

Lab 1

Jan Bazant

This lab is due 11:59 PM Saturday 2/9/19.

You should have RStudio installed to edit this file. You will write code in places marked “TO-DO” to complete the problems. Some of this will be a pure programming assignment. The tools for the solutions to these problems can be found in the class practice lectures. I want you to use the methods I taught you, not for you to google and come up with whatever works. You won’t learn that way.

To “hand in” the homework, you should compile or publish this file into a PDF that includes output of your code. Once it’s done, push by the deadline to your repository in a directory called “labs”.

- Print out the numerical constant pi with ten digits after the decimal point using the internal constant pi.

```
#TO-DO
options(digits = 10)
pi
```

```
## [1] 3.141592654
```

- Sum up the first 100 terms of the series $1 + 1/2 + 1/4 + 1/8 + \dots$

```
#TO-DO
sum(2^(0:-99))
```

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

```
sum((1/2)^(0:99))
```

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

- Find the product of the first 100 terms of $1 * 1/2 * 1/4 * 1/8 * \dots$

```
#TO-DO
prod(.5^(0:99))
```

```
## [1] 0
```

- Find the product of the first 500 terms of $1 * 1/2 * 1/4 * 1/8 * \dots$. Answer in English: is this answer correct?

```
#TO-DO
prod(.5^(0:499))
```

```
## [1] 0
```

- Figure out a means to express the answer more exactly. Not compute exactly, but express more exactly.

```
#TO-DO
-log10(2)*sum(0:499)
```

```
## [1] -37553.49196
```

```
# this gives the log of what we want
# so 10^this should be more accurate

# 10^(-37553.49196) # would be more accurate
10^-.49196
```

roughly $3.104273661e-375353$

- #TO-DO

```
## [1] 3.333326833
```

- #TO-DO

```
## [1] 0.46
```

```
mean(sample( c(0,1),size = 500, replace = TRUE, prob = c(0.1,0.9)))
```

```
mean(rbinom(1000,1,0.5)) #n-obs=100, n=1, p=0.9
```

```
lorem = "Lorem ipsum dolor sit amet, consectetur adipiscing elit. Morbi posuere varius volutpat. Morbi :
```

#TO-DO

```
## [1] " Integer dapibus mi lectus, eu posuere arcu ultricies in"
```

```
## [2] " Aenean nulla ante, iaculis sed vehicula ac, finibus vel arcu"
```

```
## [3] "Cras suscipit id nibh lacinia elementum"
```

```
## [4] " Curabitur est augue, congue eget quam in, scelerisque semper magna"
```

```
## [5] " Mauris at sodales augue"
```

```
## [6] " Donec at tempor erat"
```

```
## [7] "Lorem ipsum dolor sit amet, consectetur adipiscing elit"
```

```
## [8] " Morbi posuere varius volutpat"
```

```
## [9] "Donec vehicula sagittis nisi non semper"
```

```
## [10] " Morbi faucibus ligula id massa ultricies viverra"
```

- #TO-DO

```
criminality_levels = c("none", "infraction", "misdemeanor", "felony")
#x_2f = factor(criminality_levels, levels = criminality_levels, ordered = TRUE)
#print(x_2f)
# x 2 = sample(x_2f, 100, replace = TRUE)
```

```
x_2 = factor(sample(rep(criminality_levels,100/4)), levels = criminality_levels, ordered = TRUE)
x_2
```

```
## [1] felony      infraction  misdemeanor misdemeanor infraction
## [6] infraction  infraction  infraction  felony      misdemeanor
## [11] misdemeanor misdemeanor none          felony      felony
## [16] none        infraction  felony      infraction  felony
## [21] felony      misdemeanor infraction  misdemeanor misdemeanor
## [26] none        misdemeanor infraction  infraction  infraction
## [31] none        none          misdemeanor none          misdemeanor
## [36] misdemeanor felony      infraction  felony      none
## [41] felony      infraction  felony      infraction  infraction
## [46] felony      misdemeanor none          felony      infraction
## [51] none        misdemeanor misdemeanor misdemeanor felony
## [56] misdemeanor felony      infraction  none          none
## [61] none        none          misdemeanor infraction  felony
## [66] misdemeanor infraction  none          misdemeanor none
## [71] none        none          misdemeanor misdemeanor felony
## [76] felony      infraction  felony      felony      infraction
## [81] infraction  felony      misdemeanor misdemeanor infraction
## [86] none        infraction  felony      none          none
## [91] none        none          felony      felony      felony
## [96] none        none          misdemeanor none          infraction
## Levels: none < infraction < misdemeanor < felony
```

- Convert this variable to binary where 0 is no crime and 1 is any crime. Answer in English: is this the proper binary threshold?

```
#x_2n = as.numeric(x_2)
x_2b = rep(0,100)
for(i in 1:100){
  if(as.numeric(x_2[i])-1 > 0){x_2b[i] = 1}
  else{x_2b[i] = 0}
}
x_2b
```

```
## [1] 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 0 1 1 1 1 1 1 1 1 0 1 1 1 1 0 0 1 0 1
## [36] 1 1 1 1 0 1 1 1 1 1 1 1 0 1 1 0 1 1 1 1 1 1 1 0 0 0 0 1 1 1 1 1 0 1 0
## [71] 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 0 0 0 0 1 1 1 0 0 1 0 1
```

```
# x_2b = as.numeric(x_2 > 0)
```

It is a proper threshold to say if someone has any infractions, but not a good threshold to determine if someone is a criminal.

- Convert this variable to an unordered, nominal factor variable.

```
x_2uno = factor(x_2,levels = levels(x_2), ordered = FALSE)
x_2uno
```

```
## [1] felony      infraction  misdemeanor misdemeanor infraction
## [6] infraction  infraction  infraction  felony      misdemeanor
## [11] misdemeanor misdemeanor none          felony      felony
## [16] none        infraction  felony      infraction  felony
## [21] felony      misdemeanor infraction  misdemeanor misdemeanor
## [26] none        misdemeanor infraction  infraction  infraction
## [31] none        none          misdemeanor none          misdemeanor
## [36] misdemeanor felony      infraction  felony      none
```

```
## [41] felony      infraction felony      infraction infraction
## [46] felony      misdemeanor none        felony      infraction
## [51] none        misdemeanor misdemeanor misdemeanor felony
## [56] misdemeanor felony      infraction none        none
## [61] none        none        misdemeanor infraction felony
## [66] misdemeanor infraction none        misdemeanor none
## [71] none        none        misdemeanor misdemeanor felony
## [76] felony      infraction felony      felony      infraction
## [81] infraction felony      misdemeanor misdemeanor infraction
## [86] none        infraction felony      none        none
## [91] none        none        felony      felony      felony
## [96] none        none        misdemeanor none        infraction
## Levels: none infraction misdemeanor felony
```

- Convert this variable into three binary variables without any information loss and put them into a data matrix.

```
x_2mtx = matrix(data = rep(0,100*3), nrow = 100, ncol = 3)
for(i in 1:100){
  j = as.numeric(x_2[i])-1
  if (j != 0){ x_2mtx[i,j] = 1}
}
x_2mtx
```

```
##      [,1] [,2] [,3]
## [1,]    0    0    1
## [2,]    1    0    0
## [3,]    0    1    0
## [4,]    0    1    0
## [5,]    1    0    0
## [6,]    1    0    0
## [7,]    1    0    0
## [8,]    1    0    0
## [9,]    0    0    1
## [10,]   0    1    0
## [11,]   0    1    0
## [12,]   0    1    0
## [13,]   0    0    0
## [14,]   0    0    1
## [15,]   0    0    1
## [16,]   0    0    0
## [17,]   1    0    0
## [18,]   0    0    1
## [19,]   1    0    0
## [20,]   0    0    1
## [21,]   0    0    1
## [22,]   0    1    0
## [23,]   1    0    0
## [24,]   0    1    0
## [25,]   0    1    0
## [26,]   0    0    0
## [27,]   0    1    0
## [28,]   1    0    0
## [29,]   1    0    0
## [30,]   1    0    0
```

##	[31,]	0	0	0
##	[32,]	0	0	0
##	[33,]	0	1	0
##	[34,]	0	0	0
##	[35,]	0	1	0
##	[36,]	0	1	0
##	[37,]	0	0	1
##	[38,]	1	0	0
##	[39,]	0	0	1
##	[40,]	0	0	0
##	[41,]	0	0	1
##	[42,]	1	0	0
##	[43,]	0	0	1
##	[44,]	1	0	0
##	[45,]	1	0	0
##	[46,]	0	0	1
##	[47,]	0	1	0
##	[48,]	0	0	0
##	[49,]	0	0	1
##	[50,]	1	0	0
##	[51,]	0	0	0
##	[52,]	0	1	0
##	[53,]	0	1	0
##	[54,]	0	1	0
##	[55,]	0	0	1
##	[56,]	0	1	0
##	[57,]	0	0	1
##	[58,]	1	0	0
##	[59,]	0	0	0
##	[60,]	0	0	0
##	[61,]	0	0	0
##	[62,]	0	0	0
##	[63,]	0	1	0
##	[64,]	1	0	0
##	[65,]	0	0	1
##	[66,]	0	1	0
##	[67,]	1	0	0
##	[68,]	0	0	0
##	[69,]	0	1	0
##	[70,]	0	0	0
##	[71,]	0	0	0
##	[72,]	0	0	0
##	[73,]	0	1	0
##	[74,]	0	1	0
##	[75,]	0	0	1
##	[76,]	0	0	1
##	[77,]	1	0	0
##	[78,]	0	0	1
##	[79,]	0	0	1
##	[80,]	1	0	0
##	[81,]	1	0	0
##	[82,]	0	0	1
##	[83,]	0	1	0
##	[84,]	0	1	0

```
## [85,] 1 0 0
## [86,] 0 0 0
## [87,] 1 0 0
## [88,] 0 0 1
## [89,] 0 0 0
## [90,] 0 0 0
## [91,] 0 0 0
## [92,] 0 0 0
## [93,] 0 0 1
## [94,] 0 0 1
## [95,] 0 0 1
## [96,] 0 0 0
## [97,] 0 0 0
## [98,] 0 1 0
## [99,] 0 0 0
## [100,] 1 0 0
```

#T0-D0

- What should the sum of each row be (in English)? Verify that.

```
# the sum of each row should be 0 or 1
# this should match writing the data as a binary crime or no crime, as before
rowSums(x_2mtx)
```

```
## [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 0 1 1 1 1 1 1 1 1 0 1 1 1 1 0 0 1 0 1
## [36] 1 1 1 1 0 1 1 1 1 1 1 1 1 0 1 1 0 1 1 1 1 1 1 0 0 0 0 1 1 1 1 1 0 1 0
## [71] 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 0 0 0 0 1 1 1 0 0 1 0 1
```

- How should the column sum look (in English)? Verify that.

```
# the column sums should be the frequency of each category
colSums(x_2mtx)
```

```
## [1] 25 25 25
```

- Generate a matrix with 100 rows where the first column is realization from a normal with mean 17 and variance 38, the second column is uniform between -10 and 10, the third column is poisson with mean 6, the fourth column in exponential with lambda of 9, the fifth column is binomial with $n = 20$ and $p = 0.12$ and the sixth column is a binary variable with 24% 1's.

```
n00 = 100

datafill = c(
  rnorm(n00, mean = 17, sd = sqrt(38)),
  runif(n00, min = -10, max = 10),
  rpois(n00, lambda = 6),
  rexp(n00, rate = 1/9),
  rbinom(n00, size = 20, prob = 0.12),
  sample(c(rep(1,round(n00*.24)),rep(0,round(n00*(1.00-.24))))),size = n00, replace = FALSE)
)

mtx_B = matrix(datafill, nrow = n00, ncol = n00, byrow = FALSE)
```