Introduction to R

Session 04: Simulations, probability and statistics review

Álvaro Pérez[1]

Instituto Tecnológico Autónomo de México

Fall 2024

[1] Based on PhD Romero Londoño's notes.

Simulation

- Why do this? Why not just use real data?
- Because with real data, **we don't know what the right answer is**
- So if we do some method, and it gives us an answer, how do we know if the answer is right?
- Simulation lets us know the right answer
- And if the method works (at least in our fake scenario), we can apply it to some real data

Uncovering the truth

- When it comes down to it, **what is the purpose of data analysis?**
- When we work with data, we have this idea that there exists a **true model**
- The **true model** is the way the world actually works!
- But we don't know what that true model is

The purpose of data analysis

- So that's where the data comes in
- The true model **generated the data** (the 'data generating process' or DGP)
- By looking at the data we're trying to work backwards to figure out what is the 'data generating process'
- With simulation, **we know** what generated the data and what the true model is. Thus we can check how close we get with our data analysis

Example

- Let's generate 500 coin flips
- **True model**: generate heads with probability 1/2 and tails with probability 1/2

```r
coins <- sample(c("Heads","Tails"),500,replace=T)
```

Example

- Now let's take that data as given and analyze it in our standard way
- The proportion of heads is ‘mean(coins=='Heads')‘ ($\approx$0.496)
- And we can look at the distribution, as we would:

```r
mean(coins=='Heads')
barplot(prop.table(table(coins)))

#THE GGPLOT2 WAY
#ggplot(as.data.frame(coins),aes(x=coins))+geom_bar()
```

Example

- So what's our conclusion?
- We would "estimate" that the **true model** generates heads $\approx$0.496 of the time
- $\frac{1}{2}$ is correct, so pretty close. But not exact.
- What if it **always** errs on the same side? Then it's not a good method at all!

Simulation in a loop

- We can go a step further by doing this simulation **over and over again** in a loop!
- This will let us tell whether our method gets it right on average
- And, when it's wrong, how wrong it is!

Simulation in a loop

```
#A blank vector to hold our results
propHeads <- c()
#Let's run this simulation 2000 times
for (i in 1:2000) {
  #Re-create data using the true model
  coinsdraw <- sample(c("Heads","Tails"),500,replace=T)
  #Re-perform our analysis
  result <- mean(coinsdraw=="Heads")
  #And store the result
  propHeads[i] <- result
}
#Let's see what we get on average
stargazer(as.data.frame(propHeads),type='text')
#And let's look at the distribution of our findings
plot(density(propHeads),xlab='Proportion Heads',
main='Mean of 501 Coin Flips over 2000 Samples')
abline(v=mean(propHeads),col='red')
```

Real world

- Imagine we **didn't** know the answer was $\frac{1}{2}$
- We wan to know what proportion of the time will a coin land heads
- Collect data on coin flips
- Perform our analysis method - take proportion of heads, and get $\approx$0.496
- Conclude that the **true model** produces heads $\approx$0.496 of the time
- Statistical inference is all about formalizing this process

Random variables

- Probability/statistics allows us to analyze chance events in a logically way
- The probability of an event is a number indicating how likely that event will occur
- Probability is always between 0 (never happens) and 1 (always happens)
- Random variable assigns numbers to different outcomes (each with a probability)
- Coin toss. It's random. Each face has $\frac{1}{2}$ probability
- By assigning 1 to tail and 0 to head we created a random variable

Some clarifications

- Goal: Estimate unknown parameters

- To approximate parameters, we use an estimator, which is a function of the data

Important notation

- Greek letters (e.g., μ) are the truth (i.e., parameters of the true DGP)
- Greek letters with hats (e.g., $\hat{\mu}$) are estimates (i.e., what we *think* the truth is)
- Non-Greek letters (e.g., X) denote sample/data
- Non-Greek letters with lines on top (e.g., $\overline{X}$) denote calculations from the data (e.g., $\overline{X} = \frac{1}{N}\sum_i X_i$).
- We want to estimate the truth, with some calculation from the data ($\hat{\mu} = \overline{X}$)
- Data $\longrightarrow$ Calculations $\longrightarrow$ Estimate $\underbrace{\longrightarrow}_{\text{Hopefully}}$ Truth
- Example: X $\longrightarrow \overline{X} \longrightarrow \hat{\mu} \underbrace{\longrightarrow}_{\text{Hopefully}} \mu$

Notation example with a coin toss

- μ denotes the true probability a coin lands head ($\frac{1}{2}$ if the coin is fair)
- $\hat{\mu}$ is our estimator of the probability a coin lands head
- X is the data we gather from tossing a coin 500 times
- $\overline{X}$ is the proportion of times the coin lands head
- Data from coin tosses $\longrightarrow$ Calculate proportion of heads $\longrightarrow$ Estimator for the probability of heads $\underbrace{\longrightarrow}_{\text{Hopefully}}$ True probability
- $\mathrm{X} \longrightarrow \overline{X} \longrightarrow \hat{\mu} \underbrace{\longrightarrow}_{\text{Hopefully}} \mu$

Discreet random variables

- Takes only a discreet set of values
- Probability distribution ($P(X = x) = f(x)$): probability event x happens
- $f(x) \in [0, 1]$
- Cumulative probability distribution ($P(X \leq x) = F(x)$: probability random variable is less than or equal to x

Continuous random variables

- Takes a continuum of values
- Probability density function ($f(x)$): **not** the probability x happens
  - zero since there are infinity many possible values
  - $P(a < x < b) = \int_a^b f(x)dx$
  - $f(x)$ helps us recover the probability that a random variable is in an interval
- $f(x) \in [0, 1]$
- Cumulative probability distribution ($P(X \leq x) = F(x) = \int_{-\infty}^{x} f(x)dx$: probability random variable is less than or equal to x

Summarizing a distribution

- What are we actually doing when we do something like take a mean or a median?
- We're trying to say something about the **distribution** of that variable
- Distribution: **how often** values occur when you randomly sample over and over
  - **Distribution** of a coin toss: half the times you get "head" (other half get "tail")
  - **Distribution** of the minutes in the day: it's equally likely to be any minute
  - **Distribution** of height looks like a bell-curve shape
  - **Distribution** of income/wealth: Most people near the bottom; very few at the top

Summarizing a distribution

- Expectation attempts to capture the "mean" of the random variable
- Variance quantifies the spread of the random variable
- For a discreet random variable
  - $\mathbb{E}[X] := \sum_x f(x)x$
  - $V(X) := \mathbb{E}[(X - \mathbb{E}[X])^2] = \sum_x f(x)\,(x - \mathbb{E}[X])^2$
- For a continuous random variable
  - $\mathbb{E}[X] := \int_{-\infty}^{\infty} f(x)xdx$
  - $V(X) := \mathbb{E}[(X - \mathbb{E}[X])^2] = \int_{-\infty}^{\infty} f(x)\,(x - \mathbb{E}[X])^2\,dx$

For any constants a and b and random variables X and Y:

- $\mathbb{E}[aX+b] = a\mathbb{E}[X] + b$
- $\mathbb{E}[X+Y] = \mathbb{E}[X] + \mathbb{E}[Y]$
- $V(aX+b) = a^2 V(X)$
- $Cov(X,Y) := \mathbb{E}[(X-\mathbb{E}[X])(Y-\mathbb{E}[Y])] = \mathbb{E}[XY] - \mathbb{E}[X]\mathbb{E}[Y]$
- $Cor(X,Y) := \frac{Cov(X,Y)}{V(x)V(y)} \in [-1,1]$
- $V(X+Y) = V(X) + V(Y) + 2Cov(X,Y)$

Independence

- X and Y are independent if $P(X < x, Y < y) = P(X < x)P(Y < y)$
- If X and Y are independent then:
  - $E(XY) = E(X)E(Y)$
  - $Cov(X, Y) = 0$ (if $Cov(X, Y) = 0$ this does not imply independence)
  - $V(X + Y) = V(X) + V(Y)$

No correlation does not mean no causality

- Let X be a random variable such that $P(X = x) = \frac{1}{3}$ if $x \in \{-1, 0, 1\}$
- Let $Y = X^2$
- X and Y are not independent (in fact Y is a function of X)
- $\mathbb{E}X = 0$
- $\mathbb{E}Y = \frac{2}{3}$
- $\mathbb{E}X^3 = 0$

$$
\begin{aligned}
Cov(X, Y) &= \mathbb{E}(X - \mathbb{E}(X))(Y - \mathbb{E}(Y)) \\
&= \mathbb{E}(X)(X^2 - \frac{2}{3}) \\
&= \mathbb{E}(X^3 - X\frac{2}{3}) \\
&= \mathbb{E}(X^3) - \frac{2}{3}\mathbb{E}(X) \\
&= 0
\end{aligned}
$$

Normal distribution

Let $X \sim N(\mu, \sigma^2)$

- The probability density function (PDF) of X is given as:

$$f_X(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

- The cumulative distribution function (CDF) of X is given as:

$$P(X < x) = F_X(x) = \int_{-\infty}^{x} f_X(x)$$

- $\mathbb{E}[X] = \mu$
- $V(X) = \sigma^2$
- A standard normal has mean zero ($\mu = 0$) and variance one ($\sigma = 1$)
- $\Phi(\cdot)$: CDF of the standard normal

Normal distribution

- For $a, b \in \mathbb{R}$ and **independent** random variables $X \sim N(\mu_X, \sigma_X^2); Y \sim N(\mu_Y, \sigma_Y^2)$
  - $aX + b \sim N(a\mu_X + b, a^2\sigma_X^2)$
  - $X + Y \sim N(\mu_X + \mu_Y, \sigma_X^2 + \sigma_Y^2)$
- Therefore
$$\frac{X - \mu_X}{\sigma_X} \sim N(0,1)$$
- The cumulative distribution function (CDF) of X is given as:

$$P(X \leq x) = P\left(\underbrace{\frac{X - \mu_X}{\sigma_X}}_{\text{Standard normal}} < \frac{x - \mu_X}{\sigma_X}\right) = \Phi\left(\frac{x - \mu_X}{\sigma_X}\right)$$

Generating Normal data

- Good for many 'real-world' variable: height, intellect, log income, education level
- Especially when those distributions tend to be tightly packed around the mean!
- Less good for variables with huge huge outliers, like stock market returns
- ‘rnorm(thismanyobs)‘ will assume ‘mean=0‘ and ‘sd=1‘

```
normaldata <- rnorm(5)
normaldata

normaldata <- rnorm(2000)
hist(normaldata,
xlab="Random Value",
main="Random Data from Normal Distribution",
probability=TRUE)
```

No correlation does not mean no causality

- Let $X \sim N(0,1)$
- Let $Y = X^2$
- X and Y are not independent (in fact Y is a function of X)
- $\mathbb{E}X = 0$
- $\mathbb{E}Y = \sigma^2$
- $\mathbb{E}X^3 = 0$

$$
\begin{aligned}
Cov(X,Y) &= \mathbb{E}(X-\mathbb{E}(X))(Y-\mathbb{E}(Y)) \\
&= \mathbb{E}(X)(X^2-\sigma^2) \\
&= \mathbb{E}(X^3-X\sigma^2) \\
&= \mathbb{E}(X^3)-\sigma^2\mathbb{E}(X) \\
&= 0
\end{aligned}
$$

Uniform distribution

Let $X \sim U(a, b)$

- $f_X(x) = \begin{cases} \frac{1}{b-a} & \text{if } a \leq x \leq b \\ 0 & \text{otherwise} \end{cases}$
- $\mathbb{E}[X] = \frac{b+a}{2}$
- $V(X) = \frac{(b-a)^2}{12}$
- $cX \sim U(ca, cb)$
- $X + d \sim U(a + d, b + d)$

Generating uniform data

- Good for variables that should be bounded: e.g., “percent male” can only be 0-1
- Gives even probability of getting each value
- ‘runif(thismanyobs,min,max)‘ will draw ‘thismanyobs‘ observations from the range of ‘min‘ to ‘max‘.
- ‘runif(thismanyobs)‘ will assume ‘min=0‘ and ‘max=1‘

```
uniformdata <- runif(5)
uniformdata

uniformdata <- runif(2000)
hist(uniformdata,xlab="Random Value",
main="Random Data from Uniform Distribution",
probability=TRUE)
```

Generating Other Kinds of Data

- ‘sample()‘ picks randomly from categories (e.g., Heads/Tails) or integers (e.g., ‘1:10‘)
- R can generate random data from other distributions. See ‘help(Distributions)‘
- We have looked quickly at two:
  - The **uniform** distribution
  - The **normal** distribution
- But don’t forget there are more
- When generating “random” data: set a seed so you can reproduce the results (‘set.seed(XXX)’)

Law of large numbers

- Let $X_1, ..., X_N$ be independent and identically distributed (iid) with mean μ and variance σ^2
  - $\mathbb{E}\left[\sum_{i=1}^{N} X_i\right] = N\mu$
  - $V\left(\sum_{i=1}^{N} X_i\right) = N\sigma^2$
  - $V\left(\frac{1}{N}\sum_{i=1}^{N} X_i\right) = \frac{1}{N}\sigma^2$
  - $\mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N} X_i\right] = \mu$
- As n grows, the variance goes to zero, but the mean is always μ
- That is, the mean of the random variables ($\overline{X}$) converges (in probability) to μ

Example: Coin flips

- Throw a coin 1,000 times
- Let's create a random variable $X = \begin{cases} 1 & \text{if } \textit{coin} = \textit{Heads} \\ 0 & \text{if } \textit{coin} = \textit{tails} \end{cases}$
- $\mathbb{E}(X) = 1\frac{1}{2} + 0\frac{1}{2} = \frac{1}{2}$
- $V(X) = (1 - 0{,}5)^2\frac{1}{2} + (0 - 0{,}5)^2\frac{1}{2} = \frac{1}{4}$
- $\overline{X}$ proportion of times coin lands on heads
- $\mathbb{E}\overline{X} = \frac{1}{2}$
- $\mathbb{V}\overline{X} = \frac{1}{4N}$

Example: Coin flips

A little simulation:

```
## Generate data with 1000 coin flips
## Pprob of head and tail is the same
data <- sample(c("Heads","Tails"),1000,replace=TRUE)
## Create random variable (one if heads, zero if tails
    )
X<-as.numeric(data=="Heads")
# Calculate the proportion of heads of the first n
    observations
X_n<-cumsum(X)/(1:1000)
#Plot the results
plot(1:1000,X_n,bty="L",ylim=c(0,1),
ylab="Average",xlab="Tosses",type="l",lwd=2,
cex.lab=1.5,cex.axis=1.5,cex.main=1.5)
abline(h=0.5,lty=2,col=2,lwd=2)
```