# CLASSIFICATION OF PLACENTAL CHORIONIC SURFACE VASCULATURE NETWORK FEATURES USING MACHINE LEARNING TECHNIQUES

A THESIS

Presented to the Department of Mathematics and Statistics

California State University, Long Beach

In Partial Fulfillment

of the Requirements for the Degree

Master of Science in Mathematics

Option in Applied Mathematics

Committee Members:

Jen-Mei Chang, Ph.D.
James von Brecht, Ph.D.
William Ziemer, Ph.D.

College Designee:

Tangan Gao, Ph.D.

By Hike Hambarsoomian

B.S., 2014, University of California, Irvine

August 2017

ABSTRACT

**CLASSIFICATION OF PLACENTAL CHORIONIC SURFACE VASCULATURE NETWORK FEATURES USING MACHINE LEARNING TECHNIQUES**

**FORMAT: USE ALL CAPS AND BOLD FONT**

By

Hike Hambarsoomian

August 2017

The placenta is an organ that connects the fetus to the uterine wall of the mother. Analyzing the Placental Chorionic Surface Vasculature Network (PCSVN) has found measurable anatomical indicators that appear to differentiate placentas associated with high and low risks for Autism Spectrum Disorders (ASD). Since vessels and nerves share many guiding mechanisms when they are created and with autism being a neurodevelopmental disease, the idea that the vasculature is different in the case of autism is plausible. With this thesis, we aim to improve understanding of the PCSVN factors which are visible already at birth or even earlier to identify children that are associated with placentas at risk for ASD. Using an embedded feature selection method called Elastic Net we were able to reduce the number of features by selecting the 16 most important of the original 66 PCSVN features. Using Principal Component Analysis, the dimension of data set was further reduced to five features (nodes, tortuosity, thickness, branching angle, and growth). We then use these five features to cluster and classify the placentas into the high and low risk cohorts. Because early diagnosis and treatment reduces the effects of ASD considerably, this thesis can be used to identify the high-risk cluster earlier than before, allowing children to begin treatment as soon as the placenta is classified.

**ACKNOWLEDGEMENTS**

First and foremost, I would like to thank Dr. Jen-Mei Chang for taking me on and believing in my ability to take on this thesis. Thank you for helping create a project that allows me to gain a deeper understanding of the subject I love and enjoy.

I would like to acknowledge the works of Dr. Carolyn Salafia, Ruchit Shah and Terrie Giradi for without their dedication to the project data, code, and manual labor I would not have been able to even begin this project. I would aslo like to thank the remaining members of my committee, Dr. James von Brecht and Dr. William Ziemer, for being so patient and working with me to finish this paper.

Lastly I would like to thank my family for understanding my minimal availi- bility and letting me work deep into the night and my friends for being a stress relief and giving me the strength to power through at the toughest of times. Specifically, Debbie (for allowing me to pick her brain and helping me work out problems when I was stuck), Diana (for her emotional support and giving me a laugh when needed), and Jake and Mimsy for sitting through multiple explanations of my project and learning way more about placentas than they ever wanted to know.

## TABLE OF CONTENTS

## LIST OF TABLES

## LIST OF FIGURES

# CHAPTER 1

# INTRODUCTION

## The Placenta

The placenta is a complex organ that connects the fetus to the uterine wall of the mother, that plays a crucial role in the outcome of the pregnancy. Acting as an immune barrier to protect the fetus from antigen attacks by the maternal system, the placenta is also responsible for oxygen and carbon dioxide exchange, waste removal, and supplying nutrients to the fetus. Being the sole provider of these essential substances for the fetus during its entire development in the womb, the proper functioning of the placenta is not only crucial for a healthy pregnancy, but analyzing placental properties gives hints about possible abnormalities of the fetus [1]. Figure 1 from [2] shows the schematic view and structure of the placenta.

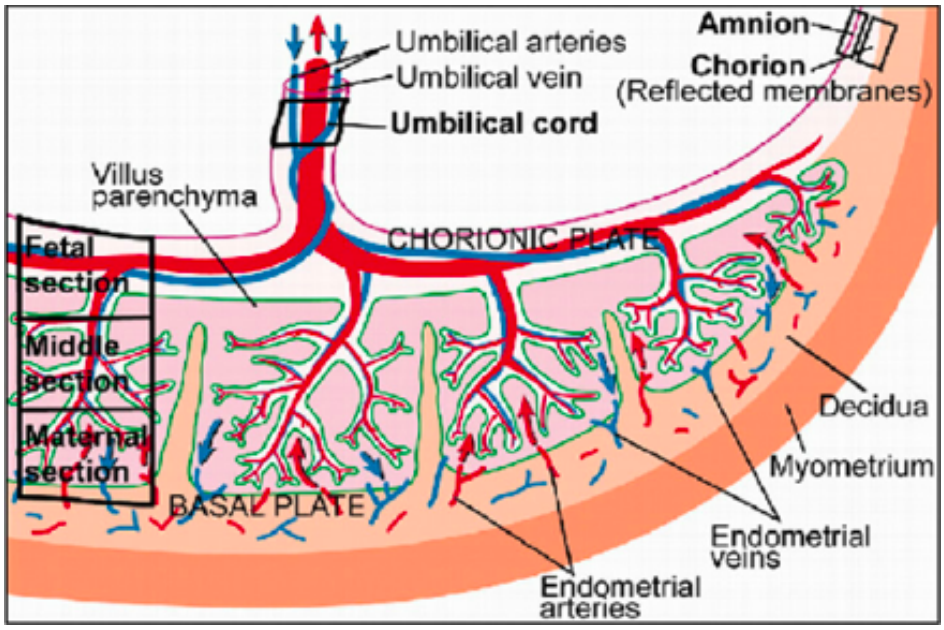



**FIGURE 1. Schematic view of chorionic plate.**

Since it contains both fetal and maternal blood, one method of using the placenta to identify abnormalities is to analyze the placenta with respect to diseases in the blood such as rubella and sickle cell disease in the mother. First indications of many other diseases and disorders show in

the form of abnormalities in the placenta. These abnormalities of the placenta can range from color and size to inflammation and lesions on the umbilical cord and to the amplification of the placental tissue [3].

The placental chorionic surface vascular network (PCSVN) is a major aspect of the placenta that has not been thoroughly studied due to the lack of reliability in the extraction of its features. With manual extraction of the PCSVN features, issues such as arteries crossing over veins hinders the ability to present exact extraction. A recent study [4], that has built the groundwork for this thesis, used geometric figures on the PCSVN for their analyses. For example, by studying the vascular structures of placentas, two measurable anatomical indicators that appear to differentiate placentas associated with high and low risks for autism spectrum disorder were identified from a general sample of placentas. The differences were seen in transport efficiency of the measured venous/arterial network of the placenta and the relative transport efficiency ratio of the venous network over the arterial network of the same placenta. Placentas with lower transport efficiency were more likely to be a part of the at risk group [5]. It is known that vessels and nerves share many guiding mechanisms when they were created and since autism is a neurodevelopmental disease, the idea that the vasculature is different in the case of autism appears plausible [4].

## Autism Spectrum Disorder

First described by psychiatrists Kanner and Asperger in the early 1940s, Autism spectrum disorders (ASD) are a heterogeneous group of lifelong neurodevelopmental disorders, comprising of autism, Asperger syndrome, and pervasive developmental disorder not otherwise specified [6]. Those affected by ASD tend to show social and communicational deficits along with restrictive and inappropriate repetitive behavior which usually start during the first three years of life [7], with the severity varying with intellectual ability ranging from average to severely disabled [8].

The focus of this thesis work is to develop an improved understanding of the PCSVN factors that are associated with placentas with a high-risk for ASD. Our experiments were conducted on

two groups of placentas, one that is population based and another that consists of placentas that are associated with families where there is already a child diagnosed with ASD as these children are at a higher risk. While there is no known cure for autism spectrum disorders, early intervention can help to at least reduce the effects considerably [9] and therefore substantially ease the life of the affected person and family. Standard screening tests of autism based on behavioural analysis can help to identify risks at the age of eighteen months [10] and a stable diagnosis is usually possible from the age of 2 years [11], whereas it would be desirable to find biomarkers for ASD which are visible already at birth or even earlier.

This thesis will proceed as follows. In Chapter 2, beginning on page 4, we will describe the data sets considered in this work including the way PCSVN images were collected and the process of measuring their respective geometric features. In Chapter 3, beginning on page 7, we analyze the relationship between these features using machine learning techniques such as Elastic Net for feature selection and Principal Component Analysis for dimensionality reduction. The accompanying MATLAB codes are provided in the appendix on page 33. We will show in Chapter 4, beginning on page 22, the prominent features resulting from the Elastic Net approach and Principal Component Analysis. Placentas from both data sets were then clustered and classified into subgroups based on further refined relationships.

The results generated in this thesis give promise to the potential of using PCSVN for early assessment and detection of ASD risks. The simplicity of taking a digital photo of the placenta hours after delivery makes this risk assessment method attractive in clinical setting as it bypasses costly and cumbersome medical procedures; however, the lack of an automated vessel extraction routine makes this method currently difficult to implement.

# CHAPTER 2

## DATA

### t

The placentas studied in this thesis came from two independently collected cohorts. The first data set contains 89 placentas from the Early Autism Risk Longitudinal Investigation (EARLI) that ”enrolls and follows a large group of mothers of children with autism at the start of another pregnancy and document the newborn childs development through three years of age” [12]. According to a study conducted by the Baby Siblings Research Consortium, compared to the total population, siblings of autistic children are at a higher risk (18.7%) of also being diagnosed with ASD with the risk increasing (32.2%) if the child has more than one older sibling with ASD [13]. Thus, the EARLI group will represent the high-risk cohort of this thesis. The second data set contains 201 placentas from the National Childrens Study (NCS) which is a ”long-term study of children designed to study environmental influences on child health and development” [14]. The children participating in this study participate without a bias towards risk and diagnoses in autism, representing the normal population with pregnancies at unknown risk for ASD and in this thesis are used as the low-risk cohort. The 290 observations, EARLI and NCS combined, will create the rows of our data matrix $X_0$.

For all studies, pictures of the fetal surface of the placentas were taken either at delivery or upon pathology evaluation with fresh tissue, keeping a consistency between the two cohorts. The placental chorionic surface vascular network on the surface was hand-traced with a protocol that identified different colors and pencil sizes to different vessel thicknesses, as can be seen in Figure 2.

Each traced image was separated from the original image and fed through an automated algorithm, written in MATLAB, that produced a skeleton graph of the network. From the skeleton graph, 66 numerical values were calculated with eight being shape-related (e.g. area, perimeter),

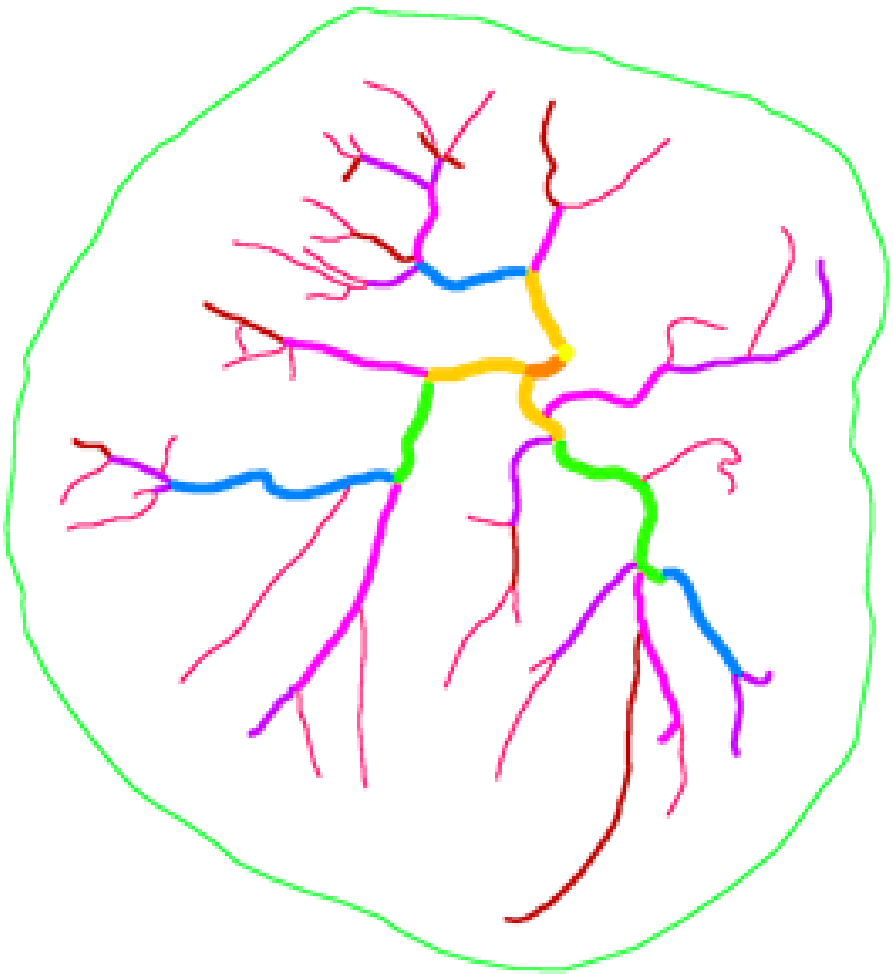


**FIGURE 2. Traced vessels.**

29 relating to the features of the arterial network (e.g., number of arterial branch points) and the remaining to the features of the venous network (e.g., number of venous branch points). To avoid redundancy in variables, such as the previous two examples, we will be focusing solely on the eight shape-related variables and the 29 features related to the arterial network.

The remaining 37 numerical values will represent the columns of our data set, $X_1$ with each placenta being a separate row. The data set is then mean subtracted to remove any bias that could have occurred in the data collection process. Some of these features include ”Surface Area” which is the surface area of the arterial and venous networks as determined from the traced images such

as Figure 2, measures of ”Thickness” such as the thickness of each placentas arterial/venous networks and values of ”Angles” which measure the branching angles of each new generation of branches, as depicted in Figure 3 [15]. Since each of these features have multiple variables (e.g., mean, standard deviation) it is clear to see that some of the values are closely related.

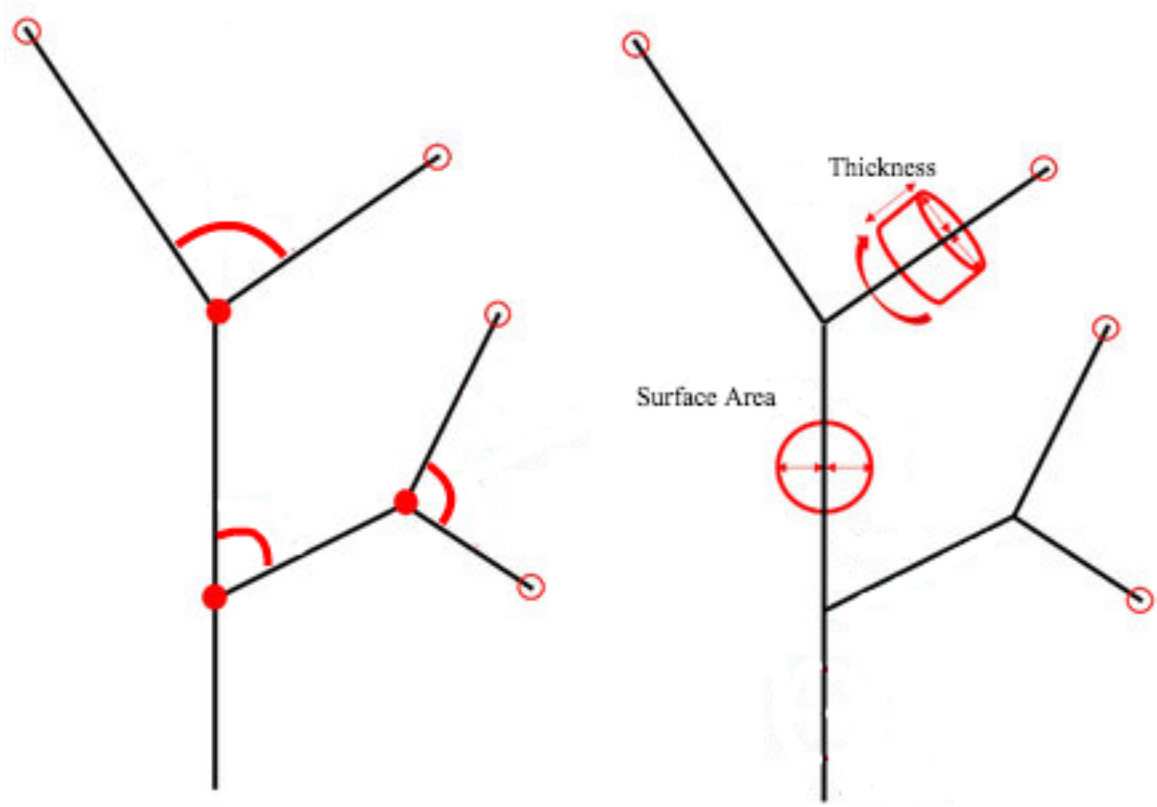



**FIGURE 3. Branching angles, surface area, thickness.**

# CHAPTER 3

# METHODS

In this chapter, we will describe the methods used to improve understanding of the PCSVN factors that are associated with placentas with a high-risk for ASD. We will use a feature selection method called Elastic Net on the data described in Chapter 2, with an aim to automatically select a subset of the PCSVN features. We will apply Principal Component Analysis to the variables selected from Elastic Net, to select an even smaller set of variables that are linearly independent. We will then study the placentas based on the results obtained from Principal Component Analysis using K-Means clustering and identify the placentas at risk of ASD using K-Nearest Neighbors.

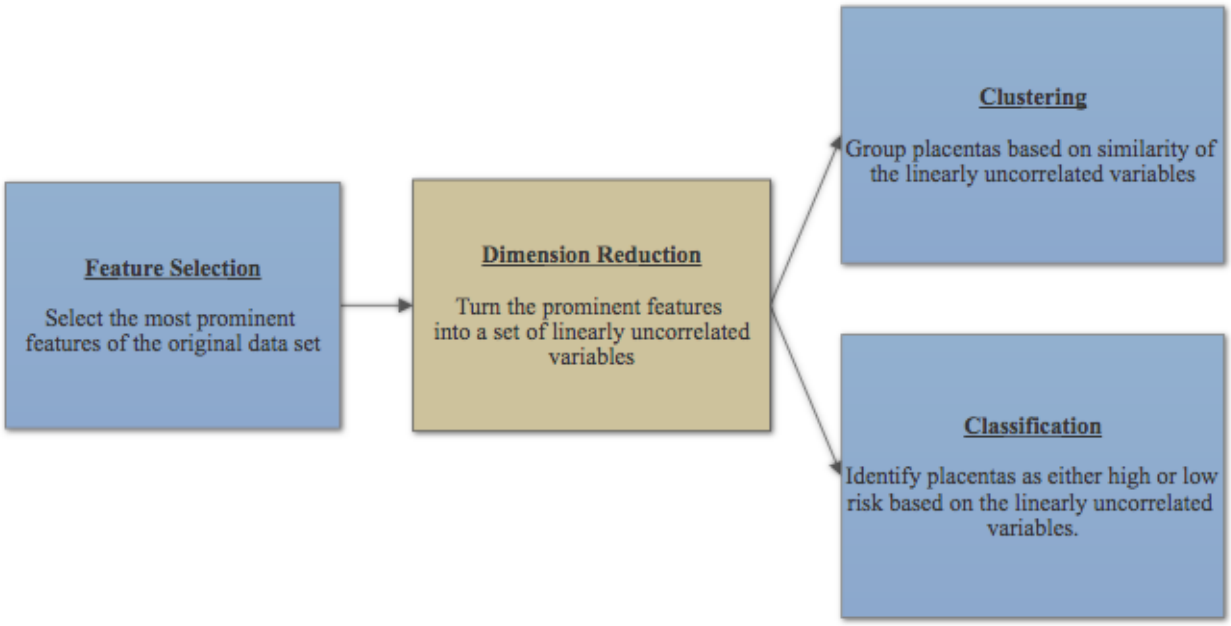



**FIGURE 4. Procedure outline.**

## Feature Selection

Feature selection, also known as variable selection, is a process of reducing or finding the most meaningful inputs for processing and analysis. Feature selection helps to improve the pre-

dictive nature of a classification algorithm by producing a list of core features that explains the underlying process used to generate the data [16]. Feature selection retains a subset of features by eliminating attributes from data that do not contribute to the accuracy, producing a data matrix that is more interpretable than the original set [17]. Feature selection methods can be categorized into three general categories: filter, wrapper, and embedded, depending on how the selections are combined.

The filter method is the most direct of the three. This method is primarily used as a preprocessing step as the selection of the prominent features is done independently from any machine learning algorithms. Filter methods typically consider the relevance of features independently, selecting features that are correlated and dependent [18]. This causes a selection of redundant variables because the relationship between variables is ignored [19], creating an overfitting of the data whilst failing to select the most useful features.

The second category of feature selection methods is the wrapper methods. In wrapper methods, a subset of features is used to train a model for feature selection. What is learned in the training set leads to features being added or removed from the subset, creating multiple combinations that keep the best performing feature at each iteration, evaluated and compared to other combinations. This ultimately reduces the problem to a search problem which, unlike filter methods, identifies possible interactions between variables [20]. Though wrapper methods may provide the most useful features, they are prone to overfitting and computationally expensive [18].

The final class is the class of embedded methods, which will be used in this thesis. Embedded methods combine some of the qualities of the methods mentioned earlier. This class is similar to the wrapper methods but is less computationally expensive and less prone to overfitting the dataset. Since algorithms perform the selection and classification simultaneously these methods tend to learn which features best contribute to the data set while features are selected [18]. However, the disadvantage of embedded methods is that it is computationally expensive compared to

filter methods. An example of an algorithm that falls into this class is the Elastic Net, which incorporates many aspects of the Ridge Regression and Least Absolute Shrinkage and Selection Operator (LASSO) algorithms. We will discuss the Elastic Net algorithm in further detail next.

In summary, algorithms that use the filter approach are generally efficient and can be combined with most learning algorithms while a wrapper approach can usually provide a good performance for the learning algorithm and the embedded methods share the advantages of the filter and wrapper methods.

**Elastic Net**

The Elastic Net feature selection method, developed by Zou and Hastie in 2005, ”produces a space model with good prediction accuracy while encouraging a grouping effect” [21]. We begin by considering the regression

$$y = \beta_1 x_1 + \beta_2 x_2 + ... + \beta_p x_p + \varepsilon \tag{3.1}$$

where $y$ is a classification label vector that provides the ground truth labels for the placentas in the data set and the values $x_1, x_2, ..., x_p$ are given and represent the columns of the data where $p$ is the number of variables in the data set. Leaving the computation and minimization of the minimization coefficients $\beta_1, \beta_2, ..., \beta_p$.

To begin the understanding of how Elastic Net came to be, we must first look at the ordinary least squares (OLS) method which finds the best-fitting curve to a given set of points by minimizing the sum of squares of the residuals. The OLS problem solves the minimization problem

$$F(\beta) = \underset{\beta}{\operatorname{argmin}} \|y - X\beta\|_2^2 = \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 \tag{3.2}$$

for a given data matrix $X$ where $N$ is the number of observations, to estimate the unknown param-

eters, $\beta_i$, in equation (3.1). To solve this minimization problem,

$$
\begin{aligned}
F(\beta) &= \underset{\beta}{\operatorname{argmin}} \|y - X\beta\|_2^2 = \underset{\beta}{\operatorname{argmin}} (y - X\beta)^T (y - X\beta) \\
&= \underset{\beta}{\operatorname{argmin}}\, y^T y - y^T X\beta - \beta^T X^T y + \beta^T X^T X\beta \\
&= \underset{\beta}{\operatorname{argmin}}\, y^T y - 2X^T y\beta - +\beta^T X^T X\beta \\
\frac{\partial F}{\partial \beta} &= -2X^T y + 2X^T X\beta = 0 \\
X^T X\beta &= X^T y \\
\beta &= (X^T X)^{-1} X^T y
\end{aligned}
$$

Thus, the optimal beta value is given by $\beta_{OLS} = (X^T X)^{-1} X^T y$. According to the Gauss Markov theorem, OLS estimates have the smallest mean squared error among linear estimators. OLS often does poorly in both prediction and interpretation because a single outlier can weigh heavily and affect the accuracy. Penalization techniques have been proposed to improve OLS; with a trade-off of a little bias the estimator will result in a larger reduction in variance [17].

In 1970, Hoerl and Kennard [22] proposed that the potential instability in the OLS estimator, such that $\beta_{OLS}$ could be improved by shrinking the regression coefficients by adding a small constant value, resulting in the Ridge regression estimator

$$
R(\beta) = \underset{\beta}{\operatorname{argmin}} \|y - X\beta\|_2^2 + \lambda \|\beta\|_2^2 \tag{3.3}
$$

where $\lambda \geq 0$ acts as a parameter that controls the amount of shrinkage in the regression coefficients by imposing a penalty on their size; the larger the value of $\lambda$, the greater the amount of

shrinkage [17]. An equivalent way to write the Ridge regression estimator

$$R(\beta) = \underset{\beta}{\operatorname{argmin}} \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \|\beta\|_2^2 \tag{3.4}$$

From OLS we know, $X^T X\beta = X^T y$. We use this to find the minimizer $\beta$ for Ridge regression by setting $X^* = \begin{pmatrix} \mathbf{X} \\ \sqrt{\lambda}\mathbf{I} \end{pmatrix}$ and $y^* = \begin{pmatrix} \mathbf{y} \\ \mathbf{0} \end{pmatrix}$ so that

$$\begin{aligned}
R(\beta) &= \underset{\beta}{\operatorname{argmin}} (y - X\beta)^T (y - X\beta) + \lambda \beta^T \beta \\
&= \underset{\beta}{\operatorname{argmin}} (y^* - X^*\beta)^T (y^* - X^*\beta) \\
X^{*T} X^* \beta &= X^{*T} y^* \\
(X^T X + \lambda I)\beta &= X^T y \\
\beta &= (X^T X + \lambda I)^{-1} X^T y \\
\beta_{Ridge} &= (X^T X + \lambda I)^{-1} X^T y
\end{aligned}$$

Ridge regression minimizes the residual sum of squares subject to a bound on the $L_2$-norm of the coefficients. As a continuous shrinkage method, ridge regression achieves its better prediction performance through a biasvariance trade-off making it a preferred method to the OLS method. However, it cannot produce a desired level of selection since variables are not removed, even if deemed unimportant [21]. This is where Least Absolute Shrinkage and Selection Operator (LASSO) becomes the preferred method.

The LASSO technique, developed by Tibshirani in 1996 [23], uses an $L_1$ since minimization of the $L_1$ norm involves taking values of the minimization coefficient vector to zero, instead of the Euclidean norm in Ridge regression (3.4). The new estimator is given by the optimization problem

$$L(\beta) = \underset{\beta}{\operatorname{argmin}} \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \|\beta\|_1 \tag{3.5}$$

The optimization problem can be cast in the equivalent form

$$L(\beta) = \underset{\beta}{\operatorname{argmin}} \|y - X\beta\|_2^2 + \lambda \|\beta\|_1 \tag{3.6}$$

with the same properties of $\lambda$ from Ridge regression. This subtle but important change penalizes the absolute values of the coefficients, introducing the ability for a sparse solution with some coefficients being shrunken all the way to zero unlike Ridge regression. With the penalty performing a sort of continuous variable selection using the $L_1$-norm for shrinkage it is intuitive why the method is called "Least Absolute Shrinkage and Selection Operator".

The minimization of (3.6) can be done similarly to that of (3.4) with a penalty term. We assume orthonormal columns in $X$, i.e., $X^T X = I$. With this,

$$\begin{aligned}
L(\beta) &= \underset{\beta}{\operatorname{argmin}} \|y - X\beta\|_2^2 + \lambda \|\beta\|_1 \\
&= \underset{\beta}{\operatorname{argmin}} y^T y - y^T X\beta - \beta^T X^T y + \beta^T X^T X\beta + \lambda \sum_{j=1}^{p} |\beta_j| \\
&= \sum_{j=1}^{p} \underset{\beta}{\operatorname{argmin}} (y^T y)_j - (y^T X)_j \beta_j - \beta_j (X^T y)_j + \beta_j^2 + \lambda |\beta_j| \\
&= \begin{cases} \operatorname{argmin}_\beta (y^T y)_j - 2(X^T y)_j \beta_j + \beta_j^2 + \lambda \beta_j & \beta_j > 0 \\ \operatorname{argmin}_\beta (y^T y)_j - 2(X^T y)_j \beta_j + \beta_j^2 - \lambda \beta_j & \beta_j < 0 \end{cases} \\
\beta_j &= \begin{cases} X^T y - \frac{1}{2}\lambda & \beta_j > 0 \\ X^T y - \frac{1}{2}\lambda & \beta_j < 0 \end{cases} \\
\beta_{LASSO_j} &= \begin{cases} X^T y - \frac{1}{2}\lambda & \beta_j > 0 \\ X^T y - \frac{1}{2}\lambda & \beta_j < 0 \end{cases}
\end{aligned}$$

LASSO improves prediction accuracy of OLS by shrinking or setting some coefficients to zero and acts as a variable selection to isolate the most influential subset of variables, making it typically a more effective method than Ridge Regression.

LASSO does have its share of shortcomings. First, for $p > N$ LASSO selects at most $N$ variables making it a possibility that some important variables are not selected. Second, in the case where there are correlations within variables, LASSO tends to blindly select only one variable from the group with the possibility of not selecting an important one. Lastly, for $N > p$ situations, if correlations between variables exist, Ridge regression dominates LASSO [17, 21]. Because of these shortcomings, Ridge regression is the ideal choice for our correlated data. But, a more effective method than both LASSO and Ridge Regression, especially when the predictors are highly correlated, is Elastic Net.

Zou and Hastie made a compromise between Ridge Regression and LASSO with Elastic Net. Similar to the LASSO, Elastic Net simultaneously does automatic variable selection and continuous shrinkage, but can select groups of correlated variables [21]. For any fixed nonnegative $\lambda_1$ and $\lambda_2$, the optimal parameter is obtained via the optimization problem

$$E(\beta) = \underset{\beta}{\operatorname{argmin}} \|y - X\beta\|_2^2 + \lambda_1 \|\beta\|_1 + \lambda_2 \|\beta\|_2^2 \tag{3.7}$$

By defining an artificial data set, ($y^*$,$X^*$) such that $y^* = \begin{pmatrix} \mathbf{y} \\ \mathbf{0} \end{pmatrix}$ and

$X^* = \frac{1}{\sqrt{1+\lambda_2}} \begin{pmatrix} \mathbf{X} \\ \sqrt{\lambda_2}\mathbf{I} \end{pmatrix}$ Elastic Net can be written as a LASSO estimate

$$E(\beta^*) = \underset{\beta^*}{\operatorname{argmin}} \|y^* - X^*\beta^*\|_2^2 + \gamma \|\beta^*\|_1 \tag{3.8}$$

by performing,

$$
\begin{aligned}
E(\beta) &= \underset{\beta}{\operatorname{argmin}} \|y - X\beta\|_2^2 + \lambda_1\|\beta\|_1 + \lambda_2\|\beta\|_2^2 \\
&= \underset{\beta}{\operatorname{argmin}} \|y - X\beta\|_2^2 + \|0 - \lambda_2\beta\|_2^2 + \lambda_1\|\beta\|_1 \\
&= \underset{\beta}{\operatorname{argmin}} \left\| \begin{pmatrix} \mathbf{y} \\ \mathbf{0} \end{pmatrix} - \begin{pmatrix} \mathbf{X} \\ \sqrt{\lambda_2}\mathbf{I} \end{pmatrix} \beta \right\|_2^2 + \lambda_1\|\beta\|_1 \\
&= \underset{\beta}{\operatorname{argmin}} \left\| \begin{pmatrix} \mathbf{y} \\ \mathbf{0} \end{pmatrix} - \frac{1}{\sqrt{1+\lambda_2}} \begin{pmatrix} \mathbf{X} \\ \sqrt{\lambda_2}\mathbf{I} \end{pmatrix} \sqrt{1+\lambda_2}\beta \right\|_2^2 + \frac{\lambda_1}{\sqrt{1+\lambda_2}} \|\sqrt{1+\lambda_2}\beta\|_1 \\
&= \underset{\beta^*}{\operatorname{argmin}} \|y^* - X^*\beta^*\|_2^2 + \gamma\|\beta^*\|_1
\end{aligned}
$$

using where $\gamma = \dfrac{\lambda_1}{\sqrt{1+\lambda_2}}$ and $\beta^* = \sqrt{1+\lambda_2}\beta$ which could be minimized similar to the LASSO minimization above.

Elastic Net could be written as

$$
E(\beta) = \underset{\beta}{\operatorname{argmin}} \|y - X\beta\|_2^2 + \lambda((1-\alpha)|\beta|_1 + \alpha|\beta|_2^2 \tag{3.9}
$$

where $\alpha = \dfrac{\lambda_2}{\lambda_1 + \lambda_2}$ for $\alpha \in [0,1]$ with $(1-\alpha)|\beta|_1 + \alpha|\beta|_2^2$ being called the penalty of Elastic Net. The first term of the penalty encourages a sparse solution, while the second term encourages highly correlated features to be averaged. When $\alpha = 0$ the $E(\beta)$ reduces to $L(\beta)$ and with $\alpha = 1$ the $E(\beta)$ reduces to $R(\beta)$. The improved estimator (3.9) can be used to balance out the strengths and overcome the deficiencies set forth by either LASSO or Ridge Regression. This allows Elastic Net to have a better prediction performance than either of the methods in the presence of highly correlated predictors [21].

This proves particularly important when working with our sub data set, $X_1$, with $N = 290$

and $p = 37$, since the variables of the vascular network tend to exhibit strong correlations as described in Chapter 2. Applying Elastic Net for feature selection will provide weights for each minimization coefficient in equation (3.9) for each of the 37 variables in our data set. The LASSO aspect of Elastic Net will help zero out variables, leaving the most prominent features in a new sub data matrix $X_2$. These results are described in further details in Chapter 4.

## Feature Extraction

In this thesis, we distinguish between feature extraction from feature selection. Even though both methods seek to reduce the number of attributes or dimension of the data space , feature extraction methods do so by creating a new combination of attributes as opposed to feature selection methods that exclude variables that did not make the cut. Many of the features extracted from Elastic Net remain correlated, making it difficult to interpret the important underlying information. We will be performing a linear transformation with Principal Component Analysis to reduce the collinearity of those selected variables.

### Principal Component Analysis

Principal Component Analysis (PCA) represents and compresses data by finding an orthogonal coordinate system that preserves the maximum amount of variance as possible. It is used to find a lower-dimensional representation of the data distribution that preserves dominant and linearly uncorrelated features. These linearly uncorrelated features are used to create the scaled feature vectors called principal components. The principal components can be found by finding the eigenvector decomposition of the covariance matrix

$$C = \frac{1}{n-1} X^T X \tag{3.10}$$

or by finding the singular value decomposition of the data matrix.

The covariance matrix measures the variance of the variables where the eigenvectors represent the principal components. To find the principal components we first diagonalize $C$. Since the

$XX^T$ is square symmetric it is diagonalizable; hence, we have

$$X^T X = S\Lambda S^{-1}$$

where $\Lambda$ is a diagonal matrix of $X^T X$ and $S$ is a matrix of eigenvectors of $X^T X$ in its columns. Since $X^T X$ is symmetric, the eigenvectors are orthonormal. So, $S$ an orthogonal matrix with $S^{-1} = S^T$. By choosing a new orthogonal coordinate system given by the principal components, the change of coordinates is accomplished by applying the linear transformation

$$Y = S^T X.$$

Now, the covariance matrix of $Y$ is

$$\begin{aligned}
C_Y &= \frac{1}{n-1} YY^T \\
&= \frac{1}{n-1} (S^T X)(S^T X)^T \\
&= \frac{1}{n-1} S^T (XX^T) S \\
&= \frac{1}{n-1} (S^T S\Lambda S^{-1} S) \\
C_Y &= \frac{1}{n-1} \Lambda
\end{aligned}$$

In this new coordinate system, the covariance matrix is a diagonal matrix, meaning that all correlation between the variables has been eliminated. The eigenvalues on the diagonal of $\Lambda$ are ordered from the largest to the smallest such that the largest eigenvalue corresponds to the dimension that has the strongest correlation in the dataset.

The other way to perform PCA is through Singular Value Decomposition (SVD) on $X$. SVD is a factorization of a matrix into a number of constitutive components. A real $m$ x $n$ matrix $X$ with rank $r$ where $m \geq n$, has the decomposition,

$$X = U\Sigma V^T \tag{3.11}$$

where $U$ ($m$ x $m$) and $V$ ($n$ x $n$) are orthogonal matrices and $\Sigma$ ($m$ x $n$) is a diagonal matrix with nonnegative singular values, $\sigma_i$, on its diagonal. Suppose the matrices $U$, $\Sigma$, and $V$ exist such that $X = U\Sigma V^T$, then the two positive-definite symmetric matrices $XX^T$ and $X^TX$ can be written as follows

$$XX^T = U\Sigma V^T V\Sigma U^T = U\Sigma^2 U^T \tag{3.12}$$

$$X^TX = V\Sigma U^T U\Sigma V^T = V\Sigma^2 V^T \tag{3.13}$$

Multiplying the decompositions (3.14) and (3.15) on the right by $U$ and $V$, respectively, gives the two self-consistent eigenvalue problems.

$$XX^TU = U\Sigma^2 \tag{3.14}$$

$$X^TXV = V\Sigma^2 \tag{3.15}$$

The values on the diagonal of $\Sigma^2$ are ordered from the largest to the smallest, where $\sigma_1 \geq \sigma_2 \geq$ $\geq \sigma_n \geq 0$. These values are the eigenvalues, such that $\sigma_i$ is the $i^{th}$ largest eigenvalue, of equations (3.14) and (3.15). Finding the normalized eigenvectors of (3.16) and (3.17) will give the orthonormal column vectors for U and V [24]. The columns of $U$ will be used to create the principal components for Principal Component Analysis. Principal components have a particular ordering with each principal component pointing in the direction of maximal variance, orthogonal to the previous principal components, accounting for the maximal amount of variance not already accounted for by the previous principal components. These directions turn out to be precisely the eigenvectors of $X^TX$ [25].

To perform PCA with SVD, we can decompose $X$, our data set, by using (3.11), such that, $X = U\Sigma V^T$. By using (3.12) and defining the transformed variable

$$Y = U^TX,$$

we can rewrite the covariance matrix

$$
\begin{aligned}
C_Y &= \frac{1}{n-1} YY^T \\
&= \frac{1}{n-1} (U^T X)(U^T X)^T \\
&= \frac{1}{n-1} U^T * XX^T)U \\
&= \frac{1}{n-1} (U^T U \Sigma^2 U^T U) \\
C_Y &= \frac{1}{n-1} \Sigma^2
\end{aligned}
$$

This shows the connection between the two methods, since $\Lambda = \Sigma^2$.

Though both eigenvector decomposition of the covariance matrix and SVD of X can be used to perform PCA, SVD will be the tool of choice in dimensionality reduction. This is because the cost of computing the covariance matrix and its eigenvalue decomposition, $O(p^2 n + p^3)$, is much more than using SVD with a cost of $O(min(p^2 n, n^2 p))$ for $n$ observations and $p$ variables. In our case, this translates to a saving of $p^3$.

With PCA, we aim to eliminate data collinearity. Recall that $X_2$ is the sub data matrix after performing Elastic Net in our experiments. By factoring $X_2$ into its singular value decomposition, $X_2 = U \Sigma V^T$, the columns of $U$, after scaling by the singular values, give the principal components. The singular values, diagonal entries of $\Sigma$, are ordered from the largest to the smallest and are all nonnegative. The first singular value corresponds to the first principal direction and captures the most variance of the data set. The variances are found through the singular value, with the first term in the diagonal matrix being the largest. Since our data is mean subtracted to remove bias, the first principal component will travel along the most varying direction in the data space. To transform the data $X_2$ onto the new feature subspace, we will create the projection matrix $X_3$.

## Clustering

Clustering is a form of machine learning that groups data points in such a way such that objects in the same cluster are more similar to each other than to those in different clusters [26]. In this thesis, we will be using clustering to group similar observations, based on the principal features extracted from PCA. Since clustering is an unsupervised learning method, we will not consider the groundtruth of our data. We will be using a centroid-based clustering method called K-Means to group data points based on the distances from centroids.

### K-Means Clustering

K-Means aims to partition observations into a predetermined number of clusters, $k$, where each observation belongs to the cluster with the nearest mean. Given a set of $N$ points $\{x_n \in X_3\}_{n=1}^{N}$, the idea is to group points which are close to the same cluster, where closeness is defined in terms of the Euclidean distance, so we can summarize each cluster by the points of their centroid.

The central idea behind K-Means can be described in simple steps. Having predetermined the number of clusters that is needed based on prior knowledge of the data, randomly select $k$ points as cluster centroids. Calculate the distance from each centroid to each data point and group each data point to the closest centroid. Recalculate the cluster centroids under new memberships. The algorithm converges when there is no longer any update on centroid assignment. The algorithm improves the clusters and centroids iteratively by minimizing the cost function in each iteration,

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} \|x_n - \mu_k\|^2 \tag{3.16}$$

with respect to the parameter $\mu_k$, where $\mu_k$ is the arithmetic mean of the points in cluster $k$ when $J$ is minimized. Thus, $J$ is the sum of the distances between input points and their cluster centers [4].

Though K-Means is NP-hard in general Euclidean space $d$, even for 2 clusters [27], it is still

low in cost compared to other clustering methods, $O(ndk)$ with $n$ being the number of observations, $d$ being the dimensions, and $k$ being the number of clusters. K-means is, thus, one of the more efficient clustering methods. But, K-Means has its downfalls, including picking a $k$ value and difficulty in clustering complex data sets. Because we know how many clusters are in our data set, selecting the value for $k$ is not an issue. Performing feature and dimension reduction, using Elastic Net and PCA, simplifies the data set making it easier to cluster. This gives us the ability to visualize how the observations are grouped based on the features from PCA. To see whether these observations are in the correct clusters, we need to use a method of classification.

## Classification

Classification gives the ability to identify which class a new observation belongs to based on previous instances, which is found in a training set whose class is known. Classification is considered an instance of supervised learning, where a labeled training set is available [28]. In this thesis, classification with k-Nearest Neighbors will be implemented to identify which placentas are being classified correctly.

### K-Nearest Neighbors

k-Nearest Neighbors is a classification method that consists of the $k$ closest training examples to output a class membership. An observation is classified to the class that the majority of its $k$ nearest neighbors from the training set could be found in, using Euclidean distance as the distance metric because of the popularity in machine learning community. The training observations are randomly selected in the same feature space, the one that was found from PCA.

To understand how to choose the $k$ in k-Nearest Neighbors, it is best to understand what influence $k$ has in the algorithm. Cross-validation can be used to obtain estimates for parameters that are unknown. The idea of cross-validation is to randomly divide the data into disjoint subsets. Then for a fixed values of $k$, apply K-Nearest Neighbors to make predictions on each segment, and evaluate the accuracy. For various $k$, the steps above are repeated and the value with

the highest classification accuracy is selected as the optimal value for $k$. As $k$ increases to the number of observations, the training set finally becomes one complete class depending on the total majority

To implement k-Nearest Neighbors, we divide $X_3$ into a training and testing set. For each of the point in the testing set, $x_i \in X_{3_{Test}}$, using the ”majority voting” of its $k$ neighbors from the training set $X_{3_{Train}}$ to classify in which class each observation fits. However, a more frequent class of training points is likely to be common among the $k$ nearest neighbors which tends to dominate the prediction of the test point. Thus, a downfall to ”majority voting” occurs when there is a skewness in class distribution. [29]. Since we only have two classes and the difference in the number of observations is not too drastic, this does not become an issue for us. And so, k-Nearest Neighbors allows us to see which of the test points are accurately put in the correct class and which low risk placentas are considered to be high risk.

# CHAPTER 4

# RESULTS

This chapter presents the results of the methods described in Chapter three in four sections. Section one presents the results gathered with Elastic Net for feature selection. It will show the subset of the prominent features and give a basic understanding of each of them. In Section two, we will be showing how these variables are grouped for dimension reduction by describing each of the principal components from Principal Component Analysis. Section three shows how the placentas are clustered using K-Means and gives a visual representation of the centroids. Section four will display the classifications of placentas in either a high or low risk cohort. To begin we started with our original data matrix $X_0$ with 290 observations and 66 variables. To remove repetitive variables and improve accuracy we strategically removed variables associated with the venous network leaving a new sub data matrix $X_1$ with 290 observations but just 37 variables. We will use this data matrix as our data matrix for Elastic Net.

## Feature Selection

As discussed in the the Chapter 3, the privilege of implementing the Elastic Net algorithm would be to benefit from both Ridge regression and LASSO. The alpha parameter sets the degree of mixing between Ridge regression and LASSO. Elastic net is the same as LASSO when $\alpha = 0$ and as $\alpha$ approaches 1, Elastic Net approaches Ridge regression. Values in between these extremes will give a result that is a blend of the two with $\alpha = 0.5$ giving a solution that is evenly affected by Ridge Regression and LASSO. As was described in Chapter 3, we would ideally use Ridge regression for a dataset with more observations than variables. But, with Ridge regression incapable of zeroing out features we use Elastic Net. Using the built-in MATLAB function for Elastic Net flips the $(1-\alpha)$ and $\alpha$ coefficients from equation (3.11) thus by setting $\alpha = 0.001$ it becomes closely similar to Ridge regression. Using the minimization of LASSO, maximizes the positive regularization parameter lambda in equation (3.11). As the $\lambda$ increases, the number of

nonzero regression coefficients, $\beta_i$, decreases. We empirically decided to select the sixteen most prominent features of the data set which was resulted when $\lambda = 134.8143$. The sixteen features are listed and briefly described in Table 1. These sixteen features are listed and ranked on the magnitude of the regression coefficients in Table 1. The sixteen features will serve as the columns of $X_2$.

**TABLE 1. 16 Prominent Features from Elastic Net**

| Rank | Variables | Definition | Beta Value |
|---|---|---|---|
| 1 | 'A_MeanThickness' | Mean thickness of the arterial network | -2.3043 |
| 2 | 'A_MurrayL1FitError' | Mean error in terms of how far $(mother^{exp} - (d_1^{exp} + d_2^{exp}))$ is from 0 | -2.2071 |
| 3 | 'A_StdThickness' | Standard deviation of the thickness | -1.9395 |
| 4 | 'A_NumEndPoints' | Total number of arterial end points | 1.1369 |
| 5 | 'A_NumBranchPoints' | Total number of arterial branch points | 1.1300 |
| 6 | 'A_MurrayBranchesUsed' | Branches without endpoint | 1.1300 |
| 7 | 'A_NumGenerations' | Max number of times an artery branches | 1.1183 |
| 8 | 'A_MaxTortuosity' | Max of all the tortuosities | 0.8201 |
| 9 | 'A_MeanTortuosity' | Mean of all the tortuosities | 0.7374 |
| 10 | 'A_MeanAngle' | Mean of all the branching angles | 0.6348 |
| 11 | 'A_StdDevTortuosity' | Standard deviation of all the tortuosities | 0.5727 |
| 12 | 'A_Volume' | Volume of the arterial network | -0.4729 |
| 13 | 'A_ArcLength' | Arc length of the arterial network | 0.4260 |
| 14 | 'A_MedianAngle' | Median of all the branching angles | 0.2947 |
| 15 | 'A_KurtosisTortuosity' | Kurtosis of all the tortuosities of arterial branches | 0.1048 |
| 16 | 'A_MeanDistEndPointTo' | Kurtosis of all the tortuosities of arterial branches | 0.0383 |

## Feature Extraction

Implementing PCA on the set of sixteen features extracted five principal components (PC) that captured 66.26% of the datas variances. These five PCs are shown in Table 2. The elements in each of the principal components with the largest magnitude are linearly independent from any of the other principal components. These principal components will be used to create the new data set $X_3$ with 290 observations and 5 columns which will be used for clustering and classifica-

tion.

**TABLE 2. Principal Components**

| Rank | Features (Variance Captured) | PC1 18.82% | PC2 14.53% | PC3 14.03% | PC4 10.56% | PC5 8.32% |
|---|---|---|---|---|---|---|
| 1 | MeanThickness | 8.0900 | -0.2576 | **-13.7455** | 1.2831 | -0.1504 |
| 2 | MurrayL1FitError | 11.4156 | -0.8427 | -10.7496 | -0.1051 | 0.3620 |
| 3 | StdThickness | 7.9417 | -1.2600 | **-13.8620** | 0.3306 | 1.3209 |
| 4 | NumEndPoints | **-16.2302** | -2.1996 | -2.7430 | 1.4168 | 0.9118 |
| 5 | NumBranchPoints | **-16.2478** | -2.1843 | -2.7652 | 1.3696 | 0.9080 |
| 6 | MurrayBranchesUsed | **-16.2478** | -2.1843 | -2.7652 | 1.3696 | 0.9080 |
| 7 | NumGenerations | -12.1303 | -1.1140 | -2.2685 | 2.7161 | -2.6842 |
| 8 | MaxTortuosity | -5.3928 | **14.9208** | -2.9881 | -4.4932 | -0.7518 |
| 9 | MeanTortuosity | -0.6993 | **13.4199** | -1.5856 | -0.1646 | 2.9513 |
| 10 | MeanAngle | 2.2398 | 7.7866 | 2.0259 | **13.1834** | 1.4548 |
| 11 | StdDevTortuosity | -1.6064 | **15.7747** | -2.1969 | -3.7158 | 0.5720 |
| 12 | Volume | -3.4578 | -2.4526 | **-15.0841** | 2.8693 | 1.3643 |
| 13 | ArcLength | -13.9789 | -2.6781 | -5.4785 | 1.6854 | 2.2004 |
| 14 | MedianAngle | 2.1530 | 7.0529 | 2.4646 | **13.8830** | 1.2744 |
| 15 | KurtosisTortuosity | -6.3551 | 9.6260 | -3.3273 | -5.4349 | -2.7048 |
| 16 | DistEndPointToPerim | -0.6021 | 1.1053 | -2.6975 | 3.5703 | **-15.8686** |

**Principal Component 1 (PC1) - Nodes**

Each vascular network has two types of nodes. The first is an end node which is the terminal point of the vessel. The second is a branch node which splits into multiple vessels. The three elements with the largest magnitude of the first principal component were ”Number of Endpoints”, ”Number of Branch Points”, and ”Murray Branches Used”. As was defined in Table 1, the element ”Number of Endpoints” and ”Number of Branch Points” are the total number of end and branch nodes in the vascular system of each placenta, respectively. ”Murray Branches Used” is the number of branches that do not have an end node which could be evaluated by taking the difference between the total number of branches and the number of end nodes. Thus all three of these elements have a relation to either branch of end nodes in the vascular network.

**Principal Component 2 (PC2) - Tortuosity**

Tortuosity is defined as the measure of how much a curve twists which is given by the ratio between the length of the curve (arc length) and the distance between the starting and end points of the curve [30]. The three elements with the largest magnitude of the second principal component were ”Max Tortuosity”, ”Mean Tortuosity” and ”Standard Deviation of Tortuosity”. By the definition in Table 1, these elements give the maximum, mean, and standard deviation of the arterial tortuosities in the vascular network, respectively.

By definition, tortuosity has no relation to the number of nodes in the vascular network making this principal component independent to any of the elements of Principal Component 1.

**Principal Component 3 (PC3) - Thickness**

The word thickness here is being treated as the diameter of the vessels. The three elements with the largest magnitude of the third principal component were ”Mean Thickness”, ”Standard Deviation of Thickness”, and ”Volume”. By the definition in Table 1, these elements give the mean and standard deviation of the vascular thickness and volume gives the sum of all the arterial vessel volumes. Treating the vessels as cylinders, volume of each artery can be calculated by

$$Volume = \pi(\sum_{i=1}^{N}(\frac{diameter_i}{2})^2 * (arclength)_i)$$

where $N$ is the number of branches in the network.

The three variables in this principal component have no relation to the number of nodes or tortuosity in the arterial network,making this principal component independent to both Principal Components 1 and 2.

**Principal Component 4 (PC4) - Branching Angle**

A branching angle is used to capture the instantaneous growth of each branch node. The two elements with the largest magnitude of the fourth principal component were ”Mean Angle” and ”Median Angle”. By the definition in Table 1, these elements give the mean and median branching angles of all the arterial vessels. Branching angle is defined as the angle between line segments created between the branch node and the fourth pixel of each branched vessel. The fourth

pixel was an empirical decision used to give a better interpretation of the instantaneous change compared to using the end node of the branches. The calculation of branching angle is independent to a Principal Components 1, 2 and 3.

**Principal Component 5 (PC5) - Growth**

The element with the single largest magnitude of the fifth principal component is ”Mean Distance from End Point to Perimeter.” By the definition in Table 1, this element gives the average distance between the end point to the nearest point on the placental chorionic surface boundary. This variable calculates the growth such that the larger the value, the further away the vascular network is to the boundary. This element can be calculated independently from any of the previously mentioned principal components. The five principal components are linearly independent.

## Clustering

Implementing K-Means on the data points in the projected space allows us to cluster similar data points regardless of them being classified as high-risk for ASD or not. This combines observations most similar to each other based on branching, thickness, tortuosity, branching angle, and growth. Though clustering is done in a 5-dimensional space, in order to visualize the clusters Figure 6 will be shown in a 3D plot using the first three principal components.

The centers of the clusters are given as $[1.6663, -0.0099, -0.0375, -0.0485, 0.0366]$ for low-risk and $[-1.9399, 0.0115, 0.0437, 0.0565, -0.0426]$ for high-risk shown in green. The placenta pictured in Figure 6(a) is the closest to the centroid of the low-risk ASD group on the plot with a Euclidean distance of 0.8515. The placenta in Figure 6(b) is the closest to the high risk centroid on the plot with a Euclidean distance of 1.1840.

As we can see there are differences between the placentas for each of the principal component. The traced placenta associated with low-risk for ASD has slimmer arteries that curve less (Thickness=-0.0375, Tortuosity=-0.0099), than that of the high-risk (Thickness=0.0437, Tortu-

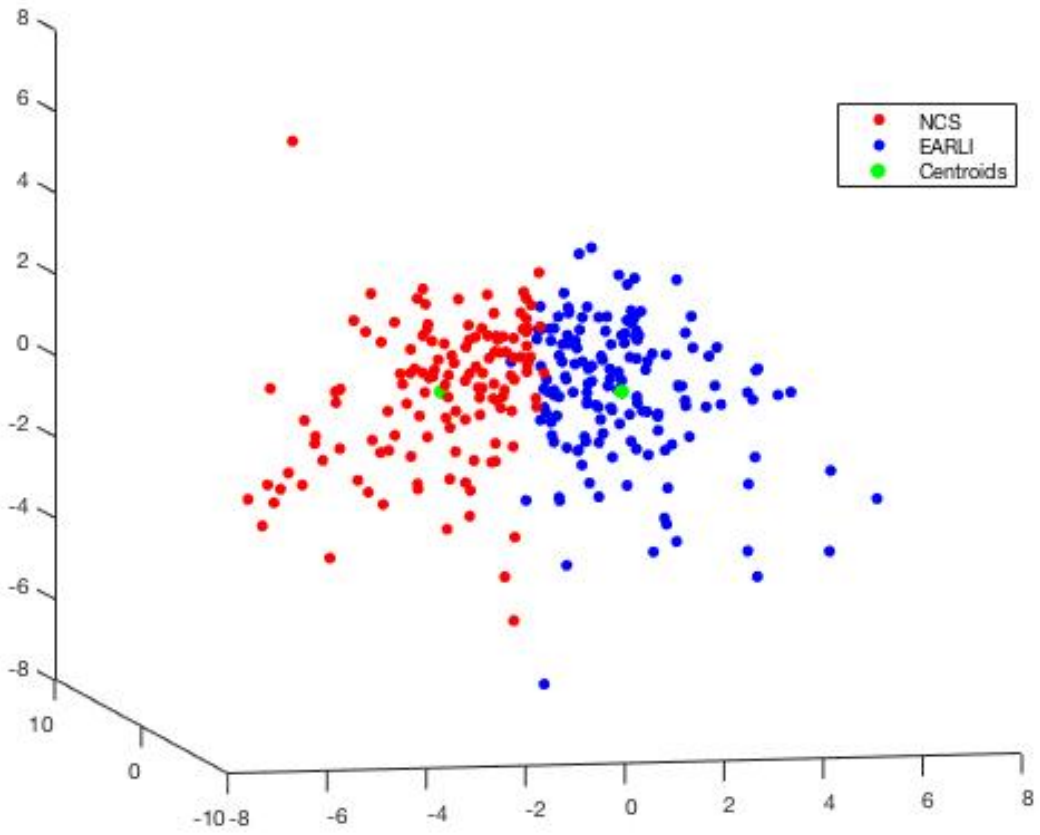


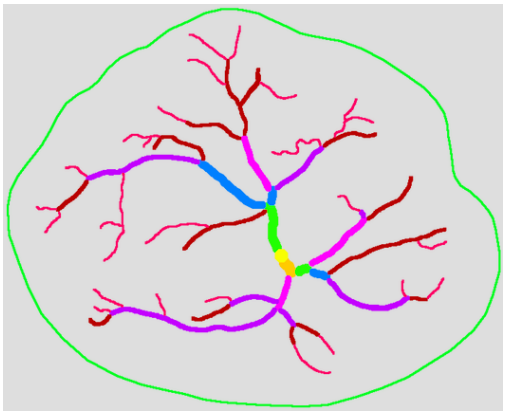

**FIGURE 5. Cluster plot in first 3 principal components.**

osity= 0.0115). The tracing of a PCSVN associated with high-risk for ASD has arteries growing to the edge of the placenta with larger branching angles (Growth=-0.0485, Branching Angle=0.0565) compared to the low-risk (Growth=-0.0426, Branching Angle=0.0565).

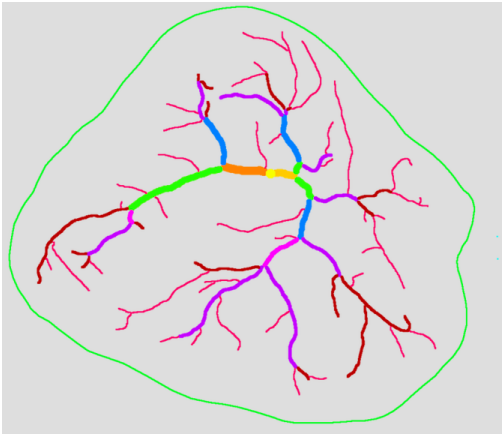
**TABLE 3. k-Means Clustering**

| Class | Classified as NCS | Classified as EARLI | Accuracy |
|---|---|---|---|
| NCS | 117 | 84 | 58.21% |
| EARLI | 17 | 72 | 72.72% |

This table shows that k-Means correctly clustered 65.17% of the observations. The remaining 34.83% represent those observations that were more closely related to the opposing class.



**(a) Low risk centroid.**



**(b) High risk centroid.**

**FIGURE 6. Centroids of each cluster.**

## Classification

Implementing K-Nearest Neighbors would classify our observations as discussed in Chapter 3. Since our data was skewed to have more NCS data points than EARLI, the testing sets would likely consist of more points from NCS. A distance-weight was added to offset this where training points further from the testing point had a smaller weight in deciding the class of said testing point

We chose to use 1 nearest neighbor to perform our Classification since a huge improvement did not occur using larger values of k. Using 70% of a randomly permutated data, 203 observations as our training set and 87 observations as our testing set, 74% - 82% of the data points were correctly classified as compared to its nearest neighbors. Table 4 shows the result of the best performance. With cross-validation, k=7 yielded the best results and performed with an average accuracy of 75% per 100 runs.

**TABLE 4. k-Nearest Neighbor Example**

| Class | Classified as NCS | Classified as EARLI | Accuracy |
|---|---|---|---|
| NCS | 57 | 5 | 91.94% |
| EARLI | 9 | 16 | 56.25% |

Performing classification on the original data with all of the shape and arterial network related features had results ranging from 61-77% with an average of 69% per hundred runs. Implementing k-Nearest Neighbors the data set after only performing Elastic Net had results ranging from 62-80% with an average of 70% per hundred runs. While classification of the data set after only performing PCA had results ranging from 61-81% with an average of 74% per hundred runs.

So, using both Elastic Net and PCA has a slight improvement on these classifications showing that both feature selection and feature extraction was necessary.

We then decided to use the result from K-Means to see if we can classify the results that were assigned to the incorrect cluster. The green points in Figure 7 were originally high-risk but identified as low-risk, and vice versa, were set as the testing points whereas the red and blue points of Figure 7 were correctly clustered and set as our training set.

**TABLE 5. k-Nearest Neighbor with k-Means Classification**

| Class | Classified as NCS | Classified as EARLI | Accuracy |
|---|---|---|---|
| NCS | 26 | 40 | 39.39% |
| EARLI | 19 | 1 | 5% |

With a 31.76% accuracy, it is clear to see that these points are still difficult to classify. These points represent placentas that have values closer to the mean for each of the principal components. Figure 7 represents 2 of the placentas in the cluster of green points.

Notice that these two placentas are very similar in each of the principal components and difficult to classify by observation. The green points in Figure 6 tend to be the ones that are incorrectly classified in most testing sets. We can interpret the green points as the placentas with the highest risk for ASD in the low-risk cluster and the lowest risk for ASD in the high-risk cluster.

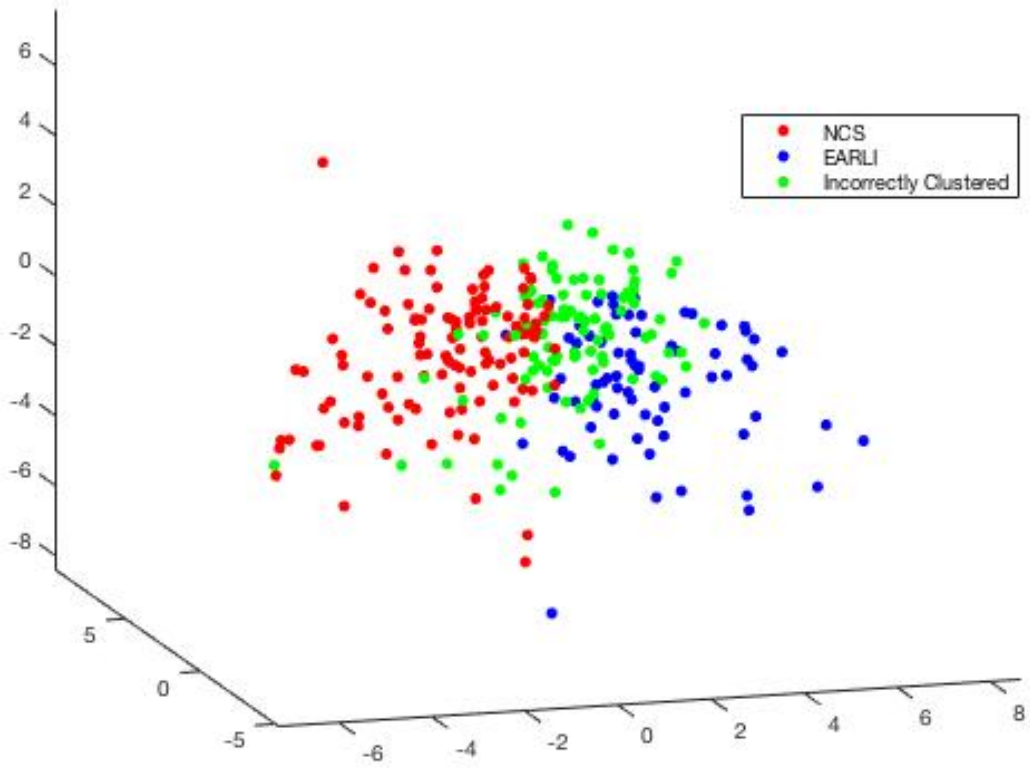
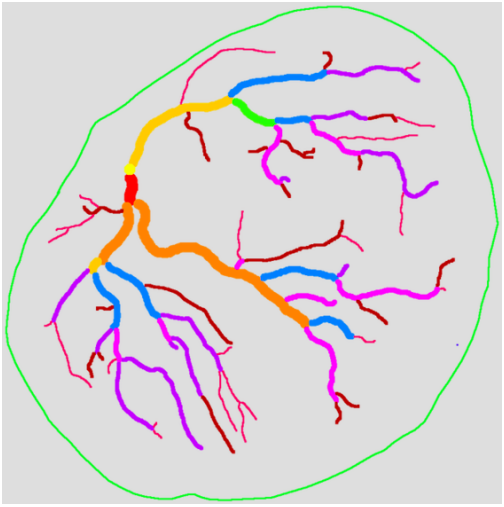
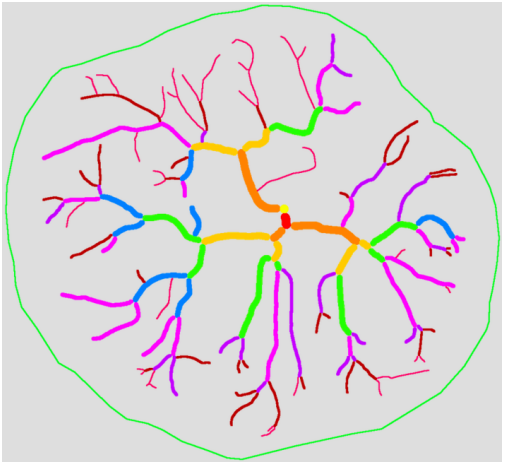



**FIGURE 7. Cluster plot with incorrectly clustered points.**



**(a) Incorrectly classified NCS.**



**(b) Incorrectly classified EARLI.**

**FIGURE 8. Commonly incorrectly classified placenta tracings.**

# CHAPTER 5

## DISCUSSION

A major contribution of this thesis is the ability to distinguish between placentas that are in a high-risk ASD group and a low-risk group based on features of the placental chorionic surface vascular network. We began with our original data set $X_0$ with 290 observations and 66 variables. To simplify the data set by removing redundency and improving accuracy, we created a sub data matrix $X_1$ which was used in Elastic Net regression to find a reduced set of features that consisted of 290 observations and 37 shape and artery related features. Elastic Net for selected the sixteen most prominent features from $X_1$ which were used as the columns of the new sub data matrix $X_2$. We then found, through Principal Component Analysis, five features that accounted for 66% of the variance in the data. These were used to project the data onto a smaller space. This projection created a data matrix $X_3$ with 290 observations and 5 columns, which allows the k-mean clustering method to group the classes based on their most prominent features. An classification accuracy ranging from 74% to 82% was achieved using k-nearest neighbor classification.

As can clearly be seen in Table 5, only one of the incorrectly clustered EARLI points was classified correctly using K Nearest Neighbors. Since EARLI consists of those observations that are considered to be at-risk of developing an ASD, it can be concluded that those classified incorrectly are less likely to develop a disorder on the spectrum than the rest of the observations in that class. On the contrary, those observations classified incorrectly that are part of NCS should be watched closely by medical professional as they are the most likely, in the class, to develop an ASD. This study helps identify the level of risk each child has of developing ASD, while taking into consideration just five factors, nodes, thickness, tortuosity, branching angle, and growth. These five components captured 66% of the variance in the data set with only twelve variables being used. This helps cut down on calculations of 66 variables and those involved in the study can focus on the remaining twelve variables used in the principal components discussed in Chap-

ter 4.

The children in the data set are too young to be diagnosed with ASD. The data is too new to actually know if those observations classified in EARLI have developed ASD. Since it takes years for the symptoms and diagnosis to come up in the children we can not truthfully state that these variables are the most important. As time passes and more observations are made and diagnosed, it will be easier to classify. In the future, this project should improve on classification of placentas with high risk of ASD and help diagnose, or at least monitor, children who are at risk of being diagnosed with an ASD.

# APPENDIX

# MATLAB CODES

## MATLAB CODES

### Elastic Net

The following Elastic Net code, Elastic_Net.m, is used for feature selection. This code utilizes the MATLAB built-in LASSO function to perform Elastic Net. The built in function takes in an $X$, $y$, and $alpha$ to perfrom Elastic Net. The $X$ is the mean subtracted data set with dimensions (290 x 37) where the rows represent each placenta and each column represents one of the 37 shape and artery related features. The binary classification label vector $y$ has dimension (290 x 1) with each element representing a row of X associating 1 with low risk and 0 with high risk. This built in functions outputs the regression coefficients in a matrix, Beta, and information of the matrix, FitInfo, including how many features each column vector has and the $\lambda$ of each. We use FitInfo to find the sixteen prominent features and the $\lambda$ used. Finally we sort the data to rank the sixteen coefficients in descending order based on the magnitude of the coefficients.

```
[Beta FitInfo] = LASSO(X,y,'Alpha',alpha);
i = find(FitInfo.DF==16);
lambda=FitInfo.Lambda(max(i));
[sortBeta,IB] = sort(abs(Beta(:,max(i))),'descend');
```

**Principal Component Analysis**

The following Principal Component Analysis code, PCA.m, is used for feature extraction. This code takes the mean subtracted data set of the sixteen prominent features from Elastic Net, $X$, as an input. Performs SVD of $X$ and uses $u$ from SVD to output the first five principal components (PC), $v$ from SVD to create the projection matrix (proj) and $s$ from SVD for the variance of each of the principal component (var).

```
function [PC,proj,variance] = PCA(X)
    [~,n] = size(X);

    [u,s,v] = svd(X); %SVD does everything

    PC = (u(:,1:5)'*X)'; %principal components
    proj = X*v(:,1:5); %projection matrix

    eigenvalue = diag(s);

    for i=1:n
        variance(i)=eigenvalue(i)/sum(eigenvalue);
    end

    variance=variance';
end
```

**K-Means**

The following K-Means code, K_Means.m, is used for clustering. This code takes the projection matrix from PCA, $X$, and a value $k$ for the number of clusters as inputs. It randomly labels the obsevations of $X$, finds the center of each of the initial clusters and continues to improve cluster predictions as long as the new label is different than the previous label of predictions. Each observation is newly labeled to the cluster that has the closest center, measured by Euclidean Distance. This code outputs the label in which cluster each item belongs in (label) and the center point for each center (cent).

```
function [label,cent] = K_Means(X, k)
    n = size(X,2); %size of data
    label = ceil(k*rand(1,n)); %label observations into a random
       cluster
    label_new = label*0; %used for while loop
    u = [1:k]; %class labels
    while any(label ~= label_new)
        im = sparse(1:n,label,1); %transform label into indicator
            matrix
        cent = X*(im*spdiags(1./sum(im,1)',0,k,k)); %compute
           centers
        label_new = label; %relabel for while loop
        for j = 1:k
            for i = 1:n
                J(j,i) = norm(X(:,i) - cent(:,j),2); %observation
                    matrix
            end
```

```
        end
        [~,label] = min(J,[],1); %assign observations to new
            cluster
    end
end
```

**K-Nearest Neighbors**

The following K-Nearest Neighbors code, K_Nearest_Neighbor.m, is used for classification. This code takes a testing and training set (test,train), the classes of the training set (labels) and a $k$ for the number of nearest neighbors as inputs. It measures the distance of each training point from the $i^{th}$ testing point and sorts the data. Finally performing a majority vote count at the end and classifying the $i^{th}$ test point to the class that occurs the most in the first $k$ training points with the smallest distance. Finally output the predictions of which class each training set belongs in (predictions).

```
function [predictions] = K_Nearest_Neighbor(test,train,labels,k)
    for i = 1 : size(test,1)
        m = size(train,1);
        B = repmat(test(i,:), m, 1); %replicate the original
            input m times
        diff = train - B; %calculate the distance of each column
        dist = sqrt(sum((diff.*diff)'));
        [~, idx] = sort(dist); %neareast neighbor
        idx = idx(1:k);
        % make a prediction
        if sum(labels(idx) == 1) > k/2  %majority voting
            predictions(i) = 1;
            else
            predictions(i) = -1;
        end
    end
    predictions=predictions';
```

```
end
```

# BIBLIOGRAPHY

## BIBLIOGRAPHY


[1] Y. Wang, "Vascular biology of the placenta," *Colloquium Series on Integrated Systems Physiology: From Molecule to Function*, vol. 2, pp. 1–98, 2010.

[2] R. Sood, J. L. Zehnder, M. L. Druzin, and P. O. Brown, "Gene expression patterns in human placenta," *Proceedings of the National Academy of Sciences*, vol. 103, pp. 5478–5483, 2006.

[3] D. J. Roberts and E. Oliva, "Clinical significance of placental examination in perinatal medicine," *The Journal of Maternal-Fetal and Neonatal Medicine*, vol. 19, pp. 255–264, 2006.

[4] D. Schubert, "Machine learning of transport networks in the human placenta," Master's thesis, Dept. of Physics, Imperical College London, 2015.

[5] Q. Xia, L. A. Croen, M. D. Fallin, C. J. Newschaffer, C. Walker, P. Katzman, R. K. Miller, J. Moye, S. Morgan, and C. Salafia, "Human placentas, optimal transportation and high-risk autism pregnancies," *Journal of Coupled Systems and Multiscale Dynamics*, vol. 4, pp. 260–270, 2016.

[6] C. Whitelaw, P. Flett, and D. J. Amor, "Recurrence risk in autism spectrum disorder: A study of parental knowledge," *Journal of Pediatrics and Child Health*, vol. 43, pp. 752–754, 2007.

[7] C. Lord, E. H. Cook, B. L. Leventhal, and D. G. Amaral, "Autism spectrum disorders," *Neuron*, vol. 28, pp. 355–363, 2000.

[8] M. T. Carter and S. W. Scherer, "Autism spectrum disorders in the genetics clinic: A review," *Clinical Genetics*, vol. 83, pp. 399–407, 2013.

[9] E. Bloom, C. Lord, L. Zwaigenbaum, E. Courchesne, S. R. Dager, C. Schmitz, R. T. Schultz, J. Crawlet, and L. J. Young, "The developmental neurobiology of autism spectrum disorder," *Journal of Neuroscience*, vol. 26, pp. 6897–6906, 2006.

[10] J. H. Miles, "Autism spectrum disorders: A genetics review," *Genetics in Medicine*, vol. 13, pp. 278–294, 2011.

[11] C. Lord, S. Risi, P. S. Dilavore, C. Shulman, A. Thurm, and A. Pickles, "Autism from 2 to 9 years of age," *Archives of General Psychiatry*, vol. 63, pp. 694–701, 2006.

[12] *EARLI - Early Autism Risk Longitudinal Investigation*, (2017), Welcome to EARLI. [Online]. Available: http://www.earlistudy.org

[13] S. Ozonoff, G. S. Young, A. Carter, D. Messinger, N. Yirmiya, L. Zwaigenbaum, S. Bryson, L. J. Carver, J. N. Constantino, K. Dobkins, T. Hutman, J. M. Iverson, R. Landa, S. J. Rogers, M. Sigman, and W. L. Stone, "Recurrence risk for autism spectrum disorders: A baby siblings research consortium study," *Pediatrics*, pp. 488–495, 2011.

[14] *National Institute of Child Health and Human Development*, (2017), National Children's Study. [Online]. Available: http://www.nichd.nih.gov/research/NCS

[15] E. Haeussner, S. Christoph, H.-G. Frank, and F. E. von Koch, "Novel 3d light microscopic analysis of IUGR placentas points to a morphological correlate of compensated ischemic placental disease in humans," *Scientific Reports*, vol. 6, pp. 1–11, 2016.

[16] I. Guyon and A. Elisseff, "An introduction to variable and feature selection," *The Journal of Machine Learning Research*, vol. 3, pp. 1157–1182, 2003.

[17] T. Hastie, R. Tibshirani, and J. H. Friedman, *The Elements of Statistical Learning Data Mining, Inference, and Prediction*. New York: Springer, 2016.

[18] S. Kaushik. (2017) Feature selection methods with example (variable selection methods). [Online]. Available: http://www.analyticsvidhya.com

[19] J. Hamon, "Combinatorial optimization for variable selection in high dimensional regression: Application in animal genetic," Ph.D. dissertation, Dept. of Computer Science, Univ. of Sciences and Technologie of Lillie, 2013.

[20] T. M. Phuong, Z. Lin, and R. B. Altman, "Choosing SNPs using feature selection," *Journal of Bioinformatics and Computational Biology*, vol. 4, pp. 241–257, 2006.

[21] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 67, pp. 301–320, 2005.

[22] A. E. Hoerl and R. W. Kennard, "Ridge regression: Biased estimation for nonorthogonal problems," *Technometrics*, vol. 12, pp. 55–67, 1970.

[23] R. Tibshirani, "Regression shrinkage and selection via the lasso: A retrospective," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 58, pp. 267–288, 1996.

[24] J. N. Kutz, *Data-Driven Modeling & Scientific Computation: Methods for Complex Systems & Big Data*. Oxford: Oxford Univ. Press, 2013.

[25] R. C. Cascaval, "Eigenvalues, singular value decomposition," *in Encyclopedia of Social Network Analysis and Mining.*, pp. 456–462, 2014.

[26] S. P. Lloyd, "Least squares quantization in pcm," *IEEE Transactions on Information Theory*, vol. 28, pp. 129–137, 1982.

[27] M. Garey, D. Johnson, and H. Witsenhausen, "The complexity of the generalized lloyd - max problem (corresp.)," *IEEE Transactions on Information Theory*, vol. 28, pp. 255–256, 1982.

[28] E. Alpaydin, *Introduction to Machine Learning*. Cambridge, MA: MIT Press, 2010.

[29] D. Coomans and D. L. Massart, "Alternative k-nearest neighbour rules in supervised pattern recognition: Part 1. k-nearest neighbour classification by using alternative voting rules," *Analytica Chimica Acta*, vol. 136, pp. 15–27, 1982.

[30] G. Dougherty, *Digital Image Processing for Medical Applications*. Cambridge: Cambridge University Press, 2014.