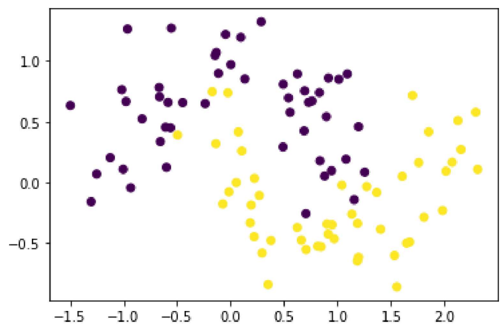
```
import numpy as np
import matplotlib.pyplot as plt
from sklearn.datasets import make_moons
import seaborn as sns
from mlxtend.plotting import plot_decision_regions

import tensorflow
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
from tensorflow.keras.layers import Dropout
from tensorflow.keras.optimizers import Adam
```

```
X, y = make_moons(100, noise=0.25,random_state=2)
```

```
import matplotlib.pyplot as plt
plt.scatter(X[:,0], X[:,1], c=y)
plt.show()
```



+ Code + Text

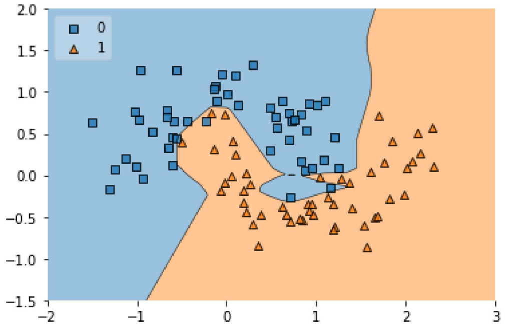
```
model1 = Sequential()

model1.add(Dense(128,input_dim=2, activation="relu"))
model1.add(Dense(128, activation="relu"))
model1.add(Dense(1,activation='sigmoid'))

model1.summary()
```

```
Model: "sequential_2"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_6 (Dense)             (None, 128)               384

 dense_7 (Dense)             (None, 128)               16512

 dense_8 (Dense)             (None, 1)                 129

=================================================================
Total params: 17,025
Trainable params: 17,025
Non-trainable params: 0
_________________________________________________________________
```

```
adam = Adam(learning_rate=0.01)
model1.compile(loss='binary_crossentropy', optimizer=adam, metrics=['accuracy'])

history1 = model1.fit(X, y, epochs=2000, validation_split = 0.2,verbose=0)
```

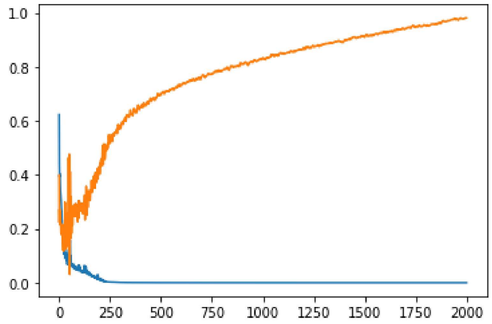
```
plot_decision_regions(X, y.astype('int'), clf=model1, legend=2)
plt.xlim(-2,3)
plt.ylim(-1.5,2)
plt.show()
```

```
/usr/local/lib/python3.7/dist-packages/mlxtend/plotting/decision_regions.py:244: Matplot
  ax.axis(xmin=xx.min(), xmax=xx.max(), y_min=yy.min(), y_max=yy.max())
```



```
plt.plot(history1.history['loss'])
plt.plot(history1.history['val_loss'])
```

```
[<matplotlib.lines.Line2D at 0x7f042d0f3210>]
```



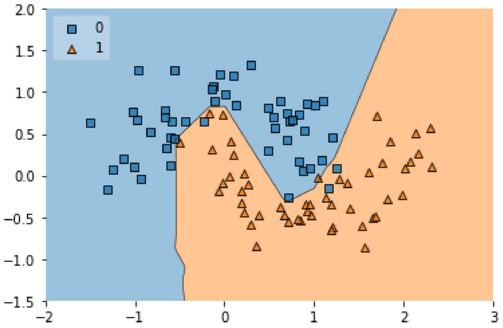
```
model2 = Sequential()

model2.add(Dense(128,input_dim=2, activation="relu",kernel_regularizer=tensorflow.keras.regularizers.l1(0.001)))
model2.add(Dense(128, activation="relu",kernel_regularizer=tensorflow.keras.regularizers.l1(0.001)))
model2.add(Dense(1,activation='sigmoid'))

model2.summary()
```

```
Model: "sequential_7"

 Layer (type)                Output Shape              Param #
=================================================================
 dense_21 (Dense)            (None, 128)               384

 dense_22 (Dense)            (None, 128)               16512

 dense_23 (Dense)            (None, 1)                 129

=================================================================
Total params: 17,025
Trainable params: 17,025
Non-trainable params: 0
```

```
adam = Adam(learning_rate=0.01)
model2.compile(loss='binary_crossentropy', optimizer=adam, metrics=['accuracy'])

history2 = model2.fit(X, y, epochs=2000, validation_split = 0.2,verbose=0)
```

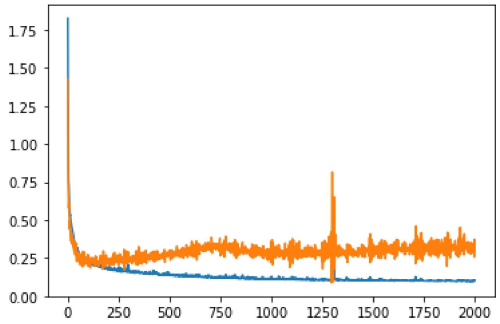
```
plot_decision_regions(X, y.astype('int'), clf=model2, legend=2)
plt.xlim(-2,3)
plt.ylim(-1.5,2)
plt.show()
```

```
/usr/local/lib/python3.7/dist-packages/mlxtend/plotting/decision_regions.py:244: Matplot
  ax.axis(xmin=xx.min(), xmax=xx.max(), y_min=yy.min(), y_max=yy.max())
```



```
plt.plot(history2.history['loss'])
plt.plot(history2.history['val_loss'])
```

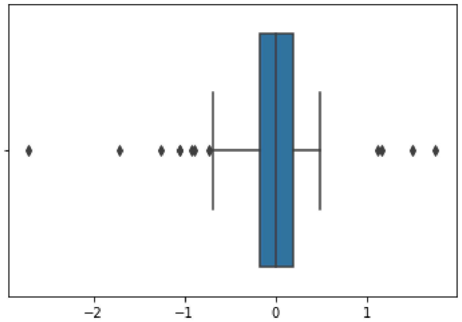
```
[<matplotlib.lines.Line2D at 0x7f042209ba10>]
```



```
model1_weight_layer1 = model1.get_weights()[0].reshape(256)
model2_weight_layer1 = model2.get_weights()[0].reshape(256)
```

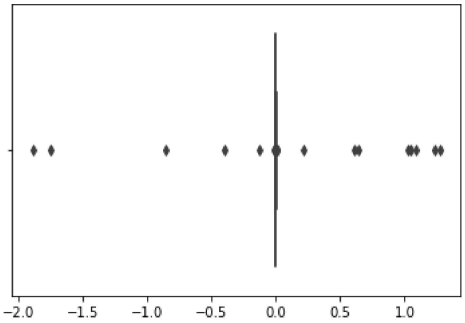
```
sns.boxplot(model1_weight_layer1)
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: Pass th
  FutureWarning
<matplotlib.axes._subplots.AxesSubplot at 0x7f04220b8b10>
```



```
sns.boxplot(model2_weight_layer1)
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: Pass th
  FutureWarning
<matplotlib.axes._subplots.AxesSubplot at 0x7f042202c390>
```



```
model1_weight_layer1.min()
```

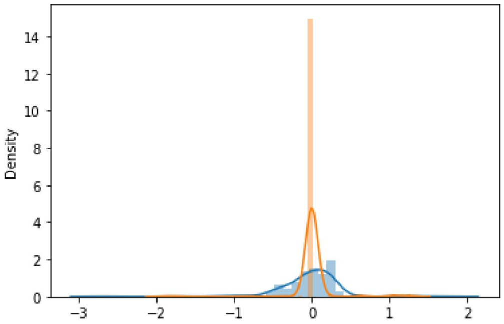
```
-2.7185578
```

```
model2_weight_layer1.min()
```

```
-1.8934479
```

```
sns.distplot(model1_weight_layer1)
sns.distplot(model2_weight_layer1)
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/distributions.py:2619: FutureWarning: `di
  warnings.warn(msg, FutureWarning)
/usr/local/lib/python3.7/dist-packages/seaborn/distributions.py:2619: FutureWarning: `di
  warnings.warn(msg, FutureWarning)
<matplotlib.axes._subplots.AxesSubplot at 0x7f0422018610>
```



```
model1.get_weights()[0].reshape(256)
```

```
array([ 8.36581886e-02,  3.62687036e-02, -4.17765856e-01, -3.93163204e-01,
       -3.96005869e-01,  2.13018522e-01,  2.28988439e-01, -3.22141826e-01,
        2.23829538e-01,  1.97904125e-01, -5.38365006e-01,  1.70224056e-01,
       -1.77298978e-01,  4.41261902e-02,  6.83471262e-02,  2.14902475e-01,
        4.95364517e-02,  8.23249444e-02, -5.02141356e-01, -1.56526896e-03,
        9.14724991e-02, -2.30933651e-02,  2.13529035e-01,  9.54458341e-02,
       -2.91170686e-01,  1.57895297e-01, -5.04917026e-01,  2.12428153e-01,
       -2.12982274e-03,  2.03352317e-01,  1.95124984e-01, -5.25625683e-02,
       -3.26485872e-01,  2.97886580e-01, -6.85776353e-01,  2.16803566e-01,
       -2.26653010e-01,  2.15190932e-01,  2.18419120e-01,  9.44057107e-02,
       -3.32020015e-01,  2.72148997e-01, -4.05541629e-01,  2.10304454e-01,
       -3.53123620e-02,  2.39017248e-01,  1.97818264e-01,  4.17221524e-02,
        2.47686148e-01,  2.60426432e-01,  7.42920935e-02, -1.97996497e-02,
        2.49996199e-03,  7.39424825e-02, -3.76863003e-01, -3.45605701e-01,
        2.64541715e-01,  1.32041499e-01,  2.02703670e-01, -5.18711984e-01,
       -1.18189253e-01,  1.92909852e-01,  6.19973317e-02,  1.69551060e-01,
        1.87287569e-01, -4.96824086e-01,  6.72358125e-02, -4.61391211e-01,
        2.53824741e-01, -4.05834556e-01,  1.26976356e-01, -5.35438359e-01,
        2.17906699e-01, -4.61450815e-01,  1.40626235e-02,  8.39711912e-03,
        1.33992940e-01,  2.37961203e-01, -3.23832154e-01,  2.20816463e-01,
        1.54505417e-01,  2.60821521e-01, -4.87969033e-02, -1.69064537e-01,
       -2.45658420e-02,  2.68627226e-01, -3.08435112e-02, -5.11454821e-01,
        1.59141541e-01, -3.70742589e-01,  7.88443834e-02,  4.04660106e-02,
        1.14555068e-01,  6.63752928e-02,  9.02682170e-03, -2.60784894e-01,
       -4.40374583e-01, -1.14047769e-02, -3.74770425e-02,  1.68620914e-01,
        1.78412095e-01,  3.17960113e-01, -1.22944184e-01,  2.22067803e-01,
 2.16269627e-01, -1.55968731e-02, -1.53711125e-01,  1.20375909e-01,
-2.82058775e-01, -4.21607405e-01, -2.38318786e-01, -3.87454391e-01,
-5.39283343e-02, -5.19582815e-02,  2.20146533e-02,  2.19618052e-01,
-3.93210024e-01,  5.31659797e-02,  1.20239288e-01, -1.07759438e-01,
 6.92361444e-02,  2.54981816e-01,  2.05702752e-01, -4.69989777e-01,
-2.02217568e-02,  2.18708590e-01, -3.57157469e-01, -4.57049876e-01,
 1.50024498e+00,  4.97291982e-01,  1.08143896e-01, -2.13980023e-02,
-3.68858159e-01, -1.55548185e-01, -2.20214128e-01, -3.67496133e-01,
 3.76471758e-01, -1.12438962e-01, -1.42594591e-01,  2.63052106e-01,
 2.86761671e-01, -1.75136283e-01,  1.42654404e-01, -1.64357230e-01,
-2.05702230e-01, -3.31739932e-01, -6.89846743e-03,  1.20316848e-01,
 2.39299849e-01,  2.02913046e-01, -1.23616964e-01,  1.96781605e-02,
 1.68995097e-01,  8.50129426e-02, -7.56526217e-02, -1.47868231e-01,
-1.22441379e-02,  1.24331288e-01,  8.93589929e-02, -3.18250328e-01,
 1.45956382e-01,  1.50910780e-01, -1.24292299e-01, -1.20857731e-01,
 3.02440643e-01, -1.56325176e-01, -8.48699063e-02,  1.12520742e+00,
 2.29321763e-01, -1.60620332e-01,  2.70279080e-01, -1.20651469e-01,
 2.99278051e-01,  4.02042896e-01, -1.20495699e-01, -2.19048098e-01,
-1.40762851e-01, -2.05702767e-01, -4.23372239e-02, -8.96272540e-01,
-1.72349977e+00, -3.21702920e-02, -4.25565869e-01,  1.55981630e-01,
-2.06218705e-01,  2.18112305e-01,  9.11024734e-02, -1.53775085e-02,
 2.21733779e-01, -1.11189082e-01, -5.94162345e-02, -7.17675462e-02,
-1.28930464e-01,  1.42400444e-04,  1.16128314e+00,  1.24267630e-01,
-1.96727023e-01,  2.12383211e-01,  3.87428761e-01, -1.30678445e-01,
 3.66538137e-01,  2.10566714e-01, -9.19768572e-01,  1.75965977e+00,
 2.43212238e-01, -1.89635113e-01,  1.17535554e-01, -1.66406915e-01,
```