Machine Learning Evaluation Tests Design Document

Remark : All these tests are **Evaluation tests** , and they will be executed once the training of our model is done and before passing to the creation of the API .

1. Data Integrity tests :

- **New Label Train Test :**

  - **Goal of the test :** Detect new labels in the test set
  - **Features to be tested :**
    The test will compare if there any new labels in the possible modalities between the columns of the train and test sets
  - **Dependencies of the test :** Train.csv , Test.csv files
  - **Tool used :** Deepchecks
  - **Feature pass/fail criteria :** This test can have either the state pass or fail . The condition to pass is Ratio of samples with new label is not greater than 0%

2. Train Test Validation tests:

- **Train Test Feature Drift :**

  - **Goal of the test :** Calculate drift between train dataset and test dataset per feature, using statistical measures.
  - **Features to be tested :**
    For numerical features, the check uses the Earth Movers Distance method and for the categorical features it uses the PSI. The check calculates drift between train dataset and test dataset per feature, using these 2 statistical measures.
  - **Dependencies of the test :** Train.csv , Test.csv files
  - **Tool used :** Deepchecks

- **Feature pass/fail criteria :** This test can have either the state pass or fail . The condition to pass is PSI <= 0.2 and Earth Mover's Distance <= 0.1

- **Train Test Label Drift :**

  - **Goal of the test :** Calculate label drift between train dataset and test dataset, using statistical measures
  - **Features to be tested :**
    The Drift score is a measure for the difference between two distributions, in this check - the test and train distributions.
    The check shows the drift score and distributions for the label.
  - **Dependencies of the test :** Train.csv , Test.csv fiéles
  - **Tool used :** Deepchecks
  - **Feature pass/fail criteria :** This test can have either the state pass or fail . The condition to pass is PSI <= 0.2 and Earth Mover's Distance <= 0.1

- **Train Test Prediction Drift :**

  - **Goal of the test :** Calculate prediction drift between train dataset and test dataset, using statistical measures
  - **Features to be tested :**
    For detecting the drift between the two distributions
    Deepchecks uses :

    - For regression problems, the Population Stability Index (PSI)
    - For classification problems, the Wasserstein Distance (Earth Mover's Distance)
  - **Dependencies of the test :** Train.csv , Test.csv files
  - **Tool used :** Deepchecks

- **Feature pass/fail criteria :** This test can have either the state pass or fail . The condition to pass is PSI <= 0.15 and Earth Mover's Distance <= 0.075 for model prediction drift

- **Whole dataset Drift :**

  - **Goal of the test :** Calculate drift between the entire train and test datasets using a model trained to distinguish between them.
  - **Features to be tested :**
    A whole dataset drift, or a multivariate dataset drift, occurs when the statistical properties of our input feature change .
    The difference between a feature drift (or univariate dataset drift) and a multivariate drift is that in the latter the data drift occurs in more than one feature.
    Practically, the check concatenates the train and the test sets, and assigns label 0 to samples that come from the training set, and 1 to those who are from the test set.
    Then, we train a binary classifier of type Histogram-based Gradient Boosting Classification Tree, and measure the drift score from the AUC score of this classifier.
  - **Dependencies of the test :** Train.csv , Test.csv files
  - **Tool used :** Deepchecks
  - **Feature pass/fail criteria :** This test can have either the state pass or fail . The condition to pass is Drift value is not greater than 0.25 .

3. Model Evaluation tests :

- **Model Inference Time - Train Dataset :**

  - **Goal of the test :** Measure model average inference time (in seconds) per sample.
  - **Features to be tested :**

This test aims to make sure that our model doesn't need a long time to make the inference .

- **Dependencies of the test :** Train.csv , Test.csv , model.pkl files
- **Tool used :** Deepchecks
- **Feature pass/fail criteria :** This test can have either the state pass or fail . The condition to pass is that the average model inference time for one sample is not greater than 0.001

- **Unused Features :**

  - **Goal of the test :** Detect features that are nearly unused by the model
  - **Features to be tested :**

    Having too many features can prolong training times and degrade model performance due to "The Curse of Dimensionality" or "Hughes Phenomenon".

    Features with low model contribution (feature importance) are probably just noise, and should be removed as they increase the dimensionality without contributing anything. Nevertheless, models may miss important features. For that reason the Unused Features check selects out of these features those that have high variance, as they may represent information that was ignored during model construction. We may wish to manually inspect those features to make sure our model is not missing important information.
  - **Dependencies of the test :** Train.csv , Test.csv , best_model.pkl files
  - **Tool used :** Deepchecks
  - **Feature pass/fail criteria :** This test can have either the state pass or fail . The condition to pass is that the number of high variance unused features is not greater than 5

- **Single Feature Contribution - Train / Test Dataset :**

  - **Goal of the test :** Return the PPS (Predictive Power Score) of all features in relation to the label
  - **Features to be tested :**
    The Predictive Power Score (PPS) is used to estimate the ability of a feature to predict the label by itself . A high PPS (close to 1) can mean that this feature's success in predicting the label is actually due to data leakage - meaning that the feature holds information that is based on the label to begin with.
  - **Dependencies of the test :** Train.csv , Test.csv , model.pkl files
  - **Tool used :** Deepchecks
  - **Feature pass/fail criteria :** This test can have either the state pass or fail . The condition to pass is that the Features' Predictive Power Score is not greater than 0.8 .

- **Performance Report :**

  - **Goal of the test :** Summarize given scores on a dataset and model
  - **Features to be tested :**
    This check helps you compare your model’s performance between two datasets. The default metric that are used are F1, Precision, and Recall for Classification and Negative Root Mean Square Error, Negative Mean Absolute Error, and R2 for Regression.
  - **Dependencies of the test :** Train.csv , Test.csv , model.pkl files
  - **Tool used :** Deepchecks
  - **Feature pass/fail criteria :** This test can have either the state pass or fail . The condition to pass is that the Train-Test scores relative degradation is not greater than 0.1

- **ROC Report - Train Dataset / Test Dataset :**

  - **Goal of the test :** Calculate the ROC curve for each class

- **Features to be tested :**

  This check shows the roc curve to check the performance of the model

- **Dependencies of the test :** Train.csv , Test.csv , model.pkl files

- **Tool used :** Deepchecks

- **Feature pass/fail criteria :** This test can have either the state pass or fail . The condition to pass is that the AUC score for all the classes is not less than 0.7

- **Simple Model Comparison :**

  - **Goal of the test :** Compare the given model score to a simple model score (according to the given model type)
  - **Features to be tested :**

    The simple model is designed to produce the best performance achievable using very simple rules. The goal of the simple model is to provide a baseline of minimal model performance for the given task, to which the user model may be compared. If the user model achieves less or a similar score to the simple model, this is an indicator of a possible problem with the model (e.g. it wasn't trained properly).

  - **Dependencies of the test:** Train.csv , Test.csv, model.pkl files

  - **Tool used:** Deepchecks

  - **Feature pass/fail criteria:** This test can have either the state pass or fail. The condition to pass is that the Model performance gain over a simple model is not less than 10%.

- **Calibration Metric - Train Dataset / Test Dataset :**

  - **Goal of the test :** Calculate the calibration curve with brier score for each class
  - **Features to be tested :**

    Calibration curves (also known as reliability diagrams) compare how well the probabilistic predictions of a binary classifier are calibrated. It

plots the true frequency of the positive label against its predicted probability, for binned predictions.

- **Dependencies of the test :** Train.csv , Test.csv , model.pkl files
- **Tool used :** Deepchecks
- **Feature pass/fail criteria :** No condition

- **Confusion Matrix - Train Dataset / Test Dataset :**

  - **Goal of the test :** Calculate the confusion matrix of the model on the given dataset
  - **Features to be tested :**

    A confusion matrix is a table that is used to define the performance of a classification algorithm. A confusion matrix visualizes and summarizes the performance of a classification algorithm.
  - **Dependencies of the test :** Train.csv , Test.csv , model.pkl files
  - **Tool used :** Deepchecks
  - **Feature pass/fail criteria :** No condition