Journal of Soft Computing in Civil Engineering 5-1 (2021) 19-37

Contents lists available at **SCCE**

Journal of Soft Computing in Civil Engineering

Journal homepage: www.jsoftcivil.com

Application and Analysis of Machine Learning Algorithms for Design of Concrete Mix with Plasticizer and without Plasticizer

Sh. Pandey[1*], V. Kumar[2], P. Kumar[1]

1. Post Graduate Student, Department of Civil Engineering, Indian Institute of Technology (IIT) BHU, Varanasi, India
2. Professor, Department of Civil Engineering, Indian Institute of Technology (IIT) BHU, Varanasi, India

Corresponding author: *shivam.pandey.civ15@iitbhu.ac.in*

https://doi.org/10.22115/SCCE.2021.248779.1257

ARTICLE INFO

Article history:
Received: 17 September 2020
Revised: 06 January 2021
Accepted: 14 January 2021

Keywords:
Machine learning;
Linear regression;
Support vector regression;
Decision tree regression;
Artificial neural network;
Feature selection.

ABSTRACT

The objective of this paper is to find an alternative to conventional method of concrete mix design. For finding the alternative, 4 machine learning algorithms viz. multi-variable linear regression, Support Vector Regression, Decision Tree Regression and Artificial Neural Network for designing concrete mix of desired properties. The multi-variable linear regression model is just a simplistic baseline model, support vector regression Artificial Neural Network model were made because past researchers worked heavily on them, Decision tree model was made by authors own intuition. Their results have been compared to find the best algorithm. Finally, we check if the best performing algorithm is accurate enough to replace the convention method. For this, we utilize the concrete mix designs done in lab for various on site designs. The models have been designed for both mixes types – with plasticizer and without plasticizer The paper presents detailed comparison of four models Based on the results obtained from the four models, the best one has been selected based on high accuracy and least computational cost. Each sample had 24 features initially, out of which, most significant features were chosen which were contributing towards prediction of a variable using f regression and p values and models were trained on those selected features. Based on the R squared value, best fitting models were selected among the four algorithms used. From the paper, the author(s) conclude that decision tree regression is best for calculating the amount of ingredients required with R squared values close to 0.8 for most of the models. DTR model is also computationally cheaper than ANN and future works with DTR in mix design is highly recommended in this paper.

How to cite this article: Pandey S, Kumar V, Kuma P. Application and analysis of machine learning algorithms for design of concrete mix with plasticizer and without plasticizer. J Soft Comput Civ Eng 2021;5(1):19–37. https://doi.org/10.22115/scce.2021.248779.1257.

1. Introduction

Progress in the preparation of concrete mix is at a moderate level. The most mainstream strategy used to get the measure of ingredients required, in a little changed structure, has been utilized for quite a long time. These techniques have numerous disadvantages - being laborious and time consuming are a few. We want to introduce a way to design a concrete mix based on a mathematical equation developed by the machine learning algorithm. AI as a field is growing powerfully as of late. . Practically speaking, AI intends to utilize different cutting edge accomplishments in software engineering to expand upon a framework that will have the option to gain from informational collections and, subsequently, look for examples and connections among factors, which would be challenging to conduct with conventional methods.

If we are successful in estimating the quantity of ingredients required using these machine lerning algorithms with sufficient accuracy, we would be saving time and resources involved in conventional design process and also include the experience of engineers due to which they are able to vary the amount of various ingredients due to different conditions. This experience is not given in any code but machine learning algorithms can also include this.

With a wrong manufacturing process, for example, poor concrete curing can cause excessive cracks and reduce concrete tightness [1]. Traditional approach is step by step design methodology. These methods have evolved from arbitrary 1-2-3 cement-sand-aggregate volumetric ratio methods which were used in early 1900s [2] to the present-day method where every ingredient is estimated by weight and definite rules are given in design codes for their estimation. The contemporary method for utilizing design codes gives a blend of emperical and statistical methods. these mix of empirical and statistical methods are often insufficient to describe such complex relationships. compared to the previously mentioned customary emperical and statistical methods, AI methods don't depend upon express conditions; rather, AI models are learning calculations that discover learning algorithms that find patterns information to foresee future vlaues. These strategies are more computationally costly than statistical procedures; be that as it may, analysts have progressively applied AI methods in concrete blend in light of their capacity to represent the intricacy of concrete blends and their properties. One AI technique is utilizing ANN. Kasperkiewicz et al. modelled compressive strength of HPC using ANNs; using 6 features and obtained a R2 of 0.757 [3]. After that, researchers have applied ANN to many different problems in cement & concrete research; they have been used to model different types of concrete properties, like slump, filling capacity, compressive strength, and segregation. For many types of concrete too such as HPC [3,4], self consolidating concrete [5,6], RMC [7], high-strength concrete [7,8], ultra high performance concrete [9], recycled aggregate concrete [10], and structural lightweight concrete [11]. Another method is using Decision Tree Regression. The first regression tree algorithm was published by Morgan et al. around the time that other machine learning algorithms were first being developed [12]. Fundamental tree-based models experience inconvenience finding the model with best prescient execution. Thus, research in machine learning since the 1960s has zeroed in on enlarging the idea of a basic tree-based model with extra design features ,variations on regression trees frequently perform in a way that is better than the other machine learning algorithms. Such as, Erdal showed that regression tree ensembles that use bagging and boosting have better performance to the simple decision tree model for concrete property prediction [13]. The model tree approach has been used to predict compressive strength for different types of concrete types including HPC [14], recycled aggregate concrete (RAC) [15,16], fiber-reinforced polymer [17], and high-volume mineral-admixture concrete [18]. Another approach is using Support Vector Regression. Vapnik and Chervonenkis invented SVM [19]; Boser et al. developed a training algorithm to optimize margin classifiers in 1992 [20]. In 2007, Gupta et al. predicted concrete compressive strength using SVM, where, R2 = 0.992 [21]. This showed that SVMs are good modeling tool for concrete property modeling specially when a dataset is small, because the user only needs to define 2 parameters. Many other researchers have used SVMs after that to predict a different concrete mixture properties, like elastic modulus [22], compressive strength [14,21,23–25], and splitting tensile strength [26]. 1 hidden layer ANN has been used by Naderpour et. al. to predict compressive strength of environmental friendly concrete [27]. The SVM performance decreases as number of features increase.

2. Experimental dataset

The dataset consists of 800 samples which were designed and tested in structural engineering lab of IIT BHU Varanasi for various on site design requirements. These designs have been used for actual construction works. To remove any effect of time, we used design values from years 2012, 2015, 2017 and 2019. For each sample, 32 parameters were initially compiled viz. grade of concrete (Grade), slump achieved (slump), 7 day compressive strength, 28 day compressive strength, amount of water (water), amount of cement (cement), amount of sand (sand), coarse aggregate of size 10 mm (CA10) , coarse aggregate of size 20 mm (CA20), plasticizer added, fineness modulus (FM) of three types of aggregates, bulk density of aggregates (BD), specific gravity of aggregates(SG), water absorption of aggregates (WA), consistency of cement paste, soundness of cement, initial setting time of cement (Initial ST), final setting time of cement (Final ST), 3 day, 7 day and 28 day compressive strength of mortar, unit weight of cement (UW) and specific gravity of cement. Out of these 32 parameters, amount of water, cement, sand, coarse aggregate of size 10mm, coarse aggregate of size 20mm and plasticizer (if needed) were targets and rest were further analyzed for their contribution towards strength prediction.

The dataset was first broken into 2 parts viz. design where plasticizer has been used and design where plasticizer has been not been used. Further these two divisions each were broken in two parts – designs where PPC has been used and designs where OPC has been used. Thus, we have made models for 2 kinds of design – concrete mix with plasticizer and concrete mix without plasticizer.

3. Feature selection

Since in total, there are 24 features, and obviously not every feature contributes toward predicting a target. It may be possible that one feature is crucial in predicting one target and at the same time, totally useless while predicting another target. It may be possible that two features are highly correlated to each other and presence of only one of them is enough. Thus, we found correlation between all the features and removed any feature with 0.9 or more correlation to another feature.

Table 1.
Correlation Values of different features.

	grade	slump	20finns	10finns	sandFinns	20BD	10BD	sandBD	20SG	10SG	sandSG
grade	1	0.149684	0.047312	0.063508	-0.06296	0.024342	-0.03676	0.109403	-0.01772	-0.02373	-0.17345
slump	0.149684	1	-0.00277	0.046469	-0.02393	-0.13837	0.073955	0.272915	-0.09765	-0.04111	-0.06235
20finns	0.047312	-0.00277	1	0.650473	0.257077	0.249478	-0.26473	0.188877	-0.00776	-0.45063	-0.04512
10finns	0.063508	0.046469	0.650473	1	0.070928	0.166533	-0.2173	0.156423	-0.03251	-0.6376	-0.05741
sandFinns	-0.06296	-0.02393	0.257077	0.070928	1	0.039942	-0.19629	0.148832	-0.12425	-0.0948	0.24006
20BD	0.024342	-0.13837	0.249478	0.166533	0.039942	1	0.340065	0.178882	0.450262	0.181763	-0.17546
10BD	-0.03676	0.073955	-0.26473	-0.2173	-0.19629	0.340065	1	0.074185	0.372482	0.55278	0.008589
sandBD	0.109403	0.272915	0.188877	0.156423	0.148832	0.178882	0.074185	1	-0.00068	0.018433	-0.03883
20SG	-0.01772	-0.09765	-0.00776	-0.03251	-0.12425	0.450262	0.372482	-0.00068	1	0.524984	0.090279
10SG	-0.02373	-0.04111	-0.45063	-0.6376	-0.0948	0.181763	0.55278	0.018433	0.524984	1	0.057379
sandSG	-0.17345	-0.06235	-0.04512	-0.05741	0.24006	-0.17546	0.008589	-0.03883	0.090279	0.057379	1
20WA	0.003462	0.017137	0.005137	0.041553	0.015112	0.068244	-0.03847	0.037838	-0.03061	-0.04014	0.067726
10WA	-0.03031	0.018975	0.019252	0.057279	0.002657	0.058344	-0.05039	0.014454	-0.03621	-0.04151	0.003194
sandWA	0.000948	0.037306	0.078757	0.061156	0.041853	0.066953	-0.02275	0.026391	-0.00292	-0.00932	0.020907
emCNSTNC	-0.05347	-0.00534	-0.04985	-0.0469	-0.11516	0.023987	-0.00956	0.020599	0.019409	0.086737	-0.04945
initialST	-0.13648	-0.00482	-0.02871	-0.05633	-0.09194	0.037306	0.031025	0.113788	0.032255	0.170886	-0.27741
finalST	0.002824	0.021846	0.085451	0.036277	0.098593	0.074755	0.035912	0.071599	-0.1022	-0.04075	0.004829
3dCem	0.031514	-0.00486	-0.43916	-0.5614	0.010455	-0.12104	0.198237	0.032127	-0.00388	0.440146	0.029607
7dCem	0.052968	0.036154	-0.07285	0.059461	0.007924	0.014526	0.056284	0.047011	0.008036	-0.02833	-0.02581
28dCem	0.114079	-0.02256	-0.10733	0.022735	-0.09763	0.020662	0.058596	0.047301	0.09472	0.044387	-0.02762
cemUW	0.149994	-0.12123	0.08255	0.1038	0.100082	0.069945	-0.14755	-0.0741	0.00167	-0.20012	0.121759
cemSG	0.125804	-0.04554	0.04032	0.078265	0.086543	0.012222	0.007085	-0.0996	0.019116	-0.16202	0.201197

Table 1 (contd.)
Correlation Values of different features.

	20WA	10WA	sandWA	cemCNSTNCY	initialST	finalST	3dCem	7dCem	28dCem	cemUW	cemSG
grade	0.003462	-0.03031	0.000948	-0.0534657	-0.13648	0.002824	0.031514	0.052968	0.114079	0.149994	0.125804
slump	0.017137	0.018975	0.037306	-0.00534322	-0.00482	0.021846	-0.00486	0.036154	-0.02256	-0.12123	-0.04554
20finns	0.005137	0.019252	0.078757	-0.04984822	-0.02871	0.085451	-0.43916	-0.07285	-0.10733	0.08255	0.04032
10finns	0.041553	0.057279	0.061156	-0.0469003	-0.05633	0.036277	-0.5614	0.059461	0.022735	0.1038	0.078265
sandFinns	0.015112	0.002657	0.041853	-0.11516208	-0.09194	0.098593	0.010455	0.007924	-0.09763	0.100082	0.086543
20BD	0.068244	0.058344	0.066953	0.02398694	0.037306	0.074755	-0.12104	0.014526	0.020662	0.069945	0.012222
10BD	-0.03847	-0.05039	-0.02275	-0.00956299	0.031025	0.035912	0.198237	0.056284	0.058596	-0.14755	0.007085
sandBD	0.037838	0.014454	0.026391	0.02059862	0.113788	0.071599	0.032127	0.047011	0.047301	-0.0741	-0.0996
20SG	-0.03061	-0.03621	-0.00292	0.01940911	0.032255	-0.1022	-0.00388	0.008036	0.09472	0.00167	0.019116
10SG	-0.04014	-0.04151	-0.00932	0.08673653	0.170886	-0.04075	0.440146	-0.02833	0.044387	-0.20012	-0.16202
sandSG	0.067726	0.003194	0.020907	-0.04944992	-0.27741	0.004829	0.029607	-0.02581	-0.02762	0.121759	0.201197
20WA	1	0.683804	0.884127	0.02469075	0.083748	-0.01704	-0.00335	0.014388	-0.05022	0.006595	-0.00423
10WA	0.683804	1	0.632288	-0.00382751	0.001257	-0.01279	0.035454	0.079675	0.028554	0.052814	0.048241
sandWA	0.884127	0.632288	1	0.01326339	0.057866	-0.00978	-0.0081	0.017536	-0.02257	0.002015	0.009083
emCNSTNC	0.024691	-0.00383	0.013263	1	0.427268	0.032225	-0.09766	-0.1659	-0.17036	-0.43018	-0.47488
initialST	0.083748	0.001257	0.057866	0.42726777	1	0.02972	-0.121	-0.22235	-0.20035	-0.69634	-0.78114
finalST	-0.01704	-0.01279	-0.00978	0.03222504	0.02972	1	-0.1485	-0.38255	-0.17288	-0.05461	-0.10948
3dCem	-0.00335	0.035454	-0.0081	-0.09765804	-0.121	-0.1485	1	0.509969	0.421801	0.22237	0.28147
7dCem	0.014388	0.079675	0.017536	-0.16590223	-0.22235	-0.38255	0.509969	1	0.764037	0.334495	0.443168
28dCem	-0.05022	0.028554	-0.02257	-0.1703584	-0.20035	-0.17288	0.421801	0.764037	1	0.313966	0.413147
cemUW	0.006595	0.052814	0.002015	-0.43018036	-0.69634	-0.05461	0.22237	0.334495	0.313966	1	0.896594
cemSG	-0.00423	0.048241	0.009083	-0.47488111	-0.78114	-0.10948	0.28147	0.443168	0.413147	0.896594	1

From these correlation values, we found that water absorption (WA) of sand and coarse aggregate are highly positively correlated for the samples we have taken. But that is only a coincidence as these quantities are independent of each other. Hence, we are keeping both. Specific gravity (SG) and unit weight (UW) of sand are highly positively correlated, and that's quite obvious. Cement specific gravity (SG) and initial setting time (ST) show high negative correlation i.e. as cement SG increases, its initial ST decreases. Fineness Modulus (FM) of CA

10 and CA 20 show high positive correlation but again, that's just a coincidence. Cement UW and initial ST also show high negative correlation. 28 day and 7 day strength of cement also show high positive correlation and that's expected too. Using f statistics and p values from Scikit Learn library, for each target we chose features with p-values less than 0.05 i.e. 95% significance level. The thumb rule of selecting the features using f statistics is that we choose a significance level (here 95%) of each feature for predicting the target variable. All the features more significant that this threshold are considered as useful in predicting.

4. Scaling and splitting the dataset for testing and training

After splitting up the dataset into 2 parts - concrete mix with plasticizer and concrete mix without plasticizer, we move to scaling and splitting of dataset. For Support Vector Regression and Neural Network, as a rule of thumb, we scaled the input features to speed up learning and faster convergence in the range 0 to 1. However, for Linear Regression and Decision Tree Regression, these algorithms don't have any significant increase in performance due to scaling. Therefore, we used the input features in their original form. For Linear model, SVR and DTR, we split the dataset into 2 parts: training and testing dataset in ratio of 80:20. For Neural Network, we split the Dataset into 3 parts: training, validation and testing dataset in the ratio of 70:10:20.After preprocessing has been completed and dataset was split for training and testing, we deployed the learning algorithms over the training dataset and used the model learnt to predict the data of test dataset. The hyper parameters which gave best predictions were finalized and model was finalized once it gave satisfactory predictions.

5. Results of different models with plasticizer added

For all ANN models, the number of hidden layers is 6 and number of nodes is twice the number of features in first 2 layers which decreases to half the value of previous layers nodes for each subsequent layer. For all these nodes, the activation function used is relu. For output layer – the number of nodes is 1 and activation function is linear.

Table 2
R square value for different models for design with Plasticizer.

	Linear	**SVR**	**DTR**	**ANN**
Water	0.39	0.45	0.88	0.57
Cement	0.63	0.63	0.76	0.33
Sand	0.69	0.43	0.77	0.81
CA10	0.35	0.37	0.79	0.72
CA20	0.55	0.36	0.66	0.64
Plasticizer	0.33	0.38	0.74	0.34

(i) Water

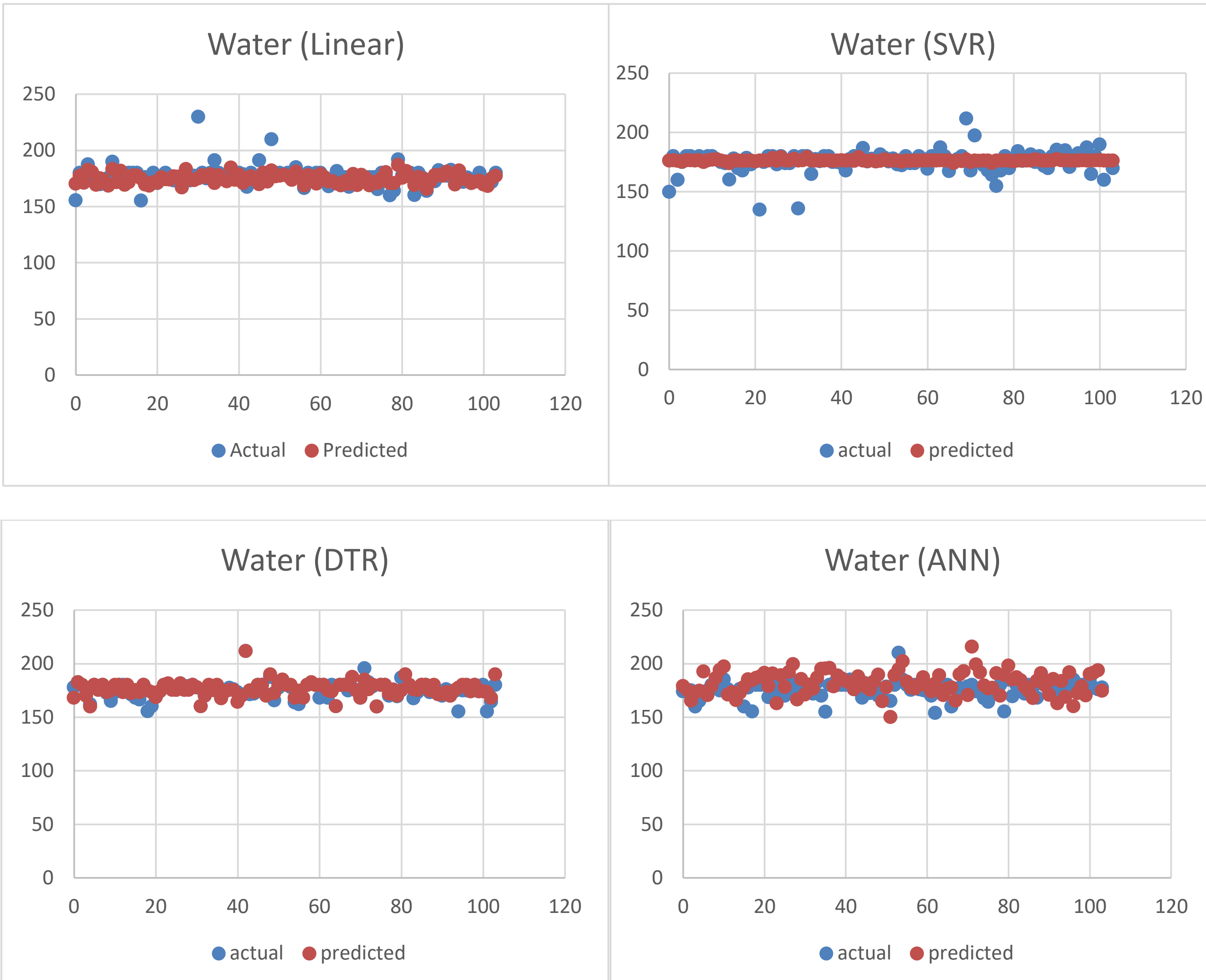

Fig. 1. Water requirement according to various models for designs with plasticizer.

Water (Linear) = 156.09 – 0.323(Grade) + 0.05(Slump) + 8.7(FM 20) – 7.105(FM 10) + 20.678(BD 10) – 0.677(SG 10) + 0.02(Consistency) +0.078(Initial ST) + 0.456(7d Cement) – 0.683(28d cement) + 3.468(UW Cement) – 8.836(SG Cement)

The DTR model predicts quantity of water required in kg per cubic meters with R2 of 0.88. Overall, there is no particular trend of error and the model gives pretty accurate results.

(ii) Cement

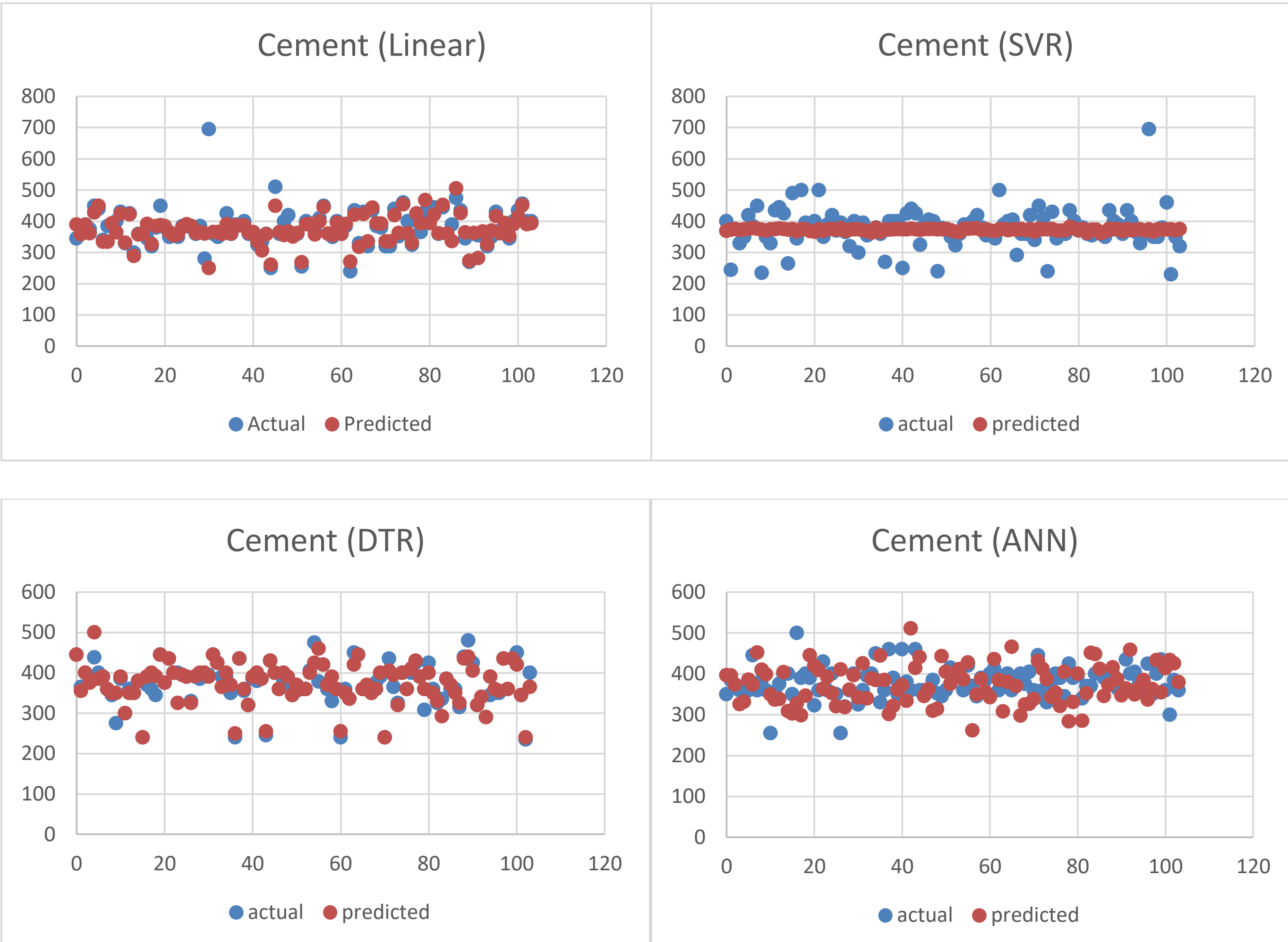

Fig. 2. Cement requirement according to various models for designs with plasticizer.

Cement (linear) = 316.842 + 5.827(Grade) + 0.044(Slump) - 9.242(FM Sand) - 29.696(SG Sand)

The DTR model shows significant variability in cement prediction with R2 value of 0.76.

This is due to the fact that cement from different companies and different types has quite variable properties such as rapid setting cement, sulphate resisting cement, etc. therefore, this is not a very good model.

(iii) Sand

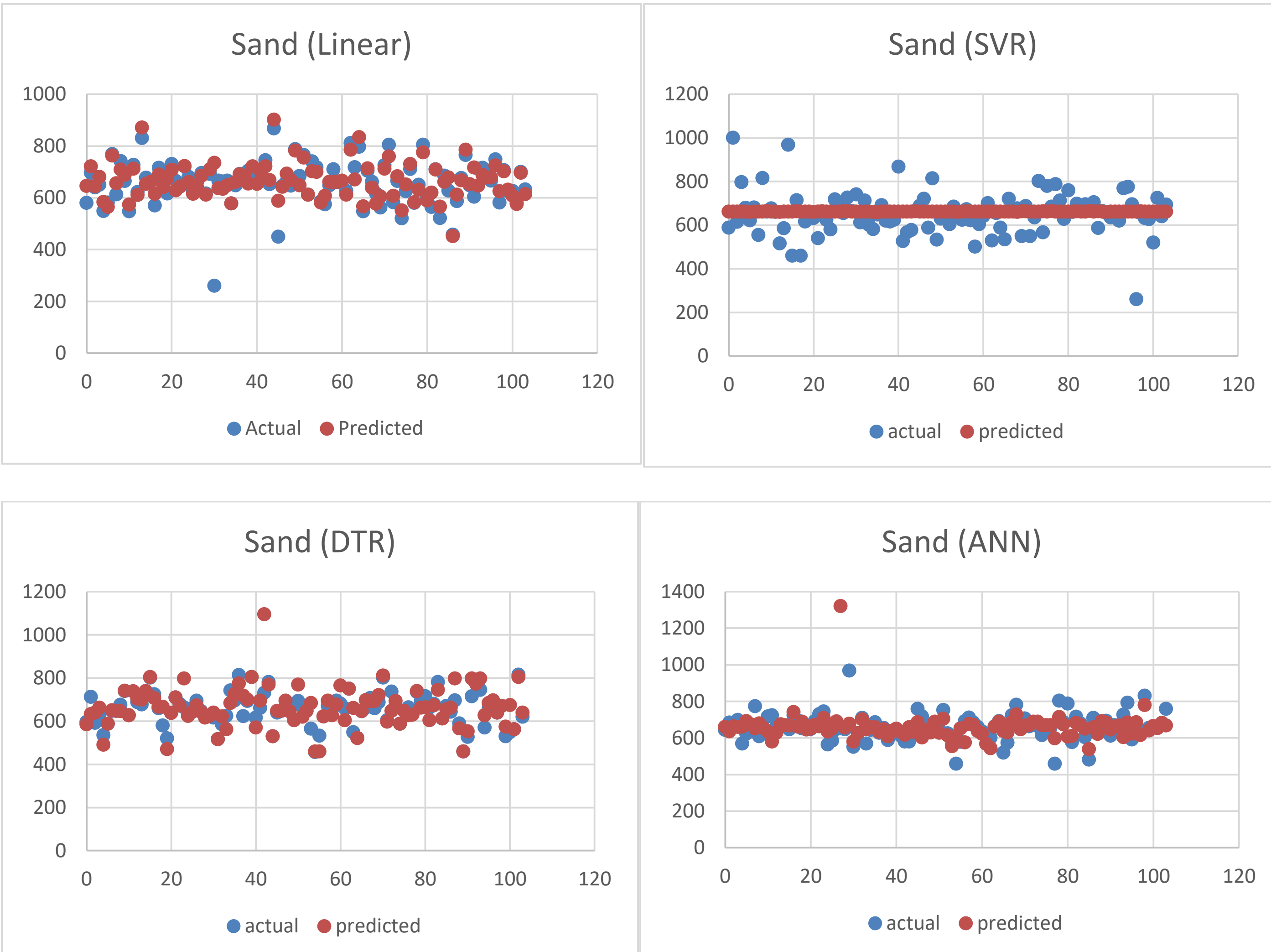

Fig. 3. Sand requirement according to various models for designs with plasticizer.

Sand (linear) = -314.28 – 6.483(Grade) + 0.567(Slump) – 5.803(FM 20) – 19.05(FM 10) + 29(FM Sand) + 78.523(BD 20) – 150.353(BD Sand) – 11.862(SG 10) + 398.98(Sand SG) – 0.156(Initial ST) + 0.052(3d Cement) + 86.058(SG Sand)

The ANN predicts sand required with R2 of 0.81. The model has been trained for mix design containing CA 10, CA20 and sand. We can see that it gave prediction of 652 kg sand for actual value of 1320 kg. This is because the design is of all fines concrete and our model performs poorly for all fines concrete. Similarly, the model cannot be used for no fine concrete too as it has not been trained for it. Other than that, the model performs quite satisfactorily.

(iv) Coarse Aggregate (10 mm)

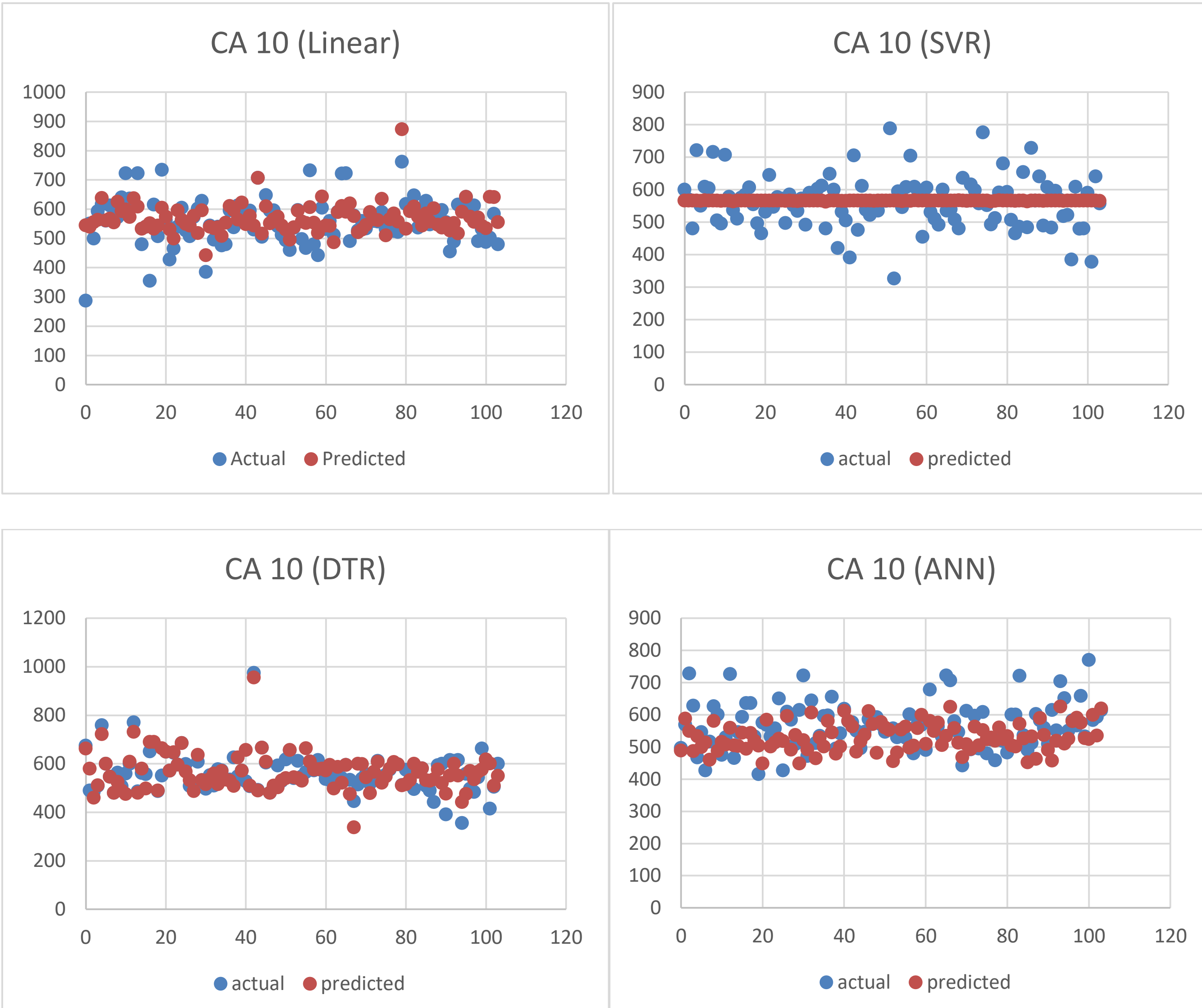

Fig. 4. CA10 requirement according to various models for designs with plasticizer.

CA10 (linear) = -804.77 + 0.757(Grade) + 0.657(Slump) + 50.135(FM 10) + 292.8(BD 10) – 98.71(BD Sand) + 119.367(SG 10) – 0.584(Consistency) – 0.105(Initial ST) + 0.006(Final ST) + 5.436(3d Cement) +142.436(UW Cement) +10.4 (SG Cement)

For mix design, we generally predetermine a ratio of CA10 to CA20 such as 40:60, 45:55 and so on. This affects the amount of CA10 and CA20 used. However in our model, we have not made any such assumption. Therefore, the predicted values differ so much with best fitting DTR model with R2 value of 0.79. The model sometimes overestimates CA10 values.

(v) Coarse Aggregate (20 mm)

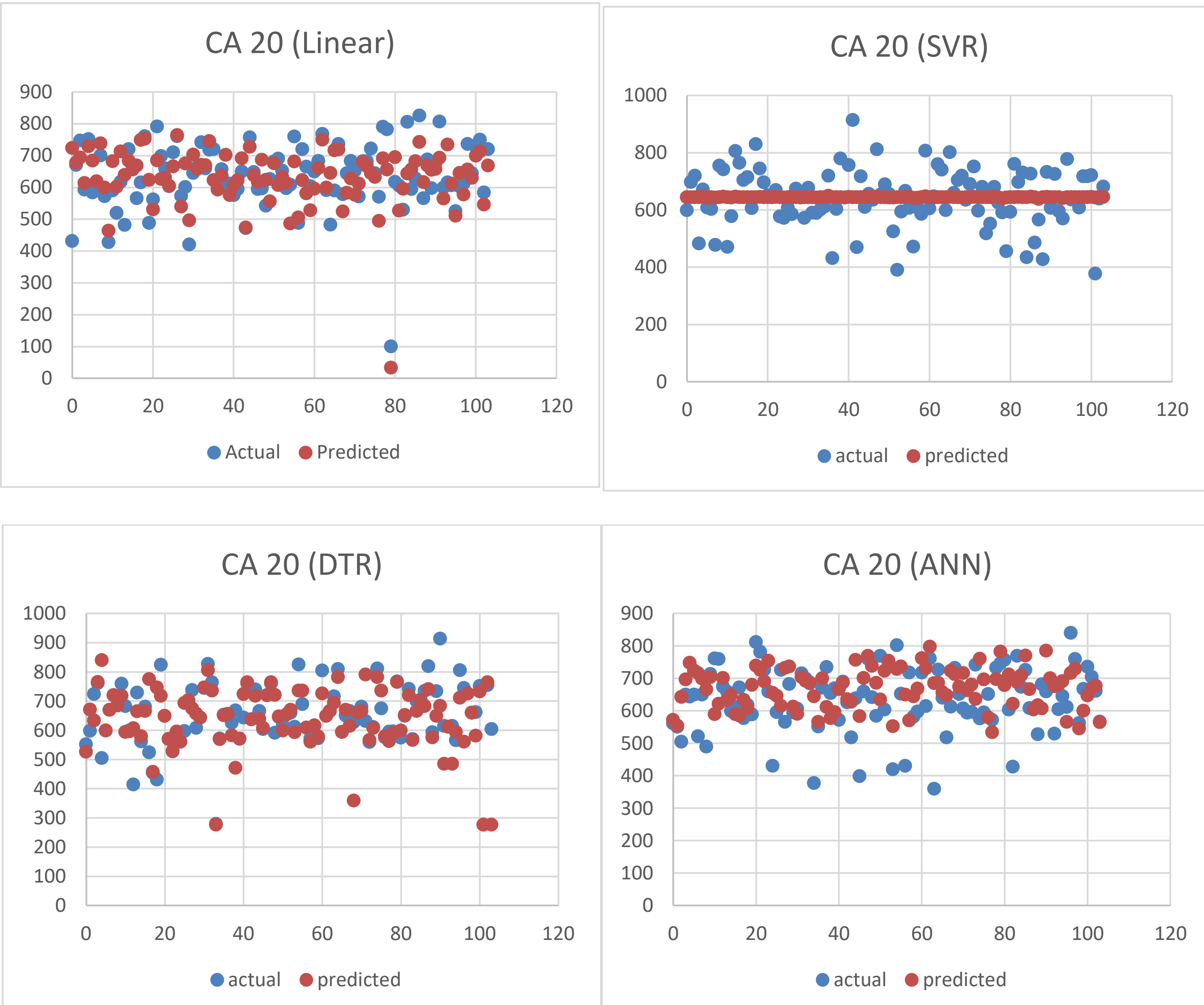

Fig. 5. CA20 requirement according to various models for designs with plasticizer.

CA20 (linear) = -377.17 – 1.288(Slump) + 92.014(FM 20) -72.78(FM 10) – 37.721(FM Sand) + 15.093(BD 20) + 30.89(BD Sand) + 261.463(SG 20) + 78.954(SG 10) – 64.614(WA Sand) – 0.004(Final ST) + 1.263(28d Cement)

The DTR model tends to predict lower values of CA20 with R2 value of just 0.66.

The reason behind this could be the predetermined CA10 to CA20 ratio. This can be verified as CA10 values are overestimated and CA20 values are underestimated.

(vi) Plasticizer

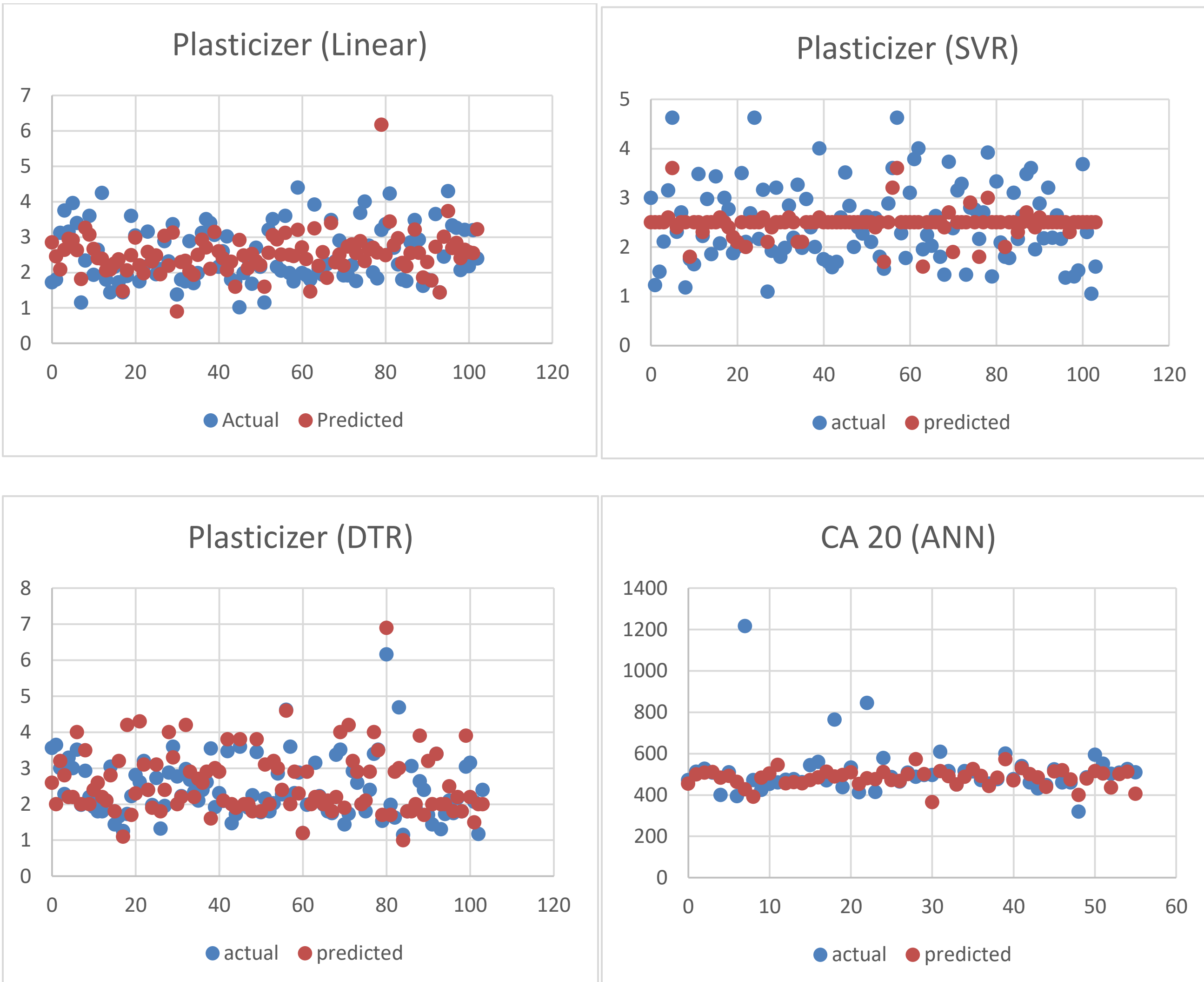

Fig. 6. Plasticizer requirement according to various models.

Plasticizer (linear) = 16.632 + 0.034(Grade) + 0.007(Slump) – 2.043(FM 20) + 0.055(FM 10) + 0.768(BD 20) – 1.304(SG 20) – 0.001(Initial ST) – 0.007(3d Cement) + 0.013(7d Cement)

The DTR model predicts plasticizer required with R2 value of 0.74. This poor prediction can be accounted for by the fact that in our mix design, we have used plasticizers of different generations and different types, some are plasticizer while other are superplasticizer, each requiring different amounts to be used. Since such difference in type of plasticizer has not been not been included in our dataset, the predictions are not excellent.

6. Results of different models without plasticizer added

For all ANN models, the number of hidden layers is 6 and number of nodes is twice the number of features in first 2 layers which decreases to half the value of previous layers nodes for each subsequent layer. For all these nodes, the activation function used is relu. For output layer – the number of nodes is 1 and activation function is linear.

Table 3
R square value for different models for design without Plasticizer.

	Linear	SVR	DTR	ANN
Water	0.28	0.59	0.84	0.81
Cement	0.53	0.47	0.83	0.59
Sand	0.74	0.39	0.91	0.72
CA10	0.45	0.28	0.69	0.73
CA20	0.42	0.46	0.77	0.59

(i) Water

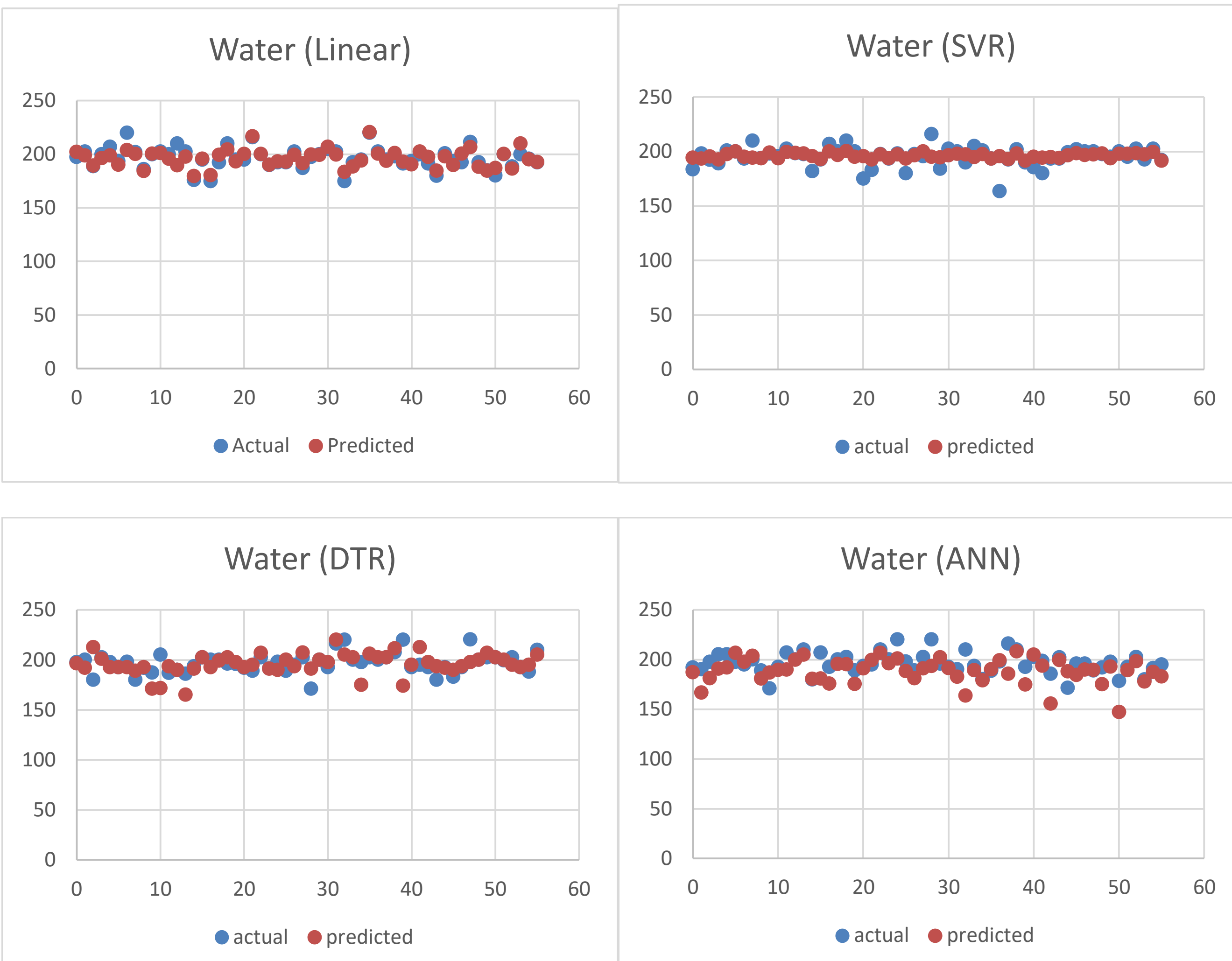

Fig. 7. Water requirement according to various models for designs without plasticizer.

Water (linear) = 227.604 + 0.151(Slump) + 16.682(SG Sand) – 0.64(Consistency) – 23.23(SG Cement)

The DTR model give quite satisfactory water requirement with R2 value of 0.84. The model sometimes gives a little less value of water required, but not always and its results can be used quite accurately.

(ii) Cement

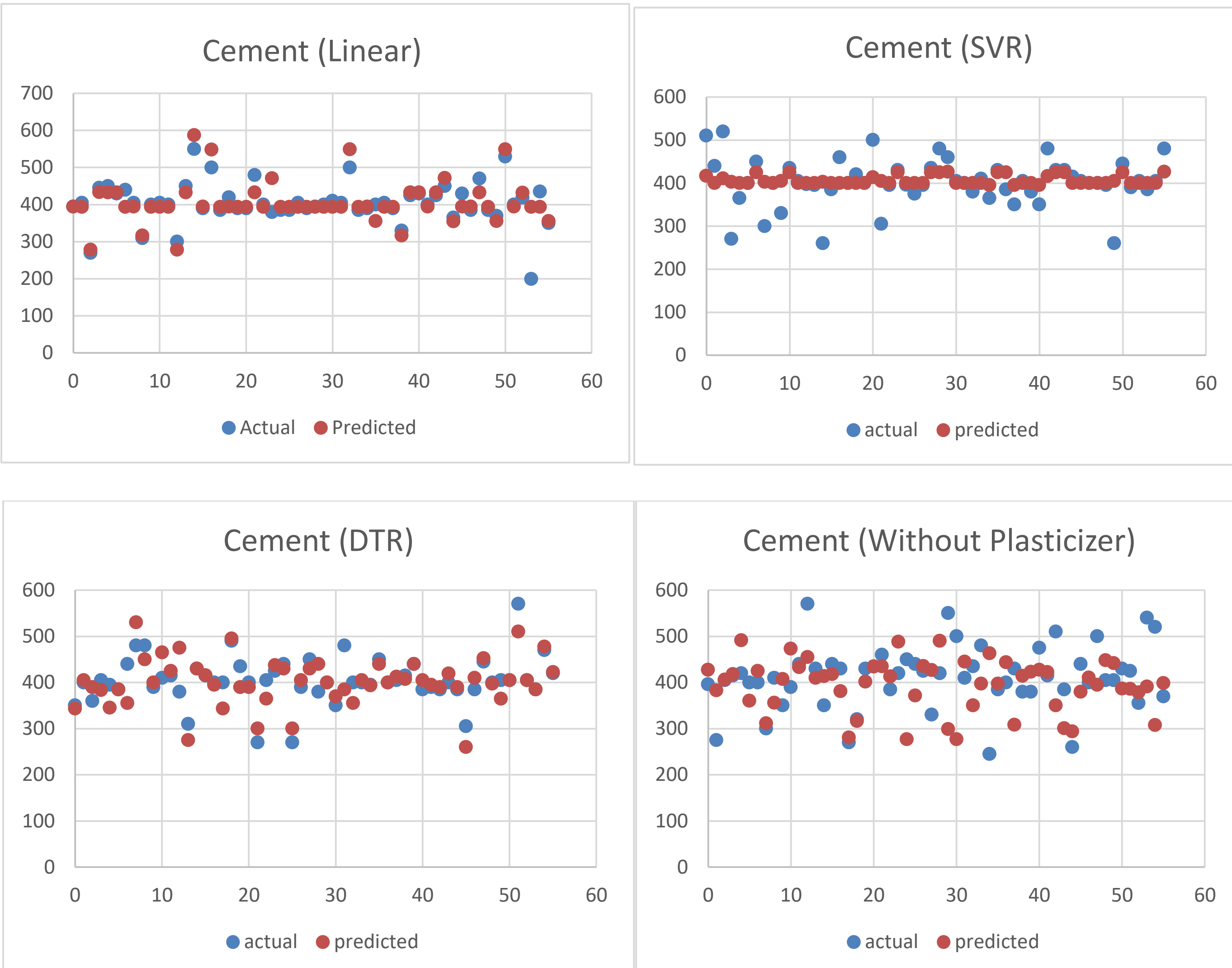

Fig. 8. Cement requirement according to various models for designs without plasticizer.

Cement (linear) = 224.465 + 7.732(Grade) – 3.318(FM 20)

The DTR model performs quite well for prediction of cement with R2 value of 0.83. However,

If cement content comes out to be less than 300 kg, the value cannot be trusted blindly as the model shows error in this region probably due to less training examples in this region. Otherwise for predictions in range of 300 kg to 450 kg, the results can be trusted with confidence.

(iii) Sand

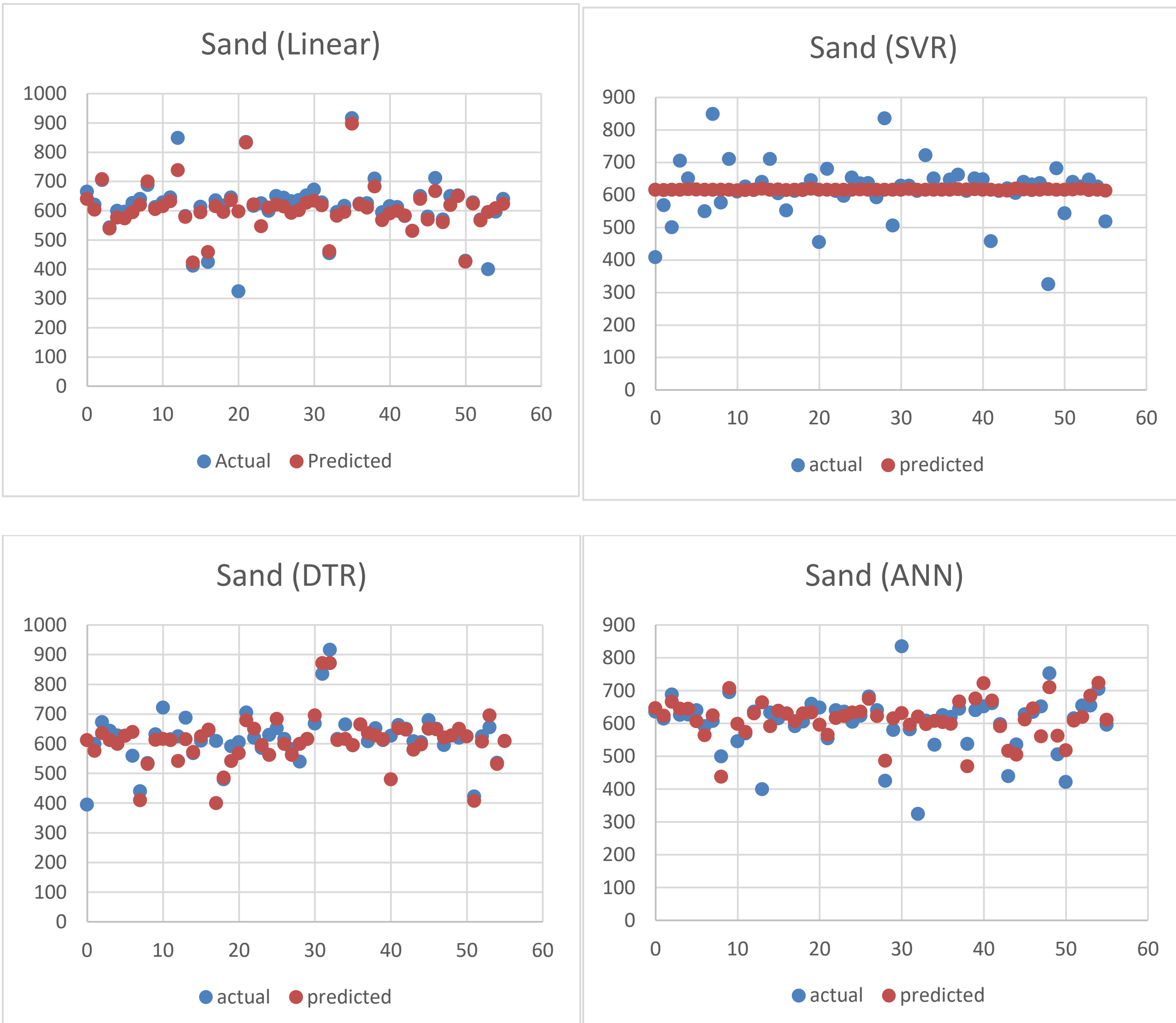

Fig. 9. Sand requirement according to various models for designs without plasticizer.

Sand (linear) = 166.125 – 7.253(Grade) + 43.25(FM Sand) + 197.195(SG Sand) – 0.063(Final ST)

The DTR model predicts sand required with R2 of 0.91. This model also has been trained specifically for mix design consisting of all 3 – CA 10, CA 20 and sand. Therefore, the model performs poorly for no fines or all fines concrete. Apart from these, the model gives the value of sand required which can be used with confidence.

(iv) Coarse Aggregate (10 mm)

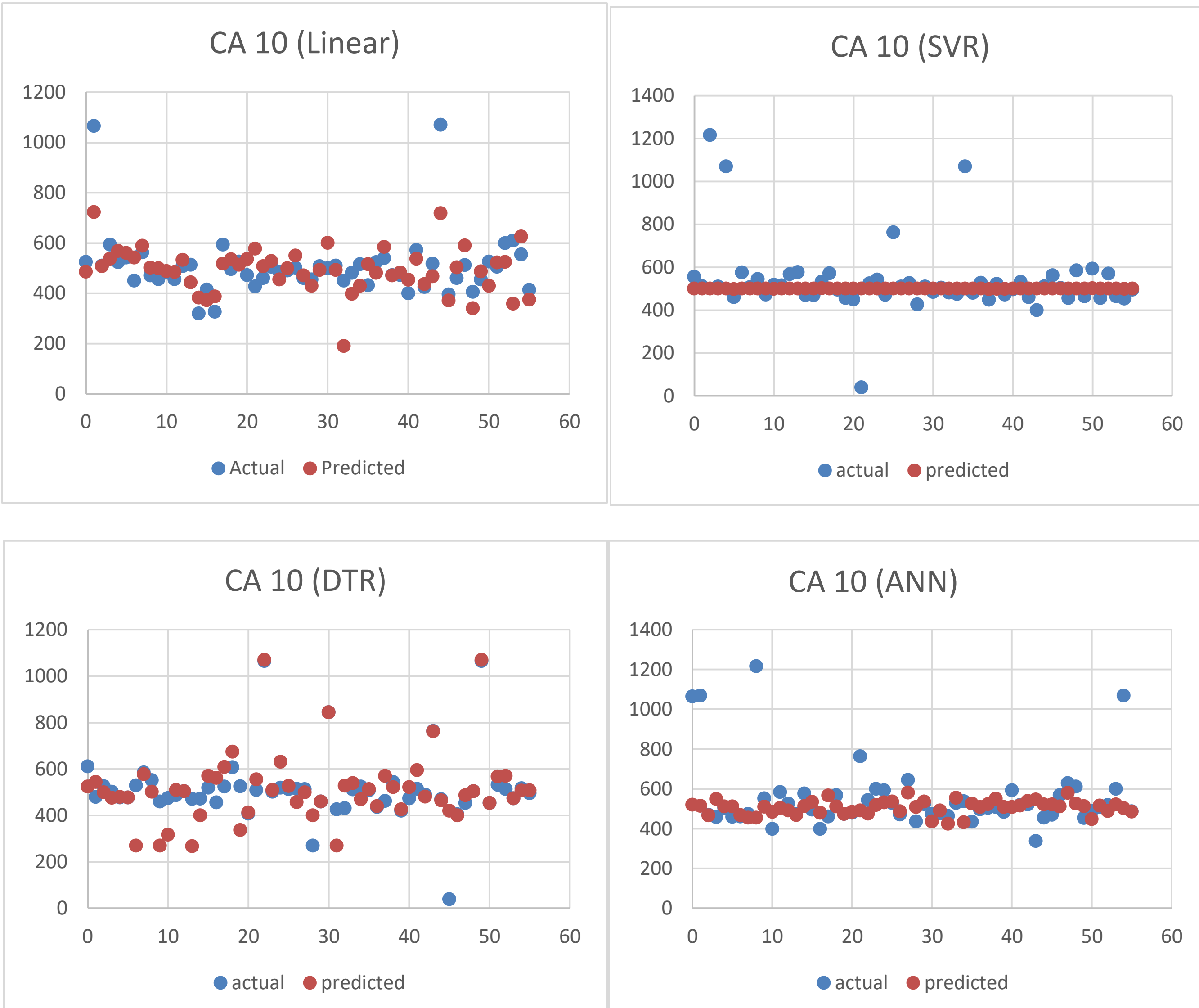

Fig. 10. CA10 requirement according to various models for designs without plasticizer.

CA10 (linear) = -2668.635 + 0.973(Slump) + 6(FM 20) + 261.35(FM 10) – 405.61(BD 20) +509.434(SG 10) + 1.567(Consistency) + 0.473(Initial ST) – 0.4(Final ST)

The ANN has a trend to slightly overestimate CA10 values with R2 value of 0.73. In some designs where only CA10 or CA12.5 were used, the predictions were significantly less than original values. Also, the error is due to predetermined CA10 to CA20 ratio, which changes with each design.

(v) Coarse Aggregate (20 mm)

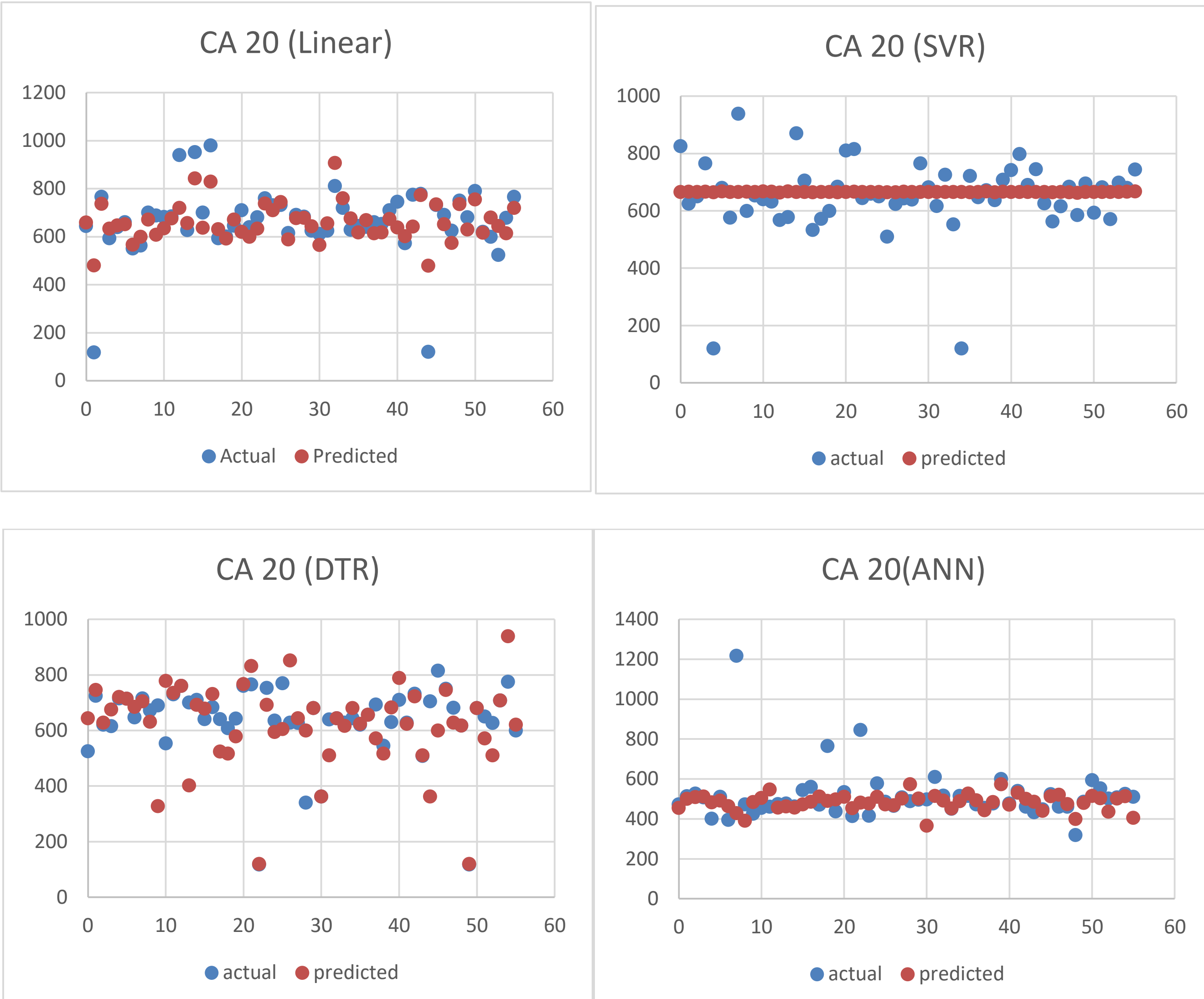

Fig. 11. CA20 requirement according to various models for designs without plasticizer.

CA20 (linear) = 1369.272 + 1.466(Grade) – 1.22(Slump) – 211.321(FM 10) + 0.261(SG 20) + 185(SG 10) – 1353.25(WA 20) + 1575.735(WA 10) – 698.7(WA Sand) + 1.143(Initial ST)

The DTR model tends to predict lower values of CA20 with R2 value of just 0.77.

The reason behind this could be the predetermined CA10 to CA20 ratio. This can be verified as CA10 values are overestimated and CA20 values are underestimated.

7. Conclusions

The objective of this paper was to first - compare and contrast between the results obtained by all 4 machine learning algorithms and find out the best performing model and then finally – check if the best performing models are able to replace the convention method of calculating the requirements of ingredients of mix. For the first objective, the best performing models have been given in results section (5 and 6). This has been summarized below in Table 4.

For second objective – could these prediction models replace the conventional design method – we observe that for water and sand requirements, the best performing models can actually replace the conventional design methods. For coarse aggregates, the best performing model over-estimates coarse aggregate of size 10 mm and underestimates the course aggregate size of 20 mm. Thus, as a whole, if we take values of both CA10 and CA20 from these models, combined course aggregate value can also be used with confidence. As for cement, the required amount of cement is not predicted with high accuracy for designs with plasticizers even with our best performing model. This is due to the fact that our dataset consists of huge variety of cement of different manufacturers with different properties. This variation in types of cement is predominant in designs with plasticizers and not in designs without plasticizers and hence, for without plasticizer designs – our best performing model can replace the conventional method. As for design with plasticizers, the variation in properties of cement was too much which resulted in poor performance of our model. However, if we do away with this variation, our model would be ready to replace conventional method. For plasticizer also, even our best performing model could not give satisfactory results. This can be accounted for by the fact that in our mix design, we have used plasticizers of different generations and different types, some are plasticizer while other are superplasticizer, each requiring different amounts to be used. Since such difference in type of plasticizer has not been not been included in our dataset, the predictions are not excellent.

Table 4
Conclusion for prediction results by best performing model for design with Plasticizer.

S. No.	Ingredient	Best performing model	Can replace conventional design method
1	Water	DTR	Yes
2	Cement	DTR	No
3	Sand	ANN	Yes
4	Coarse Aggregate 10mm	DTR	Yes only when both results are used from this model
5	Coarse Aggregate 20mm	DTR	
6	Plasticizer	DTR	No

Table 5
Conclusion for prediction result by best performing model for design without Plasticizer.

S. No.	Ingredient	Best performing model	Can replace conventional design method
1	Water	DTR	Yes
2	Cement	DTR	Yes
3	Sand	DTR	Yes
4	Coarse Aggregate 10mm	ANN	Yes only when both results are used from this model
5	Coarse Aggregate 20mm	DTR	

Overall, we saw that DTR models are best for predicting the composition of a concrete mix without too much hassles for fine tuning the hyper parameters and are computationally efficient too.

In literature review, we saw SVR models were used previously and after creating our own models, we compared the efficiency of SVR model with DTR, linear model and ANN. And from the obtained results, we can undoubtedly say that DTR gives a way better prediction.

Therefore, use of tree based models for predicting the composition of concrete mix is highly recommended by the results of this paper.

References

[1] Antoniou C, Dimitriou L, Pereira F. Mobility patterns, big data and transport analytics: tools and applications for modeling. Elsevier; 2019.

[2] Abrams DA. Design of concrete mixtures. Structural Materials Research Laboratory. Bulletin 1918;1.

[3] Kasperkiewicz J, Racz J, Dubrawski A. HPC Strength Prediction Using Artificial Neural Network. J Comput Civ Eng 1995;9:279–84. doi:10.1061/(ASCE)0887-3801(1995)9:4(279).

[4] Yeh I-C. Modeling of strength of high-performance concrete using artificial neural networks. Cem Concr Res 1998;28:1797–808. doi:10.1016/S0008-8846(98)00165-3.

[5] Nehdi M, El Chabib H, El Naggar MH. Predicting performance of self-compacting concrete mixtures using artificial neural networks. Mater J 2001;98:394–401.

[6] Siddique R, Aggarwal P, Aggarwal Y. Prediction of compressive strength of self-compacting concrete containing bottom ash using artificial neural networks. Adv Eng Softw 2011;42:780–6. doi:10.1016/j.advengsoft.2011.05.016.

[7] Dias WPS, Pooliyadda SP. Neural networks for predicting properties of concretes with admixtures. Constr Build Mater 2001;15:371–9. doi:10.1016/S0950-0618(01)00006-X.

[8] Öztaş A, Pala M, Özbay E, Kanca E, Çagˇlar N, Bhatti MA. Predicting the compressive strength and slump of high strength concrete using neural network. Constr Build Mater 2006;20:769–75. doi:10.1016/j.conbuildmat.2005.01.054.

[9] Ghafari E, Bandarabadi M, Costa H, Júlio E. Prediction of Fresh and Hardened State Properties of UHPC: Comparative Study of Statistical Mixture Design and an Artificial Neural Network Model. J Mater Civ Eng 2015;27:04015017. doi:10.1061/(ASCE)MT.1943-5533.0001270.

[10] Topçu İB, Sarıdemir M. Prediction of properties of waste AAC aggregate concrete using artificial neural network. Comput Mater Sci 2007;41:117–25. doi:10.1016/j.commatsci.2007.03.010.

[11] Alshihri MM, Azmy AM, El-Bisy MS. Neural networks for predicting compressive strength of structural light weight concrete. Constr Build Mater 2009;23:2214–9. doi:10.1016/j.conbuildmat.2008.12.003.

[12] Loh W-Y. Fifty Years of Classification and Regression Trees. Int Stat Rev 2014;82:329–48. doi:10.1111/insr.12016.

[13] Erdal HI. Two-level and hybrid ensembles of decision trees for high performance concrete compressive strength prediction. Eng Appl Artif Intell 2013;26:1689–97. doi:10.1016/j.engappai.2013.03.014.

[14] Chou J-S, Chiu C-K, Farfoura M, Al-Taharwa I. Optimizing the Prediction Accuracy of Concrete Compressive Strength Based on a Comparison of Data-Mining Techniques. J Comput Civ Eng 2011;25:242–53. doi:10.1061/(ASCE)CP.1943-5487.0000088.

[15] Deshpande N, Londhe S, Kulkarni S. Modeling compressive strength of recycled aggregate concrete by Artificial Neural Network, Model Tree and Non-linear Regression. Int J Sustain Built Environ 2014;3:187–98. doi:10.1016/j.ijsbe.2014.12.002.

[16] Deepa C, SathiyaKumari K, Sudha VP. Prediction of the compressive strength of high performance concrete mix using tree based modeling. Int J Comput Appl 2010;6:18–24.

[17] Mansouri I, Ozbakkaloglu T, Kisi O, Xie T. Predicting behavior of FRP-confined concrete using neuro fuzzy, neural network, multivariate adaptive regression splines and M5 model tree techniques. Mater Struct 2016;49:4319–34. doi:10.1617/s11527-015-0790-4.

[18] Ayaz Y, Kocamaz AF, Karakoç MB. Modeling of compressive strength and UPV of high-volume mineral-admixtured concrete using rule-based M5 rule and tree model M5P classifiers. Constr Build Mater 2015;94:235–40. doi:10.1016/j.conbuildmat.2015.06.029.

[19] Vapnik V. A note one class of perceptrons. Autom Remote Control 1964.

[20] Boser BE, Guyon IM, Vapnik VN. A training algorithm for optimal margin classifiers. Proc. fifth Annu. Work. Comput. Learn. theory - COLT '92, New York, New York, USA: ACM Press; 1992, p. 144–52. doi:10.1145/130385.130401.

[21] Gupta SM. Support vector machines based modelling of concrete strength. World Acad Sci Eng Technol 2007;36:305–11.

[22] Yan K, Shi C. Prediction of elastic modulus of normal and high strength concrete by support vector machine. Constr Build Mater 2010;24:1479–85. doi:10.1016/j.conbuildmat.2010.01.006.

[23] Cheng M-Y, Chou J-S, Roy AFV, Wu Y-W. High-performance Concrete Compressive Strength Prediction using Time-Weighted Evolutionary Fuzzy Support Vector Machines Inference Model. Autom Constr 2012;28:106–15. doi:10.1016/j.autcon.2012.07.004.

[24] Siddique R, Aggarwal P, Aggarwal Y, Gupta SM. Modeling properties of self-compacting concrete: support vector machines approach. Comput Concr 2008;5:123–9.

[25] Aiyer BG, Kim D, Karingattikkal N, Samui P, Rao PR. Prediction of compressive strength of self-compacting concrete using least square support vector machine and relevance vector machine. KSCE J Civ Eng 2014;18:1753–8. doi:10.1007/s12205-014-0524-0.

[26] Yan K, Xu H, Shen G, Liu P. Prediction of splitting tensile strength from cylinder compressive strength of concrete by support vector machine. Adv Mater Sci Eng 2013;2013.

[27] Naderpour H, Rafiean AH, Fakharian P. Compressive strength prediction of environmentally friendly concrete using artificial neural networks. J Build Eng 2018;16:213–9. doi:10.1016/j.jobe.2018.01.007.