Architecture Design

# CONCRETE COMPRESSIVE STRENGTH PREDICTION

**Document Control**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author** | **Comments** |
| 1 | 01.09.2022 | Balamurali K |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Index

|  |  |
| --- | --- |
| **Content** | **Page No** |
| Abstract | 4 |
| 1. Introduction | 4 |
| 1.1 What is Architecture Design? | 4 |
| 1.2 Scope | 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 Data Gathering | 7 |
| 5.2 Raw Data Validation | 8 |
| 5.3 Exploratory Data Analysis | 8 |
| 5.4 Feature Engineering | 8 |
| 5.5 Model Building | 8 |
| 5.6 Model Saving | 8 |
| 5.7 Flask Setup for Web Application | 8 |
| 5.8 Github | 9 |
| 5.9 Deployment | 9 |
| 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 the concrete. Taking various aspects of a dataset, and the methodology followed for building a predictive model.

# Introduction

## What is Architecture Design?

The goal of Architecture Design (AD) is to give the internal design of the actual program code for the `Insurance Premium 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.

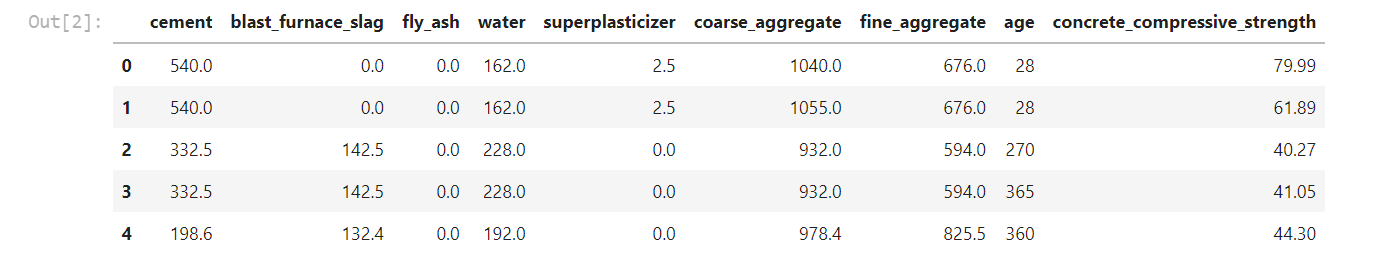
## Scope

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.

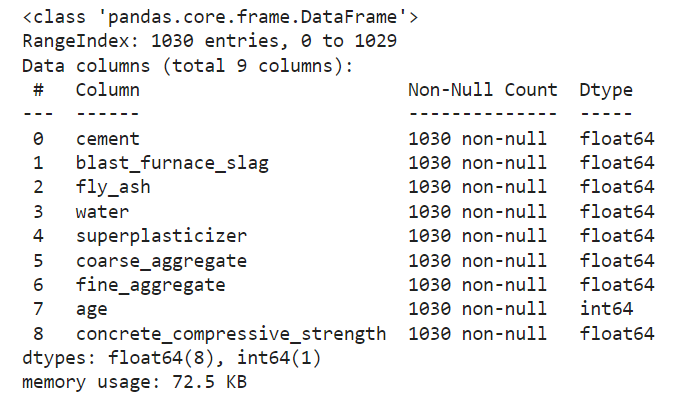
# Technical Specification

## Dataset

The dataset consists of over 1000 record with aforementioned information. 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 age, water, coarse aggregate, fine aggregate, super plasticizer, cement, fly ash and blast furnace slag . Using all the observations, it is inferred what role certain properties affect their compressive strength. The dataset looks like as follow:

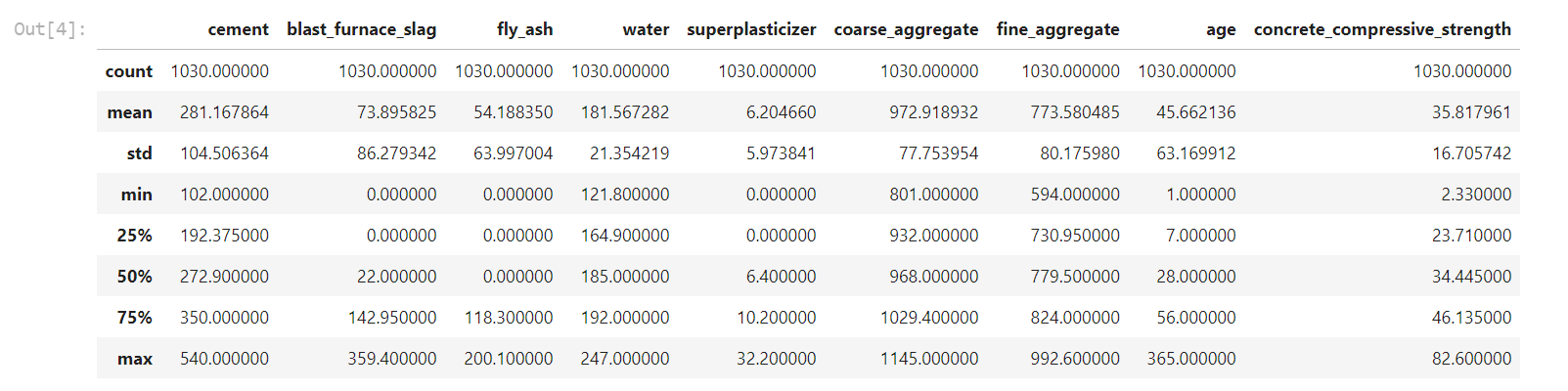


The data set consists of various data types from integer to floating to object as shown in Fig.



In the dataset, there can be various types of underlying patterns which also gives an in-depth knowledge about the preparation of the concrete. There are no categorical values present in the dataset.

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



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. 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.

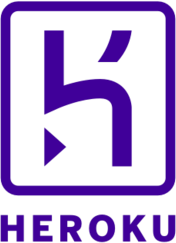
## Logging

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 just because we can easily debug issuing so logging is mandatory to do.

## Deployment

For the hosting of the project, we will use Heroku.



## Technology Stack

|  |  |
| --- | --- |
| **Front End** | HTML/CSS |
| **Backend** | Python/ Flask |
| **Deployment** | Heroku |

1. **Proposed Solution**

We will use performed EDA to find the important relation between different attributes and will use a machine-learning algorithm to estimate the concrete compressive strength. The client will be filled the required feature as input 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 pre-processed and then it will be passed to a hyperparameter tuned machine learning model to predict the final outcome.

## Architecture

Start

Model

Building

Model

Testing

Feature

gineering

Flask Setup

EDA

Data

Cleaning

Deployment

En

Data

Fetching

### Data Gathering

Data source: https://www.kaggle.com/datasets/elikplim/concrete-compressive-strength-data-set

Dataset is stored in .csv format.

### 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 duplicates, checking for complete missing values in any columns, etc. These are required because the attributes which contain these are of no use. It will not play role in contributing to the estimating compressive strength of the concrete.

### 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.

### Feature Engineering

After pre-processing, standard scalar is performed to scale down all the numeric features.

### Model Building

After doing all kinds of pre-processing operations mentioned above and performing scaling, the data set is passed through pycaret experimentation setup to find the best model after which the best model is fine tuned to increase the performance.

### Model Saving

Model is saved using dill library in .pkl format.

### 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. Whatever user will enter the data and then that data will be extracted by the model to estimate the compressive strength of the concrete, this is performed in this stage.

### GitHub

The whole project directory will be pushed into the GitHub repository.

### Deployment

The project was deployed from GitHub into the Heroku platform.

# User Input / Output Workflow.

