

twinlab.ModelSelectionParams

```
class twinlab.ModelSelectionParams(seed=None, evaluation_metric='MSLL',
val_ratio=0.2, base_kernels='restricted', depth=1, beam=None)
```

Parameter configuration for the Bayesian model selection process.

Variables:

- **seed** (*Union[int, None], optional*) – Specifies the seed for the random number generator for every trial of the model selection process. Setting to an integer is necessary for reproducible results. The default value is `None`, which means the seed is randomly generated each time.
- **evaluation_metric** (*str, optional*) – Specifies the evaluation metric used to score different configuration during the model selection process. Can be either:
 - `"BIC"`: Bayesian information criterion.
 - `"MSLL"`: Mean squared log loss.
 The default is `"MSLL"`.
- **val_ratio** (*float, optional*) – Specifies the percentage of random validation data allocated to to compute the `"BIC"` metric. The default is `0.2`.
- **base_kernels** (*Union[str, Set[str]], optional*) – Specifies the set of individual kernels to use for compositional kernel search. Can be:
 - `"all"`: The complete set of available kernels: `{"LIN", "M12", "M32", "M52", "PER", "RBF", "RQF"}`.
 - `"restricted"`: The restricted set of kernels: `{"LIN", "M32", "M52", "PER", "RBF"}`.
 - A set of strings corresponding to the individual kernels to use for kernel selection, for example `{"RBF", "PER"}`.
 The default is `"restricted"`.
- **depth** (*int, optional*) – Specifies the number of base kernels allowed to be combined in the compositional kernel search. For example, a `depth=3` search means the resulting kernel may be composed from up-to three base kernels, so examples of allowed kernel combinations would be `"(LIN+PER)*RBF"` or `"(M12*RBF)+RQF"`. The default value is `1`, which simply compares all kernel functions individually.

- **beam** (*Union[[int](#), None], optional*) – Specifies the beam width of the compositional kernel search algorithm. This uses a beam search algorithm to find the best kernel combination. This is a heuristic search algorithm that explores a graph by expanding the most promising nodes in a limited set. A `beam=1` search is exhaustive, which is algorithmically ‘greedy’. `beam=None` instead performs a breadth-first search. `beam > 1` performs a beam search with the specified beam value. The default value is `None`.

```
__init__(seed=None, evaluation_metric='MSLL', val_ratio=0.2,  
base_kernels='restricted', depth=1, beam=None)
```

Methods

```
__init__([seed, evaluation_metric, ...])
```

```
unpack_parameters()
```

< Previous
[twinlab.EstimatorParams](#)

Next >
[twinlab.TrainParams](#)