2018

Utilizing data mining techniques and ensemble learning to predict development of surgical site infections in gynecologic cancer patients

John R. McDonough
Binghamton University--SUNY, jmcdono1@binghamton.edu

UTILIZING DATA MINING TECHNIQUES AND ENSEMBLE LEARNING TO PREDICT DEVELOPMENT OF SURGICAL SITE INFECTIONS IN GYNECOLOGIC CANCER PATIENTS

BY

JOHN R. MCDONOUGH

B.S. New York Institute of Technology, 2015

THESIS

Submitted in partial fulfillment of the requirements for
the degree of Master of Science in Industrial and Systems Engineering
in the Graduate School of
Binghamton University
State University of New York
2018

Accepted in partial fulfillment of the requirements for
the degree of Master of Science in Industrial and Systems Engineering
in the Graduate School of
Binghamton University
State University of New York
2018

May 10, 2018

Dr. Mohammad T. Khasawneh, Faculty Advisor
Department of Systems Science and Industrial Engineering, Binghamton University

Dr. Changqing Cheng, Member
Department of Systems Science and Industrial Engineering, Binghamton University

Dr. Sreenath Chalil Madathil, Member
Department of Systems Science and Industrial Engineering, Binghamton University

Catherine Licitra, Member
Division of Quality and Safety

Abstract

Surgical site infections are costly to both patients and hospitals, increase patient mortality, and are the most common form of a hospital acquired infection. Gynecological cancer surgery patients are already at higher risk of developing an infection due to the suppression of their immune system. This research leverages popular data mining techniques to create a prediction model to identify high risk patients. Implemented techniques include logistic regression, naive Bayes, recursive partitioning and regression trees, random forest, feed forward neural network, k-nearest neighbor, and support vector machines with linear kernel. Weighted stacked generalization was implemented to improve upon the individual base level model's performance. The chosen meta level classifiers were support vector machines with linear kernel, logistic regression, and k-nearest neighbor. The result is a model that identifies high-risk patients immediately following a surgical procedure with an AUC of 0.6864, accuracy of 0.6744, sensitivity of 0.7, and specificity of 0.6728.

Acknowledgements

I would like to thank Dr. Mohammad Khasawneh for being an incredible advisor throughout my graduate studies and WISE career. Without his continued support and guidance, I would have never been able to pursue such a rewarding career. When I first met Dr. Khasawneh, he introduced me to a whole new field of engineering; and his passion for the program has been a continual source of inspiration to me.

Thank you to the faculty and staff of the Watson Institute for Systems Excellence (WISE) for granting me the opportunity to conduct interesting and meaningful research in an incredible setting. Being able to gain professional experience while continuing my studies was an incredibly challenging, but rewarding experience that paved the way for my career.

I would also like to thank Patrick Samedy and Scott Ogden for their support throughout the thesis process. They always made themselves available to help me with my research no matter the request and served as mentors throughout my WISE career.

Finally, I would like to thank my mother, Catherine, my father Robert, and my sister, Lauren for their support throughout my undergraduate and graduate studies. Your love has shaped me into the person that I am today and I owe my success to you.

Table of Contents

List of Tables

List of Figures

1. Introduction

According to the Centers for Medicare and Medicaid Services (CMS), in 2015 the United States spent $3.2 trillion on healthcare which equates to $9,900 dollars a person; more than any other country (CMS, 2015). The increased spending and growth in hospital care, private health insurance, physician and clinician services, Medicaid, and prescription drug services results in 17.8% of the United States Gross Domestic Product (GDP) being spent on healthcare (CMS, 2015). Therefore, there is a strong need to reform the way healthcare is delivered so that it is more economical, safe, and of a higher quality. Breaking down the spending by service shows that hospital care accounts for the largest proportion of the United States healthcare spending.

A significant portion of hospital care spending results from the increased cost of care that results from treating surgical site infections (SSI). In fact, it is estimated that each infection costs an additional $20,000 per instance (Gbegnon, 2010). Not only is there a significant monetary cost associated with patients who develop an SSI, but there is also a serious impact on patient satisfaction and quality of care. In fact, patients who develop an SSI have longer length of stays, higher mortality, and higher readmission rates (Gbegnon, 2010).

With such a significant cost associated with development of a surgical site infection, it is imperative that patients at high risk of developing an SSI are able to be determined. This research uses strategic analysis to reduce the impact a surgical site infection has on hospital care spending and patient quality of life.

1.1 Problem Statement

Surgical site infections are the most common form of hospital acquired infections, and as such should be completely preventable. According to the Centers for Disease Control and Prevention (CDC), an SSI is "an infection that occurs after surgery in the part of the body where the surgery took place" and occurs within 30 days following the surgery (CDC, 2012). SSIs are of interest to hospitals due to their expensive treatment cost and delays. In fact, SSI's account for more than 400,000 extra hospital days accounting for an additional $10 billion in care each year (WHO, 2016). There are three types of surgical site infections that are prevalent in hospitals:

1. Superficial - involves only skin and subcutaneous tissue of the incision
2. Deep - involves deep soft tissues of the incision (e.g., fascial and muscle layers)
3. Intra-abdominal - infection involves any part of the body deeper than the fascial/muscle layers, that is opened or manipulated during the operative procedure (Lachiewicz, 2015)

Among gynecological cancer patients surgical site infections can add an additional unnecessary expense to an already expensive hospital stay. SSIs are prevalent among gynecological cancer patients because the surgeries are performed on bacteria prone sites, and cancer patients are already at a higher risk for infections (Lachiewicz, 2015). SSIs increase the length of stay a patient stays in the hospital, and often require the need for readmission. Specific interventions, both pre and post surgery, can be implemented to reduce instances of SSI, but require appropriate identification of patients at risk of developing an SSI.

Prediction models using data mining techniques for SSIs have been utilized in numerous studies where significant features and interventions have been identified. Such prediction models have been successfully implemented for colorectal patients where they reduced the surgical site infection rate in cancer patients. However,

gynecological cancer patients have not been studied specifically to predict such infections.

1.2 Research Objectives

The objective of this research is to utilize data mining techniques to predict development of a Surgical Site Infection in gynecological cancer patients. The prediction is made based upon information available immediately following surgery. The prediction model will be used to assess patients wound treatment and post surgical care to ensure that high risk patients are receiving the necessary medical attention. The objective is completed through a 2-step approach.

1. Predicting Individual Gynecological Surgical Site Infection

Individual gynecological surgical site infection is predicted using seven unique data mining techniques. The techniques utilized are Logistic Regression, Naive Bayes, Random Forest, Feed Forward Neural Network, Recursive Partitioning and Regression Trees, K-Nearest Neighbors, and Support Vector Machines with Linear Kernel. The aforementioned techniques are used to predict whether or not an individual patient will develop a surgical site infection. Individual patient characteristics, including past medical history and demographics, are utilized alongside details of the surgery in each of the data mining techniques. The prediction performance of each of the techniques is compared with specific emphasis on the best performing techniques.

2. Predicting Individual Gynecological Surgical Site Infection Using Ensemble Learning

Individual gynecological surgical site infection is predicted again using ensemble learning, specifically stacking, techniques. The predictive probabilities from the best three performing techniques are utilized as inputs in three data mining techniques: Support Vector Machines with Linear Kernel, Logistic Regression, and K-Nearest

Neighbors. Higher weight was assigned to the predictive probabilities of the overall best performing model. The prediction performance of the ensemble stacking models was then compared with one another as well as with performance of the first seven data mining techniques.

1.3 Research Contributions

This research addresses a gap in the literature regarding predicting gynecological surgical site infections specifically in cancer patients. Extensive literature exists on predicting surgical site infections in patients both pre and post surgery, but there is little emphasis on cancer patients. This research utilizes data mining techniques to predict an individual patient's risk of developing a surgical site infection following gynecologic surgery based on medical history, surgical characteristics, and patient demographics.

Uniquely this research goes a step beyond individual prediction by utilizing ensemble learning, specifically stacking algorithms, to improve the predictions of patients developing an SSI based upon information available immediately following surgery. Stacking algorithms have become popularized by data science competitions such as Kaggle due to their high predictive performance (van Veen et al., 2015). Stacking algorithms have made their way into healthcare through classification of microarray cancer genes (Nagi, 2013). Another gap in the literature exists as there is no specific instance of applying stacking algorithms to prediction of surgical site infections in gynecologic cancer patients.

Finally, this research offers a comparison of seven popular individual prediction model performances which is more than what is found in the literature. Additionally, the performance of three stacking algorithms were compared to each other and the individual models in an effort to determine the most appropriate model.

In conclusion surgical site infections are a metric of poor quality of care and lead to increased treatment costs and reduced patient satisfaction. Hospitals have identified a need for models that predict the risk of surgical site infections with the most accurate results through data mining techniques. In this effort to achieve the best performing model this research applies a unique multilayer approach to generate predictions of surgical site infections in gynecological cancer patients.

1.4 Thesis Organization

This thesis is organized by the following: Chapter 2 details an overview of the literature used to determine factors of interest, individual prediction of surgical site infections, and ensemble learning. Chapter 3 details the methodology used in this thesis and starts with a description of the data acquisition and preprocessing process (Chapter 3.1 & 3.2). Next a description of the feature selection process is covered (Chapter 3.3), followed by a review of the data mining algorithms implemented (Chapter 3.4). The section concludes with a description of the ensemble models used in this research (Chapter 3.4). Chapter 4 details the metrics used to assess the performance of the chosen data mining algorithms (Chapter 4.1) and the performance of the ensemble models (Chapter 4.2). Finally, Chapter 5 summarizes and draws conclusion from this research and offers areas to consider for future work.

2. Literature Review

2.1 Factors of Interest

Shapiro et al. undertook one of the earliest studies to determine risk factors for surgical site infections following a hysterectomy in 1982. Logistic regression was applied to 1,448 patient's information who had a hysterectomy between February 1976 and April 1978. The seven significant factors found to be significant predictors of postoperative surgical site infection were increased duration of surgery, antibiotic prophylaxis, age, surgical procedure, obesity, blood loss, and surgeon.

Fagotti et al. found predictors of developing abscesses in gynecological cancer patients undergoing surgery, particularly that duration of surgery, type of surgery, and the use of absorbable hemostats were significant. They applied logistic regression to a dataset of 360 patients to accurately predict the occurrence of pelvic abscesses following gynecological surgery with an area under Receiver Operating Characteristic (ROC) curve of 0.868, as described further in section 4.1.2.

Lake et al. is another study that identifies the risk factors of developing an SSI after surgery, in this case looking at hysterectomies among 13,822 women. Using data from the national database collected by the ACS NSQIP (American College of Surgeons National Surgical Quality Improvement Program) they found that significant factors included diabetes mellitus, BMI, cancer, ASA class, duration of surgery, race, smoking, and anemia. Descriptive statistics, Student *t* test, Pearson x^2, and Fisher exact test (two-sided) were performed for bivariate analysis. Variables were added to the model in a stepwise fashion utilizing forward selection ($p \leq .05$).

Bakkum-Gamez et al. used data collected from the Mayo clinic among endometrial cancer patients between 1999 and 2008 to determine the costs of SSI. in particular, they looked to determine the risk factors for SSI in endometrial cancer patients, providing reference to managing the costs associated with SSI. They found that among 1,369 patients, 136 or 9.9% had SSI. They used a Fisher Exact test and Wilcoxon Rank test to determine the individual factors associated with the 30 day cost of SSI. Those factors include BMI, ASA score, Diabetes mellitus, pulmonary dysfunction, anemia, age, smoking, MRSA history, duration of surgery, blood loss, lymphadenectomy, bowel resection, vascular disease. They found these factors to significantly influence the cost of SSI and patient mortality.

Mahdi et al. 2014 also used data collected from the ACS NSQIP database, from 2005 to 2011 to determine the rate and predictors of SSI in gynecological cancer patients. Of 6854 patients, 369 or 5.4% were diagnosed with SSI. They used logistic and linear regression to determine risk factors, with all tests of significance being found at the level of $p < 0.005$. Most significantly they found that Endometrial cancer, obesity, ascites, ASA score ≥ 3, blood transfusion, hypoalbuminemia, respiratory comorbidities were high risk indicators.

In 2015, Lachiewicz et al. identify the risk factors contributing to SSI that occur after gynecological surgeries by assessing those risks against the use of antibiotic prophylaxis. They used a cross-sectional analysis of the ACS NSQIP database to identify the host risk factors, preoperative risk factors, intraoperative risk factors, and postoperative risk factors for SSI following any gynecological surgery. Significant factors included BMI, Diabetes mellitus, anemia, smoking, age, malnutrition, history of radiation, MRSA, length of surgery, blood loss, blood transfusion, bowel resection, lymphadenectomy, and preoperative and postoperative glucose levels.

More recently, Johnson et al. used bundled interventions to reduce SSI among gynecological surgery patients, finding that there was a reduction from 6% to 1.1% in SSI with the implementation of the bundle. They used Fisher Exact test to determine significant factors within two datasets, a pre-intervention period January 1, 2010 and December 31, 2010, and a post-intervention period August 1, 2013 and September 30, 2014. Significant factors included bowel resection, cancer location, surgical procedure (laparoscopic vs open) and duration of surgery.

A summary of the significant factors identified in the literature and their corresponding papers is shown in table 1.

Table 1 - Literature Review: Significant Factors

	Shapiro et al.	Fagotti et al.	Lake et al.	Bakkum-Gamez et al.	Mahdi et al.	Lachiewicz et al.	Johnson et al.
Age	✓	×	×	✓	×	✓	×
Anemia	×	×	✓	✓	×	✓	×
Antiobiotic Prophylaxis	✓	✓	×	×	×	×	×
ASA Score	×	×	✓	✓	✓	×	×
Blood Loss	✓	×	×	✓	×	×	×
BMI	✓	×	✓	✓	✓	✓	×
Bowel Resection	×	×	×	✓	×	✓	✓
Cancer	×	×	✓	×	✓	×	✓
Diabetes Mellitus	×	×	✓	✓	×	✓	×
Duration of Surgery	✓	✓	✓	✓	×	✓	✓
History of Radiation	×	×	×	×	×	✓	×
Laparoscopic vs Open Surgery	✓	✓	×	×	×	×	✓
Lymphadenectomy	×	×	×	✓	×	✓	×

Malnutrition	×	×	×	×	✓	✓	×
Postglucose	×	×	×	×	×	✓	×
Preglucose	×	×	×	×	×	✓	×
Race	×	×	✓	×	×	×	×
Smoking History	×	×	✓	✓	✓	✓	×
Surgeon	✓	×	×	×	×	×	×
MRSA History	×	×	×	✓	×	✓	×

2.2 Individual Models and Ensemble Learning

Sands et al. uses logistic regression, recursive partitioning and regression trees to predict if a patient would develop a surgical site infection, using data collected automatically by health care systems from 4,086 procedures. Significant predictors were the prescriptions and dispensing of specific antibiotics, outpatient diagnosis, readmission with specific diagnosis, wound culture ordered, and emergency department visit. This study found that using recursive partitioning to create decision trees is a better approach to predict development of a surgical site infection than logistic regression. The accuracy of the developed model was 0.74 and the sensitivity was 0.42.

Fowler et al. analyzed 331,429 coronary artery bypass graftings to determine predictors of postoperative surgical site infection. Utilizing logistic regression Fowler et al., found that BMI, diabetes mellitus, previous myocardial infarction, hypertension, and urgent operation were found to be significant predictors of SSI following the cardiac procedure. Based on the identified significant features a prediction model was created that was able to predict the occurrence of surgical site infection with a c-index of 0.686.

Neumayer et al. analyzed 163,624 patients undergoing vascular and general surgery to develop a model to determine patients at high risk of developing an SSI. Through the use of Logistic Regression, the team was able to predict the occurrence of an SSI with an area under the ROC curve of 0.62. The factors found significant during the development of the model were age, diabetes, dyspnoea, steroids, alcoholism, smoking, prior radiology treatment, ASA score, Albumin, wound classification, and procedure type.

In a study done by Heckerling et al. in 2007 the use of Artificial Neural Networks was used to create a model for the prediction of urinary tract infections in women, as well as determine significant predictors. The study incorporated patient information from 212 women ages 19 to 84. The determined significant predictors were urinary frequency, dysuria, urine odor, symptom duration, diabetes mellitus, red blood cells, and infection history. The developed artificial neural network model was used to classify urinary tract infections with an area under the ROC curve of 0.792.

Looy et al. investigated the use of support vector regression in 2007 to predict tacrolimus blood concentration in liver transplants. More than 16,000 blood samples were analyzed from 50 liver transplant patients. The results from the linear support vector regression were compared to the results from a multiple linear regression model developed on the same data. The mean absolute difference between the observed and predicted values was 2.31 for linear support vector regression and 2.73 for multiple linear regression. In the study gender, age, weight, days since transplantation, and 12 biochemical variables were found to be significant factors.

In 2008 Verplanke et al. compared the performance between support vector machines and logistic regression to model patient mortality for patients with hematological malignancies. 352 patients admitted to the ICU between 1997 and 2006, including those with a life-threatening complication, were analyzed. From the developed

models gender, high grade malignancy, active disease, bone marrow transplant, infection history, and ventilation were the factors identified as significant. The patient mortality predictive performance of the logistic regression model was 0.768 AUC and 0.802 AUC for the support vector machine model.

Sill et al. explored the use of feature weighted linear stacking as way to reduce the computational demand of nonlinear stacking methods, but increase the performance of linear stacking alone. The study found that stacking in general improved upon the performance of an individual model such as linear regression used in this study. When a weight is assigned to individual classifiers the predictive performance is increased when compared to stacking without weighting. Stacking in this manner is far less computationally demanding than stacking with a nonlinear algorithm and results in comparable performance.

Mu et al. analyzed data reported to the National Healthcare Safety Network for all operative procedures that took place from January 1, 2006 to December 31, 2008. In total 847 hospitals from 43 contributed 849,649 operations of which 16,147 resulted in an SSI. The goal of the study was to develop a new risk model to improve the predictive performance of SSI for each procedure category. The team implemented a stepwise logistic regression model with bootstrap resampling as a form of bagging ensemble. The developed model resulted in a median c-index of 0.67 as compared to the prior median c-index of 0.6. Additionally, a set of variables determined to be risk factors were developed.

In 2011 Al-Shayea performed a study to investigate and introduce the use of artificial neural networks to diagnose diseases. Specifically, two patient datasets were studied; acute nephritis disease and heart disease. In the study, acute nephritis disease was predicted with an accuracy of 0.99 and mean square error of 1.13e-6 and heart disease was predicted with accuracy of 0.95 and mean square error of 7.48e-2. The

strong predictive performance shown in this study proves the worth of using artificial neural networks and an ensemble of multiple neural networks in healthcare datasets.

Kawaler et al. researched the best methods to predict patients at risk of developing venothromboembolism (VTE) after they had been discharged from the hospital. The prediction model used only data that could automatically be gathered from the patient's electronic health record. The study included data from 720 subjects of which 3,330 unique variables were represented. In order to determine the best model to use for predicting patient's risk several machine learning algorithms were applied to the dataset including Naive Bayes, K-Nearest Neighbor, Support Vector Machine, Classification and Regression Tree, and Random Forest. Additionally, the study applied bagging, boosting, and stacking ensemble methods to the dataset as well in an effort to increase the predictive performance. Low blood volume, infection, inflammation, immobilization, and malnutrition were among the variables that were found to be the most significant risk factors. The study concluded that Naive Bayes, Random Forest, and Support Vector Machine were the best learners for the dataset.

Shouman et al. performed a study in 2012 that investigated the use of k-Nearest Neighbor data mining technique to assist in the diagnosis of heart disease patients. A benchmark dataset was used so that the predictive performance of KNN could be compared to other data mining techniques that had been used on the same dataset in prior studies. Additionally, a bagging ensemble using KNN and majority voting was implemented to determine if there would be an increase in predictive performance. In the prior studies the best performing model was a neural network bagging ensemble with an accuracy of 89.01%. With the implementation of k-nearest neighbors the accuracy was increased to 97.4% with a sensitivity of 93.8% and specificity of 99%. While the implementation of the KNN ensemble did increase the performance over the data mining

techniques implemented in prior studies, the performance was not as high as the individual KNN algorithm.

In 2013 Manilich et al. performed an extensive analysis to determine the key factors that are associated with post surgical complications for patients who received colorectal surgery. Data were collected from the departmental outcomes database for 3,552 who received a colorectal surgery between 2010 and 2011. Then 700 classification models with bootstrap resampling were applied to the dataset. The outputs from the bootstrap models were then used in a stacking ensemble model to further improve the predictive performance. The study found that the duration of the surgery, BMI, age, surgeon, type of surgery (laparoscopic vs open) contributed the most to post colorectal surgery complications.

A study published in 2013 by Legrand et al. looked into risk factors for development of postoperative kidney injury in patients undergoing cardiac surgery with infective endocarditis. Data were gathered between 2000 and 2010 for patients with infective endocarditis with cardiac surgery consisting of a cardio-bypass pulmonary. Ultimately 202 patients were identified to be included for analysis. A stacking ensemble technique was applied to the dataset from which number of surgeries, contrast agent, Vancomycin administration, transfusion preoperative hemoglobin, and age were found to be risk factors. The ensemble was created by using multiple stepwise regression models as the base level classifiers. From the implementation of the stacking ensemble model postoperative acute kidney injury was able to be accurately predicted with area under the ROC curve of 0.76.

Nagi et al. performed a study in 2013 where microarray cancer data was classified using an ensemble approach. 9 different cancer datasets were analyzed separately for a total of 993 entries. First the base level classifiers of decision tree, k-nearest neighbors, and naive bayes were implemented on each of the individual

datasets with their performances in terms of accuracy compared. Then bagging, boosting, and stacking were all implemented on the same datasets to determine if there was an increase in performance. For each of the ensemble methods the data mining techniques used for the individual models were also used as the classifiers for the ensemble models. In each instance implementation of ensemble techniques, specifically stacking lead to the largest gain in accuracy over the individual base level classifiers. The study concluded that the largest performance gains come from using diverse base level classifiers.

In 2013 Rose performed a thorough analysis to determine if stacking ensemble methods could predict patient mortality with higher performance than individual base level classifiers. 2,066 patients were included in this study, who were all residents of Sonoma, California and were aged 54 or more during 1993 to 1999. The machine learning techniques that were applied to these patients were Bayes logistic regression, LASSO, logistic regression, boosted logistic regression, bagging classification, random forest recursive partitioning and regression trees, and neural network. From these models' implementation gender, age, self-rated health, physical activity level, smoking history, and cardiac history were found to be significant predictors of mortality. With the implementation of the stacking ensemble technique the predictive performance was better than any individual model with a R^2 value of 0.201 and Mean Square Error of $9.04e^{-2}$ which is a 20.1% gain in model performance.

A study done by Yap et al. in 2014 aimed to determine the best methods to deal with class imbalance through the prediction of survival following cardiac surgery. The data were obtained from a local hospital and included 4,976 cases of which 4.2% of patients died. The individual base level classifier that was chosen to be implemented was classification and regression tree. Gender, age, comorbidities, surgery type, and wound infection were the factors that were found significant from the base level

classifier. Bagging and boosting ensemble models were also implemented using classification and regression tree as the chosen classification. Use of ensemble techniques facilitated better model performance and handling of the imbalanced dataset. The best performing model had an accuracy of 76.7%, sensitivity of 69.4%, and specificity of 84.5%.

Sanger et al. performed a study in 2016 in which a model was developed to identify patients at high risk of developing a surgical site infection that incorporated daily wound assessment data. 1,000 post-surgery patients were studied at a teaching hospital for 1 to 5 days following their surgery. A naive Bayes model was applied to the patient data, as well as a logistic regression model to determine baseline performance. The three most significant risk factors were c-reactive protein, duration of surgery, and wound contamination. The best performing model was a naive Bayes model that included daily updated features referred to as serial features. The naive Bayes model had an area under the ROC curve of 0.76, sensitivity of 0.8, and specificity of 0.64.

Recently Taylor et al. explored the use of machine learning in predicting mortality in sepsis patients in a hospital. The retrospective study included 5,278 visits with 4,676 unique patients admitted to the hospital after a visit to the emergency department displaying symptoms of sepsis. The chosen machine learning techniques that were implemented were random forest, classification and regression tree, and logistic regression. From these models' analysis blood pressure, age, albumin, heart rate, CO2, acuity level, potassium, heart rate, and respiratory rate were found to be significant predictors of patient mortality in hospital. The models predicted in hospital mortality for the sepsis patients with an AUC of 0.86 for random forest, 0.69 for classification and regression tree, and 0.76 for logistic regression.

2.3 Literature Review Summary

From the literature review it is evident that there is a significant interest in identifying patients at risk of developing a surgical site infection and the associated risk factors. It was found that there are many sources suggesting that the same or similar risk factors are present with respect to development of a surgical site infection post-surgical procedure. Additionally, we can see that similar models are being utilized in healthcare with specific instances of predicting patient's development of surgical site infections. Summaries of the performed literature review are shown in tables 2 and 3. There is a gap in the literature when it comes to predicting development of a surgical site infection specifically in gynecological cancer patients. Additionally, there is limited literature that explores the use of ensemble learning methods in healthcare to improve upon the performance of the individual prediction models.

Table 2 - Literature Review: Factors of Interest

Study	Objective	Methodology	Conclusions	Significant Factors
Shapiro et al., 1982	Determine factors associated with postoperative surgical site infection following a hysterectomy	Logistic Regression	Identified 7 factors found to be significant predictors of postoperative surgical site infection	Duration of operation, antibiotic prophylaxis, age, surgical procedure (laparoscopic vs open), obesity, blood loss, surgeon
Fagotti et al., 2010	Determine risk factors for developing abscesses in gynecological cancer patients undergoing surgery	Logistic Regression	Able to predict the occurrence of pelvic abscesses following gynecological surgery with an area under ROC curve of 0.868	Duration of surgery, type of surgery, use of absorbable hemostats

Lake et al., 2013	Estimate the occurrence of SSIs after a hysterectomy and the associated risk factors	Logistic Regression, Student t-test, Fisher Exact test	Identified risk factors for SSI following a hysterectomy and need for a model to predict SSIs	Diabetes mellitus, BMI, cancer, ASA class, duration of surgery, race, smoking, anemia
Bakkum-Gamez, et al., 2013	Determine risk factors for SSI in endometrial cancer patients to manage cost of treating SSI	Fisher Exact test, Wilcoxon Rank Sum test	Determined risk factors for SSI of endometrial cancer patients and the associated 30 day cost of SSI in the cohort	BMI, ASA score, Diabetes mellitus, pulmonary dysfunction, anemia, age, smoking, MRSA history, duration of surgery, blood loss, lymphadenectomy, bowel resection, vascular disease
Mahdi et al., 2014	Determine rate and predictors of surgical site infections following gynecologic cancer surgery	Logistic Regression	5.4 % of patients undergoing gynecologic cancer surgery developed an SSI. Determination of significant factors helps identify patients at risk of developing an SSI	Endometrial cancer, obesity, ascites, ASA score ≥ 3, blood transfusion, hypoalbuminemia, respiratory comorbidities

Lachiewicz et al., 2015	Review national database to determine the risk factors that increase the chance of developing an SSI following pelvic surgery	Cross sectional analysis of American College of Surgeon's National Surgical Quality Improvement Program patient files	Identified host risk factors, preoperative risk factors, intraoperative risk factors and postoperative risk factors for SSIs following gynecologic surgery	BMI, Diabetes mellitus, anemia, smoking, age, malnutrition, history of radiation, MRSA, length of surgery, blood loss, blood transfusion, bowel resection, lymphadenectomy, preglucose, postglucose
Johnson et al., 2016	Determine if implementation of a bundle containing evidence-based practices can reduce surgical site infection rate	Fisher Exact test	Infection rate reduced from 6% to 1.1% with implementation of bundle	Bowel Resection, Cancer Location, surgical procedure (laparoscopic vs open), duration of surgery

Table 3 - Literature Review: Data Mining Techniques

Study	Objective	Methodology	Conclusions	Significant Factors
Sands, et al., 1999	Develop an efficient way to predict patients who will develop an SSI based on information collected by health care systems automatically	Logistic Regression, Recursive Partitioning and Regression Trees	Able to predict SSI post discharge with accuracy of 0.74 and sensitivity of 0.42 with Recursive Partitioning and Regression Trees, the better performing of the two models	Prescription and dispensing of specific antibiotics, outpatient diagnosis, readmit with specific diagnosis, wound culture ordered, wound care, emergency department visit
Fowler, et al., 2005	Determine predictors of SSIs following Cardiac Surgery	Logistic Regression	Able to predict the occurrence of SSI following cardiac surgery with c-index of 0.686	BMI, diabetes mellitus, previous myocardial infarction, hypertension, urgent operation
Neumayer, et al., 2007	Develop a model to determine patients at high risk for an SSI	Logistic Regression	Able to predict the occurrence of an SSI with area under the ROC curve of 0.62	Age, diabetes, dyspnoea, steroids, alcoholism, smoking, prior radiology treatment, ASA score, Albumin, wound classification, procedure type

Heckerling, et al., 2007	Implement Artificial Neural Networks to determine factors of interest and create models to predict urinary tract infections in women	Artificial Neural Networks	Identified significant variables for predicting urinary tract infections and able to classify urinary tract infections with area under the ROC curve of 0.792	Urinary frequency, dysuria, urine odor, symptom duration, diabetes mellitus, red blood cells, infection history
Looy, et al., 2007	Investigate use of Linear Support Vector Regression in predicting tacrolimus blood concentration	Support Vector Regression, Multiple Linear Regression	Mean absolute difference between observed and predicted was 2.31 for support vector regression, and 2.73 for multiple linear regression	Gender, age, weight, days since transplantation, 12 biochemical variables
Verplancke, et al., 2008	Compare the performance of Logistic Regression and Support Vector Machines when predicting mortality of patients with hematological malignancies	Logistic Regression, Support Vector Machines	Predict patient mortality with AUC of 0.768 for Logistic Regression and 0.802 for Support Vector Machines	Gender, high grade malignancy, active disease, bone marrow transplant, infection history, ventilation
Sill, et al., 2009	Explore the use of Feature Weighted Linear Stacking as a less computationally demanding way of increase performance of individual classifiers	Linear Regression, Feature Weighted Linear Stacking	Assigning a weight to individual classifiers increase the predictive performance as compared to linear stacking without weights, and is less computationally demanding than weighting nonlinear stacking algorithms	Not specified

Mu et al., 2011	Develop new risk models to improve predictive performance of surgical site infection by procedure category	Stepwise Logistic Regression with Bootstrap resampling	Median c-index (area under ROC curve) increased to 0.67 from 0.6 and developed a set of variables that were determined to be risk factors	Not specified
Al-Shayea, 2011	Introduce examples in healthcare where Artificial Neural Networks were successfully implemented specifically in diagnosing diseases	Artificial Neural Networks, Stacking ensemble with multiple Neural Networks	Diagnosed Acute Nephritis with Mean Square Error of 1.13e-6 and accuracy of 0.99 and Diagnosed Heart Disease with a MSE of 7.48e-2 and accuracy of 0.95	Not specified
Kawaler, et al., 2012	Predict patients at risk for developing venothromboembolism post hospitalization based on automatically generated data electronic health records	Naive Bayes, K-Nearest Neighbor, Support Vector Machine, Classification and Regression Tree, and Random Forest. Used bagging, boosting, and stacking ensemble methods as well	Naive Bayes, Random Forest, and Support Vector Machines prove to be the best learners for this dataset	Low blood volume, infection, inflammation, immobilization, malnutrition

Shouman, et al., 2012	Determine if applying K-Nearest Neighbors will help health care professionals diagnose heart disease	K-Nearest Neighbors	The maximum accuracy on the same dataset was 89.01% using Neural Network Ensembles. When K-Nearest Neighbors was applied the accuracy increase to 97.4% while sensitivity was 93.8% and specificity 99%	Age, blood pressure, smoking, cholesterol, diabetes, hypertensions, family history, obesity, and lack of physical activity
Manilich et al., 2013	Create a model that will help determine the most important factors that are associated with post-surgical complications	Stacking Ensemble Technique	Identified factors that contribute the most to post surgery complications in an effort to minimize the occurrence of complications	Duration of surgery, BMI, age, surgeon, type of surgery
Legrand, et al., 2013	Determine the risk factors for acute kidney injury following surgery for infective endocarditis	Stacking Ensemble Technique	Predicted postoperative acute kidney injury with an AUC of 0.76 using a stacking ensemble of stepwise regression models	Number of surgeries, contrast agent, Vancomycin administration, transfusion preoperative hemoglobin, age

Nagi, et al., 2013	Compare performance of bagging, boosting, and stacking ensemble models to individual models for classification of microarray cancer data	Decision Tree, K-Nearest Neighbors, Naive Bayes	Implementation of bagging, boosting, and stacked algorithms for each of the data mining techniques increases the performance over the original models. The largest performance gain results from using stacking algorithms with diverse base classifiers	Not specified
Rose, 2013	Determine if stacked ensemble models have higher predictive performance for mortality than individual machine learning methods	Bayes Logistic Regression, LASSO, Logistic Regression, Boosted Logistic Regression, bagging classification, Random Forest Recursive Partitioning and Regression Trees, Neural Network, Stacking ensemble	The stacking ensemble model was able to achieve higher performance than any of the individual models or ensemble models with a R squared of 0.201 and Mean Square Error of 9.04e-2, which is a 20.1% gain in performance	Gender, age, self-rated health, physical activity level, smoking history, cardiac history

Yap, et al., 2014	Determine best methods to deal with imbalanced datasets through prediction of cardiac surgery survival using decision tree models	Classification and Regression Tree	Bagging and boosting techniques facilitated better model performance with accuracy of 76.7, sensitivity of 69.4 and specificity of 84.5 in the best performing model	Gender, age, comorbidities, surgery type, wound infection
Sanger et al., 2016	Develop a model to identify patients for high risk of surgical site infection that incorporates daily wound assessment	Naïve Bayes, Logistic Regression	Able to predict occurrence of SSI with area under the ROC curve of 0.76, sensitivity of 0.8, and specificity of 0.64	C-reactive protein, duration of surgery, and contamination
Taylor, et al., 2016	Explore the use of machine learning techniques to predict Sepsis patient mortality in hospital	Random Forest, Classification and Regression Tree (CART), Logistic Regression	Predicted in-hospital mortality for sepsis patients with AUC of 0.86 for Random Forest, 0.69 for CART, and 0.76 for Logistic Regression	Blood pressure, age, albumin, heart rate, CO2, acuity level, potassium, heart rate, respiratory rate

3. Methodology

The methodology used in this research is summarized in figure 1. First the area of research was focused in on by determining the scope. Next a literature review along with consultation of clinical expertise was performed in order to determine the factors of interest to be utilized in the prediction models. The relevant data were then pulled from a multitude of sources. Before the data were able to be analyzed some initial cleaning including data transformations and grouping was required, as outlined in section 3.2.1. Following preprocessing of the data feature selection needed to take place so that only the most important variables with the highest predictive power were included in the model. The feature selection process is outlined in section 3.3, and in section 3.4 a variety of data mining techniques were utilized to predict individual gynecological surgical site infection. These methods include Logistic Regression, Naive Bayes, Random Forest, Feed Forward Neural Network, Recursive Partitioning and Regression Trees, K-Nearest Neighbors, and Support Vector Machines with Linear Kernel. The performance of each individual model was compared in order to determine the models that would be combined in an ensemble stacking algorithm as outlined in section 3.5. The ensemble stack models used include Support Vector Machines with Linear Kernel, Logistic Regression, and K-Nearest Neighbors. The performance of the ensemble models was compared with one another, as well as with the individual models. The data mining techniques were performed using RStudio, open-source Integrated Development Environment version 1.0.153. Additionally, feature selection and the issue of imbalanced classes were also addressed using RStudio (Dutta, 2016).

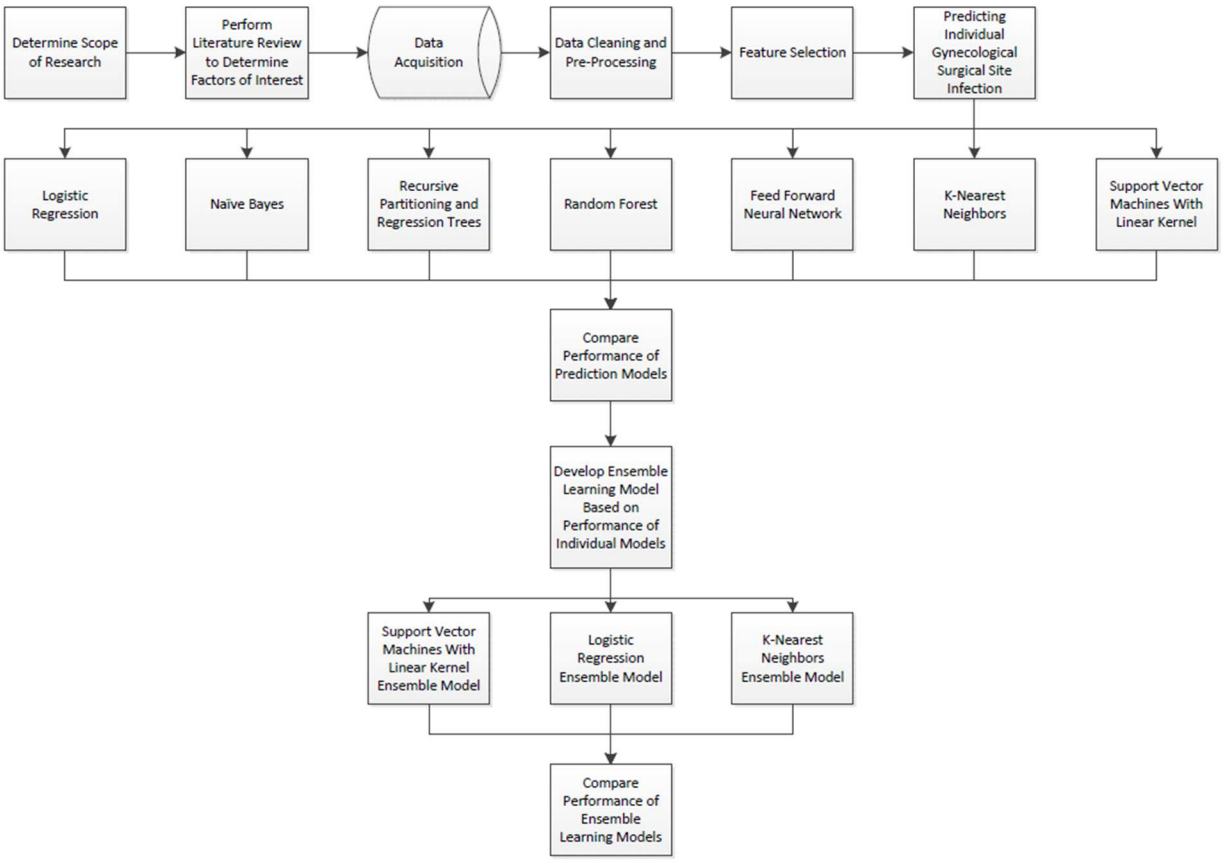

Figure 1 - Flowchart of Research Methodology

3.1 Scope and Factors of Interest

In order to be able to predict individual gynecological surgical site infection the scope of the research needs to be determined along with the factors of interest that will be utilized with the data mining techniques. This research focuses on gynecological cancer patients who require a surgical procedure as part of their treatment. As part of the requirement mandated by the Joint Commission, patients who developed a surgical site infection 30 days post-surgical procedure were documented and reported. It was decided that the scope of this research would focus on the inpatients who developed a surgical site infection after a surgical site infection reduction bundle was implemented. These patients underwent surgery more recently, and are therefore more pertinent to the research.

The factors of interest were based upon those identified in the literature review as well as their presence in the clinical systems. Additionally, there was consultation with an infection control team to identify additional factors of interest based on clinical expertise. These factors can be separated into the categories of patient medical history, patient demographics, and surgical characteristics.

Table 4 - Factors of Interest

Factor Category	Factor Name	Factor Description
Patient Medical History	Diabetes Mellitus	Patient diabetes status at the time of the surgery
	ASA Class	American Society of Anesthesiologists Classification (ASA I, ASA II, ASA III, ASA IV, ASA V, ASA VI)
	CCI	Charlson Comorbidity Index based upon 10 year survival period for patients with multiple comorbidities
	Chemo Flag	Signifies if patient has received chemotherapy prior to the surgery
	MI	Signifies if the patient has had a myocardial infarction prior to the surgery
	PVD	Peripheral vascular disease status at the time of the surgery
	COPD	Chronic obstructive pulmonary disease status at the time of the surgery
	Bowel Resection	Signifies if the patient has had a bowel resection prior to the surgery

	Smoking	Patient smoking status at the time of the surgery (Current, Former, Never)
	Alcohol	Patient alcohol use at the time of the surgery (Current, Former, Never)
	MRSA	Signifies if patient was diagnosed with Methicillin-resistant Staphylococcus aureus (MRSA) prior to surgery
	Number of Surgeries	Number of surgeries patient has had prior to surgery
	Days Between Surgeries	Number of days between prior surgery and current surgery
	Time Between Chemo and Last Surgery	Number of days between prior chemotherapy and current surgery
	Preglucose	Glucose levels prior to surgery
	Postglucose	Glucose levels following surgery
	Albumin	Albumin levels at the time of the surgery
	HGB	Hemoglobin levels at the time of the surgery
	Admissions	Number of hospital admissions the patient had one year prior to the surgery
	BMI	Body Mass Index at the time of the surgery
	WBC	White Blood Cell count at the time of the surgery

	Age	Patient age at the time of the surgery
Patient Demographics	Race	Patient identified race (White, Black, Asian, Native American, Other, Patient refused to answer, or Unknown)
	Marital Status	Marital status of patient (Married, Single, Divorced, Separated, Widowed, Unknown)
	Language	Patient preferred language (English or Non-English)
	Insurance	Insurance type for patient (Private, Medicaid, Medicare, No Insurance, Other)
	Discharge Location	Patient destination post discharge (Hospice, Other Hospital, Death, Routine Discharge, Rehabilitation Facility, Home Health Care)
	Median Income	Median Income based on the patient's zip code
Surgical Characteristics	Wound Class	Surgical wound classification (Clean, Clean-Contaminated, Contaminated, or Dirty)
	Laparoscopic Vs Open	Signifies whether the surgery was laparoscopic (many small incisions) or open (one large incision)
	Cancer Category	Location of patient’s cancer (Cervix, Ovary/Fallopian, Uterine Endometrium, Vagina, Vulva, Other)
	Surgeon	Dummy variable representing surgeon performing surgery
	Time Between Patient Surgery Start	Time between patient entering surgery room and first surgical activity
	Duration of Surgery	Length of the surgery in minutes

	Total Blood Loss	Amount of blood lost (in milliliters) during the surgery

3.2 Data Source

The data were pulled for 693 surgeries, following the surgical site infection reduction bundle that took place from February 3rd, 2015 to May 15th, 2017. The following additional fields were also pulled alongside the identified factors of interest: Patient Medical Record Number, date and time of the surgery start, date and time of the surgery end, patient admission date, date of last chemo treatment, patient date of birth, and patient zip code to aid in the calculation of the factors of interest.

The original dataset included 1,141 surgeries, and the dataset considered for this research contained 693 surgeries, post-surgical site infection reduction bundle. The reduction bundle involved implementing mandated interventions in order to reduce the surgical site infection rate. From the 693 surgeries, there were 664 unique patients. Additionally, 93.94%, or 651 surgeries, did not result in a surgical site infection while 6.06%, or 42 surgeries, resulted in a surgical site infection. The patient's zip code was used as a reference to determine median household income.

3.2.1 Data Cleaning

The data required significant transformation before they could be analyzed. Dummy variables needed to be assigned to categorical predictors. These dummy variables consisted of integers starting at one, ensuring the number zero was not used, and increasing by one until a number was assigned to each of the levels in a categorical variable. An example of this is for the variable smoking, 1 represents patients who have never smoked, 2 = Former, and 3 = Current. Additionally, certain variables had a

significant number of levels so the data were grouped in order to improve the performance of the predictions. Originally, there were 78 identified languages. These were grouped to represent patients whose identified primary language was either English or Non-English based upon a review of the literature and clinical consultation. The levels for each of the categorical variables are summarized in table 4.

Numerical values also require transformation to ensure each of the inputs utilizes the same range of values to avoid over influence of certain variables (Quackenbush, 2002). In order to achieve this result numerical variables were normalized or feature scaled between 0 and 1 utilizing equation 1.

$$X_{new} = \frac{X - min(X)}{max(X) - min\ (X)} \quad \textbf{(1)}$$

3.2.2 Data Calculations

Before the data were analyzed simple calculations were required in order to obtain the identified features of interest. The duration of the patient's surgery was calculated by taking the time difference, in minutes, between the start of the surgery and the end. Additionally, the patient's age at the time of the surgery was determined by calculating the time difference, in years, between the patient's date of birth and the start of the patient surgery. Finally, the number of days since the patient's last chemotherapy treatment was calculated by taking the difference in days between the patient's last chemotherapy treatment and the surgery start.

3.3 Feature Selection

Feature selection is an important data mining technique as it reduces the number of features in an effort to remove irrelevant attributes (KDnuggets, 2017). When features with high predictive power are combined with irrelevant attributes the result is a model

that is not as generalizable and has worse predictive performance. When the number of features has been narrowed down, correlated or related variables are eliminated to reduce the impact of negative interactions. Additionally, including more features in the models than is necessary increases the chances of data missing, as well as increasing the time and computational strain it takes to train the models (Deshpande, 2011).

3.3.1 Univariate Analysis

As an initial, and somewhat limited, form of feature selection univariate analysis was performed in order to determine individual variables that have a significant impact on patients developing a surgical site infection following surgery. This form of analysis does not consider interactions between variables, but rather individual interactions. For categorical variables, both binary and multiclass, a G-test and Fisher Exact test were implemented to determine significant variables. For continuous variables, a Student's t-test and Wilcoxon Rank Sum test were implemented to determine significant variables.

3.3.1.1 Categorical Variables

To determine the significance of individual categorical variables two tests were implemented in order to validate the results. The first test implemented was the G-test of independence. The G-test is used when you have nominal variables and are interested in determining when the proportions of one variable are different for values of different variables (Biostathandbook, 2014). The null hypothesis is that the proportions of one variable are the same for varying values of the second variable. As an example, the G-test helps determine if the proportion of patients who had a prior diagnosis Diabetes Mellitus that developed an SSI is statistically significantly different than the proportion of patients who were not previously diagnosed with Diabetes Mellitus and developed an SSI. In order to perform the G-test of Independence first the G test statistic must be

calculated which is achieved using equation 2. After the value of the G-test has been calculated the degrees of freedom must be calculated by multiplying the number of rows minus one by the number of columns minus one. The associated p-value is then calculated by using the value of the G-test and the degrees of freedom. Since the G-test is more suited for data where the sample size is large, greater than 1000 entries, an exact test is recommended for our data where the sample size is 693.

$$G = 2\sum_{i}^{n} Oi * ln\left(\frac{Oi}{Ei}\right) \quad \textbf{(2)}$$

G = Value of the G-test

n = Total number of observations

O_i = Observed frequency for each value

E_i = Expected frequency for each given value

The Fisher Exact test aims to achieve the same goal as the G-test, but is better suited for data where the number of samples is fewer than 1000 (Biostathandbook, 2014). The null hypothesis is the same as the G-test in that the proportions of one variable are the same for different values of the second variable. To calculate the Fisher Exact test Statistic for a 2 by 2 table equation 3 must be used. This value is calculated by determining the probability of the observed numbers through the hypergeometric distribution The Fisher Exact test can be applied to tables larger than 2 by, however the chi-square test statistic must be calculated for every possible set of numbers. Those with values greater than or equal to the observed data are then considered as extreme as the observed data (Biostathandbook, 2014). In order to determine the statistical significance of the Fisher Exact test value a p-value must be calculated. This is achieved by enumerating all other possible matrices and summing together the p-values of those that

have a Fisher Exact test statistic value less than the overall value of the Fisher Exact test which can be considered a cutoff value.

$$p = \frac{(a+b)!(c+d)!(a+c)!(b+d)!}{a!b!c!d!n!} \quad \textbf{(3)}$$

p = Value of the Fisher Exact test

a,b,c,d = Total count of each cell in the table

n = Total number of observations

As a form of validation if a variable was determined to be significant by both the G-test and the Fisher Exact test the variable was considered for further analysis and inclusion in the prediction models. The four categorical variables that were found to be significant by both tests, p value less than 0.05, are the patient's wound class, whether or not the patient underwent a bowel resection prior to their surgery, the patient's insurance type, and the patient's cancer category. The results of the G-test and Fisher Exact test are summarized in table 5.

Table 5 - Univariate Analysis for Categorical Variables

Categorical Variables	G-Test Results	Fisher Exact Test Results	Significant at 0.05?
Diabetes Mellitus	0.06859	0.1293	No
Wound Class	0.003214	0.001963	Yes
ASA Class	0.8396	0.8349	No
CCI	0.4431	0.6903	No

Chemo Flag	0.1257	0.1199	No
Laparoscopic Vs Open	0.53	0.5493	No
Race	0.9976	0.9845	No
Marital Status	0.9745	0.888	No
MI	0.1933	0.1611	No
PVD	0.08195	0.07959	No
COPD	0.4033	0.6089	No
Bowel Resection	7.62E-09	1.11E-08	Yes
Smoking	0.9168	0.5869	No
Alcohol	0.4531	0.6126	No
Language	0.4788	0.2574	No
Insurance	0.0247	0.0326	Yes
Cancer Category	0.0168	0.0399	Yes
Surgeon	0.8294	0.8321	No
Discharge Location	0.1105	0.1255	No
MRSA	0.0844	0.1176	No

3.3.1.2 Continuous Variables

In order to determine the significance of the continuous variables two tests were also implemented in order to validate the results of identified significant variables. The first test implemented was the Student's t-test for two samples. The Student's t-test is used when there is one continuous variable and one categorical variable where there are only two values (Biostathandbook, 2014) and is generally applied when the sample size is small. The test determines if there is a statistically significant difference in the mean of the two groups by testing the null hypothesis that the mean difference between pairs of observations is zero. For example, the Student's t-test determines if the average number of prior surgeries for patients who developed an SSI is statistically significant that the average number of surgeries who did not develop an SSI. In order to determine the significance of a variable first Student's t-test statistic must be calculated. This is achieved by using equation 4 and 5. To calculate the p value the value of the test statistic is matched with the degrees of freedom, number of observations in the groups minus 2, in order to determine significance.

$$t = \frac{\bar{X}1 - \bar{X}2}{\sqrt{\frac{S1^2}{n1} + \frac{S2^2}{n2}}} \quad \textbf{(4)}$$

$\bar{X}1$ = Mean of the first set of values

$\bar{X}2$ = Mean of the second set of values

S_1 = Standard deviation of the first set of values

S_2 = Standard deviation of the first set of values

n_1 = Total number of values in the first set

n_1 = Total number of values in the second set

$$S = \sqrt{\frac{\sum(x-\bar{x})^2}{n-1}} \quad \textbf{(5)}$$

x = values given

$\bar{x}$ = Mean

n = Total number of values

In order to validate the results of the Student's t-test a second test, the Wilcoxon Rank-Sum test was also implemented. The Wilcoxon-Rank Sum test is used in the same situation as the Student's t-Test, but it has the added stipulation that the differences are not distributed normally (Biostathandbook, 2014). As such the null hypothesis differs in that the median difference between pairs of observations is zero. The test statistic for the Wilcoxon-Rank Sum test is determined by taking the lesser value of U_1 and U_2 as defined by equation 6. In order to determine the significance of the variable the p value must be calculated which is achieved through the use of a U table which contains α, n_1, and n_2. The null hypothesis is then rejected if the value of the test statistic is less than or equal to the critical value identified by the U table (LaMorte, 2017).

$$U1 = n1n2 + \frac{n1(n1+1)}{2} - R1 \quad \textbf{(6)}$$

$$U2 = n1n2 + \frac{n2(n2+1)}{2} - R2$$

R_1 = Sum of ranks for group 1

R_2 = Sum of ranks for group 2

Again, variables needed to be determined significant, p value less than 0.05, by both test in order to be included in the prediction models. The three significant variables are the duration of the surgery, the total amount of blood lost by the patient during the surgery, and the number of admissions the patient had in the year prior to their surgery.

The results of the Student's t-Test and Wilcoxon Rank Sum Test are summarized in table 6.

Table 6 - Univariate Analysis for Continuous Variables

Continuous Variables	Student's t-Test Results	Wilcoxon Rank-Sum Test Results	Significant at 0.05?
Number of Surgeries	0.195	0.02198	No
Days Between Surgeries	0.5717	0.3349	No
Time Between Patient Surgery Start	0.04243	0.1882	No
Duration of Surgery	0.0007442	0.0002831	Yes
Time Between Chemo and Last Surgery	0.2039	0.007578	No
Preglucose	0.6312	0.9685	No
Postglucose	0.1776	0.6485	No
Albumin	0.2494	0.9023	No

HGB	0.2144	0.1445	No
Total Blood Loss	0.02956	0.001848	Yes
Admissions	0.01782	0.001669	Yes
BMI	0.4798	0.5097	No
WBC	0.3645	0.4538	No
Age	0.7226	0.5858	No
Median Income	0.6502	0.3361	No

3.3.2 Boruta Feature Selection Algorithm

Due to the limited nature of using only univariate analysis as a feature selection method it comes as no surprise that the model's predictive performance is lackluster. Therefore, in an effort to improve the performance of the models a more robust feature selection method must be implemented. More specifically a feature selection method that accounts for interaction between the variables rather than treating the variables individually. In order to achieve this there are three main methods of feature selection, filters, wrappers, and embedded methods (Kaushik et al., 2017). These three methods

both work to obtain the same goal of selecting the features that are most relevant to what is being predicted. The methods however do vary in the process in which this goal is achieved. Filter methods are the easiest to apply in that they require the least computational demand. They work by calculating a statistical measure such as Chi squared test in order to assign a score to each feature. Based on the assigned score the features are ranked and determined whether they will be retained in the model, shown in figure 2. Since the scores are generally determined by univariate methods each feature will only be considered independently (Machine Learning Mastery, 2016). This can result in redundant features being selected, which is the opposite of the desired effect of feature selection. As this form of feature selection is similar to the univariate analysis performed in section 3.3.1, it was not considered for the purposes of this research. The risk of failing to select the most important features is too great with a filter approach.

Figure 2 - Filter Feature Selection

The next feature selection method falls under the category of wrapper methods. Wrapper methods are far more computationally demanding than filter methods since they consider many different feature combinations. The performance of these combinations is compared to one another through the use of a prediction model (Kaushik et al., 2017). The combination of features is assigned a score based on the predictive models accuracy and those features included in the model with the highest score are retained as the most important features (Machine Learning Mastery, 2016). Wrapper methods are extremely robust and are considered to find the most important features due to the use of cross validation along with many wrapper algorithms, shown

in figure 3. These algorithms include random-hill climbing algorithm, forward passes and backward passes to select features.

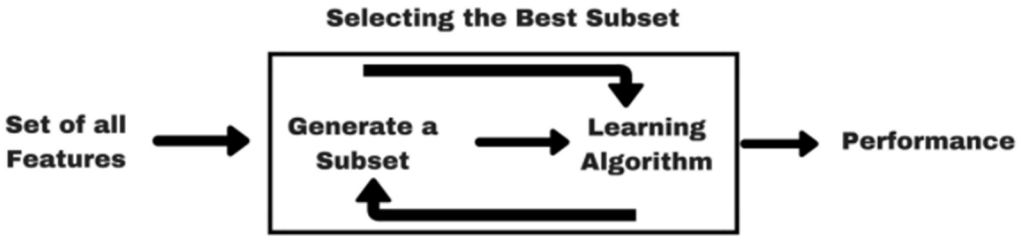

Figure 3 - Wrapper Feature Selection

The third feature selection method is embedded methods. These methods aim to combine the advantages of both filter and wrapper methods. These methods work by learning which features contribute to the highest accuracy while the model is being created, shown in figure 4. In other words, embedded methods perform classification and feature selection at the same time (Kaushik et al., 2017). These methods are most commonly implemented using regularization methods. In regularization methods, additional constraints are introduced to the optimization of the predictive model. These constraints introduce bias in the model to focus on choosing fewer constraints.

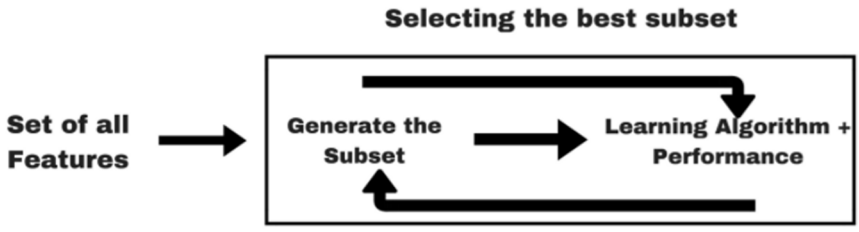

Figure 4 - Embedded Feature Selection

For the purposes of this research a wrapper method was implemented due to its robustness, better performance with chosen features, and availability of algorithms in RStudio. Specifically, the Boruta algorithm, a wrapper method based on the Random

Forest model was utilized for feature selection (Dutta et al., 2016). This algorithm identifies the most important features through a series of 4 steps. First shadow features are created in order to add randomness to the given dataset. These shadow features are created through shuffled copies of all the features in the dataset. Next a Random Forest classifier is trained on the dataset including the shadow features. A feature importance measure is then applied in order to evaluate the importance of each feature. The measure utilized is Mean Decrease Accuracy where a higher value signifies importance. Third, during each iteration it is determined if all of the features are better than the best of its shadow features. This is achieved by comparing the Z score of each feature and comparing it to the maximum Z score of the best shadow feature. Features that are determined to be highly unimportant are then removed from the dataset. The final step is for the algorithm to determine to stop. This occurs when either all features have been confirmed or rejected, or more commonly after a specified number of runs have been completed (Dutta et al., 2016). For the purposes of this research the specified number of runs was set as 1,000.

When the Boruta algorithm is implemented in RStudio, the resulting output is figure 5. The chart includes all tested features rank in order of importance, along with the minimum, mean, and maximum shadow variables. From the chart, it is concluded that 5 features were determined to be important, an importance larger than the max shadow variable which is represented by shadowMax in figure 5. These features, in order of importance, are if the patient had a prior bowel resection, the duration of the surgery, the patient's BMI at the time of the surgery, the patient's wound class, and the patient's cancer category. These are the five variables to be included in the creation of the seven prediction models.

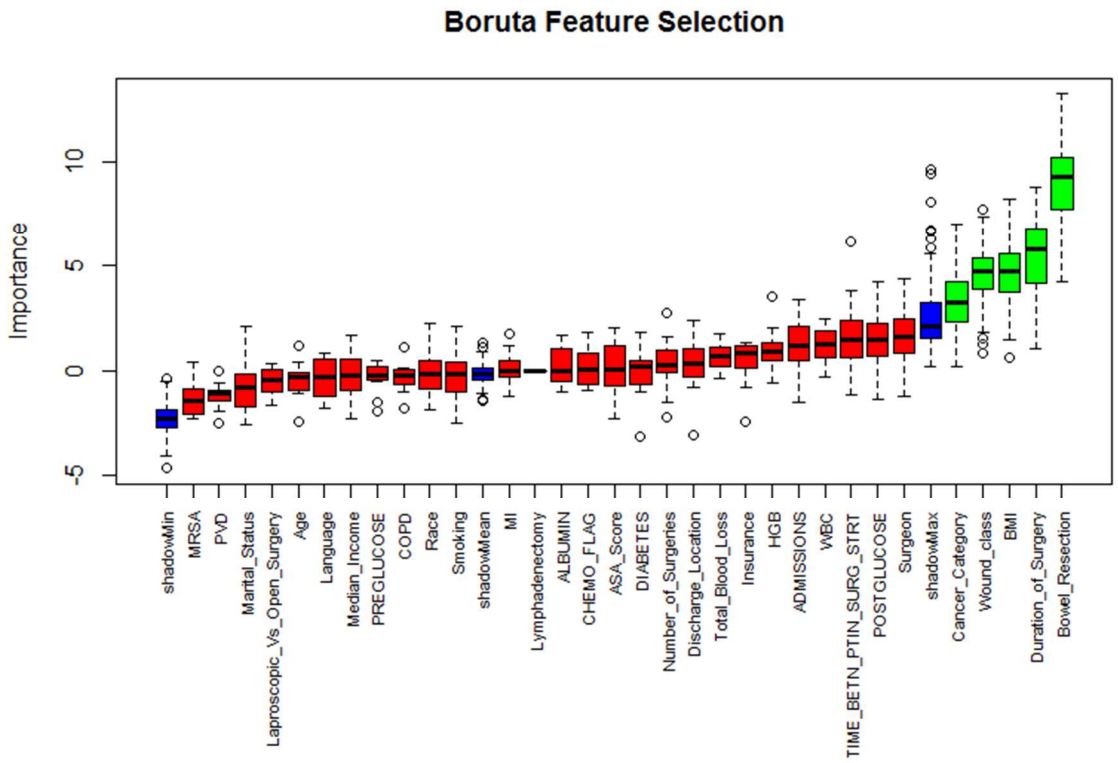

Figure 5 - Boruta Feature Selection

However, before the models can be created it is important to check the correlation between the variables. Collinearity or multicollinearity can result in misinterpreted data or erroneous results. Each feature is included specifically to increase the accuracy of the predictions being made. If features are highly correlated, features that are not statistically significant when considered independently may appear to be significant when considered in conjunction with a highly correlated variable (Tu et al., 2005). This is known to result in more frequent Type I errors or false positive results. Alternatively, features may not appear statistically significant due to the wide confidence intervals associated with high correlation. This is known to increase the rate of Type II errors or false negative results. Thus, to avoid an increase in both Type I and II errors the correlations must be analyzed between the identified significant features. Those with very high correlations, above 0.5, will have to be eliminated. This will be achieved by

keeping the variable that was ranked more important by the Boruta algorithm, so long as the other variables correlations do not increase. The correlation plot in figure 6 shows the correlation between the five identified significant variables. Since no variables have a high correlation, above 0.5, it is a fair assumption that all five variables can be retained for use in the prediction models (Statistics Solutions, 2018). The performance of the models, as discussed in sections 4.1.4 & 4.2.3, were significantly improved when using the features identified as significant by the Boruta algorithm when compared to the features identified as significant by univariate analysis alone. As such, the prediction models and associated results displayed in this research are developed and based upon the five variables identified as significant by the Boruta algorithm.

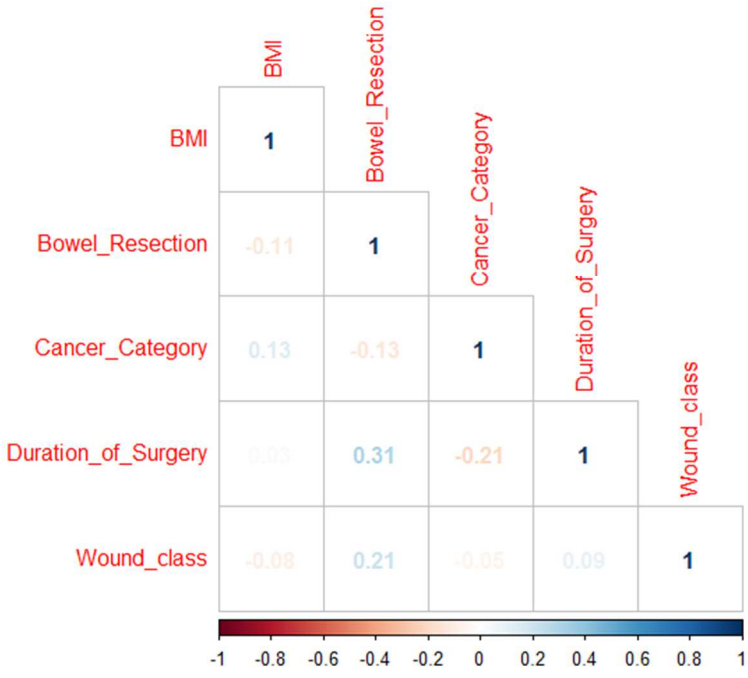

Figure 6 - Correlation Among Significant Features

3.4 Predicting Gynecological Surgical Site Infection

Section 3.4.1 discusses the methodology used to address the imbalance between classes. Sections 3.4.2 to 3.4.8 outline data mining techniques implemented for the prediction of surgical site infection following gynecological surgery.

3.4.1 Class Balancing

Before data mining techniques can be implemented it is important to understand the distribution of the classes. In fact, most datasets have a different number of cases in each class, but small differences are not significant. It is when there is a vastly imbalanced distribution between the cases that an intervention needs to be made. This is to avoid the “accuracy paradox” that is inherent in datasets with unbalanced classes (Brownlee, 2016). This accuracy paradox occurs when you achieve very high accuracy with a prediction model, but it is only reflecting the distribution of the classes. Using this research as an example, prediction models are able to achieve 93.94% accuracy, which is very high, just by predicting all patients as not developing an SSI. While this is a very accurate model, it is not a good model at all as the more important and rare class, those who did develop an SSI (6.06%), were not taken into consideration. This occurs because many models are biased towards only achieving high accuracy. Since there would be no purpose to implement a model that behaves as such it is important that the severely unbalanced classes are addressed prior to application of data mining techniques.

There are four main algorithms that aim to deal with balancing of classes; oversampling the minority class, undersampling the majority class, Synthetic Minority Over-Sampling Technique (SMOTE), and Random Over Sampling Exercises (ROSE). Over-sampling the minority class involves randomly duplicating entries in the minority until the two classes are balanced. This is more beneficial in smaller datasets where eliminating entries would have a significant impact on the predictive performance

(Analytics Vidhya Team, 2016). Undersampling is similar to over-sampling except that it deals with the majority class. Undersampling randomly removes entries from the majority class until the two classes are balanced. This is more beneficial in larger datasets where removing entries will not have as significant of an impact on the performance of the model. These two methods are more basic in dealing with unbalanced classes and are prone to overfitting in the case of oversampling, and information loss in the case of undersampling. Therefore, there are two hybrid methods developed in order to overcome the class imbalance problem through synthetic data generation.

The first method, SMOTE, works by looking at the difference between a feature and its nearest neighbor. A random data point is then placed in between the two features (Analytics Vidhya Team, 2016). For the purpose of this research the ROSE class balancing technique was implemented. Developed by Menardi and Torelli in 2014, ROSE works by utilizing a smoothed bootstrap approach to generate synthetic data points from the classes with emphasis on the minority class (Lunardon, 2013). The below steps outline the steps for generating a single synthetic data point.

1. Select $y^* = Y_j$ with probability π_j **(7)**
2. Select $(x_i , y_i) \in T_n$, such that $y_i = y^*$, with probability $1\ n_j$
3. Sample x^* from $KH_j(\cdot, x_i)$, with KH_j a probability distribution centered at x_i and covariance matrix H_j

y^* = Synthetic generated y coordinate

T_n = Training set of size n

(x_i , y_i) = generic row in training set T_n

x^* = Synthetic generated x coordinate

In other words, an observation from either class is taken from the training set T_n and a new data point (x^*, y^*) is generated in the neighborhood of the original observation. Steps one through three are then repeated m times, number of observations in the training dataset, in order to create a new synthetic training set T_m^* (Lunardon, 2013). By synthetically generating data points the pitfalls from under or oversampling can be avoided resulting in the mitigation of information loss and overfitting while improving the predictive performance of the developed models.

3.4.2 Logistic Regression

Logistic Regression is among the most popular data mining techniques used in predicting individual surgical site infection, as well other predictions. It is a form of regression analysis that is used to determine a relationship between a dichotomous dependent variable and one or more categorical or continuous independent variables (CMU, 2013). Logistic regression works through the use of the logit function shown in equation 8, to create a linear function of x (CMU, 2013).

$$log\frac{p(x)}{1-p(x)} = \beta_0 + x * \beta \quad \textbf{(8)}$$

$log\frac{p(x)}{1-p(x)}$ = logit function

p(x) = probability of event occurring

β_0= Y intercept

x = Input variable value

β = coefficient of input variable

Consequently, the probability that an event will occur is determined using equation 9. The probability is calculated based upon the inputs from the associated predictors (CMU, 2013).

$$p(x) = \frac{e^{\beta 0 + x * \beta}}{1 + e^{\beta 0 + x * \beta}} = \frac{1}{1 + e^{-(\beta 0 + x * \beta)}} \quad \textbf{(9)}$$

The coefficient estimates of β are fitted through the use of a Maximum Likelihood Estimator (MLE). Through the use of a MLE the overall error of the estimates is reduced. For the purposes of this research, Logistic Regression was applied to the features identified as important by the Boruta algorithm. The results are shown in equation 10.

$$log\frac{p(x)}{1-p(x)} = 2.1162 + 0.5108 * \text{Wound Class} + 2.3208 * \text{Duration of Surgery} - 0.4248 * \text{BMI} + 1.1479 * \text{Bowel Resection} - 0.2447 * \text{Cancer Category} \quad \textbf{(10)}$$

3.4.3 Naive Bayes

Naive Bayes is another popular classification technique and relies on the application of Bayes' Theorem. The algorithm provides a way to calculate the probability of a class given the predictors, which is referred to as the posterior probability (Sayad, 2010). Within the classifier lies the assumption that the effect of a predictor's value on a class is independent of the other predictors, which is referred to as class conditional independence (Sayad, 2010). Additionally, the probability of a class, known as prior probability, and the prior probability of a predictor are used to calculate the posterior probability. Prior probability is based on the associated percentage of the classes. The prior probability of a predictor is based on the associated percentage of levels within a predictor. The equation for calculating posterior probability is shown in equation 11.

$$P(c|x) = P(x1|c) * P(x2|c) *\ldots* P(xn|c) * P(c) \quad \textbf{(11)}$$

P(c|x) = Posterior probability

P(c) = Prior probability

P(x|c) = Probability of predictor given class.

P(x) = Prior probability of predictor

In order to obtain the best performing Naive Bayes prediction model it is important to tune the model against the performance metric of choice; Receiver Operating Characteristic (ROC). This metric was utilized when tuning each of the individual and ensemble models and is further discussed in section 4.1.2. For the purposes of this research two distribution types were considered for use; Gaussian and Nonparametric due to their availability in RStudio. Additionally, a Laplace Correction was applied in order to address the problem that arises when conditional probabilities are equal to zero. From figure 7 it is evident that the best performing model, retained for use in this research, arises when the distribution type is Nonparametric and the Laplace Correction is two.

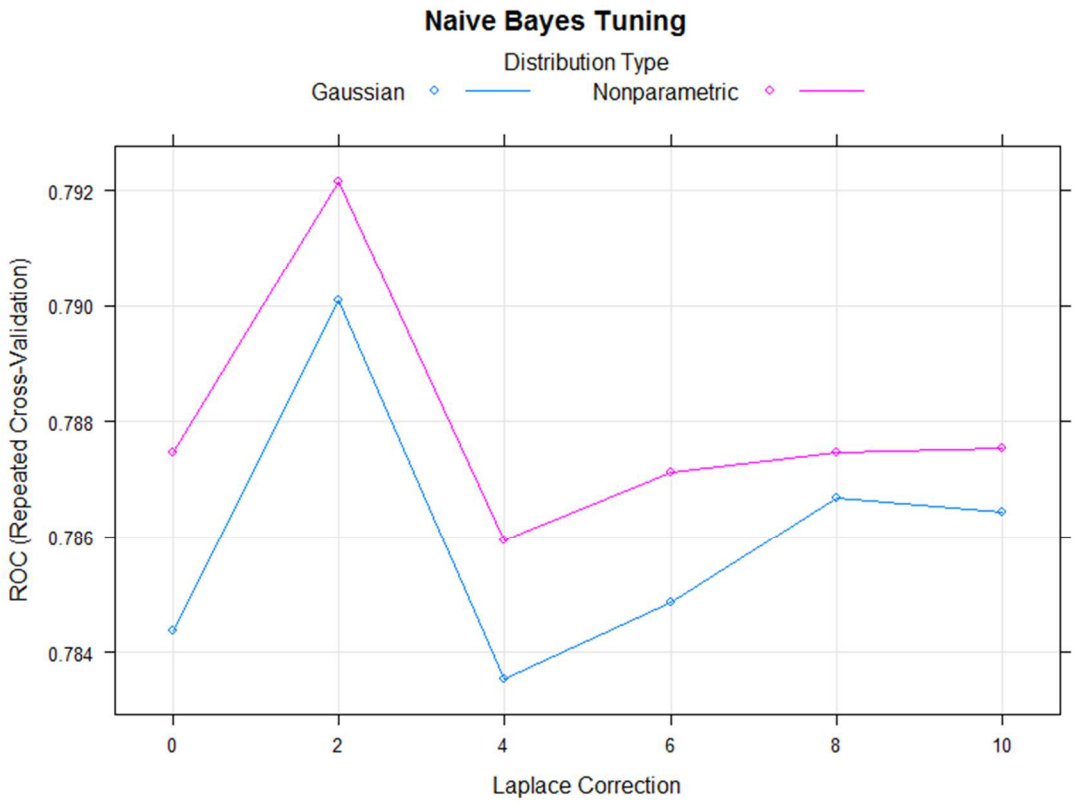

Figure 7 - Naive Bayes Tuning

3.4.4 Recursive Partitioning and Regression Trees

A recursive partitioning and regression trees algorithm is an algorithm that is vital in the implementation of classification or regression trees (CART). Recursive partitioning refers to the iterative process in which a decision tree is constructed by either splitting or not splitting a node on the tree into further nodes (Izenman, 2008). The initial node or root node consists of the entire dataset of predictors. The root node then splits into two daughter nodes based upon whether or not a condition is satisfied. This condition is determined by the observed value of the variable being used as the node as is referred to as the threshold value. This process is repeated until a tree with k splits is constructed. When a node does not split it is referred to as a terminal node and is

assigned a class label. Each node is determined by the predictor that would result in the most information gain which is determined by the entropy function in equation 12.

$$i(T) = -\sum_{K=1}^{K} p(k|T) log p(k|T) \quad \textbf{(12)}$$

K = number of classes

p(k|T) = Estimate of probability that observation x belongs to a class given it is in node T

Once a predictor for use in a node has been chosen for the best split the threshold value for us in the split must be determined. This is again achieved by calculating information gain for each of the possible threshold values for the predictor. The value that results in the largest information gain is retained as the threshold value.

Figure 8 gives a simple example of a decision tree where X_1 and X_2 represent the two predictors. Θ_1, Θ_2, Θ_3, and Θ_4 represent the associated threshold values and T_1, T_2, T_3, T_4, and T_5 represent the terminal nodes for which a class label is applied.

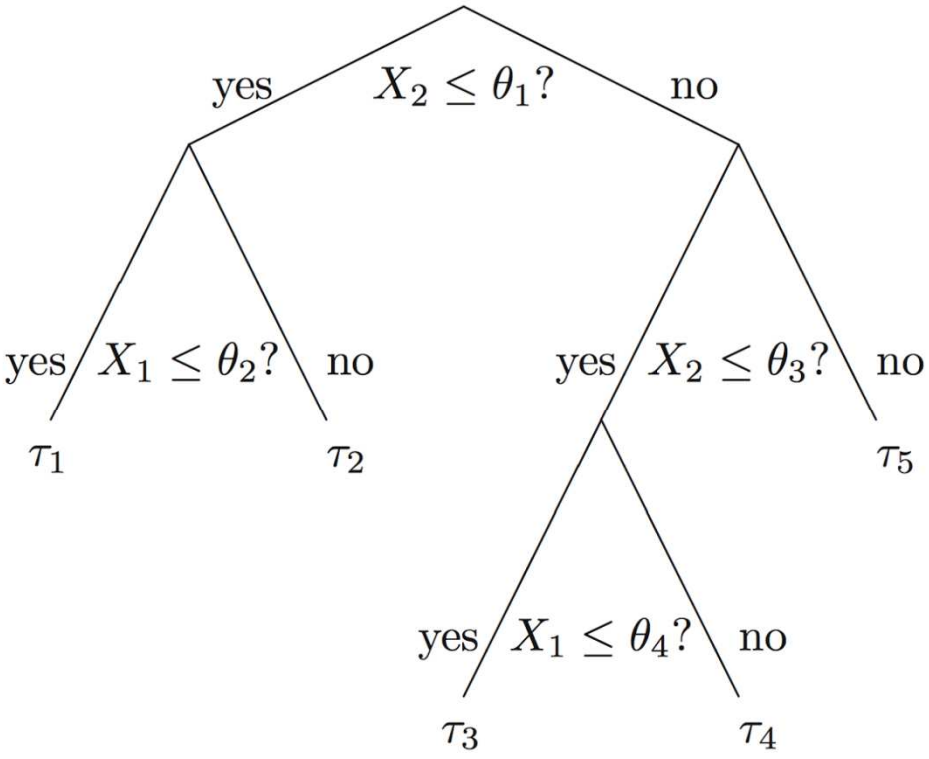

Figure 8 - Decision Tree Example

Again, it is important to tune the model in order to obtain the model that results in the best predictive performance. For the implemented classification tree, the tuning parameter is the complexity parameter; cp. This acts as a threshold value for which if the value of R^2, measure of how close data fit a regression line (Frost, 1970), does not increase by a value of cp during a split, the split is not implemented. This is a way to reduce the computational effort needed. From figure 9 a cp value of 0.032 results in the best performing model in terms of ROC. Thus, this is the model that is retained for further analysis.

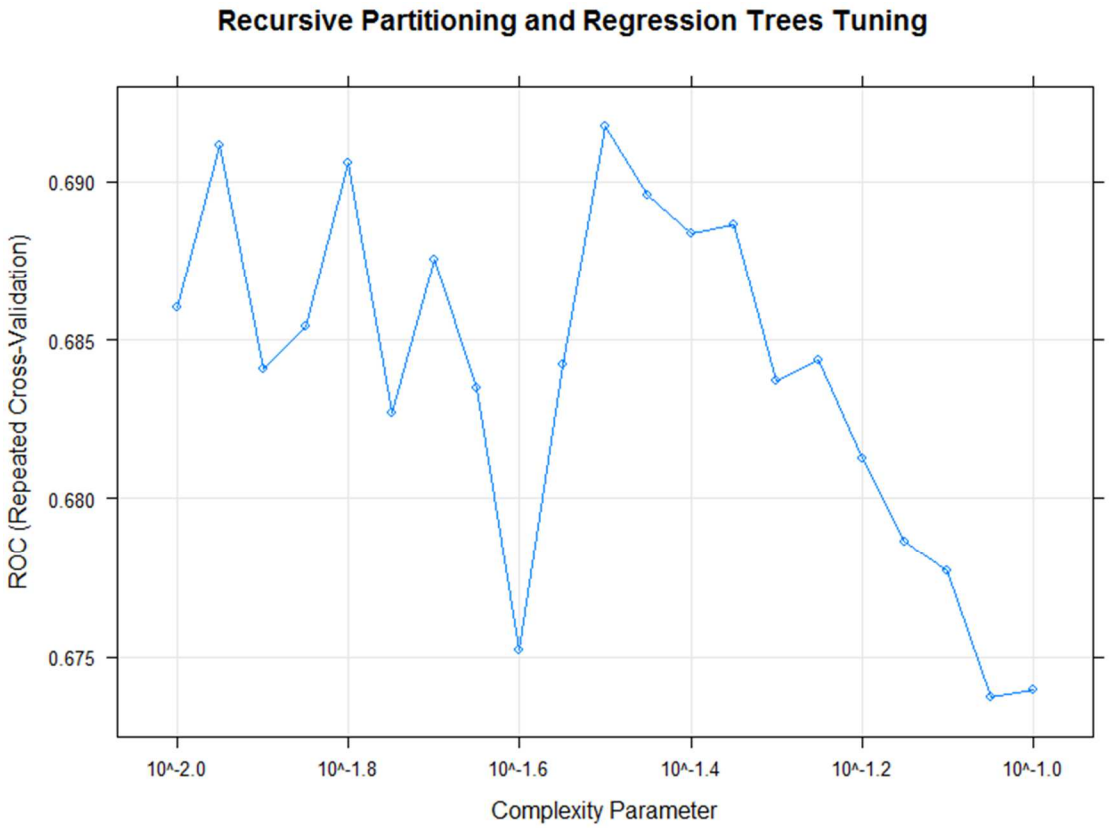

Figure 9 - Recursive Partitioning and Regression Trees Tuning

3.4.5 Random Forest

Random Forest is a type of ensemble learning that groups poorly performing models together in order to form a better performing model. Random Forest is akin to a decision tree algorithm, discussed above in section 3.3.4, but takes it one step further by combining many repetitions of generated trees. First subsets of the data are created by sampling N cases of the data. The subsets should contain a majority proportion of the data. At each node in the decision tree m predictor variables are randomly chosen from all predictor variables. The predictor variable that results in the best split is then retained for a binary split. Performance regarding the best split is based upon variable importance as determined by an embedded objective function relating to the error for each data point. At the next node, the process is repeated by again selecting m predictor variables

from all predictors. A terminal node is reached when a predictor variable in the set m leading to the best split has already been utilized. This process is repeated for any number of trees T (Benyamin, 2012). In regard to this research a majority voting technique was implemented when combining the generated trees to reach a final prediction due to the dependent variable being categorical.

In an effort to produce the best performing random forest model, the model was tuned across many values for the number of randomly selected predictors, m. From figure 10 it is evident that 1 randomly selected predictor results in the best performing model. Subsequently this was the model retained for the purposes of this research.

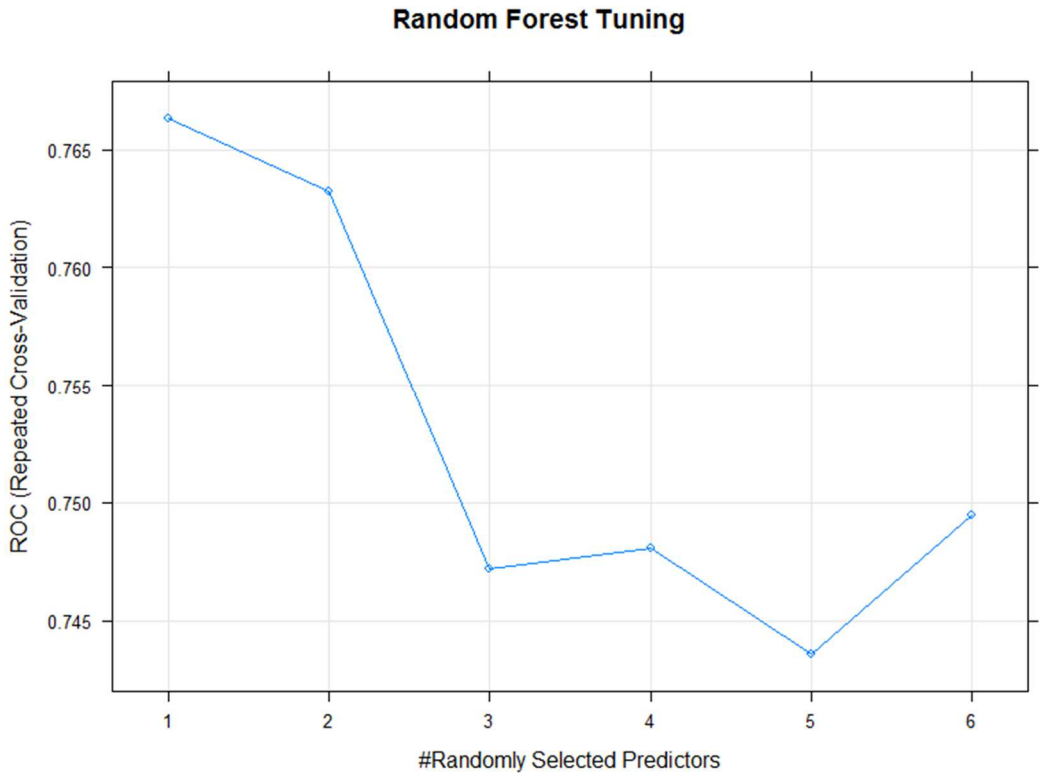

Figure 10 - Random Forest Tuning

3.4.6 Feed Forward Neural Network

Feed Forward Neural Networks are a classification technique that maps inputs to categories in a fashion that is akin to how the human brain operates. This research implements a Feed Forward Neural Network which signifies that no feedback connections exist that feed outputs of the model back into itself (Gupta, 2017). A neural network comprises of an input layer, hidden layers, and an output layer. Each individual node in a layer is referred to as a neuron and contains the basic computations of the neural network (Gupta, 2017). Figure 11 shows the basic structure of a Feed Forward Neural Network.

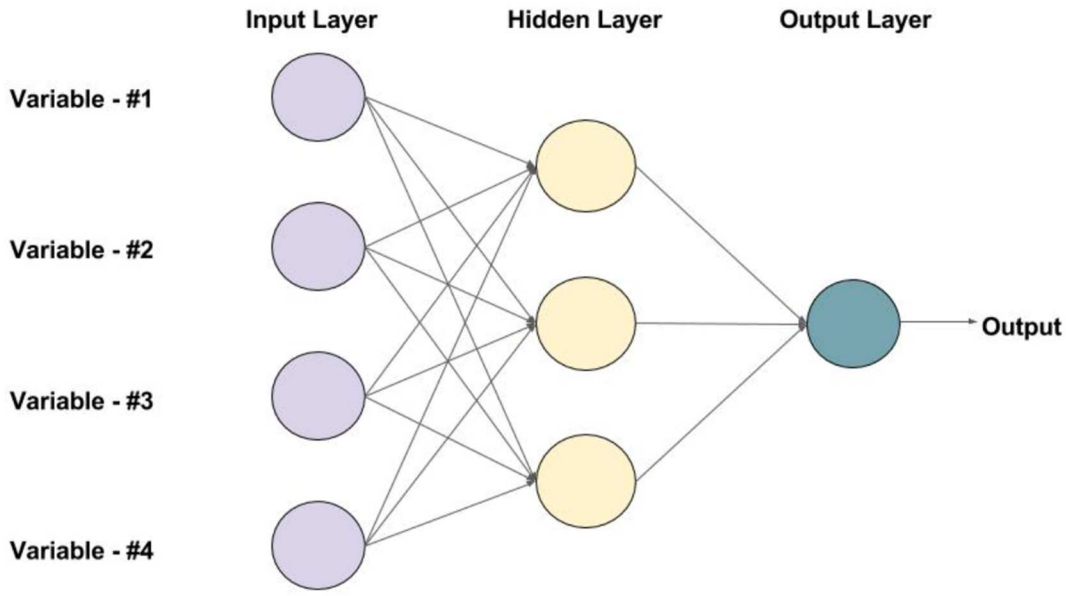

Figure 11 - Feed Forward Neural Network Structure

Figure 12 shows the composition of a neuron in a neural network. X_1 - X_n are the inputs, w_1 - w_n are the inputs corresponding weights, b is the bias, and f is the activation function. In each neuron, first the weighted sum of the inputs is calculated, then an activation function is applied so that the weighted sum is normalized (Gupta, 2017). The

weights associated with the input variables are learned through the training process. For the purposes of this research a sigmoid activation function was implemented. Specifically, a logistic function was implemented based on the equation 13. The purpose of the activation function is to provide a way to make a decision as to which class the input belongs to at the output of each neuron.

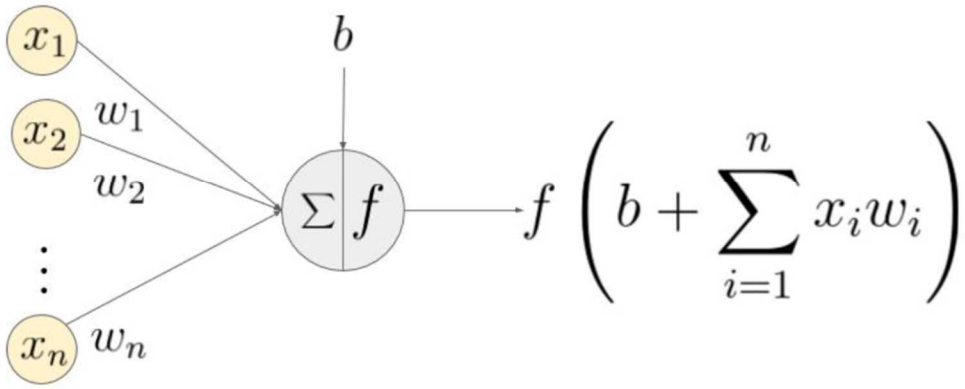

Figure 12 - Composition of a Neuron

$$S(x) = \frac{1}{1+e^{-X}} \quad \textbf{(13)}$$

The predictors of features of the input data are the first layer of the neural network, referred to as the input layer. The output layer is where the predictions are displayed, either 0 or 1 in the case of this research. The hidden layers are a series of functions that are applied to the input (Gupta, 2017). These functions allow the model to detect complex relationships that are not linear in nature. The model then learns by a backpropagation algorithm in which training samples are passed through the neural network and the outputs are compared to the actual outputs (Gupta, 2017). As data is passed through the associated weights in the neurons are updated so that the error is

reduced. In order to determine the neural network that provides the best predictive performance it is important to tune the model for various hidden layers and weight decay. The weight decay is a factor by which the weights are multiplied after each iteration to ensure that the weights do not grow too large (Metacademy, 2012). For the purposes of this research a neural network with one hidden layer and a weight decay of 0.1 offered the best performance, as shown in figure 13.

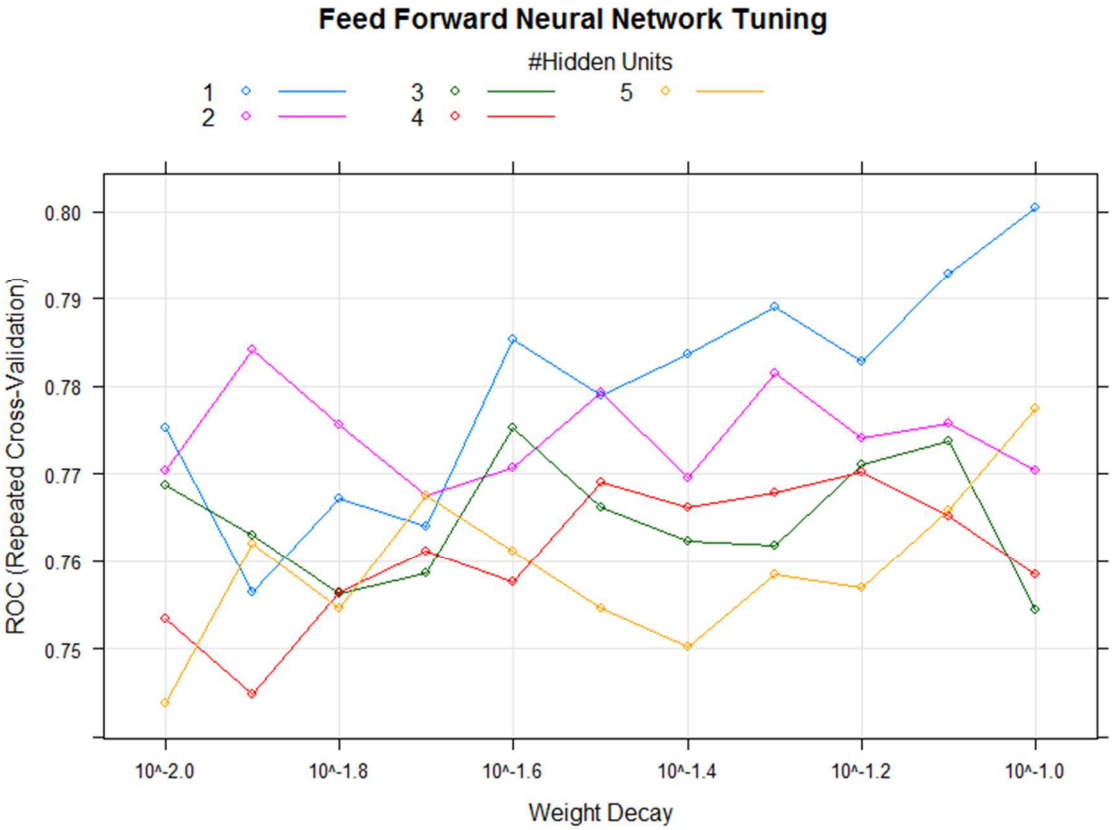

Figure 13 - Feed Forward Neural Network Tuning

3.4.7 K-Nearest Neighbors

K nearest neighbors is another popular classification technique due to its interpretability and fast computation. This is due to the predictions being made in real time as the model does not require any prior learning. K nearest neighbors works by

looking at a new object, determining the K most similar objects to the new object, referred to as neighbors, and then taking a summary of the neighbors (Machine Learning Mastery, 2016). In a classification problem, a new object's class is determined by the most frequent class in the K nearest neighbors. If there is a tie, K is increased by 1 to determine the majority class. Similarities between a new instance and the neighbors are calculated through the use of a distance measure. The most popular distance measure is Euclidean distance, shown equation 14, where p represents a new point and q represents an existing point (Statsoft, 2018).

$$d(p,q) = \sqrt{(p_1\text{-}q_1)^2 + (p_2\text{-}q_2)^2 + \ldots + (p_n\text{-}q_n)^2} \qquad \textbf{(14)}$$

Deciding what K should be is very important in implementing K Nearest Neighbors algorithm. Figure 14 shows the model's performance for varying values of K. The model with the best predictive performance results from when K is equal to 28.

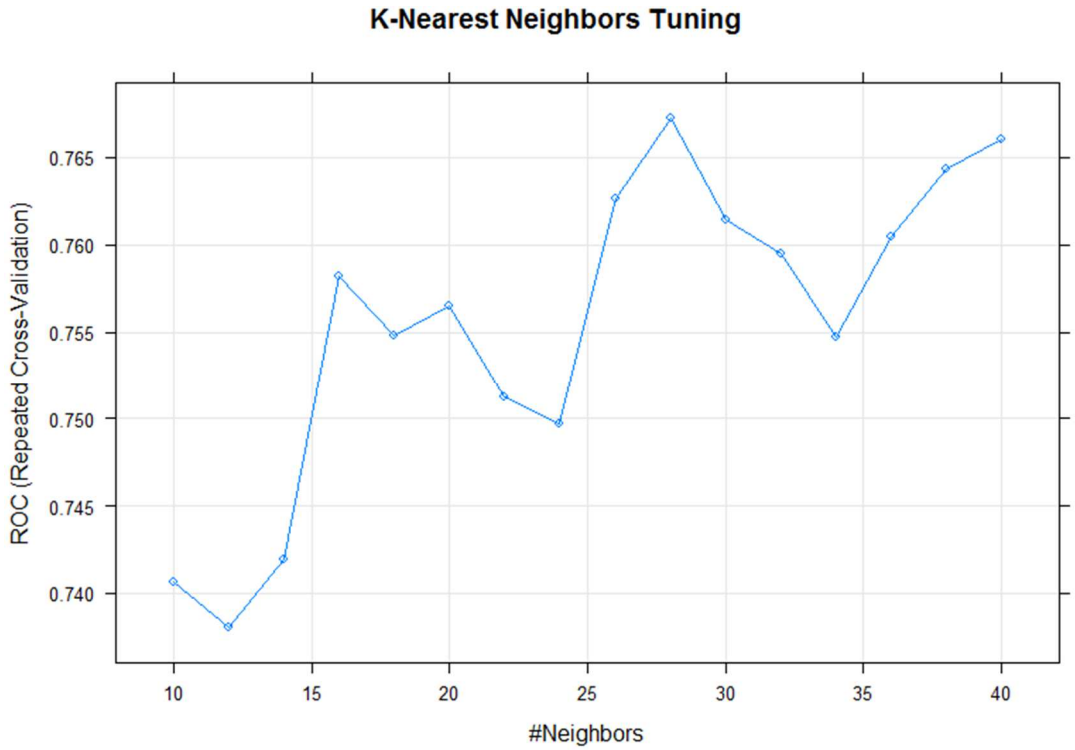

Figure 14 - K-Nearest Neighbors Tuning

3.4.8 Support Vector Machines with Linear Kernel

Support Vector Machines (SVM) is a classification algorithm that finds an optimal hyperplane to categorize new data (OpenCV, 2014). This hyperplane is referred to as the decision boundary which is the separator between the two classes. One side of the decision boundary will be classified as class 1, and the other side of the decision boundary will be classified as class 2. The optimal hyperplane is determined by the line that offers the maximum distance between the nearest element of each case, the support vectors (Stencanella, 2017). Twice this distance is referred to as the margin. For the purposes of this research a linear hyperplane was implemented as it offered the best separation between the two classes. A summary of the optimal hyperplane can be found in figure 15.

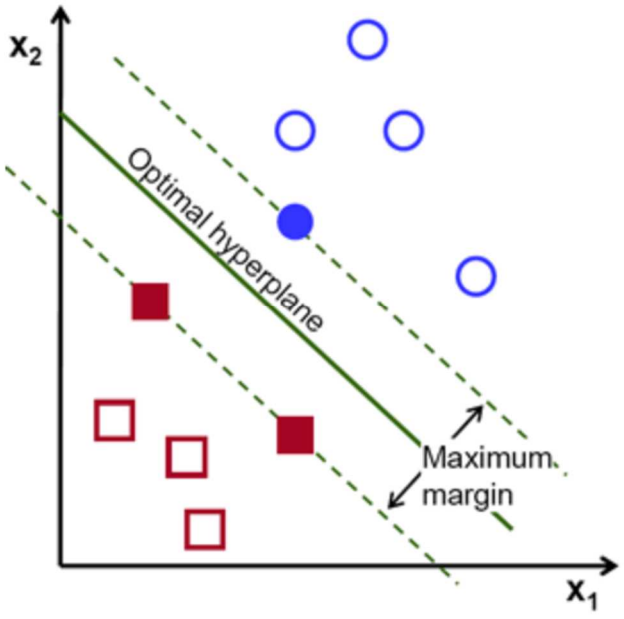

Figure 15 - Optimal Linear Hyperplane

The equation for a hyperplane is shown by equation 15 where β is the weight vector, β_0 is the bias, x the support vector is the training example that is closest to the hyperplane, and T represents the dot product.

$$f(x) = \beta 0 + \beta^T X \quad \textbf{(15)}$$

Rewriting the equation in order to find the optimal hyperplane yields equation 16.

$$|\beta 0 + \beta^T X| = 1 \quad \textbf{(16)}$$

The distance between a point and the hyperplane (β, β_0) is then calculated by equation 17.

$$distance = \frac{|\beta 0 + \beta^T X|}{\|\beta\|} \quad \textbf{(17)}$$

Finally, the margin, M, is calculated by taking multiplying the distance to the support vectors by two, as shown in equation 18. The optimal hyperplane is then determined by maximizing the margin M. This is achieved by minimizing the function L(β) shown by equation 18 where y_i represents the class labels for the training data. The constraints that the function is subject to enable the hyperplane that best classifies the data from the training set x_i.

$$M = \frac{2}{\|\beta\|} \quad \textbf{(18)}$$

$$min_{\beta,\beta 0}, L(\beta) = \frac{1}{2}\|\beta\|^2 \; subject\; to\; y_i(\beta^T x_i + \beta_0) \geq 1 \forall i$$

As mentioned above a SVM with Linear Kernel was implemented in this research due to the reduced computational effort needed along with offering the best separation between the two classes. Kernel functions apply a transformation to map the data into higher dimensional space. The kernel functions then are used to calculate the dot product, or similarity between all pairs of data in this higher dimensional space without

the need to calculate the coordinates (Souza, 2010). In doing such the computational strain is greatly reduced. The equation for a linear kernel is shown by equation 19.

$$k(x, y) = X^T y + c \quad \textbf{(19)}$$

c is an optional constant that is representative of the cost of classification. A small value of c will yield a large margin hyperplane resulting in a larger separation between the support vectors. On the other hand, a large value of c will result in a lower misclassification rate (Fumera & Roli, 2002). As shown in figure 16, a value of c equal to 90 resulted in the model with the best predictive performance.

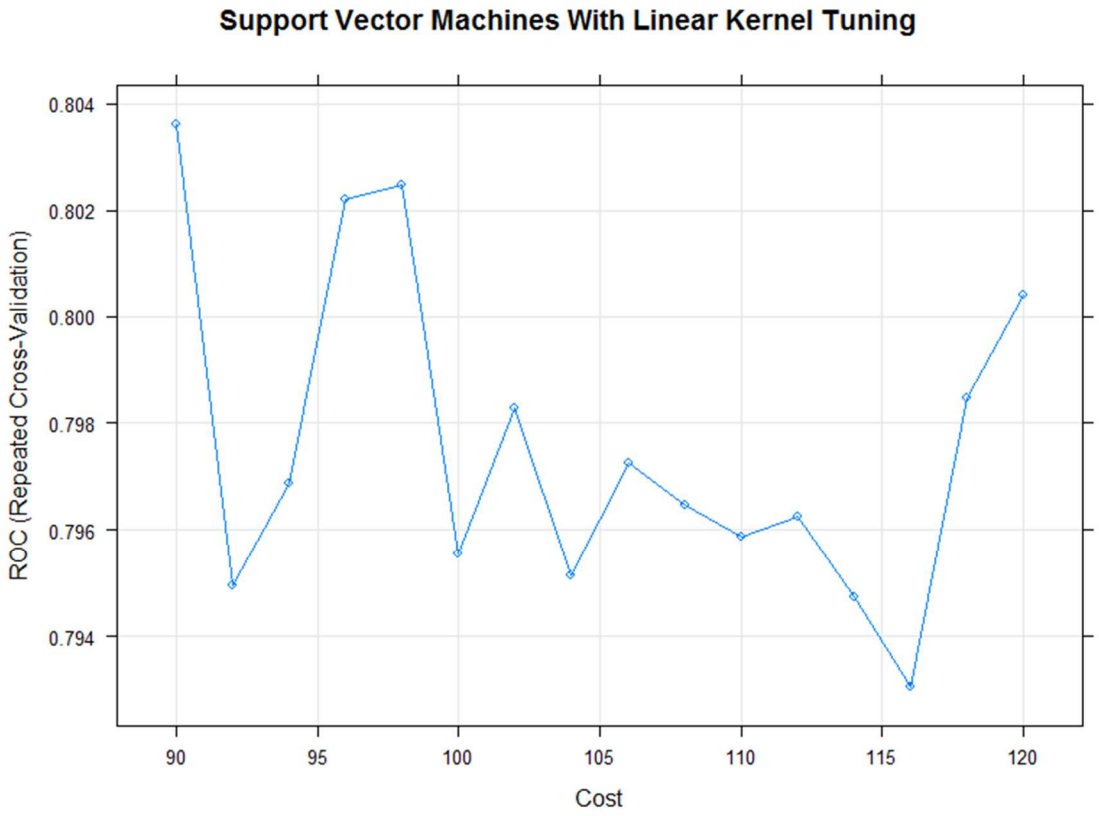

Figure 16 - Support Vector Machines with Linear Kernel Tuning

3.5 Ensemble Learning Model

Sections 3.5.1 to 3.5.5 outline types of ensemble models, the development of the ensemble learning models, the prediction performance metrics of interest, and a comparison of the results.

3.5.1 Types of Ensemble Models

Ensemble modeling is a method in which several weaker performing models, base learners, are combined in order to improve the predictive power of a model. Since these base learners differ in the methods used to classify the data, they each have varied predictions on how the data is to be classified. Ensembling takes into consideration all of the base learners to create a more accurate and robust prediction model that is less likely to be biased (Kaushik, 2017).

When it comes to ensemble models there are three main methods of combining base learners in an effort to improve their predictive performance; bagging, boosting, and stacking. Bagging, or bootstrap aggregation, refers to the process where a single base learner is applied to different training sets. A bootstrap technique is applied to resample the training data in a way that ensures diversity in the smaller training sets. The chosen algorithm is then applied to each training set, and a predicted class is determined. A final prediction is achieved by a majority voting from each of the individual classifiers. Majority voting is when the final prediction is chosen by the class that was predicted most often by the individual classifiers (Kaushik, 2017). Figure 17 shows the structure of the bagging technique where D represents the class prediction by each of the classifiers. It is important to note that bagging focuses on reduction in variance.

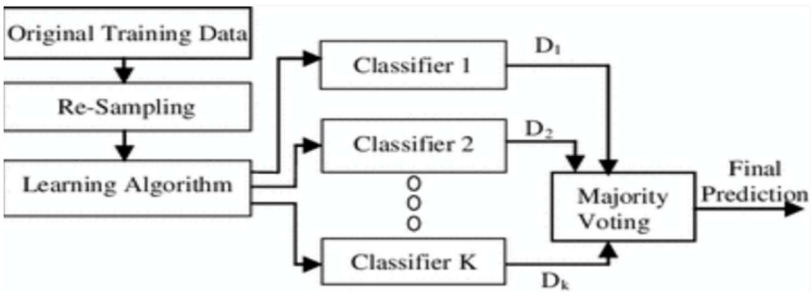

Figure 17 - Bagging Ensemble Technique

Boosting also uses a resampling technique, but it is different than the one used in bagging. A training set is generated based upon its sample distribution (Nagi, 2013). The first classifier is then created on an equally weighted dataset. Next a second training set is created where higher weight is placed on the samples correctly predicted, and lower weight on the samples incorrectly predicted. This process is then iteratively repeated. A final class prediction is then made based upon a weighted linear combination of the classifier outputs. Higher weight is applied to the more accurate classifiers, while a lower weight is applied to the less accurate classifiers. Figure 18 outlines the boosting process where D again refers to the class prediction by its associated classifier. The main focus of the boosting technique is to reduce the bias.

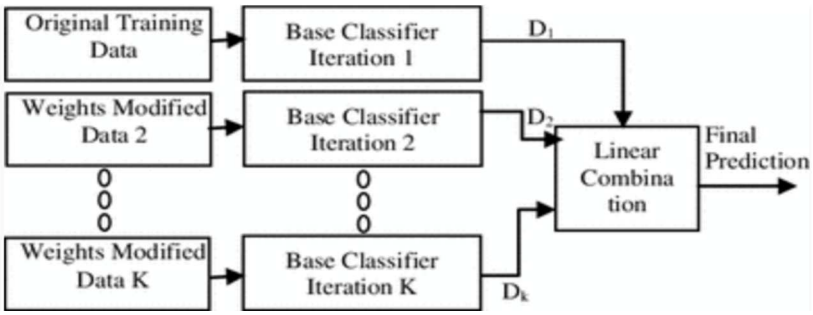

Figure 18 - Boosting Ensemble Technique

The third most popular ensemble technique is stacked generalization or stacking which is broken up into two parts. First base level classifiers are trained on the training

dataset, and the predicted classes are recorded. The outputs of these base level classifiers are then combined into a second dataset. This dataset is then used as the training data for the second level or meta level classifier. The predictions from the meta level classifier are then used as the final predictions. Implementing a stacking technique ensures that the data have properly learned from the training set (Nagi, 2013). If a base level classifier incorrectly learns a part of the training set, it will lead to misclassifications stemming from the incorrectly learned part. The goal of the meta classifier then is to learn this behavior and combine it with the behavior of the other base classifiers to correct the improper learning (Nagi, 2013). In an effort to improve the performance of the stacking ensemble the predictive probabilities from each of the base level classifiers are used instead of the class labels, as suggested by Polikar. Figure 19 shows the how the stacking technique is implemented. For the purposes of this research a stacking ensemble model was implemented based on prominence of its use in popular data science competitions such as Kaggle, as well as its balanced reduction of both bias and variance.

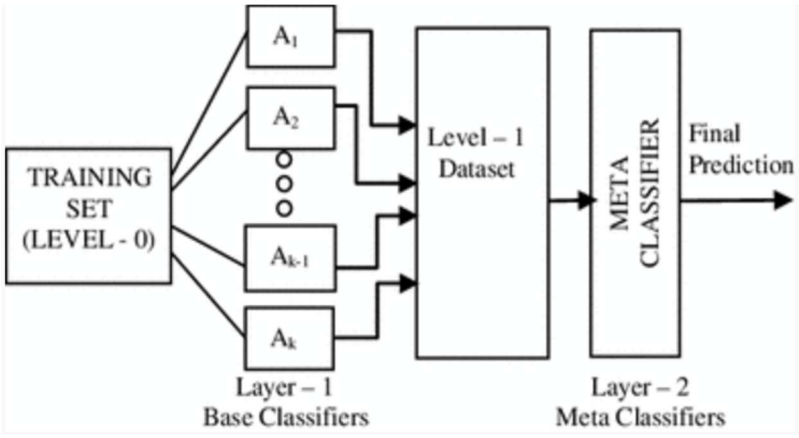

Figure 19 - Stacking Ensemble Technique

3.5.2 Development of Ensemble Learning Model

Once it was decided that a stacked generalization technique was going to be implemented the specific base level classifiers, as well as the specific meta level classifiers needed to be chosen. For the base level classifiers, it is important to choose classifiers that are not too closely related so that the meta level classifier has ample opportunity to learn from the training set. Additionally, the base level classifiers have relatively strong predictive performance, further discussed in section 4.1.2. For these reasons Support Vector Machines with Linear Kernel, K-Nearest Neighbors, and Logistic Regression were chosen as the base level classifiers. It is important to note that not every individual prediction model was included due to some similarities between the techniques and the associated risk that the stacking algorithm would learn the overlapping pitfalls of these methods rather than the areas of strong performance.

Before the stacking algorithm could be implemented on the chosen base level classifiers it is important to make sure the prediction results are not highly correlated due to the same reasons as discussed in section 3.3. Figure 20 shows the associated correlation matrix for all of the base level classifiers, where Logistic Regression is represented by glm, K-Nearest Neighbors by knn, Naive Bayes by nb, Feed Forward Neural Network by nnet, Random Forest by rf, Recursive Partitioning and Regression Trees by rpart, and Support Vector Machines with Linear Kernel by svmLinear. Since no probabilities are highly correlated, greater than 0.5 we do not need to remove the predictive probabilities from any data mining technique. This allows the stacking algorithm to be implemented on the chosen base level classifiers.

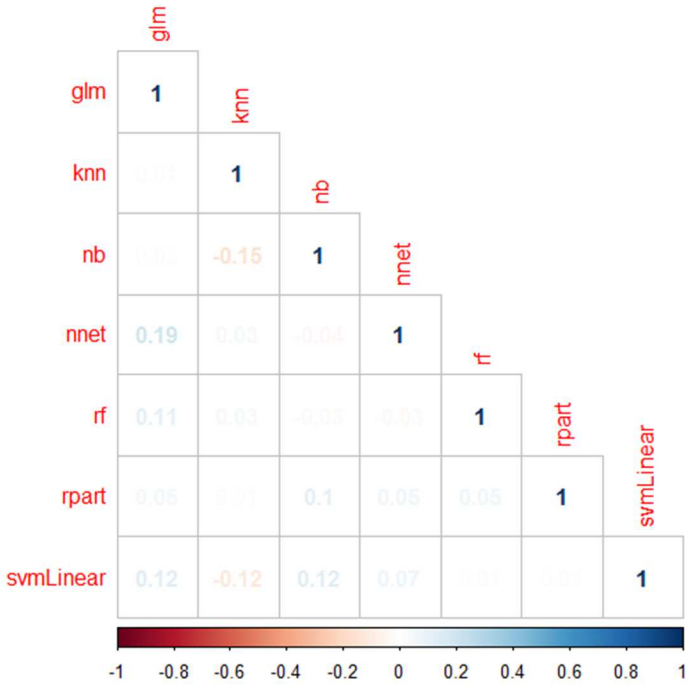

Figure 20 - Correlation Amongst Individual Predictions

Once the base level classifiers were chosen their predictive probabilities, from the training set, were utilized as inputs to train the meta level classifier. The meta level classifier then utilized the testing set predictive probabilities as the inputs in the trained meta level classifier in order to determine predictive performance. In order to compare performance of the stacking ensemble models three meta level classifiers were chosen; Support Vector Machines with Linear Kernel due to its strong performance for all measures as an individual model, K-Nearest Neighbors again due to its strong predictive performance as an individual model, and Logistic Regression due to its ease of implementation and interpretability.

In an effort to improve the performance of the stacking ensemble a series of weights were applied to the individual base level classifiers. Specifically, extra emphasis was placed on Support Vector Machines with Linear Kernel due to the emphasis on sensitivity in the development of the prediction model. Further discussion on the impact

of the performance metrics is discussed in section 4.1.2. Therefore, the predictive probabilities from Support Vector Machines with Linear Kernel received 50% of the weight while K-Nearest Neighbors and Feed Forward Neural Network received 25% of the weight each due to their balanced performance.

3.5.3 Support Vector Machines with Linear Kernel Ensemble

The same classification algorithm, as discussed in section 3.4.8, was applied to the combined predictive probabilities of the individual Support Vector Machines with Linear Kernel, K-Nearest Neighbors, and Feed Forward Neural Network. Again, in an effort to improve the performance of the model the value of the cost of classification, c, was varied. From diagram 21 a value of c=70 yields the best performance in terms of ROC.

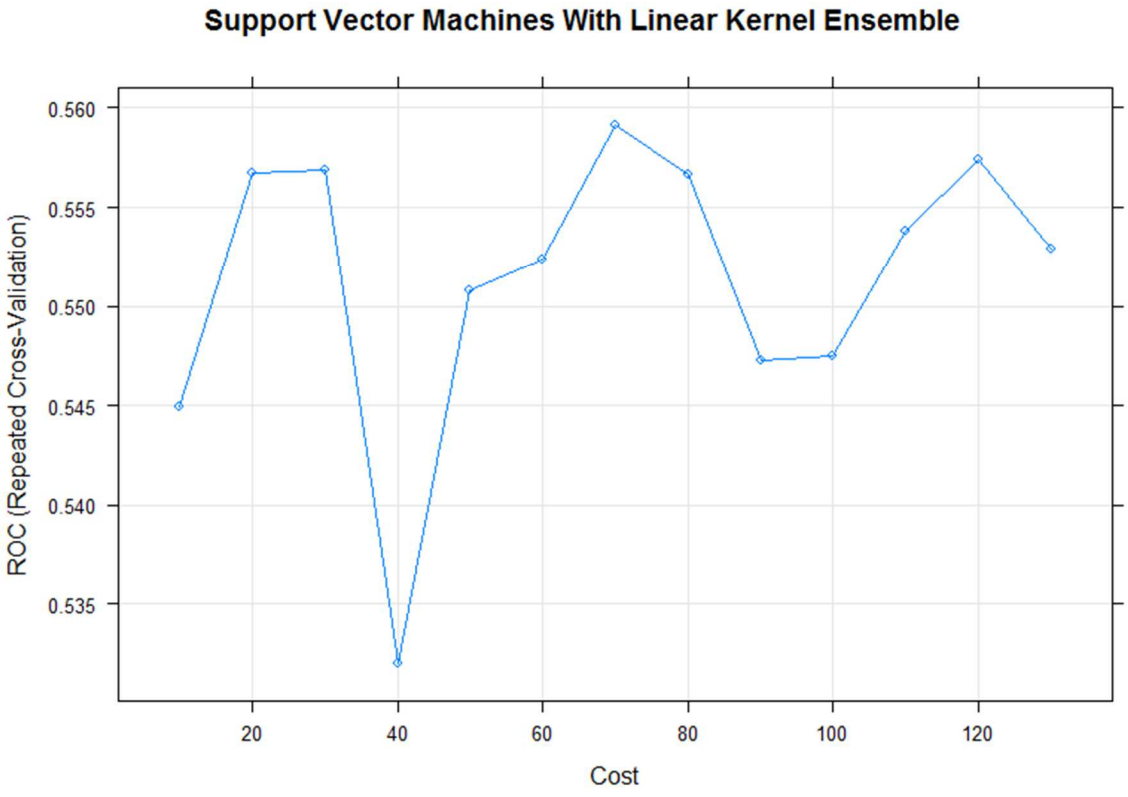

Figure 21 - Support Vector Machines with Linear Kernel Ensemble Tuning

3.5.4 Logistic Regression Ensemble

The same generic algorithm as discussed in section 3.4.2 was applied again to the base level classifiers predictive probabilities outputs. Equation 20 shows the results of using Logistic Regression as a meta level classifier.

$$log\frac{p(x)}{1-p(x)} = 0.1418 - 1.6077 * KNN - 0.7232 * NNET + 1.0898 * SVMLinear \quad \textbf{(20)}$$

3.5.5 K-Nearest Neighbors Ensemble

The third meta level classifier implemented was K-Nearest Neighbors. Utilizing the algorithm from section 3.4.7 above a stacking ensemble was created. A very important part of implementing K-Nearest Neighbors is choosing an appropriate value of k. From figure 22, k=28 yields the best results in terms of ROC.

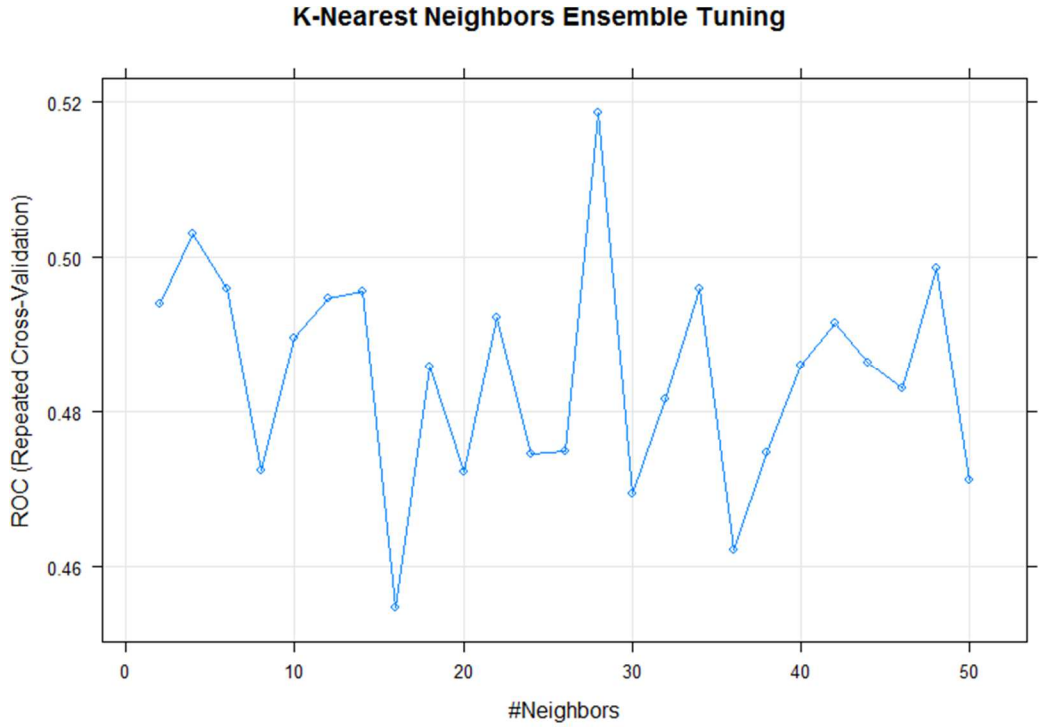

Figure 22 - K-Nearest Neighbors Ensemble Tuning

4. Results

4.1 Gynecological Surgical Site Infection Prediction Results

Section 4.1.1 to section 4.1.4 outline the creation of a training and testing set, performance metrics, a comparison of the performance metrics, and discussion of the individual prediction model results.

4.1.1 Model Training and Testing

In order to determine the effectiveness of the data mining techniques, the algorithms are trained on a portion of the data and then evaluated on a testing set. The purpose of splitting the data in this manner is to determine how the developed models behave on new data that has not been seen before. For the purposes of this research the data were split into 75% training, 521 observations and 25% testing, 172 observations. This same training and testing set was used for both the development of the individual and ensemble prediction models. In order to compare the results of the prediction models in a more robust way, a five fold cross validation, repeated ten times, was implemented.

K fold cross validation is a validation technique that is used to evaluate the performance of machine learning techniques. First the training dataset is randomly divided into k subsets which are referred to as folds. It is important that there is an equal distribution of the class being predicted and the sample sizes remain close across the folds (Kohavi, 1995). During each of the folds, k-1 of the subsets are used as the training dataset which allows the machine learning algorithms to learn the features and behavior

of the dataset. The model is then tested on the fold that was left out in order to determine the performance of the model. This process is then repeated with the withheld fold inserted back into the dataset and the next subset is withheld. This is repeated until every data point is in the test set k times and is a part of a training set k-1 times (Carnegie Mellon Computer Science, 1995). Validating the data in such a manner is more computationally demanding, however it offers lower variance since each data point is used for validation only once. In order to ensure the model was validated the k-fold cross validation algorithm was implemented with 5 folds and repeated 10 times. The results of the folds across the 10 repetitions were averaged together to gain a strong understanding of the model performance. Figure 23 hows the schematic for how a k-fold cross validation is implemented (Esmaeelzadeh et al., 2014).

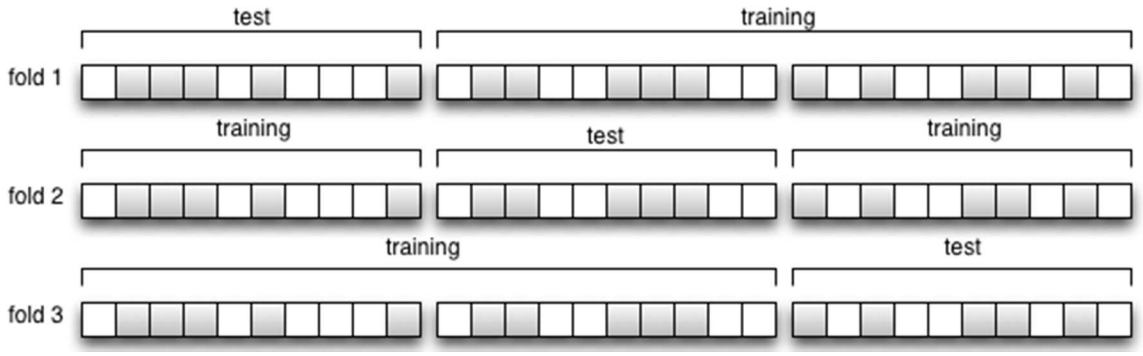

Figure 23 - k-Fold Cross Validation Implementation

4.1.2 Prediction Performance Metrics

Five measures were used to compare the performances of the seven individual data mining techniques; accuracy, sensitivity, specificity, Receiver Operator Characteristic (ROC), and Area under the Receiver Operating Characteristic Curve (AUC). For the first three performance metrics, a confusion matrix must be used in order to perform the calculations and compare the performances. Table 7 shows the

framework for the confusion matrix with regards to this research. In this research, a patient who did not develop an SSI was considered a negative output, and a patient who did develop an SSI was considered a positive output. The outputs were structured this way to increase the emphasis on the patients who did develop an SSI.

There are four sections in a confusion matrix; True Negative (TN), True Positive (TP), False Negative (FN), and False Positive (FP). A true negative indicates that a patient did not develop an SSI and it was predicted that the patient would not develop an SSI. Alternatively, a true positive indicates that a patient was predicted as developing an SSI and did develop an SSI. A false negative represents a patient who developed an SSI but was predicted as not developing an SSI. Finally, a false positive indicates that a patient was predicted as developing an SSI and did not develop an SSI.

Table 7 - Surgical Site Infection Confusion Matrix

True Negative Patient was predicted as not developing an SSI, Patient did not develop SSI	**False Negative** Patient was predicted as not developing an SSI, Patient did develop SSI
False Positive Patient was predicted as developing an SSI, Patient did not develop SSI	**True Positive** Patient was predicted as developing an SSI, Patient did develop SSI

Accuracy, sensitivity, and specificity draw from the confusion matrix in order to be calculated. Their respective equations are shown by equations 21, 22, and 23 where the output is a number between zero and one. Values closer to one are indicative of better performance with one being a perfect score. Accuracy is a measure of how well patients are predicted correctly, both developing an SSI and not developing an SSI (Baratloo et al., 2015). As mentioned in section 3.4.1 accuracy is strongly dependent on the

distribution of the classes. Therefore, it was not considered as the sole performance metric.

Sensitivity is a measure of how well the prediction model is able to determine patients who did develop an SSI as such. This was considered a very important performance metric for the purposes of this research as there is a strong interest in ensuring that patients who developed SSIs are correctly being identified.

On the other side of sensitivity is specificity, which is a measure of how well the prediction model is able to classify the patients that did not develop an SSI as such. Again, this is an important measure since it is detrimental to over predict patients as developing an SSI.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad \textbf{(21)}$$

$$Sensitivity = \frac{TP}{TP + FN} \quad \textbf{(22)}$$

$$Specificity = \frac{TN}{TN + FP} \quad \textbf{(23)}$$

Receiver Operating Characteristic, specifically the ROC curve, is a powerful metric that is used to compare the performance of many different classification techniques. ROC is especially powerful when dealing with unbalanced classes such as the case with this research (Vogler, 2016). In a two class problem, the ROC curve is created by plotting the specificity, decreasing, on the X axis and the sensitivity, increasing, on the Y axis. The points on the plot represent the associated sensitivity and specificity of the models at various thresholds. A threshold is representative of a probabilistic value to distinguish between the classes. Probability outputs larger than the

threshold are classified as one class, and probabilities below the threshold are classified as the other class (Zygmunt, 2013). Additionally, a line is drawn to represent a ROC of 0.5 as this is representative of a random classifier. Models that perform poorer than this are less accurate than random classification and are considered ineffective (Hanley & Mcneil, 1982). A ROC score of 1 corresponds to a perfect classifier and would be represented by a curve that extends to the top left of the plot.

When calculating ROC, the value is determined by calculating the area under the curve or AUC. Therefore, ROC and AUC are used to represent the same metric, but for the purposes of this research they correspond to different datasets. When the metric ROC is referred to it is the area under the ROC curve associated with the performance of the training set. When the metric AUC is referred to it is the area under the ROC curve associated with the performance of the training set.

The ROC curves comparison for the performance of the individual prediction models on the testing dataset is shown in figure 24. The results of the comparison are discussed in section 4.1.3.

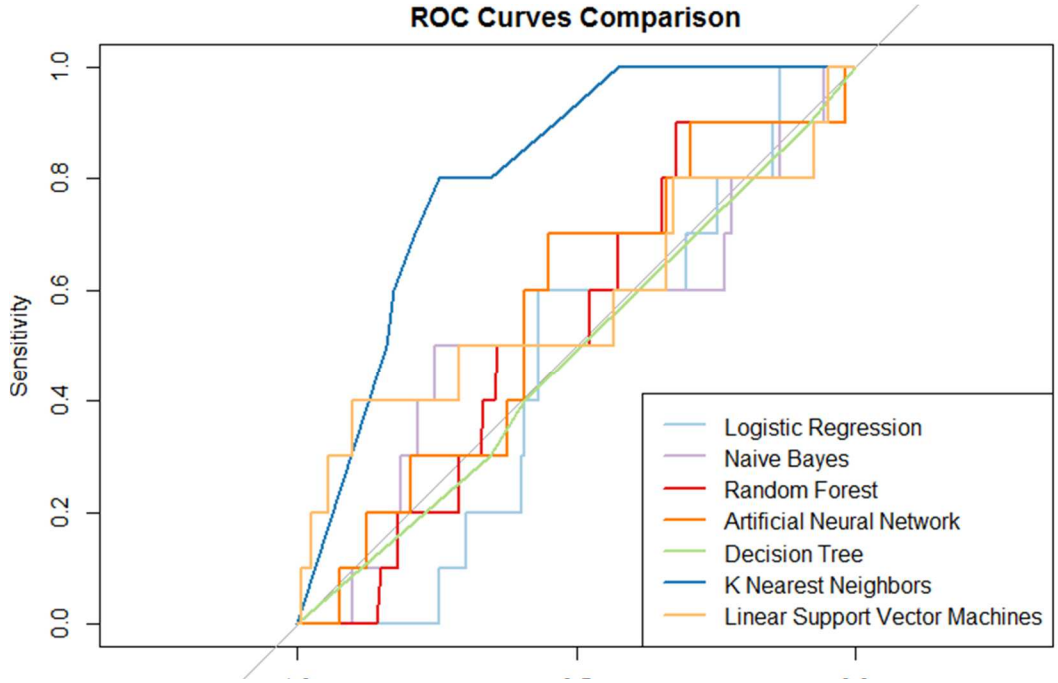

Figure 24 - Individual Predictions ROC Curves Comparison

4.1.3 Individual Prediction Results Comparison

The results from the seven implemented data mining techniques are summarized in table 8. Since a variety of performance metrics were used to assess the predictive performance of the various techniques it is very rare to find one model that has the best performance across all metrics. Therefore, the techniques that performed best for each metric were noted.

In terms of accuracy Random Forest and K-Nearest Neighbors performed the best with scores of 0.7791 and 0.7558 respectively. For sensitivity Support Vector Machines with Linear Kernel had by far the best performance with 0.7. For specificity, Random Forest and K-Nearest Neighbors again performed the best with scores of 0.8086 and 0.7840 respectively. For ROC of the training set, Support Vector Machines with Linear Kernel had the best performance with a score of 0.8036. Finally, for AUC K-Nearest Neighbors has by far the highest performance with a score of 0.8086.

Table 8 - Comparison of Individual Model Performance

Model	Accuracy	Sensitivity	Specificity	ROC	AUC
Logistic Regression	0.6163	0.3	0.6358	0.7575	0.4605
Naïve Bayes	0.657	0.3	0.67901	0.7923	0.5154
Random Forest	0.7791	0.3	0.80864	0.7663	0.5287

Feed Forward Neural Network	0.6977	0.5	0.70988	0.7752	0.5611
Recursive Partitioning and Regression Trees	0.5698	0.3	0.58642	0.6917	0.4827
K Nearest Neighbors	0.7558	0.3	0.78396	0.7672	0.8086
Support Vector Machines with Linear Kernel	0.657	0.7	0.65432	0.8036	0.5747

4.1.4 Individual Prediction Results Discussion

It is difficult to determine which one model had the best overall performance across all the metrics. Of note are Support Vector Machines with Linear Kernel due to its high sensitivity and K-Nearest Neighbors due to its high accuracy, specificity, and AUC. Feed Forward Neural Network did not perform the best in any single category, however when considering the predictive performance, it had high results for all metrics.

4.2 Ensemble Learning Prediction Results

Section 4.2.1 to section 4.2.3 outline the metrics used in the ensemble prediction models, a comparison of those metrics, and a discussion of the ensemble prediction models.

4.2.1 Ensemble Prediction Performance Metrics

The same performance metrics, as mentioned above in section 4.1.2, were used to assess the predictive performance of the ensemble models. Again, a confusion matrix

needed to be created, and the same layout and definitions from section 4.1.2 were used for the ensemble model confusion matrix. Additionally, ROC for the training set and AUC for the testing set were calculated. The associated ROC curve comparison for the testing set of the weighted ensemble models is shown in section 4.2.2.

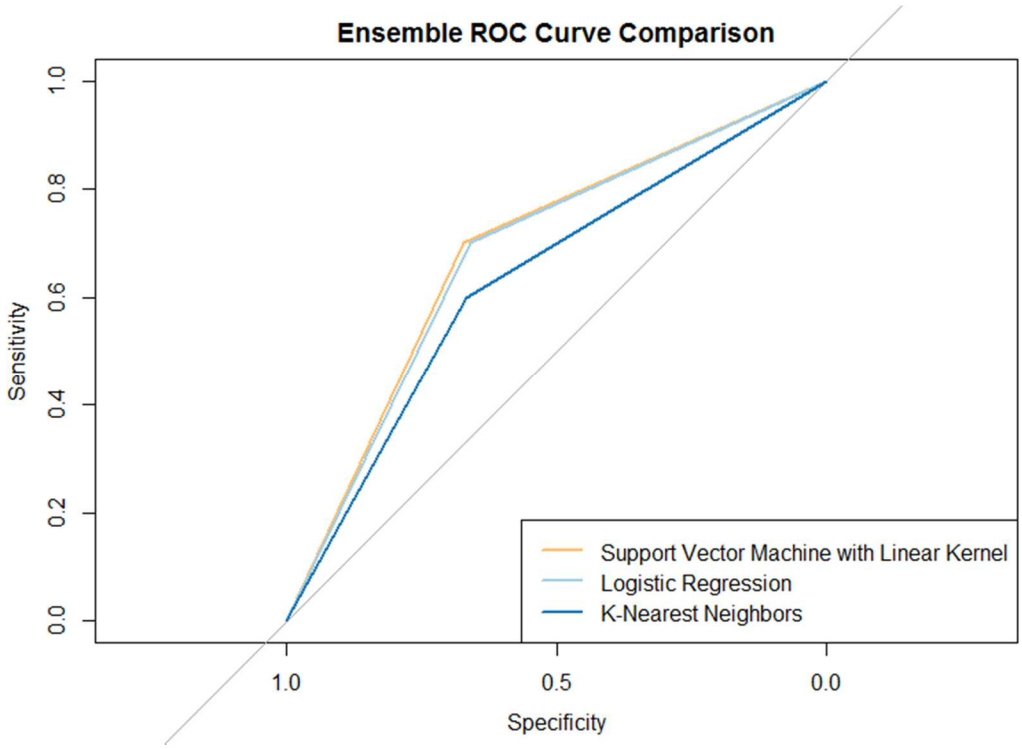

Figure 25 - Weighted Ensemble Models ROC Curve Comparison

4.2.2 Non-Weighted Ensemble Prediction Results Comparison

Table 9 shows the results of the three chosen ensemble stacking models for the identified performance metrics. In terms of accuracy Support Vector Machines with Linear Kernel has the best performance with 0.6628. Support Vector Machines with Linear Kernel, Logistic Regression, and k-Nearest Neighbors have the best performance, with regards to sensitivity, of 0.5. For specificity Support Vector Machines with Linear Kernel again has the best performance with 0.6728. Logistic Regression has the best ROC performance with 0.5008. Finally Support Vector Machines with Linear Kernel has the best AUC performance of 0.5864.

Table 9 – Comparison of Non-Weighted Ensemble Model Performance

Model	Accuracy	Sensitivity	Specificity	ROC	AUC
Support Vector Machines with Linear Kernel Ensemble Model	0.6628	0.5	0.6728	0.4945	0.5864
Logistic Regression Ensemble	0.6105	0.5	0.6173	0.5008	0.5586
K Nearest Neighbors Ensemble	0.657	0.5	0.6667	0.4783	0.5833

4.2.3 Non-Weighted Ensemble Prediction Results Discussion

Based on the performance of the ensemble models, it is evident that Support Vector Machines with Linear Kernel offers the best performance. k-Nearest Neighbors is not far behind Support Vector Machines with Linear Kernel in terms of performance, and Logistic Regression has the worst results. However, none of these ensemble models are able to achieve the sensitivity, 0.7, that the base level classifier of Support Vector Machines with Linear Kernel was able to. Thus, this research investigates the use of weighting to increase the performance of the stacked generalization models with specific emphasis on the Support Vector Machines with Linear Kernel ensemble model.

Figure 26 shows the results of varying the weight on Support Vector Machines with Linear Kernel during the stacking process. Specifically, the weight was varied from 0.1 to 0.9, while the remaining delta was split equally amongst Feed Forward Neural Network and k-Nearest Neighbors. Figure 26 depicts the resulting change in performance for the Support Vector Machines with Linear Kernel ensemble model with

regards to the varying weights. Form the figure it is evident that the best performance for all four performance metrics results from 50% of the weight on Support Vector Machines with Linear kernel, and 25% on both Feed Forward Neural Network and k-Nearest Neighbors. Thus, these weights were retained for the stacked generalization models.

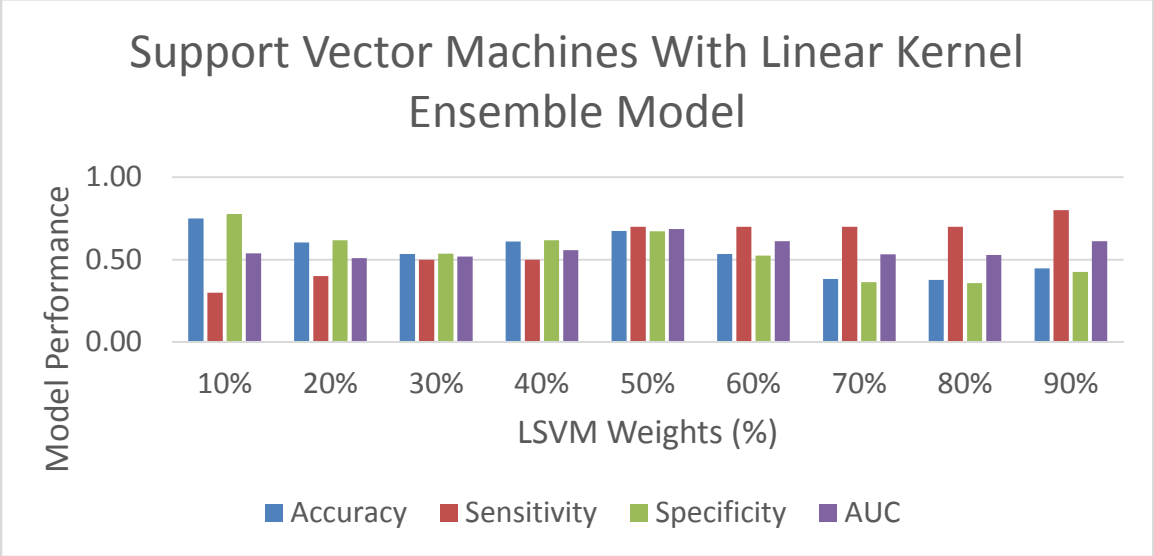

Figure 26 - Varying Weights During Stacked Generalization

4.2.4 Weighted Ensemble Prediction Results Comparison

Table 10 shows the results of the same three chosen ensemble models, but with 50% of the weight on Support Vector Machines with Linear Kernel, and 25% of the weight on both Feed Forward Neural Network and k-Nearest Neighbors. In terms of accuracy Support Vector Machines with Linear Kernel has the best performance with 0.6744. Both Support Vector Machines with Linear Kernel and Logistic Regression have the best performance, with regards to sensitivity, of 0.7. For specificity Support Vector Machines with Linear Kernel again has the best performance with 0.6728. Logistic Regression has the best ROC performance with 0.567. Finally Support Vector Machines with Linear Kernel has the best AUC performance of 0.6864.

Table 10 - Comparison of Weighted Ensemble Model Performance

Model	Accuracy	Sensitivity	Specificity	ROC	AUC
Support Vector Machines with Linear Kernel Ensemble Model	0.6744	0.7	0.67284	0.5591	0.6864
Logistic Regression Ensemble	0.6628	0.7	0.66049	0.567	0.6802
K Nearest Neighbors Ensemble	0.6628	0.6	0.66667	0.5191	0.6333

4.2.5 Weighted Ensemble Prediction Results Discussion

Based on the results for the various performance metrics it is evident that Support Vector Machines with Linear Kernel ensemble has the best performance of the three chosen ensemble models. However Logistic Regression ensemble is not far off in terms of performance. K Nearest Neighbors is the worst performing of the three tested ensemble models. It is important to compare the performance of the ensemble models to that of the individual predictions though. Support Vector Machines with Linear Kernel ensemble outperforms the individual Support Vector Machines with Linear Kernel in every metric besides ROC. Additionally, the Logistic Regression and K Nearest Neighbors ensemble out perform their respective individual performances for nearly every metric.

Since sensitivity was the performance metric of highest significance when developing the model, more weight was placed on the model with the highest sensitivity; Support Vector Machines with Linear Kernel. Additionally, the individual model

performed very well across the other performance metrics. Thus, the increase in performance, specifically sensitivity, can be attributed to the higher weight placed on Support Vector Machines with Linear Kernel during the stacking process. As weighted Support Vector Machines with Linear Kernel ensemble was the best performing of the ensemble and individual predictions, it was retained as the final model for the purposes of this research.

The features included in the base level classifiers were bowel resection, duration of surgery, BMI, wound class, and cancer category. From utilization of these factors three meta level classifiers were retained for use in the stacked generalization process; Support Vector Machines with Linear Kernel, Feed Forward Neural Network, and k-Nearest Neighbors. Weighting the meta level classifiers so that 50% went to Support Vector Machines with Linear Kernel, and 25% went both to Feed Forward Neural Network and k-Nearest Neighbors facilitated higher performance during the stacked generalization process. Weighted Support Vector Machines with Linear Kernel as the stacked generalization classifier achieved the best performance for predicting development of an SSI following gynecologic surgery in cancer patients.

5. Conclusion and Future Work

5.1 Summary

In chapter 2 a review of the available literature was performed in order to determine factors of interest that were used in the prediction of surgical site infections. Additionally, throughout the literature review a series of data mining techniques were identified as the most popular methods of developing prediction models. This thesis utilizes seven data mining techniques in order to predict individual risk of surgical site infection immediately following surgery. In an effort to further improve upon the performance of the individual models, a stacking algorithm was also implemented.

From the significant features identified in the literature 5 features were retained for use in predictions based on the output of the Boruta algorithm. These 5 features are whether or not the patient had a bowel resection prior to surgery, the time between the surgery start and stop in minutes, the patient's BMI at the time of the surgery, the surgical wound classification, and specific location of the patient's cancer. Only one of these variables, duration of the surgery, is not available prior to the operation. It is however available immediately following the surgery which aids in the timeliness of the predictions. The other preoperative variables can be entered into a model prior to the patient's surgery as they are easily obtainable through a patient's medical record. Then once the surgery concludes only the duration of the surgery needs to be added to the model in order to ascertain the patient's risk of developing a surgical site infection.

The seven implemented data mining techniques used to predict individual surgical site infection were Logistic Regression, Naive Bayes, Random Forest, Feed Forward Neural Network, Recursive Partitioning and Regression Trees, K-Nearest

Neighbors, and Support Vector Machines with Linear Kernel. These techniques represent a diverse range of classification techniques whose performance has not been previously compared in prior work. Due to the model's diverse nature, their performances differed across the chosen performance metrics used to compare the models.

Three models stood out due to their high performance; Support Vector Machines with Linear Kernel had the highest sensitivity of 0.7, K-Nearest Neighbors had the highest AUC of the testing set of 0.8086, and Feed Forward Neural Network due its high performance across all of the metrics. Thus, the predictive probabilities of these three models were used as the base level classifiers in three meta level classifiers where performance was compared using the same metrics.

The three meta level classifiers used in the stacking algorithm were Support Vector Machines with Linear Kernel, Logistic Regression, and K-Nearest Neighbors. Support Vector Machines and K-Nearest Neighbors were chosen due to their high performance in the individual models, while Logistic Regression was chosen due to its ease of implementation and interpretability. Of the three meta level classifiers used in the stacking algorithm, Support Vector Machines with Linear Kernel had the best performance in terms of accuracy, sensitivity, specificity, and AUC compared to the other stacking algorithms. Additionally, the implemented Support Vector Machine stacking algorithm performed better than its associated individual model.

5.2 Conclusion

It is often difficult to determine the machine learning technique that best applies to your dataset. The techniques chosen to implement have a significant impact on predictive performance with regards to the chosen performance metrics. This research implements seven popular machine learning techniques to ascertain which model

facilitates the highest predictive performance relevant to predicting a gynecological cancer patient's likelihood of developing a surgical site infection. Additionally, the use of ensemble learning, specifically stacked generalization, was implemented in an effort to improve upon the individual model's predictive performance.

Support Vector Machines with Linear Kernel was chosen as the most effective individual model. This was due to its providing the highest sensitivity along with moderately high accuracy and specificity. It is important to note that the performance of the support vector machine with linear kernel model is specifically tied to the size and characteristics of the dataset utilized in this research. Therefore, one cannot conclude that support vector machines with linear kernel is the technique that facilitates the highest predicting performance for predicting patients at risk of developing an SSI. Additionally, there are significantly more kernel functions that exist and were not tested in this research. When implementing data mining techniques on a similar dataset it may be a good idea to start with support vector machines.

This thesis also compares performance resulting from the use of stacked generalization to the performance of the individual models. Again, the classifier that resulted in the best performance was support vector machines with linear kernel. Using this machine learning technique as a meta level classifier further improved upon the predictive performance resulting from support vector machines with linear kernel being used as a base level classifier. This is another indicator that support vector machines are an appropriate start for implemented data mining techniques on similar datasets.

This research goes even further and explores the used of weighted stacked generalization. Varying the weight placed on Support Vector Machines with Linear Kernel during the stacking process shows the resulting change in performance. The weights that result in the best performance for the four chosen metrics are 50% on Support Vector Machines with Linear Kernel, and 25% on both Feed Forward Neural

Networks and k-Nearest Neighbors. Thorough the utilization of weighting the performance of the stacked generalization models increased slightly over the non-weighted models. This slight increase in performance is significant due to the precise nature of healthcare, and the necessity to have accurate models.

The predictive performance resulting from the implementation of data mining techniques on the dataset used in this research is comparable to, or better than, the performance in associated literature. With the dataset including only 693 patients it is difficult to gather the minute details and relationships among factors that one would be able to discern from significantly larger datasets, up to nearly 850,000 patients, such as what was found in the literature. The performance of the implemented machine learning techniques is attributable to the specific dataset the techniques are being implemented on.

This research most significantly concludes utilizing stacking algorithms promotes better performance than the individual base level classifiers. For the three chosen meta level classifiers all the stacking methods had better performance than their corresponding base level classifier performance. Thus, stacking facilitates generation of a better performing predictive model than any other individual model. Additionally, this research proves the worth of comparing multiple, diverse base level classifiers to the performance of stacking ensemble methods. Additionally, this research concludes that stacking algorithms are appropriate to use in healthcare and more specifically, the prediction of surgical site infections in gynecological cancer patients. Through proactive management of high risk patients, the significant negative impact on the health system resulting from the development of an SSI can be further reduced.

5.3 Future Work

In order to expand upon the work that was done in this thesis there are a few areas that warrant further investigation. First is the collection of more data. Since there were only 693 cases included in this study the predictive power is not as strong as models developed on a larger dataset. Additionally, as the dataset expands it may become more appropriate to predict each specific type of surgical site infection rather than just if a patient will develop a surgical site infection or not. Knowing the likelihood of each severity of SSI can allow the care team to make more appropriate interventions.

Alongside collection of more data is the testing of additional variables. While a large number of factors of interest were identified and tested, further investigation can be performed on the patient's past medical history and demographics related to the patient's area of residence.

This research includes a mix of preoperative and postoperative variables without distinguishing between the three types. As such the predictions regarding whether or not a patient will develop a surgical site infection are only available upon the surgery completion. This can limit the interventions that can take place in order to reduce likelihood of a surgical site infection. In an effort to provide the most accurate information to the care team two models can be developed. One model consisting of only preoperative features so that the care team can take proactive measure to address high risk patients, and another model that is updated during and at the completion of the surgery. A two step model of this nature ensures that the care team has access to pertinent and up to date patient information which can result in the most appropriate care plan.

In order to operationalize the predictive models a decision support system (DSS) should be implemented. Through the implementation of a DSS a surgeon would be able to input information regarding a patient and get clear results about their risk of a Surgical

Site Infection. Additionally, the DSS would have clear indicators of necessary next steps to ensure the most appropriate care plan is followed for a patient. To ease in the implementation of a DSS, one can investigate the development of an index similar to the Charlson Comorbidity Index (CCI). An index places weights on each of the features incorporated in the index according to their importance. Due to the reduced dimensionality of the dataset following feature selection, development of an index is appropriate for the five features identified as significant. Developing an index can aid in the operationalization and reduce future computational strain.

Additionally, different feature selection techniques could be implemented in an effort to perform the predictive performance of the models. Specifically, embedded feature selection can be implemented in order to have the individual prediction models determine the significant features to include in each specific model. Based on the literature performing feature selection in this manner is shown to be more robust as compared to the wrapper method that was implemented in this research.

Higher predictive performance should always be sought after especially in industries such as healthcare where an accurate prediction can mean the difference between a patient's life and death. Additional base level classifiers and meta level classifiers should be tested to determine if there are different models that are more accurately able to predict whether or not a patient will develop a surgical site infection. An increase in the predictive performance of the models will result in the most accurate care plan being developed for a patient, and will ultimately result in a reduced surgical site infection rate.

For future work, even further down the road, a similar form of analysis should be applied to additional types of surgical site infections for cancer patients and ultimately all types of patients. Accurate prediction models that are able to determine patients that are high risk of resulting in a surgical site infection will result in the best proactive care being

provided to patients. Patients who are receiving the appropriate care are at lower risk of developing a surgical site infection, and as a result significantly less money will need to be spent on treatment and readmissions.

References

1. “An Introduction to Feature Selection.” Machine Learning Mastery, 30 Oct. 2016, machinelearningmastery.com/an-introduction-to-feature-selection/.

2. “Cross Validation.” Carnegie Mellon Computer Science, 7 Feb. 1995, www.cs.cmu.edu/~schneide/tut5/node42.html.

3. “Dispel Tutorial 0.8 documentation.” Case studies — Dispel Tutorial 0.8 documentation, homepages.inf.ed.ac.uk/pmartin/tutorial/case_studies.html.

4. “Global Guidelines on the Prevention of Surgical Site Infection.” World Health Organization, World Health Organization, Nov. 2016, www.who.int/gpsc/ssi-prevention-guidelines/en/.

5. “Handbook of Biological Statistics.” Fisher's exact test of independence - Handbook of Biological Statistics, 04 Dec. 2014, www.biostathandbook.com/fishers.html.

6. “Handbook of Biological Statistics.” G–test of independence - Handbook of Biological Statistics, 04 Dec. 2014, www.biostathandbook.com/gtestind.html.

7. “Healthcare-Associated Infections.” Centers for Disease Control and Prevention, Centers for Disease Control and Prevention, 17 May 2012, www.cdc.gov/hai/ssi/ssi.html.

8. “Home.” RStudio, www.rstudio.com/.

9. “Introduction to Support Vector Machines.” OpenCV, 2014, docs.opencv.org/2.4/doc/tutorials/ml/introduction_to_svm/introduction_to_svm.html.

10. “K-Nearest Neighbors for Machine Learning.” Machine Learning Mastery, 21 Sept. 2016, machinelearningmastery.com/k-nearest-neighbors-for-machine-learning/.

11. “k-Nearest Neighbors for Machine Learning.” Statsoft, 2018, www.statsoft.com/Textbook/k-Nearest-Neighbors.

12. “Naive Bayes Classifier.” Statsoft, 2018, www.statsoft.com/textbook/naive-bayes-classifier.

13. “Normalized Data / Normalization.” Statistics How To, www.statisticshowto.com/normalized/.

14. “Pearson's Correlation Coefficient.” Statistics Solutions, 2018, www.statisticssolutions.com/pearsons-correlation-coefficient/.

15. “The Practical Importance of Feature Selection.” KDnuggets Analytics Big Data Data Mining and Data Science, www.kdnuggets.com/2017/06/practical-importance-feature-selection.html.

16. “Two-Sample t–test.” Handbook of Biological Statistics, 04 Dec. 2014, www.biostathandbook.com/twosamplettest.html.

17. “Weight Decay in Neural Networks.” Metacademy, 2012, metacademy.org/graphs/concepts/weight_decay_neural_networks.

18. “Wilcoxon Signed-Rank test.” Handbook of Biological Statistics, 04 Dec. 2014, www.biostathandbook.com/wilcoxonsignedrank.html.

19. Al-Shayea, Qeethara Kadhim. “Artificial Neural Networks in Medical Diagnosis.” IJCSI International Journal of Computer Science, vol. 8, no. 2, Mar. 2011.

20. Amato, Filippo. “Artificial Neural Network for Medical Diagnosis.” Journal of Applied Biomedicine, vol. 11, no. 2, 2013, pp. 47–58., doi:https://doi.org/10.2478/v10136-012-0031-x.

21. Bakkum-Gamez, Jamie N., et al. “Predictors and Costs of Surgical Site Infections in Patients with Endometrial Cancer.” Gynecologic Oncology, vol. 130, no. 1, 2013, pp. 100–106., doi:10.1016/j.ygyno.2013.03.022.

22. Baratloo, Alireza et al. “Part 1: Simple Definition and Calculation of Accuracy, Sensitivity and Specificity.” Emergency 3.2 (2015): 48–49. Print.

23. Benyamin, Dan. “CitizenNet Blog.” A Gentle Introduction to Random Forests, Ensembles, and Performance Metrics in a Commercial System, blog.citizennet.com/blog/2012/11/10/random-forests-ensembles-and-performance-metrics.

24. Brownlee, Jason. “8 Tactics to Combat Imbalanced Classes in Your Machine Learning Dataset.” Machine Learning Mastery, 6 June 2016, machinelearningmastery.com/tactics-to-combat-imbalanced-classes-in-your-machine-learning-dataset/.

25. CMS. “National Health Expenditures 2015 Highlights .” 2015.

26. CMU. “Logistic Regression.” Stat.cmu, 21 Feb, 2013. pp. 223–237.

27. Culver, DH., et al. “Surgical Wound Infection Rates by Wound Class, Operative Procedure, and Patient Risk Index. National Nosocomial Infections Surveillance System.” PubMed, 16 Sept. 1991.

28. Deshpande, Bala. “Reasons Why Feature Selection is Important in Predictive Analytics.” Analytics made accessible: for small and medium business and beyond, 05 Jul. 2011, www.simafore.com/blog/bid/61099/Reasons-why-feature-selection-is-important-in-predictive-analytics.

29. Dutta, Debarati, et al. “How to Perform Feature Selection (Pick imp. variables) - Boruta in R?” Analytics Vidhya, 25 Mar. 2016, www.analyticsvidhya.com/blog/2016/03/select-important-variables-boruta-package/.

30. E Manilich. “Key Factors Associated With Postoperative Complications in Patients Undergoing Colorectal Surgery.” Diseases of the Colon & Rectum, vol. 56, no. 1, Jan. 2013, pp. 64–71., doi:10.1097/DCR.0b013e31827175f6.

31. Engemann, John J., et al. “Adverse Clinical and Economic Outcomes Attributable to Methicillin Resistance among Patients with Staphylococcus aureus Surgical Site Infection.” Clinical Infectious Diseases, vol. 36, no. 5, Mar. 2003, pp. 592–598., doi:10.1086/367653.

32. Esmaeelzadeh, Seyed Reza, et al. “Long-Term streamflow forecasts by Adaptive Neuro-Fuzzy Inference System using satellite images and K-Fold cross-Validation (Case study: Dez, Iran).” KSCE Journal of Civil Engineering, vol. 19, no. 7, 2014, pp. 2298–2306., doi:10.1007/s12205-014-0105-2.

33. Fagotti, Anna, et al. “Risk of Postoperative Pelvic Abscess in Major Gynecologic Oncology Surgery: One-Year Single-Institution Experience.” Annals of Surgical Oncology, vol. 17, no. 9, Sept. 2010, pp. 2452–2458., doi:10.1245/s10434-010-1059-3.

34. Fowler, Vance G., et al. “Clinical Predictors of Major Infections After Cardiac Surgery.” Circulation, vol. 112, no. 9, 30 Aug. 2005, doi:https://doi.org/10.1161/CIRCULATIONAHA.104.525790.

35. Frost, Jim. “Regression Analysis: How Do I Interpret R-Squared and Assess the Goodness-of-Fit?” Minitab, 30 May 1970, blog.minitab.com/blog/adventures-in-statistics-2/regression-analysis-how-do-i-interpret-r-squared-and-assess-the-goodness-of-fit.

36. Fumera, Giorgio, and Fabio Roli. “Cost-Sensitive learning in Support Vector Machines.” Sept. 2002, Department of Electrical and Electronic Engineering.

37. Gbegnon, Akpene, et al. “Predicting Surgical Site Infections in Real-Time.” Drexel, 2010.

38. Gupta, Tushar. “Deep Learning: Feedforward Neural Network – Towards Data Science.” Towards Data Science, Towards Data Science, 5 Jan. 2017, medium.com/towards-data-science/deep-learning-feedforward-neural-network-26a6705dbdc7.

39. Gupta, Vikas. “Home.” Learn OpenCV, 9 Oct. 2017, www.learnopencv.com/understanding-feedforward-neural-networks/.

40. Haider Mahdi., et al. “Surgical Site Infection in Women Undergoing Surgery for Gynecologic Cancer.” International Journal of Gynecological Cancer, vol. 24, no. 4, May 2014, doi:10.1097/IGC.0000000000000126.

41. Hanley, J A, and B J Mcneil. “The Meaning and Use of the Area Under a Receiver Operating Characteristic (ROC) Curve.” Radiology, vol. 143, no. 1, 1982, pp. 29–36., doi:10.1148/radiology.143.1.7063747.

42. Heckerling, P, et al. “Predictors of Urinary Tract Infection Based on Artificial Neural Networks and Genetic Algorithms.” International Journal of Medical Informatics, vol. 76, no. 4, 2007, pp. 289–296., doi:10.1016/j.ijmedinf.2006.01.005.

43. Hengpraprohm, Supoj, and Prabhas Chongstitvatana. “A Genetic Programming Ensemble Approach to Cancer Microarray Data Classification.” 2008 3rd International Conference on Innovative Computing Information and Control, Sept. 2013, doi:10.1109/icicic.2008.35.

44. Izenman, A.J. “Modern Multivariate Statistical Techniques.” Springer Science, 2008, doi:10.1007/978-0-387-78189-1 9.

45. Jacob van Veen, Hendrik, et al. “Kaggle Ensembling Guide.” MLWave, 11 June 2015, mlwave.com/kaggle-ensembling-guide/.

46. Johnson, Megan P., et al. “Using Bundled Interventions to Reduce Surgical Site Infection After Major Gynecologic Cancer Surgery.” Obstetrics & Gynecology, vol. 127, no. 6, 2016, pp. 1135–1144., doi:10.1097/aog.0000000000001449.

47. Kaushik, Saurav, et al. “Feature Selection Methods with Example (Variable selection methods).” Analytics Vidhya, 16 Apr. 2017, www.analyticsvidhya.com/blog/2016/12/introduction-to-feature-selection-methods-with-an-example-or-how-to-select-the-right-variables/.

48. Kaushik, Saurav, et al. “How to Build Ensemble Models in Machine Learning? (with code in R).” Analytics Vidhya, 5 May 2017, www.analyticsvidhya.com/blog/2017/02/introduction-to-ensembling-along-with-implementation-in-r/.

49. Kawaler, Emily et al. “Learning to Predict Post-Hospitalization VTE Risk from EHR Data.” AMIA Annual Symposium Proceedings 2012 (2012): 436–445. Print.

50. Kohavi, Ron. “A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection.” International Joint Conference on Artificial Intelligence, 1995.

51. Lachiewicz, Mark P., et al. “Pelvic Surgical Site Infections in Gynecologic Surgery.” Infectious Diseases in Obstetrics and Gynecology, vol. 2015, Feb. 2015, pp. 1–8., doi:10.1155/2015/614950.

52. Lake, AeuMuro G. et al. “Surgical Site Infection after Hysterectomy.” American journal of obstetrics and gynecology 209.5 (2013): 10.1016/j.ajog.2013.06.018. PMC. Web. 19 Nov. 2017.

53. LaMorte, Wayne W. “Nonparametric Tests.” Mann Whitney U Test (Wilcoxon Rank Sum Test), 2017, sphweb.bumc.bu.edu/otlt/mph-modules/bs/bs704_nonparametric/BS704_Nonparametric4.html.

54. Lee, Michael J., et al. “Predicting Surgical Site Infection After Spine Surgery: A Validated Model Using a Prospective Surgical Registry.” The Spine Journal, vol. 14, no. 9, 1 Sept. 2014, pp. 2112–2117., doi:10.1016/j.spinee.2013.12.026.

55. Legrand, Matthieu, et al. “Incidence, Risk Factors and Prediction of Post-Operative Acute Kidney Injury Following Cardiac Surgery for Active Infective Endocarditis: An Observational Study.” Critical Care, vol. 17, no. 5, 4 Oct. 2013, doi:10.1186/cc13041.

56. Looy, Stijn Van, et al. “A Novel Approach for Prediction of Tacrolimus Blood Concentration in Liver Transplantation Patients in the Intensive Care Unit Through Support Vector Regression.” Critical Care, vol. 11, no. 4, 26 July 2007, doi:10.1186/cc6081.

57. Lunardon, Nicola. “ROSE: A Package for Binary Imbalanced Learning.” Contributed Research Articles, 2013, pp. 79–79.

58. Mu, Yi, et al. “Improving Risk-Adjusted Measures of Surgical Site Infection for the National Healthcare Safely Network.” Infection Control & Hospital Epidemiology, vol. 32, no. 10, 2011, pp. 970–986., doi:10.1086/662016.

59. Nagi, Sajid, and Dhruba Kr. Bhattacharyya. “Classification of Microarray Cancer Data Using Ensemble Approach.” Network Modeling Analysis in Health Informatics and Bioinformatics, vol. 2, no. 3, Sept. 2013, pp. 159–173., doi:https://doi.org/10.1007/s13721-013-0034-x.

60. Naive Bayesian, 2010, www.saedsayad.com/naive_bayesian.htm.

61. Neumayer, Leigh. “Multivariable Predictors of Postoperative Surgical Site Infection after General and Vascular Surgery: Results from the Patient Safety in Surgery Study.” Journal of the American College of Surgeons, vol. 204, no. 6, June 2007, pp. 1178–1187., doi:10.1016/j.jamcollsurg.2007.03.022.

62. Perner, Petra, et al. Medical data analysis: 4th international symposium, ISMDA 2003, Berlin, Germany, October 9-10, 2003: proceedings. Springer, 2003.

63. Polikar, Robi. “Ensemble Learning.” Scholarpedia, www.scholarpedia.org/article/Ensemble_learning.

64. Quackenbush, John. “Microarray Data Normalization and Transformation.” Nature Publishing Group, 2002, doi:10.1038/ng1032.

65. Rose, Sherri. “Mortality Risk Score Prediction in an Elderly Population Using Machine Learning.” American Journal of Epidemiology, vol. 177, no. 5, 2013, pp. 443–452., doi:10.1093/aje/kws241.

66. Sands, Kenneth, et al. “Efficient Identification of Postdischarge Surgical Site Infections: Use of Automated Pharmacy Dispensing Information, Administrative Data, and Medical Record Information.” The Journal of Infectious Diseases, vol. 179, no. 2, 1999, pp. 434–441., doi:10.1086/314586.

67. Sanger, Patrick C., et al. “A Prognostic Model of Surgical Site Infection Using Daily Clinical Wound Assessment.” Journal of the American College of Surgeons, vol. 223, no. 2, 2016, doi:10.1016/j.jamcollsurg.2016.04.046.

68. Shapiro, Mervyn., et al. “Risk Factors for Infection at the Operative Site After Abdominal or Vaginal Hysterectomy.” American Journal of Infection Control, vol. 11, no. 4, 1983, pp. 158–159., doi:10.1016/0196-6553(83)90034-2.

69. Shouman, Mai, et al. “Applying k-Nearest Neighbour in Diagnosing Heart Disease Patients.” International Journal of Information and Education Technology, June 2012, pp. 220–223., doi:10.7763/ijiet.2012.v2.114.

70. Sill, Joseph. “Feature-Weighted Linear Stacking.” Arixv, 4 Nov. 2009, doi:arXiv:0911.0460.

71. Souza, Cesar. Kernel Functions for Machine Learning Techniques. 17 Mar. 2010, crsouza.com/2010/03/17/kernel-functions-for-machine-learning-applications/#kernel_trick.

72. Stencanella, Bruno. “An Introduction to Support Vector Machines (SVM).” MonkeyLearn Blog, 22 June, 2017monkeylearn.com/blog/introduction-to-support-vector-machines-svm/.

73. Taylor, R. Andrew, et al. “Prediction of In-Hospital Mortality in Emergency Department Patients With Sepsis: A Local Big Data-Driven, Machine Learning Approach.” Academic Emergency Medicine, vol. 23, no. 3, 2016, pp. 269–278., doi:10.1111/acem.12876.

74. Team, Analytics Vidhya Content, et al. “Practical Guide to Deal with Imbalanced Classification Problems in R.” Analytics Vidhya, 27 Mar. 2016, www.analyticsvidhya.com/blog/2016/03/practical-guide-deal-imbalanced-classification-problems/.

75. Tu, Y-K, et al. “Problems of Correlations Between Explanatory Variables in Multiple Regression Analyses in the Dental Literature.” British Dental Journal, vol. 199, no. 7, Aug. 2005, pp. 457–461., doi:10.1038/sj.bdj.4812743.

76. Verplancke, T, et al. “Support Vector Machine Versus Logistic Regression Modeling for Prediction of Hospital Mortality in Critically Ill Patients with Haematological Malignancies.” BMC Medical Informatics and Decision Making, vol. 8, no. 1, 2008, doi:10.1186/1472-6947-8-56.

77. Vogler, Raffael. “Illustrated Guide to ROC and AUC.” R-Bloggers, 11 June 2016, www.r-bloggers.com/illustrated-guide-to-roc-and-auc/.

78. Yap, Bee Wah, et al. “An Application of Oversampling, Undersampling, Bagging and Boosting in Handling Imbalanced Datasets.” Lecture Notes in Electrical Engineering Proceedings of the First International Conference on Advanced Data and Information Engineering (DaEng-2013), 2013, pp. 13–22., doi:10.1007/978-981-4585-18-7_2.

79. Z., Zygmunt. “What You Wanted to Know About AUC .” FastML, 19 Sept. 2013 fastml.com/what-you-wanted-to-know-about-auc/.