Supervised Learning and Performance

Machine Learning and Deep Learning
Lesson #3

AImage Lab
University of Modena and Reggio Emilia

The data and the goal

- Data: A set of data records (also called examples, instances or cases) described by
  - k **attributes**: A_1, A_2, … A_k.
  - a class: Each example is labelled with a pre-defined class.
- Goal: To learn a classification model from the data that can be used to predict the classes of new (future, or test) cases/instances.

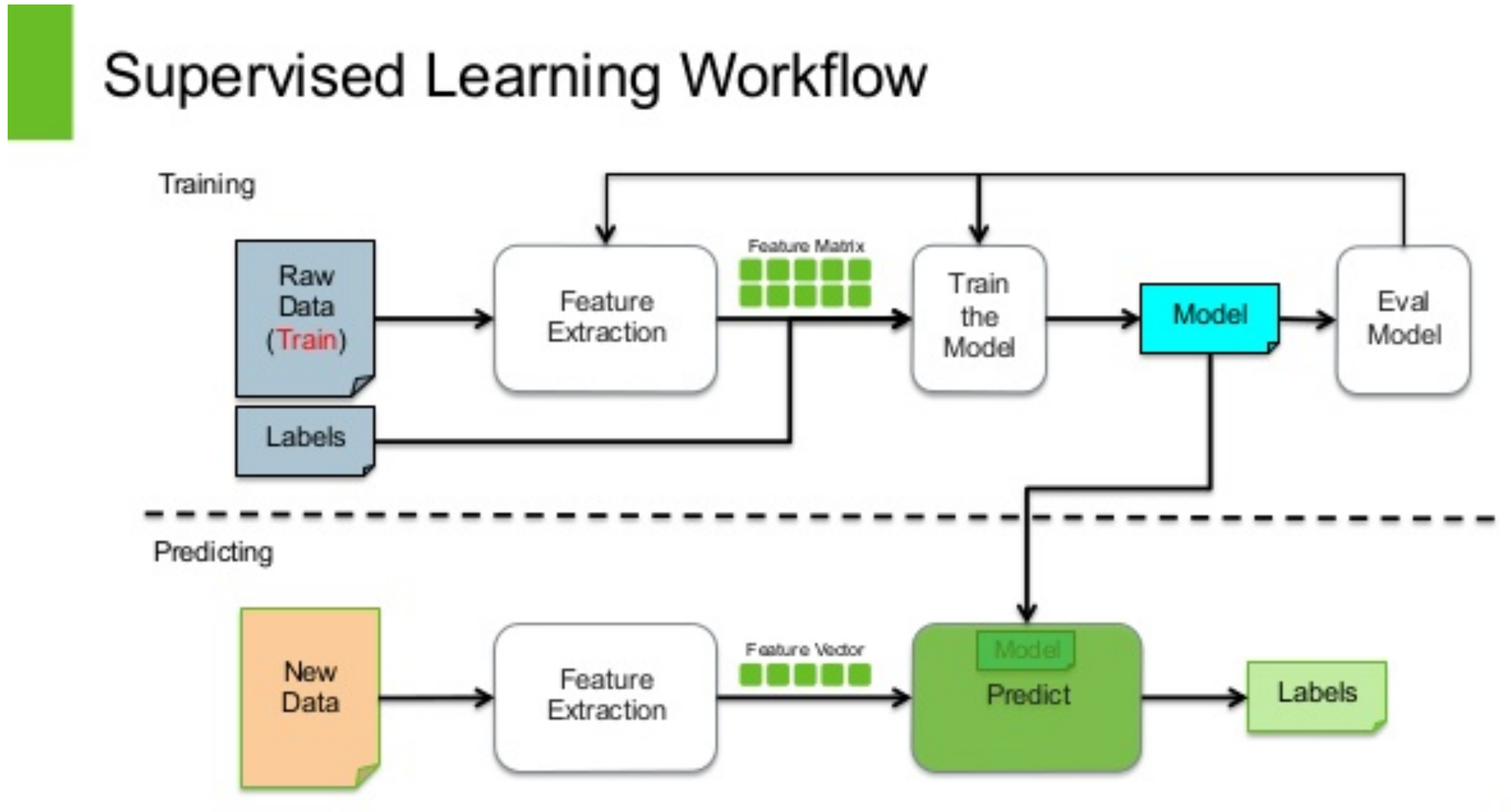
Supervised vs. unsupervised Learning

- Supervised learning: classification is seen as supervised learning from examples.
  - Supervision: The data (observations, measurements, etc.) are labeled with pre-defined classes. It is like that a “teacher” gives the classes (supervision).
  - Test data are classified into these classes too.
- ~~Unsupervised learning (clustering)~~ DONE ☺️
  - Class labels of the data are unknown
  - Given a set of data, the task is to establish the existence of classes or clusters in the data

Supervised learning process: two steps

1. Learning (training): Learn a model using the training data

2. Testing: Test the model using unseen test data to assess the model accuracy



$$Accuracy = \frac{\text{Number of correct classifications}}{\text{Total number of test cases}},$$

What do we mean by learning?

- Given
  - a data set *D*,
  - a task *T*, and
  - a performance measure *M*,

  a computer system is said to learn from *D* to perform the task *T* if after learning the system's performance on *T* improves as measured by *M*.

- In other words, the learned model helps the system to perform *T* better as compared to no learning.

Fundamental assumption of learning

Assumption: The distribution of training examples is identical to the distribution of test examples (including future unseen examples).

- In practice, this assumption is often violated to certain degree.
- Strong violations will clearly result in poor classification accuracy.
- To achieve good accuracy on the test data, training examples must be sufficiently representative of the test data.

EVALUATING CLASSIFICATION METHODS

- Predictive accuracy

$$Accuracy = \frac{\text{Number of correct classifications}}{\text{Total number of test cases}}$$

- Efficiency
  - time to construct the model
  - time to use the model
- Robustness: handling noise and missing values
- Scalability: efficiency in disk-resident databases
- Interpretability:
  - understandable and insight provided by the model
- Compactness of the model: size of the tree, or the number of rules.

Evaluation methods

- Holdout set: The available data set *D* is divided into two disjoint subsets,
  - the ***training set*** D_{train} (for learning a model)
  - the ***test set*** D_{test} (for testing the model)

- Important: training set should not be used in testing and the test set should not be used in learning.
  - Unseen test set provides a unbiased estimate of accuracy.

- The test set is also called the holdout set. (the examples in the original data set *D* are all labeled with classes.)

- This method is mainly used when the data set *D* is large.

Evaluation methods (cont…)

- n-fold cross-validation: The available data is partitioned into ***n*** equal-size disjoint subsets.

  - Use each subset as the test set and combine the rest ***n***-1 subsets as the training set to learn a classifier.

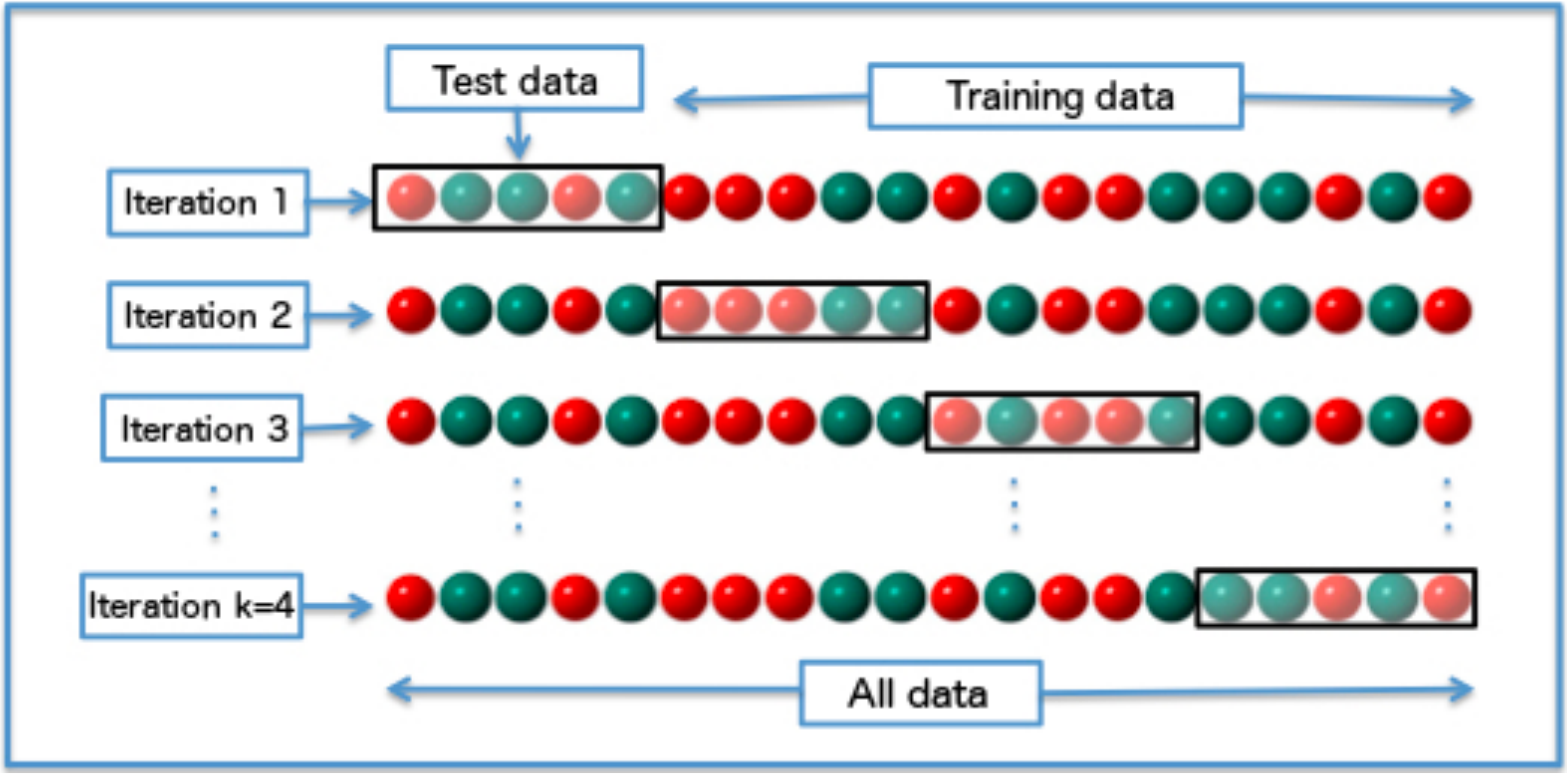
  - The procedure is run ***n*** times, which give ***n*** accuracies.

  - The final estimated accuracy of learning is the average of the ***n*** accuracies.

  - 10-fold and 5-fold cross-validations are commonly used.

- This method is used when the available data is not large.

Cross Validation



Evaluation methods (cont…)

- Leave-one-out cross-validation: This method is used when the data set is very small.

  - It is a special case of cross-validation

  - Each fold of the cross validation has only a single test example and all the rest of the data is used in training.

  - If the original data has m examples, this is m-fold cross-validation

Evaluation methods (cont…)

- Validation set: the available data is divided into three subsets,
  - a training set,
  - a validation set and
  - a test set.

- A validation set is used frequently for **estimating parameters** in learning algorithms.

- In such cases, the **values that give the best accuracy** on the validation set are used as the final parameter values.

- **Cross-validation** can be used for parameter estimating as well.

Classification measures

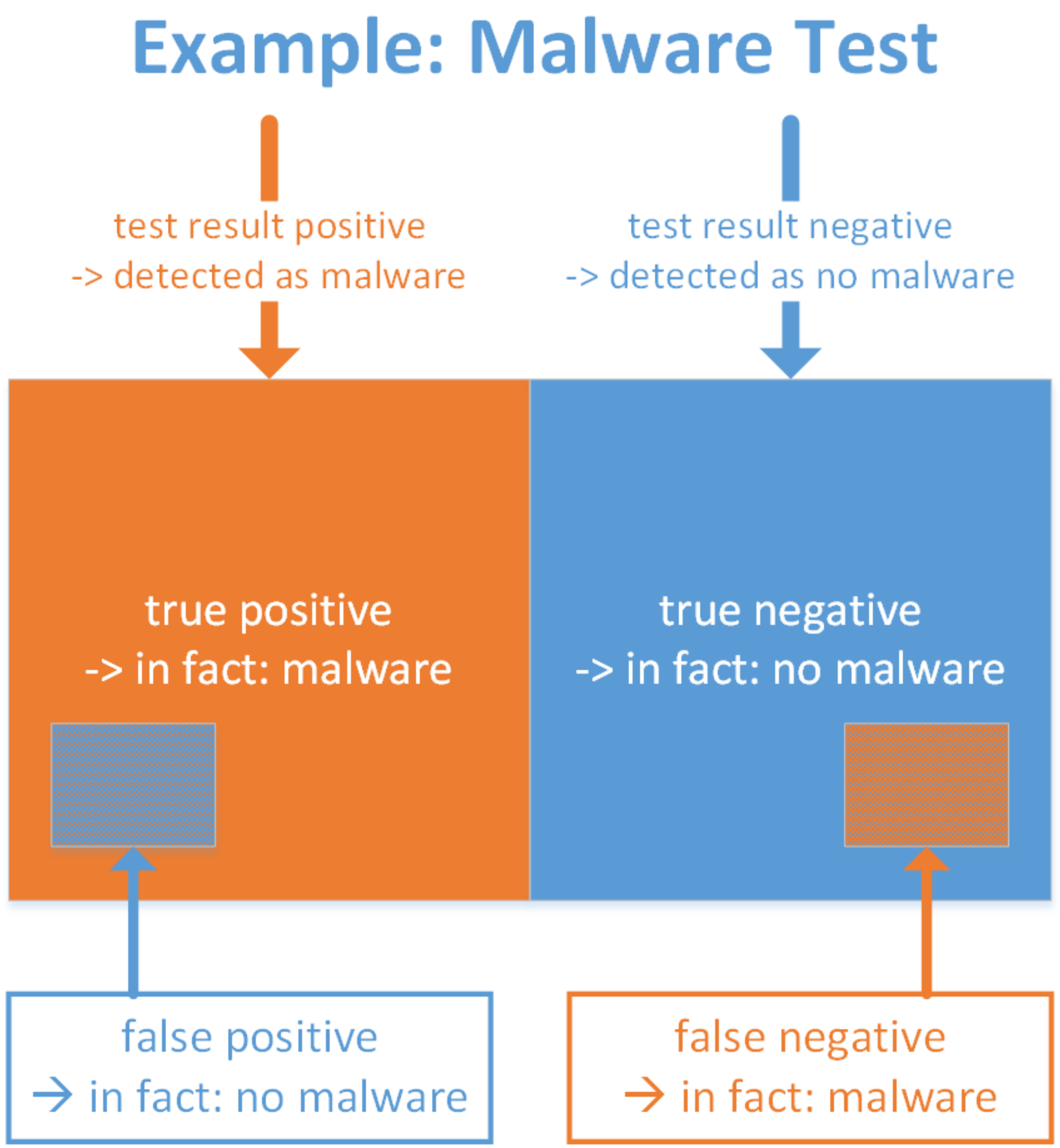
- Accuracy is **only one measure** (error = 1-accuracy).
- Accuracy is **not suitable** in some applications.
- In text mining, we may only be interested in the documents of a particular topic, which are only a small portion of a big document collection.
- In classification involving skewed or highly imbalanced data, e.g., network intrusion and financial fraud detections, we are interested only in the minority class.
  - High accuracy does not mean any intrusion is detected.
  - E.g., 1% intrusion. Achieve 99% accuracy by doing nothing.
- The class of interest is commonly called the positive class, and the rest negative classes.

PRECISION AND RECALL MEASURES

- Used in **information retrieval** and text classification.
- We use a **confusion matrix** to introduce them.

		Truth		
		Positive	**Negative**	
Test	**Positive**	True Positive	False Positive Type I α	Total Testing Positive
	Negative	False Negative Type II β	True Negative	Total Testing Negative
		Total Truly Positive	Total Truly Negative	Total

EXAMPLE MALWARE DETECTION



Precision and recall measures (cont…)

$$p = \frac{TP}{TP + FP}. \qquad r = \frac{TP}{TP + FN}.$$

Precision p is the number of correctly classified positive examples divided by the total number of examples that are classified as positive.

Recall r is the number of correctly classified positive examples divided by the total number of actual positive examples in the test set.

F1-VALUE (ALSO CALLED F1-SCORE)

- It is hard to compare two classifiers using two measures. F_1 score combines precision and recall into one measure
- $F_1$is the harmonic mean of precision and recall

$$F_1 score = \frac{1}{\frac{1}{P} + \frac{1}{R}}$$

- The harmonic mean of two numbers tends to be closer to the smaller of the two.

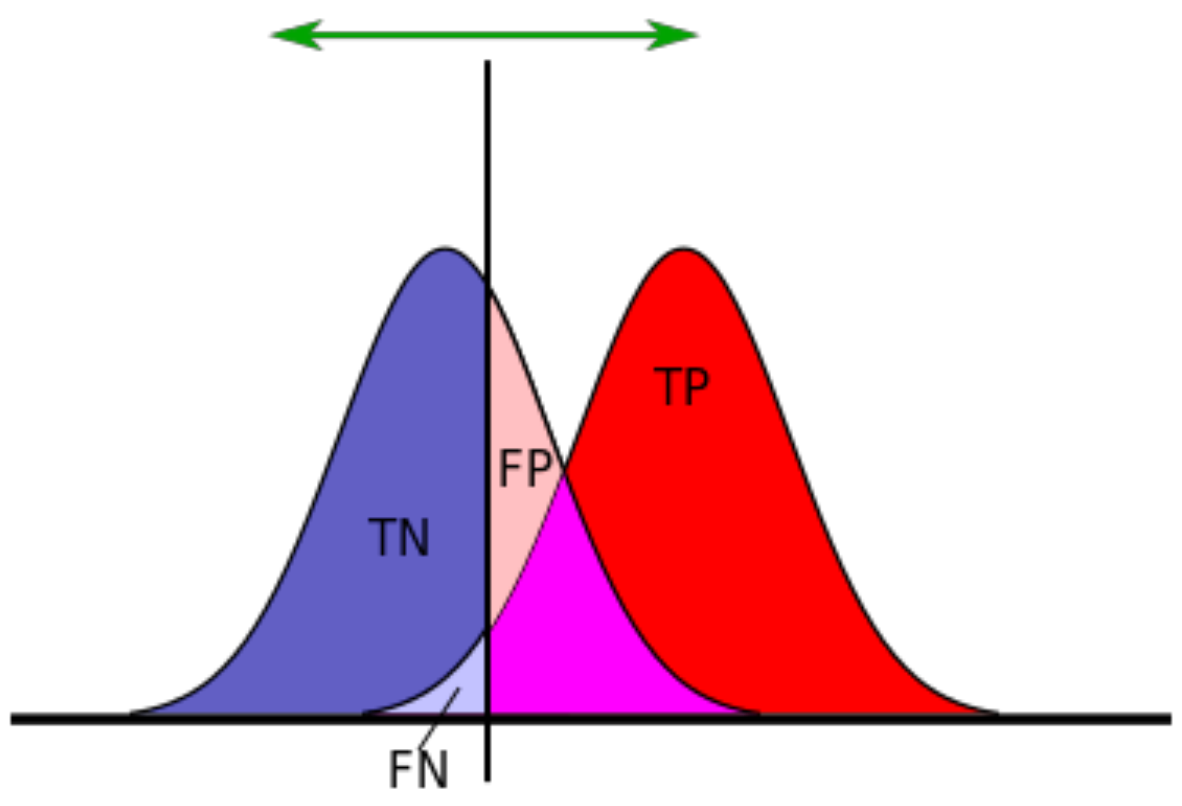
- For F_1-value to be large, both *P* and *R* much be large.

ROC Curve

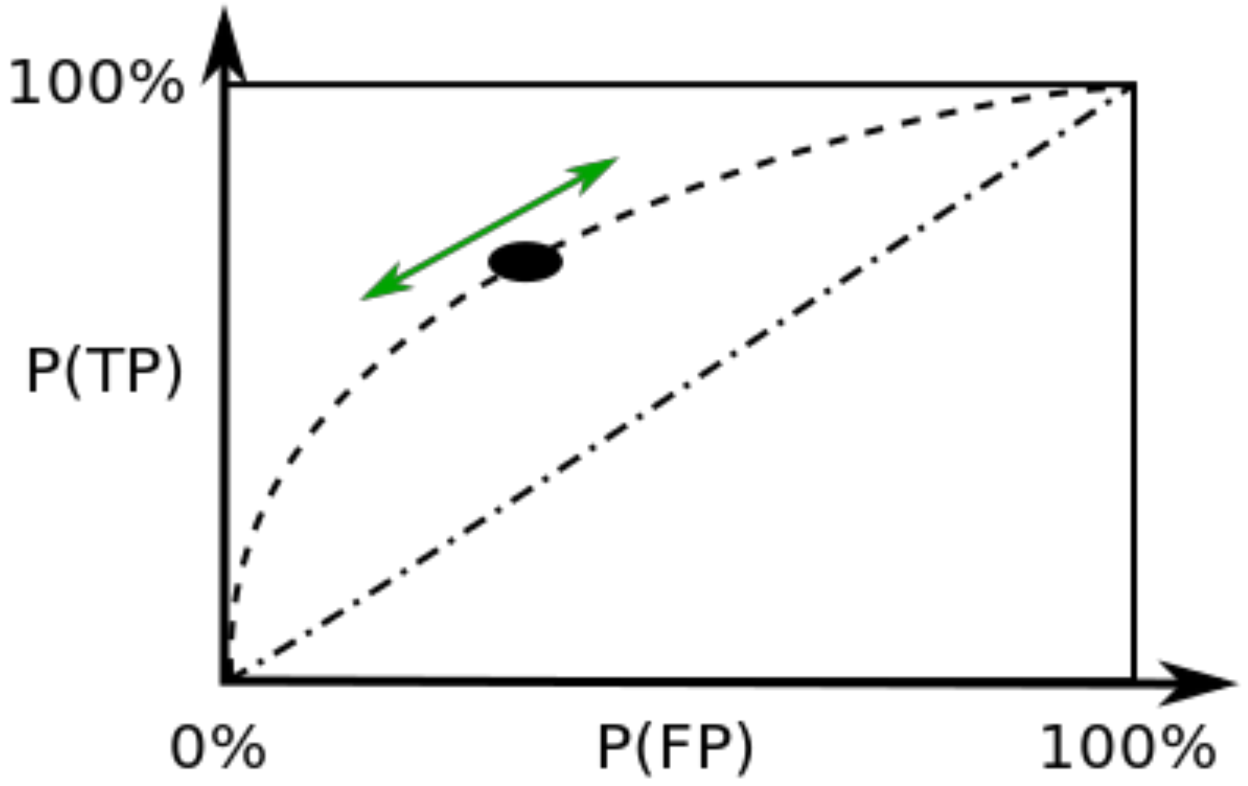
- **ROC** curve measure the correlation between precision recall
- Typically used when the **output** of the **system is a score**

- To obtain a category from a score we need a **threshold**

- **Varying threshold** varies P and R accordingly



TP	FP
FN	TN



Performance with statistical output

PERFORMANCE WITH STATISTICAL OUTPUT

- If the output is a probability measure or distribution the performance are evaluated:

1. Thresholding the probability and using discrete class values
2. Using distance between distributions

DISTRIBUTIONS DISTANCES

- **Bhattacharrya coefficient**
  - The Bhattacharyya coefficient is an approximate measurement of the amount of overlap between two statistical samples

$$BC(p, q) = \int_x p(x)\ q(x)\ dx$$

where p and q are discrete distributions

where p(x) and q(x) are probality distributions

- **KL Divergence**
  - It is a **non symmetric** measure of lost information when distribution Q is used to approx distribution P
  - Not a metric -> NON-symmetric

$$KL(P\,||\,Q) = \int_x P(x)\, log \frac{P(x)}{Q(x)} dx$$

CROSS ENTROPY

- Cross entropy is a **information theory** measure related for coding messages using number of bit
- "I*t evaluates the* ***average number of bits*** *for discovering a* ***datum coded by a distribution q*** *while the* ***original one was p***"
- it is related to **both the entropy and KL divergence**

$$H(P, Q) = H(P) + KL(P\,||\,Q)$$

where P *is the true distribution and* Q *the estimate*

Discrete case :

$$H(P, Q) = \sum p\ log(p) - \sum p\ log(p) - \sum p\ log(q) = -\sum p\ log(q)$$

- It make use of **Kraft–McMillan** theorem that states a value x_i can be identified by l_i bits with probability

  $q(x_i) = 2^{-l_i}$

  trying to take the expected value of l w.r.t. p

  $$E_p[l] = \sum p\ log_2(\frac{1}{q}) = -\sum p\ log(q)$$

CROSS ENTROPY AND CLASSIFICATION

- **Cross entropy** can be used as a measure of classification Error
- Several **classifiers minimize cross entropy** as the objective measure

- Consider p a discrete distribution with k possible values and the problem being a k class classification
- suppose p_i (x)=1 iff x belongs to class i
- Suppose q_i(x) the probability the classifier attributes to class i for element x
- The Expected cross Entropy over the dataset D of N elements is

$$E_D[H(p,q)] = \frac{1}{N}\sum_{x \in D}\left(-\sum_i p_i(x)\ log\ q_i(x)\right)$$

where every element of D has probability 1/N

BINARY CASE

- In the binary case only **2 classes exists** and
  - if it is y the **probabilty p** of one class it is 1-y the probability for the second class
  - if it is y' the **probabilty q** of one class 1-y' is the probability for the second class

- Recalling the original definition

$$E_D[H(p,q)] = \frac{1}{N}\sum_{x\in D}\left(-\sum_i p_i(x)\; log\; q_i(x)\right)$$

i goes from 1 to 2 and the inner summation and with **p=y and q=y'** reduces to

$$E_D[H(p,q)] = \frac{1}{N}\sum_{x\in D}\left(y\, log(y') + (1-y)\, log(1-y')\right)$$

That is also the objective of logistic regression classifier.