

```
from itertools import product
import numpy as np
import pytest
from keras import regularizers, models, layers
from lib.model import normalization
from lib.utils import get_backend
from tests.lib.model.layers import test import layer_test
@pytest.mark.parametrize('dummy', [None], ids=[get_backend().upper()])...
@pytest.mark.parametrize('dummy', [None], ids=[get_backend().upper()])...
_PARAMS = ['center', 'scale']
_VALUES = list(product([True, False], repeat=len(_PARAMS)))
_IDS = ['{[{}}]'.format(''.join([_PARAMS[idx] for idx, b in enumerate(v) if
    b]), get_backend().upper()) for v in _VALUES]
@pytest.mark.parametrize(_PARAMS, _VALUES, ids=_IDS)...
@pytest.mark.parametrize(_PARAMS, _VALUES, ids=_IDS)...
_PARAMS = ['partial', 'bias']
_VALUES = [(0.0, False), (0.25, False), (0.5, True), (0.75, False), (1.0, True)]
_IDS = ['partial={}|bias={}|{...}'.format(v[0], v[1], get_backend().upper()) for
    v in _VALUES]
@pytest.mark.parametrize(_PARAMS, _VALUES, ids=_IDS)...
```

