Linear Regression for Predicting the Quality of Red Wine

Pérez Mariana, ISC A01206747, Tecnológico de Monterrey Campus Querétaro

***Abstract*— This document presents the implementation and explanation of an analysis of the quality of the wine based on their characteristics the dataset consists of different variants of the Portuguese "Vihno verde" wine. The document also explains the model used to predict the goal result.**

I. INTRODUCTION

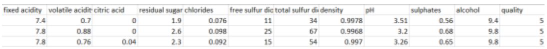
Wine producers constantly brag about the quality ratings that their wines receive from critics, because a high wine quality commonly translates into an increase in sales for that specific wine, but quality in wines come in all colors, degrees of sweetness and dryness, and flavor profiles. The following report is divided into 6 sections that will give insight and information about the state of the art, the used data set, the proposal of the model, the test and validation of the model and the conclusions of the report and the implemented model.

II. STATE OF THE ART

The study of simple predictive models has always been a topic of interest in machine learning, where a high predictive accuracy is needed. Linear regression is a statistical model which attempts to model the relationship between a dependent variable and one or more independent variables by fitting a linear equation. [1].

III. DATA SET

The dataset comes from a Kaggle [2] problem which contains records of the quality of the red wine based on different characteristics. The dataset is composed of 12 columns which are the fixed acidity, the volatile acidity, the citric acid, the residual sugar, the amount of chlorides, the total sulphur dioxide, the amount of free sulphur dioxide, the density of the water, the ph. of the wine, the amount of sulphates, the percent of alcohol and the quality of the wine. Figure 1 shows an example of the given dataset.



fixed acidity	volatile acidit	citric acid	residual sugar	chlorides	free sulfur dic	total sulfur di	density	pH	sulphates	alcohol	quality
7.4	0.7	0	1.9	0.076	11	34	0.9978	3.51	0.56	9.4	5
7.8	0.88	0	2.6	0.098	25	67	0.9968	3.2	0.68	9.8	5
7.8	0.76	0.04	2.3	0.092	15	54	0.997	3.26	0.65	9.8	5

Fig 1. Shows the given data without any scaling

The dataset is only one file which consists of 1599 records, so the file was divided intro two sections for the purpose of having a dataset to train and a dataset to test the parameters. The training data consists of 70% of the file meanwhile the test data is the remaining 30%.

IV. MODEL PROPOSAL

All the values of the dataset are numeric which means there was no transformation of alphabetic characters into numbers, but some of the fields, such as the free sulphur dioxide field, had a high variation or dispersion between the values given, therefore a scale of all the values had to be done in order to help the model achieve the goal results.

A standard scaling was implemented for all the fields in which it sums all the values in the field and get the average and the standard deviation for it, then every value is scaled using the following formula, where μ is the mean and σ is the standard deviation. This type of scaling is used to standardize the range of the independent variables.

$$X_i = \frac{X_i - \mu}{\sigma}$$

The resulting data is the given to the linear regression part of the model. To get the result the models uses the following gradient descent function to minimize the cost until the new parameters are the same as the older parameters or when the number of epochs or iterations is equal to 1000.

$$\theta j = \theta j - \alpha \sum_{i=1}^{m} \left(h\left(x^{i}\right) - y^{i}\right) xj^{i}$$

Alpha is the learning rate and is set in the model to 0.01 and the hypothesis used in the gradient descent is the linear function which is

$$y = \theta_0 b + X_1\theta_1 + X_2\theta_2 + \cdots$$

The linear function is used to map the given input and the calculated parameters by the gradient descent to give a prediction of a continuous value.

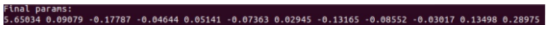
The cost function used in linear regression is the square mean error, which is shown below,

$$\sum_{i=1}^{m} (h(y_i) - y_i)^2$$

Therefore, every iteration a copy of the current parameters is made and saved, the parameters are then changed according to the result of minimizing the cost using gradient descent and the hypothesis function, and finally the mean square error that the model calculates is used to validate the performance of the model and decide if the given model is overfitting or underfitting the dataset.

V. TEST AND VALIDATION

The results of running the previously presented model [3] using the train dataset outputs the following parameters.



```
Final params:
5.65034 0.09079 -0.17787 -0.04644 0.05141 -0.07363 0.02945 -0.13165 -0.08552 -0.03017 0.13498 0.28975
```

The model achieved a square mean error of 0.00037 in the training part and a 0.00091. This means that the in general the presented model performs well and is getting most of the general pattern but the difference between the error tells that there is a small percentage of overfitting and that the model could behave better. Fig 3 shows how the model reaches the local minimum using the gradient descent and linear equation.

VI. CONCLUSIONS

The presented model performs overall well but it is possible to improve it to minimize the small overfitting. In this particular case the most first step to improve the model is by reducing the model complexity because all the 12 columns were used, making the model have big complexity. In order to reduce the complexity, there must be first an analysis of the correlation of the variables and the goal variable, the quality of the wine, to verify which columns have a high correlation and which don't.

VII. REFERENCES

[1]Malte J. (2016). Simple Regression Models. Retrieved from http://proceedings.mlr.press/v58/lichtenberg17a/lichtenberg17a.pdf

[2]Linear Regression (n.d.). Retrieved from https://www.kaggle.com/uciml/red-wine-quality-cortez-et-al-2009

[3]Perez M. (2019). Square Mean Error. Retrieved from https://github.com/pirty6/Machine-Learning/tree/master/SquareMeanError