**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 andstandard 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 whencalculating 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**

->to calculate t-confidence for 2 different groups(g1 and g2):

M1—mean of 1st group, M2 –mean of 2nd group, N1—size of 1st group, N2—size of 2nd group

Mean(mn)<-M2-M1

Standard deviation(S)<- sqrt((N1\*var(g1)+ N2 \* var(g2))/(N1+N2-2))

Putting in formula:

**Mn+ c(-1,1)\*qt(.975,N1+N2-2)\*S\*sqrt(1/N1+1/N2)**

R function for the same:

**t.test(g2,g1,paired=FALSE,var.equal=TRUE)$conf**

**5. Hypothesis testing**

* making decision about population using observed data.
* Null Hypothesis: Baseling against which we test our hypothesis.
* Reject H\_0 to favour alternative hypothesis.
* Type I error: rejecting a TRUE null hypothesis.
* Type II error: accepting a false Null hypothesis.

Suppose our first alternative, H\_a, is that mu < mu\_0. We would reject H\_0 (and accept H\_a) when our observed sample mean is significantly less than mu\_0. That is, our test statistic

**(X'-mu) / (s/sqrt(n)) is less than Z\_alpha.**

Alpha is probability of Type I error, which is fixed to be 5%, so if we are not able to reject Null Hypothesis, we will say we “fail to reject” null hypothesis.

Beta is probability of Type II error, which is not fixed,

**Power=1-beta,** which represents the probability of rejecting a null hypothesis when it is false.

If you fail to reject the one sided test, you know that you will fail to reject the two sided.

Test static tells us : number of estimated errors between sample and hypthosized means

**6. P-Values**

The p-value is the probability under the null hypothesis of obtaining evidence as or more extreme than your test statistic (obtained from your observed data) in the direction of the alternative hypothesis.

**NOTE: Test-statistic** means **Observed Sample Results**

**WIKI:** In [frequentist inference](https://en.wikipedia.org/wiki/Frequentist_inference" \o "Frequentist inference), the *p*-value is widely used in [statistical hypothesis testing](https://en.wikipedia.org/wiki/Statistical_hypothesis_testing), specifically in [null hypothesis significance testing](https://en.wikipedia.org/wiki/Null_hypothesis_significance_testing). In this method, as part of [experimental design](https://en.wikipedia.org/wiki/Experimental_design), before performing the experiment, one first chooses a model (the [null hypothesis](https://en.wikipedia.org/wiki/Null_hypothesis)) and a threshold value for *p*, called the [significance level](https://en.wikipedia.org/wiki/Significance_level) of the test, traditionally 5% or 1% [[6]](https://en.wikipedia.org/wiki/P-value#cite_note-nature506-6) and denoted as *α*. If the *p*-value is less than or equal to the chosen significance level (*α*), the test suggests that the observed data is inconsistent with the [null hypothesis](https://en.wikipedia.org/wiki/Null_hypothesis), so the null hypothesis must be rejected. However, that does not prove that the tested hypothesis is true. When the *p*-value is calculated correctly, this test guarantees that the[Type I error rate](https://en.wikipedia.org/wiki/Type_I_error_rate) is at most *α*. For typical analysis, using the standard *α* = 0.05 cutoff, the null hypothesis is rejected when *p* < .05 and not rejected when *p* > .05. The *p*-value does not in itself support reasoning about the probabilities of hypotheses but is only a tool for deciding whether to reject the null hypothesis.

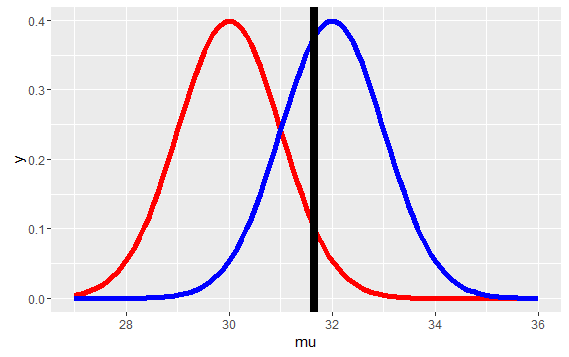
**NOTE:** if the p-value is less than the specified alpha you reject the null hypothesis and if it's greater you fail to reject.

**7. Power**

Probability of rejecting the null hypotheis when it is false.

Power give opportunity to detect if Alternative hypothesis is true.

Power=1-beta(TYPE II error, accepting a false null hypothesis.)



Red graph is H\_0 and Blue graph is H\_a.

Power is area under blue curve to the right of H\_a.

However, these hypothesis is considered to be normalized and have equal variance.

**H\_0, X'~ N(mu\_0, sigma^2/n) , H\_a, X'~ N(mu\_a, sigma^2/n**

If you will not consider it as normal, then taking consideration the red curve, its 95th percentile will be somewhere else.

Function to make this type of plot:

***function(mua) {***

***g = ggplot(data.frame(mu = c(27, 36)), aes(x = mu))***

***g = g + stat\_function(fun=dnorm, geom = "line",***

***args = list(mean = mu0, sd = sigma / sqrt(n)),***

***size = 2, col = "red")***

***g = g + stat\_function(fun=dnorm, geom = "line",***

***args = list(mean = mua, sd = sigma / sqrt(n)),***

***size = 2, col = "blue")***

***xitc = mu0 + qnorm(1 - alpha) \* sigma / sqrt(n)***

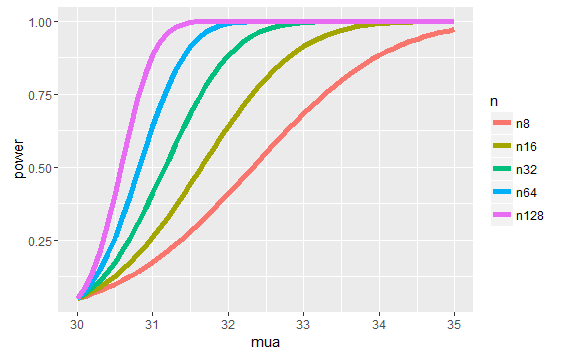
***g = g + geom\_vline(xintercept=xitc, size = 3)***

***print(g)***

***invisible()***

***}***

**Power is a function that depends on a specific value of an alternative mean, mu\_a, which is any value greater than mu\_0, the mean hypothesized by H\_0.**

****

This plot shows us that:

**Power gets bigger as mua gets bigger.**

**Power gets bigger as sample size incearses.**

Our test for determining rejection of H\_0 involved comparing a test statistic, namely

Z=(X'-30)/(sigma/sqrt(n)),

against some quantile, say Z\_95,which depended on our level size alpha (.05 in this case). H\_a proposed that mu > mu\_0, so we tested if Z>Z\_95.

This is equivalent to X' > Z\_95 \* (sigma/sqrt(n)) + 30

Calling pnorm with 30 + Z\_95 \* (sigma/sqrt(n)) as the quantile and mu\_a, say 32, as the mean and lower.tail=FALSE does:

**returns the area under the blue curve to the right of the line**

**Example:**

Z<-qnorm(.95)

Pnorm(30+z,30,lower.tail=FALSE): both distribution are same.

Pnorm(30+z,32,lower.tail=FALSE):Power(rejeceting mull hypothesis)—--sd=1

pnorm(30+z\*2,mean=32,sd=2,lower.tail = FALSE):Power decreases when sd increases(flatter distributions).

Power depends on:

->alpha (proportional)

->variance (inverse)

->alternative mean (proportional)

->sample size (proportional)

->When we use t-distribution,

**(Mu\_a-mu\_0)/sigma** –--Effect size : diff. in means in std deviation units.

We use R function for this, **power.t.test(n,delta,sd,type=”one.sample”,alt=”one.sided”)$power**

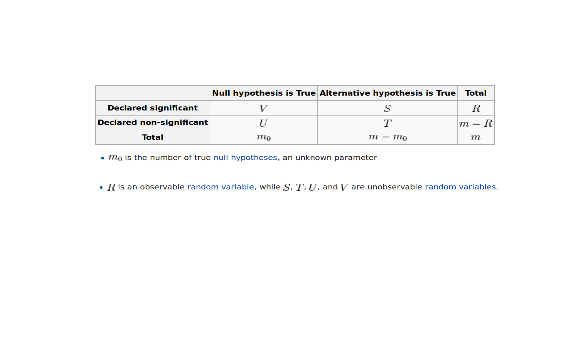
where delta/sd is effect size, which is constant for power

Other Variants:

**power.t.test(power, delta, sd, type = "one.sample", alt ="one.sided")$n**

**8. Multiple testing**

We need to perform multiple tests.

****

**E(V/R):**False discovery rate.

**E(V/m\_0):**False positive rate

**V :** Type I error rate

**T:** Type II error rate

**P(V>=1)**: probability of atleast one false positive(Type I error), should be minimium.

There are 2 methods for minimizing false positives:

BH(FDR) and Bonferooni(FWER)

sum(p.adjust(pValues,method="bonferroni") < 0.05)

sum(p.adjust(pValues,method="BH") < 0.05)

**9. Resampling**

**Bootstraping:**

Constructing confidence intervals and calculating std errors