609 HW#2

Jose Mawyin

2/22/2020

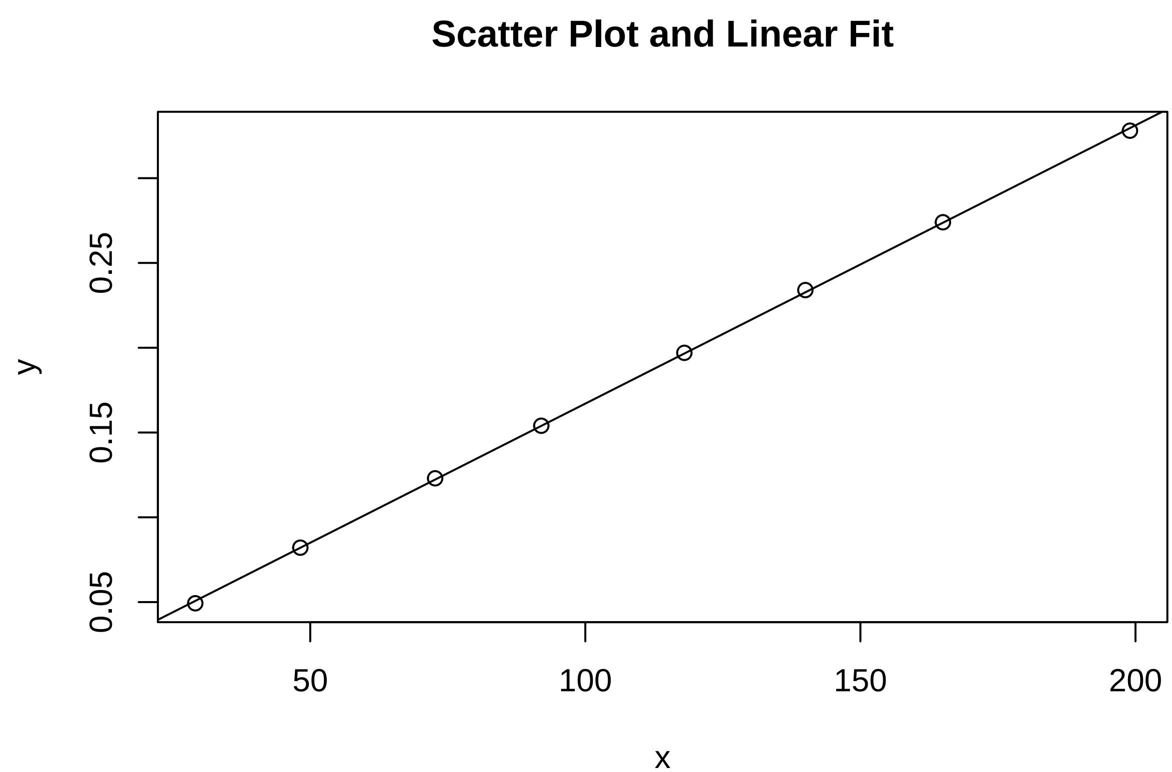
1.For each of the following data sets, formulate the mathematical model that minimizes the largest deviation between the data and the line y = ax+b. If a computer is available, solve for the estimates of a and b. (Chapter3: Model Fitting, Pg.121)

```
## X:
```

```
## [1]  29.1  48.2  72.7  92.0 118.0 140.0 165.0 199.0
```

```
## Y:
```

```
## [1] 0.0493 0.0821 0.1230 0.1540 0.1970 0.2340 0.2740 0.3280
```




```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
```

```
##        Min         1Q     Median         3Q        Max
## -0.0015241 -0.0002946  0.0001804  0.0004996  0.0013066
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 2.926e-03  8.316e-04   3.518   0.0126 *
## x           1.641e-03  6.869e-06 238.929 3.63e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.001063 on 6 degrees of freedom
## Multiple R-squared:  0.9999, Adjusted R-squared:  0.9999
## F-statistic: 5.709e+04 on 1 and 6 DF,  p-value: 3.627e-13
```

The y intercept is 2.926e-03 and the slope is 1.641e-03.

===

2. a. In the following data, W represents the weight of a fish (bass) and l represents its length. Fit the model $W = kl^3$ to the data using the least-squares criterion. (Chapter3: Model Fitting, Pg.136)

```
##
## Call:
## lm(formula = weight ~ length^3)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.1317 -1.2943 -0.1228  1.0279  2.9519
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -57.4862     5.0859  -11.30 2.87e-05 ***
## length        5.8329     0.3481   16.76 2.89e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.884 on 6 degrees of freedom
## Multiple R-squared:  0.9791, Adjusted R-squared:  0.9756
## F-statistic: 280.8 on 1 and 6 DF,  p-value: 2.885e-06
```

b. In the following data, g represents the girth of a fish. Fit the model $W = klg^2$ to the data using the least-squares criterion.

```
##
## Call:
## lm(formula = weight ~ length * (girth^2))
##
## Residuals:
##        1        2        3        4        5        6        7        8
##  0.57095  0.75027 -0.03051 -0.42905  0.08118  0.00134 -0.02537 -0.91882
##
## Coefficients:
```

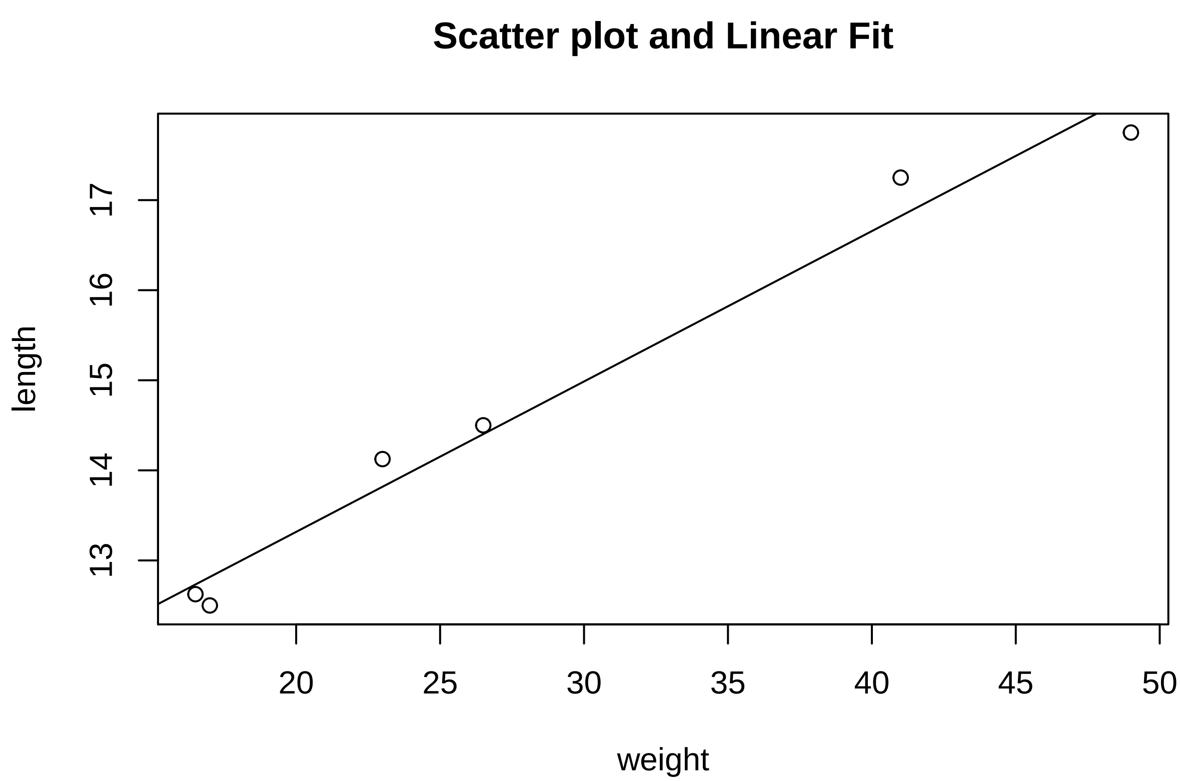
```
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -5.5702    21.8792  -0.255   0.8116
## length         0.2729     1.2227   0.223   0.8343
## girth         -2.0399     2.8403  -0.718   0.5124
## length:girth   0.3390     0.1455   2.330   0.0802 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6938 on 4 degrees of freedom
## Multiple R-squared:  0.9981, Adjusted R-squared:  0.9967
## F-statistic: 703.6 on 3 and 4 DF,  p-value: 6.703e-06
```

c. Which of the two models fits the data better? Justify fully. Which model do you prefer? Why?

The second model $W = klg^2$ fits the data better. The goodness-of-fit measure R-squared of 0.9981 indicates that almost all of the variability in the data is explainec by the regresion model.

==

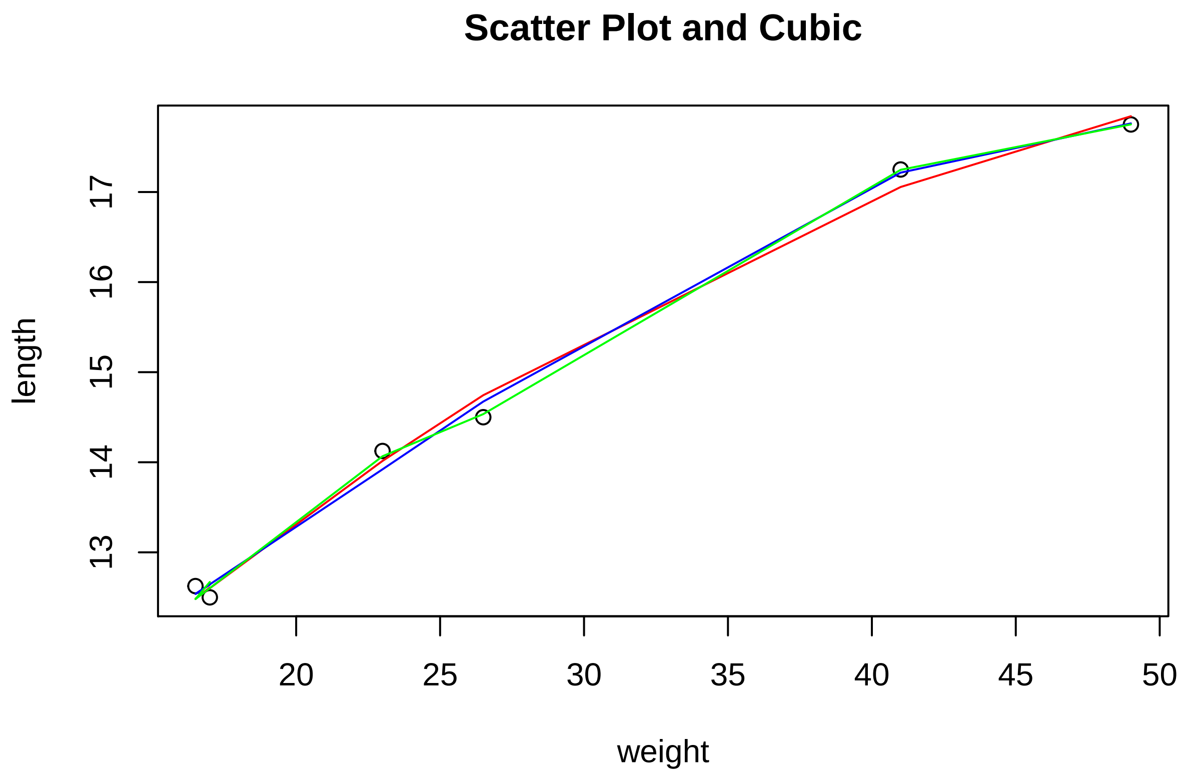
3.Construct a scatterplot of the given data. Is there a trend in the data? Are any of the data points outliers? Construct a divided difference table. Is smoothing with a low-order polynomial appropriate? If so, choose an appropriate polynomial and fit using the least-squares criterion of best fit. Analyze the goodness of fit by examining appropriate indicators and graphing the model, the data points, and the deviations. (Chapter4: Experimental Modeling, Pg.169)



```
##
## Call:
## lm(formula = length ~ poly(weight, 2))
##
## Coefficients:
##      (Intercept)  poly(weight, 2)1  poly(weight, 2)2
##          14.7917            4.9705           -0.6441
```

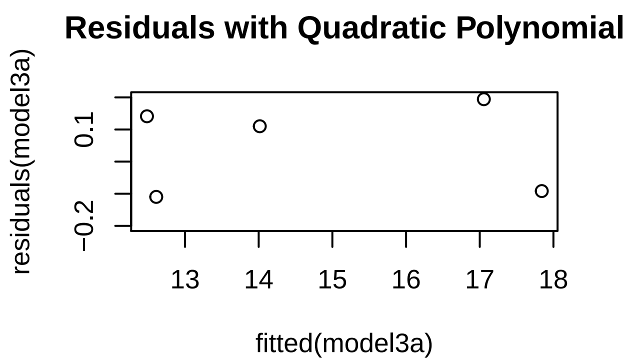
```
##
## Call:
## lm(formula = length ~ poly(weight, 3))
##
## Coefficients:
##      (Intercept)  poly(weight, 3)1  poly(weight, 3)2  poly(weight, 3)3
##          14.7917            4.9705           -0.6441           -0.2225
```

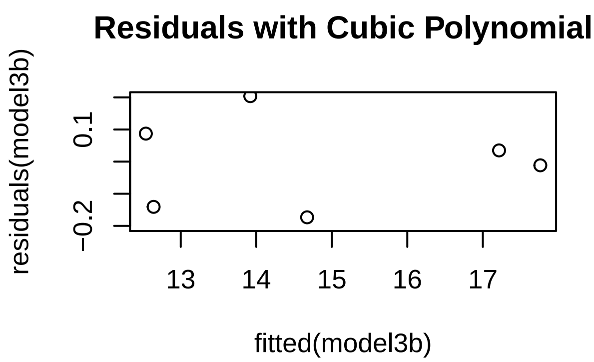
```
##
## Call:
## lm(formula = length ~ poly(weight, 4))
##
## Coefficients:
##      (Intercept)  poly(weight, 4)1  poly(weight, 4)2  poly(weight, 4)3
##          14.7917            4.9705           -0.6441           -0.2225
## poly(weight, 4)4
##          -0.2138
```

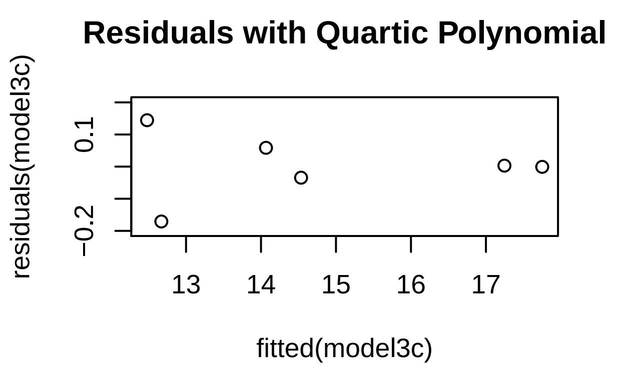


Data x_i	y_i	Δ	Δ^2	Δ^3	Δ^4	Δ^5
12.5	17					
		-4				
12.625	16.5		5.13			
		4.33		-1.23		
14.125	23		2.67		0.08	
		9.33		-0.86		0.06
14.5	26.5		-1.30		0.42	
		5.27		1.27		
17.25	41		3.30			
		16				
17.75	49					

Figure 1: Difference Table







We see above how the residuals decrease as the order of the polynomial increases improving the fit. However, the risk with higher order polynomials is that of overfitting the data.

===

4. Use the middle-square method to generate (Chapter5: Simulation Modeling, Pg.194):

a. 10 random numbers using x0 = 1009.

```
## n1_rdn_list
## 1808 2656 3805 4169 4780 5433 6875 6886 8484 9782
##    1    1    1    1    1    1    1    1    1    1
```

b. 20 random numbers using x0 = 653217.

```
## n2_rdn_list
## 111857 115502 119884 177129 302674 310673 401635 407120 485617 551929
##      1      1      1      1      1      1      1      1      1      1
## 625621 692449 721734 746694 746826 749074 761776 823870 899966 938801
##      1      1      1      1      1      1      1      1      1      1
```

c. 15 random numbers using x0 = 3043.

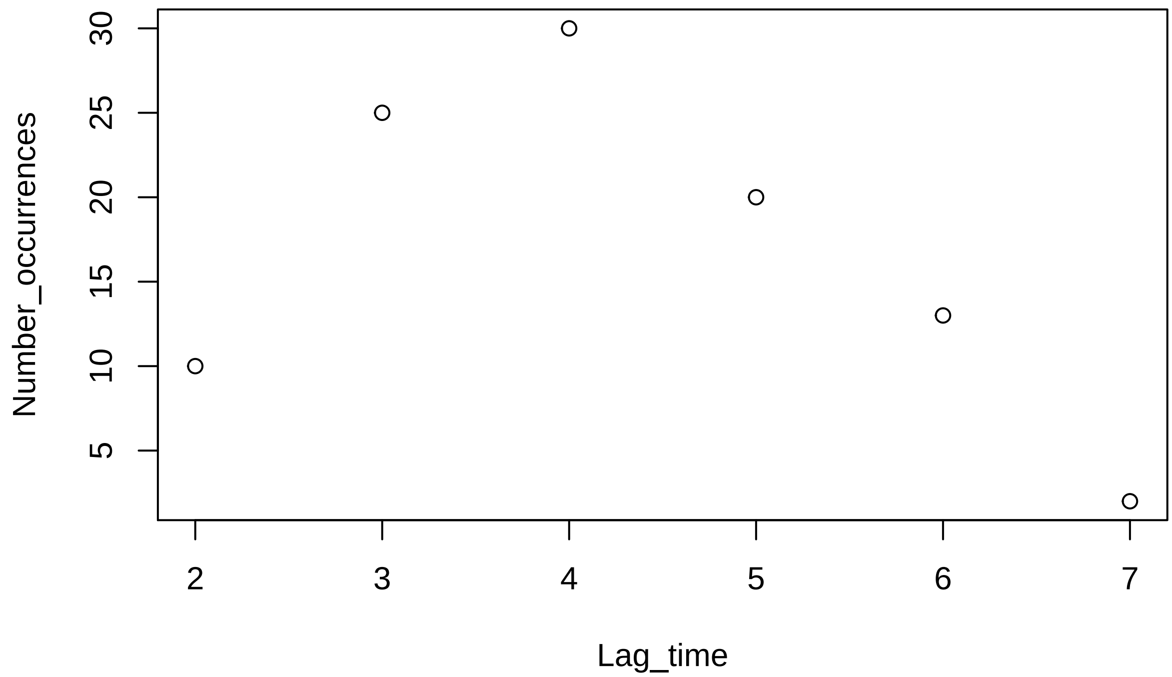
```
## n3_rdn_list
## 1429 1569 1761 2796 3187 3736 4204 5984 6176 6736 6997 7764 8082 9576 9580
##    1    1    1    1    1    1    1    1    1    1    1    1    1    1    1
```

d. Comment about the results of each sequence. Was there cycling? Did each sequence degenerate rapidly?

We can see above that the middle-square method produced non-repeated results based on the given seed and output number. Other seed numbers could have shown cycling and sequence degeneration but the given values for this exercise did not.

===

5.In many situations, the time T between deliveries and the order quantity Q is not fixed. Instead, an order is placed for a specific amount of gasoline. Depending on how many orders are placed in a given time interval, the time to fill an order varies. You have no reason to believe that the performance of the delivery operation will change. Therefore, you have examined records for the past 100 deliveries and found the following lag times, or extra days, required to fill your order:



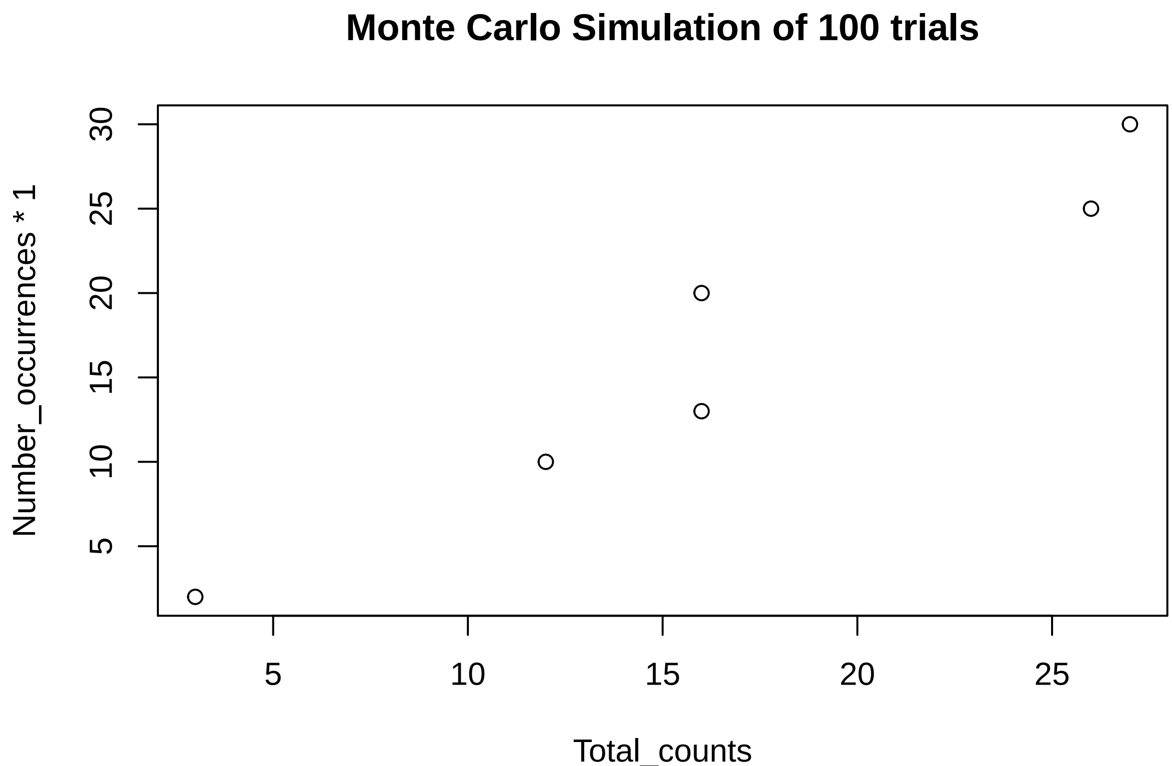
Construct a Monte Carlo simulation for the lag time submodel. If you have a handheld calculator or computer available, test your submodel by running 1000 trials and comparing the number of occurrences of the various lag times with the historical data. (Chapter5: Simulation Modeling, Pg.211)

Random number	Corresponding Lag	Percent occurrence
0 <= x < 0.1	2	0.1
0.1 <= x < 0.35	3	0.25
0.35 <= x < 0.65	4	0.3
0.65 <= x < 0.85	5	0.2
0.85 <= x < 0.98	6	0.13
0.98 <= x <= 1	7	0.02

In the three following plots, we see how accurately the Monte Carlo model matches the observed data when the number of trials are:

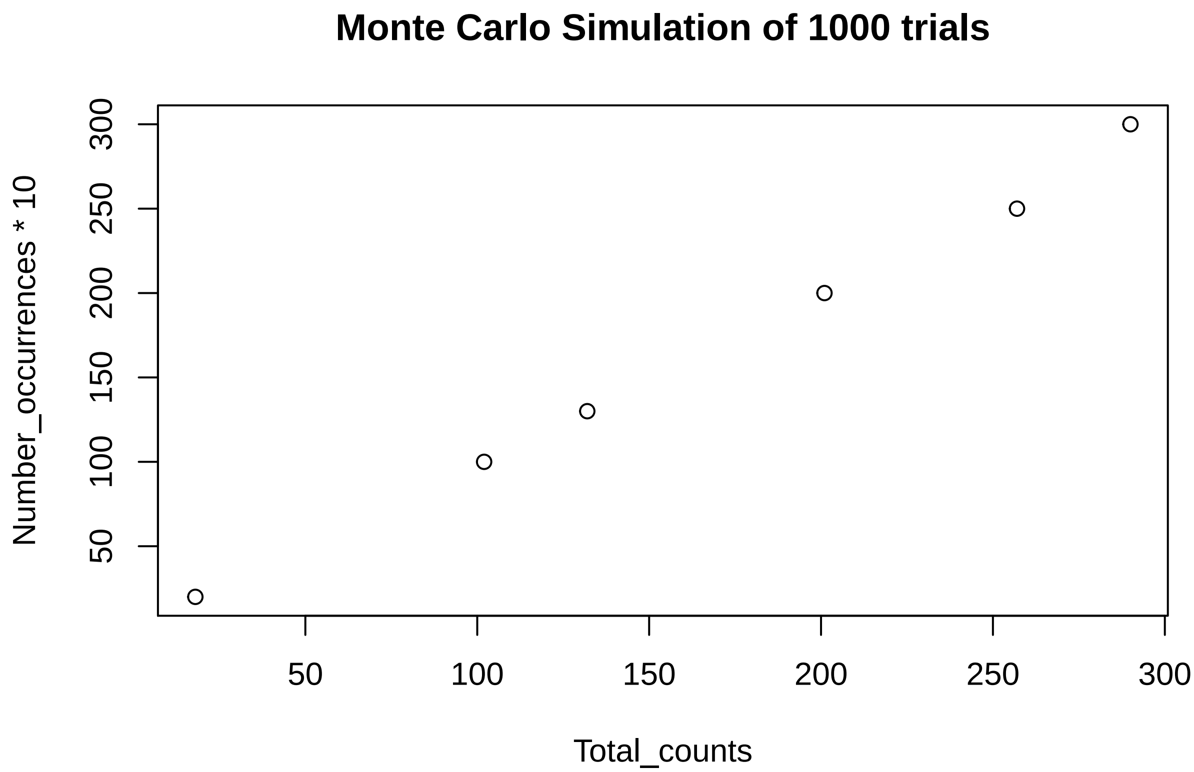
100 Trials

```
## Simulated Occurrences:
##  12 26 27 16 16 3
## Observed Occurrences:
##  10 25 30 20 13 2
```



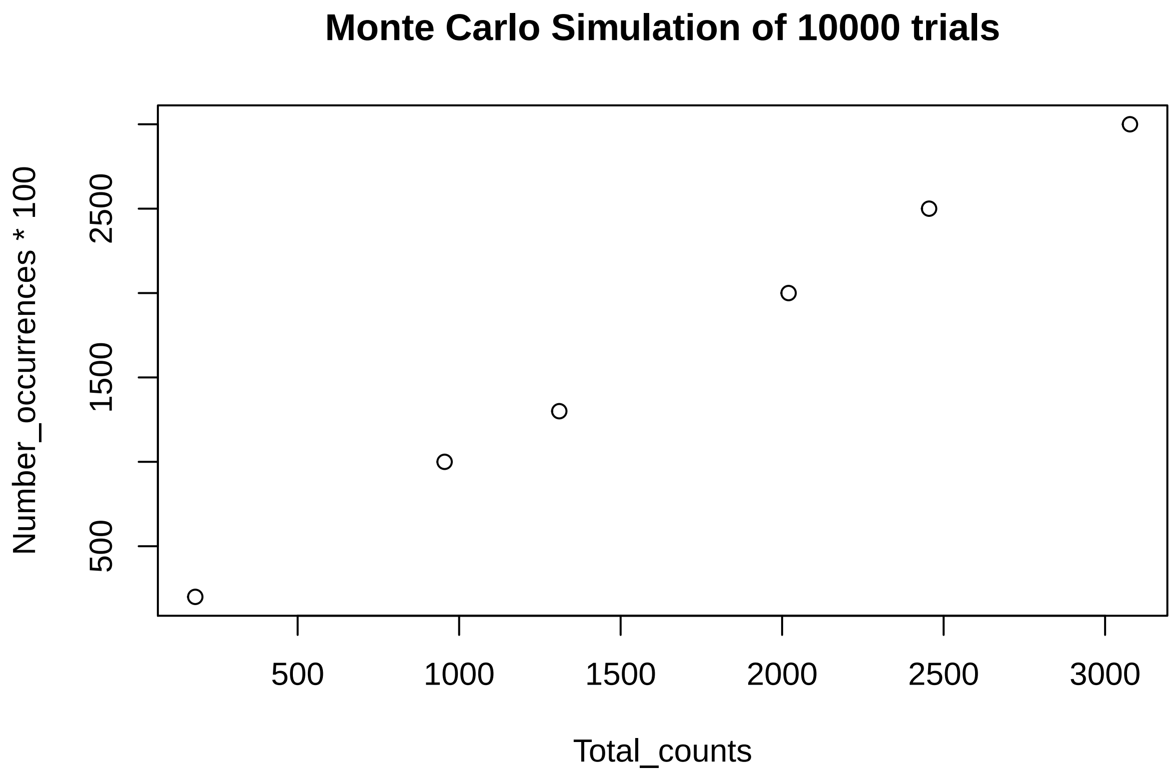
1000 Trials

```
## Simulated Occurrences:
##  102 257 290 201 132 18
## Observed Occurrences:
##  100 250 300 200 130 20
```



10000 Trials

```
## Simulated Occurrences:
##  955 2455 3077 2020 1310 183
## Observed Occurrences:
##  1000 2500 3000 2000 1300 200
```


Monte Carlo Simulation of 10000 trials
Number_occurrences * 100
500
1500
2500
500
1000
1500
2000
2500
3000
Total_counts