Important:

Sample class balance -> cases in which you have samples with few positives and many negatives. Also see : <http://machinelearningmastery.com/tactics-to-combat-imbalanced-classes-in-your-machine-learning-dataset/>

Also see : <https://svds.com/learning-imbalanced-classes/>

Python: https://github.com/scikit-learn-contrib/imbalanced-learn

Feature selection.

Techniques for classification:

-kNN

-SVM

-decision tress and random forests

-naïve bayes

-neuronal networks

Techniques for regression:

-SVM

-decision tress and random forests

-naïve bayes

-neuronal networks

In order to improve the predictions:

-Detecting if your model present high bias or high variance and if so, applying techniques to solve it.

-Missing values? if so….

- Outlier detection

- Features selection

- Esemble Learnig.

FOR MACHINE LEARNING:

scikit-learn library in python

<http://scikit-learn.org/stable/tutorial/basic/tutorial.html>

http://ipython-books.github.io/featured-04/

times series : http://stackoverflow.com/questions/30346605/time-series-forecasting-with-scikit-learn

FOR OPTIMIZATION:

Scipy.optimize in python -> differential evolution