Estimator

Learn how to use TensorFlow's Estimator API for model development.

Chapter Goals:

- Create an `Estimator` object for the regression model

A. Creating an `Estimator`

TensorFlow's `Estimator` object provides an organized and simple API for model execution. It handles model training, saving and restoring checkpoints, evaluating a model, and making predictions.

To initialize an `Estimator` object, we pass in the model function as a required argument. The model function should follow the same template as `regressor_fn`. The function must return an `ExampleSpec` object, which specifies the model results for training, evaluation, or prediction.

The two main keyword arguments to know are `model_dir` and `params`. The `model_dir` argument represents the directory we save model checkpoints to. The `params` argument represents the values we wish to pass into the model function. The argument should be set to a dictionary, which then corresponds to the model function's `params` argument.

```
import tensorflow as tf
params = {
    'feature_columns': feature_columns,
    'hidden_layers': hidden_layers
}
regressor = tf.estimator.Estimator(
    regressor_fn,
    model_dir=ckpt_dir,
    params=params)
```

Creating the Estimator object for the regression model.

In our example, we initialized the `Estimator` object with `regressor_fn` as its model function. We set the checkpoint directory to `ckpt_dir`, and passed in

the feature columns and number of hidden layers through `params`.