Comparing the Performance of 17 Machine Learning Models in Predicting Human Population Growth of Countries

Mohammad Mahmood Otoom

Department of Computer Science and Information, College of Science in Zulfi
Majmaah University
Al-Majmaah 11952, Saudi Arabia

Summary
Human population growth rate is an important parameter for real-world planning. Common approaches rely upon fixed parameters like human population, mortality rate, fertility rate, which is collected historically to determine the region's population growth rate. Literature does not provide a solution for areas with no historical knowledge. In such areas, machine learning can solve the problem, but a multitude of machine learning algorithm makes it difficult to determine the best approach. Further, the missing feature is a common real-world problem. Thus, it is essential to compare and select the machine learning techniques which provide the best and most robust in the presence of missing features. This study compares 17 machine learning techniques (base learners and ensemble learners) performance in predicting the human population growth rate of the country. Among the 17 machine learning techniques, random forest outperformed all the other techniques both in predictive performance and robustness towards missing features. Thus, the study successfully demonstrates and compares machine learning techniques to predict the human population growth rate in settings where historical data and feature information is not available. Further, the study provides the best machine learning algorithm for performing population growth rate prediction.
Key words:
Machine Learning, Population Growth rate, decision tree, random forest, artificial neural network, bagging, boosting.

1. Introduction

Human population dynamics, especially human population growth rate, is critical for any real-world field applications. Traditionally, the population growth rate is estimated from a series of the census for an area using pre-defined empirical methods[1, 2]. Census parameters may be collected physically or using various methods. A study in Bangladesh uses iterative proportional fitting to estimate infant mortality rate of an area[3]. A study in Nepal uses the change in the number of active cellular users to estimate population change in an area[4]. A study in France and Portugal uses cellular data to estimate the population density of an area[5].
Machine learning (ML) models are gaining prominence in human population sciences. A Peruvian study uses regression and decision trees to determine an area's population density[6]. Studies from USA, India, Nigeria and Bangladesh have used different base learners like regression, decision trees, k-Nearest Neighbors and artificial neural network and ensemble learners like bagging and boosting to determine the population of an area[7-11]. ML model employs techniques that learn and train using the existing data to build the relationship between the input data and the outcome. Successfully trained ML model could predict the outcome of the unseen data.
ML methods have not been used to predict the human population growth rate for a region. Further, it is crucial to create an ML model when the study area's historical data is not available and/or some input features are missing since missing features is a common issue in the real world. However, ML encompasses multiple techniques whose performance varies with application domain[12]. A Bangladesh study for population prediction showcases the high performance of gradient boosting for data-based input, but artificial neural network performed best for satellite image-based input[9]. Thus, it is essential to evaluate different ML techniques before preparing an ML model for the problem at hand. The current study strives to address these research gaps.
The current study compares the performance of 17 ML models in predicting the human growth rate of an area. We prepare the models for the scenario where historical data for an area may not be available and with missing features. Our selected algorithms include base learners and ensemble learners. While some learners have been used previously in other studies related to population dynamics, some are not tested in the population growth rate prediction. Such an extensive model evaluation has not been done in the context of population growth rate prediction. The article structure is as follows: Section 2 introduces the methodology used in the study. Section 3 summarises the results and discussions with the conclusion in Section 4.

Manuscript received January 5, 2021
Manuscript revised January 20, 2021
https://doi.org/10.22937/IJCSNS.2021.21.1.28

2. Methodology

The study performs a five-stage methodology to evaluate the different ML models in predicting an area's population growth rate. The stages description is given below.

2.1 Stage 1: Dataset collection Stage

The study uses countries as the study area for current research. Accordingly, the data used for evaluating the ML models in predicting the human population growth rate is collected. The study uses the United Nations database[13] to collect the data. The seven demographic features are used in the study as input. These input features for any given area are the total human population, human population density, infant mortality rate (IMR), under-five mortality rate, life expectancy at birth, female life expectancy at birth and total fertility rate. These features are commonly associated with the population growth rate of any area.

Total fertility rate depicts the number of children born to a woman in a lifetime which influences the birth rate of population and hence population growth rate[14]. The population growth rate is estimated using the birth rate, mortality rate and migration rate of an area. Hence, change in mortality rate or life expectancy would influence growth rate[15]. Population density influences the development status of an area[16] which alters the growth rate of an area[17]. The database contains 1950-2015 data for 205 countries, providing a total sample size of 13530.

2.2 Step 2: Pre-Processing and Transformation Stage

The UN data is cleaned and transformed to enable analysis. The outcome, i.e., population growth rate (*pgr*), is continuous and is transformed into a categorical outcome. The study categorized the outcome into 'high growth rate' (*HGR*), 'medium growth rate' (*MGR*) and 'negative to low growth rate' (*NLGR*) using K-mean clustering with Euclidean distance as a distance metric. K-mean clustering is a commonly used categorization approach that groups similarly related values into a single cluster[18].

2.3 Step 3: Feature Selection Stage

Highly correlated features are removed from the analysis. The study uses pearson correlation coefficient to remove highly correlated features. Consequently, study drops the infant mortality rate (IMR) and female life expectancy at birth.

2.4 Step 4: Model Training Stage

All the 17 ML techniques are trained in this step to predict *pgr* from the input features set. The study has trained nine base learners and eight ensemble learners to build models. The nine base learners used in the model are artificial neural network (ann), decision tree (*dt*), k-nearest neighbors (*knn*), linear discriminant analysis (*lda*), logistic regression (*lr*), Localized Generalized Matrix Learning Vector Quantization (*Lgmlvq*), naïve Bayes (*nb*), quadratic discriminant analysis (*qda*) and support vector classifier (*svc*). The eight ensemble learners are created using either bootstrap aggregation (bagging) or boosting approach. The different bagging based learners used in the study are bagging based linear discriminant analysis (*ldaBG*), bagging based logistic regression (*lrBG*), bagging based naïve Bayes (*nbBG*), bagging based quadratic discriminant analysis (*qdaBG*) and random forest (*rf*). The different boosting based learners used in the study are adaptive boosting based decision tree (*dtADA*), adaptive boosting based logistic regression (*lrADA*) and adaptive boosting based naïve Bayes (*nbADA*).

The study divided data into training and dataset (80%) and validation dataset (20%) through stratified sampling with stratification done on *pgr*. Further, all the features are normalized. Before model training, it is important to optimize the hyperparameters used in different learners. Accordingly, hyperparameter tuning (Supplementary 1) is performed before training using three-fold cross-validation. We use a grid search approach to identify the best hyperparameters and use them for model building. The hyperparameter tuning and model training is performed in *Python* using the *Scikit* library[19]. The learners are trained in six different scenarios based on missing input features, which generates six different models for each learner. The different scenarios used in the study are given in Table 1.

Table 1: Different scenarios used in the study.

Scenario	*Missing Input Feature*	*Code*
1	None	ALL
2	Life Expectancy at Birth (LEB)	NLEB
3	Population (Pop)	NPOP
4	Population Density (PopD)	NPD
5	Total Fertility Rate (TFR)	NTFR
6	Under Five Mortality Rate (U5MR)	NUMR

2.5 Step 5: Model Evaluation Stage

We evaluate the trained models for prediction performance of pgr on test dataset. Accuracy is used as the performance metric for the study and evaluated the overall prediction performance of pgr and prediction performance of individual pgr category. Accuracy metric is calculated as follows:

Accuracy = number of correctly predicted instances/total number of instances

3. Results and Discussions

3.1 Descriptive Analysis

Table 2 provides descriptive statistics of the dataset. The range of the values is large and distinct across the features. Further, *pgr* is directly proportional to mortality and fertility rate but inversely proportional to population, population density and life expectancy. In regards to *pgr* categorization, very few samples (2.36%) are in the *HGR* category.

Table 2: Table Type Styles

#	*Parameter*	*Full Sample (n=13530)*	*Low (n=5979)*	*Medium (n=7205)*	*High (n=346)*
1	Population (in `00,000)*				
	Mean (sd)	241.6 (968.1)	302.5 (1168.6)	201.8 (787.3)	17.8 (28.9)
	Median (Min,Max)	45.0 (0.1, 13970.3)	50.1 (0.3, 13970.3)	43.6 (15.1, 11535.7)	4.8 (0.2, 161.7)
2	Pop Density (per sq km) *				
	Mean (sd)	198.2 (929.9)	205.9 (654.8)	196.4 (1124.4)	102.5 (261.3)
	Median (Min,Max)	50.7 (0, 20098.4)	85.7 (0.3, 8320)	31.7 (0.3, 20098.4)	15 (0, 1864.4)
3	Life expectancy at birth (years)*				
	Mean (sd)	62 (12.2)	68.4 (10.6)	56.6 (10.8)	61.9 (11.4)
	Median (Min,Max)	64.8 (18.9, 83.8)	70.7 (18.9, 83.8)	57 (30.1, 83.7)	64.3 (34.8, 79.2)
4	Under-five mortality (deaths under age 5 per 1000 live births) *				
	Mean (sd)	99.5 (92.1)	54 (78.1)	137.1 (85.9)	101.4 (85.8)
	Median (Min,Max)	68.2 (2.1, 435.2)	27 (2.1, 435.2)	123.7 (2.6, 424.8)	74.6 (7.5, 331.7)
5	Total fertility (live births per woman) *				
	Mean (sd)	4.3 (2)	2.8 (1.5)	5.5 (1.5)	5.6 (1.8)
	Median (Min,Max)	4.2 (0.8, 8.9)	2.3 (1, 8.1)	5.8 (0.8, 8.9)	6.3 (1.6, 8.4)
6	Population growth rate percentage [OUTCOME] *				
	Mean (sd)	1.9 (1.5)	0.7 (0.8)	2.7 (0.6)	7.4 (2.6)
	Median (Min,Max)	1.9 (-7.1, 17.7)	0.8 (-7.1, 1.7)	2.6 (1.7, 5)	6.4 (5, 17.7)

* Results are significant for p<0.001

3.2 ML Models Evaluation

The study compared ML models using the accuracy metric to classify the country's *pgr* based on the demographic data. The comparison is performed in six scenarios based on the missing feature. As shown in Table 3, the ML models ability to predict *pgr* of the country using all the available demographic data varied from 0.58 to 0.96; this suggests that the choice of ML model is critical in developing the predictive model for *pgr*. Random forest (*rf*) provided the best model to predict *pgr,* while naïve Bayes (*nb*) provided the worst model. Secondly, ensemble models performed better or similar to their base models. For example, quadratic discriminant analysis (*qda*) based classifier had an accuracy of 0.61 but its ensemble classifier, i.e., bagging based quadratic discriminant analysis (*qdaBG*), had an accuracy of 0.8.

Table 3: Prediction performance of different machine learning models in terms of overall accuracy in the test dataset.

#	*Technique*	*Overall pgr Performance (Accuracy)*					
		Scenario					
		All	*NLEB*	*NPOP*	*NPD*	*NTFR*	*NUMR*
1	ann	0.88	0.88	0.88	0.88	0.89	0.88
2	dt	0.94	0.94	0.94	0.94	0.94	0.94
3	dtADA	0.94	0.93	0.93	0.93	0.94	0.93
4	knn	0.93	0.93	0.93	0.93	0.93	0.93
5	lda	0.82	0.82	0.82	0.82	0.82	0.82
6	ldaBG	0.82	0.82	0.82	0.82	0.82	0.82
7	Lgmlvq	0.86	0.86	0.86	0.86	0.86	0.86
8	lr	0.84	0.84	0.84	0.84	0.84	0.84
9	lrADA	0.83	0.83	0.83	0.83	0.83	0.83
10	lrBG	0.84	0.84	0.84	0.84	0.84	0.84
11	nb	0.58	0.58	0.58	0.58	0.58	0.58
12	nbADA	0.78	0.78	0.78	0.78	0.78	0.78
13	nbBG	0.80	0.79	0.80	0.80	0.79	0.79
14	qda	0.61	0.61	0.61	0.61	0.61	0.61
15	qdaBG	0.80	0.80	0.80	0.80	0.80	0.80
16	rf	0.96	0.95	0.95	0.95	0.96	0.96
17	svc	0.86	0.86	0.86	0.86	0.86	0.86

With regards to individual *pgr* category, our study observes that most learners, except artificial neural network (*ann*) and decision tree-based learners, struggled to identify the *HGR* category (Table 4). Among the learners that could predict the HGR category, rf provided the best predictive performance and performance remained consistent with missing features. In the *MGR* and *NLGR* category, all learners could perform prediction, but *rf* provided the best prediction performance (Table 5 and Table 6). Interestingly, *ann* could not provide the best performance in *pgr* prediction, which could be due to shallow hidden layers.

Table 4: Prediction performance of different machine learning models for HGR category in the test dataset.

#	*Technique*	*HGR Category Performance (Accuracy)*					
		Scenario					
		All	*NLEB*	*NPOP*	*NPD*	*NTFR*	*NUMR*
1	ann	0.67	0.67	0.60	0.67	0.77	0.65
2	dt	0.75	0.75	0.75	0.75	0.75	0.75
3	dtADA	0.76	0.84	0.73	0.73	0.73	0.72
4	knn	0.69	0.69	0.69	0.69	0.69	0.69
5	lda	0.00	0.00	0.00	0.00	0.00	0.00
6	ldaBG	0.00	0.00	0.00	0.00	0.00	0.00
7	Lgmlvq	0.00	0.00	0.00	0.00	0.00	0.00
8	lr	0.00	0.00	0.00	0.00	0.00	0.00
9	lrADA	0.00	0.00	0.00	0.00	0.00	0.00
10	lrBG	0.00	0.00	0.00	0.00	0.00	0.00
11	nb	0.05	0.05	0.05	0.05	0.05	0.05
12	nbADA	0.00	0.00	0.00	0.00	0.00	0.00
13	nbBG	0.00	0.00	0.00	0.00	0.00	0.00
14	qda	0.07	0.07	0.07	0.07	0.07	0.07
15	qdaBG	0.00	0.00	0.00	0.00	0.00	0.00
16	rf	0.96	0.94	0.98	0.98	0.98	0.96
17	svc	0.00	0.00	0.00	0.00	0.00	0.00

Table 5: Prediction performance of different machine learning models for MGR category in the test dataset.

#	*Technique*	*MGR Category Performance (Accuracy)*					
		Scenario					
		All	*NLEB*	*NPOP*	*NPD*	*NTFR*	*NUMR*
1	ann	0.88	0.88	0.88	0.88	0.88	0.88
2	dt	0.94	0.94	0.94	0.94	0.94	0.94
3	dtADA	0.93	0.93	0.94	0.94	0.94	0.93
4	knn	0.94	0.94	0.94	0.94	0.94	0.94
5	lda	0.84	0.84	0.84	0.84	0.84	0.84
6	ldaBG	0.84	0.84	0.84	0.84	0.84	0.84
7	Lgmlvq	0.84	0.84	0.84	0.84	0.84	0.84
8	lr	0.84	0.84	0.84	0.84	0.84	0.84
9	lrADA	0.82	0.82	0.82	0.82	0.82	0.82
10	lrBG	0.84	0.84	0.84	0.84	0.84	0.84
11	nb	0.87	0.87	0.87	0.87	0.87	0.87
12	nbADA	0.77	0.77	0.77	0.77	0.77	0.77
13	nbBG	0.84	0.84	0.84	0.84	0.84	0.84
14	qda	0.91	0.91	0.91	0.91	0.91	0.91
15	qdaBG	0.84	0.84	0.84	0.84	0.84	0.84
16	rf	0.95	0.95	0.95	0.95	0.96	0.95
17	svc	0.84	0.84	0.84	0.84	0.84	0.84

Table 6: Prediction performance of different machine learning models for NLGR category in the test dataset.

#	Technique	*NLGR Category Performance (Accuracy)*					
		Scenario					
		All	*NLEB*	*NPOP*	*NPD*	*NTFR*	*NUMR*
1	ann	0.89	0.89	0.90	0.89	0.90	0.89
2	dt	0.94	0.94	0.94	0.94	0.94	0.94
3	dtADA	0.95	0.93	0.94	0.94	0.94	0.94
4	knn	0.94	0.94	0.94	0.94	0.94	0.94
5	lda	0.81	0.81	0.81	0.81	0.81	0.81
6	ldaBG	0.81	0.81	0.81	0.81	0.81	0.81
7	Lgmlvq	0.89	0.89	0.89	0.89	0.89	0.89
8	lr	0.83	0.83	0.83	0.83	0.83	0.83
9	lrADA	0.84	0.84	0.84	0.84	0.84	0.84
10	lrBG	0.83	0.83	0.83	0.83	0.83	0.83
11	nb	0.79	0.79	0.79	0.79	0.79	0.79
12	nbADA	0.80	0.80	0.80	0.80	0.80	0.80
13	nbBG	0.75	0.74	0.76	0.76	0.74	0.75
14	qda	0.83	0.83	0.83	0.83	0.83	0.83
15	qdaBG	0.77	0.77	0.77	0.77	0.77	0.77
16	rf	0.96	0.96	0.96	0.96	0.96	0.96
17	svc	0.89	0.89	0.89	0.89	0.89	0.89

Interestingly, our study finds that models performance is robust with missing features, which could be due to multiple reasons. Firstly, large sample size (10,824) for training could provide sufficient information to the models for learning. Secondly, the preliminary analysis suggested an association between individual input features and *pgr*, which could have made the input features redundant. Thirdly, most *pgr* data is present in either *MGR* or *NLGR* category, which could hide the feature performance in the *HGR* category. Further, Table 4, 5 and 6 show that missing feature did not affect the performance in *MGR* or *NLGR* category. However, among the ML models, which could predict the HGR category, Artificial Neural Network (*ann*) and adaptive boosting based decision tree (*dtADA*) showed variation in *HGR* category performance with the missing feature. However, this variation is absent in the overall performance. Overall, *rf* has outperformed all the learners by providing the best and robust prediction performance for each *pgr* category.

4. Conclusion

The population growth rate is essential for any real-world decision making. One of the significant limitations is that current strategies rely on the use of historical data to predict the population growth rate in the area. The study successfully addresses the limitation of relying on the historical data of the area through ML models trained on global population growth rate behavior. Secondly, the study compared the 17 ML learners to predict the population growth rate in the test data with and without missing features. While all models could classify an area's population growth rate, random forest convincingly outperformed all the learners both in prediction and robustness. These findings suggest that random forest could be the best ML model to train the population growth rate models. Thirdly, the study provides *rf* models capable of predicting a region's population growth rate using all or any five of the demographic parameters: total human population, human population density, under-five mortality rate, life expectancy at birth, and total fertility rate.

The study has certain limitations that future research could focus on addressing. Future research could test the sub-national or supra national demographic data to perform model prediction for sub-national or supra-national regions. Further, the future study could incorporate more missing features and demographic details to make a more flexible model for predicting the population growth rate. Additionally, the future models could focus on predicting population growth rate over a time rather than at one single time point.

Acknowledgments

The author would like to thank the Deanship of Scientific Research at Majmaah University for supporting this work under project number No. R-2021-20

References

[1] K. W. Kintigh and M. A. Peeples, "Estimating Population Growth Rates and Instantaneous Population from Periodized Settlement Data," *Journal of Computer Applications in Archaeology,* vol. 3, 2020.

[2] C. Rich, "The measurement ofthe rate of population growth," *Journal of the Institute of actuaries», voi. LXV, parte I,* 1934.

[3] A. N. Rose and N. N. Nagle, "Validation of spatiodemographic estimates produced through data fusion of small area census records and household microdata," *Computers, Environment and Urban Systems,* vol. 63, pp. 38-49, 2017.

[4] R. Wilson, E. zu Erbach-Schoenberg, M. Albert, D. Power, S. Tudge, M. Gonzalez*, et al.*, "Rapid and near real-time assessments of population displacement using mobile phone data following disasters: the 2015 Nepal Earthquake," *PLoS currents,* vol. 8, 2016.

[5] P. Deville, C. Linard, S. Martin, M. Gilbert, F. R. Stevens, A. E. Gaughan*, et al.*, "Dynamic population mapping using mobile phone data," *Proceedings of the National Academy of Sciences,* vol. 111, pp. 15888-15893, 2014.

[6] W. Anderson, S. Guikema, B. Zaitchik, and W. Pan, "Methods for estimating population density in data-limited areas: Evaluating regression and tree-based models in peru," *PLoS one,* vol. 9, p. e100037, 2014.

[7] G. Bandyopadhyay and S. Chattopadhyay, "An Artificial Neural Net approach to forecast the population of India," *arXiv preprint nlin/0607058,* 2006.

[8] O. Folorunso, A. Akinwale, O. Asiribo, and T. Adeyemo, "Population prediction using artificial neural network," *African Journal of Mathematics and Computer Science Research,* vol. 3, pp. 155-162, 2010.

[9] N. Ahmed, N. A. Diptu, M. S. K. Shadhin, M. A. F. Jaki, M. F. Hasan, M. N. Islam*, et al.*, "Artificial Neural Network and Machine Learning Based Methods for Population Estimation of Rohingya Refugees: Comparing Data-Driven and Satellite Image-Driven Approaches," *Vietnam Journal of Computer Science,* vol. 6, pp. 439-455, 2019.

[10] V. Riiman, A. Wilson, R. Milewicz, and P. Pirkelbauer, "Comparing Artificial Neural Network and Cohort-Component Models for Population Forecasts," *Population Review,* vol. 58, 2019.

[11] M. M. Otoom, M. Jemmali, Y. Qawqzeh, K. N. SA, and F. Al Fay, "Comparative Analysis of Different Machine Learning Models for Estimating the Population Growth Rate in Data-Limited Area," *IJCSNS,* vol. 19, p. 96, 2019.

[12] N. Cesare, C. Grant, Q. Nguyen, H. Lee, and E. O. Nsoesie, "How well can machine learning predict demographics of social media users?," *arXiv preprint arXiv:1702.01807,* 2017.

[13] United Nations, World Population Prospects 2017. 2017.

[14] G. Nargund, "Declining birth rate in Developed Countries: A radical policy re-think is required," *Facts, views & vision in ObGyn,* vol. 1, p. 191, 2009.

[15] J. Bongaarts, "Human population growth and the demographic transition," *Philosophical Transactions of the Royal Society B: Biological Sciences,* vol. 364, pp. 2985-2990, 2009.

[16] R. Boucekkine, D. De la Croix, and D. Peeters, "Early literacy achievements, population density, and the transition to modern growth," *Journal of the European Economic Association,* vol. 5, pp. 183-226, 2007.

[17] D. Ahlburg and R. Cassen, "Population and development," 2008.

[18] D. Steinley, "K-means clustering: a half-century synthesis," *British Journal of Mathematical and Statistical Psychology,* vol. 59, pp. 1-34, 2006.

[19] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel*, et al.*, "Scikit-learn: Machine learning in Python," *the Journal of machine Learning research,* vol. 12, pp. 2825-2830, 2011.

SUPPLEMENTARY TABLE: HYPER-PARAMETER TUNING PERFORMED IN THE STUDY

ML technique	**Hyper-parameter (Range)**
svc	C (0.1 to 1), kernel (linear, poly, rbf, sigmoid), degree (2 to 5)
lr	Penalty (none, L1, L2, Elasticnet), l1_ratio (0.0 to 1.0)
knn	Neighbors (1 to 2706), distance weights (uniform, distance), metric (euclidean, manhattan, chebyshev, minkowski, hamming, canberra, braycurtis)
dt	Criterion (gini, entropy), maximum Depth of Tree (1 to 2706), minimum samples per leaf (1 to 2706), minimum samples per split (2 to 2706), maximum features in each tree (6, sqrt, log)
ann	Hidden layer nodes (5 to 50), number of hidden layers (1 to 3), activation function (identity, logistic, tanh, relu), learning rate (constant, adaptive, invscaling), L2 penalty (regularization term) parameter (0.00006 to 0.0003)
Lgmlvq	Prototypes per class (1 to 10), regularization (0 to 1)
rf	Criterion (gini, entropy), maximum Depth of Tree (1 to 2706), minimum samples per leaf (1 to 2706), minimum samples per split (2 to 2706), maximum features in each tree (6, sqrt, log), number of trees (500, 1000)
bagging	Number of base learners (50,100), maximum samples (0.5 to 0.8), maximum features (6, sqrt, log)
adaptive boosting	Number of base learners (50,100), learning rate (0.1 to 2.0)