Classification using Machine Learning Algorithms (MALA)

Rodrigo Farnham
Department of Mathematics
CSU Long Beach

Tuyen Ly
Department of Mathematics
CSU Long Beach

Jason Wang
Department of Mathematics
CSU Long Beach

May 20, 2011

Abstract

This report summarizes the results of our work on trying to predict the health of a baby. We used two different machine learning algorithms, Weka and our own Naive Bayes Classifier. We discovered that placental ratio and Term/Preterm Birth yield interesting results, based on our list of 19 features. While the placental ratio results are puzzling, we learned that the two features Eclampsia and Chronic Hypertension had a significant relationship between the label Term/Preterm.

1 Introduction and Overview

According to the US Census Bureau, over 350,000 children are born in the world everyday.[1] Many of these babies are unfortunately born with medical conditions, such as hypertension, diabetes, etc. In some cases, fetuses are aborted because either its medical condition or the mother's condition is threatened. With these facts, our group's goal was to try to find a way to assist doctors and expecting mothers by predicting a baby's health.

We first started off with using the various data of around 1,500 cases, data such as mother's health, family's poverty index, gestational days, etc. In addition to this plethora of data, we extracted additional data from the given placenta images. We then used all this information to train Weka, a learning-machine system. Therefore, our ultimate goal was to obtain a prediction of a pregnancy outcome, given specific data input. Along the way, we also wanted to see if certain characteristics had more impact on the outcome of the pregnancy versus other characteristics.

2 Machine-Learning Softwares

2.1 Classification Algorithms

We used two different machine learning algorithms when analyzing our data and testing our proposed features: (a) we used a software called WEKA that used Support Vector Machine (SVM) models, as well as (b) our own Naive Bayes Classifier.

These two algorithms were used on the same data set but explored different facets of the data. Support Vector Machines are able to cope with high dimensional data, both discrete and otherwise. We applied the SVM primarily towards the full feature classification problem, i.e. deducing classification from the entirety of our feature set. On the other hand, the Bayes Classifier is more suitable for binary features, which are either present or absent. Although Naive Bayes classifiers are able to cope with very high dimensional data as well - even to a greater extent than SVMs - we were restricted to only a handful of binary maternal and fetal features available to us. Therefore, since some of our data was in binary form (i.e. Absence/Presence of Hypertension) while others were in high-dimensional form (i.e. Gestational Days), we wanted to use both algorithms when testing our data.

Further description of Support Vector Machines and Naive Bayes Classifiers can be found in Appendix A.

2.2 The Function of Machine-Learning Software

Before we describe our research methods, we first introduce Weka, the learning-machine system that we used. Since machine-learning softwares are used to identify patterns within data, we wanted to use a learning-machine system that would help us in our predictions. We chose Weka because of its many features as well as awards that it has won from the data-mining community. [2]

We first generally describe how Weka works: Given a data set of cases containing input data and output data, Weka takes a percentage of the cases and attempts to produce a prediction model, based on the input and output data of that specific percentage. It then looks at the remaining input and applies its prediction model and compares its predicted output to the actual remaining output. We use a simple example to illustrate this concept:

In Figure 1, we see see 6 different cases containing 3 data input categories (Average Temperature, Month, State) and 2 data output categories (Weather, Average Population Health). Based upon such data, potential questions could be:

- Which input factor(s) have a greater influence on Weather?

- Which input factor(s) have a greater influence on Average Population Health?

- Are there factors that have negligent influence on either/both Weather and Average Population Health?

- If Average Temperature was 30 degrees in the month of August in the state of Alaska, what would be the predicted Weather and Average Population Health?

Average Temperature	Month	State	Weather	Average Population Health
10 degrees	January	Alaska	cold	sick
80 degrees	August	Florida	hot	well
50 degrees	April	Oregon	cold	well
60 degrees	December	California	cold	sick
70 degrees	June	California	hot	well
50 degrees	January	Florida	cold	sick

Figure 1: Potential Factors Affecting Weather & Average Population Health

- etc.

Let us assume that Weka takes all the input and output data for only the first 4 cases. It then analyzes the input and output data for these 4 cases and looks for a pattern or algorithm. Having come up with an algorithm, it then looks at the input data for the remaining 2 cases and predicts the output data. Let us assume that it predicts that the outcome for the remaining 2 cases is:"(hot, well)" and "(cold, well)". Finally, it compares its predicted output data and the actual output data. Therefore, it would have had 100% accuracy for predicting Weather and 50% accuracy for predicting Average Population Health in this example. As can be seen through the above example, a machine-learning system is very useful, as it aims to predict answers to many questions.

3   Research and Analysis Methods

In this section, we will discuss the two types of research and analysis methods that were carried out: methods used to analyze the already obtained data from Dr. Carolyn Salafia and Amy Mulgrew, and methods used to analyze the placental images that were also obtained from Dr. Salafia and Ms. Mulgrew.

3.1   Analysis of Obtained Data

We first started out by analyzing an Excel file from Dr. Salafia that contained a vast amount of data of the mothers and babies in around 1,500 cases. This data contained information such as: the presence/absence of hypertension, the gestational days, the baby's birth weight, mother's ethnicity, etc. Our goal was to look for features that we believed would have an impact on the outcome of the baby's health. While looking at the long list of features, we used three criteria: (a) reasonability, (b) dispersal, and (c) scientific evidence (if applicable).

We used reasonability as the first criteria, in order to remove features that we thought were

not relevant to the outcome of the child. This removed features such as "Mother's Height" from our choices. It also removed redundant features from our choices such as "Mother's Pre-Pregnancy Weight," as there was another feature named "Weight Gain Total" that we believed would be more relevant to our goal.

Next, we used data dispersal as a criteria because we wanted there to be enough of a spread of data, so that Weka could learn from the results. For example, we discounted "Number of Previous Pregnancies that were Ectopic" because there were only 27 cases, or roughly 1.7% of the total number of cases, and we preferred to submit data to Weka that was not heavily concentrated in one answer.

Finally, we also used scientific evidence to look for a further refinement of the features we were interested in. For example, we believed that "Family's Poverty Index Level" would be a significant feature because of an article written by Dr. Charles Oberg from the University of Minnesota entitled, "The Impact of Childhood Poverty on Health and Development" [3]

For these above reasons, we chose to look whether the following 16 features had a significant influence on the outcome of the birth:

1. Body Mass Index (BMI)

2. Gestational Days

3. Mother's total weight gain [4]

4. Number of previous pregnancies

5. Summary Index number of Prenatal Care Adequacy [5]

6. Family's Poverty Level Index

7. Eclampsia

8. Gestational Diabetes

9. Pregnancy Induced Hypertension

10. Chronic Hypertension

11. Acute Inflammation

12. Chronic Inflammation

13. Vascular Pathology

14. Maternal Vascular Pathology

15. Fetal Vascular Pathology

16. Other Vascular Pathology

3.2   Placental Image Analysis

In addition to the analysis of already-obtained data, we analyzed and obtained additional information. Whereas the analysis described in Section 2.1 related to the physical and medical features of the mother and child, we also wanted to see if the characteristics of the placenta had any impact on the health of the child. We believed that the shape of not only the placenta but of the vessel network could have significance. By analyzing the placental images, we were able to add 3 additional features that we believed could be significant to the predicted health of the baby: (a) Heat Map Value, (b) Dominant Direction, and (c) Signed Deviation Vector.
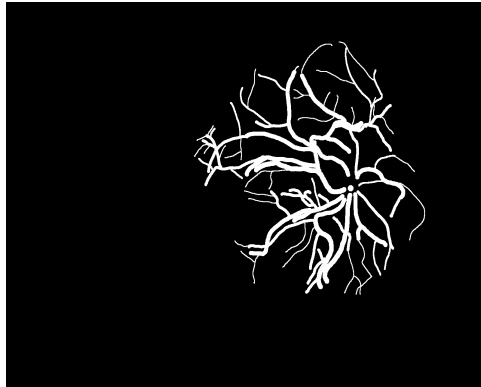

Figure 2: Traced Placental Vessel Network

3.2.1   Heat Map Value

When we began looking at the various placental images, we hypothesized that the dispersion of the placental blood vessel network may have significance on the health of the baby. [6] An example of a placenta's blood vessel network is shown in Figure 2. Because the placental blood vessel network assists in transferring nutrients to the baby [7], we theorized that a less evenly dispersed vessel network may have more negative effects on the baby's health. In order to evaluate the dispersion of the placental blood vessel network, we created a new feature called "Heat Map Value" that would analyze the farthest distance of a point on a placenta to the nearest blood vessel.

We created "Heat Map" to be an indicator of the efficiency of a placental vessel network. We will demonstrate the idea using an example.

Figure 3 shows a sample vessel network contained within a placenta. We observe that the vessel network is relatively evenly dispersed throughout the placenta. Our "Heat Map" software first labels all pixels containing the vessel network as "0", as shown in Figure 4. Please note that for this example, there are "Inf" labels along the border because those pixels are missing adjacent pixels, due to the fact that they comprise the border. However, in a real placenta, there will always be adjacent pixels present, and these error values will not show.
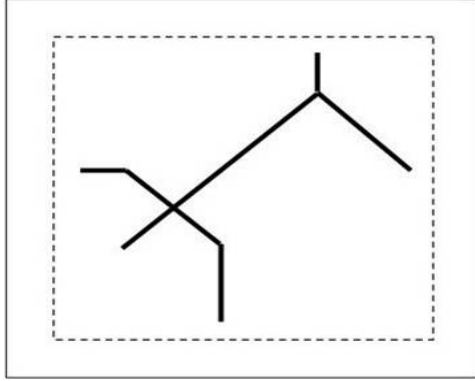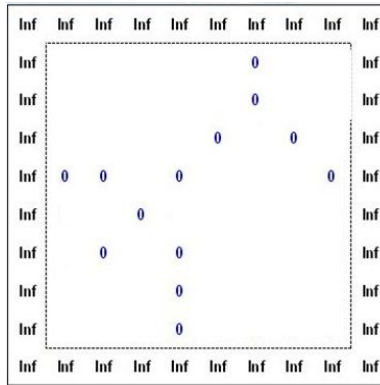
Figure 3: Sample Vessel Network

Figure 4: Matlab Numerical Representation of the Vessel Network

We then assign "1" to the pixels that are either above, below, to the right of, or to the left of pixels labeled as "0" as shown in Figure 5. Please note that these numbers do not imply Euclidean distance. Instead, they are simply an indicator of blood vessel efficiency to that specific pixel.

Next, we assign "1.41" to the pixels that are diagonal to the pixels labeled as "0," as shown in Figure 6.

The software then moves further and further out from the main vessel network, assigning values of 2, 2.41, 2.83, 3, etc., until the entire grid is filled, as shown in Figure 7.

As we can see in Figure 7, the greatest numerical value is 4, located at the bottom right-hand corner of the grid. Because this is the greatest numerical value, it is then assigned as the Heat Map Value. Therefore, if another sample vessel network was bunched up in the lower left-hand corner, we would conclude that the Heat Map Value would be a large number that would be located in the top right-hand corner. Thus, we hypothesized that the greater a placenta's Heat Map Value is, the less efficient the placenta would be, and the greater negative impact it would
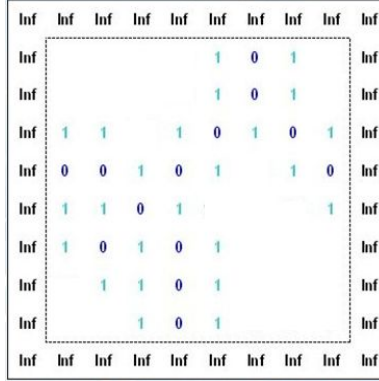
Figure 5: Matlab Numerical Representation of the Vessel Network with Indicator of 1
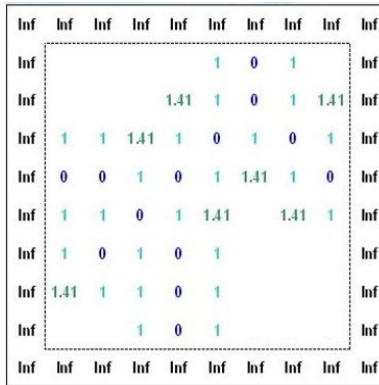

Figure 6: Matlab Numerical Representation of the Vessel Network with Indicator of 1.41

have on the baby.

3.2.2 Dominant Direction

As helpful as the Heat Map Value is, it has limitations. Different placental vessel networks could have the same Heat Map Value while having completely different vessel network dispersions. Therefore, we also wanted to look at the vessel network as a whole and decided to analyze the average direction and magnitude of the overall vessel network. Using the umbilical cord insertion point as the origin, we were able to assign each vessel network an Average Magnitude Value, ρ (rho), and an Average Direction Value, θ (theta). Dominant Direction is described by these two values.

For each placental vessel network image, every pixel is assigned a value of 1 or a 0, denoting the presence or absence of a vessel at that pixel. Therefore, the dominant direction is nothing more than the average of all the vectors going from the insertion point to each vessel pixel. The hope is that this feature would capture any overt tendencies of the vessel network.
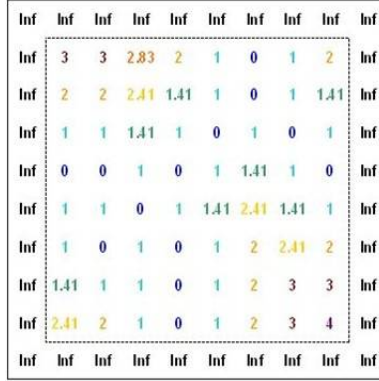
7

Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf
Inf	3	3	2.83	2	1	0	1	2	Inf
Inf	2	2	2.41	1.41	1	0	1	1.41	Inf
Inf	1	1	1.41	1	0	1	0	1	Inf
Inf	0	0	1	0	1	1.41	1	0	Inf
Inf	1	1	0	1	1.41	2.41	1.41	1	Inf
Inf	1	0	1	0	1	2	2.41	2	Inf
Inf	1.41	1	1	0	1	2	3	3	Inf
Inf	2.41	2	1	0	1	2	3	4	Inf
Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf

Figure 7: Complete Matlab Numerical Representation of the Vessel Network

3.2.3   Signed Deviation Vector

Whereas Heat Map Value and Dominant Direction describe the placental vessel network, we also hypothesized that the shape of the placenta itself might have significant influence on the health of the baby. Our hypothesis was also based on various medical research and publications written about the impact of placental surface shape. [8] [9]

We started out by looking at the 139 placental images that were available to us and used these images to produce the shape of the average placenta, or mean placenta. We would then carry out the following procedure for each placenta:

Starting at the umbilical cord insertion point, we would rotate a full revolution by moving in increments of 1 degree and compare the length deviation between that specific placenta and the mean placenta. A positive value implied that the specific placenta was longer than the mean placenta at that degree, while a negative value implied that the specific placenta was shorter than the mean placenta at that degree. Thus, each placenta's Signed Deviation Vector would be a vector containing 360 values, 1 value per degree.
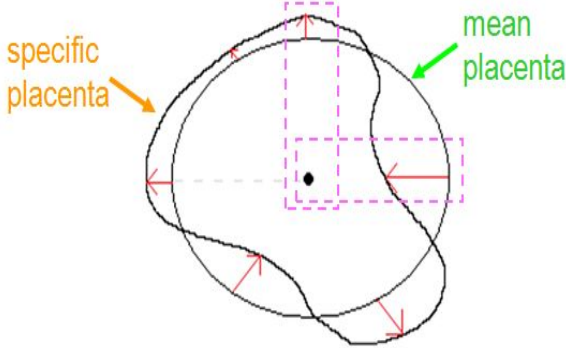

Figure 8: Sample Placental Signed Deviation Vector

An illustration of the calculation of the Signed Deviation Vector is shown in Figure 8: At 0

degrees, the specific placenta has smaller length than the mean placenta. Therefore, the value at 0 degrees would be negative. In the same way, the value at 90 degrees would be positive, since the specific placenta has larger length than the mean placenta. Thus,we want our SDV values to be as close to 0 as possible.

4   Implementation and Results

4.1   Number of Cases

The initial data that we looked at contained information of over 1,500 mothers and babies. However, we only had 139 placental images. Thus, out of the initial 1,500, we narrowed our data set to 139 cases. Out of the 139 cases, we wanted each case to have information for each of the 19 features (the initial 16 features plus the 3 new Placental Image features) that we were testing. Therefore, after we removed cases that were missing data, we ended up implementing data of 122 cases into our learning-machine software.

4.2   Implementation of Weka

In Weka, we used the supplied (linear kernel) support vector machine under its SMO method (which is different from libSVM).

4.3   Over-fitting and Prediction Significance

When investigating the predictive power of a classification system, there were two things that we had to keep in mind: (a) over-fitting and (b) significance.

When training a classifier, care must be taken to guard against over-fitting. Over-fitting occurs when relationships between variables are predicted, but in reality, they are not true and most likely were created due to noise. To avoid over-fitting in our experiments, we used leave-one-out cross validation (LOOCV). In this mode, the model is trained with all but one of the data points and then tested against that excluded data point. The reported classification accuracy is the accuracy over all possible singular exclusions.

Once the accuracy is measured, the next important question is how significant the predictions are are. A user might assume that a machine-learning system is good if it can make predictions with 50% or more accuracy. However, that may not always be the case. For example, if a fair 3-sided die with sides X, Y, and Z is tossed, the probability of the die showing either Y or Z is 2/3 or 66.6%. Therefore, if a machine-learning system's algorithm predicts Y or Z with 55% accuracy, it is actually not predicting very well, because it is lower than a user simply guessing Y or Z for every toss. Instead, the algorithm should have a significant improvement above 66.6% in order for the algorithm to be considered as "good". Thus, we see that the standard of "50% of more" does not necessarily mean that a prediction algorithm is good.

Significance is given by the tail of the cumulative distribution function of repeated independent Bernoulli trials.

$$\text{classify}(f_1, \ldots, f_n) = \arg\max_c p(C = c) \prod_{i=1}^{n} p(F_i = f_i \mid C = c)$$

This number is frequently called alpha.

The accepted alpha for significance is 0.05 and sometimes 0.01. For the analysis of our resutls, we are using the 0.05 convention.

Therefore, according to the above formula, to establish statistical significance, the formula dictates that the number of instances must be large or the classification accuracy must be significantly higher than the prior probability.

5   Analysis and Conclusions of Results

5.1   Analysis

We investigated the classification accuracy against the following four labels:

1. placental ratio (the quotient of Placental Weight and Birth Weight)

2. beta (the quotient of the logarithm of Placental Weight and the logarithm of Birth Weight)

3. birth weight, and

4. term or preterm birth

Of these, only placental ratio and term/preterm yielded interesting results for our particular combination of features. For beta and birth weight, not only were the results statistically insignificant, but also the two labels are plagued with certain medical arbitrariness. In regards to birth weight, it was unclear as how to appropriately segregate the copious birth weight values. In addition, when we studied beta, the cut-off value of 0.75 displayed no correlation (opinions on its accuracy differ substantially). [1] For these reasons, we found our results for those two labels unsatisfactory. On the other hand, results for both placental ratio and term/preterm were encouraging.

The results are detailed below, along with their statistical significance. The primary way of presenting classification accuracy is through a confusion matrix - a handy way of separating all correct and incorrect classifications. Additionally, while interpreting the results, it is important to keep in mind the prior distribution. Finally, the terms "positive" and "negative" refer to the subjective predicted outcomes, such that a positive classification implies that the classifier deems the particular instance a "good outcome".

---

[1]http://en.wikipedia.org/wiki/Kleiber%27s_law

5.1.1 Placental Ratio

Two experiments were performed with the placental ratio labeling. The first, in Figure 9, used all attributes but weight attributes, as they are directly related to the definition of placental ratio. The second experiment also excluded the signed deviation vector information. The prior distribution had approximately 54.03% of the data labeled Good and a total of 124 instances.

Confusion Matrix:	Classification	
True Label / Classification	Bad	Good
Bad	29	28
Good	20	47

Figure 9: Placental Ratio Results Using All Attributes and Features

The classification accuracy achieved with all the attributes and features, was 61.29%, as can be seen in Figure 9. The statistical significance of this result is 0.062, which does not meet our criteria of 0.05.

Confusion Matrix:		
	Classification	
True Label / Classification	Bad	Good
Bad	20	37
Good	9	58

Figure 10: Placental Ratio Results Excluding SDV

The classification accuracy of all attributes and features, but excluding SVD, was 62.90%, as shown in Figure 10. The statistical significance of this result is 0.029, which is considered statistically significant.

Surprisingly, excluding the SDV data from the training set improved accuracy. While the second result is arguably statistically significant, there are a couple of issues. Firstly, the placental ratio bucketing scheme was arbitrarily chosen; thus, the linking of our results to any medical predictability is a daunting task. Secondly, the second experiment's improved results come with significantly skewed classification bias. In particular, the second experiment made many false

positive classifications. Because of these two points, it seems unlikely that these placental classification results can be extended in any medically meaningful way.

5.1.2 Term/Preterm

There were three salient experiments performed with the Term/Preterm labeling and WEKA's SVM facilities. The first made use of all features, save for gestational age. The second concentrated on only the shape of the placenta by way of SDV feature. Finally, the third experiment only included the maternal and fetal characteristics. The prior distribution was very skewed, with approximately 78.22% of the 124 instances labeled term. Because the prior distribution is so heavily biased towards term, only very high classification accuracies are deemed significant.

Furthermore, relationship between Term/Preterm labeling and the binary subset of maternal and fetal characteristics was investigated using a Naive Bayes Classifier.

Confusion Matrix

True Label / Classification	Term	Preterm
Term	89	8
Preterm	9	18

Figure 11: Term/Preterm Results Using All Attributes and Features

The classification accuracy achieved using all features was 86.29%., as shown in Figure 11. The statistical significance of this result is 0.016, which is considered statistically significant. Moreover, there appears to be good separation between term and preterm classifications, inasmuch as the false positive and false negative rates are relatively small.

The classification accuracy using merely the SDV feature was 82.26%, as shown in Figure 12. The statistical significance of this result is 0.163, which is very unimpressive. Furthermore, there are many false positives. It bears mentioning that if less strict cross validation is used (or none at all), misleading results are obtained.

The classification accuracy using only the maternal and fetal characteristics was 84.68%, as shown in Figure 13. The statistical significance of this result is 0.047, which is considered statistically significant. However, there are very many false-positives.

Since the results based solely on maternal and fetal characteristics were so promising (with respect to their simplicity), we furthered the investigation when restricted only to the binary

Confusion Matrix

Classification

True Label / Classification	Term	Preterm
Term	90	7
Preterm	15	12

Figure 12: Results Using SDV Only

Confusion Matrix

Classification

True Label / Classification	Term	Preterm
Term	96	1
Preterm	18	9

Figure 13: Term/Preterm Results Using Maternal and Fetal Characteristics

attributes with a Naive Bayes Classifier, the results which are shown in Figure 14.

*Note: In this case, we removed two instances which had missing information, because we deemed their presence detrimental to our goals, given the sparsity of our data.

The classification accuracy achieved was 81.97%, which is very impressive for such simple features. The statistical significance is 0.187, which is not significant.

While the accuracy is not high enough to be deemed statistically significant, it is notable that such simple features would elicit such results. By listing the attributes the Bayesian Classifier found most relevant, we learned that Eclampsia and Chronic Hypertension were pivotal in determining the Term/Preterm label. The relevance of these two features were corroborated when we later used WEKA's feature selector, which listed Eclampsia and Chronic Hypertension as the two most important features out of the binary features.

This relationship is illustrated by the scatter plots in Figure 15.

As can be seen in Figure 15, there is a very pronounced separation between term and preterm in Chronic Hypertension and Eclampsia space.

Confusion Matrix

True Label / Classification	Classification	
	Term	Preterm
Term	88	7
Preterm	15	12

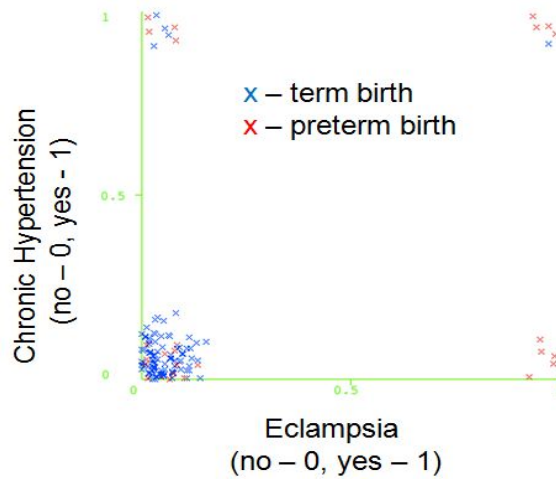Figure 14: Term/Preterm Results Using Maternal Binary Attributes

Figure 15: Sample Placental Signed Deviation Vector

5.2 Conclusions

It is very unclear as how to interpret the placental ratio label. There is a statistically significant relationship being picked up by the support vector machine classifier, but additional medical work needs to be done to properly interpret what it is. Furthermore, it was very puzzling that the exclusion of data fostered a superior classification accuracy. This contradicts the classification power attributed to the signed deviation vector by the work of other researchers as well as some of our own experiments.

On the other hand, the Term/Preterm label has a very clear-cut interpretation, and while at the moment our methodology can only attack the classification problem aposteriori, hopefully it can be extended by means of imaging techniques or different feature selection. In addition, for medical purposes, we would like to minimize the number of false-positives while still maintaining a balance with respect to classification accuracy, since that would be the most economical. As of now, it seems that the combination of all features does indeed form a capable predictor of the Term/Preterm label.

Surprisingly, the lone binary attributes manage to achieve notable, if not statistically significant, classification accuracy. However, upon further investigation, we have discovered that the Eclampsia attribute attribute is suspect: when a mother suffers from eclampsia, preterm birth is induced. Therefore, the results obtained through the Naive Bayes Classifier must be taken with a dose of skepticism.

Finally, we proposed two novel features - heatmap and dominant direction. In a series of undocumented experiments, we were led to the conclusion that neither has a significant impact in classification accuracy. In fact, this is to be expected because they distill the complex, featureful image to mere numbers. The two ideas are salvageable by extending their dimensionality (or refining granularity), but that is left for future work. As for SDV, there is literature speaking to its capabilities, but we are as of now not entirely convinced that by itself SDV is a sufficient predictor - more work is needed in this area.

6 Future Work

We believe that based on the conclusions that we have presented, further work can be done, in order to better refine the prediction of a baby's health. A few points of observation:

6.1 The Need for Improved Data

Due to the fact that our findings were based on a total of 122 cases, we feel that the machine-learning systems will be more accurate if it were given more data to train itself on.

6.2 The Disadvantage of Single Numerical Values

In addition, we also felt that using a single numerical value to describe a placenta oversimplifies the situation, as many idiosyncrasies and unique qualities of that placenta may be lost. For example, a placenta's Dominant Direction values do not tell tell the user how evenly dispersed its vessel network is, as a vessel network with 5 branches may have the same Dominant Direction as a vessel network with 20 branches.

6.3 The Possible Influence of Eclampsia

Finally, while we were delighted to know that our suspicions above Eclampsia were confirmed, we also recognize that including Eclampsia in our initial training may have negatively influenced our conclusions. Therefore, we believe that further work may include the removal of Eclampsia from the list of features

A   Appendix A - Further Description of Support Vector Machines and Naive Bayesian Classifier

A.1   Support Vector Machines

A support vector machine is a classification scheme which seeks to find the optimal separating hyperplane. It operates on 2-class datasets which are assumed to have a linear decision boundary. However, the linearity assumption can be relaxed through kernel methods. The training of an SVM consists of finding the hyperplane with the maximum margin (margin being the minimum distance between any data point and the hyperplane). Additionally, allowance for misclassification in the training data can be incorporated through the addition of a penalty term to the objective function. And interesting property of the SVM method is that finding the optimal hyperplane is a convex optimization problem, and therefore the solution is guaranteed to be a global optimum, this is in stark contrast with other classification schemes, such as neural networks, where local optima abound. [10] The classification step is simpler still: given a point x

$$h(x) = w \bullet x + b c(x) = \begin{cases} 1, \text{if } h \text{ is positive} \\ -1, \text{if } h \text{ is negative} \end{cases} \quad \text{where the hyperplane is given by } w \bullet x + b = 0$$

The literature for support vector machines is very rich and the reader is invited to read further if he is interested.

A.2   Naive Bayesian Classifier

The Naive Bayesian Classifier is based on Bayes Theorem and the (naive) assumption that each feature is independent. Classification is done by finding the class label which maximizes P(Class = c — F1, F2, F3). This estimate itself is computed as the product of several simpler likelihood estimates. Training consists of recording the frequencies of occurrences of each class label, each feature, and when they occur simultaneously.

There are many implementation details which can be found on online resources, but particularly important ones:

1. Dirichlet Prior (aka pseudocount)
   If something has occurred 0 out of N times, count it as having occurred as 1 time. Asymptotically it does not affect the probabilities, but this simple modification prevents events that have not been observed from wiping out an otherwise nonzero probability estimate.

2. Log likelihood
   Rather than computing the conditional probability as a product through Bayes Theorem, compute the log of it as the sum of the associated log quantities. This prevents underflow in the computer arithmetic and is very important.

B Appendix B - MATLAB codes for Dominant Direction

```
function [rho theta] = dominance(source)
    [n m] = size(source);
    vec = [0 0];
    count = 0;
    origin = [ceil(n/2), ceil(m/2)];
    for i = 1:n
        for j = 1:m
            if source(i, j) ~= 0
                vec = vec + ([i j] - origin);
                count = count + 1;
            end
        end
    end
    vec = vec./count;
    rho = sqrt(vec * vec');
    theta = -atan(vec(1)/vec(2));
return
```

C Appendix C - MATLAB codes for HeatMap

Let $f(i,j)$ be the number assigned to a pixel at coordinates (i,j) and $d((i,j),(l,k))$ be the metric

$$f(i,j) = \begin{cases} f(i,j) = 0, & \text{if } (i,j) \text{ is a vessel pixel} \\ \min(f(i \pm 1, j \pm 1) + d((i,j),(i \pm 1, \pm 1))), & \text{otherwise} \end{cases}$$

```
function [img] = floodfill(source, mask)
    [n m] = size(source);
    img = source;
    queue = zeros(2,n*m);
    k = 1;
    for i = 1:n
        for j = 1:m
            if img(i,j) == 0
                img(i,j) = Inf;
            else
                img(i,j) = 0;
                queue(:,k) = [i,j];
                k = k + 1;
            end
        end
    end
```

```
l = 1;
while l < k
    i = queue(1,l);
    j = queue(2,l);
    cross = img(i,j) + 1;
    diagonal = img(i,j) + sqrt(2);

    if (img(i+1, j) > cross) && mask(i+1,j)
        img(i+1,j) = cross;
        queue(:,k) = [i+1,j];
        k = k + 1;
    end
    if (img(i-1,j) > cross) && mask(i-1,j)
        img(i-1,j) = cross;
        queue(:,k) = [i-1,j];
        k = k + 1;
    end
    if (img(i,j+1) > cross) && mask(i,j+1)
        img(i,j+1) = cross;
        queue(:,k) = [i,j+1];
        k = k + 1;
    end
    if (img(i,j-1) > cross) && mask(i,j-1)
        img(i,j-1) = cross;
        queue(:,k) = [i, j-1];
        k = k + 1;
    end
    if (img(i+1, j+1) > diagonal) && mask(i+1,j+1)
        img(i+1,j+1) = diagonal;
        queue(:,k) = [i+1,j+1];
        k = k + 1;
    end
    if (img(i+1, j-1) > diagonal) && mask(i+1,j-1)
        img(i+1, j-1) = diagonal;
        queue(:,k) = [i+1,j-1];
        k = k + 1;
    end
    if (img(i-1,j+1) > diagonal) && mask(i-1,j+1)
        img(i-1,j+1) = diagonal;
        queue(:,k) = [i-1,j+1];
        k = k + 1;
    end
    if (img(i-1,j-1) > diagonal) && mask(i-1,j-1)
        img(i-1,j-1) = diagonal;
```

```
            queue(:,k) = [i-1,j-1];
            k = k + 1;
        end

        l = l + 1;
    end

return
```

D   Appendix D - Codes for the Naive Baynesian classifier

```python
from math import log, exp

class BayesianClassifier(object):
    def __init__(self):
        self.observation_total = 0
        self.observations = {}
        self.observations_dual = {}
        self.labels = {}
        self.feats = {}

    def train(self, event, evidence):
        self.observations.setdefault(event, {})
        self.labels.setdefault(event, 0)
        self.observation_total += 1
        self.labels[event] += 1
        for v in evidence:
            self.observations[event].setdefault(v, 0)
            self.observations_dual.setdefault(v, {})
            self.observations_dual[v].setdefault(event, 0)
            self.feats.setdefault(v, 0)
            self.observations[event][v] += 1
            self.feats[v] += 1
            self.observations_dual[v][event] += 1

    def classify(self, evidence, complement=False):
        estimates = [(self.cond_comp(event, evidence, complement), event)
                      for event in self.labels]
        highest_prob, likely_event = max(estimates)
        estimates = [(exp(prob - highest_prob), event)
                      for prob, event in estimates]
        highest_prob = 1 / sum(prob for prob, event in estimates)
        estimates = [(highest_prob * prob, event)
```

```python
                    for prob, event in estimates]

        return likely_event, highest_prob

    def cond_comp(self, event, evidence, complement):
        if event not in self.labels:
            return 0
        else:
            #Using log probabilities to prevent underflow
            P_event = log(self.labels[event]) - log(self.observation_total)
            P_condt = 0
            for Bk in evidence:
                if Bk not in self.feats: continue
                P_condt += ( log(self.observations[event].get(Bk,0) + 1) -
                                log(self.labels[event]) )
            if complement:
                for Bk in self.feats:
                    if Bk in evidence: continue
                    P_condt += ( log(self.labels[event] -
                                        self.observations[event].get(Bk, 0) + 1) -
                                    log(self.labels[event]) )
            return P_event + P_condt

    def cull(self, **kargs):
        try:
            threshold = kargs['threshold']
        except KeyError:
            threshold = float('inf')
        try:
            cull_rare = kargs['cull_rare']
        except KeyError:
            cull_rare = False
        for feature in self.feats.keys():
            D = self.observations_dual[feature]
            entropy = -sum(D[event] * 1.0 / self.feats[feature] *
                            log(D[event] * 1.0 / self.feats[feature])
                            for event in D)
            if entropy > threshold or (cull_rare and self.feats[feature] == 1):
                self.feats.pop(feature)
                self.observations_dual.pop(feature)
                for event in self.observations:
                    try:
                        self.observations[event].pop(feature)
                    except KeyError: pass
```

```python
def train(dataset):
    B = BayesianClassifier()
    for event, evidence in dataset:
        B.train(event, evidence)
    return B


def test(dataset, classifier, complement):
    correct = [0, 0]
    for event, evidence in dataset:
        correct[1] += 1
        guess, prob = classifier.classify(evidence, complement)
        if event == guess:
            correct[0] += 1
    return correct, correct[0]*1.0/correct[1]


def testsuite(threshold = None, cull_rare = False, complement = False):
    print 'Importing Dataset . . . .',
    from nltk.corpus import movie_reviews
    import time
    def word_feats(words):
        return set(word for word in words)

    negids = movie_reviews.fileids('neg')
    posids = movie_reviews.fileids('pos')
    negfeats = [['neg', word_feats(movie_reviews.words(fileids=[f]))]
                for f in negids]
    posfeats = [['pos', word_feats(movie_reviews.words(fileids=[f]))]
                for f in posids]
    negcutoff = len(negfeats)*3/4
    poscutoff = len(posfeats)*3/4
    trainfeats = negfeats[:negcutoff] + posfeats[:poscutoff]
    testfeats = negfeats[negcutoff:] + posfeats[poscutoff:]
    print 'DONE!'


    print 'Training Classifier . . .',
    B = train(trainfeats)
    if threshold:
        B.cull(threshold=threshold, cull_rare=cull_rare)
    print 'DONE!'
    print 'Testing . . . . . . . . .',
    t = time.time()
    R = test(testfeats, B, complement)
    t = time.time() - t
```

```
print 'DONE! (in %f seconds)'%t
print 'Accuracy . . . . . . . . ',
print (100*R[1]), '%\n', R

return B
```

References

[1] US Census Bureau. World vital events. preprint (2011), available at `http://www.census.gov/cgi-bin/ipc/pcwe`.

[2] Kdnuggets, news. preprint (2005), available at `http://www.census.gov/cgi-bin/ipc/pcwe`.

[3] C. Oberg. The impact of childhood poverty on health and development. *Healthy Generations*, 4(1), 2003.

[4] High risk individual counseling protocol for registered dietitians - inadequate weight gain or weight loss during pregnancy. Texas Department of State Health Services, 2007.

[5] F. Ziyo, F. Matly, G. Mehemd, and E. Dofany. Relation between prenatal care and pregnancy outcome at benghazi. *Sudanese Journal of Public Health*, 4(8), 2009.

[6] N. Almoussa, B. Dutra, B. Lampe, P. Getreuer, T. Wittman, C. Salafia, and L. Vese. Automated vasculature extraction from placenta images. In *Proc. SPIE Medical Imaging Conference. San Diego, CA*, 2009.

[7] American Pregnancy Association. Kdnuggets, news. preprint (2007), available at `http://www.americanpregnancy.org/duringpregnancy/fetallifesupportsystem.html`.

[8] C.M. Salafia, M. Yampolsky, D.P. Misra, O.Shlakhter, D. Haas, B. Eucker, and J. Thorp. Placental surface shape, function, and effects of maternal and fetal vascular pathology. *Placenta*, 31(11):958–962, 2010.

[9] M. Toal, S. Keating, G. Machin, J. Dodd, S.L. Adamson, R.C. Windrim, and J.C.P. Kingdom. Determinants of adverse perinantal outcome in high-risk women with abnormal uterine artery doppler images. *American Journal of Obstetrics and Gynecology*, 198(3):330.e1–330.e7, 2008.

[10] Christopher J.C. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2:121–167, 1998.

Acknowledgements

We would like to thank Dr. Salafia for providing us our data and the research topic. Also, we would like to thank Dr. Chang for provide us this opportunity to research this topic. Lastly, we would like to thank Amy Mulgrew for the data she provided us.