
# AA ML Course – Theoretical Session 5

Assaf Hallak,

courtesy of Yahav Shadmi & Lev Faivishevsky

Advanced Analytics

We mine your business

# Agenda

- Linear Regression
- Logistic Regression
- Ensembles Intro / Random Forest
- Kmeans
- T test
- Chi Square

# The Regression Problem

- Regression is a functional approximation of the relations between explanatory variables and a response variable
  - Response (dependent, output) variable – the variable we wish to explain
  - Explanatory (independent, output) variables – the variables used to explain the dependent variable

- Regression analysis is used to
  - Predict the value of a dependent variable based on the values of the independent variables
  - Explain the impact of changes in an independent variable on the dependent variable

# Classification vs. Regression

- The difference is in the output space and the error measurement
  - Classification is a mapping from a feature space to categories (class memberships, labels), so as to minimize the probability of being wrong

$$Pr_{(X,Y)\sim D}(c(x) \neq y)$$

  - Regression is a mapping from a feature space to a numeric output space (e.g. real numbers), so as to minimize the error, e.g. minimize the squared error

$$E_{(X,Y)\sim D}\left[(f(x) - y)^2\right]$$

- These two problems are highly related and one can be reduced to the other

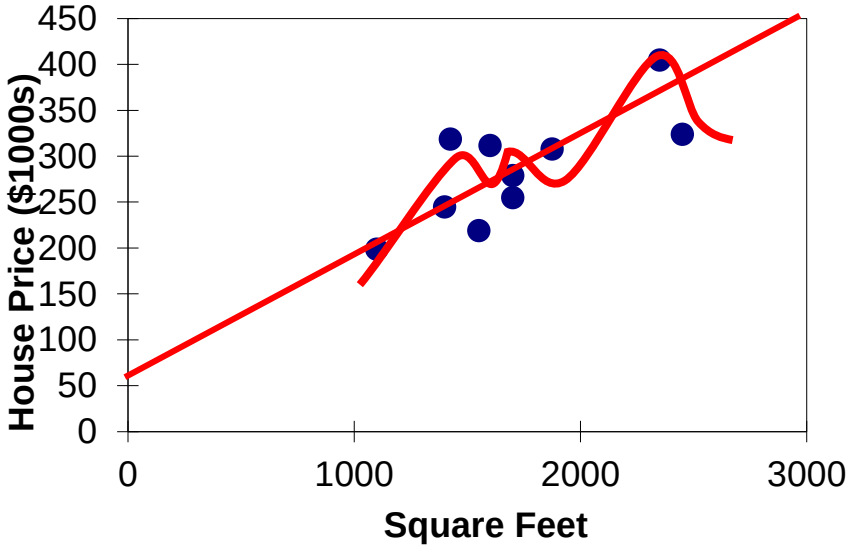
# Example – House Price

- Assume we want to know what the price of a house would be for a given size?
  - Given house of size x what is the price y=f(x) of the house?
- A random sample of 10 houses is selected as “training set”:
  - **Dependent** variable (y) = house price in $1000s
  - **Independent** variable (x) = square feet

| Square Feet (x) | House Price in $1000s (y) |
|---|---|
| 1400 | 245 |
| 1600 | 312 |
| 1700 | 279 |
| 1875 | 308 |
| 1100 | 199 |
| 1550 | 219 |
| 2350 | 405 |
| 2450 | 324 |
| 1425 | 319 |
| 1700 | 255 |

# Graphical Representation

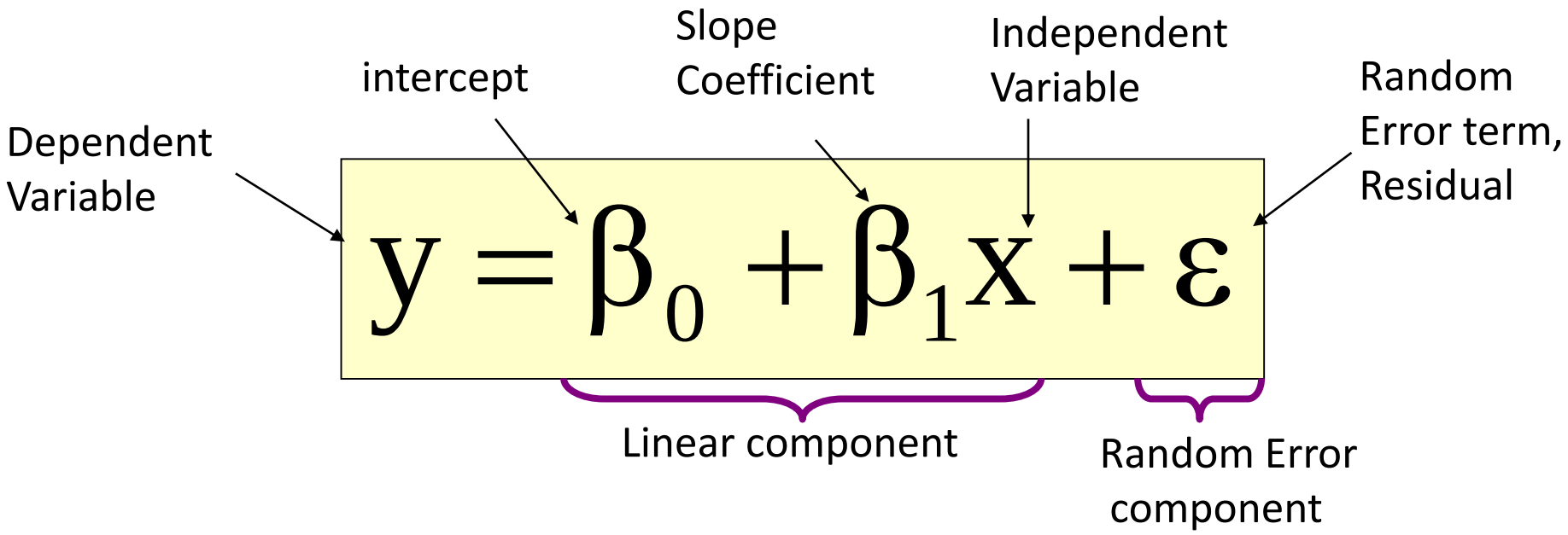
- The House Price scatter plot



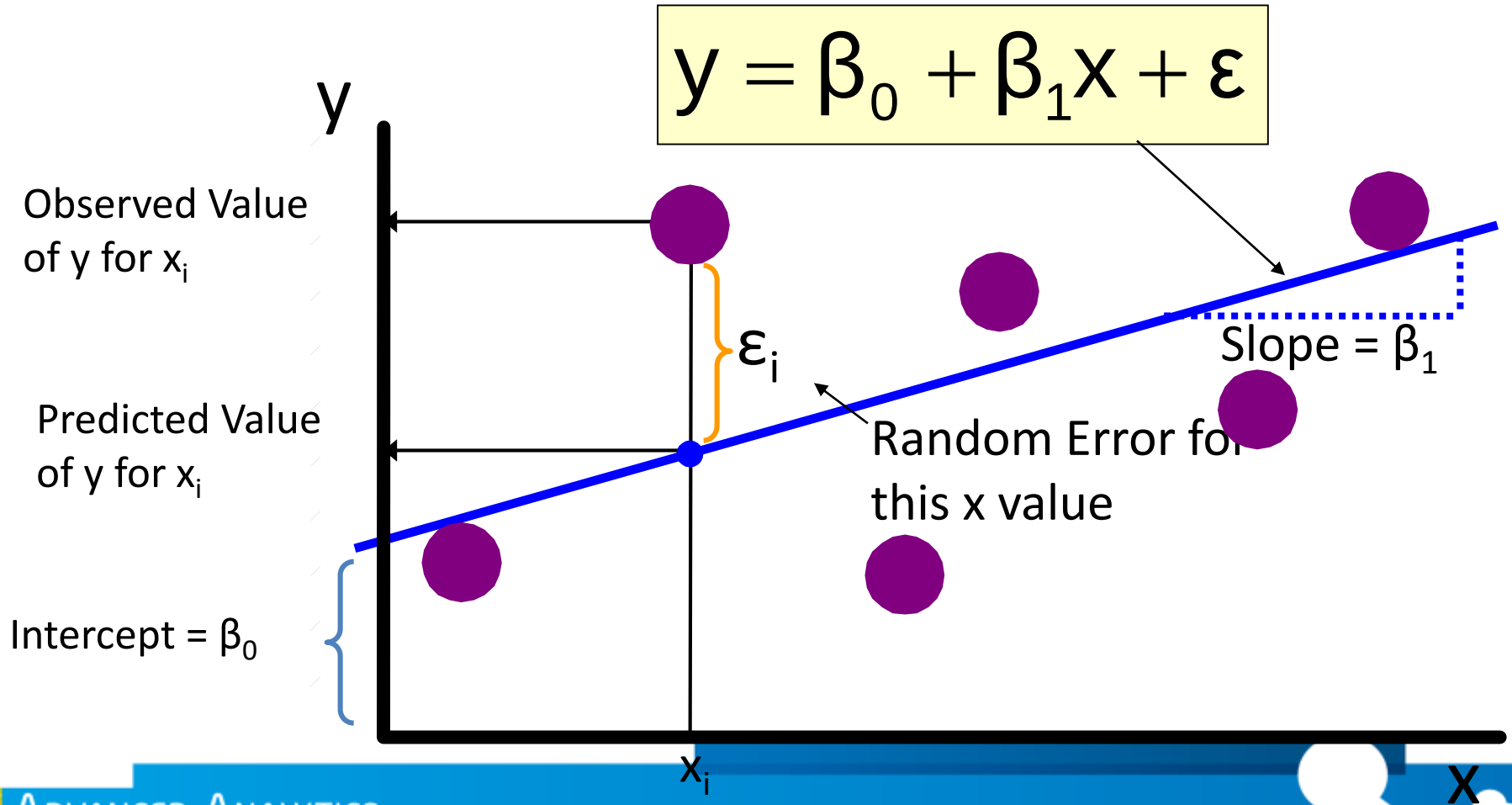

- Which function y=f(x) to choose?
- Assuming a *linear* connection between dependent and independent variables limits the search space

# A Linear Model for the House Prices

- One independent variable (house size) which "explains" the dependent variable (house price)



# Graphical Representation

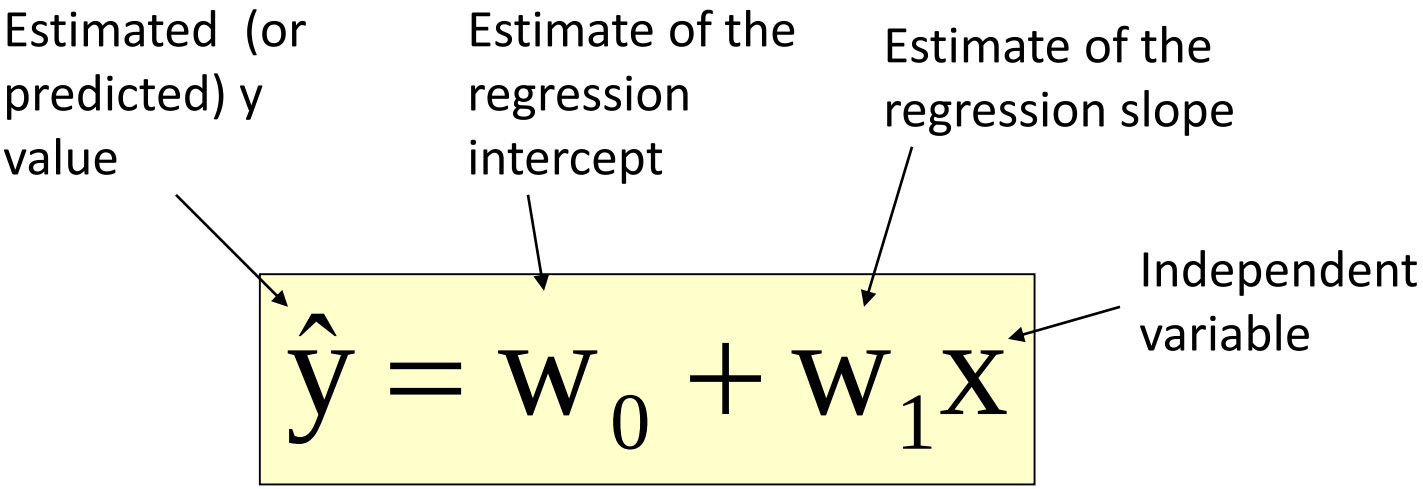


# Linear Regression Assumptions

- The underlying relationship between the x variable and the y variable is linear
- Error values (ε) are statistically independent
- The probability distribution of the errors is normal and independent of x
  - With mean 0 and an equal but unknown variance for all values of x
- $E[\varepsilon] = 0$ , hence $E[y|x] = \beta_0 + \beta_1 x$ is a linear function
  - Given a training set, the goal of linear regression is to estimate this function (the regression model)

# Estimated Regression Model

- The sample regression line provides an estimate of the population regression line




The individual random error terms $e_i$ have a mean of zero

# Least Squares Criterion

- $w_0$ and $w_1$ are obtained by minimizing the sum of the squared residuals

$$\sum \epsilon^2 = \sum (y - \hat{y})^2 = \sum \big(y - (w_0 + w_1 x)\big)^2$$

# Analytic Solution

$$E(w_0, w_1) = \sum \epsilon^2 = \sum (y - \hat{y})^2 = \sum \big(y - (w_0 + w_1 x)\big)^2$$

- $\frac{\partial}{\partial w_0} E = \sum 2\big(y - (w_0 + w_1 x)\big) = 0$
  - $\sum y - n w_0 - w_1 \sum x = 0$

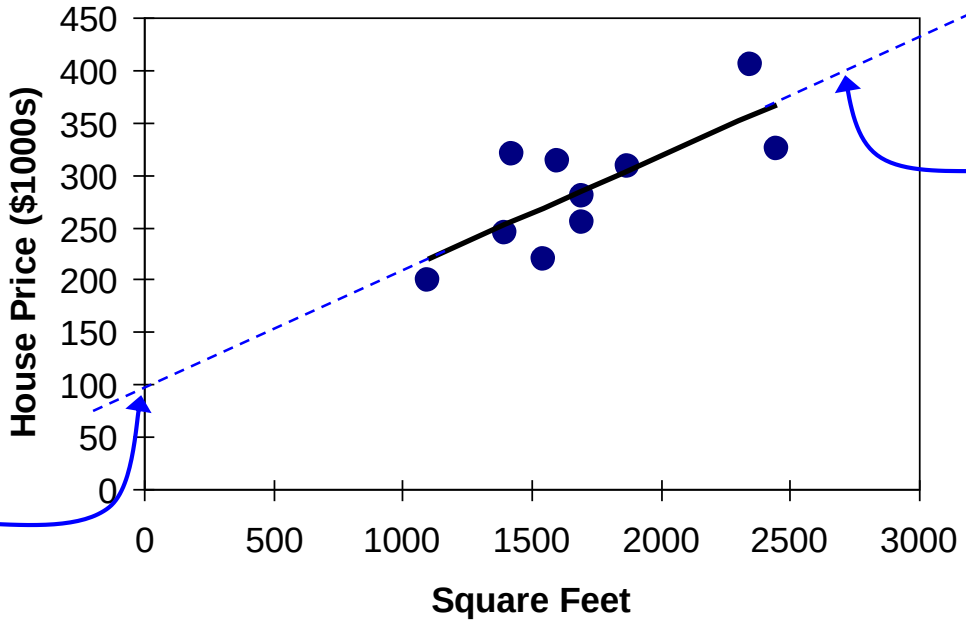
$$w_0 = \frac{\sum y - w_1 \sum x}{n} = \bar{y} - w_1 \bar{x}$$

- $\frac{\partial}{\partial w_1} E = \sum 2\big(y - (w_0 + w_1 x)\big)(-x) = 0$
  - $\sum xy - w_1 \sum x^2 - (\bar{y} - w_1 \bar{x}) \sum x = 0$
  - $\sum xy - \bar{y} \sum x = w_1 \sum x^2 - w_1 \bar{x} \sum x$

$$w_1 = \frac{\sum xy - \bar{x} \sum y}{\sum x^2 - \bar{x} \sum x} = \frac{\sum (y - \bar{y})(x - \bar{x})}{\sum (x - \bar{x})^2} = \frac{Cov\,(y, x)}{Var(x)}$$

# House Price Model

## *Scatter Plot and Regression Line*




Intercept = 98.248

Slope = 0.10977

$$\widehat{\text{house price}} = 98.24833 + 0.10977\,(\text{square feet})$$

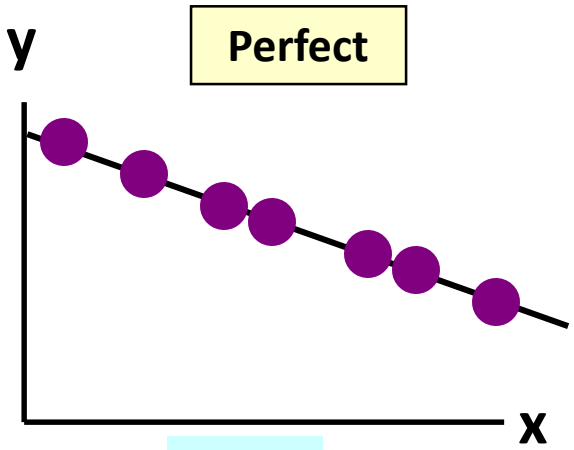
# Interpretation of the Parameters

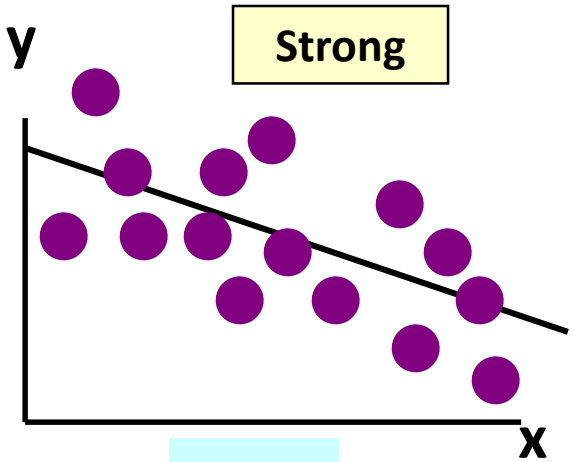
- The Intercept
  - $w_0$ is the estimated average value of Y when the value of X is zero
    - *For the house price model, no houses had 0 square feet, so $w_0 = 98.24833$ just indicates that, for houses within the range of sizes observed, $98,248.33 is the portion of the house price not explained by square feet*

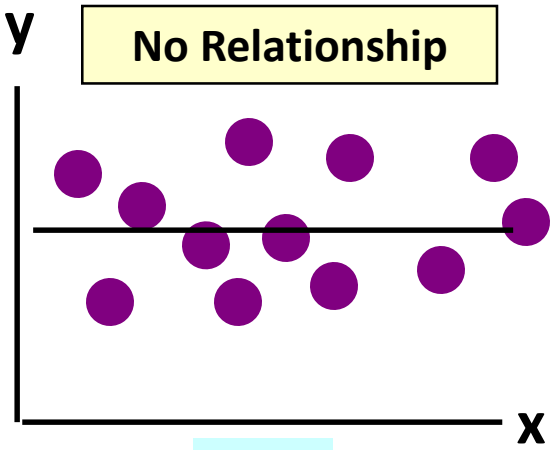
- The Slope
  - $w_1$ measures the estimated change in the average value of Y as a result of a one-unit change in X
    - *For the house price model, $w_1 = 0.10977$ tells us that the average value of a house increases by 0.10977($1000) = $109.77, on average, for each additional one square foot of size*

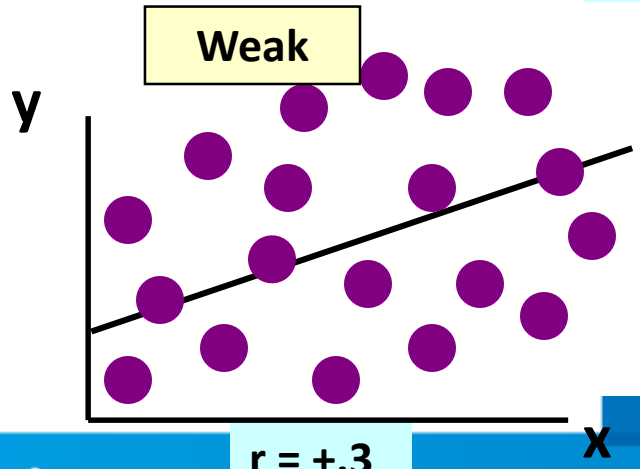
$$\widehat{\text{house price}} = 98.24833 + 0.10977\,(\text{square feet})$$

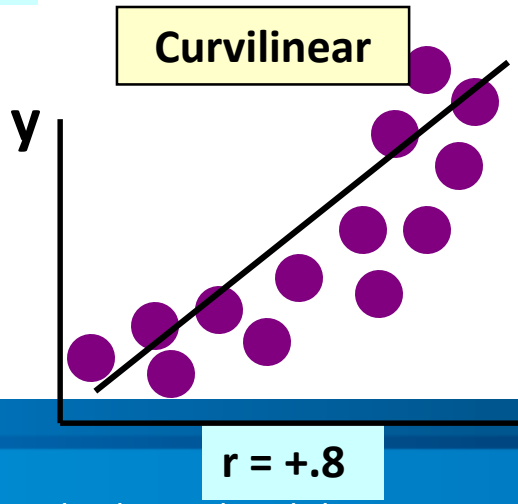
# Correlation and Linear Relationships



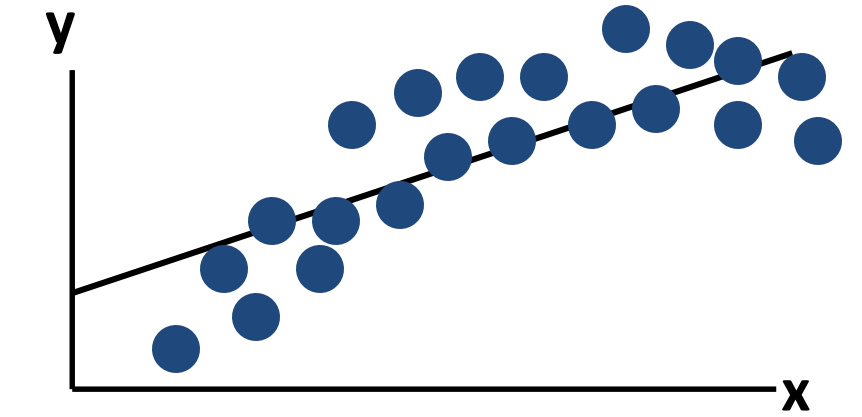



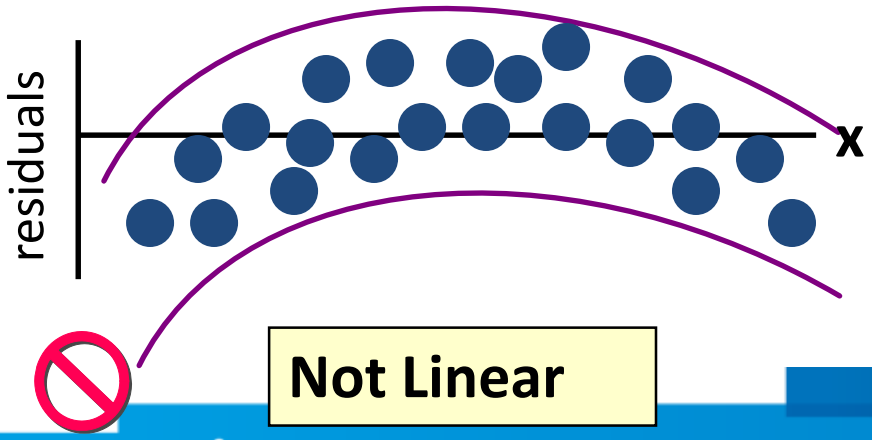
# Residual Analysis

- Purposes
  - Examine for linearity assumption
  - Examine for constant variance for all levels of x
  - Evaluate normal distribution assumption

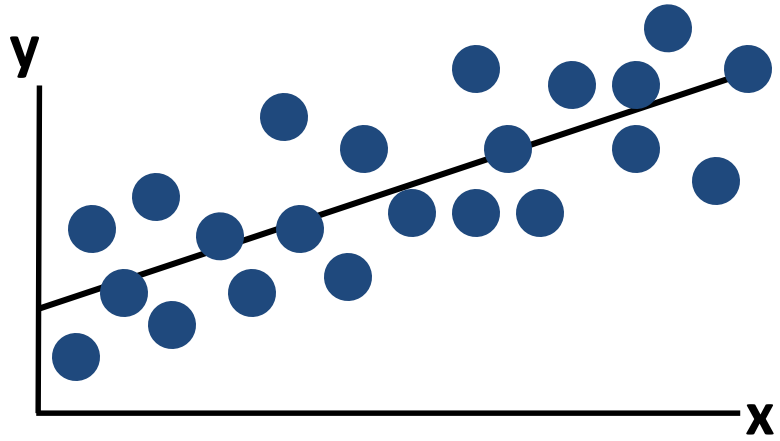
- Graphical Analysis of Residuals
  - Can plot residuals vs. x
  - Can create histogram of residuals to check for normality

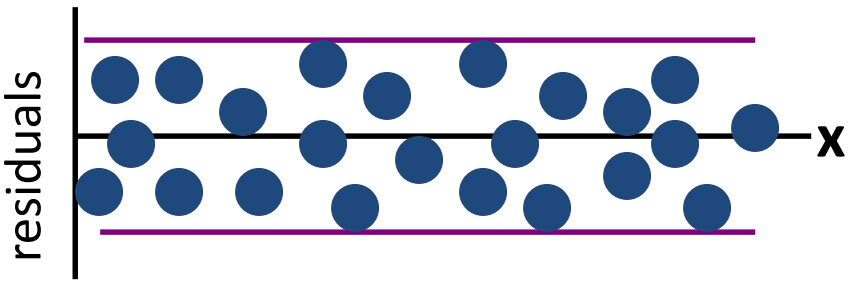
# Residual Analysis for Linearity



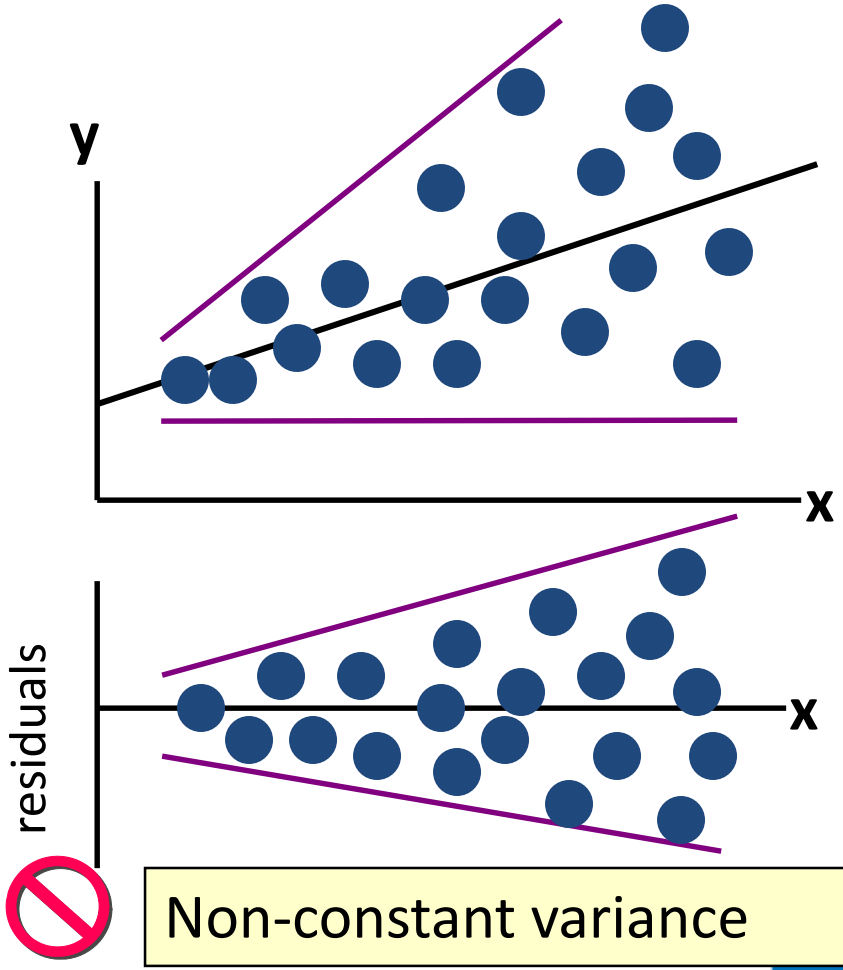

Not Linear



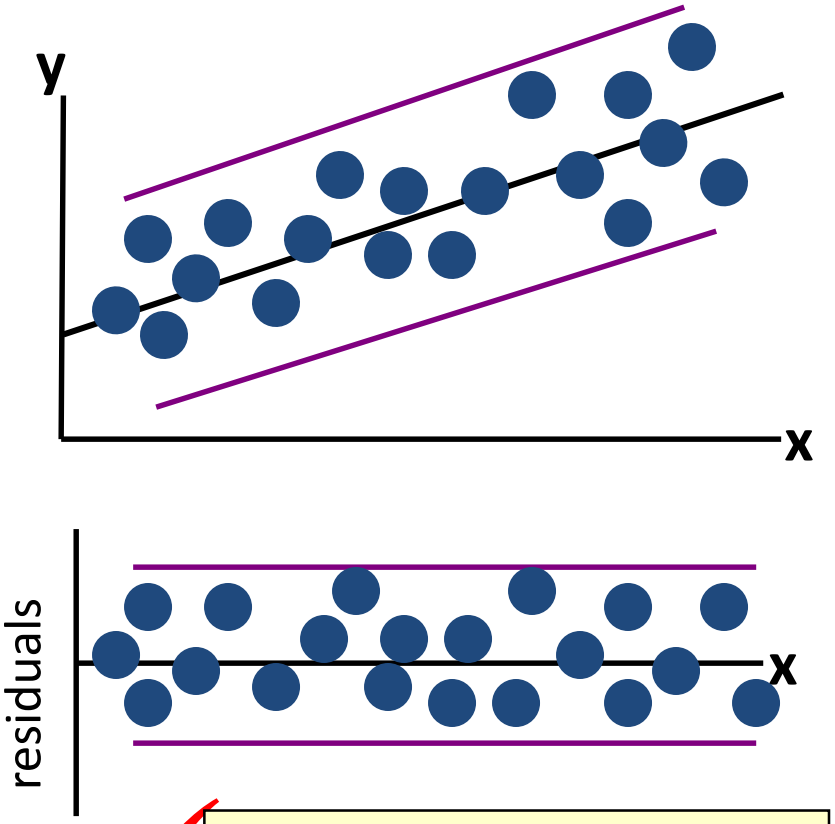




Linear

# Residual Analysis for Constant Variance




Non-constant variance




Constant variance

# Explained and Unexplained Variation

The *Variation* is made up of two parts

$$\sum(y-\bar{y})^2 =$$

$$\sum(y-\hat{y}+\hat{y}-\bar{y})^2 =$$

$$\sum(y-\hat{y})^2+\sum(y-\hat{y})(\hat{y}-\bar{y})+\sum(\hat{y}-\bar{y})^2 =$$

$$\sum(y-\hat{y})^2+\sum\epsilon\hat{y}-\bar{y}\sum\epsilon+\sum(\hat{y}-\bar{y})^2 =$$

$$\sum(y-\hat{y})^2+\sum(\hat{y}-\bar{y})^2$$

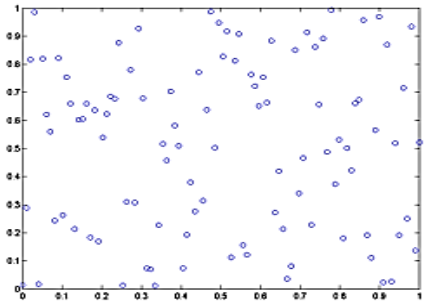
| **SST** | | **SSE** | | **SSR** |
|---|---|---|---|---|
| Total Sum of Squares $\sum(y-\bar{y})^2$ | = | Sum of Squares Error $\sum(y-\hat{y})^2$ | + | Sum of Squares Regression $\sum(\hat{y}-\bar{y})^2$ |
| Total Variation of the $y$ values around their mean $\bar{y}$ | | Unexplained Variation attributed to factors other than the relationship between $x$ and $y$ | | Explained variation attributed to the relationship between x and y |

# Coefficient of Determination, $R^2$

- The *coefficient of determination* (also called *R-squared,* $R^2$ )
  - The portion of the total variation in the dependent variable that is explained by variation in the independent variable

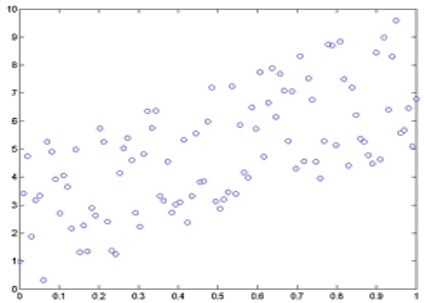
$$R^2 = \frac{\text{SSR}}{\text{SST}} = \frac{\sum(\hat{y} - \bar{y})^2}{\sum(y - \bar{y})^2} = 1 - \frac{SSE}{\text{SST}} = 1 - \frac{\sum(y - \hat{y})^2}{\sum(y - \bar{y})^2}$$

  - Where $0 \leq R^2 \leq 1$, and $R^2 = 1$ if there is a perfect linear relationship



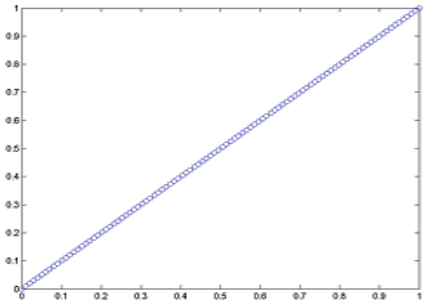
$R^2 = 0$

None of the variation in y is explained by variation in x



$R^2 = 0.5$

Some but not all of the variation in y is explained by variation in x



$R^2 = 1$

100% of the variation in y is explained by variation in x

# Multivariate Linear Regression Model

Examine the linear relationship between one dependent variable (y) and two or more independent variables ($x_i$)

**Population model:**

Y-intercept | Population slopes | Random Error

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_k x_k + \varepsilon$$

---

**Estimated regression model:**

Estimated (or predicted) value of y | Estimated intercept | Estimated slope coefficients

$$\hat{y} = w_0 + w_1 x_1 + w_2 x_2 + \ldots + w_k x_k$$

# Example – House Price

| Square Feet (x1) | Years since built (x2) | Time on market in months (x3) | Number of rooms (x4) | House Price in $1000s (y) |
|---|---|---|---|---|
| 1400 | 25 | 5 | 5 | 245 |
| 1600 | 32 | 3 | 5 | 312 |
| 1700 | 15 | 10 | 7 | 279 |
| 1875 | 5 | 22 | 3 | 308 |
| 1100 | 0 | 1 | 5 | 199 |
| 1550 | 10 | 0 | 2 | 219 |
| 2350 | 154 | 2 | 7 | 405 |
| 2450 | 2 | 3 | 4 | 324 |
| 1425 | 2 | 15 | 5 | 319 |
| 1700 | 22 | 8 | 3 | 255 |

# Agenda

- Linear Regression
- Logistic Regression
- Ensembles Intro / Random Forest
- Kmeans
- T test
- Chi Square

# Logistic Regression

## *The Two-Class Case*

- The optimal decisions are based on the posterior class probabilities $\Pr(y|x)$
- For binary classification problems, we can write these decisions as

$$y = \begin{cases} 1, & \log \frac{\Pr(y=1|x)}{\Pr(y=0|x)} > 0 \\ 0, & \text{Otherwise} \end{cases}$$

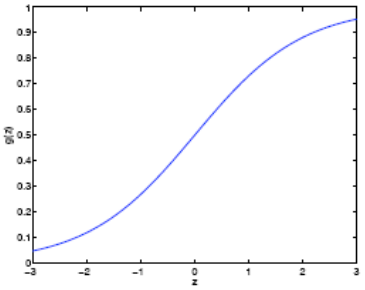
- We generally don't know $\Pr(y|x)$ but we can parameterize the possible decisions according to

$$\log \frac{\Pr(y=1|x)}{\Pr(y=0|x)} = w_0 + x^T w$$

- This log-odds model gives rise to a specific form for the conditional probability over the labels
- The logistic model

$$\Pr(y=1|x, w_0, w) = g(w_0 + x^T w)$$

  - Where $g(z) = (1 + \exp(-z))^{-1}$ Is a *logistic function* that turns linear predictions into probabilities

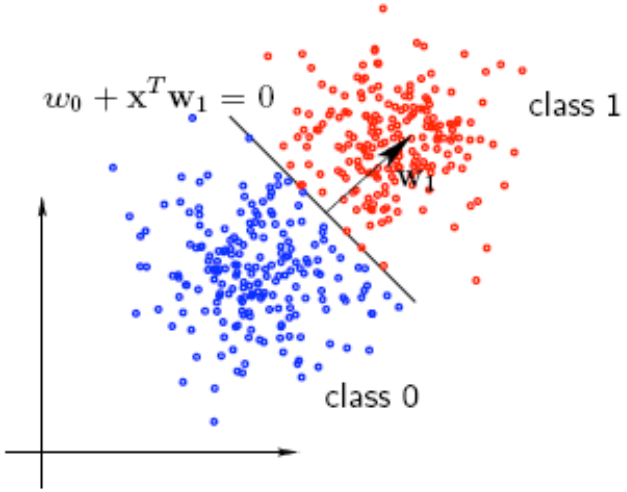


# Two-Class Logistic Regression – Decisions

- Logistic regression models imply a linear decision boundary

$$\log \frac{\Pr(y=1|x)}{\Pr(y=0|x)} = w_0 + x^T w = 0$$

$$\log \frac{P(y=1|\mathbf{x})}{P(y=0|\mathbf{x})} = w_0 + \mathbf{x}^T \mathbf{w}_1 = 0$$



# Fitting Logistic Regression Models

- Denote $\Pr(y_i = 1|x_i, \beta) = p_i = p(x_i; \beta)$, and $\Pr(y_i = 0|x_i, \beta) = 1 - p_i$
- The log-likelihood of $n$ observations

$$l(\beta) = \sum_{i=1}^{n} \{y_i \log p_i + (1 - y_i) \log(1 - p_i)\} = \sum_{i=1}^{n} \{y_i \beta^T x_i - \log(1 + \exp(\beta^T x_i))\}$$

- To maximize the log-likelihood, we set the derivative to zero

$$\frac{\partial}{\partial \beta} l(\beta) = \sum_{i=1}^{n} \{x_i (y_i - \log p_i)\} = 0$$

  - These is a set equations nonlinear in $\beta$

- IRLS (iteratively reweighted least squares)
  - Particularly, for two-class case, solve using Newton-Raphson algorithm

$$\beta^{new} = \beta^{old} + (X^T W X)^{-1} X^T (\boldsymbol{y} - \boldsymbol{p}) = (X^T W X)^{-1} X^T W \boldsymbol{z}$$

  - where
    - $\boldsymbol{z} = X\beta^{old} + W^{-1}(\boldsymbol{y} - \boldsymbol{p})$
    - $W$ is a diagonal $n \times x$ matrix, $W_{i,i} = p_i(1 - p_i)$
    - $\boldsymbol{y}$ is the vector of $y_i$, and $\boldsymbol{p}$ is the vector of $p_i$

# A Few Comments on Logistic Regression

- When it is used
  - Binary responses (two classes)
  - As a data analysis and inference tool to understand the role of the input variables in explaining the outcome
- Feature selection
  - One way is to repeatedly drop the least significant coefficient, and refit the model until no further terms can be dropped
  - Another strategy is to refit each model with one variable removed, and perform an analysis of deviance to decide which one variable to exclude
- Regularization
  - Maximum penalized likelihood $l(\beta) - \frac{C}{2}\|\beta\|^2$
  - Shrinking the parameters via an $L_1$ constraint, imposing a margin constraint in the separable case

# Agenda

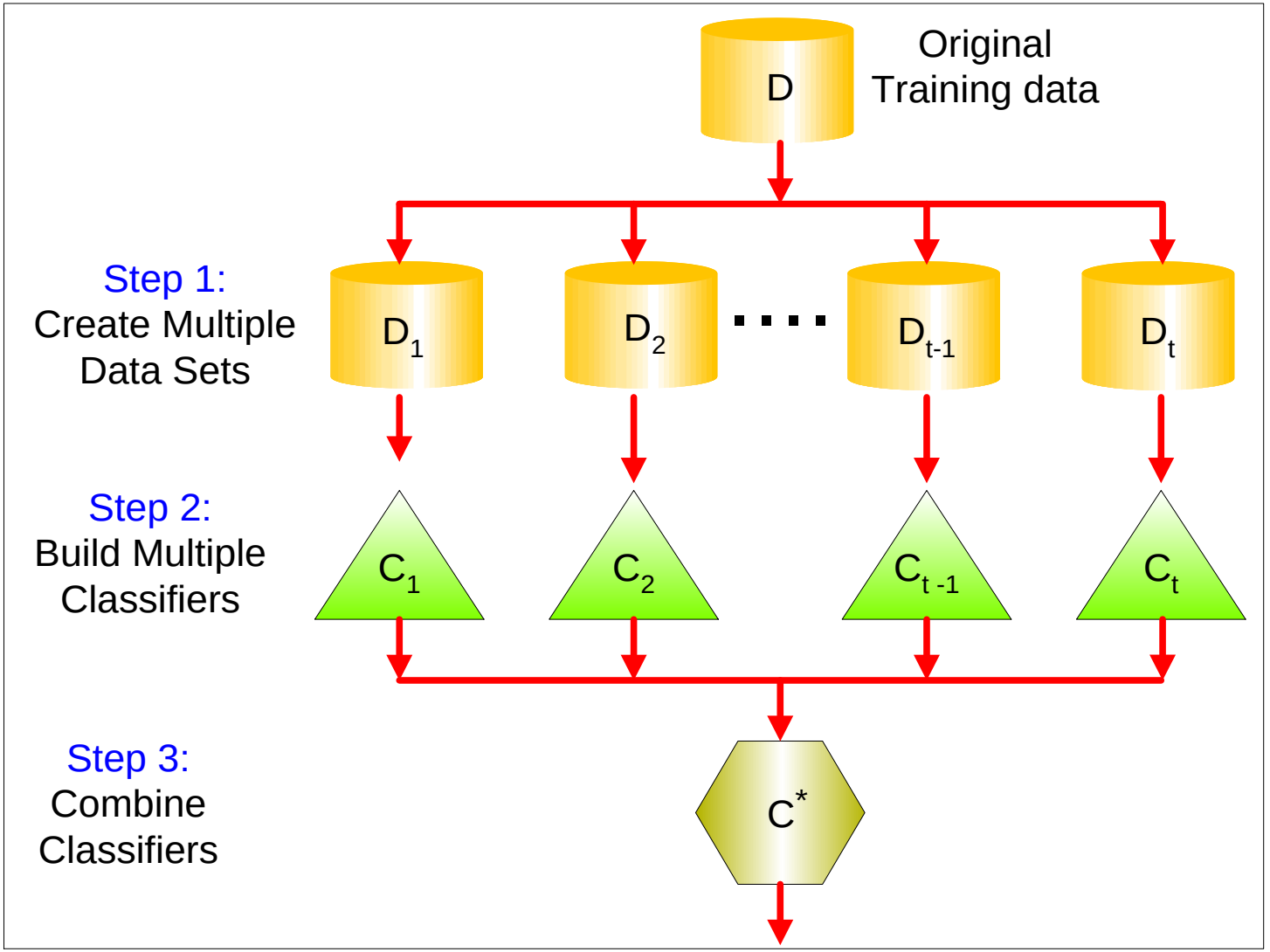
- Linear Regression
- Logistic Regression
- Ensembles Intro / Random Forest
- Kmeans
- T test
- Chi Square

# Ensemble Intro

# Introduction & Motivation

- Suppose that you are a patient with a set of symptoms
- Instead of taking opinion of just one doctor (classifier), you decide to take opinion of a few doctors!
- Is this a good idea? Indeed it is.
- Consult many doctors and then based on their diagnosis, you can get a fairly accurate idea of the diagnosis.
- Majority voting - **'bagging'**
- More weight to the opinion of some 'good' (accurate) doctors - **'boosting'**
- In bagging, you give equal weightage to all classifiers, whereas in boosting you give weightage according to the accuracy of the classifier.

# General Idea



# Ensemble Classifiers (EC)

- An ensemble classifier constructs a set of 'base classifiers' from the training data
- Methods for constructing an EC
  - Manipulating training set
  - Manipulating input features
  - Manipulating class labels
  - Manipulating learning algorithms

# Ensemble Classifiers (EC)

- Manipulating training set
  - Multiple training sets are created by resampling the data according to some sampling distribution
  - Sampling distribution determines how likely it is that an example will be selected for training – may vary from one trial to another
  - Classifier is built from each training set using a particular learning algorithm
  - Examples: Bagging & Boosting

# Ensemble Classifiers (EC)

- Manipulating input features
  - Subset of input features chosen to form each training set
  - Subset can be chosen randomly or based on inputs given by Domain Experts
  - Good for data that has redundant features
  - Random Forest is an example which uses DT as its base classifiers

# Ensemble Classifiers (EC)

- Manipulating class labels
  - When no. of classes is sufficiently large
  - Training data is transformed into a binary class problem by randomly partitioning the class labels into 2 disjoint subsets, A0 & A1
  - Re-labeled examples are used to train a base classifier
  - By repeating the class labeling and model building steps several times, and ensemble of base classifiers is obtained
  - Example – error correcting output coding

# Ensemble Classifiers (EC)

- Manipulating learning algorithm
  - Learning algorithms can be manipulated in such a way that applying the algorithm several times on the same training data may result in different models
  - Example – ANN can produce different models by changing network topology or the initial weights of links between neurons
  - Example – ensemble of DTs can be constructed by introducing randomness into the tree growing procedure – instead of choosing the best split attribute at each node, we randomly choose one of the top k attributes

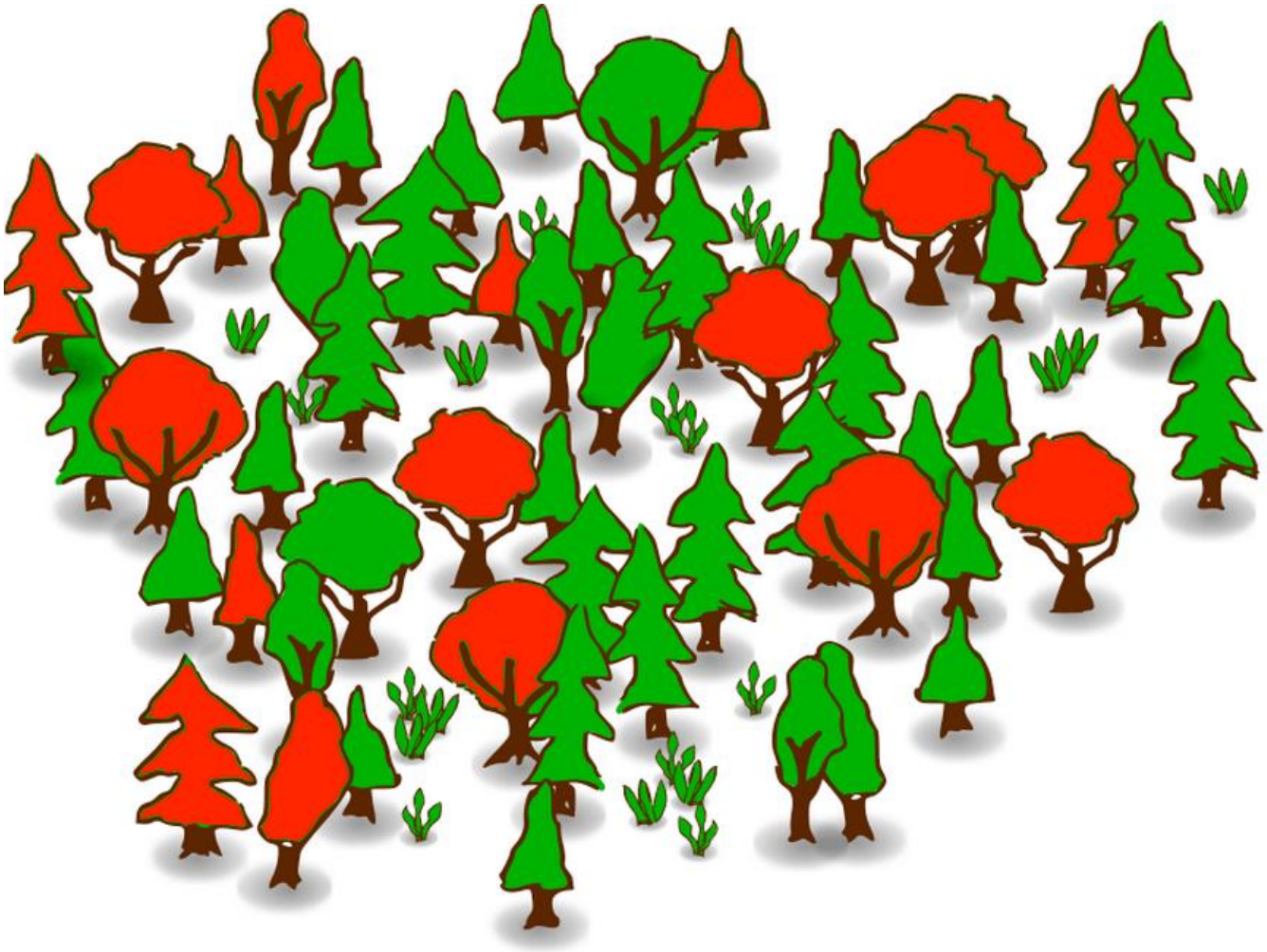
# Ensemble Classifiers

- Ensemble methods work better with 'unstable classifiers'
- Classifiers that are sensitive to minor perturbations in the training set
- Examples:
  - Decision trees
  - Rule-based
  - Artificial neural networks

# Bagging- Final Points

- Works well if the base classifiers are unstable
- Increased accuracy because it reduces the variance of the individual classifier
- Does not focus on any particular instance of the training data
- Therefore, less susceptible to model overfitting when applied to noisy data
- What if we want to focus on a particular instances of training data?

# Random Forest



# Random Forest - Definition

- **Random forest** (or **random forests**) is an ensemble classifier that consists of many decision trees and outputs the class that is the mode of the class's output by individual trees.
- The term came from **random decision forests** that was first proposed by Tin Kam Ho of Bell Labs in 1995.
- The method combines Breiman's "bagging" idea and the random selection of features.

# Algorithm

Each tree is constructed using the following algorithm:

1. Let the number of training cases be *N*, and the number of variables in the classifier be *M*.
2. We are told the number *m* of input variables to be used to determine the decision at a node of the tree; *m* should be much less than *M*.
3. Choose a training set for this tree by choosing *n* times with replacement from all *N* available training cases (i.e. take a bootstrap sample). Use the rest of the cases to estimate the error of the tree, by predicting their classes.
4. For each node of the tree, randomly choose *m* variables on which to base the decision at that node. Calculate the best split based on these *m* variables in the training set.
5. Each tree is fully grown and not pruned (as may be done in constructing a normal tree classifier).

For prediction a new sample is pushed down the tree. It is assigned the label of the training sample in the terminal node it ends up in. This procedure is iterated over all trees in the ensemble, and the average vote of all trees is reported as random forest prediction.

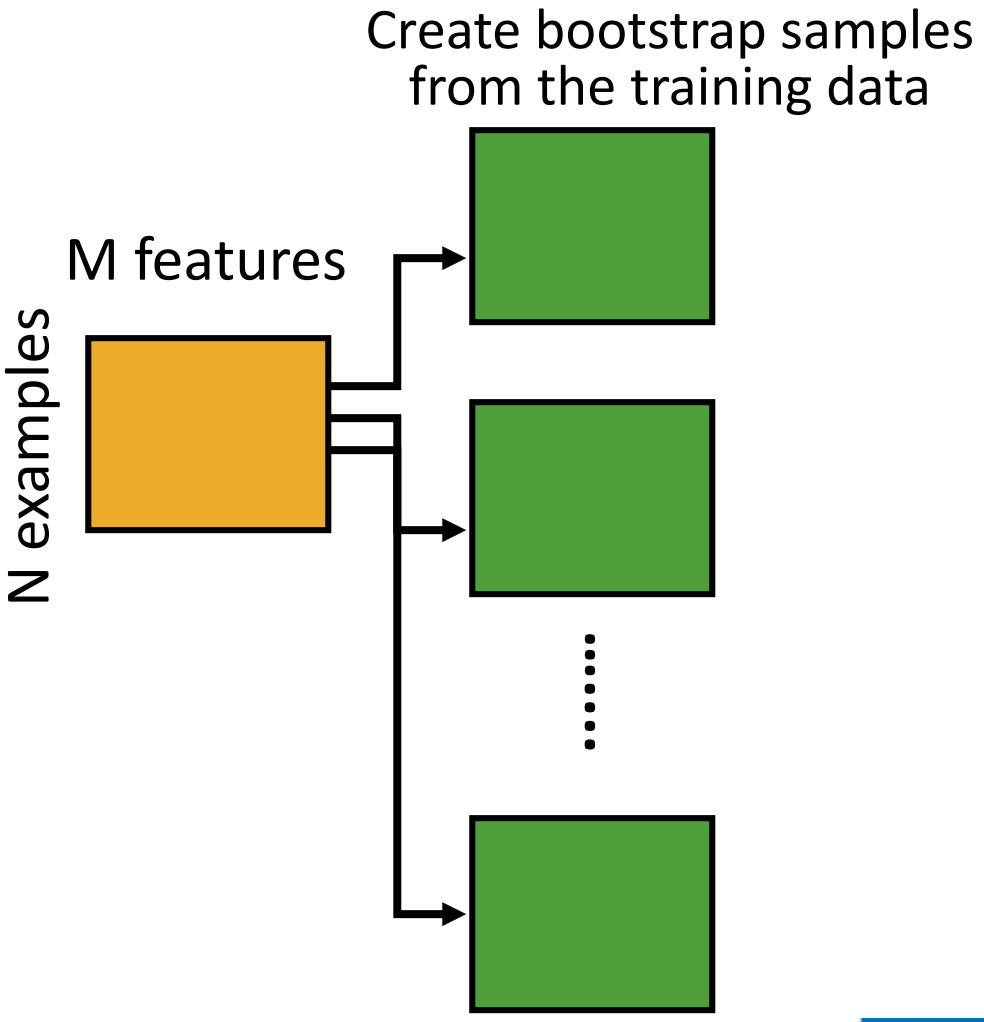
# Random Forest Classifier

**Training Data**

M features

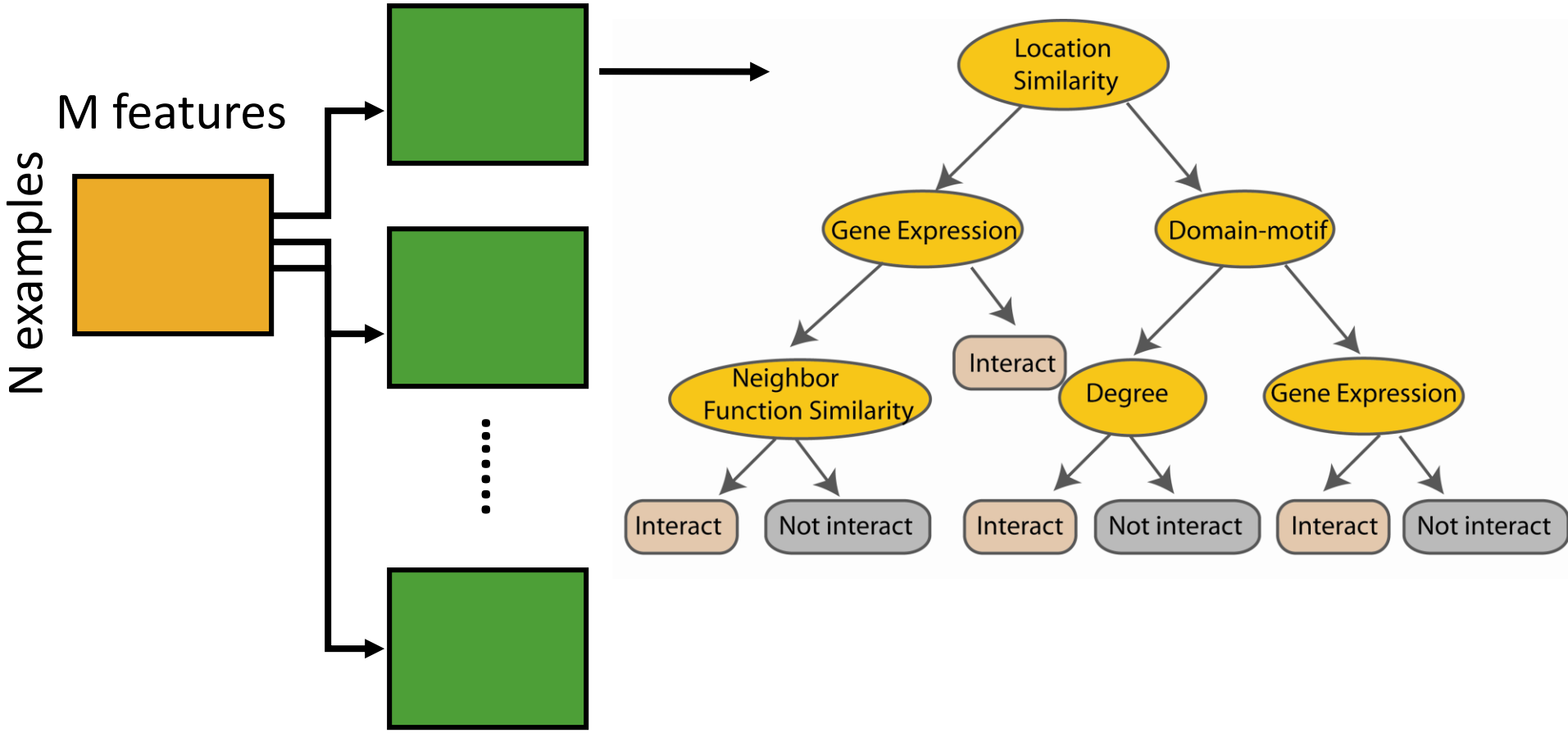
N examples

# Random Forest Classifier



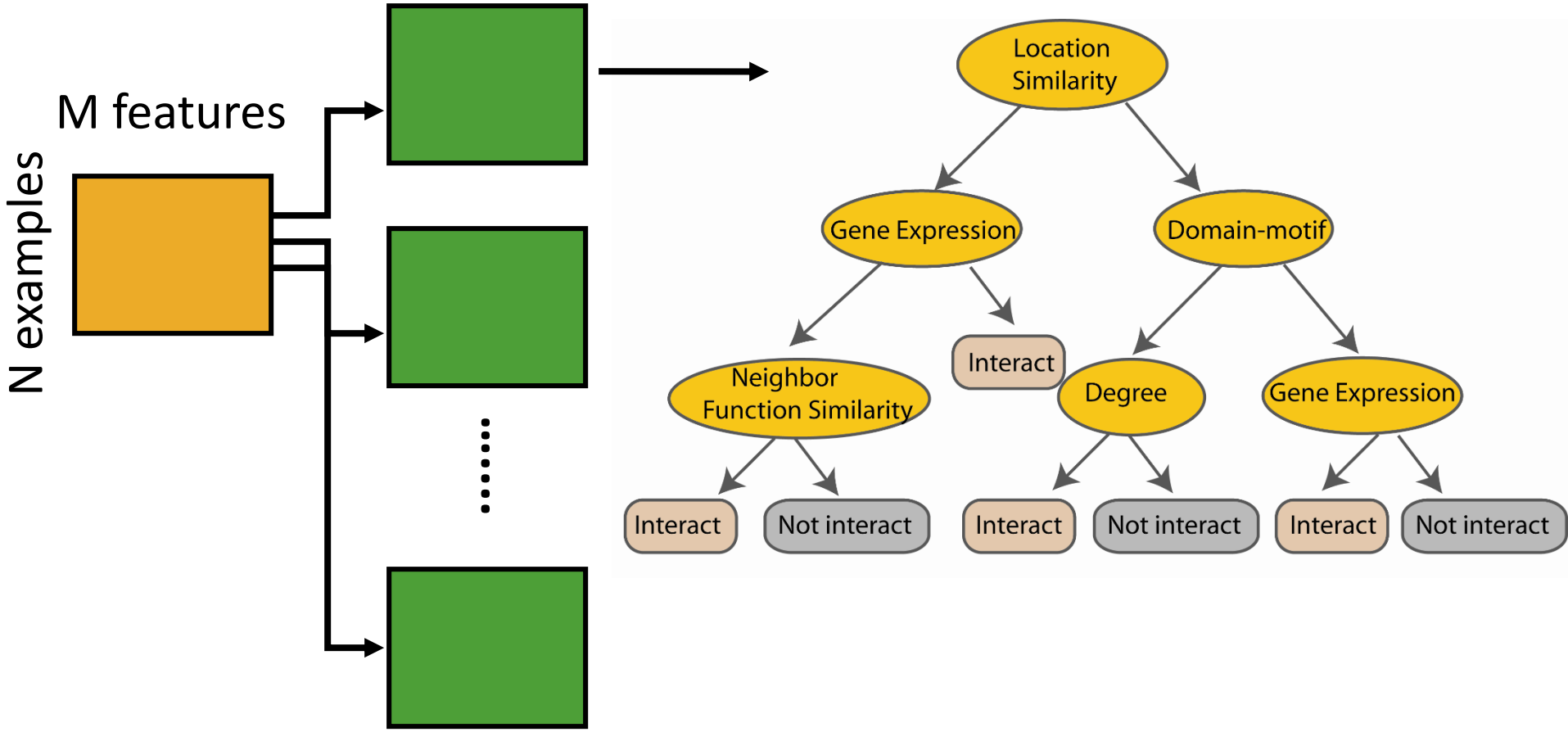
# Random Forest Classifier

Construct a decision tree

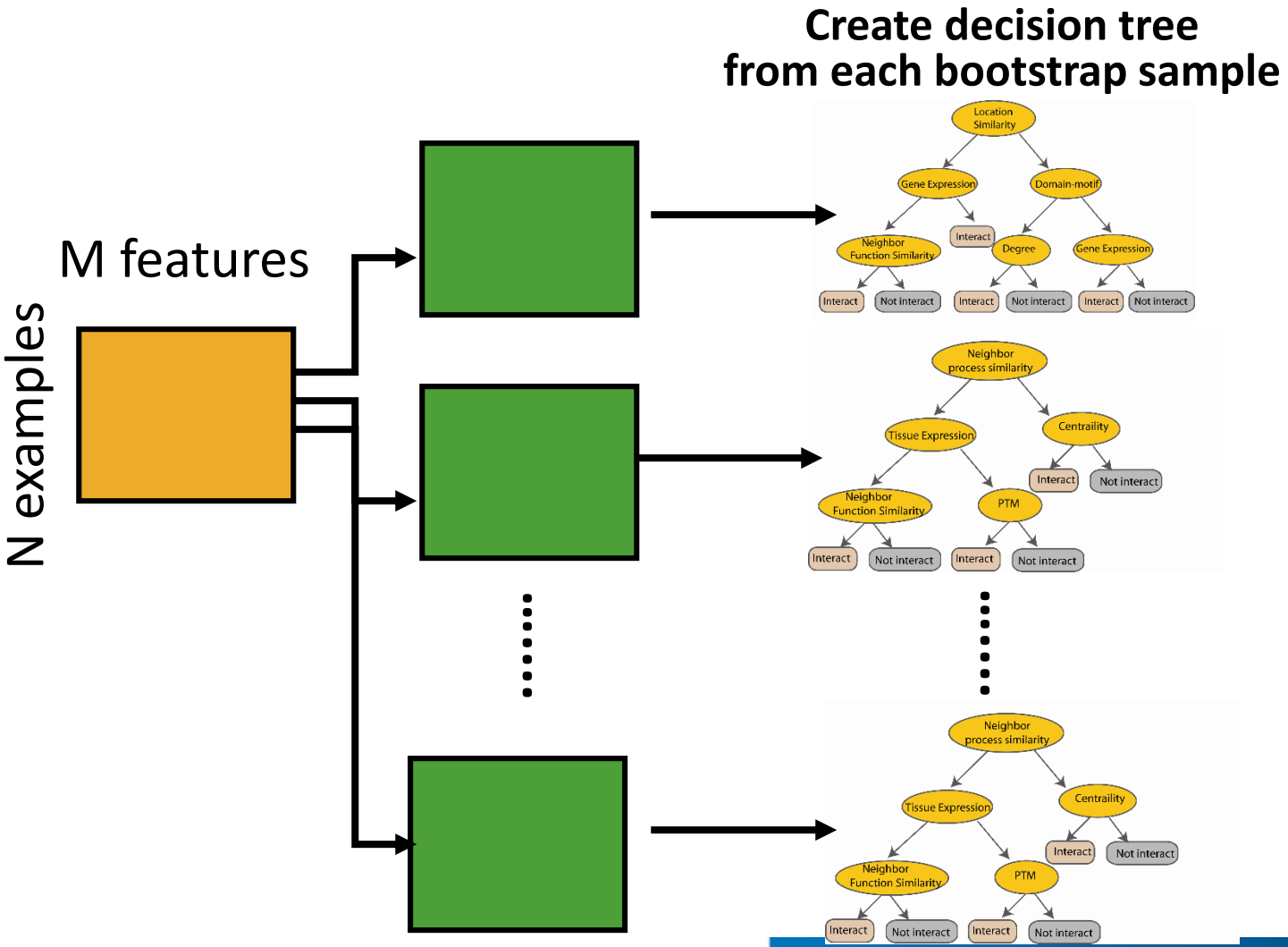


# Random Forest Classifier

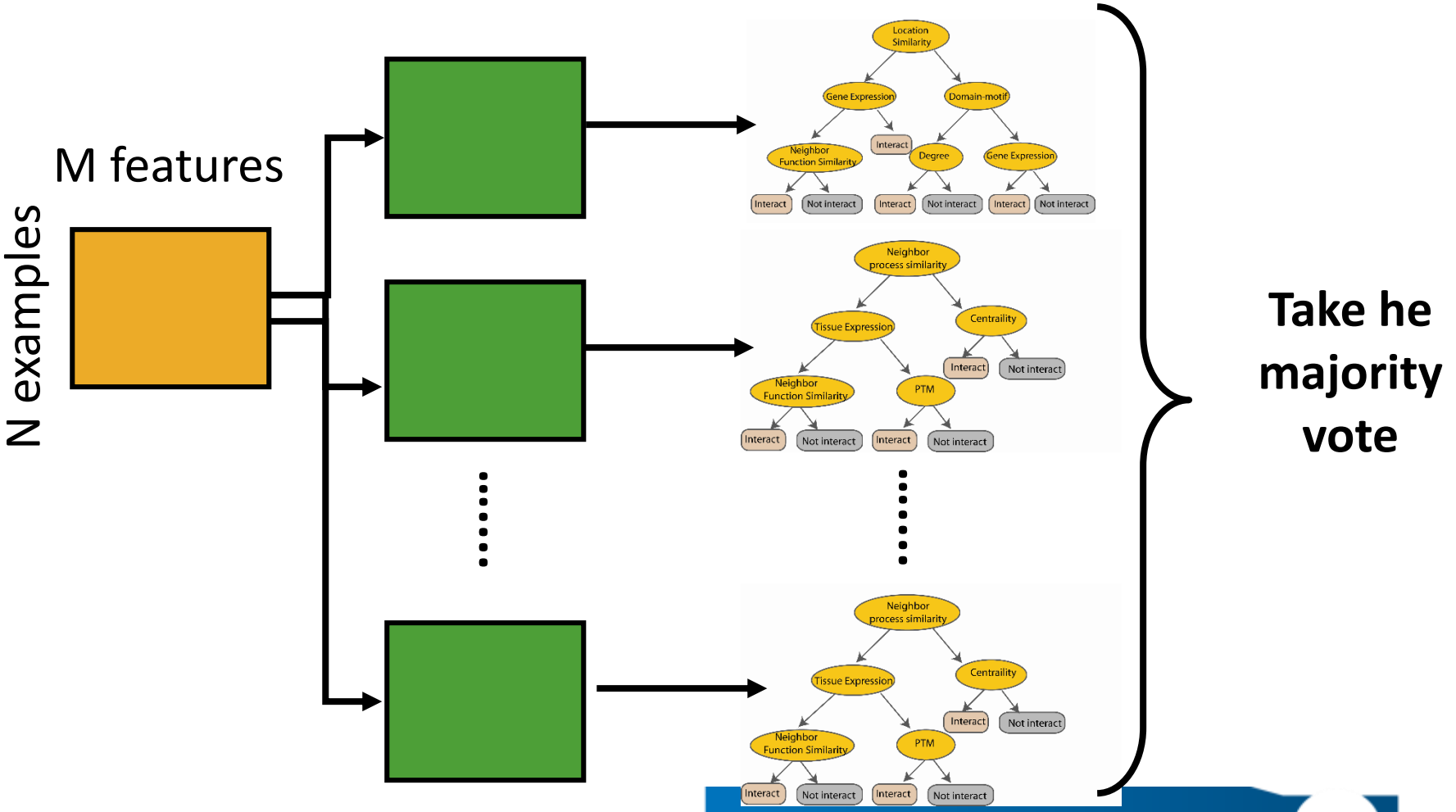
At each node in choosing the split feature
choose only among $m<M$ features



# Random Forest Classifier



# Random Forest Classifier



# Random Forest – practical consideration

- Splits are chosen according to a purity measure:
  - E.g. squared error (regression), Gini index or devinace (classification)
- How to select number of trees?
  - Build trees until the error no longer decreases
- How to select ***m*** number of features to split a node?

# Features and Advantages

The advantages of random forest are:

- It is one of the most accurate learning algorithms available. For many data sets, it produces a highly accurate classifier.
- It runs efficiently on large databases.
- It can handle thousands of input variables without variable deletion.
- It gives estimates of what variables are important in the classification.
- It generates an internal unbiased estimate of the generalization error as the forest building progresses.
- It has an effective method to maintains accuracy when a large proportion of the data are missing.

# Disadvantages

- Random forests have been observed to overfit for some datasets with noisy classification/regression tasks.
- For data including categorical variables with different number of levels, random forests are biased in favor of those attributes with more levels. Therefore, the variable importance scores from random forest are not reliable for this type of data.

# Overfitting

- You can perfectly fit to any training data
- Zero bias, high variance

Two approaches:

- 1. Stop growing the tree when further splitting the data does not yield an improvement
- 2. Grow a full tree, then prune the tree, by eliminating nodes.

# RF - Additional information

Estimating the test error:

- While growing forest, estimate test error from training samples
- For each tree grown, 33-36% of samples are not selected in bootstrap, called out of bootstrap (OOB) samples
- Using OOB samples as input to the corresponding tree, predictions are made as if they were novel test samples
- Through majority vote (classification), average (regression) is computed for all OOB samples from all trees.
- Such estimated test error is very accurate in practice, with reasonable N

# RF - Additional information

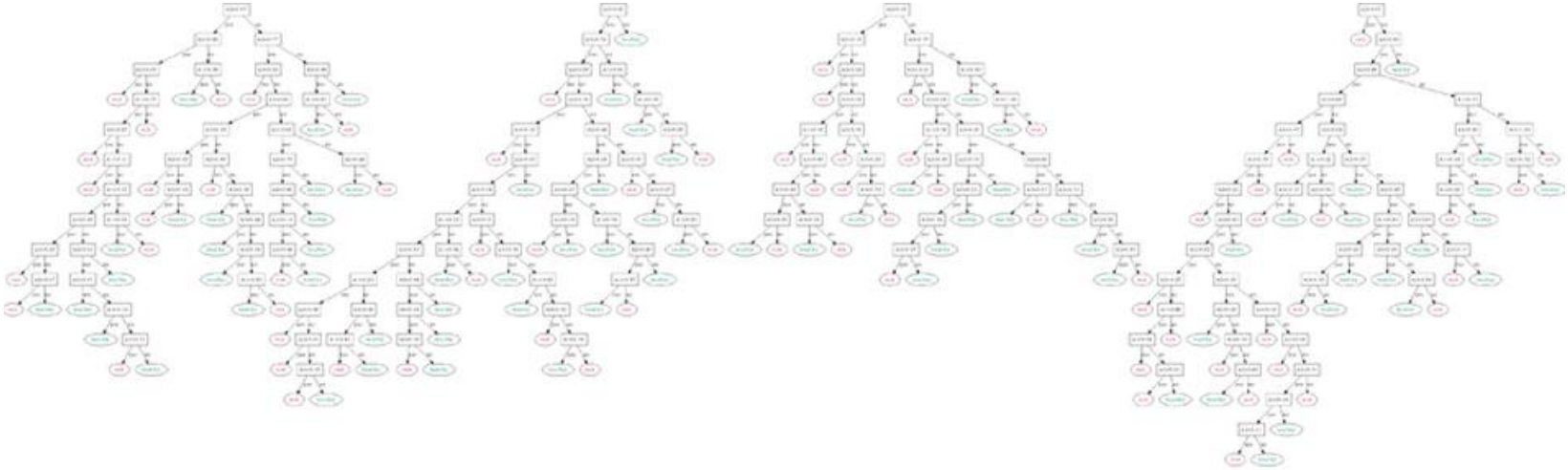
Estimating the importance of each predictor:

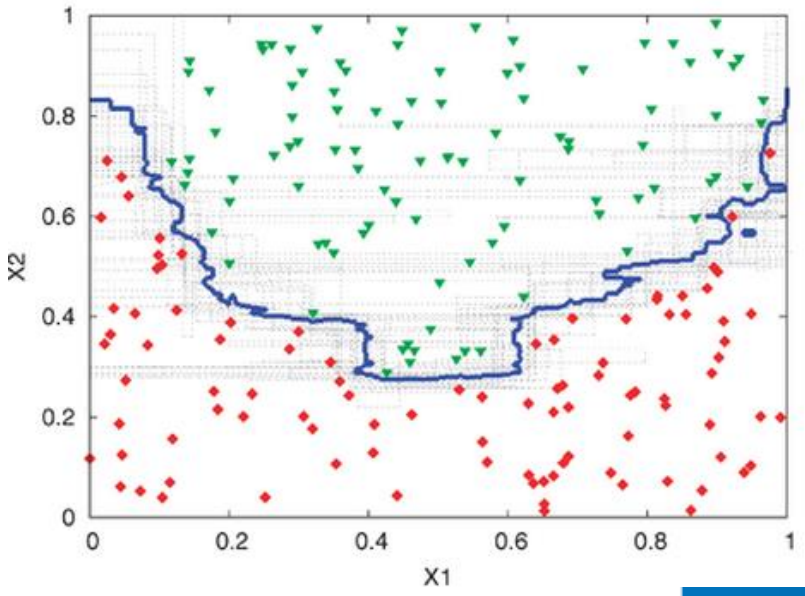
- Denote by ê the OOB estimate of the loss when using original training set, D.
- For each predictor xp where $p \in \{1,..,k\}$
  - Randomly permute $p_{th}$ predictor to generate a new set of samples $D' = \{(y_1,x'_1),\ldots,(y_N,x'_N)\}$
  - Compute OOB estimate $\hat{e}_k$ of prediction error with the new samples
- A measure of importance of predictor xp is êk – ê, the increase in error due to random perturbation of pth predictor

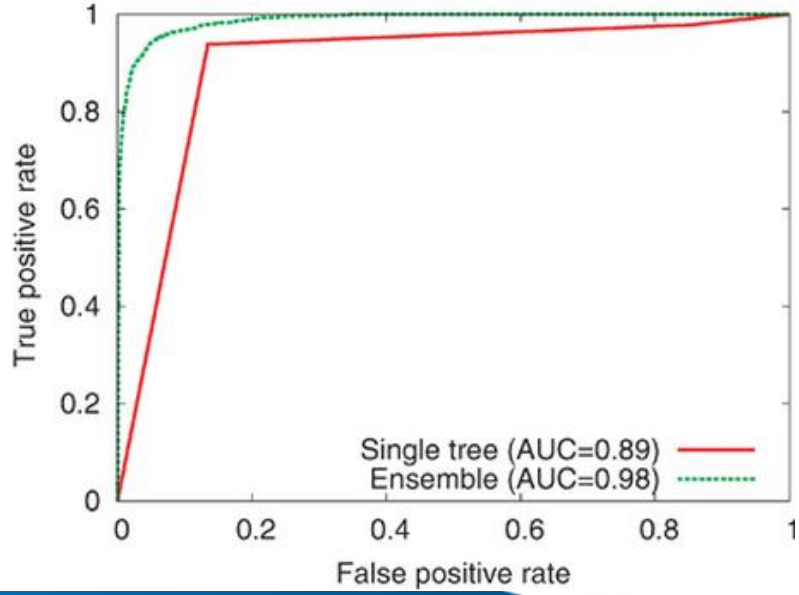
# Conclusions & summary:

- Fast fast fast!
  - RF is fast to build. Even faster to predict!
  - Practically speaking, not requiring cross-validation alone for model selection significantly speeds training by 10x-100x or more.
  - Fully parallelizable … to go even faster!
- Automatic predictor selection from large number of candidates
- Resistance to over training
- Ability to handle data without preprocessing
  - data does not need to be rescaled, transformed, or modified
  - resistant to outliers
  - automatic handling of missing values
- Cluster identification can be used to generate tree-based clusters through sample proximity

# Conclusions & summary:



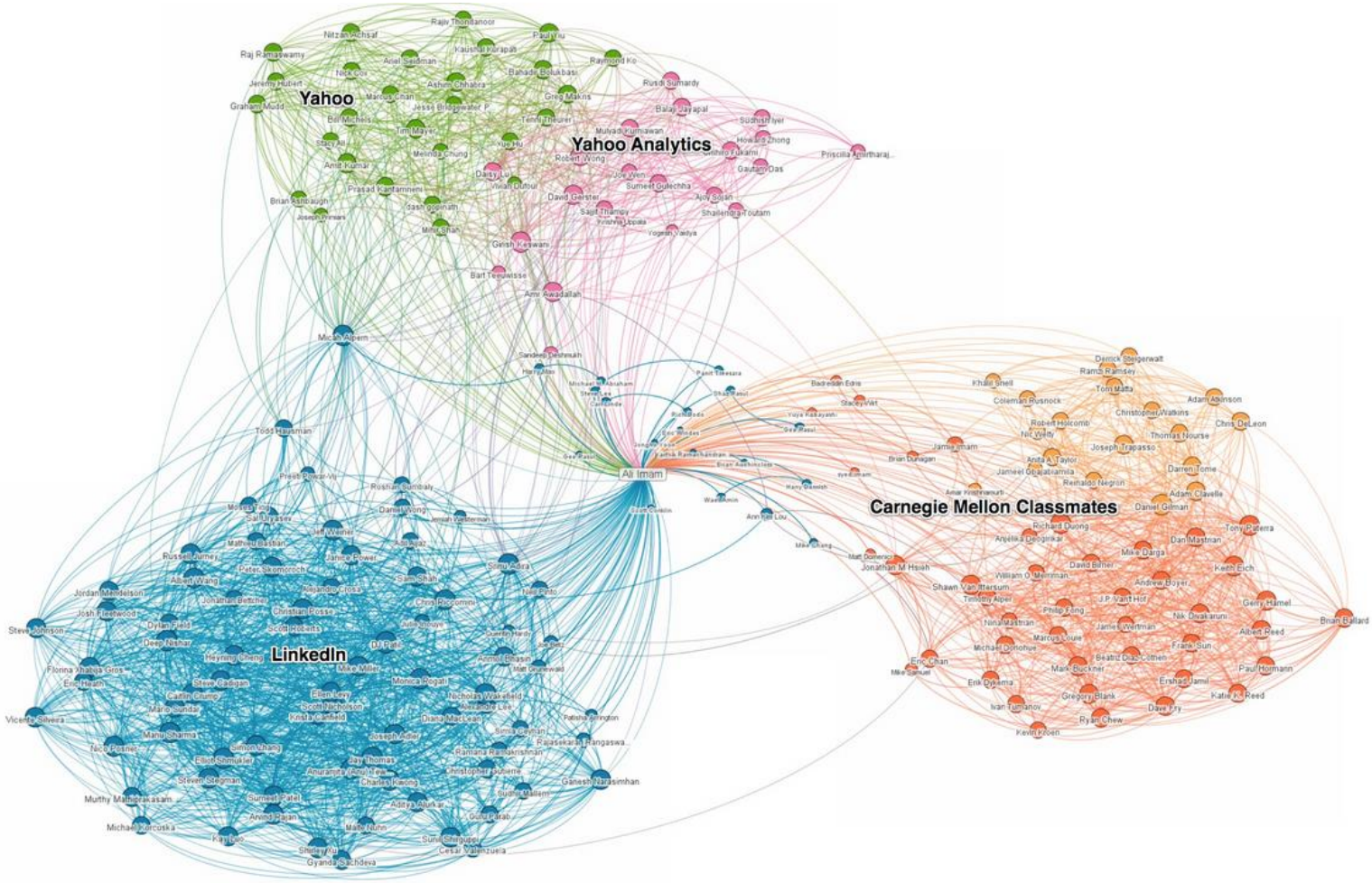


# Agenda

- Linear Regression
- Logistic Regression
- Ensembles Intro / Random Forest
- Kmeans
- T test
- Chi Square

# Clustering - K Means



# What is Clustering?

- Attach label to each observation or data points in a set
- You can say this "unsupervised classification"
- Clustering is alternatively called as "grouping"
- Intuitively, if you would want to assign same label to a data points that are "close" to each other
- Thus, clustering algorithms rely on a distance metric between data points
- Sometimes, it is said that the for clustering, the distance metric is more important than the clustering algorithm

# Distances: Quantitative Variables

Some examples

Identity (absolute) error

$$d_j(x_{ij}, x_{i'j}) = I(x_{ij} \neq x_{i'j})$$

Squared distance

$$d_j(x_{ij}, x_{i'j}) = (x_{ij} - x_{i'j})^2$$

$L_q$ norms

$$L_{qii'} = \left[\sum_j |x_{ij} - x_{i'j}|^q\right]^{1/q}$$

Canberra distance

$$d_{ii'} = \sum_j \frac{|x_{ij} - x_{i'j}|}{|x_{ij} + x_{i'j}|}$$

Data point:

$x_i = [x_{i1} \ldots x_{ip}]^T$

# Distances: Ordinal and Categorical Variables

- Ordinal variables can be forced to lie within (0, 1) and then a quantitative metric can be applied:

- For categorical variables, distances must be specified by user between each pair of categories.

- Often weighted sum is used:

$$D(x_i, x_j) = \sum_{l=1}^{p} w_l d(x_{il}, x_{jl}), \quad \sum_{l=1}^{p} w_l = 1, \; w_l > 0.$$

# *K*-means Overview

- An unsupervised clustering algorithm
- "*K*" stands for number of clusters, it is typically a user input to the algorithm; some criteria can be used to automatically estimate *K*
- It is an approximation to an NP-hard combinatorial optimization problem
- *K*-means algorithm is iterative in nature
- It converges, however only a local minimum is obtained
- Works only for numerical data
- Easy to implement

# *K*-means Assumptiopns

- K is known
- The Data is sampled from K spherical clusters

# K-means: Setup

- $x_1, \ldots, x_N$ are data points or vectors of observations
- Each observation (vector $x_i$) will be assigned to one and only one cluster
- $C(i)$ denotes cluster number for the $i^{th}$ observation
- Dissimilarity measure: Euclidean distance metric
- *K*-means minimizes within-cluster point scatter:

$$W(C) = \frac{1}{2}\sum_{k=1}^{K}\sum_{C(i)=k}\sum_{C(j)=k}\left\|x_i - x_j\right\|^2 = \sum_{k=1}^{K} N_k \sum_{C(i)=k}\left\|x_i - m_k\right\|^2$$

$m_k$ is the mean vector of the $k^{th}$ cluster

$N_k$ is the number of observations in $k^{th}$ cluster

# K-means Algorithm

- For a given cluster assignment *C* of the data points, compute the cluster means $m_k$:

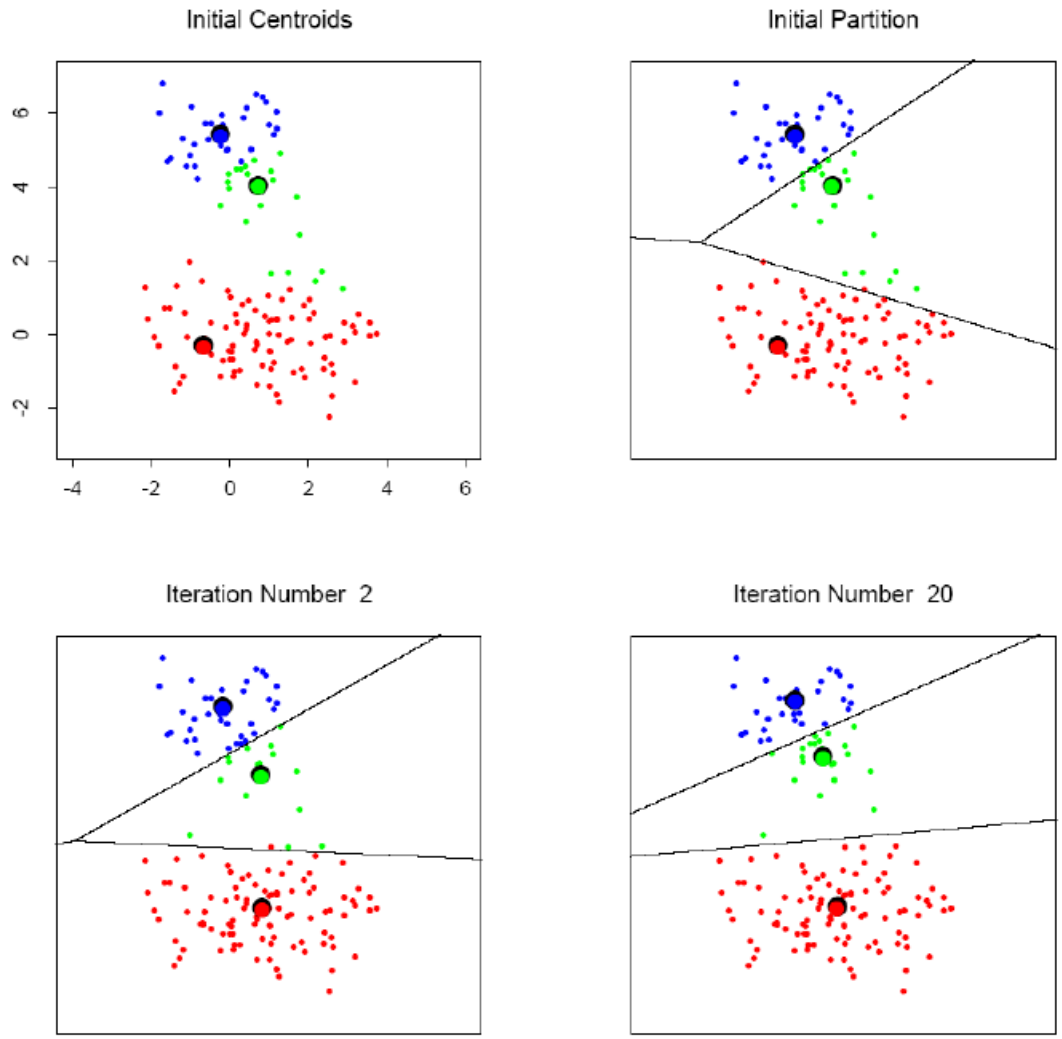
$$m_k = \frac{\sum_{i:C(i)=k} x_i}{N_k},\ k = 1,\ldots,K.$$

- For a current set of cluster means, assign each observation as:

$$C(i) = \arg\min_{1\le k\le K} \|x_i - m_k\|^2,\ i = 1,\ldots,N$$

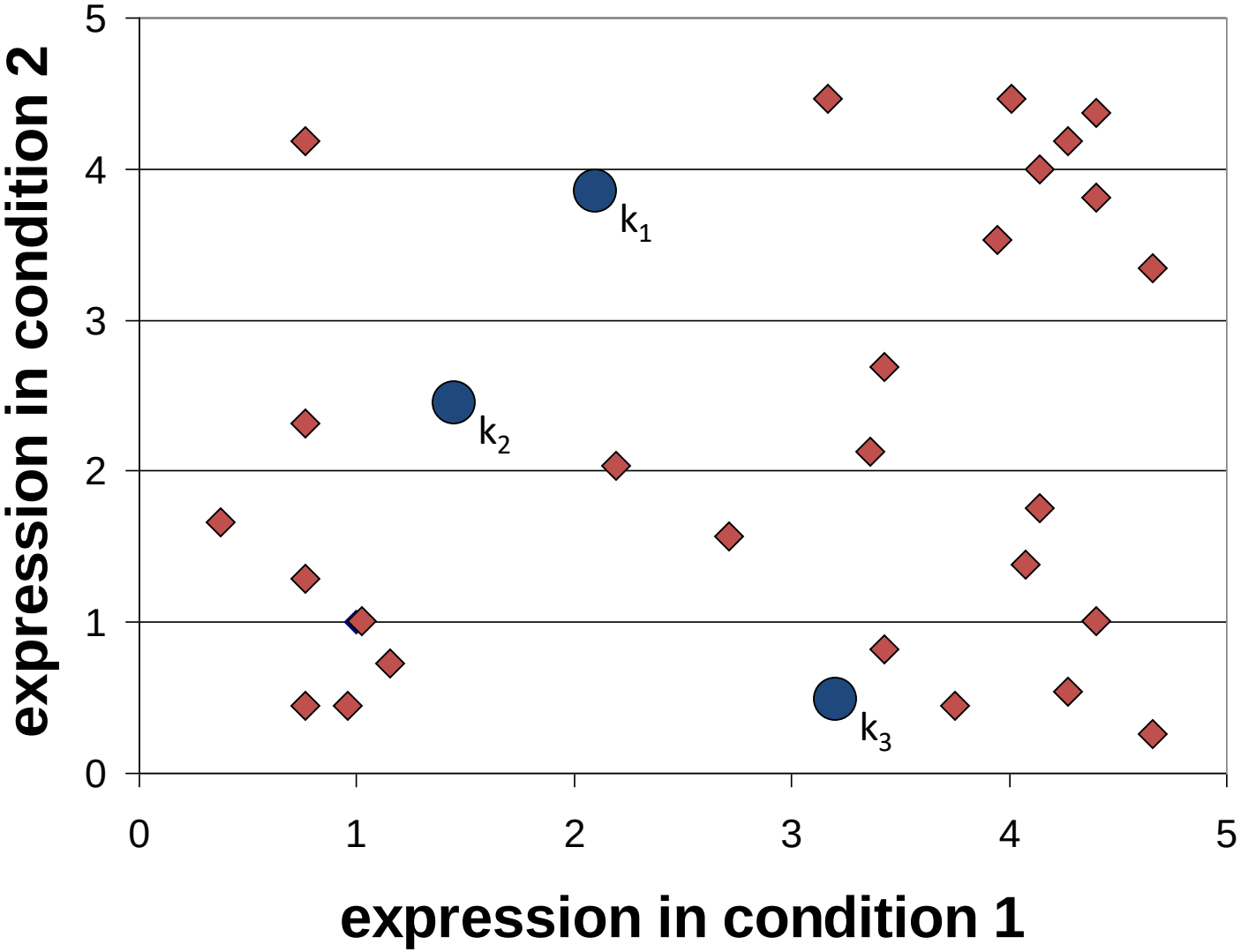
- Iterate above two steps until convergence

# K-means clustering example



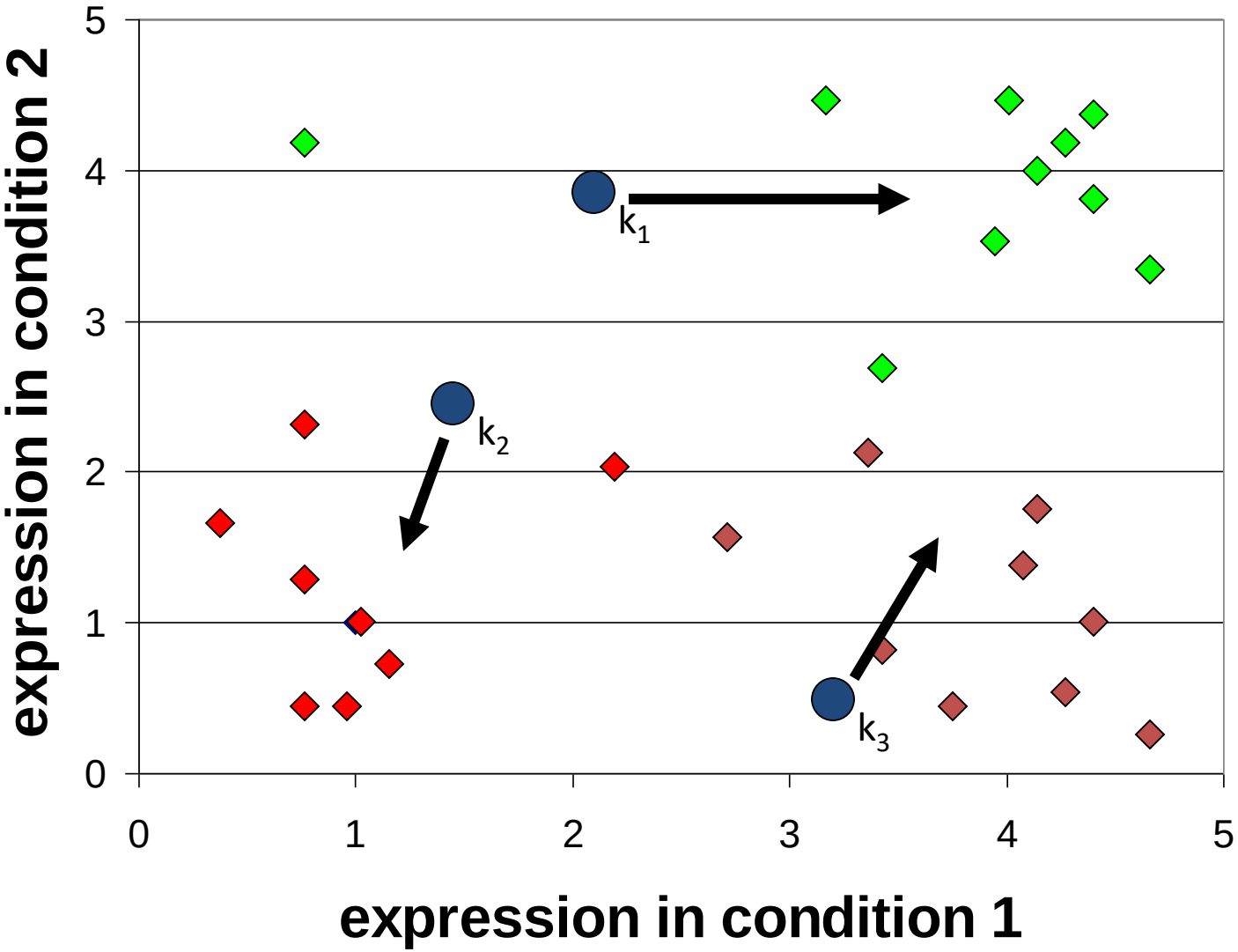
# Clustering: Example 2, Step 1

Algorithm: k-means, Distance Metric: Euclidean Distance



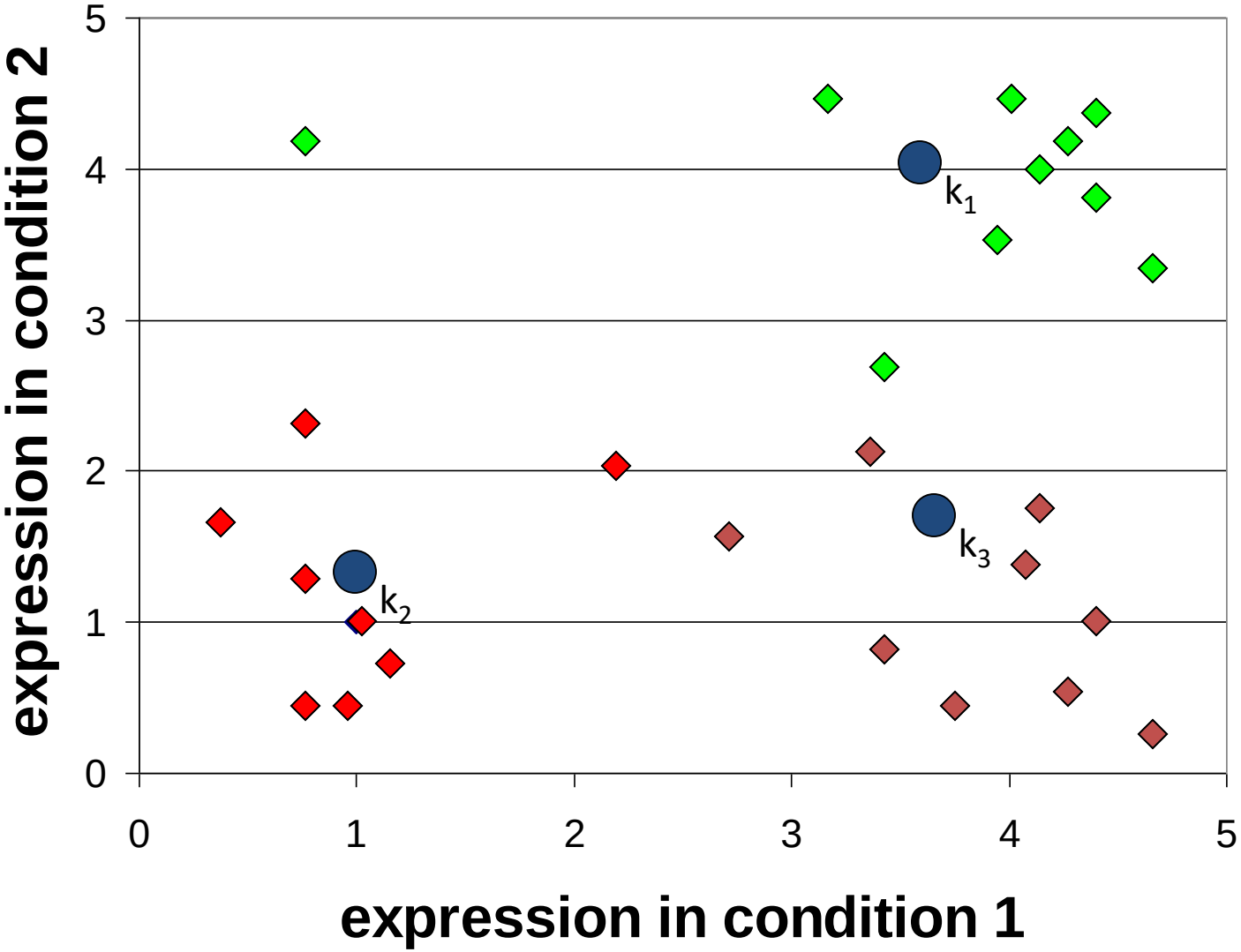
# Clustering: Example 2, Step 2

Algorithm: k-means, Distance Metric: Euclidean Distance



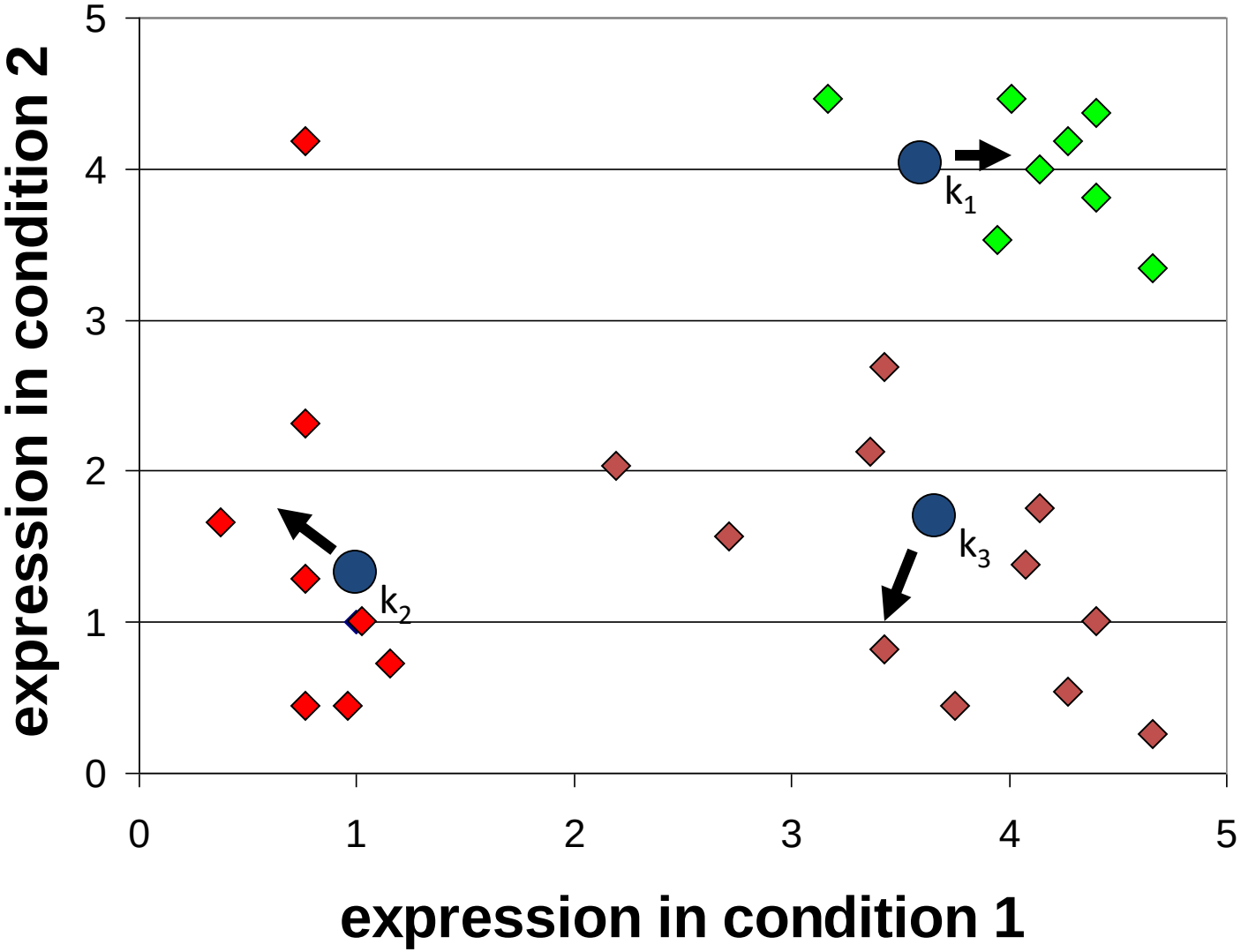
# Clustering: Example 2, Step 3

Algorithm: k-means, Distance Metric: Euclidean Distance



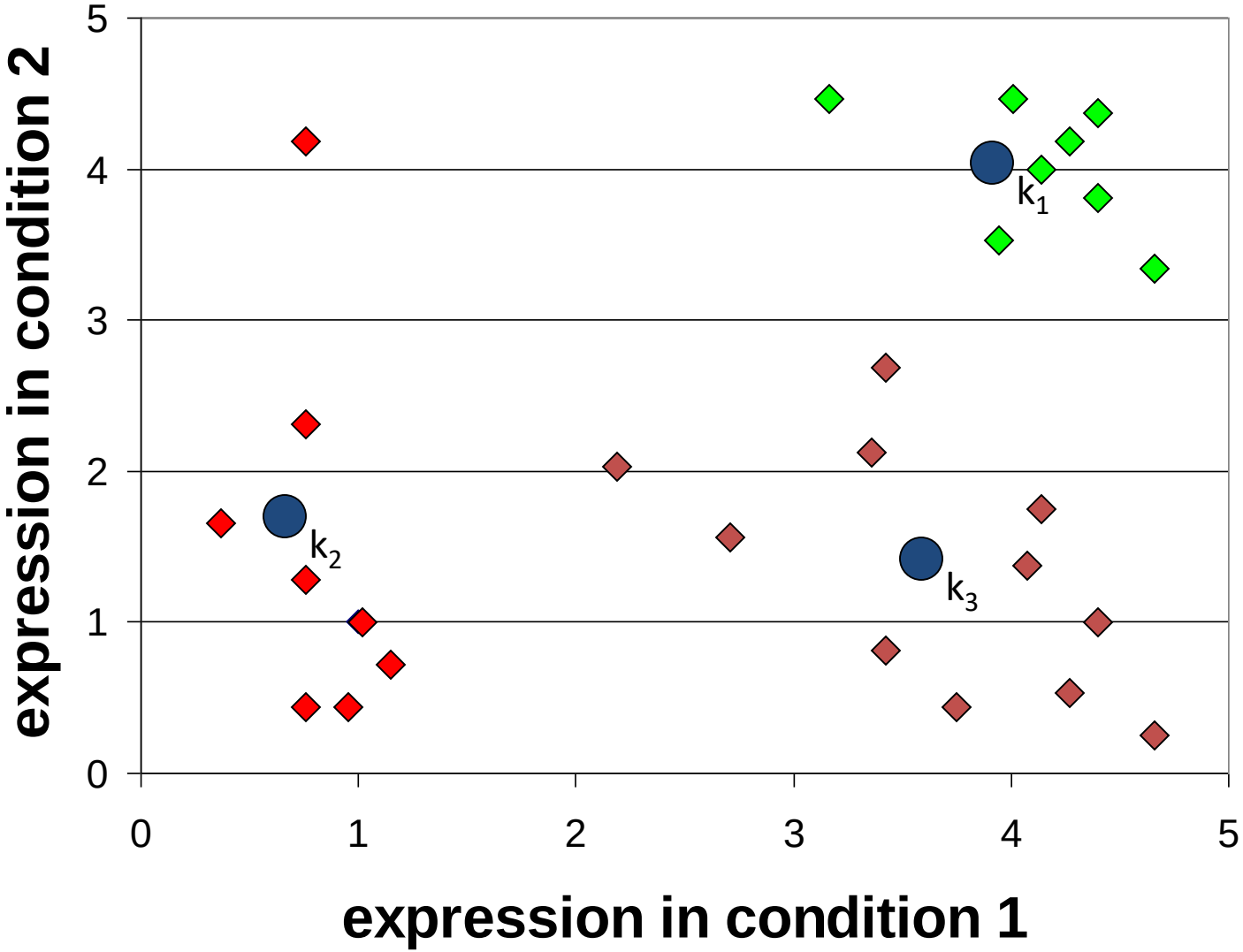
# Clustering: Example 2, Step 4

Algorithm: k-means, Distance Metric: Euclidean Distance



# Clustering: Example 2, Step 5

Algorithm: k-means, Distance Metric: Euclidean Distance



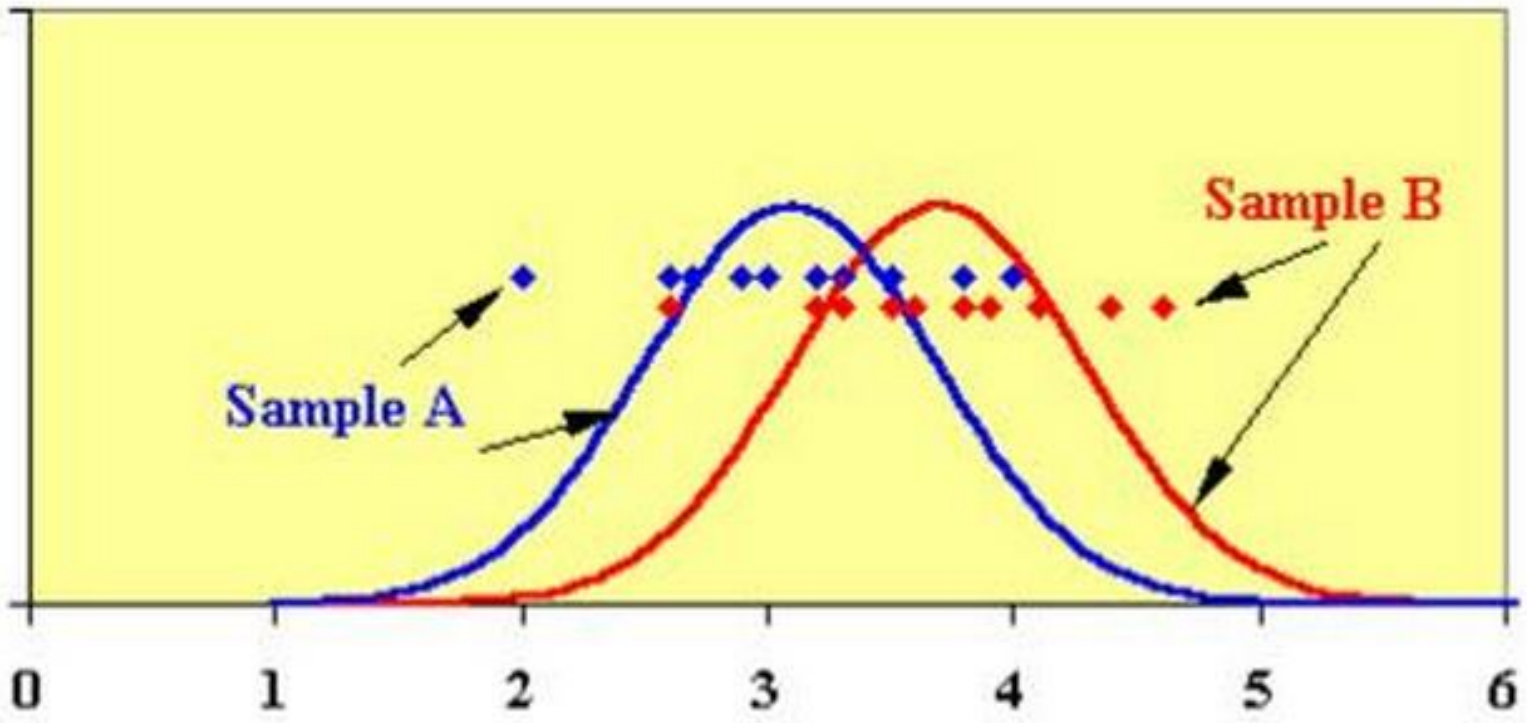
# *K*-means: summary

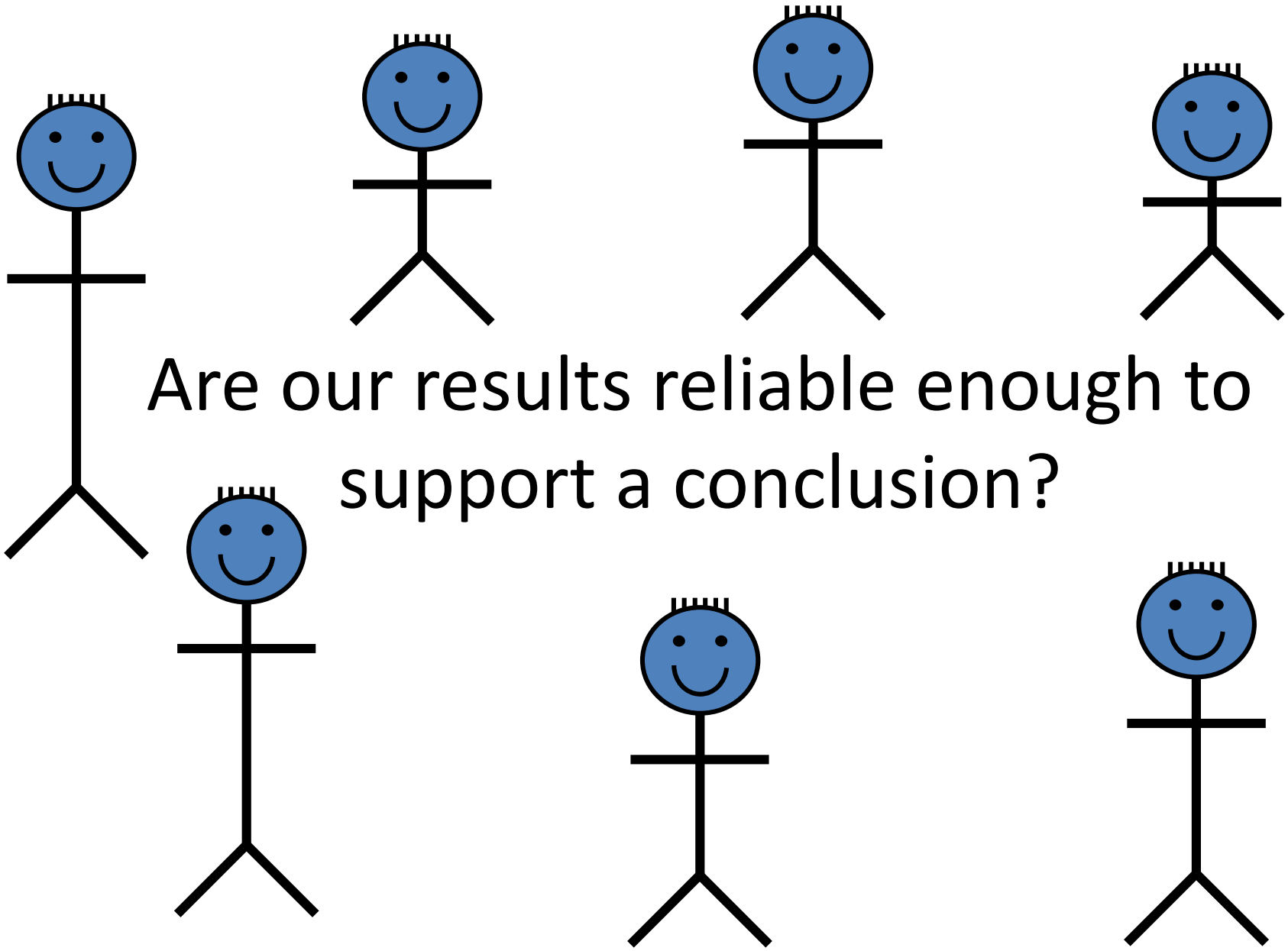
- Algorithmically, very simple to implement
- *K*-means converges, but it finds a local minimum of the cost function
- Works only for numerical observations
- *K* is a user input; alternatively BIC (Bayesian information criterion) or MDL (minimum description length) can be used to estimate *K*
- Outliers can considerable trouble to *K*-means

# Agenda

- Linear Regression
- Logistic Regression
- Ensembles Intro / Random Forest
- Kmeans
- T test
- Chi Square

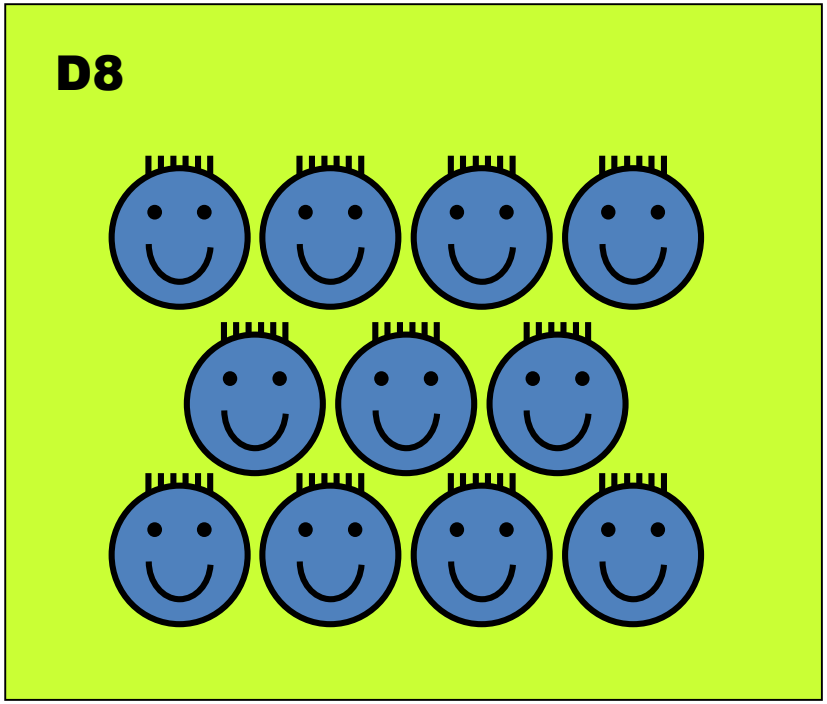
# T test

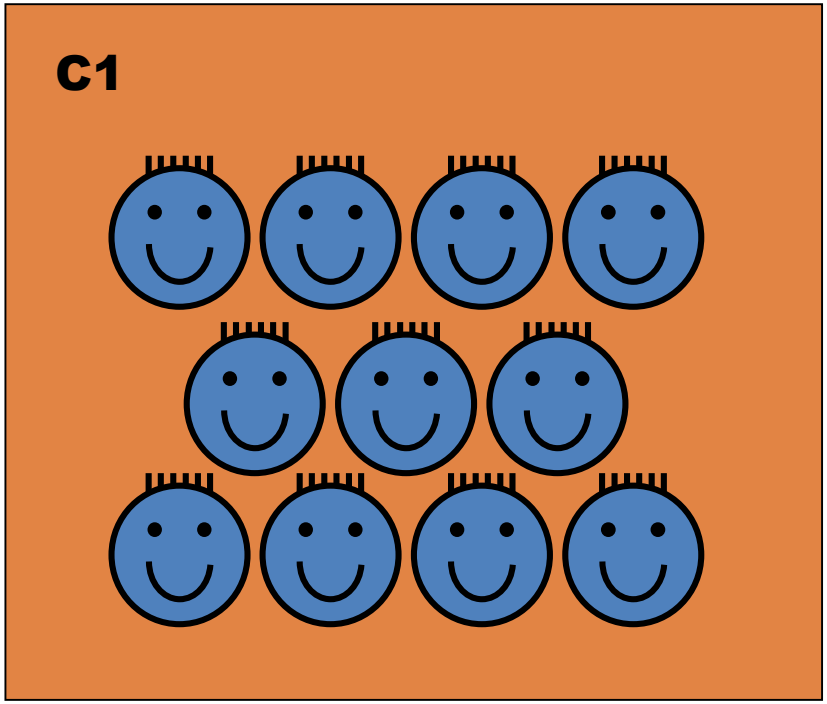




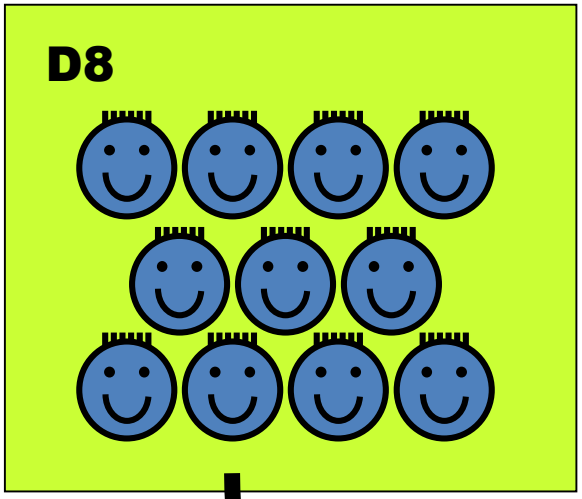
Are our results reliable enough to support a conclusion?

Imagine we chose two children at random from two class rooms…

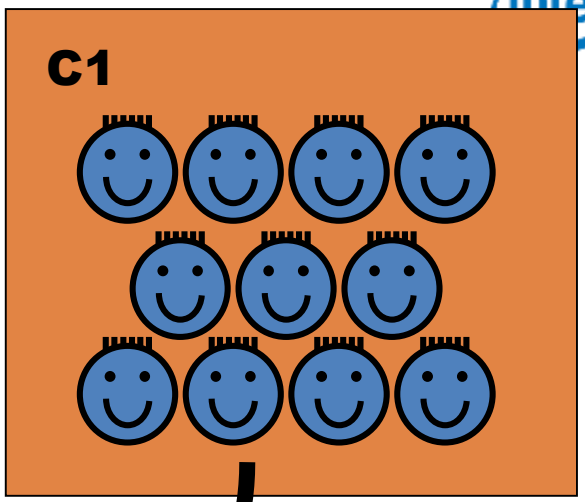


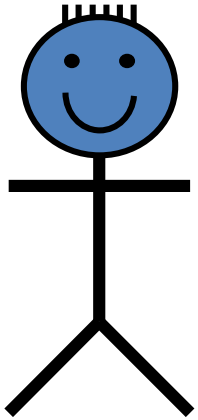



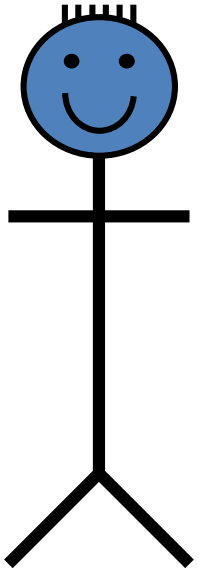

… and compare their height …




… we find that one pupil is taller than the other

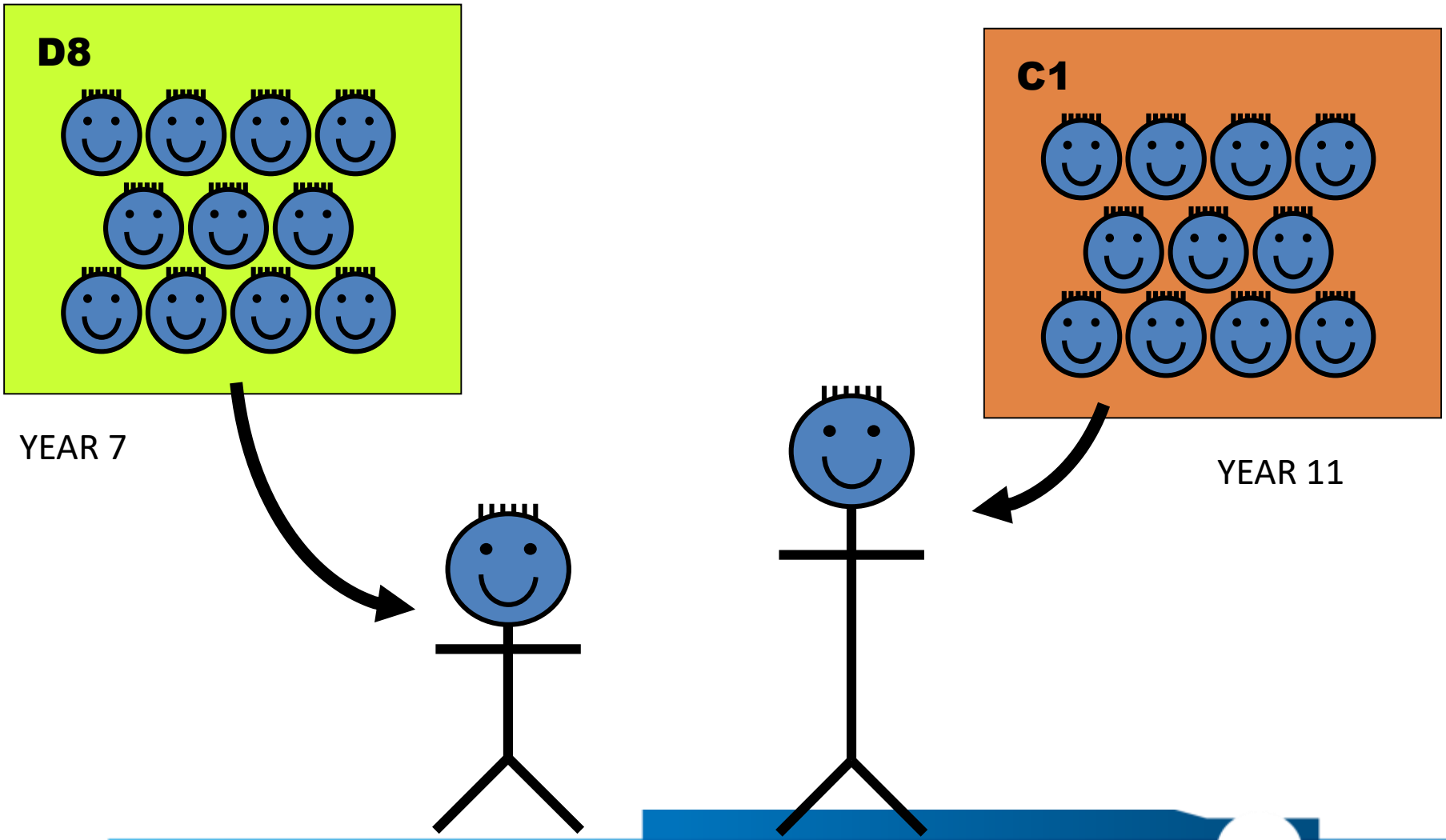


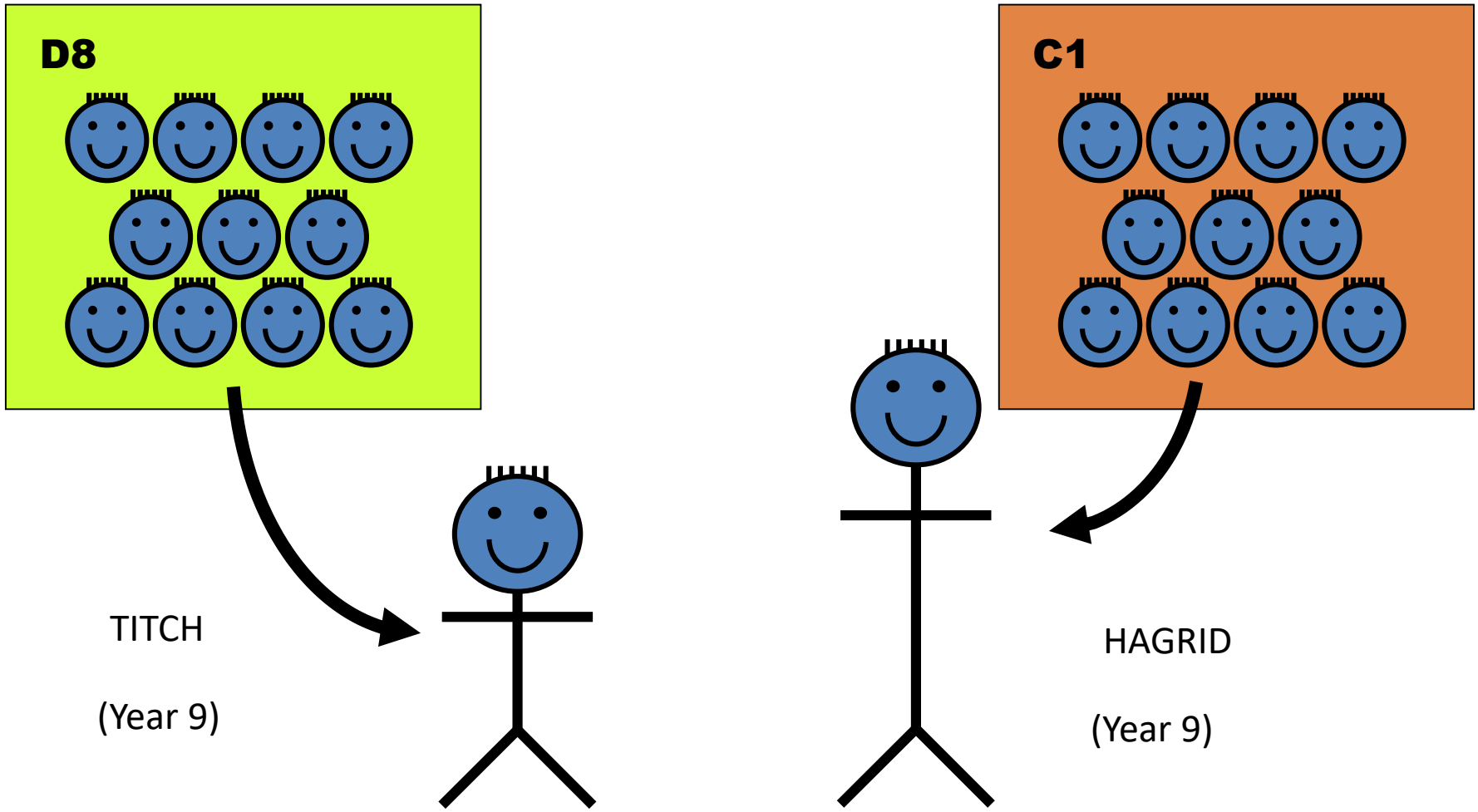


WHY?

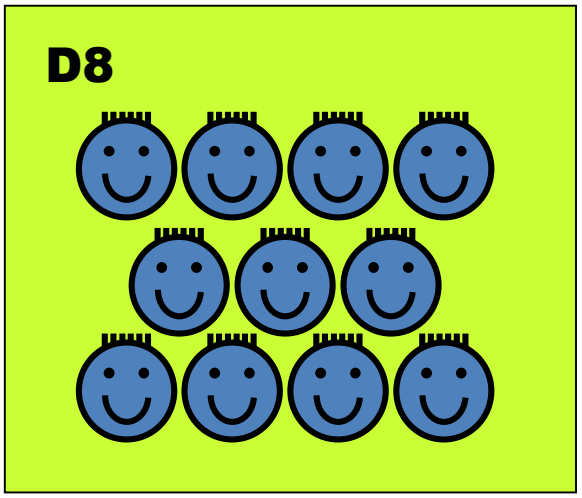
REASON 1: There is a significant difference between the two groups, so pupils in C1 are taller than pupils in D8



REASON 2: By chance, we picked a short pupil from D8 and a tall one from C1

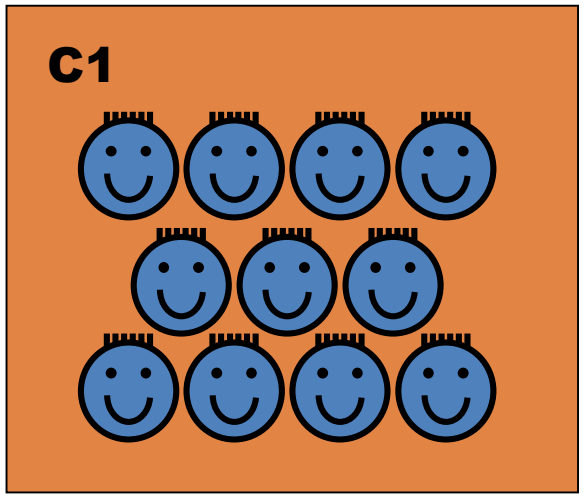


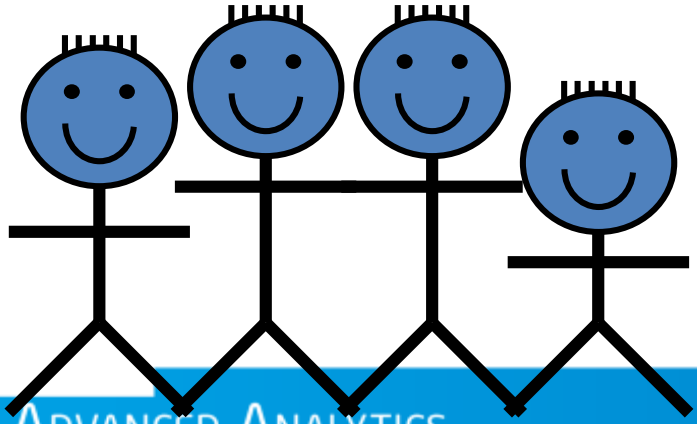
If there is a significant difference between the two groups…

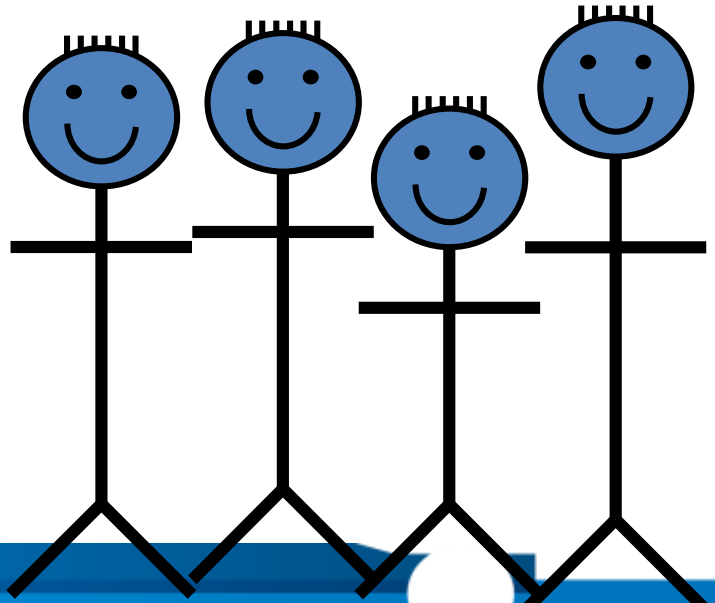



… the average or mean height of the two groups should be very…

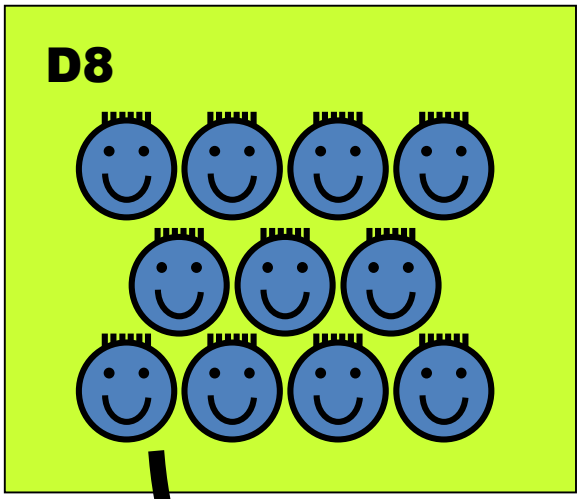
… DIFFERENT



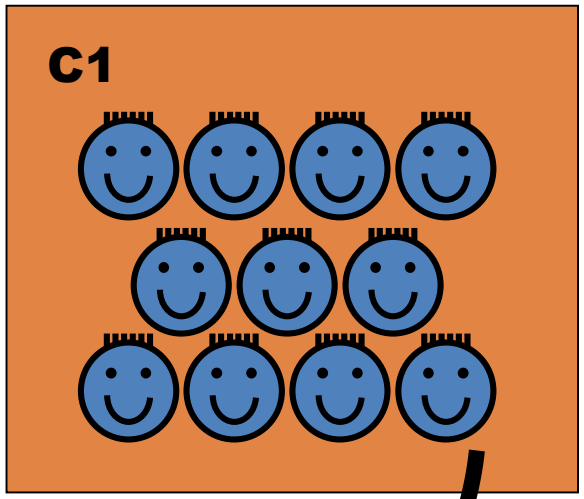


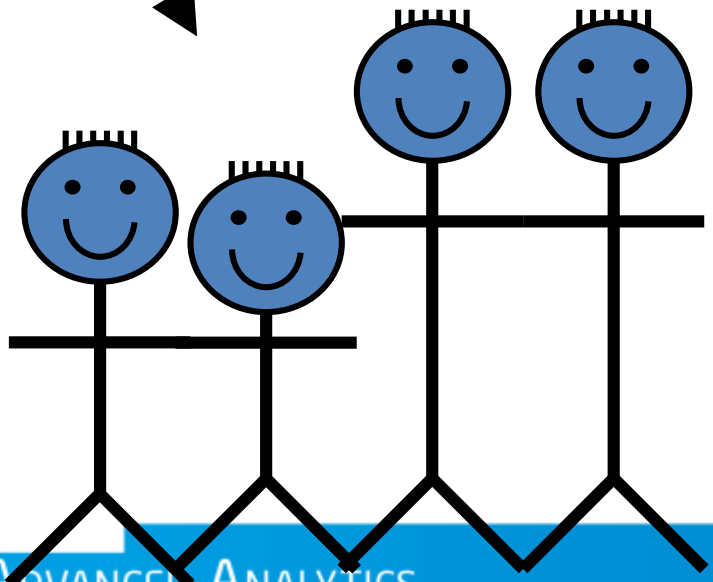
If there is no significant difference between the two groups…

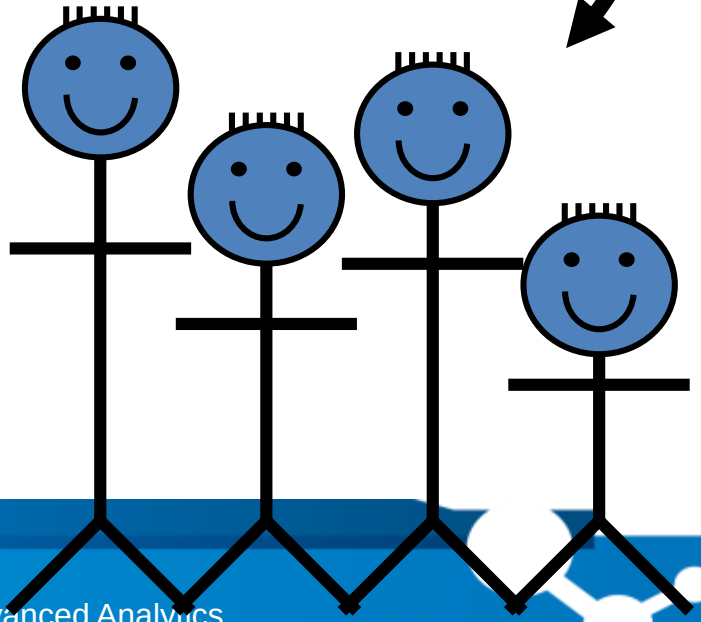


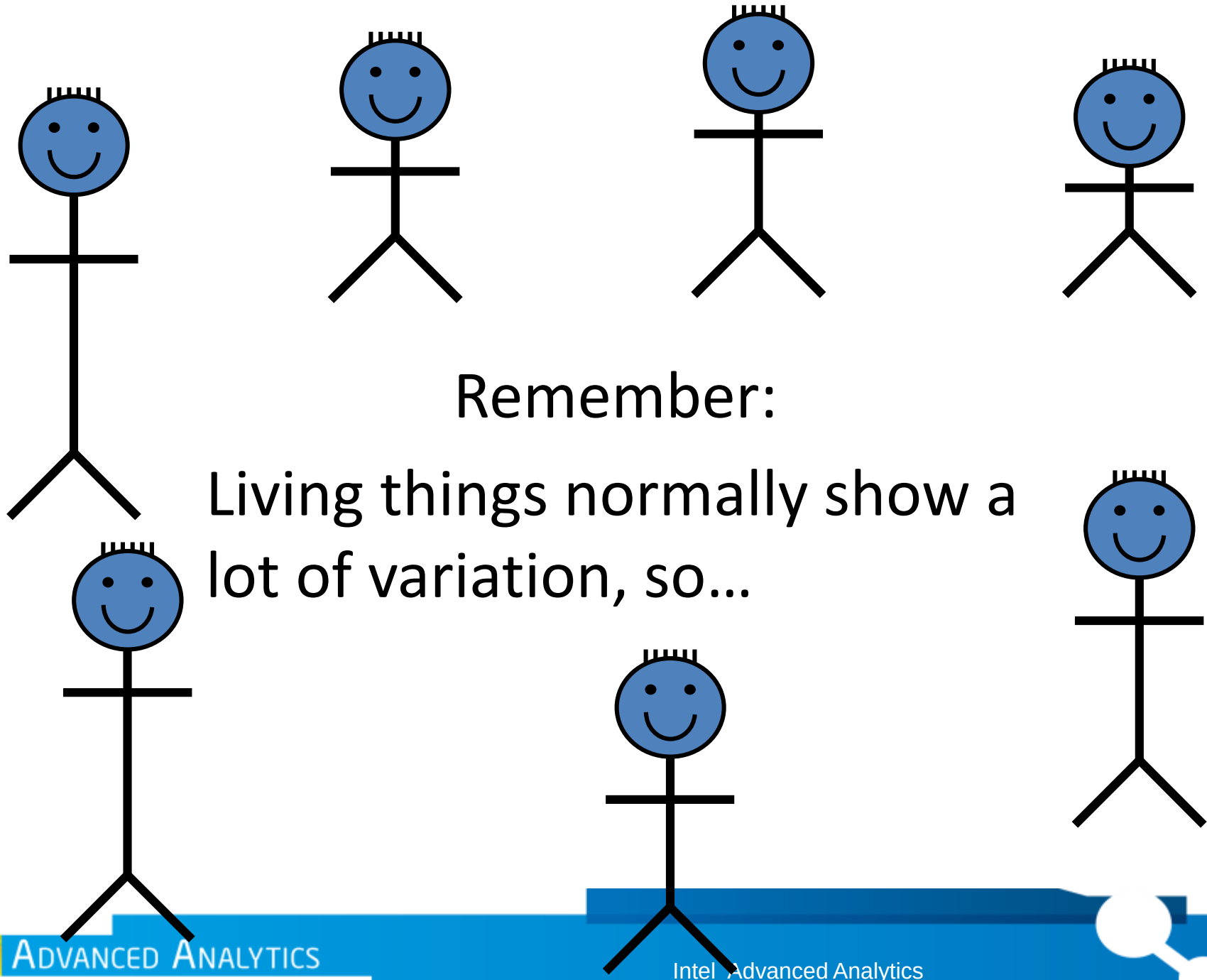

… the average or mean height of the two groups should be very…

… SIMILAR





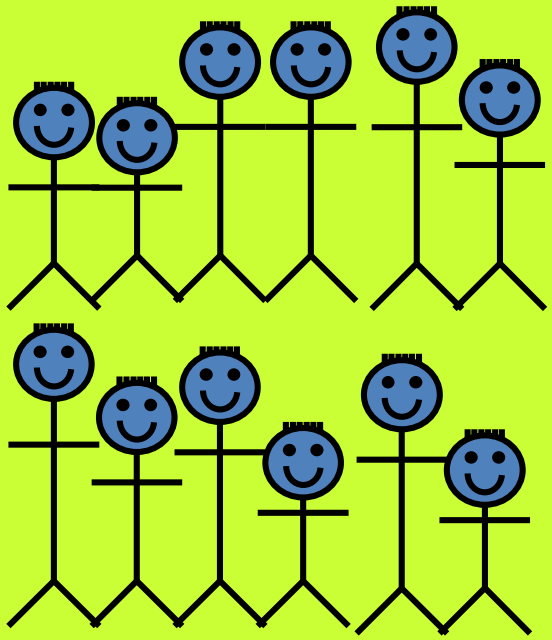


Remember:

Living things normally show a lot of variation, so…

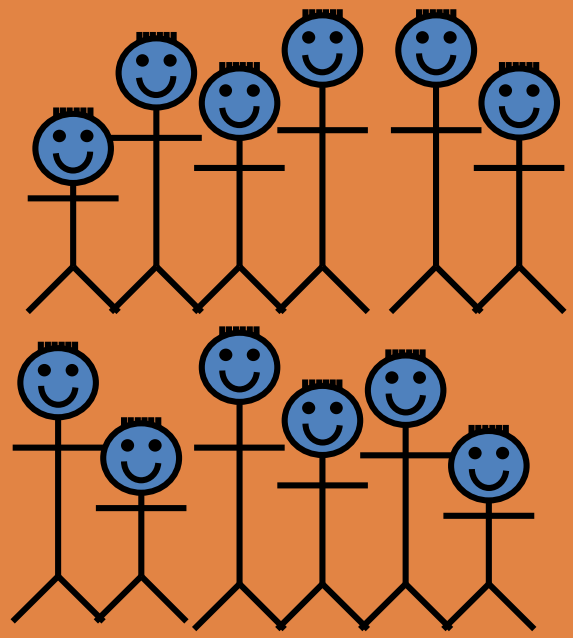
It is *VERY* unlikely that the mean height of our two samples will be exactly the same

C1 Sample



Average height = 162 cm

D8 Sample



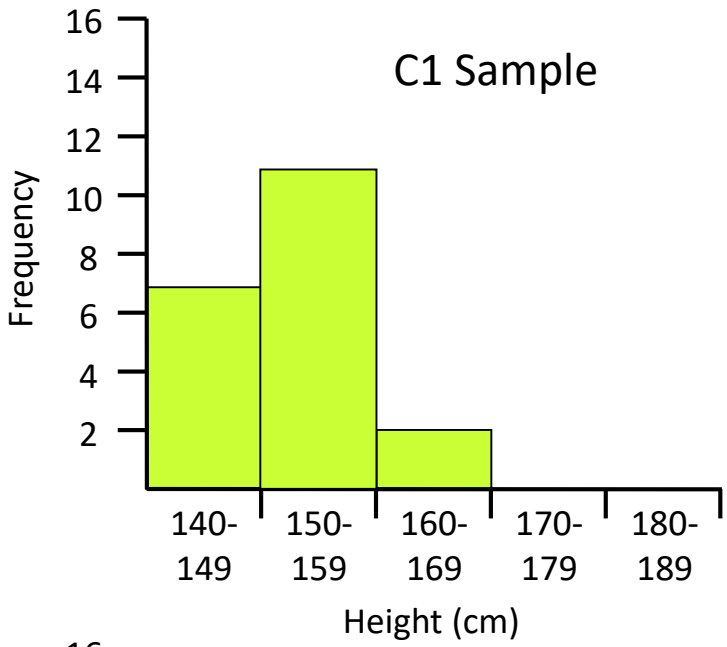
Average height = 168 cm

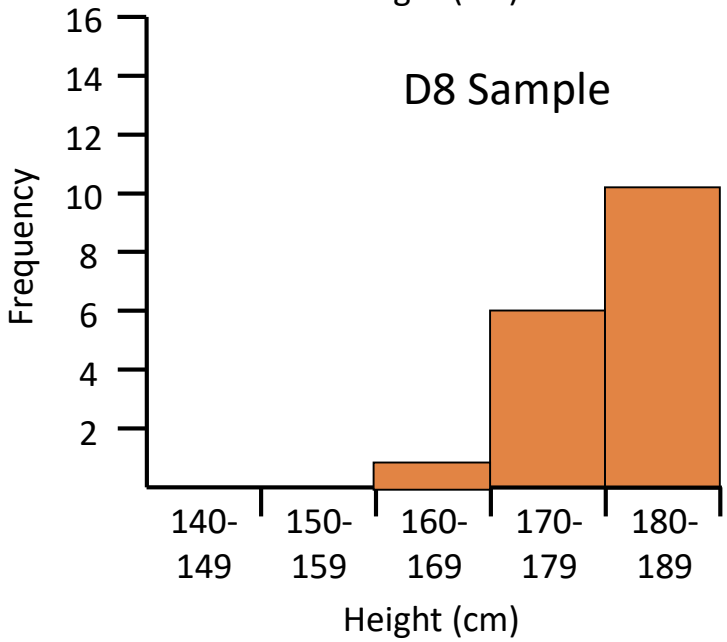
Is the difference in average height of the samples large enough to be significant?

We can analyse the spread of the heights of the students in the samples by drawing *histograms*

Here, the ranges of the two samples have a small overlap, so…

… the difference between the means of the two samples *IS* probably significant.

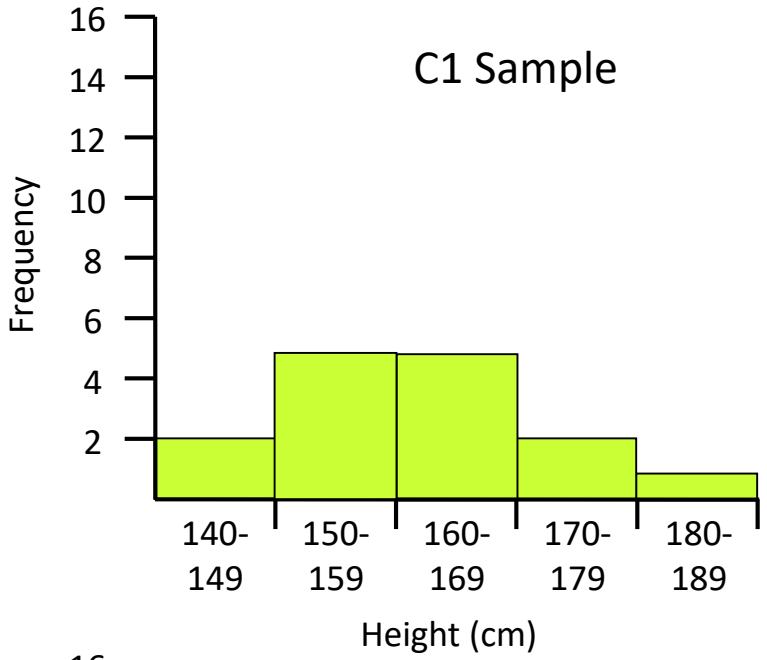


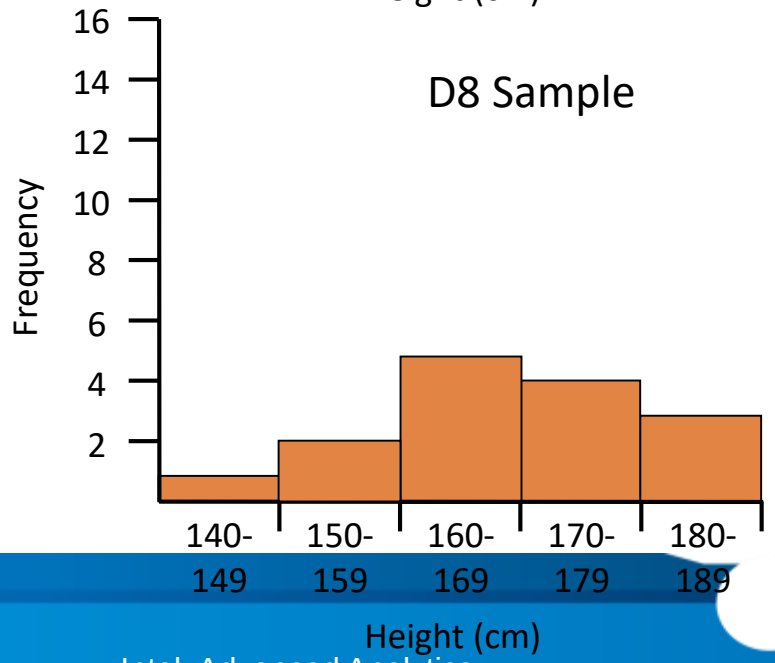



Here, the ranges of the two samples have a large overlap, so…

… the difference between the two samples may *NOT be* significant.

The difference in means is possibly due to *random sampling error*

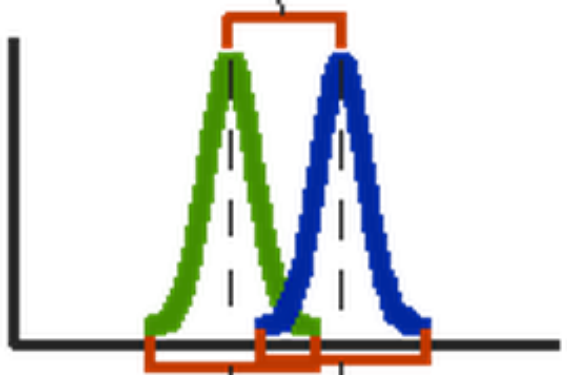


$$\frac{\text{signal}}{\text{noise}} = \frac{\text{difference between group means}}{\text{variability of groups}}$$

$$= \frac{\bar{X}_T - \bar{X}_C}{SE(\bar{X}_T - \bar{X}_C)}$$

$$= \text{t-value}$$



$$t = \frac{\bar{x} - \bar{y}}{\sqrt{\frac{S_x^2}{n} + \frac{S_y^2}{m}}}$$

# T-Tests

- Single Sample
  - One group to known Value [ e.g IQ in team]
- Paired Samples
  - Relation Between Two Groups
    - Same Object [ e.g Lab Examples]
    - Close Relation [ e.g Husband and Wife]
- Independent Samples
  - Relation Between General Two Groups [e.g classroom height]

# The Robust Nature of the *t* Statistics

- The only situation in which using a *t* test is likely to give a seriously distorted result is when you are using a *one-tailed test and the population is highly skewed.*

# Scenarios When you would use a Single Sample *t* test

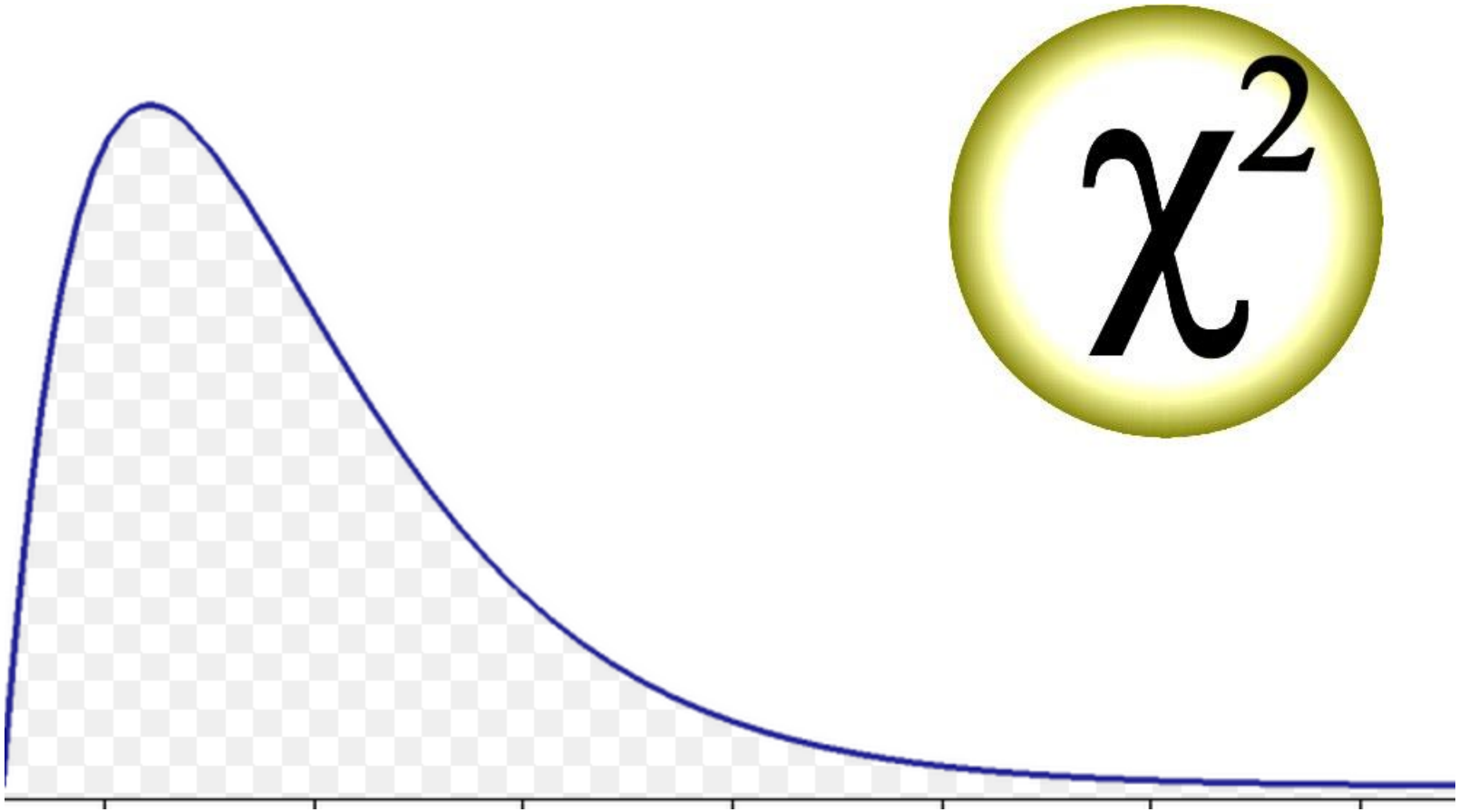
- A newspaper article reported that the typical American family spent an average of $81 for Halloween candy and costumes last year. A sample of N = 16 families this year reported spending a mean of M = $85, with s = $20. What statistical test would we use to determine whether these data indicate a significant change in holiday spending?

- Many companies that manufacture light bulbs advertise their 60-watt bulbs as having an average life of 1000 hours. A cynical consumer bought 30 bulbs and burned them until they failed. He found that they burned for an average of *M = 1233, with a standard deviation of s = 232.06. What statistical test would this consumer use to determine whether the average burn time of light bulbs differs significantly from that advertised?*

# Agenda

- Linear Regression
- Logistic Regression
- Ensembles Intro / Random Forest
- Kmeans
- T test
- Chi Square

# Chi-Square



# The Chi Square Test

- A statistical method used to determine goodness of fit
  - Goodness of fit refers to how close the observed data are to those predicted from a hypothesis
- Note:
  - The chi square test does not prove that a hypothesis is correct
    - It evaluates to what extent the data and the hypothesis have a good fit

# The Chi Square Test

- The general formula is

$$\chi^2 = \Sigma \frac{(O - E)^2}{E}$$

- where
  - O = observed data in each category
  - E = observed data in each category based on the experimenter's hypothesis
  - Σ = Sum of the calculations for each category

- Consider the following example in *Drosophila melanogaster*

- Gene affecting wing shape
  - $c^+$ = Normal wing
  - $c$ = Curved wing
- Gene affecting body color
  - $e^+$ = Normal (gray)
  - $e$ = ebony

- Note:
  - The wild-type allele is designated with a + sign
  - Recessive mutant alleles are designated with lowercase letters
- The Cross:
  - A cross is made between two true-breeding flies ($c^+c^+e^+e^+$ and *ccee).* The flies of the $F_1$ generation are then allowed to mate with each other to produce an $F_2$ generation.

- The outcome
  - $F_1$ generation
    - All offspring have straight wings and gray bodies
  - $F_2$ generation
    - 193 straight wings, gray bodies
    - 69 straight wings, ebony bodies
    - 64 curved wings, gray bodies
    - 26 curved wings, ebony bodies
    - 352 total flies
- Applying the chi square test
  - Step 1: Propose a null hypothesis ($H_0$) that allows us to calculate the expected values based on Mendel's laws
    - The two traits are independently assorting

– Step 2: Calculate the expected values of the four phenotypes, based on the hypothesis

- According to our hypothesis, there should be a 9:3:3:1 ratio on the $F_2$ generation

| Phenotype | Expected probability | Expected number | Observed number |
|---|---|---|---|
| straight wings, gray bodies | 9/16 | 9/16 X 352 = 198 | 193 |
| straight wings, ebony bodies | 3/16 | 3/16 X 352 = 66 | 64 |
| curved wings, gray bodies | 3/16 | 3/16 X 352 = 66 | 62 |
| curved wings, ebony bodies | 1/16 | 1/16 X 352 = 22 | 24 |

## – Step 3: Apply the chi square formula

$$\chi^2 = \frac{(O_1 - E_1)^2}{E_1} + \frac{(O_2 - E_2)^2}{E_2} + \frac{(O_3 - E_3)^2}{E_3} + \frac{(O_4 - E_4)^2}{E_4}$$

$$\chi^2 = \frac{(193 - 198)^2}{198} + \frac{(69 - 66)^2}{66} + \frac{(64 - 66)^2}{66} + \frac{(26 - 22)^2}{22}$$

$$\chi^2 = 0.13 + 0.14 + 0.06 + 0.73$$

$$\chi^2 = 1.06$$

| Expected number | Observed number |
|---|---|
| 198 | 193 |
| 66 | 64 |
| 66 | 62 |
| 22 | 24 |

- Step 4: Interpret the chi square value
  - The calculated chi square value can be used to obtain probabilities, or P values, from a chi square table
    - These probabilities allow us to determine the likelihood that the observed deviations are due to random chance alone
  - Low chi square values indicate a high probability that the observed deviations could be due to random chance alone
  - High chi square values indicate a low probability that the observed deviations are due to random chance alone
  - If the chi square value results in a probability that is less than 0.05 (ie: less than 5%) it is considered ***statistically significant***
    - The hypothesis is rejected

- Step 4: Interpret the chi square value

  - Before we can use the chi square table, we have to determine the degrees of freedom (*df*)
    - The *df* is a measure of the number of categories that are independent of each other
    - If you know the 3 of the 4 categories you can deduce the $4^{th}$ (total number of progeny – categories 1-3)
    - $df = n - 1$
      - where $n$ = total number of categories
    - In our experiment, there are four phenotypes/categories
      - Therefore, df = 4 – 1 = 3

## TABLE 2.1

### Chi Square Values and Probability

| Degrees of Freedom | $P = 0.99$ | 0.95 | 0.80 | 0.50 | 0.20 | 0.05 | 0.01 |
|---|---|---|---|---|---|---|---|
| 1 | 0.000157 | 0.00393 | 0.0642 | 0.455 | 1.642 | **3.841** | **6.635** |
| 2 | 0.020 | 0.103 | 0.446 | 1.386 | 3.219 | **5.991** | **9.210** |
| 3 | 0.115 | 0.352 | 1.005 | 2.366 | 4.642 | **7.815** | **11.345** |
| 4 | 0.297 | 0.711 | 1.649 | 3.357 | 5.989 | **9.488** | **13.277** |
| 5 | 0.554 | 1.145 | 2.343 | 4.351 | 7.289 | **11.070** | **15.086** |
| 6 | 0.872 | 1.635 | 3.070 | 5.348 | 8.558 | **12.592** | **16.812** |
| 7 | 1.239 | 2.167 | 3.822 | 6.346 | 9.803 | **14.067** | **18.475** |
| 8 | 1.646 | 2.733 | 4.594 | 7.344 | 11.030 | **15.507** | **20.090** |
| 9 | 2.088 | 3.325 | 5.380 | 8.343 | 12.242 | **16.919** | **21.666** |
| 10 | 2.558 | 3.940 | 6.179 | 9.342 | 13.442 | **18.307** | **23.209** |
| 15 | 5.229 | 7.261 | 10.307 | 14.339 | 19.311 | **24.996** | **30.578** |
| 20 | 8.260 | 10.851 | 14.578 | 19.337 | 25.038 | **31.410** | **37.566** |
| 25 | 11.524 | 14.611 | 18.940 | 24.337 | 30.675 | **37.652** | **44.314** |
| 30 | 14.953 | 18.493 | 23.364 | 29.336 | 36.250 | **43.773** | **50.892** |

1.06

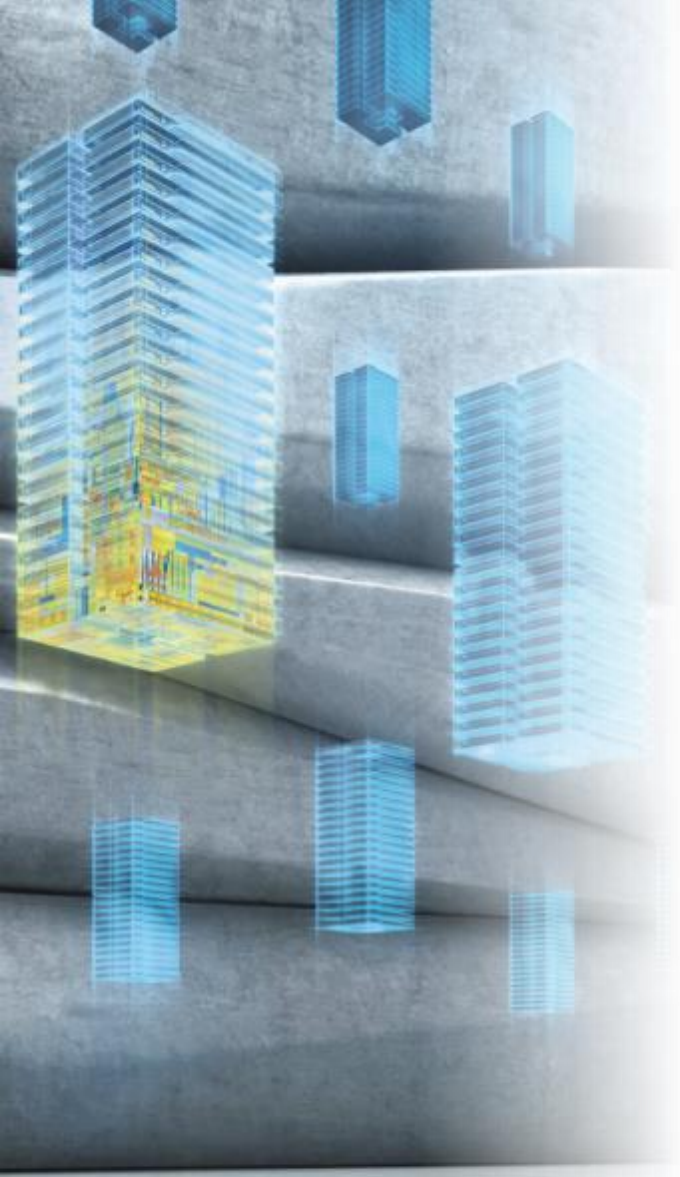
From Fisher, R. A., and Yates, F. (1943) *Statistical Tables for Biological, Agricultural, and Medical Research.* Oliver and Boyd, London.

- Step 4: Interpret the chi square value

  – With *df* = 3, the chi square value of 1.06 is slightly greater than 1.005 (which corresponds to P-value = 0.80)

  – P-value = 0.80 means that Chi-square values equal to or greater than 1.005 are expected to occur 80% of the time due to random chance alone; that is, when the null hypothesis is true.

  – Therefore, it is quite probable that the deviations between the observed and expected values in this experiment can be explained by random sampling error and *the null hypothesis is not rejected*. What was the null hypothesis?



# Backup

# Generalized Linear Model

- Quadratic discriminant

$$y(x|W, w, w_0) = x^T W x + w^T x + w_0$$

- More generally, linear combination of *nonlinear basis functions*

$$y(x|w, \phi) = \sum_{j=1}^{M} w_j \phi_j(x) = w^T \phi(x)$$

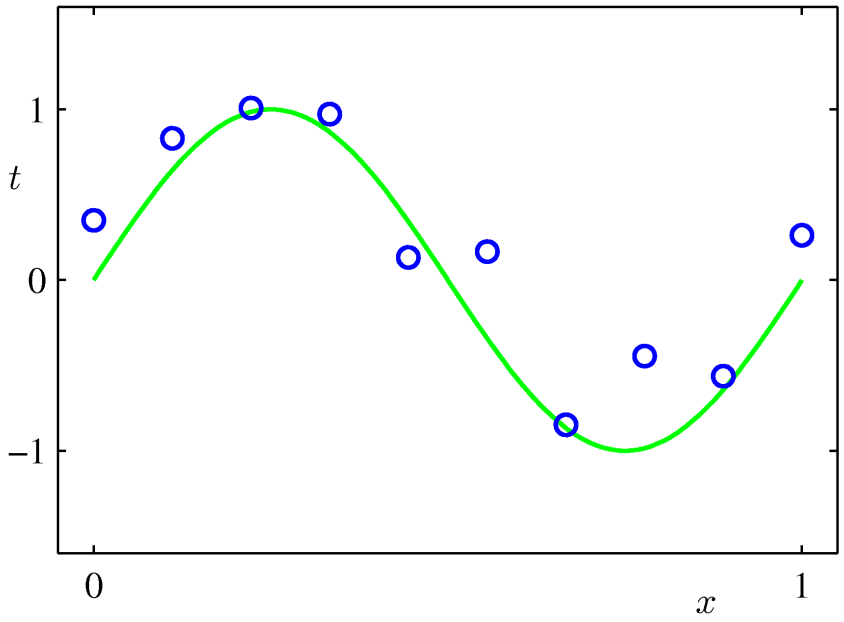
  - Interpretation
    - A nonlinear mapping of x to *z*-space: $z_j = \phi_j(x)$
    - A linear discriminant in the *z*-space: $y(x|w, \phi)$

- $\phi(x)$ are known as the *basis functions*
  - Typically, $\phi_0(x) = 1$, so that $w_0$ acts as the *bias*
  - In the simplest case, we use a linear basis functions

$$\phi_d(x) = x_d$$

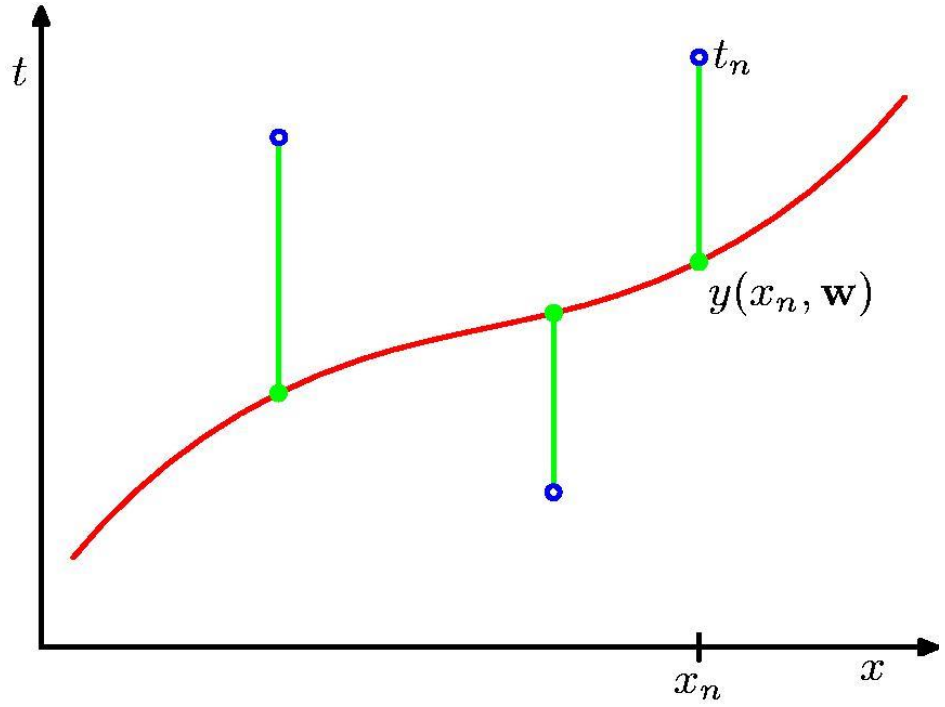
# Linear Basis Function Models

- Example: Polynomial Curve Fitting



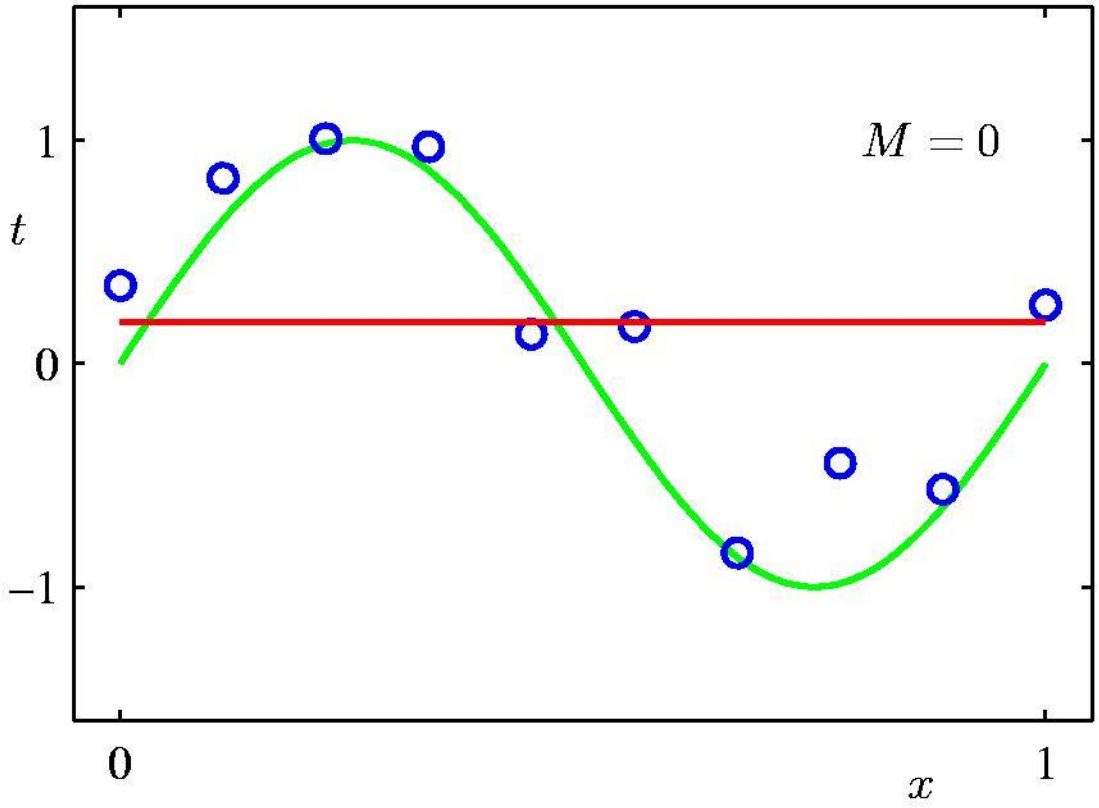

$$y(x, \mathbf{w}) = w_0 + w_1 x + w_2 x^2 + \ldots + w_M x^M = \sum_{j=0}^{M} w_j x^j$$

# Sum-of-Squares Error Function

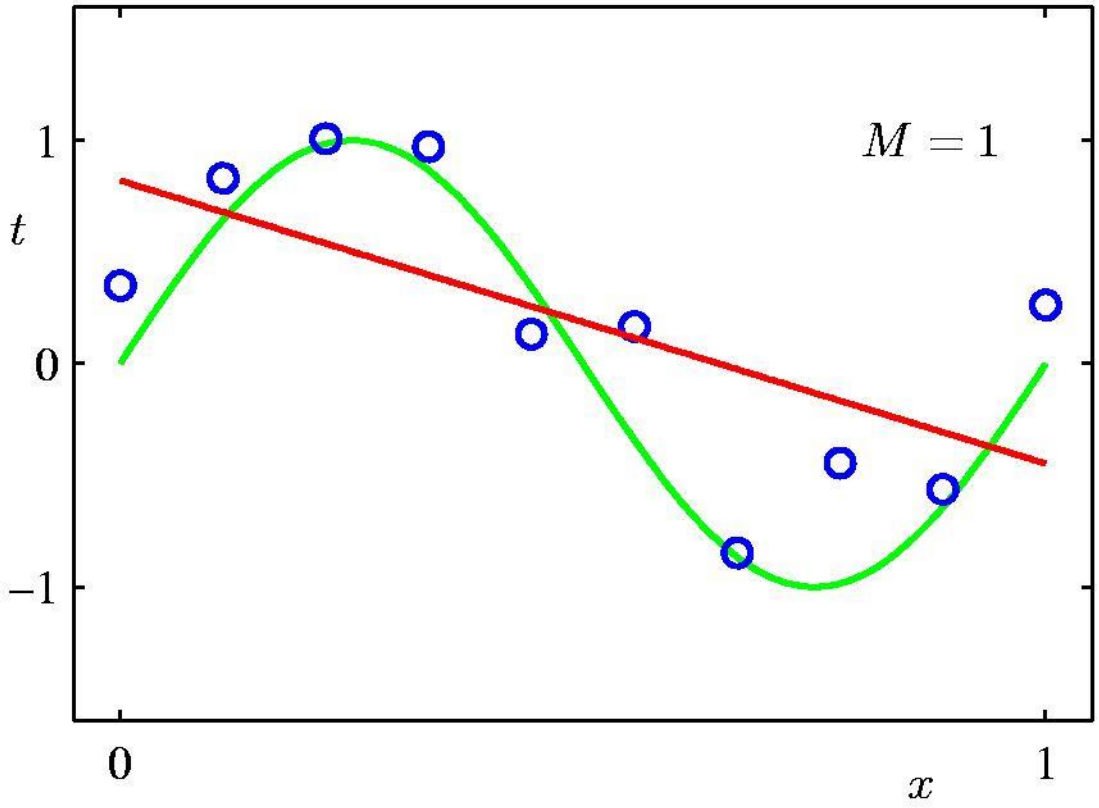



$$E(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^{N} \{y(x_n, \mathbf{w}) - t_n\}^2$$

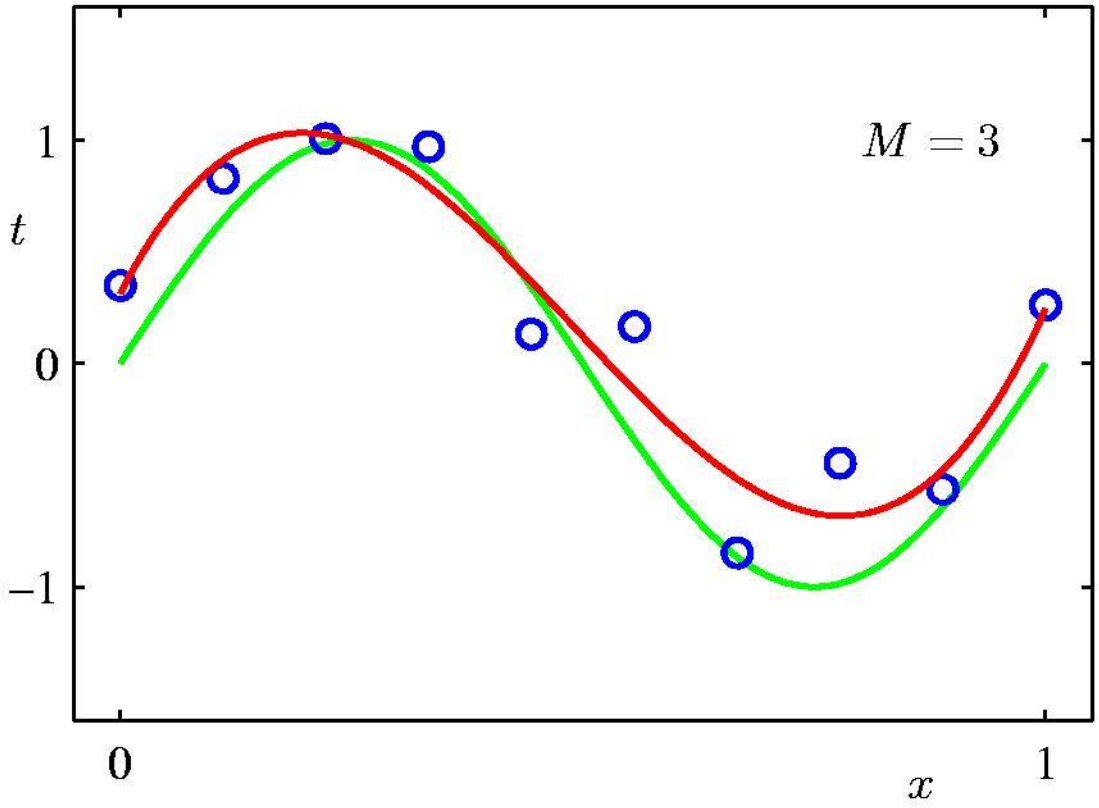
# $0^{th}$ Order Polynomial



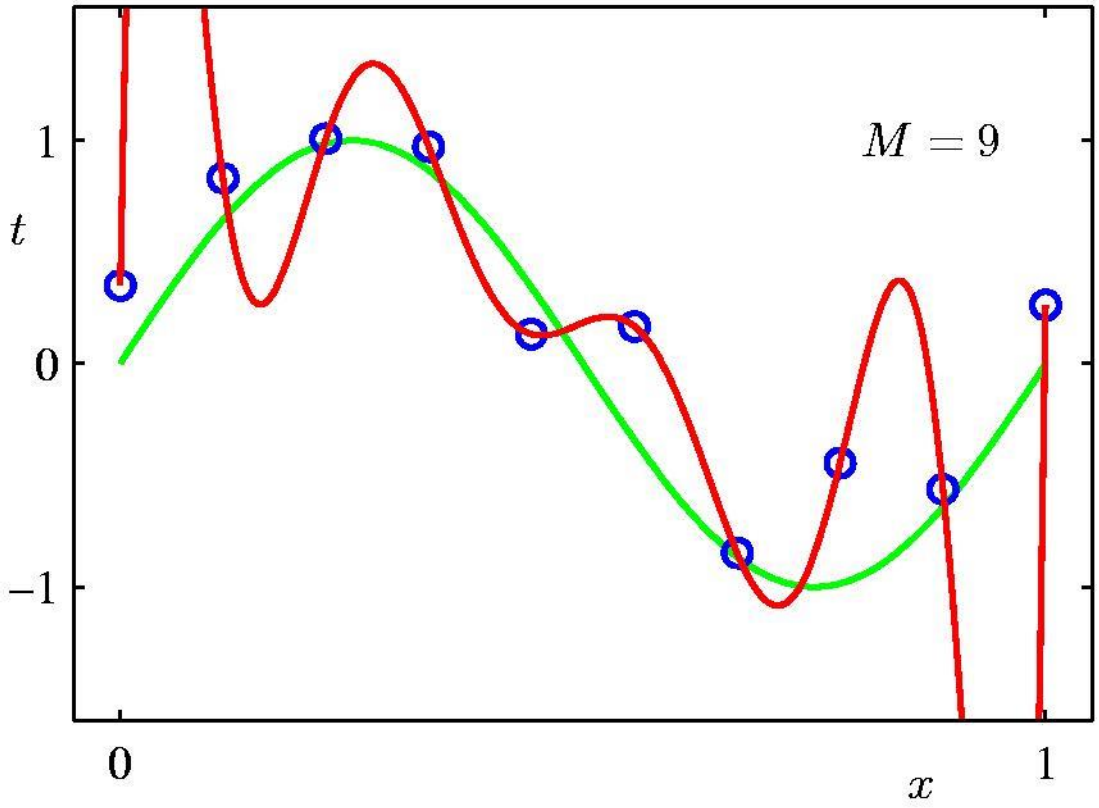
# 1st Order Polynomial



# 3rd Order Polynomial



# 9th Order Polynomial

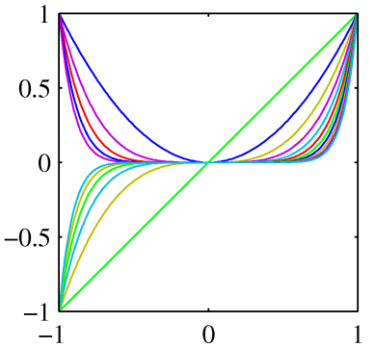


# Basis Functions

- **Polynomial basis functions**

$$\phi_j(x) = x^j$$

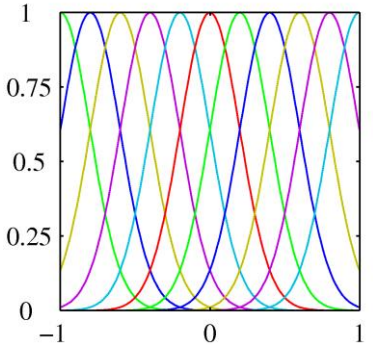
  - These are global; a small change in x affect all basis functions




- **Gaussian basis functions**

$$\phi_j(x) = \exp\left\{-\frac{(x-\mu_j)^2}{2s^2}\right\}$$

  - These are local; a small change in x only affect nearby basis functions
  - $\mu_j$ and $s$ control location and scale (width)



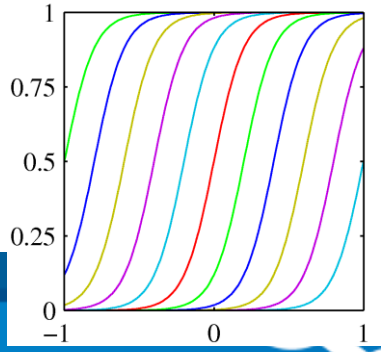

- **Sigmoidal basis functions**

$$\phi_j(x) = \sigma\left(\frac{x-\mu_j}{s}\right)$$

  - where

$$\sigma(a) = \frac{1}{1+\exp(-a)}$$

  - These are local; a small change in x only affect nearby basis functions
  - $\mu_j$ and $s$ control location and scale (slope)



# Closed-form Solution

$$\mathbf{X} \cdot \mathbf{w} = y$$

$$\begin{pmatrix} x_{10} & x_{11} & \cdots & x_{1d} \\ x_{20} & x_{21} & \cdots & x_{2d} \\ \vdots & \vdots & & \\ x_{n0} & x_{n1} & & x_{n1} \end{pmatrix} \begin{pmatrix} w_0 \\ \vdots \\ w_d \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_2 \end{pmatrix}$$

- The parameters $\boldsymbol{w}$ are obtained by minimizing the sum of the squared residuals

$$E(\boldsymbol{w}) = \|\boldsymbol{X}\,\boldsymbol{w} - \boldsymbol{y}\,\|^2 = \sum_{i=1}^{n} \left(w^T x^i - y^i\right)^2$$

- Forming the derivative yields

$$\nabla E(\boldsymbol{w}) = \sum_{i=1}^{n} 2\left(w^T x^i - y^i\right) x_i = 2\boldsymbol{X}^T(\boldsymbol{X}\,\boldsymbol{w} - \boldsymbol{y}\,)$$

- Setting the derivative to zero yields the necessary condition for minimum

$$\boldsymbol{X^T X w} = \boldsymbol{X^T y}$$

- Now, $\boldsymbol{X^T X}$ is square and often nonsingular and so we can solve for $\vec{\boldsymbol{w}}$ uniquely as

$$\boldsymbol{w} = \left(\boldsymbol{X^T X}\right)^{-1} \boldsymbol{X^T y}$$

- The $d \times d$ matrix $\left(\boldsymbol{X^T X}\right)^{-1} \boldsymbol{X^T}$ is the *Pseudo Inverse of* $\boldsymbol{X}$

# Maximum Likelihood and Least Squares

- Assume observations from a deterministic function with added Gaussian noise

$$t = y(x, w) + \epsilon$$

  - where $\Pr(\epsilon) = N(0, \sigma^2)$

- which is the same as saying

$$\Pr(t|x, w, \sigma^2) = N(t|y(x, w), \sigma^2)$$

- Given observed inputs, $\boldsymbol{X} = \{x_1, x_2, \dots, x_n\}$, and targets, $\boldsymbol{t} = [t_1, t_2, \dots, t_n]^T$, the likelihood function is

$$\Pr(\boldsymbol{t}|\boldsymbol{X}, w, \sigma^2) = \prod_{i=1}^{n} N(t_i|y(x_i, w), \sigma^2) = \prod_{i=1}^{n} N(t_i|w^T\phi(x_i), \sigma^2)$$

- Taking the logarithm, we get

$$\ln(\Pr(\boldsymbol{t}|\boldsymbol{X}, w, \sigma^2)) = \sum_{i=1}^{n} \ln\left(N(t_i|w^T\phi(x_i), \sigma^2)\right) = -\frac{n}{2}\ln(2\pi\sigma^2) - \frac{E(w)}{\sigma^2}$$

  - Where $E(w)$ is the sum-of-squares error

$$E(w) = \frac{1}{2}\sum_{i=1}^{n}\left(t_i - w^T\phi(x_i)\right)^2$$

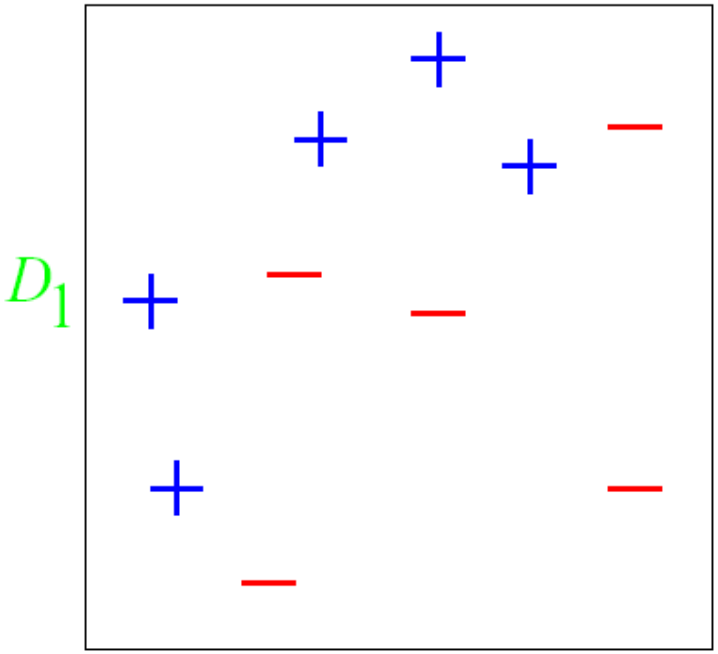
# Boosting

- An iterative procedure to adaptively change distribution of training data by focusing more on previously misclassified records
  - Initially, all N records are assigned equal weights
  - Unlike bagging, weights may change at the end of a boosting round

# Boosting

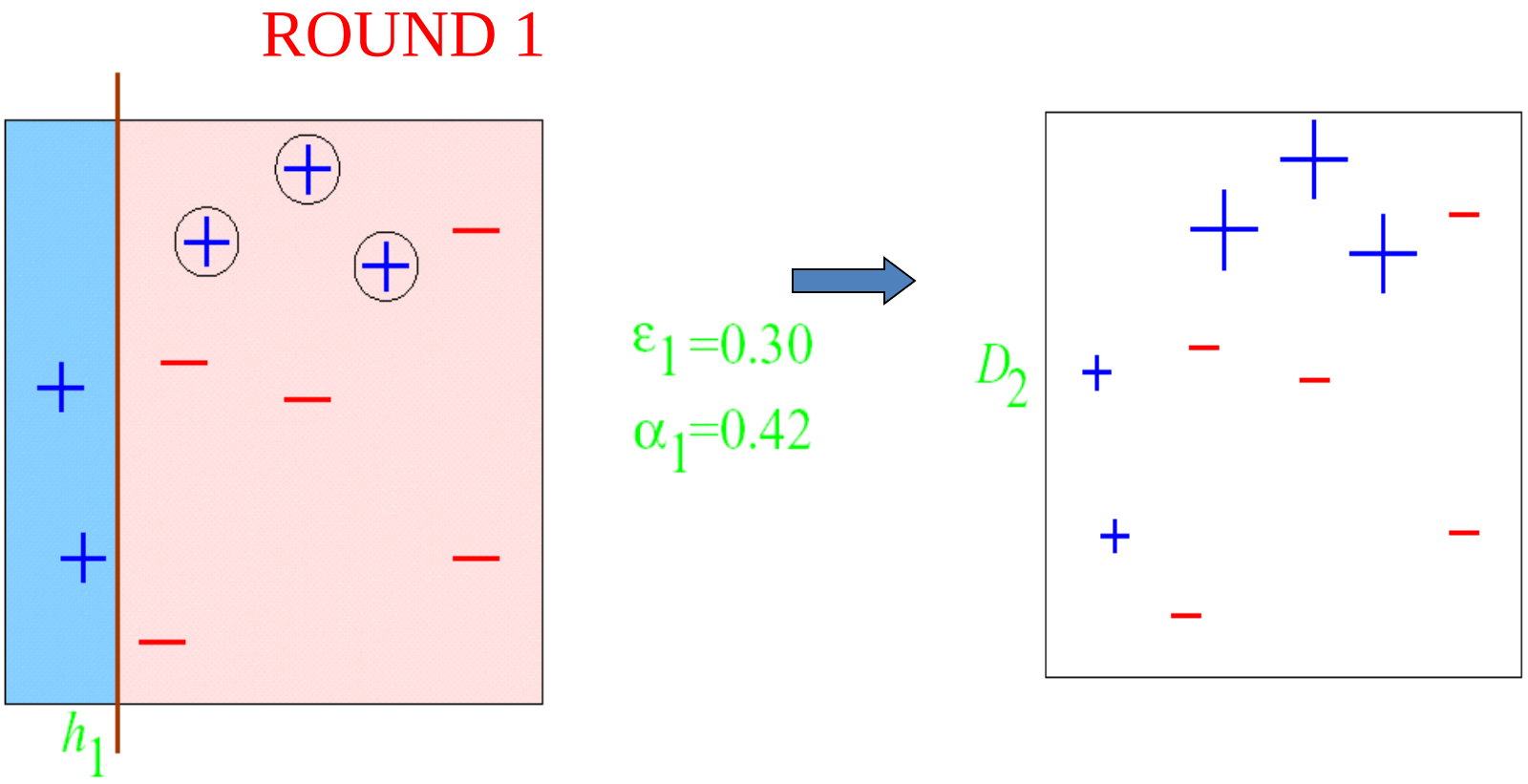
- Equal weights are assigned to each training tuple (1/d for round 1)
- After a classifier $M_i$ is learned, the weights are adjusted to allow the subsequent classifier $M_{i+1}$ to "pay more attention" to tuples that were misclassified by $M_i$.
- Final boosted classifier M* combines the votes of each individual classifier
- Weight of each classifier's vote is a function of its accuracy
- Adaboost – popular boosting algorithm

# And in animation



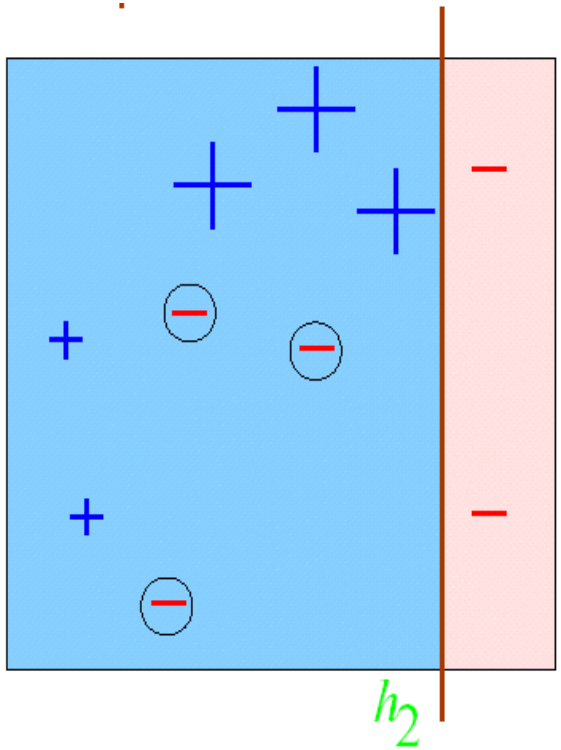

Original Training set : Equal Weights to all training samples

# AdaBoost(Example)



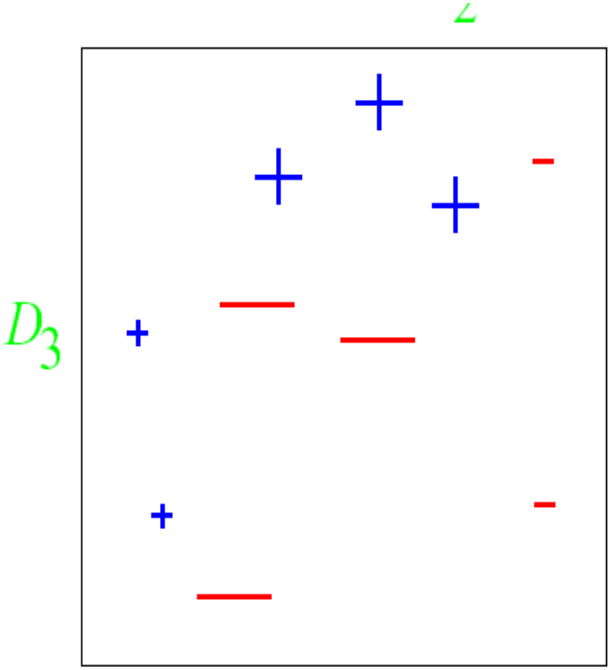
# AdaBoost(Example)

## ROUND 2



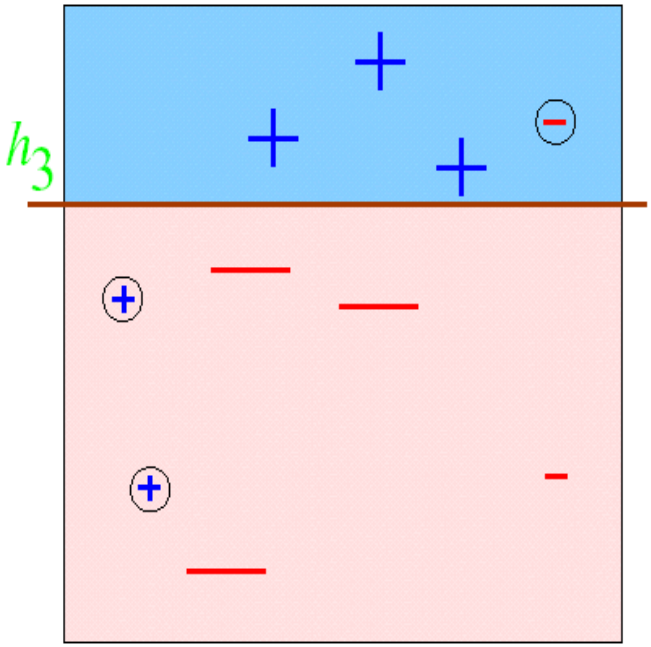

$\varepsilon_2 = 0.21$

$\alpha_2 = 0.65$



# AdaBoost(Example)

ROUND 3

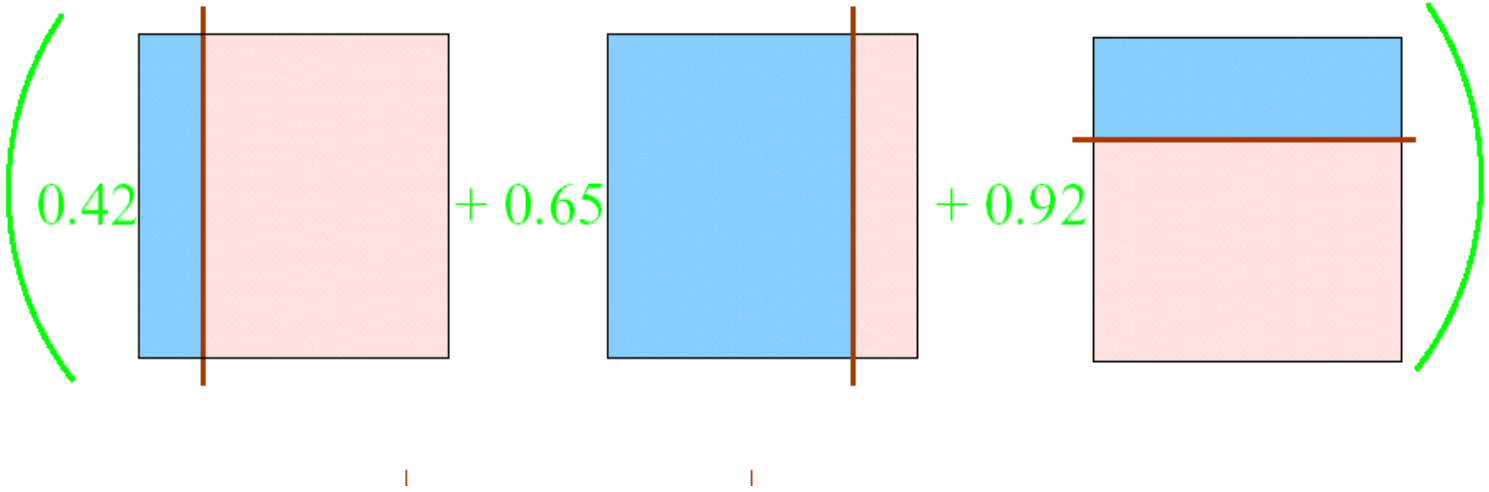



$\varepsilon_3 = 0.14$

$\alpha_3 = 0.92$

# AdaBoost(Example)

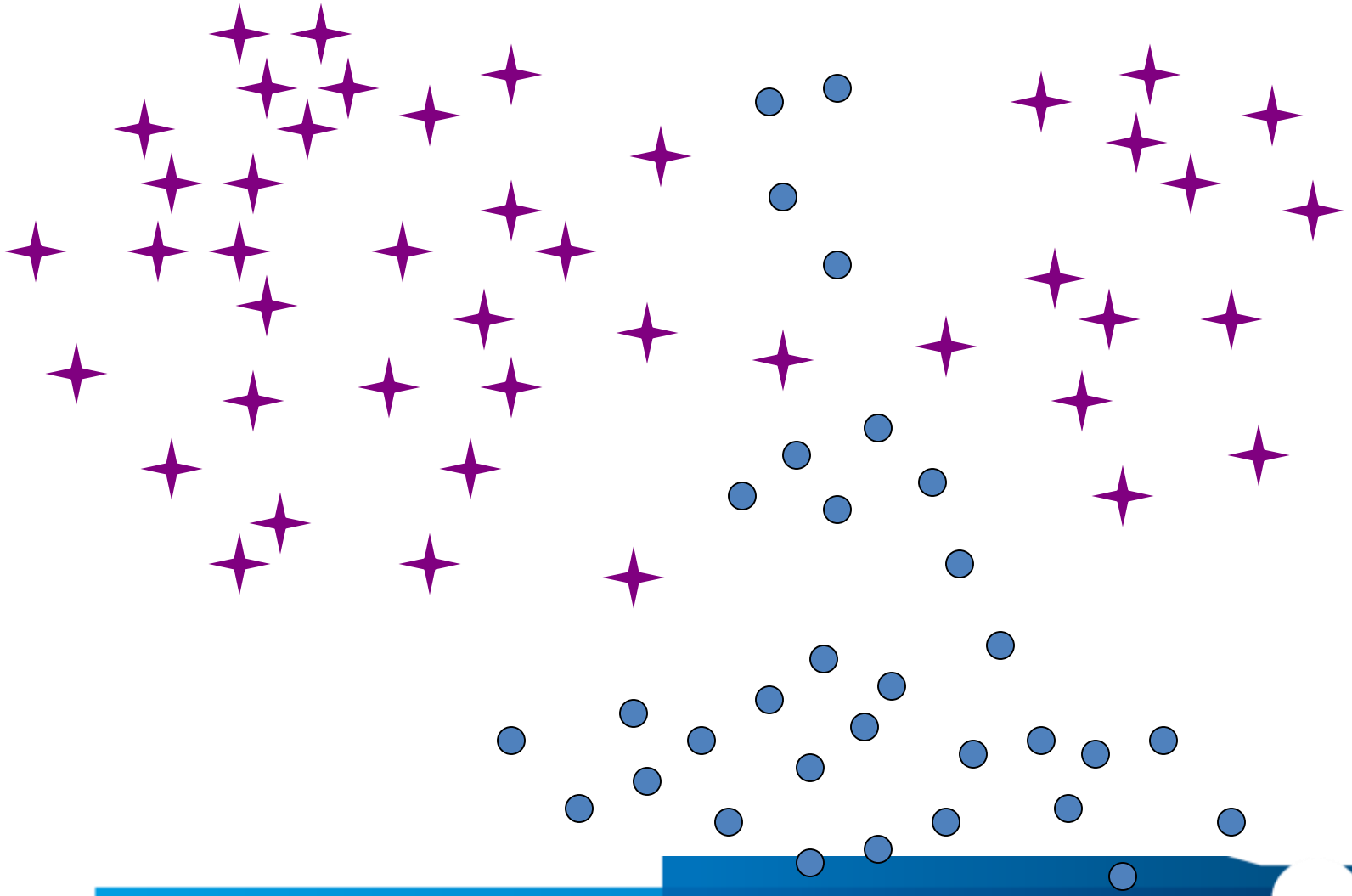
$$H_{\text{final}} = \text{sign}\left( 0.42 \; + 0.65 \; + 0.92 \right)$$



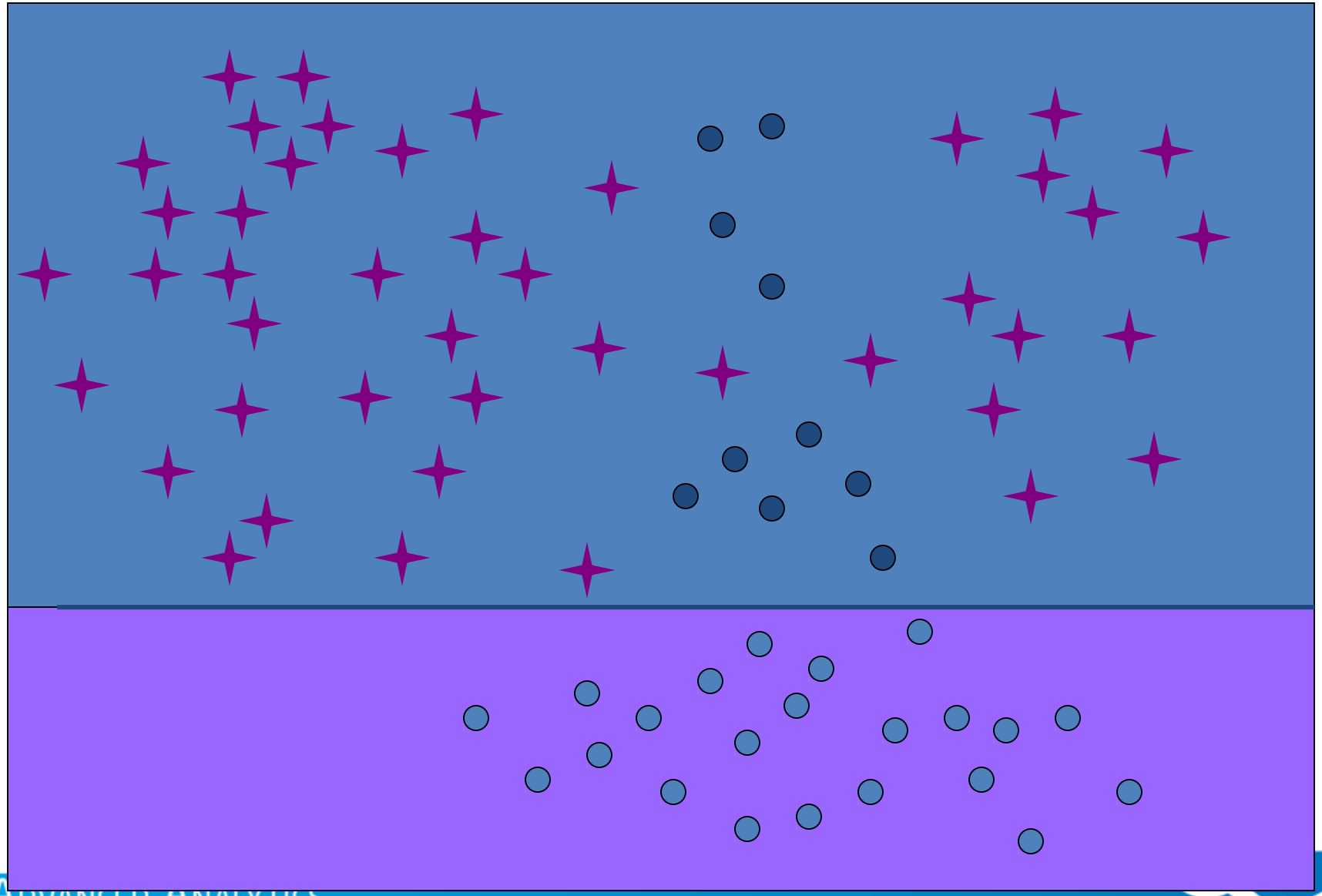
# Ada-Boosting

- Input:
  - Training samples $S = \{(x_i, y_i)\}$, $i = 1, 2, \ldots, N$
  - Weak learner h
- Initialization
  - Each sample has equal weight $w_i = 1/N$
- For k = 1 … T
  - Train weak learner $h_k$ according to weighted sample sets
  - Compute classification errors
  - Update sample weights $w_i$
- Output
  - Final model which is a linear combination of $h_k$

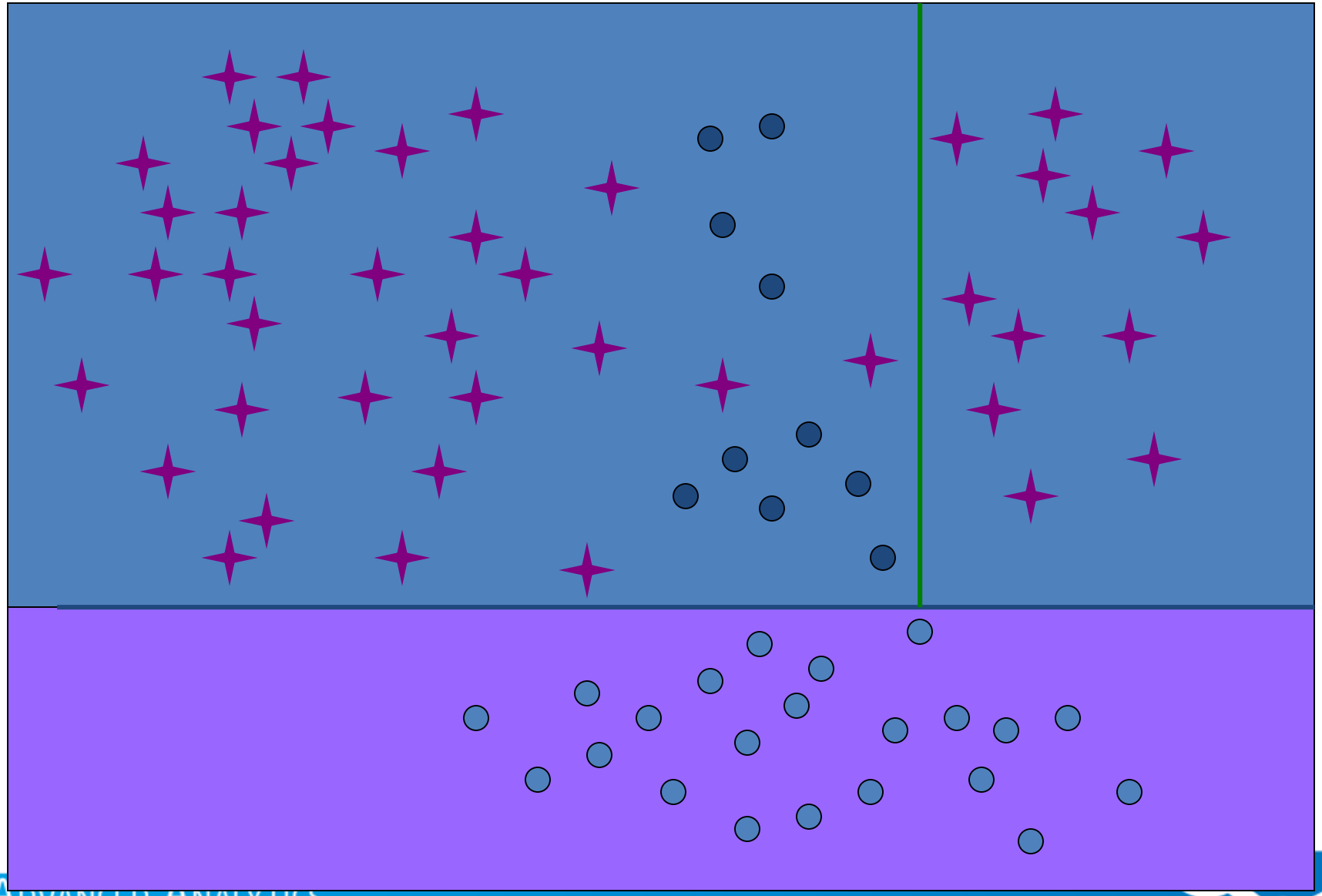
# Ada-Boosting



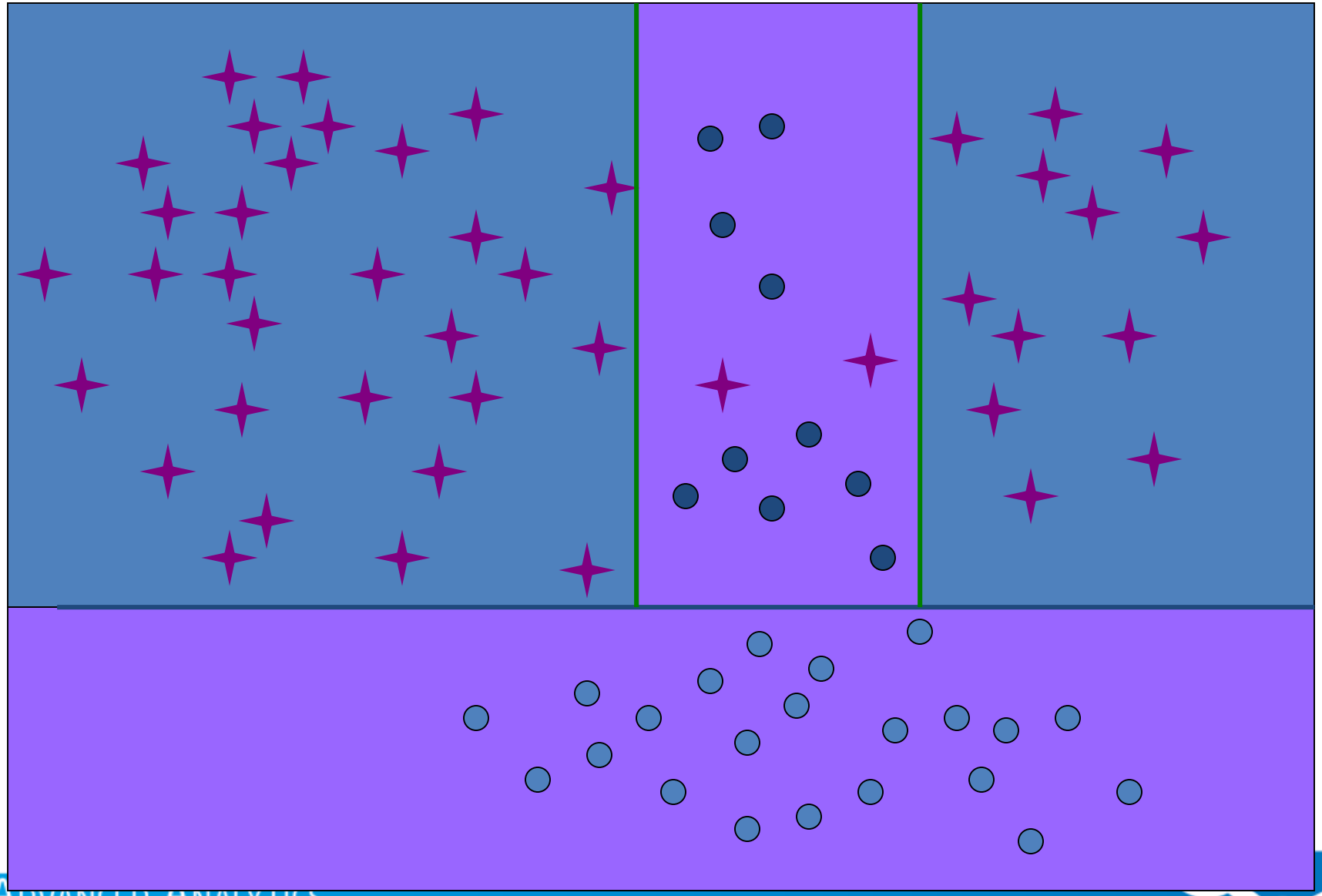
# Ada-Boosting



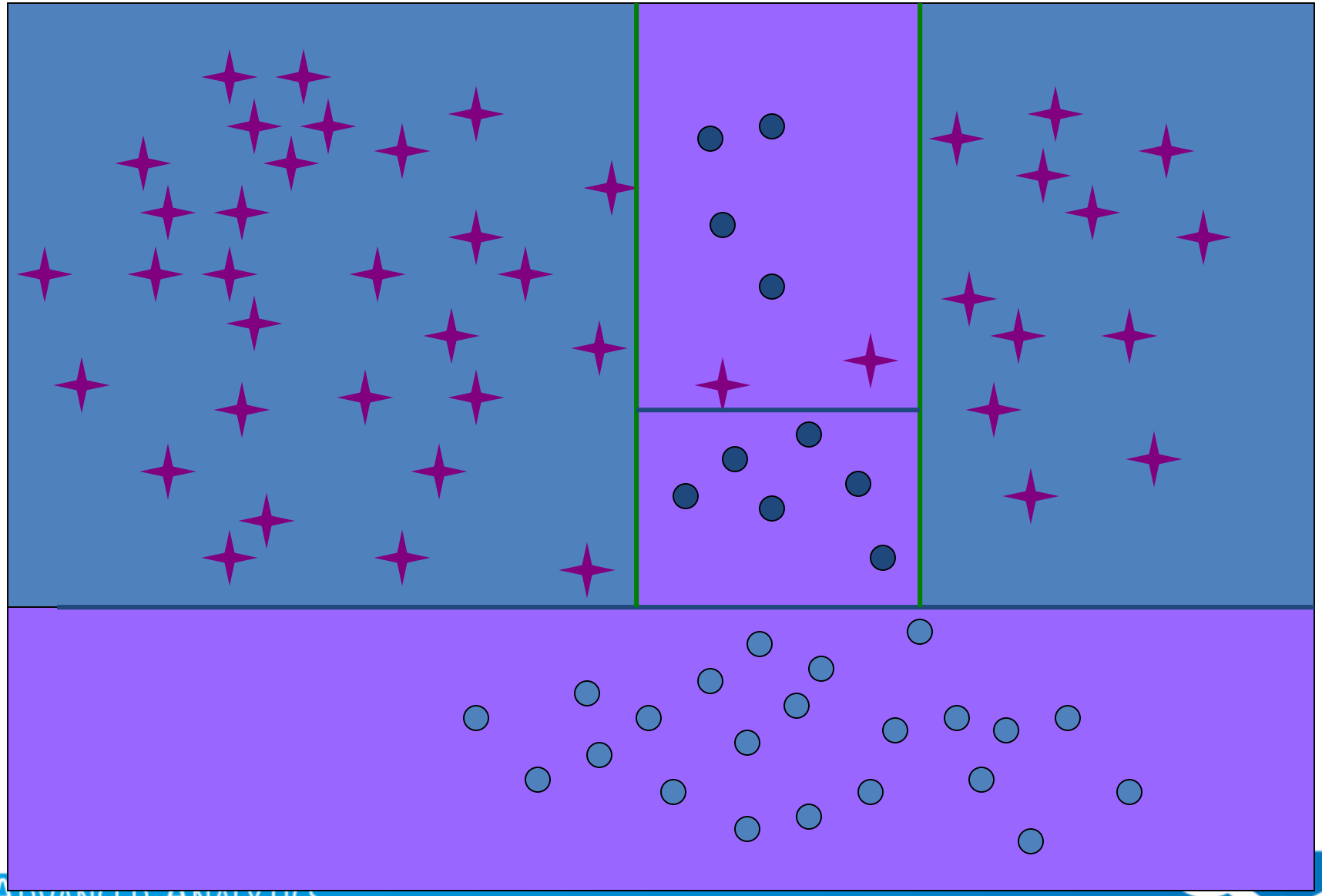
# Ada-Boosting



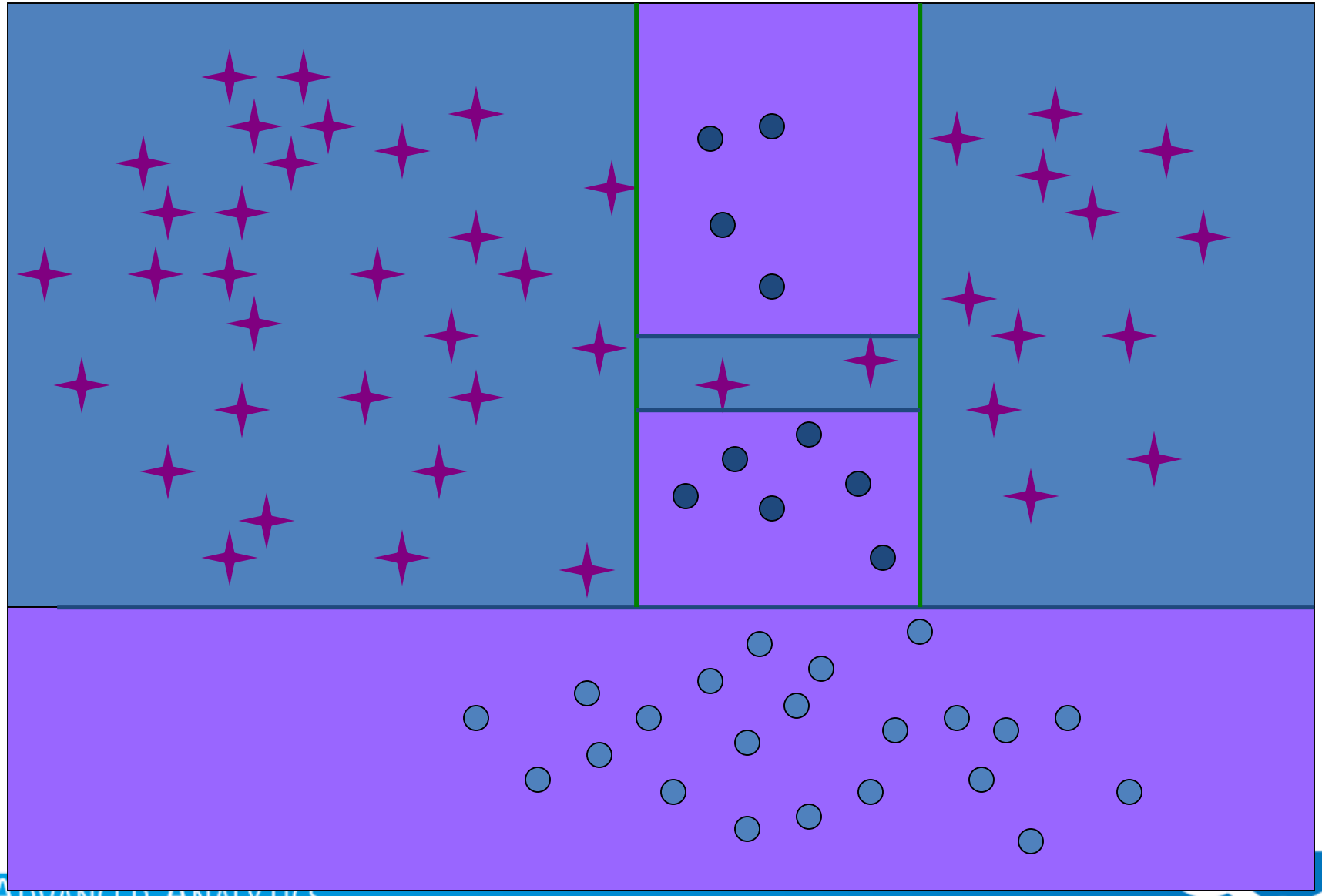
# Ada-Boosting



# Ada-Boosting



# Ada-Boosting



# Difference Between Calculating *z* Statistic and *t* Statistic

**z Statistic**

$$\sigma = \sqrt{\frac{\Sigma(X-\mu)^2}{N}}$$

$$\sigma_m = \frac{\sigma}{\sqrt{N}}$$

$$z = \frac{(M-\mu_M)}{\sigma_M}$$

**t Statistic**

$$s = \sqrt{\frac{\Sigma(X-M)^2}{N-1}}$$

$$s_m = \frac{s}{\sqrt{N}}$$

$$t = \frac{(M-\mu_m)}{s_m}$$

# Estimating Population from a Sample

- Main difference between *t* Tests and *z* score:
  - use the standard deviation of the sample to estimate the standard deviation of the population.
- How? Subtract 1 from sample size! (called degrees of freedom)

$$SD = \sqrt{\frac{\Sigma(X-\mu)^2}{N}} \qquad s = \sqrt{\frac{\Sigma(X-M)^2}{N-1}}$$

- Use degrees of freedom (df) in the *t* distribution chart

# *t* Distribution Table

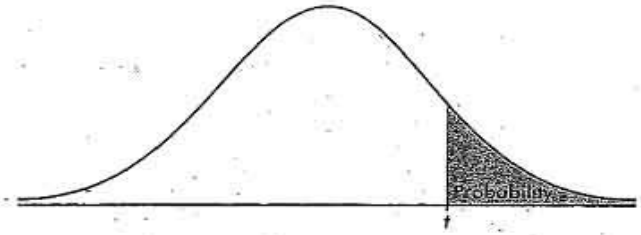



**TABLE B: *t*-DISTRIBUTION CRITICAL VALUES**

| | Tail probability $p$ | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| df | .25 | .20 | .15 | .10 | .05 | .025 | .02 | .01 | .005 | .0025 | .001 | .0005 |
| 1 | 1.000 | 1.376 | 1.963 | 3.078 | 6.314 | 12.71 | 15.89 | 31.82 | 63.66 | 127.3 | 318.3 | 636.6 |
| 2 | .816 | 1.061 | 1.386 | 1.886 | 2.920 | 4.303 | 4.849 | 6.965 | 9.925 | 14.09 | 22.33 | 31.60 |
| 3 | .765 | .978 | 1.250 | 1.638 | 2.353 | 3.182 | 3.482 | 4.541 | 5.841 | 7.453 | 10.21 | 12.92 |
| 4 | .741 | .941 | 1.190 | 1.533 | 2.132 | 2.776 | 2.999 | 3.747 | 4.604 | 5.598 | 7.173 | 8.610 |
| 5 | .727 | .920 | 1.156 | 1.476 | 2.015 | 2.571 | 2.757 | 3.365 | 4.032 | 4.773 | 5.893 | 6.869 |
| 6 | .718 | .906 | 1.134 | 1.440 | 1.943 | 2.447 | 2.612 | 3.143 | 3.707 | 4.317 | 5.208 | 5.959 |
| 7 | .711 | .896 | 1.119 | 1.415 | 1.895 | 2.365 | 2.517 | 2.998 | 3.499 | 4.029 | 4.785 | 5.408 |
| 8 | .706 | .889 | 1.108 | 1.397 | 1.860 | 2.306 | 2.449 | 2.896 | 3.355 | 3.833 | 4.501 | 5.041 |
| 9 | .703 | .883 | 1.100 | 1.383 | 1.833 | 2.262 | 2.398 | 2.821 | 3.250 | 3.690 | 4.297 | 4.781 |
| 10 | .700 | .879 | 1.093 | 1.372 | 1.812 | 2.228 | 2.359 | 2.764 | 3.169 | 3.581 | 4.144 | 4.587 |
| 11 | .697 | .876 | 1.088 | 1.363 | 1.796 | 2.201 | 2.328 | 2.718 | 3.106 | 3.497 | 4.025 | 4.437 |
| 12 | .695 | .873 | 1.083 | 1.356 | 1.782 | 2.179 | 2.303 | 2.681 | 3.055 | 3.428 | 3.930 | 4.318 |
| 13 | .694 | .870 | 1.079 | 1.350 | 1.771 | 2.160 | 2.282 | 2.650 | 3.012 | 3.372 | 3.852 | 4.221 |
| 14 | .692 | .868 | 1.076 | 1.345 | 1.761 | 2.145 | 2.264 | 2.624 | 2.977 | 3.326 | 3.787 | 4.140 |
| 15 | .691 | .866 | 1.074 | 1.341 | 1.753 | 2.131 | 2.249 | 2.602 | 2.947 | 3.286 | 3.733 | 4.073 |
| 16 | .690 | .865 | 1.071 | 1.337 | 1.746 | 2.120 | 2.235 | 2.583 | 2.921 | 3.252 | 3.686 | 4.015 |
| 17 | .689 | .863 | 1.069 | 1.333 | 1.740 | 2.110 | 2.224 | 2.567 | 2.898 | 3.222 | 3.646 | 3.965 |
| 18 | .688 | .862 | 1.067 | 1.330 | 1.734 | 2.101 | 2.214 | 2.552 | 2.878 | 3.197 | 3.611 | 3.922 |
| 19 | .688 | .861 | 1.066 | 1.328 | 1.729 | 2.093 | 2.205 | 2.539 | 2.861 | 3.174 | 3.579 | 3.883 |
| 20 | .687 | .860 | 1.064 | 1.325 | 1.725 | 2.086 | 2.197 | 2.528 | 2.845 | 3.153 | 3.552 | 3.850 |
| 21 | .686 | .859 | 1.063 | 1.323 | 1.721 | 2.080 | 2.189 | 2.518 | 2.831 | 3.135 | 3.527 | 3.819 |
| 22 | .686 | .858 | 1.061 | 1.321 | 1.717 | 2.074 | 2.183 | 2.508 | 2.819 | 3.119 | 3.505 | 3.792 |
| 23 | .685 | .858 | 1.060 | 1.319 | 1.714 | 2.069 | 2.177 | 2.500 | 2.807 | 3.104 | 3.485 | 3.768 |
| 24 | .685 | .857 | 1.059 | 1.318 | 1.711 | 2.064 | 2.172 | 2.492 | 2.797 | 3.091 | 3.467 | 3.745 |
| 25 | .684 | .856 | 1.058 | 1.316 | 1.708 | 2.060 | 2.167 | 2.485 | 2.787 | 3.078 | 3.450 | 3.725 |
| 26 | .684 | .856 | 1.058 | 1.315 | 1.706 | 2.056 | 2.162 | 2.479 | 2.779 | 3.067 | 3.435 | 3.707 |
| 27 | .684 | .855 | 1.057 | 1.314 | 1.703 | 2.052 | 2.158 | 2.473 | 2.771 | 3.057 | 3.421 | 3.690 |
| 28 | .683 | .855 | 1.056 | 1.313 | 1.701 | 2.048 | 2.154 | 2.467 | 2.763 | 3.047 | 3.408 | 3.674 |
| 29 | .683 | .854 | 1.055 | 1.311 | 1.699 | 2.045 | 2.150 | 2.462 | 2.756 | 3.038 | 3.396 | 3.659 |
| 30 | .683 | .854 | 1.055 | 1.310 | 1.697 | 2.042 | 2.147 | 2.457 | 2.750 | 3.030 | 3.385 | 3.646 |
| 40 | .681 | .851 | 1.050 | 1.303 | 1.684 | 2.021 | 2.123 | 2.423 | 2.704 | 2.971 | 3.307 | 3.551 |
| 50 | .679 | .849 | 1.047 | 1.299 | 1.676 | 2.009 | 2.109 | 2.403 | 2.678 | 2.937 | 3.261 | 3.496 |
| 60 | .679 | .848 | 1.045 | 1.296 | 1.671 | 2.000 | 2.099 | 2.390 | 2.660 | 2.915 | 3.232 | 3.460 |
| 80 | .678 | .846 | 1.043 | 1.292 | 1.664 | 1.990 | 2.088 | 2.374 | 2.639 | 2.887 | 3.195 | 3.416 |
| 100 | .677 | .845 | 1.042 | 1.290 | 1.660 | 1.984 | 2.081 | 2.364 | 2.626 | 2.871 | 3.174 | 3.390 |
| 1000 | .675 | .842 | 1.037 | 1.282 | 1.646 | 1.962 | 2.056 | 2.330 | 2.581 | 2.813 | 3.098 | 3.300 |
| ∞ | .674 | .841 | 1.036 | 1.282 | 1.645 | 1.960 | 2.054 | 2.326 | 2.576 | 2.807 | 3.091 | 3.291 |
| | 50% | 60% | 70% | 80% | 90% | 95% | 96% | 98% | 99% | 99.5% | 99.8% | 99.9% |
| | Confidence level $C$ | | | | | | | | | | | |

# Example of Single Sample *t* Test

- The mean emission of all engines of a new design needs to be below 20ppm if the design is to meet new emission requirements. Ten engines are manufactured for testing purposes, and the emission level of each is determined. Data:
- 15.6, 16.2, 22.5, 20.5, 16.4, 19.4, 16.6, 17.9, 12.7, 13.9
- Does the data supply sufficient evidence to conclude that type of engine meets the new standard, assuming we are willing to risk a Type I error (false alarm, reject the Null when it is true)) with a probability = 0.01?
- **Step 1: Assumptions:** dependent variable is scale, Randomization, Normal Distribution
- **Step 2: State $H_0$ and $H_1$:**
  - **$H_0$** Emissions are equal to (or greater than) 20ppm;
  - **$H_1$** Emissions are lesser than 20ppm (One-Tailed Test)

# Example of Single Sample *t* Test

- The mean emission of all engines of a new design needs to be below 20ppm if the design is to meet new emission requirements. Ten engines are manufactured for testing purposes, and the emission level of each is determined. Data:
- 15.6, 16.2, 22.5, 20.5, 16.4, 19.4, 16.6, 17.9, 12.7, 13.9
- **Step 3: Determine Characteristics of Sample**

  Mean =

  Standard Deviation of Sample =

  Standard Error of Sample =

$$s = \sqrt{\frac{\Sigma(X - M)^2}{N - 1}}$$

$$s_m = \frac{s}{\sqrt{N}}$$

- **Step 4: Determine Cutoff**
  - **df = N-1 = 10-1 =9**
  - ***t* statistic cut-off = -2.822**

# Example of Single Sample *t* Test

- The mean emission of all engines of a new design needs to be below 20ppm if the design is to meet new emission requirements. Ten engines are manufactured for testing purposes, and the emission level of each is determined. Data:
- 15.6, 16.2, 22.5, 20.5, 16.4, 19.4, 16.6, 17.9, 12.7, 13.9
- **Step 3: Determine Characteristics of Sample**

  Mean M = 17.17

  Standard Deviation of Sample $s$ = 2.98

  Standard Error of Sample $s_m$ = 0.942

- **Step 4: Determine Cutoff**
  - **df = N-1 = 10-1 =9**
  - ***t* statistic cut-off = -2.822**

$$s_m = \frac{s}{\sqrt{N}} \qquad s = \sqrt{\frac{\Sigma(X-M)^2}{N-1}}$$

$$t = \frac{(M - \mu_m)}{s_m}$$

# Example of Single Sample *t* Test

- The mean emission of all engines of a new design needs to be below 20ppm if the design is to meet new emission requirements. Ten engines are manufactured for testing purposes, and the emission level of each is determined. Data:
- 15.6, 16.2, 22.5, 20.5, 16.4, 19.4, 16.6, 17.9, 12.7, 13.9
- Mean M = 17.17 Standard Deviation of Sample ***S*** = 2.98
  Standard Error of Sample $s_m$ = 0.942

$$t = \frac{(M - \mu_m)}{s_m} = \frac{(17.17 - 20)}{0.942} = -3.00$$