**1.VARIANCE**

\*\*Mean: Center of distribution

\*\*Variance & SD: Distribution's spread around mean

--> Var(X) = E( (X-mu)^2 ) = E( (X-E(X))^2 )=E(X^2)-E(X)^2

where E(X) is polulation mean

implying higher variance means more spread around a mean than lower variance

-->Var(aX)=a^2\*Var(X)

-->sample variance is ALMOST the average squared deviation from

the sample mean.(we divide sum of n squared distance by n-1, coz last n is dependent on sample mean)

But, sample mean is same as population mean, but in sample vairance

it is not the case.

-->E(X') = mu, where X' represents a sample mean and mu is the population mean.

Variance of sample mean:Var(X') = sigma^2 / n, where sigma^2 is population variance adn n is sample size

-->SD--Standard error

SD of sample mean -- sqrt of variance of sample mean --sqrt(Var(X'))==s/sqrt(n),s is sample SD

-->s:sample standard deviation:how variable population is

standard error:how much averages of random samples of size n from population vary

**2.Common Distributions**

2.1 Bernoulli

->only 2 possible outcomes

->PMF:p^x\*(1-p)^(1-x)

->Mean:x\*p(x), where x=0,1, hence mean =p

->E(X^2):x^2\*p(x),where x=0,1 hence =p

->Var(X)= E(X^2)-(E(X))^2=p(1-p)

->pbinom()

2.2 Normal(or Gausian)

->qnorm(quantile=)

->X=mu+sigma\*Z, where X is normally distributed and Z is distribution.

->pnorm(1200,mean=?,sd=?,lower.tail=FALSE)

\*\*\*\*Pnorm and qnorm are inverses

2.3 Poison distribution

->takes integer values

->mean & variance is lambda

->used to model rates

->expressed as X~Poisson(lambda\*t)

NOTE:: **Poisonn distribution is similar to Binominal distribution when size of sample(n) is large and probability (p) is less**

**3.Asymptotics**

1. LLN : forms basis of frequency style thinking

coinPlot

function(n){

means <- cumsum(sample(0 : 1, n , replace = TRUE)) / (1 : n)

g <- ggplot(data.frame(x = 1 : n, y = means), aes(x = x, y = y))

g <- g + geom\_hline(size=1.5 ,yintercept = 0.5,alpha=0.6,

linetype="longdash") + geom\_line(size = 1)

if(n<100){

g <- g + geom\_point(colour="red",size=3,alpha=0.8)

}

g <- g + labs(x = "Number of obs", y = "Cumulative mean")

g <- g + scale\_x\_continuous(breaks=seq(0,n+1,ceiling(n/10)))

print(g)

invisible()

}

2. CLT : It states that the distribution of averages of iid variables (properly normalized) becomes that of a standard normal as the sample size increases. OR

CLT that for large n, the sample mean is normal with mean mu and standard deviation sigma/sqrt(n)

🡪Properly normalized: (X’(sample mean)-mu(population mean))/(sigma/sqrt(n))

🡪for large N, normalized variable (X’-mu)/(sigma/sqrt(n)), is almost normally distributed with mean =0 & variance=1

**IMP:: We can assume normality of a sample mean no matter what kind of population we have, as long as our sample size is large enough and our samples are independent.**

3.Confidence Intervals

->Note that for a 95% confidence interval we divide (100-95) by 2 (since we have both left and right tails) and add the result to 95 to compute the quantile we need.

->The quantity X' plus or minus 2 sigma/sqrt(n) is called a 95% interval for mu.

**i.e. est mean+c(-1,1)\*qnorm(.975)\*sqrt(est var)**

->For **binominal distribution**,95% confidence interval is:

p' +/- 2\*sqrt(p(1-p)/n),, where p’ is sample mean

and p is also unknown, so in order to maximize we givae p=1/2(maximize acc. To calculus)

which is deduced into: **p’+/- 1/sqrt(n)**

->Alternative to this approch we have **Wald confiddence interval,** replace p by p’

p'+/- qnorm(.975)\*sqrt(p'(1-p')/100),,qnorm is used to calculate more precise quantile value than 2(i.e.1.96)

->anoter alternative: **binom.test()**

This function"performs an exact test of a simple null hypothesis about the probability of success in a Bernoulli experiment." (This means it guarantees the coverages,uses a lot of computation and doesn't rely on the CLT.)

🡪Wald is not useful when N is small

Fix is using **Agresti/Coull interval**. This simply means we add 2 successes and 2 failures to the counts when calculating the proportion p'. Specifically, if X is the number of successes out of the 20 coin flips, then instead of setting p'=X/20, let p'=(X+2)/24. We use 24 as the number of trials since we've added 2 successes and 2 failures to the counts.

🡪**For Poisson distribution**

Est mean=x/t

To verify: **poisson.test()**

**4. T-Confidence Intervals**

-> Used for small sample size

-> We use t-statistic similar to z-statistic,

**t=(X'-mu)/(s/sqrt(n)),** where s is sample SD.

-> Formula used for t-confidence interval is similar to Confidence interval:

**Est +/- t-quantile \*std error(Est). T**

->instead of having two parameters, mean and variance, as the normal does, the t distribution has only one - the number of degrees of freedom (df)

* As df increases t distribution becomes more like normal.
* t interval is not good for skewed and discrete data(binary)
* to calculate:

**mean+c(-1,1)\*qtquantile(.975),n-1)\*s/sqrt(n)**

R function for same functionality:

**t.test(difference)$conf.int**