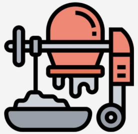
Concrete Compressive Strength Regression Problem With Machine Learning

SDSC3006: Fundamentals of Machine Learning I

Ha Quang Minh 57040174

Date: 27/11/2022

Background



What is Concrete?
Most widely used building materials in the world

Application of Concrete:
Concrete Dams
Residential Building
Roads or Driveways
Culverts & Sewers

Problem

Prediction:
Ingredients and age are highly non-linear
→ Difficult to establish an analytical formula and a perfect model for prediction

Interpretation:
The relationship between the strength and the input variables is weak (< 0.5)
→ Difficult to find which variables are essential in determining concrete compressive strength

What is Concrete Compressive Strength?
The capacity of concrete to withstand loads before failure

Is it Important?

Strength & Durability
Represent the ability of concrete to support heavy structures over long periods of time

Advantage of Solving Problem:

- Increasing economic benefit
  - Lower material cost
    - → Higher strength
- Quality control
  - Tests of International Standards that consist of the breaking of specimens
  - Less time-consuming & Low human-effort
    - The model could be used as the first filtering to separate the unqualified cases

Data Pre-processing and Cleaning

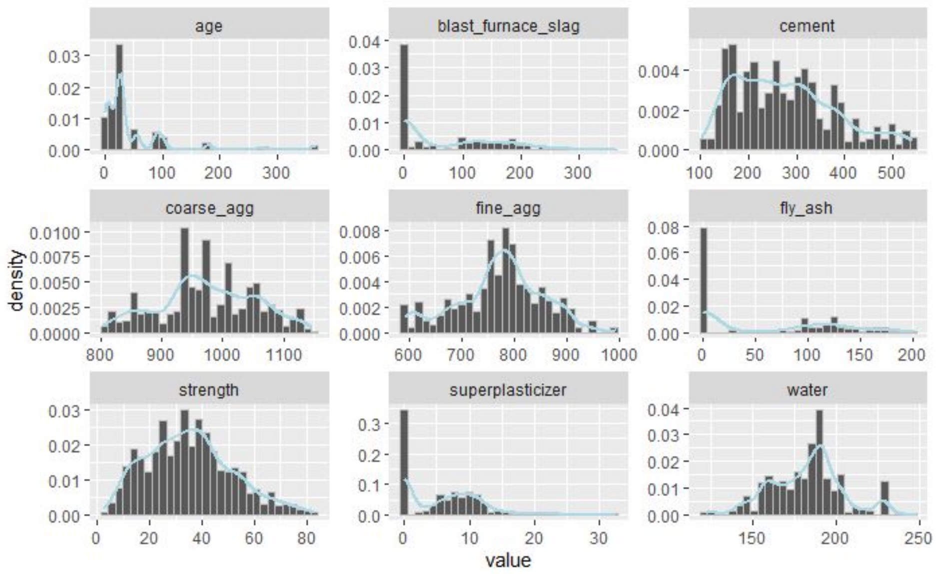
We remove 25 duplicate observations from our dataset. During finding the duplicate instances, we realize that some samples are identical in proportions of all features, except for the compressive strength

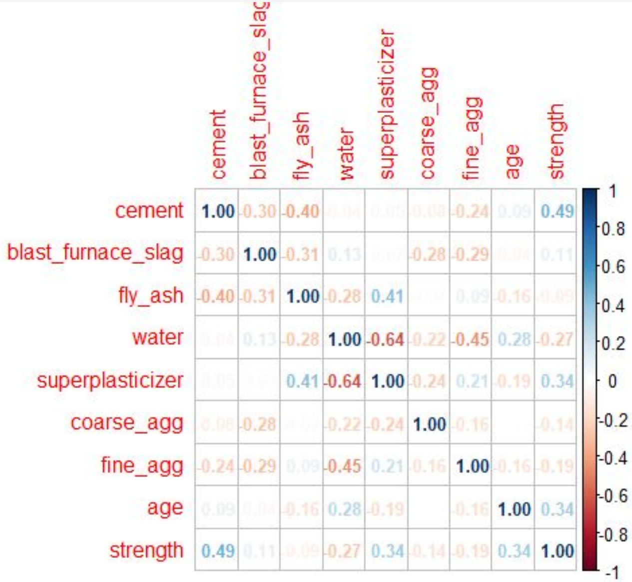
	cement	blast_furnace_slag	fly_ash	water	superplasticizer	coarse_agg	fine_agg	age	strength
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
100	362.6	189	0	164.9	11.6	944.7	755.8	7	55.90
106	362.6	189	0	164.9	11.6	944.7	755.8	7	22.90
448	446.0	24	79	162.0	11.6	967.0	712.0	28	57.03
449	446.0	24	79	162.0	11.6	967.0	712.0	28	44.42
450	446.0	24	79	162.0	11.6	967.0	712.0	28	51.02
452	446.0	24	79	162.0	11.6	967.0	712.0	3	35.36

This is probably due to differences in the building process, hence we assign all of the samples with similar features the same id and calculate their mean compressive strength. We only keep 992 observations for further analysis.

Split our dataset into 2 parts: 80% is for training data and 20% is for test data for evaluating our models.

Data Visualizations about Predictors and Response





- Some of the features have bell-curve distribution such as fine aggregate and water, while the others are heavily right-skewed
- Target response compressive strength has a bell-curve with small skewness, which is good for our models’ performance later
- None of the features have strong correlations with the target strength, however we can see that water and superplasticizer or fine aggregate are slightly correlated → multicollinearity

Method & Justification

Features Selections for Linear Models with Detect Multicollinearity, Best Subset Selection with 10-fold CV, and Shrinkage Methods

Regression with Tree-based Methods and Comparing Results with and without Feature Selections

Ensemble Models and Tuning Hyperparameters to Achieve Better Performance

Features Selection: Least Squares

We conduct OLS for our target with all of the features in our training dataset. Two components coarse and fine aggregate have p-value ≈ 0.27 > 0.05 level of significance. And the VIF values of our features are extremely high.

	vif.lm.fit. <dbl>
cement	7.188423
blast_furnace_slag	7.200445
fly_ash	5.813000
water	6.584524
superplasticizer	2.763241
coarse_agg	4.885534
fine_agg	6.663086
age	1.127499

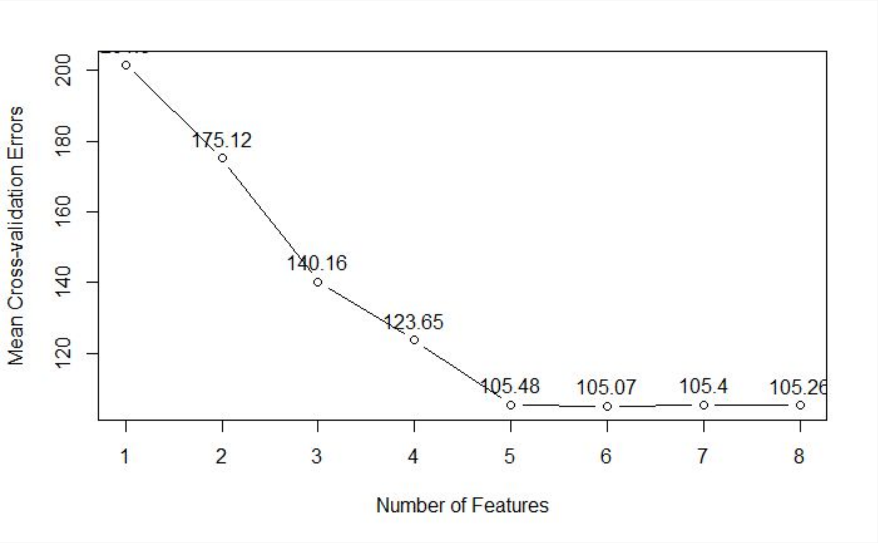
Hence, we remove those 2 features and conduct OLS again for the remaining attributes. All of VIF values now are smaller than 5. Or we also can combine the collinear variables together into a single predictor and make our OLS model better.

	vif.lm.fit1. <dbl>
cement	1.868412
blast_furnace_slag	1.779754
fly_ash	2.310474
water	1.860285
superplasticizer	2.296751
age	1.114977

Alternative fitting procedures can yield better *prediction accuracy* and *model interpretability.* We will check them to see the differences with least squares fitting results.

Features Selection: Subset Selection and Shrinkage Methods

Best subset selection with 10-fold CV
6-variable model has the smallest cross-validation errors. Coarse and fine aggregate are excluded




```
      (Intercept)            cement blast_furnace_slag            fly_ash              water
      26.23355282        0.10603855         0.08196717         0.06966380        -0.20031695
superplasticizer               age
       0.27341647        0.10425305
```

The LASSO
Coefficient estimates of coarse and fine aggregate are also shrunk to zero

```
10 x 1 sparse Matrix of class "dgCMatrix"
                              s1
(Intercept)          26.28502028
(Intercept)           .
cement                0.10320448
blast_furnace_slag    0.07846224
fly_ash               0.06451969
water                -0.19356489
superplasticizer      0.29684840
coarse_agg            .
fine_agg              .
age                   0.10216140
```

→ The results are the same as OLS and multicollinearity suggestion

Tree-based Models: Training and Test Results

Training RMSE and R^2 of all methods are becoming worse when we exclude coase_agg and fine_agg. However, the test results are better with 3 models: Decision Trees, Bagging, and Random Forest

→ Can **consider** about linear models' features selection, in case there's lots of predictors and some of them are highly correlated or unimportant

Table: Without Features Selection

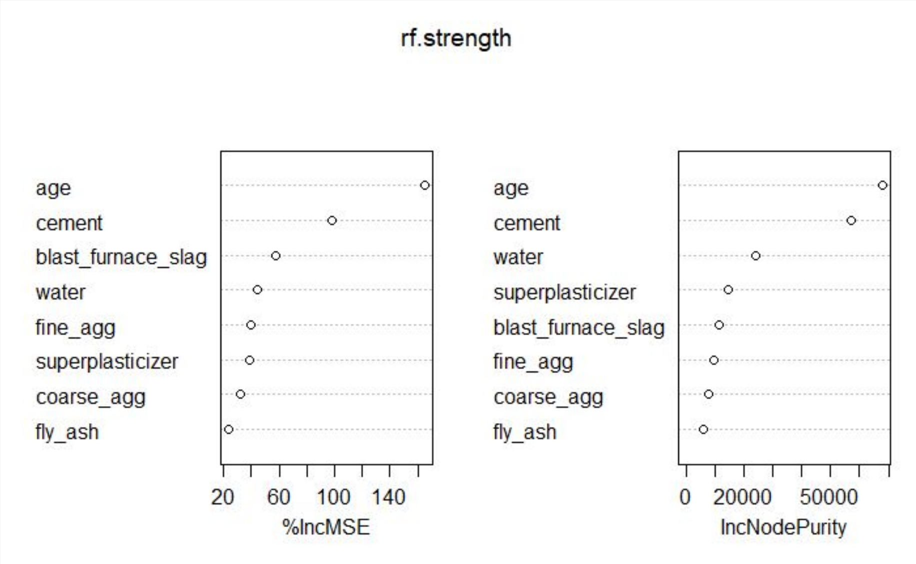
Methods	Training RMSE	Training R^2	Test RMSE	Test R^2
Decision Trees	7.8258	0.7585	10.1323	0.6614
Bagging	2.1559	0.9816	5.5099	0.8998
Random Forest	2.2161	0.9806	5.3012	0.9073
Boosting	3.4950	0.9518	4.6068	0.9300
BART	2.6022	0.9733	4.0348	0.9463

Table: With Features Selection

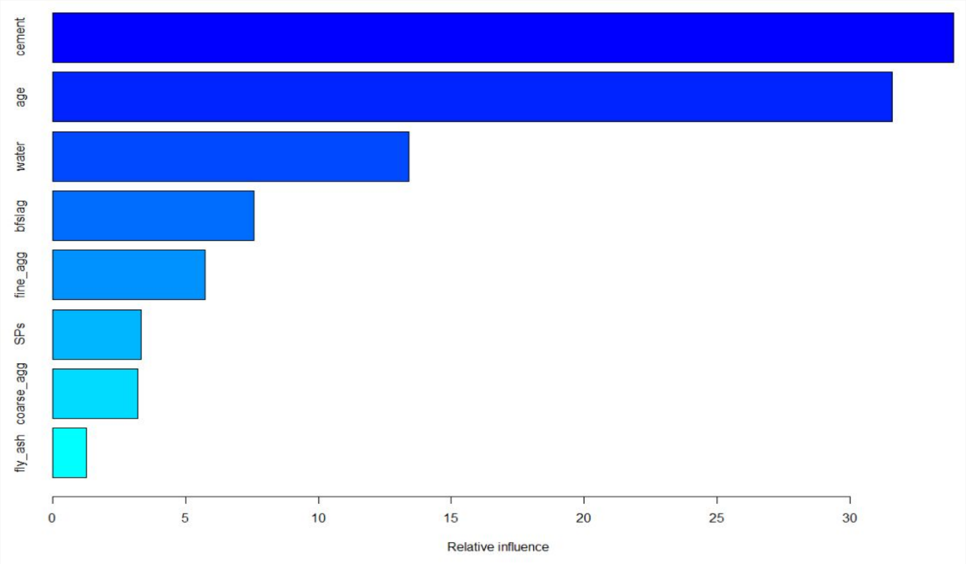
Methods	Training RMSE	Training R^2	Test RMSE	Test R^2
Decision Trees	7.9883	0.7484	10.0217	0.6687
Bagging	2.2929	0.9792	5.2286	0.9098
Random Forest	2.3664	0.9779	5.2204	0.9101
Boosting	3.9412	0.9387	4.7714	0.9249
BART	3.0293	0.9638	4.2023	0.9417

Tree-based Methods: Features Importance

Random Forest



Boosting



The two most important features in our tree-based models with all of predictors are age and cement, outperforming the other predictors with double the importance

While fine aggregate and fly ash are the two least important features

Ensemble Models with Tuning Hyperparameters

The ensemble models we use are: **AdaBoost** ,**GBM (Gradient Boosting Machine)**, **LightGBM**, **XGBoost (eXtreme Gradient Boosting),** and **CatBoost**.

Before tuning the hyperparameters, CatBoost is outperform all the opponents for the test RMSE and R^2, however XGBoost is the best model for fitting our training data with 0.9988 R^2 value (mostly overfitting)

After we tune the hyperparameters of the models with 5-fold CV, all of them have better results for both training and test RMSE with R^2 score except XGBoost training performance slightly decreases. However, there is an increase in test results, which means overfitting is reduced

Comparing to the best tree-based models, BART surpass AdaBoost, GBM, and LightGBM in test RMSE and R^2, but CatBoost and XGBoost are still the two best models.

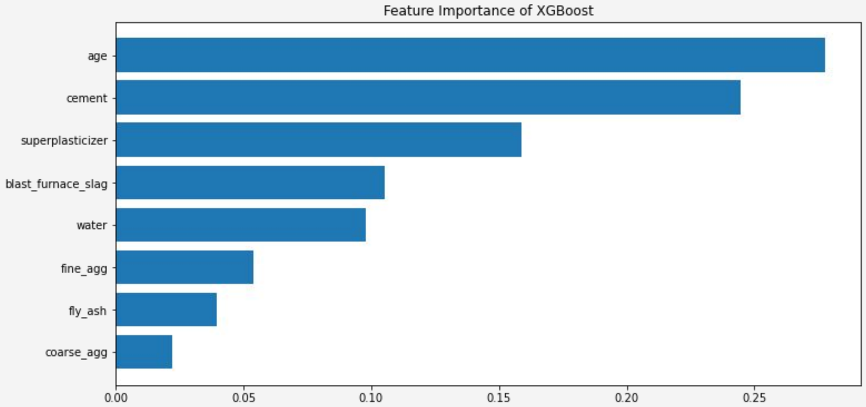
Table: Before Tuning Hyperparameters

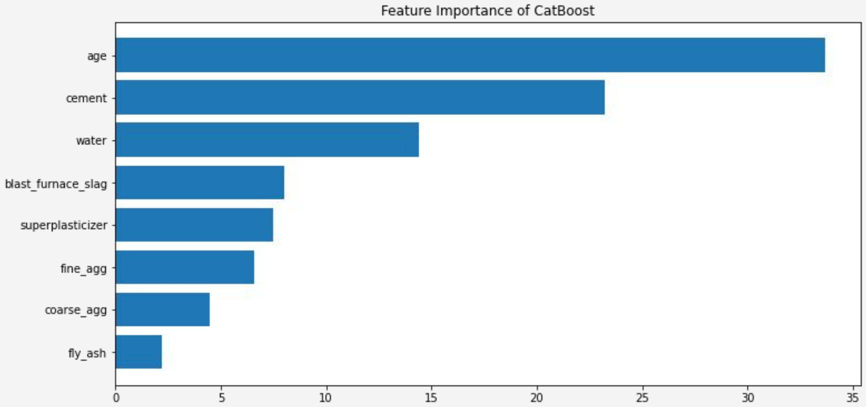
Methods	Training RMSE	Training R^2	Test RMSE	Test R^2
AdaBoost	5.6714	0.8731	7.4087	0.8189
GBM	3.6685	0.9469	5.1705	0.9118
LightGBM	2.0871	0.9828	4.6759	0.9278
XGBoost	0.5289	0.9988	4.0507	0.9458
CatBoost	1.5816	0.9901	3.6388	0.9563

Table: After Tuning Hyperparameters with 5-fold CV

Methods	Training RMSE	Training R^2	Test RMSE	Test R^2
AdaBoost	5.2782	0.8901	6.7973	0.8476
GBM	0.8706	0.9970	4.0800	0.9450
LightGBM	1.0927	0.9952	4.1763	0.9424
XGBoost	1.3934	0.9923	3.9167	0.9494
CatBoost	1.2091	0.9942	3.6158	0.9568

Ensemble Models: Features Importance





Age and cement are still the two most important variables in XGBoost and CatBoost, meanwhile fly ash is still the least important feature in our ensemble models.

Conclusion & Discussion

- Linear models’ features selection can help non-linear models like Decision Trees, Bagging, and Random Forest improve their performance. For the prediction in future, we can consider about this for our ensemble models since we may have more predictors for our response compressive strength and highly correlated or unimportant features should be removed.

- For interpretation, age and cement are the two most important variables in helping increase prediction accuracy of our ensemble models, while fly ash seems to be the least important feature.

- For prediction accuracy, with the tree-based models we learnt in our course, Boosting and BART seems to be the two best models for regression problems. However, they cannot beat the strong gradient boosting algorithms models like XGBoost and CatBoost.

- Tuning hyperparameters of ensemble models can help increasing the prediction accuracy as well as reducing the overfitting with our training dataset.

- The models built present satisfactory results and prove that the compressive strength of concrete can be predicted relatively easily. CatBoost with tuned hyperparameters gives us lowest test RMSE = 3.6158 and highest test R^2 = 0.9568. For further prediction, we can use this tuned model or the blending of two best models, XGBoost and CatBoost since their performance are just slightly different.