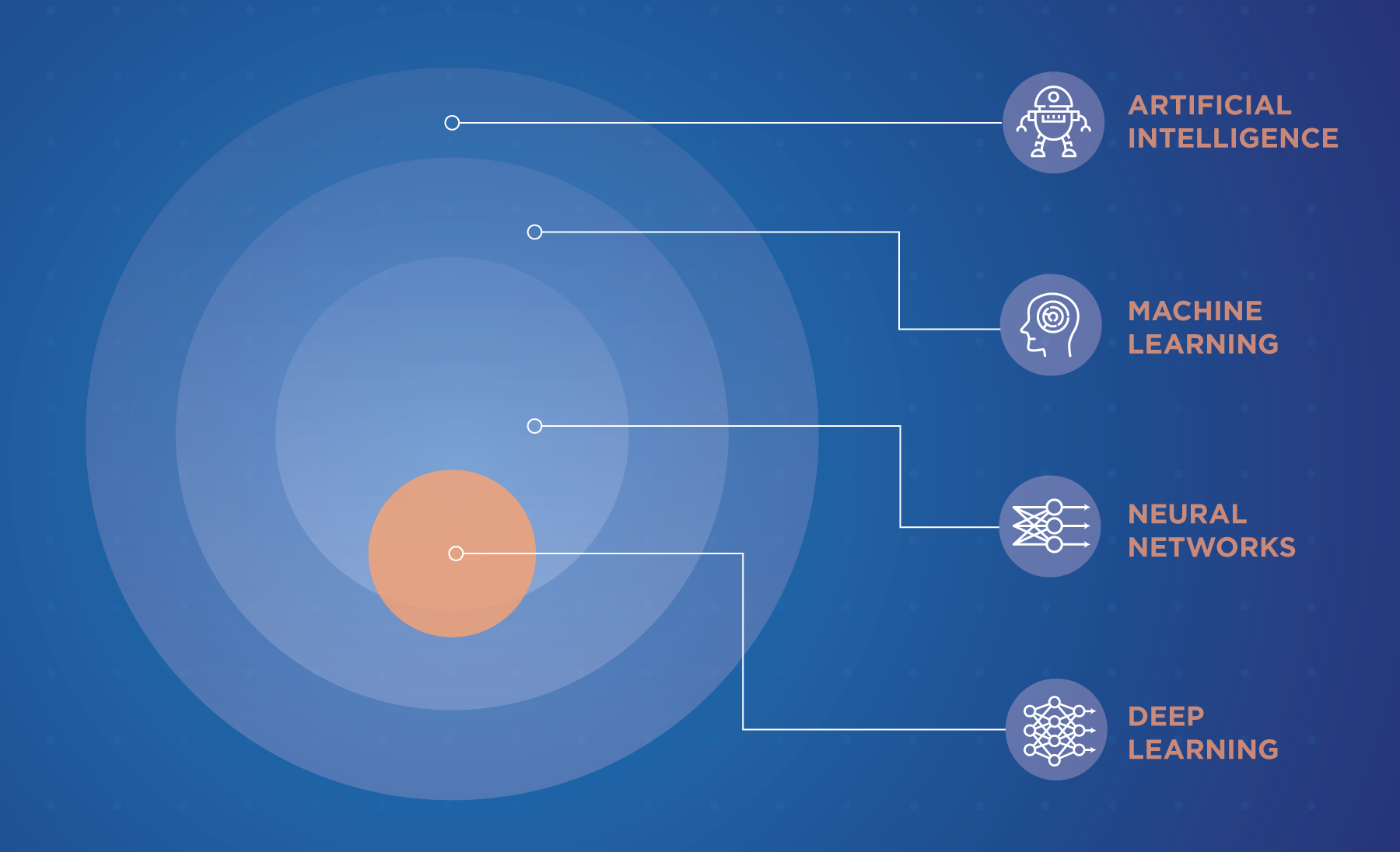
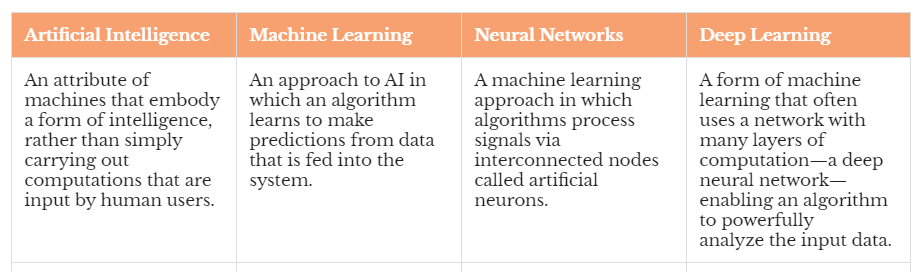
Machine learning

# Introduction

When people think about Artificial Intelligence, they typically have Neural Nets on mind. However, AI refers to any machine that can replicate human cognitive skills, such as problem solving.

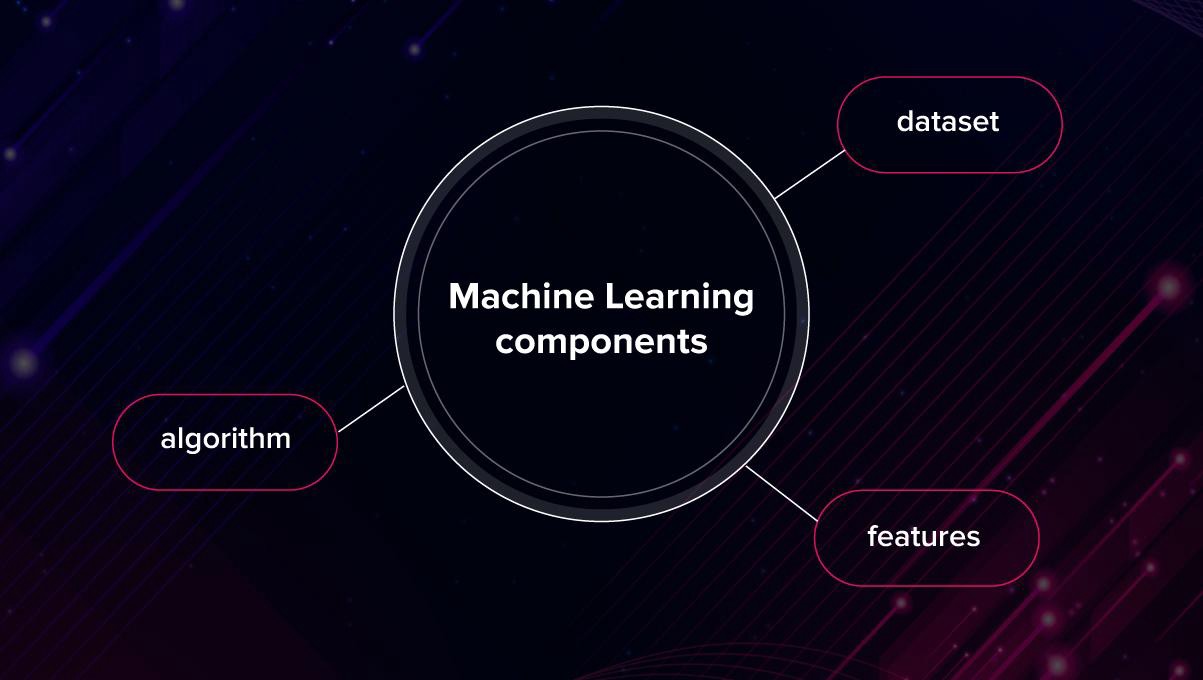




* Machine learning is a subset of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed. In ML, there are different algorithms (e.g. neural networks) that help to solve problems.
* Deep learning, or deep neural learning, is a subset of machine learning, which uses the neural networks to analyse different factors with a structure that is similar to the human neural system.

# Machine Learning

Machine learning focuses on teaching computers how to learn without the need to be explicit programmed for specific tasks. In fact, the key idea behind ML is that it is possible to create algorithms that learn from data and make predictions on data.



* **Datasets**: datasets can include numbers, images, texts or any other kind of data.
* **Features**: important pieces of data that work as the key to the solution of the task. They demonstrate to the machine what to pay attention to. Features may be size, name, type, weight, etc. Some features are special, and they are called **labels**. Labels are the features that you are trying to predict based on the other features. There is also term **target** / **target attribute**. The term label and target are often used interchangeably.
* **Algorithm**: different algorithms are used to solve the same problems. Algorithms may be combined.

Example:

* Problem – predicting the price of a house based on the previous transactions.
* Dataset – housing transactions from the last year.
* Feature - location, standard, no of rooms and bathrooms, furnished/unfurnished, price.
* Label – price because you are trying to predict price based on other features. Values are known, as that is a dataset of transactions from the past.

The problem can be solved by finding relationship between the price (dependent variable) and other features (independent variables). The good approach when you have numeric dataset and try to predict numeric variable is to use regression algorithms. Simple linear regression is a type of regression analysis where the number of independent variables is one and there is a linear relationship between the independent(x) and dependent(y) variable. In that case the challenge for machine is to find the best values of linear regression parameters, which would provide the best fit line for the data points. The bigger challenge is when datasets have many independent variables. Regression algorithms train model using dataset by comparing computed (predicted) result to the known values (labels). Algorithm may run iteratively and during each iteration try to reduce errors (distance between predicted result and known target/labelled values) by changing parameters. The result of training is a **machine learning model**, that can be used for predicting unknow values. Such type of ML model is called **regression model**. It is used to predict **continuous values**.

**Datasets** can be:

* **labelled dataset** – dataset that contains the known and correct target value/answer. Example: a transactional price of house. Labelled data are used in **supervised learning**.
* **Unlabelled dataset** – dataset that does not contain result/target value, because they were not labelled by human. Unlabelled data are used in **unsupervised learning**.

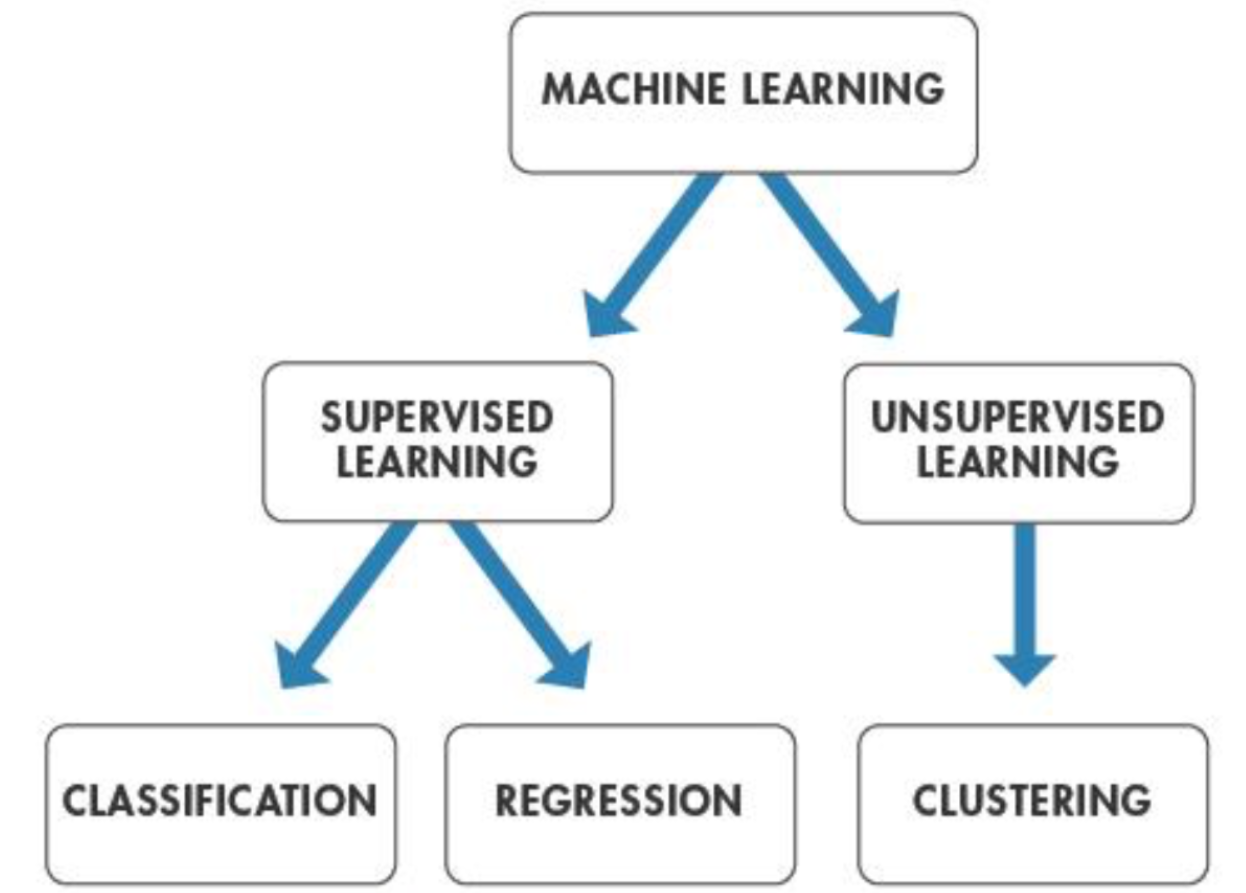
**Supervised learning** means, that machine learning algorithm uses labelled data to train model. In supervised learning, the algorithm builds a model of a set of data that contains both the inputs (features) and the desired outputs (labels). **Regression algorithms** and **classification algorithms** are types of supervised learning.

**Unsupervised learning** means, that machine learning algorithm uses unlabelled dataset to train model. In the unsupervised learning, the algorithm builds a model using a set of data, which contains only inputs and no desired outputs. Unsupervised learning algorithms are used to find the clusters of data. The algorithm figures out the data, tries to find pattens and makes clusters of data with new labels. **Clustering** is used in unsupervised learning.

Labelled training dataset is much better for training purpose. The algorithm can run iteratively to find the best set of parameters. In every iteration it compares its output to the labelled data and in next step it tries to reduce the number of wrong answers by adjusting **parameters**. During the learning process, the program learns how to get to the “right” solution, or simply it finds the best parameters.

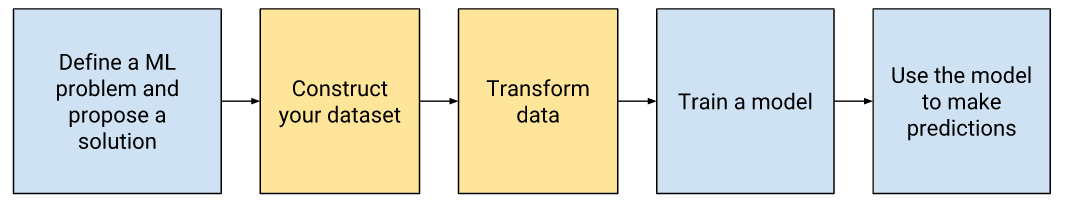
Categorical dependent variables are classified into classes or categories according to shared qualities or characteristics. For example, a dataset of animals containing features width, height, weight can be classified into classes and labelled accordingly i.e.: cow, cat, mouse, etc. Classification algorithms are used to build a machine learning **classification model**. It is used to predict **categorical values**.

Unlabelled datasets are also useful. Machine can check if there is something in a dataset. Machine may find similarities between subsets of data and create **clusters**. This process is called **clustering**. The goal of this machine learning technique is to find similarities in the data point and group similar data points together and to figure out to which cluster new data should belong. Clustering is used for knowledge discovery rather than prediction. Example: Dublin house transactions dataset, unlabelled. Price is treated as a feature. Machine learning model clustered data. It was discovered that one cluster contains expensive houses that are located, surprisingly, along Luas line.



How supervised learning works?

* Implement machine learning algorithm or use the existing one.
* Construct **training dataset**. Usually it requires **data pre-processing** (transforming data from string to numbers, cleaning dataset, dealing with missing data, outliers, etc.)
* **Train model** using training dataset. Machine learning algorithms train a **machine learning model** based on training dataset.
* Test the performance of model using **test dataset**.
* Use metrics (i.e., **accuracy**) to check the model performance.
* If model performance is good enough, use it for **predicting** result of target datasets.



Supervised machine learning algorithms use training dataset, where the **input** and the **output** are known. Input are features data and output are labelled data. The algorithm learns by comparing its actual production with correct outputs (labels) to find errors. Then it modifies the model parameters to reduce errors.

And then, the validation dataset is used to **tune hyperparameters** to avoid **overfitting**. Your model is overfitting when it performs extremely well during training and very poorly on test dataset. In such case hyperparameters are tuned. The validation dataset is used to avoid overfitting.

**Parameters** are configuration variables that can be thought to be internal to the model as they can be estimated from the training data. Algorithms have mechanisms to optimize parameters. On the other hand, **hyperparameters** cannot be estimated from the training data. Hyperparameters of a model are set and tuned depending on a combination of some heuristics and the experience and domain knowledge of the data scientist.

The term test dataset and validation dataset are often used interchangeably. But the “**validation dataset**” is used to describe the evaluation of models when tuning hyperparameters, and the “**test dataset**” is used to describe the evaluation of a final tuned model when comparing it to other final models.

**Accuracy**: Accuracy is a metric for evaluating **classification models**. Informally, accuracy is the fraction of predictions our model got right.

Be careful with interpretation of model accuracy. Example: a training dataset of images, half of them with number 1. ML model predicts which image contains 1. If model accuracy is 0.5 it may be, that the model just ‘guesses’ the result, instead of doing any useful job. The probability of ‘guessing’ correct result is also 0.5. But if there are 10 different numbers (probability of ‘guessing’ is 0.1) and accuracy of model 0.5 – it is quite good. Similarly, recognition of fake news (true, false) and accuracy 0.5. Another pitfall is with **class-imbalanced data set**. For example, a training dataset contains 90% non-fake news and 10% fake news. ML model accuracy may be 0.9 just because all news were classified as non-fake news. It is good to check if any of the fake news were classified correctly. There are also other metrics.

**Scikit-learn**, also known as **sklearn** is a machine learning library for the Python programming language. It features various classification, regression and clustering algorithms. Scikit-learn provides dozens of built-in machine learning algorithms and models, called **estimators**. Each estimator can be fitted to some data using its **fit method**.

Popular algorithms used in machine learning and implemented in scikit-learn library:

* **Support Vector Machines (SVMs)** implemented in classification (SVC Support Vector Classification) and regression (SVR Support Vector Regression)
* **K-Nearest Neighbours (KNN, k-NN)** implemented in classification (KNeighbours Classifier) and regression (Nearest Neighbours Regression)
* **K-Mean** implemented in clustering.

I made an example of Machine Learning in Jupyter using Support Vector Classification. Here are some basic tutorials about **Support Vector Classification**, if you want to know how it works.

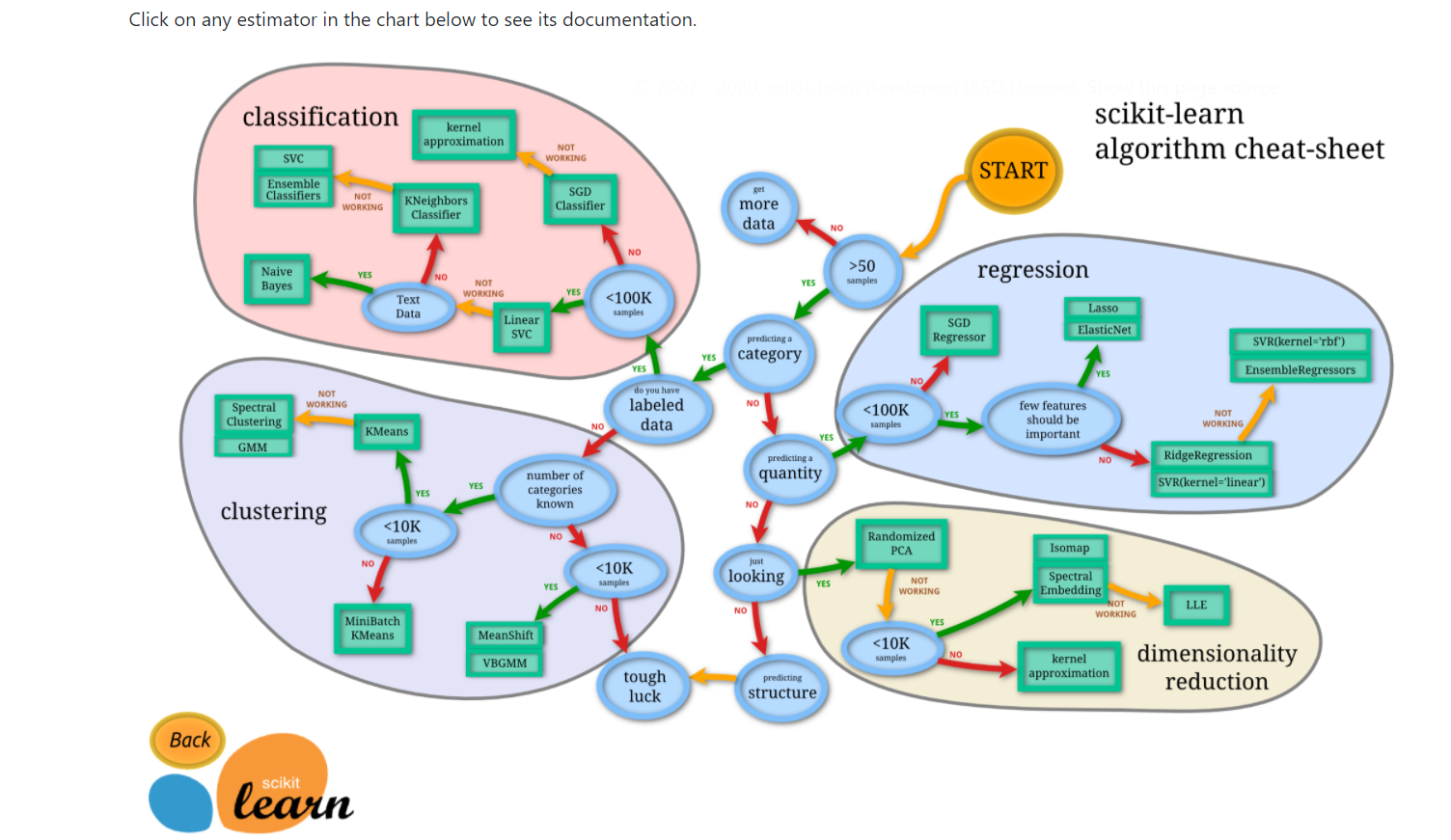
Basic (just 3 minutes but good): <https://www.youtube.com/watch?v=N-sPSXDtcQw>

More stuff: <https://www.youtube.com/watch?v=Y6RRHw9uN9o>

Practical example: <https://www.youtube.com/watch?v=N1vOgolbjSc>

The scikit-learn cheat-sheet helps to find the right estimator for the job:

<https://scikit-learn.org/stable/tutorial/machine_learning_map/index.html>



**Dimensionality reduction** part, that is on right-bottom scikit-learn cheat-sheet, includes algorithms to reduce dimensions of dataset. In machine learning area **dimensions** means the number of features (10 dimensions means that there are 10 features). To many dimensions may have a bad impact on training model. It is called **the curse of dimensionality**. This phenomenon states that with a fixed number of training samples, the average (expected) predictive power of a classifier or regressor first increases as the number of dimensions or features used is increased but beyond a certain dimensionality it starts deteriorating instead of improving steadily. **Principal component analysis (PCA)** algorithm is commonly used to reduce the number of dimensions.

See tutorial if you want to learn more about scikit-learn:

Getting started: <https://scikit-learn.org/stable/getting_started.html>

Tutorial: <https://scikit-learn.org/stable/tutorial/index.html>