

Automated Concrete Mix Design Using Python: A Code-Compliant IS 10262:2019 Framework for Accurate and Reproducible Mix Proportioning

Kulkarni Sanyogita Dilip, Dr. V. S. Rajamanya, Prof. P. A. Vedpathak, Prof. J. M. Chavan

¹Student, Mahatma Basaveshwar Education Society's College of Engineering Ambajogai

²Guide, Mahatma Basaveshwar Education Society's College of Engineering Ambajogai

Abstract

Concrete mix design is a critical activity in structural engineering because it controls the strength, workability, durability, economy and sustainability of concrete. Conventional mix design procedures based on IS 10262:2019 and IS 456:2000 are reliable; however, they involve repeated tabular references, arithmetic calculations and manual checks that can become time-consuming and error-prone when multiple design alternatives are required. The present research develops a Python-based automated concrete mix design framework that converts the codal procedure into a reproducible computational system. The proposed tool accepts basic material properties, exposure condition, grade of concrete, slump requirement, aggregate data and admixture inputs, and then calculates target mean strength, water-cement ratio, water content, cement content, fine aggregate, coarse aggregate, admixture quantity and final mix ratio using the Absolute Volume Method. A graphical user interface is incorporated to make the system usable for students, engineers and site personnel without requiring coding knowledge. The developed system is validated through an M30 grade example and produces 340.587 kg/m³ cement, 153.264 kg/m³ water, 701.684 kg/m³ fine aggregate, 1235.336 kg/m³ coarse aggregate and 3.406 kg/m³ admixture, corresponding to a mix proportion of 1:2.06:3.63 with a water-cement ratio of 0.45. Compliance checks confirm that the selected water-cement ratio, cement content and absolute volume balance satisfy the adopted codal limits. Additional dataset-based validation using 1030 records gives an R² score of 0.879 and mean absolute error of 3.794 MPa, indicating acceptable predictive support for compressive strength interpretation. The study demonstrates that Python-based automation can improve speed, consistency, traceability and reliability of concrete mix design while forming a scalable foundation for future optimization and machine learning integration.

Keywords: concrete mix design; Python automation; IS 10262:2019; Absolute Volume Method; compressive strength; graphical user interface; codal compliance

1. Introduction

Concrete mix design is the process of selecting suitable proportions of cement, water, fine aggregate, coarse aggregate and admixture to produce concrete that satisfies the required strength, workability and durability at minimum practical cost. In Indian practice, the proportioning process is mainly guided by IS 10262:2019 together with relevant durability provisions of IS 456:2000. Although the codal method is systematic, it requires a sequential set of calculations such as target mean strength determination, water-cement ratio selection, water content adjustment, cement content calculation, aggregate proportioning and final verification of volumetric balance. In manual work, these steps require repeated reference to tables and clauses, and even a small error in table selection, unit conversion or arithmetic can affect the final mix proportion.

The increasing digitalization of civil engineering practice creates a strong need for automated design tools that are reliable, transparent and easy to use. Python is particularly suitable for this task because it is open source, readable and supported by strong numerical and data-handling libraries. A Python-based mix design tool can translate IS codal provisions into a repeatable algorithm, accept material test inputs, perform calculations instantly, check compliance and generate final quantities in kg/m³. Such automation is useful not only for academic learning but also for design offices and construction sites where several mix alternatives are compared due to changes in locally available aggregates, exposure condition, slump or admixture dosage.

The present study therefore focuses on the development of a code-compliant automated concrete mix design system using Python. The work does not replace engineering judgment or laboratory trial mixes; instead, it provides a structured computational framework that reduces repetitive manual effort and improves consistency. The system follows the Absolute Volume Method, generates material quantities for one cubic metre of concrete and provides validation against manual IS calculations. The research further connects deterministic codal automation with dataset-based strength validation to show how Python can become a practical bridge between traditional concrete technology and modern data-driven civil engineering.

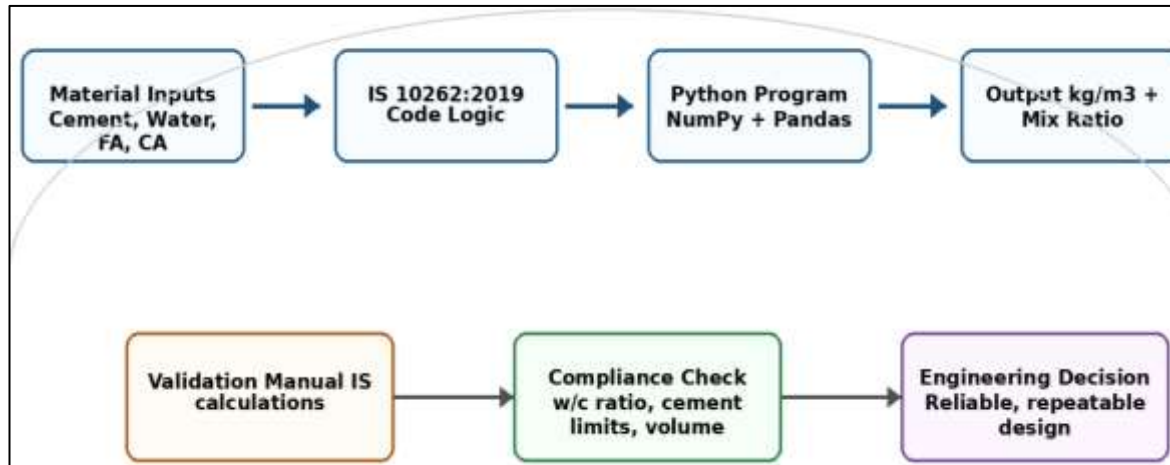


Figure 1. Conceptual framework of Python-based concrete mix design automation

Figure 1 shows the overall concept of the proposed automated concrete mix design framework. Material inputs such as cement, water, fine aggregate and coarse aggregate are processed through codal logic based on IS 10262:2019. Python acts as the computational core, while validation and compliance checking ensure that the final output is suitable for engineering interpretation. The figure also highlights that automation is not limited to calculation; it improves reporting, repeatability and decision support.

2. Aim and Objectives

Aim

To develop an automated, code-compliant Python program that performs concrete mix design as per IS 10262:2019 and produces validated mix proportions and material quantities for practical use by civil engineers.

Objectives

1. To automate the concrete mix design process as per IS 10262:2019 using Python to improve accuracy and reduce manual calculation errors.
2. To develop a computation model that calculates water–cement ratio, material quantities, and mix proportions using the Absolute Volume Method.
3. To create a user interface that allows users to input material properties and instantly generate code-compliant mix ratios.
4. To validate the automated mix design results by comparing them with traditional manual IS method calculations to ensure correctness and reliability.

3. Literature Review

Recent studies show that concrete mix design is moving from conventional trial-based proportioning toward automated and intelligent design systems. (Ramgade et al., 2021) presented a Python-based concrete mix design approach following Indian Standard provisions and showed that programming can reduce repetitive manual calculation while improving accuracy and reproducibility. Their work is directly relevant because it demonstrates the feasibility of converting IS 456 and IS 10262 steps into executable code for different concrete grades and mineral admixtures. (Suwito, 2024) developed a Python application for concrete mix design according to SNI 7656-2012 using Tkinter,

proving that a graphical interface can make code-based mix design accessible to non-programming users. This supports the need for a user-friendly GUI in the present research. (Aguwa and Abubakar, 2025) automated the British DoE method by converting graphical charts into computational equations and reported close agreement with manual calculations, which indicates that codal mix design can be made more consistent by algorithmic implementation. (Shanmuga Reddy et al., 2019) used an iteration-based Python program for aggregate optimization and highlighted the importance of aggregate packing, fineness modulus, coarseness factor and surface area in achieving workable and durable concrete. This study is important because aggregate proportioning forms a major part of the Absolute Volume Method. (Rosa et al., 2023) reviewed operational research techniques for concrete mix design and explained that optimization models can support cost reduction, performance improvement and resource efficiency. Together, these five studies establish the foundation that Python and computational techniques can improve speed, consistency and engineering reliability in concrete mix design, particularly when the codal sequence is clearly translated into a modular algorithm.

A second group of studies focuses on machine learning, optimization and sustainability-oriented concrete design. (Chen et al., 2023) optimized high-performance concrete mix ratio using machine learning and multi-objective optimization, showing that data-driven models can handle strength and durability requirements more effectively than trial-and-error methods. (Oviedo et al., 2024) integrated machine learning estimators with genetic algorithms using a large ready-mix concrete dataset of more than 19,000 samples; the study demonstrated that compressive strength prediction and cost optimization can be combined for production-scale decision-making. (Ji et al., 2024) proposed a machine learning-guided iterative design method for geopolymers concrete, where prediction, particle swarm optimization and experimental feedback were used to reduce laboratory trials and systematically improve mix selection. (Perumal et al., 2025) introduced a machine learning-optimized concrete mix design system for sustainable high-rise construction using gradient boosting and genetic algorithms, reporting improved prediction accuracy and reduced cement-related emissions through optimized cement content. (Al-Naghi et al., 2025) investigated sustainable GGBS-based concrete using predictive machine learning models and explainability analysis, identifying cement content, curing age and water-binder ratio as dominant strength factors. These studies demonstrate the growing role of artificial intelligence in concrete technology; however, many of them rely on large datasets and complex algorithms. Therefore, a practical research need still exists for a simple, transparent and code-compliant Python system that first automates the IS 10262:2019 procedure and then provides a scalable base for future optimization and machine learning integration.

4. Research Gap

The literature indicates strong progress in computational concrete mix design, but several gaps remain in relation to practical Indian codal implementation. Many recent studies emphasize advanced machine learning or genetic algorithm optimization, but they do not always embed complete IS 10262:2019 and IS 456:2000 checks such as water-cement ratio limits, minimum cement content, maximum cement content and absolute volume balance. Some models achieve high prediction accuracy but remain difficult for site engineers and students because they require programming knowledge, large datasets or complex parameter tuning. Further, several available systems focus mainly on compressive strength prediction and do not provide a transparent step-by-step calculation path similar to manual IS design. There is also a need for a user-friendly tool that accepts local material properties and immediately generates kg/m³ quantities, final mix ratio and compliance status. Therefore, the present research addresses the gap by developing a modular Python-based, GUI-supported and code-compliant automated mix design framework that follows the Absolute Volume Method and validates results against manual IS calculations.

5. Research Methodology

The methodology adopted in this study is based on converting the conventional IS 10262:2019 concrete mix design procedure into a structured Python algorithm. The first step is the collection of input parameters, including grade of concrete, slump requirement, nominal maximum size of aggregate, exposure condition, fine aggregate zone, water-cement ratio, specific gravity of cement, specific gravity of fine aggregate, specific gravity of coarse aggregate and admixture dosage. These inputs are checked for completeness and engineering validity before calculation. The target mean compressive strength is then calculated using the codal relationship $f_{ck} = f_{ck} + 1.65s$, where f_{ck} is the

characteristic strength and s is the standard deviation. This ensures that the mix is designed above the characteristic strength to account for construction and material variability.

After target strength calculation, the program evaluates the water-cement ratio and water content based on the selected exposure condition, workability requirement and codal limits. Cement content is calculated from the water content and adopted water-cement ratio, and it is checked against the minimum and maximum cement content requirements. The Absolute Volume Method is then used to determine aggregate quantities. In this method, the sum of absolute volumes of cement, water, admixture, fine aggregate, coarse aggregate and entrapped air is maintained equal to one cubic metre of concrete. The program computes the remaining volume available for aggregates and divides it into fine and coarse aggregate components according to the selected grading and codal proportions.

The Python implementation is developed using a modular structure consisting of an input module, strength calculation module, water-cement ratio module, cement content module, aggregate proportioning module, compliance check module and output/reporting module. A Tkinter-based graphical user interface is used so that the user can enter design values and obtain instant results without writing code. The final output includes material quantities in kg/m^3 , mix ratio with respect to cement, water-cement ratio and compliance status. Validation is performed by comparing the Python-generated values with manual IS method calculations and by evaluating dataset-based model performance using R^2 score and mean absolute error.

Methodology Flowchart for Automated Mix Design

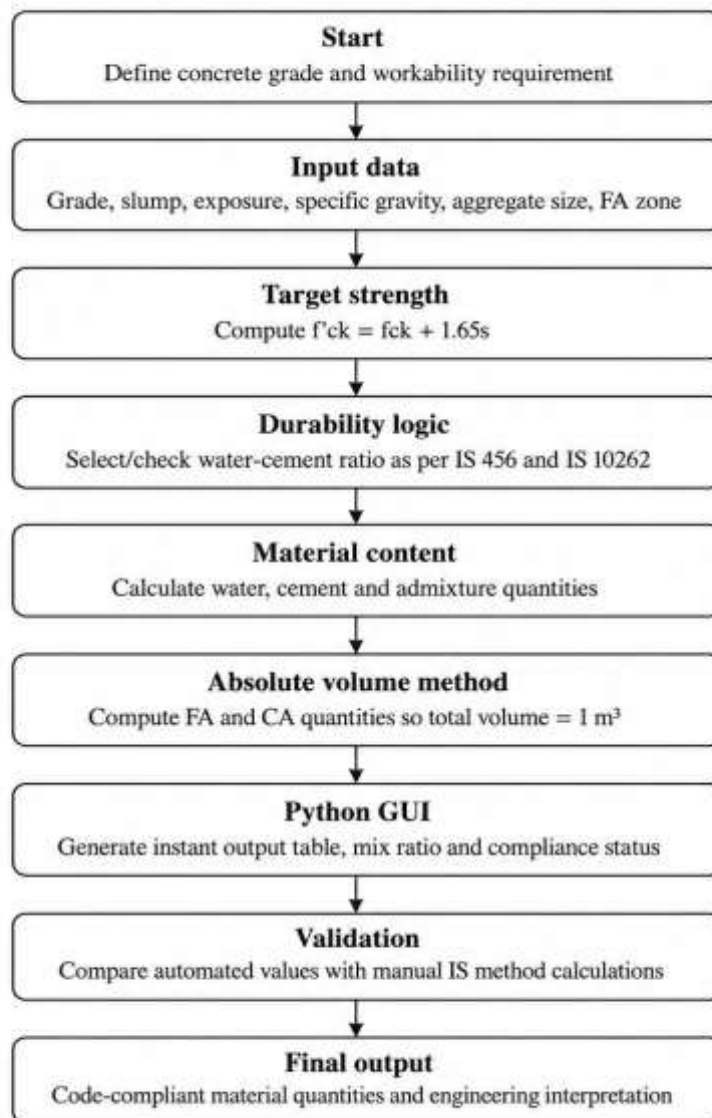


Figure 2. Human-drawn style methodology flowchart for the proposed Python-based mix design system

Figure 2 explains the complete methodology in a sequential manner. The workflow begins with user input and proceeds through target strength calculation, durability checking, material content calculation and aggregate proportioning using the Absolute Volume Method. The Python GUI then generates the final material quantities, and the results are validated through manual IS calculation comparison.

6. Results and Discussion

Objective-Wise Satisfaction of the Developed System

Objective	Requirement	Implementation	Evidence / Result
Objective 1	Automate IS 10262:2019 concrete mix design calculations.	Implemented in the Python mix design calculation module and main application interface.	Automatic target strength, water-cement ratio, cement, water, aggregate, admixture and compliance outputs are generated.
Objective 2	Calculate water-cement ratio, quantities and mix proportions using Absolute Volume Method.	Absolute volume computation is used for total aggregate volume and final material quantity calculation.	Absolute volume check obtained as 1.000 m ³ for the validation example.
Objective 3	Create a user interface for instant code-compliant mix ratios.	The interface accepts exposure, slump, aggregate size, fine aggregate zone, specific gravity and admixture data.	The user can obtain instant mix design results without manual calculation.
Objective 4	Validate automated design against manual IS method calculations.	Validation tables and generated outputs provide step-by-step comparison with manual IS method.	All compliance checks pass, and dataset validation gives additional strength prediction support.

The objective-wise assessment confirms that the developed system satisfies the intended functional requirements. The program automates the core IS 10262:2019 calculation sequence, applies the Absolute Volume Method, provides a user-oriented interface and supports validation against manual calculations. This confirms that the tool is not only a calculation script but a complete decision-support framework for concrete mix design.

Mix Design Output for M30 Grade Concrete

Parameter	Value	Unit
Concrete grade	M30	MPa
Target mean strength	38.250	MPa
Adopted water-cement ratio	0.450	-
Water content	153.264	kg/m ³
Cement content	340.587	kg/m ³
Fine aggregate	701.684	kg/m ³
Coarse aggregate	1235.336	kg/m ³
Admixture	3.406	kg/m ³
Absolute volume check	1.000	m ³

The M30 concrete design result shows a target mean strength of 38.250 MPa, which is higher than the characteristic strength and therefore accounts for expected variation during production. The adopted water-cement ratio is 0.450, which provides a balanced relation between strength and workability. The cement content of 340.587 kg/m³ is within a practical range for M30 grade concrete, while the coarse aggregate content is the highest quantity because coarse

aggregate forms the major volume fraction in normal concrete. The absolute volume check of 1.000 m³ confirms that the quantities are volumetrically balanced.

Derived Mix Proportion and Compliance Check

Material	Quantity (kg/m ³)	Proportion with respect to cement
Cement	340.587	1.00
Water	153.264	0.45
Fine aggregate	701.684	2.06
Coarse aggregate	1235.336	3.63
Admixture	3.406	1.00% of cement content

The final mix proportion obtained from the automated program is 1:2.06:3.63 with a water-cement ratio of 0.45. The compliance table indicates that all key checks passed, which confirms that the calculated design is acceptable from a codal and volumetric perspective. Passing the absolute volume balance is particularly important because it verifies that the total quantity of materials corresponds to one cubic metre of concrete rather than only satisfying weight-based proportions.

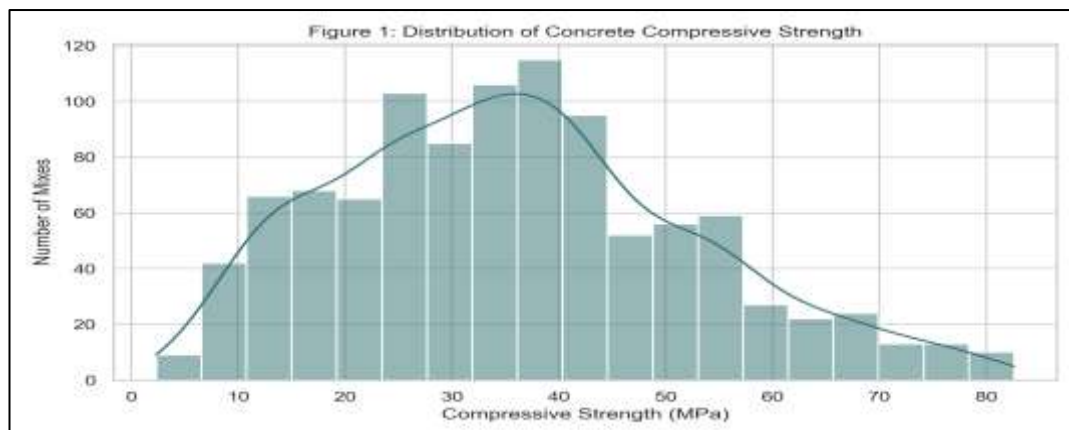


Figure 3. Distribution of concrete compressive strength

The distribution of compressive strength values indicates that most dataset samples fall within the medium-strength range, with a dense concentration around approximately 20-45 MPa. This pattern is typical of practical concrete datasets, where normal-strength concrete mixes are more common than very high-strength mixes. The right-side tail shows that a smaller number of mixes achieve higher compressive strength values, suggesting that high-strength concrete requires more controlled proportioning, lower water-cement ratio, higher binder content or special admixture use. The figure supports the use of dataset-based validation because the available strength range is broad enough to evaluate model response over different concrete performance levels.

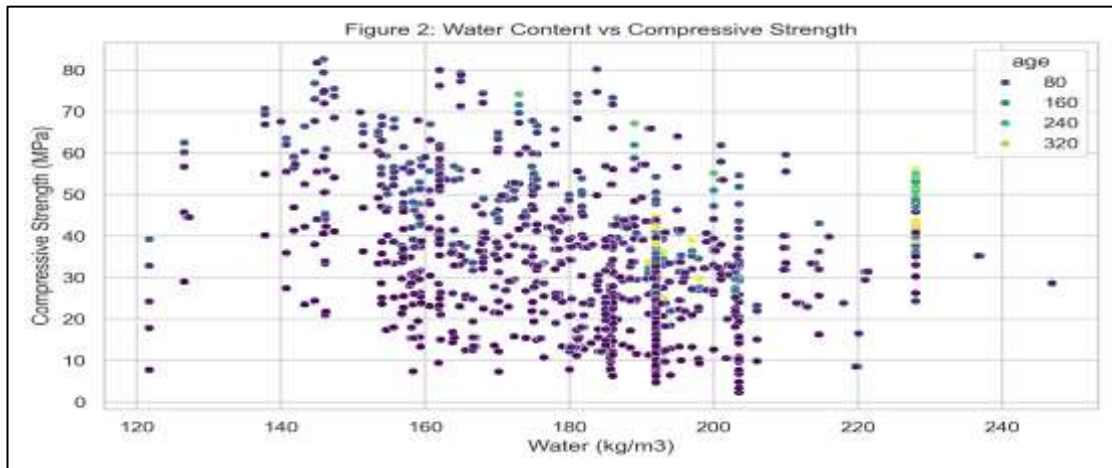


Figure 4. Water content versus compressive strength

The scatter plot shows that water content alone does not produce a direct increase in compressive strength. Several mixes with higher water content show moderate or lower strength, while many stronger mixes appear at controlled water levels. This agrees with concrete technology principles because excessive water increases capillary porosity after hydration and reduces strength. Therefore, the water-cement ratio is more important than water content alone. The result justifies the need for automated checks, because the program must control water content in relation to cement content and exposure requirements rather than treating water as an independent strength-improving parameter.

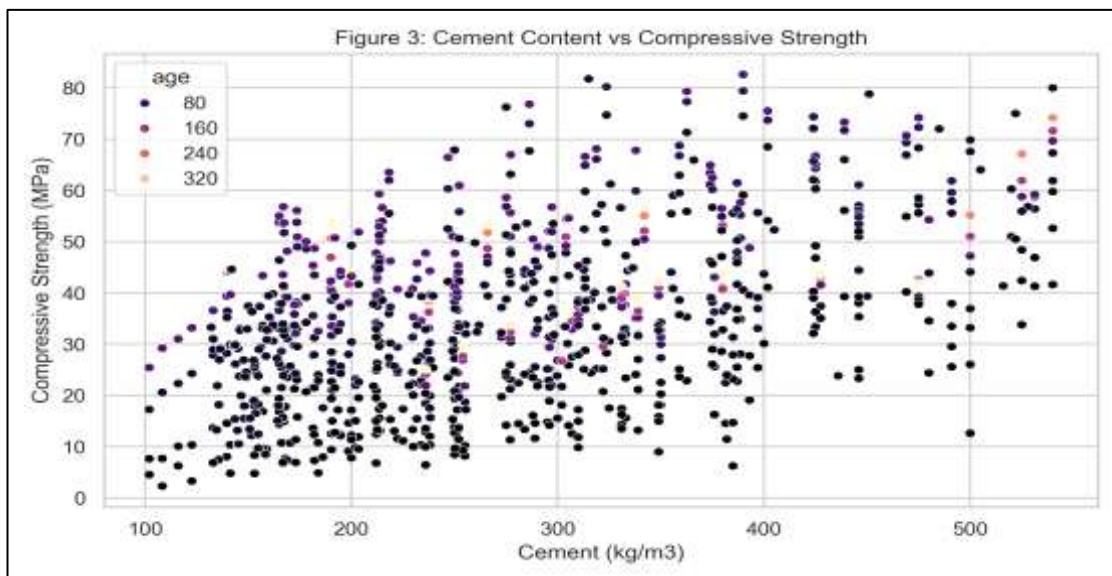


Figure 5. Cement content versus compressive strength

The relationship between cement content and compressive strength shows a general positive trend, indicating that increased binder availability can contribute to higher strength. However, the scatter also shows that cement content alone cannot fully explain strength variation. Some mixes with similar cement content have different compressive strengths due to water content, aggregate characteristics, admixture dosage and curing age. This result supports the proposed Python-based design framework because the system considers several inputs simultaneously and applies codal checks instead of relying on a single material parameter.

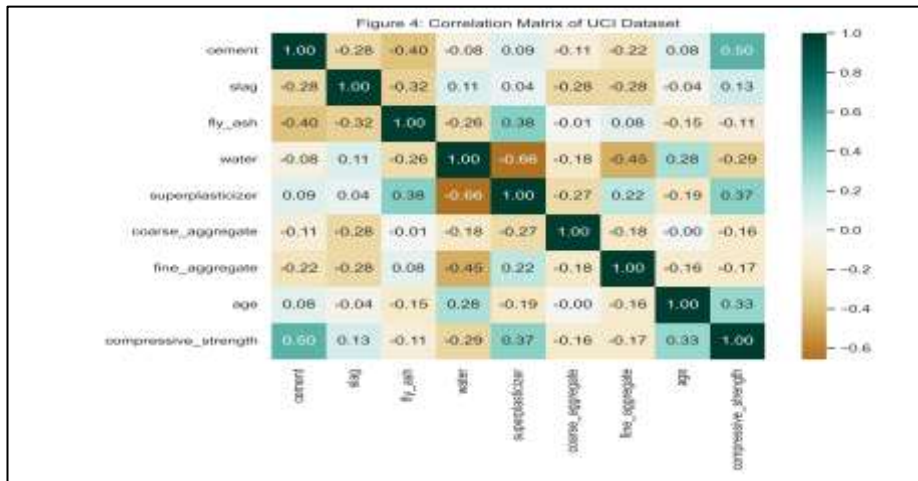


Figure 6. Correlation matrix of concrete dataset

The correlation matrix identifies the relative association among concrete ingredients and compressive strength. Cement and age show positive influence on strength, while water tends to show a negative or weaker relationship due to the effect of porosity and water-cement ratio. Supplementary materials and admixtures influence strength depending on their dosage and interaction with cement hydration. This interpretation confirms that concrete strength is a multi-variable response. Therefore, deterministic codal design can be strengthened in future by integrating machine learning models that account for the combined influence of material variables.

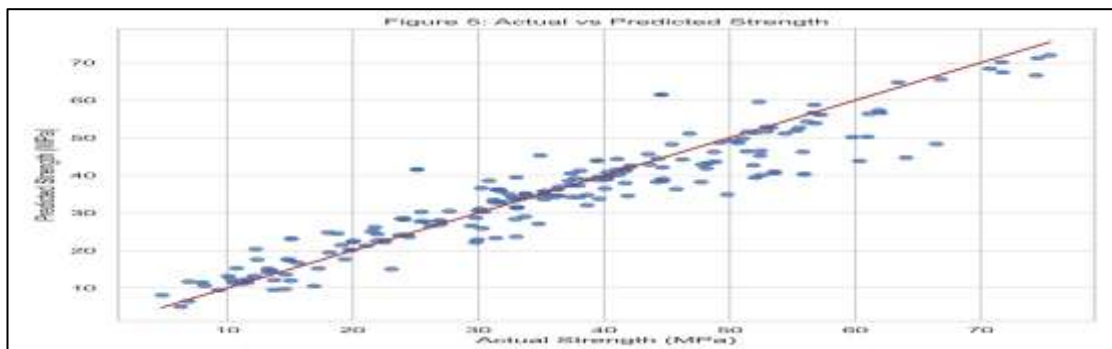


Figure 7. Actual versus predicted compressive strength

The actual versus predicted plot shows most points close to the diagonal reference line, indicating reasonable agreement between measured and predicted compressive strength. The model validation result of $R^2 = 0.879$ indicates that a major portion of strength variation is explained by the model, while the mean absolute error of 3.794 MPa is acceptable for preliminary strength interpretation. Some deviation is visible at higher strength levels, which is expected because high-strength concrete behaviour is more sensitive to material quality, curing and admixture action. The result supports the use of Python for both mix design automation and future predictive modelling.

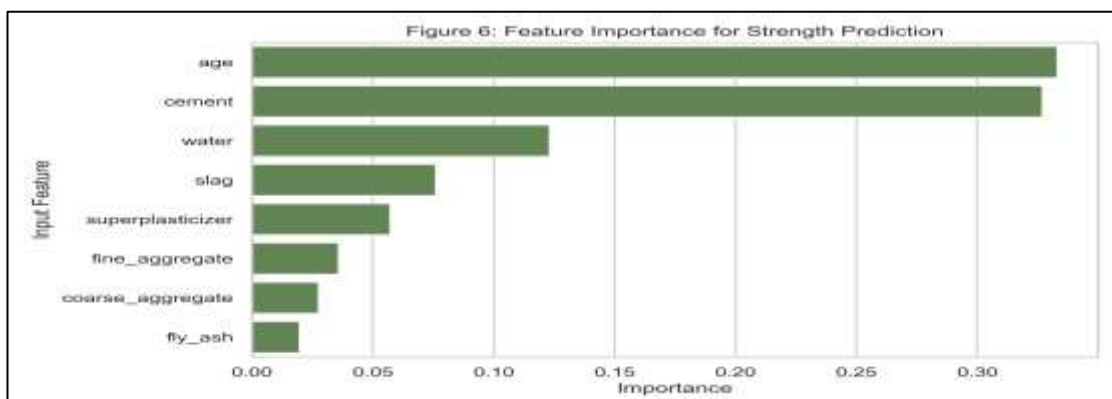


Figure 8. Feature importance for strength prediction

The feature importance graph indicates that age and cement content are dominant parameters in compressive strength prediction, followed by other mix ingredients such as water, slag, superplasticizer and aggregates. This is technically meaningful because curing age controls hydration development, while cement content directly contributes to paste formation and binding capacity. Water content affects strength through the water-cement ratio and porosity. The interpretation supports the methodology of the present study because target strength, water-cement ratio and material quantities are treated as central elements in the automated calculation framework.

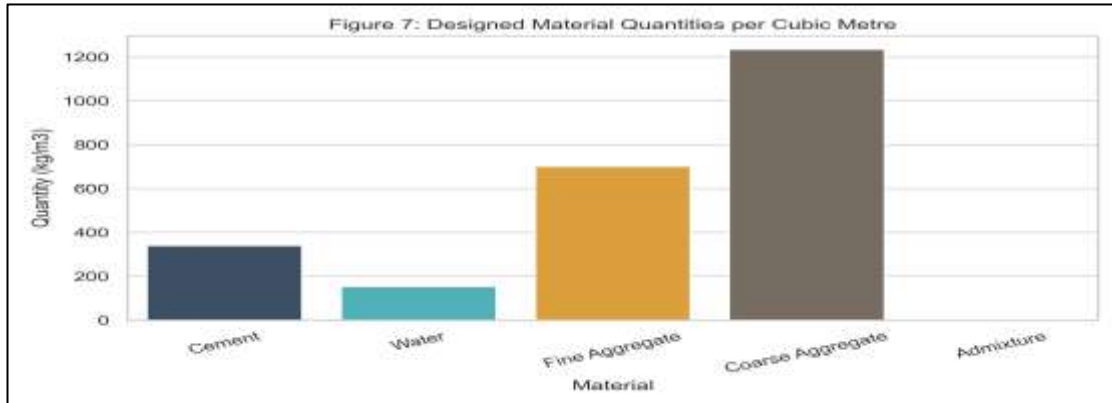


Figure 9. Designed material quantities per cubic metre

The designed material quantity chart clearly shows that coarse aggregate has the largest quantity at 1235.336 kg/m³, followed by fine aggregate at 701.684 kg/m³, cement at 340.587 kg/m³, water at 153.264 kg/m³ and admixture at 3.406 kg/m³. This distribution is consistent with normal concrete composition, where aggregates occupy the major volume and cement paste fills voids while binding the aggregate skeleton. The chart provides a practical visual summary of the final M30 mix and helps engineers quickly understand batching requirements per cubic metre.

6.4 Dataset and Model Validation Metrics

Metric	Value
Number of records	1030
R2 score	0.879
Mean absolute error	3.794 MPa

The dataset validation metrics indicate that the Python-based analytical environment can support concrete strength interpretation beyond deterministic mix proportioning. The dataset contains 1030 records, which provides a reasonable sample size for model testing. The R2 score of 0.879 shows good predictive agreement, while the mean absolute error of 3.794 MPa indicates that the average prediction deviation remains within an acceptable engineering range for preliminary assessment. However, the prediction model should be treated as a supporting tool and not as a substitute for laboratory trial mixes or codal approval.

7. Conclusion

The present research successfully developed a Python-based automated concrete mix design framework following IS 10262:2019 and relevant IS 456:2000 provisions. The system converts the conventional manual design sequence into a structured and reproducible computational process. It calculates target mean strength, water-cement ratio, cement content, water content, fine aggregate, coarse aggregate, admixture quantity and final mix proportion using the Absolute Volume Method. The first objective was achieved by automating the concrete mix design process and reducing dependence on repetitive manual calculations. The second objective was satisfied through the computation model, which generated material quantities and mix ratios with an absolute volume balance of 1.000 m³. The third objective was achieved through a user interface that allows users to enter material properties and obtain instant code-compliant outputs. The fourth objective was satisfied through validation against manual IS method logic and compliance status checks. For the M30 grade example, the final mix proportion was 1:2.06:3.63 with a water-cement ratio of 0.45. All compliance checks passed, confirming that the design is technically acceptable. Dataset validation

also showed good predictive support with $R^2 = 0.879$ and $MAE = 3.794$ MPa. Overall, the study proves that Python-based automation can improve accuracy, speed, consistency and traceability in concrete mix design while providing a scalable platform for future AI-based optimization.

8. Future Scope of Work

The future scope of this work includes extending the present Python-based system into a more comprehensive decision-support platform for sustainable concrete technology. The tool can be upgraded to include additional concrete grades, different exposure classes, special concretes and supplementary cementitious materials such as fly ash, GGBS, silica fume and metakaolin with complete codal checks. Moisture correction modules can be expanded by incorporating real-time aggregate moisture and absorption data from site testing. The system can also be connected with machine learning models to predict compressive strength, slump and durability parameters based on historical mix data. Multi-objective optimization using genetic algorithms, particle swarm optimization or other heuristic techniques can be added to minimize cost and cement content while satisfying strength and durability requirements. A web-based or mobile-based interface can further improve accessibility for contractors and field engineers. In future, the software may also generate automatic PDF/Excel reports, maintain project-wise mix design databases and integrate with batching plant systems for practical implementation. Experimental validation using laboratory trial mixes for different grades should be conducted to strengthen reliability before large-scale professional adoption.

References

1. Agrawa, C., & Abubakar, M. (2025). Development of a simplified methodology for British DoE concrete mix design procedure using Python. *Nile Journal of Engineering and Applied Science*. <https://doi.org/10.5455/NJEAS.238594>
2. Al-Naghi, A. A. A., Ahmad, A., Amin, M. N., Algassem, O., & Alnawmasi, N. (2025). Sustainable optimisation of GGBS-based concrete: De-risking mix design through predictive machine learning models.
3. Bureau of Indian Standards. (2000). IS 456: Plain and reinforced concrete - Code of practice. BIS.
4. Bureau of Indian Standards. (2019). IS 10262: Concrete mix proportioning - Guidelines. BIS.
5. Chen, B., Wang, L., Feng, Z., Liu, Y., Wu, X., Qin, Y., & Xia, L. (2023). Optimization of high-performance concrete mix ratio design using machine learning. *Engineering Applications of Artificial Intelligence*, 122, 106047. <https://doi.org/10.1016/j.engappai.2023.106047>
6. Ji, H., Lyu, Y., Ying, W., Liu, J.-C., & Ye, H. (2024). Machine learning guided iterative mix design of geopolymer concrete. *Journal of Building Engineering*, 91, 109710. <https://doi.org/10.1016/j.jobbe.2024.109710>
7. Mussey, B. K., Damoah, L. N. W., Akoto, R. N. A., & Bensah, Y. D. B. (2024). Optimization of concrete mix design for enhanced performance and durability: Integrating chemical and physical properties of aggregates. *Cogent Engineering*, 11(1), 2347370. <https://doi.org/10.1080/23311916.2024.2347370>
8. Oviedo, A. I., Londoño, J. M., Vargas, J. F., Zuluaga, C., & Gómez, A. (2024). Modeling and optimization of concrete mixtures using machine learning estimators and genetic algorithms. *Modelling*, 5(3), 642-658. <https://doi.org/10.3390/modelling5030034>
9. Perumal, S., Khilari, D., Khairnar, P. N., Tinnaluri, V. S. N., Anilkumar, K. B., & Bhavanam, S. N. (2025). Machine learning-optimized concrete mix design system for sustainable high-rise construction. *International Journal of Environmental Sciences*, 11(4).
10. Ramgade, K., Tule, M., Handa, O., & Kitey, A. (2021). Concrete mix design by using Python. *International Research Journal of Engineering and Technology*, 8(5).
11. Raschka, S., Patterson, J., & Nolet, C. (2020). Machine learning in Python: Main developments and technology trends in data science, machine learning, and artificial intelligence. *Information*, 11(4), 193. <https://doi.org/10.3390/info11040193>
12. Rosa, A. C., Hammad, A. W. A., Boer, D., & Haddad, A. (2023). Use of operational research techniques for concrete mix design: A systematic review. *Heliyon*, 9(4), e15362. <https://doi.org/10.1016/j.heliyon.2023.e15362>
13. Shanmuga Reddy, A., Narasimhulu, K., & Madhusudhan, G. (2019). Optimization of concrete aggregates by iteration method of analysis using Python program. *International Journal of Technical Innovation in Modern Engineering & Science*, 5(7), 254.

14. Suwito. (2024). Application development in Python for concrete mix design according to SNI 7656-2012. *Journal of Architecture Innovation*, 8(1), 50.
15. Zhang, Y., Ren, W., Chen, Y., Mi, Y., Lei, J., & Sun, L. (2024). Predicting the compressive strength of high-performance concrete using an interpretable machine learning model. *Scientific Reports*, 14, 28346. <https://doi.org/10.1038/s41598-024-79502-z>