Lab_5

Register Number:19BLC1186

Name:Tarun Sidhu

Lab Exercise No:5

Date:22/2/2021

Dataset: Cleaned_data

Task: I) For the given dataset design binary classifiers that can predict the severity level of Covid-19. Demonstrate how resampling methods can improve the estimation of the model performance.

II. For the given dataset design binary classifiers that can predict the difficulty in breathing for Covid-19 cases. Demonstrate how model selection and regularisation techniques can be employed to improve the model accuracy.

```
library(tidyverse)
library(boot)
Auto=read.csv("/Users/tarunsidhu/Desktop/Sem 4/ML/ML(Lab)/Data Sets/Cleaned-Data.c
sv")
names(Cleaned_Data)
ggplot(Cleaned_Data, aes(Severity_Severe,Severity_None)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE) +
  geom_smooth(method = "lm", formula = y ~ poly(x, 1), se = FALSE, linetype = 1) +
  geom_smooth(method = "lm", formula = y ~ poly(x, 2), se = FALSE, linetype = 2) +
  geom_smooth(method = "lm", formula = y ~ poly(x, 3), se = FALSE, linetype = 3)
set.seed(1)
sample <- sample(c(TRUE, FALSE), nrow(Cleaned_Data), replace = T, prob = c(0.6,0.4
))
train <- Cleaned_Data[sample, ]
test <- Cleaned_Data[!sample, ]

# loop for first ten polynomial
mse.df <- tibble(degree = 1:10, mse = NA)

for(i in 1:10) {
  lm.fit <- lm(Severity_None ~ poly(Severity_Severe, i), data = train)
  mse.df[i, 2] <- mean((test$Severity_None - predict(lm.fit, test))^2)
}

ggplot(mse.df, aes(degree, mse)) +
  geom_line() +
  geom_point() )

glm.fit <- glm(Severity_None ~ Severity_Severe, data = Cleaned_Data)
coef(glm.fit)
```

```
glm.fit <- glm(Severity_None ~Severity_Severe , data = Cleaned_Data)

loocv.err <- cv.glm(Cleaned_Data, glm.fit)

str(loocv.err)
loocv.err$delta[1]
loocv_error <- function(x) {
  glm.fit <- glm(Severity_None ~ poly(Severity_Severe, x), data =Cleaned_Data)
  cv.glm(auto, glm.fit)$delta[1]
}

library(purrr)
1:5 %>% map_dbl(loocv_error)

kfcv_error <- function(x) {
  glm.fit <- glm(Severity_None ~ poly(Severity_Severe, x), data = Cleaned_Data)
  cv.glm(auto, glm.fit, K = 10)$delta[1]
}
```

```
## Error: <text>:26:16: unexpected ')'
## 25:   geom_line() +
## 26:   geom_point() )
##                    ^
```