# BIOSTATISTICS COURSE #3

# STATISTICAL MODELING #2

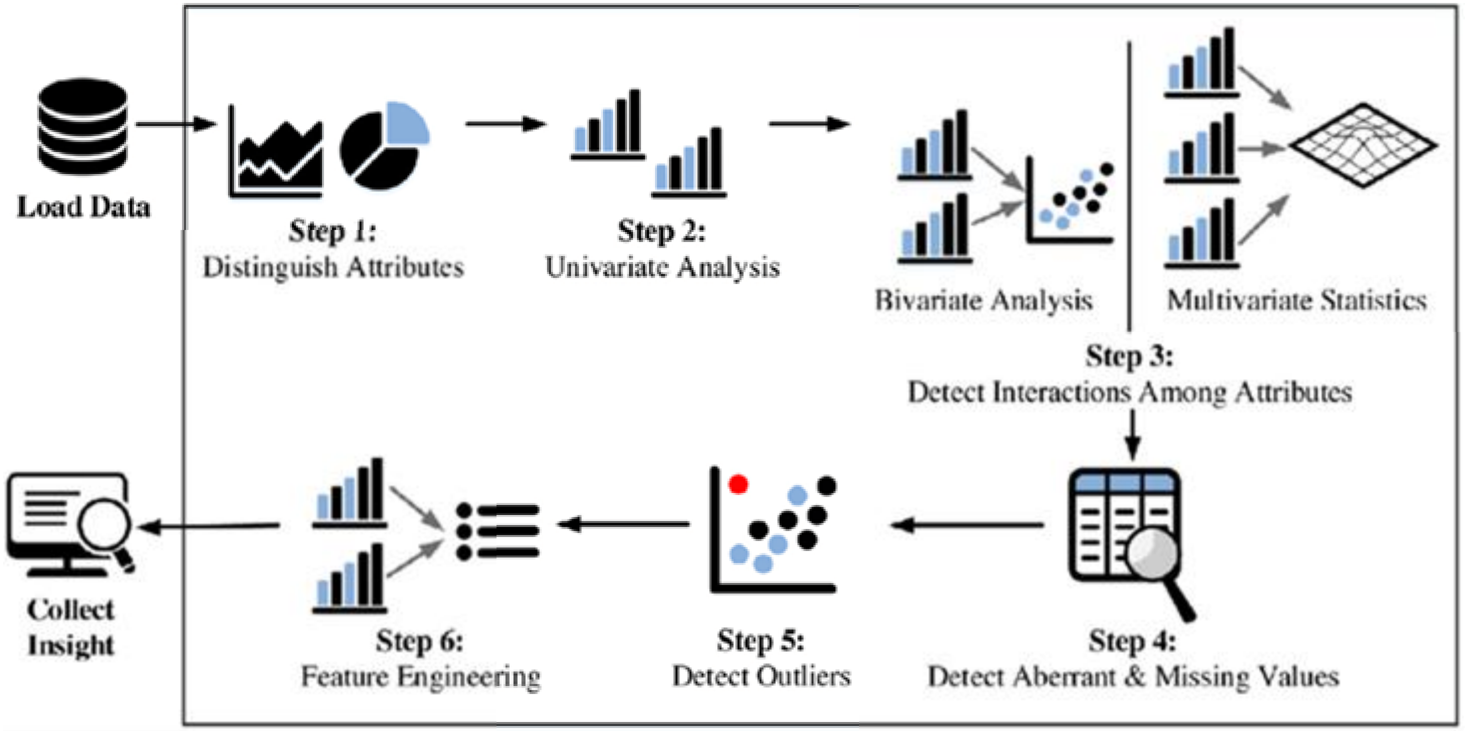
SEPTEMBER 2025

## SUMMARY OF THE COURSE #3

# INTRODUCTION

# 01

# INTRODUCTION
## DATA ANALYSIS WORKFLOW

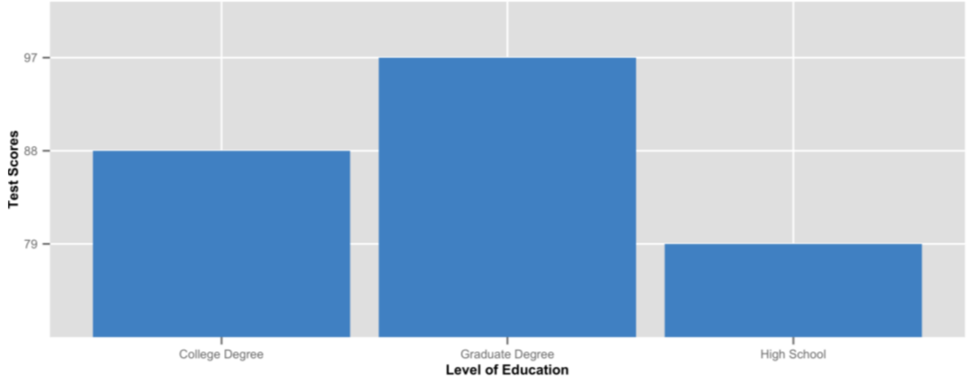


# ANOVA, MANOVA, ANCOVA AND MANCOVA

# 02

# ONE-WAY ANOVA
## COMPARISON OF MEANS BETWEEN GROUPS (ANOVA)

- **Goal** : compare **means** of a variable between more **than 2 independent groups** (normally distributed variable and equivalent variances)




- Hypotheses :
  - $H_0$ : the **means** are equivalent
  - $H_1$ : At least two **means** are different

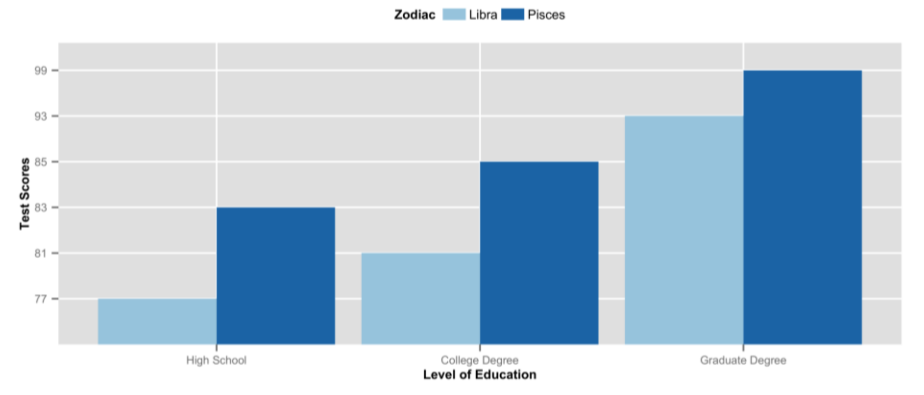
- Need to perform **post-hoc pairwise tests** in case of overall significant pvalue with **Tukey's tests**

- In R function ***aov***(*y ~ group) and* ***tukey_hsd(****y ~ group*) (package ***rtatix***)

- **Outputs** : ANOVA table and pairwise comparisons (for post-hoc tests)

# TWO-WAY ANOVA
## COMPARISON OF MEANS ACROSS SUBGROUPS

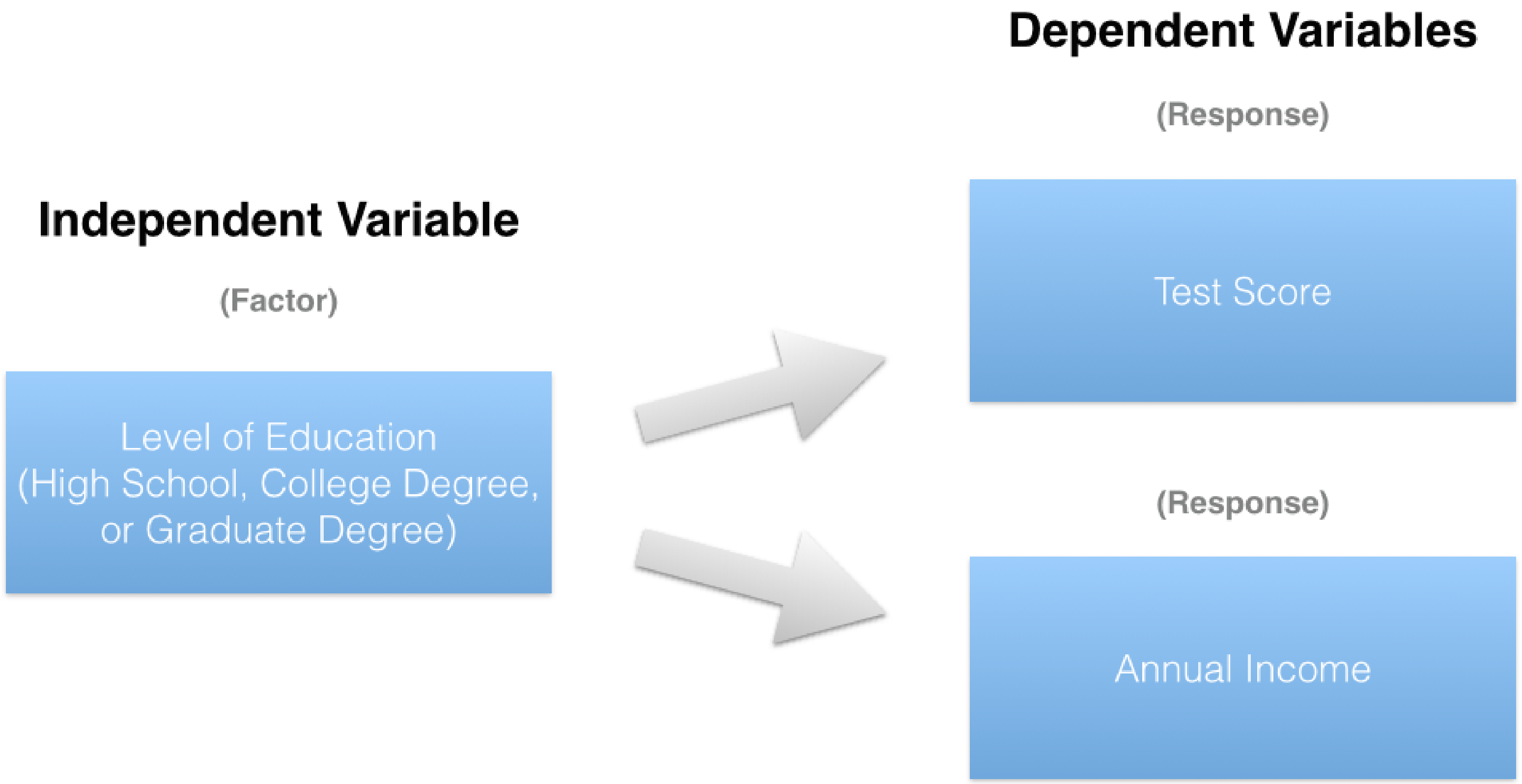
- **Goal** : assess impact of two (or more) categorical parameters (factor) on a continuous outcome (normally distributed variable and equivalent variances)



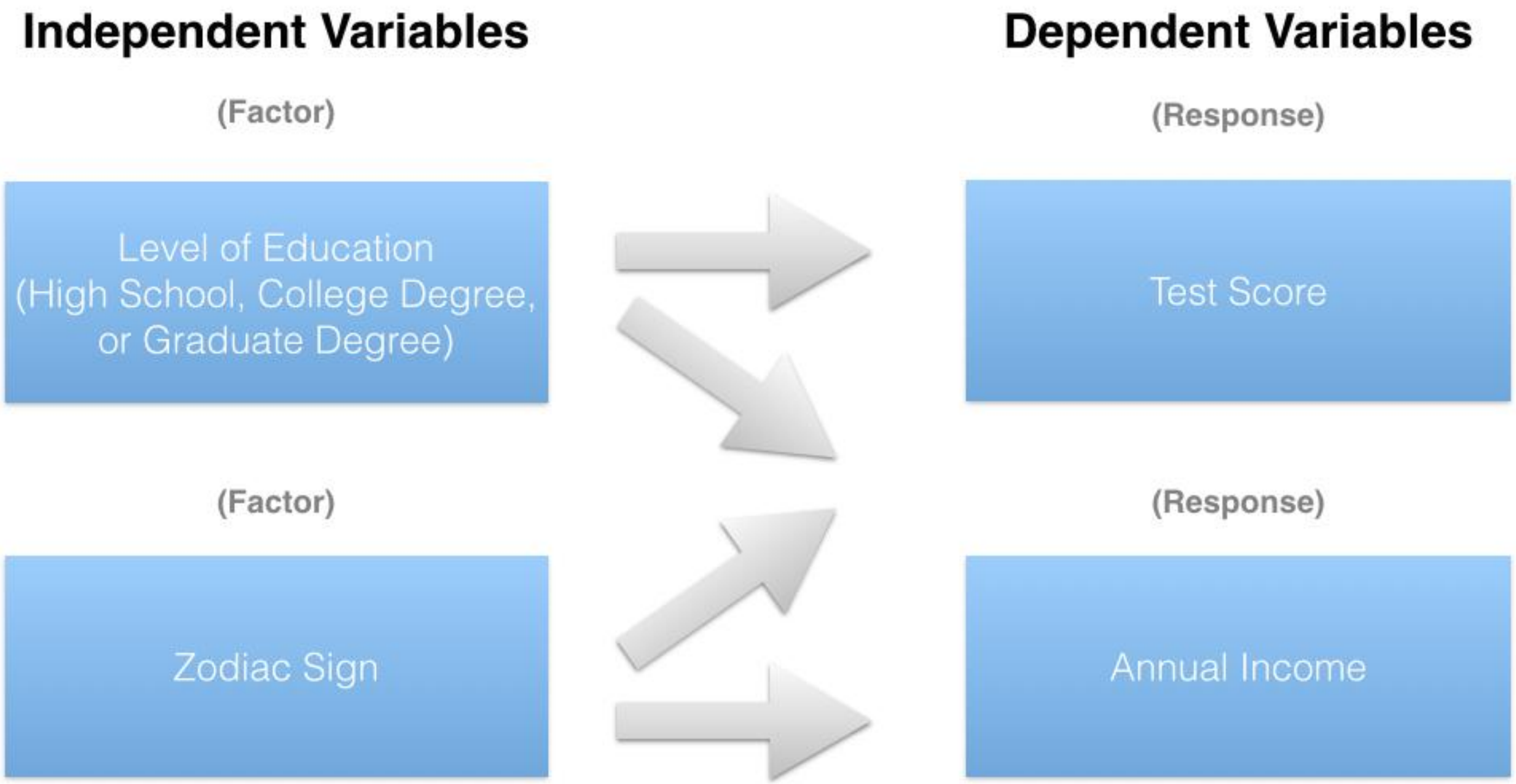

- **Many hypotheses (2 factors)** :
  - $H_{A0}$ : no difference in the means of factor A
  - $H_{A1}$ : means are not equal for factor A
  - $H_{B0}$ : no difference in the means of factor B
  - $H_{B1}$ : means are not equal for factor B
  - $H_{AB0}$ : no interaction between factors A and B
  - $H_{AB1}$ : significant interaction between factors A and B

- In R function ***anova_test***(*y ~ factors) and* ***tukey_hsd(****y ~ parameters)* (package ***rtatix***)

# MANOVA (MULTIVARIATE ANOVA)
## ONE-WAY MANOVA - EXAMPLE



# MANOVA (MULTIVARIATE ANOVA)
## TWO-WAY MANOVA - EXAMPLE



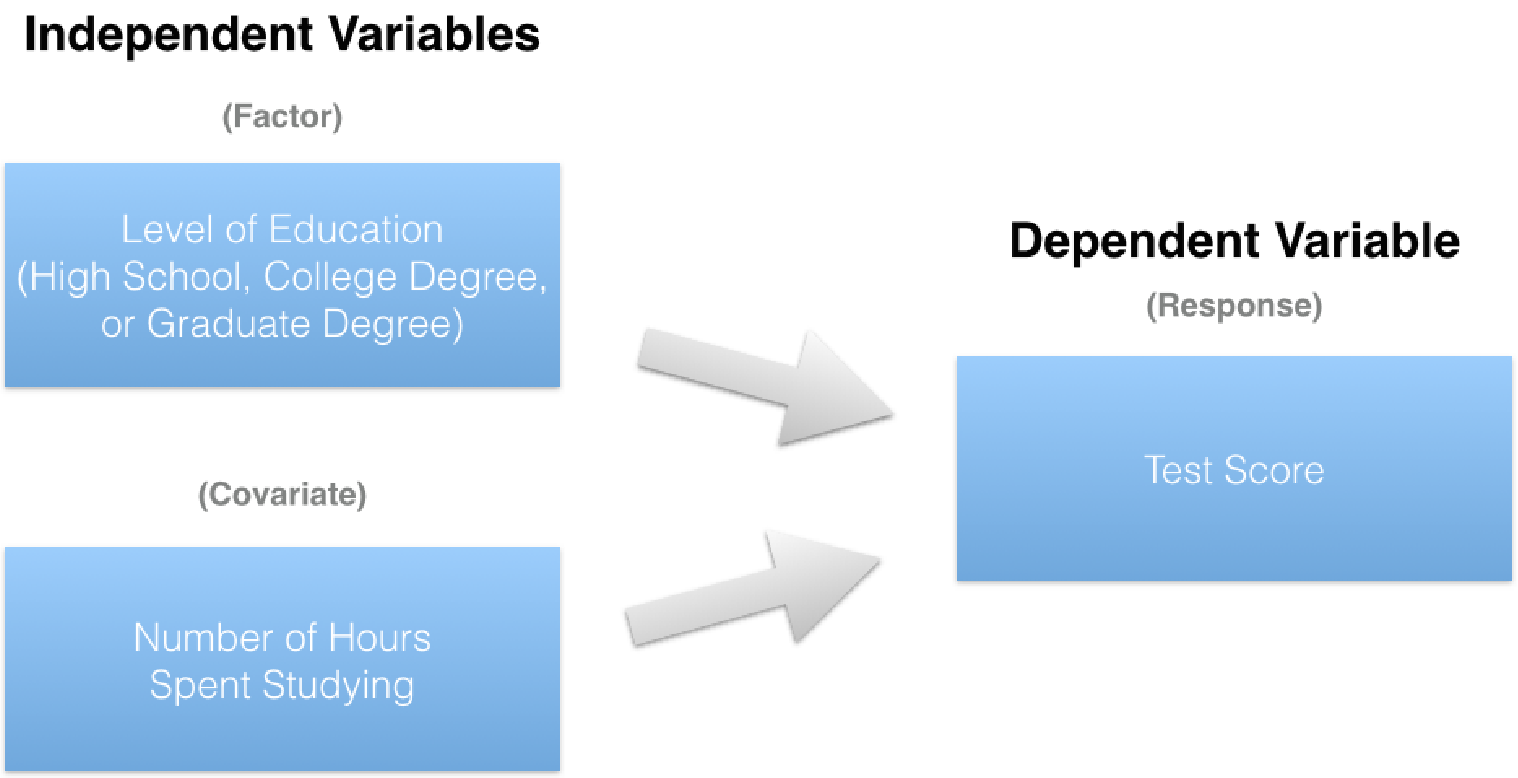
# MANOVA (MULTIVARIATE ANOVA)
## HIGHLIGHTS

- **Goal** : assess impact of one (or many) categorical parameter (factor) on two (or more) continuous outcomes

- **Assumptions :**
  - dependent variables should be **normally distributed within groups** (checked with *mshapiro.test* function in *mvnormtest* package)
  - **Homogeneity of variances** across the range of predictors
  - **No multicollinearity between dependent variables :** checked with *cor_test* function (package *rstatix*)

- In R : function *manova(cbind(y1,y2…) ~ factor1 + factor2…* (package *stats*)

# ANCOVA (ANALYSIS OF COVARIANCE)
## EXAMPLE



# ANCOVA (ANALYSIS OF COVARIANCE)
HIGHLIGHTS

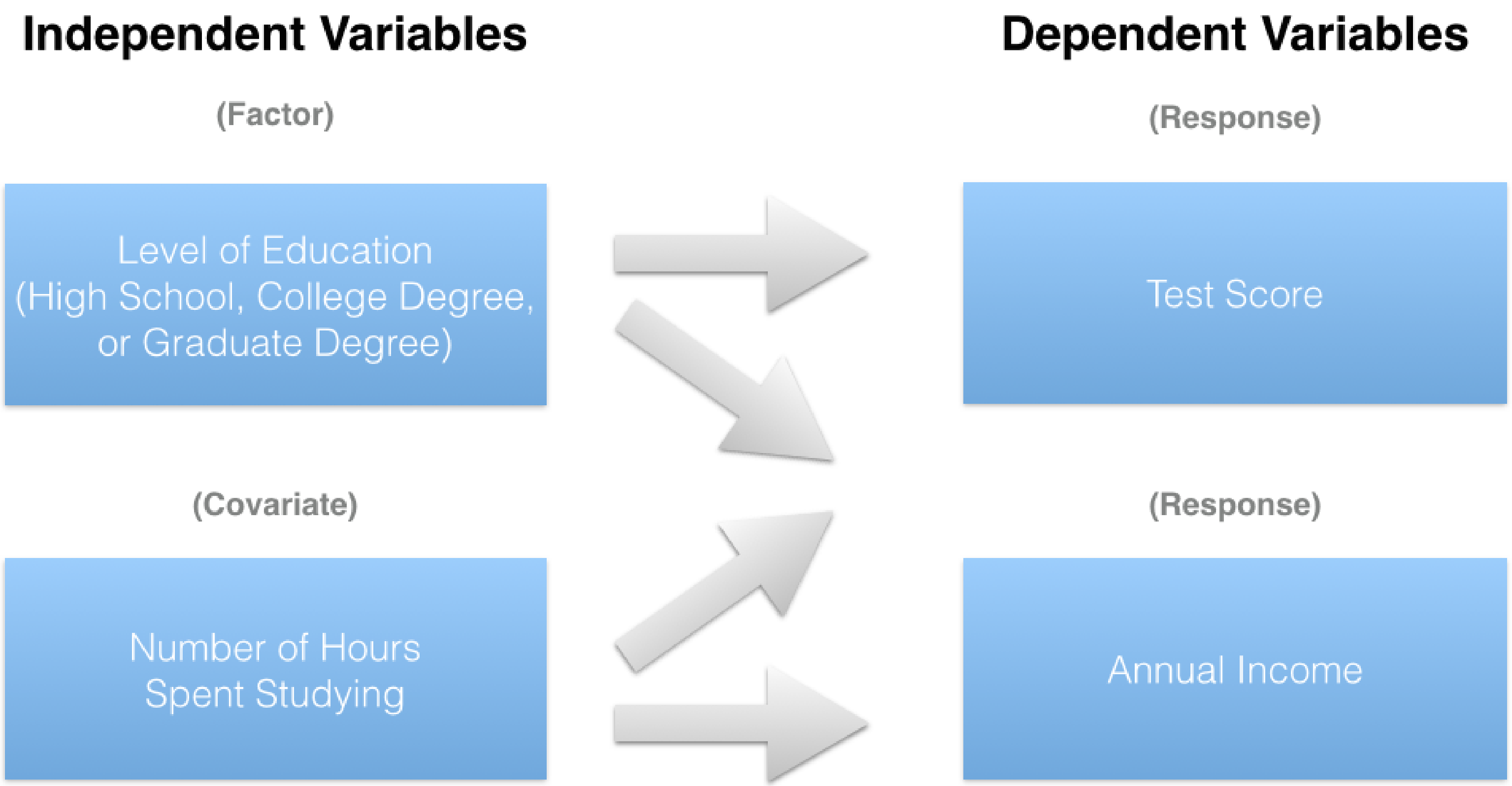
- Similar to ANOVA but with covariate(s) (continuous factor)

- Assumptions :
  - The covariate(s) and the factor are **independent**
  - **Homogeneity of variances** across the modalities of the factor
  - **Homogeneity of regression slopes** within each group

- Need to perform **post-hoc pairwise tests** in case of overall significant pvalue with **Tukey's tests**

- In R function ***manova***(*y ~ factor + covariate) and* ***tukey_hsd***(*y ~ group)* (package ***rstatix***)

# MANCOVA (MULTIVARIATE ANALYSIS OF COVARIANCE)
## EXAMPLE



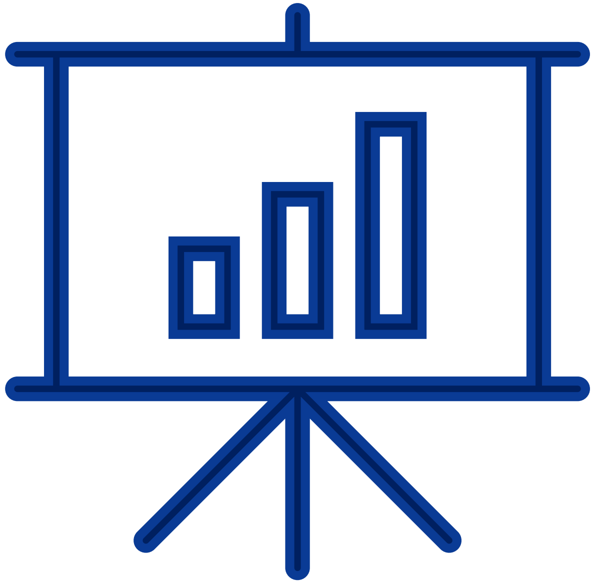
# MANCOVA (MULTIVARIATE ANALYSIS OF COVARIANCE)
## HIGHLIGHTS

- Similar to ANCOVA but with more than one response variable (continuous)

- Assumptions :
  - ➢ dependent variables should be **normally distributed within groups** (checked with ***mshapiro.test*** function in ***mvnormtest*** package)
  - ➢ The covariate(s) and the factor(s) are **independent**
  - ➢ **Homogeneity of variances** across the range of predictors.
  - ➢ **Homogeneity of regression slopes** within each group

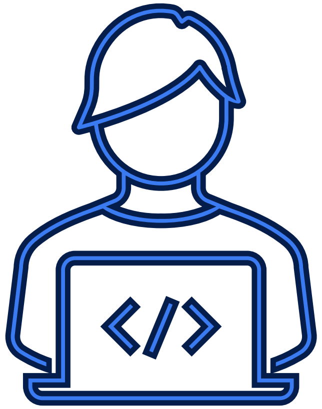
- In R : function ***manova***(*cbind(y1,y2…) ~ factor + covariate* (package ***stats***)

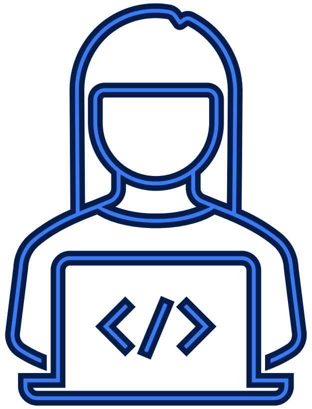
# ANOVA, MANOVA, ANCOVA AND MANCOVA



Live demo

# ANOVA, MANOVA, ANCOVA AND MANCOVA





Time to play !
(20 minutes)

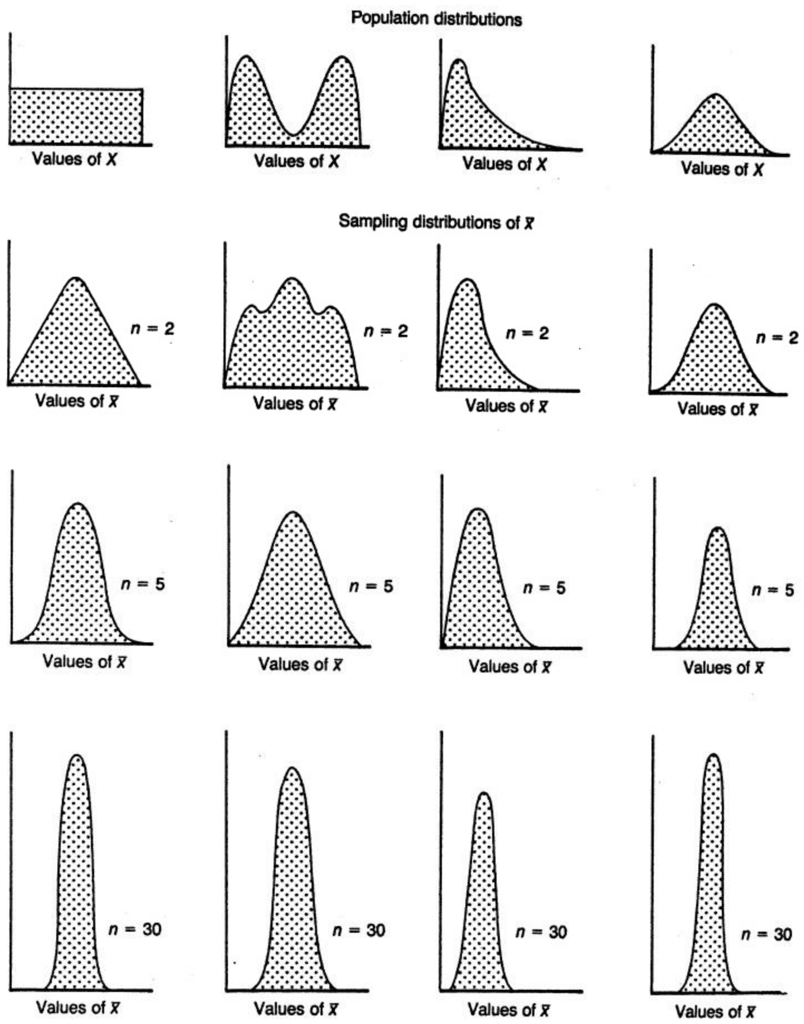
# NON-LINEAR MODELS

# 03

# NON-LINEAR MODELS
## INTRODUCTION

- **Central limit theorem (CLT)** : under appropriate conditions the distribution of a **normalized version** of the sample mean converges to a standard normal distribution.

- i.e : whatever the form of the population distribution, **the sampling distribution tends to a Gaussian**, and its dispersion is given by the central limit theorem.



# NON-LINEAR MODELS
## INTRODUCTION

- In most of cases, when N is big (> 30), CLT is valid and **linear models** (univariate or multivariate) **are applicable**.

- In case of **non-linear** relationship between a **continuous dependent variable** and factors, two options are available :

  - **Transform** the explanatory variables with logarithm or square-root
  - Use **non-linear models**

- Many **non-linear models** are available : quadratic models, cubic models, exponential models, logarithmic models, logistic models…

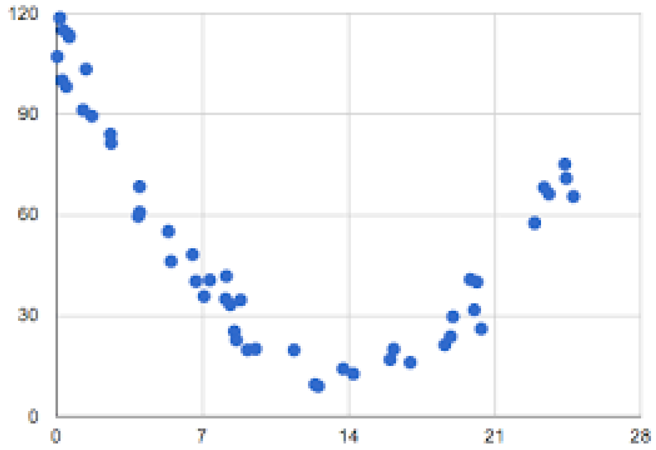
# NON-LINEAR MODELS
## QUADRATIC MODEL

- **Goal** : explain a continuous variable $Y$ with a continuous variable $X$

- **Hypothesis to test** : the relationship between the variable **is non-linear (quadratic shape)**

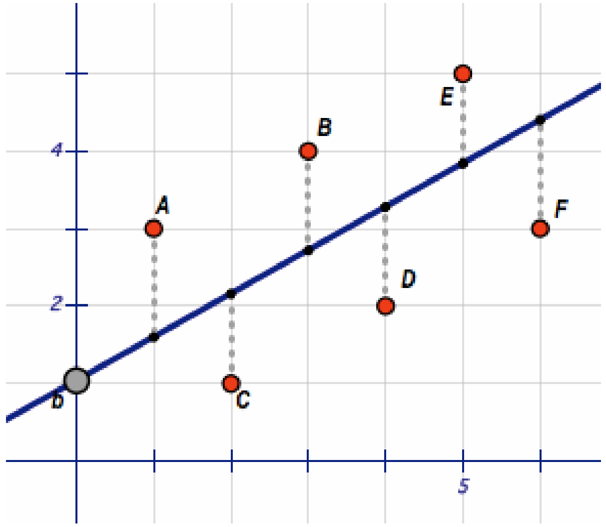
- **Equation** : $Y = a \times X^2 + b \times X + c + \varepsilon$

with
$Y$ : variable to explain (continuous)
$X$ : explanatory variable (continuous)
$a$ : coefficient of the quadratic term $X^2$
$b$ : coefficient of the linear term $X$
$c$ : intercept (value of $Y$ when $X$ = 0)
$\varepsilon$ : model residuals (proportion of the variability of $Y$ not explained)



# NON-LINEAR MODELS
## QUADRATIC MODEL

- **Adjustment method : least square algorithm:** plays on parameters $a$, $b$ and $c$ in the equation in order to minimize the sum of squares (sum of $\varepsilon^2$) between real values $Y$ and the model (estimated values $\hat{Y}$)




- **Equation :** $$Y = a \times X^2 + b \times X + c + \varepsilon$$

with
$Y$ : variable to explain (continuous)
$X$ : explanatory variable (continuous)
$a$ : coefficient of the quadratic term $X^2$
$b$ : coefficient of the linear term $X$
$c$ : intercept (value of $Y$ when $X$ = 0)
$\varepsilon$ : model residuals (proportion of the variability of $Y$ not explained)

# NON-LINEAR MODELS
## QUADRATIC MODEL – MODELING WITH R

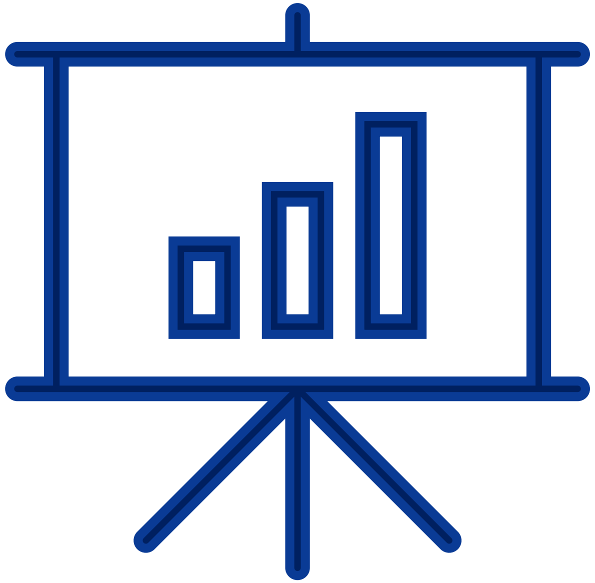
*lm* function (*stats* package)

Parameters :
- *formula* = $Y \sim I(X^2) + X$ with $Y$ and $X$ are continuous
- *data* = dataset (number of points : N)
- *subset* = train model only on a subset of the dataset
- *weight* = optional vector with the weights of points
- *na.action* = handling of NA values

Output :
- **Coefficients** of the equation
- **Residuals**
- **Fitted values**
- **$R^2$ & Adjusted $R^2$** (also called $Q^2$) & **RMSE**

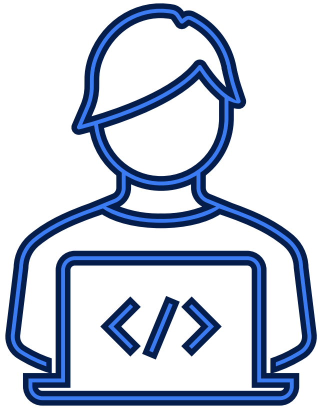
Two ways to write the **formula** : $I(X^2) + X$ vs $poly(X, 2)$

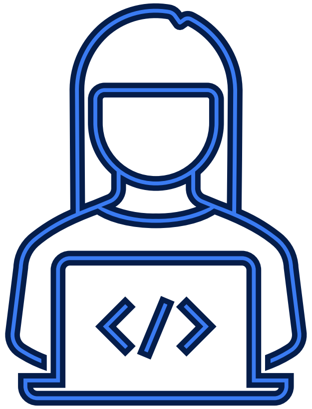
# NON-LINEAR MODELS
## QUADRATIC MODEL



Live demo

# NON-LINEAR MODELS
## QUADRATIC MODEL

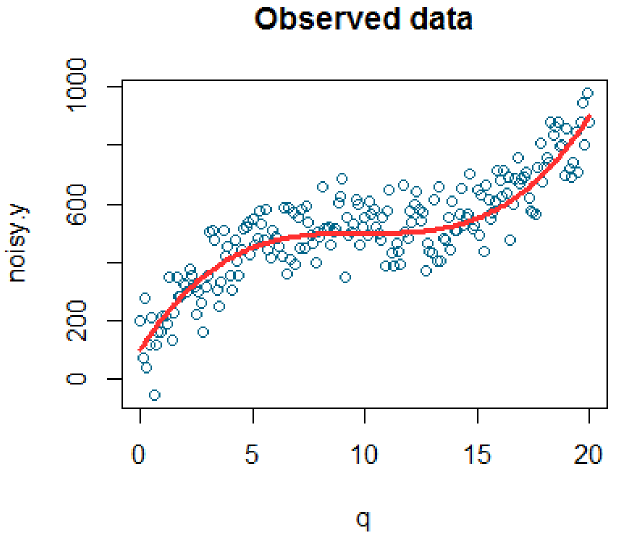




Time to play !
(20 minutes)

# NON-LINEAR MODELS
## POLYNOMIAL MODEL

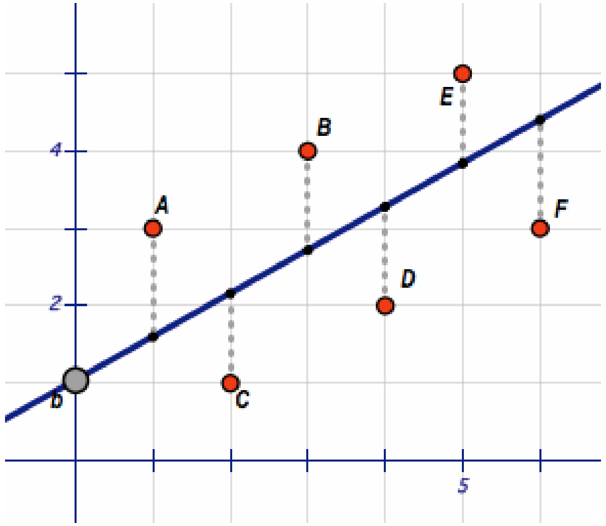
- **Goal** : explain a continuous variable $Y$ with a continuous variable $X$
- **Hypothesis to test** : the relationship between the variable **is non-linear (polynomial shape)**
- **Equation** : $$Y = a_1 \times X^K + a_2 \times X^{K-1} \cdots + b + \varepsilon$$




with
$Y$ : variable to explain (continuous)
$X$ : explanatory variable (continuous)
$K$ : degree of the polynom
$a_K$ : coefficient of the term $X^K$
$b$ : intercept (value of $Y$ when $X$ = 0)
$\varepsilon$ : model residuals (proportion of the variability of $Y$ not explained)

# NON-LINEAR MODELS
## POLYNOMIAL MODEL

- **Adjustment method : least square algorithm:** plays on parameters of the equation in order to minimize the sum of squares (sum of $\varepsilon^2$) between real values $Y$ and the model (estimated values $\hat{Y}$)




- **Equation :** $Y = a_1 \times X^K + a_2 \times X^{K-1} \cdots + b + \varepsilon$

with
$Y$ : variable to explain (continuous)
$X$ : explanatory variable (continuous)
$K$ : degree of the polynom
$a_K$ : coefficient of the term $X^K$
$b$ : intercept (value of $Y$ when $X$ = 0)
$\varepsilon$ : model residuals (proportion of the variability of $Y$ not explained)

# NON-LINEAR MODELS
## POLYNOMIAL MODEL – MODELING WITH R

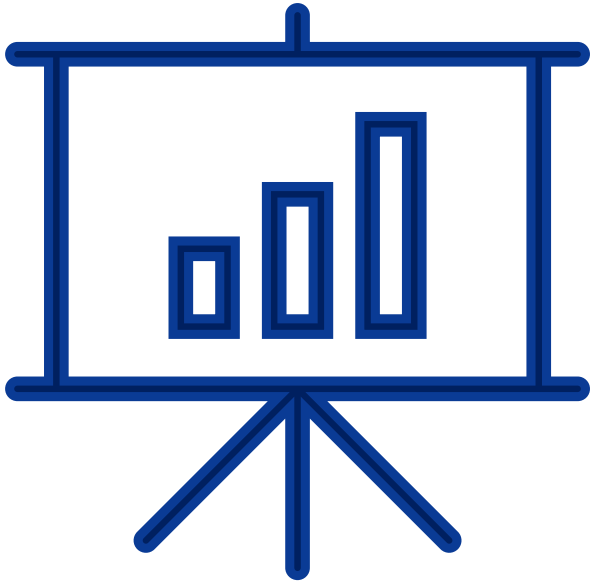
*lm* function (*stats* package)

Parameters :
- *formula* = $Y \sim poly(X, K)$ with $Y$ and $X$ are continuous
- *data* = dataset (number of points : N)
- *subset* = train model only on a subset of the dataset
- *weight* = optional vector with the weights of points
- *na.action* = handling of NA values

Output :
- **Coefficients** of the equation
- **Residuals**
- **Fitted values**
- **$R^2$ & Adjusted $R^2$** (also called $Q^2$) & **RMSE**

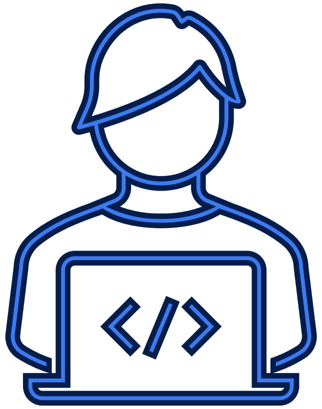
*anova(model1,model2)* function allows to compare models (built on same data)

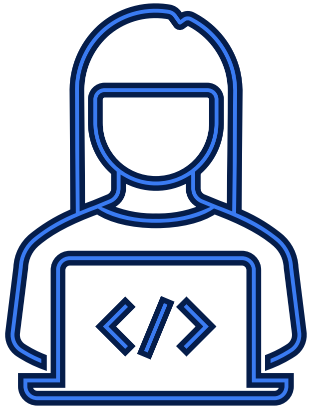
# NON-LINEAR MODELS
## POLYNOMIAL MODEL



Live demo

# NON-LINEAR MODELS
## POLYNOMIAL MODEL





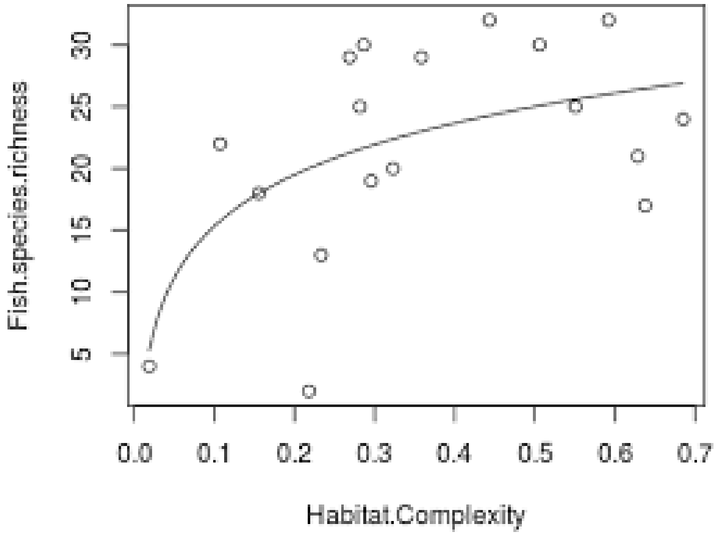
Time to play !
(30 minutes)

# NON-LINEAR MODELS
## LOGARITHMIC MODEL

- **Goal** : explain a continuous variable $Y$ with a continuous variable $X$

- **Hypothesis to test** : the relationship between the variable **is non-linear (logarithmic shape)**

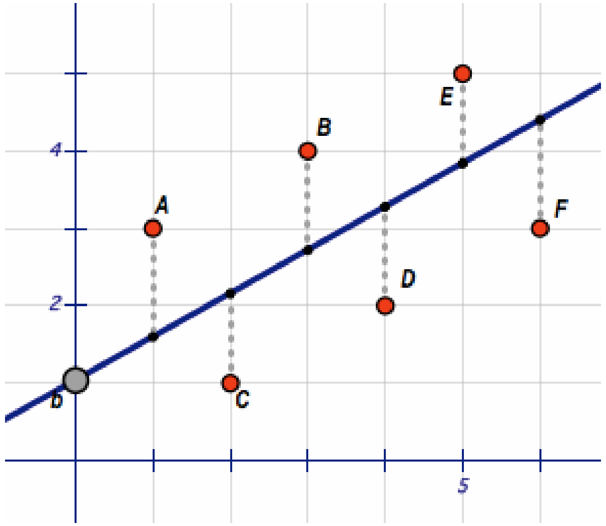
- **Equation** : $$Y = a \times \log(X) + X + b + \varepsilon$$




with
$Y$ : variable to explain (continuous)
$X$ : explanatory variable (continuous)
$a$ : coefficient of the logarithmic term $X$
$b$ : intercept (value of $Y$ when $X$ = 0)
$\varepsilon$ : model residuals (proportion of the variability of $Y$ not explained)

# NON-LINEAR MODELS
## LOGARITHMIC MODEL

- **Adjustment method : least square algorithm:** plays on parameters $a$ and $b$ in the equation in order to minimize the sum of squares (sum of $\varepsilon^2$) between real values $Y$ and the model (estimated values $\hat{Y}$)




- **Equation :** $$Y = \mathrm{a} \times \log(X) + X + b + \varepsilon$$

with $Y$ : variable to explain (continuous)
$X$ : explanatory variable (continuous)
$a$ : coefficient of the logarithmic term $X$
$b$ : intercept
$\varepsilon$ : model residuals (proportion of the variability of $Y$ not explained)

# NON-LINEAR MODELS
## LOGARITHMIC MODEL – MODELING WITH R

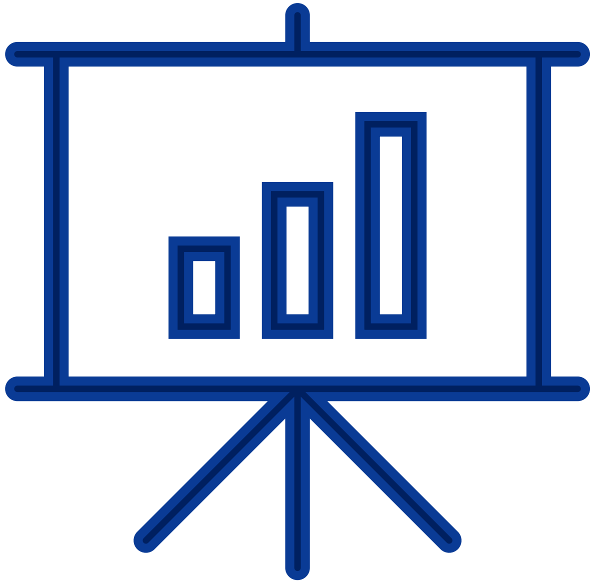
*lm* function (*stats* package)

Parameters : *formula* = $Y \sim \log(X)$ with $Y$ and $X$ are continuous
*data* = dataset (number of points : N)
*subset* = train model only on a subset of the dataset
*weight* = optional vector with the weights of points
*na.action* = handling of NA values

Output : **Coefficients** of the equation
**Residuals**
**Fitted values**
**$R^2$ & Adjusted $R^2$** (also called $Q^2$) & **RMSE**

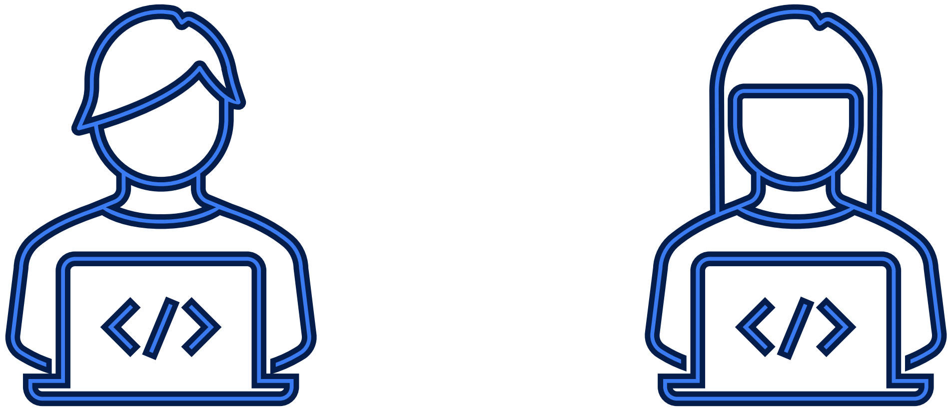
# NON-LINEAR MODELS

## LOGARITHMIC MODEL



Live demo

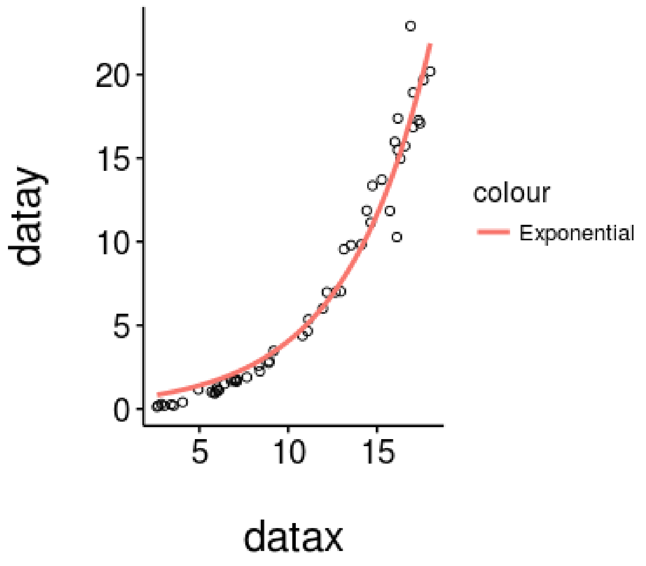
# NON-LINEAR MODELS
## LOGARITHMIC MODEL



Time to play !
(20 minutes)

# NON-LINEAR MODELS
## EXPONENTIAL MODEL

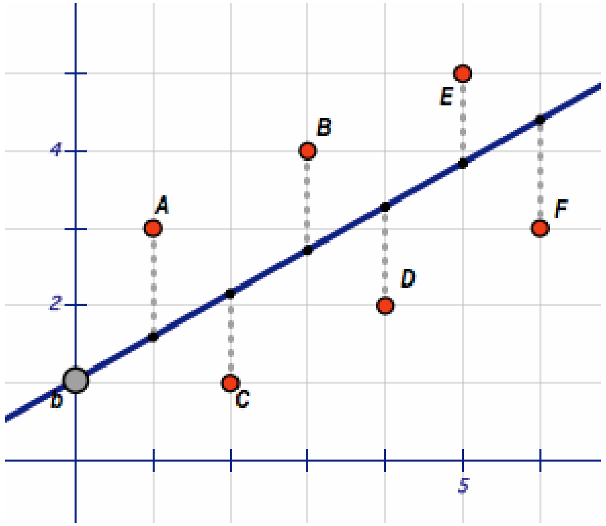
- **Goal** : explain a continuous variable $Y$ with a continuous variable $X$
- **Hypothesis to test** : the relationship between the variable **is non-linear (exponential shape)**
- **Equation** : $Y = \mathrm{a} \times e^{X} + b + \varepsilon$




with $Y$ : variable to explain (continuous)
$X$ : explanatory variable (continuous)
$a$ : coefficient of the exponential term $X$
$b$ : intercept (value of $Y$ when $X$ = 0)
$\varepsilon$ : model residuals (proportion of the variability of $Y$ not explained)

# NON-LINEAR MODELS
## EXPONENTIAL MODEL

- **Adjustment method** : **least square algorithm:** plays on parameters $a$ and $b$ in the equation in order to minimize the sum of squares (sum of $\varepsilon^2$) between real values $Y$ and the model (estimated values $\hat{Y}$)




- **Equation** : $Y = \mathrm{a} \times \exp(X) + X + b + \varepsilon$

with
$Y$ : variable to explain (continuous)
$X$ : explanatory variable (continuous)
$a$ : coefficient of the exponential term $X$
$b$ : intercept (value of $Y$ when $X$ = 0)
$\varepsilon$ : model residuals (proportion of the variability of $Y$ not explained)

# NON-LINEAR MODELS
## EXPONENTIAL MODEL – MODELING WITH R

*lm* function (*stats* package)

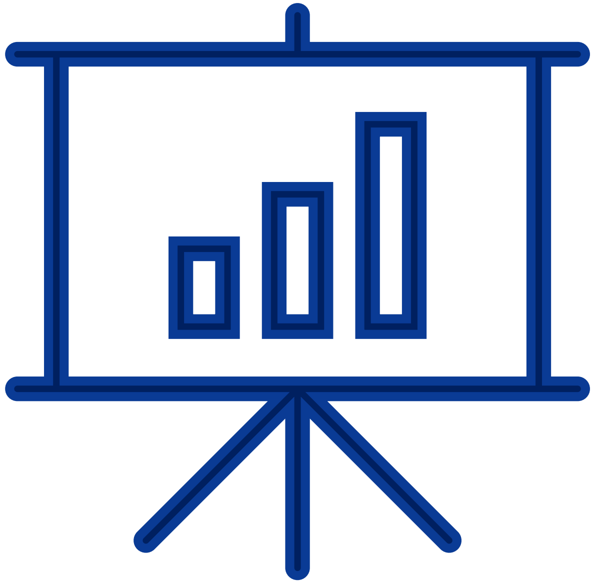
Parameters :

- *formula* = $Y \sim e^X$ with $Y$ and $X$ are continuous
- *data* = dataset (number of points : N)
- *subset* = train model only on a subset of the dataset
- *weight* = optional vector with the weights of points
- *na.action* = handling of NA values

Output :

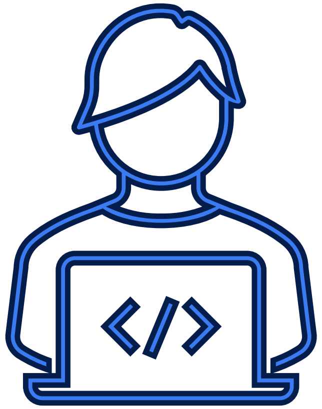
- Coefficients of the equation
- Residuals
- Fitted values
- $R^2$ & Adjusted $R^2$ (also called $Q^2$) & RMSE

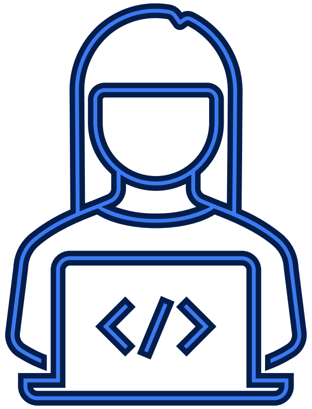
# NON-LINEAR MODELS
## EXPONENTIAL MODEL



Live demo

# NON-LINEAR MODELS
## EXPONENTIAL MODEL





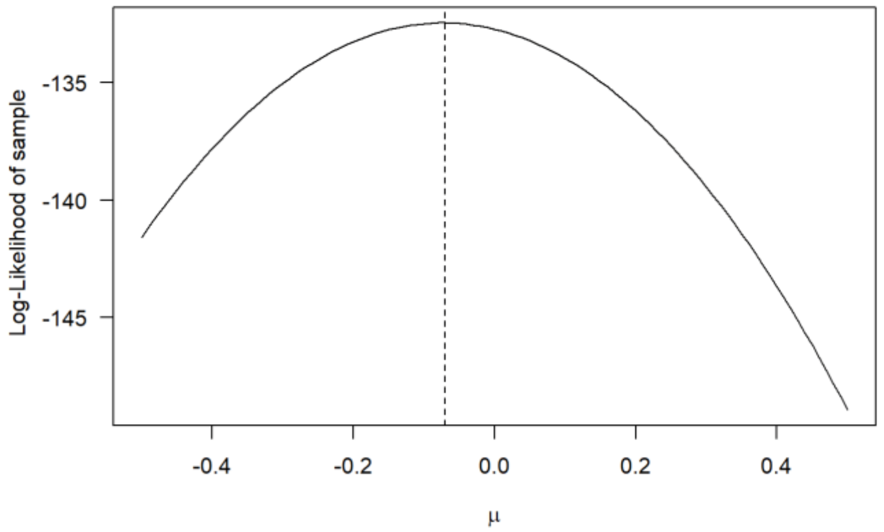
Time to play !
(20 minutes)

# GENERALIZED LINEAR MODELS

# 04

# GENERALIZED LINEAR MODELS
## INTRODUCTION

- When equations are too complex and not linear, **Least Square method** is not the most converging algorithm to use when trying to fit a model to data.




- A powerful alternative way exists :
  **Maximum Likelihood Estimation** (MLE)

- **Principle** : the algorithm try to find the **parameters of the distribution** of data (and not trying to minimize residuals).

# GENERALIZED LINEAR MODELS
## INTRODUCTION

Two main models using MLE, also called “**Generalized Linear Models**” exists and aim to link a discrete $Y$ to continuous and/or discrete factors :

- **Poisson model** (and alternative models : quasi-poisson, negative-binomial) : often use to model **count data**

- **Logistic regression** : commonly used in the research of **risk factors of a disease** in clinical trials for example

These two models are available in R in with *glm* function (package *stats*).

# GENERALIZED LINEAR MODELS
## INTRODUCTION

Thanks to Maximum Likelihood function, **two indexes** are available for model comparisons :

- **Akaike information criterion** (AIC) : trade-off between the goodness of fit of the model and the simplicity of the model: balance between overfitting and underfitting.

$$AIC = 2 \times k - 2 \times \ln(\hat{L})$$

with $\boldsymbol{k}$ : number of parameters $(\boldsymbol{X})$

$\hat{L}$ : Maximum Likelihood value

**Lower AIC : best model**

# GENERALIZED LINEAR MODELS
## INTRODUCTION

Thanks to Maximum Likelihood function, **two indexes** are available for model comparisons :

- **Bayesian information criterion** (BIC) : like AIC but with penalization of complex models when the number of points is > 8.

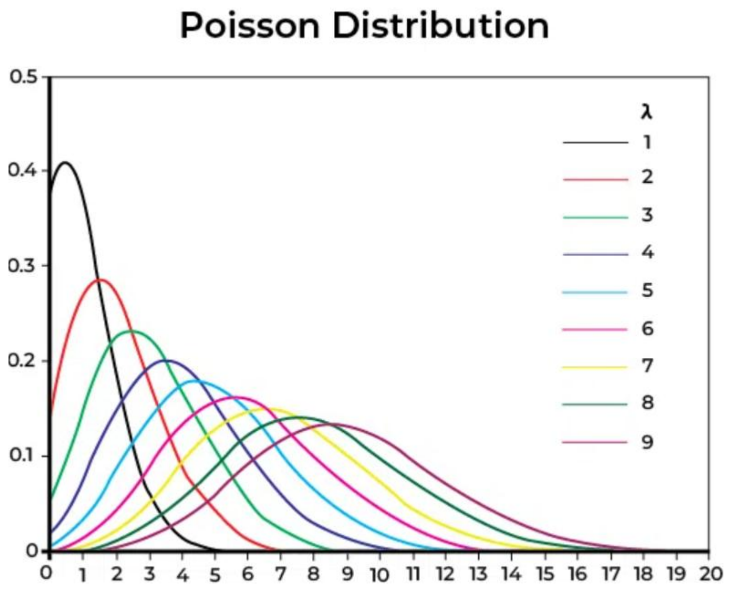
$$BIC = k \times \ln(n) - 2 \times \ln(\hat{L})$$

with $k$ : number of parameters $(X)$
$n$ : number of observations
$\hat{L}$ : Maximum Likelihood value

**Lower BIC : best model**

# GENERALIZED LINEAR MODELS
## POISSON MODEL

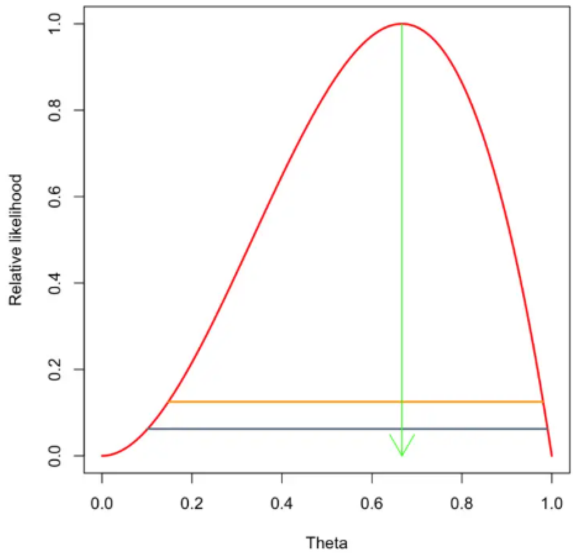
- **Goal** : explain a discrete variable $Y$ (counts ) with a variable $X$
- **Hypothesis to test** : $Y$ follows a poisson distribution (with parameter $\lambda$ = mean and standard-deviation)
- **Equation** : $Y = \mathrm{a} \times X + b + \varepsilon$




with $Y$ : variable to explain (discrete)
$X$ : explanatory variable (continuous ou discrete)
$a$ : coefficient of $X$
$b$ : intercept (value of $Y$ when $X$ = 0)
$\varepsilon$ : model residuals (proportion of the variability of $Y$ not explained)

# GENERALIZED LINEAR MODELS
## POISSON MODEL

- **Adjustment method** : **Maximum Likelihood Algorithm :** the model will try to guess the value of the parameter $\lambda = \exp(X \times \beta)$ in order to minimize the sum of squares (sum of $\varepsilon^2$) between real values $Y$ and the model (estimated values $\hat{Y}$)



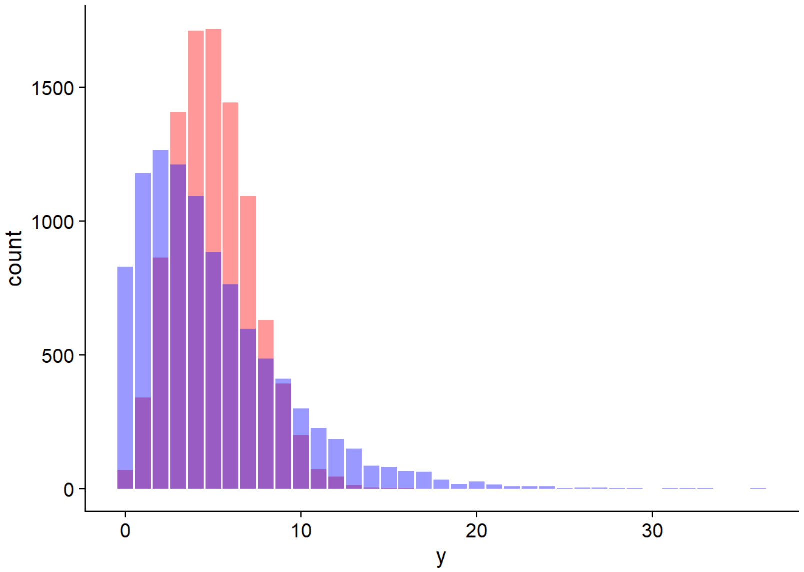

- **Equation** : $$P(Y{=}y/\lambda) = \frac{e^{-\lambda}\lambda^{y}}{y!}$$

with $Y$ : variable to explain (discrete)
$y$ : observations (continuous ou discrete)
$\lambda$ : parameter of Poisson distribution

**Validity** : distribution of $Y$ values follows a Poisson law : $\bar{y} = \sigma$

# GENERALIZED LINEAR MODELS
## POISSON MODEL

- **Overdispersion** : the observation that variation is higher than would be expected according to the Poisson law (assessed when $\frac{residual\ deviance}{ddl} > 1$)




- In case of overdispersion : **quasi-poisson** or **negative-binomial** model

- **$R^2$ of Generalized Linear Models** does not have the same meaning compared to linear model : the % represents the **% of reduction of deviance** (difference between log-likelihood functions) of the current model vs the **null model** (with only the intercept in the equation).

# GENERALIZED LINEAR MODELS
## POISSON MODEL – MODELING WITH R

*glm* function (*stats* package)

Parameters :
- *formula* = $Y \sim X$ with $Y$ discrete and $X$ continuous or discrete
- *family* = poisson(link="log")
- *data* = dataset (number of points : N)
- *subset* = train model only on a subset of the dataset
- *weight* = optional vector with the weights of points
- *na.action* = handling of NA values

Output :
- **Coefficients** of the equation
- Residuals
- Fitted values
- Dispersion analysis
- AIC

# GENERALIZED LINEAR MODELS
## QUASI-POISSON MODEL – MODELING WITH R

*glm* function (*stats* package)

Parameters : *formula* = $Y \sim X$ with $Y$ discrete and $X$ continuous or discrete
*family* = quasipoisson()
*data* = dataset (number of points : N)
*subset* = train model only on a subset of the dataset
*weight* = optional vector with the weights of points
*na.action* = handling of NA values

Output : Coefficients of the equation
Residuals
Fitted values
Dispersion analysis
AIC

# GENERALIZED LINEAR MODELS

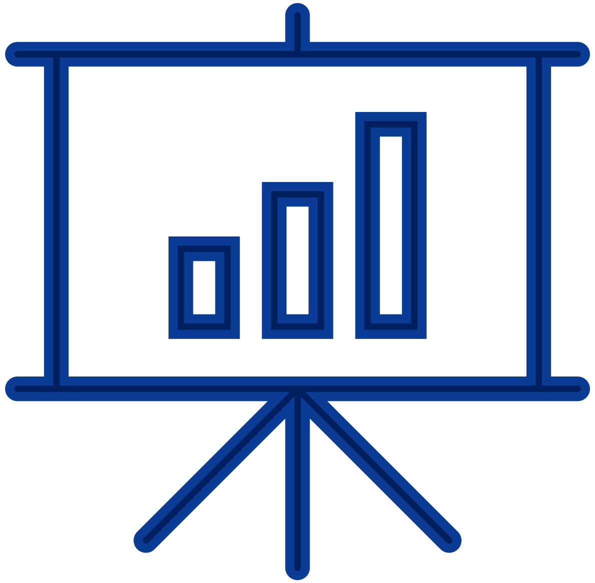
## NEGATIVE-BINOMIAL MODEL – MODELING WITH R

*glm.nb* function (*MASS* package)

Parameters :
*formula* = $Y \sim X$ with $Y$ discrete and $X$ continuous or discrete
*family =* quasipoisson()
*data* = dataset (number of points : N)
*subset* = train model only on a subset of the dataset
*weight* = optional vector with the weights of points
*na.action* = handling of NA values

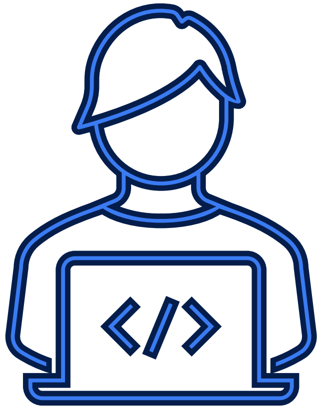
Output :
Coefficients of the equation
Residuals
Fitted values
Dispersion analysis
AIC

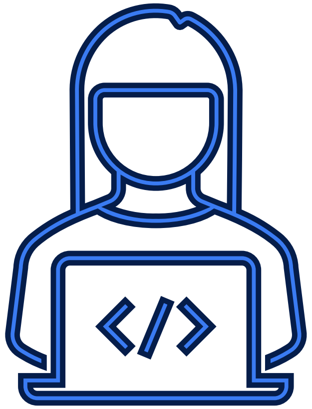
# GENERALIZED LINEAR MODELS
## POISSON MODEL



Live demo

# GENERALIZED LINEAR MODELS
## POISSON MODEL



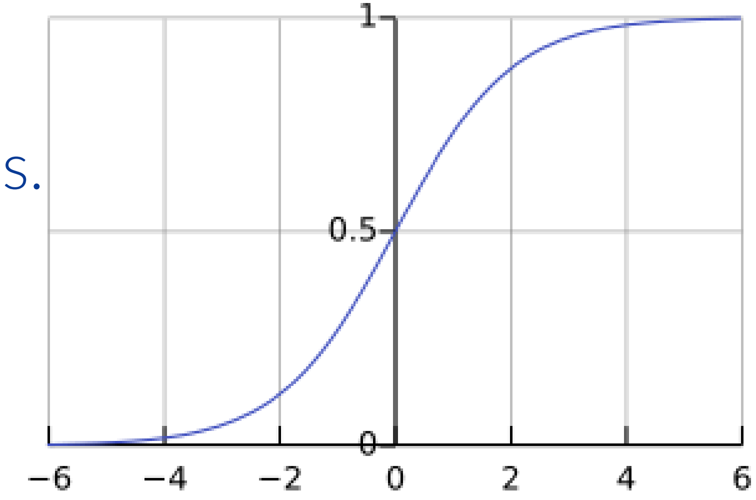


Time to play !
(15 minutes)

# GENERALIZED LINEAR MODELS
## LOGISTIC REGRESSION

- **Goal** : explain a binary variable $Y$ with a continuous or discrete variable(s) $X$

- **Assumptions** : no colinearity between explanatory variables, independance of observations, no outliers, binary response well balanced between the two classes.



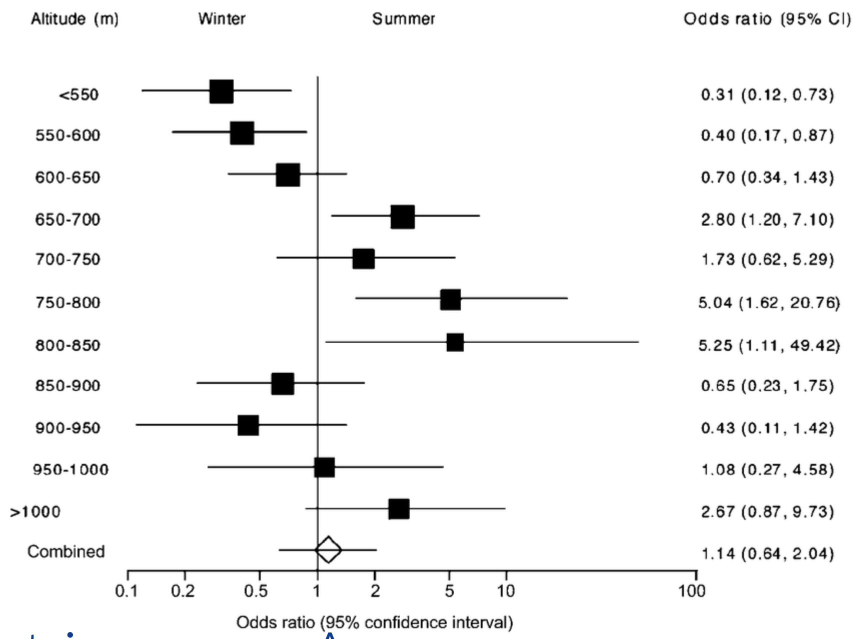

- **Link function** : logit

- No residuals normality study : **predicted vs observed classes table**

- Results : $\hat{Y}$ : **probability** to belong to group 1
  **Odds-ratios** : **effect size** of each explanatory variable

# GENERALIZED LINEAR MODELS
## LOGISTIC REGRESSION – ODDS-RATIOS (OR)

$$\text{Odds} - \text{ratio} = \frac{\dfrac{p}{1-p}}{\dfrac{q}{1-q}} = \frac{p(1-q)}{q(1-p)}$$




with : $p$ : probability of event in group A
$q$ : probability of event in group B

- Values (always positive) : < 1 : event is less frequent in group A
  = 1 : no difference in frequency of event in both groups
  > 1 : event is more frequent in group A

- OR is always presented with **95% confidence interval** (95%CI) and pvalue

# GENERALIZED LINEAR MODELS
## LOGISTIC REGRESSION – QUALITY OF MODEL

- No $R^2$ in logistic regression

- Instead, a **confusion matrix** is used :

| Predicted / Actual | Sick | Healthy |
|---|---|---|
| Positive test | TRUE positive (TP) | FALSE positive (FP) |
| Negative test | FALSE negative (FN) | TRUE negative (TN) |

# GENERALIZED LINEAR MODELS
## LOGISTIC REGRESSION – QUALITY OF MODEL

- Many metrics are derived from this matrix :

$$\textbf{Specificity} = \frac{\boldsymbol{TP}}{\boldsymbol{TP+FN}}$$

(True Positive Rate : probability the test is positive in the sick pop)

$$\textbf{Sensitivity} = \frac{\boldsymbol{TN}}{\boldsymbol{TN+FP}}$$

(False Positive Rate : probability the test is negative in the healthy pop)

| Predicted / Actual | Sick | Healthy |
|---|---|---|
| Positive test | TRUE positive (TP) | FALSE positive (FP) |
| Negative test | FALSE negative (FN) | TRUE negative (TN) |

# GENERALIZED LINEAR MODELS
## LOGISTIC REGRESSION – QUALITY OF MODEL

- Many metrics are derived from this matrix :

$$\textbf{Positive Predicted Value} = \frac{TP}{TP+FP}$$

(probability the illness is present when test is positive)

$$\textbf{Negative } \boldsymbol{Predicted\ Value} = \frac{TN}{TN+FN}$$

(probability the illness is absent when test is negative)

| Predicted / Actual | Sick | Healthy |
|---|---|---|
| Positive test | TRUE positive (TP) | FALSE positive (FP) |
| Negative test | FALSE negative (FN) | TRUE negative (TN) |

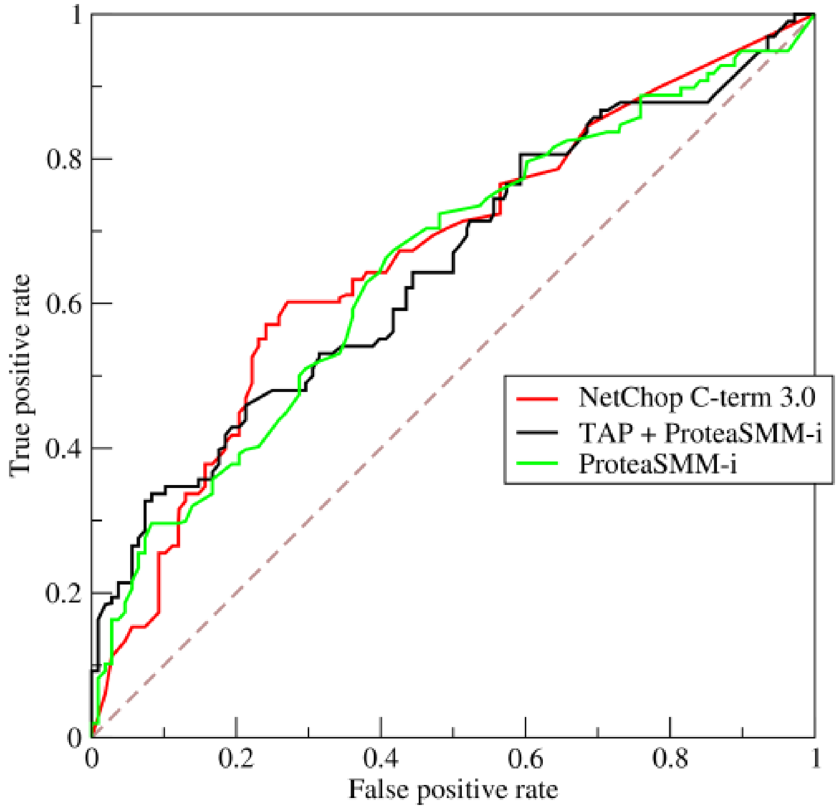
# GENERALIZED LINEAR MODELS
## LOGISTIC REGRESSION – ROC CURVE

- Helpful tool to **compare models** : **ROC curve** (Receiver Operative Characteristic)

- Displays the **evolution of TPR and FPR** in the population

- Dataset is sorted by ascending **score** (probability) calculated by the logistic regression and **TPR** and **FPR** are calculated

- The **Area Under the Curve** (AUC) is calculated (from 0 to 1) : **roc** from pROC package




- The diagonal represents the **random classification** of individuals (AUC of 0.5).

# GENERALIZED LINEAR MODELS

## LOGISTIC REGRESSION – MODELING WITH R

*glm* function (*stats* package)

Parameters :

- *formula* = $Y \sim X$ with $Y$ discrete and $X$ continuous or discrete
- *family* = binomial
- *data* = dataset (number of points : N)
- *subset* = train model only on a subset of the dataset
- *weight* = optional vector with the weights of points
- *na.action* = handling of NA values

Output :

- Coefficients of the equation (to convert into OR with exponential)
- Residuals
- Fitted values
- Dispersion analysis
- AIC

# GENERALIZED LINEAR MODELS
## LOGISTIC REGRESSION – MODELING WITH R

*predict* function (*stats* package)
Parameters : *model* = logistic model
*type* = “response”
Output : Probability of class 1 for each individual. If > 0.5 : class 1

*roc* function (*pROC* package) (same parameters as above).
Output : roc object

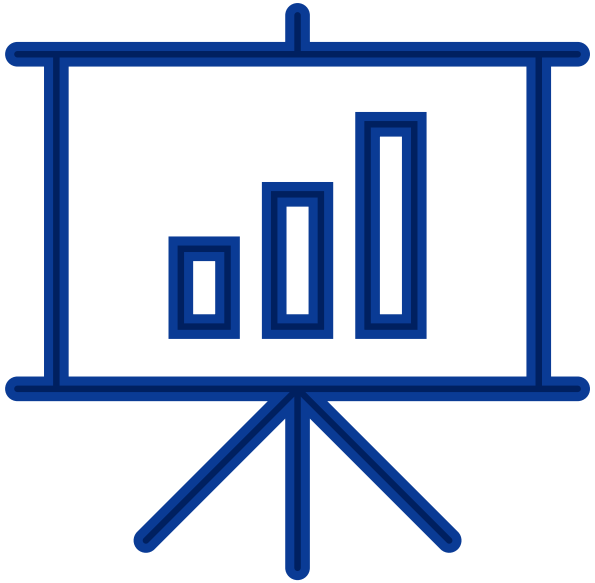
*plot.roc* function (*pROC* package) (parameter : roc object)
Parameters : *x* = roc object
Output : ROC curve

# GENERALIZED LINEAR MODELS
## LOGISTIC REGRESSION – MODELING WITH R

*geom_roc* function (*plotROC* package)
Output : ROC curve with ggplot2

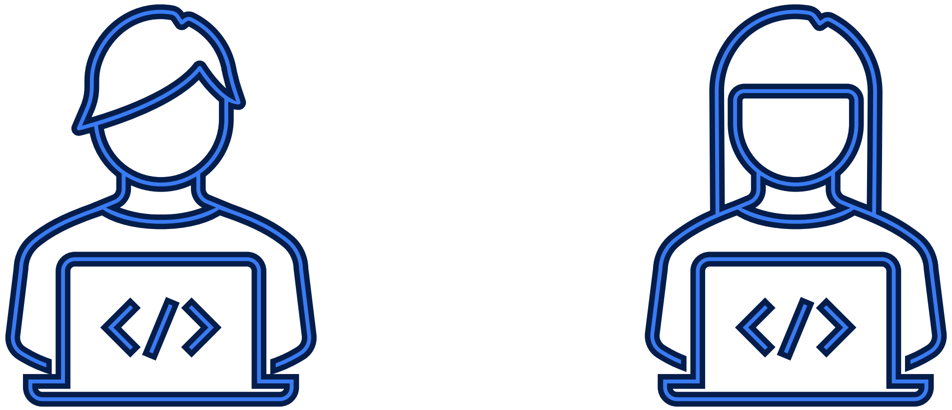
*confusionMatrix* function (*caret* package)
Parameters : *table* = cross table with real vs predicted outcome
table(real,predicted)
Output : Confusion Matrix
Quality classification indexes (Accuracy, TPR, FPR, sensibility, specificity)

# GENERALIZED LINEAR MODELS
## LOGISTIC REGRESSION



Live demo

# GENERALIZED LINEAR MODELS
## LOGISTIC REGRESSION



Time to play !
(20 minutes)

# QUESTIONS

# 05

# THANK YOU FOR YOUR ATTENTION

SEPTEMBER 2025