2020 Summer Olympics Predictions Using Machine Learning

Brian C. Dobkowski
Master's Student
Department of Mechanical Engineering
Stanford University
bdobkows@stanford.edu

A dataset of world development indicators is used to estimate the medal counts of each country in the 2020 Summer Olympic Games. Historic Olympic performance data was merged with historic world economic indicators in order to set the stage for a regression problem in which the efficacy of multiple machine learning models is analyzed. In the course of experimentation, it was found that a binary classification algorithm can greatly aid in the formulation of the regression problem. A number of classification and regression models are experimented upon, and a final solution is given to predict the outcomes of the 2020 Summer Olympics.

1 Introduction

The Summer Olympic Games always offers an exciting contest for which country can earn the most medals, and the 2020 Summer Olympics, planned to take place in 2021 due to COVID-19, should prove no exception.

This paper concerns a high-level approach to make predictions about the 2020 games. That is, indicators such as GDP, population, land area, etc. are used on a country-level scale, as opposed to looking at individual athlete performance. The problem is best modeled as a regression problem: given the chosen indicators, how many medals will a nation win in the Olympic games this year? To accomplish the task, a dataset is constructed for training and validation, and a number of classification and regression models are iterated upon in the development of a two-phase algorithm design.

Seeing as the 2020 Olympic Games have yet to take place, this paper will use the 2016 games as a dataset against which to test the effectiveness of the algorithm, which can then later be applied to predict the 2020 outcomes.

2 Related Work

Olympic medal forecasting is a common outlet for statistical estimation techniques. Goldman Sachs in an Olympics-related economics study found that the economic and demographic features GDP and population tend to dominate the prediction landscape, which has been corroborated in other sources [1] [2] [3].

In [4], the authors find that a two-phased approach to the Olympic estimation problem bodes well, and that a Random Forest Regressor tends to provide the best results over boosting methods and neural networks - this informed the model selection in this paper's Models and Model Selection section. Interestingly, they also find that the COVID-19 crisis does not have a significant impact on the 2020 medal predictions, which gives reason for omitting COVID-related statistics in this study. Other researchers have demonstrated success using a MLP neural network model in olympic medal predictions, which informed the use of a MLP in this paper [5].

Many scholars, such as Scelles, Andreff et. al., agree that classically speaking, the Tobit and Hurdle models offer fair representations of the data, given that they naturally excel with distributions of data that have a large point mass at zero [6]. Zhao, Qian, and Yang attempt to tackle the point-mass problem directly by developing a model of the data that inherently includes this point-mass at zero when building a gradient boosted Tweedie model to predict insurance claims [7]. These efforts inspired the idea in this paper to consider a two-phased approach of first classifying the data to eliminate the training examples with zero medals won, and then using regression algorithms to fit the remaining data.

3 Data Collection and Inspection

Two main datasets were used in this project. One was a Kaggle dataset containing Olympic medal results on an individual athlete basis. This data can be found in [8]. To get the socioeconomic indicators, a csv of World Development Indicators was downloaded from worldbank.org in [9]. A considerable amount of work was required to merge the two datasets and resolve compatibility issues to create a coherent dataset for learning.

3.1 Dataset Organization and Feature Selection

It was decided that the best way to organize the data would be to have each country's performance in a given sum-

Features Selected		
Year	GDP	GDP Per Capita
GDP Growth	% World GDP	Total Pop.
% World Pop.	Pop. Growth	Total Land Area
Medals Last Games	Total Medals	Total Athletes
% Athletes		

Table 1. Features Selected

mer Olympics year be a training example. The output data would be medals won. The country would be dropped from the set of features, since it is desired to study the medaling performances as a function of economic indicators independent of country.

The features chosen are shown in Table 1, and the data was split according to Table 2 for use in model iteration and selection.

3.2 Dataset Nuances

Below is a list of peculiarities in the data that were resolved.

1. Only one medal was counted for each event, regardless of team size.
2. Training examples with any null data have been removed entirely. Figure 1 shows the sparsity in the data.
3. USA and a number of other countries boycotted the Olympics in 1980 due to the Cold War. The Soviet Union led a number of countries to boycott the following games in 1984. For this reason, and because less data was available from earlier in the 20th century, only data from 1988 onwards was used.
4. Country naming discrepancies between the two data sources had to be resolved.
5. Some features (e.g., the percentages) had to be engineered.

3.3 Preliminary Data Analysis

To get a sense for which features have a positive correlation on the number of Olympic medals won in a given year, and also to get a sense for how these features are distributed, scatter plots were created on all the features vs the number of medals won.

Figure 2 shows an example of the plots that were made from each feature. Observations from these plots will be drawn upon later in the Experiments section.

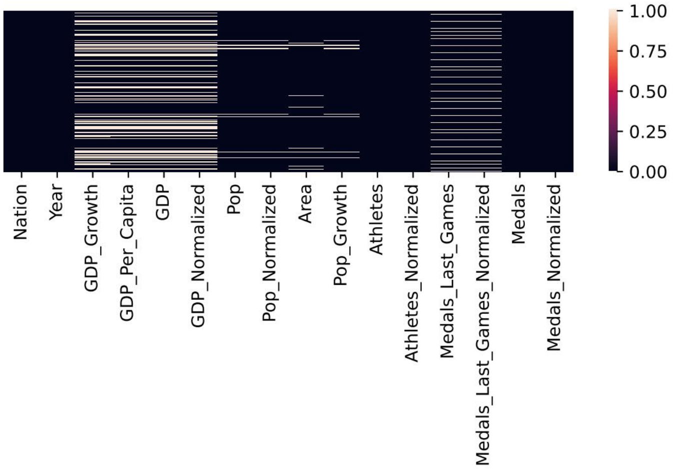

Fig. 1. Heatmap showing the null values over all training examples for given features. Examples with any null space have been removed.

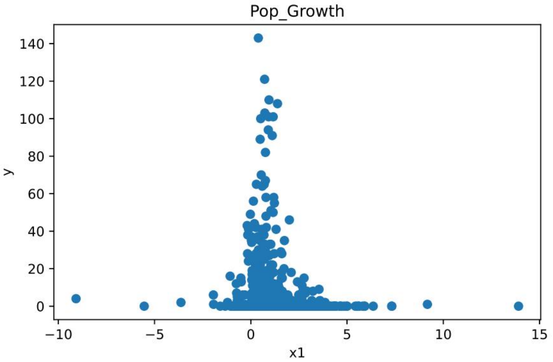

Fig. 2. Scatter plot of population growth (percentage) vs total medals won for all training examples

Purpose	Years Used
Model Tuning	1988-2008
Model Validation and Selection	2012
Algorithm Test	2016

Table 2. Train, Valid, Test Split

4 Models and Model Selection

4.1 Objective

The average squared loss (Eq. 1) was taken as the primary objective in this regression problem. Each model would aim to minimize the squared loss between the predicted and actual medal counts. Another objective was also considered: the average squared loss of the top 10 scoring countries in a given olympic games (Eq. 2). The thinking behind this loss function is that the top scoring countries' counts matter more if this algorithm is to predict the winning country correctly. Due to an already small dataset, and the qualitative decision that estimating every country's medal count is more important than just predicting the winner, the first loss function was implemented as the objective when training each model. The squared loss of the top scoring countries is considered a "nice-to-have" and though not implemented as an objective function, was still included as a metric.

$$\frac{1}{n}\sum_{i=1}^{n}(y_{predict}^{(i)} - y_{valid}^{(i)})^2 \tag{1}$$

$$\frac{1}{n}\sum_{i=1}^{n}(y_{predict}^{(i)} - y_{valid}^{(i)})^2 1_{\{y_{valid}^{(i)} \in top\ ten(year^{(i)})\}} \tag{2}$$

4.2 Regression Models

Below is a list of all models considered in this project. All models' objective functions were the average squared loss (Eq. 1).

1. **Linear Regression**
Linear regression served as the baseline model, and was also used as an intermediary model to explore kernel functions and locally weighted linear models, so as to introduce a method to minimize the second loss function in Eq. 2.
2. **Regularized Linear Regression**
To minimize any overfitting brought on by linear regression, both Ridge and Lasso regression were used to obtain fits of the data. By minimizing the L2-norm and L1-norm of the fitted coefficients, these algorithms would reduce the size of the linear coefficients and thereby avoid overfitting.
3. **Poisson Regression**
The Poisson distribution was included because it is a classic model for discrete counts. However, the particularity that the Poisson expected value equals its variance causes issues in a problem where there is a large number of points clustered at zero [6]. This distribution was included anyway out of desire to understand its struggles in this context.
4. **Support Vector Regression**
Support Vector Machines were also used in regression analysis because of their inherent ability to reduce overfitting by including a weight minimizer in their objective function. Kernels were heavily explored using SVRs to examine the data in different dimensions, to try to find hyperplanes separating the data and leverage the nonlinearities of the features.
5. **Random Forest Regressor**
To include a less classical (and less ”linear”) algorithm, but also reduce the overfitting that decision trees introduce, a random forest regressor was used.

4.3 Classification Models

As mentioned earlier, one of the peculiarities of this dataset is that many countries do not win any medals in the Olympic games. Therefore, this takes the form of a point mass at zero when trying to fit regressors to the data. In classical statistics, this may be modeled as a mixture distribution like Eq. 3, using the Poisson distribution as an example.

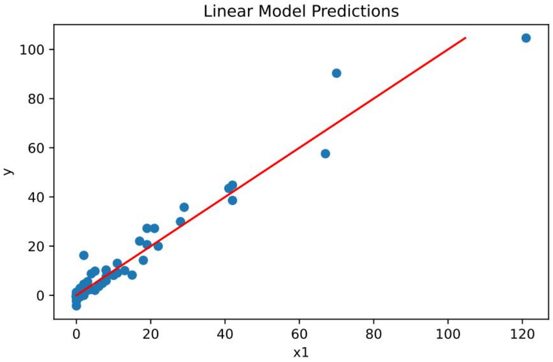

Fig. 3. Baseline results for estimated medals won (y-axis) vs actual medals won (x-axis) in 2016 Summer Olympic Games.

Using this equation, a maximum likelihood estimate can be derived for the parameter γ.

$$X_{(i)} \sim \gamma\delta_0 + (1-\gamma)Poisson(\lambda) \tag{3}$$

By training a binary classification algorithm to determine whether (0) no medals were won, or (1) at least one medal was won, the same objective can be accomplished. The result is a two-stage machine learning algorithm - first a binary classifier is trained on the data, and the regression algorithm is trained on the examples which are predicted (1) by the classifier. The below binary classifiers were examined for this application.

1. **Logistic Regression**
This model would serve as the baseline for all classification models by trying to create a linear decision boundary between the training examples.
2. **Support Vector Classification**
Support vectors would allow the application of the kernel techniques from the regression section of this project, and also introduce natural regularization.
3. **Gaussian Naive Bayes**
The naive assumption that all examples are independent may not hold, especially because the medals won by a country in one year directly affects the medals won by another country in that same year. However, between Olympic games, this assumption may be more valid.
4. **Multilayer Perceptron**
This algorithm attempts to leverage the nonlinearities in the data, especially when experimenting with the number of hidden layers a different types of activation functions.
5. **Random Forest Classifier**
To again introduce a decision tree algorithm into the methods, and study nonlinearities in the data, a random forest classifier was used.

5 Experiments

5.1 Feature, Variable Normalization

One of the key issues with this dataset is that a different number of medals are awarded each Olympic games, due to the addition of sports over time. This creates uncertainty in the predictions of the medal counts for a given Olympic games since the training examples include medal counts from a variety of different games. Two things were explored to mitigate this effect.

1. Instead of total medals won as the dependent variable, a variable "Percentage of Total Medals Won" was created. The regression algorithms would be trained to estimate this fraction, and then the estimate would be multiplied by the number of total medals awarded that year to obtain estimates of each country's final score.
2. Two additional features were added: year of Olympic games and total medals awarded that year. The dependent variable was kept at total medals won by a specific country in a specific year. This way, the correlation between year and total medals would be implicitly given to the algorithm, which could learn the correlation on its own. This method was chosen as the winner due to its cleaner implementation, and slightly better results.

5.2 Kernels and Feature Maps

From exploratory data analysis earlier in the phase of the project, it could be seen that the relationship between the total medals won and certain features was not always linear. First, some manual feature mappings, in the form of simple functions, were experimented with using the baseline linear regression model. For example, the $\log n_{atheletes}$ was tried instead of $n_{athletes}$, due to the relationship seen in the exploratory data analysis plots. However, due to the wide distribution of behavior across the feature space, none of these maps provided much benefit.

Different off-the-shelf kernels were also used to explore nonlinearities in the data. The main kernels examined were linear, polynomial (degrees of 2 and 3), radial basis function (RBF), and sigmoid kernels. The polynomial kernels were motivated by desires to fit the correlations of certain features that were known to be positive, but not exactly linear. The RBF kernels were inspired by Gaussian-looking behavior exhibited in the features such as "GDP Growth" and "Population Growth" where a positive correlation was not always found (see Fig. 2).

Interestingly, the kernels that performed the best were simple linear kernels. This is due to the fact that in general, the higher the feature values, the more medals won by a particular country (e.g. higher population, higher GDP, bigger land area had an overwhelmingly positive, and linear-looking, correlation with total medals won).

5.3 Hyperparameter Tuning

In order to tune the chosen regression and classification models, cross-validation techniques were used on the "Model Tuning" data in Table 2. K-fold cross validation was used for most algorithms, sometimes with $K = 3$, and other times with $K = 5$. With K-fold cross-validation, K randomly selected divisions of the training data would be created. By shuffling the data in this way and providing multiple test sets within the training space, it is less likely that the models will overfit the training data.

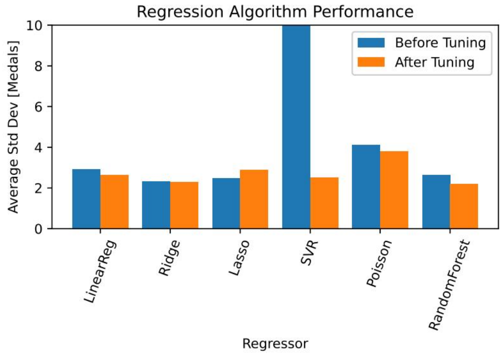

Fig. 4. Performance of all regression models before and after tuning. Dependent variable is the loss in Eq. 1 using the validation set (2012).

For the more classical algorithms, an exhaustive grid search cross validation technique was used, which tested all combinations of parameters in order to come up with the optimal fit of the 1988-2008 data. The advantage of using the 2012 data to validate these models was to prove that the cross-validated models were not overfitting the 1988-2008 data.

For the more computationally intensive algorithms, such as the Multilayer Perceptron and Random Forest algorithms, a randomized search cross-validation technique was used. In this technique, not every pairing of parameters was attempted on the tuning data, but parameter distributions were provided, and a subset of the random combinations of these parameters were evaluated on the K folds. The distributions of parameters used in the grid search and randomized search tuning can be seen in the code.

Figure 4 shows the performance of the regression models before and after cross-validation hyperparameter tuning. For both of these experiments, the baseline Logistic Regression classifier was used, so the regressors only fit the data that the classifier predicted as $y = 1$ (at least one medal won). Interestingly, the performance of some models were worse after parameter tuning (see Lasso performance), because the models overfit the tuning set and offered a worse fit than the default values on the validation set. Similar figures were developed for evaluating the classification algorithms, but those have not been included.

5.4 Weighted Regression

Locally weighted regression was explored in the linear regression models (linear, ridge, lasso). This was an attempt to reduce the loss in Eq. 2 without drastically impacting the

Classifier	Accuracy	Regressor	Avg Std Dev
Logistic Reg.	0.877	Weighted LR	2.63
SVM Clf	0.884	SVM Reg	2.50
Random Forest	0.877	Random Forest	2.20
Gaussian NB	0.891	Poisson	3.75
MLP	0.855	Lasso	2.87
		Ridge	2.27

Table 3. Results of Models on Validation Set

foremost objective in Eq. 1.

$$w^{(i)} = exp\left(\frac{-(y^{(i)} - y)^2}{2\tau^2}\right) \tag{4}$$

Equation 4 shows the weighting scheme used, where y and τ were parametrically varied. y can be thought of as the number of medals won where we want to give the most weight to the input feature, and τ is a standard-deviation-like division factor. Values of 50 and 1, respectively, were found to be optimal for y and τ.

6 Results

Table 3 shows the results of the tuned classifier models on the 2012 validation set. Accuracy, or the proportion of examples estimated correctly over all examples, is the primary metric used, as it was the objective function driving each of these classifiers in training. Given the best performing classifier, the Gaussian Naive Bayes model, the classifier predictions were fed into the regression algorithms, which performed according to Table 3. It was decided to choose the final regressor to minimize a linear combination of Eq. 1 and 0.25 * Eq. 2, so as to not use a model that will wildly favor fitting the lower scoring countries. Thus, the optimal algorithm in this problem is a two stage algorithm consisting of a Gaussian Naive Bayes classifier, and tuned Ridge Regression model. To predict the 2016 outcomes, the algorithm was trained on the tuning and validation data - results of the predictions can be seen in Fig. 5. This final prediction had an average standard deviation of 2.26. For reference, the baseline linear regression model (with no classification step) had an average standard deviation of 3.25.

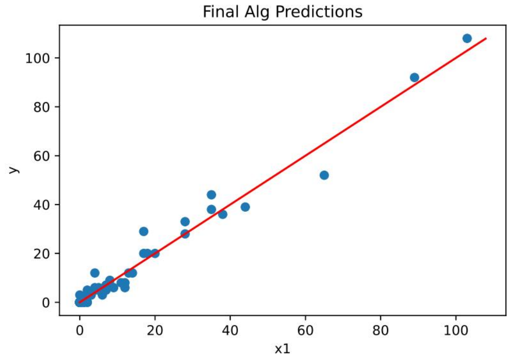

Fig. 5. Final algorithm performance on test set (2016). Y-axis is predicted medals won, whereas the x-axis is actual medals won. Compare with baseline model in Fig. 3.

Country	Actual Medals	Predicted Medals
USA	103	103
China	89	85
UK	65	52
Germany	44	44
Japan	38	45

Table 4. Final algorithm predictions for 2016 Olympic medal counts for top five scoring countries

7 Conclusions and Future Work

Through this work, a two-step machine learning approach to predicting the 2020 Olympic Games medal counts has been proposed. The first step applies binary classification algorithms to determine which countries even medal. Given these results, several regression algorithms are investigated to form a fit to the resultant data. The application of this two-phased approach helped mitigate the issue of a large presence of the dependent variable (medals won) at 0, and was found to greatly improve performance.

The most obvious next step is to actually make predictions for the 2020 games - this was not done in this paper because there would be no way to grade the predictions since the games have yet to happen, and the dataset used in this project did not have 2020 economic data.

To improve performance, a few adjustments can be made to the data and features. A lot of the important world development indicator information does not exist for the smaller nations. The solution for this project was to remove all training examples with null data entries - however, this resulted in a relatively small training set. Application of the imputation technique may help improve performance [10]. A last data-related improvement would be including data from in-between the games in the predictions, and not only the years that the Olympics takes place.

In terms of algorithm development, it would be better to pare down the number of considered models so that more work can go into optimizing and tuning this models. Also, a new objective for training that can incorporate loss functions 1 and 2 simultaneously would be beneficial, so that the second objective is actually optimized by the algorithm itself.

8 Code

The cross-validation and most of the models were implemented using the Scikit-Learn library [11]. All code for this project can be found at `https://github.com/bdobkowski/2020_Olympics_Predictions`.

References

[1] Sachs, G., 2012. “The olympics and economics 2012”. *Goldman Sachs Global Economics, Commodities and Strategy Research*.

[2] Jayantha, K., and Ubayachandra, E., 2015. “Going for gold medals: factors affecting olympic performance”. *International Journal of Scientific and Research Publications,* **5**(6), pp. 2250–3153.

[3] Shailaja, V., Lohitha, R., Musunuru, S., Reddy, K. D., and Priya, J. P., 2020. “Predictive analytics of performance of india in the olympics using machine learning algorithms”. *International Journal,* **8**(5).

[4] Schlembach, C., Schmidt, S. L., Schreyer, D., and Wunderlich, L., 2020. Forecasting the olympic medal distribution during a pandemic: a socio-economic machine learning model.

[5] Fazlollahi, P., Afarineshkhaki, A., and Nikbakhsh, R. “Predicting the medals of the countries participating in the tokyo olympic games (2020) using the test of networks of multilayer perceptron (mlp)”. *Annals of Applied Sport Science*, pp. 0–0.

[6] Scelles, N., Andreff, W., Bonnal, L., Andreff, M., and Favard, P., 2020. “Forecasting national medal totals at the summer olympic games reconsidered”. *Social Science Quarterly,* **101**(2), pp. 697–711.

[7] Zhou, H., Qian, W., and Yang, Y., 2020. “Tweedie gradient boosting for extremely unbalanced zero-inflated data”. *Communications in Statistics-Simulation and Computation*, pp. 1–23.

[8] rgriffin, 2018. “120 years of olympic history: athletes and results”. *Kaggle*.

[9] Bank, T. W., 2021. World development indicators.

[10] Lakshminarayan, K., Harp, S. A., Goldman, R. P., Samad, T., et al., 1996. “Imputation of missing data using machine learning techniques.”. In KDD, pp. 140–145.

[11] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E., 2011. “Scikit-learn: Machine learning in Python”. *Journal of Machine Learning Research,* **12**, pp. 2825–2830.