Predicting Ratings For a Variety of Dark Chocolates

Manvir Kohli, Julie Song, Kelvin Wong

2022-11-26

Contents


```
## Warning: package 'kableExtra' was built under R version 4.2.2
```

```
## Warning: package 'pandoc' was built under R version 4.2.2
```

Summary

Four regression models were built using decision tree, support vector machine, ridge (linear regression) and k-nearest neighbors algorithms, and compared to determine the best model for predicting a chocolate's rating based on characteristics such as the number and type of ingredients in the chocolate, its amount of cocoa, location for the manufacturing company, memorable characteristics, and several others. The ratings are predicted based on a scale of 1 to 5. Our models performed _____ on a test set containing _____ observations...

Introduction

How well-recieved a new chocolate may be, can depend on a variety of factors. The project present here is interested in predicting the rating for a type of dark chocolate on a scale of 1 to 5, given some of its characteristics. This information can be useful for product analysis, and may be able to aide in predicting the popularity of a new product. It may also be useful for determining some important factors in the development of the chocolate, and suggest characteristics to focus on in order to develop a successful product.

Methods

Data

The data set is provided by the Manhattan Chocolate Society (@cocoa), and was found and retrieved from the tidytuesday data project (@tidytuesday), specifically through this link: https://github.com/rfordatascience/tidytuesday/blob/master/data/2022/2022-01-18/readme.md

The data set contains observations for different types of dark chocolate, including the manufacturing company and its location, origin of the cocoa beans used to make the chocolate, the other ingredients in the chocolate, the amount of cocoa in the chocolate, and others. They have also provided a feature column that contains descriptive words relating to the characteristics of the chocolate flavor, and a final rating.

Before the models were built, the data also had to be processed. The information in the `ingredients` feature column was split apart into several columns, including one column for the number of ingredients in the chocolate, and one column for each of the recorded ingredients. The company location and cocoa bean origin features were modified such that only some categories that had total observations meeting a minimum threshold were considered, while the remaining categories were considered "Other". The amount of cocoa in the chocolate feature was also converted from a character data type to a numerical datatype.

Analysis

Four regression models were built after splitting the original dataset into training and testing dataset in a 70%-30% split. Apart from the processing performed above, some features were also dropped due to being practically unique to each observations, such as the reference number and the company manufacturer. Table 1 shows the hyperparameters that were considered for each regression model. Hyperparameter optimization was performed for each model, using 20 iterations of 5-fold cross-validation each.

Table 1: Model Hyperparameters

Model	Hyperparameter
KNN	Leaf Size
	Number of Neighbors
	Weights
Ridge	Alpha
SVM RBF	C
	Gamma
Decision Tree	Max Depth

The initial exploratory data analysis for this dataset was performed in R (R Core Team (2022)) using the following packages: dplyr (Wickham et al. (2022)), tidyverse (Wickham et al. (2019)), docopt (de Jonge (2020)), cowplot (Wilke (2020)), knitr (Xie (2014)), and kableExtra (Zhu (2021)).

The modelling was performed in Python (Van Rossum and Drake (2009)) using the following packages, available in the provided environment file: matplotlib (Hunter (2007)), scikit-learn (Pedregosa et al. (2011)), pandas (McKinney (2010)), imbalanced-learn (Lemaître, Nogueira, and Aridas (2017)), lightgbm (Ke et al. (2017)), joblib (Joblib Development Team (2020)), docopt (Keleshev (2014)), and dill (Van Rossum (2020)).

Results and Discussions

The distributions for the numerical and categorical features were first analyzed to check for any unreasonable skewness. These distributions are present in Figures 1 and 2. The top company locations which had at least 20 observations and the top bean origin countries with at least 10 observations were kept, while the remaining categories in each feature were grouped as Other. It is revealed that the manufacturing company feature has an overwhelming number of categories, and even taking into considering only the top 50 companies, the Other category has far more points compared to the others. This feature was considered too unique to the observations, and was omitted from the modelling.

Table 2 shows the optimal hyperparameters, their associated values, and the resulting validation scores for

each model. The hyperparameter optimization and cross-validation was performed for each of the models in this regression problem.

Table 2: Optimized Hyperparameters With Validation Scores

Model	Hyperparameter	Optimized Value
KNN	Leaf Size	1
	Number of Neighbors	2
	Weights	3
Ridge	Alpha	4
SVM RBF	C	5
	Gamma	6
Decision Tree	Max Depth	7

The models performed _____ on the test datasets (how they compare with the validation results, how they compare with each other, were they accurate) Table 3 shows a comparison of scores for the models. The scoring metric used was the negative mean absolute percentage error metric, hence the scores are negative. Looking only at their magnitudes, _____model had the smallest error (so was the best model, ____% difference compared to the actual rating)

There are several ares for improvement with these models. For instance, perhaps we could use a sentiment analysis package on the memorable characteristics text feature, and manually aggregate some of the similar terminology (such as "acidic" and "acid"). Since most of our features are categorical features, we could also do some feature crosses and determine if there are any relations between the features. This might require some more careful examination of the data, but for instance, we could begin with some crosses of the ingredients present in the chocolate. As well, we could introduce a noise attribute for regularization, as it may help us identify exactly how well the model is actually performing with these features. Finally, we may also do some feature selection based on coefficient weightings, for instance for the Ridge model. Perhaps these can allow us to make the models more generalizable, and able to predict well for brand-new types of chocolate, which is the ultimate goal for the project.

References

de Jonge, Edwin. 2020. *Docopt: Command-Line Interface Specification Language*. https://CRAN.R-project.org/package=docopt.

Hunter, J. D. 2007. "Matplotlib: A 2D Graphics Environment." *Computing in Science & Engineering* 9 (3): 90–95. https://doi.org/10.1109/MCSE.2007.55.

Joblib Development Team. 2020. *Joblib: Running Python Functions as Pipeline Jobs*. https://joblib.readthedocs.io/.

Ke, Guolin, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. 2017. "Lightgbm: A Highly Efficient Gradient Boosting Decision Tree." *Advances in Neural Information Processing Systems* 30: 3146–54.

Keleshev, Vladimir. 2014. *Docopt: Command-Line Interface Description Language*. https://github.com/docopt/docopt.

Lemaître, Guillaume, Fernando Nogueira, and Christos K. Aridas. 2017. "Imbalanced-Learn: A Python Toolbox to Tackle the Curse of Imbalanced Datasets in Machine Learning." *Journal of Machine Learning Research* 18 (17): 1–5. http://jmlr.org/papers/v18/16-365.html.

McKinney, Wes. 2010. "Data Structures for Statistical Computing in Python." In *Proceedings of the 9th Python in Science Conference*, edited by Stéfan van der Walt and Jarrod Millman, 56–61. https://doi.org/10.25080/Majora-92bf1922-00a .

Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, et al. 2011. "Scikit-Learn: Machine Learning in Python." *Journal of Machine Learning Research* 12: 2825–30.

R Core Team. 2022. *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. https://www.R-project.org/.

Van Rossum, Guido. 2020. *The Python Library Reference, Release 3.8.2*. Python Software Foundation.

Van Rossum, Guido, and Fred L. Drake. 2009. *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace.

Wickham, Hadley, Mara Averick, Jennifer Bryan, Winston Chang, Lucy D'Agostino McGowan, Romain François, Garrett Grolemund, et al. 2019. "Welcome to the tidyverse." *Journal of Open Source Software* 4 (43): 1686. https://doi.org/10.21105/joss.01686.

Wickham, Hadley, Romain François, Lionel Henry, and Kirill Müller. 2022. *Dplyr: A Grammar of Data Manipulation*. https://CRAN.R-project.org/package=dplyr.

Wilke, Claus O. 2020. *Cowplot: Streamlined Plot Theme and Plot Annotations for 'Ggplot2'*. https://CRAN.R-project.org/package=cowplot.

Xie, Yihui. 2014. "Knitr: A Comprehensive Tool for Reproducible Research in R." In *Implementing Reproducible Computational Research*, edited by Victoria Stodden, Friedrich Leisch, and Roger D. Peng. Chapman; Hall/CRC. http://www.crcpress.com/product/isbn/9781466561595.

Zhu, Hao. 2021. *kableExtra: Construct Complex Table with 'Kable' and Pipe Syntax*. https://CRAN.R-project.org/package=kableExtra.