Spark: Predicting Patient Survival with Logistic Regression

Scalable Systems Programming: Project 100%

Polina Prinii
Postgraduate Diploma in Science in Data Analytics
School of Computing, National College of Ireland, Mayor Street Lower, IFSC, Dublin1, D01Y300, Ireland
Email: x21137757@student.ncirl.ie

Abstract—**Regression analysis is of no new practise in the medical domain. Since the advancement of artificial intelligence, regression analysis has been a popular method of machine learning application in various medical settings. Recent studies have examined multiple applications of regression analysis within the medical domain, presenting a wide range of related work for reference. One of particular interest is Logistic Regression. Previous studies have indicated that Logistic Regression is popular in many machine learning fields, with applications ranging from machine learning algorithms, to deep neural networks, to distributed systems such as Apache Spark. The research looks to identify the application of a Logistic Regression model against a set of medical data which outlines patient survival history based on common medical data. The following work was facilitated by Apache Spark, to allow for fast execution considering the size of the proposed data. Various open-source libraries supported the building and testing of the model. Overall the Logistic Regression model was successful in achieving an accuracy of 80%, giving way for additional future work. All technical aspects of the research were undertaken in Python.**

Keywords: **- logistic regression, machine learning, target, features, x, variables, y variable, dependent variable, independent variables, apache spark, big data, distributed systems, virtual machine, instances.**

I. INTRODUCTION

Medical practice has tremendously evolved since the first recordings of human history [1]. In the last quarter of the millennia alone medical practice has significantly spiked in evolution because of new discoveries and modernization. Major factors such as the discovery of the germ theory led to the cure of many infectious diseases, the rapid growth of urbanization emplaced systematic sanity measures, as well as the revolutionary discovery of antibiotics. The twenty-first century alone was seen as a hyper-accelerator in providing medical care at its best. Likewise medical practice was heavily professionalized with the introduction of field specialization, allowing for medical care to excel.

Medical care has seen a rise from simple remedies curated and administered from home to countries such as the United States being recognized as leading leaders in the provision of the utmost medical care to its patients. Nonetheless with medical care constantly advancing, patient survival remains a concerning risk for both the caring doctor and the hospital to which they fall for care.

Alongside the rapid advancement within the medical domain, came the advancement within computing. One of particular interest is the rising popularity of machine learning application. The use of machine learning application within the medical domain is of no new concept, machine learning is used in various settings to draw inferences from patterns in data. Applications of machine learning within the medical domain are endless, potentially of even not having a limit as the advancements within the industry continue.

The research looked to evaluate a certain aspect of machine learning – regression analysis, specifically the application of Logistic Regression. Logistic Regression a popular method for the calculation or prediction of an event occurring, which very often is evaluated by the dependent variable, also depicted as the *y* variable, is of binary nature.

In the research, it was identified the target variable is the binary measure of hospital deaths, with this a model was constructed which evaluated a medical set of data describing patient survival upon admittance to a hospital.

With the large popularity of machine learning application within the medical domain, the research looked to compare its findings against some of the other published works as well as address the following questions:

"To what extent can the survival rate be predicted using Logistics Regression?"

As the field of research is within the medical domain, the relative success of the Logistic Regression model cannot be solely evaluated on the accuracy metric also known as R^2. In such case the crucial metrics for evaluation are the measures of Specificity and Sensitivity. These measures will be the deciding factors behind the performance of one or all models undertaken throughout the research. However, the overall relative success of the Logistic Regression model is dependent on appropriate data cleansing, and tuning actions as well as adequate feature selection.

The research hopes to comprehensively evaluate the application of regression analysis to the chosen dataset, in aid to address the set-out question. Based on the drawn inferences, the research looks to draft recommendations for future work which may look to evaluate various other factors with the aim of improving the application of Logistic Regression for predictive analysis within the medical domain.

The following research is undertaken utilizing the open-source software Apache Spark, facilitating distributed system to handle big data, whist being built and execute using PySpark. All supporting materials of the researched have been uploaded to GitHub for the publics reference [2].

II. Related Work

Machine Learning regression analysis within the medical domain is of no novelty application, many have researched and published works exploring various techniques against different settings. An area of particular interest is the application of Logistic Regression for predictive analysis. Whilst many literatures available focused on specific medical condition for predictive analysis [3][4][5], a small number are considered by the research which have guided decisions throughout the study.

In the past several models have been developed to predict the onset of life-threatening illnesses such as heart diseases or acute liver failure. Anchana et al. [3] compared the performance of logistic regression against other popular methods such as decision trees and neural network for classification of heart disease patients. Achieving strong results in both True Positives at 81% and True Negatives at 73%. While the comparison of machine learning models is clear and concise in addressing the set-out objective, the study failed to clearly outline reproducible model parameters.

Tang et al. [4] made the comparison between a logistic regression model and other machine learning methods to predict acute kidney injury (AKI) in severely burned patients. Whilst the overall analysis was conducted outside of a Spark framework, Tang et al. proofed the application of s logistic regression analysis however, similarly to Anchana et al. failed to expand in detail of model parameter specification.

Wojciech et al. [5] proposed a new machine learning model based on logistic regression analysis with genetic algorithm optimization. By varying the regularization penalty between the three experiments, the study had achieved strong results with accuracy ranging between 80% to 95%, alongside Specificity and Sensitivity. Having clearly outlined each experimental model, Wojciech et al. grant reusability and replicability, facilitating for a comparable figure by the proposed research.

More specific to the proposed research, Vaggellis et al. [5] used random forest and logistic regression to predict early onset of Acute Respiratory Distress Syndrome. Whilst the medical field of study differs, the architecture outline is re-purposeful. The architectural platform established by Vaggellis et al. accommodates both static and streaming data-streams, with historic data staged within HDFS and streaming data handled by Apache Kafta & Zookeeper. Similarly, the proposed research looks to utilize HDFS from which a high-level API data frame would reach for the historic data for distributed parallel processing. Although, the research handles a significantly smaller dataset compared to the MIMIC III, certain aspects of the outlined data-streams can prove useful to implementing additional data-streams by the proposed research due to the clear data-stream setup outline, as Vaggelis et al. ingest data points from the ICU's, mechanical home ventilators and other smart health devices. By expanding the size of the dataset, the proposed research can potentially increase the overall relative success of the models as more data can be trained prior to validation. Granted, the study achieved high results potentially as a factor of parameter tunning. The proposed research hopes to achieve similar scores with the AUC, Accuracy, Specificity, and Sensitivity metrics, as such metrics have been widely recognized by previous studies as a measure of relative success.

Diego et al. [7] similarly to the proposed research applied a Logistic Regression model to predict in-hospital mortality with the focus on intensive care unit patients and their corresponding variables and descriptors. Similarly, to Vaggellis et al. [6], Resilient Distributed Datasets (RDD) were applied to structure the source data accordingly. Whilst the use of RDD's is appropriate for use when dealing with streams of text, a more appropriate method for working with. static, yet structured data is Spark DataFrames. Arguably the size difference of the data source between Diego et al. and the proposed research is immense, the use of Spark for big data analytics has become the de facto [8]. Diego et al. have suggested the calculation of p-values on the train set of data for further feature selection based on statistical significance of ($p < 0.001$). A notably important action to consider prior to regularization. Diego et al. applied default regularization parameters, removing a small percentage of weights at each iteration through the L2 penalty. Based on the results derived, using a Logistic Regression with LBFGS can yield greater results.

Mir Junaid et al. [9], have applied Logistic Regression analysis to a similarly sized dataset using Spark framework. A rather simple yet robust architecture is drafted and applied, to which the proposed research may closely align. A point to note, Mir Junaid et all did not address the class imbalance present within the ingested, ultimately resulting in a higher accuracy. A quite evident point when referring to the breakdown of True Positives and True Negatives. With 8431 True Positives predicted in comparison to 12 True Negatives. A considerable point to address as the proposed research explores the identified dataset. Considering the size of data to be ingested by the proposed research, the Spark architecture composed by Mir Junaid et al. may not suffice and thus may look to a more powerful setup within Spark.

Most studies within the field of predictive regression analysis focus on comparison research against other machine learning methods to evaluate the relative success of each constructed model. Such approaches, however, have failed to mention in detail aspects of the fine tuning of logistic regression models, leaving parameter setting such as regularization open to interpretation. Whilst a few studies have been found to describe in detail the penalization of logistic regression, it was found that in comparison to other machine learning model Logistic Regression could not outperform. Although, extensive research has been caried out on predicting patient survival using logistic regression, no such study for reference was found. A degree of difference would always be present, whether differentiating in method of application and analysis or with the medical topic chosen.

While most studies on logistic regression within the medical domain are undertaken using traditional machine learning architectures, as the data quantities grows more turn to the use of big data tools such as Apache Spark. Most research varied out in Spark accommodate for large quantities of data, providing insight into potential architectures as new studies arise.

III. METHODOLOGY

The methodology section outlines all technical aspects undertaken to extract, cleanse and process the data to draw inferences from the regression analysis undertaken. The research aligned to the Knowledge Discovery in Databases framework to execute the technical workflow, the KDD framework outlines a sophisticated data mining technique to identify, curate and evaluate patterns from a given data sample. However, the research takes into consideration that the illustrated framework by Figure 1., is not a fixed sequential process and that the various stages of the KDD can be revisited as appropriately needed.

Fig. 1. KDD Process Framework

To facilitate the implementation of the research a virtual instance was built, operating a Debian GNU/Linux 10 version alongside Python 3. Within the virtual instance the latest version of Apache Spark was installed to facilitate the data processing set out by the research.

A. The Dataset & Data Dictionary:

The research identified the Patient Survival Prediction Dataset [3] from Kaggle to aid in the Logistic Regression analysis set out. The original data was sourced from "The Global Open-Source Severity of Illness Score (Gossis)" [11], from where the compiled dataset on Kaggle came to be. The dataset consists of 91,713 rows and 186 columns, outlining patient information, vitals, chronic illnesses, and much more medical data, which ultimately determined the survival rate of a given patient. Both the dataset and data dictionary are of Comma Separated Value (.csv) format.

A mix of various data types are present within the dataset, such as integer, binary, numerical and string type. The attributes of the chosen dataset can be further examined within the data dictionary here:. Alternatively, the first 6 columns of the dataset are included in the report to aid in the understanding of the data worked.

TABLE I. DATA DICTIONARY

Category	**Patient Survival Prediction Dataset**			
	Variable name	Unit of Measure	***Data Type***	***Description***
Identifier	Encounter _id	None	Integer	Unique identifier associated with a patient unit stay
Identifier	Hospital _id	None	Integer	Unique identifier associated with a hospital
Identifier	Patient_id	None	Integer	Unique identifier associated with a patient
Outcome	Hoptial _death	None	Binary	Whether the patient died during this hospitalization
Demographic	Age	Years	Numeric	The age of the patient on unit admission
Demographic	bmi	Kilo / Metres	String	The body mass index of the person on unit admission

Fig. 2. Table 1: Data Dictionary – First 6 Rows

Due to the nature of the identified dataset, no data preparation activities were undertaken within the research prior to ingesting the file into Spark.

B. Architecture & Application Workflow:

An Apache Spark architecture is employed within the research due to the benefits of the computing framework for real-time processing. As the quantity of data increases the processing capabilities of traditional machine learning methods are strained, exposing a given research to larger wait time and excessive use of computation resources. Having near 92,000 records present within the dataset, the research found the use of Apache Spark appropriate due to its distributed and parallel processing functionality. This allows for faster and less consuming processing capabilities by partitioning the pieces of data between worker nodes.

A Spark Cluster was instantiated with a master / worker architecture employed, a total of 6 worker nodes each equipped with 4 core CPU's and 16GB of RAM were set and managed by one master node, in the case of the research the master node is a YARN – 'Yet Another Resource Negotiator' cluster manager. Figure 3 illustrates the set architecture at a high-level.

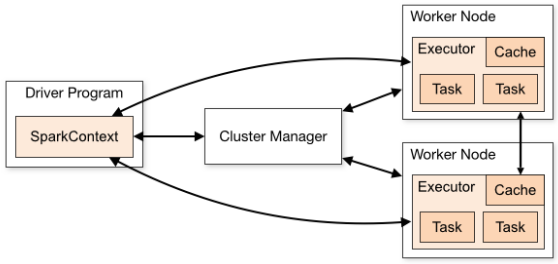

Fig. 3. Cluster Model Overview – Apache Spark

As highlighted previously, the research is primarily implement using the Python programming language, alongside some Linux commands to operate the virtual instance. Using the PySpark interface the following application workflow, illustrated by Figure 4 was built and executed to achieve the end results discussed within the Results & Evaluation section.

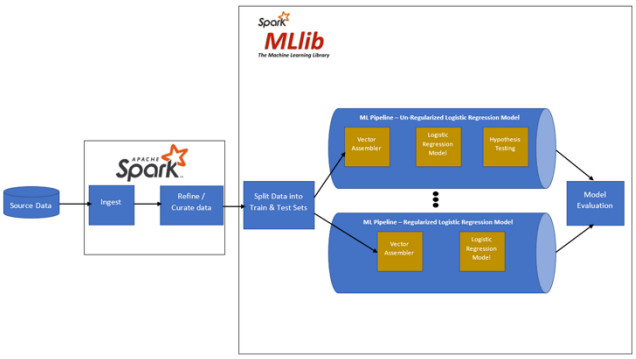

Fig. 4. Application Workflow Overview

C. Data Selection:

The relative success of a given model within regression analysis heavily depends on the data selection, as regression analysis evaluates the relationship between a target and feature variables [5]. As the data selection originates from Kaggle, the use of a Kaggle API was utilized. The API was

directly installed within the virtual instance. A connection was established following:

- A Kaggle.json file containing the connection key extracted from Kaggle.
- A new directory for Kaggle created.
- Permissions set for the .json file – chmod 600 set.

Following the above steps, the identified dataset is pulled from Kaggle using the '!kaggle datasets download' command, followed by the unzipping of the downloaded file. The success of the extraction was evaluated using the 'ls' Linux command and any poor naming formats addressed using the 'mv' command. The extracted files are then copied from the local folder to HDFS – Hadoop Distributed File System, from which Spark accesses via YARN.

Kaggle is an open-source platform for data scientist, thus no infringement was observed upon the extraction of the chosen dataset.

D. Exploratory Data Analysis (EDA):

Prior to addressing any issues present within the chosen data a small sample was selected for analysis. A total of 6% was selected, equating to 5,459 records for exploration. The data was grouped by target variable which identified the presence of a strong class imbalance present. Based on the sample selected almost 92% are false outcomes.

Alas PySpark does not support plotting functionality, to facilitate the drafting of various plots, the data frame set within the Spark Session was brought out. By converting the sample data frame from Spark to a Pandas, a fraction of the data to left the Spark Context and entered a local Python session in which plots can be constructed using libraries such as Seaborn. Whilst Figure 5 highlighted that more males than females do not survive given a set of presented circumstances.

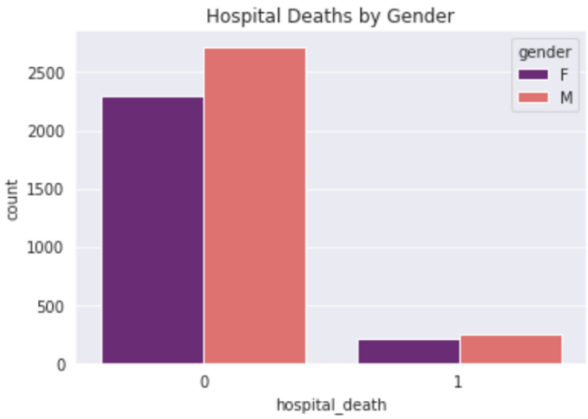

Fig. 5. Target – Hospital Death by Gender

Figure 6 confirmed that potentially said circumstances may originating for the emergency department as most to not survive originated from the emergency department.

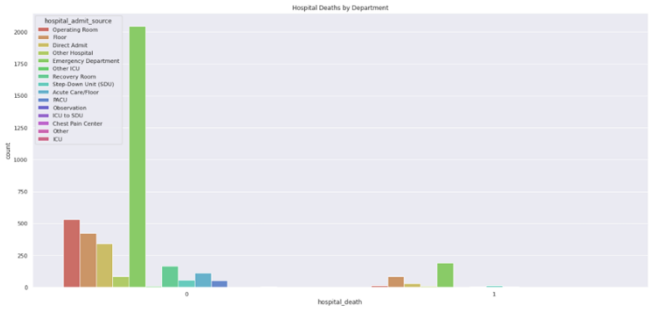

Fig. 6. Target – Hospital Death by Department Origin

E. Data Processing:

To ensure a strong model performance the data is cleansed prior to training and validation of the dataset. A variety of data processing actions were performed to address the abundance of null values as well missing values present within the dataset upon investigation, illustrated by Figure 7.

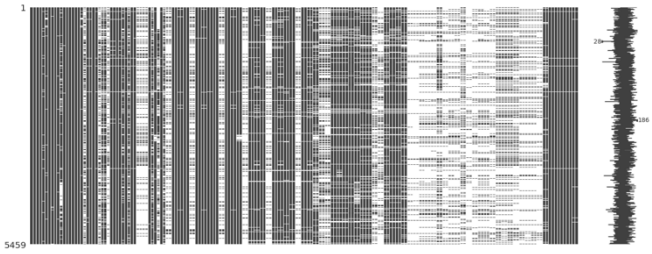

Fig. 7. Sample DataFrame: Matrix of Null or Missing Values

It was discovered that many of the 186 columns present within the dataset contained null or missing values. Typically, null, or missing values are dropped from the overall dataset, especially when working with large datasets. However, as many columns contained up to 90% of null or missing values, the research made the conscious decision to drop said columns. This was achieved by aggregating the overall data frame, to collect a percentage of null or missing values from each column as an array, from which the array was then converted to a dictionary for a loop to be applied to create a list of columns meeting a threshold of greater than 10% of null or missing values. Lastly, the list was then used to drop said columns, additionally other binary type columns were dropped leaving the overall dataset with 68 columns.

Having eliminated a large portion of null or mission values, a small portion remained. The remaining values were addressed with the use of two functions, one to collect the mean of a given column and the other to fill said null or missing values by the mean of the column in which they are present. Once all null or mission values were address, all string / categorical type columns present within the dataset were converted using a StringIndexer, having done so the original columns were dropped.

Prior to the setup of the Logistic Regression model the cleansed dataset was under-sampled based on the majority class to address the class imbalance present within the target variable. Resulting in a 60:40 ration between the False and True outcomes. Having addressed the class imbalance, the data frame was prepped further to facilitate the build of a binary classification model. Firstly, the target variable was encoded as 0 being False and 1 being True to prepare the data for a binary classification model. Secondly, the Spark data frame was converted into a sparse format as expected by

the Machine Learning algorithms within Spark, this was achieved using a VectorAssembler and only performed on the feature variables. Thirdly, the label also known as the target variable was converted into the required index format using a StringIndexer. Lastly, the data was split between train and test sets using stratified sampling to ensure an appropriate distribution of the True/False is present in both the train and test set. A total of 60% was selected from the negative class and a total of 85% from the positive class.

F. Logistic Regression Modelling

Throughout the Logistic Regression Modelling phase, two Logistic Regression models were fitted against the dataset. The research looked to compare the relative success having applied regularization to one of the models. Table 2 illustrates the parameter difference between the two models built.

TABLE II. LOGISTIC REGRESSION

Model Type	**Parameter Outline**		
	maxIter	***regParam***	***elasticNetParam***
Base	50	0.0	0.0
Regularized	100	0.5	0.3

Fig. 8. Table 2: Parameter Oveview – Logistic Regression

Additionally, the regularized model differed from the base with the number of features used. Both models followed a train and test process.

IV. RESULTS

In this section, the results obtained with the models proposed in the Methodology are presented. The following section aims the overall performance of the Spark Architecture and to evaluate a set of metrics regarded as key indicators to the relevant success of a Logistic Regression model, when working with a classification algorithm. The proposed research set out to understand to what extent can patient survival be predicted utilizing a Logistic Regression model. Whilst, one model would suffice to address the question, the research proposed an additional regularized model in hopes it yields superior results.

To measure the relative success of a model, the research looks to the overall Accuracy of both the base and regularized model. The Accuracy score indicates the number of correct predictions made by the model and is illustrated as a percentage however, as the research has undertaken a study within the medical domain using data acquired from hospital admission, the overall Accuracy is supported by the scores of both Sensitivity (True Positives Rate) and Specificity (True Negatives). Additionally, the research looks to the F1 score, a harmonic mean between precision and recall, the Log Loss score to measure the amount of divergence of the predicted probability with the actual label as well as the Area Under Curve (ROC) which indicates how well the models classified the positive and negative outcomes at all possible cutoffs. Table 3 illustrates a breakdown of metrics for both Logistic Regression models applied throughout the research.

TABLE III. LOGISTIC REGRESSION MODEL RESULTS

Model Type	**Metrics**			
	Accuracy:	***Log Loss:***	***F1:***	***AUC***
Base (Train)	0.784	0.460	0.783	N/A
Base (Test)	0.800	0.462	0.818	0.88
Regularized (Train)	0.784	0.460	0.783	N/A
Regularized (Test)	0.807	0.552	0.817	0.83

Fig. 9. Table 3: Model Results: Base & Regularized

It can be seen from Table 3 that both Logistic Regression models have performed quite well, yielding strong results between 78% to 81%. The research determined that the yielded results are relatively strong having referred to the study undertaken by D. R. Sarvamangala & Raghavendra V. Kulkarni [1]. Whilst the area of application differs, the algorithm at hand is of the same nature, thus a good gauge to the relative success of the implemented models. With both train instances of the models yielding exact result, both models are excluded from the overall evaluation and not considered in the final factor of the relative success. It is apparent from Table 3, that by introducing a regularization penalty into the, model a higher Accuracy can be achieve however small. Even so, with the regularization penalty came an increase within the Log Loss score, with the divergence of the predicted probability increasing by almost 10%, ultimately meaning that the prediction probability for some cases within the dataset reduced in comparison to the base model. In contrast to the overall Accuracy and Log Loss increasing with the regularized model, the base model outperformed the regularized model with the Area Under Curve being higher by 5%, with the curve illustrated by Figure 10.

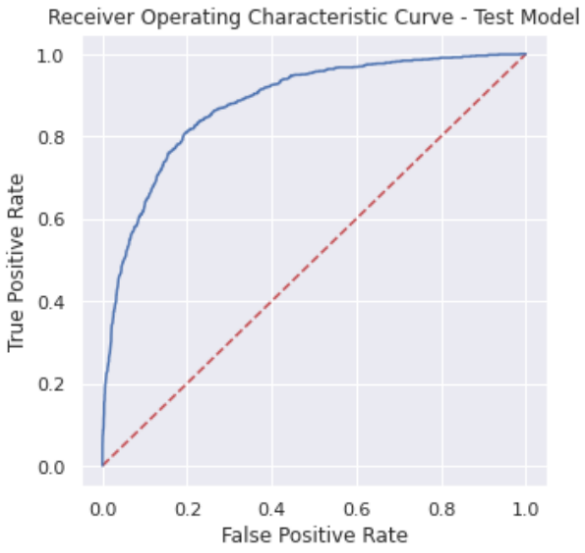

Fig. 10. areaUnderROC: Base Test Model

Whilst the difference in ROC between the two model is low, it is noted that the trade-off between sensitivity and specificity is better in the base model.

Having partially addressed the set-out question with the metrics derived from Table 3 alone, the derived Sensitivity and Specificity can determine the best model fit for the research at hand. With Sensitivity being the measure of True Positives (proportion of actual positive cases predicted as positive) and Specificity being the measure of True Negatives (proportion of actual negative cases predicted as negatives), a relatively high score for each metric is a good indicator of strong relative success of the models.

The base model achieved a score of 78% for Sensitivity and 81% for Specificity using the test set, a well distributed balance between the two as illustrated by Figure 11.

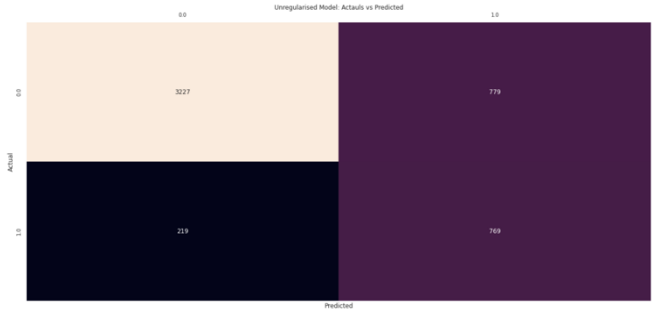

Fig. 11. Sensitivity vs Specificity: Base Model (TN = 0 & TP =1)

In contrast the Regularized model achieved a score of 69% for Sensitivity and 84% for Specificity, illustrating a decrease in Sensitivity of 9% and increase in Specificity of 3% compared to the base model. Figure 12 illustrates the distribution of prediction for the regularized model. Whilst the Accuracy improved with the implementation of a regularization penalty Sensitivity however, dropped. Accuracy alone is a poor indicator of the relative success of a given model thus, metrics such as the above are utilized to finalize the relative success. The drop in Sensitivity can potentially be explained by the applied regularization parameter itself, as the research had considered a default arbitrary value having reviewed official documentation. However, it is not to say that the Sensitivity scored achieved by the regularized model is poor, in fact it is a relatively good score.

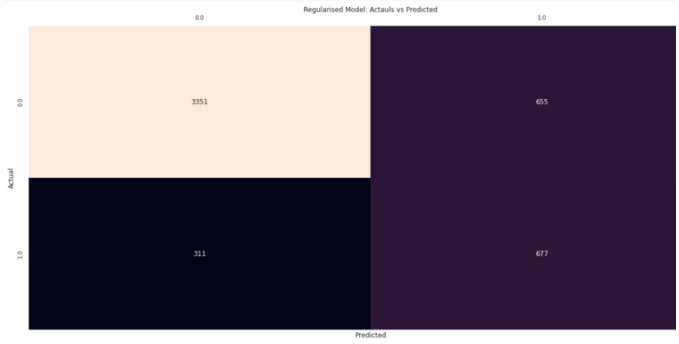

Fig. 12. Sensitivity vs Specificity: Regularized Model (TN = 0 & TP =1)

Together these results provide important insights into the parameter selection of the Logistic Regression models at hand when considering regularization, while simultaneously answering the question set out by the research. Considering the medical domain setting both the score of Sensitivity and Specificity take precedence. Taken together, these results suggest that both models have demonstrated good performance, suggesting that the patient survival rate can be predicted of upwards 80% with the predictions having a Sensitivity and Specificity of 78% and 81% respectively.

To conclude this section, the Spark run-time is evaluated. Overall, the proposed Spark architecture is promising, whilst the chosen dataset is of a small sample in comparison to the research reviewed, Spark managed to execute the cleansing, transforming, and the analysis in a total of 1047 seconds (17 minutes). Although this execution is considerably slow in comparison to related work, this potentially is a direct result of the small data sample worked. Spark is a great tool for working with large data, however a given job with a small data sample can disproportionately take longer to execute due to the excess overhead given by the cluster. In the case of the research, potentially 5 worker nodes were underutilized as the proposed data simple fit the accommodation of a single worker node. Nevertheless, the derived run-time is not entirely bad as the sub-tasks themselves executed within milliseconds. Taking this into consideration, the outlined architecture has great potential for scalability as the data under analyses increases.

V. Conclusion & Future Work

The proposed research set out to apply Logistic Regression analysis on a chosen dataset derived from a hospital setting to determine to what extend patient survival can be predicted. Utilizing Spark, a relatively large dataset was ingested, cleansed, analyzed and inferences drawn. Additionally, the research implemented a regularized model in hopes a greater result is yielded from the model. The research has shown that Logistic Regression can yield models with accuracy averaging upward of 80%, suggesting that the application of an appropriate regression analysis against a dataset can be just as effective if not better than other methods such as neural networks [13]. Whilst accuracy alone is not an accurate representation of the success of a given model, the relative success can be determined with Specificity and Sensitivity of a model. In the case of the conducted research, was successful in deriving strong scores for both.

Since the study was limited to a small sample size in comparison to referenced related work, it was difficult to acquire a good sample of records from the dataset for the regression models as the dataset was under sampled due to the presence of a class imbalance. However, despite the small sample, the findings of the Logistic Regression models suggest that a relatively high patient survival rate can be predicted. Should the research not have been limited in data sample size, it is possible even higher result could be achieved as larger data samples yield more data for training and better trained models yield to better validation of the test set.

A further study utilizing hyperparameter tuning is hoped to be implement as part of continuous improvement and development of the proposed models. Model selection is an important task and whilst feature selection was kept to the results of data cleansing, the proposed research hopes to implement tuning in future work. Tuning can help identify not only the best model but also the best parameters for a given task. Depending on the aims and objectives of future work a model can be tuned and best option selected using Cross-Validation, whilst Train Validation can assist with feature selection by evaluating a set of parameters once and

determining which is best. While this can be applied and implement on a relatively large dataset such as the work dataset through the research, a larger sample may yield greater results.

In conclusion, the research has determined that use of Logistic Regression can yield a positively strong rate for patient survival. Alongside, confirming that with the application of a regularization penalty overall score can increase. However, caution to be noted as other metrics can decrease and depending on the area of study may not entirely be a good case.

REFERENCES

[1] Koutsouris, D. The evolution of medical care: from the beginnings to personalized medicine. *Health Technol.* **7,** 3–4 (2017). https://doi.org/10.1007/s12553-016-0163-1

[2] Github, "SSP_Portfolio", [Online], Available at: https://github.com/polinaprinii/SSP_Portfolio

[3] A. Khemphila and V. Boonjing, "Comparing performances of logistic regression, decision trees, and neural networks for classifying heart disease patients," *2010 International Conference on Computer Information Systems and Industrial Management Applications (CISIM)*, 2010, pp. 193-198, doi: 10.1109/CISIM.2010.5643666.

[4] Tang CQ, Li JQ, Xu DY, et al. [Comparison of machine learning method and logistic regression model in prediction of acute kidney injury in severely burned patients]. Zhonghua Shao Shang za zhi = Zhonghua Shaoshang Zazhi = Chinese Journal of Burns. 2018 Jun;34(6):343-348. DOI: 10.3760/cma.j.issn.1009-2587.2018.06.006. PMID: 29961290.

[5] Wojciech Książek, Michał Gandor, Paweł Pławiak, Comparison of various approaches to combine logistic regression with genetic algorithms in survival prediction of hepatocellular carcinoma, Computers in Biology and Medicine, Volume 134, 2021, 104431, ISSN0010-4825, https://doi.org/10.1016/j.compbiomed.2021.104431.

[6] V. Chaniotakis, L. Koumakis, H. Kondylakis, G. Notas, D. Plexousakis and M. Tsiknakis, "Predictive Analytics Based on Open Source Technologies for Acute Respiratory Distress Syndrome," *2021 IEEE 34th International Symposium on Computer-Based Medical Systems (CBMS)*, 2021, pp. 68-73, doi: 10.1109/CBMS52027.2021.00019.

[7] Gachet, D., de la Luz Morales, M., de Buenaga, M., Puertas, E., Muñoz, R. (2016). Distributed Big Data Techniques for Health Sensor Information Processing. In: García, C., Caballero-Gil, P., Burmester, M., Quesada-Arencibia, A. (eds) Ubiquitous Computing and Ambient Intelligence. UCAmI 2016. Lecture Notes in Computer Science(), vol 10069. Springer, Cham. https://doi.org/10.1007/978-3-319-48746-5_22

[8] Salloum, S., Dautov, R., Chen, X. *et al.* Big data analytics on Apache Spark. *Int J Data Sci Anal* **1,** 145–164 (2016). https://doi.org/10.1007/s41060-016-0027-9

[9] Junaid Rasool, Mir & Kang, Hardeep & Brar, Amanpreet. (2020). Machine Learning on Healthcare Big Data Using Apache Spark. 10.5281/zenodo.6374971.

[10] Sadia Anzum, "Patient Survival Prediction Dataset", [Online], Available at: https://www.kaggle.com/datasets/sadiaanzum/patient-survival-prediction-dataset

[11] Critical Care Medicine: January 2019 - Volume 47 - Issue 1 - p 17doi: 10.1097/01.ccm.0000550825.30295.dd

[12] Sarvamangala, D.R., Kulkarni, R.V. Convolutional neural networks in medical image understanding: a survey. Evol. Intel. 15, 1–22 (2022). https://doi.org/10.1007/s12065-020-00540-3

[13] Navdeep Tangri, David Ansell, David Naimark, Predicting technique survival in peritoneal dialysis patients: comparing artificial neural networks and logistic regression, *Nephrology Dialysis Transplantation*, Volume 23, Issue 9, September 2008, Pages 2972–2981, https://doi.org/10.1093/ndt/gfn187

[14]