

tf.contrib.learn.learn_runner.tune

```
tune(  
    experiment_fn,  
    tuner  
)
```

Defined in [tensorflow/contrib/learn/python/learn/learn_runner.py](#).

Tune an experiment with hyper-parameters.

It iterates trials by running the Experiment for each trial with the corresponding hyper-parameters. For each trial, it retrieves the hyper-parameters from `tuner`, creates an Experiment by calling `experiment_fn`, and then reports the measure back to `tuner`.

Example:

```
def _create_my_experiment(run_config, hparams):  
    hidden_units = [hparams.unit_per_layer] * hparams.num_hidden_layers  
  
    return tf.contrib.learn.Experiment(  
        estimator=DNNClassifier(config=run_config, hidden_units=hidden_units),  
        train_input_fn=my_train_input,  
        eval_input_fn=my_eval_input)  
  
    tuner = create_tuner(study_configuration, objective_key)  
  
    learn_runner.tune(experiment_fn=_create_my_experiment, tuner)
```

Args:

- `experiment_fn`: A function that creates an `Experiment`. It should accept an argument `run_config` which should be used to create the `Estimator` (passed as `config` to its constructor), and an argument `hparams`, which should be used for hyper-parameters tuning. It must return an `Experiment`.
- `tuner`: A `Tuner` instance.

Except as otherwise noted, the content of this page is licensed under the [Creative Commons Attribution 3.0 License](#), and code samples are licensed under the [Apache 2.0 License](#). For details, see our [Site Policies](#). Java is a registered trademark of Oracle and/or its affiliates.

Last updated November 2, 2017.

Stay Connected

[Blog](#)

[GitHub](#)

[Twitter](#)

Support

[Issue Tracker](#)

[Release Notes](#)

[Stack Overflow](#)

English

[Terms](#) | [Privacy](#)