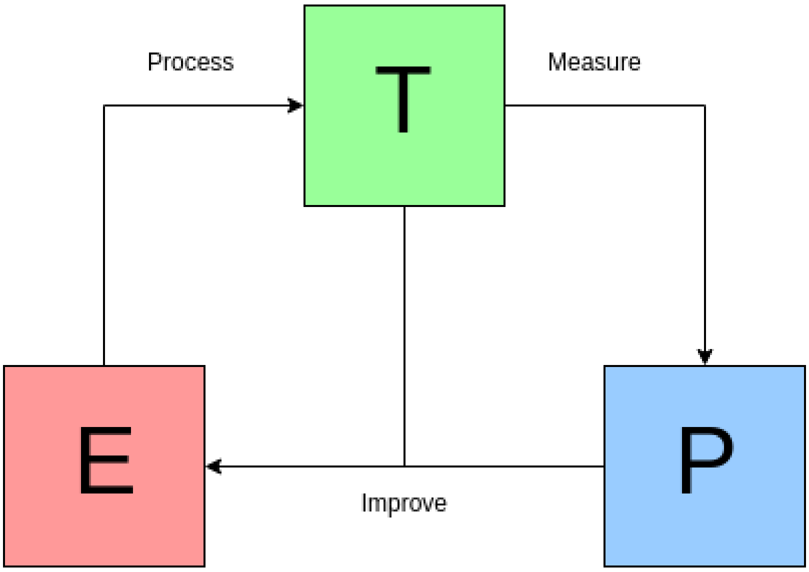
Introduction about Machine Learning

What is Machine Learning?

- Machine learning is not something like a futuristic fantasy. Many people when hearing machine learning, they often pictures a robot, but it's not, it's already here. For example, that is *spam filters* that can learn to flag spam given examples of spam emails.
- To answer for the question what is we can simply understand that → Machine learning is just a program that can **learn from data**
- From the science view point:

  A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E. (Tom Mitchell



Why we use Machine Learning?

to make your life easier 😇.

Imagine that you have 1 millions of emails and your task is to classify what is spam and not spam email. Can you measures how much of your time spend on this task compare to computers. Its cost you much effort than computers that you can not imagine 🙂

Types of Machine Learning:

▼ Supervised/Unsupervised Learning:

  ▼ Supervised Learning:

  - The training data you feed to the algorithms that have *labels*. **The computer try to learn with a teacher**
  - For example, the spam filters model is trained with many emails along with their *labels* (spam or not spam) and after learn from training data, the model can predict new emails that is spam or not.
  - Some popular supervised learning algorithm:
    1. Linear Regression
    2. Logistic Regression
    3. K-Nearest Neighbors
    4. Support Vector Machines (SVMs)
    5. Decision Trees and Random Forests
    6. Neural networks

  ▼ Unsupervised Learning

  - As you can guess from the name, that unsupervised learning given the training data is unlabeled. **The computer try to learn without a teacher.**
  - For example, you want to detect group of similar visitors (what group a visitors belong to). When visitors come to your webpage, the computer will clustering them into several group for demands and use this data to recommends the visitors something that they **might need.**
  - There are many application of unsupervised learning such as *visualization algorithms, dimensionality reduction, anomaly detection, association rule learning*... I will go through of this later. You can search key word of these term on the Internet to understand the idea of each.

  - Some popular algorithm:
    - ▼ Clustering
      - K-Means
      - DBSCAN
      - Hierarchical Cluster Analysis (HCA)
    - ▼ Anomaly detection and novelty detection
      - One-class SVM
      - Isolation Forest
    - ▼ Visualization and dimensionality reduction
      - Principal Component Analysis (PCA)
      - Kernel PCA
      - Locally-Linear Embedding (LLE)
      - t-distributed Stochastic Neighbor Embedding (t-SNE)
    - ▼ Association rule learning
      - Apriori
      - Eclat

- ▼ Semi-supervised Learning
  - This algorithm can deal with *labeled and unlabeled training data.* **The teacher can help a half and the other half you must learn by yourself**
  - For example, Google Photos automatically recognize the same person A shows up in the photo 1,5,11. While another person B is shows up in photos 2,5,7. This is unsupervised part. But the algorithm needs you to tell it who these people are so you need to name everyone in the photo (this is as label data). And after that you can search the name of person and algorithm will return you photos that this person shows up. This is supervised part.
  - Some popular algorithms:
    - Deep belief networks (DBNs)
    - Restricted Boltzmann machines (RBMs)

- ▼ Reinforcement Learning
    - *This is a very different beast*
    - The learning system (*agent*) observe the environment, select and perform actions, and get *rewards (or penalties)*. The task is that it must learn by itself what is the best strategy, called *policy* to get the most reward over time.
    - For example DeepMind's AlphaGo

▼ Batch and Online Learning

- ▼ Batch Learning (Offline learning)
    - In this the system can not learn incrementally, it must be trained with all the available data.
    - First it will be trained with all available data and then it is deploy into production and runs without learning anymore (this is call *offline learning*). When you want the system know about the new data, there is only way that you must trained the system from the beginning with all data include old data and new data. After you have the new model then stop the old system and replace it with the new one.
    - This solution is very simple and also works fine, but training again the whole dataset that costs a lot of computing resources.
- ▼ Online Learning
    - For solving the problem from the offline learning we have the new one, *online learning*. In this, you train the system incrementally by feeding it data instances sequentially (individuals or small groups call *mini-batches*). Each step of learning is very fast and cheap, so the system can learn about new data on the fly.
    - This is great for adapting to change rapidly or autonomously and also your limited computing resources. For a huge dataset, the algorithms loads part of the data, training in this part and repeat until run all of the dataset.

Two main approaches to generalization:

▼ **Instance-based Learning**

- This is simply learn by heart (the most trivial way 👎). The system learns the examples by heart, then generalizes to new cases by comparing them to the learned examples, using *similarity measure*
- For example, the spam filter in this way would cluster all the emails that are identical or very similar to emails that have already been labeled by user. That will count the number of words they have in common and flag as spam if it has many words in common with spam emails

▼ Model-based Learning

- From the assumption about the given data are made explicit in the form of a model. We imagine or generalize from a set of data to build a model from given dataset, then use that model to make predictions. This is called model-based learning
- For example, the data is given and from this we use any tools to plot the graph and recognize that seem to be a trend in the graph. Although the data is *noisy,* it still look like linearly. So you decide to select *linear model* to make a prediction this step is called *model selection*.