Letter Recognition using Google Prediction API

About Data

Before I write about goodness of predictions, I used Letter Recognition data set available at UCI Machine Learning repository, here the task is to identify images as one of 26 capital letters and trained using 16 features describing image properties. The data has 20,000 items and it is divided into train and test using 80:20 split. Each category has around 600 and around 150 items for training and testing respectively with 26 categories. The python script to divide data and generate csv files in Prediction API's training format [code: data_div.py].

Using Prediction API:

I felt that the developer's guide is limited and there are only few examples about usage. I had hard time figuring out how to actually use the API productively and ended up using other online resources to understand how to train a model and make predictions

Results:

I didn't find how to make a batch request to predict for all test data at once so each item in test data is predicted separately. I used classification report from scikit-learn metrics to assess the model and the results are bad, with 0.42 average precision. As the model is a black box I cannot further investigate on results. Even Individual category predictions didn't seem to be good as some classes had no single correct prediction i.e., zero precision [code: the_api_call.py].

I used the same data to build a model using SVM and gridsearchcv using scikit-learn. The model worked well using rbf kernel with 0.98 average precision and all individual categories had above 95 percent accuracy [SVM code: py_predicition.py].

The Classification Report results are available at report.txt

Final Thoughts:

-The Prediction API promises automatic model selection and tuning but how much can you trust machine tweaked parameters?

-Black box models provide no insight into your data and as data is in cloud you cannot make any visualizations on it. Ability to make visualizations on cloud will possibly solve this problem.

Note: This report is solely based on my experience.