hwk2

Patrick Wheeler

9/21/2020

Reaction time, measured from the moment the driver first sees the danger until he or she gets a foot on the brake pedal, is thought to follow a Normal model with a mean of 1.5 seconds and a standard deviation of 0.18 seconds.
(a) (not to turn in) Use the 68-95-99.7 rule to draw and label the Normal model for yourself.
(b) Write a sentence or two describing the reaction times. The middle 95% of drivers take between 1.14 and 1.86 seconds to put their foot on the pedal.
(c) What percentage of drivers have a reaction time of less than 1.25 seconds?

```r
mean = 1.5
sd = 0.18
ans_1c <- pnorm(1.25, mean=mean, sd=sd)*100
```

8.243327%
(d) What percentage of drivers have reaction times between 1.6 and 1.8 seconds?

```r
norm_d16 <- pnorm(1.6, mean=mean, sd=sd)
norm_d18 <- pnorm(1.8, mean=mean, sd=sd)
ans_1d <- (norm_d18-norm_d16)*100
```

24.1467008%
(e) Describe the reaction times of the slowest 1/3 of all drivers.

```r
#slowest means high reaction times
minSlowestTime <- qnorm(1/3, mean=mean, sd=sd, lower.tail=FALSE)
z_e <- (minSlowestTime-mean)/sd
```

The reaction times of the slowest 1/3 of drivers are at least 1.5775309 seconds at 0.4307273 standard deviations from the mean.
(f) What is the interquartile range of reaction times? (hint: run pnorm(-.675))

```r
#z = (x-m)/s, x = z*s+m
iqr_f <- 2*abs(qnorm(0.25, mean=mean, sd=sd))
```

Interquartile range: 2.7571837

2. Here are the summary statistics for the weekly payroll of a small company: lowest salary = \$300, mean salary = \$700, median = \$500, range = \$1200, IQR = \$600, Q1 = \$350, standard deviation = \$400. highest salary = \$300+\$1200 = \$1500

(a) Do you think the distribution of salaries is symmetric, skewed right, or skewed left? Explain why. The distribution is likely skewed right since the mean salary is higher than the median.

(b) Between what two values are the middle 50% of the salaries found? The middle 50% are between \$350 and \$950. #Q3 = IQR + Q1

(c) Suppose business has been good and the company gives every employee a \$50 raise. State the new value of each of the above summary statistics.
lowest salary = \$350, mean salary = \$750, median = \$550, range = \$1200, IQR = \$600, Q1 = \$400, standard deviation = \$400.

(d) Suppose instead the company gives each employee a 10% raise. State the new value of each of the summary statistics.
lowest salary = \$330, mean salary = \$770, median = \$`500*1.1`, range = \$1980, IQR = \$600, Q1 = \$385, standard deviation = \$ 440.

3. A cereal factory has a machine that fills its "16 ounce" boxes. The distribution of weight of cereal it actually puts into a box can be well approximated with a Normal distribution with a mean of 16.3oz and a standard deviation of 0.2oz.

(a) What percentage of boxes are under weight?

```
mean_3 <- 16.3
sd_3 <- 0.2
underweight <- 100*pnorm(16, mean=mean_3, sd=sd_3)
```

6.6807201% are underweight.
(b) Suppose that the company lawyer insists that no more than 5% of the boxes can be underweight. Unfortunately nothing can be done about the standard deviation of the machine's distribution, however the mean can be altered. To what mean value should the machine be set to satisfy the lawyer and minimize waste? #come back and think on #b

```
mean_3b <- 16.3
#means_3b <- c(16.4, 16.5, 16.6, 16.7, 16.8, 16.9, 17, 17.1, 17.2, 17.3)
means_3b <- c(16.31, 16.32, 16.33, 16.34)
sd_3 <- 0.2
ans_3b <- c()
for (m in means_3b) {
  print(m)
  ans <- 100*pnorm(16, mean=m, sd=sd_3)
  print(ans)
  ans_3b <- append(ans_3b, ans)
}
```

```
## [1] 16.31
## [1] 6.057076
## [1] 16.32
## [1] 5.479929
## [1] 16.33
## [1] 4.947147
## [1] 16.34
## [1] 4.456546
```

The best mean would be about 16.33oz since it keeps 4.9471468% of the boxes to be underweight, which is about 5%.

4. Consider the following four points: (200,1950), (400,1650), (600,1800), and (800,1600). The mean of the x values is m_x = 500 and the standard deviation s_x = 258.2, while the mean of the y values is m_y = 1750 and the standard deviation s_y = 158.1.

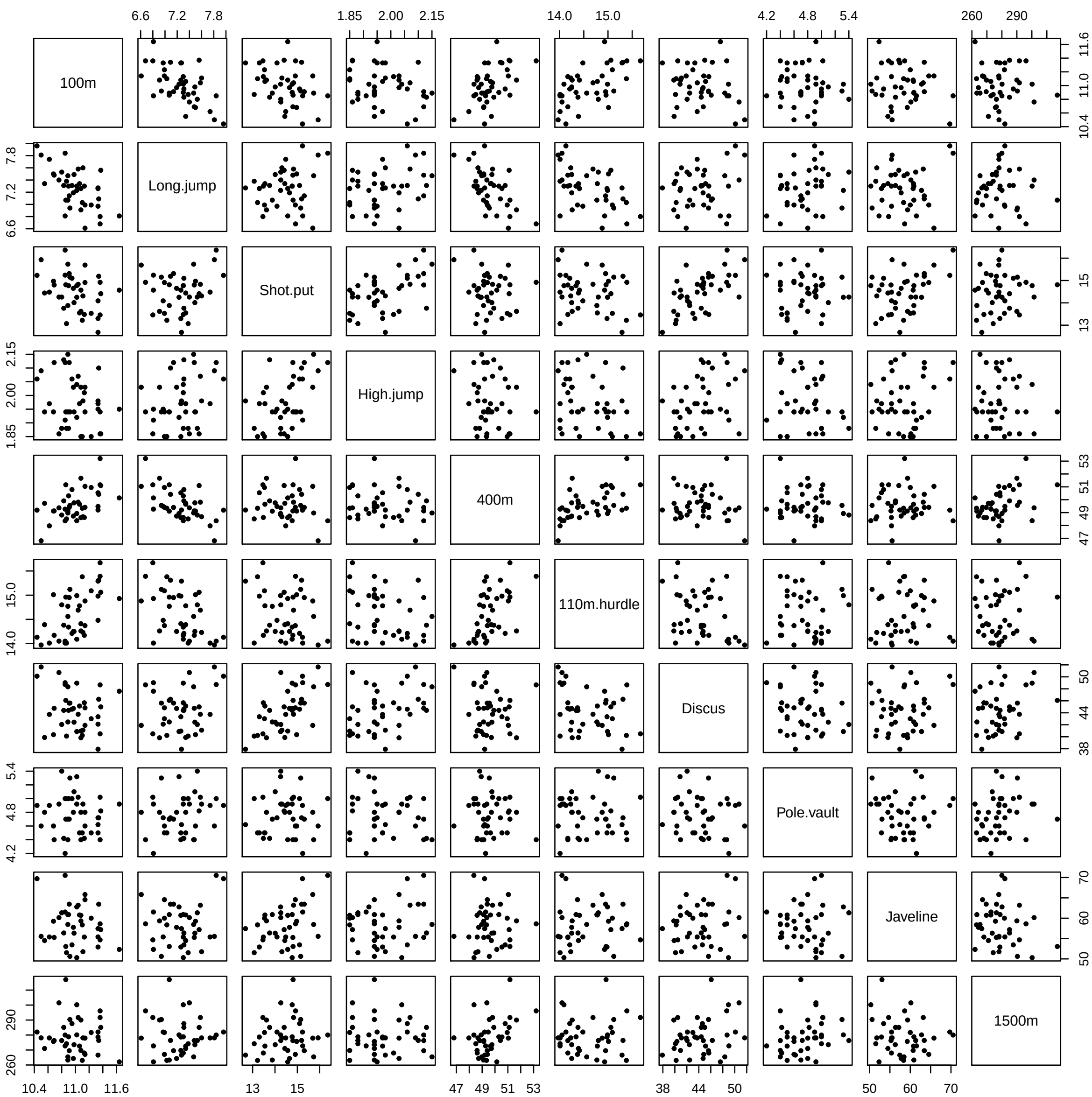
(a) Use these summary statistics to calculate the z-scores for each x and y value, then calculate the correlation coefficient (r).

```
m_x <- 500
s_x <- 258.2
m_y <- 1750
s_y <- 158.1
x <- c(200, 400, 600, 800)
y <- c(1950, 1650, 1800, 1600)
z_4x <- c()
z_4y <- c()
productSum <- 0
i <- 1
while (i <= 4) {
  z_4x <- append(z_4x, (x[i]-m_x)/s_x)
  z_4y <- append(z_4y, (y[i]-m_y)/s_y)
  productSum <- productSum+z_4x[i]*z_4y[i]
  i <- i+1
}
correlationCoef = productSum/3
```

Z-Scores for each pt (x, y):
(200,1950): -1.16189, 1.2650221
(400,1650): -0.3872967, -0.6325111
(600,1800): 0.3872967, 0.3162555
(800,1600): 1.16189, -0.9487666
r = -0.7349083
(b) We will rarely give much weight on to a summary statistic like the correlation coefficient for a sample size so small. In a sentence or two explain why.
The smaller the dataset, the higher the likelyhood that any correlation is coincidental. We need a large dataset to be confident that the correlation isn't just an accident.

5. The Decathlon is a track and field competition in which contestants participate in 10 events (long jump, 100 meter, shot put, etc). How might we compare contestants' results across multiple events?
We could make a distribution for each event and compare the contestants' results by their z-scores (how many standard deviations above the mean time or score their result was). This comparison is imperfect because it doesn't account for the possibility that all the contestants for one event may perform better than another group does at theirs.
Include and run the following code chunk:

```
pairs(decathlon[1:10],pch = 20)
```




```r
R = cor(decathlon[, 1:10])
round(R, 3)
```

```
##              100m Long.jump Shot.put High.jump   400m 110m.hurdle Discus
## 100m        1.000    -0.599   -0.356    -0.246  0.520       0.580 -0.222
## Long.jump  -0.599     1.000    0.183     0.295 -0.602      -0.505  0.194
## Shot.put   -0.356     0.183    1.000     0.489 -0.138      -0.252  0.616
## High.jump  -0.246     0.295    0.489     1.000 -0.188      -0.283  0.369
## 400m        0.520    -0.602   -0.138    -0.188  1.000       0.548 -0.118
## 110m.hurdle 0.580    -0.505   -0.252    -0.283  0.548       1.000 -0.326
## Discus     -0.222     0.194    0.616     0.369 -0.118      -0.326  1.000
## Pole.vault -0.083     0.204    0.061    -0.156 -0.079      -0.003 -0.150
## Javeline   -0.158     0.120    0.375     0.172  0.004       0.009  0.158
```

```
## 1500m       -0.061    -0.034    0.116    -0.045  0.408       0.038  0.258
##             Pole.vault Javeline  1500m
## 100m            -0.083   -0.158 -0.061
## Long.jump        0.204    0.120 -0.034
## Shot.put         0.061    0.375  0.116
## High.jump       -0.156    0.172 -0.045
## 400m            -0.079    0.004  0.408
## 110m.hurdle     -0.003    0.009  0.038
## Discus          -0.150    0.158  0.258
## Pole.vault       1.000   -0.030  0.247
## Javeline        -0.030    1.000 -0.180
## 1500m            0.247   -0.180  1.000
```

(a) Briefly describe what this plot shows and how to read it.

(b) Choose two events whose results have a strong positive correlation, create a scatter plot for just these two, state the linear correlation coefficient. Why do you think these two events in particular are positively correlated?
r = 0.520
The 100m and 400m are positively correlated because both test runners for their speed, therefore both events will result in fast runners winning.

(c) Choose two events whose results have a strong negative correlation, create a scatterplot for just these two, state the linear correlation coefficient. Why do you think these two events in particular are negatively correlated?

```r
#pairs(decathlon[1:10],pch = 20)
decathlon[1]
```

```
##              100m
## SEBRLE      11.04
## CLAY        10.76
## KARPOV      11.02
## BERNARD     11.02
## YURKOV      11.34
## WARNERS     11.11
## ZSIVOCZKY   11.13
## McMULLEN    10.83
## MARTINEAU   11.64
## HERNU       11.37
## BARRAS      11.33
## NOOL        11.33
## BOURGUIGNON 11.36
## Sebrle      10.85
## Clay        10.44
## Karpov      10.50
## Macey       10.89
## Warners     10.62
## Zsivoczky   10.91
## Hernu       10.97
## Nool        10.80
## Bernard     10.69
## Schwarzl    10.98
```

```
## Pogorelov   10.95
## Schoenbeck  10.90
## Barras      11.14
## Smith       10.85
## Averyanov   10.55
## Ojaniemi    10.68
## Smirnov     10.89
## Qi          11.06
## Drews       10.87
## Parkhomenko 11.14
## Terek       10.92
## Gomez       11.08
## Turi        11.08
## Lorenzo     11.10
## Karlivans   11.33
## Korkizoglou 10.86
## Uldal       11.23
## Casarsa     11.36
```

r = -0.602
The long jump requires a relatively short burst of speed relative to the 400 meter so athletes who perform well in one are unlikely to succeed in the other.