Canadian Art Project Funding Ranges Prediction Using Random Forest

Amelia Tang

1/18/2022

Contents

Audiance Persona

Astrid is enrolled in a Data Science boot camp trying to switch career from freelance graphic design to data science. She has taken basic math and statistics in college but might not be able to do math proofs for machine learning algorithms. She has learned most of the commonly used terminologies in data science such as training/testing data, different metrics etc. She also learned the general ideas of popular machine learning algorithms from the boot camp. She is curious about how we can use apply machine learning to the real world. As a former art practitioner, she is interested in how big data is used in the art industry.

Abstract / Executive Summary

The Government of Canada plays a vital role in preserving Canadian heritage through funding art projects. The eligible Canadian artists and art organizations may want to know the potential ranges of the funding they can receive for budgeting purposes. In this project, we built a predictive model utilizing popular machine learning algorithms for multi-class classifications, including Logistics Regression, Multinomial Naïve Bayes, Support Vector Classification (SVC) and Random Forest. The model predicted art projects' funding ranges using features, such as the projects' locations, disciplines and target audiences, that were not indicative of visual properties. We compared the performance of each algorithm using three metrics, weighted average f-1, weighted-average recall and weight-average precision. The results showed that the random forest algorithm outperformed others with a weighted-average f-1 score of 0.69, a weight-average recall of 0.69, and a weight-

average precision of 0.69. Therefore, we could use the features non-indicative of visual characteristics to predict the funding ranges that the government would grant for art projects in Canada.

Introduction

The Canadian Arts Presentation Fund ("the Fund") provides financial assistance to arts festivals presenters and performing artists (Canada, n.d.). The Fund's performance was evaluated based on the diversity of the art projects it covered and broadness of the communities it reached. The key mission of the Fund was to support a variety of art projects in all parts of the country based on economic values that the projects could bring to local communities ("Grouped Arts Evaluation: Canada Arts Presentation Fund, Canada Cultural Spaces Fund, and Canada Cultural Investment Fund 2013-14 to 2017-18_2019" 2019).

In recent years, many studies used neural network algorithms and features indicative of visual properties to predict values of art work. However, the models with visual features, such as the image of the art work, did not outperform the models using numeric and textual data only (Aubry et al. 2022). In most cases, image data presenting the art work alone failed to appraise the values (Ayub, Orban, and Mukund 2017).

In this project, we built classification models to predict approved funding ranges of Canadian art projects based on textual features that reflected social, cultural and geographical diversities. We did not add any features related to the visual characteristics of the art projects. After comparing the performances of four algorithms for multi-class classification, logistics regression, Naïve Bayes, Support Vector Classification (SVC) and Random Forest, we constructed a final predictive model using Random Forest.

Methods

Data Collection

From fiscal year 2016-2017 and fiscal year 2017-2018, the Government of Canada collected data on the art projects accepted and the funding sizes approved. The data set is available on the Canada's Open Data website and can be found here. The data set has 1358 rows of data. Each raw of the data contains the project's name, location information, including community, city, region and province, presenter information, such as organization names, disciplines and target audience. The Fund also reported the funding size for each art project in the data set. All the features included in this data set were categorical features.

Data Processing

We observed that the funding sizes reported were not continuous, so we divided the funding size data into five ranges, less than 8 thousand Canadian dollars (`less than $8.0k`), from 8 to 12 thousand Canadian dollars (`$8.0k-12.0k`), from 12 to 23 thousand Canadian dollars (`$12.0k-23.0k`), from 23 to 50 Canadian dollars (`$23.0k-$50.0k`), and greater than 50 thousand Canadian dollars (`over $50.0k`). We used 10%, 25%, 50% and 75% percentiles of the funding size data as the cutoffs for the ranges.

When selecting features for our classification model, we noticed the similar patterns of the funding ranges for each discipline in both fiscal years. Therefore, fiscal year would not be a meaningful predictor for funding ranges and we discarded this feature. Figure 1 showed the detailed distributions for both fiscal years.

We also discarded the data on regions because there was already data on provinces of the art projects. It would be repetitive had we included both. In addition, we excluded the organization names of the art projects because there were too many unique names and thus would not be informative when predicting funding ranges. Similarly, we did not choose to use data on other disciplines because we already included data on disciplines and there were many unique values in other disciplines. The key features we included for our projects were province, community type, grant or contribution, presenter type, project type, project sub-type, disciplines and audiences.

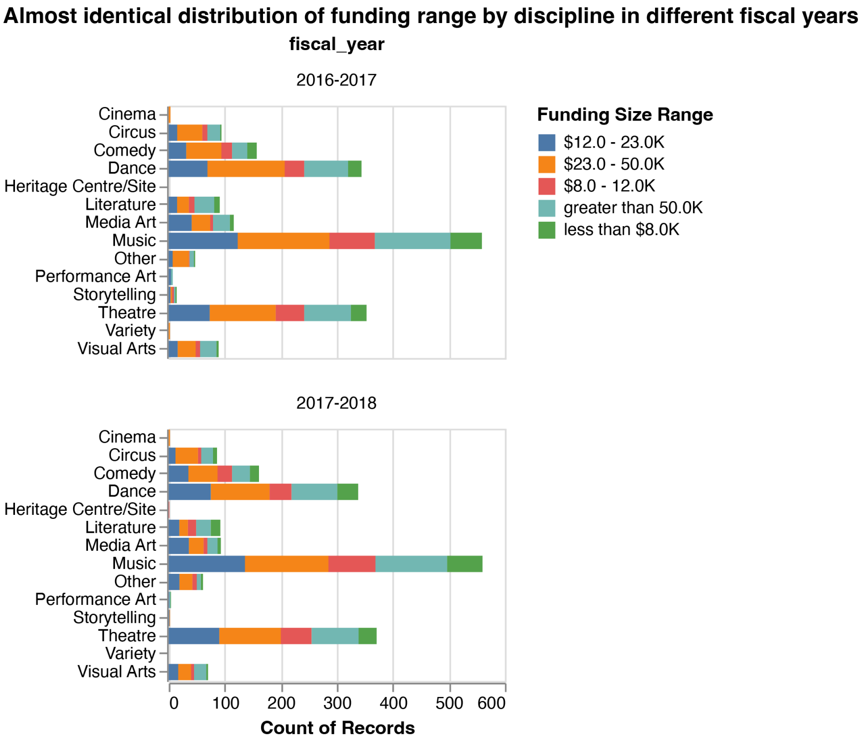

Figure 1: Comparison of the distribution of funding ranges 2016-2017 vs 2017-2018

Data Analysis

We used the Python programming language (Van Rossum and Drake 2009) and the following Python packages to perform the analysis: numpy (Harris et al. 2020), pandas (McKinney et al. 2010) and sikit-learn(Pedregosa et al. 2011). The code used to perform the analysis and create this report can be found here.

We conducted a conventional 80:20 split to create the training and testing data. We created a base-case model using the dummy classifier in the scikit-learn package (Pedregosa et al. 2011) to serve as a reference. Further, we chose four machine learning algorithms commonly used for multi-class classification problems, Logistics Regression, Multinomial Naïve Bayes, Support Vector Classification (SVC) and Random Forest. We again used scikit-learn package (Pedregosa et al. 2011) to build classification models.

There were class imbalances in our training data set. Table 1 showed that we had different sample sizes for the five ranges. To deal with this issue, we set the class weight parameter to "balanced" for Logistic Regression, Support Vector Classification (SVC) and Random Forest. When set to balanced, the class weight parameter would tune the model to assign a class weight inversely proportional to sample size in each class. We also set the max iteration parameter of the Logistic Regression to 1000 for the solver to converge.

Table 1: Count of each funding ranges (Observed class imbalances)

Funding ranges	Count
over $50.0K	250
$23.0-50.0K	288
$12.0-23.0K	266
$8.0-12.0K	161
less than $8.0K	121

Because our problem was a multi-class classification problem, we used relevant metrics, weighted-average f-1, weighted-average recall and weighted-average precision to evaluate the model performances.

Results

After comparison, we reached that Random Forest model performed the best among the four algorithms we used given our training data set. We also noticed that the Random Forest model required the longest fit time for our data though all four algorithms were reasonably time-efficient and computationally inexpensive. Table 2 showed fit time, weighted-average f-1, weighted-average recall and weight-average precision for each model.

Table 2: Random Forest performs the best

	Dummy Classifier	Logistic Regression	Multinomial Naive Bayes	SVC	Random Forest
fit time	0.03	0.31	0.03	0.19	0.43
score time	0.02	0.02	0.02	0.04	0.04
weighted f1	0.11	0.57	0.44	0.52	0.63
weighted recall	0.27	0.57	0.45	0.52	0.63
weighted precision	0.07	0.58	0.47	0.53	0.64

To further tune the Random Forest model, we conducted hyperparameter optimization and optimized the maximum number of features, class weight (balanced or none) and maximum number of depths. The maximum number of depths was important for tree-based models like Random Forest to avoid overfitting.

Our tuned classification model using Random Forest algorithm had a weighted-average f-1 score of 0.69, a weighted-average recall of 0.69 and a weighted-average precision of 0.69 when tested using our test data. Table 3 showed the f-1 score, recall, and precision for each class. The test scores were not the most ideal but the model performed reasonably well comparing to the base-line model as well as the other algorithms that we explored.

Table 3: Test Scores of the Tuned Random Forest Model

	precision	recall	f1-score
less than $8.0K	0.79	0.70	0.74
$8.0-12.0K	0.74	0.68	0.71
$12.0-23.0K	0.61	0.60	0.60
$23.0-50.0K	0.58	0.63	0.61
over $50.0K	0.81	0.83	0.82
weighted average	0.69	0.69	0.69

Conclusion

We provided a machine learning model to predict the funding ranges of Canadian art projects using the Random Forest algorithm. This model achieved some improvements over our base-case dummy classifier and other algorithms that we explored, Logistics Regression, Multinomial Naïve Bayes, and Support Vector Classification (SVC). We also reached that categorical features that stated the basics of the art project, such as location and presenter type, but did not contain information on the visual properties could reasonably predict the funding ranges approved by the Canadian Arts Presentation Fund.

However, our model was not without limitations. We could further explore techniques to solve class imbalance problem for the Multinomial Naïve Bayes algorithm. We might improve the model performance by adjusting the cutoffs for the ranges to create more balanced classes. We could expand our choices by exploring other algorithms such as k-Nearest Neighbors. Moreover, we could add some features that were indicative of visual properties of the art projects, such as the images of the costumes, to see if the model would perform better.

References

Aubry, Mathieu, Roman Kräussl, Gustavo Manso, and Christophe Spaenjers. 2022. "Biased Auctioneers." *SSRN*. Journal of Finance. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3347175#.

Ayub, Rafi, Cedric Orban, and Vidush Mukund. 2017. "Art Appraisal Using Convolutional Neural Networks." *Cs229: Machine Learning*. Stanford University. http://cs229.stanford.edu/proj2017/final-reports/5229686.pdf.

Canada. n.d. *Canada Arts Presentation Fund*. The Government of Canada. https://www.canada.ca/en/canadian-heritage/services/funding/arts-presentation-fund.html.

"Grouped Arts Evaluation: Canada Arts Presentation Fund, Canada Cultural Spaces Fund, and Canada Cultural Investment Fund 2013-14 to 2017-18_2019." 2019. The Government of Canada. https://www.canada.ca/en/canadian-heritage/corporate/publications/evaluations/grouped-art-evaluation.html#a1.

Harris, Charles R., K. Jarrod Millman, Stéfan J van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, et al. 2020. "Array Programming with NumPy." *Nature* 585: 357–62. https://doi.org/10.1038/s41586-020-2649-2.

McKinney, Wes et al. 2010. "Data Structures for Statistical Computing in Python." In *Proceedings of the 9th Python in Science Conference*, 445:51–56. Austin, TX.

Pedregosa, Fabian, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, et al. 2011. "Scikit-Learn: Machine Learning in Python." *Journal of Machine Learning Research* 12 (Oct): 2825–30.

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