Summary

Xining Yuan

9/12/2020

Build the data

```
library(bis557)
library(usethis)
lm_patho<-read.table("C:/Users/Xining/Dropbox/My PC (DESKTOP-Q77OU8J)/Desktop/Campus/First Year/First S
use_data(lm_patho, overwrite = TRUE)
#> v Setting active project to 'C:/Users/Xining/Dropbox/My PC (DESKTOP-Q77OU8J)/Desktop/Campus/First Ye
#> v Saving 'lm_patho' to 'data/lm_patho.rda'
#> * Document your data (see 'https://r-pkgs.org/data.html')
```

#Build the linear model function

```
linear_model<-function(fo,d, contrasts = NULL){
  d_no_na<-model.frame(fo, d)
  X<-model.matrix(fo, d_no_na, contrasts)
  y_name<-as.character(fo)[2]
  Y<-matrix(d_no_na[,y_name], ncol=1)
  beta<-gradientR(Y,X,2.65,10000000000)
  ret <- list("coefficients"=t(beta$coefficients),"cost"=t(beta$cost))
  ret
}
```

#Build the gradient descent function

```
gradientR<-function(Y, X, eta, iters){
  epsilon=0.00000001
  set.seed(0)
  N = dim(X)[1]
  theta.init = as.matrix(rnorm(n=dim(X)[2], mean=0,sd = 1))
  theta.init = t(theta.init)
  e = t(Y) - theta.init%*%t(X)
  grad.init = -(2/N)%*%(e)%*%X
  theta = theta.init - eta*(1/N)*grad.init
  l2loss = c()
  for(i in 1:iters){
    l2loss = c(l2loss,sqrt(sum((t(Y) - theta%*%t(X))^2)))
    e = t(Y) - theta%*%t(X)
    grad = -(2/N)%*%e%*%X
    theta = theta - eta*(2/N)*grad
    if(sqrt(sum(grad^2)) <= epsilon){
      comments = c("no","yes","NA")
      for (l in 1:length(comments)) {
        #if (!is.na(comments[l])) print(comments[l])
```

```
        if (comments[1] != "NA") print(comments[1])
      }
      break
      }
  }
  values<-list("coefficients" = t(theta),"cost"=t(l2loss))
  return(values)
}
```