# Data Collection

## Samples

* Part of a Population

## Parameter

* Factual information about **population**

## Satistics

* Can be computed to **estimate parameter**

## Accuracy

* How close is the **estimate statistic** to the actual **true parameter**

## **Bias**

* Factor that allows unequal variations of probability among **responses or samples** (intended or not)

### Selection Bias

* + When there is **underlying nature** in **determination** of **sample for study**

### Measurement bias

* + Recall bias
  + Sensitive bias
  + Misinterpretation bias
  + Wording bias

## Testing Methods

## Double blind test

* + Testing method where both **investigator & subject** go into test without knowledge of **what group anyone is in**

## Observational study

* + When investigator **observe** the effect of **risk factor/treatment** without trying to change or control **who is and isnt exposed to it**
    - Smoking vs non-smoking impacts

## Confounders

* **Dependant variable** that was not considered

### Method to control confounders

* + - Divide up the groups with respect of confoudner

## Simposons paradox

* When a clear trend in **individual groups** either **dissapear or reverse** when groups are pooled together
  + Clear sign of existance of confounder

# CHI-SQUARE

## General Structure of Report Layout

It is always **One sided test** where because we are looking for SIGNIFICANCE relationship

### H­0 Null hypothesis

* Conventional belief/ No statistical significance between factors

### H1 Alternative Hypothesis

* What you are trying to prove/ Existance of statistically significant relationship between factors

### Assumptions

* Facts about the population that is assumed to be true.

### Test statistics

* Numerical value that provide information about **General lnature of sample**
* Deviation away from expected parameters.

### P-Val

* Probability that we observe a **test statistic** that is **as or more extreme** than our observation

## Degree of Freedom

Number of freely determined cell

## Goodness of fit

Study to test one categorical factor and it’s frequency against eachother.

### Hypothesis

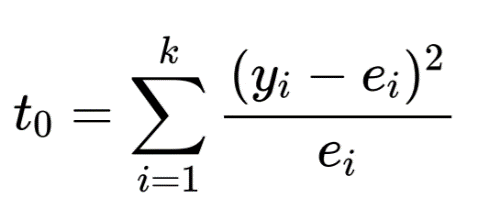
H0 : Frequencies acrross all categories in a factor is the same

H1: There is at least one group is not the same. There exists a where

### Assumptions

* Each observation is chosen at random from population and is independent from eachother
* Expected frequency per category is 5

### Test statistics

General equation that tries to test whether

Significance is present between **expected frequency vs observed frequency** per each **Factor Categories**

### P-Val

P( where k = number of categorie

## R Examples

Two methods in R

1 – pchisq(t0, df = (degree of freedom))

Chisq.test(y, p = c(probabilities))

# Poisson Distribution

Poisson random variable represents the probability of a given number of event occuring in a fixed interval **GIVEN** their independence and with known average rate **λ**

## Hypothesis

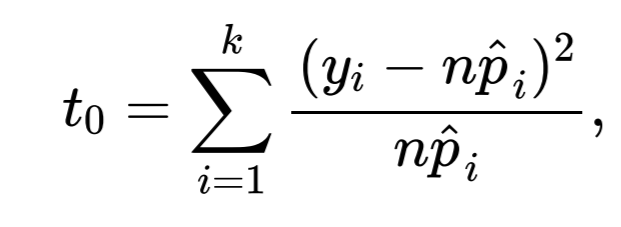
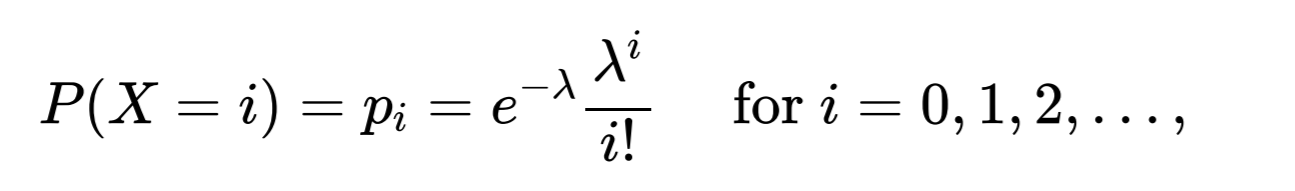
H0: Sample comes from a **poisson distribution**

H1: Sample does not come from **poisson distribution**

## Assumption

* Each observation is chosen at random from population and is independent from eachother
* Expected frequency per category is 5

## Test Statistics

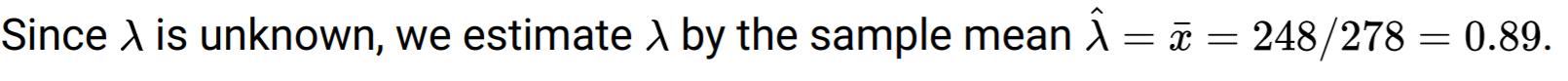
 given 

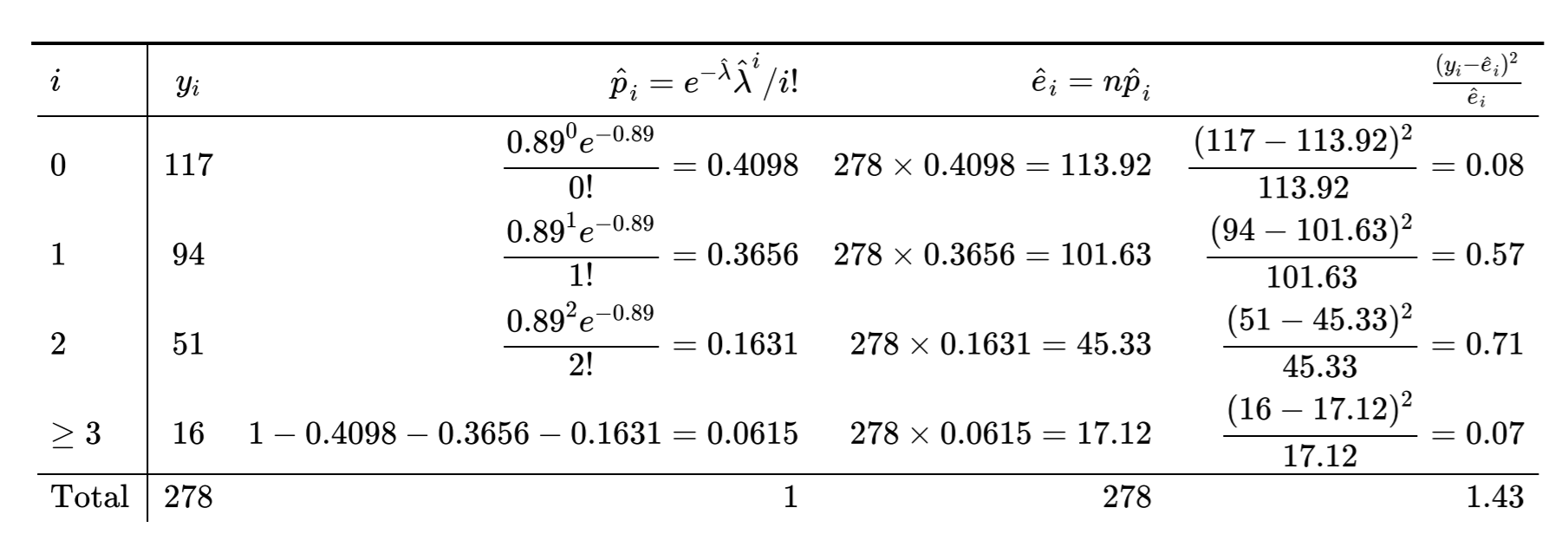
Where **λ = Sample mean**

## P-Val

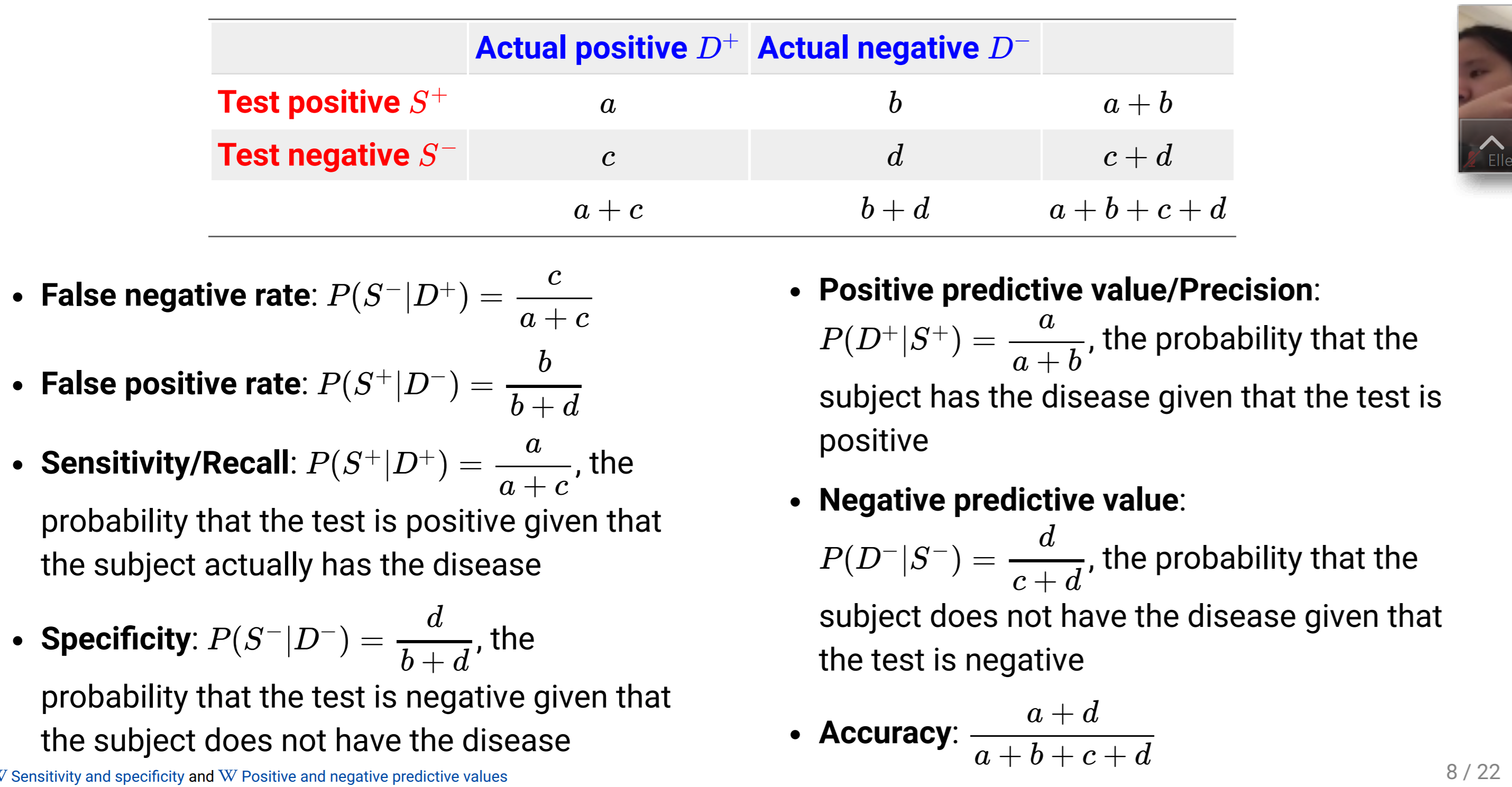
* Degree of freedom is k-2 because we made an assumption on a parameter **λ**

## Example





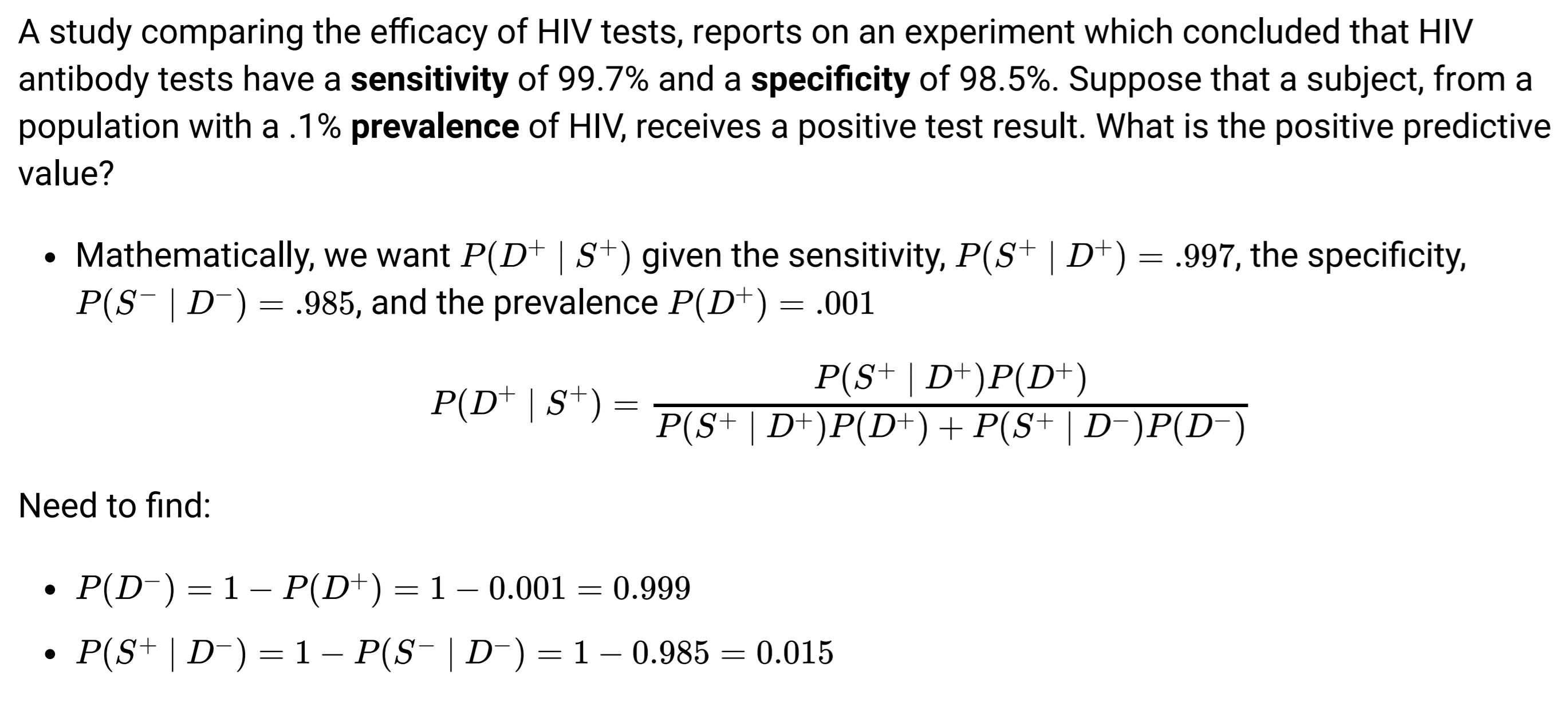
# Types of Errors



# Conditional Definition

Probability of A **GIVEN** event B Occurred

## Bayes Rule



# Measurement of Risk

## Prospective Study

* Stidies based on **subject** who are **initially identified disease free** and **classified by their exposure to risk factor**, they are then observed **through time** to see whether they **become diseased or not**
* Difficult to do because of ethics
* or

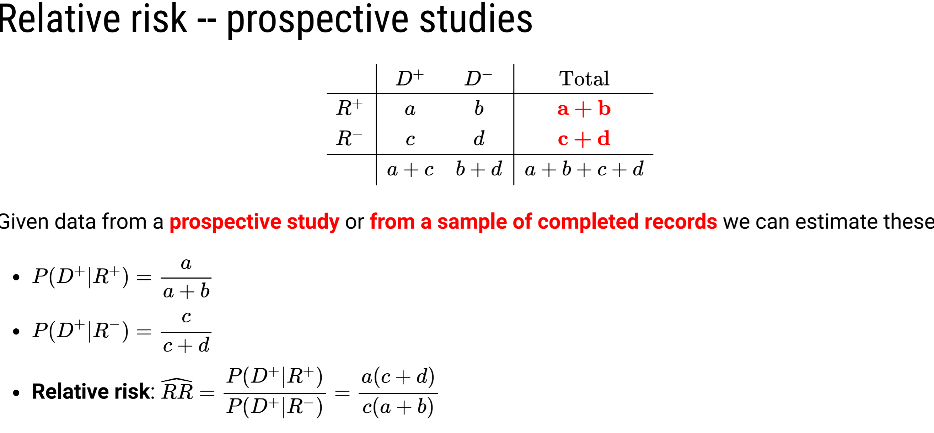
## Retrospective Study

* Study based ob **random sampling** from each of the **category groups** and through **interogative research** observe their exposure to **risk factor**
* or

## Relative Risk

* Ratio of Probability of **having the disease** **GIVEN** exposure to **risk factor** to probability of not
* Only applicable for **prospective studies**

If the Risk factor is independent from **presence of Disease** then RR should be 1.

If:

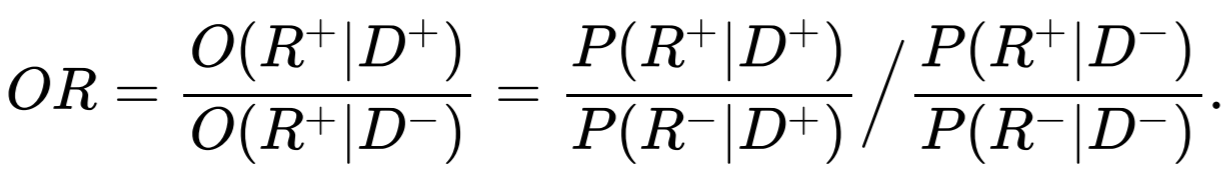
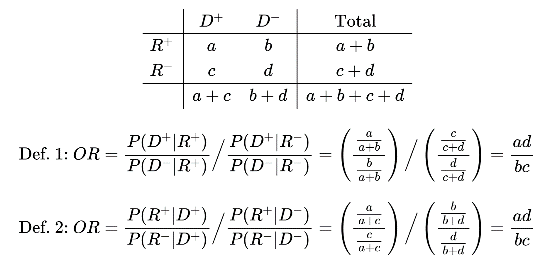
RR > 1: Disease is **more** likely from group **with risk factor**

RR = 1: Disease and Risk factor is independent

RR < 1: Disease is **less** likely from group **with risk factor**

## Odds

* Ratio of Diseased/Non-diseased **given** the presence of Risk factor.
* Applicable for both **prospective and retrospective**



OR > 1: Disease is **more** likely from group **with risk factor**

OR = 1: Disease and Risk factor is independent

OR < 1: Disease is **less** likely from group **with risk factor**

## Confidence interval and Standard Error

SE(log(OR)) =

Log(OR) ± 1.96 \* SE(log(OR))

# Testing for Homogeneity

## Hypothesis

H0: All the categories’s probability is the same, P11 = P21 & P12 = P22

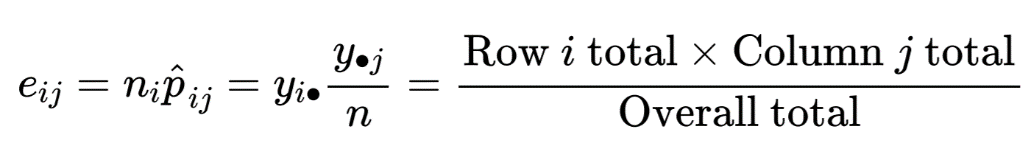
Probability distribution of Categories are the same over **Two different population**

H1: At least one categorical relationship is significantly different, P11 ≠ P21 & P­12 ≠ P22

## Assumptions

* Each observation is chosen at random from population and is independent from eachother
* Expected frequency per category is 5

## Test Statistics



## P-Val

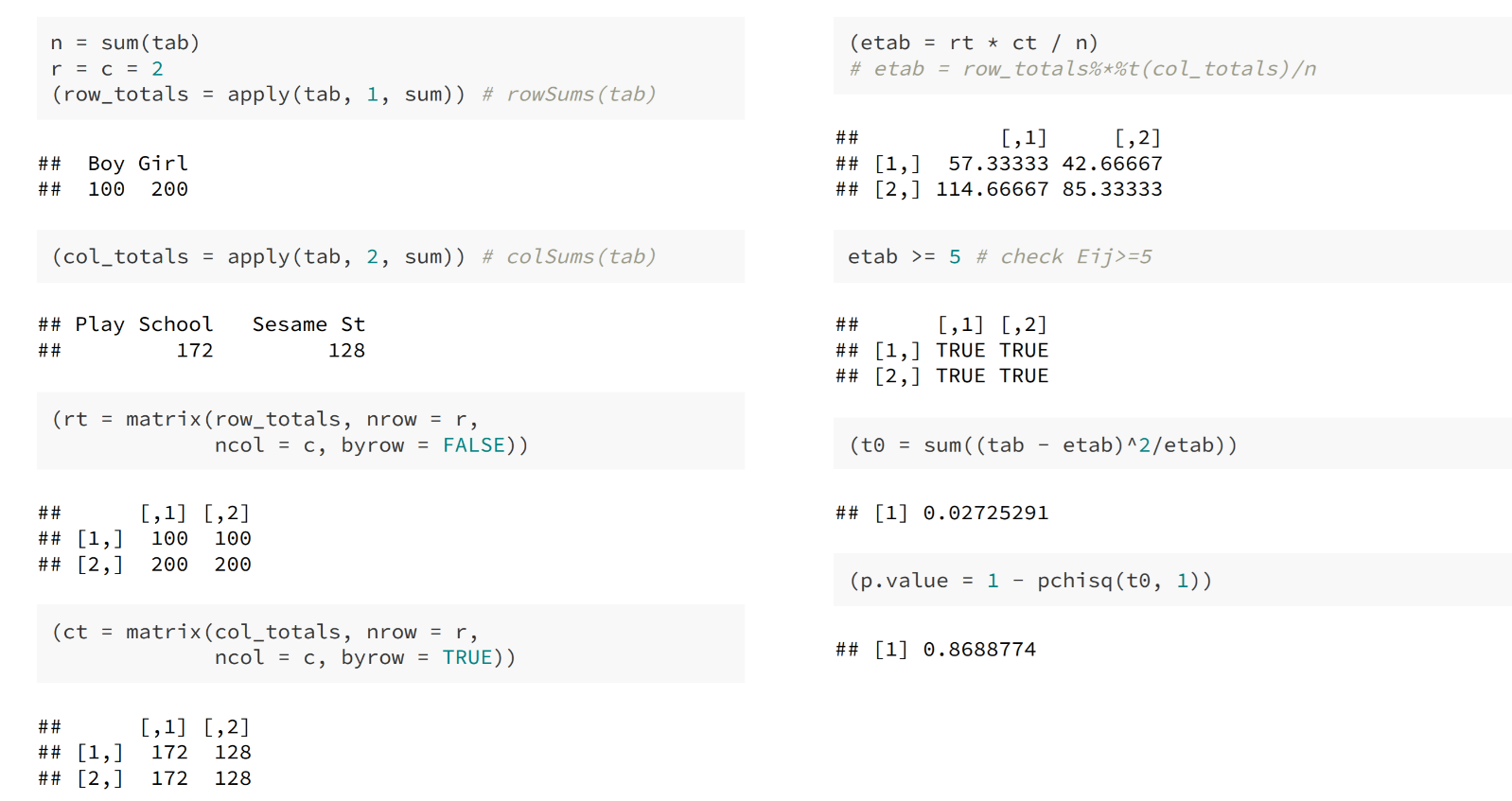
where r,c = number of category per factor

## R Code Example

**Built in R function**

Tab = table(category1, category2)

Chisq.test(tab)

**Manual** **Method**

# Test for Independence

## Hypothesis

H­0: Frequency for all the categories are the same Pij = PiPj, for I = j = 1,2,3,4….

Probability distribution of Categories are the same over **different Categories**

H1: Not all Equality hold

Test for Independence and Homogeneity is essentially the same with same processes. **HOWEVER** Differs by the main focus of sample

**Test for homogeneity**

* Analaysis onto whether **One categorical variable** holds **same distribution** over **Two different populations**
  + Sample is **randomly selected** from **two different population**

**Test for independence**

* Analysis onto the underlying nature of the **dependence relationiship** between **two variables OF THE SAME POPULATION**
  + Sample is selected **from one population**

# Tests in a small sample size

## Fisher’s exact test

* Use of **Hypergrometric distribution** to calculate the probability of getting **same or more extreme observation** than actual observation **given same margin total**
* Uses permutation to see the probability of getting the same or more extreme number **WITH TOTAL SUM FIXED**

### R Code Example

Tab = table(factor1, factor2)

Fisher.test(tab, alternative = ‘’)

* Uses Odds ratio to calculate the P-Value, therefore, alternative would be in respect of OR

### Draw Back of Fisher’s exact

* Assumes rows and column’s margins are fixed
* Computationally stressful

## Chi-square test with Yates correction

* Aims to reduce error made by **Overestimation of statistical significance** in a **small data analysis**
* Tends to **overcorrect** resulting in a **overly conservative conclusion** that fails to reject Null hypothesis
  + Reduces **Power**

Chisq.test(data, correct = TRUE)

## Permutation Testing in Categorical setting, Monte Carlo Simulation

When we sample from the sample to find the **proportion** of the **resampled Test Statistic** that is **same or more extreme** than the one calculated from **original sample**

Chisq.test(data, simulate.p.value = TRUE, B = (# of tests))

# Testing mean, numerical analysis & t-test

Main aim of the t-test is to validate whether there is a **significant differences amongst the mean value of sample(s)**

## Normality Characteristics

* Sample mean from a normal sample is in itself **normally distributed**
* **Sample variance** from a Normal sample has a **scaled x2 distribution**
* **Sample mean** and **sample variance** from a normal sample are **statistically independent**

Combination of all these gives

## Normality Assumptions

t-test has an underlying assumption that the Dataset/population is Normally distributed.

* Box plot
* QQ plot

However, this is also very difficult to satisfy if there is an **outlier**

## T-Test Test Statistics

|  |  |
| --- | --- |
| = Sample Mean | =Theoretical mean |
| = Standard Deviation | = Population Standard  Deviation |

## One Sample T-Test

H0: Sample mean is equal to Given Pupulation mean,

H1: Sample mean is not equal to Given Population mean, ,

## Assumptions

* Each Sample is independent and identically Distributed (*iid)*
* Population Follows a normal distribution around the population mean

## P-Val

|  |  |  |  |
| --- | --- | --- | --- |
| When H1: | P(T ≥t0) | t.test(x,mu = ‘’, alternative = “greater”) | pt(t0, n-1) |
| When H1: | P(T ≤ t0) | t.test(x,mu = ‘’, alternative = “less”) | 1-pt(t0, n-1) |
| When H1: | 2P(T ≥ |t0|) | t.test(x,mu = ‘’, alternative = “two.sided) (default) | 2pt(t0, n-1) |

## Paired Two sample T-Test

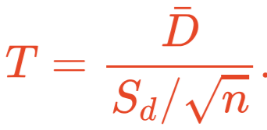
Paird T-test is the same as normal single sample t-test but in terms of differences

### Assumptions

H0: There is no significant differences between two Samples, = 0;

H1: There is a significant differences between two samples, ≠ 0;

### Test Statistics

 This comes from the fact that

## Two sample T-Test

### Hypothesis

H0: Two sample’s mean is identical,

H1: Two sample’s mean is different

### Assumptions

* Independent and Identically Distributed *iid*
* Population is Normally distributed around the mean
* Xi is independent from Yi
* Equal variance between two groups

## Test Statistics

## P-Val

|  |  |  |
| --- | --- | --- |
| When H1: | P(Tnx – ny - 2 ≥t0) | t.test(x,y, alternative = “greater”, var.equal = TRUE) |
| When H1: | P(Tnx – ny - 2 ≤t0) | t.test(x,y, alternative = “less” , var.equal = TRUE) |
| When H1: | P(Tnx – ny - 2 ≥|t0|) | t.test(x,y, alternative = “two.sided” , var.equal = TRUE) (default) |

## Equal Variance

Tested using **Box plot** and comparing the **Standard Deviation** of both samples

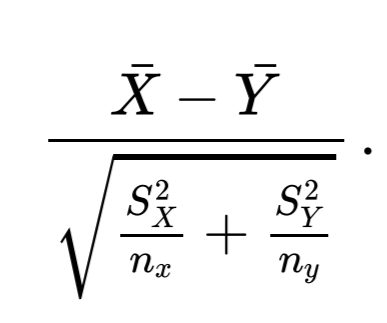
## Welch Two Sample T-Test

When Equal Variance is Not possible

## Assumption

* The sample is normally distributed
* Two sample is independent
* Indepentent and identically distributed

## Test Statistics



Therefore

|  |  |  |
| --- | --- | --- |
| When H1: | P(Tnx – ny - 2 ≥t0) | t.test(x,y, alternative = “greater”) |
| When H1: | P(Tnx – ny - 2 ≤t0) | t.test(x,y, alternative = “less”) |
| When H1: | P(Tnx – ny - 2 ≥|t0|) | t.test(x,y, alternative = “two.sided) (default) |

# Critical value

Critial value can be seen as a value that located at the α percentile of the

normal graph.

**Critical value** is a point of the test distribution that is compared to the

test statistics to determin whether to reject the null hypothesis or not

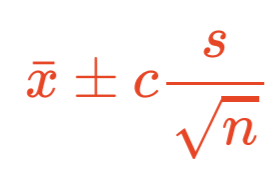
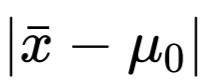
It can be calculated by doing

**C = qt(α, (degree of freedom))**

If two sided, α needs to be

Therefore, if α = 0.05, then either c = qt(0.025,df) or qt(0.975,df)

# T-Test

Delcare H0 to be rejected if,  >

# Confidence Interval

### Standard Error

When trying to find the confidence interval, we use Standard Error to find the variation in expected mean range

# Coverage Probability

* Probability that the true value of the **unknown parameter** lies **inside the** confidence interval
  + **Coverage probability =** 1- α

# Rejection Region

|  |  |
| --- | --- |
| When H1: | Tn-1(α) = qt(1-α, n-1) |
| When H1: | Tn-1(α) = qt(α, n-1) |
| When H1: | Tn-1(α/2) = qt(1- |

## Power

|  |  |  |
| --- | --- | --- |
|  | H0 True | H0 False |
| Don’t Reject H0 | Correct Decision | Type 2 Error (β) |
| Reject H­0 | Type 1 Error (α) | Correct Decision |

Power is the ability to

“Reject H0 when H­0 is false”

Power = 1-β

## Factors that Effect Power

### **Sample Size (n)**

As **Sample Size (n) increase**, Power **increases**

### **Significance level (α)**

As **Significance level (α)**  **increase**, Critical Value **decrease**, and Power **Increases**

### **Sample Variance ()**

As **Sample Variance increase (),** Confidence interval & Confidence value **decrease**, and power **increase**

### **Cohen’s d, (**)/

As **Cohen’s d decrease,** power **increase**

## Pwr package

Pwr.t.test(n = , d = , sig.level = , power = ,

type = {“two.sample”,”one.sample”,”paired”},

alternative = {“two.sided”, “less”,”greater”})

if you leave one of the parameter as NULL, it will provide the corresponding parameter’s expected values

* Using this, you can find out the expected value for

# Sign Test

Goes around the t-test restriction of Normality by getting rid scaling factor.

This is only applicable for **paired t-test** or **single sample t-test**

Diff = xi – xy ­or xi -

## Hypothesis

H0: number of positive signed differences = number of negative signed differences, P+ =

H1: number of positive signed differences ≠ number of negative signed differences, P+ ≠

## Assumption

* The Differences is equally likely to be **negative** as it is **positive**
* No ties are considered

## Test Statistics

**t0 = #P+ =** number of positive differences

## P-Val

|  |  |
| --- | --- |
| When H1: | P(Tt0) |
| When H1: | P(Tt0) |
| When H1: & t0 > | 2P(Tt0) |
| When H1: & t0 < | 2P(Tt0) |

## R Code Example

binom.test(c(P+,P-), p = 0.5, alternative = “greater”)

**alternative** is refering to whether there is :

* More positive value than negative value

## Advantages

* Non-Parametric
* Robust

## Disadvantages

* Ignores a lot of information
* Less powerful than t-test

# Wilcoxon signed-rank test

The aim of Wilcoxon sign rank test is to eliminate the loss of information (scaling factor) that the sign test has.

## Theory

If both positive and negative differences between X and Y are the same, then we should also see the mean rank differences to be near to symmetry

* This is beause of the fact that if magnitude of X and Y are the same, then they would also have same sum of ranks

## Assumptions

* Data is paird and comes from the same Population
* Each pair is chosen randomly and independently
* Sample is collected from symetric distribution around diff = 0;

## Test statistics

W+ = for **one sided**

W = min(w+, w-) for **two sided**

## P-val

|  |  |  |
| --- | --- | --- |
| H1 = | P(W+ ≥ w+) | Psignrank(W+ -1 ,n, lower.tail = False) |
| H1 = | P(W+ ≤ w+) | Psignrank(W+ ,n) |
| H1 = | 2P(W+ ≤ w+) | 2\*Psignrank(W,n) |

Or you can do

**wilcox.test(d, alternative = “greater”|”less”|”two.sided”)**

**wilcox.test(x,y,alternative = “greater”|”less”|”two.sided”, paired = TRUE)**

## Normal approximation of Wilcoxon.test

* For a large number of n, we can assume **normal distribution**

## Test Statistics

**Without Ties**

**With ties**

|  |  |  |
| --- | --- | --- |
| H1 = | P(Z ≥ t0) | pnrom(t­0, lower.tail = FALSE) |
| H1 = | P(Z ≤ t0) | 1 - pnorm(t­0) |
| H1 = | 2P(Z ≥ |t0|) or 2P(Z ≤ |t0|) depending on t0 | pnorm(t0{,lower.tail = FALSE}) |

If a tie is present, R will automatically calculate it and run wilcox.test accordingly.

# Wilcoxon rank-sum test

This study is aimed to conuct **TWO SAMPLE T-TEST** without normality or symmetry assumption

## Methodology

The dependant factor is concatenated into one single variable and ranked over the whole set

We are focusing on the **sum of ranks** for **one of the sample** and their **proportional t0**

## Theory

If H0: no differences between the two samples then:

E(W) = Proportion X Total Rank Sum = =

Respectively, if H1: ( , E(Wx) small (large) then we should see,

## Assumptions

* X and Y are independent
* Follow the same distribution but differ by shift

## Test Statistics

w = r1 + r2+…+ ~ WRS(nx,xy) distribution

## P-Value

|  |  |  |  |
| --- | --- | --- | --- |
| H1 = | | P(W≥w) | pwilcox(w-minw-1, m = nx, n = ny, lower.tail = FALSE) |
| H1 = | | P(W≤w) | pwilcox(w-minw, m = nx, n = ny) |
| H1 = | >  **w > E(w)** | 2P(W≥w) | 2\*pwilcox(w-minw-1, m = nx, n = ny, lower.tail = FALSE) |
| H1 = | <  **w < E(w)** | 2P(W≤w) | 2\* pwilcox(w-minw, m = nx, n = ny) |

# Wilcoxon Rank sum test **with ties**

## Test Statistics

t0 =

|  |  |  |
| --- | --- | --- |
| H1 = | P(Z ≥ t0) | pnrom(t­0, lower.tail = FALSE) |
| H1 = | P(Z ≤ t0) | 1 - pnorm(t­0) |
| H1 = | 2P(Z ≥ |t0|) or 2P(Z ≤ |t0|) depending on t0 | pnorm(t0{,lower.tail = FALSE}) |

## R-Code

**wilcox.test(A, B , alternative = “greater”|”less”|”two.sided”)**

R will automatically know whether there is a tie or not and react accordingly.

# Permutation testing

process of **sampling from a sample without replacement** and see the probability of having test statistics that is **same or more extreme** than observed value.

In a small samlpe size, this will allow **greater power** as it allows **greater number of test size (n)**.

What is unique about permutation testing is its **sampling without replacement**. Because of the use of ALL THE SAMPLESwe can use this to perform **statistical testing** and **extract p-val**

## Assumptions

**Exchangability**

* Swapping data points keeps the data just as likely as the original
* Assumption that depends on numerical assignment onto variables to have equal probability

## Test-Statistics

We can use

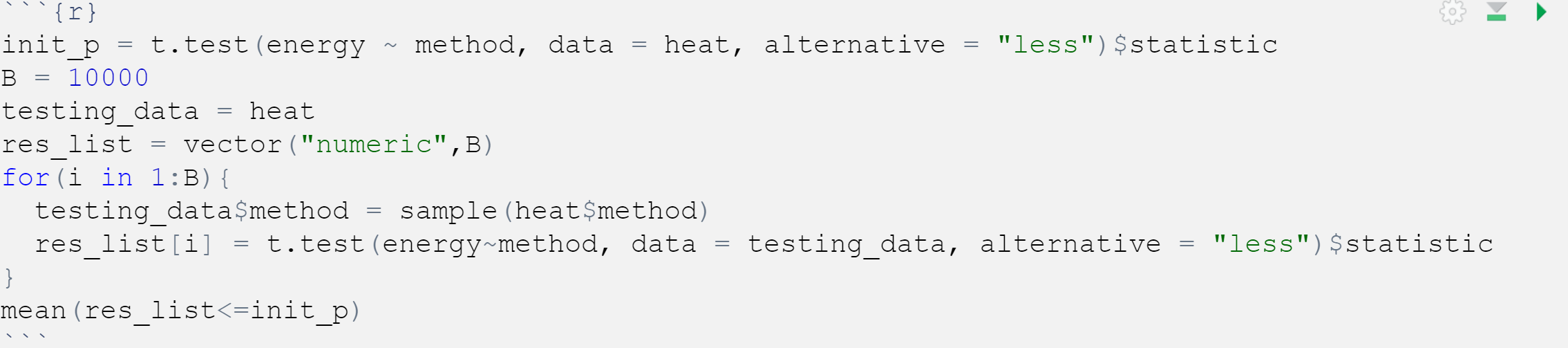
* T-test
* Wilcoxon sign-rank
* Wilcoxon rank-sum
* Rebustly stadardise the median differences
  + where MAD() is the Median Absolute Deviation

To find test statistics for the original set

* Although we are not strictly comparing the result with distributions, **t-test test statistics** are **parametric** by nature, therefore, we need to do **rank test** if there is a **distinct outlier**

Then we perform **sample from sample** and store the **test statisstics** per samples.

Then we find the observations from collected samples that is **same or more extreme** than initially calculated test statistics



|  |  |
| --- | --- |
| Alternative = | Inequality in mean() |
| Greater | >= |
| Less | <= |
| Two.sided | Abs(test\_list)>=origin\_p\_val |

Alternative does not need to be stated in here because of the fact that alternative is only relevant when **finding p value** but we are only concentrating on **test statistics alone**

# Bootstrap

## Estimate vs Hypothesis testing

Estimate (bootstraping)

* Aims to estimate the population parameter using sample statistics

Hypothesis tesing

* Test statistics are generated to either support or reject **null hypothesisb**

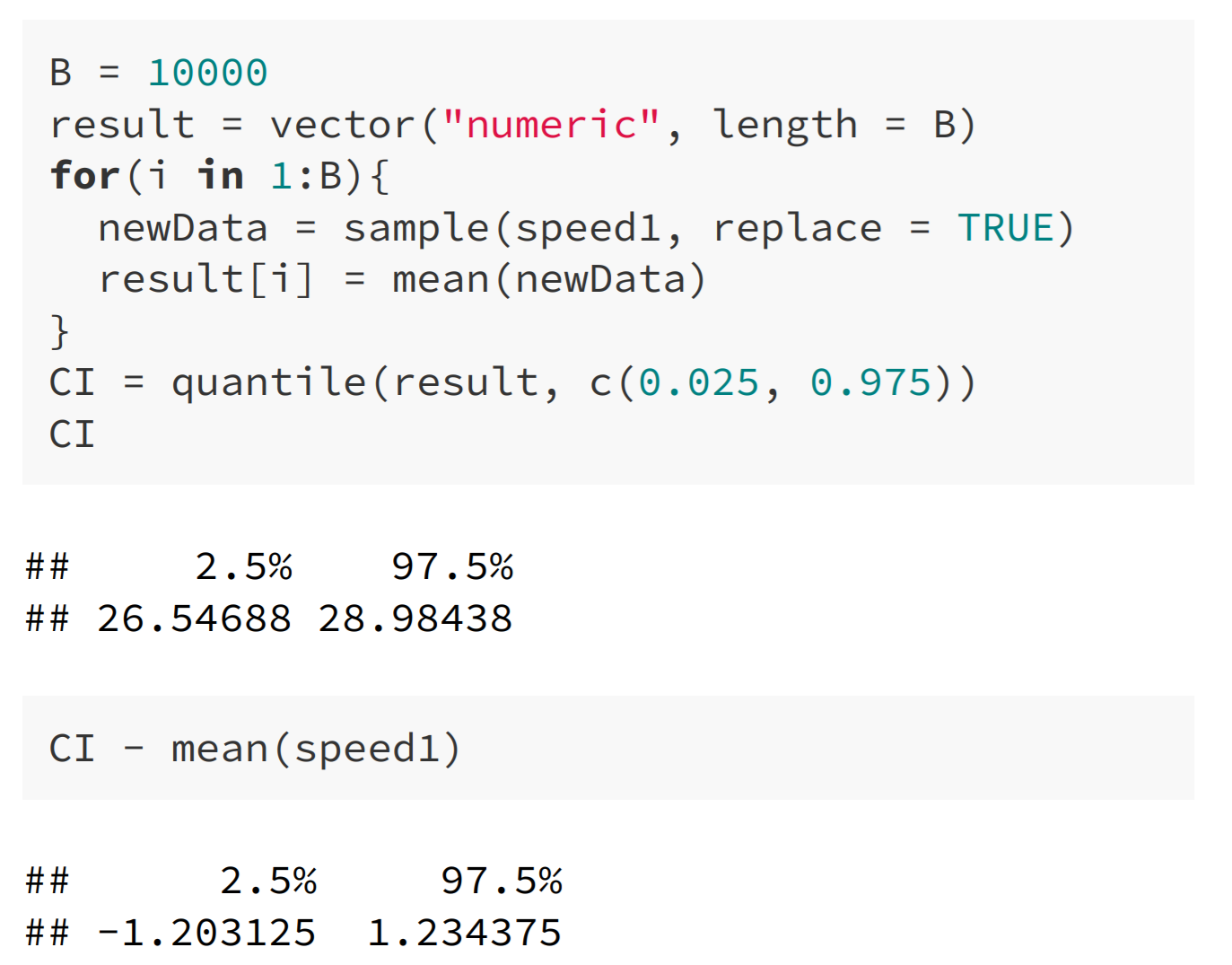
Bootstrap aims to find **confidence intervals** or generation of **large sampel standard error**

## The meaning of Confidence interval

**Confidence interval** does NOTmean **porbability of population mean** being between **(a,b)**

It does mean that if we **draw large number of random samples,** and compute **confidence interval** from those samples, 95% of CI will have **population mean**

**Confidence ≠ probability**



This system is done to calculate the population parameter due to the use of **sampling with repetition**

## Bootstrap’s advantage

* Useful when we don’t know the **theroetical distribution of a statistic**
* Sample size is **too small** to make any **sensible parametric inference**
* Minimal **parametric assumption**
* Can be used to **verify & check the stability of result**

# Types of errors when doing multiple testing

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | | Trueth | | #Test |
| H0 = True | H0 = False |
| Conclusion | H­0 = true | U | T | m-R |
| H0 = false | V | S | R |
| #Test | | m0 | m-m0 | m |

**False positive rate E()**

The rate at which **null results are significant** but was not considedred **significant**

**Family wise error rate** **E(V≥1)**

The probability of at least one **false positive**

**Family Discovery Rate E**

Rate at which **claims of significance is false**

## Controlling Family wise error rate

FWER = probability of at least one false positive

If we conducted ***m***number of tests simultaneously, then we are expected to get

**False positive** = 1-(1-α)m

Meaing, if we do 20 tests with α = 0.05, then we have 1-(1-0.05)20, we have 64% probability of getting **at least one false positive result**

### Bonferroni correction

This is to accommodate the number of simultaneous tests and to reduce FMER accordingly

This means new

It is simple to calculate but may be **more concervative than desired**

## Controlling False discory rate

To keep the expected proportion of **false positive** in your **rejected test** close to

### Benjamin – Hochberg Procedure

1. Conduct m number of tests
2. Calculate p value normally
3. Order results in ascending order of p value
4. Find J\* = max j such that
5. Reject all H0 that has p value less than p(J\*)

This is pretty simple to calculate but this allows for more false positives

# Simple ANOVA tests

## Analysis of variance

* Two sided **two sample** t-test done over multiple factor variable values.

Out of all the types of t-test

* Paired t-test
* Welch t-test
* 2 sample t-test

2 sample t-test is used as a base test to maximise the power, so we should always check for

* Normality
* Observations are Individual and identically distributed within each groups
* Groups are independent of eachother
* Equal variance

## Hypothesis

H0: All Mean are equal

H1: There is at least one pair that is not equal to the rest

## Assumptions

* Observations are independent across different **g samples**
* Each **g population** has the same **variance**
* Each **g population** are normally distributed

## Test statistics

~ Fg-1, N-g

### Total Sum of Squares

= Sample Variance + Sample Mean

= residual SS + Treatment SS

Weighted average across all the datasets, this can be used to find the total variance to show significance between sample means

### 

### Residual Sum of Sqares and Mean Square

By Dividing Residual SS by (N-g), we get **unbiased estimator of**

Comparison between **each value** with **associated group mean** to allow **unbiased values**

### Treatment Sum of Sqares and Mean Squares

The comparison between **group mean** with **overall mean** would result in the **group mean** **inconsistency** to be **visible**

## Anova Test Statistics and P-val

Anova’s P-value is always **one sided** because of the fact that, any **variance of group** means will result in **TMS to** **increase while RMS is consistent**, resulting in test statistics to increase and never decrease in any cercumstances.

## R-Code

**aov(dependant variable ~ factor variable, Data = “”)**

**summary(aov\_obj)**

## Anova Contrasts

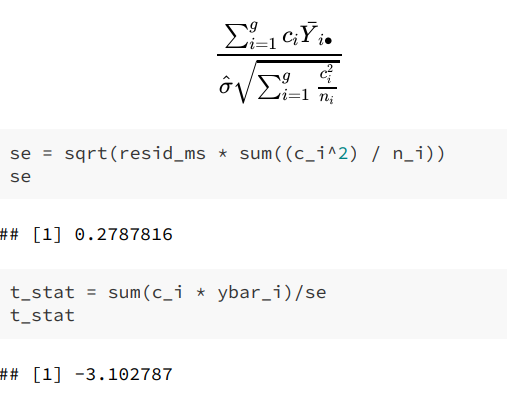
Constrasts are a **linear combination** where **coefficients add to zero,** In anova, it is the **Linear combination of means**

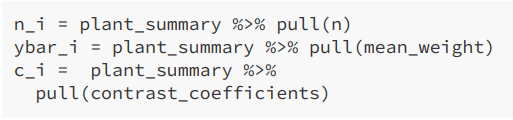
ANOVA puts a contrasts on the **factor variable** to determin whether there is a **group mean variance or not**.

However, in **null hypothesis**, we assume that **all means are the same**, then

However, this only finds the presence of **group mean** variance **across the table** and not **specific relationships**

We can select which variables to use by specifying the **contrast coefficience**

Where c = c(1,-1,0), depending on which variables to do the testings on.

This holds a better estimation of therefore it is a better option than normal t-test.

## Confidence Interval for ANOVA

Where t = qt(1-,df = (n-g)) , quantile against normal distribution

# ANOVA Post hoc tests

Post hoc testing is when you do further analysis **after** initial analysis of the data.

Normal Assumptions can be checked individually, but we can sum it all together as and test the “residual” in terms of their satisfaction to Normality assumption:

If we find that all groups are Interesting in ANOVA context, we have two choices

* T statistic constrution for each pairwise differences
* T – based confidence interval for each pairwise “population” differences

## Practical application for t based confidence interval with ANOVA

Because we are concerned with pairwise, we only put 1 as value of **contract coefficience** for those interested

Where I and h is the two groups in focus, t = quantiles.

## Emmeans package

Package called emmeans will construct this for us so that we don’t need to manually calculate all the nP2 relationships

**Emmeans(aov\_obj, factor variable)**

|  |  |  |
| --- | --- | --- |
|  | **Individual** | **Pairwise** |
| **Confidence interval** | **Confint(emmeans\_obj, adjust = “”)** | **Confint(pairs(emmeans\_obj,adjust=””)** |
| **t-test** | **test(emmeans\_obj, adjust = “”)** | **test (pairs(emmeans\_obj,adjust=””)** |

if the confidence interval does not contail 0, it means they are “interesting”

## Adjustment Methods

### Bonferroni

* + Most suitable for small number of groups
  + It will become very conservative very fast due to the quadratic growth (g^2)
  + As g increase
    - Confidence interval will also icnrease
    - Power will decrease

## Tukey’s

* + Very useful when sample size is equal
  + Also advantageous when there is a lot of groups

## Scheffe

* + Consider all the possible combination to derrive correction
  + Useful when doing data snoopings.
    - Result of refining parameters to fit a certain trend shown in a **small section of a data**
  + More conservative

# Assumption modification with ANOVA

There are two main assumptions from ANOVA

* Equal Variance
* Normality

## Violation of **equal variance** in ANOVA

* When there are outliars within the datasets
* **Normality assumption satisfied** but **common variance is not**
* **Welch t test** 
  + We can conduct **welch t tet across all pairwie** and apply **bonferroni correction**

**T test**

Pairwise.t.test(dependent\_var, factor\_Var,p.adjust.method = “bonfferoni”, pool.sd = FALSE)

**Confidence Interval**

Pairwise.t.test(dependent\_var, factor\_Var,p.adjust.method = “bonfferoni”, pool.sd = FALSE)$conf.int

## Violation of **normality** in ANOVA

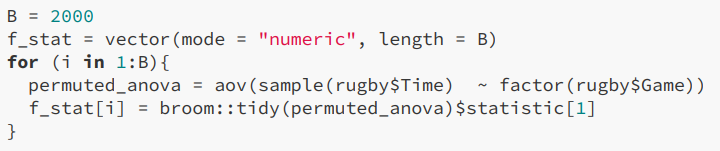
### Permutation Tests

#### Anova with F-Statistics

* + Under null hypothesis of

“No differences between groups” = “all possible allocations are equally likely”

* + Randomly allocate the **opjbservation vector** keeping the **factor vector fixed**
    - Pbserved proportion of the times the **statistics exceeded t0** becomes an bestimate fo the exact value



* + In here we can see a sampling of “observed”/ dependent variable while keeping the factor variable static.
  + This is to control the “no differences **between group**” section of the test

#### Using Kruskal-wallis test with binomial statistics

* + - Assumption
      * Same as anova for most
      * Different groups follow same distribution shifted by location

Kruskal.test(observed\_var ~ factor\_var, data = “”)

# Two way ANOVA

Conducting ANOVA with two factor variables

**Total Sum of Squares**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Source of variation | Sum of squares | Df | Mean Square | F-Ratio |
| Blocks | Block Sum Sq | n-1 | - | - |
| Treatments | Treatement sum sq | g-1 | Trt MS = |  |
| Residual | Residual Sum Sq | (n-1)(g-1) | Res Ms = | - |
| Total | Total Sum sq | ng-1 | - | - |

We are not interested in the **effect of Blocking variable**, however we are using it to further define the **treatment level in each pair**

**by adding another determinant factor variable**

* Residual’s degree of freedom will decrease
* Residual sum of Square and Mean Square will decrease
* Confidence interval will increase
* P-Value will decrease

AOV(observation variable ~ factor\_variable, blocking variable)

## Friedman test

* Friedman’s test is a two way rank test for Kruskal wallis test

Friedman.test(observational variable ~ factor variable | blocking variable)

# Two way ANOVA with interactions

Interactions

* The secondary variable **does have an imapact** in which we want to also focus our study into

## Aim

* To see whether we can study the **two factor variables** independently or if there exists an internal interaction that are present between variables that **needs to be considered when conducting studies**

## Difference between Blocking and Interactions

#### Blocking

* + We do not consider the interactio nbetween factor variable and blocking variable.
  + We are only concerned to adjust the Factor variables

#### Two way with Interaction

* + Each mean is affected by **both factor variables** and we are trying to find the presences of **interaction that impacts individual means per group**

## Hypothesis

* H0: There is no interaction between variable,
* H1: There exists an interaction between Factor variables

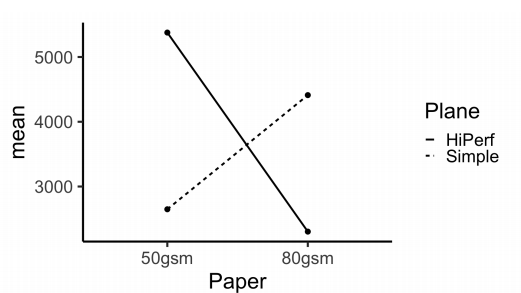
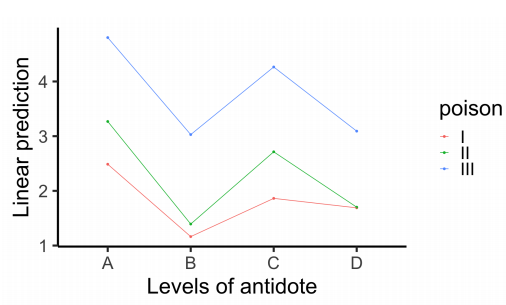
## F-Statistics

## R Code Example

## Interaction Plot

* Graphically examin the interaction using interaction plot
* If interaction is present, there will be **crossings** else **just parallel lines**

With interaction Without interaction



## Post hoc analysis

* When we do post hoc analysis after interaction checking we need to adjust the p-value according to both variables.
* We create two separate Emmean objects and combine them using update()

**Update(emmean\_obj\_1 + emmean\_obj\_2)**

This will fix the adjustment values

# Regression model

* Linear Regression model
  + Linear equation aimed to predict outcome given **predictor variable values**
  + and which are both calculated **to minimise regressions from observations to line.**

**lm(dependent\_variable ~ predictor/factor variable)**

## Assumptions

* Linearity
  + Conduct line-graph with **geom\_smooth** to see whether each point has **linear trend**
    - If not, we can conduct transformations accordingly.
* Independence
  + Independence between **each observation’s errors/residual**
  + Usually controlled in **experimental design**
* Homoskedasticity
  + Construct **scatter plot** to check whethere there is **equal separation of points in the value**
  + This does not affect the prediction, but it deos effect the p-value and test statistics
* Normality
  + QQ plot and Box plot
  + We can also rely on  **central limit theorem** if we have enough samples

## Inference

* When we use the model to test whether **the variable has significance in linear relationship of data**
* If = 0, it means that factor variable x is irrelevant in determination of Yi

We reject if this includes 0

## Coefficient of Determination r2

Measures the proportion of total variabnles in Y explained by the linear regression model

If r2 = 66%, it means the linear regression model describes 66% of observed data variations.

# Multiple Regression and Model selection

When you want to create a regression model using **multiple predictor variable**

## Log transformations

Log linear:

* On average, a one unit increase in x will result in change in Y

Linear Log:

* On average, one percent increase in x will result in changes in Y

Log Log:

* On average, a one percent increase in x will result in changes in Y

lm(dependent variable ~ predictor1 + predictor 2+ predictor 3…)

## Model Interpretation

* **One degree increase** in **temperature** will result in **5% increase in ozon**
* **One increase in solar radiation** result in **0.3% increase in ozon**
* **One m/h increase in wind speed** will result in **6% decrease in ozon**

## Model Selection

### Backward Variable selection

1. Start with model cnotaining **all possible explanatory variable**
2. For each variable in turn, **investigate effect of removing variable from current model**
3. **Remove the least informative variable**, unless this variabel is **supplying enough information about the responces**
4. Repeat untill all are significant

Step(full\_model, direction = “backward”, trace = FALSE)

Round(summary(step\_result)$coef, 3)

### Forward Variable selection

1. Start with model containing **no explanatory variable**
2. For each variable in turn, **investigate effect of adding vartiable from current model**
3. **Add most informative/significant variable,** unless it is not supplying significant information
4. Repeat untill **all significant variables are selected**

Step(null\_model , scope = list(lower = null\_model, upper = null\_model), direction = “foward”, trace = FALSE)

Round(summary(step\_result)$coef, 3)

## AIC Akaike information criterion

* Assessment of the performance of the Model Selection
* The lower the AIC the better the model selection
* If AIC is the same, the model that differs by 1~2 AIC is considered simmilar in model fitting

# Prediction

There is two ways of going about performing a prediction

* When you want to **predict a specific value** given a condition
  + Prediction interval
* When you want to **get average value** given a condition
  + Confidence interval

**Predictor interval** is greater than **confidence interval** by

## Prediction

1. You need to set your predictor parameter
2. Give it to predict() along with that type of prediction you want to get

new\_obs = data.frame(radiation = 200, temperature = 90, wind = 15)

**prediction interval**

predict(Linear\_model, new\_obs, interval = “prediction”, level = 0.90)

**confidence interval**

predict(Linear\_model, new\_obs, interval = “confidence”, level = 0.9)

## Effect of variance on intervals

* Smallerpopulation variance , the **better the fit** and in terns, **smaller the variance for B0 and Y0**
* **Larger the spread of our x variable,** the more information we have, **smaller the variance for B0 and Y0**
* Larger the sample size, **smaller the variance for B0 and Y0**
* The closer the X0 is to True X, **smaller the variance for B0 and Y0**

# Performance

## RMSE (Root Mean Square Error)

Comparison between observed value and predicted value.

Mean((observed-predicted)^2)

## MAE (Mean Absolute Error)

Mean(abs(observed-predicted))

RMSE is more sensitive to outliers and errors than MAE,

MAE is more linearly defined, meaning it clearly shows the differences between the values without scaling factors.

# K fold cross validation estimation

* Data is randomly divided into k subsets of nearly equal size
* Estimate your model by leaving one subset out
* Use your estimated model to predict the observation left out
* Compute error rates aon the left out set
* Repeat k times
* Average error rate over the k run

## Caret Package

Train(dependent variable ~ predictor 1 + predictor 2 + predictor 3…, method = “lm”, trControl = trainConrtolvar)

trainControlvar = trainControl(method = “cv”, number = 10, verboseIter = FALSE)

Definition of trainControl allows parameter control over how the mode training is to be conducted.

Wrongful decision of k would result in unwanted bias:

* Small k = not enough training data
* Large k = not enough testing data, in sample prediction.

# Logistic regression

If the **dependent variable** is a binary value, it is recommended to do **logistic regression**

* There is no value beyond 0~1.
* Result is the “probability of being either one”

glm(binary dependent variable ~ predictor 1+ predictor 2+ predictor 3…, family = binomial, data = x)

this will create the logistic linear model and from here we can make a **fitted model**

**logit(binary dependent variable) = fitted model**

logit = log-odds

## Interpretation of Logit fitted model

* Intercept
  + Log odds of survival for an individual traveling in 1st class who is female and aged 0 years old
* pclass2nd coefficient represents the **difference in the log odds** between soemone travelign in **first class** and someone traveling in **2nd class**
* sexmale is negative meaning that **if the person was male, he would be 2.5 times less likely to survive**
* age is also negative meaning, **as individual gets older, they are 0.03\*age times less likely to survive**

we can also convert this other way around.

*“Work out the estimated probability of survival for a new born male travelling in first class”*

=

Therefore this person would have 73% survival rate!

## Evaluation of Prediction

### In sample performance

* + Confusion matrix

confusionMatrix(data = as.factor(predicted\_values), reference = as.factor(actual\_value))

this also shows sensitivity, specificity, predictive values.

### Out sample performance

* + Kw folds testing

Train(**binary dependent variable** ~ predictor 1 + predictor 2 + predictor 3…, **method = “glm”, family = “binomial**”, trControl = trainConrtolvar)

trainControlvar = trainControl(method = “cv”, number = 10, verboseIter = FALSE)

# Decision tree

Package called **rPart** is used to build decision tree

Rpart(dependent variable ~ predictor factor variable, method = “class”)

Rpart.plot(rpart\_obj)

Method = “class” means we want to make this into a classification tree

Each node shows

* Predicted class
* Predicted probability of each class
* Percentage of observation In each node

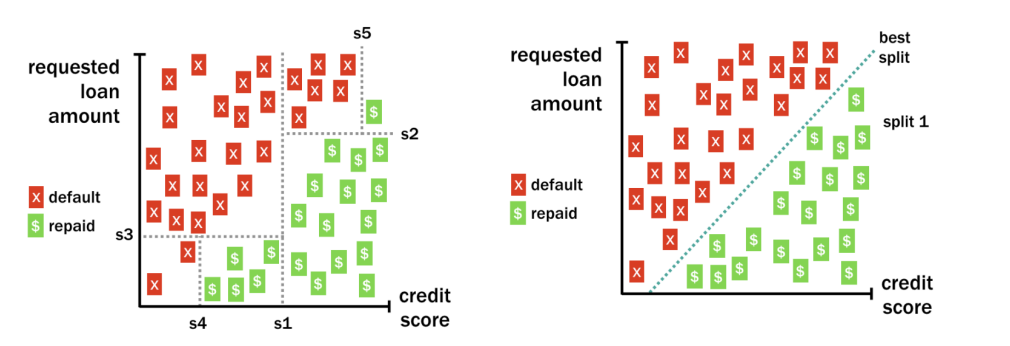
Construction of tree is made by finding the **critical point** in the data in which **decisions** could be made

### Complexity parameters

* Determins if a **new prooposed split** sufficiently improves the **pridctive power or not**
* Value that says “we have split enough”
* This helps avoid **overfitting**
* this can be calculated using caret’s train function
* Train(**Categorical dependent variable** ~ predictor 1 + predictor 2 + predictor 3…, **method = “rpart**”, trControl = trainConrtolvar)
* trainControlvar = trainControl(method = “cv”, number = 10, verboseIter = FALSE)

disadv of Decision tree

* It can be complex really quickly, without the help of complexity penelty, it would just continue untill all the individual points are assigned it’s own value.
* Might not be very sensitive to the complexity penelty
* Parallel decisions only



# Random Forest

It’s the process of constructing many tree and getting concensus result from all the trees

1. Choose the **number of decision tree** to grow and **number of variables** to consider in each tree
2. Randomly select the row of data frame **with replacement**
3. Randomly selecting the **appropriate number of variables** from the **data frame**
4. **Building a decision tee** on the **resulting datase**t
5. Repeating this procedure a **large number of times**
6. A prediction is made by **Majority rule**

randomForest(dependent variable ~ predictor variables)

predict(randomForest\_obj, new\_data)

# Nearest K neighbours

k-Nearest neighbour is a **data classification** method that uses the **classification of k neighbvours** surrounding to **identify their classification**

it identifies “distance” in Euclidean distance equation

It requires a base classified values and from there it compares with k nearest neighbours to decide what classification the point should have.

Knn(original\_data, new\_data, cl = “classification\_variable”, k = “number of neighbours”)

The number of k is important as it could also introduce bias into the classification model,

## K value decision

Train(**classification dependent variable** ~ predictor 1 + predictor 2 + predictor 3…, **method = “knn”**, trControl = trainConrtolvar)

trainControlvar = trainControl(method = “repeatedcv”, number = 5, repeat = 100)

This should show the most appropriate number of k.

## Advantages of knn

* Easy to understand
* Does not require preprocessing (if scales are correctly made)

## Disadvantages of knn

* Time consuming
* Scales need to be norminalised if it is extreme