

Importing Library

```
In [1]: import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

```
In [2]: from keras.layers.core import Dense
from keras import Sequential
```

Importing Dataset

```
In [3]: df=pd.read_csv('concrete.csv')
df
```

Out[3]:

	cement	slag	ash	water	superplastic	coarseagg	fineagg	age	strength
0	141.3	212.0	0.0	203.5	0.0	971.8	748.5	28	29.89
1	168.9	42.2	124.3	158.3	10.8	1080.8	796.2	14	23.51
2	250.0	0.0	95.7	187.4	5.5	956.9	861.2	28	29.22
3	266.0	114.0	0.0	228.0	0.0	932.0	670.0	28	45.85
4	154.8	183.4	0.0	193.3	9.1	1047.4	696.7	28	18.29
...
1025	135.0	0.0	166.0	180.0	10.0	961.0	805.0	28	13.29
1026	531.3	0.0	0.0	141.8	28.2	852.1	893.7	3	41.30
1027	276.4	116.0	90.3	179.6	8.9	870.1	768.3	28	44.28
1028	342.0	38.0	0.0	228.0	0.0	932.0	670.0	270	55.06
1029	540.0	0.0	0.0	173.0	0.0	1125.0	613.0	7	52.61

1030 rows × 9 columns

```
In [4]: x=df.drop(['strength'],axis=1)
y=df['strength']
```

```
In [5]: x.shape
```

Out[5]: (1030, 8)

Performing Sequential model

```
In [6]: model=Sequential()  
model.add(Dense(12,input_dim=8,kernel_initializer='normal',activation='relu'))  
model.add(Dense(8,kernel_initializer='normal',activation='relu'))  
model.add(Dense(1,kernel_initializer='normal'))
```

```
In [7]: model.compile(loss='mean_squared_error',optimizer='adam')
```

Fitting The Model

```
In [8]: model.fit(x,y,epochs=100,batch_size=10)
```

```
Epoch 1/100  
103/103 [=====] - 0s 1ms/step - loss: 909.4157  
Epoch 2/100  
103/103 [=====] - 0s 1ms/step - loss: 247.3357  
Epoch 3/100  
103/103 [=====] - 0s 668us/step - loss: 223.2814  
Epoch 4/100  
103/103 [=====] - 0s 818us/step - loss: 181.0602  
Epoch 5/100  
103/103 [=====] - 0s 771us/step - loss: 148.0419  
Epoch 6/100  
103/103 [=====] - 0s 893us/step - loss: 131.0341  
Epoch 7/100  
103/103 [=====] - 0s 870us/step - loss: 120.7765  
Epoch 8/100  
103/103 [=====] - 0s 977us/step - loss: 117.5387  
Epoch 9/100  
103/103 [=====] - 0s 879us/step - loss: 114.7991  
Epoch 10/100  
103/103 [=====] - 0s 842us/step - loss: 114.1702
```

Predicting the model

```
In [9]: y_pred=model.predict(x)  
y_pred
```

```
Out[9]: array([[25.07229 ],  
               [21.619337],  
               [31.282114],  
               ...,  
               [45.798264],  
               [43.349617],  
               [48.476955]], dtype=float32)
```

```
In [10]: y.shape
```

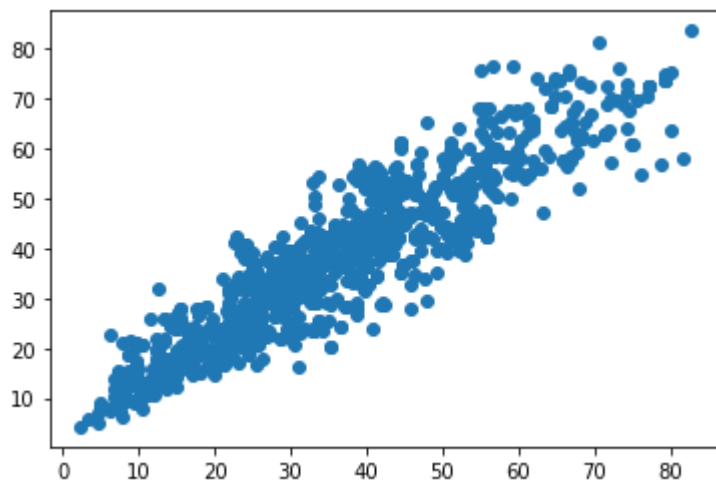
```
Out[10]: (1030,)
```

```
In [11]: y_pred.shape
```

```
Out[11]: (1030, 1)
```

```
In [12]: plt.scatter(y,y_pred)
```

```
Out[12]: <matplotlib.collections.PathCollection at 0x1a5a8c39280>
```



```
In [ ]:
```