COMP47750/COMP47990
Machine Learning with Python

Tutorial 13

Model Selection

Preliminaries

The notebook **`13 Model Selection Tutorial`** shows how to fit a *k*-NN model for the HotelRevHelpfulness dataset. It assesses three options:

- whether to use a StandardScaler, MinMaxScaler or no scaler.
- what *k* to use for *k*-NN
- what weighting policy

Q1

Two Naive Bayes options for building classifiers on the Hotels dataset are GaussianNB and CategoricalNB with discretization. Scikit-learn provides a basic discretizier KBinsDiscretizer https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.KBinsDiscretizer.html. Compare three options using pipelines and cross validation:

- Gaussian Naive Bayes
- Gaussian Naive Bayes on scaled data
- Categorical Naive Bayes with discretization, try (KBinsDiscretizer(encode = 'ordinal'))

Q2

Find the best decision tree model for the HotelRevHelpfulness dataset considering max_leaf_nodes and the splitting criterion. The splitting criterion can be either 'gini' or 'entropy', you can select your own options for max_leaf_nodes.

Q3

One of the not-so-good things about pipelines is that they hide a lot of the detail about what is going on. **`scikit-learn`** classifiers (estimators) provide **`fit`** and **`predict`** methods and transformers provide **`fit`** and **`transform`** methods.

Consider the following pipeline code - in terms of what you know the pipeline must achieve:

```
1. kNNpipe   = Pipeline(steps=[
2.     ('imputer', KNNImputer(missing_values = np.nan)),
3.     ('scaler', StandardScaler()),
4.     ('classifier', KNeighborsClassifier())])
In [150]:
5. kNNpipe.fit(X_train, y_train)
6. y_pred = kNNpipe.predict(X_test)
7. print("Accuracy: {0:4.2f}".format(accuracy_score(y_test,y_pred)))
8. confusion_matrix(y_test, y_pred)
```

1. In line 5 what **`fit`**, **`predict`** and **`transform`** methods (on what classes) are called? What data is passed to these methods?
2. In line 6 what **`fit`**, **`predict`** and **`transform`** methods (on what classes) are called? Again, what data is passed to these methods?

Hint: It may help to visualise the pipeline in terms of the pipeline diagrams shown in the Model Selection slides.

Q4

'Cheating' is a term used in ML to describe scenarios where test data is used in model selection or model training. If test data is to be used to estimate generalisation accuracy it should not be accessible during model selection or training.

However, some scenarios where test data is accessed during model setup are much more serious than others. In the following scenarios where data is divided into train and test sets, score on a scale of 0 to 4 the seriousness of performing the operation before splitting the data. Zero is harmless and four is serious (you go straight to hell!).

a) Missing value imputation using mean value replacement is performed before data is split.
b) Missing value imputation using k-NN imputation is performed before data is split.
c) Cyclical encoding is used to generate new representations of temporal features before splitting.
d) Feature selection is performed before splitting.
e) Model is trained before the data is split.
f) Data is scaled before splitting.
g) Model hyper-parameters are determined before splitting.
h) SMOTE is used (on all the data) to up-sample the minority class.