TẠP CHÍ KHOA HỌC & CÔNG NGHỆ ĐẠI HỌC DUY TÂN
DTU Journal of Science and Technology 5(60) (2023) 39-43

TRƯỜNG ĐẠI HỌC DUY TÂN
DUYTAN UNIVERSITY

Linear regression models for predicting the compressive strength of rice husk ash-blended concrete

Ứng dụng các mô hình hồi quy tuyến tính cho việc dự báo cường độ chịu nén của bê tông có chứa tro trấu

Hoang Nhat Duc[a,b*], Nguyen Quoc Lam[b]
Hoàng Nhật Đức[a,b*] , Nguyễn Quốc Lâm[b]

[a]Institute of Research and Development, Duy Tan University, Da Nang, 550000, Vietnam
[a]Viện Nghiên cứu và Phát triển Công nghệ Cao, Đại học Duy Tân, Đà Nẵng, Việt Nam
[b]Faculty of Civil Engineering, Duy Tan University, Da Nang, 550000, Vietnam
[b]Khoa Xây dựng, Trường Đại học Duy Tân, Đà Nẵng, Việt Nam

(Ngày nhận bài: 20/02/2023, ngày phản biện xong: 23/4/2023, ngày chấp nhận đăng: 09/5/2023)

Abstract

This paper constructs linear regression models for estimating the compressive strength (CS) of rice husk ash (RHA)-blended concrete. The conventional multiple linear regression model and multivariate power function-based model are employed. Experimental results show that the performance of the latter is better than that of the former. The multivariate power function-based regression model can achieve good prediction results with a mean absolute percentage error of 14%. This model can provide explanation for roughly 84% of the sample variation in the CS. Prediction intervals are also computed in addition to point estimations of the CS. The models are coded in Python to support their implementations.

Keywords: Linear models, regression analysis, rice husk ash, concrete strength, Python.

Tóm tắt

Bài báo này xây dựng mô hình hồi quy tuyến tính để ước lượng cường độ chịu nén (CS) của bê tông trộn tro trấu. Mô hình hồi quy tuyến tính truyền thống và mô hình dựa trên hàm mũ đa biến đã được sử dụng. Kết quả tính toán cho thấy mô hình sử dụng hàm mũ đa biến cho kết quả tốt hơn mô hình hồi quy thông thường. Mô hình hồi quy này có thể đạt được kết quả dự báo tốt với sai số phần trăm tuyệt đối trung bình là 14%. Mô hình này có thể giải thích khoảng 84% dao động của cường độ chịu nén của bê tông. Kết quả dự đoán theo khoảng cũng được tính hợp trong mô hình sử dụng hàm mũ đa biến. Các mô hình này đã được xây dựng với Python để có thể được sử dụng một cách thuận tiện.

Từ khóa: Mô hình tuyến tính, phân tích hồi quy, tro trấu, cường độ bê tông, Python.

1. Introduction

Rice husk ash (RHA) is a very attractive partial replacement for ordinary cement in agricultural producing countries. This material has been shown to have a high silica content, high pozzolanic activity, and a smaller carbon footprint compared to ordinary Portland cement [1]. Estimating the CS is a crucial task in concrete mix design. Based on records of testing results, it is able to establish data-driven

[*] *Corresponding Author:* Hoang Nhat Duc
Email: hoangnhatduc@duytan.edu.vn

approaches to predict this mechanical parameter of RHA-blended concrete mixtures. These approaches can immensely help reduce time and effort required in sample casting and testing procedures.

This study focuses on multiple linear regression (MLR) models to construct data-driven methods for the task of interest. Regression analysis generally refers to the process of establishing a mathematical model that best fits a set of collected data samples [4]. Herein, the modeled variable or response variable is the CS. The variables, including the mixture's component and concrete age, are used as independent variables. In addition, this study also relies on the MLR that is based on a multivariate power function [6]. This method can be used to explain the nonlinear relationship between the response and independent variables. A dataset, including 349 samples and 7 independent variables, are used to train and test the MLR models.

2. Research method

2.1. Multiple linear regression model

The general form of a MLR model is given by [4]:

$$Y = \sum_{d=0}^{D} \beta_d X_d \qquad (1)$$

where Y denotes the estimated CS value; D is the number of independent variables; X_d (d = 1,2,…,D) is an independent variable and X_0 is always 1, which accounts for the bias term; β denotes the parameters of the MLR model.

Using the least squares method, the model's parameters can be estimated as follows:

$$\beta = (X^T X)^{-1} X^T Y$$

where X is the matrix of independent variables, including the bias terms.

2.2. Multivariate power function-based multiple linear regression model

It is worth noticing that in the aforementioned MLR model, the variable X_d can be a function of other variables. Herein, the linearity refers to the parameters of the model, not its variables [5]. Based on that notion, a multivariate power function (MPF) can be used to cope with the nonlinearity in a dataset. The MPF-based MLR model is given by [6]:

$$Y = \beta_0 \prod_{d=1}^{D} X_d^{\beta_d} \qquad (2)$$

Using the laws of logarithms, it is able to express the model as follows:

$$\log(Y) = \log(\beta_0) + \beta_d \sum_{d=1}^{D} \log(X_d) \qquad (3)$$

Let Y_{TF} be $\log(Y)$ and $X_{d,TF}$ be $\log(X_d)$, the aforementioned model can be neatly stated as follows:

$$Y_{TF} = \beta_d \sum_{d=0}^{D} X_{d,TF} \qquad (4)$$

The least squares method can be again used to compute the parameter of the MPF-based MLR model. In addition, because the logarithm and the exponential are inverse functions, to convert the transformed CS value into the original value, the following equation is used:

$$Y = \exp(Y_{TF}) \qquad (5)$$

2.3. Prediction interval for a compressive strength value

To account for the uncertainty in CS estimation, the method used for deriving its prediction interval can be used. A prediction interval for the CS of a novel mixture x_{new} is given by [2,4]:

$$PI = Y_{PE} \pm t_{n-p}(1-\frac{\alpha}{2})\hat{\sigma} \times \sqrt{1 + x_0^T (X^T X)^{-1} x_0} \qquad (6)$$

where $t_{n-p}(1-\frac{\alpha}{2})$ denotes the *t*-multiplier, which is inverse of the Student's *t* distribution function with $n-p$ degrees of freedom and α critical value. For instance, to compute a 95% prediction interval, a critical value of α = 0.05 can be used. Hence, for a 95% prediction interval, $1-\frac{\alpha}{2}=1-0.025=0.975$. Moreover, σ^2 represents the variance of random errors (ε).

In addition, the variance of random errors is unknown and must be estimated from the data. Herein, σ^2 is best estimated via the sum of squares error (SSE) [4] as follows:

$$s^2=\frac{SSE}{N-(D+1)} \qquad (7)$$

where s^2 is the estimated value of σ^2; *N* is the number of samples; *D* is the number of independent variables. The term *s* is also known as the estimated standard error of the model.

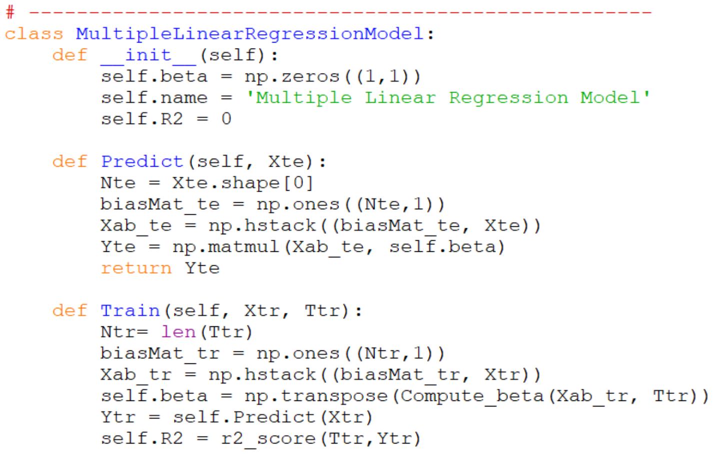
The term SSE is computed as follows:

$$SSE=\sum_{i=0}^{N-1}(T_i-Y_i) \qquad (8)$$

where T_i and Y_i are actual and estimated CS values, respectively.

3. Results and discussions

To train and test the performance of the MLR models, a dataset, including 349 samples and 7 independent variables, is used. The dataset was compiled in [3]. The original dataset consists of 527 samples. To avoid the computing of the logarithm of 0s as required by the MPF-based MLR model, all records that contain 0 elements are cast out. The independent variables include the content of cement RHA, water, fine aggregate, coarse aggregate, superplasticizer, and age of concrete. It is noted that 90% of the data samples are employed for training the models and the rest of the dataset is reserved for testing the established models. Moreover, the mean absolute percentage error (MAPE), the coefficient of determination (R^2), and the coefficient of variation (CV) are computed to inspect the models' performance [4]. In addition, the models are coded in Python to support its implementation (refer to **Fig. 3.1**). The functions used for computing s and the prediction interval are demonstrated in **Fig. 3.2**.

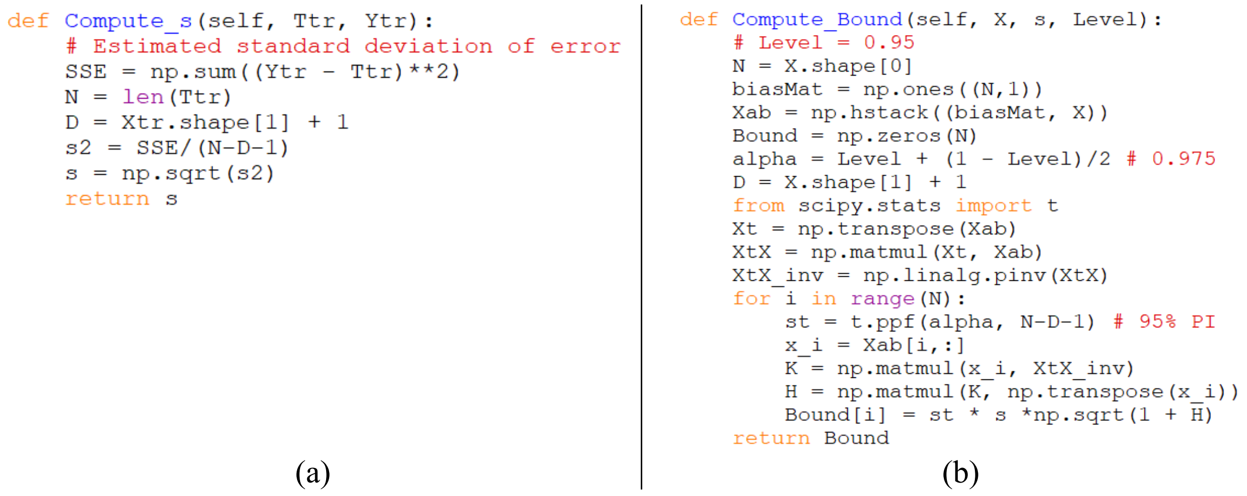


```
# -----------------------------------------------------
class MultipleLinearRegressionModel:
    def __init__(self):
        self.beta = np.zeros((1,1))
        self.name = 'Multiple Linear Regression Model'
        self.R2 = 0

    def Predict(self, Xte):
        Nte = Xte.shape[0]
        biasMat_te = np.ones((Nte,1))
        Xab_te = np.hstack((biasMat_te, Xte))
        Yte = np.matmul(Xab_te, self.beta)
        return Yte

    def Train(self, Xtr, Ttr):
        Ntr= len(Ttr)
        biasMat_tr = np.ones((Ntr,1))
        Xab_tr = np.hstack((biasMat_tr, Xtr))
        self.beta = np.transpose(Compute_beta(Xab_tr, Ttr))
        Ytr = self.Predict(Xtr)
        self.R2 = r2_score(Ttr,Ytr)
```

Fig. 3.1 The model's class coded in Python



```
def Compute_s(self, Ttr, Ytr):
    # Estimated standard deviation of error
    SSE = np.sum((Ytr - Ttr)**2)
    N = len(Ttr)
    D = Xtr.shape[1] + 1
    s2 = SSE/(N-D-1)
    s = np.sqrt(s2)
    return s
```

(a)

```
def Compute_Bound(self, X, s, Level):
    # Level = 0.95
    N = X.shape[0]
    biasMat = np.ones((N,1))
    Xab = np.hstack((biasMat, X))
    Bound = np.zeros(N)
    alpha = Level + (1 - Level)/2 # 0.975
    D = X.shape[1] + 1
    from scipy.stats import t
    Xt = np.transpose(Xab)
    XtX = np.matmul(Xt, Xab)
    XtX_inv = np.linalg.pinv(XtX)
    for i in range(N):
        st = t.ppf(alpha, N-D-1) # 95% PI
        x_i = Xab[i,:]
        K = np.matmul(x_i, XtX_inv)
        H = np.matmul(K, np.transpose(x_i))
        Bound[i] = st * s *np.sqrt(1 + H)
    return Bound
```

(b)

Fig. 3.2 The supporting functions: (a) computing s and (b) prediction interval calculation

Table 1. Result comparison

Metrics	MLR	MPF
MAPE (%)	30.71	14.30
R^2	0.60	0.84
COV (%)	37.54	23.78

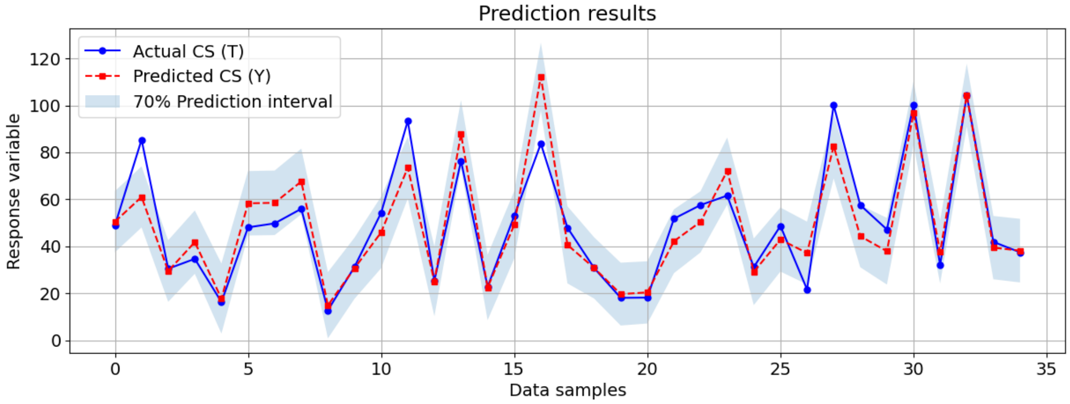


Fig. 3.3 Interval estimation of the CS of RHA-blended concrete

The prediction performances of the two linear models are summarized in **Table 1**. The standard MLR obtains a MAPE of 30.71% and a COV of 37.54%. It can explain roughly 60% of the variation in the CS of concrete mixes. The MPF-based model achieves a MAPE of 14.30% and a COV of 23.78%. The proportion of variation in the CS that can be explained by this model is roughly 84%. Hence, the performance of the MPF-based model is significantly better than the regular MLR model. This outcome can be attributed to the fact that the MPF-based model can better deal with nonlinearity in the collected dataset via the employed data transformation. The results coupled with prediction intervals of the MPF-based model is provided in **Fig. 3.3**. Herein, a confidence level of 70% is used. The result shows that the average width of the prediction interval is 27.55 MPa and the proportion of the testing data samples that lie within the interval is roughly 83%.

4. Conclusions

This paper utilizes linear regression models for estimating the CS of RHA-blended concrete. Experimental results point out good performance of the MPF-based model which attains a MAPE of 14.30%, a COV of 23.78%, and a R^2 of 0.84. Accordingly, about 84% of the sample variation in the CS can be explained by the independent variables and the established model. Besides, the MPF-based model has been coded in Python to support its implementations. One advantage of this model is that its prediction interval can be easily derived from the formulation of the conventional MLR model. Presenting the estimated results with prediction intervals is very useful for mix design process because they account for the uncertainty in CS of RHA-blended concrete. Future extensions of this work may include the use of other nonlinear methods (e.g. neural networks) as well as advanced approaches for constructing prediction intervals.

Supplementary material

The Python code and the dataset used to support the findings of this study have been deposited in the Github repository at: https://github.com/NhatDucHoang/LM_RHAC.

References

[1] Aslam F., Elkotb M.A., Iqtidar A., Khan M.A., Javed M.F., Usanova K.I., Khan M.I., Alamri S., Musarat M.A. (2022). Compressive strength prediction of rice husk ash using multiphysics genetic expression programming. *Ain Shams Engineering Journal* (13), 1-10. doi:https://doi.org/10.1016/j.asej.2021.09.020

[2] Fahrmeir L., Kneib T., Lang S., Marx B. (2013). *Regression: Models, Methods and Applications*. Berlin: Springer

[3] Hoang N.-D. (2022). Compressive Strength Estimation of Rice Husk Ash-Blended Concrete Using Deep Neural Network Regression with an Asymmetric Loss Function. *Iranian Journal of Science and Technology, Transactions of Civil Engineering* (47), 1547–1565. doi:10.1007/s40996-022-01015-4

[4] Mendenhall W., Sincich T. (2011). *Second Course in Statistics, A: Regression Analysis*. USA: Pearson

[5] Seber G.A.F., Lee A.J. (2003). *Linear Regression Analysis*. USA: John Wiley & Sons

[6] Zain M., Abd S. (2009). Multiple Regression Model for Compressive Strength Prediction of High Performance Concrete. *Journal of Applied Sciences* (9), 155-160. doi:https://doi.org/10.3923/jas.2009.155.160