INTRODUCTION TO MACHINE LEARNING

Challenges, Trends and Solutions in Life Sciences

Fotis E. Psomopoulos
INAB|CERTH

Shakuntala Baichoo
University of Mauritius

H3ABioNet

CERTH
CENTRE FOR RESEARCH & TECHNOLOGY HELLAS

INAB
INSTITUTE OF APPLIED BIOSCIENCES
ΙΝΣΤΙΤΟΥΤΟ ΕΦΑΡΜΟΣΜΕΝΩΝ ΒΙΟΕΠΙΣΤΗΜΩΝ
CERTH

SOME HOUSEKEEPING

Please remain muted throughout the course, unless you are invited to speak by the Instructors

Please use the "hand-raising function" to indicate you would like to contribute directly

Cameras are optional but might lead to bandwidth issues

This meeting will be run in line with the ELIXIR Code of Conduct. If you have any concerns, please refer to the Code of Conduct, found on the ELIXIR website

Please use the "Chat" or the Gdoc to raise questions for further discussions

If you have any questions during the course, please use the Gdoc document

It would be helpful to include the Country abbreviation after your name using the "rename" function.

Example: Johann Schmidt (DE)

CODE OF CONDUCT

Our values: a place to feel respected, a place to feel safe!

This course falls under the **ELIXIR Hub Code of Conduct** (full document here)

As defined in the ELIXIR Hub Code of Conduct, we encourage the following kinds of behaviours:

- Use welcoming and inclusive language
- Be respectful of different viewpoints and experiences
- Foster scientific and technical rigour and curiosity with constructive and facts-based critique
- Gracefully accept constructive criticism
- Show courtesy and respect towards other participants
- Be mindful of your own biases and do not let them get in the way of respectful interaction
- Speak up if you believe the spirit of the Code has not been upheld. Ideally, where feasible, directly address the issue with the person who committed the transgression
- Adjust the behaviour where it was seen to be short of the requirements indicated in this Code.

A QUICK ROUND OF INTRODUCTIONS

Shakuntala Baichoo

- Associate Professor
- University of Mauritius

Wandrille Duchemin

- Bioinformatics Support and Training
- Swiss Institute of Bioinformatics – SIB / sciCORE UNIBAS

Geert van Geest

- Bioinformatician and Trainer
- Swiss Institute of Bioinformatics - SIB/Universität Bern, CH

Thuong Van Du Tran

- Staff scientist
- Vital-IT, SIB Swiss Institute of Bioinformatics

Fotis Psomopoulos

- Assistant Research Professor
- Institute of Applied Biosciences Centre for Research and Technology Hellas

Monique Zahn

- Training Manager and neXtProt Quality Manager
- SIB Swiss Institute of Bioinformatics

COMMUNICATION

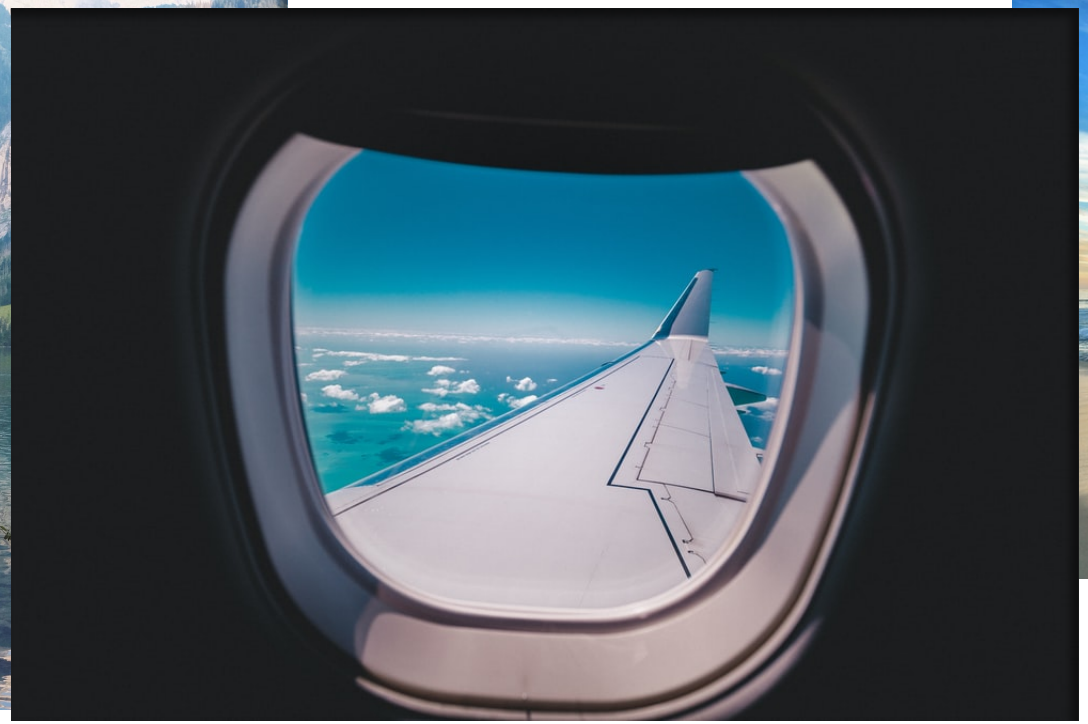
We will be using GDoc to exchange information:

https://tinyurl.com/sib-ml-course-2020

Access the Gdoc and sign in! ☺

AN ICEBREAKER

"If I could be on vacation anywhere right now (pandemic-free 😂), I'd go to…, because…"



COURSE AGENDA

09:00 - 09:30	**Day 1: Course Introduction.** - Welcome. - Introduction and CoC. - Way to interact - Practicalities (agenda, breaks, etc). - Setup
09:30 - 10:00	**Introduction to Machine Learning** (*theory*)
10:00 - 11:30	**What is Exploratory Data Analysis (EDA) and why is it useful?** (*hands-on*) - Loading omics data - PCA
11:30 - 11:45	*Coffee Break*
11:45 - 12:15	**Introduction to Unsupervised Learning** (*theory*)
12:15 - 13:00	**Agglomerative Clustering: k-means** (*practical*)
13:00 - 14:00	*Lunch break*
14:00 - 14:45	**Agglomerative Clustering: k-means** (*practical*) (cont'd)
14:45 - 15:30	**Divisive Clustering: hierarchical clustering** (*practical*)
15:30 - 15:45	*Coffee Break*
15:45 - 16:30	**Divisive Clustering: hierarchical clustering** (*practical*) (cont'd)
16:30	*Closing of Day 1*

09:00 - 09:30	**Welcome Day 2.** - Questions from Day 1 - Agenda
09:30 - 10:00	**Introduction to Supervised Learning** (*theory*) - Overview of multiple algorithms - Advantages and Disadvantages
10:00 - 10:30	**Classification Metrics** (*theory*) - F1 Score, Precision, Recall - Confusion Matrix, ROC-AUC
10:30 - 11:30	**Classification** (*practical*) - Decision trees - Random Forests
11:30 - 11:45	*Coffee Break*
11:45 - 12:30	**Classification** (*practical*) (cont'd)
12:30 - 13:30	*Lunch break*
13:30 - 14:00	**Regression** (*theory*)
14:00 - 15:15	**Regression** (*practical*) - Linear regression - Generalized Linear Model (GLM)
15:15 - 15:30	*Coffee Break*
15:30 - 16:00	**Regression** (*practical*) (cont'd)
16:00 - 16:30	*Closing questions, Discussion*

SESSION OVERVIEW

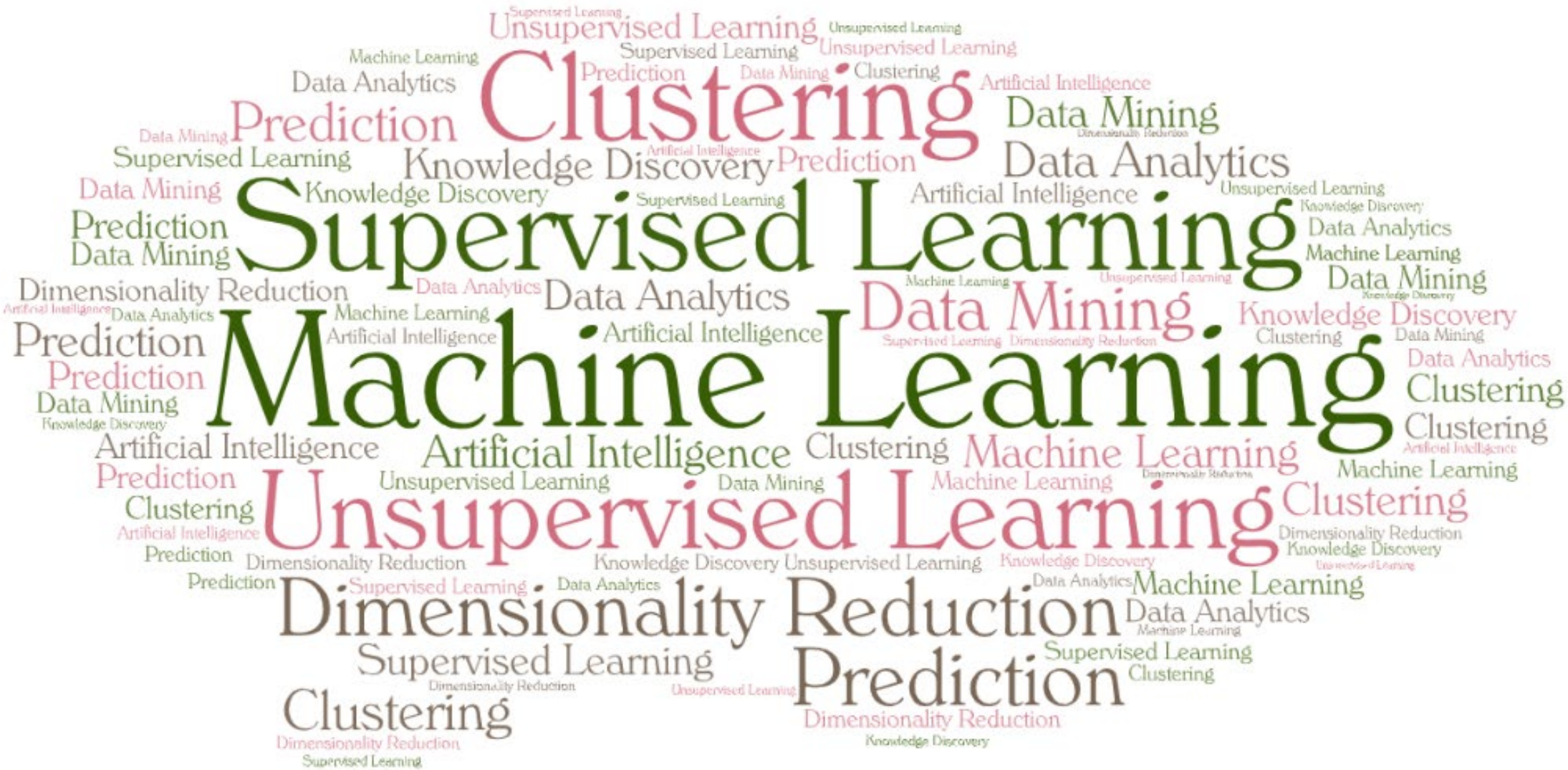
01 • Introduction to basic concepts of Data mining and Machine learning

02 • Machine learning taxonomy

03 • Supervised classification vs unsupervised classification

04 • Algorithms examples

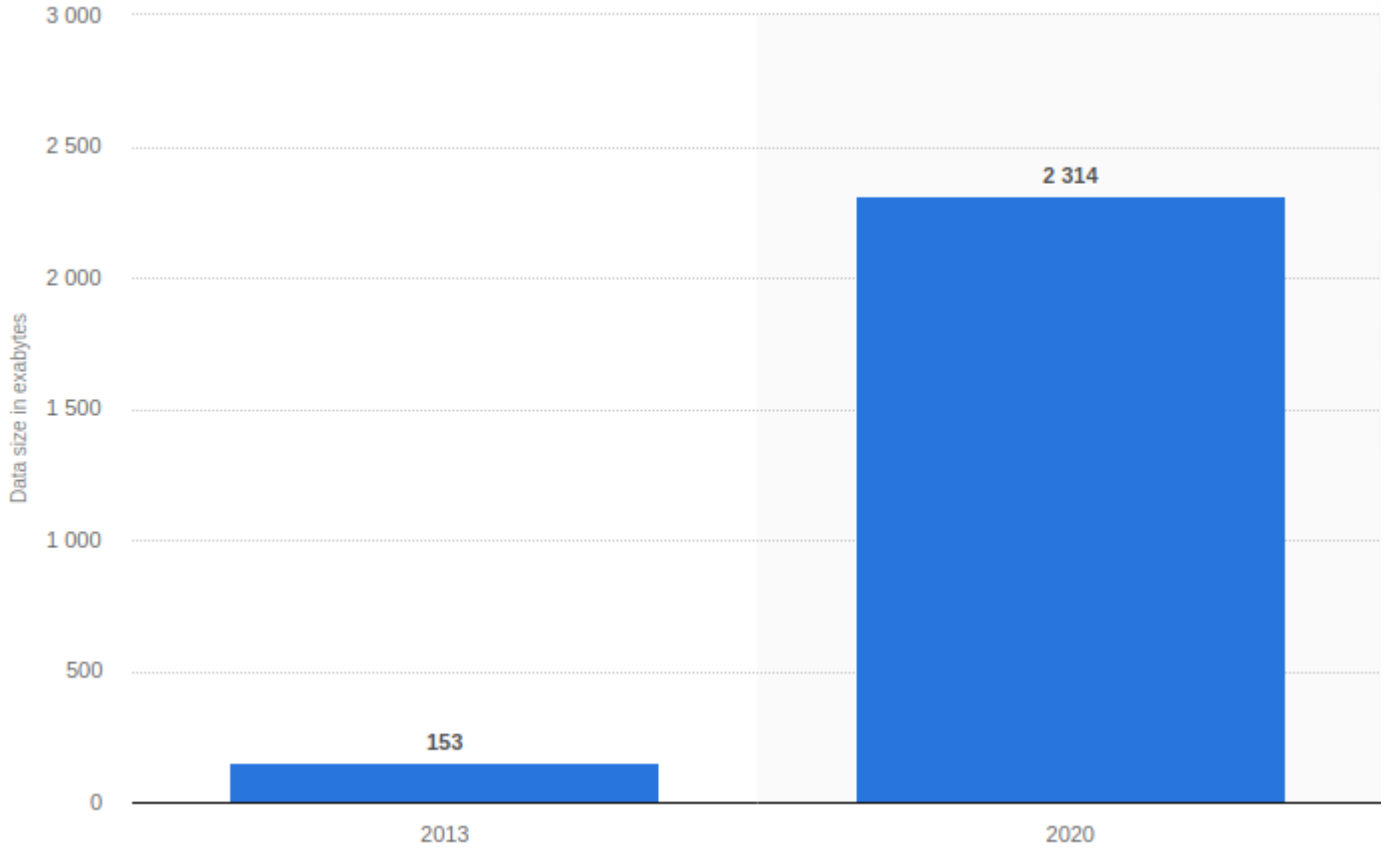
05 • Examples of applications in Bioinformatics


Supervised Learning
Unsupervised Learning
Unsupervised Learning
Machine Learning
Supervised Learning
Unsupervised Learning
Data Analytics
Prediction
Data Mining
Clustering
Artificial Intelligence
Clustering
Prediction
Data Mining
Dimensionality Reduction
Supervised Learning
Artificial Intelligence
Knowledge Discovery
Prediction
Data Analytics
Data Mining
Knowledge Discovery
Supervised Learning
Artificial Intelligence
Unsupervised Learning
Knowledge Discovery
Prediction
Supervised Learning
Data Analytics
Data Mining
Machine Learning
Dimensionality Reduction
Data Analytics
Data Analytics
Machine Learning
Unsupervised Learning
Data Mining
Artificial Intelligence
Data Analytics
Machine Learning
Data Mining
Knowledge Discovery
Prediction
Artificial Intelligence
Artificial Intelligence
Supervised Learning
Dimensionality Reduction
Clustering
Data Mining
Prediction
Data Analytics
Data Mining
Clustering
Knowledge Discovery
Artificial Intelligence
Artificial Intelligence
Clustering
Machine Learning
Prediction
Unsupervised Learning
Data Mining
Machine Learning
Machine Learning
Clustering
Clustering
Artificial Intelligence
Prediction
Dimensionality Reduction
Knowledge Discovery Unsupervised Learning
Knowledge Discovery
Prediction
Supervised Learning
Data Analytics
Data Analytics
Machine Learning
Dimensionality Reduction
Data Analytics
Machine Learning
Supervised Learning
Clustering
Dimensionality Reduction
Unsupervised Learning
Prediction
Clustering
Dimensionality Reduction
Supervised Learning

Technology Timeline

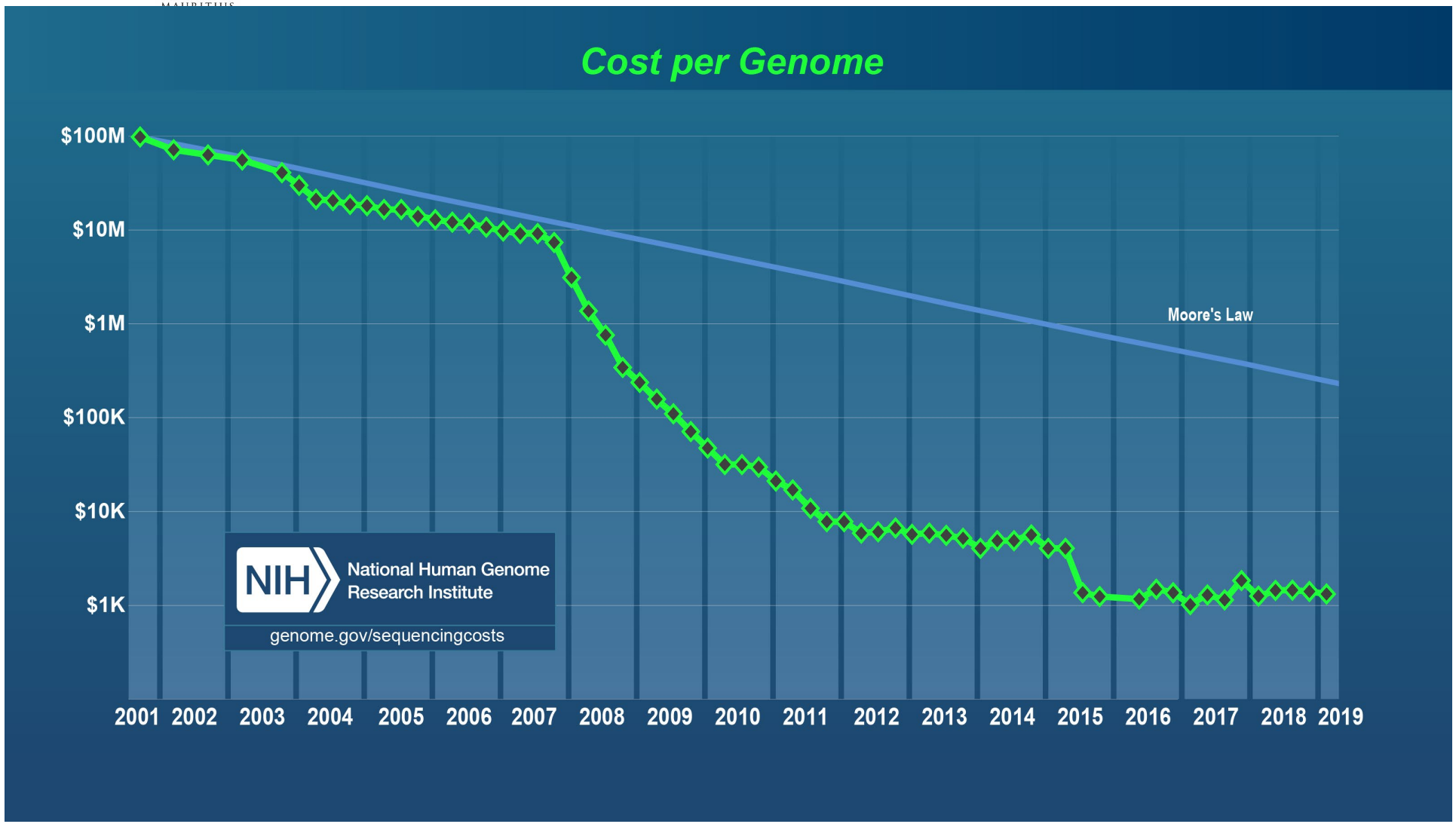
https://www.linkedin.com/pulse/technology-increase-vs-department-budgets-sam-errington/

TOTAL AMOUNT OF GLOBAL HEALTHCARE DATA GENERATED AND PROJECTIONS FOR END 2020 (IN EXABYTES)



1 exabyte = 1 billion gigabytes

Source: https://www.statista.com/statistics/1037970/global-healthcare-data-volume/



https://www.genome.gov/about-genomics/fact-sheets/DNA-Sequencing-Costs-Data

FROM DATA TO KNOWLEDGE

AI & ML

AI is a broader concept than ML which addresses the use of computers to mimic the cognitive functions of humans.

When machines carry out tasks based on algorithms in an intelligent manner, that is AI

ML is a subset of AI and focuses on the ability of machines to receive a set of data and learn from it, improve algorithms as they learn more about information being processed

ML & DATA MINING

ML embodies the principles of DM

DM and ML have the same foundation but in different ways

- DM requires human interaction
- DM can't see the relationship between different data aspects with the same depth as ML
- ML learns from the data and allows the machine to teach itself

DM is typically used as an information source for ML to pull from

ML is more about building the prediction model

AI, ML & DM

Data mining produces insights

ML produces predictions

AI produces actions

https://medium.freecodecamp.org/using-machine-learning-to-predict-the-quality-of-wines-9e2e13d7480d

DEEP LEARNING

Deep learning is a subset of ML

Deep learning algorithms go a level deeper than classical ML involving many layers

Layers: set of nested hierarchy of related concepts

The answer to a question is obtained by answering other related deeper questions

DATA IS AT THE HEART OF ML

Machine learning algorithms are driven by the data used

Data quality is very important!

Identifying incomplete, incorrect and irrelevant parts of the data is an important step

Preprocessing data before applying ML is crucial step

HOW DO WE HUMAN MAKE DECISIONS? DO WE ALL MAKE THE SAME DECISIONS?

Observations

Experiences

External information

Beliefs, creativity, common sense

Compare to expectations

Analyze differences

Creativity, Limited memory

HOW DOES A COMPUTER WORK?

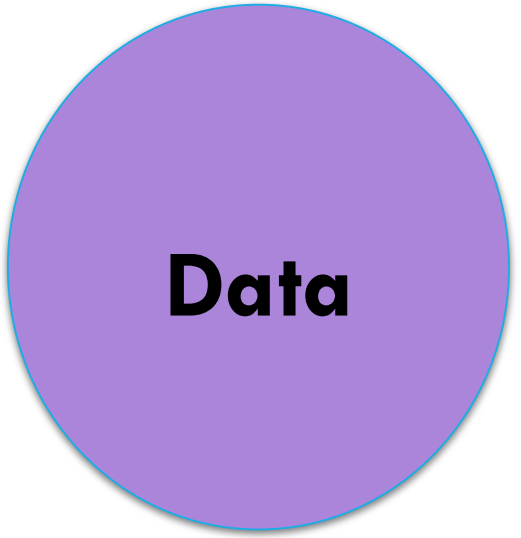
Follow instructions given by human

ARTIFICIAL INTELLIGENCE

Stimulate human behavior and cognitive process

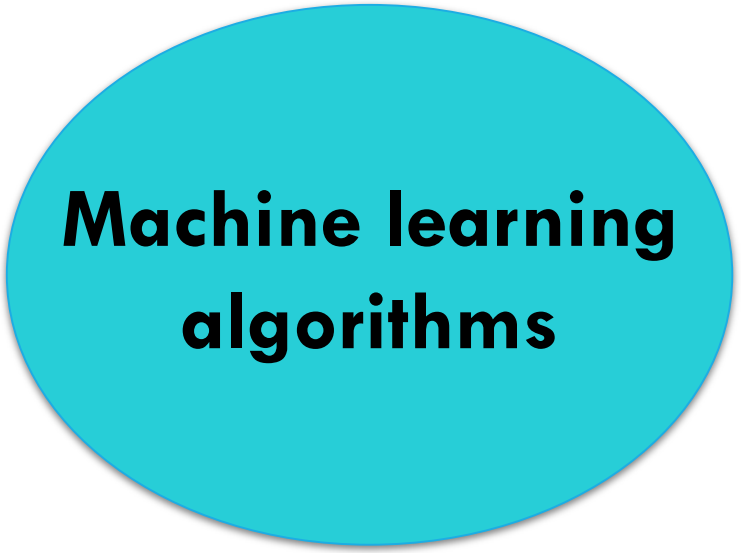
Capture and preserve human expertise

Fast response
Ability to memorize big amounts of data



Computing
+
Storage

ARTIFICIAL INTELLIGENCE



Data

Results
Prediction and
Rules

HOW DO MACHINES LEARN?

Data to model

Decision

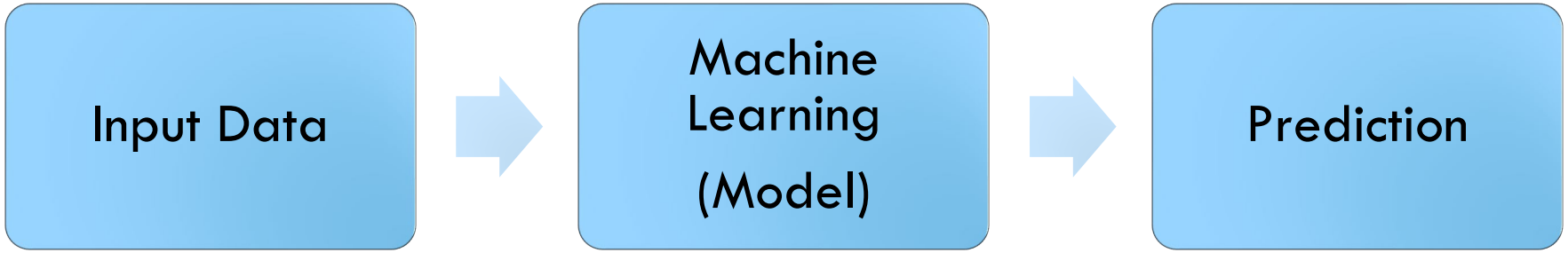
Create models

Evaluate models

Refine models

Prediction, categorization

WHAT IS MACHINE LEARNING?



Learning begins with observations or data

- Examples: direct experience, or instruction

The system looks for patterns in data and makes better decisions in the future based on the examples that we provide

The primary aim is to allow the computers learn automatically without human intervention or assistance and adjust actions accordingly.

MACHINE LEARNING AND GENOMICS

In the context of genome annotation, a machine learning system can be used to:

- 'learn' how to recognize the locations of transcription start sites (TSSs) in a genome sequence
- identify splice sites and promoters

In general, if one can compile a list of sequence elements of a given type, then a machine learning method can probably be trained to recognize those elements.

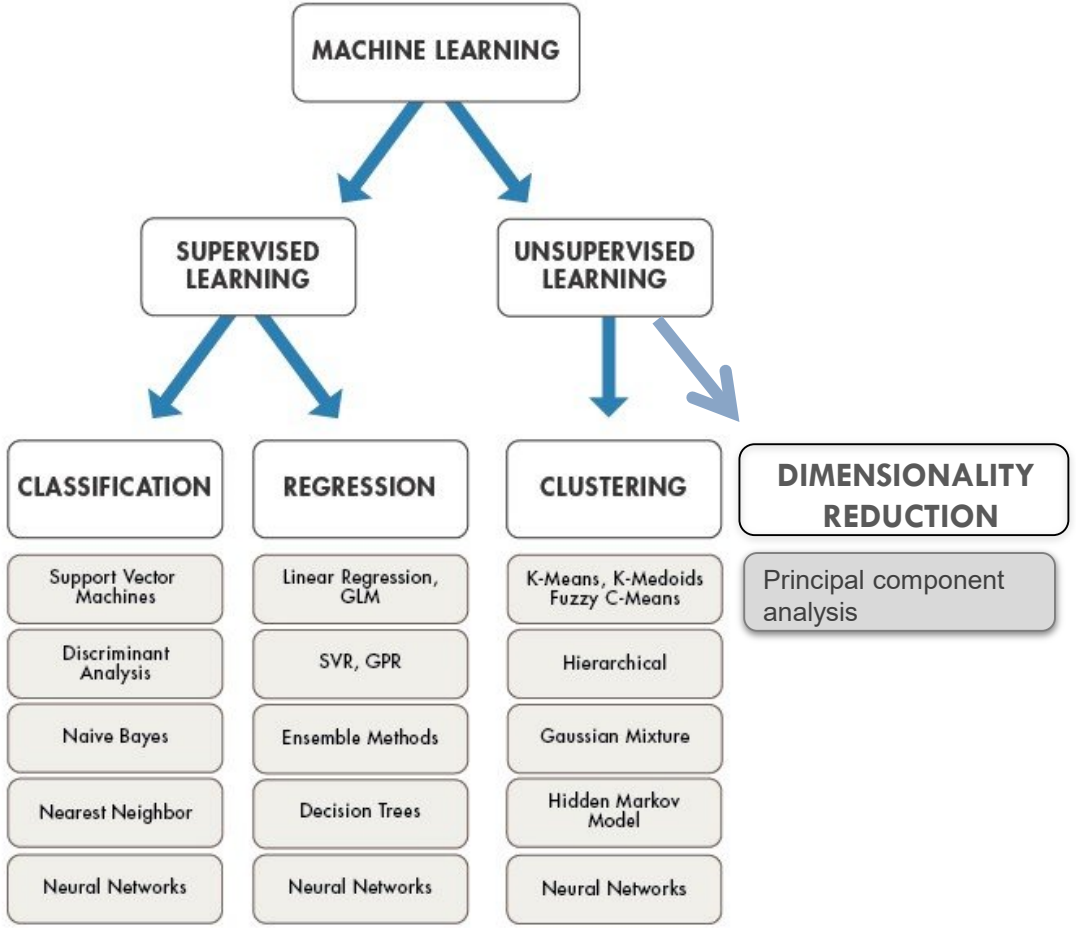
More info about this task can be obtained from: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5204302/)

MACHINE LEARNING CONCEPTS

Any machine learning problem can be represented with the following three concepts:

- We will have to learn to solve a task T.
  - For example, perform genome annotation.
- We will need some experience E to learn to perform the task. Usually, experience is represented through a dataset.
  - For the gene prediction, experience comes as a set of DNA sequences provided as input to a learning procedure, along with binary labels indicating whether each sequence is centered on a TSS or not. The learning algorithm produces a model which can then be subsequently used, in conjunction with a prediction algorithm, to assign predicted labels to unlabeled test sequences.
- We will need a measure of performance P to know how well we are solving the task and also to know whether after doing some modifications, our results are improving or getting worse.
  - The percentage of genes that our gene prediction model is correctly classifying as genes could be P for our gene prediction task.

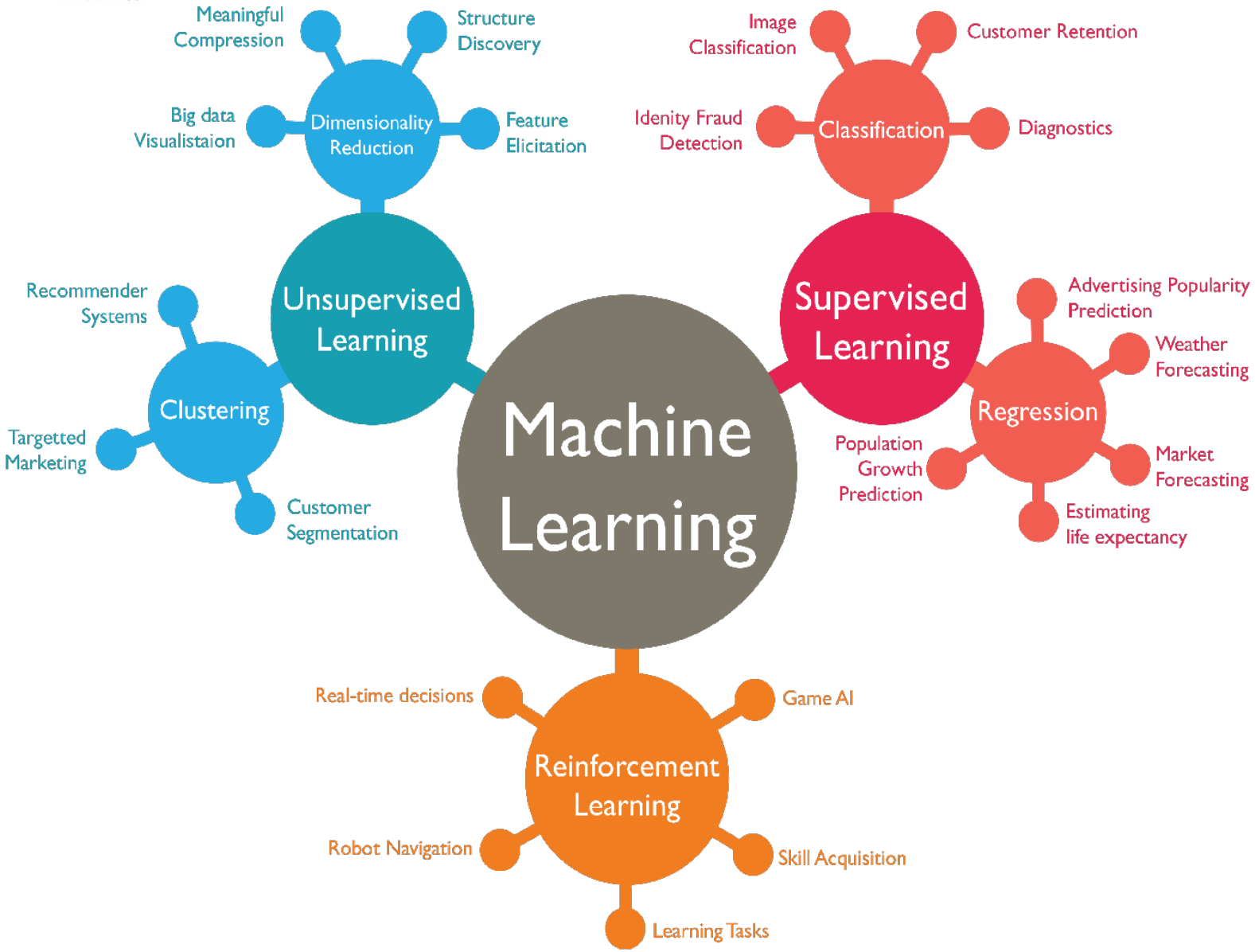
THE ML TAXONOMY



THE ML TAXONOMY

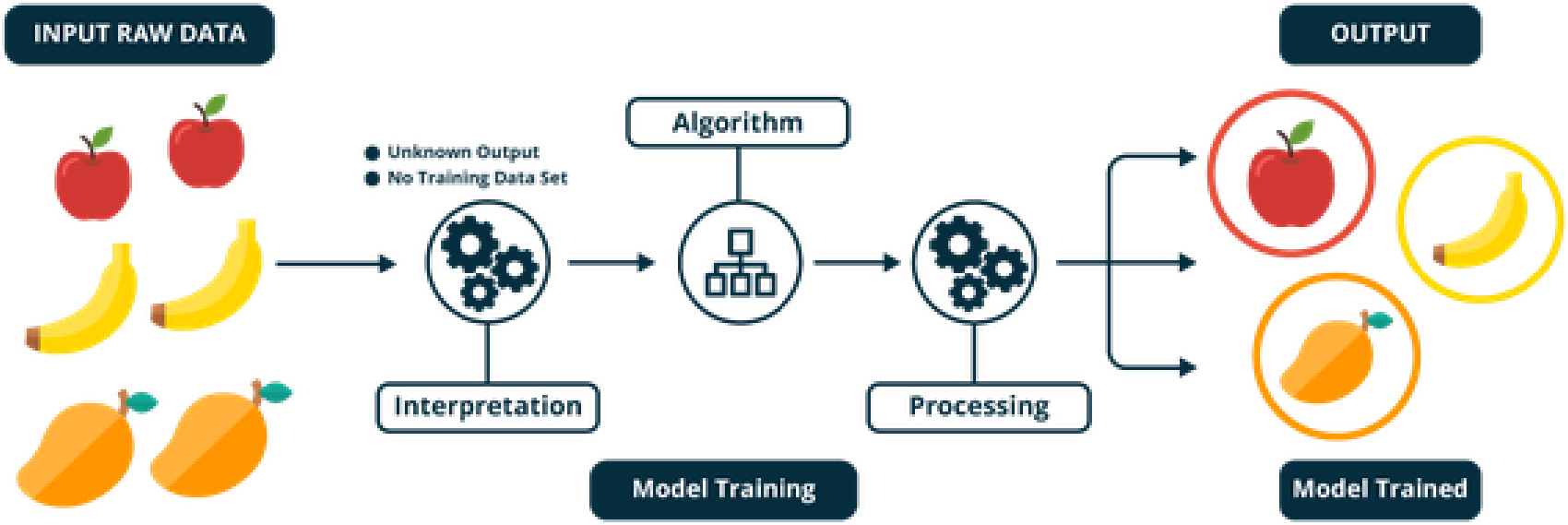
Machine learning algorithms are often categorized as **supervised** or **unsupervised**.

We also have **semi-supervised** machine learning and **reinforcement** machine learning.


Machine Learning
Unsupervised Learning
Dimensionality Reduction
Meaningful Compression
Structure Discovery
Big data Visualistaion
Feature Elicitation
Clustering
Recommender Systems
Targetted Marketing
Customer Segmentation
Supervised Learning
Classification
Image Classification
Customer Retention
Idenity Fraud Detection
Diagnostics
Regression
Advertising Popularity Prediction
Weather Forecasting
Population Growth Prediction
Market Forecasting
Estimating life expectancy
Reinforcement Learning
Real-time decisions
Game AI
Robot Navigation
Skill Acquisition
Learning Tasks

Unsupervised Learning

UNSUPERVISED LEARNING



https://www.quora.com/What-is-the-difference-between-supervised-and-unsupervised-learning-algorithms

UNSUPERVISED MACHINE LEARNING ALGORITHMS[1]

In contrast to supervised machine learning algorithms, they:

- are applied when the information used to train is ***neither classified nor labeled***.
- can infer a function to describe a hidden structure from unlabeled data.
- do not figure out the right output, but explore the data and can draw inferences from datasets to describe hidden structures from unlabeled data.

The goal for unsupervised learning is to model the underlying structure or distribution in the data in order to learn more about the data.

- Algorithms are left to their own devises to discover and present the interesting structure in the input data.

UNSUPERVISED MACHINE LEARNING ALGORITHMS[2]

Unsupervised learning problems can be further grouped into clustering, association and dimensionality reduction problems:

- **Clustering**: A clustering problem is where you want to discover the inherent groupings in the data, such as clustering DNA sequences into functional groups.
- **Association**: An association rule learning problem is where you want to discover rules that describe large portions of your data, such as using association analysis-based techniques for pre-processing protein interaction networks for the task of protein function prediction.
- **Dimensionality Reduction**: Often we are working with data of high dimensionality—each observation comes with a high number of measurements—a dimension reduction procedure is usually conducted to reduce the variable space before the subsequent analysis is carried out..
  - For example in a gene-expression analysis, dimension reduction can be used to find a list of candidate genes with a more operable length ideally including all the relevant genes.
  - Leaving many uninformative genes in the analysis can lead to biased estimates and reduced power.

UNSUPERVISED MACHINE LEARNING ALGORITHMS[3]

Clustering is an exploratory data analysis technique that allows us to organize a pile of information into meaningful subgroups (clusters) without having any prior knowledge of their group memberships.

Each cluster that arises during the analysis

- defines a group of objects that share a certain degree of similarity but are more dissimilar to objects in other clusters, which is why clustering is also sometimes called **unsupervised classification**.

UNSUPERVISED MACHINE LEARNING ALGORITHMS[5]

Taking the example of the gene-finding model, when a labeled training set is not available, unsupervised learning is required.

Consider the interpretation of a heterogeneous collection of epigenomic data sets, such as those generated by the Encyclopedia of DNA Elements (ENCODE) Consortium and the Roadmap Epigenomics Project.

UNSUPERVISED MACHINE LEARNING ALGORITHMS[6]

A priori, we expect that the patterns of chromatin accessibility, histone modifications and transcription factor binding along the genome should be able to provide a detailed picture of the biochemical and functional activity of the genome.

- We may also expect that these activities could be accurately summarized using a fairly small set of labels.

To discover what types of label best explain the data, rather than imposing a pre-determined set of labels on the data, unsupervised learning method can be applied.

UNSUPERVISED MACHINE LEARNING ALGORITHMS[7]

- It will use only the unlabeled data and the desired number of different labels to assign as input to automatically partition the genome into segments and assign a label to each segment, with the goal of assigning the same label to segments that have similar data.

The unsupervised approach requires an additional step in which semantics must be manually assigned to each label, but it provides the benefits of enabling training when labeled examples are unavailable and has the ability to identify potentially novel types of genomic elements.

EXAMPLES OF UNSUPERVISED LEARNING ALGORITHMS

PRINCIPAL COMPONENT ANALYSIS (PCA) (UNSUPERVISED)

PCA provides dimensionality reduction.

Sometimes you have a wide range of features, probably highly correlated between each other, and models can easily overfit on a huge amount of data. Then PCA can be applied.

Advantage:

- in addition to the low-dimensional sample representation, it provides a synchronized low-dimensional representation of the variables. The synchronized sample and variable representations provide a way to visually find variables that are characteristic of a group of samples.

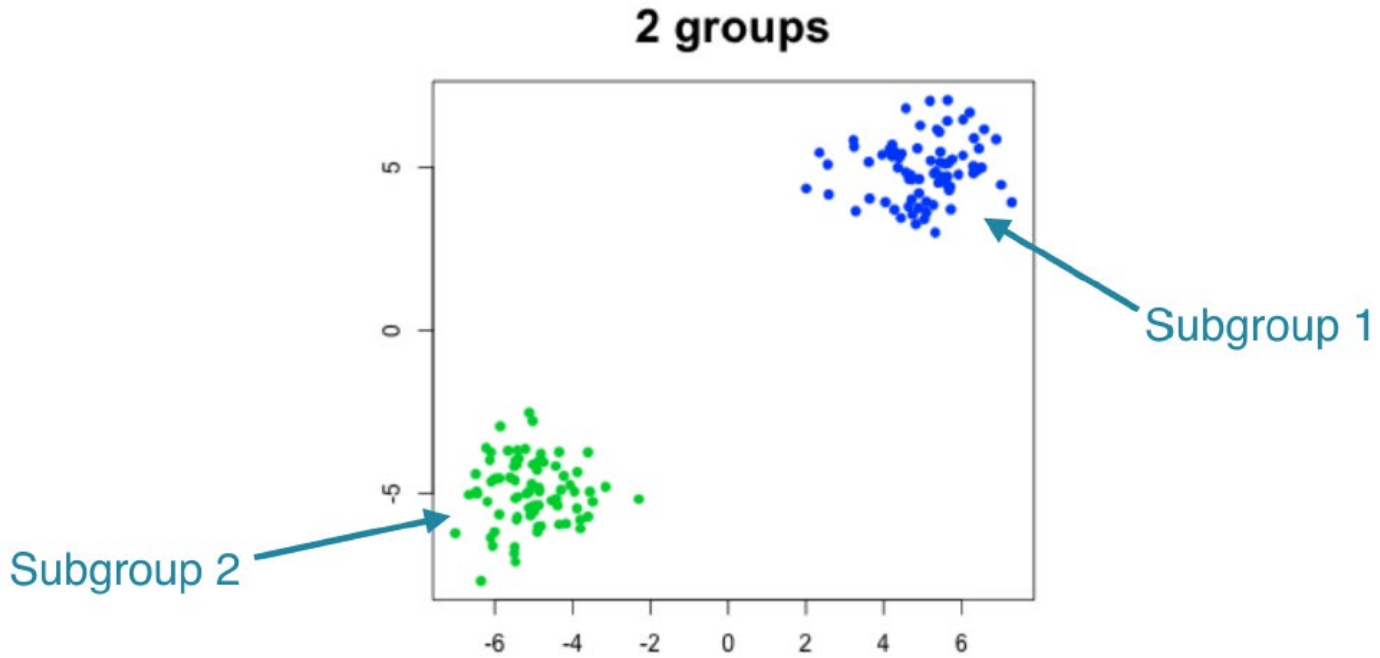
PCA can be used in bioinformatics to:

- Analyse gene expression data

K-MEANS (UNSUPERVISED) [1]

An algorithm used to find homogeneous subgroups in a population

- Breaks observations into a pre-defined number of clusters



K-MEANS (UNSUPERVISED) [2]

The Algorithm

1. Divide the data into K clusters

   Initialize the centroids with the mean of the clusters

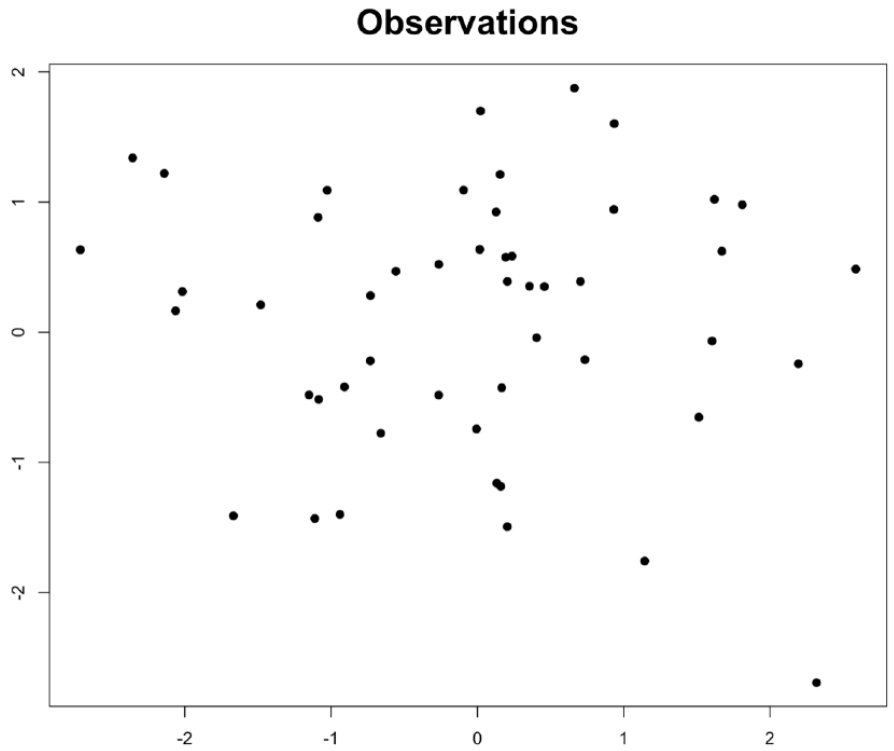
2. Assign each item to the cluster with closest centroid

3. When all objects have been assigned, recalculate the centroids (mean)

4. Repeat 2-3 until the centroids no longer move

K-MEANS (UNSUPERVISED) [3]

The Algorithm in action

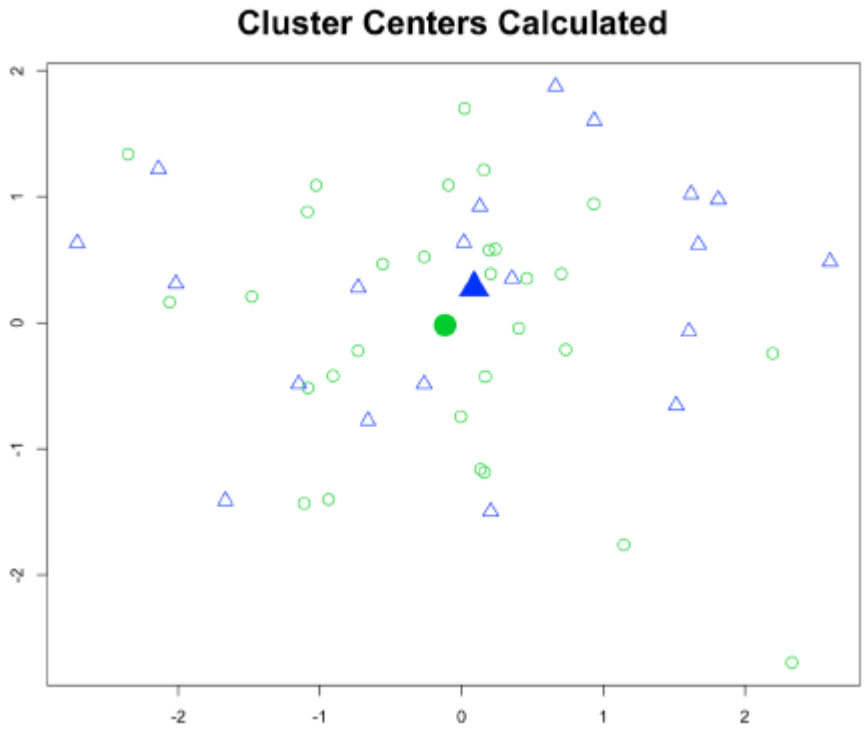


K-MEANS (UNSUPERVISED) [4]

The Algorithm in action

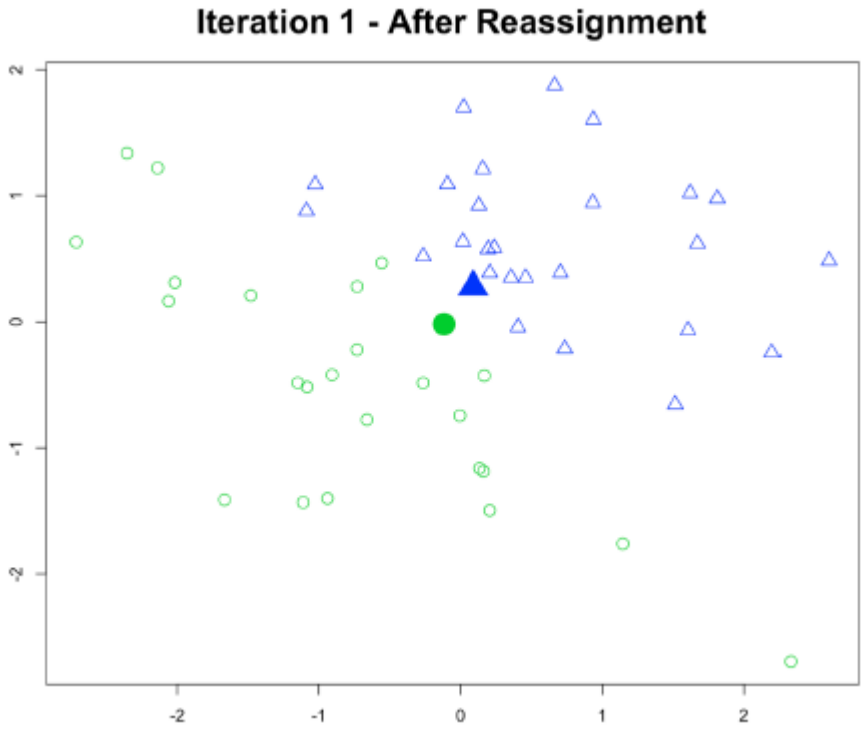
K-MEANS (UNSUPERVISED) [5]

The Algorithm in action



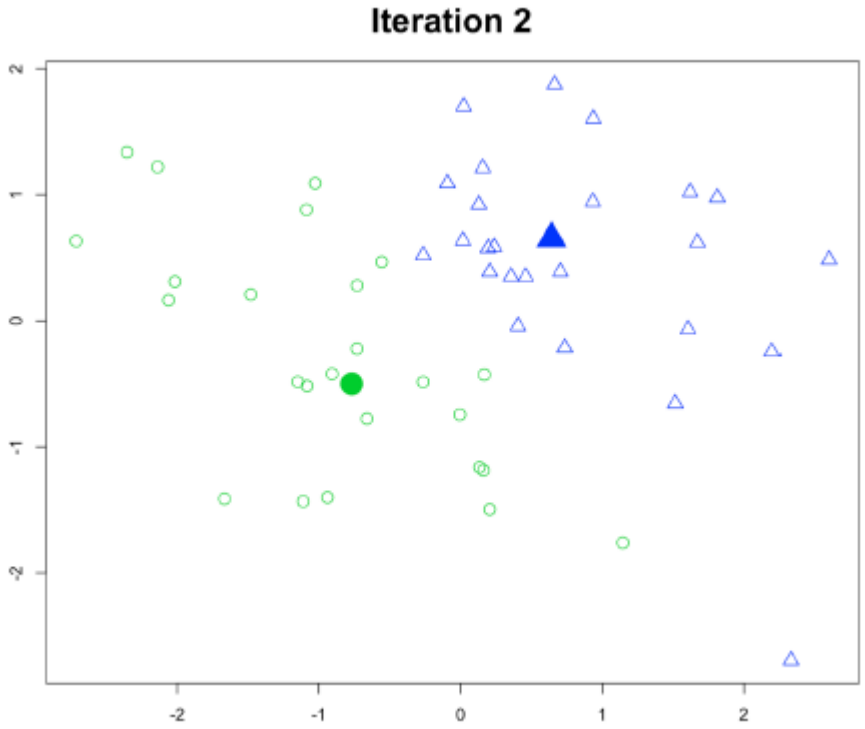
K-MEANS (UNSUPERVISED) [6]

The Algorithm in action



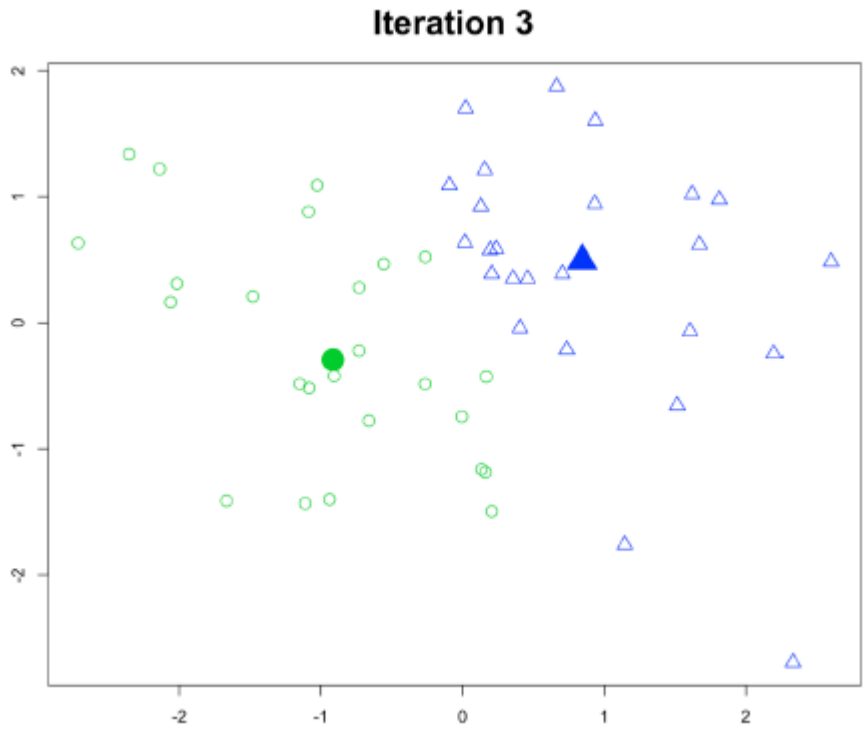
K-MEANS (UNSUPERVISED) [7]

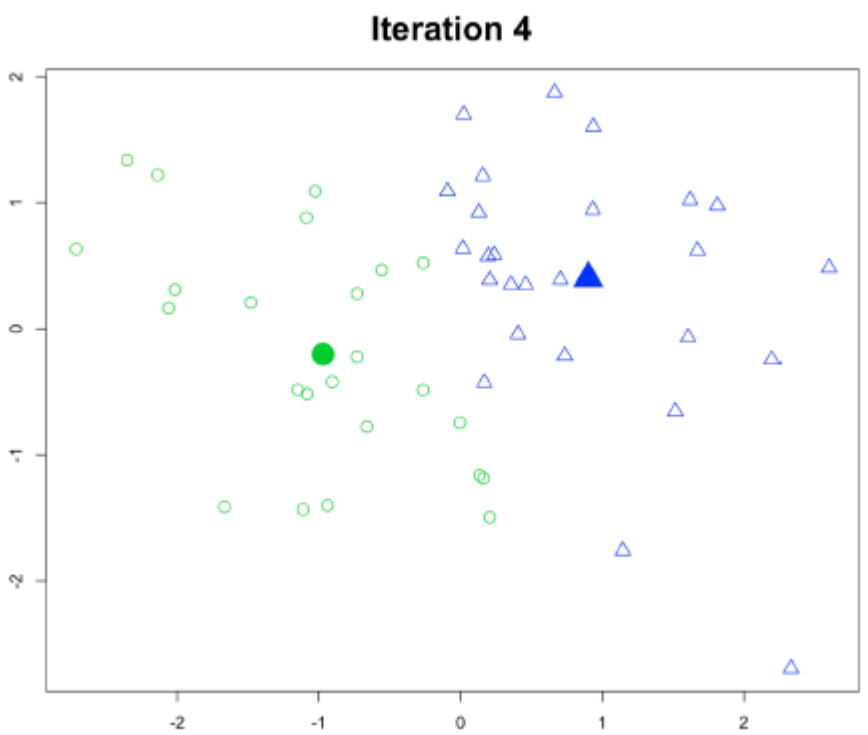
The Algorithm in action



K-MEANS (UNSUPERVISED) [8]

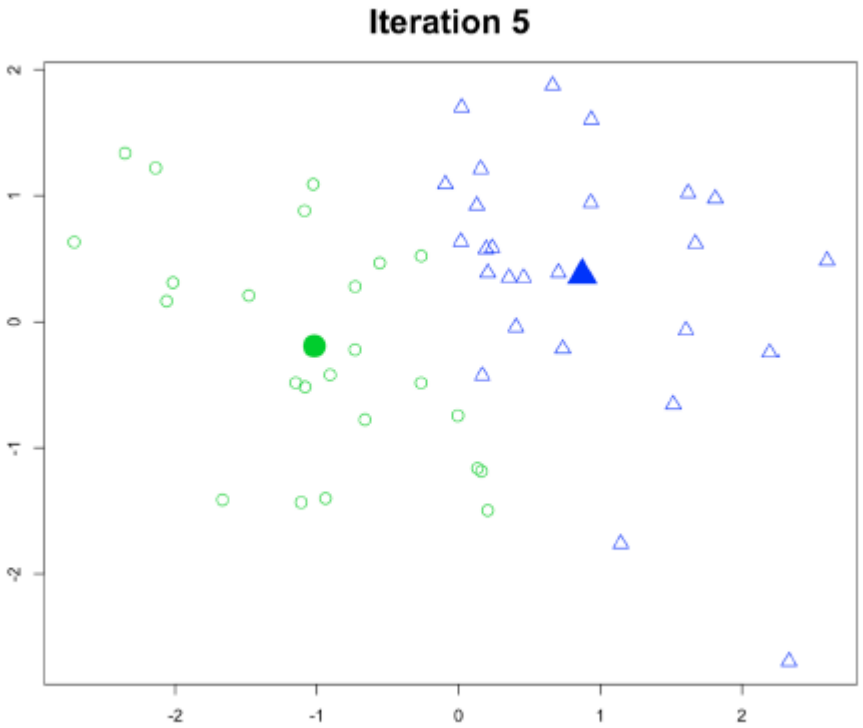
The Algorithm in action





K-MEANS (UNSUPERVISED) [9]

The Algorithm in action



Remember k-means has a random component!

K-MEANS (UNSUPERVISED) [10]

The goal of k-means is to find groups in the data, with the number of groups represented by the variable *K*.

The algorithm works iteratively to assign each data point to one of *K* groups based on the features that are provided. Data points are clustered based on feature similarity.

Advantage: Easy to implement and fast and efficient in terms of computational cost

Disadvantage include:

- Initial seeds have a strong impact on the final results
- The order of the data has an impact on the final results
- K-Means needs to know in advance how many clusters there will be in your data, so this may require a lot of trials to “guess” the best K number of clusters to define.

Example: popular and simple partition computational models for clustering microarray data

HIERARCHICAL CLUSTERING (UNSUPERVISED)

It is yet another clustering algorithm that groups similar objects together into *clusters*.

Number of clusters is not known ahead of time

There are two kinds - Bottom-up (most common) & Top-down

- Algorithm:

  It starts by treating each observation as a separate cluster.

  Then, it repeatedly executes the following two steps:

  (1) identify the two clusters that are closest together, and

  (2) merge the two most similar clusters.

  This iterative process continues until all the clusters are merged together.

- The main output of Hierarchical Clustering is a dendogram, which shows the hierarchical relationship between the clusters

Example: hierarchical Clustering is the most popular method for gene expression data analysis - genes with similar expression patterns are grouped together and are connected by a series of branches.

NEURAL NETWORKS (CAN BE SUPERVISED OR UNSUPERVISED)

Neural Networks take in the weights of connections between neurons. When all weights are trained, the neural network can be utilized to predict the class or a quantity.

With Neural networks, extremely complex models can be trained and they can be utilized as a kind of black box.

Disadvantages:

- parameterization is extremely difficult in neural networks.
- They are also very resource and memory intensive.

NN can be joined with the "deep approach" to build models that can pick previously unpredictable cases.

They may be applied for classification, predictive modelling and biomarker identification within data sets of high complexity such as transcript or gene expression data generated from DNA microarray analysis, or peptide/protein level data generated by mass spectrometry.

Supervised Learning

WHAT IS SUPERVISED MACHINE LEARNING?

Supervised learning is done using a ***ground truth, i.e.*** we have prior knowledge of what the output values for our samples should be.

- It occurs when we have input variables – say x, and an output variable (outcome/target), say Y, and we use an algorithm to learn the mapping function from the input to the output.

  Y = h(X)

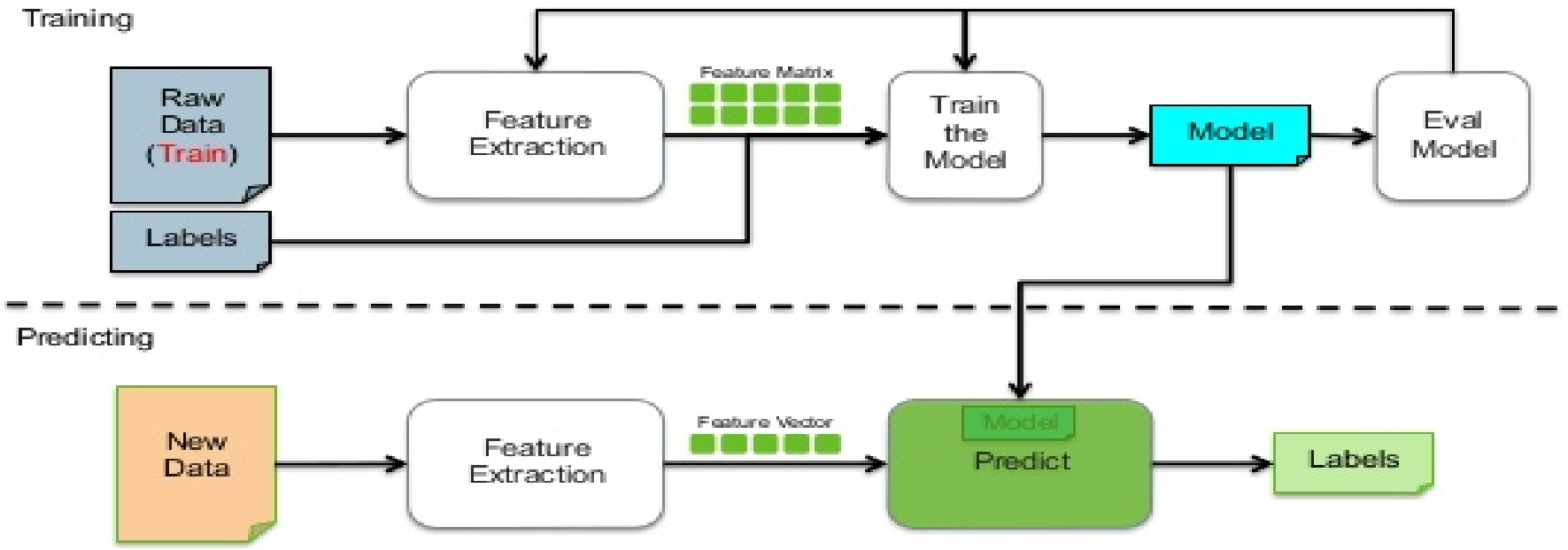
The goal is to approximate the mapping function so well that when you have new input data (x) that you can predict the output variables (Y) for that data.

It is called supervised learning because the process of an algorithm learning from the training dataset can be thought of as a teacher supervising the learning process.

- We know the correct answers, the algorithm iteratively makes predictions on the training data and is corrected by the teacher.
- Learning stops when the algorithm achieves an acceptable level of performance.

SUPERVISED MACHINE LEARNING -WORKFLOW

Supervised Learning Workflow



Page 41 ©Hortonworks Inc. 2011 – 2014. All Rights Reserved

Hortonworks

SUPERVISED LEARNING

Let's assume our simple predictor has this form:

$$h(x) = \vartheta_0 + \vartheta_1 x$$

- where θ_0 and θ_1 are constants.
- Our goal is to find the perfect values of θ_0 and θ_1 to make our predictor work as well as possible.

Optimizing the predictor h(x) is done using training examples.

- For each training example, we have an input value **x_train**, for which a corresponding output, **y**, is known in advance.
- For each example, we find the difference between the known, correct value y, and our predicted value h(x_train).
- With enough training examples, these differences give us a useful way to measure the "wrongness" of h(x).
- We can then tweak h(x) by tweaking the values of θ_0 and θ_1 to make it "less wrong".
- This process is repeated over and over until the system has converged on the best values for θ_0 and θ_1
- In this way, the predictor becomes trained, and is ready to do some real-world predicting.

SUPERVISED LEARNING ALGORITHMS

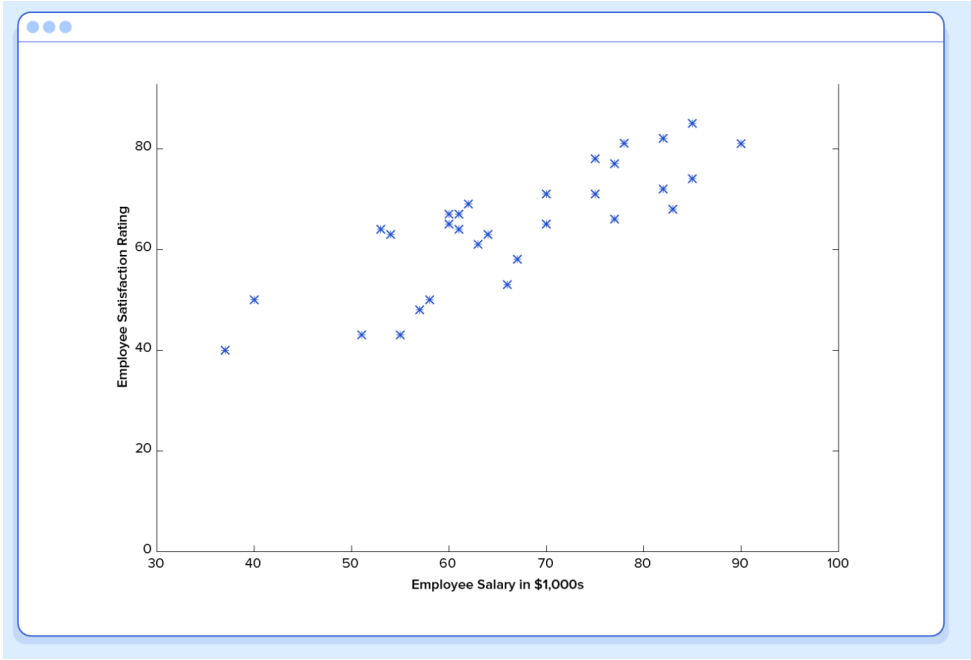
Apply what has been learned in the past to new data using labeled examples to predict future events.

Starting from the analysis of a known training dataset, the learning algorithm produces a prediction model that can provide targets for any new input (after sufficient training).

The learning algorithm can also compares its output with the correct, intended output and finds errors in order to modify and improve the prediction model accordingly.

SUPERVISED LEARNING EXAMPLE

Consider the following training data, wherein company employees have rated their satisfaction on a scale of 1 to 100:



First, notice that the data is a little noisy.

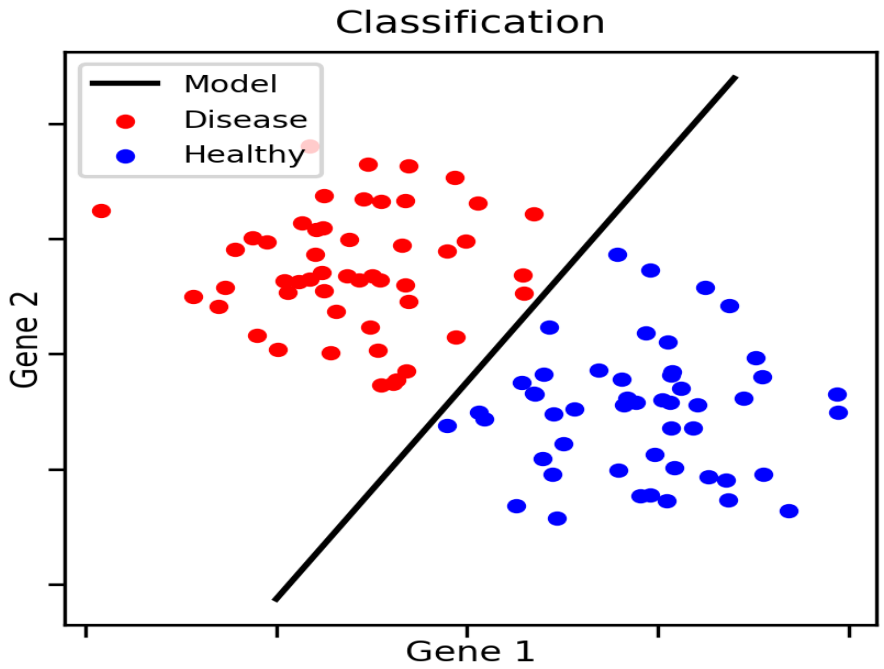
- That is, while we can see that there is a pattern to it (i.e. employee satisfaction tends to go up as salary goes up), it does not all fit neatly on a straight line.
- This will always be the case with real-world data (and we absolutely want to train our machine using real-world data!).
- So then how can we train a machine to perfectly predict an employee's level of satisfaction?
  - The answer, of course, is that **we can't**.
- The goal of ML is never to make "perfect" guesses, because ML deals in domains where there is no such thing.
  - **The goal is to make guesses that are good enough to be useful.**

SUPERVISED MACHINE LEARNING – TWO TYPES

Under supervised ML, two major subcategories are:

- **Regression machine learning systems**: Systems where the value being predicted falls somewhere on a continuous spectrum.
  - These systems help us with questions of "How much?" or "How many?".
- **Classification machine learning systems**: Systems where we seek a yes-or-no prediction, such as "Is this tumor cancerous?", "Does this cookie meet our quality standards?", and so on.
- The underlying Machine Learning theory is more or less the same for both.

CLASSIFICATION VS REGRESSION



https://aldro61.github.io/microbiome-summer-school-2017/sections/basics/

CLASSIFICATION VS REGRESSION

Classification	Regression
Discrete, categorical variable	Continous (real number range)
Supervised classification problem	Supervised regression problem
Assign the output to a class (a label)	Predict the output value using training data
Predict the type of tumor (harmful vs not harmful)	Predict gene expression patterns, predict survival time

EXAMPLES OF SUPERVISED LEARNING ALGORITHMS

LINEAR REGRESSION (SUPERVISED)

Regression algorithms can be used for example when some continuous value needs to be computed as compared to classification where the output is categorical.

So whenever there is a need to predict some future value of a process which is currently running, regression algorithm can be used.

Operating on a two dimensional set of observations (two continuous variables), simple linear regression attempts to fit, as best as possible, a line through the data points.

The regression line (our model) becomes a tool that can help uncover underlying trends in our dataset.

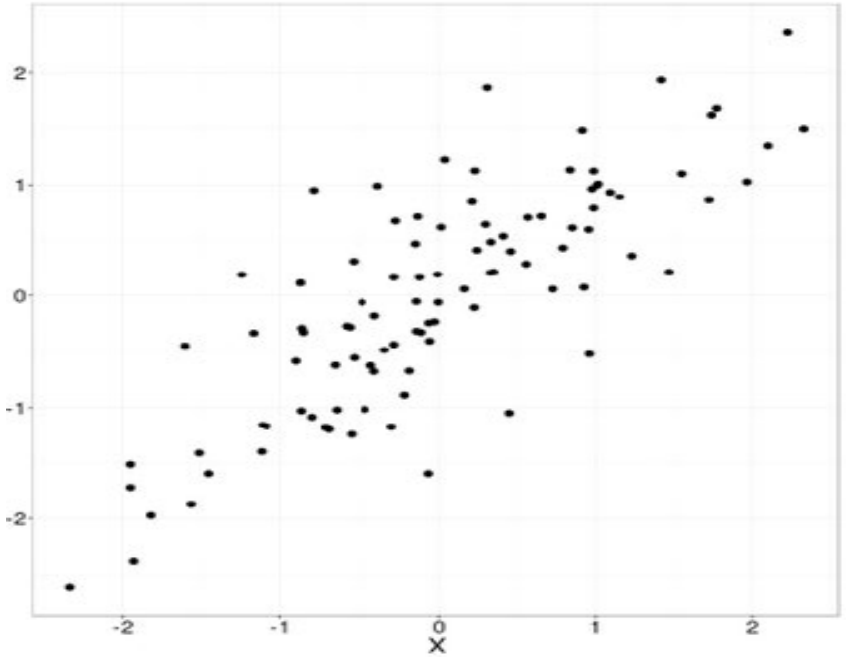
The regression line, when properly fitted, can serve as a predictive model for new events.

Linear Regressions are however unstable in case features are redundant, i.e. if there is multicollinearity

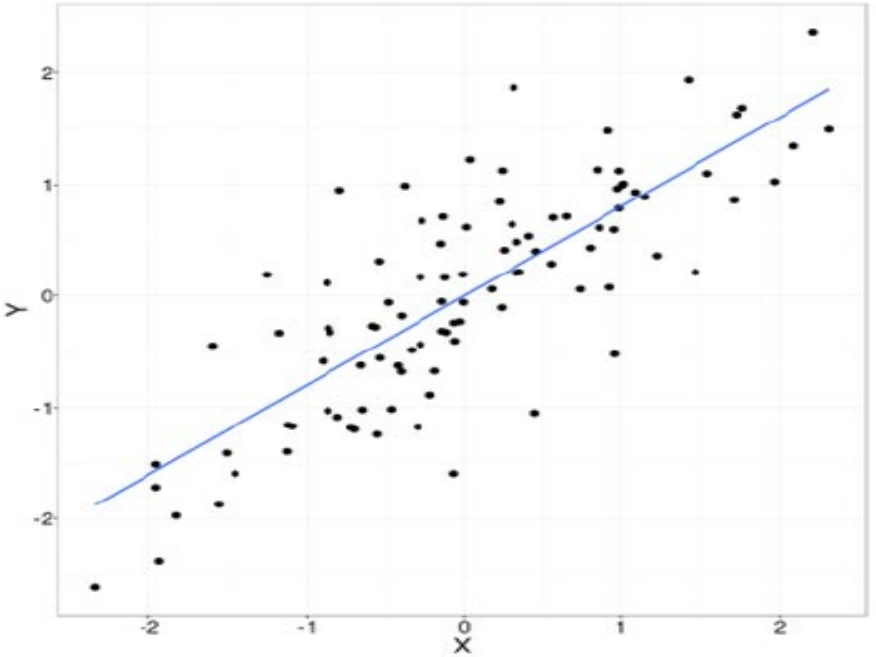
Example where linear regression can be used is:

- predicting drug resistance by correlating genotypic information with phenotypic profiles

APPLYING LINEAR REGRESSION



Scatterplot of our dataset.



Fitting of the regression line (blue).

LINEAR REGRESSION – PROS AND CONS

Pros

- Easy to fit and apply
- Concise
- Less prone to over-fitting
- Interpretable

```
Call:
lm(formula = blood_pressure ~ age + weight, data = bloodpressure)

Coefficients:
(Intercept)          age       weight
    30.9941       0.8614       0.3349
```

Cons

- Can only express linear and additive relationships

DECISION TREES (SUPERVISED)

A decision tree is a tree-like graph with nodes representing the place where we pick an attribute and ask a question; edges represent the answers to the question; and the leaves represent the actual output or class label.

They are used in non-linear decision making with simple linear decision surface.

It is one of the most widely used and practical methods for supervised learning.

Single trees are used very rarely, but in composition with many others they build very efficient algorithms such as Random Forest or Gradient Tree Boosting.

Decision trees easily handle feature interactions and they are non-parametric, so there is no need to worry about outliers or whether the data is linearly separable.

They are used for both classification and regression tasks.

DECISION TREES (SUPERVISED)

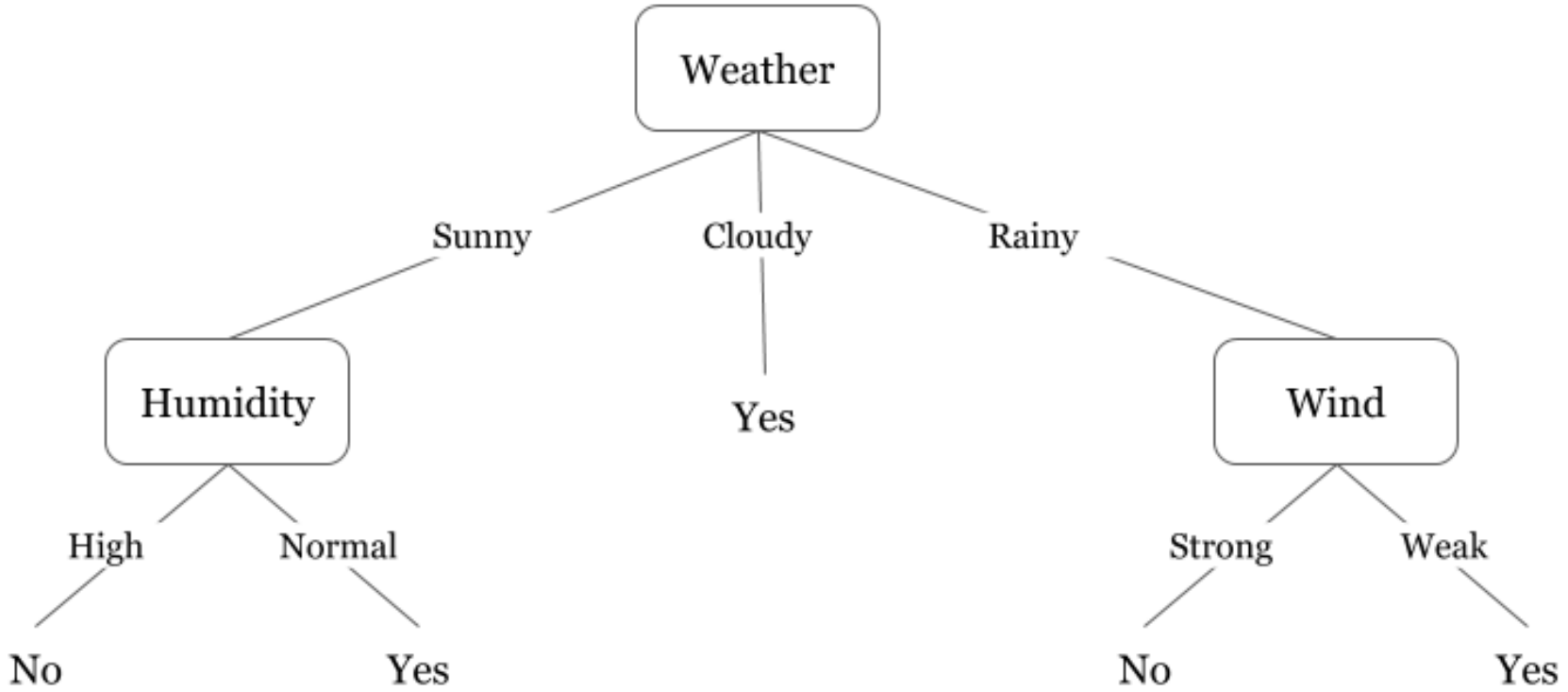
Disadvantages are:

- Often the tree needs to be rebuilt when new examples come on.
- Decision trees easily overfit, but ensemble methods like random forests (or boosted trees) take care of this problem.
- They can also take a lot of memory (the more features you have, the deeper and larger your decision tree is likely to be)

Trees are excellent tools for helping to choose between several courses of action.

Example: Classification of genomic islands using decision trees and ensemble algorithms

DECISION TREES (SUPERVISED)

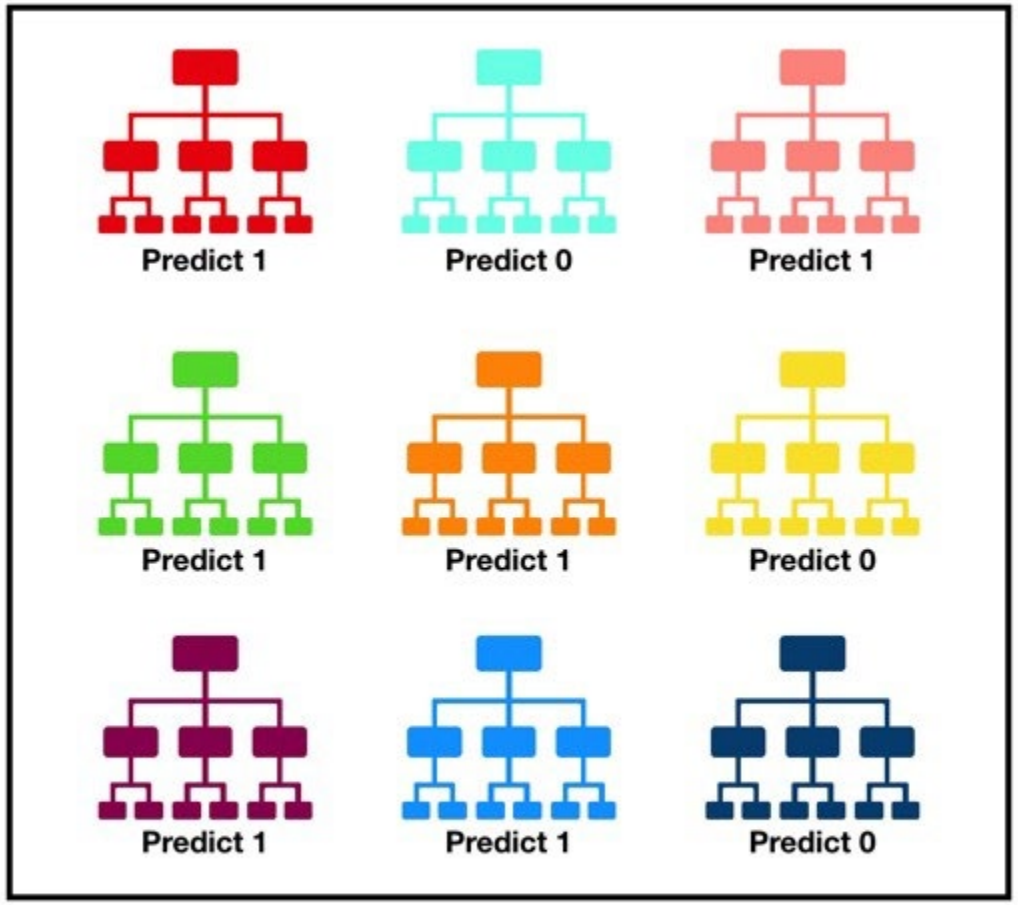


RANDOM FOREST (SUPERVISED)

Random forest, like its name implies, consists of a large number of individual decision trees that operate as an ensemble.

Each individual tree in the random forest spits out a class prediction and the class with the most votes becomes our model's prediction.

Put simply: random forest builds multiple decision trees and merges them together to get a more accurate and stable prediction.



RANDOM FOREST (SUPERVISED)

It can solve **both regression and classification** problems with large data sets.

It also helps identify most significant variables from thousands of input variables.

Random Forest is highly scalable to any number of dimensions and has generally quite acceptable performances.

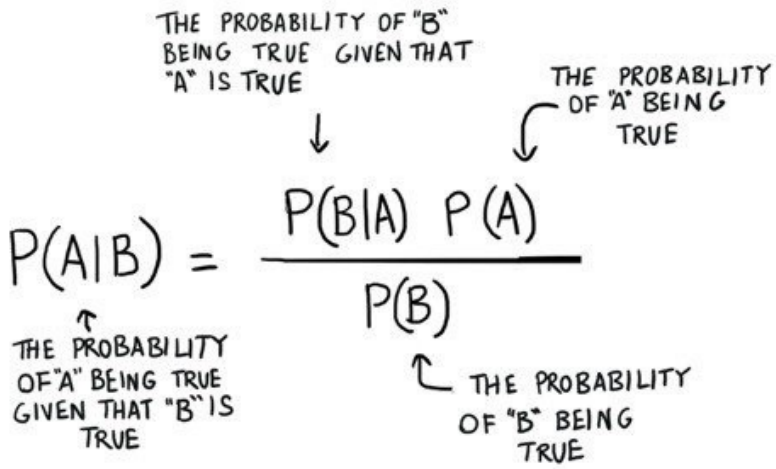
However with Random Forest, ***learning may be slow*** (depending on the parameterization) and it is not possible to iteratively improve the generated models

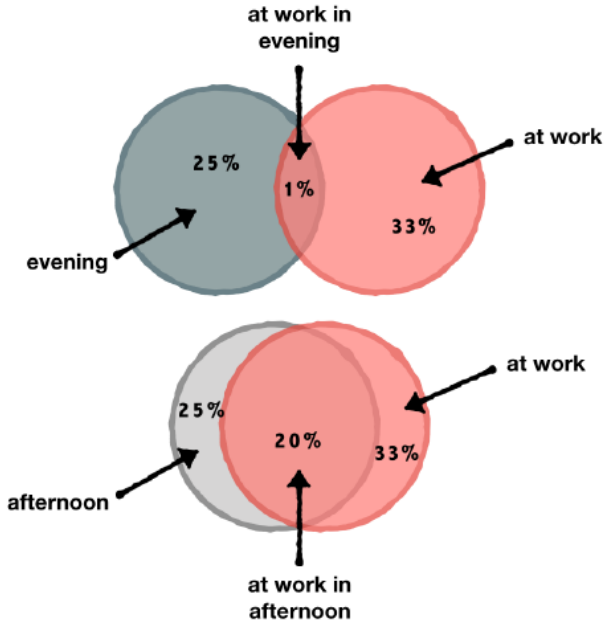
Random Forest can be used in real-world applications such as:

- Predict patients for high risks for certain diseases

NAIVE BAYES (SUPERVISED)

It is a classification technique based on Bayes' theorem (conditional probability and dependent events).





The **conditional probability** of events A and B is denoted P(A | B)

- P(A | B) = P(A and B) / P(B)
- P(work | evening) = 1 / 25 = 4%
- P(work | afternoon) = 20 / 25 = 80%

With some modifications it **can** be **used** for **regression** as well

NAIVE BAYES (SUPERVISED)

Advantages include:

- very easy to build and particularly useful for very large data sets.
- outperform even highly sophisticated classification methods.
- a good choice when CPU and memory resources are a limiting factor.
- A good method if something fast and easy that performs pretty well is needed.

Its main disadvantage is that it does not consider the interactions between features.

Naive Bayes has been used in real-world applications such as:

- Mining housekeeping genes
- genetic association studies
- discovering Alzheimer genetic biomarkers from whole genome sequencing (WGS) data

SUPPORT VECTOR MACHINES (SUPERVISED)

Support Vector Machine (SVM) is a supervised machine learning technique that is widely used in pattern recognition and classification problems—when your data has exactly two classes.

In the SVM algorithm, we plot each data item as a point in n-dimensional space (where n is number of features we have) with the value of each feature being the value of a particular coordinate.

Then, we perform classification by finding the hyper-plane that differentiates the two classes very well.

- It constructs a hyperplane in multidimensional space to separate different classes.
- SVM generates optimal hyperplane in an iterative manner, which is used to minimize an error.
- The core idea of SVM is to find a maximum marginal hyperplane(MMH) that best divides the dataset into classes.

SUPPORT VECTOR MACHINES (SUPERVISED)

Advantages include high accuracy and even if the data is not linearly separable in the base feature space, SVM can work well with an appropriate kernel.

It can be used for both classification and regression, but mostly used in classification problems

However SVMs are memory-intensive, hard to interpret, and difficult to tune.

SVM is especially popular in text classification problems where very high-dimensional spaces are the norm.

SVM can be used in real-world bioinformatics applications such as:

- Detecting persons with common diseases such as diabetes
- Classification of genomic islands
- Classification of genes

LOGISTIC REGRESSION (SUPERVISED)

It is a regression model that predicts probabilities

- Predicting *whether* an event occurs (yes/no): **classification**
- Predicting *the probability* that an event occurs: **regression**
- Linear regression: predicts values in $[-\infty, \infty]$
- Probabilities: limited to [0,1] interval
  - So we'll call it non-linear

Note: Classification refers to predicting whether an event will occur (Yes/No). While **regression** refers to the probability that an event will occur.

GENERALIZED LINEAR MODELS (GLM)

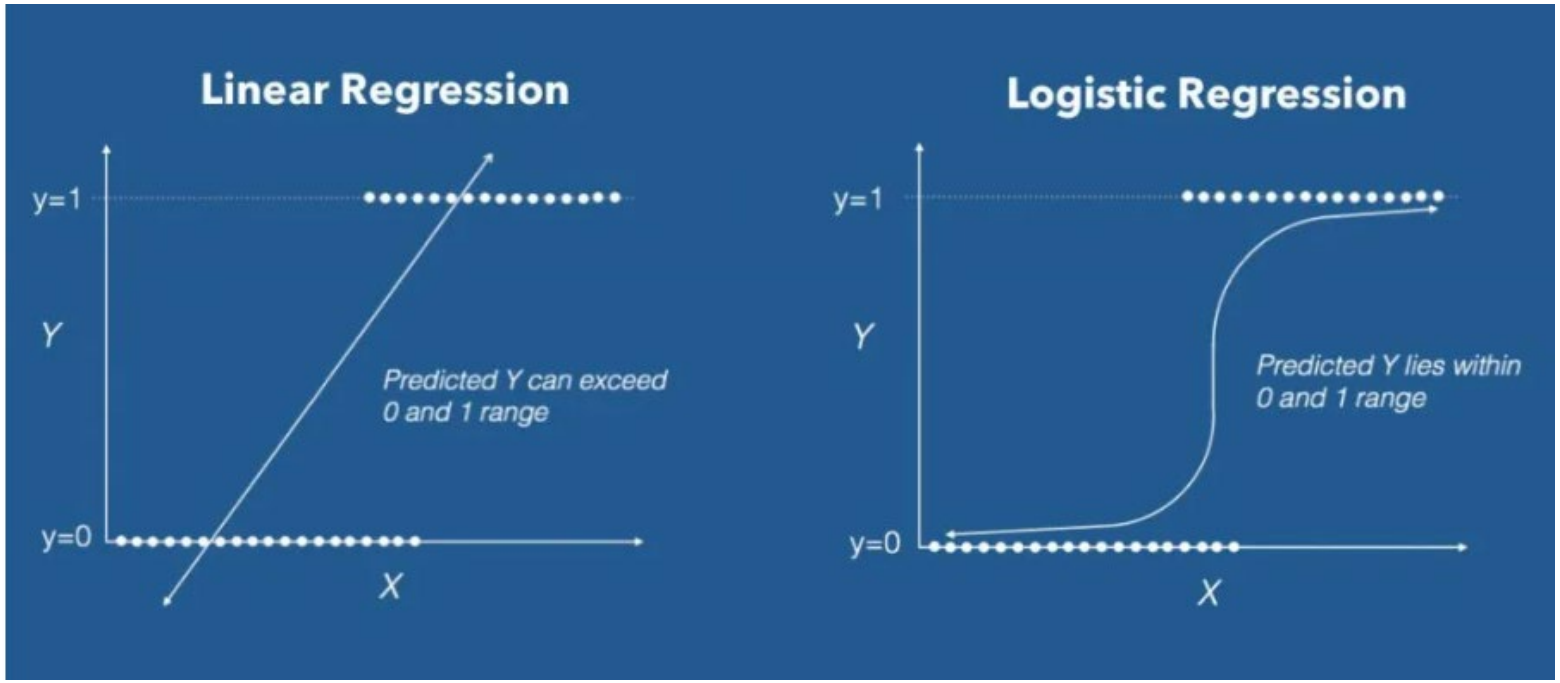
- The term ***generalized linear model*** (GLIM or GLM) refers to a larger class of models
- In these models, the response variable y_i is assumed to follow an exponential family distribution with mean μ_i, which is assumed to be some (often nonlinear) function.
- GLMs are a broad class of models that include linear regression, ANOVA, Poisson regression, log-linear models etc.
- Some of the models are:

Model	Probability Distribution
Linear Regression	Normal
Logistic Regression	Binomial
Poisson Regression	Poisson

EXAMPLE OF LOGISTIC REGRESSION – PREDICTING DUCHENNE MUSCULAR DYSTROPHY (DMD)

We want to develop a test to detect the gene for DMD in women.

- The test uses the measurements of 2 enzymes in the blood (CK and H).
- What is the probability that a woman is a DMD carrier based on her CK and H levels?
- We cannot use linear regression (where the outcome is 0:False and 1:True), because the linear model will predict probabilities outside the range of 0 and 1.



SUPERVISED LEARNING - DATASETS

Training dataset

- The subset of the dataset provided to the algorithm for learning is called the training set.

Validation dataset

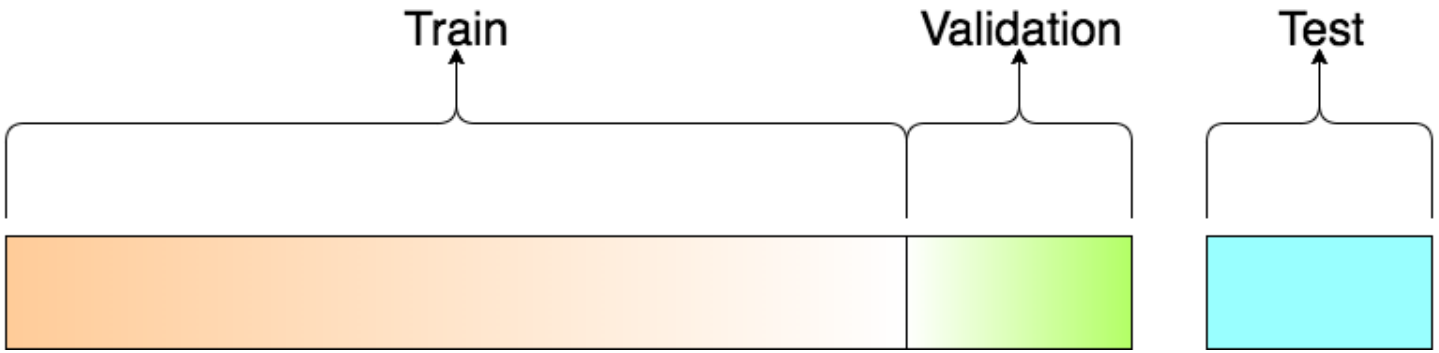
A set of examples used to tune the parameters of a classifier/regressor, for example to choose the number of hidden units in a neural network.

Test dataset

A set of examples used only to assess the performance of a fully-specified classifier/regressor

SUPERVISED LEARNING DATASETS

Train -test-validation split ratio



- This mainly depends on 2 things.
  - The total number of samples in your data
  - The actual model we are training

VALIDATION OF SUPERVISED ML ALGORITHMS RESULTS

To test the performance of the learning system:

- The system can be tested with sequences where the labels are known (and were excluded from the training set because they were intended to be used for this purpose).
- Based on the results of the test data, the performance of the learning system can be assessed.

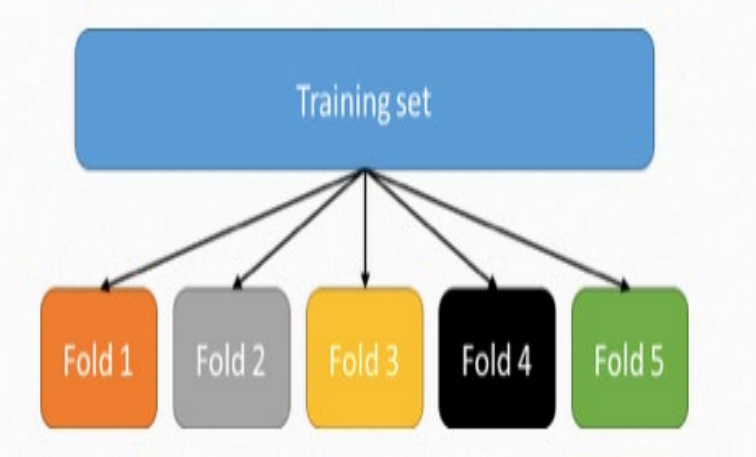
TRAINING SET AND TEST SET

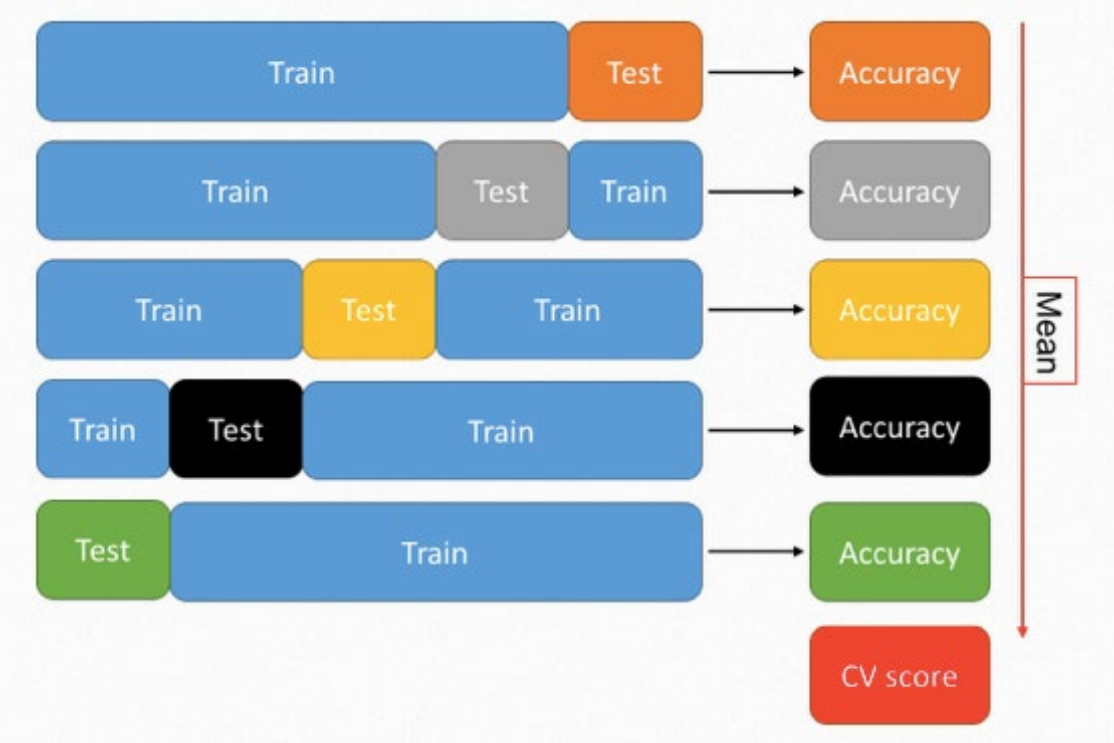
Used to train the algorithm

Estimate the accuracy of the model

Split the dataset randomly!
Use cross-validation
Underfitting and over fitting problems

K-FOLD CROSS VALIDATION



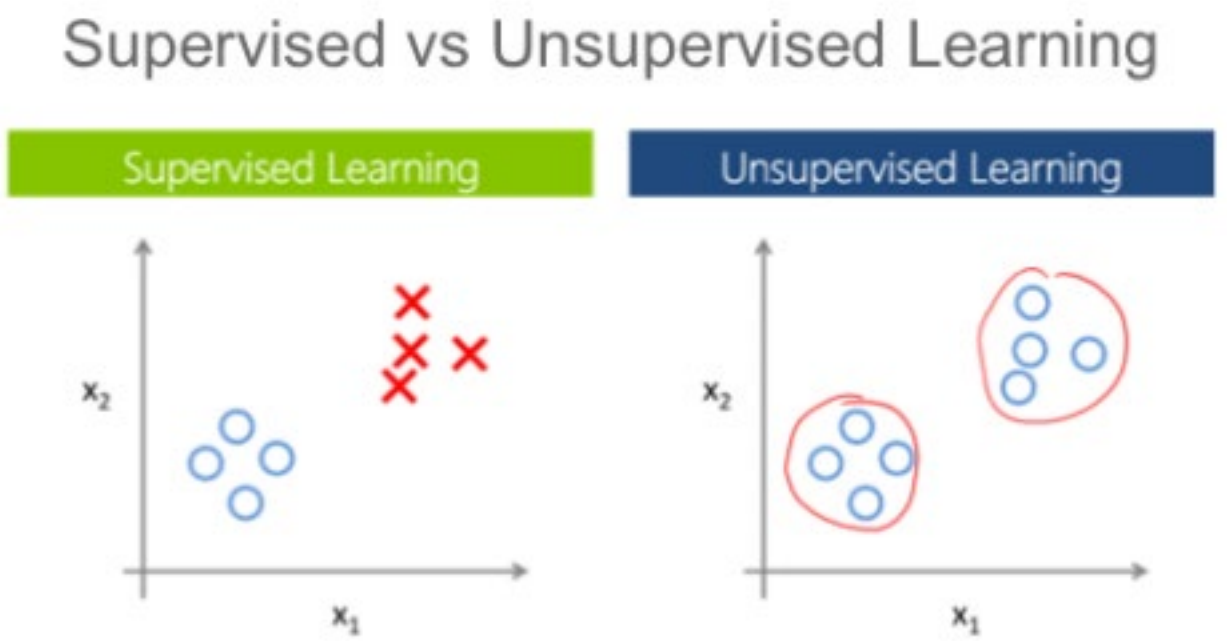


https://aldro61.github.io/microbiome-summer-school-2017/sections/basics/#type-of-learning-problems

K-FOLD CROSS VALIDATION

- In informed cross validation we partition the data into subsets (suppose we use k=3).
  - Let's call the subsets A, B & C
- First train the model using the data from sets A & B, and use that model to make prediction on C.
- Then train the model using the data from sets B & C, and use that model to make prediction on A.
- And train the model using the using the data from sets A & C, and use that model to make prediction on B.

SUPERVISED VS UNSUPERVISED LEARNING



https://www.cisco.com/c/m/en_us/network-intelligence/service-provider/digital-transformation/get-to-know-machine-learning.html

SUPERVISED VS UNSUPERVISED

Supervised	Unsupervised
Input data is labelled	Input data is unlabelled
Uses training dataset	Uses just input dataset
Known number of classes	Unknown number of classes
Guided by expert (labelled data provided)	Self guided learning (using some criteria)
Goal: predict class or value label	Goal: analyse data, determine data structure/grouping
Classification and regression	Clustering, dimensionality reduction, density estimation

Classification metrics

WHY THE NEED TO EVALUATE?

- Multiple methods are available to classify or predict
- For each method, multiple choices are available for settings
- To choose best model, need to assess each model's performance

MISCLASSIFICATION ERROR

- **Error** = classifying a record as belonging to one class when it belongs to another class.

- **Error rate** = percent of misclassified records out of the total records in the validation data

NAÏVE RULE

Naïve rule: classify all records as belonging to the most prevalent class

- Often used as benchmark: we hope to do better than that
- Exception: when goal is to identify high-value but rare outcomes, we may do well by doing worse than the naïve rule

SEPARATION OF RECORDS

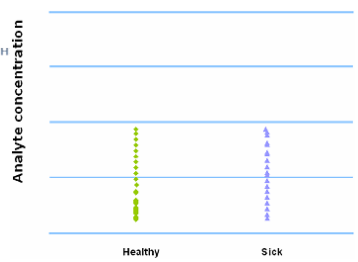
"High separation of records" means that using predictor variables attains low error

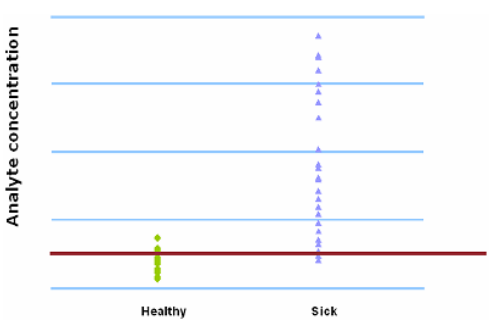
"Low separation of records" means that using predictor variables does not improve much on naïve rule

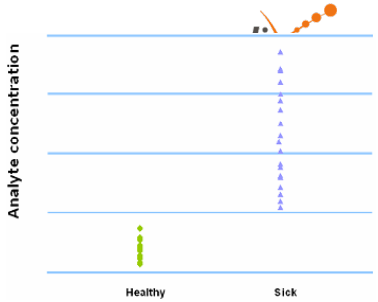
DIFFERENT SCORING METRICS

1. Confusion Matrix
   - True positives
   - False negatives
   - False positives
   - True negatives
2. Sensitivity and Specificty
3. Precision and Recall
4. F-measure
5. Overall accuracy and Cohen's kappa

MAIN DEFINITIONS



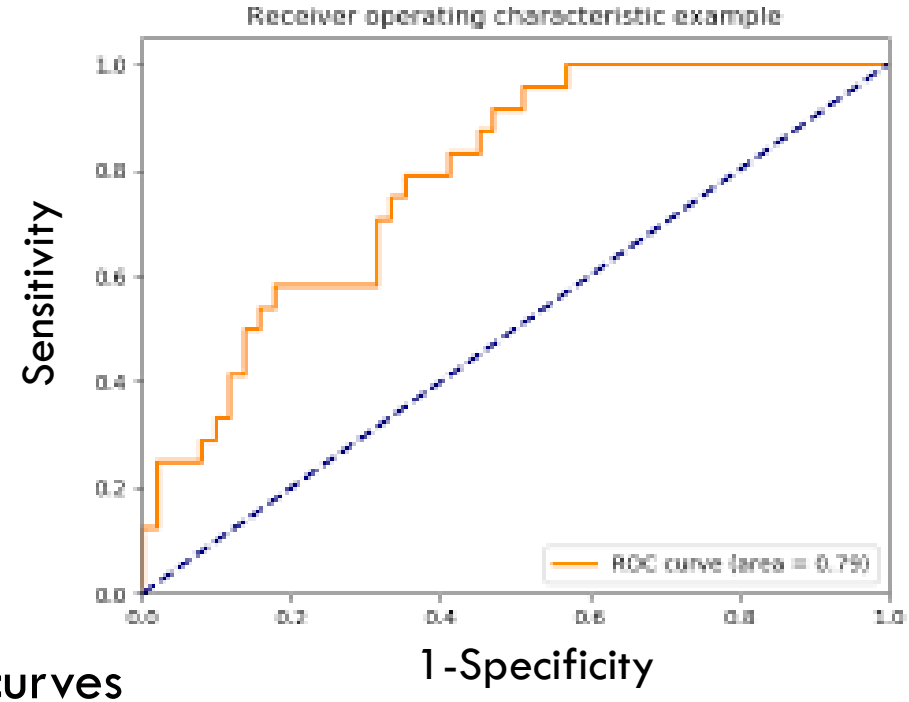




- Confusion matrix

n=165	Predicted: NO	Predicted: YES
Actual: NO	50	10
Actual: YES	5	100

- Precision $\frac{tp}{tp + fp}$
- Specificity $\frac{TN}{FP+TN}$
- Recall / Sensitivity $\frac{tp}{tp + fn}$
- Receiver Operating Characteristic (ROC) and AUC curves



https://scikit-learn.org/stable/auto_examples/model_selection/plot_roc.html

F-MEASURE

$$\text{F-measure} = 2 * \frac{precision * recall}{precision + recall}$$

Harmonic mean of precision and recall

Are ALL and ONLY positive class events found by the model?

OVERALL ACCURACY

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+FN+TN}$$

! Target class distribution must be balanced!

Probability of classifying a positive OR negative class event correctly.

WHY DIFFERENT METRICS?

1. What is your objectives?

2. What is the target class distribution?

3. Is the target binomial or multinomial?

Supervised Learning - Regression

WHAT IS REGRESSION?

Regression: Predict a numerical outcome ("dependent variable") from a set of inputs ("independent variables").

Statistical Sense: Predicting the expected value of the outcome.

Casual Sense: Predicting a numerical outcome, rather than a discrete one.

WHAT IS REGRESSION?

How many patients will come to the emergency unit on a Sunday evening? (**Regression**)

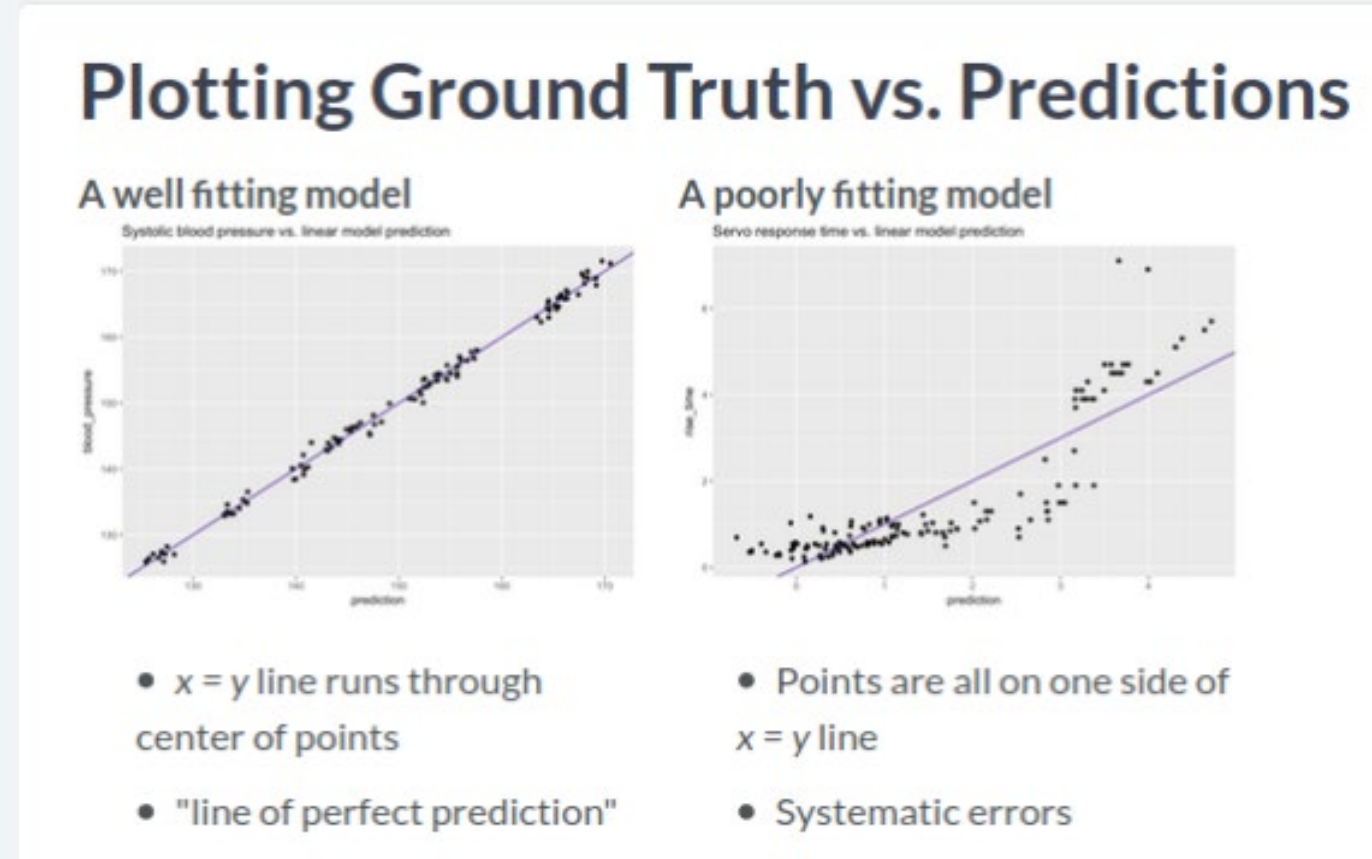
Is this histopathological image classified as "cancer" or "non-cancer" type? (**Classification**)

How many days will this patient spend in the hospital? (**Regression**)

Evaluating a Regression Model

EVALUATING OUR REGRESSION MODEL GRAPHICALLY

First of all we can visualize our ground truths vs the predicted values to see how well our model has performed the predictions.

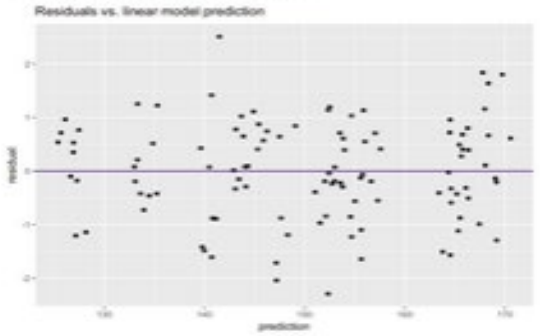


EVALUATING OUR REGRESSION MODEL GRAPHICALLY

Secondly we can also visualize the residuals against the predictions

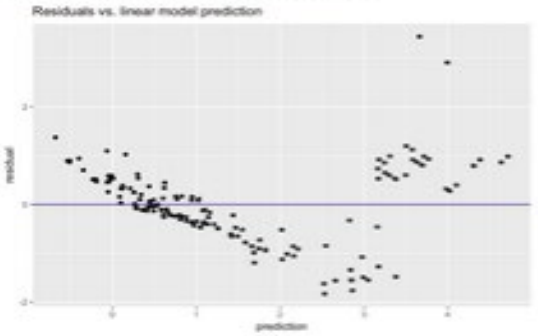
The Residual Plot

A well fitting model



- Residual: actual outcome - prediction
- Good fit: no systematic errors

A poorly fitting model



- Systematic errors

EVALUATION OF OUR REGRESSION MODEL – USING RMSE (ROOT MEAN SQUARE ERROR)

$$RMSE = \sqrt{\overline{(pred - y)^2}}$$

where

- $pred - y$: the error, or residuals vector
- $\overline{(pred - y)^2}$: mean value of $(pred - y)^2$

EVALUATION OF OUR REGRESSION MODEL – USING R^2

Coefficient of Determination or R^2 is another metric used for evaluating the performance of a regression model.

It helps us to compare our current model with a constant baseline and tells us how much our model is better.

The constant baseline is chosen by taking the mean of the data and drawing a line at the mean.

R^2 is a scale-free score that implies it doesn't matter whether the values are too large or too small, the R^2 will always be less than or equal to 1.

The closer the value of R^2 to 1, the better is our model

EVALUATION OF OUR REGRESSION MODEL – USING R^2

Calculating R^2

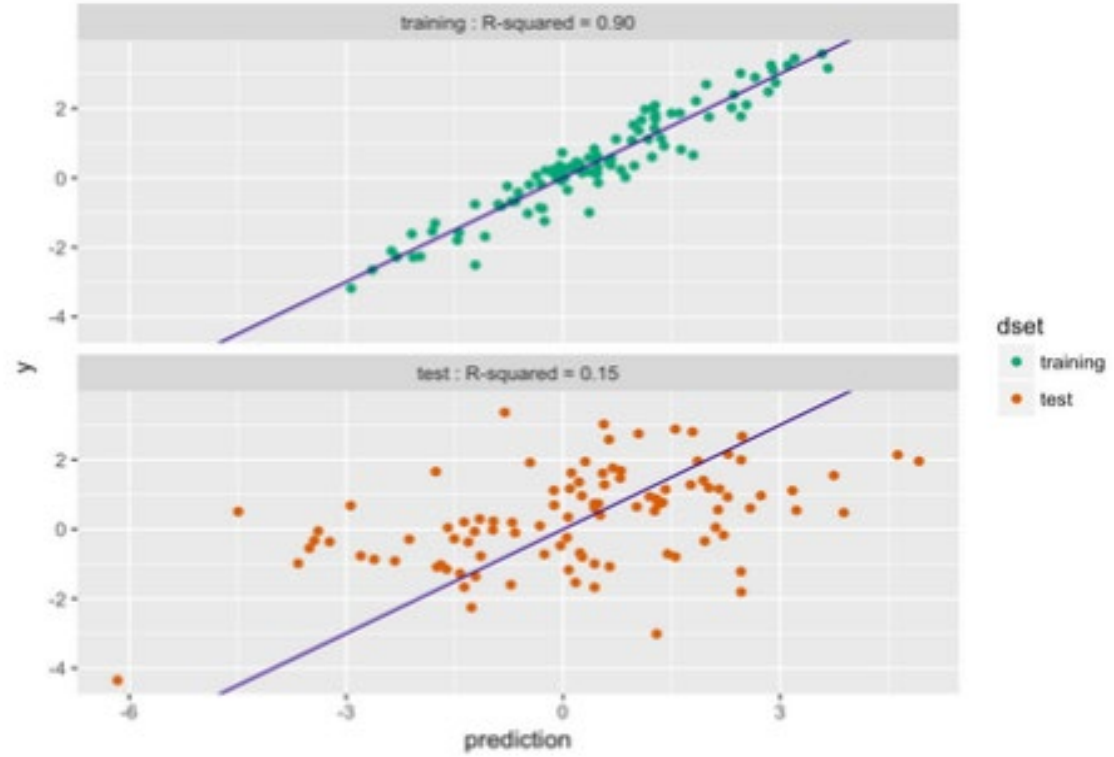
R^2 is the *variance explained by the model.*

$$R^2 = 1 - \frac{RSS}{SS_{Tot}}$$

where

- $RSS = \sum (y - prediction)^2$
  - Residual sum of squares (variance from model)
- $SS_{Tot} = \sum (y - \overline{y})^2$
  - Total sum of squares (variance of data)

REGRESSION – **PROPERLY TRAINING A MODEL**



- Training R^2: 0.9; Test R^2: 0.15 -- **Overfit**

REGRESSION – PROPERLY TRAINING A MODEL

In general models can perform much better on training than on data they have not yet seen.

For simple models, this difference between training data and test data results is often not severe.

But for more complex models or even for linear model with too many variables, using only the training data to evaluate the model can produce misleading results.

In the previous slide example we get the value of R^2 as 0.9 on training data but 0.15 on new data.

- It means this model was overfit.

When we have a lot of data, the best thing to do is to split your data into 2, one set to train the model and another set to test it.

When we don't have enough data we must do cross-validation

Going into the “grey” area

SEMI-SUPERVISED MACHINE LEARNING ALGORITHMS

In supervised learning, the algorithm receives as input a collection of data points, each with an associated label, whereas in unsupervised learning the algorithm receives the data but no labels.

- The semi-supervised setting is a mixture of these two approaches: the algorithm receives a collection of data points, but only a subset of these data points have associated labels.

So, they fall somewhere in between supervised and unsupervised learning, since they use both labeled and unlabeled data for training – **typically a small amount of labeled data and a large amount of unlabeled data**.

The systems that use this method are able to considerably improve learning accuracy.

SEMI-SUPERVISED MACHINE LEARNING ALGORITHMS

Consider the gene finding model where the system is provided with labeled data and unlabeled data.

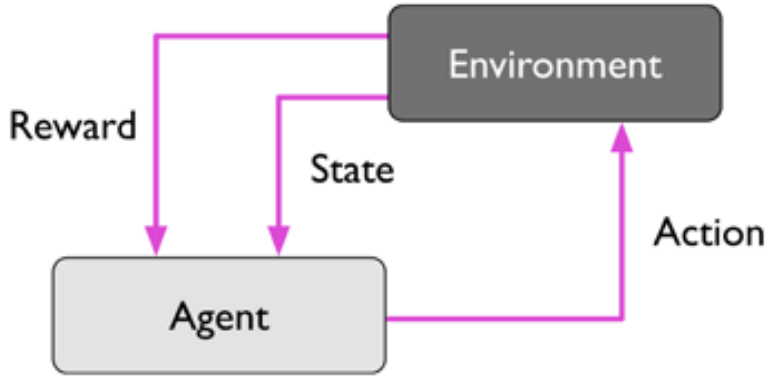
- The learning procedure begins by constructing an initial gene-finding model on the basis of the labeled subset of the training data alone.
- Next, the model is used to scan the genome, and tentative labels are assigned throughout the genome.
- These tentative labels can then be used to improve the learned model, and the procedure iterates until no new genes are found.

SEMI-SUPERVISED MACHINE LEARNING ALGORITHMS

In practice, gene-finding systems are often trained using a semi-supervised approach, in which the input is a collection of annotated genes and an unlabeled whole-genome sequence.

The semi-supervised approach can work much better than a fully supervised approach because the model is able to learn from a much larger set of genes — all of the genes in the genome — rather than only the subset of genes that have been identified with high confidence.

REINFORCEMENT MACHINE LEARNING ALGORITHMS



The learning system interacts with the environment by producing actions and discovers errors or rewards.

- The goal is to develop a system (agent) that improves its performance based on interactions with its environment.

Through its interaction with the environment, an agent can then use reinforcement learning to learn a series of actions that maximizes this reward via an exploratory trial-and-error approach or deliberative planning.

REINFORCEMENT MACHINE LEARNING ALGORITHMS

The idea behind ***Reinforcement Learning*** is that an agent will learn from the environment by interacting with it and receiving rewards for performing actions.

Learning from interaction with the environment comes from our natural experiences.

- Consider a child in a living room who sees a fireplace and approaches it.
- It's warm, it's positive, the child feels good (*Positive Reward* +1) and understands that fire is a positive thing.
- Next he tries to touch the fire and it burns his hand (Negative reward -1). He then understands that fire is positive when he is a sufficient distance away, because it produces warmth. But getting too close to it, he will be burned.

DEEP LEARNING ALGORITHMS

Also known as deep structured learning or hierarchical learning

It is a subfield of machine learning concerned with algorithms inspired by the structure and function of the brain called artificial neural networks.

Can perform learning in supervised and/or unsupervised manners.

Teach computers to do what comes naturally to humans: **learn by example**

- key technology behind driverless cars, enabling them to recognize a stop sign, or to distinguish a pedestrian from a lamppost.
- **In medical Research**
  - Cancer researchers are using deep learning to automatically detect cancer cells.
  - Teams at UCLA built an advanced microscope that yields a high-dimensional data set used to train a deep learning application to accurately identify cancer cells.

DEEP LEARNING ALGORITHMS

While deep learning was first theorized in the 1980s, there are two main reasons it has only recently become useful:

- Deep learning requires large amounts of labeled data.
  - For example, driverless car development requires millions of images and thousands of hours of video.
- Deep learning requires substantial computing power.
  - High-performance GPUs have a parallel architecture that is efficient for deep learning.
  - When combined with clusters or cloud computing, this enables development teams to reduce training time for a deep learning network from weeks to hours or less.

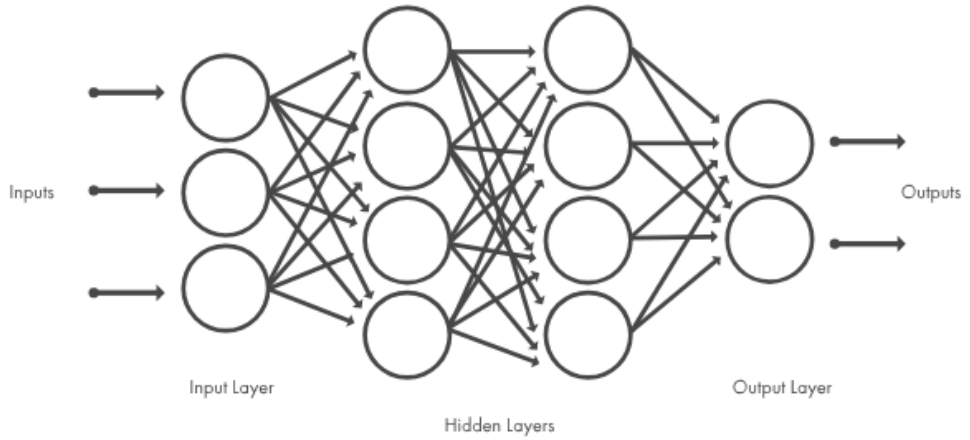
DEEP LEARNING ALGORITHMS

Most deep learning methods use neural network architectures, which is why **deep learning models** are often referred to as **deep neural networks**.

The term "**deep**" usually refers to the number of hidden layers in the neural network.

- Traditional neural networks only contain 2-3 hidden layers, while deep networks can have as many as 150.

Deep learning models are trained by using large sets of labeled data and neural network architectures that learn features directly from the data without the need for manual feature extraction.



DEEP LEARNING ALGORITHMS

Deep learning is now one of the most active fields in machine learning and has been shown to improve performance in image and speech recognition.

The potential of deep learning in high-throughput biology is clear

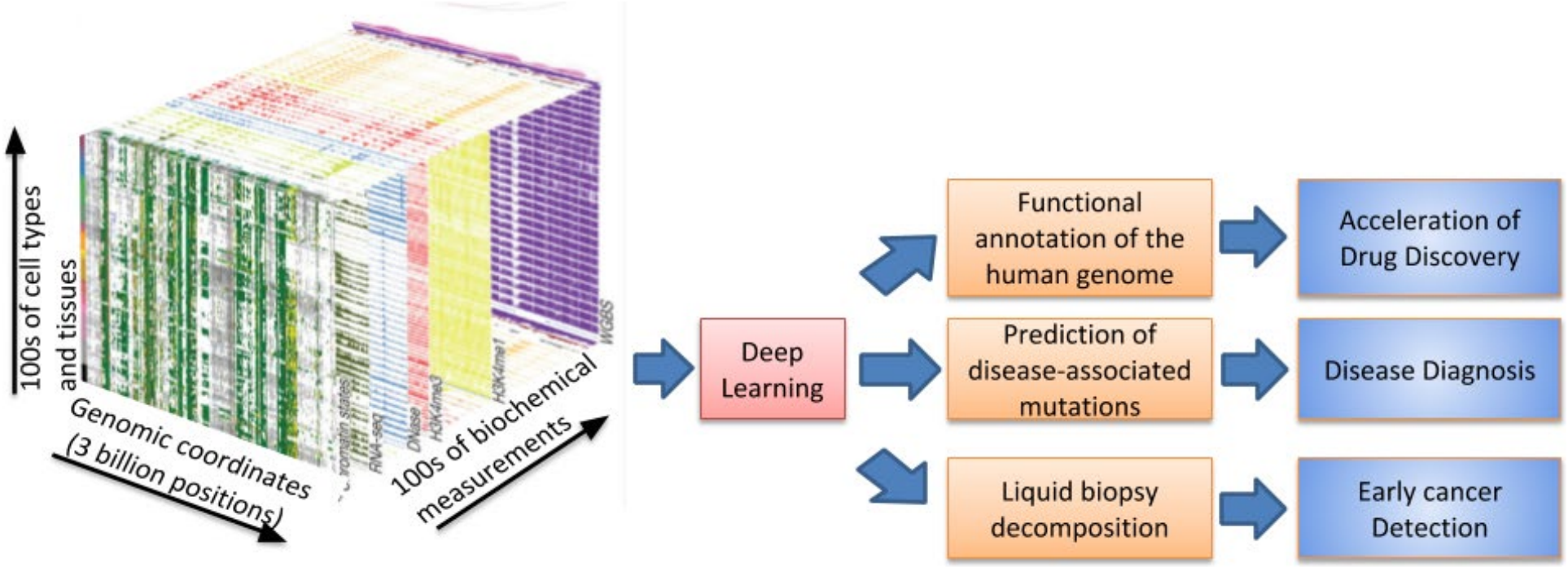
- it allows to better exploit the availability of increasingly large and high-dimensional data sets (e.g. from DNA sequencing, RNA measurements, flow cytometry or automated microscopy) by training complex networks with multiple layers that capture their internal structure

DEEP LEARNING ALGORITHMS

Example

- **Multi-label Deep Learning for Gene Function Annotation in Cancer Pathways** [Renchu Guan, Xu Wang, Mary Qu Yang, Yu Zhang, Fengfeng Zhou, Chen Yang & Yanchun Liang Scientific Reports volume 8, (2018)]
- Applied deep learning to explore full texts of biomedical articles containing detailed methodologies, experimental results, critical discussions and interpretations can be found, for the analysis of gene multi-functions relevant to cancer pathways derived from full-text biomedical publications.
  - Without the involvement of a biologist to do a feature study about the data.
- Experimental results on eight KEGG cancer pathways revealed that this new system is not only superior to classical multi-label learning models, but it can also achieve numerous gene functions related to important cancer pathways.

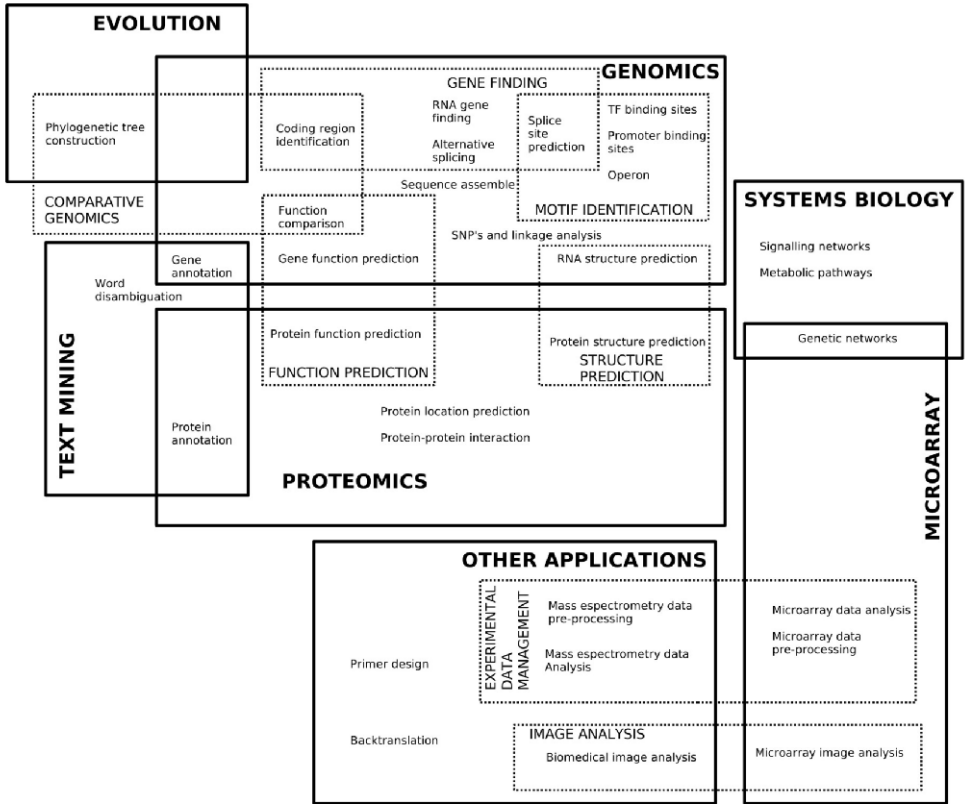
OPPORTUNITIES FOR DEEP LEARNING IN GENOMICS



https://towardsdatascience.com/opportunities-and-obstacles-for-deep-learning-in-biology-and-medicine-6ec914fe18c2

In closing

APPLICATIONS OF ML IN BIOINFORMATICS



From: Machine learning in bioinformatics
Brief Bioinform. 2006;7(1):86-112. doi:10.1093/bib/bbk007

IS THERE A PERFECT ML TECHNIQUE?

There is not one solution (one machine learning algorithm) or one approach that fits all problems.

For each problem, there is not one single solution.

WHICH TECHNIQUE TO USE?

Size, quality and nature of the data to be analysed.

The question, the answer expected, and also expected accuracy.

How the result will be used

Time and computing resources available.

Always good to check performance of different algorithms and compare results.

WHAT KIND OF DATA DO YOU HAVE?

If the data to be analysed is unlabelled and the aim is to find structure, it is an unsupervised learning problem.

If the aim is to optimize an objective function by interacting with an environment, it is a reinforcement learning problem.

When supervised learning is feasible, it is often the case that additional, unlabelled data points are easy to obtain.

How do you decide whether it's a supervised or semi-supervised approach?

A good rule of thumb is to use semi-supervised learning if you do not have very much labelled data and you have a very large amount of unlabelled data

WHAT IS THE EXPECTED OUTPUT?

If the output of your model is a number, it is a regression problem.

- Two-class classification of gene expression data

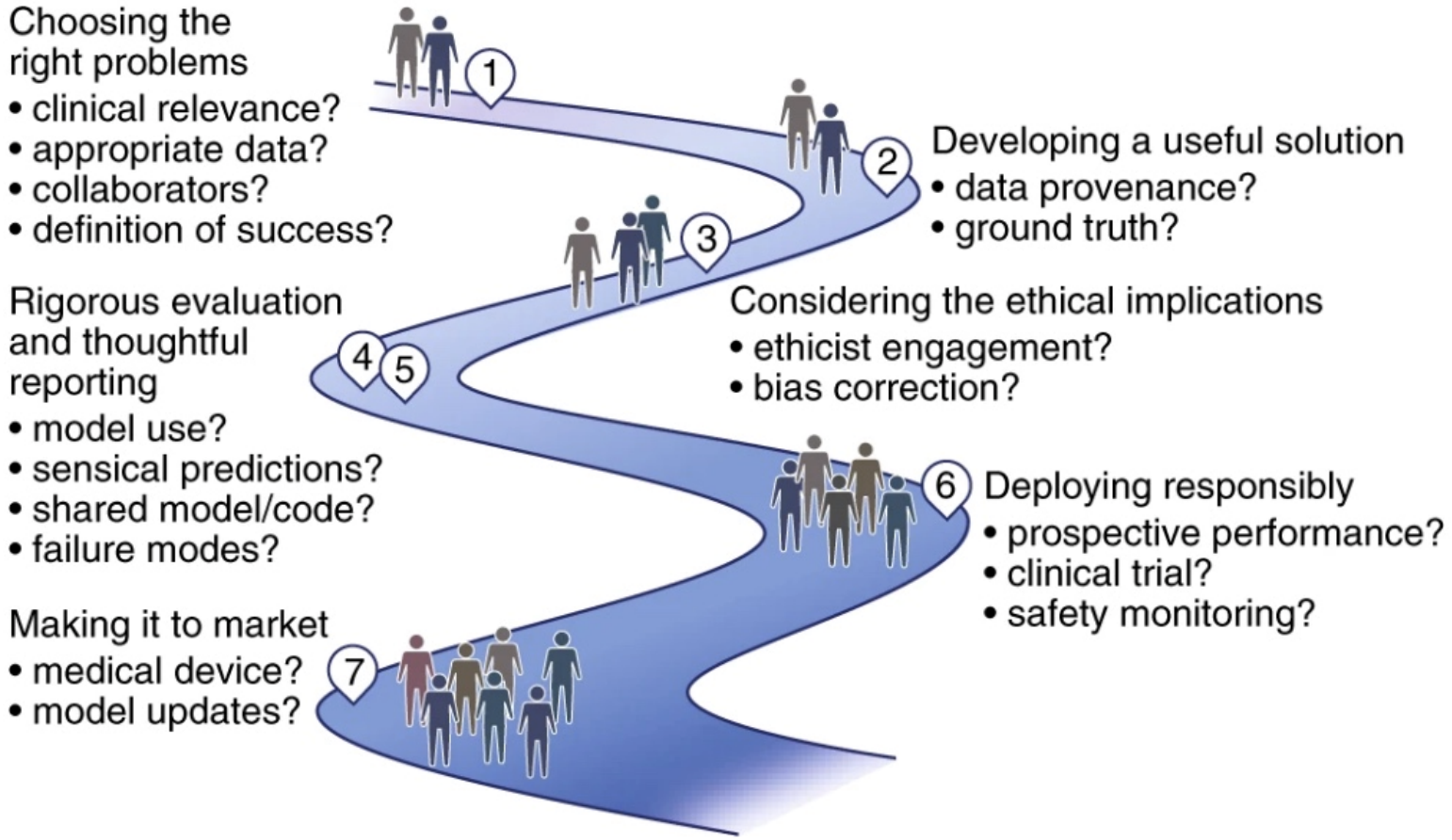
If the output of your model is a class, it is a classification problem.

- Genomic classification of AML

If the output of your model is a set of input groups, it is a clustering problem.

- Patterns in gene expression at different developmental stages of zebrafish

DO NO HARM: A ROADMAP FOR RESPONSIBLE MACHINE LEARNING FOR HEALTH CARE



https://www.nature.com/articles/s41591-019-0548-6/figures/1

TOOLS

All the methods listed above are already available either in Python, R (https://www.r-project.org/about.html) or Matlab using existing packages. Some basic code needs to be written.

If you are not used to writing code, you may use a tool like WEKA (https://www.cs.waikato.ac.nz/ml/weka/) or RapidMiner (https://rapidminer.com/) – the methods are already implemented and you simply need to load your data in either csv, arff,… format and run the selected methods.

Some useful R packages R implementing many ML techniques: https://cran.r-project.org/web/views/MachineLearning.html

SOME ONLINE RESOURCES

https://machinelearningmastery.com/start-here/

https://www.datascience.com/blog

https://www.mathworks.com/discovery/machine-learning.html

https://www.coursera.org/browse/data-science

SOURCES

http://www.sthda.com/english/articles/29-cluster-validation-essentials/97-cluster-validation-statistics-must-know-methods/#data-preparation

https://medium.mybridge.co/30-amazing-machine-learning-projects-for-the-past-year-v-2018-b853b8621ac7

Shakuntala Baichoo and Zahra Mungloo slides (H3ABionet, ML group)

NOW GO FORTH AND ML! ☺