Ghoshal_HW1

Gourav Ghoshal

September 8, 2018

Questions 1.

Data generation and matrix indexing

1) Generate a vector with 25 elements and each element independently follows a normal distribution (with mean =0 and sd=1);

```
x <- rnorm(n=25, mean = 0, sd = 1)
print(x)
```

```
##  [1] -0.606725380  1.547943646  1.115017265  0.650105568 -0.771196676
##  [6] -0.039830773  0.750422042  1.660862565 -1.442221802  0.002157856
## [11]  0.754334617 -0.121668261 -0.349785505 -0.656366908 -1.446677238
## [16] -1.152988247  0.299417361  0.873632321 -0.391963595 -0.293937258
## [21] -0.254738603 -1.098431881  1.298658656  0.712130057 -0.851868865
```

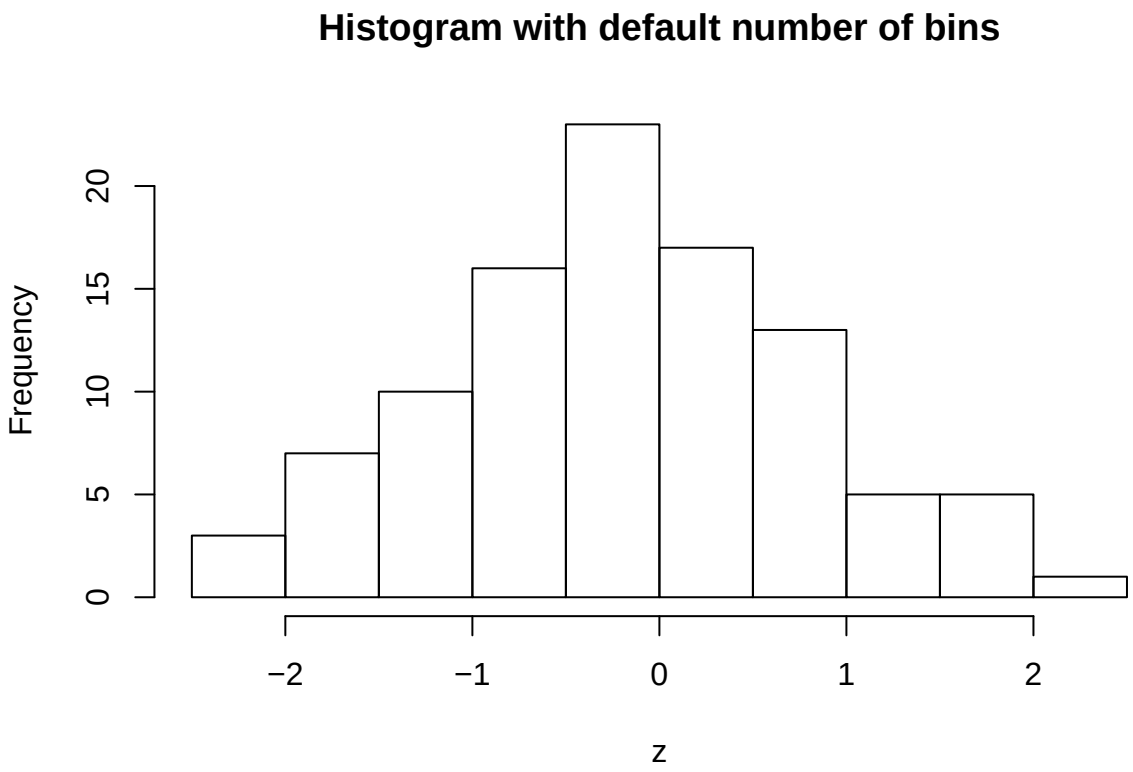
2) Reshape this vector into a 5 by 5 matrix in two ways (arranged by row and column);

```
y <- matrix(x, nrow = 5, byrow = T)
print(y)
```

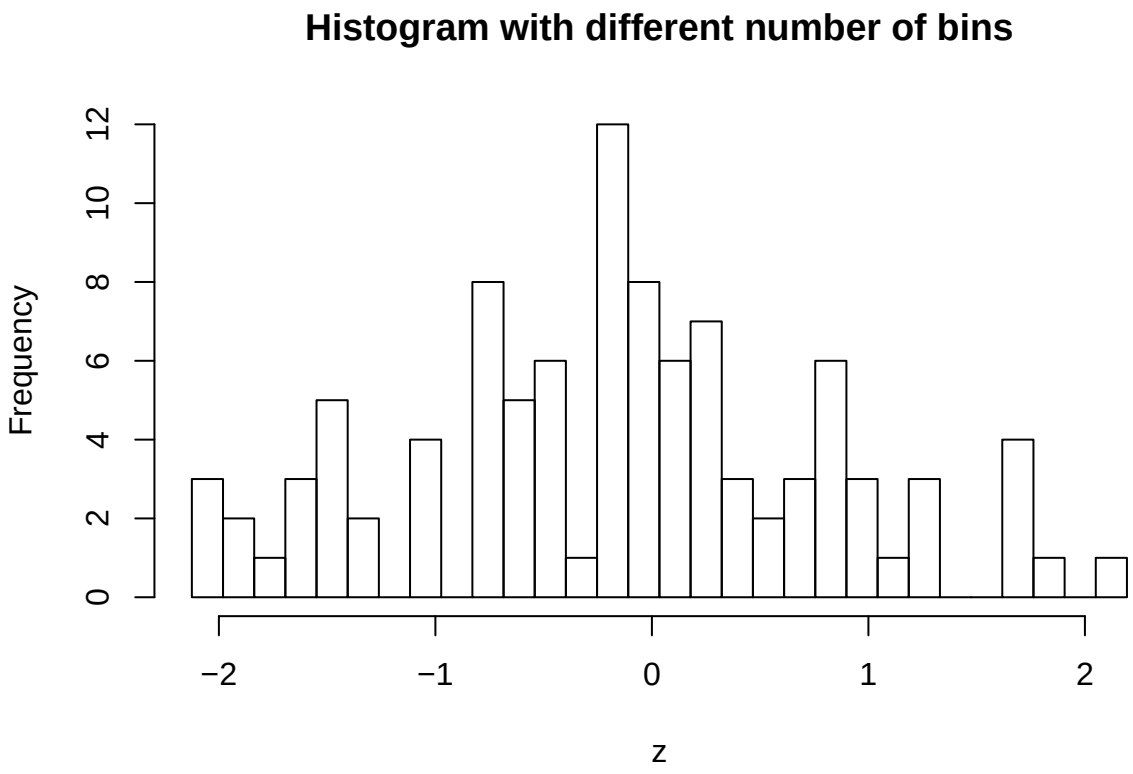
```
##             [,1]       [,2]       [,3]       [,4]         [,5]
## [1,] -0.60672538  1.5479436  1.1150173  0.6501056 -0.771196676
## [2,] -0.03983077  0.7504220  1.6608626 -1.4422218  0.002157856
## [3,]  0.75433462 -0.1216683 -0.3497855 -0.6563669 -1.446677238
## [4,] -1.15298825  0.2994174  0.8736323 -0.3919636 -0.293937258
## [5,] -0.25473860 -1.0984319  1.2986587  0.7121301 -0.851868865
```

3) Similarly, generate another vector with 100 elements and plot its histogram.

```
z <- rnorm(n=100, mean = 0, sd = 1)
hist(z,main = 'Histogram with default number of bins')
```




```
number_of_bins <- 30
hist(z, breaks = seq(min(z), max(z), l = number_of_bins+1),
     main = 'Histogram with different number of bins')
```



4) Provide screenshots of the R code used for the above questions as well as the plots in the report.Explain the plots in your own words.

ans -

- **R-Code -** The "code chunk" contains the code and corresponding plots are also available.
- **explanation of plots -** The vector of 100 elements was generated from a standard normal distribution. The default histogram generated from this vector looks like normal and the histogram with 20 bins has more of a normal shape.

Questions 2.

Upload the Auto data set, which is in the ISLR library. Understand information about this data set by either ways we introduced in class (like "?Auto" and names(Auto))

```
library(ISLR)
```

```
## Warning: package 'ISLR' was built under R version 3.4.2
```

```
print('About the Auto data...')
```

```
## [1] "About the Auto data..."
```

```
names(Auto)
```

```
## [1] "mpg"          "cylinders"    "displacement" "horsepower"
## [5] "weight"       "acceleration" "year"         "origin"
## [9] "name"
```

```
dim(Auto)
```

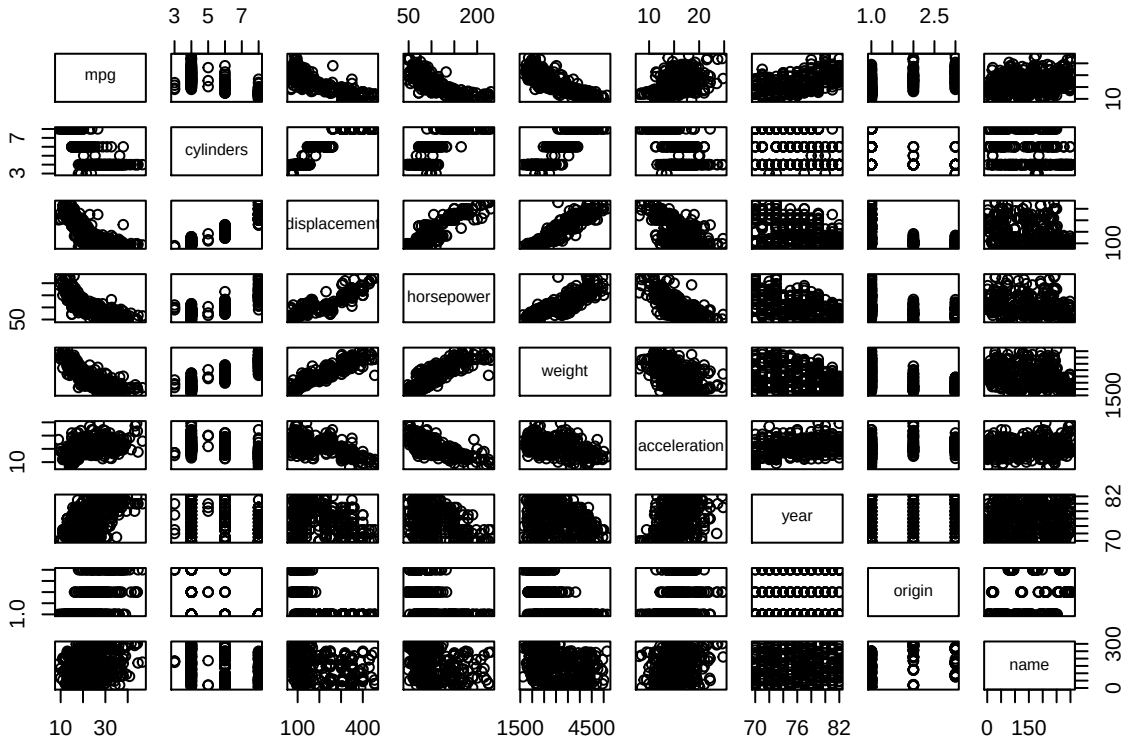
```
## [1] 392   9
```

```
?Auto
```

```
## starting httpd help server ... done
```

Question 3.Make a scatterplot between every pair of the following variables (try to plot all scatterplots in one figure; hint: use pairs() command): "mpg", "displacement", "horsepower", "weight", "acceleration". By observing the plots, do you think the two variables in each scatterplot are correlated? If so, how?

```
pairs(Auto)
```



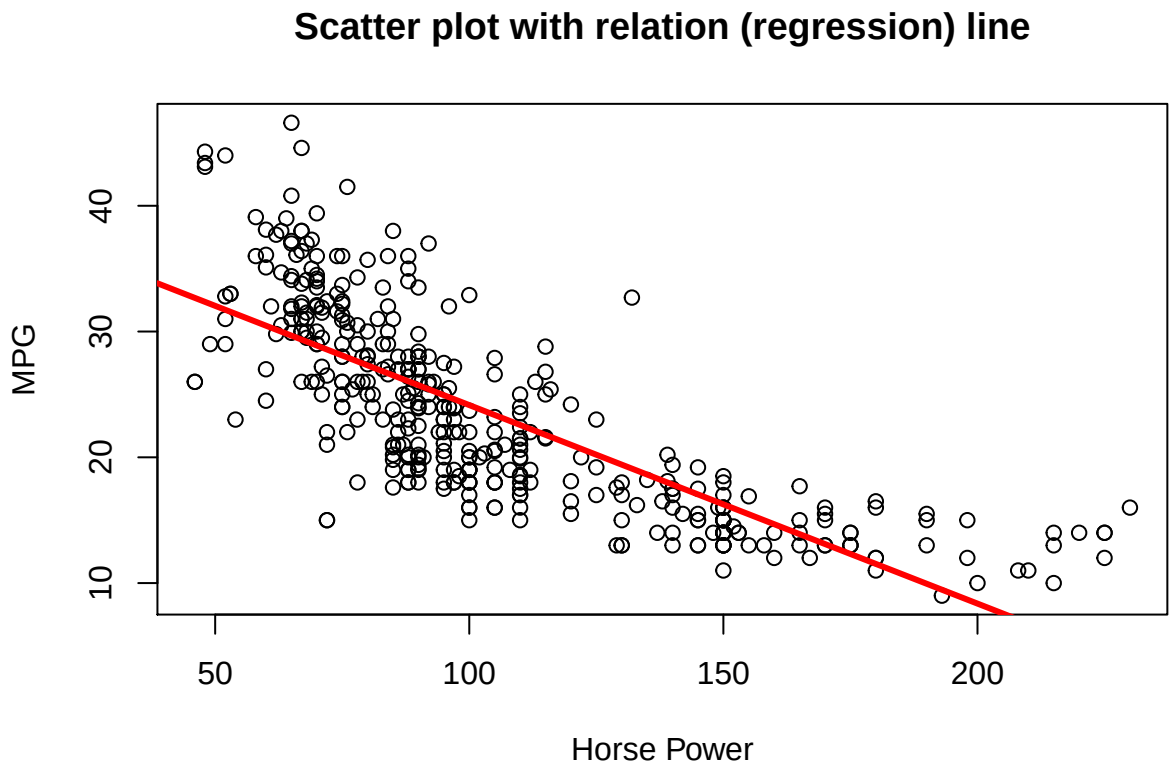
ans -

- Some of the variables are highly correlated, which is evident from the pair plot.
- The variable 'mpg' is highly corelated with 'displacement', 'horsepower', 'weight', 'year' and the correlation reduces with 'acceleration', and there is no correlation with 'name'.
- Similarly, variable 'displacement' is positively corelated with 'horsepower','weight' and negatively correlated with 'accelerations'.
- Horsepower is positively correlated with Weight and negatively correlated (weakly) with acceleration
- Weight is weakly, negatively correlated with acceleration.
- In addition, origin and name variables are not useful while checking correlation! ... ...

4. Draw a line on the scatterplot of mpg vs. horsepower to represent relationship between the two variables.

```
# scatter plot
plot(Auto$horsepower,Auto$mpg, xlab = 'Horse Power', ylab = 'MPG', main = 'Scatter plot with relation (

# line to express relationship
abline(lm(Auto$mpg ~ Auto$horsepower), col = 2, lwd = 3)
```



5. Is there a better way to represent their relationship rather than the linear model you just drew? (No need to use mathematical formula. Just draw something on the figure)

```
# scatter plot
plot(Auto$horsepower,Auto$mpg, xlab = 'Horse Power', ylab = 'MPG', main = 'Scatter plot with relation (

# line to express relationship
fit_poly <- lm(Auto$mpg ~ poly(Auto$horsepower, 2, raw = T))
lines(sort(Auto$horsepower), fitted(fit_poly)[order(Auto$horsepower)], col = 4, lwd = 5)
```