

In [1]:

```
import pandas as pd
```

load the required columns to a dataframe

In [2]:

```
df = pd.read_csv("movie_metadata.csv", usecols=['num_critic_for_reviews', 'director_facebook_
```

Preprocess by dropping rows with any null values or zero values

In [3]:

```
df = df.dropna()  
df = df[(df.movie_facebook_likes != 0) & (df.director_facebook_likes != 0.0)  
        & (df.num_user_for_reviews != 0.0) & (df.num_critic_for_reviews != 0.0)  
        & (df.actor_1_facebook_likes != 0.0) & (df.imdb_score != 0.0)]
```

Split input and output

In [4]:

```
x=df[['num_critic_for_reviews', 'director_facebook_likes', 'actor_1_facebook_likes', 'movie_fa  
y=df['imdb_score']  
y=y.round()
```

Split as training data and test data

In [5]:

```
from sklearn.model_selection import train_test_split  
xTrain,xTest,yTrain,yTest=train_test_split(x,y,test_size=0.15)
```

In [6]:

```
from keras.models import Sequential  
from keras.layers import Dense
```

Using TensorFlow backend.

Build model and compile

In [7]:

```
model = Sequential()
model.add(Dense(5, input_dim=5, activation='relu'))
model.add(Dense(8, activation='relu'))
model.add(Dense(1, activation='sigmoid'))
```

In [8]:

```
model.compile(loss='mean_squared_error', optimizer='adam', metrics=['accuracy'])
```

Train model by fitting training data and validate by predicting for test data

In [9]:

```
import numpy as np
results = model.fit(xTrain, yTrain, epochs= 50, batch_size = 20, validation_data = (xTest, yTe
print("Test-Accuracy:", np.mean(results.history["val_acc"])))
```

Train on 1853 samples, validate on 328 samples

Epoch 1/50

1853/1853 [=====] - 0s 184us/step - loss: 41.5219
- acc: 0.0000e+00 - val_loss: 40.2530 - val_acc: 0.0000e+00

Epoch 2/50

1853/1853 [=====] - 0s 53us/step - loss: 41.5219
- acc: 0.0000e+00 - val_loss: 40.2530 - val_acc: 0.0000e+00

Epoch 3/50

1853/1853 [=====] - 0s 53us/step - loss: 41.5219
- acc: 0.0000e+00 - val_loss: 40.2530 - val_acc: 0.0000e+00

Epoch 4/50

1853/1853 [=====] - 0s 54us/step - loss: 41.5219
- acc: 0.0000e+00 - val_loss: 40.2530 - val_acc: 0.0000e+00

Epoch 5/50

1853/1853 [=====] - 0s 54us/step - loss: 41.5219
- acc: 0.0000e+00 - val_loss: 40.2530 - val_acc: 0.0000e+00

Epoch 6/50

1853/1853 [=====] - 0s 58us/step - loss: 41.5219
- acc: 0.0000e+00 - val loss: 40.2530 - val acc: 0.0000e+00