Lab3

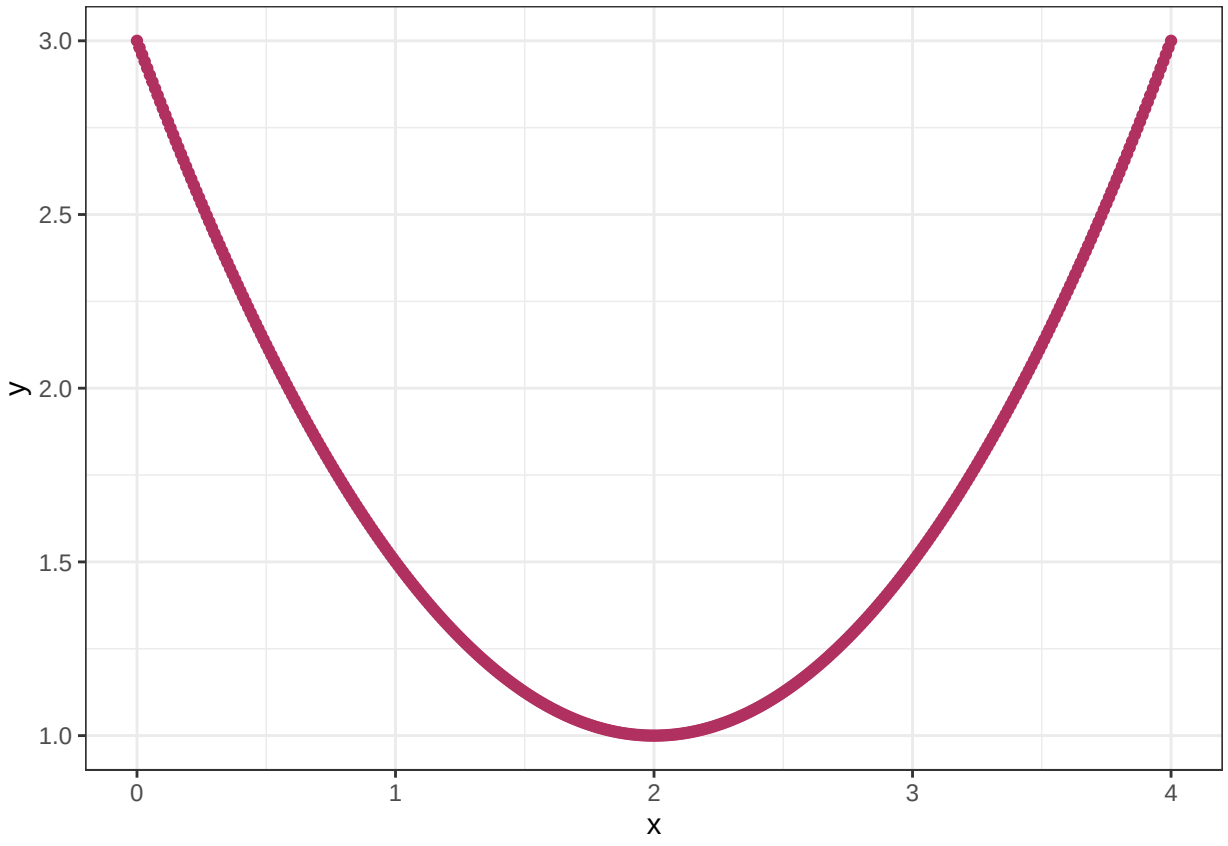
Kevin Luo

2019/9/23

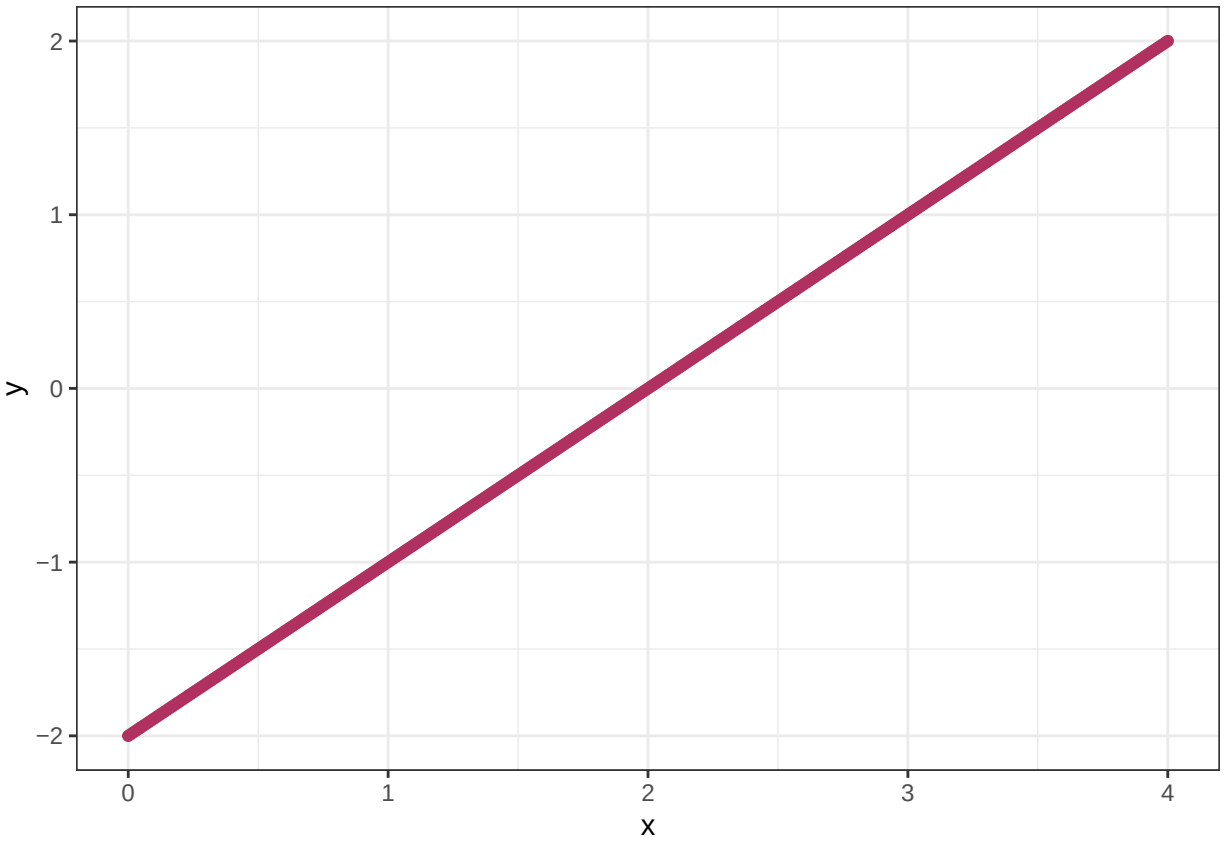
Part 1: Minimizing a quadratic

1.1

```
f <- function(x){
  return(0.5*(x-2)^2+1)
}
df <- function(x){
  return(x-2)
}
temp <- seq(0,4,0.01)
tempplotf <- data.frame(x = temp, y = f(temp))
tempplotdf <- data.frame(x = temp, y = df(temp))
tempplotf %>% ggplot() + theme_bw() +
  geom_point(aes(x = x, y = y), color = 'maroon')
```




```
tempplotdf %>% ggplot() + theme_bw() +
  geom_point(aes(x = x, y = y), color = 'maroon')
```



1.2

```
x = rnorm(1)
alpha = 1
for (i in 1:10){
  x = x - alpha*df(x)
}
x
```

```
## [1] 2
```

```
# x = 2 is obviously the minimum for f(x)
```

1.3

```
alpha <- 0.1
x <- rnorm(2)
A <- diag(c(1,2), nrow = 2)
b <- matrix(c(1,1), ncol = 1)
for (i in 1:100){
  x = x - alpha*(A %*% x - b)
}
x
```

```
##          [,1]
## [1,] 0.999963
## [2,] 0.500000
```

Part 2: OLS Regression

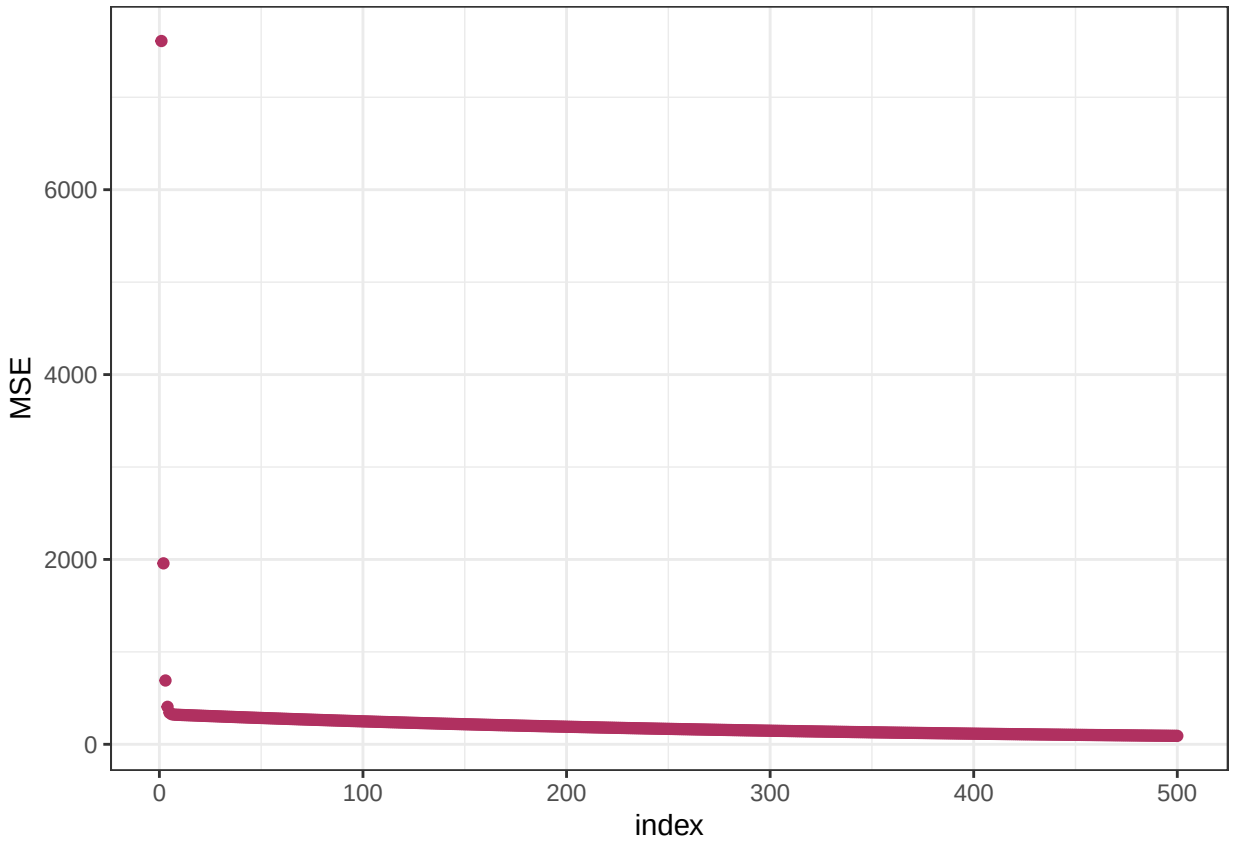
```
response <- 'mpg'
predictors <- c('hp', 'qsec', 'wt')
M <- as.matrix(mtcars[ ,predictors])
X <- cbind(intercept = 1, M)
y <- mtcars[ ,response]
y <- as.matrix(y, ncol = 1)
```

```
test <- lm(mpg~hp+qsec+wt, data = mtcars)
test$coefficients / length(y)
```

```
##  (Intercept)           hp         qsec           wt
##  0.862828964 -0.000556946  0.015963553 -0.136212413
```

```
MSE <- function(beta){
  return(t(X %*% beta - y) %*% (X %*% beta - y) / nrow(X))
}
dMSE <- function(beta){
  return(2 * (t(X) %*% X %*% beta - t(X) %*% y) / nrow(X))
}
```

```
beta <- matrix(rnorm(4), ncol = 1)
alpha <- 0.00001
iteration <- 500
plotdata <- data.frame(index = 1:iteration, MSE = rep(0,iteration))
for (i in 1:iteration){
  beta = beta - alpha * (dMSE(beta))
  plotdata[i,'MSE'] = MSE(beta)
}
plotdata %>%
  ggplot(aes(x = index, y = MSE)) + theme_bw() +
  geom_point(color = 'maroon')
```

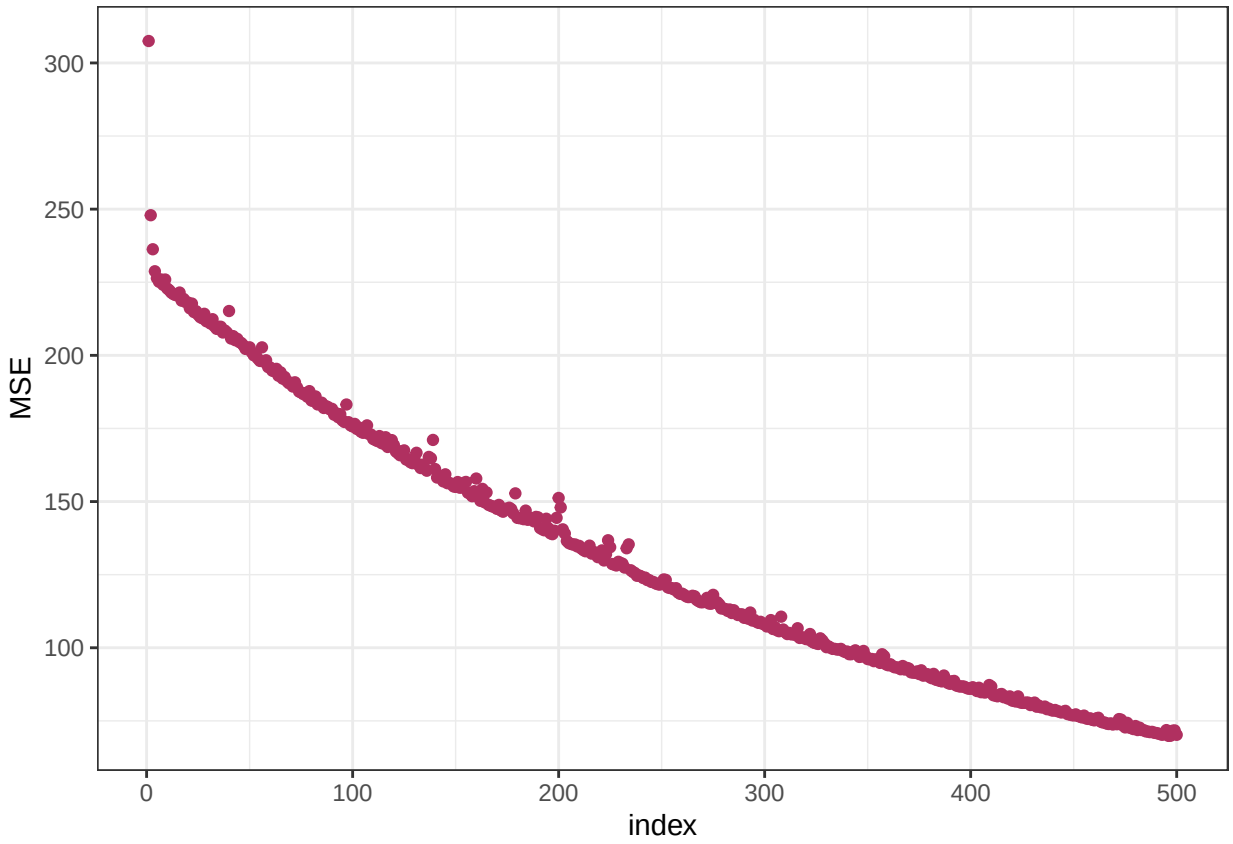



```
sample_batch <- function(n, B){
  return(sample(1:n,B))
}
B <- 16
alpha <- 0.00001
beta <- matrix(rnorm(4), ncol = 1)
plotdata2 <- data.frame(index = 1:iteration, MSE = rep(0,iteration))

for (i in 1:500){
  sampleRows <- sample_batch(nrow(X), B)
  tempX <- X[sampleRows,]
  tempy <- matrix(y[sampleRows,], ncol = 1)
  beta = beta - alpha * 2 * (t(tempX) %*% tempX %*% beta - t(tempX) %*% tempy) / nrow(tempy)
  plotdata2[i,'MSE'] = MSE(beta)
}
beta
```

```
##                 [,1]
## intercept -0.07583735
## hp         0.02642154
## qsec       0.58433473
## wt         0.66847470
```

```
plotdata2 %>%
  ggplot(aes(x = index, y = MSE)) + theme_bw() +
  geom_point(color = 'maroon')
```


MSE
300
250
200
150
100
0
100
200
300
400
500
index