University of Pittsburgh

Leveraging Machine Learning

To Model Patient Readmittance

Connor Stout, Joseph Datz, Eric Peterson, Myles Onosko, Brian Moum, Uriah West, Michael Franusich

March 23, 2021

Overview

- Problem Statement
- Who Cares?
- Finding Data
- Our Techniques
  - Lasso Regression (Unsupervised)
  - Support Vector Machine (Supervised)
  - Naive Bayes Classifier (Supervised)
- Results
- Questions

Craneware®

Who is Craneware?

Craneware is a enterprise software and data analytics company from Edinburgh, Scotland. They provide software to hospitals meant to reduce costs and develop consistent data standards across all departments.

Reducing Readmittance

Hospital readmittance is a very costly problem, and predicting and preventing patient readmittance is a very strong value proposition for a company.

Problem Statement

Problem Statement

Given anonymized data about a patient's health care, is it possible to determine which factors predict a patient's readmittance within 30 days using statistical and supervised machine learning methods?

Machine Learning (ML) : Defining Terms

- Supervised vs. Unsupervised
  - Supervised Learning: Data set knows what model should predict.
  - Unsupervised Learning: Draws inferences and finds correlation between data points without knowledge of what that data represents.
- Data Set - Used to train (teach) a ML model
- Labels - Actual outcome model should predict
- Error/Cost - A measurement of how off the model is. ML models minimize this.
- Training - Using the Data set to minimize the error of the model in order to improve predictions.

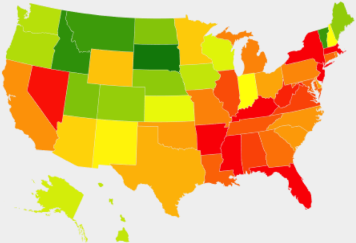
Who Cares?

Hospitals

Hospitals are charged a penalty for patient readmittance within 30 days. Hospitals incurred $528 million in penalties in 2017.

Medicare

There was an estimated $17 billion dollars of avoidable Medicare costs associated with patient readmittance in 2015.



Modern Medicine, Dark Age Data

Inconsistent Data Standards

Hospitals have notoriously inconsistent data storage standards. There is only one publicly available database of patient level admissions data.

MIMIC-II

This database provided anonymized data of 30,000 patient hospital admissions. MIMIC-III provided better formatted data but lacked the dates of patient admittance needed to determine readmittance.

Vectorization

This large data set was converted to large binary vectors to train our model.

Data Extraction/Interpretation

Approach

- Data is mostly categorical. Computers are good at interpreting numbers.
- Solution? Make a long vector, where each index represents a category.
- Place a 1 if a patient belongs to that category. Place a 0 if the patient does not belong to that category.

Feature Selection

Dataset had at least 10,000 elements per patient record

Had to determine which features are the most important for predicting readmitted patients

Three different types of regression methods for determining most important features:

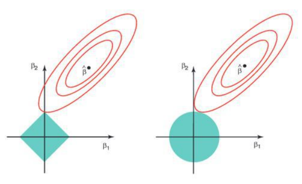
- Ridge Regression
- Lasso Regression
- Elastic Net Regression

Determined that 25 features were sufficient to feed into the Support Vector Machine Algorithm

Ridge Regression

- N: The number of examples in a dataset (# of Patients)
- $y^{(i)}$: The output of a dataset (Readmittance)
- $x^{(i)}$: The input of a dataset (Condition, drugs taken, etc.)
- β_n: Regression coefficents.
- λ: Optimal value which minimizes MSE.

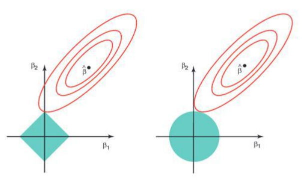
$$\min_{\beta_0,\beta}\left(\frac{1}{2N}\sum_{i=1}^{n}(y^{(i)}-\beta_0-x^{(i)T}\beta)+\lambda\sum_{j=1}^{p}\|\beta_j\|_2^2\right)$$



Lasso Regression

- N: The number of examples in a dataset (# of Patients)
- $y^{(i)}$: The output of a dataset (Readmittance)
- $x^{(i)}$: The input of a dataset (Condition, drugs taken, etc.)
- β_n: Regression coefficents.
- λ: Optimal Value which minimizes MSE.

$$\min_{\beta_0,\beta}\left(\frac{1}{2N}\sum_{i=1}^{n}(y^{(i)}-\beta_0-x^{(i)T}\beta)+\lambda\sum_{j=1}^{p}\|\beta_j\|_1\right)$$

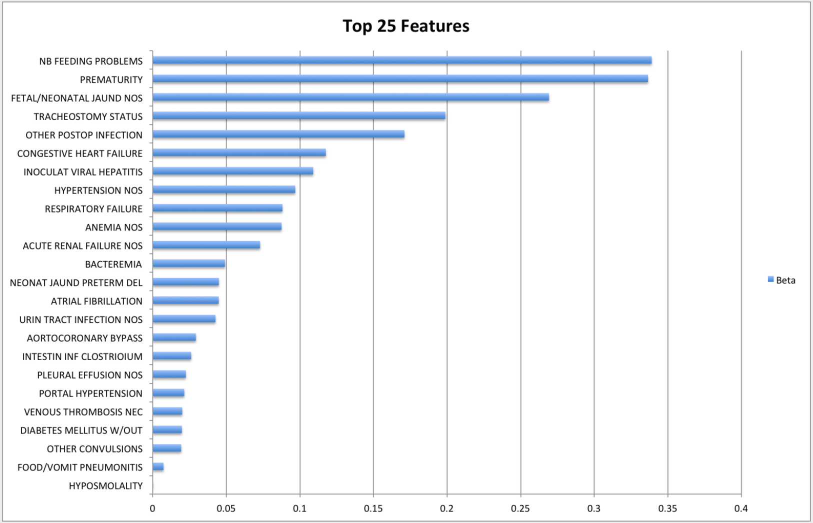


Elastic Net Regression

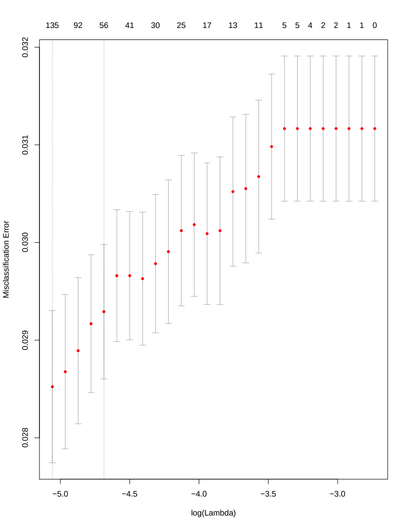
- N: The number of examples in a dataset (# of Patients)
- $y^{(i)}$: The output of a dataset (Readmittance)
- $x^{(i)}$: The input of a dataset (Condition, drugs taken, etc.)
- β_n: Regression coefficents.
- λ: Optimal value which minimizes MSE.

$$\min_{\beta_0,\beta}\left(\frac{1}{2N}\sum_{i=1}^{n}(y^{(i)}-\beta_0-x^{(i)T}\beta)+\lambda\sum_{j=1}^{p}\|\beta_j\|_2^2+\lambda\sum_{j=1}^{p}\|\beta_j\|_1^2\right)$$

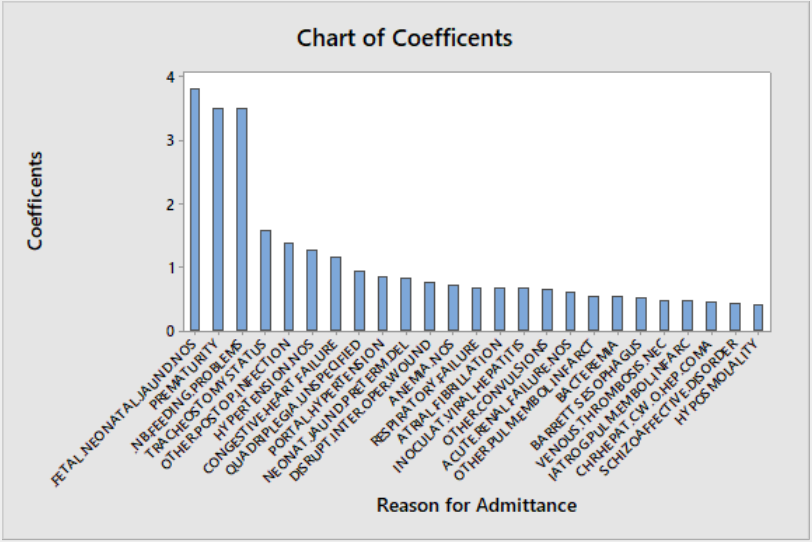
Results: Lasso



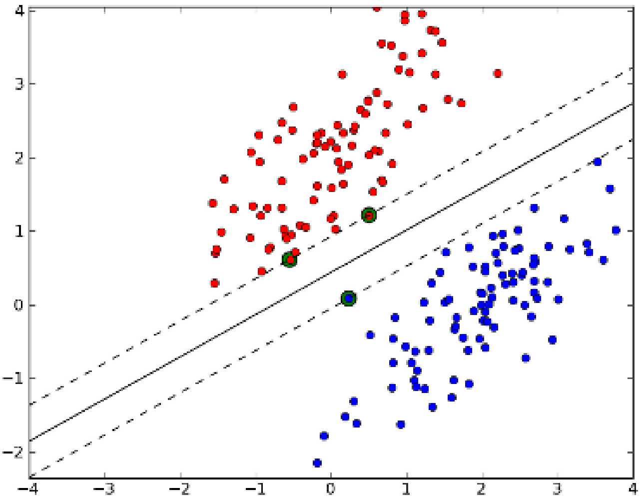
Results: Elastic Net Regression Lambda



Results: Elastic Net Regression



Support Vector Machine



Defining a Support Vector Machine

Define:

- w: a vector/matrix of constants.
- b: a bias shifting the position of the SVM.
- α_i: A datapoint existing directly on the margin.
- C: a parameter which is free to vary.

We want to build a hyperplane s.t

$$w^T x^{(i)} - b = 0$$

And we want to have two other parallel planes s.t

$$w^T x^{(i)} - b \geq 1, \quad \textit{if} \quad y^{(i)} = 1$$

$$w^T x^{(i)} - b \leq -1, \quad \textit{if} \quad y^{(i)} = -1$$

The distance between these two planes is called the **margin.**

We can rewrite this as:

$$y^{(i)} * (w^T x^{(i)} - b) \geq 1, \quad \forall i$$

Note that the length of the margin is $\frac{2}{||w||}$. Therefore, to maximize the margin, we want to find:

$$min \quad ||w|| \quad s.t \quad y^{(i)} * (w^T x^{(i)} + b) \geq 1 \quad for \quad i = 1, ..., m$$

In Optimization, this is known as a **primal** problem. It's **dual** form can be written as:

$$\max W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} y_i y_j \alpha_i \alpha_j \langle x^{(i)}, x^{(j)} \rangle$$

$$s.t \quad 0 \leq \alpha_i \leq C \quad i = 1, ..., m, \quad \sum_{i=1}^{m} \alpha_i y^{(i)} = 0$$

Where α_i are Support Vectors and C is a free parameter.

Next, we define a **kernel function** to be:

$$K(x,z) := \phi(x) * \phi(z)$$

Where z is another feature from our training set X. A Kernel function's goal is to measure the "similarity" between two features. There are many valid kernels:

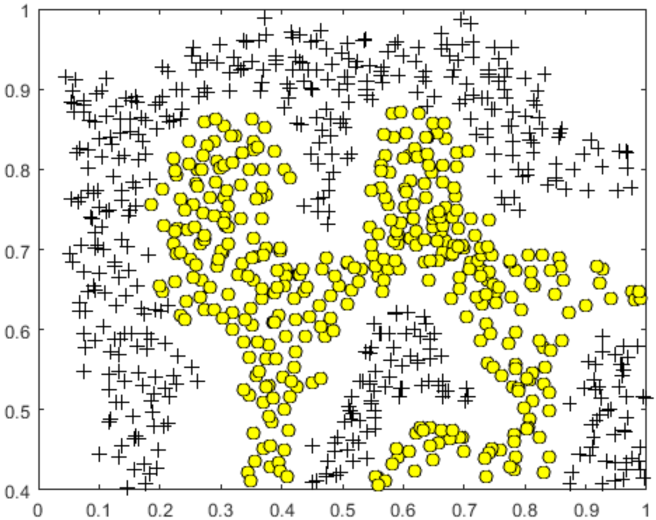
$$K(x,z) = (x^T z)^2$$

$$K(x,z) = \tanh(\gamma * x^T z + c)$$

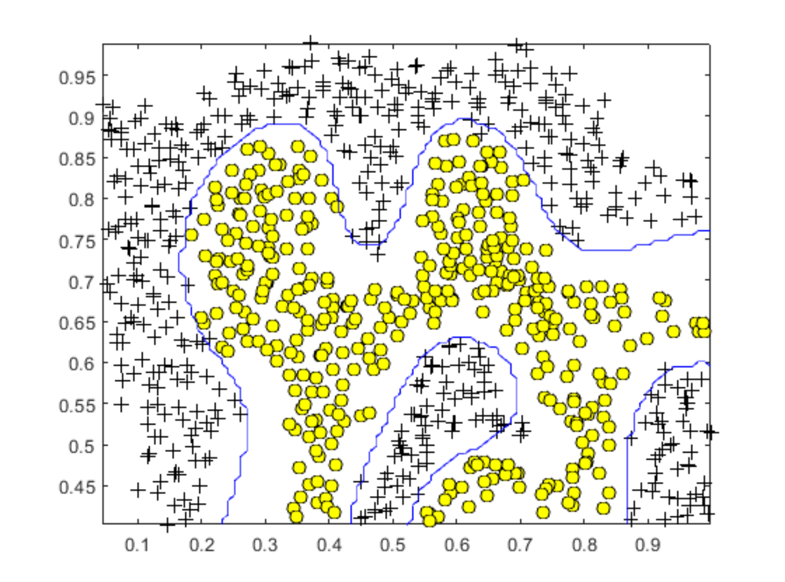
$$K(x,z) = e^{-\frac{||x-z||^2}{2\sigma^2}}$$

Where the inner product $\langle x^{(i)}, x^{(j)} \rangle$ occurred in our dual problem, we can replace with K(x,z).

Why bother with a Kernel?



Why bother with a Kernel?



Results: SVM

Training

The model was trained using the top 25 most influential features from our Lasso model and can accurately predict patient readmittance with a **97.20%** total accuracy and a **17.59**% accuracy on only readmitted patients.

Similarly with our Ridge Model we can accurately predict readmittance with **97.36%** total accuracy and a **38.31%** accuracy on only readmitted patients

Bayesian Classifier

A Naive Bayes classifier attempts to answer the question, "what is the probability that a problem instance is of a certain class C_k given a vector $\vec{x} = \{x_1, x_2, ..., x_n\}$ of n independent parameters." Applying Bayes Theorem, this problem can be written as:

$$P(C_k|\vec{x}) = \frac{P(C_k)P(\vec{x}|C_k)}{P(\vec{x})}$$

The classifier is a function which combines Bayes theorem with a decision rule; in this case the *maximum a posteriori*, or MAP decision rule. In plain English, the classifier calculates the probability of each hypothesis and chooses the one with the highest probability.

$$\hat{y} = \underset{k \epsilon \{1..,K\}}{\text{argmax}}\, P(C_k) \prod_{i=1}^{n} P(x_i|C_k)$$

Bernoulli Naive Bayes

The assumptions on distributions of features are called the event model of the Naive Bayes classifier. Based on the nature of the data, the most relevant event model was the multivariate Bernoulli event model, which excels in handling binary features. The model calculates the probability

$$p(\vec{x}|C_k) = \prod_{i=1}^{n} p_{k_i}^{x_i}(1 - p_{k_i})^{(1-x_i)}$$

where p_{k_i} is the probability of class C_k generating the term x_i. A direct improvement of this model over the SVM is that it has the benefit of explicitly modeling the absence of terms, which can be extremely useful on this particular dataset.

Results

The dataset was split randomly over ten separate trials and fed through the classifier, which returned the following results:

- Average Total Accuracy: 92.4%
- Average Readmittance Prediction Accuracy: 69.4%
- Average True Positive to False Positive Ratio: 1:3

Acknowledgements

Mentors

Special thanks to Dr. Jeffrey Wheeler, Nathan Ong, Dr. Andrew Walsh of Health Monitoring Systems, Eric Bentley of Conduent, and David Hunter, Jourdain Lamperski and Chris McCord of MIT for their assistance and guidance.

University of Pittsburgh

We extend our appreciation to the University of Pittsburgh Department of Mathematics for offering this course, and providing the resources that aided in this project's success.

Craneware

Thank you to Korin Reid of Craneware for giving us this opportunity to apply the skills we learned at Pitt.

Questions

Thank you!