Tutorial3

Swarag T

2023-08-19

Sets, Counting, Samples

Sampling With Replacement

Let us start from the basics. Mostly we are going to measure in the context of a sample. Which in a way is a set. What is a set? a collection of well-defined objects or elements

We have familiarized yourself with the function sample. Let's start with that to understand probability. We can say that the outcomes of a die is a set, since it is a collection of well defined objects, which are numbers from 1 to 6. Now we want to understand the probability distribution for the die. Do you have any guess about how it would look like? Let's define a function to sample from it 1000 time.

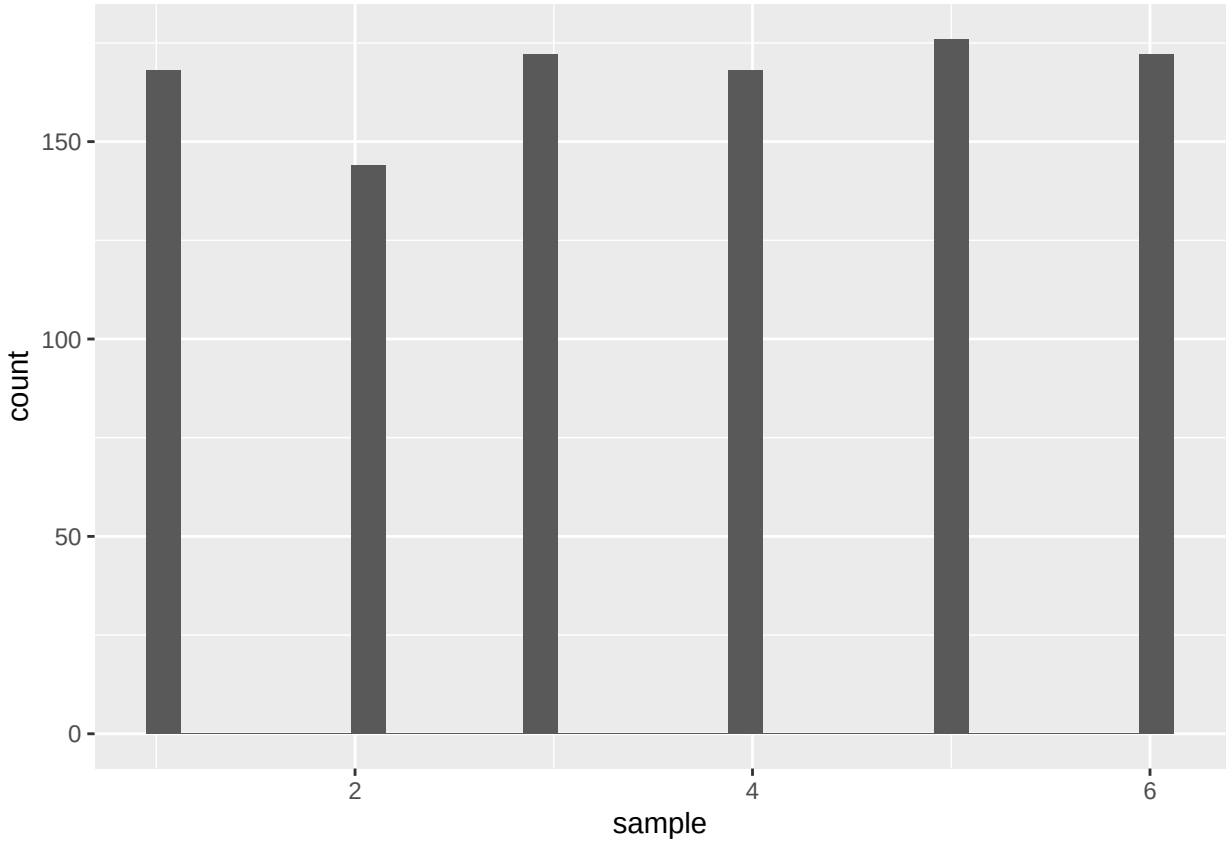
```
library(ggplot2)
library(tidyr)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
die <- c(1,2,3,4,5,6)

sampling_distribution <- sample(die, size = 1000, replace = TRUE)

df <- data.frame(sample = sampling_distribution)

# Plot the distribution
my_plot <- ggplot(df, aes(x=sample))
my_plot + geom_histogram(density = TRUE)
```

```
## Warning in geom_histogram(density = TRUE): Ignoring unknown parameters:
## `density`
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

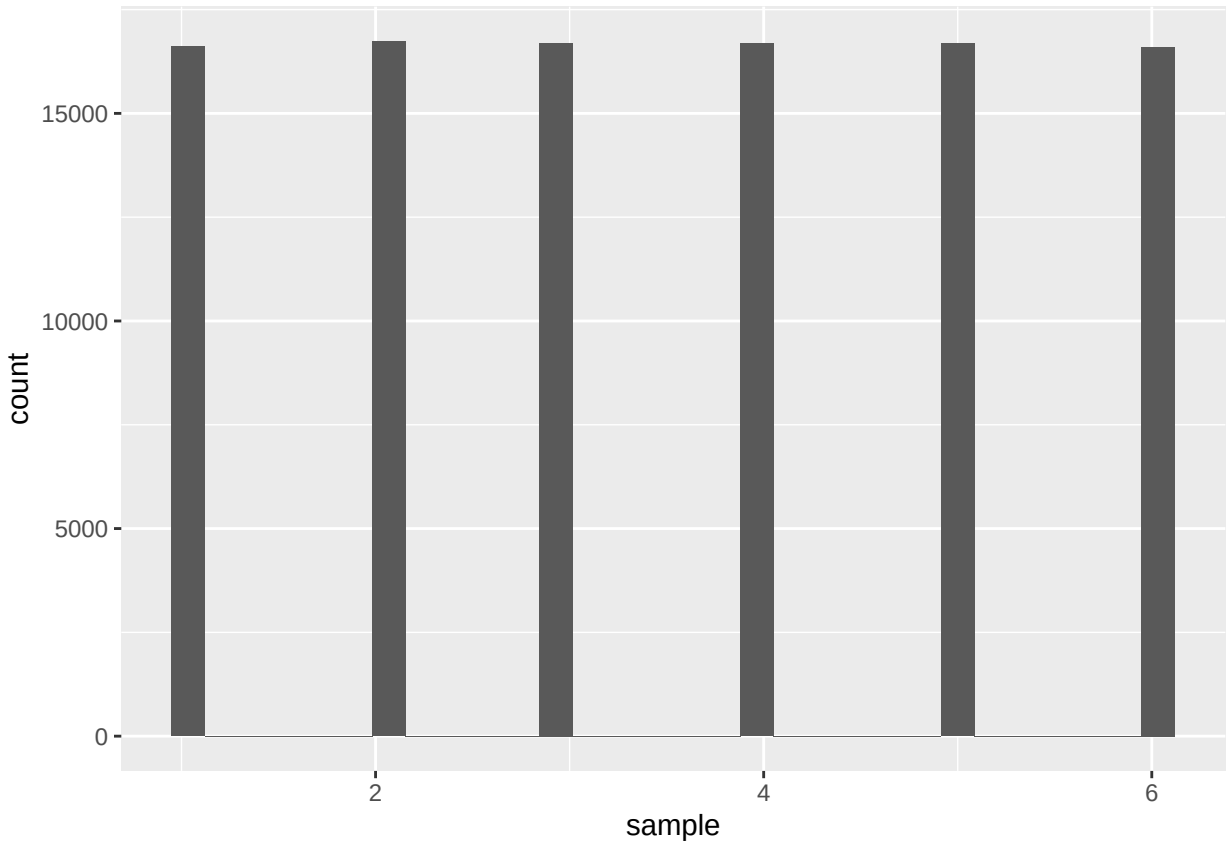


So, we can see that it is almost uniform. Let's further increase the numbers to see what happens!

```
library(ggplot2)
library(tidyr)
library(dplyr)
die <- c(1,2,3,4,5,6)

sampling_distribution <- sample(die, size = 100000, replace = TRUE)

df <- data.frame(sample = sampling_distribution)

# Plot the distribution
my_plot <- ggplot(df, aes(x=sample))
my_plot + geom_histogram(density = TRUE)
```

```
## Warning in geom_histogram(density = TRUE): Ignoring unknown parameters:
## `density`
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



It is almost the same for all numbers. What do you infer from this? Any guesses? What are these counts translating to?

Sampling Without Replacement

Permutations

So far we discussed how to sample with replacement. In the case of a die, we can't remove one face of the die after drawing an outcome. But, suppose we are considering a bag of objects, for simplicity, let it be the same numbers that we used in a die, then, is it possible to sample the same number again? No! There we uses sampling without replacement. It is here, the concept of permutations are becoming important. Let's consider bag containing numbers 1 to 3. In this case, the possible permutations(arrangements) we can have are: [1 2 3] [2 3 1] [3 1 2] [1 3 2] [3 2 1] [2 1 3]

$$^{n}P_{k} = \frac{n!}{(n-k)!}$$

This expression gives the total number of arrangements we can have, if we are choosing k objects from n objects. We can use combinat package in R for getting the permutations.

```
library(combinat)
```

```
##
## Attaching package: 'combinat'
```

```
## The following object is masked from 'package:utils':
##
##     combn
```

```
?permn
```

```
print ("Permutations of 3")
```

```
## [1] "Permutations of 3"
```

```
permn(3)
```

```
## [[1]]
## [1] 1 2 3
##
## [[2]]
## [1] 1 3 2
##
## [[3]]
## [1] 3 1 2
##
## [[4]]
## [1] 3 2 1
##
## [[5]]
## [1] 2 3 1
##
## [[6]]
## [1] 2 1 3
```

Combinations

In the case of permutations, the arrangement matters. But, what if we only care about the set we are getting and not the arrangement? Here, we would want to use combinations.

$$^{n}C_{k} = \frac{n!}{(k!)(n-k)!}$$

```
library(combinat)
?combn
```

```
## Help on topic 'combn' was found in the following packages:
##
##   Package               Library
##   utils                 /usr/lib/R/library
##   combinat              /home/swarag/R/x86_64-pc-linux-gnu-library/4.3
##
##
## Using the first match ...
```

```r
print ("Combinations of 3 from list of numbers from 1 to 3")
```

```
## [1] "Combinations of 3 from list of numbers from 1 to 3"
```

```r
combn(3,3)
```

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

```r
print ("Combinations of 3 from list of numbers from 1 to 10")
```

```
## [1] "Combinations of 3 from list of numbers from 1 to 10"
```

```r
combn(10,3)
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12] [,13] [,14]
## [1,]    1    1    1    1    1    1    1    1    1     1     1     1     1     1
## [2,]    2    2    2    2    2    2    2    2    3     3     3     3     3     3
## [3,]    3    4    5    6    7    8    9   10    4     5     6     7     8     9
##      [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23] [,24] [,25] [,26]
## [1,]     1     1     1     1     1     1     1     1     1     1     1     1
## [2,]     3     4     4     4     4     4     4     5     5     5     5     5
## [3,]    10     5     6     7     8     9    10     6     7     8     9    10
##      [,27] [,28] [,29] [,30] [,31] [,32] [,33] [,34] [,35] [,36] [,37] [,38]
## [1,]     1     1     1     1     1     1     1     1     1     1     2     2
## [2,]     6     6     6     6     7     7     7     8     8     9     3     3
## [3,]     7     8     9    10     8     9    10     9    10    10     4     5
##      [,39] [,40] [,41] [,42] [,43] [,44] [,45] [,46] [,47] [,48] [,49] [,50]
## [1,]     2     2     2     2     2     2     2     2     2     2     2     2
## [2,]     3     3     3     3     3     4     4     4     4     4     4     5
## [3,]     6     7     8     9    10     5     6     7     8     9    10     6
##      [,51] [,52] [,53] [,54] [,55] [,56] [,57] [,58] [,59] [,60] [,61] [,62]
## [1,]     2     2     2     2     2     2     2     2     2     2     2     2
## [2,]     5     5     5     5     6     6     6     6     7     7     7     8
## [3,]     7     8     9    10     7     8     9    10     8     9    10     9
##      [,63] [,64] [,65] [,66] [,67] [,68] [,69] [,70] [,71] [,72] [,73] [,74]
## [1,]     2     2     3     3     3     3     3     3     3     3     3     3
## [2,]     8     9     4     4     4     4     4     4     5     5     5     5
## [3,]    10    10     5     6     7     8     9    10     6     7     8     9
##      [,75] [,76] [,77] [,78] [,79] [,80] [,81] [,82] [,83] [,84] [,85] [,86]
## [1,]     3     3     3     3     3     3     3     3     3     3     3     4
## [2,]     5     6     6     6     6     7     7     7     8     8     9     5
## [3,]    10     7     8     9    10     8     9    10     9    10    10     6
##      [,87] [,88] [,89] [,90] [,91] [,92] [,93] [,94] [,95] [,96] [,97] [,98]
## [1,]     4     4     4     4     4     4     4     4     4     4     4     4
## [2,]     5     5     5     5     6     6     6     6     7     7     7     8
## [3,]     7     8     9    10     7     8     9    10     8     9    10     9
##      [,99] [,100] [,101] [,102] [,103] [,104] [,105] [,106] [,107] [,108]
## [1,]     4      4      5      5      5      5      5      5      5      5
## [2,]     8      9      6      6      6      6      7      7      7      8
## [3,]    10     10      7      8      9     10      8      9     10      9
##      [,109] [,110] [,111] [,112] [,113] [,114] [,115] [,116] [,117] [,118]
```

```
## [1,]      5      5      6      6      6      6      6      6      7      7
## [2,]      8      9      7      7      7      8      8      9      8      8
## [3,]     10     10      8      9     10      9     10     10      9     10
##      [,119] [,120]
## [1,]      7      8
## [2,]      9      9
## [3,]     10     10
```

Events

In probability theory, an event is a set of outcomes of an experiment to which a probability is assigned. Let's now see examples of an independent event and dependent event. ## Events And Probabilities ### Birthday Problem If there is a group of n people, what is the probability that at least two of them would share the same birthday? Write a function and plot the probability against n. Answer: Suppose you have n people. The event of having at least two people having a common birthday is complementary to the event of none of them having a common birthday. Total number of such permutations or arrangements are

$$^{365}P_n = \frac{365!}{(365-n)!}$$

And the total number of possible arrangements are

$$365^n$$

. Then the probability of not getting same birtday for any of them is that:

$$P(\bar{B}) = \frac{\frac{365!}{(365-n)!}}{365^n}$$

. Just negate this from the sure event, which is the set f all possible outcomes, which has a probability =1, we will get the probability of at least two of them sharing the same birthday.

```
same_bday <- function(n){

  permutations <- factorial(365)/factorial(365-n)
  print(permutations)
  total_arrangements <- 365^n
  print(total_arrangements)
  p_b_bar <- permutations/total_arrangements
  return(1-p_b_bar)
}
same_bday(23)
```

```
## [1] NaN
## [1] 8.565168e+58
```

```
## [1] NaN
```

Doesn't work, right? How will you solve? What is the problem we are encountering?

```r
same_bday2 <- function(n){

  permutations <- lfactorial(365)-lfactorial(365-n)
  print(permutations)
  total_arrangements <- log(365^n)
  print(total_arrangements)
  p_b_bar <- exp(permutations - total_arrangements)
  return(1-p_b_bar)
}
same_bday2(23)
```

```
## [1] 134.9898
## [1] 135.6976
```

```
## [1] 0.5072972
```

Now, let's consider a weather data, that of Kanpur, for the year 2022

```r
data <- read.csv('kanpur-weather.csv')
data[1:10,]
```

```
##      name   datetime tempmax tempmin temp feelslikemax feelslikemin feelslike
## 1  kanpur 2022-01-01    20.3    11.0 14.4         20.3         11.0      14.4
## 2  kanpur 2022-01-02    20.5     7.7 13.4         20.5          7.7      13.4
## 3  kanpur 2022-01-03    21.0     6.7 13.2         21.0          6.7      13.2
## 4  kanpur 2022-01-04    22.9     7.7 14.6         22.9          7.7      14.6
## 5  kanpur 2022-01-05    17.1     9.8 13.8         17.1          9.8      13.8
## 6  kanpur 2022-01-06    18.5    13.0 14.3         18.5         13.0      14.3
## 7  kanpur 2022-01-07    18.9    12.6 15.8         18.9         12.6      15.8
## 8  kanpur 2022-01-08    19.6    15.0 16.9         19.6         15.0      16.9
## 9  kanpur 2022-01-09    21.4    14.8 16.5         21.4         14.8      16.5
## 10 kanpur 2022-01-10    21.4    14.0 17.3         21.4         14.0      17.3
##     dew humidity precip precipprob precipcover preciptype snow snowdepth
## 1  10.6     80.5    0.0          0        0.00         NA   NA        NA
## 2  10.4     83.5    0.0          0        0.00         NA   NA        NA
## 3  10.0     83.1    0.0          0        0.00         NA   NA        NA
## 4  11.4     82.7    0.0          0        0.00         NA   NA        NA
## 5  12.7     92.6    0.0          0        0.00         NA   NA        NA
## 6  13.2     93.2   10.8        100       20.83       rain   NA        NA
## 7  14.6     92.8    3.0        100        4.17       rain   NA        NA
## 8  15.7     92.8    3.2        100       12.50       rain   NA        NA
## 9  15.2     92.1    0.8        100        4.17       rain   NA        NA
## 10 15.7     90.3    0.0          0        0.00          0    0
##    windgust windspeed winddir sealevelpressure cloudcover visibility
## 1        NA       9.4   281.0           1021.6       38.3        1.5
## 2        NA       9.4   270.5           1019.8       47.3        1.8
## 3        NA       7.6   283.2           1018.1        8.0        1.3
## 4        NA       8.6   240.2           1017.4        2.6        1.0
## 5        NA       7.6    94.2           1017.3       75.9        0.7
## 6        NA      16.9    79.4           1016.9       92.2        1.2
## 7        NA      10.8    81.9           1017.9       90.6        1.5
## 8        NA      18.4   111.0           1016.7       93.5        2.1
```

```
## 9        NA      18.4    96.3           1014.8       85.4        2.1
## 10      9.4       9.4   283.5           1016.2       74.9        2.7
##    solarradiation solarenergy uvindex severerisk             sunrise
## 1           169.9        14.6       6         NA 2022-01-01T06:56:48
## 2           159.3        13.8       6         NA 2022-01-02T06:57:04
## 3           151.3        13.0       6         NA 2022-01-03T06:57:19
## 4           164.2        14.3       6         NA 2022-01-04T06:57:32
## 5           118.0        10.4       6         NA 2022-01-05T06:57:44
## 6            80.3         7.1       4         NA 2022-01-06T06:57:54
## 7            61.1         5.1       3         NA 2022-01-07T06:58:03
## 8            68.7         6.1       3         NA 2022-01-08T06:58:11
## 9            95.2         8.2       4         NA 2022-01-09T06:58:17
## 10          184.1        16.0       7         10 2022-01-10T06:58:21
##                   sunset moonphase            conditions
## 1  2022-01-01T17:27:37      0.95       Partially cloudy
## 2  2022-01-02T17:28:17      0.98       Partially cloudy
## 3  2022-01-03T17:28:59      0.00                 Clear
## 4  2022-01-04T17:29:41      0.05                 Clear
## 5  2022-01-05T17:30:23      0.08       Partially cloudy
## 6  2022-01-06T17:31:06      0.11         Rain, Overcast
## 7  2022-01-07T17:31:50      0.15         Rain, Overcast
## 8  2022-01-08T17:32:34      0.18         Rain, Overcast
## 9  2022-01-09T17:33:19      0.25 Rain, Partially cloudy
## 10 2022-01-10T17:34:04      0.25       Partially cloudy
##                                                                    description
## 1                                          Partly cloudy throughout the day.
## 2                                          Partly cloudy throughout the day.
## 3                                        Clear conditions throughout the day.
## 4                                        Clear conditions throughout the day.
## 5                                          Partly cloudy throughout the day.
## 6  Cloudy skies throughout the day with a chance of rain throughout the day.
## 7                   Cloudy skies throughout the day with early morning rain.
## 8                                Cloudy skies throughout the day with rain.
## 9                     Partly cloudy throughout the day with morning rain.
## 10                                         Partly cloudy throughout the day.
##                icon                                          stations
## 1  partly-cloudy-day 42367099999,42369099999,VILK,remote,42469099999
## 2  partly-cloudy-day        42367099999,42369099999,VILK,42469099999
## 3          clear-day 42367099999,42369099999,VILK,remote,42469099999
## 4          clear-day 42367099999,42369099999,VILK,remote,42469099999
## 5                fog        42367099999,42369099999,VILK,42469099999
## 6               rain 42367099999,42369099999,VILK,remote,42469099999
## 7               rain 42367099999,42369099999,VILK,remote,42469099999
## 8               rain        42367099999,42369099999,VILK,42469099999
## 9               rain 42367099999,42369099999,VILK,remote,42469099999
## 10 partly-cloudy-day 42367099999,42369099999,VILK,remote,42469099999
```

```
summary(data)
```

```
##      name              datetime            tempmax         tempmin
##  Length:365         Length:365         Min.   :13.20   Min.   : 4.00
##  Class :character   Class :character   1st Qu.:27.00   1st Qu.:13.00
##  Mode  :character   Mode  :character   Median :32.30   Median :22.50
##                                        Mean   :31.58   Mean   :20.61
```

```
##                                     3rd Qu.:37.00   3rd Qu.:27.00
##                                     Max.   :45.30   Max.   :31.00
##
##       temp        feelslikemax    feelslikemin     feelslike
##  Min.   : 8.40   Min.   :13.20   Min.   : 4.00   Min.   : 8.20
##  1st Qu.:19.30   1st Qu.:26.80   1st Qu.:13.00   1st Qu.:19.30
##  Median :27.90   Median :38.80   Median :22.50   Median :29.60
##  Mean   :25.57   Mean   :36.99   Mean   :21.85   Mean   :28.78
##  3rd Qu.:31.10   3rd Qu.:46.20   3rd Qu.:29.60   3rd Qu.:37.20
##  Max.   :36.60   Max.   :55.00   Max.   :39.70   Max.   :47.60
##
##       dew           humidity         precip          precipprob
##  Min.   : 6.50   Min.   :29.20   Min.   :  0.000   Min.   :  0.00
##  1st Qu.:13.20   1st Qu.:64.70   1st Qu.:  0.000   1st Qu.:  0.00
##  Median :20.60   Median :76.00   Median :  0.000   Median :  0.00
##  Mean   :19.34   Mean   :73.01   Mean   :  2.575   Mean   : 18.63
##  3rd Qu.:25.70   3rd Qu.:84.60   3rd Qu.:  0.000   3rd Qu.:  0.00
##  Max.   :28.20   Max.   :96.30   Max.   :122.000   Max.   :100.00
##
##   precipcover      preciptype             snow     snowdepth    windgust
##  Min.   : 0.000   Length:365         Min.   :0   Min.   :0   Min.   : 4.30
##  1st Qu.: 0.000   Class :character   1st Qu.:0   1st Qu.:0   1st Qu.:15.40
##  Median : 0.000   Mode  :character   Median :0   Median :0   Median :23.95
##  Mean   : 1.347                      Mean   :0   Mean   :0   Mean   :25.11
##  3rd Qu.: 0.000                      3rd Qu.:0   3rd Qu.:0   3rd Qu.:32.40
##  Max.   :25.000                      Max.   :0   Max.   :0   Max.   :59.40
##                                      NA's   :9   NA's   :9   NA's   :9
##    windspeed        winddir      sealevelpressure   cloudcover
##  Min.   : 0.00   Min.   :  2.3   Min.   : 995.8   Min.   :  0.00
##  1st Qu.: 9.40   1st Qu.:100.7   1st Qu.:1001.5   1st Qu.:  2.50
##  Median :14.80   Median :273.9   Median :1006.0   Median : 28.90
##  Mean   :14.49   Mean   :212.4   Mean   :1007.2   Mean   : 37.14
##  3rd Qu.:18.40   3rd Qu.:300.3   3rd Qu.:1013.5   3rd Qu.: 69.30
##  Max.   :38.20   Max.   :356.5   Max.   :1021.6   Max.   :100.00
##
##    visibility     solarradiation   solarenergy       uvindex
##  Min.   : 0.500   Min.   : 34.8   Min.   : 3.20   Min.   : 1.000
##  1st Qu.: 3.000   1st Qu.:177.1   1st Qu.:15.20   1st Qu.: 7.000
##  Median : 4.000   Median :212.8   Median :18.20   Median : 8.000
##  Mean   : 4.015   Mean   :222.4   Mean   :19.17   Mean   : 8.079
##  3rd Qu.: 4.600   3rd Qu.:281.0   3rd Qu.:24.20   3rd Qu.: 9.000
##  Max.   :15.500   Max.   :336.4   Max.   :29.00   Max.   :10.000
##
##    severerisk     sunrise             sunset             moonphase
##  Min.   :10.0   Length:365         Length:365         Min.   :0.0000
##  1st Qu.:10.0   Class :character   Class :character   1st Qu.:0.2300
##  Median :10.0   Mode  :character   Mode  :character   Median :0.4700
##  Mean   :14.3                                         Mean   :0.4772
##  3rd Qu.:10.0                                         3rd Qu.:0.7500
##  Max.   :60.0                                         Max.   :0.9800
##  NA's   :9
##   conditions        description            icon             stations
##  Length:365         Length:365         Length:365         Length:365
##  Class :character   Class :character   Class :character   Class :character
```

```
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
```

What is the probability of getting an above median temperature in Kanpur? From this what you can infer about the data? What about precipitation? ### Conditional Probability

Now, some people had argued that the temperature that we would feel, is a function of both the ambient temperature and the humidity level. We define three events 1) Maximum temperature raising above the median of maximum temperature a year(32.30) 2) Humidity raises above the median value(76.00) 3) Maximum perceived temperature raises above the median value(38.80) Determine whether event (3) is independent of event (1) and (2)? Use columns "tempmax", "humidity" and, "feelslikemax".

```
data$tempmax <- 1*(data$tempmax>32.30)
data$humidity <- 1*(data$humidity>76.00)
data$feelslikemax <- 1*(data$feelslikemax>38.80)
```

```
check_independence <- function(data, event1, event2){
  data$intersection <- data[[event1]]*data[[event2]]
  p_intersection <- mean(data$intersection)
  p_event1 <- mean(data[[event1]])
  p_event2 <- mean(data[[event2]])
  p_multiple <- p_event1*p_event2
  p_multiple == p_intersection
}
print("ambient temperature and the temperature we feel are independent?")
```

```
## [1] "ambient temperature and the temperature we feel are independent?"
```

```
check_independence(data,"tempmax","feelslikemax")
```

```
## [1] FALSE
```

```
print("humidity and the temperature we feel is independent?")
```

```
## [1] "humidity and the temperature we feel is independent?"
```

```
check_independence(data,"humidity","feelslikemax")
```

```
## [1] FALSE
```

But, we still would like to know the conditional probability, the probability that event2 would happen, given event1 had happened. Why?

```
conditional_prob <- function(data, event1, event2){
  data$intersection <- data[[event1]]*data[[event2]]
  p_intersection <- mean(data$intersection)
  p_event1 <- mean(data[[event1]])
  p_intersection/p_event1
}
print("probability of experiencing an above average temperature given that the ambient temperature is al
```

```
## [1] "probability of experiencing an above average temperature given that the ambient temperature is a
```

```
conditional_prob(data,"tempmax","feelslikemax")
```

```
## [1] 0.9
```

```
print("probability of experiencing an above average temperature given that the humidity is above median"
```

```
## [1] "probability of experiencing an above average temperature given that the humidity is above media
```

```
conditional_prob(data,"humidity","feelslikemax")
```

```
## [1] 0.4364641
```

But, are we missing somthing here? Why the conditional probability for humidity is very low?

Humidity is not a primary factor. Let's see the case when the temperature is already high

```
data_high_temp <- subset(data, tempmax == 1)
print("probability of experiencing an above average temperature given that the humidity is above median"
```

```
## [1] "probability of experiencing an above average temperature given that the humidity is above media
```

```
conditional_prob(data_high_temp,"humidity","feelslikemax")
```

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

But, when we are selecting only those data points were the ambient temperature is above median, event of experiencing an above average temperature becomes a sure event. Then this conditional probability may not tell us anything about the humidity.

Question: Sun radiates energy in the form of solar radiation, check the conditional probability of getting an above median temperature when the solar radiation is above median? But, clouds would reflect it, then, how would you find the same if cloud cover was given?

```
data$solarradiation <- 1*(data$solarradiation>212.8)
data$cloudcover <- 1*(data$cloudcover>28.90)
print("probability of above average temperature given that the solar radiation is above median?")
```

```
## [1] "probability of above average temperature given that the solar radiation is above median?"
```

```
conditional_prob(data,"solarradiation","tempmax")
```

```
## [1] 0.8241758
```

```
print("probability of above average temperature given that the cloud cover is above median?")
```

```
## [1] "probability of above average temperature given that the cloud cover is above median?"
```

```r
conditional_prob(data,"cloudcover","tempmax")
```

```
## [1] 0.5934066
```

Questions

1. Find the probability of having above median temperature for each month, plot it(bar graph). What do you observe?
2. I have a biased coin, which has a probability of 0.8 for getting a head. What is the number of tosses required to get the probability of getting at least one head from these tosses to be >=0.9? Define a function that would compute the probability of getting at least one head from n tosses(n is the input) and plot this probability against n. Make the function more generic, where it would take the probability of getting head in a single toss also and add this probability as a legend in the graph.