**Measuring Data Quality**

There are commonly used parameters to calculate data quality;

*Completeness*: the percent of data fields that have values entered into them

*Accuracy*: the percentage of values that are correct compared to the actual value (for age feature being in range ~[0-120])

*Consistency*: the percent of values that match across different tables/files

*Validity*: the percentage of data that have values within the domain of acceptable values (being in an acceptable data format etc.)

*Timeliness*: if the data is dependent on time, it has a negative effect (trying to predict house prices using data from the 1990s, etc.)

Other features which should be considered;

* Correlation between columns: if it is too high, some columns do not contribute to the model, just consume memory
* imbalanced data: it is better to have balanced data
* Data distribution: a balanced distribution have a positive effect
* The volume of data: the bigger the data is, the better for the models, especially for deep learning

We can form a metric using all the parameters above. There is also an open-source python library we may use or inspire:<https://github.com/ydataai/ydata-quality>

Another important thing we have to think about is the structure of data. Are we going to have it as a plain csv file or json file? The data can also be in a relational database format with lots of tables, then we have to know how to join them. Apart from them, the data may be in the no-sql format as well. Then we might have to do some preprocess and give the data as plain csv files, which is easier to use. I can be represented as a dataframe that is the basic input format for most of the models.

**Validating Model Code**

We can check some functions related to copying data, but the code could be in many different languages and normally, the code would be encrypted. They may not want us to see their code. Even we can detect copying data etc. some execution might be confused with other commands (like in virus scans), the user may really need to copy data inside the code in preprocessing steps. Another point here is, can the user use an external service (api) to send information outside?

What are we going to do if we are suspicious? Are we going to ask the user if they are copying the data?

The only solution I can think of currently is that we can check the size of the model and see if it is reasonable. I will search more on this and update here when I find anything.