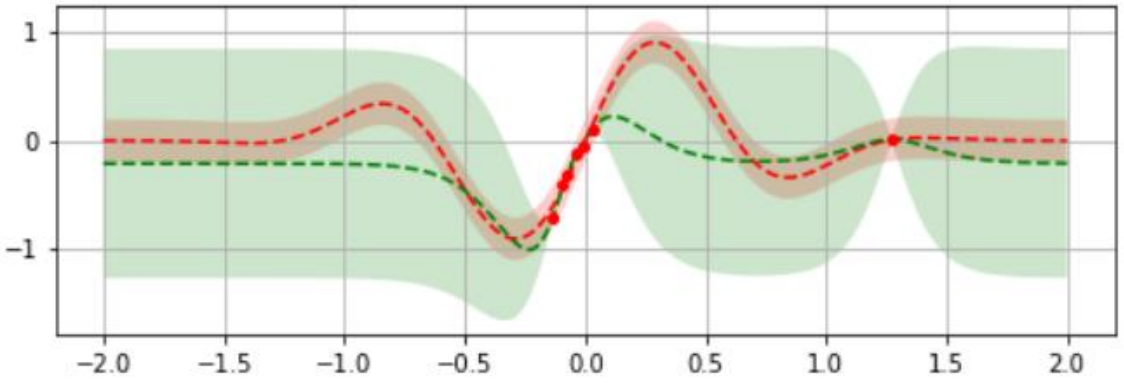
Train In Data

TPE

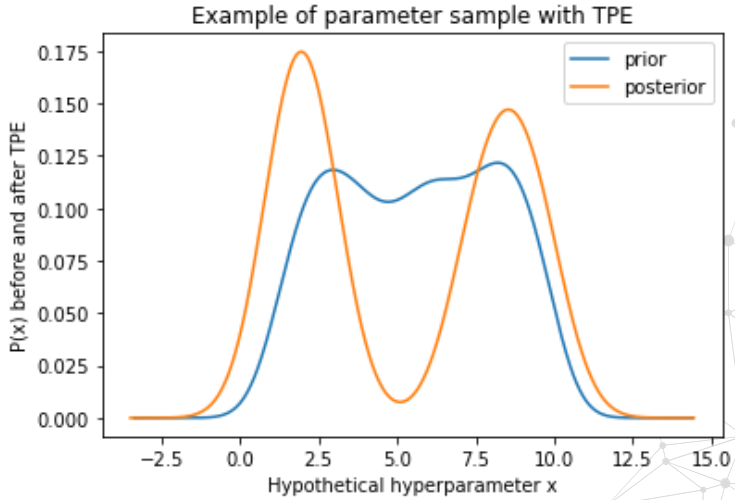
Why tree structured?

GP vs TPE

- *Surrogate model for f(x)* $-P(y\,|x)$
- *f(x) takes a value given all hyperparameters*



- *Surrogate model for* $P(x\,|y)$
- *1 probability per hyperparameters*
  - *Hyperparameter spaces can be nested*



Why Tree-structured?

Because TPE estimates a distribution for each hyperparameter, it will only estimate distributions for parameters when they "exist".

```
space = hp.choice('classifier_type', [
{
    'type': 'svm',
    'C': hp.lognormal('svm_C', 0, 1),
    'kernel': hp.choice('svm_kernel', [
        {'ktype': 'linear'},
        {'ktype': 'RBF', 'width': hp.lognormal('svm_rbf_width',
0, 1)},
        ]),
},
{
    'type': 'dtree',
    'criterion': hp.choice('dtree_criterion', ['gini',
'entropy']),
    'max_depth': hp.choice('dtree_max_depth',
        [None, hp.qlognormal('dtree_max_depth_int', 3, 1, 1)]),
    'min_samples_split':
hp.qlognormal('dtree_min_samples_split', 2, 1, 1),
}])
```

Node 1

Parameters only make sense inside each Node

Node 2

Why Tree-structured?

Because TPE estimates a distribution for each hyperparameter, it will only estimate distributions for parameters when they "exist".

```
space = hp.choice('classifier_type', [
{
    'type': 'svm',
    'C': hp.lognormal('svm_C', 0, 1),
    'kernel': hp.choice('svm_kernel', [
        {'ktype': 'linear'},
        {'ktype': 'RBF', 'width': hp.lognormal('svm_rbf_width',
0, 1)},
        ]),
},
{
    'type': 'dtree',
    'criterion': hp.choice('dtree_criterion', ['gini',
'entropy']),
    'max_depth': hp.choice('dtree_max_depth',
        [None, hp.qlognormal('dtree_max_depth_int', 3, 1, 1)]),
    'min_samples_split':
hp.qlognormal('dtree_min_samples_split', 2, 1, 1),
}])
```

Sub - Node 1

Sub - Node 2

Node 1

Parameters only make sense inside each Node

Node 2

Train In Data

THANK YOU

www.trainindata.com