Predicting Life Expectancy with Regression Analysis

Seah Kit Han, Jovyn Tan, Amanda Lim

Abstract

We predict life expectancy from widely available national metrics such as income, schooling and child mortality. We evaluate the suitability of 5 different machine learning models. Using Random Forest, a root mean square error of only 1.92 years is achieved.

1 Introduction

1.1 Motivations and Applications

Life expectancy refers to the average number of years a person is expected to live. It is a key metric in assessing a country's population health as well as the overall standard of living. Together with the Education Index and Gross National Income per capita, it forms the Human Development Index used by the United Nations Development Programme [1].

A better understanding and assessment of life expectancy helps countries improve this key metric, by diverting national spending to areas that more effectively contribute to the overall health of citizens. Furthermore, these insights could help develop better estimators of life expectancy and by extension, indicators of population health [2].

2 Research Methodology

2.1 Overview

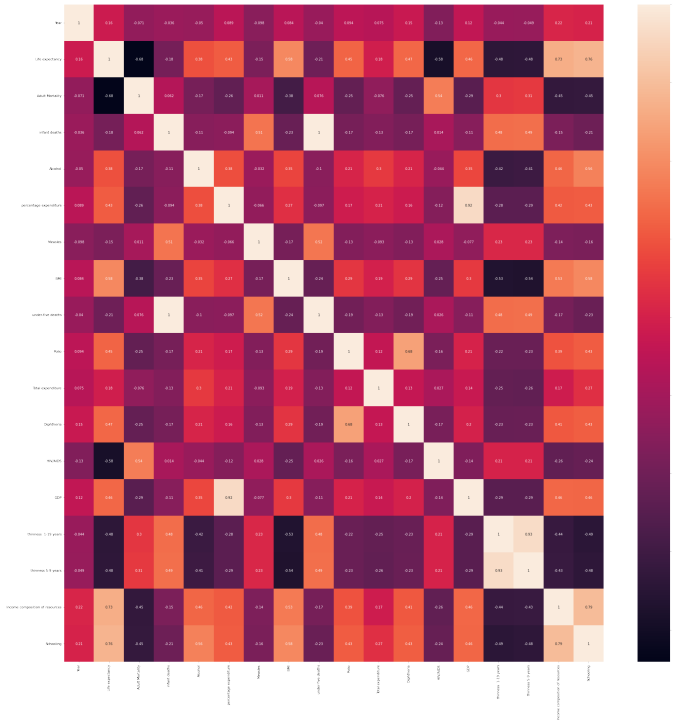
We run 5 models: Linear Regression, Ridge Regression, Support Vector Regressor, Random Forest and Neural Networks, before comparing and evaluating their effectiveness in predicting life expectancy. Additionally, we further optimise these models by investigating tuning parameters within each model.

2.2 Data Pre-Processing

The dataset was obtained from Kaggle, which was collected from WHO and the United Nations website, with the help of Deeksha Russell and Duan Wang [3].

We first remove rows and columns with significant NaN values. After cleaning, we split the data into numerical and categorical data, and encode the categorical data with labels. We then identify the features that correlate strongly to life expectancy, the variable we are trying to predict. Figure 1 is a correlation map of all the variables.

Figure 1: Correlation map



The top five variables that correlated most strongly to life expectancy were *Adult Mortality*, *HIV/AIDS*, *Income*, *Resources* and *Schooling* (see Table 1). We also observe that many features show correlation with each other, such as *Thinness* and *BMI*.

Table 1: Top correlators with life expectancy

Feature	Correlation
Life expectancy	1.000
Schooling	0.762
Income composition	0.732
Adult Mortality	-0.682
BMI	0.580
HIV/AIDS	-0.576

Finally, we scale the variables and train our models using this streamlined dataset. Our data has a 80-20 split for training and testing respectively, as our dataset is small and this split allows for sufficient testing data for reliability while maintaining enough data for training the model itself.

2.3 Multi-Linear Regression

We used the `LinearRegression` method of `sklearn` to perform an initial regression analysis on our data. This method uses the Ordinary Least Squares (OLS) method as its optimization function by minimizing the sum of square differences between the observed and predicted values. We chose to start with OLS as it is relatively straightforward and is a good fit for our data - using multiple features to predict life expectancy.

2.4 Ridge Regression

L2 ridge regression adds a penalty term to the linear regression described above. This results in higher bias but lower variance, which prevents overfitting. This could be suitable in our dataset which has features that are linearly correlated each other (Fig. 1).

We used additional values of [0.0001, 0.001, 0.01, 0.1, 1, 10] as the hyper-parameter alphas in order to allow more fine-grained tuning of our model. These values represent the regularization strength. Regularization improves the conditioning of the problem and reduces the variance of the estimates. We also used cross-validation through the `RidgeCV` method of `sklearn` to try out different alpha values and choose the best method.

2.5 Support Vector Regressor

We chose to use `sklearn`'s `SVR` method for our third model as it finds the best hyper-plane within a threshold value that fits our data. This is different from the previous 2 models which aim to minimize the sum of squares.

In addition to a linear kernel, a Radial-Basis Function (RBF) kernel was also investigated to determine if a linear or non-linear kernel would be more suitable for the classifier. This is the most generalized form of kernelization and most similar to the Gaussian distribution [4]. Although a linear kernel may be simpler and require less computational load, the RBF kernel may give a more accurate prediction.

2.6 Random Forest

With regression, the mean or average prediction of the individual trees is returned. Random decision forests further correct for decision trees' habit of overfitting the training data. Thus, it generally outperforms decision trees, but their accuracy is lower than gradient boosted trees [5].

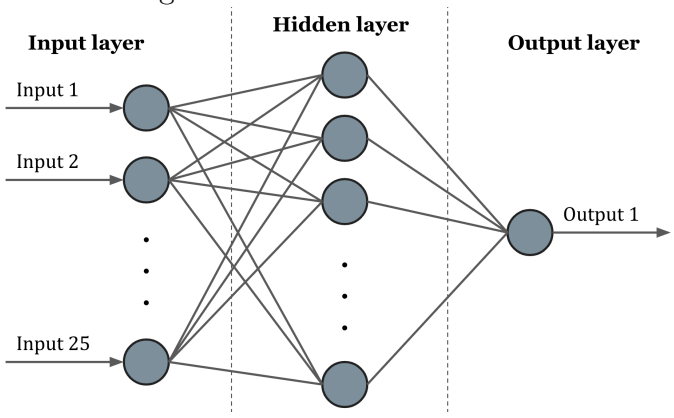
Random forests are also good for non-linear relationships, which is different from the linear models we have run thus far, and could further improve results.

Additionally, we investigated how the number of trees used in the random forest classifier affects the RMSE. The optimal number of trees was determined using the average of the root mean squared error over training 20 classifiers for each sampled number of trees.

2.7 Neural Network

Since the number of features we had was small (5), through manual tuning of the hyper-parameters, we found that a Neural Network of 25 neurons in the input layer, and 1 hidden layer of 15 neurons work best. This network was then trained for 100 iterations.

Figure 2: Neural network structure



The Adam optimiser was used, which is generally the best optimiser for Machine Learning models [6]. The

loss function uses mean squared error. We included only 1 hidden layer to reduce unnecessary complexity and reduce computational load in training. We chose RELU as the activation function for the output and hidden layers. One major benefit of RELU over sigmoid functions is the reduced likelihood of the gradient to vanish when $a > 0$ [7]. Additionally, when $a \leq 0$, sigmoids are likely to generate non-zero values resulting in dense representations.

3 Results & Discussion

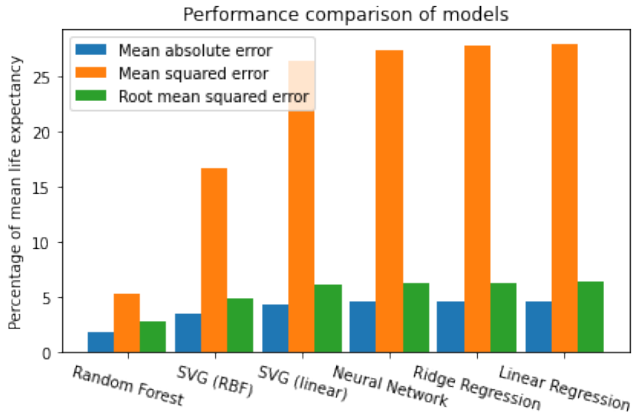
3.1 Overview

Root Mean Square Error (RMSE) is used as the metric to compare the 5 models. Random Forests performed the best, with a RMSE of 1.92, while Ridge Regression performed the worst, with a RMSE of 4.39. Table 2 shows an overview of the results obtained through the Mean Absolute Error (MAE), Mean Squared Error (MSE) and RMSE for each model. Figure 3 illustrates the results obtained as a percentage of the mean life expectancy.

Table 2: Overview of results

Model	MAE	MSE	RMSE
Linear Regression	3.162	19.31	4.394
Ridge Regression	3.159	19.27	4.390
SVR (Linear)	2.981	18.31	4.279
SVR (RBF)	2.398	11.52	3.395
Random Forest	1.271	3.702	1.924
Neural Network	3.155	19.01	4.360

Figure 3: Results



3.2 Multi-Linear Regression

Our multi-linear regression, using `sklearn`'s `LinearRegression()` method, showed promising results and suggests that a simple linear regression could be used to model life expectancy with respect to factors like schooling and adult mortality. However, non-linear models showed better performance as will be discussed below.

3.3 Ridge Regression

Surprisingly, adding the ridge penalty to the Multi-linear model increased the RMSE compared to the linear regression model. This is possibly due to a sub-optimal value chosen for the regularization parameter. Another reason could be that the number of rows of data (2301) is significantly larger than the number of features (19). As such, the benefit of using Ridge Regression over the Multi-linear model is negligible.

3.4 Support Vector Regressor

The SVR with a RBF kernel had the second-lowest RMSE out of all the classifiers. Thus, the SVR model performed better than both linear models by using a non-linear kernel. This indicates that the data could possibly have a nonlinear relationship, and using a linear-model would not be appropriate.

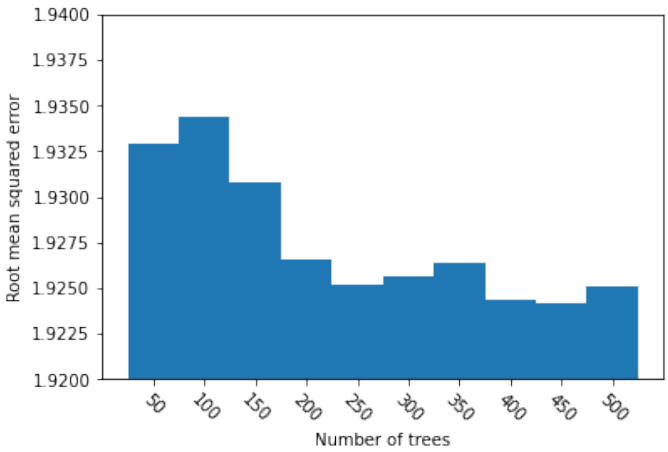
3.5 Random Forest

The random forest classifier had the lowest RMSE out of the classifiers that we attempted. This could be because random forest classifiers can implicitly handle collinearity in features, making it better suited for this particular datasets where several features may exhibit collinearity. For instance, features such as *Adult Mortality* and *HIV/AIDS* may be correlated, as well as *Income* and *Schooling*. This is because random forest uses bootstrap sampling and feature sampling such as row sampling and column sampling to pick different features for different models, reducing the effect of multicollinearity.

Furthermore, we found a positive relationship be-

tween performance and the number of trees. In general, as the number of trees increased, the RMSE decreased. Figure 4 shows the general decreasing trend between the number of trees used and RMSE.

Figure 4: RMSE against number of trees



The number of trees that gave rise to the lowest RMSE is 450. With 450 trees, we the average RMSE across training 20 such classifiers was 2.78% of the mean life expectancy.

However, the variance in RMSE across classifiers trained with the same number of trees was high, and the marginal improvement gained from increasing the number of trees is low. Increasing the number of trees from 100 to 450 yields only a 0.01 improvement in RMSE, which may not be a worthy trade-off given the increased computational load and time with a greater number of trees.

3.6 Neural Network

Using the values from *Adult Mortality*, *HIV/AIDS*, *Income*, *Resources* and *Schooling* as input features for the neural network, we then derive a prediction value. Through manual tuning, we found that one hidden layer gave the best results. The RMSE of the network is 4.36, which was 6.29% of the mean life expectancy.

Surprisingly, the added complexity of a neural network offered no improvement over a linear SVG, and only a minor improvement over a simple linear regression. This could be due to neural networks having a greater tendency to overfit compared to simpler regression models. Our relatively small dataset may also have been unsuitable for neural networks, which generally require more data than traditional machine learning algorithms.

3.7 Limitations

During data cleaning, we chose to drop columns that have a high percentage of NaN values. This resulted in a sizeable 22% loss of information, which could have affected our results. An alternative approach could have been to fill the missing data with the mean or mode of the remaining values in the column.

Additionally, the regularization parameters of [0.0001, 0.001, 0.01, 0.1, 1, 10] that were used for Ridge regression may not have been optimal. Another way would have been to test more parameter values and compare their accuracy, before selecting the best value. This could have further improved our results at the cost of greater computational load.

4 Conclusion

In conclusion, the random forest classifier performed the best among the classifiers used. This could suggest significant multicollinearity in the dataset, which is not unexpected due to the similar nature of features such as adult mortality and child mortality. The random forest classifier also had a RMSE of only 2.78% of the mean, suggesting that it could be useful in accurately predicting life expectancy.

Since the two best classifiers (Random Forest and SVR with RBF kernel) were non-linear, it could suggest that there is a non-linear relationship between the features used and life expectancy. Future work could include a deeper exploration into non-linear models which could yield even better results.

This research outcome has significant applications. Besides improving the accuracy of national life expectancy predictions given known factors like national schooling and income levels, policymakers may also be able to make more informed decisions in budget planning should they want to choose specific areas to focus legislation in.

5 References

[1] Murillo, P. I. L. (2021, July 14). *The life expectancy: What is it and why does it matter.* CENIE. Retrieved November 22, 2022, from https://cenie.eu/en/blogs/age-society/life-expectancy-what-it-and-why-does-it-matter

[2] Freeman, T., Gesesew, H. A., Bambra, C., Giugliani, E. R. J., Popay, J., Sanders, D., Macinko, J., Musolino, C., & Baum, F. (2020, November 10). *Why do some countries do better or worse in life expectancy relative to income? an analysis of Brazil, Ethiopia, and the United States of America - International Journal for equity in health.* BioMed Central. Retrieved November 22, 2022, from https://equityhealthj.biomedcentral.com/articles/10.1186/s12939-020-01315-z

[3] Mattson. (2020, October 6). *WHO national life expectancy.* Kaggle. Retrieved November 22, 2022, from https://www.kaggle.com/datasets/mmattson/who-national-life-expectancy

[4] Sreenivasa, S. (2020, October 12). *Radial basis function (RBF) kernel: The go-to kernel.* Towards Data Science. Retrieved November 22, 2022, from https://towardsdatascience.com/radial-basis-function-rbf-kernel-the-go-to-kernel-acf0d22c798a

[5] Sharma, A. (2022, June 15). *Decision Tree vs. Random Forest - which algorithm should you use?* Analytics Vidhya. Retrieved November 22, 2022, from https://www.analyticsvidhya.com/blog/2020/05/decision-tree-vs-random-forest-algorithm/

[6] Ruder, S. (2017, June 15). *An overview of gradient descent optimization algorithms.* arXiv.org. Retrieved November 22, 2022, from https://arxiv.org/abs/1609.04747v2

[7] Thakur, A. (2020, August 19). *Relu vs. sigmoid function in deep neural networks.* W&B. Retrieved November 22, 2022, from https://wandb.ai/ayush-thakur/dl-question-bank/reports/ReLU-vs-Sigmoid-Function-in-Deep-Neural-Networks–VmlldzoyMDk0MzI