Architecture Design

CONCRETE COMPRESSIVE STRENGTH PREDICTION

Document Control

Version	Date	Author	Comments
1	03/07/2023	Shailendra Sahu	

Index

Content	Page No
Abstract	4
1. Introduction	4
1.1 What is Architecture Design?	4
1.2 Scope	4
1.3 Constraints	4
2. Technical Specification	5
2.1 Dataset	5
2.2 Logging	6
2.3 Deployment	6
3. Technology Stack	7
4. Proposed Solution	7
5. Architecture	7
5.1 Architecture Description	8
6. User Input/Output Workflow	9

Abstract

Machine Learning is a category of algorithms that allows software applications to become more accurate in predicting outcomes without being explicitly programmed. The basic premise of machine learning is to build models and employ algorithms that can receive input data and use statistical analysis to predict an output while updating outputs as new data becomes available. These models can be applied in different areas and trained to match the expectations of management so that accurate steps can be taken to achieve the organization's target. In this project, we will estimate the compressive strength of concrete on the basis of the various informations provided to us. Taking various aspects of a dataset collected from client, and the methodology followed for building a predictive model.

1. Introduction

1.1 What is Architecture Design?

The goal of Architecture Design (AD) is to give the internal design of the actual program code for the ` Concrete Compressive Strength Prediction`. AD describes the class diagrams with the methods and relation between classes and program specification. It describes the modules so that the programmer can directly code the program from the document.

1.2 Scope

Architecture Design (AD) is a component-level design process that follows a step-by-step refinement process. This process can be used for designing data structures, required software, architecture, source code, and ultimately, performance algorithms. Overall, the data organization may be defined during requirement analysis and then refined during data design work. And the complete workflow.

1.3 Constraints

We only predict the Compressive Strength of Concrete based on the information provided by client.

2. Technical Specification

2.1 Dataset

The dataset containing verified data, consisting of the informations about contents and things mixed during the concrete making process. The objective is to find a way to estimate the value in the "Concrete_compressive_strength" column using the values in the other columns like Cement, Blast Furnace Slag, Fly Ash, Water, Superplasticizer, Coarse Aggregate, Fine Aggregate and Age_day.

The dataset looks like as follow:

In [3]:
```
df
```

Out[3]:

	Cement (component 1)(kg in a m^3 mixture)	Blast Furnace Slag (component 2)(kg in a m^3 mixture)	Fly Ash (component 3)(kg in a m^3 mixture)	Coarse Aggregate (component 6)(kg in a m^3 mixture)	Fine Aggregate (component 7)(kg in a m^3 mixture)	Age (day)	Concrete compressive strength(MPa, megapascals)
0	540.0	0.0	0.0	1040.0	676.0	28	79.99
1	540.0	0.0	0.0	1055.0	676.0	28	61.89
2	332.5	142.5	0.0	932.0	594.0	270	40.27
3	332.5	142.5	0.0	932.0	594.0	365	41.05
4	198.6	132.4	0.0	978.4	825.5	360	44.30
...	...	...	...	...	...	...	...
1025	276.4	116.0	90.3	870.1	768.3	28	44.28
1026	322.2	0.0	115.6	817.9	813.4	28	31.18
1027	148.5	139.4	108.6	892.4	780.0	28	23.70
1028	159.1	186.7	0.0	989.6	788.9	28	32.77
1029	260.9	100.5	78.3	864.5	761.5	28	32.40

1030 rows × 7 columns

The data set consists of various data types from integer to floating as shown in Fig.

In [4]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1030 entries, 0 to 1029
Data columns (total 7 columns):
 #   Column                                              Non-Null Count  Dtype
---  ------                                              --------------  -----
 0   Cement (component 1)(kg in a m^3 mixture)           1030 non-null   float64
 1   Blast Furnace Slag (component 2)(kg in a m^3 mixture)  1030 non-null   float64
 2   Fly Ash (component 3)(kg in a m^3 mixture)          1030 non-null   float64
 3   Coarse Aggregate  (component 6)(kg in a m^3 mixture)   1030 non-null   float64
 4   Fine Aggregate (component 7)(kg in a m^3 mixture)   1030 non-null   float64
 5   Age (day)                                           1030 non-null   int64
 6   Concrete compressive strength(MPa, megapascals)     1030 non-null   float64
dtypes: float64(6), int64(1)
memory usage: 56.5 KB
```

Architecture Design

Various factors important by statistical means like mean, standard deviation, median, count of values and maximum value, etc. are shown below for numerical attributes.

```
In [5]: df.describescribe()
```

Out[5]:

	Cement (component 1)(kg in a m^3 mixture)	Blast Furnace Slag (component 2)(kg in a m^3 mixture)	Fly Ash (component 3)(kg in a m^3 mixture)	Coarse Aggregate (component 6)(kg in a m^3 mixture)	Fine Aggregate (component 7)(kg in a m^3 mixture)	Age (day)	Concrete compressive strength(MPa, megapascals)
count	1030.000000	1030.000000	1030.000000	1030.000000	1030.000000	1030.000000	1030.000000
mean	281.167864	73.895825	54.188350	972.918932	773.580485	45.662136	35.817961
std	104.506364	86.279342	63.997004	77.753954	80.175980	63.169912	16.705742
min	102.000000	0.000000	0.000000	801.000000	594.000000	1.000000	2.330000
25%	192.375000	0.000000	0.000000	932.000000	730.950000	7.000000	23.710000
50%	272.900000	22.000000	0.000000	968.000000	779.500000	28.000000	34.445000
75%	350.000000	142.950000	118.300000	1029.400000	824.000000	56.000000	46.135000
max	540.000000	359.400000	200.100000	1145.000000	992.600000	365.000000	82.600000

Preprocessing of this dataset includes doing analysis on the independent variables like checking for null values in each column and then replacing or filling them with supported appropriate data types so that analysis and model fitting is not hindered from their way to accuracy. Shown above are some of the representations obtained by using Pandas tools which tell about variable count for numerical columns and model values for categorical columns. Maximum and minimum values in numerical columns, along with their percentile values for median, play an important factor in deciding which value to be chosen at priority for further exploration tasks and analysis. Data types of different columns are used further in label processing.

2.2 Logging

We should be able to log every activity done by the user

- The system identifies at which step logging require.
- The system should be able to log each and every system flow.
- The system should be not be hung even after using so much logging.
  Logging is done so that we can easily debug issues, so logging is mandatory to do.

2.3 Deployment

For the hosting of the project, we will use GCP (Google Cloud Platform).


Google Cloud

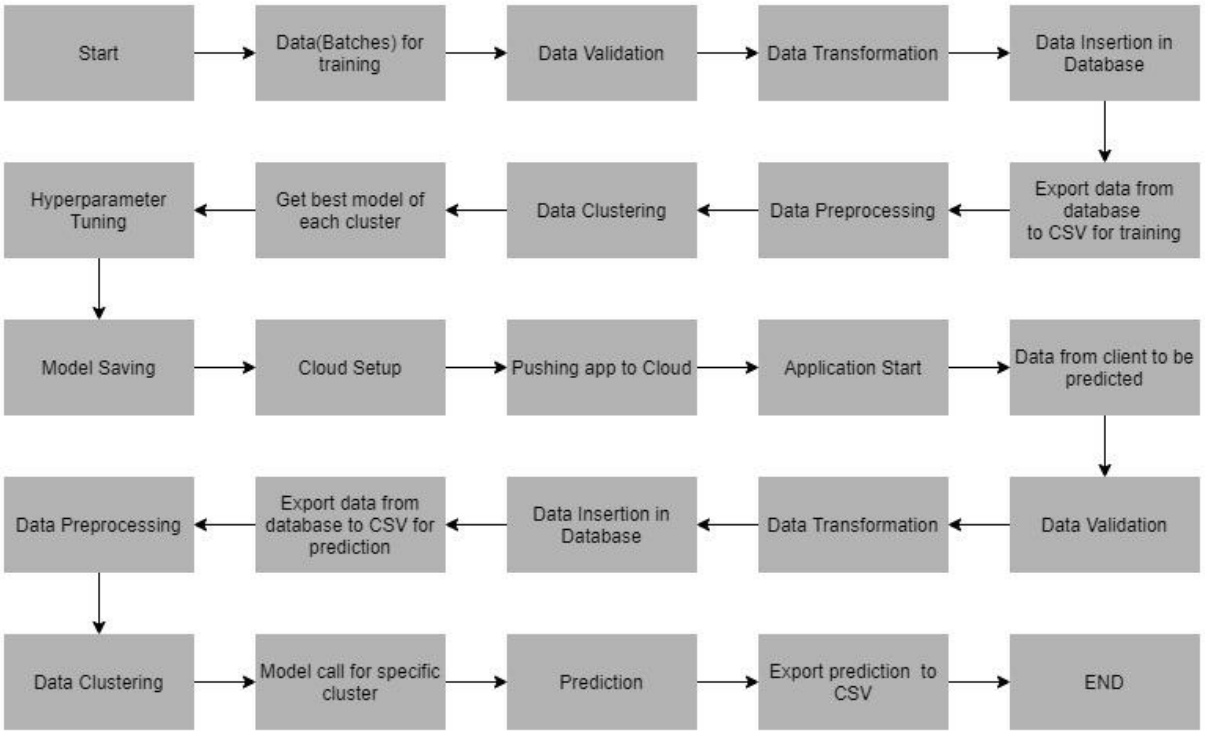
3. Technology Stack

Front End	HTML/CSS
Backend	Python/ Flask
Deployment	GCP

4. Proposed Solution

We will perform EDA to find the important relation between different attributes and willuse a machine-learning algorithm to estimate the compressive strength. The client will give the input with required features and will get results through the web application. The system will get features and it will be passed into the backend where the features will be validated and preprocessed and then it will be passed to a hyperparameter tuned machine learning model to predict the final outcome.

5. Architecture



5.1 Architecture Description

5.1.1 Data Gathering

Data source: Data is provided by client.
Dataset is stored in .csv format.

5.1.2 Raw Data Validation

After data is loaded, various types of validation are required before we proceed further with any operation. Validations like checking for zero standard deviation for all the columns, checking for complete missing values in any columns, etc. These are required because the attributes which contain these informations are of no use and it will not play role in contributing to the estimating of the compressive strength.

5.1.3 Exploratory Data Analysis

Visualized the relationship between the dependent and independent features. Also checked relationship between independent features to get more insights about the data.

5.1.4 Feature Engineering

After pre-processing, log transformations and standard scalar is performed to scale down all the numeric features. For this process, pipeline is created to log transform and to scale numerical features.

5.1.5 Model Building

After doing all kinds of pre-processing operations mention above and performing scaling and transformation, the data set is passed through a clustering algorithm (kmeans) to divide the dataset into different clusters, so that we can apply different algorithms in different clusters of the dataset. The dataset of each clusters are passed through a pipeline to both the models, Linear Regression and Random Forest. Then performance score is calculted for both the model and the model with the best score is selected for prediction for that perticular cluster.

5.1.6 Model Saving

The best model for each cluster is saved. Model is saved using pickle library in pickle` format.

5.1.7 Flask Setup for Web Application

After saving the model, the API building process started using Flask. Web application

creation was created in Flask for testing purpose. Whenever the user will enter the data in the path and give this path as an input in the web application then that data will be extracted by the model to predict the compressive strength of the concrete, this is performed in this stage.

5.1.8 GitHub

https://github.com/shailendra24sahu/Concrete_Compressive_Strength_Prediction
The whole project directory is pushed into the GitHub repository.

5.1.9 Deployment

The project was deployed into the GCP platform.

6. User Input / Output Workflow.