**Modules**

1. Model construction
2. model training
3. model testing
4. model evaluation

**Model construction** depends on machine learning algorithms. In this projects case, it was neural networks.

Such an algorithm looks like:

1. begin with its object: model = Sequential()
2. then consist of layers with their types: model.add(type\_of\_layer())
3. after adding a sufficient number of layers the model is compiled. At this moment Keras communicates with TensorFlow for construction of the model. During model compilation it is important to write a loss function and an optimizer algorithm. It looks like: model.comile(loss= ‘name\_of\_loss\_function’, optimizer= ‘name\_of\_opimazer\_alg’ ) The loss function shows the accuracy of each prediction made by the model.

Before model training it is important to scale data for their further use.

After model construction it is time for **model training.**In this phase, the model is trained using training data and expected output for this data.

It’s look this way: model.fit(training\_data, expected\_output).

Progress is visible on the console when the script runs. At the end it will report the final accuracy of the model.

Once the model has been trained it is possible to carry out **model testing.**During this phase a second set of data is loaded. This data set has never been seen by the model and therefore it’s true accuracy will be verified.

After the model training is complete, and it is understood that the model shows the right result, it can be saved by: model.save(“name\_of\_file.h5”).

Finally, the saved model can be used in the real world. The name of this phase is **model evaluation**. This means that the model can be used to evaluate new data.