Problem Set 0

CS 169 Problem Set 0

Due Friday, September 15, 2023

1. Poisson and Binomial Simulations

a. (4 points) Exercise 1.7 in text: Generate, and save in a numeric vector called `instances`, 100 instances of a Poisson random variable with lambda = 3. What are the mean and variance of the simulated values? What is the second highest value in the vector?

```
# insert your R code here
set.seed(123)
instances<-rpois(100,lambda=3)
instances
```

```
##   [1] 2 4 2 5 6 0 3 5 3 3 6 3 4 3 1 5 2 0 2 6 5 4 3 8 4 4 3 3 2 1 6 5 4 4 0 3 4
##  [38] 2 2 2 1 2 2 2 1 1 2 3 2 5 0 3 4 1 3 2 1 4 5 2 4 1 2 2 4 3 4 4 4 3 4 3 4 0
##  [75] 3 2 2 3 2 1 2 4 2 4 1 3 7 5 5 1 1 4 2 4 2 1 4 1 3 3
```

```
mean(instances)
```

```
## [1] 2.94
```

```
var(instances)
```

```
## [1] 2.622626
```

```
sort(instances)[length(instances)-1]
```

```
## [1] 7
```

Any text you want to write should go in here

b. (4 points) Modified from the exercise in textbook section 1.3.3 (p.6): With a binomial distribution function, count the number of simulated mutations along a genome with 5,000 positions, where a "success" (i.e., a simulated value of 1) represents a mutation. Use a mutation rate of 8e-4. Do 1000 such simulations and count the total number of mutations observed in each; save these counts in a vector called `sims`. (You should not need to use a for loop for this; it can be done with a single binomial distribution function.)

Print out the first five entries in `sims`, corresponding to the total number of mutations you found in each of your first five trials.

```r
set.seed(123)
sims = rbinom(n=1000, prob=8e-4, size=5000)
sims
```

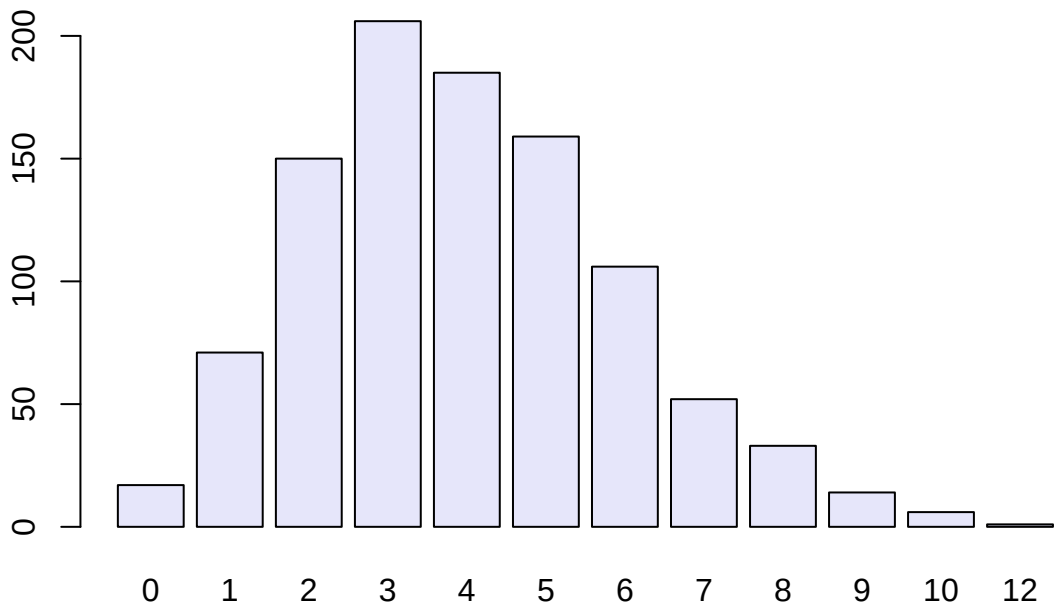
```
##    [1]  3  6  3  6  7  1  4  7  4  4  8  4  5  4  2  7  3  1  3  8  7  5  5 10
##   [25]  5  5  4  4  3  2  8  7  5  6  1  4  5  2  3  2  2  3  3  3  2  2  2  4
##   [49]  3  6  1  4  6  2  4  2  2  5  7  3  5  2  3  3  6  4  6  6  6  4  5  5
##   [73]  5  0  4  2  3  4  3  2  3  5  3  6  2  4  9  7  6  2  2  5  3  5  3  2
##   [97]  5  2  4  4  4  3  4  8  4  7  7  4  3  2  7  3  1  7  5  2  4  8  4  3
##  [121]  5  3  3  2  3  9  2  1  2  5  4  7  5  5  4  5  6  6  9  4  3  3  0  2
##  [145]  6  2  3  1  3  5  6  4  3  3  2  3  4  2  4  2  4  3  5  3  3  4  5  2
##  [169]  3  3  5  2  6  5  5  4  3  4  6  4  6  3  5  3  4  4  3  4  7  7  3  3
##  [193]  9  4  7  4  3  5  2  4  3  8  4  4  3  6  3  3  2  2  4  3  2  5  1  5
##  [217]  3  3  6  7  3  8  5  5  1  3  4  4  5  7  4  3  4  1  3  3  2  6  2  6
##  [241]  4  5  2  5  3  5  3  8  8  5  3  2  4  3  4  6  2  3  4  6  7  6  5  8
##  [265]  4  4  3  3  1  4  6  0  1  2  5  5  8  4  1  5  5  2  3  2  1  3  1  2
##  [289]  1  5  3  2  1  6  5  6  9  2  2  6  5  0  5  5  5  4  2  0  4  4  3  4
##  [313]  5  1  3  6  6  2  3  6  6  3  2  5  2  1 12  1  3  7  4  3  5  6  3  4
##  [337]  5  5  5  8  3  2  4  2  4  3  9  3  2  4  2  8  4  2  4  9  5  3  3  6
##  [361]  2  2  7  3  3  5  2  0  3  4  3  3  6  4  4  8  5  2  3  8  4  5  3  5
##  [385]  4  7  2  3  2  2  8  3  5  6  2  3  3  2  2  9  9  2  7  4  3  4  5  1
##  [409]  3  5  3  6  2  4  3  2  8  3  4  1  4  5  4  3  7  4  3  3  2  6  8  4
##  [433]  3 10  2  4  2  4  2  2  3  4  5  2  6  5  5  2  3  5  4  3  3  1  2  7
##  [457]  6  5  2  2  8  4  4  2  4  3  2  5  3  6  2  6  3  5  5  4  4  0  2  5
##  [481]  2  6  3  5  7  2  2  5  3  6  8  1  4  5  1 10  3  1  5  6  3  3  3  1
##  [505]  3  2  4  4  7  3  4  1  6  3  5  2  4  5  1  6  3  3  3  1  4  5  7  1
##  [529]  9  3  7  5  4  6  1  5  3  6  4  3  6  2  6  3  5  8  1  6  6  3  3  2
##  [553]  4  6  4  4  5  4  4  0  1  7  5  2  5  5  3  6  4  6  4  7  4  3  8  5
##  [577]  3  1  4  4  6  6  6  3  6  2  3  5  8  4  4  1  8  2  1  6  4  3  7  1
##  [601]  2  5  2  3  2  6  2  6  3  3  5  2  1 10  4  5  7  5  9  1  7  6  5  3
##  [625]  1  3  3  6  3  3  5  6  4  5  2  2  8  5  6  3  4  6  6  3  1  4  5  1
##  [649]  3  4  5  7  4  4  5  6  7  1  4  3  6  6  2  5  2  1  7  2  5  2  1  4
##  [673]  1  3  3  4  6  2  4  4  5  5  6  6  8  4  4  2  4  3  1  5  4  2  3  4
##  [697]  2  2  3  3  6  3  1  6  3  5  4  3  2  3  5  4  5  5  4  3  0  0 10  2
##  [721]  3  5  5  6  4  5  6  4  6  1  3  6  5  7  2  3  6  9  3  0 10  2  1  4
##  [745]  3  4  6  1  5  7  3  3  2  1  4  4  4  6  1  2  4  2  4  4  4  2  2  3
##  [769]  5  5  7  6  5  6  2  3  3  3  5  6  3  6  0  5  6  4  2  1  3  7  7  4
##  [793]  2  4  6  3  2  3  3  3  4  3  2  1  3  8  4  4  5  5  1  5  3  6  5  2
##  [817]  3  2  4  4  5  3  4  9  5  1  4  1  3  5  0  5  4  5  5  4  5  5  3  2
##  [841]  8  5  4  2  4  3  7  0  2  2  3  5  9  5  4  4  7  2  1  6  3  1  5  4
##  [865]  4  3  2  2  4  4  2  2  5  4  6  5  0  5  7  5  2  7  5  2  5  7  3  3
##  [889]  3  2  4  5  5  4  6  5  1  3  1  6  7  4  6  4  5  4  5  7  6  1  0  1
##  [913]  6  4  3  8  4  4  3  3  6  1  3  6  5  5  3  4  2  5  6  2  5  4  8  2
##  [937]  4  6  6  1  8  4  3  3  2  2  3  3  3  3  4  2  8  1  3  8  5  1  3  3
##  [961]  4  4  4  2  7  4  4  2  5  4  5  1  6  3  3  5  7  0  3  3  4  5  7  4
##  [985]  7  5  2  2  2  3  1  2  4  3  4  6  5  3  5  2
```

```r
print(sims[1:5])
```

```
## [1] 3 6 3 6 7
```

c) (4 points) Plot the distribution of the number of mutations seen in the replicates using the `barplot` function, as in Figure 1.5 in your text.

```
barplot(table(sims), col="lavender")
```



2. Extreme values; epitope example (8 points) Modifying the epitope example in Ch 1, use simulation to find the empirical probability of having a maximum value of 8 or more in 100 Poisson random variables with lambda = 0.75. (Do enough replicates to get a probability greater than zero.)

```
set.seed(123)
maxes=replicate(100000,
                {max(rpois(100,0.75))})

mean(maxes >=8)
```

```
## [1] 0.00015
```

3. The Mendelian Genetics of Oversleeping

3. Durumexcitari (DXC) is an autosomal recessive disease that causes a person to have a very hard time waking up in time for classes that start before 3pm. The incidence of DXC is roughly 1 in 2,500 individuals; approximately 1 individual in 50 is a carrier.

   Assume that a phenotypically (e.g., apparently) normal woman, Mia, marries a phenotypically normal man, Lucas, who had a biological son (Oliver) with DXC by a previous marriage. To amswer the following questions, use the symbol "A" for the normal or wildtype allele and "a" for the mutant allele. (If you don't recall enough about genetics to answer this question, see syllabus reading links for September 11.)

(a) (2 points) List all possible genotypes for Oliver, the son with DXC from Lucas' previous marriage.

Write your answer here AA, Aa, aa

(b) (2 points) What are Lucas' possible genotypes?

Write your answer here Aa

(c) (2 points) What are the possible genotypes of his second wife, Mia?

Write your answer here AA/ Aa

(d) (4 points) What is the probability that Lucus' and Mia's first child will have DXC? Show and explain your computation.

Write your answer here A dominant allele produces its phenotype whether the organism is homozygous or heterozygous, So there are two possibilities for mia's genotype: AA,Aa If mia is AA, their first child won't have DXC. Because Lucas's genotype is Aa,half of chirdren will have AA,and half probability will have Aa. If mia is Aa, the probability is 1/4. Mia is Aa, and Lucas is Aa, then ,there will be 4 probabilities. AA,Aa,Aa,aa. aa is 1/4.

(e) (6 points) If their first child does have DXC, what is the chance that the second child will be phenotypically normal? Show and explain your computation.

Write your answer here 3/4 if the first child have DXC, then the mia is Aa. And the mia is Aa, lucas is Aa. There be 4 probalities: AA, Aa, Aa, aa. A dominant allele produces its phenotype whether the organism is homozygous or heterozygous So, the second child could be AA, Aa. 75% chance that the second child will be phenotypically normal

4. Working with Data in R

a. (2 points) Use the `read.table()` command (Ch 5 in the online Irizarry text) to read the tab-delimited data from the file `testgrades.txt` into a data frame in R. This file contains the grades from three midterms and a final exam for a (fictional) class of 100 students; the first two columns contain the students' first and last names. Show your code below:

```
file_path <- "~/Desktop/testgrades.txt"
data<-read.table(file_path, header=TRUE,sep = "\t")
data
```

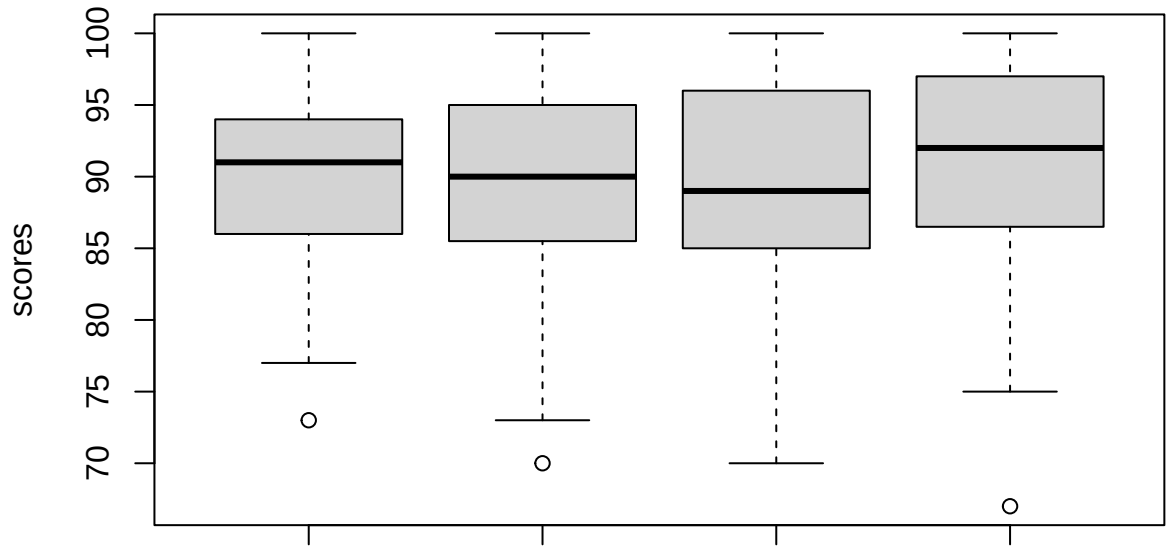
```
##          First      Last Midterm1 Midterm2 Midterm3 Final
## 1       Annika Alexander       85       87       85    77
## 2        Annie     Allan       82       89       86    81
## 3       Fallon  Anderson       85       96      100    95
## 4    Kristofer Armstrong       97       90       95    99
## 5         Lexi    Barlow       82       97       83    87
## 6      Belinda     Bowes       88       99       83    83
## 7        Sarah      Byrd       87       92       96    89
## 8     Roseanne  Caldwell       97       85       92    81
## 9       Hammad  Cantrell       77       81       85    85
```

```
## 10       Ramon    Castro       73       82       87    89
## 11     Xinghao      Chen       95       86       99    88
## 12      Jeremy     Chung       91       96       96    93
## 13        Bibi     Coles       85       97       95    94
## 14      Aleeza    Colley       92       91       97    92
## 15      Zaydan    Cooley       84       84       86    78
## 16     Michael    Cormac       94       97       86    83
## 17    Isla-Rae    Corona       88       93       90    94
## 18      Joseph     Couch       83       95       95    92
## 19     Matylda Cresswell       89       83       96    93
## 20     Everett  Cummings       93       90      100   100
## 21        Ryan Daugherty      100       93       92    90
## 22      Yasser      Dawe       97       94       97   100
## 23     Georgia    Devine       95       89       92    95
## 24       Aiden   Doherty       93       81       80    98
## 25       Aeryn  Driscoll      100       92       89    97
## 26      Junayd    Fenton       95       89       89    88
## 27        Emyr    Finley       82       96       92    81
## 28     Bridget    Finney       95       91       98   100
## 29      Keziah  Fletcher       92       96       86   100
## 30       Della    Fulton       84       79       87    85
## 31      Katrin  Gallegos       90       89       84    92
## 32        Eman     Glass       88       88       98    94
## 33  Lily-Grace   Greaves       90       88       85    95
## 34       Ellie Greenaway       94       86       88    89
## 35      Misbah    Haines       93       88       83    82
## 36      Harley     Hanna       89       83       92   100
## 37        Leon     Healy       90       85       82    85
## 38     Allison    Hebert      100       98       99    99
## 39      Montel    Herman       91      100       91    99
## 40       Kiana     Hines       91       88       89    90
## 41        Eryn    Holder       94       93       94    87
## 42  Lillie-Mae    Hughes       96       88       96    98
## 43       Vicki  Johnston      100       97      100    98
## 44    Felicity    Jordan      100       90       98    97
## 45     Brianna    Joyner       96      100       90    97
## 46      Calvin      Kidd       84       88      100    97
## 47       Alisa    Kinney       96       98       91    92
## 48     Madison    Landry       93       82       85    92
## 49        Ally       Lee       89       91       84    83
## 50         Ari    Levine       93       92       95   100
## 51    Chandler    Lucero       86       90       82    90
## 52     Rosanna    Mackay      100       94       90    90
## 53      Vikram  Manpreet      100       98      100    99
## 54        Lola   Matthis       93       90       88   100
## 55        Omer     McKee       92       85       98    85
## 56     Domonic  Mclellan       95       90       92    87
## 57      Farrah   McNeill       90       85       95    90
## 58       Kevin   Merritt       87       97       89    91
## 59      Yassin     Miray       82       70       73    80
## 60       Amani  Mohammed       94       91       89    93
## 61        Kuba    Morley       84       80       95    92
## 62      Harvey    Norman       90       85       78    88
## 63     Padraig    Obrien       80       88       85   100
```

```
## 64       Javan     Olsen       91       88       85    88
## 65      Joseph       Orr       84       85       86    82
## 66        Jack     Poole       84      100       82    98
## 67        Lyle    Porter       97       96       87    97
## 68    Harrison    Powell       81       85       75    84
## 69       Elina    Powers       97       98       83   100
## 70     Samirah     Rahim       86       90       84    86
## 71    Priyanka Rajagopal       86       92       86   100
## 72        Roan      Read       92       92       97    96
## 73      Yvonne    Reader       90       90       99    91
## 74        Safa Rodrigues       86       86       83    92
## 75   Frederick    Romero       82       84       85    77
## 76      Meerab     Rossi       97      100       95    99
## 77     Humaira    Rowley       88       91       84    93
## 78        Tony Schneider       92       95      100   100
## 79      Daisie   Sherman       92       95       91   100
## 80        Kain    Simran       96       97       87    90
## 81     Maleeha   Skinner       84       85       86    84
## 82     Makayla   Solomon       82       95       91    82
## 83     Indiana    Spears       91       90      100    91
## 84      Ashraf    Tamsin       94      100      100    92
## 85   Christine  Thornton       98       95      100   100
## 86     Russell    Thorpe       91       80       88    86
## 87      Yazmin      Tran       88       94       87    95
## 88      Brooks  Villegas       91       88       92    89
## 89       Brock    Vinson       86       95       83    98
## 90     Beverly    Weaver       88       90       89    75
## 91     Octavia     Weiss       88       87       81    85
## 92      Eileen     Wells       79       85      100    88
## 93      Steven      West       91       99       89    92
## 94      Jolyon    Weston       98       86       92    99
## 95     Tiernan     White       84       87       87    87
## 96     Montana  Whitmore       94       83       96    84
## 97      Sidney   Wormald       86       73       70    67
## 98      Jaidan     Young       86       80       96    87
## 99         Mei     Zhang       93       86       89    97
## 100     Justin      Zhou       93       83       88    93
```

b. (2 points) Using basic R graphics, create a boxplot (which by default shows the median, inter-quartile range, and outliers) for each of the four tests. Show the code and plot here:

```
boxplot(data$Midterm1, data$Midterm2, data$Midterm3, data$Final,
        data=file_path,
        ylab = "scores")
```



c. (2 points) Look at the help page for `boxplot()` to answer the following: how far do the whiskers of each box extend, relative to the size of the box itself (using default parameters)? How many outliers (i.e., students whose grade is more than that distance from the median) are there for the final exam? Write your answers below as text in the R Markdown file.

Write your answer here Each whisker extends to the furthest data point in each wing that is within 1.5 times the IQR. one outlier

d. (4 points) Which two tests have medians that are farthest apart? By how many points do they differ?

```
m1<-median(data$Midterm1)
m2<-median(data$Midterm2)
m3 <- median(data$Midterm3)
fin<-median(data$Final)
m1
```

```
## [1] 91
```

```
m2
```

```
## [1] 90
```

```
m3
```

```
## [1] 89
```

```
fin
```

```
## [1] 92
```

```
a<-c(m1,m2,m3,fin)
median_diffs <- combn(a, 2, FUN = function(x) abs(diff(x)))
median_diffs
```

```
## [1] 1 2 1 1 2 3
```

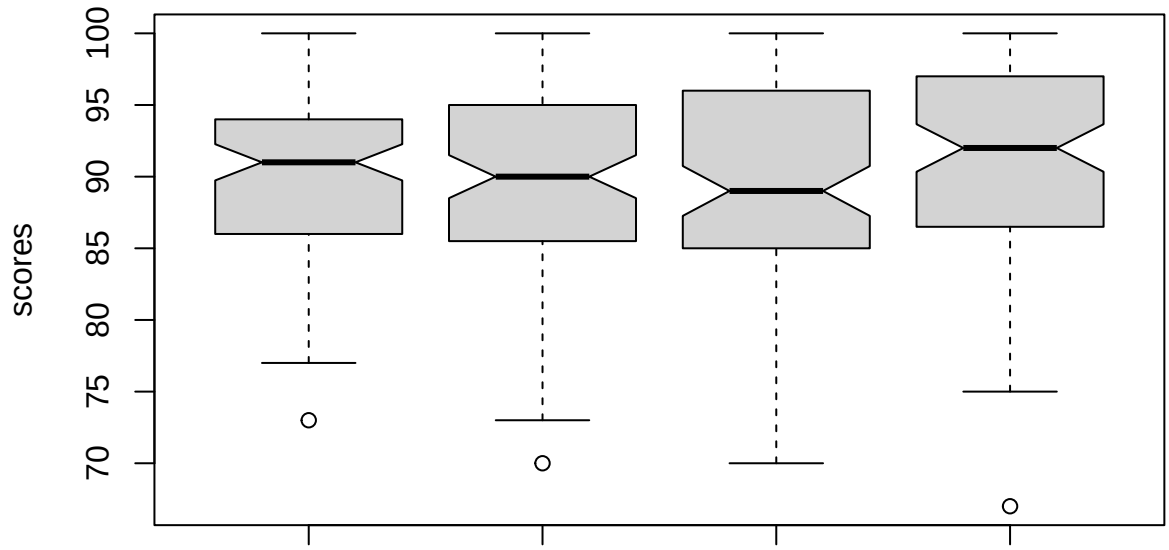
```
max<-max(median_diffs)
max
```

```
## [1] 3
```

Write your answer here The largest median is fianl:92, and the lowesr median is midterm3:89. So the fianl test and the midterm3 have medians that are farthest apart, their difference is 3.

e. (4 points) Although we haven't discussed this in class yet, do you believe those differences show one exam to be truly harder than the other? What options, among those you can find on the help page for boxplot(), might help you answer this question? Show an updated plot if necessary, and explain your conclusions.

```
boxplot(data$Midterm1, data$Midterm2, data$Midterm3, data$Final,
        data=file_path, notch = TRUE,
        ylab = "scores")
```



Write your answer here No, the difference is not a strong evidence. Both in the graph and in the data. We can all conclude that the median gap is not particularly significant. One should look for other data to determine if the test is harder.