Cardiovascular Disease Prediction

A Clustering-Based Classification Approach

Jamie Chen
Computational and Applied Math, Cognitive Science & Data Science
Rice University
Houston TX, U.S.
lc70@rice.edu

Bhavesh Shah
Computer Science & Data Science
Rice University
Houston TX, U.S.
bs72@rice.edu

Griffin Coccari
Transnational Asian Studies & Data Science
Rice University
Houston, TX, U.S.
gjc5@rice.edu

ABSTRACT

Cardiovascular diseases (CVDs) are one of the largest and most impactful class of diseases worldwide, encompassing an array of serious diseases from coronary artery disease to high blood pressure, and affecting the lives of millions each year. Nonetheless, research shows that CVDs are often preventable or, at a minimum, addressable in earlier stages to help curb future worsening. There are copious applications in medical fields for diagnosis and prevention or treatment, including many for cardiovascular or other highly related, similarly lethal, and widespread diseases. However, one of the most common applications is disease prediction – to predict the likelihood of a patient developing heart disease. Different methods have been tried and applied to such problems, and there continues to be huge efforts in this domain, accompanied by strong desires for novel and improved applications of advanced methods in artificial intelligence and machine learning to be deployed in the medical field. Given the wide array of algorithms utilized on similar problems, in our experiment, we choose to employ a comparatively unconventional approach. Here, we apply a cluster then classify strategy; we attempt to segment the entire dataset into distinct groups with similar characteristics, then train an array of classification models - logistic regression, SVM, KNN, random forest, neural nets – on each of the individual clusters to find the optimal model and set of parameters for each group of similar patients. The advantages of our approach are the additional cluster information for each patient that may inform other medical decisions, as well as more targeted models for each cluster that may promote improved overall model performance.

CCS CONCEPTS

• Artificial Intelligence • Machine Learning • Life and medical sciences

KEYWORDS

Cardiovascular disease prediction; medical machine learning; clustering; classification

1 Introduction

1.1 Medical Applications of Machine Learning

Medical applications of machine learning have become increasingly prevalent over the past decade. From computer vision to anomaly detection, the use cases have been growing; they are extensive and diverse. As machine learning algorithms improve and novel techniques are developed, the accuracy and robustness of the applications also increase. Thus, the medical field, among other popular areas of application such as finance and entertainment, has witnessed a drastic increase in adoption of machine learning and more advanced techniques to help support and inform the medical professionals (e.g., doctor, nurses, and other healthcare professionals) in the field. For example, in the realm of medicine and healthcare, machine learning algorithms are being introduced and utilized for clinical, private, and public applications - from diagnosis and treatment to insurance pricing and outbreak prediction. More specifically, for clinical applications, diagnosis can be both disease identification through health records and testing results to medical imaging and object detection. In addition, personalized medicine is also becoming increasingly commonplace in some more advanced societies today. Nonetheless, these trends are only possible with the developments in machine learning research. For example, serviceable imaging applications such as tumor detection is only recently enabled and bolstered by the latest advancements in artificial intelligence (AI), and more specifically, convolutional neural networks (CNN).

Lastly, one cannot overlook the drastic advancements made in computing hardware in the last decade that has enabled the realization of the software and mathematical progress and breakthroughs behind these algorithms. Improved GPU, and even the eventual potential mass adoption of quantum computers, has, and will, be crucial in facilitating the practical employment of these research methods. Thus, these advancements in machine learning, AI, and computing technology undoubtedly are, and will, play an important part in the shifting field of medicine as well as the changing role of medical practitioners around the world.

1.2 Cardiovascular Diseases (CVDs)

CVDs is a general, all-encompassing term referring to conditions affecting the heart or blood vessels. There are four main types of CVDs consisting of coronary heart disease, strokes and transient ischemic attacks (TIAs), peripheral arterial disease, and aortic disease, with 85% of CVD deaths attributable to heart attack and

stroke (Felman, 2019; WHAT IS CARDIOVASCULAR DISEASE?, n.d.).

Furthermore, CVDs are the leading cause of death and disability globally. In 2019, approximately 18 million people died from CVDs around the world, constituting 32% of total global deaths. In the United States, people currently die from CVDs at an astounding rate of one death every 36 seconds (Heart Disease Facts, 2021). The WHO also estimate that, by 2030, 23.6 million people will die from CVDs-related causes annually (Cardiovascular diseases (CVDs), 2021). Moreover, CVDs cost the United States a whopping $363B in a single year (from 2016 to 2017) (Heart Disease Facts, 2021).

Despite its severity, according to WHO and NHS, the majority of CVDs can be prevented by leading a healthy lifestyle and addressing behavioral risk factors such as tobacco use, unhealthy diet and obesity, physical inactivity, and excessive alcohol. In addition to the aforementioned lifestyle risk factors, other major known risk factors include high blood pressure, high blood cholesterol, radiation therapy, sleep apnea, stress, air pollution, and chronic obstructive pulmonary disorder or other lung function hindering diseases, among many others with varying levels of correspondence (Cardiovascular disease, 2018). However, the exact cause of CVDs remains unclear; thus, current medical and healthcare professionals can only rely on risk factors to assess a patient's likelihood of developing CVDs. Hence, accurate prediction of CVDs would be extremely beneficial to the healthcare community currently reliant on heuristics to diagnose high-potential CVDs patients. Furthermore, the combination of characteristics most conducive to CVDs would also be helpful for both doctors and patients.

2 Related Work

There has been a plethora of past and current work on medical diagnoses via machine learning methods, including a substantial amount of research on machine learning models for CVDs or various subcategories of CVDs, especially major, most significant heart-related diseases such as coronary artery disease, heart failure, stroke, and cardiac arrhythmias among others.

In 2016, Krittanawong et al. evaluated and summarized the overall predictive ability of machine learning algorithms applied to CVDs, with a focus on machine learning algorithms of coronary artery disease, heart failure, stroke, and cardiac arrhythmias, accumulated over the past few decades in a comprehensive research project. Their primary motive was to produce a composition of the predictive ability of machine learning algorithms of CVDs through an extensive survey of 344 identified studies, 104 cohorts, and a total of 3,377,318 individuals. In their investigation, Krittanawong et al. used pooled area under the curve (AUC) as their comparison metric for the competing models included in their study.

For prediction of coronary artery disease, custom-build algorithms (unable to be adequately classified under any of the generic machine learning algorithm types as model details are not publicly disclosed) performed the best with a pooled AUC of 0.93, with boosting algorithms coming in second with a pooled AUC of 0.88. Meanwhile, for stroke prediction, the top three performing algorithms had very close pooled AUC. Support vector machine (SVM) algorithms had a pooled AUC of 0.92, while boosting algorithms and convolutional neural network (CNN) algorithms resulted in pooled AUCs of 0.91 and 0.90 respectively. Lastly, the studies of heart failure and cardiac arrhythmias prediction algorithms remain inclusive, as the confidence intervals overlap between the various methods. However, we can conclude that the predictive ability of machine learning algorithms for CVDs is generally quite promising, with special mentions to SVM and boosting algorithms for, overall, excellent performances (Krittanawong et al, 2016).

In 2017, Tripoliti et al. conducted an even more detailed survey specifically targeted for prediction of heart failure – a subset of CVDs becoming increasingly prevalent around the world with an estimated 26 million cases each year, and an unfortunate, but common end to many CVDs (Roadmap for Heart Failure, 2019). The paper aims to present the latest and greatest machine learning methods for a variety of health failure prediction assessments, from predicting the presence of heart failure and estimating its specific subtype to general severity assessments and predicting the probability of the occurrence of a range of adverse events such as destabilizations, re-hospitalizations, and even mortality.

For heart failure detection, most of the examined studies utilized heart rate variability (HRV) as a key measure to determine whether a subject was a patient with heart failure. Methods considered range from Bayesian classifiers (Asyali et al., 2003), SVM (Jovic et al., 2011; Yu et al., 2012) and kNN (Isler et al., 2007) to CART with feature selection (Mellilo et al., 2011), supervised multi-layer perceptron (MLP) and unsupervised self-organizing maps (SOP) (Elfadil et al., 2011). Among the different studies, Liu et al. achieved 100% SVM accuracy, precision, and sensitivity for 30 normal subjects and 17 congestive heart failure (CHF) patients (2014) while Jovic et al. obtained 98.4% sensitivity and 99.2% specificity using Bayesian classifiers for a balanced data set of 25 normal subjects and 25 CHF patients (2011). Moreover, without HRV measures, Gharehchopogh et al. were able to attain true positive, precision, and recall of 95% with 26 normal subjects and 14 heart failure patients (2011). For classifying heart failure subtypes, SVM PEGASOS and PLP were key methods used that delivered the best results (Betanzos et al., 2015; Isler, 2016). For heart failure severity estimation problems, researchers often transformed the predictor to be a two or three class classification problem; however, the definitions varied across different studies, thus is difficult to compare. Lastly, the prediction of various adverse events is often reliant on the flexible deep learning architectures for accurate predictions. And most impressively, a method has been proposed where, according to the authors, it can successfully predict heart failure onsets nine months before current doctor diagnosis (Roger, 2010).

3 Methods

We started this project with Exploratory Data Analysis (EDA) where we analyzed the data's observations and features and performed necessary preprocessing. After that, we worked on

clustering the data into an optimal number of groups. The last step involved training five classification model for each cluster to determine our ideal cluster-specific models.

3.1 Data

The data for this project came from a Kaggle CVD dataset. It contains 11 features and a binary target classifying whether a patient has a CVD or not. The features comprise of objective features (age, gender, height, weight), examination features (systolic BP, diastolic BP, cholesterol, glucose), and subjective features (smoking, alcohol intake, physical activity).

In our EDA, we started off with an observation (tuple) analysis. There were exactly 70,000 observations with no missing values and duplicate rows. Additionally, the number of unique elements for the binary and ternary variables matched was as expected. The distributions of the numerical data were mostly normal except for age, which was skewed to the right (meaning that we mostly had older participants in this study). The categorical data distributions were a bit more unbalanced. For example, the number of physically active participants was almost 3x the number of inactive participants. However, our hypothesis was that our clustering strategy would mitigate the effect of these unbalanced features by distributing the imbalances across the clusters. Because our dataset seemed clean, we didn't do any data preprocessing for the first iteration of our model training. However, as we will discuss later, we ended up noticing that some of the observations were a bit unrealistic (certain blood pressures were too high, some weights were extremely low, etc.). During our second iteration of this project, we ended up going back to preprocess the dataset and remove these outliers.

The next part of our EDA involved a feature analysis. Firstly, we decided to measure feature-to-feature correlation for our numerical features using a Pearson Correlation matrix. With Pearson, our highest correlation coefficients were between height and gender (.5) and glucose and cholesterol (.45), with all the other coefficients being quite lower. Because none of the coefficients exceeded .5, which is usually considered the minimum threshold for high dependence, we concluded that our data did not have multicollinearity. We also measured feature-to-target correlation using Anova F-Test (as our target was binary). From this, we observed that all features had a P-value less than .05, implying that they were statically significant. To confirm that we do not need to reduce our feature space, we decided to do a PCA analysis, where we saw that percent of data represented by the top 2 PCA components was only 37%. Ultimately, with the correlation and PCA analysis, we concluded that we did not need to reduce our feature space; this was helpful since we wanted to maximize the number of features used by advanced learners like neural nets.

3.2 Strategy

Most of the prior work that had been done for CVD prediction had involved training classification models on entire datasets. We decided to approach this a bit differently and rather train classification models on subsets (clusters) of the dataset. The reasoning behind this was that different groups of patients might have different features contributing to why they have a CVD.; for example, for older patients, blood pressure may be the most significant factor. In addition to different feature importance, clusters could also have different types of optimal models based on the structure of the data. Thus, our end goal became to create a more personalized pipeline that sorts a patient into a cluster, and then invokes the best model (for that cluster) on the patient.

3.3 Clustering

3.3.1 K-Means. K-Means is the first algorithm we decided to try (and ended up using in our pipeline). K-Means is an unsupervised algorithm that groups data points into clusters using a centroid-based clustering approach. Specifically, it uses hard clustering, which ensures that a point cannot belong to more than 1 cluster. One of the most important hyperparameters to tune for K-means is the number of desired clusters; we decided to try a range of clusters from 2 to 20. Using the Elbow method, we determined that 6 clusters was the best fit for our dataset.

3.3.2 GMM. In order to verify that 6 clusters are the ideal number of clusters for our dataset, we decided to train a GMM model as well. GMM is another unsupervised algorithm that groups points using a distribution-based clustering approach. It uses soft clustering which allows for clusters to overlap. We plotted the AIC and BIC scores (information criteria scores) for each GMM model on a range of clusters from 2 to 20. In order to minimize AIC and BIC, we found out that using 19 clusters was the most ideal, which conflicted with what we had observed during K-Means. Ultimately, we decided to stick with our K-Means results since using 19 clusters would reduce the individual cluster dataset sizes, which wasn't optimal since we didn't have a lot of data to begin with. Additionally, when we used 6 clusters, we noticed that our clusters were a lot more interpretable.

3.3.3 Cluster Interpretations. To better understand how our clusters varied, we decided to plot cluster averages by feature. Our goal with this was to figure out which features better characterized clusters. We came up with this list based on our results:

- Cluster 0: high BP, older crowd, majority female, shortest
- Cluster 1: high glucose, high cholesterol, older crowd
- Cluster 2: alcoholics, some smokers, fewest members
- Cluster 3: smokers, majority male
- Cluster 4: majority male, tallest
- Cluster 5: essentially all female, shortest, lightest, most members

Some of these clusters have very distinguishable characteristics such as cluster 1 having high glucose/ cholesterol patients, and cluster 2 having lot of alcoholics/ smokers. This information was very helpful to know because it helped us understand why clusters had certain feature importances during our classification analysis.

Additionally, we also decided to visualize our clusters using PCA (as our feature space was 11-dimensional). As seen in the figure 1, most of the clusters do not have much overlap with the exception of cluster 0 and cluster 1 (this overlap is not something we necessarily needed to worry about since using higher dimensions helps with separating out the data points).

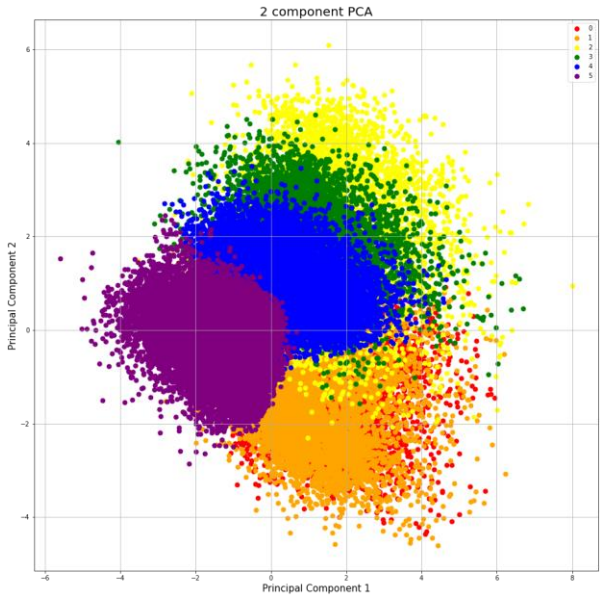

Figure 1: Cluster Visualizations using 2-Component PCA

3.4 Classification

3.4.1 Logistic Regression. Logistic Regression is one of the most fundamental classification regression models. Unlike a linear regression, which has a continuous output, the logistic regression model uses probability to predict a binary output. This model rests on a few assumptions. The model assumes there is little to no multicollinearity between variables, that there are no outliers, that the data sample sizes are larger, and that each variable is important (Joby).

We think that the Logistic Regression model was a good choice for our first model because it is, like linear regression, very interpretable. The hypothesis is as follows.

$$h(\emptyset) = \frac{1}{1 + e^{-\emptyset^t x}}$$

$\emptyset$ is our vector of parameters and x is our observations. Since $0 \leq h(\emptyset) \leq 1$, we can think of $h(\emptyset)$ as a probability of the observation belonging to the "1" class. If $h(\emptyset) \leq .5$, then the model will predict a 0 as the outcome, and if $h(\emptyset) \geq .5$, then the model will predict 1 as the outcome.

To optimize our logistic model, we found the optimal hyperparameters using grid search. Through our grid search, we determined that, in general, a l2 penalty, or lasso regression, was optimal for our clusters. Lasso regression helped shrink our least important features to 0, and in this way, it helped us with feature selection. The "C" value was also another parameter we had to tune. We tested "C" values from .5 to 2 at increments of .2. Because the "C" value indicates how much weight to assign to the training data, a middle to high "C" value indicates that model assigned significant weight to the training data.

This model scored the highest accuracy on clusters 0 and 4. It ended up working so well most likely because of our large data size and non-correlated features.

3.4.2 KNN. K-Nearest-Neighbors (kNN) is a classification algorithm that classifies x according to the majority class of its nearest neighbors. The algorithm is one of the simplest machine learning classifiers, but it has a few downsides. The model is easily affected by features with different scales, for example. We resolved this issue by z-scoring all the numerical features. Also, this algorithm is prone to suffer from the curse of dimensionality, which is a problem one may run into when dealing with a high number of variables. In this scenario, unimportant variables may swamp out the information from more relevant attributes. However, since we determined from our Anova F-Test than all the features are statistically significant, we decided to not explicitly handle the curse of dimensionality when training. We did run grid search on each cluster to determine the best hyperparameters (neighbors, distance metric, etc.) for each cluster. For the clusters, the optimal number of neighbors were 31, 29, 29, 29, 21, and 31, respectively.

Our kNN algorithm was our weakest model for each cluster. We believe that it was not the right model for this data because of the high dimensionality of the data as well as the presence of lots of noise.

.

3.4.3 Random Forest. Random Forest (RF) model is a collection of decision trees. While decision trees are immune to multicollinearity, they are prone to overfit data, and it is not a stable algorithm. The RF model attempts to reduce the instability of decision trees by averaging their effects. RF models rely on the assumption that the features are identically distributed.

Random Forest relies on a few methods of ensemble learning: bootstrap bagging and feature bagging. Bootstrap bagging is where multiple trees are learned on different subsets of the training data. Feature bagging is where the algorithm selects a random subset of features at each decision node. These two methods help reduce overfitting of the individual decision trees. The RF model then measures its accuracy with Out Of Bag (OOB) Classification Error, the mean prediction error on each training sample, xi, using only the trees that did not have xi in their bootstrap sample. The number of trees chosen is where the OOB error changes minimally.

While we did not expect overfitting to be a major problem based on our previous experience running models on the dataset, we wanted to try the random forest model. The model scored the highest for model 2. The high efficacy of this model may be thanks to its ability to utilize feature bagging and bootstrap bagging to balance the impact of features and observations on the classified outcome.

3.4.4 Support Vector Machines. Support Vector Machines (SVM) is another classification algorithm we ended up leveraging. SVM has a couple key components: the optimal hyperplane and the margin. The optimal hyperplane can be thought of as the typical

boundary that separates the classes. The margin is the distance from the hyperplane to the closest point for each class. The optimal hyperplane can be found through the objective function.

$$max_{w,b}(min_i(\frac{(y_i * (w^t x + b)}{\|w\|})$$

This optimal hyperplane is the line that maximizes margin, or the closest distance from the hyperplane to the support vectors.

Support Vector Machines were one of our best models because they have many useful qualities. They are effective in high dimensional spaces and are very versatile because they can be used with different kernels on nonlinear and non-linearly separable data (Pedregosa). For our clusters, an RBF kernel seemed to work best. Other hyperparameters we tuned using grid search included C and gamma.

Overall, our SVM models did very well but tended to slightly underperform when compared to logistic regression/ neural nets. Because they never were the best model for any of our clusters, we did not end up using them in our final pipeline.

3.4.5 Neural Networks. The last model we trained on each cluster was a feed forward neural net (FFNN). FFNNs are great at modeling non-linear relationships in the data and tend to work really well with large amounts of data (unlike many other machine learning algorithms that stop improving accuracy-wise after a certain data size). FFNNs are the most standard variant of neural nets and are basically powered by backpropagation and gradient descent techniques under the hood. We chose to use FFNNs specifically because our data wasn't sequential or spatial, so a simple neural net was sufficient to model relationships in our data.

When training our FFNN's, we decided to try a couple different architectures comprising of 1-3 hidden layers. We also ran a grid search for each of these architectures to optimize the number of neurons, batch sizes, and epochs. We noticed that these 3 architectures all resulted in very similar accuracies but decided to end up going with a 2 hidden-layer architecture since it performed slightly better than the others. Overall, FFNNs resulted in some of the highest accuracies across the clusters, and specifically performed the best for clusters 1 and 3.

4 Discussion

Listed above in the methodology section are a few important algorithms that we applied to each cluster. Each algorithm has strengths and weaknesses. For example, logistic regression works well with a large amount of data but will suffer more than other algorithms from multicollinearity. Random Forest, while immune to multicollinearity, is computationally expensive and assumes that features are identically distributed.

We used clustering to create interpretable groups that our algorithms might be able to understand better. In the same way that each model is used in different scenarios, each cluster would be a bit different. It was our hope that by utilizing many models for each cluster, we could then utilize the strengths of each model to meet the specific needs of the clusters.

w	Baseline	KNN	Logistic	SVM	Random Forest	Neural Nets
Cluster0	0.71	0.74	0.77	0.78	0.77	0.77
Cluster1	0.7	0.79	0.79	0.81	0.81	0.8
Cluster2	0.53	0.64	0.74	0.74	0.74	0.74
Cluster3	**0.63**	0.66	0.69	0.69	0.68	0.69
Cluster4	0.85	0.81	0.82	0.82	0.84	0.82
Cluster5	0.52	0.7	0.74	0.74	0.75	0.7
Cluster6	**0.52**	0.68	0.71	0.71	0.7	0.71
Cluster7	**0.51**	0.67	0.7	0.7	0.68	0.69
Cluster8	**0.51**	0.66	**0.68**	0.67	0.66	0.67
Entire Dataset	**0.50**	**0.58**	**0.72**	**0.72**	**0.72**	**0.71**
	Cluster-wise average:	**0.71**	**0.74**	**0.74**	**0.74**	**0.73**

Table 1: Classification Model Performances (Iteration 1)

As mentioned earlier, we went through 2 iterations of our model training process. During our first iteration, we determined that the optimal number of clusters was 9. Table 1 shows the optimal models for each cluster. There are some clusters which have multiple optimal clusters. For example, cluster 2's best models were logistic, RF, and SVM, and neural nets. However, we later came to realize that the noise in the un-processed data was adversely affecting our models. For iteration 2, we decided to process our data and rerun our models. After doing so, we obtained the results shown in Table 2.

w	Baseline	KNN	Logistic	SVM	Random Forest	FFNN
Cluster0	0.71	0.813	0.813	0.813	0.811	0.813
Cluster1	0.7	0.620	0.683	0.680	0.683	0.687
Cluster2	0.53	0.559	0.744	0.747	0.769	0.744
Cluster3	**0.63**	0.538	0.763	0.757	0.758	0.764
Cluster4	0.85	0.561	0.701	0.697	0.697	0.700
Cluster5	0.52	0.675	0.704	0.705	0.707	0.704
Entire Dataset	**0.66**	**0.70**	**0.727**	**0.731**	**0.723**	**0.730**
	Cluster-wise average:	**0.63**	**0.735**	**0.733**	**0.738**	**0.735**

Table 2: Classification Model Performances (Iteration 2)

After processing the data, we obtained more interpretable clusters, and then we found new optimal models. The best models are logistic, NN, RF, NN, Logistic, and RF for each cluster respectively. This time there were only 6 clusters, each one more distinct than when we had 9 clusters. Our results varied after pre-processing. Indeed, our accuracies decreased marginally in most areas, but removing our outliers ultimately gave us more interpretable clusters and model outcomes. Pre-processing, while not actually beneficial, was the step in the right direction to improving accuracy. If we had more time or resources at our expenses, we could reproduce this experiment in an even more effective way.

The next time this research is repeated, we hope that there might be access to more rows and features of data. Other ML research in CVD used more features with their own proprietary data. With 72 features, The NIH models had high degrees of separability. They believe that "a large number of clinical variables" (Haichen) are necessary and required to predict CVD outcomes. While we didn't do as well as the NIH models, our ensemble model scored better than a project done on the same dataset at Brown University, as well as better than notebooks on Kaggle. For example, Turhan Kargin, a graduate student at PUT, got a logistic accuracy of 72% on the same data. Our cluster wide average for logistic was 73.5%.

While accuracy is important, we contributed uniquely to cardiovascular ML research through our application of the clustering-first strategy. According to all the prior research we have seen on this data up to this point, we were the first to apply

this clustering-first approach. We believe that there is potential in this clustering-first approach improve accuracy of scores overall. However, in our analysis, this was only the case for half of our clusters. We believe that through more extensive hypertuning and testing of different models, we could have achieved better results on these underperforming clusters as well. That being said, we definitely believe that more experiments using this approach need to be done before we can conclude that clustering is a viable approach for medical analysis. While it did produce mostly better results locally - for this experiment - it might not work in every case. Thus, it might not be the most effective or ethical strategy to use when dealing with peoples' lives on the line. We recommend more people repeat this experiment with clustering first to verify that clustering is a dependable strategy.

6 Conclusion

As Cardiovascular Disease is one of the largest and most impactful diseases worldwide, our goal with this project was to predict how certain risk factors affect the possibility of developing a CVD. The root cause of CVD is not known, so doctors rely on a set of risk factors to predict whether a person is prone to develop a CVD. Our hope was that our novel clustering-first approach would result in higher accuracies when compared to current models that are being used for this same task.

While we did have to go through 2 iterations of model training, we were ultimately able to create pretty accurate and personalized models that did not factor in extreme patient values when determining general data relationships. Half of our models ended up exceeding the best model accuracy for the entire dataset. The other half were slightly lower than the entire dataset accuracy. However, the overall cluster-wide average still was greater than the entire dataset average, even if only by a marginal amount. Other than accuracy, another metric we decided to evaluate our models by was through an "Incremental" which is basically how much better the models performed than by just guessing "1' for every prediction. When it came to incremental, some of our cluster-specific models performed really well; for ex. cluster 2 had a 24% incremental, while the entire dataset incremental was only 7.46%. However, because cluster 4 had a really bad incremental, our overall incremental was affected quite a bit. Overall, though, our average of cluster-specific model incrementals still exceeded the entire dataset incremental.

Once we finished the second iteration of our model training, we created the pipeline shown in figure 2. These models are all deployed in this pipeline fashion and there is now a beta web app that users can try to get predictions on. Because our overall cluster-wide accuracy is still only 74%, it is not production-ready yet; for something as important as CVD diagnosis, we would ideally like to reach the 90% prediction accuracy threshold at least before we market it as a viable tool that doctors can use. As for next steps, we plan to look into cluster 4 a bit more to see why the incremental is so low. We would also like to do more parameter training and testing of different models on the clusters that yielded lower accuracies than the entire dataset accuracy. If we are able to continue improving the accuracy, that will be a good indication that this clustering-first approach is a good technique for medical analysis.

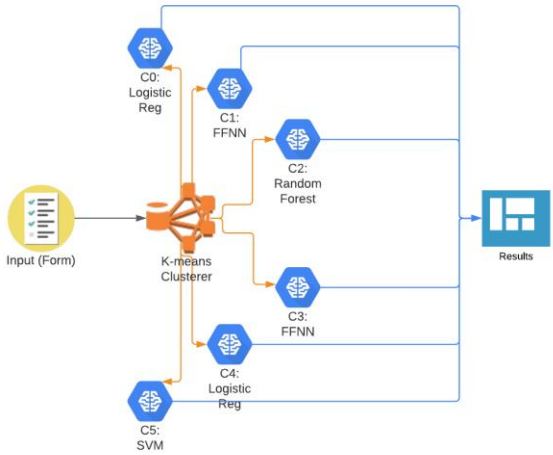

Figure 2: Finalized Pipeline

ACKNOWLEDGMENTS

We would like to thank our DSCI 303 Professor, Dr. Sano, our TAs, Cheng and Han, and everyone else who made this monumental accomplishment possible.

REFERENCES

[1] Krittanawong, C., Virk, H.U.H., Bangalore, S., Wang, Z., Johnson, K.W., Pinotti, R., Zhang, H., Kaplin, S., Narasimhan, B., Kitai, T. and Baber, U., 2020. Machine learning prediction in cardiovascular diseases: a meta-analysis. *Scientific reports, 10*(1), pp.1-11.

[2] Tripoliti, E.E., Papadopoulos, T.G., Karanasiou, G.S., Naka, K.K. and Fotiadis, D.I., 2017. Heart failure: diagnosis, severity estimation and prediction of adverse events through machine learning techniques. *Computational and structural biotechnology journal*, *15*, pp.26-47.

[3] World Heart Federation. 2019. *Roadmap for Heart Failure*. [online] Available at: <https://world-heart-federation.org/cvd-roadmaps/whf-global-roadmaps/heart-failure/> [Accessed 11 December 2021].

[4] World Heart Federation. n.d. *WHAT IS CARDIOVASCULAR DISEASE?*. [online] Available at: <https://world-heart-federation.org/what-is-cvd/> [Accessed 11 December 2021].

[5] World Health Organization. 2021. *Cardiovascular diseases (CVDs).* [online] Available at: <https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds)> [Accessed 11 December 2021].

[6] nhs.uk. 2018. *Cardiovascular disease.* [online] Available at: <https://www.nhs.uk/conditions/cardiovascular-disease/> [Accessed 11 December 2021].

[7] Felman, A., 2019. *What to know about cardiovascular disease*. [online] MedicalNewsToday. Available at: <https://www.medicalnewstoday.com/articles/257484> [Accessed 11 December 2021].

[8] Centers for Disease Control and Prevention. 2021. Heart Disease Facts. [online] Available at: <https://www.cdc.gov/heartdisease/facts.htm> [Accessed 11 December 2021].

[9] Işler Y., Kuntalp M. Combining classical HRV indices with wavelet entropy measures improves to performance in diagnosing congestive heart failure. *Comput Biol Med*. 2007;37:1502–1510.

[10] Asyali M.H. 2003. Discrimination power of long-term heart rate variability measures.

[11] Elfadil N., Ibrahim I. 2011. Self organizing neural network approach for identification of patients with congestive heart failure.

[12] Jovic A., Bogunovic N. Electrocardiogram analysis using a combination of statistical, geometric, and nonlinear heart rate variability features. *Artif Intell Med.* 2011;51:175–186.

[13] Melillo P., Fusco R., Sansone M., Bracale M., Pecchia L. Discrimination power of long-term heart rate variability measures for chronic heart failure detection. *Med Biol Eng Comput.* 2011;49:67–74.

[14] Yu S.-N., Lee M.-Y. Conditional mutual information-based feature selection for congestive heart failure recognition using heart rate variability. *Comput Methods Programs Biomed.* 2012;108:299–309
[15] Liu G., Wang L., Wang Q., Zhou G., Wang Y., Jiang Q. A new approach to detect congestive heart failure using short-term heart rate variability measures. PLoS One. 2014;9:e93399.
[16] Gharehchopogh F.S., Khalifelu Z.A. 2011. Neural network application in diagnosis of patient: a case study, Abbottabad.
[17] Alonso-Betanzos A., Bolón-Canedo V., Heyndrickx G.R., Kerkhof P.L. Exploring guidelines for classification of major heart failure subtypes by using machine learning. Clin Med Insights Cardiol. 2015;9:57–71
[18] Isler Y. Discrimination of systolic and diastolic dysfunctions using multi-layer perceptron in heart rate variability analysis. Comput Biol Med [accepted paper] 2016
[19] Roger V.L. The heart failure epidemic. *Int J Environ Res Public Health. 2010;*7:1807–1830.
[20] Amal Joby. "What is Logistic Regression? Learn When to Use it." *Learn Hub*, July 29, 2021 Description of LR algorithm applications and optimal application
[21] Pedregosa F. "Sci-Kit Learn: Machine Learning in Python" *Journal Of Machine Learning Research*, *https://scikit-learn.org/stable/about.html#citing-scikit-learn.* Accessed December 10, 2021.

CONTRIBUTIONS

For the project, Jamie Chen worked on EDA, k-Means, cluster summaries, logistic regression, and SVM; Bhavesh Shah worked on GMM, PCA, neural networks, and the Web App pipeline deployment; Griffin Coccari worked on EDA, random forests and KNN models. For the paper, Jamie Chen worked on the Abstract, Introduction, and Literature Review; Bhavesh Shah worked on most of the Methods (Clustering, Data, Strategy, Neural Nets) and Conclusion; Griffin Coccari worked on Methods (all the classification models) and Discussion.