Outline - Hands-On Machine Learning

with Scikit-Learn, Keras, and TensorFlow

2nd Edition by Aurélien Géron

**Machine Learning** is about making machines get better at some tasks by learning from data, instead of having to explicitly code rules.

**There are many different types of ML systems:**

* Supervised or not supervised,
* Batch or Online,
* Instance-based or Model-based.

**In an ML project**

1. you gather data in a training set, and
2. you select a (model or) learning algorithm
   1. **Model selection** consists in choosing the type of model and fully specifying its architecture.
3. you feed the training set to a learning algorithm (train the model/learning algorithm)
   1. **Training a model** means running an algorithm to find the model parameters that will make it best fit the training data (and hopefully make good predictions on new data).
      1. If the algorithm is model based, it tunes some parameters to fit the model to the training set (i.e., to make good predictions on the training set itself), and then hopefully it will be able to make good predictions on new cases as well.
      2. If the algorithm is instance-based, it just learns the examples by heart and generalizes to new instances by using a similarity measure to compare them to the learned instances.
   2. **How can you know which parameter values will make your model perform best?** 
      1. To answer this question, you need to specify a performance measure. You can either define a ***utility function*** (or *fitness function*) that measures how *good* your model is, or you can define a ***cost function***that measures how *bad* it is. (*For Linear Regression problems, people typically use a cost function that measures the distance between the linear[[1]](#footnote-1) model’s predictions and the training examples; the objective is to minimize this distance.*)
4. **Challenges – If your model does not make good predictions:** two things that can go wrong are **“bad data”** and **“bad algorithm”**
   1. **“BAD DATA”**
      1. **Insufficient Quantity of Training Data –** Data matters more than algorithms for complex problems. It has been shown that where there are enough data to train an algorithm, very different Machine Learning algorithms, including simple ones, performed almost identically well on a complex problem.
      2. **Nonrepresentative Training Data –** in order to generalize well, It is crucial to use a training set that is representative of the (new) cases you want to generalize to.
         1. if the sample is too small, you will have ***sampling noise*** (i.e., nonrepresentative data because of chance),
         2. but even very large samples can be non-representative if the sampling method is flawed. This is called ***sampling bias***.
      3. **Poor-Quality Data –** if your training data is full of errors, outliers, and noise (e.g., due to poor quality measurements), it will make it harder for the system to detect the underlying patterns, so your system is less likely to perform well.
      4. **Irrelevant Features –** A critical part of the success of a Machine Learning project is coming up with a good set of features to train on. This process, called *feature engineering*, involves the following steps:
         1. *Feature selection* – selecting the most useful features to train on among existing features.
         2. *Feature extraction*– combining existing features to produce a more useful one—as we saw earlier, dimensionality reduction algorithms can help.
         3. Creating new features by gathering new data.
   2. **“BAD ALGO”**
      1. **Overfitting the Training Data –** your model needs to be neither too complex.
      2. **Underfitting the Training Data –** your model needs to be nor too simple.
5. **Once you have trained a model** you want to **evaluate** it and **finetune** it if necessary.

1. A linear model makes the assumption that the data is fundamentally linear and that the distance between the instances and the straight line is just noise, which can safely be ignored. [↑](#footnote-ref-1)