Predicting Breast Cancer Prognosis

Roberto Cárdenas[a]

[a] *Colegio de Ciencias e Ingenierías "El Politécnico", Universidad San Francisco de Quito USFQ, Quito 170157, Ecuador.*

Abstract

This work explores the use of different regression methods for cancer prognosis. In particular, the studied data-set aims for prediction of survival and recurrence years of breast cancer. Out of eight measured regression models: Polynomial, Gaussian, Decision Tree, Random Forest, K Nearest Neighbours, Support Vector Machines, Neural Network, and LASSO, the LASSO regression method showed best results in comparison with lower measures of Mean Squared Error, Root Mean Squared Error and Absolute Squared Error across all the other implemented models.

Keywords: Forecasting, Regression.

1. Introduction

Nowadays, Regression methods are used to analyze the relationship between a dependent variable and one or more independent variables. The purpose of regression analysis is to determine if there is a relationship between the dependent variable and one or more independent variables. (Beers, 2022) It is also useful for modeling the future relationship between them.

Some of the problems that regression models have solved are: forecasting sales, stock predictions, predicting consumer behavior, analyzing survey data for customer satisfaction, product preferences, etc.

This work aims to explore eight regression methods to maximize the prediction of time survival and cancer recurrence on a breast cancer data set.

2. Materials and Methods

2.1. MammaPrint data set

We use the MammaPrint breast cancer (BC) dataset, which contains 70 features and 2 target columns which are Time Survival Years and Time Recurrence Years, as their names suggest, these columns represent the time in years of a particular patient of either recurrence of cancer cells or survival with the disease. (Aguilera-Mendoza et al., 2015)

In particular, the features of the dataset consist of a 70-gene expression profile that was initially created to identify individuals with early-stage BC who are unlikely to experience metastases and are eligible for remission following adjuvant chemotherapy. Its usage as a prognostic biomarker has received extensive retrospective and prospective validation. (Brandão-M et al., 2019)

2.2. Regression methods

Regression models are widely used nowadays as a powerful tool for uncovering the associations between variables observed in data, but cannot easily indicate causation. Regression in synthesis is a statistical method used in all areas of finance, investing, and other disciplines where it's objective consists in finding the strength and character of the relationship between one dependent variable commonly denoted by Y and a series of other variables also known as independent variables. (Beers, 2022)

2.2.1. Polynomial

Polynomial regression is a type of linear regression that describes the relationship between dependant variables and independent variable modelled as an nth degree polynomial in x. It is also called an special case of Multiple Linear Regression in Machine Learning because some polynomial terms are added to the Multiple Linear regression equation to convert it into Polynomial Regression. (Raghav, 2021) For this regression, it is important to preprocess the input variables into polynomial terms using a selected degree. The equation of a polynomial expression goes by the form of:

$$y = a_0 + a_1 x_1 + a_2 x_1^2 + ... + a_n x1^n$$

2.2.2. Gaussian

The Gaussian processes model is a probabilistic supervised machine learning framework that has been widely used for regression and classification tasks. A Gaussian processes regression (GPR) model can make predictions incorporating prior knowledge (kernels) and provide uncertainty measures over predictions (Sit, 2019)

In GPR, it is first assumed a Gaussian process prior, which can be specified using a mean function, m(x), and covariance function, k(x, x'):

$$f(x)\ GP(m(x), k(x, x'))$$

A Gaussian process is hence comparable to an infinite-dimensional multivariate Gaussian distribution in which any grouping of the dataset's labels is jointly distributed.

2.2.3. Decision Tree

The Decision Tree algorithm, as it name suggests, uses a tree-like model of decisions as a predictive model to draw conclusions about a set of observations. Decision trees where the target variable can take continuous values (typically real numbers) are called regression trees. More generally, the concept of regression tree can be extended to any kind of object equipped with pairwise dissimilarities such as categorical sequences. (Prasad, 2021)

It is important to keep in mind that the algorithm is susceptible to overfitting. Therefore, it is preferable to cross-validate and to always define the minimum number of children per leaf node in advance.

2.2.4. Random Forest

Random Forest can be used as a regression method that operates by constructing a group of decision trees at training. The mean prediction of the

individual trees is returned. This model is used to solve the overfitting problem with decision trees.

The low correlation between models is the key. Uncorrelated models have the ability to generate ensemble forecasts that are more precise than any single prediction. As long as they don't consistently all err in the same direction, the trees shield each other from their individual errors, which accounts for this lovely result. Many trees will be right while some may be wrong, allowing the group of trees to move in the proper direction. (Yiu, 2019)

2.2.5. k Nearest Neighbours

The k-nearest neighbors algorithm, sometimes referred to as KNN or k-NN, is a supervised learning classifier that employs proximity to produce classifications or predictions about the grouping of a single data point. Although it can be applied to classification or regression issues, it is commonly employed as a classification algorithm because it relies on the idea that comparable points can be discovered close to one another. (Harrison, 2018)

For regression, the average of the k nearest neighbors is used to forecast a classification. Here, the primary difference is that classification is used for discrete data whereas regression is utilized for continuous values. However, defining the distance is necessary before a categorization can be determined. The most typical measurement is euclidean distance described as:

$$d(x, y) = \sqrt{\sum_{i=1}^{n} (yi - xi)^2}$$

2.2.6. Support Vector Machines

The machine learning algorithm known as a support vector machine (SVM) examines data for regression and classification purposes. This algorithm sorts data into one of two groups after looking at it. The sorted data are output as a map by an SVM, with the margins between the two being as far away as possible. SVMs are employed in the sciences, picture classification, handwriting

recognition, and text categorization. SVM regression is considered a non parametric technique because it relies on kernel functions. (Ghandi, 2018) The goal is to find a function f(x) that deviates from yn by a value no greater than ϵ for each training point x, and at the same time is as flat as possible.

Linear SVM Regression Primal Formula:

$$f(x) = x'\beta + b$$

2.2.7. Neural Network

An Artificial Neural Networks is a model that simulate the human brain by implementing neurons. It consists of Input layer, Hidden layers, Output layer. The hidden layer can be more than one in number. Each layer consists of n number of neurons. Each layer will be having an Activation Function associated with each of the neurons. The activation function is the function that is responsible for introducing non-linearity in the relationship (Srivignesh, 2021). Each layer can also have regularizers associated with it. Regularizers are responsible for preventing overfitting.

Training data is essential for neural networks to develop and enhance their accuracy over time. However, these learning algorithms become effective tools in computer science and artificial intelligence once they are adjusted for accuracy, enabling us to quickly and effectively analyze, classify, or create data.

2.2.8. LASSO

LASSO regression is another type of linear regression. LASSO stands for Least Absolute Shrinkage and Selection Operator. It consists in the use of shrinkage that is defined as where data values are shrunk towards a central point, similar to the mean. The lasso procedure emphasizes simple and sparse models with fewer parameters, in order to show at high fidelity models with high levels of muticollinearity. It is also particularly useful for automation of certain parts of model selection or elimination like parameters or variables. (Glen, 2022a)

Regression with L1 Regularization: If a regression model uses the L1 Regularization technique, then it is called Lasso Regression. If it used the L2 regularization technique, it's called Ridge Regression.

The L1 regularization process used by Lasso regression results in a penalty proportional to the absolute magnitude of the coefficients. A sparse model with few coefficients may be produced by this kind of regularization; certain coefficients may go to zero and be removed from the model. Greater penalties provide coefficient values that are closer to zero, which is great for creating more straightforward models. However, L2 regularization (such as Ridge regression) does not eliminate coefficients or sparse models. Because of this, Lasso is much simpler to understand than the Ridge. (Glen, 2022a)

Following is the formula of how the regression is performed:

$$\sum_{i=1}^{n}(yi - \sum_{j} xij\beta j)^2 + \lambda \sum_{j=1}^{p} |\beta j|$$

The lambda tuning parameter controls the strength of L1 penalty. Lambda is in synthesis the amount of shrinkage:

If lambda = 0, no parameters are eliminated. The estimate is equal to the one found with linear regression. As lambda increases, coefficients are eliminated. As lambda increases, bias increases and finally, as lambda decreases there is an increase in variance.

2.3. Experimental methodology

This section outlines the experimental evaluation of the model used for this project. The 8 models presented above were implemented using python programming techniques and the use of specific scientific libraries adjusted to every model independently from the others. The following preparations and configurations were made:

2.3.1. Data set normalization

The min-max method was used to normalize the data. In this way, inconsistencies with the model are avoided and a better training can be done.

The initial data is transformed linearly using min-max normalization, also known as feature scaling. Using this method, all data is scaled within the range from (0, 1). The following is the formula to accomplish this:

$$x_{scaled} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

(O'Reilly, 2023)

2.3.2. Training, validation, and test sets

The dataset was separated by 10 percent for the test set (which does not participate in the training of the models). This is used to feed the best model selected by each method. A repeated 10-fold cross validation was applied to the remaining 90% of the data that make up the training set, and as a result of this process, the train or validation folds were formed.

2.3.3. Evaluation metrics

The evaluation metrics used for this projects were: Mean Squared Error, Mean Absolute Error, and Root Mean Squared Error. This metrics were measured across 10-fold cross validation scheme, so the mean of all 10 folds was represented in the results. This process gives much more information about the algorithm performance, and draw better informed conclusions. (Shulga, 2018)

Mean Squared Error: It is possible to determine how closely a regression line resembles a set of points using the mean squared error (MSE). This is accomplished by squaring the distances between the points and the regression line (also known as the "errors"). The squaring is required to eliminate any unfavorable indications. Additionally, it emphasizes bigger discrepancies. Since it is averaging a collection of errors, this error type is known as the mean squared error. The forecast is more accurate the lower the MSE. (Glen, 2022b)

Root Mean Squared Error: Root Mean Square Error (RMSE) is the standard deviation of the residuals (prediction errors). The distance between the data points and the regression line is measured by residuals, and the spread of these residuals is measured by RMSE. In other words, it provides information on how tightly the data is clustered around the line of best fit. In climatology, forecasting, and regression analysis, root mean square error is frequently used to validate experimental results. (Glen, 2022c)

Mean Absolute Error: Mean Absolute Error is a model evaluation metric used with regression models. The mean absolute error of a model with respect to a test set is the mean of the absolute values of the individual prediction errors on over all instances in the test set. Each prediction error is the difference between the true value and the predicted value for the instance. (Sammut, 2011)

3. Results and Discussion

3.0.1. Performance in the training

Below are the tables of results obtained by each of the regression methods used in this project. The average MSE, MAE, and RMSE measurements through cross validation and the standard deviation of each result are represented here.

Time Survival Years						
Model	MSE	MSE std	RMSE	RMSE std	MAE	MAE std
Polynomial	0.09	0.03	0.30	0.05	0.24	0.04
Gaussian	0.05	0.02	0.23	0.04	0.19	0.03
Decision Tree	0.08	0.02	0.28	0.04	0.23	0.03
Random Forest	0.05	0.01	0.22	0.03	0.18	0.03
KNN	0.05	0.01	0.22	0.03	0.18	0.03
SVM	0.06	0.02	0.24	0.03	0.19	0.03
Neural Network	0.06	0.02	0.23	0.04	0.19	0.03
LASSO*	0.05	0.01	0.21	0.03	0.17	0.03

Time Recurrence Years						
Model	MSE	MSE std	RMSE	RMSE std	MAE	MAE std
Polynomial	0.10	0.03	0.32	0.04	0.25	0.03
Gaussian	0.06	0.02	0.24	0.04	0.19	0.03
Decision Tree	0.10	0.02	0.31	0.03	0.24	0.03
Random Forest	0.06	0.01	0.23	0.03	0.19	0.02
KNN	0.05	0.01	0.23	0.03	0.19	0.03
SVM	0.07	0.02	0.25	0.03	0.20	0.02
Neural Network	0.07	0.02	0.25	0.04	0.20	0.03
LASSO*	0.05	0.02	0.23	0.03	0.19	0.03

Lasso model, as shown in the table above, has the best performance on the studied dataset. It is possible that the model is showing high levels of muti-collinearity as in when there are high correlations between two or more predictor variables. In other words, one predictor variable can be used to predict the other. This creates redundant information, skewing the results in a regression model. For this reason, the shrinkage property of the LASSO model works well minimizing the error metrics given the 70 features of the dataset.

3.0.2. Performance in the test set

Here we present the results obtained by the best selected model, the one that minimizes the desired metric. In our case it has been the LASSO model. The table below shows the performance of the model on the 10% of the data excluded from the training dataset with their respective MSE and MAE evaluations.

LASSO testing performance		
Target	Mean Squared Error	Mean Absolute Error
Time Survival Years	0.0467	0.1776
Time Recurrence Years	0.0562	0.1945

It is clear that the model performance metrics in the training phase with 90% of the dataset is consistent with the results obtained in the testing phase with 10% of the dataset. The result for Mean Absolute Error is sill not ideal and other model configurations should be taken into account in order to lower even further these results considering the sensibility of these type of predictions.

4. Conclusions

In this project, 8 different regression algorithms were implemented and tested for breast cancer survival and recurrence years prognosis. Different techniques were used in each algorithm being measured by the same metrics for comparison. Then, conclusions were drawn by which algorithm has been observed to yield the best results across both metrics and target values.

Accordingly, it was seen in our results that prognosis of breast cancer with MammaPrint technology seems to be better to work with LASSO regression due to its lower mean squared error, root mean squared error and mean absolute error across all other regression models. It is possible that the studied model was showing high levels of muticollinearity and for this reason the shrinkage property of LASSO worked well in minimizing the error metrics given the 70 features of the dataset.

References

Aguilera-Mendoza, L., Marrero-Ponce, Y., Tellez-Ibarra, R., Llorente-Quesada, M. T., Salgado, J., Barigye, S. J., & Liu, J. (2015). Overlap and diversity in antimicrobial peptide databases: compiling a non-redundant set of sequences. *Bioinformatics*, *31*, 2553–2559. doi:10.1093/bioinformatics/btv180.

Beers, B. (2022). What is regression? definition, calculation, and example. URL: https://www.investopedia.com/terms/r/regression.asp#.

Brandão-M, Pondé-N, & Piccart-Gehbart, M. (2019). Mammaprint™: a comprehensive review. *Future Oncology*, *15*, 207–224. URL: https://doi.org/10.2217/fon-2018-0221. doi:10.2217/fon-2018-0221.336arXiv:https://doi.org/10.2217/fon-2018-0221.

Ghandi, R. (2018). Support vector machine — introduction to machine learning algorithms. URL: https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47.

Glen, S. (2022a). Mse: Mean squared error. URL: https://www.statisticshowto.com/lasso-regression/.

Glen, S. (2022b). Mse: Mean squared error. URL: https://www.statisticshowto.com/probability-and-statistics/statistics-definitions/mean-squared-error/.

Glen, S. (2022c). Rmse: Root mean square error. URL: https://www.statisticshowto.com/probability-and-statistics/regression-analysis/rmse-root-mean-square-error/.

Harrison, O. (2018). Machine learning basics with the k-nearest neighbors algorithm. URL: https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761.

O'Reilly (2023). Min–max normalization. URL: https://www.oreilly.com/library/view/hands-on-machine-learning/9781788393485/fd5b8a44-e9d3-4c19-bebb-c2fa5a5ebfee.xhtml.

Prasad, A. (2021). Decision trees fro regression. URL: https://medium.com/analytics-vidhya/regression-trees-decision-tree-for-regression-machine-learning-e4d7525d8047.

Raghav, A. (2021). All you need to know about polynomial regression. URL: https://www.analyticsvidhya.com/blog/2021/07/all-you-need-to-know-about-polynomial-regression/.

Sammut, C. (2011). Mean absolute error. URL: https://link.springer.com/referenceworkentry/10.1007/978-0-387-30164-8_525.

Shulga, D. (2018). 5 reasons why you should use cross-validation in your data science projects. URL: https://towardsdatascience.com/5-reasons-why-you-should-use-cross-validation-in-your-data-science-project-8163311a1e79#.

Sit, H. (2019). Quick start to gaussian process regression. URL: https://towardsdatascience.com/quick-start-to-gaussian-process-regression-36d838810319.

Srivignesh, R. (2021). A walk-through of regression analysis using artificial neural networks in tensorflow. URL: https://www.analyticsvidhya.com/blog/2021/08/a-walk-through-of-regression-analysis-using-artificial-neural-networks-in-tensorflow/.

Yiu, T. (2019). Understanding random forest. URL: https://towardsdatascience.com/understanding-random-forest-58381e0602d2.