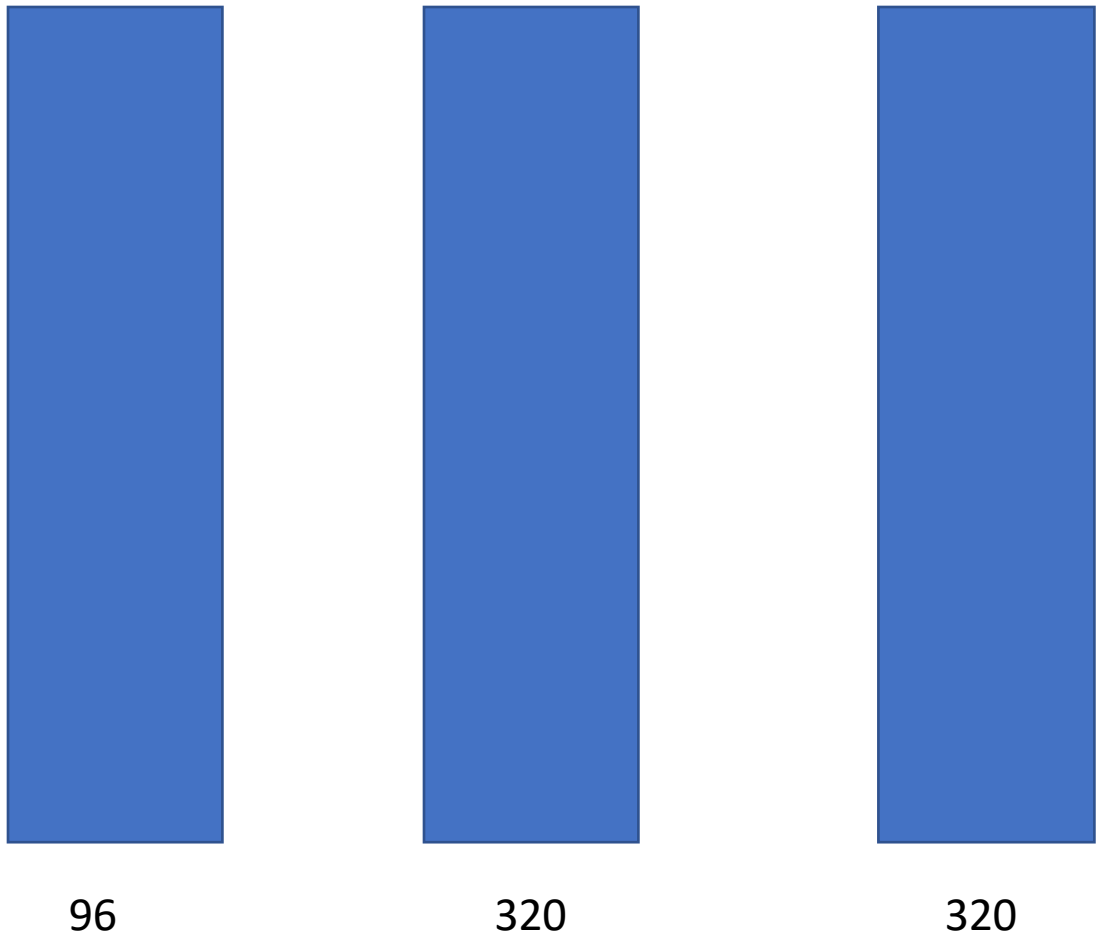

96
320
320

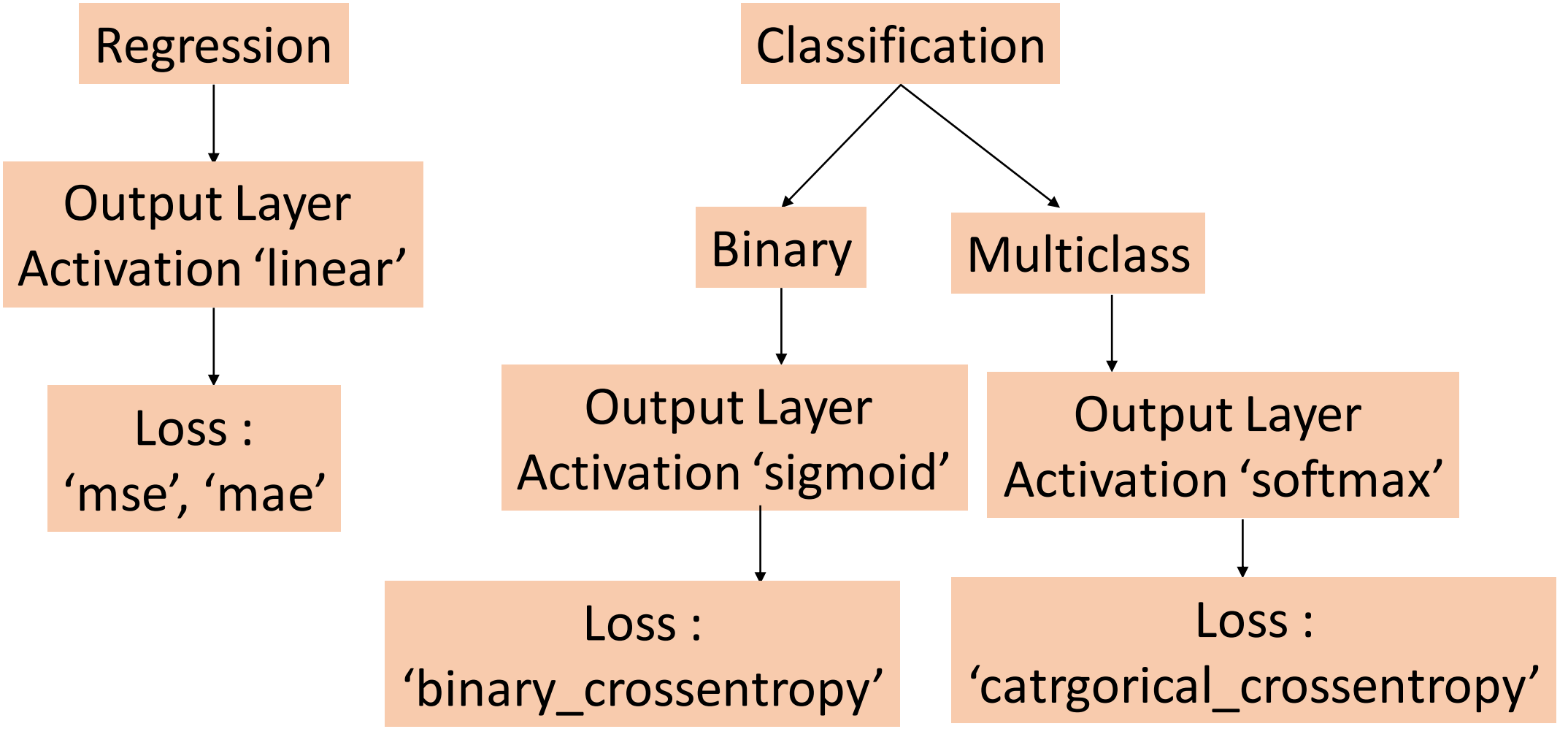
Will be back in 5 mins

- Create a model builder function

[16]:

```
def build_model(hp):
    nn = Sequential()

    hp_units1 = hp.Int('units1',min_value=32, max_value=512, step=32)
    hp_units2 = hp.Int('units2',min_value=32, max_value=512, step=32)
    hp_units3 = hp.Int('units3',min_value=32, max_value=512, step=32)

    nn.add(Dense(units=hp_units1,input_dim = xtrain.shape[1],activation='relu'))
    nn.add(Dense(units=hp_units2,activation='relu'))
    nn.add(Dense(units=hp_units3,activation='relu'))

    nn.add(Dense(units=1,activation='sigmoid'))

    nn.compile(loss='binary_crossentropy',optimizer='adam',metrics=['accuracy'])
    return nn
```

Summary


Regression
Output Layer
Activation ‘linear’
Loss :
‘mse’, ‘mae’
Classification
Binary
Multiclass
Output Layer
Activation ‘sigmoid’
Output Layer
Activation ‘softmax’
Loss :
‘binary_crossentropy’
Loss :
‘catrgorical_crossentropy’