Chapter 2 summary

- **Business and ML objectives:**

  It's important to integrate between business and ML metrics, and most of ML developers just and data scientists focus on ML metrics without considering business metrics and that causes to kill the system.

  So, what metrics do companies care about?

  The ultimate goal of any project within a business is, therefore, to increase profits and cutting costs.

  What are the requirements for ML Systems?

- **Requirements for ML Systems:**

  1- **Reliability:**

  The system should continue to perform correctly even in fault cases such as (SW, HW…) or even humans' errors, also the system should be able to modify wrong predictions and not giving false predictions to end users.

  2- **Scalability:**

  The ML systems should consider growth in future such as using new models, more predictions handling, use bigger models or more complicated ones and many other reasons may cause breaks for the system if those reasons weren't considered from the beginning of developing the system.

  3- **Maintainability**:

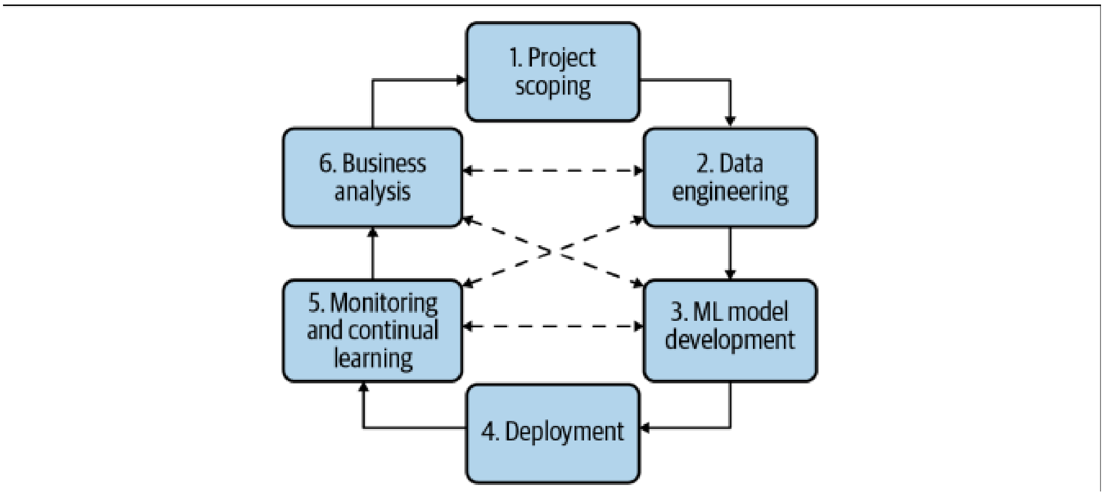
  The system with is different parts (code, models, data …. etc.) should be easy and will be documented to make other people (ML engineers, DevOps developer…. Etc.) able to contribute with the system and define problems to find solutions easily.

  4- **Adaptability:**

  The system should be adaptable with different distributions of data since the data is changeable very quickly so the system should have the capacity to evolve.

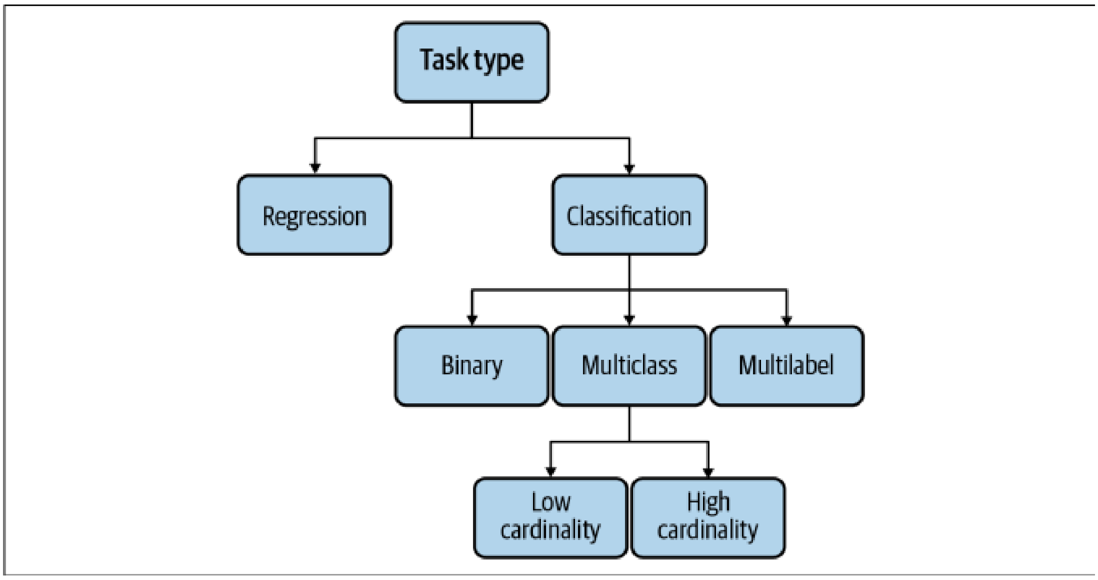
- **Iterative Process:**

  Developing an ML system is an iterative process not linear one.



- **Framing ML Problems:**

  Types of ML Tasks:



Classification models classify inputs into different categories, when regression models outputs continuous values, regression problem can easily be converted into classification one and vice versa.

Binary VS Multiclass classification:

in binary classification we try to classify data into 2 classes whatever they are such as (cat or dog) image classification, in other side multi class classification is to classify data into mor than 2 classes such as **Hand digit recognition** since we try to classify the digit into 10 classes (0~9).

Multiclass versus multilabel classification:

In both multiclass and binary classification, the input belongs to one class in cases it belongs to more than one we have **Multilabel classification.**

we can treat multilabel classification as a multiclass or binary classification, as multiclass by representing the prediction as a vector, let's say we have a multilabel classification problem and we have 4 classes then the vector [0 1 1 0] means that the input belongs to 2^{nd} and 3^{rd} classes.

Also, we can treat it by have binary classification model for each class and gathering the result of all binary models, each model represents if the input belongs to the class or not.

Multiple ways to frame a problem:

As any problem in the real life, there are many ways to solve it.

Also, frame ML problem and consider that the way that we frame the problem can make it easier or harder to solve. Like using multiclass classification in a problem that has continuous changing classes, it's bas approach to frame the problem as multiclass classification because we need to retrain the model whenever the number of classes changes.

Objective Functions:

An objective function is also called a loss function, because the objective of the learning process is usually to minimize (or optimize) the loss caused by wrong predictions.

Decoupling objectives:

when there are multiple objectives, it’s a good idea to decouple them first because it makes model development and maintenance easier.

Mind Versus Data:

In ML community there is continuous debate about what's important data or learning algorithms, we can't ignore that data is very important for ML systems and without good data there is no good ML systems but also the data in changing over time and more data isn't always helpful for your system maybe it affects the system in negative way.

In my opinion there should be balance between data analyzing and improving learning algorithms to make successful ML systems in the end.