

The 'config.yaml' is as defined in the following:

```
deploymentUri: "gs://iris-sklearn/iris_20181119_050517"
runtimeVersion: '1.8'
framework: "SCIKIT_LEARN"
pythonVersion: "3.5"
```

Run the following command to create a prediction service.

```
source ./scripts/create-prediction-service.sh
```

```
Creating model...
```

```
Created ml engine model [projects/quantum-ally-219323/models/iris_sklearn].
```

```
Creating model version...
```

```
Creating version (this might take a few minutes).....done.
```

## Make Online Predictions from the Scikit-learn Model

The code to make an online prediction from the Scikit-learn model is shown in the following and is stored in the file 'online-prediction.sh'. In online predictions, the input data is passed directly as a JSON string.

```
export JOB_NAME=iris_sklearn_prediction
export MODEL_NAME=iris_sklearn
export MODEL_VERSION=v1
export TEST_FILE_GCS=gs://iris-sklearn/test-sample.json
export TEST_FILE=./test-sample.json

# download file
gsutil cp $TEST_FILE_GCS .

# submit an online job
gcloud ml-engine predict --model $MODEL_NAME \
    --version $MODEL_VERSION \
    --json-instances $TEST_FILE

echo "0 -> setosa, 1 -> versicolor, 2 -> virginica"
```

The input data stored as a JSON string is shown in the following.

```
[5.1, 3.5, 1.4, 0.2]
```

Run the following command to execute an online prediction request to the hosted model on Cloud MLE.

```
source ./scripts/online-prediction.sh

Copying gs://iris-sklearn/test-sample.json...
/ [1 files][ 20.0 B/ 20.0 B]
Operation completed over 1 objects/20.0 B.
[0]
0 -> setosa, 1 -> versicolor, 2 -> virginica
```

In this chapter, we discuss training large-scale models using Google Cloud Machine Learning Engine, which is a part of the Google AI Platform. In the examples in this chapter, we trained the models using the Estimator High-level API and Scikit-learn. It is important to mention that the Keras high-level API can also be used to train large-scale models on Cloud MLE.

In the next chapter, we will cover training custom image recognition models with Google Cloud AutoML.

## CHAPTER 42

# Google AutoML: Cloud Vision

Google Cloud AutoML Vision facilitates the creation of custom vision models for image recognition use cases. This managed service works with the concepts of transfer learning and neural architecture search under the hood to find the best network architecture and the optimal hyper-parameter configuration of that architecture that minimizes the loss function of the model. This chapter will go through a sample project of building a custom image recognition model using Google Cloud AutoML Vision. In this chapter, we will build an image model to recognize select cereal boxes.

## Enable AutoML Cloud Vision on GCP

Step through the following steps to enable AutoML Cloud Vision on GCP:

1. Open Cloud Vision by clicking the triple dash at the top-left corner of the GCP dashboard. Select **Vision** under the product section **ARTIFICIAL INTELLIGENCE** as shown in Figure 42-1.