Satej Soman
CAPP30254: Machine Learning for Public Policy
Spring 2019

HW 3
MACHINE LEARNING PIPELINE IMPROVEMENTS & EVALUATION

Contents

Notes

- Representative code snippets are interspersed with analysis and explanations below; all code is available on GitHub: `https://github.com/satejsoman/capp30254/tree/master/hw3`.
- The pipeline library lives in the `code/pipeline` directory, while the sample application which imports the library is `code/donors_choose.py`.

1 Coding: Pipeline Improvements

1.1 Improvements

The following improvements have been made to the `pipeline` library:

- A stage to generate train/test data splits. Due to the object-oriented design, the current implementation can easily be overriden by subclassing or monkey-patching in specific applications.
- Additional metrics, besides accuracy, have been added to the model evaluation. Specifically, precision, recall, and ROC-AUC have been added to the model evaluation stage.

1.2 Additional Models

In the original pipeline design, the model was not hard-coded, so pipeline runs can be parametrized by the model implementation:

```
donors_choose_preprocessors = [
    ...
]
donors_choose_feature_generators = [
    ...
]

models_to_run = {
    ...
}

def model_parametrized_pipeline(description, model):
    return Pipeline(input_path, "funded_in_60_days",
        summarize=False,
        data_preprocessors=donors_choose_preprocessors,
        feature_generators=donors_choose_feature_generators,
        name="donors-choose-" + description,
        model=model
        output_root_dir="output")

for (description, model) in models_to_run.items():
    model_parametrized_pipeline(description, model).run()
```

The above code is purely representative; in implementation, the data generation and feature preprocessing are done in a separate pipeline. The results of this pipeline are serialized and fed in as the inputs to a pipeline that solely trains and tests models. In this way, computation to process data and create feature vectors is not repeated for every trial run.