****

**ABHINAV VIJAYAKUMAR**

**19BCE1311**

**CSE3506 – ESSENTIALS OF DATA ANALYTICS LAB-10**

**DR. LAKSHMI PATHI JAKKAMPUTI (L21 + L22)**

**--------------------------------------------------------------------------------------------**

**Tasks for Week-10: Gradient Descent with Momentum Optimizer**

**Aim:** Apply multiple regression on mtcars dataset using momentum optimized gradient descent.

**Algorithm:**

**1.** Select columns for multiple regression

**2.** Give learn rate, gamma(momentum) and max iterations in function

**3.** Pick values for m1, m2 & c.

**4.** Initialize values for nu\_m1, nu\_m2 and nu\_c to be 0.

**5.** Initialize iteration=0

**6.** If iteration < max\_iteration

**a.** Calculate y\_pred

**b.** Calculate loss function

**c.** Update nu\_m1, nu\_m2 and nu\_c using the below formula:

* + 1. Nu\_m1 = gamma \*nu\_m1 + alpha \* sum((y\_pred-y) \* x1)
    2. Nu\_m2 = gamma \*nu\_m2 + alpha \* sum((y\_pred-y) \* x2)
    3. Nu\_c = gamma \* nu\_c+alpha \* sum(y\_pred-y)

**d.** Update m1, m2, c and Lf

**e.** Print intercept, slope and loss function

**7.** Repeat step 5 continuously.

**8.** Use lm function to check for linear model.

**STATISTICS:**

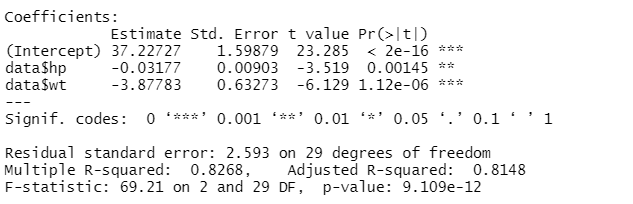
**1. Values using Momentum Optimizer:**

|  |  |
| --- | --- |
| FIELD | VALUE |
| C | 37.2272414172067 |
| M1 | -3.87782187933926 |
| M2 | -0.0317729604979703 |

****

**2. Values using lm function for Multilinear Regression:**

|  |  |
| --- | --- |
| FIELD | VALUE |
| C | 37.22727 |
| M1 | -0.03177 |
| M2 | -3.87783 |

****

**INFERENCES :**

1. In momentum gradient descent loss function is not important but in gradient descent loss function is important as it is used for convergence.
2. If we put gamma as 0 the model behaves like gradient descent.
3. We can decrease the learning rate or increase the number of iterations to increase the accuracy.

**Program:**

#Momentum based Gradient Descent

mgd=function(x1,x2,y,m1,m2,c,alpha,gamma,iter){

iterations=0

#Lf=0

nu\_m1=0

nu\_m2=0

nu\_c=0

while(iterations<=iter){

y\_pred=m1\*x1+m2\*x2+c

Lf\_new=0.5\*sum((y\_pred-y)^2)

nu\_m1=gamma\*nu\_m1+alpha\*sum((y\_pred-y)\*x1)

nu\_m2=gamma\*nu\_m2+alpha\*sum((y\_pred-y)\*x2)

nu\_c=gamma\*nu\_c+alpha\*sum(y\_pred-y)

m1=m1-nu\_m1

m2=m2-nu\_m2

c=c-nu\_c

Lf=Lf\_new

iterations=iterations+1

}

paste("Optimal intercept:",c,"Optimal slope:",m1,m2,"Loss function",Lf)

}

data=mtcars

mgd(data$wt,data$hp,data$mpg,-0.2,-0.2,32,0.000002,0.98,50000)

model=lm(data$mpg~data$hp+data$wt)

summary(model)