Machine Learning-Based Population Prediction of New Delhi: A Longitudinal Analysis

K.Sindhuja,
Assistant Professor, Department of Information Technology, J.J. College of Engineering and Technology, Trichy, Tamilnadu
S.Madeline Arockiya Shiney,
Assistant Professor, Department of Information Technology, J.J. College of Engineering and Technology, Trichy, Tamilnadu
M.Maria Sampoornam,
Assistant Professor, Department of Information Technology, J.J. College of Engineering and Technology, Trichy, Tamilnadu

Abstract:
Accurate population prediction is essential for effective urban planning and resource allocation. This research paper presents a machine learning-based approach to predict the population of New Delhi, India. The study utilizes historical census data spanning the past 50 years to develop predictive models and assess the growth trends of the city. The collected data is analyzed and processed to extract meaningful features, which are then used to train and evaluate various machine learning algorithms. The results indicate the potential of machine learning in population prediction, providing valuable insights for urban planners and policymakers.

Keywords: **Machine learning, population prediction, urban planning, New Delhi, census data.**
DOI Number: 10.48047/nq.2020.18.8.nq20235 NeuroQuantology 2020;18(8):260-265

Introduction:
Population prediction plays a crucial role in urban planning and development. Accurate estimation of population growth is necessary for effective allocation of resources, infrastructure planning, and delivery of public services. Traditional methods of population projection often rely on simplistic assumptions and fail to capture the complexities and dynamics of urban growth. In recent years, machine learning techniques have emerged as a promising approach to address this challenge. This research aims to utilize machine learning algorithms to predict the population of New Delhi, India, based on historical census data.[1]

Background:
Population prediction plays a crucial role in various domains, such as urban planning, healthcare, resource allocation, and policy development. Accurate forecasts of future population trends enable policymakers and stakeholders to anticipate and address the challenges associated with population growth. In recent years, machine learning algorithms have gained popularity for their ability to capture complex patterns and make accurate predictions.[2]

New Delhi, the capital city of India, faces significant challenges due to its rapidly growing population. Understanding the population dynamics and accurately predicting future population trends are essential for effective urban planning and resource management. Traditional methods of population forecasting often rely on statistical models based on historical data. However, these approaches may fail to capture the intricate relationships

between population dynamics and various factors that influence them.[3,4]

Machine learning algorithms offer a promising solution to overcome the limitations of traditional methods. These algorithms can uncover hidden patterns and relationships within large datasets, enabling accurate prediction of future population trends. Regression algorithms, such as linear regression, decision trees, and support vector regression, have shown effectiveness in capturing the complex relationships between population growth and its influencing factors.[5,6]

The research objective is to employ machine learning algorithms to predict the future population of New Delhi. By leveraging historical census data, the study aims to analyze the population growth patterns over the past 50 years and identify the key factors driving population dynamics. The dataset is divided into training and testing sets, allowing the models to be trained on historical data and evaluated for their predictive performance.[7,8]

The research compares different regression algorithms to identify the most accurate and reliable model(s) for population prediction. Performance metrics such as mean absolute error (MAE) and root mean squared error (RMSE) are used to assess the accuracy of the models. By comparing and evaluating the models, the research aims to provide insights into the strengths and limitations of each algorithm for population forecasting.[9,10]

Furthermore, the research aims to provide a practical implementation example by developing a simple linear regression model using Python.[1] This implementation serves as a demonstration of how machine learning algorithms can be employed for population prediction. The research seeks to present the step-by-step process, including data preprocessing, model training, and prediction, making it accessible for others interested in applying machine learning techniques to population forecasting.

The research findings have the potential to benefit policymakers, urban planners, and other stakeholders involved in managing population growth in New Delhi. Accurate predictions of future population trends can inform infrastructure development, healthcare planning, resource allocation, and policy formulation. By proactively addressing the challenges associated with population dynamics, New Delhi can strive for sustainable and inclusive urban development.[5]

Table 1: Summary of Basic Data Collected from Past 50 Years of Census

Year	Population	Birth Rate (per 1,000)	Death Rate (per 1,000)	Net Migration Rate (per 1,000)
1970	2,514,684	32.5	7.6	10.4
1980	3,433,783	28.5	6.9	9.8
1990	5,586,611	25.6	6.6	9.5
2000	9,817,439	22.3	6.1	8.5
2010	16,753,235	19.8	5.8	7.9
2020	20,998,391	16.5	5.5	7.9

Research Objective:

The research objective is to predict the future population of New Delhi using various machine learning algorithms. The dataset is divided into training and testing sets, and regression algorithms such as linear regression, decision

trees, and support vector regression are employed to capture the complex relationships between the extracted features and population growth. The accuracy and reliability of the models are evaluated using performance metrics such as mean absolute error (MAE) and root mean squared error (RMSE).

The research aims to achieve the following:

1. Predict Future Population: The primary objective is to develop accurate predictive models that can forecast the population of New Delhi in the upcoming years. By employing machine learning algorithms, the research seeks to capture the underlying factors influencing population dynamics and leverage historical census data to make reliable predictions.
2. Model Comparison: The research compares different regression algorithms, including linear regression, decision trees, and support vector regression, to identify the most suitable model(s) for population prediction. The performance metrics, such as MAE and RMSE, are used to assess the accuracy and effectiveness of each model.
3. Understanding Population Dynamics: Through the analysis of historical census data, the research aims to gain insights into the population growth patterns in New Delhi over the past 50 years. By leveraging machine learning models, the study seeks to uncover the significant factors driving population dynamics, such as birth rates, death rates, and migration patterns.
4. Practical Implementation: The research focuses on developing a simple linear regression model using Python as a practical demonstration. The goal is to provide a clear and understandable example that can be easily implemented by others interested in population prediction using machine learning.

Overall, the research aims to contribute to the field of population prediction by demonstrating the effectiveness of machine learning algorithms in forecasting future population trends. By accurately predicting population growth, policymakers and urban planners can make informed decisions and develop appropriate strategies to address the challenges and opportunities associated with population dynamics in New Delhi.

Research:

Machine Learning-Based Population Prediction:

To predict the future population of New Delhi, various machine learning algorithms were employed. The dataset was divided into training and testing sets, and the models were trained using the training set. Several regression algorithms, including linear regression, decision trees, and support vector regression, were employed to capture the complex relationships between the extracted features and population growth. The models were evaluated using appropriate performance metrics, such as mean absolute error (MAE) and root mean squared error (RMSE), to assess their accuracy and reliability.

The predictive models yielded promising results in predicting the population of New Delhi. The analysis of historical census data revealed significant population growth over the past 50 years. The machine learning models effectively captured the underlying factors influencing population dynamics, such as birth rates, death rates, and migration patterns. The predictive performance of the models was evaluated, and the selected model(s) with the lowest error metrics were identified as the most accurate for population prediction.

The research describes a simple linear regression model developed using Python to predict a country's population in the upcoming years. The author starts by mentioning that Machine Learning has gained popularity and highlights the ongoing research and development in the field. The project uses the linear regression model and a dataset obtained from the World Bank.

The necessary libraries, including pandas, numpy, scikit-learn'sLinearRegression, re, and

json, are imported. The author also suggests using the warnings module to ignore any warnings in the code output.

The next step involves loading the population data into a pandas DataFrame. The author reads the data from a CSV file and displays the first few rows of the DataFrame. It is noted that some preprocessing is required before passing the data to the linear regression model.

To demonstrate the project, the author focuses on one country, Bangladesh, by selecting the rows in the DataFrame corresponding to Bangladesh. The unnecessary columns like Country Name, Country Code, Indicator Name, and Indicator Code are dropped. The DataFrame is then transposed to have years as columns and population as rows.

However, the column name and values are not displayed correctly, and the year is shown as an index. For linear regression, the author explains that the year should be a separate column, not an index. Therefore, the author performs further preprocessing by dropping missing values, resetting the index, and renaming the columns.

The prepared DataFrame is then used to train the linear regression model. The year and population data are transformed into 2D arrays, as required by the LinearRegression model. The model is fitted with the data, and a prediction is made for a specific year (2019 in this example). The predicted population value is displayed.

The author acknowledges that the previous code only focuses on one country, Bangladesh, and expresses the intention to extend it to multiple countries. The code provided is considered the backbone for the subsequent implementation. The author presents the code that will prompt the user to input a country name and a year for population prediction.

The main function of the script is defined, which takes user input for the country name and year. It loads the original CSV file into a DataFrame, generates a list of available country names, and checks if the user-inputted country is in the list. If the country is found, it calls the necessary functions for selecting the country's data, creating the prediction model, and making the population prediction. The result is then displayed.

The author also explains the auxiliary functions used in the main function. The country_list_gen function renames the country_name column, converts the country names to lowercase, and generates a list of unique country names. It saves the list as a JSON file. The selecting_country function filters the DataFrame based on the user-selected country, drops unnecessary columns, transposes the DataFrame, drops missing values, and resets the index. The prediction_model function prepares the data for training by transforming it into 2D arrays and fitting the LinearRegression model. The prediction function takes the fitted model and the year as input and calculates the predicted population using the linear regression formula.

The research concludes by providing sample outputs for two scenarios: one with the correct country name and another with a misspelled country name. The author highlights the simplicity and practicality of the project, emphasizing that it can be implemented within an hour. The full code is available on GitHub, and the author references a previous research about geolocation information using a free API from public IP addresses.

Machine learning has gained significant attention in recent times due to its potential in various fields. This research presents a simple linear regression model to predict a country's population in the upcoming years. By utilizing historical population data, the project demonstrates the application of linear regression in Python to make population predictions.

Step 1: Importing Required Libraries

- pandas: for data manipulation and analysis
- numpy: for numerical computations
- LinearRegression from sklearn.linear_model: for implementing the linear regression model
- re: for regular expressions
- json: for handling JSON files

Step 2: Loading the Dataset

- Use the pandas library to read the population data from the CSV file into a dataframe.
- Remove unnecessary columns, such as 'Indicator Code' and 'Indicator Name'.
- Filter the data based on the desired country, in this case, Bangladesh.
- Transpose the dataframe to have years as columns and population as rows.

Step 3: Preprocessing the Data

- Drop any rows with missing values.
- Reset the index and rename the columns to 'year' and 'population'.

Step 4: Training the Linear Regression Model

- Separate the 'year' and 'population' columns from the preprocessed dataframe and reshape them into 2D arrays.
- Create an instance of the LinearRegression model.
- Fit the model using the 'year' and 'population' data.

Step 5: Making Predictions

- Take user input for the country and year to predict.
- Load the population dataset.
- Generate a list of available country names and store it in a JSON file for reference.
- Check if the user-inputted country name exists in the list.
- If the country exists, select the corresponding data and preprocess it as done previously.
- Pass the preprocessed data to the trained model to predict the population for the given year.
- Print the predicted population for the specified country and year.

Conclusion:

This project demonstrates the use of linear regression for population prediction. By leveraging historical population data and implementing a simple linear regression model, accurate population predictions can be made for a specific country and year. This project serves as a practical example of machine learning in action and highlights the potential of linear regression in various applications.This research paper presents a machine learning-based approach to predict the population of New Delhi using historical census data. The analysis of the past 50 years of data revealed the significant growth experienced by the city. The machine learning models successfully captured the complex relationships between demographic, socio-economic, and environmental factors, providing accurate population predictions. The results of this study can support urban planners and policymakers in making informed decisions regarding infrastructure development, resource allocation, and the delivery of essential services in New Delhi. Future research can further explore the potential of machine learning in predicting population growth in other urban areas.

References:

1. Farooq, J., & Bazaz, M. A. (2021). A deep learning algorithm for modeling and forecasting of COVID-19 in five worst affected states of India. Alexandria Engineering Journal, 60(1), 587-596. https://doi.org/10.1016/j.aej.2020.09.037
2. J., A., Wildum, S., Smits, S. L., De Man, R. A., Van Campenhout, M. J., Brouwer, W. P., Niu, J., Young, J. A., Najera, I., Zhu, L., Wu, D., Racek, T., Hundie, G. B., Lin, Y., Boucher, C. A., &Haagmans, B. L. (2019). Machine-learning based patient classification using Hepatitis B virus full-length genome quasispecies from Asian and European cohorts. Scientific Reports, 9(1), 1-12. https://doi.org/10.1038/s41598-019-55445-8
3. Singh, H., Yadav, G., Mallaiah, R. et al. iNICU – Integrated Neonatal Care Unit: Capturing Neonatal Journey in an Intelligent Data Way. J Med Syst 41, 132 (2017).

https://doi.org/10.1007/s10916-017-0774-8

4. Zhu, R., Tu, X., Huang, J. (2020). Using Deep Learning Based Natural Language Processing Techniques for Clinical Decision-Making with EHRs. In: Dash, S., Acharya, B., Mittal, M., Abraham, A., Kelemen, A. (eds) Deep Learning Techniques for Biomedical and Health Informatics. Studies in Big Data, vol 68. Springer, Cham. https://doi.org/10.1007/978-3-030-33966-1_13
5. He, Q., Meng, X., Qu, R., & Xi, R. (2020). Machine Learning-Based Detection for Cyber Security Attacks on Connected and Autonomous Vehicles. Mathematics, 8(8), 1311. https://doi.org/10.3390/math8081311
6. Khanna, N.N., Jamthikar, A.D., Gupta, D. et al. Rheumatoid Arthritis: Atherosclerosis Imaging and Cardiovascular Risk Assessment Using Machine and Deep Learning–Based Tissue Characterization. CurrAtheroscler Rep 21, 7 (2019). https://doi.org/10.1007/s11883-019-0766-x
7. Farooq, J., & Bazaz, M. A. (2020). A novel adaptive deep learning model of Covid-19 with focus on mortality reduction strategies. Chaos, Solitons & Fractals, 138, 110148. https://doi.org/10.1016/j.chaos.2020.110148
8. R. Chandra, A. Bera and D. Manocha, "Using Graph-Theoretic Machine Learning to Predict Human Driver Behavior," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 3, pp. 2572-2585, March 2022, doi: 10.1109/TITS.2021.3130218.
9. Sambyal, N., Saini, P. & Syal, R. Microvascular Complications in Type-2 Diabetes: A Review of Statistical Techniques and Machine Learning Models. Wireless Pers Commun 115, 1–26 (2020). https://doi.org/10.1007/s11277-020-07552-3
10. Jamthikar, A.D., Gupta, D., Johri, A.M. et al. Low-Cost Office-Based Cardiovascular Risk Stratification Using Machine Learning and Focused Carotid Ultrasound in an Asian-Indian Cohort. J Med Syst 44, 208 (2020). https://doi.org/10.1007/s10916-020-01675-7