
Gianluca Malato
Supervised Machine Learning Workflow
YOUR DATA TEACHER

The purpose of this e-book

In this e-book, I propose the typical **Supervised Machine Learning workflow** that data scientists follow when they need to create a supervised model. This workflow **may change** from project to project, but the steps I'm going to show you in the next sections are the most **common steps** that are required. For each step of the workflow, I'll give you a brief introduction and I'll suggest one or more **Python libraries** that may be used in a project.

Step 1: Data extraction

The first step of a machine learning model is related to data. Any supervised model **learns from data** we feed it with. Data can be extracted from a Data Lake, a Data Warehouse, some Excel and CSV files, an SQL database or other sources. The purpose of this phase is to **collect data** and **build the dataset** we are going to work with. The dataset is a **huge table** made of several columns (sometimes hundreds of columns) and thousands of rows. Each column is called “feature” and they are needed to predict the “target” column.

Python libraries: pandas, numpy

Step 2: Exploratory Data Analysis

After we have gathered our data, we must **take a look at it**. It’s the purpose of the Exploratory Data Analysis (EDA). In this phase, we use **graphical representations** of our dataset in order to discover the most important variables, the correlations, the orders of magnitude, and the **statistical properties** of the features with respect to the target. The purpose of EDA is to **extract information** from our dataset to better understand the phenomena inside our data and to start figuring out **which features are useful** and which others are useless.

Python libraries: pandas, numpy, scipy, seaborn, matplotlib

Step 3: Data cleaning

Once we have defined our dataset, we need to clean it **filling the missing values** in the features. This procedure is called “cleaning” and it’s very important because not every model is able to handle missing data, so we need to fill the blanks in some way.

Python libraries: scikit-learn

Step 4: Encoding

If our variables are **categorical** (i.e. not numerical), some models may not work properly with them. The majority of the models, in fact, can only handle numerical data. So, we need to convert categorical features to numerical features. This procedure is called “encoding”.

Python libraries: scikit-learn

Step 5: Transforming

Some models require particular **transformations** to be applied to a dataset before it can be used. For example, scaling the features to the same order of magnitude, symmetrize their probability distributions and other types of transformations. These numerical transformations are necessary to make our model **work properly** and they change according to the model.

Python libraries: scikit-learn, imblearn, scipy

Step 6: Dimensionality reduction

After we have transformed the features, a good choice is to **reduce the dimensionality** of our dataset removing the useless features. A first reduction has been made in the EDA phase, but a new reduction can be done as well if our dataset is still large and the features are still **correlated with each other**.

Python libraries: scikit-learn

Step 7: Model selection

We can now **select a supervised model** that is able to **generalize** the training dataset and make **accurate predictions** even on datasets it has not been trained on. Every model has **its own needs** about the encoding of the categorical features and the transformations to be applied, so when we perform this research, we must keep in mind such **requirements**.

Python libraries: scikit-learn

Step 8: Hyperparameter tuning

Once the model has been selected, we can **fine-tune its hyperparameters**, which are some parameters whose values are set before the training phase. Hyperparameter tuning is necessary in order to make a model that **better generalizes** the training dataset.

Python libraries: scikit-learn

Step 9: Recursive Feature Elimination

Once the model has been created and its hyperparameters have been optimized, we can use it for **further feature selection** and **dimensionality reduction**. It's not mandatory, but it **can be helpful**. One common choice is to use a procedure called Recursive Feature Elimination.

Python libraries: scikit-learn

Step 10: Feature importance and model interpretation

Finally, we must **interpret and explain** our model and calculate the **importance of the features** in order to **catch and understand the information** behind data and explain the phenomena that created our dataset. This is the last part of a machine learning workflow, but it's probably **the most important one**. Understanding how our model works

and how **information flows through data** is the core of every data science project.

Python libraries: scikit-learn, shap

Suggested courses

For the **data manipulation** and transformation process, I suggest attending my *Data pre-processing for Machine Learning in Python* online course.

For the **model selection**, the **hyperparameter** tuning, the **recursive feature elimination** and the calculation of **feature importance**, I suggest attending my *Supervised Machine Learning in Python* online course.

If you need a custom training program, you can benefit from my **one-to-one coaching** program. Just send me a message and we'll talk about building a training program made by remote video lessons.

Who am I?

My name is **Gianluca Malato**, I'm **Italian** and have a Master's Degree *cum laude* in **Theoretical Physics** of disordered systems at "La Sapienza" University of Rome.

I'm a **Data Scientist** who has been working for years in the banking and insurance sector. I have extensive experience in software programming and project management and I have been dealing with **data analysis** and **machine learning** in the corporate environment for several years.

I've written **many articles** about Machine Learning, R and Python and I've been a **Top Writer** on Medium.com in Artificial Intelligence category.

I teach Data Science on YourDataTeacher.com

My e-mail address is gianluca@yourdatateacher.com

YOUR
DATA TEACHER

Table of contents