Week 7 7.2 Note

Lecture 22: Law of Averages

---

1. Chance Processes

- Every time you toss a fair coin, there is a **chance variability**:
    - Numbers of Head = Half the number of tosses + Chance Error
    - **Observed value = Expected Value + Chance Error**

> **Law of Large Numbers/Averages**: the proportion of heads become more stable as the **length of the simulation increases** and approaches a fixed number called **relative frequency**.

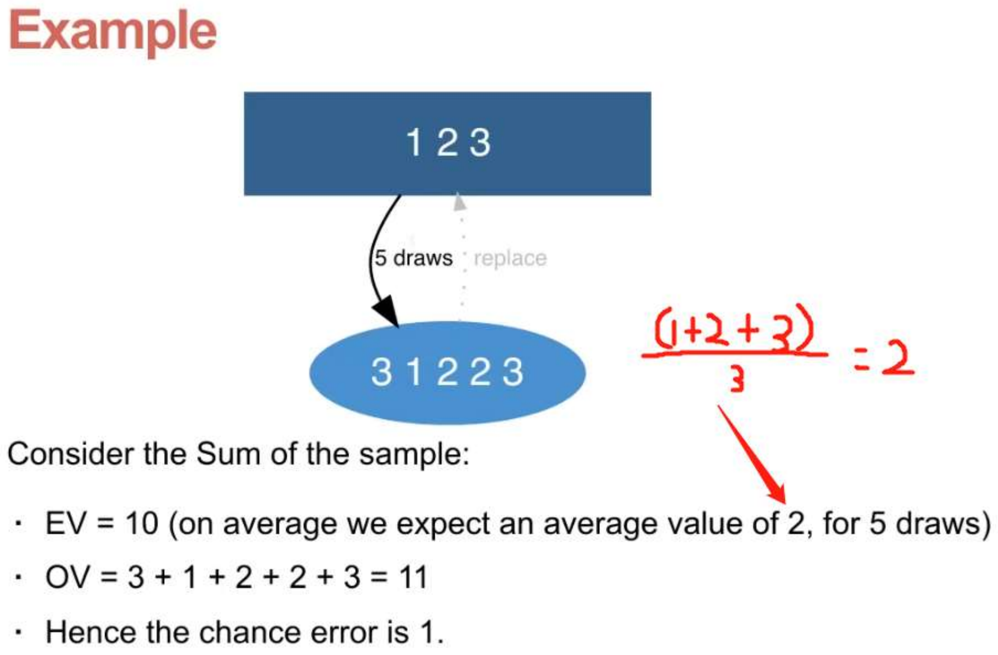
- The chance error in the **number of heads** is likely to be **large**, but **small** relative to the **number of tosses**.

> **Gambler's Fallacy（赌徒谬论）**: For independent events, it is **wrong to assume the chance of observing an event over time** even if it has not occurred for a long time.
>
> - As the number of tosses increases:
>   - The absolute size of the chance error increases
>   - The absolute percentage of the chance error increases
>   - The proportion of event will coverage to the expected proportion.

2. Box Model

- A simple wat to describe **chance processes**.
- We need to know:
    - The **distinct numbers** in the box ("tickets")
    - The **number** of each tickets in the box
    - The number of **draws** from the box
- Steps:
    - Think of the **box** as a summary of the **population**
        - What's in there, and in what proportions.
    - Take **draws** from the box to create the **sample**
    - Consider the *sum or mean* of the *sample*
        - What is the expected value (EV)?
        - What is the observed value (OV)?

- The *chance error* is *OV - EV*, modelled by **standard error (SE)**.



Example

$\frac{(1+2+3)}{3} = 2$

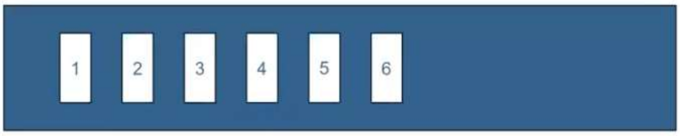
Consider the Sum of the sample:

- EV = 10 (on average we expect an average value of 2, for 5 draws)
- OV = 3 + 1 + 2 + 2 + 3 = 11
- Hence the chance error is 1.

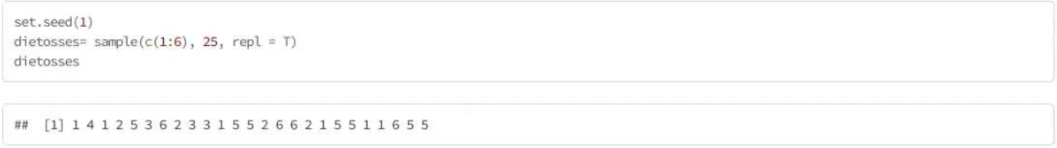
3. Applying Box Model to gambling

- Concepts in gambling:
  - The *tickets* represent the amount *won (+) and lost (-) in each play*.
  - The chance of *drawing a particular value* is the chance of *winning that amount in 1 play*.
  - The number of *draws* is the number of *plays*.
  - The *net gain* is the *sum of the draws*.
- Example: Throw a fair dice 25 times.
  - Box Composition:
    - The distinct tickets are 1,2,3,4,5,6, representing the 6 unique faces of the dice.
    - There is 1 of each ticket, as the dice is fair.

Box model of toss of fair dice



  - Simulation:



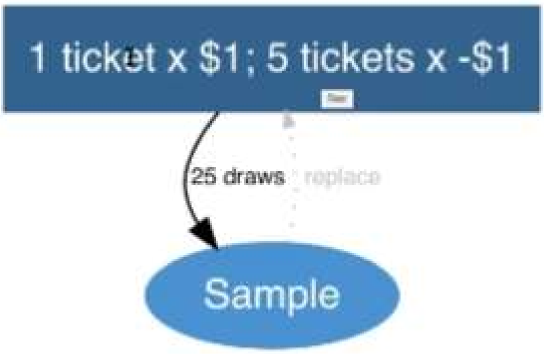
```
set.seed(1)
dietosses= sample(c(1:6), 25, repl = T)
dietosses
```

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

This simulation represents one possible sample formed by 25 throws of the dice.

- Example 2: Game

- Model:
  - Suppose it costs $1 to play a game.
  - If you roll a “6”, you get back your $1, plus win another dollar.
  - If you get any other number, you lose your $1.
  - Play 25 times. What is your net gain/loss?


1 ticket x $1; 5 tickets x -$1

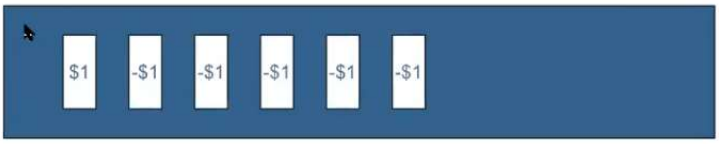
25 draws

replace

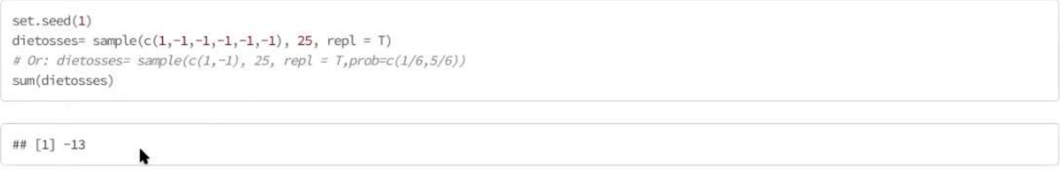
Sample

- Box Composition:
  - The distinct tickets are $1 and -$1, representing the “win” and “loss”.
  - There is 1 ticket with $1 (equivalent to tossing a “6”), and 5 tickets with -$1 (equivalent to tossing a “1”,“2”,“3”,“4”,“5”).

Box model of throw of fair dice



- Simulation:



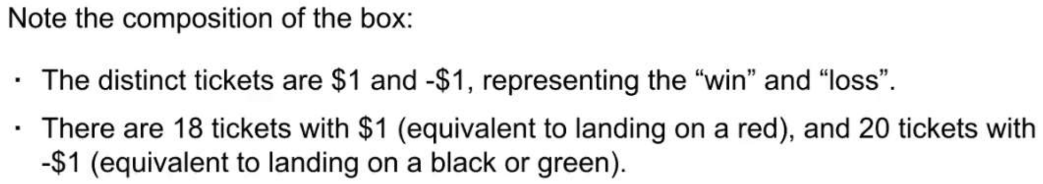
```
set.seed(1)
dietosses= sample(c(1,-1,-1,-1,-1,-1), 25, repl = T)
# Or: dietosses= sample(c(1,-1), 25, repl = T,prob=c(1/6,5/6))
sum(dietosses)
```

```
## [1] -13
```

This simulation represents one possible net “winnings”, after playing the games 25 times.

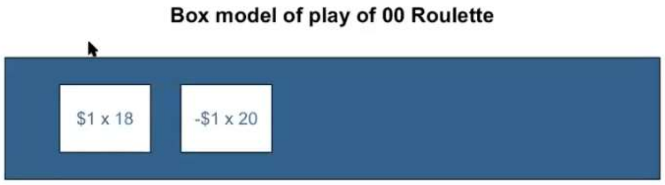
- Example 3: Roulette
  - Model:
    - The Star Casino 00 roulette wheel has 38 pockets,numbered 0 (green), 00 (green) and 1-36 (alternate red and black).
    - Suppose you place a bet on red. This costs $1.
    - The croupier spins the wheel until a ball lands in a pocket.
      - If it lands on a red, you get your $1 back plus an extra $1.
      - If it doesn’t land on red, you lose your $1.
    - You play 10 times. What is your expected net gain/loss?

- Box Composition:

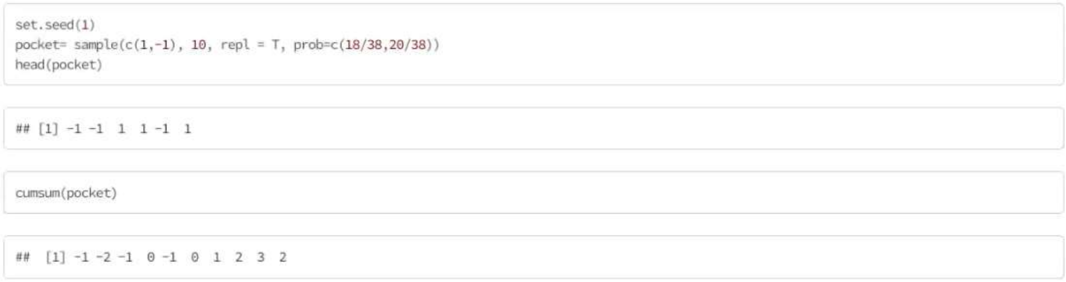
  Note the composition of the box:

  - The distinct tickets are \$1 and -\$1, representing the “win” and “loss”.
  - There are 18 tickets with \$1 (equivalent to landing on a red), and 20 tickets with -\$1 (equivalent to landing on a black or green).

  Box model of play of 00 Roulette

- Simulation:

```
set.seed(1)
pocket= sample(c(1,-1), 10, repl = T, prob=c(18/38,20/38))
head(pocket)
```

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

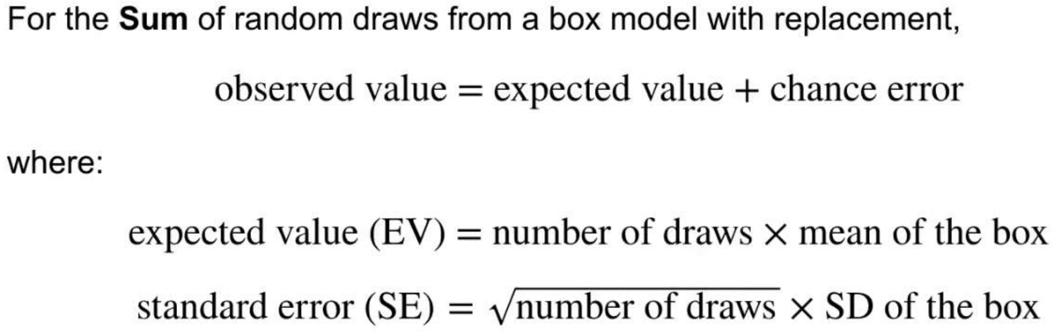
```
cumsum(pocket)
```

```
##  [1] -1 -2 -1  0 -1  0  1  2  3  2
```

Lecture 2: The Box Model

---

1. Modelling the Sum/Mean of a Sample

- **Sum of draws** from a box model with replacement:
  - 

    For the **Sum** of random draws from a box model with replacement,

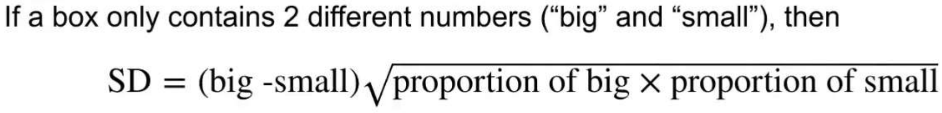
    $$\text{observed value} = \text{expected value} + \text{chance error}$$

    where:

    $$\text{expected value (EV)} = \text{number of draws} \times \text{mean of the box}$$

    $$\text{standard error (SE)} = \sqrt{\text{number of draws}} \times \text{SD of the box}$$

  - SD of the box:
    - The result for SE is **square root law** (sqrt(number of draws))
    - Box is a population so the SD of the box is **population SD**.
    - Calculations:
      1. Use `popsd()` in `multicon` package
      2. Formula: RMS(gaps) = Root of the Mean of the Square Gaps
      3. Shortcut: Simple **Binary Boxes** only

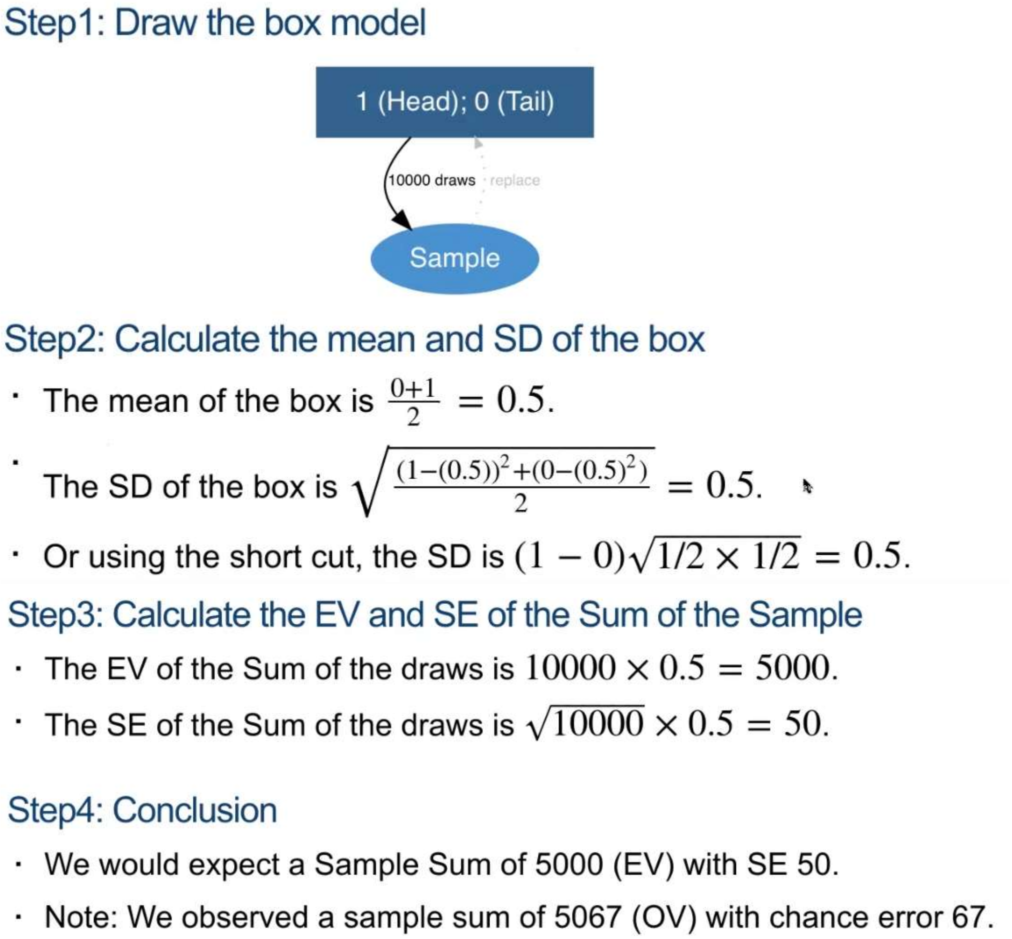
         

         If a box only contains 2 different numbers (“big” and “small”), then

         $$\text{SD} = (\text{big -small})\sqrt{\text{proportion of big} \times \text{proportion of small}}$$

  - How does chance error *relate to* standard error?
    - An OV is likely to be around its EV, with a **chance error similar to the SE**.
    - OVs are rarely more than 2 or 3 SEs away from the EVs.
  - Example: WWII Coin Tossing

- Steps:



Step1: Draw the box model

Step2: Calculate the mean and SD of the box

- The mean of the box is $\frac{0+1}{2} = 0.5.$
- The SD of the box is $\sqrt{\frac{(1-(0.5))^2+(0-(0.5)^2)}{2}} = 0.5.$
- Or using the short cut, the SD is $(1-0)\sqrt{1/2 \times 1/2} = 0.5.$

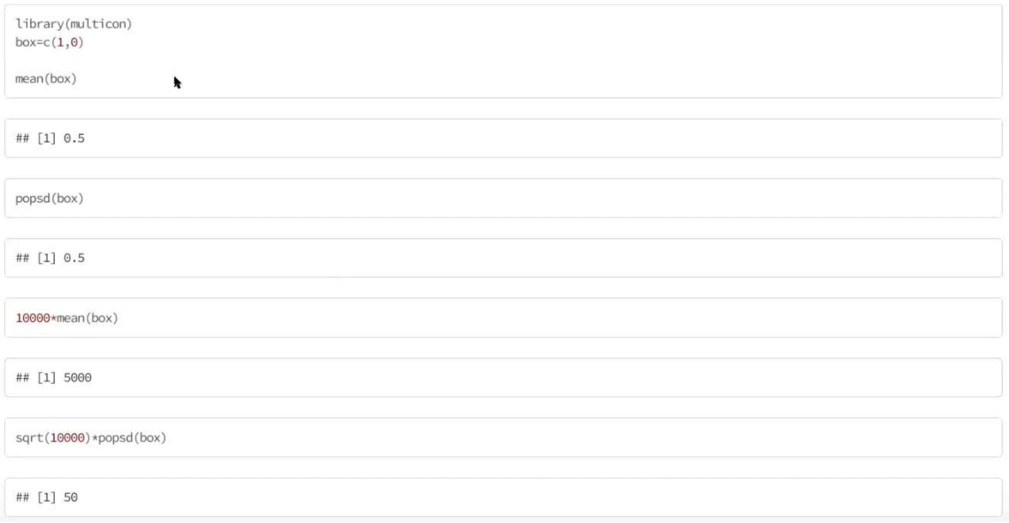
Step3: Calculate the EV and SE of the Sum of the Sample

- The EV of the Sum of the draws is $10000 \times 0.5 = 5000.$
- The SE of the Sum of the draws is $\sqrt{10000} \times 0.5 = 50.$

Step4: Conclusion

- We would expect a Sample Sum of 5000 (EV) with SE 50.
- Note: We observed a sample sum of 5067 (OV) with chance error 67.

- We say that **the chance error 67 is within 2 standard errors (SE = 50)**.



```
library(multicon)
box=c(1,0)

mean(box)
```

```
## [1] 0.5
```

```
popsd(box)
```

```
## [1] 0.5
```

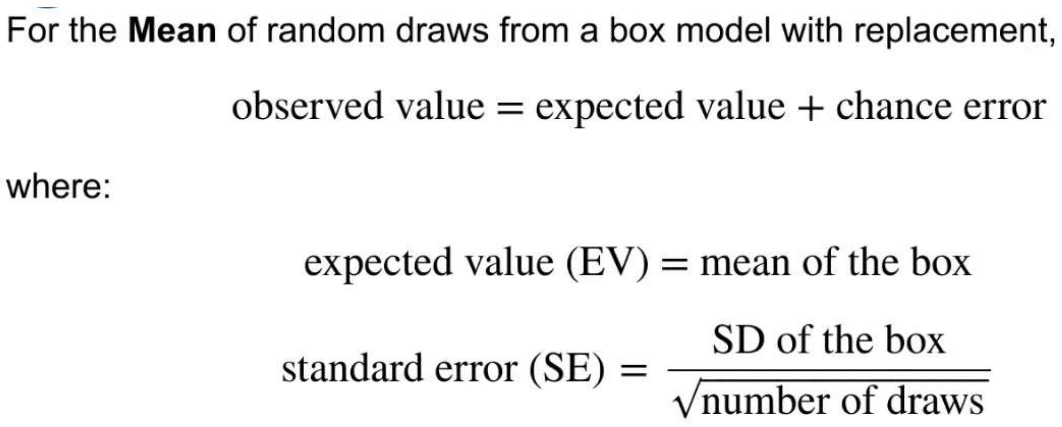
```
10000*mean(box)
```

```
## [1] 5000
```

```
sqrt(10000)*popsd(box)
```

```
## [1] 50
```

- **Mean of draws** from a box model with replacement:
  - **Mean** of the sample = **Sum** of the sample / the **number** of draws
  -



For the **Mean** of random draws from a box model with replacement,

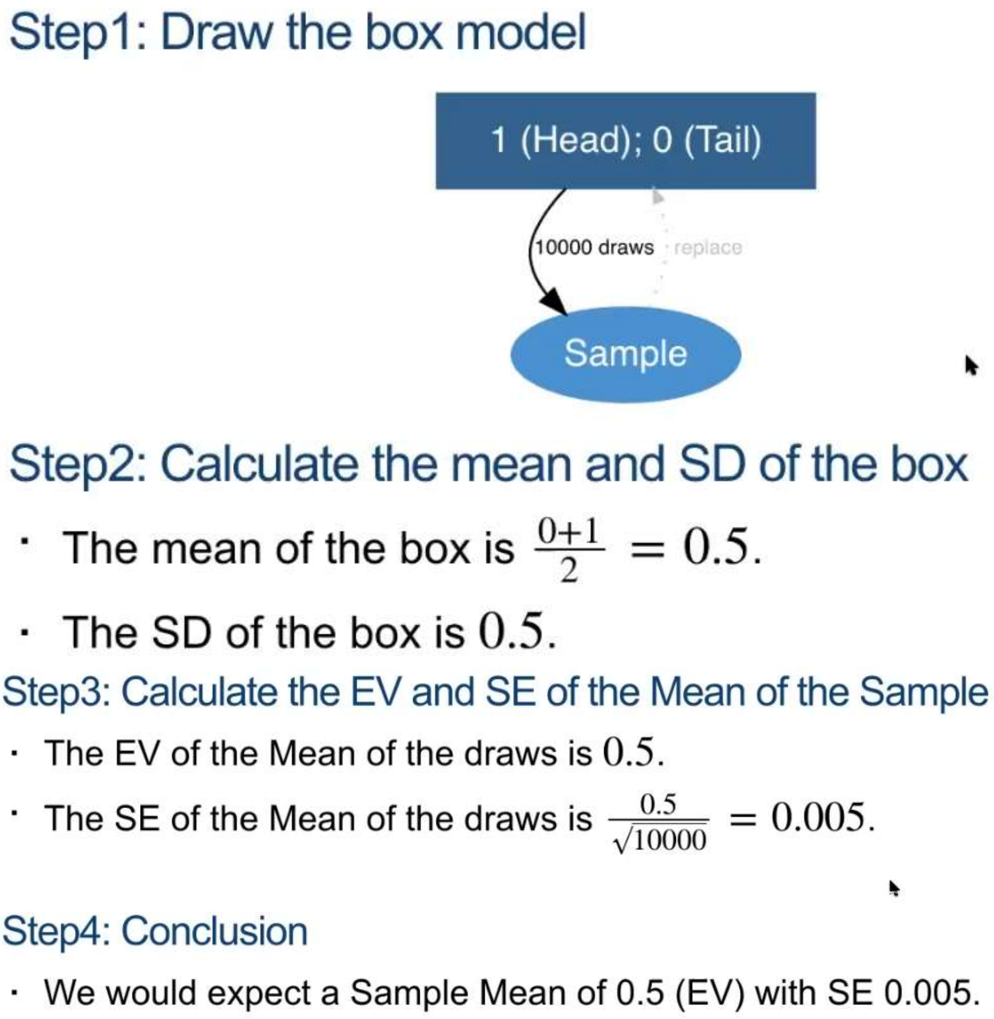
$$\text{observed value} = \text{expected value} + \text{chance error}$$

where:

$$\text{expected value (EV)} = \text{mean of the box}$$

$$\text{standard error (SE)} = \frac{\text{SD of the box}}{\sqrt{\text{number of draws}}}$$

- Example: WWII Coin Tossing
  - Steps:



Step1: Draw the box model

Step2: Calculate the mean and SD of the box

- The mean of the box is $\frac{0+1}{2} = 0.5$.
- The SD of the box is 0.5.

Step3: Calculate the EV and SE of the Mean of the Sample

- The EV of the Mean of the draws is 0.5.
- The SE of the Mean of the draws is $\frac{0.5}{\sqrt{10000}} = 0.005$.

Step4: Conclusion

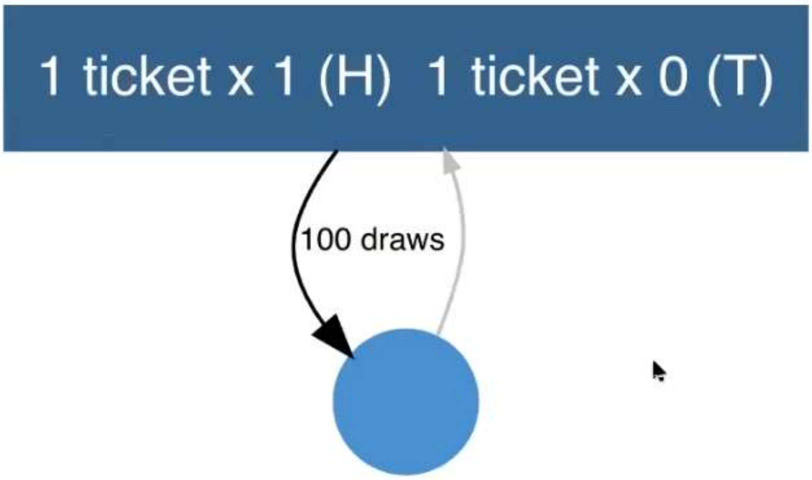
- We would expect a Sample Mean of 0.5 (EV) with SE 0.005.

2. The Normal Curve for Modelling

- For **large amounts** of draws from the box, the **observed value** of the Sum/Mean often **follows the normal curve**.
- Given a box model, we can work out EV and SE to model.
- Example: A coin is rolled *100 times*. What is the *chance of getting between 40 and 60 heads*?

- Steps:

Step1: Draw the box model



Step2: Calculate the mean and SD of the box

- The mean is $\frac{1+0}{2} = 0.5$.
- The SD is $(1 - 0)\sqrt{1/2 \times 1/2} = 0.5$.

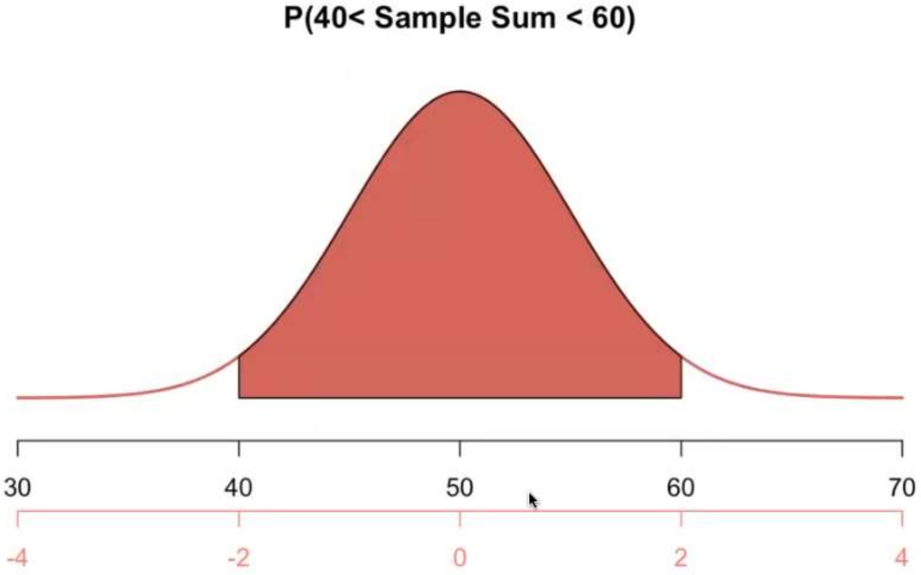
Step3: Calculate the EV and SE of the Sum of the Sample

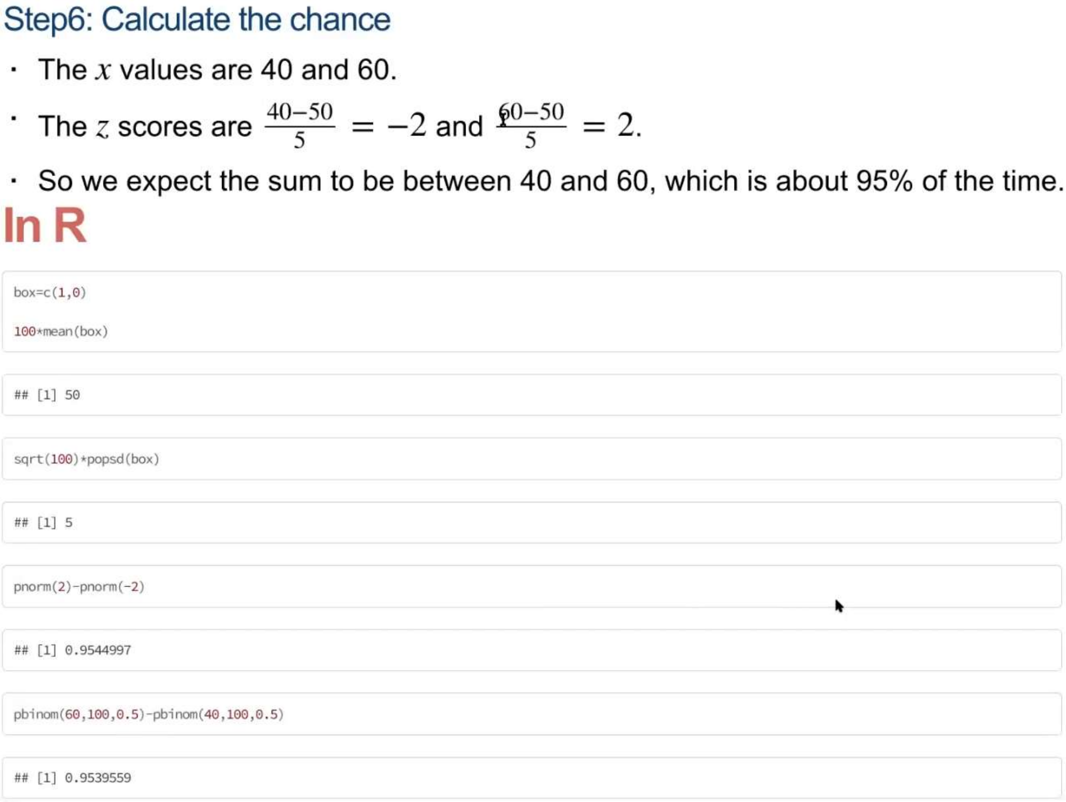
- The EV of the Sum of the draws is $100 \times 0.5 = 50$.
- The SE of the Sum of the draws is $\sqrt{100} \times 0.5 = 5$.

Step4: Conclusion

- We would expect a Sample Sum of 50 (EV) with SE 5.
- So now model the Sample Sum by a Normal, with mean = 50 and SD = 5. ie Sample Sum ~ N(50, 5^2).

Step5: Draw the Normal curve





Step6: Calculate the chance

- The x values are 40 and 60.
- The z scores are $\frac{40-50}{5} = -2$ and $\frac{60-50}{5} = 2$.
- So we expect the sum to be between 40 and 60, which is about 95% of the time.

In R

```
box=c(1,0)

100*mean(box)
```

```
## [1] 50
```

```
sqrt(100)*popsd(box)
```

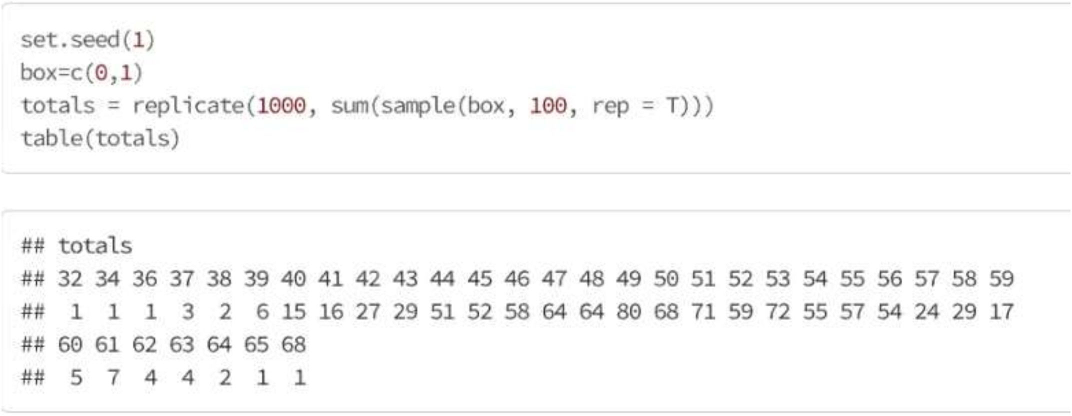
```
## [1] 5
```

```
pnorm(2)-pnorm(-2)
```

```
## [1] 0.9544997
```

```
pbinom(60,100,0.5)-pbinom(40,100,0.5)
```

```
## [1] 0.9539559
```

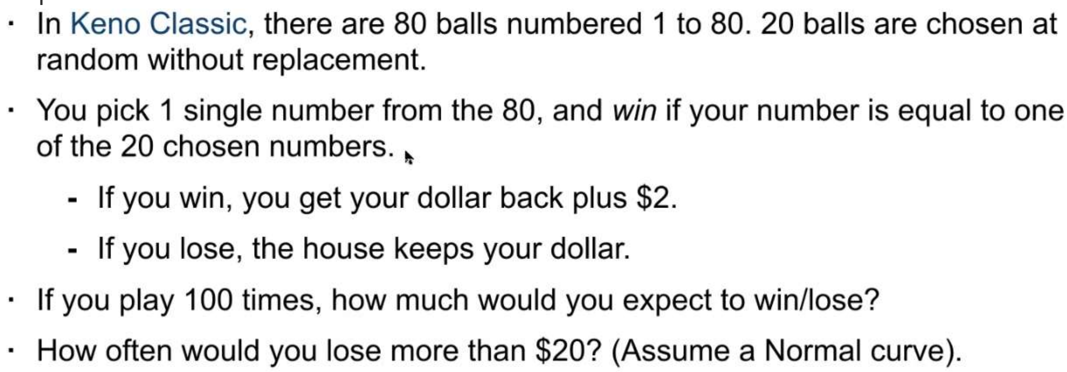


```
set.seed(1)
box=c(0,1)
totals = replicate(1000, sum(sample(box, 100, rep = T)))
table(totals)
```

```
## totals
## 32 34 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59
##  1  1  1  3  2  6 15 16 27 29 51 52 58 64 64 80 68 71 59 72 55 57 54 24 29 17
## 60 61 62 63 64 65 68
##  5  7  4  4  2  1  1
```

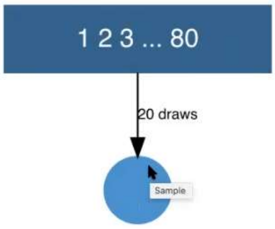
3. Using Box Model for Classifying and Counting

- Often we are just interested in 1 particular ticket in the box (binary box), which may summarize other tickets.
- Example 1: Toss a dice 100 times and count the number of 6s. The box would have one 1 (representing "6"), and five 0 (representing non "6")
- Example 2: Keno Classic
    - Steps:



- In Keno Classic, there are 80 balls numbered 1 to 80. 20 balls are chosen at random without replacement.
- You pick 1 single number from the 80, and *win* if your number is equal to one of the 20 chosen numbers.
    - If you win, you get your dollar back plus $2.
    - If you lose, the house keeps your dollar.
- If you play 100 times, how much would you expect to win/lose?
- How often would you lose more than $20? (Assume a Normal curve).

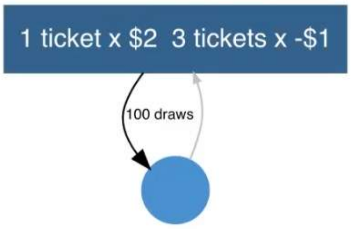
Preliminary set up (draw 20 numbers from the 80, without replacement)



Step1: Draw the box model (for game)

20 win → $\frac{1}{4}$

60 lose → $\frac{3}{4}$



Step2: Calculate the mean and SD of the box

- The mean of the box is $\frac{2-1-1-1}{4} = -0.25.$
- The SD of the box is $(2-(-1))\sqrt{1/4 \times 3/4} = 1.299038.$

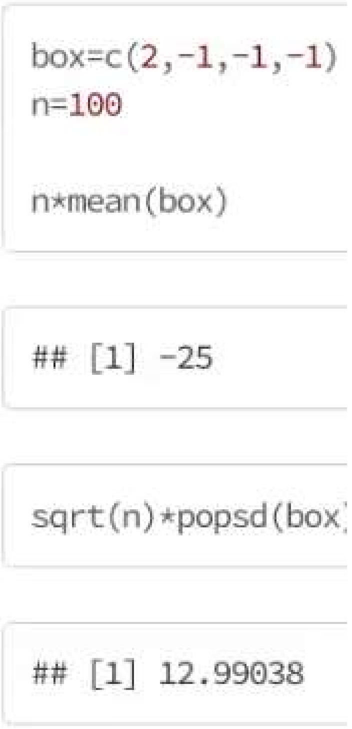
Step3: Calculate the EV and SE of the Sum of the Sample

- The EV of the Sum of the draws is $100 \times -0.25 = -25.$
- The SE of the sum of draws is $\sqrt{100} \times 1.299038 = 12.99.$

Step4: Conclusion

- In 100 plays of Classic Keno we expect to lose $25 (EV) with a SE of $13.
- Hence, it would be very common to lose between $12 and $38.

In R



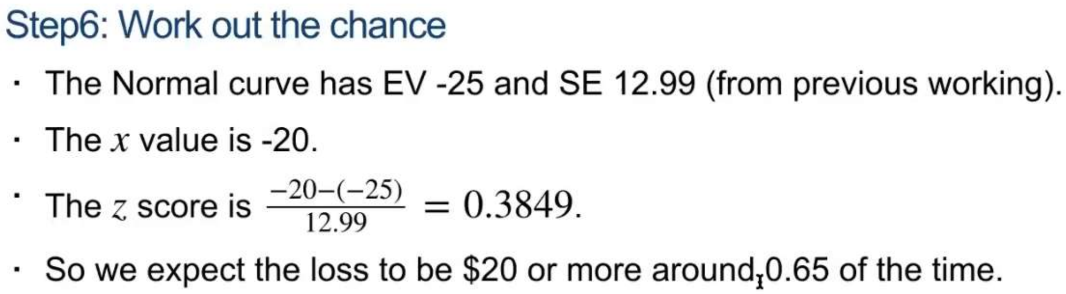
```
box=c(2,-1,-1,-1)
n=100

n*mean(box)
```

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

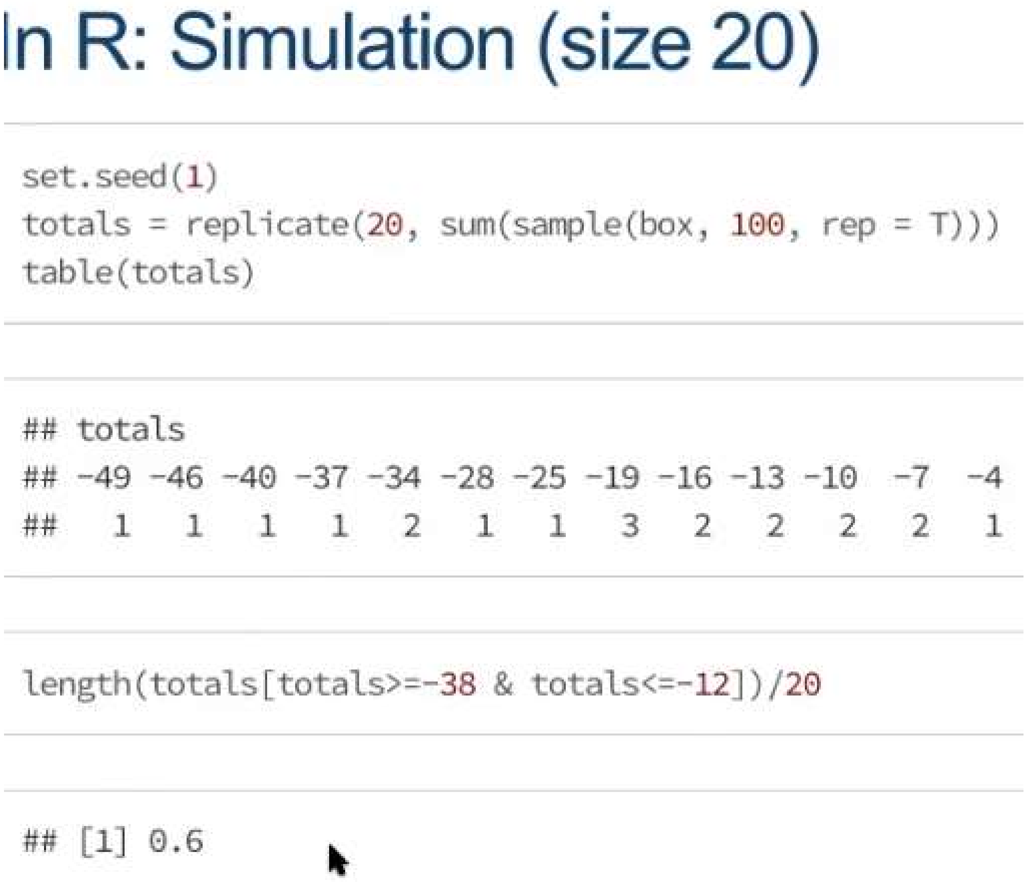
```
sqrt(n)*popsd(box
```

```
## [1] 12.99038
```



Step6: Work out the chance

- The Normal curve has EV -25 and SE 12.99 (from previous working).
- The x value is -20.
- The z score is $\frac{-20-(-25)}{12.99} = 0.3849.$
- So we expect the loss to be $20 or more around 0.65 of the time.



In R: Simulation (size 20)

```
set.seed(1)
totals = replicate(20, sum(sample(box, 100, rep = T)))
table(totals)
```

```
## totals
## -49 -46 -40 -37 -34 -28 -25 -19 -16 -13 -10  -7  -4
##   1   1   1   1   2   1   1   3   2   2   2   2   1
```

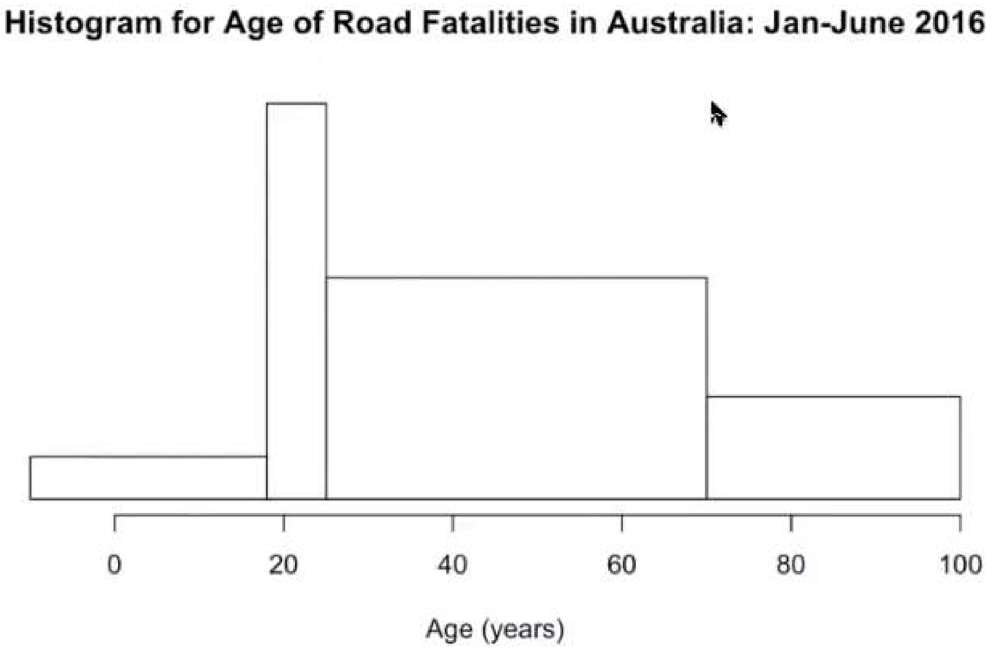
```
length(totals[totals>=-38 & totals<=-12])/20
```

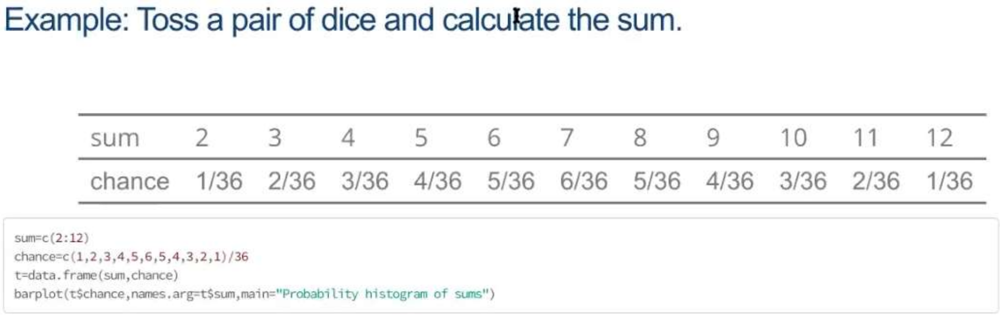
```
## [1] 0.6
```

Lecture 24: Normal Approximation

1. The Probability Histogram

- 3 Types of Histogram:
    - Data Histogram `hist()`:
        - Represents data by area

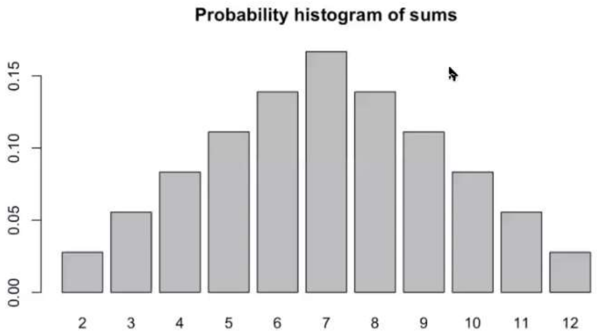
- 

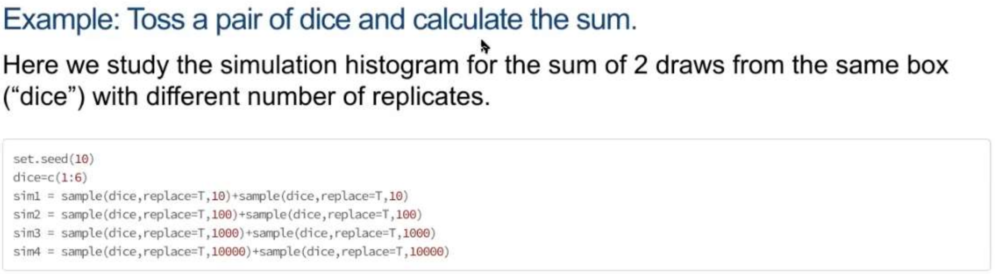
- Probability Histogram:
  - Represents chance by area
  - The **EV** measures the **centre** on x-axis.
  - The **SE** measures the **spread** on x-axis.
  - 

    Example: Toss a pair of dice and calculate the sum.

sum	2	3	4	5	6	7	8	9	10	11	12
chance	1/36	2/36	3/36	4/36	5/36	6/36	5/36	4/36	3/36	2/36	1/36

    ```
    sum=c(2:12)
    chance=c(1,2,3,4,5,6,5,4,3,2,1)/36
    t=data.frame(sum,chance)
    barplot(t$chance,names.arg=t$sum,main="Probability histogram of sums")
    ```

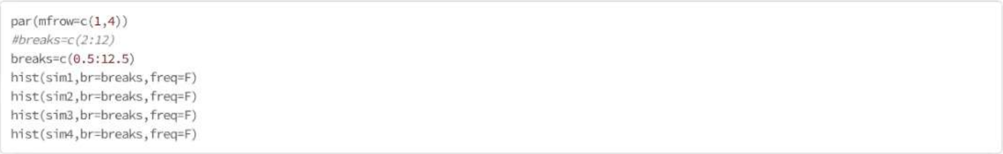
    

- Simulation (empirical) Histogram:
  - Represents chance by area for a simulation of a chance process
  - Convergence: for repeated simulations of a chance process resulting in a sum, the simulation histogram of the observed values converges to the probability histogram
  - 

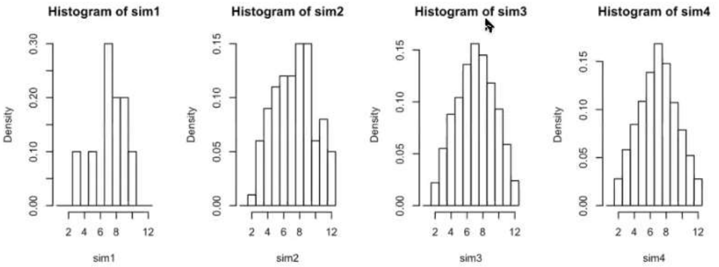
    Example: Toss a pair of dice and calculate the sum.

    Here we study the simulation histogram for the sum of 2 draws from the same box ("dice") with different number of replicates.

    ```
    set.seed(10)
    dice=c(1:6)
    sim1 = sample(dice,replace=T,10)+sample(dice,replace=T,10)
    sim2 = sample(dice,replace=T,100)+sample(dice,replace=T,100)
    sim3 = sample(dice,replace=T,1000)+sample(dice,replace=T,1000)
    sim4 = sample(dice,replace=T,10000)+sample(dice,replace=T,10000)
    ```



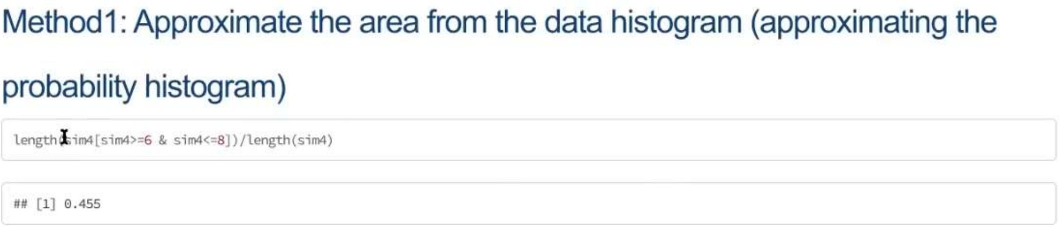
```
par(mfrow=c(1,4))
#breaks=c(2:12)
breaks=c(0.5:12.5)
hist(sim1,br=breaks,freq=F)
hist(sim2,br=breaks,freq=F)
hist(sim3,br=breaks,freq=F)
hist(sim4,br=breaks,freq=F)
```

- ■ 

2. Central Limit Theorem（中心极限定理）

> Central Limit Theorem: When **drawing at random with replacement** from a box, if the **sample size** for the sum (or average) is **sufficiently large**, then:

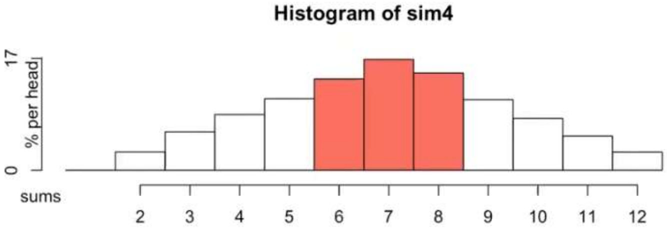
- The **probability histogram** for the sum (or average) will closely **follow the normal curve**.
- Even if the contents of the box does not.
- **More generally, the distribution (behavior of the chances) for the sum (or average) will closely follow the normal curve.**
- Example: Toss a pair of dice 10000 times and calculate the sum. What is the probability of getting a sum between 6 and 8?
  - Method 1:



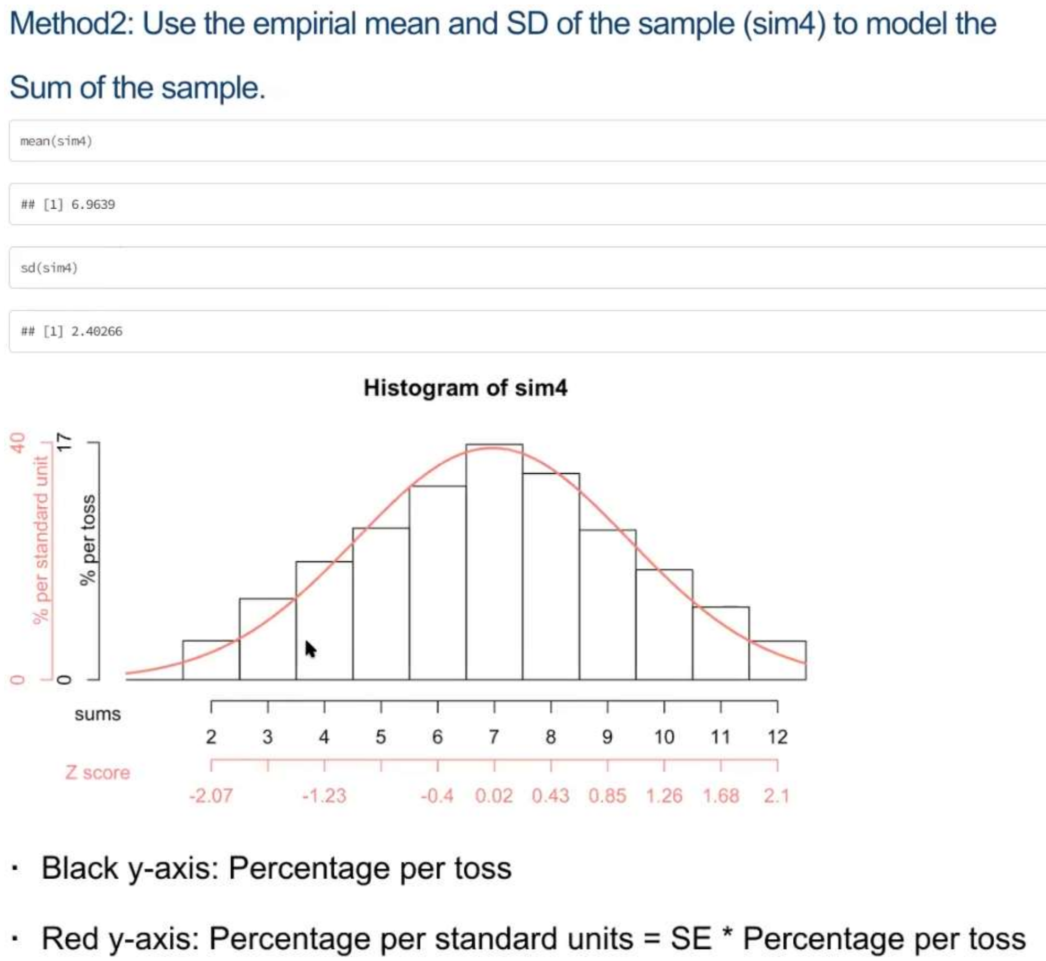
Method1: Approximate the area from the data histogram (approximating the probability histogram)

```
length(sim4[sim4>=6 & sim4<=8])/length(sim4)
```

```
## [1] 0.455
```



- Method 2:



Method2: Use the empirial mean and SD of the sample (sim4) to model the Sum of the sample.

```
mean(sim4)
```

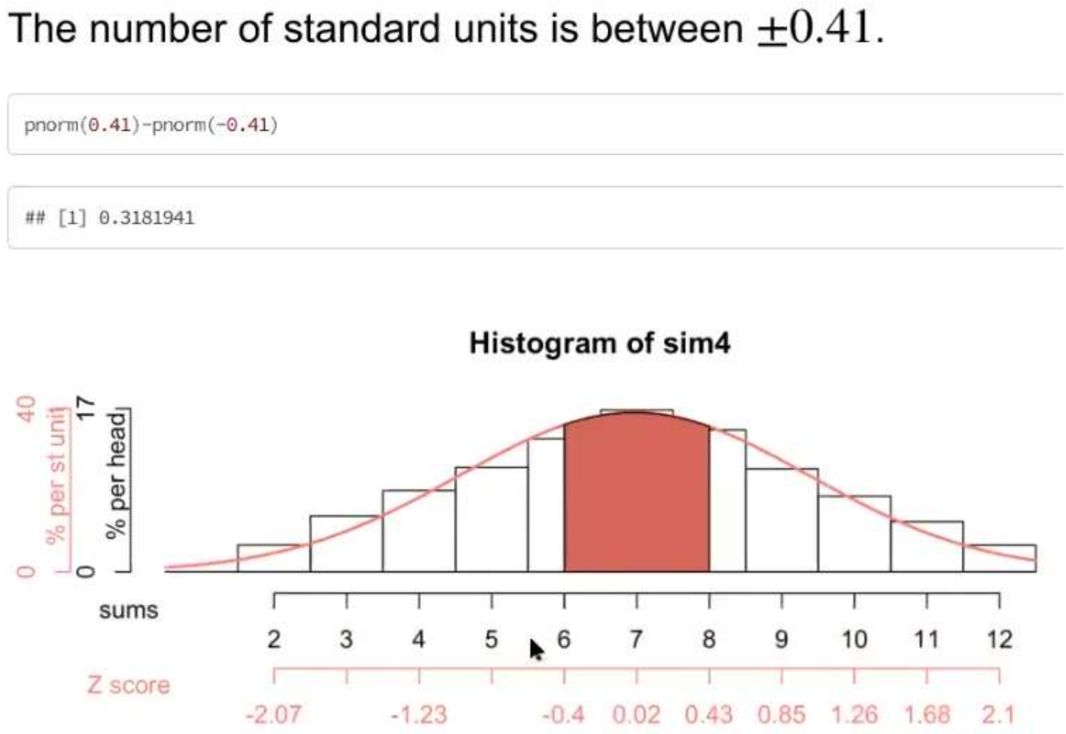
```
## [1] 6.9639
```

```
sd(sim4)
```

```
## [1] 2.40266
```

Histogram of sim4

- Black y-axis: Percentage per toss
- Red y-axis: Percentage per standard units = SE * Percentage per toss



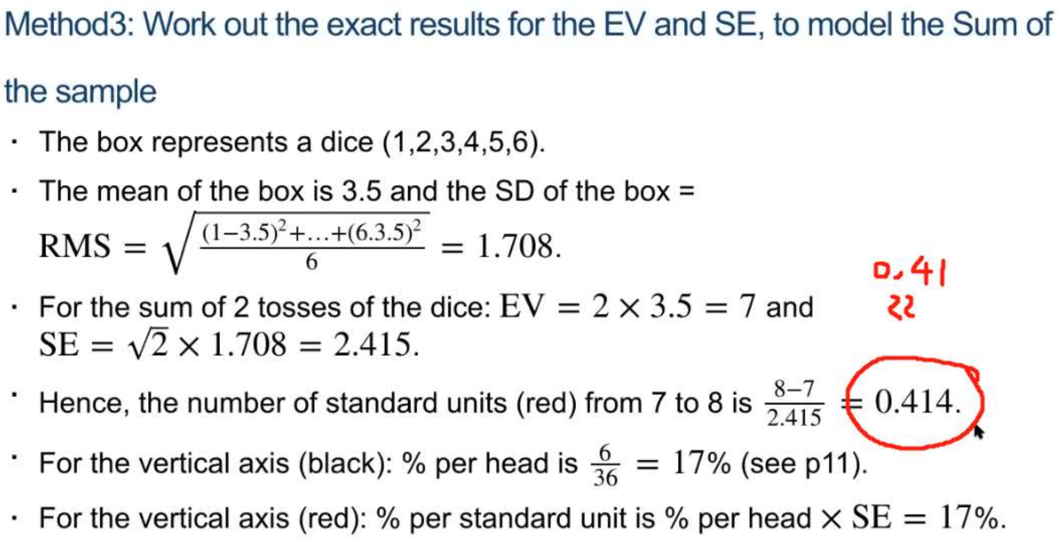
The number of standard units is between ± 0.41.

```
pnorm(0.41)-pnorm(-0.41)
```

```
## [1] 0.3181941
```

Histogram of sim4

- Method 3:

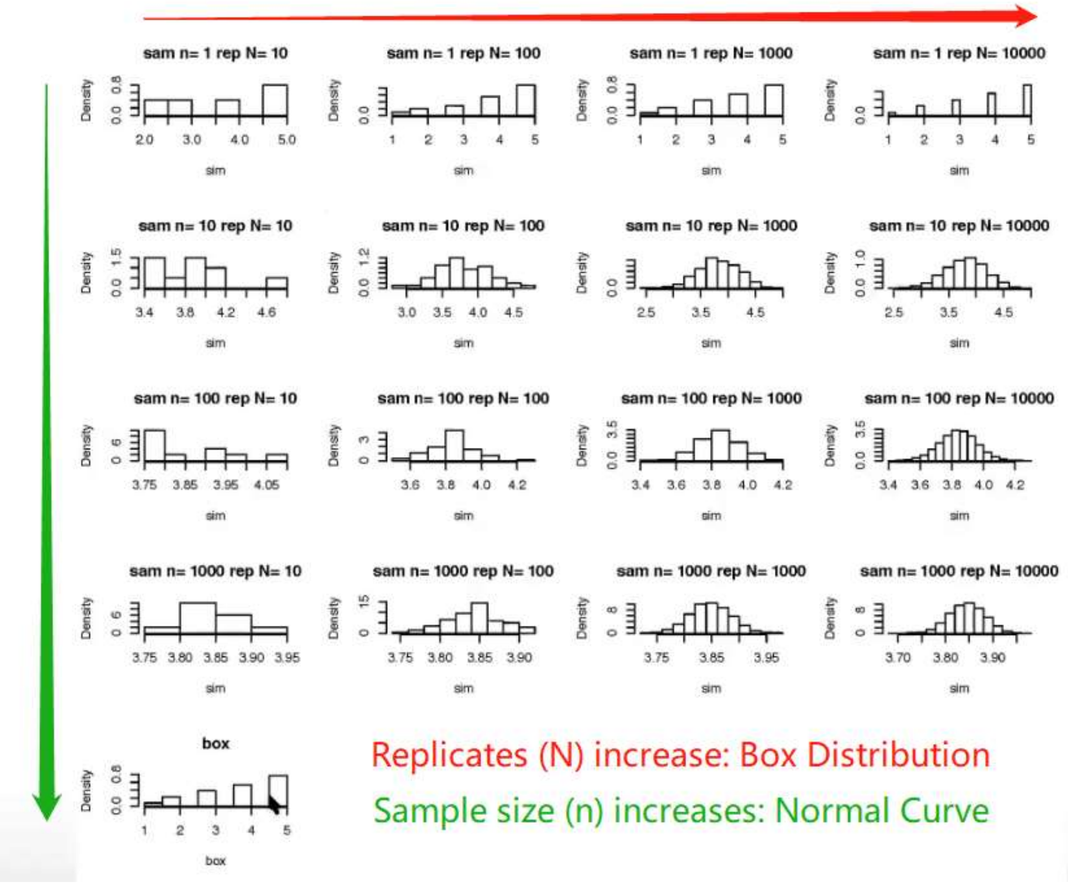


Method3: Work out the exact results for the EV and SE, to model the Sum of the sample

- The box represents a dice (1,2,3,4,5,6).
- The mean of the box is 3.5 and the SD of the box = $\text{RMS} = \sqrt{\frac{(1-3.5)^2+\ldots+(6.3.5)^2}{6}} = 1.708.$
- For the sum of 2 tosses of the dice: $\text{EV} = 2 \times 3.5 = 7$ and $\text{SE} = \sqrt{2} \times 1.708 = 2.415.$
- Hence, the number of standard units (red) from 7 to 8 is $\frac{8-7}{2.415} = 0.414.$
- For the vertical axis (black): % per head is $\frac{6}{36} = 17\%$ (see p11).
- For the vertical axis (red): % per standard unit is % per head $\times$ SE $= 17\%$.

0.41
22

> Continuity Correction (for edges): to approximate a discrete distribution by the normal distribution (continuous), we **adjust the endpoints by 0.5**.
>
> - The normal curve is **missing part of the area calculated by the data histogram**. To remedy this we adjust by 0.5 either side.
>   - Lower Threshold = 6 - 5.5, this has standard unit = -0.6 (approx)
>   - Upper Threshold = 8 - 8.5, this has standard unit = 0.6 (approx)
> - To workout whether add of minus 0.5, **draw a sketch of the histogram**.

- Sample Size vs. Replicates
  - Increasing the **replicate** will approach the **box distribution**.
  - Increasing the **sample size** will approach the **normal curve**.
  - 

Replicates (N) increase: Box Distribution

Sample size (n) increases: Normal Curve