Homework 02 - STAT440

Joseph Sepich (jps6444)

09/04/2020

Problem 1

Which of the following is an appropriate variable name?

- (a) 1st_var
- (b) first_var
- (c) first.var

first_var or choice b is the appropriate variables name of the three choices. Variables cannot start with a number and using a dot in the variable name can be confused with function syntax.

Problem 2

Recall that if $x := (x_1, ..., x_d) \in R^d$, then the euclidean norm of x is $||x||_2 = \sqrt{\Sigma_{i=1}^d x_i^2}$. Let

$$V = [v_1, v_2, v_3, v_4, v_5] = \begin{vmatrix} 1 & 2 & 4 & -1 & 0 \\ 2 & 1 & -4 & 1 & 3 \\ 3 & 0 & 1 & -1 & 5 \end{vmatrix}$$

Create matrix V in R:

```
mat_v <- matrix(c(1, 2, 3, 2, 1, 0, 4, -4, 1, -1, 1, -1, 0, 3, 5), nrow = 3, ncol=5)
mat_v
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]    1    2    4   -1    0
## [2,]    2    1   -4    1    3
## [3,]    3    0    1   -1    5
```

Use R to do the following

2a

Create a matrix D made out of the norm of all pairwise distances of the column vectors of V. That is, the ij^{th} entry of D is $||v_i - v_j||_2$.

```r
l2_norm <- function(vec) {
  sqrt(sum(vec^2))
}

num_cols <- dim(mat_v)[2]
mat_d <- matrix(1:25, nrow = num_cols, ncol = num_cols)
for (i in 1:num_cols) {
  for (j in 1:num_cols) {
    mat_d[i, j] <- l2_norm(mat_v[,i] - mat_v[,j])
  }
}
mat_d
```

```
##          [,1]     [,2]     [,3]     [,4]     [,5]
## [1,] 0.000000 3.316625 7.000000 4.582576 2.449490
## [2,] 3.316625 0.000000 5.477226 3.162278 5.744563
## [3,] 7.000000 5.477226 0.000000 7.348469 9.000000
## [4,] 4.582576 3.162278 7.348469 0.000000 6.403124
## [5,] 2.449490 5.744563 9.000000 6.403124 0.000000
```

2b

Use D to compute the average and standard deviation of these distances. Be careful not to double count.

```r
dists <- mat_d[upper.tri(mat_d,diag=TRUE)]
print('Average:')
```

```
## [1] "Average:"
```

```r
print(mean(dists))
```

```
## [1] 3.63229
```

```r
print('Standard Dev:')
```

```
## [1] "Standard Dev:"
```

```r
print(sd(dists))
```

```
## [1] 3.140712
```

2c

Find vectors y_j so that the j^{th} of D_{y_j} is the average distance from v_j to all other points. Report these numbers.

Problem 3

3a

Build a simple linear regression function using ordinary least squares that takes two inputs x and y, fits y to x, and returns the slope and intercept. Use it to fit the **iron** column to the **calcium** column in the **nutrient** dataset.

```
ols_regress <- function(x, y) {
  slope_numerator <- cov(x, y)
  slope_denom <- var(x)
  slope <- slope_numerator / slope_denom
  inter <- mean(y) - slope * mean(x)
  return(list("slope" = slope, "intercept" = inter))
}
```

```
# load dataset
nutrient_df <- read.csv('./data/nutrient.csv')

# perform regression
model <- ols_regress(nutrient_df$calc, nutrient_df$iron)
print('Slope:')
```

```
## [1] "Slope:"
```

```
print(model$slope)
```

```
## [1] 0.005956363
```

```
print('Intercept:')
```

```
## [1] "Intercept:"
```

```
print(model$intercept)
```

```
## [1] 7.412836
```

3b

Learn how to use the R function **lm** and use it to fit iron to calcium. Use the **summary** function on the output of **lm** and compare it to the output of your function in (a).

```
model <- lm(iron~calc,data=nutrient_df)
summary(model)
```

```
##
## Call:
## lm(formula = iron ~ calc, data = nutrient_df)
##
```

```
## Residuals:
##     Min      1Q  Median      3Q     Max
## -16.029  -3.432  -0.799   2.401  45.907
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 7.4128358  0.3774502   19.64   <2e-16 ***
## calc        0.0059564  0.0005103   11.67   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.5 on 735 degrees of freedom
## Multiple R-squared:  0.1564, Adjusted R-squared:  0.1552
## F-statistic: 136.2 on 1 and 735 DF,  p-value: < 2.2e-16
```

The output of the lm function regression of fitting **iron** to **calcium** has the same estimate for the intercept and slope.