TikTok Trending Tracks

Project of *Applied Statistics and Data Analytics*

Peisen Li Junhao Yu Yi Zheng

August 22, 2022

Abstract

The popularity of the songs on TikTok may be relative to the characteristics of the songs, and a regression model can be used to describe the relationship. In this work, linear model, support vector machine and gradient boosting trees are applied to fit the data and give predictions. The cross-validation result of each model is given.

1 Introduction

TikTok is a short-form video hosting service owned by Chinese company ByteDance. It hosts a variety of short-form user videos, from genres like pranks, stunts, tricks, jokes, dance, and entertainment with durations from 15 seconds to 10 minutes. Due to TikTok's unique recommendation algorithm, it has attracted a very large number of users around the world to publish and watch videos.

Songs are an integral part of TikTok, and different songs will bring people different audiovisual experiences and therefore have different popularity. What kind of songs are more popular? Can we predict whether a song will be popular according to its characteristics? In the TikTok dataset provided by Team Dan, the songs are characterized by several numerical metrics, including instrumenalness, liveness, tempo, etc. Besides, the corresponding popularity is provided in a numerical form as well. Therefore, it'll be easier for us to analyze the relationship between the characteristics and popularity of songs.

In our work, we try to formalize a regression problem, regarding "popularity" as the response variable and the other quantitative characteristics as the explanatory variables. We use linear regression as our main method, applying diagnostics and regularization to make it more precise. Also, support vector machine and tree-based methods are used as comparison. Based on these methods, we build a popularity-prediction model.

2 Data Overview and Preprocessing

The dataset contains 4 `.csv` files about the songs in TikTok: one is the global data, one is the local data of Philippines, one is the data about dance and the last one is the data about Original Pilipino Music (OPM).

There is no NaN element in our dataset, and therefore we don't need to deal with the missing values. See the next section for detailed EDA.

3 Model Specification

Since the 4 datasets share the similar analyzing process, we only include the detailed process for the global data without loss of generality. But we will list the results of all the 4 datasets at the end of each section.

3.1 Linear Regression

3.1.1 EDA

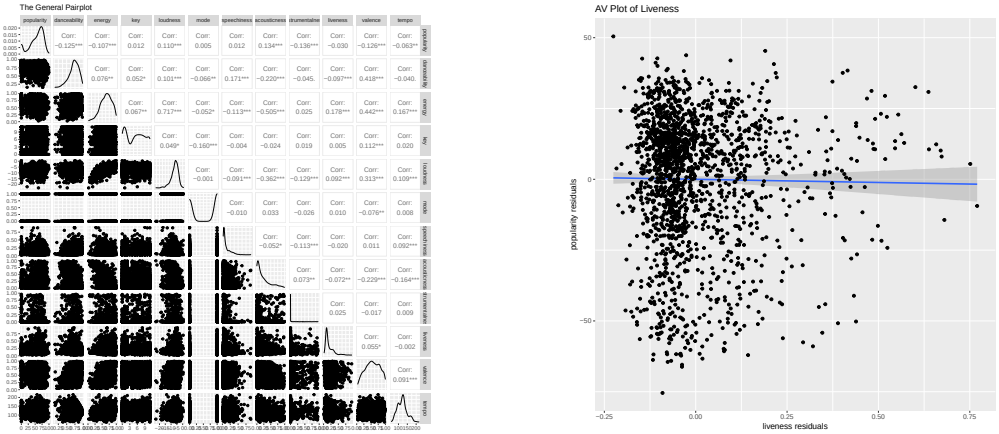


Figure 1: Results of EDA

From the correlation plot, it is obvious that some of the variables are remarkably right skewed, such as “speechiness”. What’s more, variables like “mode” are qualitative, which probably indicates the necessity of the introduction of interaction model.

From the partial regression plot, it seems that most variables do not have conspicuous linear relationship with response variable, which suggests some transformations might be crucial. We here only take “liveness” as an example.

3.1.2 Primitive Regression Model

We hereby used all the variables as explanatory variables, regardless of whether they’re qualitative or quantitative.

```
Analysis of Variance Table

Response: popularity
                   Df Sum Sq Mean Sq  F value    Pr(>F)
danceability        1  16318   16318  30.4299 3.991e-08 ***
energy              1  10050   10050  18.7424 1.583e-05 ***
key                 1    650     650   1.2118 0.2711221
loudness            1  79835   79835 148.8800 < 2.2e-16 ***
mode                1    389     389   0.7251 0.3945907
speechiness         1    877     877   1.6356 0.2011034
acousticness        1   4859    4859   9.0609 0.0026495 **
instrumentalness    1   7970    7970  14.8624 0.0001199 ***
liveness            1    144     144   0.2680 0.6047652
valence             1   1819    1819   3.3913 0.0657132 .
tempo               1   1879    1879   3.5031 0.0614228 .
Residuals        1713 918571     536
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
Residuals:
    Min      1Q  Median      3Q     Max
-75.635 -12.256   5.573  16.119  49.949

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)      118.64927    6.54035  18.141  < 2e-16 ***
danceability     -19.54655    4.48861  -4.355 1.41e-05 ***
energy           -36.36954    5.02412  -7.239 6.80e-13 ***
key                0.17635    0.15482   1.139 0.254844
loudness           2.90925    0.26466  10.993  < 2e-16 ***
mode              -1.01410    1.14251  -0.888 0.374875
speechiness        6.46233    4.77801   1.353 0.176389
acousticness       8.90288    2.71869   3.275 0.001079 **
instrumentalness -12.76498    3.30710  -3.860 0.000118 ***
liveness          -2.24063    4.02416  -0.557 0.577740
valence           -4.87019    2.82261  -1.725 0.084632 .
tempo             -0.04101    0.02191  -1.872 0.061423 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 23.16 on 1713 degrees of freedom
Multiple R-squared:  0.1196,    Adjusted R-squared:  0.1139
F-statistic: 21.16 on 11 and 1713 DF,  p-value: < 2.2e-16
```

Figure 2: Results of Primitive Model

From the regression and anova table, there is no doubt that the model is significant since p-value is far less than 0.05. However, the model can only explain 11.96% of the response variable

and MSE is 536, which is almost unbearable. Furthermore, half of the partial regression p-values are larger than 0.05, suggesting corresponding variables to be less significant. However, we should focus on the extra sum of squares to determine whether there is suppressor variable later rather than delete such variables.

From the diagnostic plots, it seems that the error variance is not constant and error distribution is not normal. Also, there might be non-linear relationship.

Furthermore, we used variance inflation factor to detect the multicollinearity. From the table above, however, such problem is not noteworthy since all the factors are less than 10.

3.1.3 Outlier, High Leverage or High Influence

In order to detect outliers, two methods were taken into consideration. The first one is intuitive – standardized residuals. We calculated the variance and mean of residuals of the full model, and then scaled the residuals. If the standardized residual is larger than 2 or less than -2, we have good reason to claim the corresponding case is an outlier.

The second method, Studentized Deleted Residuals, utilizes a concept Professor Wang once mentioned in the class – Leave One Out Validation.

To detect high leverage cases, hat values were straightforward. Usually we declare case i has extreme X values when $h_{ii} > \frac{2p}{n}$.

To detect influential cases, we used several different methods and combined their results to reach the final conclusion.

In order to measure whether removal of an outlier can cause dramatic change in regression results, we used DIFFITS.

To show the effect of the ith case on all n fitted values, we used Cook's Distance. Nevertheless, we didn't construct the test explicitly since such result had already been included in diagnostic plots.

To measure influence in the sense of regression coefficients, we used DFBETAS.

To combine the above results, we used influential.measures in R to detect all the high influential cases in any sense. We also constructed a plot to show the result of outlier, high leverage and high influence altogether.

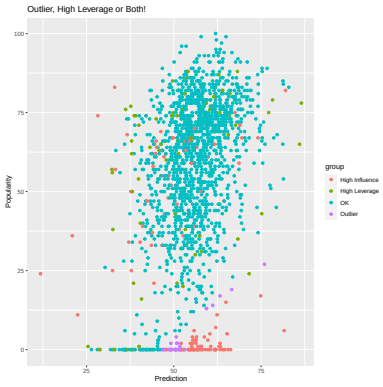
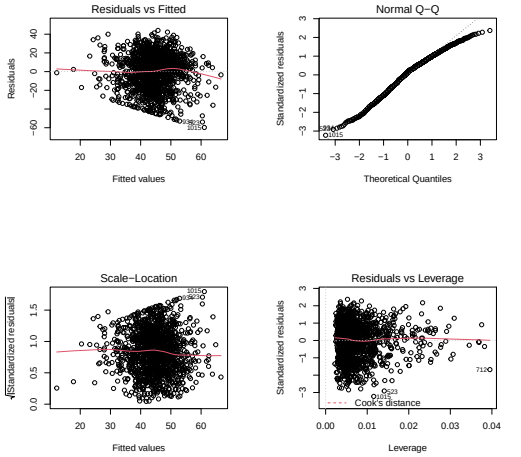


Figure 3: Outlier, High Leverage or High Influence

3.1.4 Box-Cox Transformation

Since there exists nonlinear relationship, we shall try transformations to reduce such problem. In this project, we tried Box-Cox transformation. The details are shown in the code. Here

we only present the result of Box-Cox as well as the test result of transformed model.




```
Residuals:
    Min      1Q  Median      3Q     Max
-59.738 -12.638   2.758  13.351  44.047

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)       82.10047    5.77899  14.207  < 2e-16 ***
danceability     -11.54083    3.53169  -3.268  0.00111 **
energy           -21.20872    4.22444  -5.020 5.80e-07 ***
key                0.14867    0.13499   1.101  0.27093
loudness           2.23592    0.23635   9.460  < 2e-16 ***
mode               0.07456    1.00236   0.074  0.94072
speechiness       -5.46677    4.12284  -1.326  0.18506
acousticness      10.01484    2.35693   4.249 2.28e-05 ***
instrumentalness  -9.07844    3.02028  -3.006  0.00269 **
liveness          -3.04924    3.42431  -0.890  0.37336
tempo             -0.01472    0.01892  -0.778  0.43676
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 18.63 on 1441 degrees of freedom
Multiple R-squared:  0.09622,   Adjusted R-squared:  0.08994
F-statistic: 15.34 on 10 and 1441 DF,  p-value: < 2.2e-16
```

Figure 4: Results of Box-Cox Model

From the diagnostic plots, we can see that all assumptions are satisfied. Also, the train MSE is reduced to 347, which is a large leap. But the explanatory proportion remains low, which is 9.622%. The test MSE is 405.2806, which is tolerable compared to train MSE.

However, this model is not interpretable. The best λ is around 1.5, causing the transformation to be $Y^\star = \frac{Y^{1.5}-1}{1.5}$. Since we have little domain knowledge, we can hardly explain why this parameter is appropriate. Hence, we shall try to better the performance without using transformation.

3.1.5 Variable Selection

From the primitive model, it is obvious that some variable is not significant. Hence, to improve MSE, a straightforward way is to select significant variable and delete the rest of them. Again, since we have little domain knowledge, we tried Step-wise Selection by different criterion as well as Mallows' Cp.

The criterion we considered include Adjusted R^2 (to better the fitting ability), AIC, BIC and PRESS (to better the predicting ability).

Luckily enough, all three criterion gave the same result. While carrying out the Step-wise, we carefully checked the extra sum of squares. It turned out that there wasn't enough evidence supporting suppressor variables.

```
> SignifReg(nullmodel,alpha = 0.05, criterion = "PRESS",scope = scope,direction = "both")

Call:
lm(formula = popularity ~ instrumentalness + acousticness + loudness +
    energy + danceability, data = tracks4_df)

Coefficients:
     (Intercept)  instrumentalness      acousticness          loudness            energy      danceability
         115.388           -13.028             8.775             2.911           -40.510           -20.759
```

Figure 5: Step-wise Result

The result of Mallows' Cp is presented below.

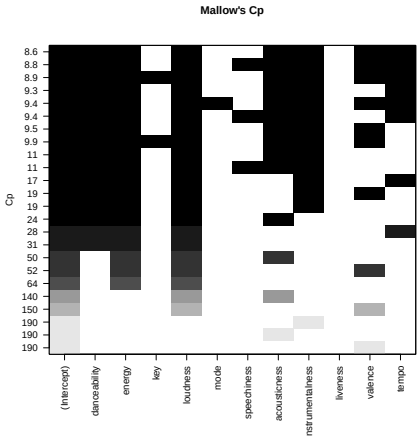


Figure 6: Result of Mallows' Cp

3.1.6 Introduction of Interaction Model

From EDA, it seems that "key" and "mode" are both qualitative variables. However, "key" is ordinal variable and it has 10 levels, hence we can only consider the interaction of "mode" for simplicity. Again, we used Step-wise to detect which interaction is significant and the result is shown below.

```
Call:
lm(formula = popularity ~ instrumentalness + acousticness + loudness +
    energy + danceability + mode:valence + mode:speechiness,
    data = tracks4_df)

Coefficients:
     (Intercept)  instrumentalness      acousticness          loudness            energy      danceability      mode:valence  mode:speechiness
         114.146           -12.913             9.530             2.919           -38.187           -20.271            -5.418            12.473
```

Figure 7: Results of Interaction Model

Figure 8: Results of Interaction Model

From the result and the plot, there are two main problems. First, the Step-wise didn't include the main effect of "valence" and "speechiness", which cannot be properly interpreted. Second, there is no distinct difference between the prediction results of "mode" 0 and "mode" 1, suggesting the interaction model might not be appropriate.

3.1.7 Regularization

Ridge and lasso have been tried but there is no obvious progress.

3.1.8 Experiment and Results

We use 5-fold cross-validation and the MSEs are listed as below.

Table 1: Results of linear model

Dataset	cv-MSE of primitive model	cv-MSE of improved model
DANCE	687.7562	626.4124
OPM	452.7946	399.5197
Philippines	592.8523	513.4178
global	574.6189	482.4694

3.2 Support Vector Machine

3.2.1 Intuition

Using the linear model to fit the regression problem is based on the assumption that there does exist a linear relationship between the response variable and the explanatory variables. Unfortunately, however, this assumption is too ideal and can hardly be satisfied in real world. Actually, nonlinear models exist more widely in nature. Therefore, we may try using a nonlinear model to solve this problem.

3.2.2 Reasons to Use SVM

One of the most common methods to introduce nonlinearity to a model is to add high-order terms or cross terms. For example, suppose we have a model that is $y = a_0x_1 + b_0x_2$, then we can introduce nonlinearity by adding terms such as x_1^2, x_2^2 or x_1x_2, and then refit the model to the form $y = ax_1 + bx_2 + cx_1^2 + dx_2^2 + ex_1x_2$.

However, using this method requires some prior knowledge to construct nonlinear terms. Otherwise, it'll be time-consuming to find the proper terms to add, and even if you find any, they may not make sense or be hard to be interpreted in terms of science.

SVM seems to be a better choice when introducing nonlinearity because of the kernel trick, which is widely known in machine learning. It uses a nonlinear transformation to map the data features from the original space to a new space, where the original space is a low-dimensional input space (Euclidean space or discrete set), and the new space is a high-dimensional feature space (Hilbert space). Then, we can solve the nonlinear problem using a linear model on the new space without specifying the nonlinear terms.

In general, under the given conditions of the kernel function $\kappa(x, z)$, the nonlinear regression problem can be solved by the method of solving the linear regression problem. The learning is implicitly carried out in the feature space and does not need to be explicitly defined.

3.2.3 Experiment and Results

We use support vector regression from package `caret` and we choose the RBF kernel. 5-fold cross-validation and grid search are applied to determine the optimal parameters, and MSE is used as the evaluation metric. The results are listed as below.

Table 2: Results of SVM

Dataset	cv-MSE
DANCE	657.6885
OPM	403.8998
Philippines	569.3052
global	551.8435

3.3 Gradient Boosting Trees

3.3.1 Intuition

Tree-based methods partition the feature space into a set of rectangles, and then fit a simple model (like a constant) in each one. They are conceptually simple yet powerful, which provide us another possible way to deal with our regression problem. Compared with linear models, tree-based methods have the following advantages:

- For decision trees, data preparation is often simple or unnecessary. Other techniques often require the data to be generalized first, such as removing redundant or blank attributes.
- Decision trees can handle both data type and general type attributes. Other techniques often require a single data attribute.
- Decision tree is a white-box model. Given an observed model, it is easy to derive the corresponding logical expression from the resulting decision tree.

Boosting is one of the ensemble methods. The main idea of this ensemble method is to use certain means to learn multiple base learners, and these multiple base learners are required to be weak learners, and then multiple base learners are combined. The boosting method builds the model in a stage-wise way, and the weak learner built in each step is to make up for the shortcomings of the existing model.

Gradient refers to the method used to minimize the loss function. The traditional boosting model, such as `Adaboost`, minimizes the loss function by updating the sample weight distribution after each iteration (the sample weight of the pair becomes smaller, the weight of the wrongly classified samples becomes larger), so that the latter basic learner pays more attention to the wrongly classified samples and finally the model should achieve the goal of minimizing the loss function. Gradient boosting uses the negative gradient of the loss function to fit the weak learner at each step of iteration, in order to achieve the purpose of minimizing the loss function.

3.3.2 Experiment and Results

We use the `gbm` package from R, setting maximum number of trees equal to 100 and shrinkage step size equal to 0.05, and we can get the following results after applying 5-fold cross-validation. The green line represents the cross-validation error and the black line represents the

training error, while the blue dotted line represents the best iteration of the training process.[1] The results are listed as below.

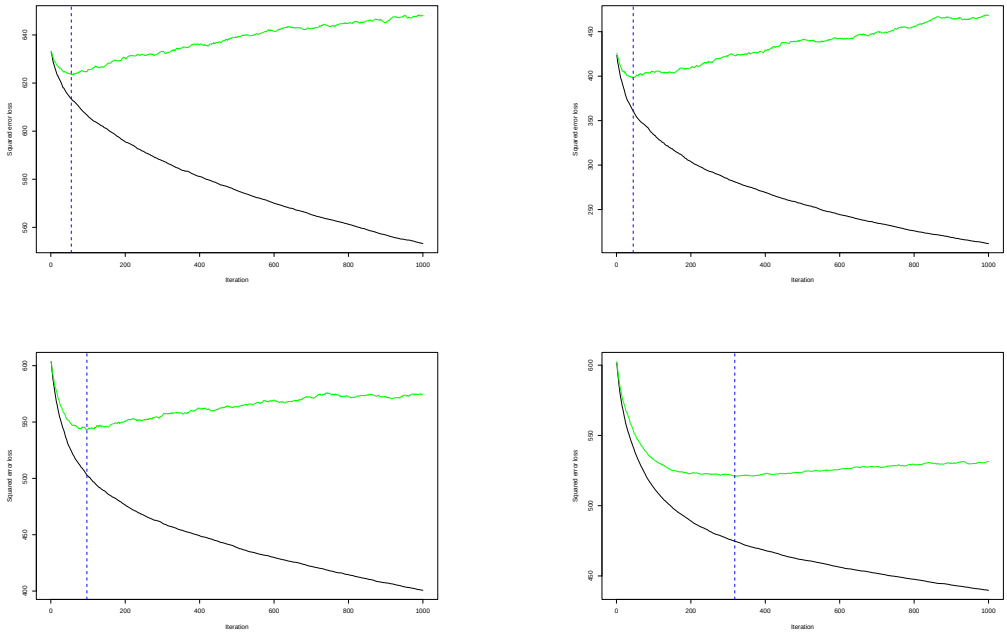


Figure 9: Results of GBM

Table 3: Results of GBM

Dataset	cv-MSE
DANCE	624.1726
OPM	398.7096
Philippines	542.4182
global	520.5417

4 Conclusion and Discussion

According to the analysis above, we've made several models to fit the data and give some predictions. However, as the data comes from real world and may include noises, our models are still far from perfect. Generally speaking, boosted tree should be the strongest model, but our improved linear model beat the boosted tree model on some datasets.

"All models are wrong, but some are useful." This attempt of data analysis taught us that even the simplest linear model can be powerful if applied on proper dataset and used in proper way. The principles of data analysis matter more than the result itself.

[1] The figures and their corresponding dataset: top left-DANCE; top right-OPM; lower left-Philippines; lower right-global

5 Acknowledgement

We sincerely thank Team Dan for providing the dataset.

6 Roles and Responsibilities of the Team

Yi Zheng (team leader): making plans, building of linear model, parameter tuning of support vector machine

Junhao Yu: building of support vector machine, remedies of linear model

Peisen Li: building and parameter tuning of gradient boosting trees, designing poster

Appendix

A Mallows' Cp

Consider a model of $p-1$ variables:

$$\hat{Y}^p = X_p(X_p^T X_p)^{-1} X_p^T Y = H_p Y$$

$$E(\hat{Y}_p) = H_p E(Y) = H_p \mu, Var(\hat{Y}_p) = \sigma^2 H_p$$

where $\mu = (\mu_1, ..., \mu_n)^T$ be the true mean responses at the X_i's.

Recall the Bias-Variance trade-off:

$$E(\hat{Y}_p - \mu_i)^2 = (E(\hat{Y}_p - \mu_i))^2 + Var(\hat{Y}_p) = Bias^2 + Variance$$

Total mean squared error:

$$\sum_{i=1}^{n} E(\hat{Y}_p - \mu_i)^2 = \sum_{i=1}^{n} (E(\hat{Y}_p - \mu_i))^2 + \sum_{i=1}^{n} Var(\hat{Y}_p)$$

The variance part:

$$\sum_{i=1}^{n} Var(\hat{Y}_p) = tr(Var(\hat{Y}_p)) = tr(\sigma^2 H_p) = p\sigma^2$$

The bias part:

$$\begin{aligned} &\left(E\left(\hat{Y}^p\right) - \mu\right)^T \left(E\left(\hat{Y}^p\right) - \mu\right) \\ &= \left(H_p\mu - \mu\right)^T \left(H_p\mu - \mu\right) = \mu^T \left(I - H_p\right) \mu \\ &= E\left(Y^T \left(I - H_p\right) Y\right) - \sigma^2 \,\mathrm{tr}\left(I - H_p\right) \\ &= E(\mathrm{SSE}(p)) - (n-p)\sigma^2 \end{aligned}$$

Total mean squared error:

$$\sum_{i=1}^{n} E\left(\hat{Y}_i^p - \mu_i\right)^2 = E(SSE(p)) - (n-2p)\sigma^2$$

The model's performance measure, sum squared prediction error (SSPE) :

$$\Gamma_p = \frac{\sum_{i=1}^{n} E\left(\hat{Y}_i^p - \mu_i\right)^2}{\sigma^2} = \frac{E(SSE(p))}{\sigma^2} - (n-2p)$$

Mallows' Cp:

$$C_p = \hat{\Gamma}_p = \frac{E(SSE(p))}{\hat{\sigma}^2} - (n-2p)$$

where $\hat{\sigma}^2 = MSE(P)$

When a model is unbiased, $C_p \approx p$; Among all unbiased models, prefer (choose) model with small C_p. $C_p \gg p$ means significant bias, indicating missing relevant predictors. $C_p \ll p$ may indicate overfitting. Use the plot of C_p versus p.

B Studentized Deleted Residuals

We hereby define Studentized Deleted Residuals as $d_i = Y_i - \hat{Y}_{i(-i)}$. According to mathematical derivation, we have the following result:

$$d_i = \frac{e_i}{1 - h_{ii}}$$

$$\text{Var}(d_i) = \frac{\sigma^2}{1 - h_{ii}}$$

Furthermore, we use statistic to construct a t-test:

$$t_i = \frac{d_i}{s(d_i)} = \frac{e_i}{\sqrt{\text{MSE}_{(-i)}(1 - h_{ii})}} \sim t(n - p - 1)$$

To lessen calculative pressure, however, we use the same decision rule as standardized residuals to detect outlier.

C DIFFITS and DFBETAS

C.1 DIFFITS

$$(\text{DIFFITS})_i = \frac{\hat{Y}_i - \hat{Y}_{i(-i)}}{\sqrt{\text{MSE}_{(-i)}\, h_{ii}}}$$

We consider influential if DIFFITS > $\frac{2\sqrt{p}}{\sqrt{n}}$.

C.2 DFBETAS

$$(\text{DFBETAS})_{k(-i)} = \frac{b_k - b_{k(-i)}}{\sqrt{\text{MSE}_{(-i)}((X'X)^{-1})_{kk}}}$$

Usually, we consider a case influential if | DFBETAS | > $\frac{2}{\sqrt{n}}$

D AIC, BIC and PRESS

The definitions of AIC, BIC and PRESS are listed as below.

$$\text{AIC}(p) = n \log(\frac{\text{SSE}(p)}{n}) + 2p$$

$$\text{BIC}(p) = n \log(\frac{\text{SSE}(p)}{n}) + log(n)p$$

$$\text{PRESS}(p) = \Sigma_{i=1}^{n}(Y_i - \hat{Y}_{i(-i)})^2$$