deep_learning

January 15, 2023

0.1 Importation of librairies

[44]:
```
import pandas as pd
import numpy as np
import keras
from keras.models import Sequential
from keras.layers import Dense
import sklearn
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error
from sklearn.preprocessing import StandardScaler
```

0.2 Data

[45]:
```
concrete_data = pd.read_csv('https://s3-api.us-geo.objectstorage.softlayer.net/
↪cf-courses-data/CognitiveClass/DL0101EN/labs/data/concrete_data.csv')
concrete_data.head()
```

[45]:

	Cement	Blast Furnace Slag	Fly Ash	Water	Superplasticizer	Coarse Aggregate	Fine Aggregate	Age	Strength
0	540.0	0.0	0.0	162.0	2.5	1040.0	676.0	28	79.99
1	540.0	0.0	0.0	162.0	2.5	1055.0	676.0	28	61.89
2	332.5	142.5	0.0	228.0	0.0	932.0	594.0	270	40.27
3	332.5	142.5	0.0	228.0	0.0	932.0	594.0	365	41.05
4	198.6	132.4	0.0	192.0	0.0	978.4	825.5	360	44.30

0.3 Dimensions

[46]:
```
concrete_data.shape
```

[46]: (1030, 9)

[47]:
```
concrete_data.describe()
```

[47]:

	Cement	Blast Furnace Slag	Fly Ash	Water
count	1030.000000	1030.000000	1030.000000	1030.000000
mean	281.167864	73.895825	54.188350	181.567282
std	104.506364	86.279342	63.997004	21.354219
min	102.000000	0.000000	0.000000	121.800000
25%	192.375000	0.000000	0.000000	164.900000
50%	272.900000	22.000000	0.000000	185.000000
75%	350.000000	142.950000	118.300000	192.000000
max	540.000000	359.400000	200.100000	247.000000

	Superplasticizer	Coarse Aggregate	Fine Aggregate	Age
count	1030.000000	1030.000000	1030.000000	1030.000000
mean	6.204660	972.918932	773.580485	45.662136
std	5.973841	77.753954	80.175980	63.169912
min	0.000000	801.000000	594.000000	1.000000
25%	0.000000	932.000000	730.950000	7.000000
50%	6.400000	968.000000	779.500000	28.000000
75%	10.200000	1029.400000	824.000000	56.000000
max	32.200000	1145.000000	992.600000	365.000000

	Strength
count	1030.000000
mean	35.817961
std	16.705742
min	2.330000
25%	23.710000
50%	34.445000
75%	46.135000
max	82.600000

0.4 Definition of predictor and label

[48]:
```
concrete_data_columns = concrete_data.columns

predictors = concrete_data[concrete_data_columns[concrete_data_columns !=␣
  ↪'Strength']] # all columns except Strength
target = concrete_data['Strength'] # Strength column
```

[49]:
```
# Consulatation de target
target.head()
```

[49]:
```
0    79.99
1    61.89
2    40.27
3    41.05
```

```
4    44.30
Name: Strength, dtype: float64
```

[50]:
```
# Consultation of predictor
predictors.head()
```

[50]:
```
   Cement  Blast Furnace Slag  Fly Ash  Water  Superplasticizer  \
0   540.0                 0.0      0.0  162.0               2.5
1   540.0                 0.0      0.0  162.0               2.5
2   332.5               142.5      0.0  228.0               0.0
3   332.5               142.5      0.0  228.0               0.0
4   198.6               132.4      0.0  192.0               0.0

   Coarse Aggregate  Fine Aggregate  Age
0            1040.0           676.0   28
1            1055.0           676.0   28
2             932.0           594.0  270
3             932.0           594.0  365
4             978.4           825.5  360
```

[51]:
```
scaler = StandardScaler()
predictors = scaler.fit_transform(predictors)
```

```
/home/jupyterlab/conda/envs/python/lib/python3.7/site-
packages/sklearn/preprocessing/data.py:625: DataConversionWarning: Data with
input dtype int64, float64 were all converted to float64 by StandardScaler.
  return self.partial_fit(X, y)
/home/jupyterlab/conda/envs/python/lib/python3.7/site-
packages/sklearn/base.py:462: DataConversionWarning: Data with input dtype
int64, float64 were all converted to float64 by StandardScaler.
  return self.fit(X, **fit_params).transform(X)
```

0.5 Devision of Data Set

[52]:
```
x_train, x_test, y_train, y_test = train_test_split(
...     predictors, target, test_size=0.3, random_state=4
... )
```

[53]:
```
# Analyse of shape
x_test.shape
```

[53]:
```
(309, 8)
```

[54]:
```
x_train.shape
```

[54]:
```
(721, 8)
```

0.6 Model

[63]:
```
# define regression model
def regression_model():
    # create model
    model = Sequential()
    model.add(Dense(50, activation='relu', input_shape=(8,)))
    model.add(Dense(10, activation='relu'))
    model.add(Dense(10, activation='relu'))
    model.add(Dense(10, activation='relu'))
    model.add(Dense(1))

    # compile model
    model.compile(optimizer='adam', loss='mean_squared_error')
    return model
```

[64]:
```
# build the model
model = regression_model()
```

[65]:
```
# fit the model
model.fit(predictors, target, epochs=100)
```

```
Epoch 1/100
1030/1030 [==============================] - 0s 478us/step - loss: 1506.9586
Epoch 2/100
1030/1030 [==============================] - 0s 159us/step - loss: 1317.4613
Epoch 3/100
1030/1030 [==============================] - 0s 167us/step - loss: 818.3439
Epoch 4/100
1030/1030 [==============================] - 0s 154us/step - loss: 294.5958
Epoch 5/100
1030/1030 [==============================] - 0s 137us/step - loss: 204.8063
Epoch 6/100
1030/1030 [==============================] - 0s 134us/step - loss: 186.8280
Epoch 7/100
1030/1030 [==============================] - 0s 121us/step - loss: 175.9045
Epoch 8/100
1030/1030 [==============================] - 0s 129us/step - loss: 168.5591
Epoch 9/100
1030/1030 [==============================] - 0s 137us/step - loss: 159.8558
Epoch 10/100
1030/1030 [==============================] - 0s 136us/step - loss: 153.1933
Epoch 11/100
1030/1030 [==============================] - 0s 132us/step - loss: 147.4898
Epoch 12/100
1030/1030 [==============================] - 0s 116us/step - loss: 142.1067
Epoch 13/100
1030/1030 [==============================] - 0s 123us/step - loss: 136.8375
```

```
Epoch 14/100
1030/1030 [==============================] - 0s 132us/step - loss: 131.2220
Epoch 15/100
1030/1030 [==============================] - 0s 119us/step - loss: 125.4504
Epoch 16/100
1030/1030 [==============================] - 0s 122us/step - loss: 119.1211
Epoch 17/100
1030/1030 [==============================] - 0s 130us/step - loss: 113.5035
Epoch 18/100
1030/1030 [==============================] - 0s 120us/step - loss: 109.4906
Epoch 19/100
1030/1030 [==============================] - 0s 131us/step - loss: 101.6363
Epoch 20/100
1030/1030 [==============================] - 0s 116us/step - loss: 96.2569
Epoch 21/100
1030/1030 [==============================] - 0s 119us/step - loss: 91.0926
Epoch 22/100
1030/1030 [==============================] - 0s 131us/step - loss: 85.1668
Epoch 23/100
1030/1030 [==============================] - 0s 120us/step - loss: 80.8972
Epoch 24/100
1030/1030 [==============================] - 0s 133us/step - loss: 76.2222
Epoch 25/100
1030/1030 [==============================] - 0s 120us/step - loss: 71.8755
Epoch 26/100
1030/1030 [==============================] - 0s 122us/step - loss: 67.1222
Epoch 27/100
1030/1030 [==============================] - 0s 134us/step - loss: 63.8018
Epoch 28/100
1030/1030 [==============================] - 0s 116us/step - loss: 60.4808
Epoch 29/100
1030/1030 [==============================] - 0s 116us/step - loss: 57.8803
Epoch 30/100
1030/1030 [==============================] - 0s 117us/step - loss: 55.3647
Epoch 31/100
1030/1030 [==============================] - 0s 176us/step - loss: 52.9635
Epoch 32/100
1030/1030 [==============================] - 0s 117us/step - loss: 51.1304
Epoch 33/100
1030/1030 [==============================] - 0s 116us/step - loss: 49.8438
Epoch 34/100
1030/1030 [==============================] - 0s 118us/step - loss: 48.4597
Epoch 35/100
1030/1030 [==============================] - 0s 134us/step - loss: 46.8676
Epoch 36/100
1030/1030 [==============================] - 0s 114us/step - loss: 46.4301
Epoch 37/100
1030/1030 [==============================] - 0s 116us/step - loss: 44.5156
```

```
Epoch 38/100
1030/1030 [==============================] - 0s 118us/step - loss: 43.6932
Epoch 39/100
1030/1030 [==============================] - 0s 116us/step - loss: 42.6657
Epoch 40/100
1030/1030 [==============================] - 0s 98us/step - loss: 41.4997
Epoch 41/100
1030/1030 [==============================] - 0s 114us/step - loss: 41.5087
Epoch 42/100
1030/1030 [==============================] - 0s 101us/step - loss: 40.5616
Epoch 43/100
1030/1030 [==============================] - 0s 115us/step - loss: 40.8780
Epoch 44/100
1030/1030 [==============================] - 0s 113us/step - loss: 39.5471
Epoch 45/100
1030/1030 [==============================] - 0s 117us/step - loss: 38.1386
Epoch 46/100
1030/1030 [==============================] - 0s 118us/step - loss: 37.5534
Epoch 47/100
1030/1030 [==============================] - 0s 116us/step - loss: 37.3047
Epoch 48/100
1030/1030 [==============================] - 0s 130us/step - loss: 37.3124
Epoch 49/100
1030/1030 [==============================] - 0s 100us/step - loss: 36.3035
Epoch 50/100
1030/1030 [==============================] - 0s 114us/step - loss: 35.8925
Epoch 51/100
1030/1030 [==============================] - 0s 115us/step - loss: 35.5018
Epoch 52/100
1030/1030 [==============================] - 0s 113us/step - loss: 35.5047
Epoch 53/100
1030/1030 [==============================] - 0s 115us/step - loss: 34.8541
Epoch 54/100
1030/1030 [==============================] - 0s 118us/step - loss: 34.1871
Epoch 55/100
1030/1030 [==============================] - 0s 96us/step - loss: 33.8145
Epoch 56/100
1030/1030 [==============================] - 0s 117us/step - loss: 33.4878
Epoch 57/100
1030/1030 [==============================] - 0s 112us/step - loss: 33.5086
Epoch 58/100
1030/1030 [==============================] - 0s 104us/step - loss: 33.0225
Epoch 59/100
1030/1030 [==============================] - 0s 114us/step - loss: 32.7794
Epoch 60/100
1030/1030 [==============================] - 0s 113us/step - loss: 32.0736
Epoch 61/100
1030/1030 [==============================] - 0s 101us/step - loss: 31.3726
```

```
Epoch 62/100
1030/1030 [==============================] - 0s 95us/step - loss: 31.1453
Epoch 63/100
1030/1030 [==============================] - 0s 112us/step - loss: 31.6485
Epoch 64/100
1030/1030 [==============================] - 0s 117us/step - loss: 30.9405
Epoch 65/100
1030/1030 [==============================] - 0s 119us/step - loss: 31.2059
Epoch 66/100
1030/1030 [==============================] - 0s 127us/step - loss: 30.9042
Epoch 67/100
1030/1030 [==============================] - 0s 104us/step - loss: 30.6113
Epoch 68/100
1030/1030 [==============================] - 0s 115us/step - loss: 30.5531
Epoch 69/100
1030/1030 [==============================] - 0s 112us/step - loss: 29.6071
Epoch 70/100
1030/1030 [==============================] - 0s 114us/step - loss: 29.3359
Epoch 71/100
1030/1030 [==============================] - 0s 115us/step - loss: 29.1102
Epoch 72/100
1030/1030 [==============================] - 0s 117us/step - loss: 28.7129
Epoch 73/100
1030/1030 [==============================] - 0s 119us/step - loss: 28.2586
Epoch 74/100
1030/1030 [==============================] - 0s 113us/step - loss: 27.8329
Epoch 75/100
1030/1030 [==============================] - 0s 118us/step - loss: 27.7430
Epoch 76/100
1030/1030 [==============================] - 0s 115us/step - loss: 27.6281
Epoch 77/100
1030/1030 [==============================] - 0s 115us/step - loss: 27.1796
Epoch 78/100
1030/1030 [==============================] - 0s 115us/step - loss: 27.5924
Epoch 79/100
1030/1030 [==============================] - 0s 118us/step - loss: 27.4977
Epoch 80/100
1030/1030 [==============================] - 0s 115us/step - loss: 26.6872
Epoch 81/100
1030/1030 [==============================] - 0s 135us/step - loss: 26.8352
Epoch 82/100
1030/1030 [==============================] - 0s 132us/step - loss: 26.8624
Epoch 83/100
1030/1030 [==============================] - 0s 117us/step - loss: 27.0052
Epoch 84/100
1030/1030 [==============================] - 0s 131us/step - loss: 25.6046
Epoch 85/100
1030/1030 [==============================] - 0s 104us/step - loss: 25.6527
```

```
Epoch 86/100
1030/1030 [==============================] - 0s 103us/step - loss: 25.4325
Epoch 87/100
1030/1030 [==============================] - 0s 114us/step - loss: 24.8010
Epoch 88/100
1030/1030 [==============================] - 0s 116us/step - loss: 24.5155
Epoch 89/100
1030/1030 [==============================] - 0s 112us/step - loss: 24.4991
Epoch 90/100
1030/1030 [==============================] - 0s 116us/step - loss: 24.4923
Epoch 91/100
1030/1030 [==============================] - 0s 102us/step - loss: 24.5743
Epoch 92/100
1030/1030 [==============================] - 0s 115us/step - loss: 23.6388
Epoch 93/100
1030/1030 [==============================] - 0s 113us/step - loss: 24.9230
Epoch 94/100
1030/1030 [==============================] - 0s 113us/step - loss: 24.0836
Epoch 95/100
1030/1030 [==============================] - 0s 103us/step - loss: 23.2328
Epoch 96/100
1030/1030 [==============================] - 0s 117us/step - loss: 22.9001
Epoch 97/100
1030/1030 [==============================] - 0s 112us/step - loss: 23.4743
Epoch 98/100
1030/1030 [==============================] - 0s 120us/step - loss: 22.9035
Epoch 99/100
1030/1030 [==============================] - 0s 104us/step - loss: 23.1486
Epoch 100/100
1030/1030 [==============================] - 0s 115us/step - loss: 22.9298
```

[65]:
```
<keras.callbacks.History at 0x7ff24c55afd0>
```

0.7 Evaluation

[66]:
```
y_predic = model.predict(x_test)
mean_squared_error(y_test, y_predic)
```

[66]:
```
25.284352040678694
```

0.8 Repeatition

[68]:
```
# fit the model
model.fit(predictors, target, validation_split=0.3, epochs=100, verbose=2)
```

```
Train on 721 samples, validate on 309 samples
Epoch 1/100
```

```
 - 0s - loss: 13.4256 - val_loss: 75.7776
Epoch 2/100
 - 0s - loss: 13.1097 - val_loss: 64.6520
Epoch 3/100
 - 0s - loss: 13.2168 - val_loss: 74.2051
Epoch 4/100
 - 0s - loss: 13.0523 - val_loss: 73.2361
Epoch 5/100
 - 0s - loss: 13.0915 - val_loss: 64.3363
Epoch 6/100
 - 0s - loss: 13.6648 - val_loss: 61.5256
Epoch 7/100
 - 0s - loss: 13.0598 - val_loss: 67.5709
Epoch 8/100
 - 0s - loss: 13.1417 - val_loss: 68.3128
Epoch 9/100
 - 0s - loss: 13.0047 - val_loss: 68.6848
Epoch 10/100
 - 0s - loss: 12.8592 - val_loss: 70.6955
Epoch 11/100
 - 0s - loss: 13.3324 - val_loss: 70.7432
Epoch 12/100
 - 0s - loss: 13.4433 - val_loss: 77.4948
Epoch 13/100
 - 0s - loss: 13.5469 - val_loss: 76.9288
Epoch 14/100
 - 0s - loss: 13.4124 - val_loss: 67.4824
Epoch 15/100
 - 0s - loss: 13.2812 - val_loss: 67.2256
Epoch 16/100
 - 0s - loss: 13.0535 - val_loss: 77.5228
Epoch 17/100
 - 0s - loss: 12.6363 - val_loss: 69.5713
Epoch 18/100
 - 0s - loss: 13.1752 - val_loss: 74.5311
Epoch 19/100
 - 0s - loss: 12.9473 - val_loss: 65.5585
Epoch 20/100
 - 0s - loss: 12.6338 - val_loss: 67.8161
Epoch 21/100
 - 0s - loss: 13.0620 - val_loss: 72.1384
Epoch 22/100
 - 0s - loss: 13.0533 - val_loss: 73.9381
Epoch 23/100
 - 0s - loss: 13.0698 - val_loss: 64.6708
Epoch 24/100
 - 0s - loss: 12.7081 - val_loss: 67.1629
Epoch 25/100
```

```
 - 0s - loss: 12.4111 - val_loss: 77.0453
Epoch 26/100
 - 0s - loss: 12.7837 - val_loss: 73.6234
Epoch 27/100
 - 0s - loss: 13.4002 - val_loss: 65.2097
Epoch 28/100
 - 0s - loss: 12.7478 - val_loss: 78.6409
Epoch 29/100
 - 0s - loss: 12.2270 - val_loss: 67.4903
Epoch 30/100
 - 0s - loss: 12.9385 - val_loss: 77.7052
Epoch 31/100
 - 0s - loss: 12.6273 - val_loss: 74.2342
Epoch 32/100
 - 0s - loss: 12.1415 - val_loss: 73.6336
Epoch 33/100
 - 0s - loss: 12.2631 - val_loss: 64.7755
Epoch 34/100
 - 0s - loss: 12.7238 - val_loss: 69.9865
Epoch 35/100
 - 0s - loss: 12.4870 - val_loss: 75.1600
Epoch 36/100
 - 0s - loss: 12.4510 - val_loss: 71.9675
Epoch 37/100
 - 0s - loss: 12.6895 - val_loss: 77.5258
Epoch 38/100
 - 0s - loss: 12.1253 - val_loss: 72.3039
Epoch 39/100
 - 0s - loss: 12.5205 - val_loss: 72.8348
Epoch 40/100
 - 0s - loss: 12.3786 - val_loss: 77.2941
Epoch 41/100
 - 0s - loss: 12.1106 - val_loss: 85.5619
Epoch 42/100
 - 0s - loss: 12.4478 - val_loss: 83.5360
Epoch 43/100
 - 0s - loss: 13.0858 - val_loss: 65.2775
Epoch 44/100
 - 0s - loss: 12.6480 - val_loss: 93.1334
Epoch 45/100
 - 0s - loss: 12.1044 - val_loss: 74.9671
Epoch 46/100
 - 0s - loss: 11.9075 - val_loss: 71.8757
Epoch 47/100
 - 0s - loss: 12.0686 - val_loss: 84.1292
Epoch 48/100
 - 0s - loss: 12.7972 - val_loss: 72.7567
Epoch 49/100
```

```
 - 0s - loss: 12.2865 - val_loss: 83.1297
Epoch 50/100
 - 0s - loss: 12.3349 - val_loss: 81.9839
Epoch 51/100
 - 0s - loss: 12.3228 - val_loss: 75.9242
Epoch 52/100
 - 0s - loss: 12.0252 - val_loss: 82.0757
Epoch 53/100
 - 0s - loss: 11.8106 - val_loss: 73.3921
Epoch 54/100
 - 0s - loss: 12.2628 - val_loss: 79.1296
Epoch 55/100
 - 0s - loss: 12.7029 - val_loss: 72.0290
Epoch 56/100
 - 0s - loss: 11.8563 - val_loss: 84.5778
Epoch 57/100
 - 0s - loss: 12.0009 - val_loss: 77.3357
Epoch 58/100
 - 0s - loss: 11.5609 - val_loss: 73.5120
Epoch 59/100
 - 0s - loss: 11.8272 - val_loss: 72.7173
Epoch 60/100
 - 0s - loss: 11.8179 - val_loss: 86.4883
Epoch 61/100
 - 0s - loss: 12.1586 - val_loss: 76.7799
Epoch 62/100
 - 0s - loss: 11.7825 - val_loss: 78.0443
Epoch 63/100
 - 0s - loss: 11.6812 - val_loss: 83.5384
Epoch 64/100
 - 0s - loss: 11.6802 - val_loss: 80.8589
Epoch 65/100
 - 0s - loss: 11.6783 - val_loss: 70.6855
Epoch 66/100
 - 0s - loss: 11.6940 - val_loss: 82.7418
Epoch 67/100
 - 0s - loss: 11.4833 - val_loss: 71.2330
Epoch 68/100
 - 0s - loss: 11.4452 - val_loss: 80.2679
Epoch 69/100
 - 0s - loss: 11.6612 - val_loss: 75.3906
Epoch 70/100
 - 0s - loss: 11.5096 - val_loss: 77.3468
Epoch 71/100
 - 0s - loss: 11.6315 - val_loss: 83.8992
Epoch 72/100
 - 0s - loss: 11.4088 - val_loss: 73.8027
Epoch 73/100
```

```
 - 0s - loss: 12.1336 - val_loss: 75.2958
Epoch 74/100
 - 0s - loss: 12.2297 - val_loss: 88.0799
Epoch 75/100
 - 0s - loss: 12.1359 - val_loss: 78.0346
Epoch 76/100
 - 0s - loss: 11.9203 - val_loss: 69.2402
Epoch 77/100
 - 0s - loss: 12.2767 - val_loss: 87.9652
Epoch 78/100
 - 0s - loss: 11.3304 - val_loss: 81.9367
Epoch 79/100
 - 0s - loss: 11.4455 - val_loss: 78.5691
Epoch 80/100
 - 0s - loss: 11.1248 - val_loss: 82.4798
Epoch 81/100
 - 0s - loss: 11.6202 - val_loss: 78.4367
Epoch 82/100
 - 0s - loss: 11.3225 - val_loss: 75.6113
Epoch 83/100
 - 0s - loss: 11.3507 - val_loss: 77.1128
Epoch 84/100
 - 0s - loss: 12.2211 - val_loss: 85.5213
Epoch 85/100
 - 0s - loss: 11.2889 - val_loss: 75.4534
Epoch 86/100
 - 0s - loss: 11.1750 - val_loss: 75.1505
Epoch 87/100
 - 0s - loss: 11.7446 - val_loss: 82.7328
Epoch 88/100
 - 0s - loss: 11.5119 - val_loss: 78.8066
Epoch 89/100
 - 0s - loss: 11.5574 - val_loss: 86.3327
Epoch 90/100
 - 0s - loss: 11.7480 - val_loss: 76.7042
Epoch 91/100
 - 0s - loss: 11.4734 - val_loss: 78.0112
Epoch 92/100
 - 0s - loss: 11.0857 - val_loss: 88.3520
Epoch 93/100
 - 0s - loss: 12.0690 - val_loss: 77.2319
Epoch 94/100
 - 0s - loss: 11.2455 - val_loss: 89.1284
Epoch 95/100
 - 0s - loss: 12.0787 - val_loss: 86.6433
Epoch 96/100
 - 0s - loss: 11.6268 - val_loss: 82.5307
Epoch 97/100
```

```
 - 0s - loss: 10.9542 - val_loss: 87.8015
Epoch 98/100
 - 0s - loss: 11.0496 - val_loss: 90.6165
Epoch 99/100
 - 0s - loss: 11.8841 - val_loss: 78.5144
Epoch 100/100
 - 0s - loss: 10.8533 - val_loss: 84.6049
```

[68]: <keras.callbacks.History at 0x7ff21c5c62d0>

[ ]:

[ ]: