Analysis of 'concrete' Data and finding the relation between 'strength' and other variables

In [1]:
```
#importing libraries
import pandas as pd
import numpy as np
import matplotlib
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
```

In [6]:
```
df=pd.read_csv("https://raw.githubusercontent.com/Mukund94/Datasets/main/concrete.csv"
```

In [7]:
```
df.shape
```

Out[7]:
```
(1030, 9)
```

In [8]:
```
df.isnull().sum()
```

Out[8]:
```
cement          0
slag            0
ash             0
water           0
superplastic    0
coarseagg       0
fineagg         0
age             0
strength        0
dtype: int64
```

In [10]:
```
df.dtypes
```

Out[10]:
```
cement          float64
slag            float64
ash             float64
water           float64
superplastic    float64
coarseagg       float64
fineagg         float64
age               int64
strength        float64
dtype: object
```

In [13]:
```
df.describe(include="all")
```

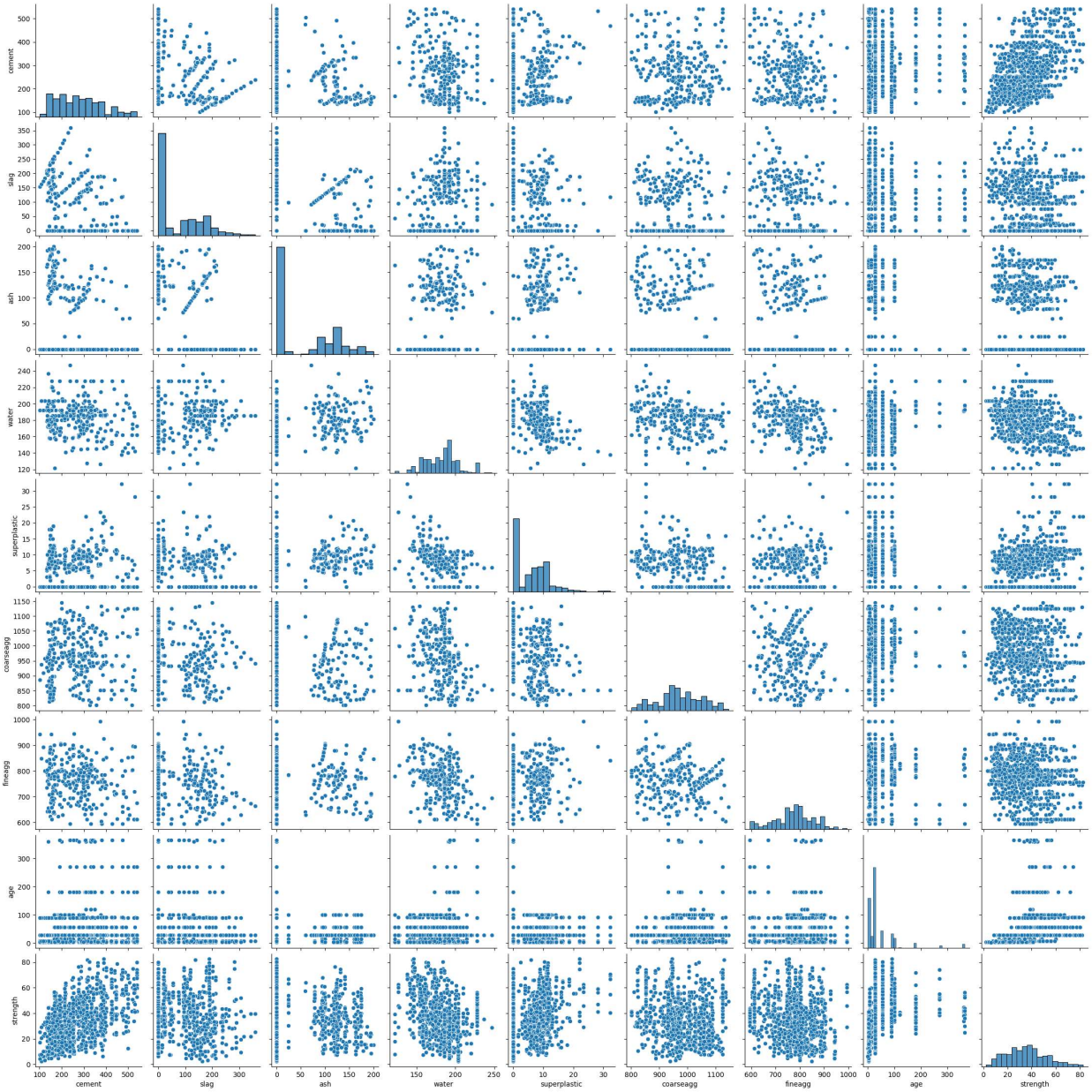
Out[13]:

	cement	slag	ash	water	superplastic	coarseagg	fineagg	
count	1030.000000	1030.000000	1030.000000	1030.000000	1030.000000	1030.000000	1030.000000	10
mean	281.167864	73.895825	54.188350	181.567282	6.204660	972.918932	773.580485	
std	104.506364	86.279342	63.997004	21.354219	5.973841	77.753954	80.175980	
min	102.000000	0.000000	0.000000	121.800000	0.000000	801.000000	594.000000	
25%	192.375000	0.000000	0.000000	164.900000	0.000000	932.000000	730.950000	
50%	272.900000	22.000000	0.000000	185.000000	6.400000	968.000000	779.500000	
75%	350.000000	142.950000	118.300000	192.000000	10.200000	1029.400000	824.000000	
max	540.000000	359.400000	200.100000	247.000000	32.200000	1145.000000	992.600000	3

In [14]:
```
sns.pairplot(df)
```

```
/opt/conda/lib/python3.10/site-packages/seaborn/axisgrid.py:118: UserWarning: The fig
ure layout has changed to tight
  self._figure.tight_layout(*args, **kwargs)
```

Out[14]:
```
<seaborn.axisgrid.PairGrid at 0x79fc4325ffd0>
```



In [18]:
```
X=df.drop(['strength'],axis=1)
Y=df[['strength']]
```

In [19]:
```
X_Train,X_Test,Y_Train,Y_Test=train_test_split(X,Y,test_size=0.2)
```

In [20]:
```
Model1=LinearRegression()
```

In [21]:
```
Model1.fit(X_Train,Y_Train)
```

Out[21]:
```
▾ LinearRegression
LinearRegression()
```

In [23]:
```
Model1.score(X_Train,Y_Train)
```

Out[23]:
```
0.6175704286565591
```

In [24]:
```
Model1.score(X_Test,Y_Test)
```

Out[24]:
```
0.5998363616297944
```

In [ ]:
```

```