Ve406 Lecture 2

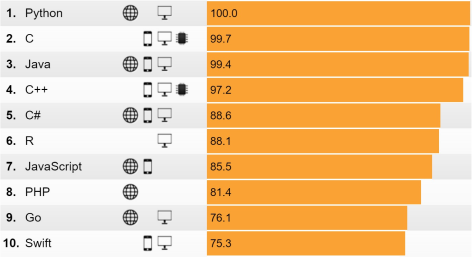
Jing Liu

UM-SJTU Joint Institute

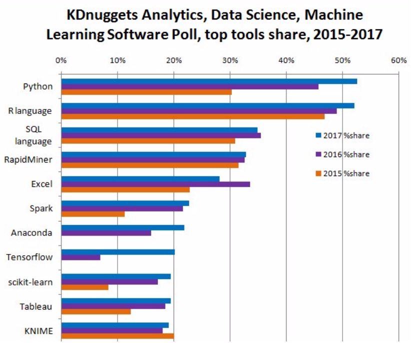
May 16, 2018

- R is a statistical programming language used by many when comes to data.

2017 IEEE top programming languages



1.	Python	🌐 🖥	100.0
2.	C	📱 🖥 ▦	99.7
3.	Java	🌐 📱 🖥	99.4
4.	C++	📱 🖥 ▦	97.2
5.	C#	🌐 📱 🖥	88.6
6.	R	🖥	88.1
7.	JavaScript	🌐 📱	85.5
8.	PHP	🌐	81.4
9.	Go	🌐 🖥	76.1
10.	Swift	📱 🖥	75.3



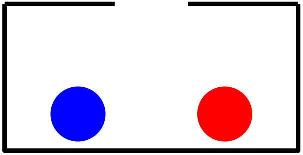
- We will be using R to illustrate a few basics in statistics.
- R can be downloaded from HERE
- RStudio is an IDE for R, and can be downloaded from HERE

Q: What does it mean when we say a sequence of random variables

$$\{X_1, X_2, \ldots, X_n\}$$

is independent and identically distributed (i.i.d.)?

- Suppose there is a box that has one red ball and one blue ball,



  and I draw them out successively. Let X_1 denote whether the first ball drawn is blue or red, and X_2 denote whether the second ball drawn is blue or red.

Q: Does $\{X_1, X_2\}$ satisfy the i.i.d. condition?

```
> xvec = c("Red", "Blue")
> # draw without replacement
> tmp = sample(xvec, 2, replace = FALSE)
> tmp
```

```
[1] "Blue" "Red"
```

```
> x.df = data.frame(X1 = integer(), X2 = integer())
> n = 1e3 # Number of repetition
> for (i in 1: n){
+   x.df[i, ] =                      # ith row
+     sample(
+       c(0,1), 2, replace = TRUE)  # red(0), blue(1)
+ }
>
> # fX2( x2 = blue )
> prob.X2.blue = sum(x.df[,2]) / n
> prob.X2.blue
```

```
[1] 0.515
```

- The last version of this simulation is very readable, but is horribly inefficient

```
> # A better version
> n = 1e8
>
> X1 = rbinom(n, size = 1, prob = 1/2)
> # binomial random variable
> X2 = 1 - X1
> # R converts 1 into a vector automatically
> prob.X2.blue = sum(X2) / n
>
> prob.X2.blue
```

```
[1] 0.5000055
```

Q: What does this result suggest? Do you think it is identically distributed?

- Of course, a sequence can be independent but not identically distributed

- You should treat the data you have on the response,

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

is just one realisation of the sequence

$$\{Y_1, Y_2, \cdots, Y_n\}$$

Q: Now recall the response in a typical dataset,

$$\{Y_1, Y_2, \cdots, Y_n\}$$

that you have encounted, does it satisfy the i.i.d. condition?

- Until the second part of this course, we will only consider independent Y_i.

Q: Men in the class: would you like to have daughters?

- It was said that some men are more likely to have daughters than others e.g.

  a deep-sea diver, a fighter pilot and a heavy smoker

- If you prefer sons, easy! just become US president.

The Facts

- The 45 US presidents from George Washington to Donald Trump have had a total of 158 children, comprising 91 sons and only 67 daughters.

  about 1.4 sons for very daughter

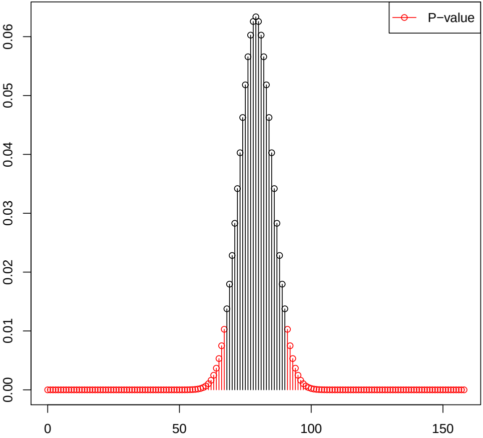
- Two studies of deep-sea divers revealed that the men had a total of 190 children, comprising 65 sons and 125 daughters:

  about 1.9 daughters for every son

- Let consider testing the two hypotheses one at time.

```
> # President -----------------------------
> rm(list = ls()) # Clean environment
> # Suppose the chance is 50-50
> # for a president as well
> p = 1/2
>
> # Total number of trials
> n = 158
>
> # All possible values 0 to 158 daughters
> x = 0:n
>
> # Binomial density/mass function
> fX = dbinom(x, size = n, prob = p)
> # Produce a plot of the density function
> # With the p-value region highlighted in red
```

- Recall P-value is the probability of getting a result at least as extreme as 65 daughters under $H_0 : p = 0.5$.



```
> p.lowerTail = pbinom(67, size = n, prob = p)
> 2*p.lowerTail                # P-value
```

```
[1] 0.06694264
```

```
> # Lower tail
> tmpL = fX[x <= 67] # As extreme as 67

> # Upper tail probability
> x.upperTail = 1 + qbinom(
+   p.lowerTail, size = n, prob = p,
+   lower.tail = FALSE)

> tmpU = fX[x >= x.upperTail]

> # Middle
> M = x[ x>67 & x<x.upperTail]
> tmpM = fX[ x>67 & x<x.upperTail ]
> # Actual plotting
> plot(x, fX, type = "n", xlab = "", ylab = "")

> lines(0:67, tmpL, type = "h", col = "red")

> points(0:67, tmpL, col = "red")

> lines(x.upperTail:n, tmpU,
+       type = "h", col = "red")
> points(x.upperTail:n, tmpU, col = "red")

> lines(M,tmpM, type = "h")
> points(M, tmpM)

> legend("topright", legend = "P-value",
+        col ="red", lty = 1, pch = 1)
```

- If we do the same to

```
> # Deep-sea divers -----------------------
> 2 * pbinom(65, size = 190, prob = 0.5)
```

```
[1] 1.603136e-05
```

- This is a little more than one chance in 100 thousand.
- The following is what we can say:

> We conclude that it is extremely unlikely that this observation could have occurred by chance, if the deep-sea divers had equal probabilities of having sons and daughters. The data are not compatible with H_0.

Q: How the president case?

```
> 2*p.lowerTail
```

```
[1] 0.06694264
```

- For the president case, the following is what we can say:

> We conclude that there is no real evidence that presidents are more likely to have sons than daughters or vice versa. The observations are compatible with the possibility that there is no difference.

Q: Does this mean presidents are equally like to have sons and daughters?

Q: Back to the deep-sea divers case, does p-value of 1.603136e-05 say about the actual probability of a diver having a daughter instead of a son?

- Let p denote the probability of a deep-sea diver has a daughter, and X be the number of daughters out of 190 children that deep-sea divers have

$$X \sim \text{Binomial}(190, p)$$

Q: Which single value would you say p is equal to?

Definition

The process of using data to suggest a value for a parameter is called estimation.

- The value suggested is called the estimate of the parameter. It is clear that the estimate should depend on the data.
- An estimator is a function of the data, that gives estimates of the parameter

$$\hat{\theta} = h(X)$$

- Notice $\hat{\theta}$ is also random, its randomness is inherited from the data.
- The distribution for $\hat{\theta}$ is called the sampling distribution of the estimator.

- Consider the following

```
> # Function ------------------------------
> myp_func = function(p){
+   n = 190
+   p125 = pbinom(125, size = n, prob = p)
+   p124 = pbinom(124, size = n, prob = p)
+   p125 - p124
+ }
```

```
> myp_func(p = 0.5)
```

```
[1] 3.972689e-06
```

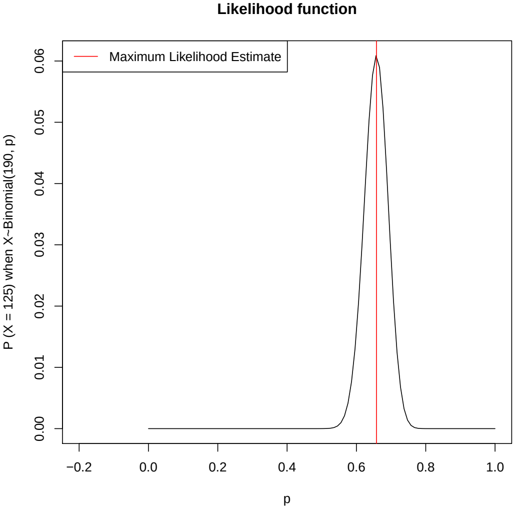
```
> myp_func(p = 0.6)
```

```
[1] 0.01576121
```

- Having $p = 0.6$ still looks unlikely, but it is almost 4000 times more likely.

- It is not hard to understand the best choice of p in this case is MLE.

Q: What is MLE?



```
> # many functions in R can take vector input
> pbinom(125, size = n, prob = c(0.5,0.6))
```

```
[1] 0.9999960 0.9567501
```

- Thus the tidy function can actually take vector input as well

```
> myp_func = function(p){
+   n = 190
+   p125 = pbinom(125, size = n, prob = p)
+   p124 = pbinom(124, size = n, prob = p)
+   p125 - p124
+ }
```

-
  ```
  > # Create a vector contains a sequence of numbers
  > pvec = seq(0, 1, length.out = 100)
  >
  > lvec = myp_func(pvec)
  ```

```
> plot(pvec, lvec,
+      type = "l",
+      xlab = "p",
+      ylab = "P(X = 125) when X~Binomial(190, p)",
+      main = "Likelihood function",
+      xlim = c(-0.2,1)
+      )

> phat = 125/190 # common-sense

> abline(v = phat, col = "red") # MLE
>
> legend("topleft",
+        legend = "Maximum Likelihood Estimate",
+        lty = 1, col = 2)
```

Q: Is there any way to attach some kind of “strength” to out estimate?

$$\hat{p} = \frac{125}{190}$$

- Because there clearly is a difference between the following scenarios
  - 125 daughters out of 190 children
  - 1250 daughters out of 1900 children

  despite of having the same MLE $\hat{p} = \dfrac{25}{28}$.
- I have been hiding things from you.

```
> # confidence interval: Clopper-Pearson
> binom.test(125, n = 190, p = 0.5)
```

```
        Exact binomial test

data:  125 and 190
number of successes = 125, number of trials = 190, p-value = 1.603e-05
alternative hypothesis: true probability of success is not equal to 0.5
95 percent confidence interval:
 0.5857408 0.7250337
sample estimates:
probability of success
             0.6578947
```

- When people say

  > We are 95% confident that the true probability of having a daughter is between $(0.586, 0.725)$ for a male deep-sea diver.

  they actually meant

  > The method used to obtain the interval
  >
  > $$(0.586, 0.725)$$
  >
  > will contain the true probability 95% of time if it is to be repeated.

```
> install.packages("binom")
> binom::binom.confint(125, n = 190)
```

```
          method   x   n      mean     lower     upper
1  agresti-coull 125 190 0.6578947 0.5878351 0.7216963
2     asymptotic 125 190 0.6578947 0.5904374 0.7253521
3          bayes 125 190 0.6570681 0.5895875 0.7236296
4        cloglog 125 190 0.6578947 0.5857332 0.7205325
5          exact 125 190 0.6578947 0.5857408 0.7250337
6          logit 125 190 0.6578947 0.5876377 0.7218476
7         probit 125 190 0.6578947 0.5882529 0.7225372
8        profile 125 190 0.6578947 0.5886447 0.7229128
9            lrt 125 190 0.6578947 0.5886470 0.7229437
10     prop.test 125 190 0.6578947 0.5852078 0.7240821
11        wilson 125 190 0.6578947 0.5879068 0.7216245
```