### ML 24/25-02 Investigate Input reconstruction by using Classifiers

Please first read about classifiers in general. Try to understand the differences and explain them. Additionally, explain why and how they can be used with HTM. Then, investigate both existing classifiers (HtmClassifier and KNN), which were implemented last year by students.  
<https://github.com/numenta/nupic-legacy/blob/master/tests/integration/nupic/algorithms/knn_classifier_test/classifier_test.py>  
Examples of Classifiers - Engineering / Machine Learning - [HTM Forum](https://discourse.numenta.org/t/examples-of-classifiers/6897) (numenta.org). Please take a look into MultisequenceLEarning experiment to see how classifier is used.

Classifiers inside HTM implement the following interface:  
IClassifier<TIN, TOUT>.

Example: How to learn using classifier:  
cls.Learn(key, actCells.ToArray());

Example: How to read predicting SDR from classifier:  
var predictedInputValues = cls.GetPredictedInputValues(lyrOut.PredictiveCells.ToArray(), 3);

The experiment [*SpatialLearning*](https://github.com/ddobric/neocortexapi/blob/master/source/Samples/NeoCortexApiSample/SpatialPatternLearning.cs) demonstrates learning. Inspired by this experiment, your task is to implement a new experiment (it does not have to be completely different), which regenerates the input from the generated SDR.

The general approach of classifiers can be described as the reverse function of the HTM encoding:

INPUT -> SCALER ENCODER -> SP -> SDR -> HTM CLASSIFIER -> INPUT

-> KNN CLASSIFIER -> INPUT

The black part above is already implemented in the ***SpatialLearning* experiment**. The red part represents the new part, which must be implemented in this project.  
  
The new code must:

1. Load many scaler inputs from a folder.
2. Run the experiment and wait for it to complete.
3. At the end of the experiment, invoke the classifier via the **IClassifier interface** to reconstruct the input.
4. Finally, compare the reconstructed input with the original input using similarity functions available in the ***neocortexapi*.** The difference between inputs and reconstructed inputs must be clearly illustrated.
5. Provide diagrams that explain result.
6. Compare results of both classifiers.
7. Discuss results.

Intended visualisations:

1. HTM output vs original input (similarity graph)
2. KNN output vs original input (similarity graph)