other variables.

```
> model=glm(default~.,data=Default, family="binomial")
```

3. Model Summary

```
> summary(model)
```

```
> summary(model)

Call:
glm(formula = default ~ ., family = "binomial", data = Default)

Coefficients:
              Estimate Std. Error z value
(Intercept) -1.065e+01  7.833e-01 -13.595
studentYes  -6.393e-01  2.431e-01  -2.629
balance      5.587e-03  5.622e-04   9.938
income       2.601e-06  8.251e-06   0.315
prob         2.196e-02  1.040e+00   0.021
cutoff       2.124e-01  3.654e-01   0.581
            Pr(>|z|)
(Intercept)  < 2e-16 ***
studentYes   0.00856 **
balance      < 2e-16 ***
income       0.75261
prob         0.98315
cutoff       0.56098
---
Signif. codes:
  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’
  0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 2920.6  on 9999  degrees of freedom
Residual deviance: 1570.8  on 9994  degrees of freedom
AIC: 1582.8

Number of Fisher Scoring iterations: 9
```

We can see that intercept, student, balance are significant for our model prediction whereas income is not significant.

4. Predicting Values and calculating accuracy

```
> Default$prob<-predict(model,type='response')
> Default$cutoff<-ifelse(Default$prob<0.5,0,1)
> ct=table(Default$default,Default$cutoff)
> accuracy=sum(diag(ct)/sum(ct))
> accuracy
```

```
> Default$prob<-predict(model,type='response')
> Default$cutoff<-ifelse(Default$prob<0.5,0,1)
> ct=table(Default$default,Default$cutoff)
> accuracy=sum(diag(ct)/sum(ct))
> accuracy
[1] 0.9732
```

We are getting an accuracy score of 0.9732 (i.e. we can say our model is predicting target variable with almost 97% accuracy)

```
> ct

     0   1
0 9627  40
1  228 105
```

We can see the contingency table also. Correct predictions for 0 (i.e not default) is 9627 whereas for 1 (i.e. default) is 105. More number of true not default predictions are there compared to true default predictions. This might be happening because of class imbalance in the data.

Result:

We fitted a logistic regression model to predict whether a customer will default on credit card payment or not. We also came to know that there is class imbalance in our dataset because of which we are getting incorrect prediction for default class.