Predicting Cervical Cancer Cases Resulting in Biopsies Using Ensemble Methods

Laura Mann | 0582478

Friday, December 8th, 2017

COIS 4400: Data Mining Fall 2017

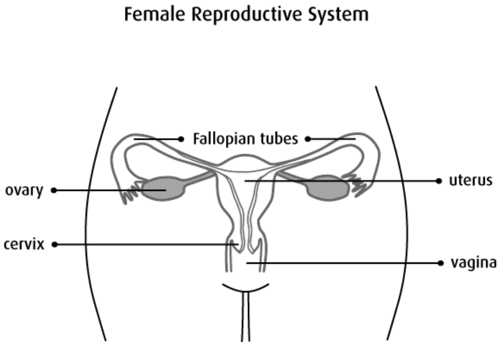
Sabine McConnell

Table of Contents

Description of Dataset

The dataset, "Cervical Cancer Risk Factors for Biopsy" was obtained from the UCI Repository. The data was collected by Kelwin Fernandes, Jamie S. Cardoso, and Jessica Fernandes in 2017 at 'Hospital Universitario de Caracas' in Caracas, Venezuela. The dataset contains habits, demographic information, and medical history of 858 patients from the hospital. There are many missing values in this dataset, due to many patients not answering questions because of privacy concerns. The dataset consists of 858 instances, with 36 attributes.

What is Cervical Cancer?



Cervical cancer is a malignant tumour starting in the cells of a woman's cervix, and possibly spreading or metastasizing to other parts of her body. The cervix is part of a woman's reproductive system, located below the uterus. In most cervical cancer cases, the tumours develop from precancerous changes in the cervix, and can take several years to develop. According to the Canadian Cancer Society, the known risk factors of cervical cancer are HPV, smoking, giving birth many times, sexual activity, weakened immune system, socio-economic status, DES, and oral contraceptives. All of these factors are considered in the dataset.

Although the number of cases of cervical cancer have been declining in recent years due to more advanced screening and early detection with the Pap test, 300,000 women worldwide die each year due to cervical cancer. However, there are multiple ways of treating cervical cancer, depending on the size of the tumour and the stage of the cancer. One of the treatment options is to have a biopsy. There are two types of biopsies that can be done: A Sentinel lymph node biopsy or a cone biopsy. A Sentinel lymph node biopsy is done in the early stages to see if there is cancer in the Sentinel lymph node. The cone biopsy is a surgery which removes a cone-shaped piece of tissue from the cervix and part of the endocervical canal. This surgery is done mainly for women who have Stage 1A1 cervical cancer, or women who still want to become pregnant. If the cancer has progressed further, some cervical cancer cases result in a trachelectomy, hysterectomy, pelvic exenteration, or ovarian transposition. This report focuses on predicting whether a woman will result in having a biopsy due to cervical cancer.

Attributes in the Dataset

There are 36 attributes in the dataset, consisting of 32 risk factors, and 4 target variables (the last four attributes):

Name	Type	Description
Age	Integer	Age of patient
Number of sexual partners	Integer	Total number of previous sexual partners
First sexual intercourse (age)	Integer	Age at which the patient first had sexual intercourse
Num of pregnancies	Integer	Total number of previous pregnancies
Smokes	Boolean	Yes/No

Smokes (years)	Real	Number of years they smoked
Smokes (packs/year)	Real	Number of packs they smoke per year
Hormonal contraceptives	Boolean	Yes/No: Have they used hormonal contraceptives
Hormonal contraceptives (years)	Real	Number of years they used hormonal contraceptives
IUD	Boolean	Yes/No: Have they had an IUD (form of hormonal contraceptive)
IUD (years)	Real	Number of years they used an IUD
STD	Boolean	Yes/No: Have they had an STD (Sexually Transmitted Disease)
STDs (number)	Integer	Number of STD's they've had
STDs: condylomatosis	Boolean	Yes/No: Have they had condylomatosis
STDs: cervical condylomatosis	Boolean	Yes/No: Have they had cervical condylomatosis
STDs: vaginal condylomatosis	Boolean	Yes/No: Have they had vaginal condylomatosis
STDs: vulvo-perineal condylomatosis	Boolean	Yes/No: Have they had vulvo-perineal condylomatosis
STDs: syphilis	Boolean	Yes/No: Have they had syphilis
STDs: pelvic inflammatory disease	Boolean	Yes/No: Have they had pelvic inflammatory disease
STDs: genital herpes	Boolean	Yes/No: Have they had genital herpes
STDs: molluscum contagiosum	Boolean	Yes/No: Have they had molluscum contagiosum
STDs: AIDS	Boolean	Yes/No: Have they had AIDS
STDs: HIV	Boolean	Yes/No: Have they had HIV
STDs: Hepatitis B	Boolean	Yes/No: Have they had Hepatitis B
STDs: HPV	Boolean	Yes/No: Have they had HPV
STDs: Number of diagnosis	Integer	Number of diagnoses of STDs
STDs: Time since first diagnosis	Integer	Years since first STD diagnosis
STDs: Time since last diagnosis	Integer	Years since last STD diagnosis
Dx: Cancer	Boolean	Yes/No: Have they had a dx test for cervical cancer
Dx: CIN	Boolean	Yes/No: Have they had a dx test for CIN (Cervical Intraepithelial Neoplasia)
Dx: HPV	Boolean	Yes/No: Have they had a dx test for HPV
Dx	Boolean	Yes/No: Have they had a dx test
Hinselmann: target variable	Boolean	Yes/No: Have they had a Hinselmann test (colonoscopy)
Schiller: target variable	Boolean	Yes/No: Have they had a Schiller test (using iodine to detect cancer cells)
Cytology: target variable	Boolean	Yes/No: Have they had a cytology-based test (Pap test)
Biopsy: target variable	Boolean	Yes/No: Have they had a biopsy

Bias in the Dataset

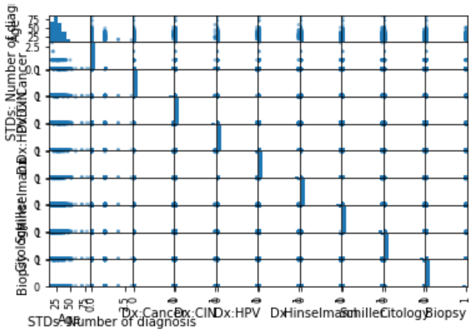
There is some bias in this dataset, based on the data collection method and the missing values. Since the data was only collected from one hospital in Vennezueala, the results could be very different than if the data had been collected from a hospital in Canada, or from multiple locations around the world. Although there are some attributes in the dataset corresponding to lifestyle choices which could result in a higher risk of cervical cancer (eg. smoking, hormonal contraceptives, ect.), there are still some major differences between Venezuealan women and Canadian women, not shown in the dataset, which could have some effect on the results (such as living conditions, access to health care, access to contraceptives, etc.).

Additionally, because there is some sensitive data involved, the issue of missing data introduces some bias to the dataset, since there is no right way to replace/remove those values.

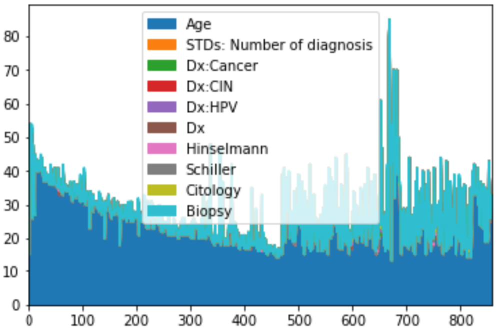
Visualization of the Data

Before doing any preprocessing, I created a few graphs in python with the dataset to see if any interesting correlations came up. Since there are many attributes which have binary values, it was difficult to show a good representation of the correlation between all attributes in the dataset. The first graph I plotted was the **scatter matrix**, as they are used to show every single pairwise scatterplot possible. The scatter matrix is advantageous in revealing strong correlations between specific attributes, or showing that two attributes don't have any sort of correlation. In looking at the "Biopsy" column (the furthest right), I could see many scatterplots that didn't have strong correlations, leading me to believe there are some potential attributes that are redundant or irrelevant and could be removed from training. The **area plot** graph shows each attribute on a different curve, and then stacks all of the curves on top of each other. It was interesting to see that the two dominating features on this graph were "Age" and "Biopsy". As a result of both of my previous findings, I plotted a histogram showing the relation between "Age" and "Biopsy", to which there is is a strong correlation. The graph shows that out of the women included in the dataset, most cases of cervical cancer resulting in a biopsy occurred when the women were between the ages of 20-30. Before doing any other investigating, this supports the fact given earlier that biopsies are often performed on women who still want to have children, since most women still wanting to have children would be between the ages of 20-30.

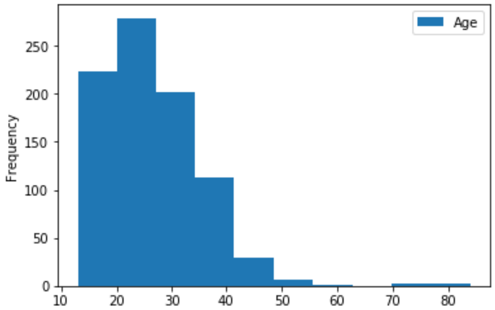
Scatter Matrix



Area Plot



Histogram Showing Age & Biopsy



Preprocessing

An advantage to this dataset is that it contains all numerical values, meaning that no transformations have to take place to convert categorical values to numerical values.

Standardization & Normalization

Due to the fact that some of the attributes in the dataset are binary values and some are not, standardizing and normalizing the dataset is advantageous in evening out the values and helping to solve imbalance problems. Standardizing makes the entire dataset have a specific value, and in this case, I standardized the mean and normalized the variance. It's important to do this do datasets of this nature so larger values don't dominate when applying models on them later. Additionally, many classification methods perform better when the dataset is standardized and/or normalized, and some even require the data to be standardized. Using a built-in function for standardizing/normalizing data in scikit-learn, the dataset now has a mean of -1.4389896095932891e-17 and a variance of 0.9561828874675149.

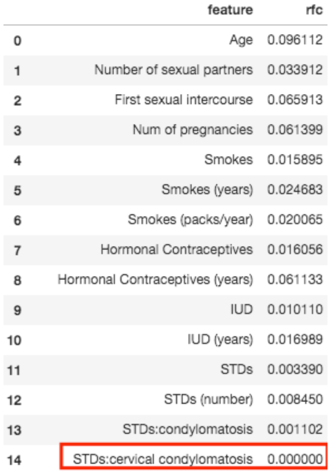
Missing Values

As mentioned, there are many missing values in this dataset, due to the fact that it contains very sensitive information which some women did not feel comfortable answering. One of the first preprocessing techniques I performed on the dataset was to deal with missing values. Because of the large number of missing values in this dataset, I decided to replace the missing values instead of eliminating them. This way I could be sure not to lose any important data, and try to make the most accurate estimate of the missing data. I chose to replace the values with the most frequent value of the row or column in which the missing value was located. This seemed more logical than choosing to replace the missing values with the mean or median of the values on the same axis, since it would skew the binary attributes to have a decimal in the mix.

Dimensionality Reduction

Another important step in preprocessing the data is trying to reduce the dimensionality of the dataset. Reducing the number of attributes in the dataset is very beneficial in removing irrelevant features and noisy data, preventing the curse of dimensionality, and making the dataset and its results easier to understand. As shown earlier when initially visualizing the data, it's apparent that it's easier to understand correlations between attributes and find unique relations when only comparing 2 or 3 attributes instead of all of them at once. In reducing the dimensionality of the dataset, we are not only making it easier to visualize and understand, but we're helping to improve the performance of the classifier being used on the data. In turn, this helps to prevent against the curse of dimensionality: the phenomenon that as the dimensionality of a dataset increases, the sparser the data becomes. This can be especially detrimental to data mining techniques that depend on the density of data, such as clustering.

One method of reducing the dimensionality of a dataset is feature subset selection, in which we only use a subset of the attributes. Any form of eliminating or ignoring data comes with the risk of losing important information that could affect the results, which is why it's important to only remove irrelevant and/or redundant data. During the initial visualization stage, it was apparent that there were some features that had very weak correlations with the “Biopsy” attribute, which is what we are trying to classify in the data. After applying a Random Forrest classifier on the dataset, I used the scikit-learn feature_selection_ tool to find out the important that each attribute has on the final performance of the model. The snippet of the results below shows a small portion of the attributes in the dataset and their importance, but you can see one attribute which has an importance value of 0, meaning that this attribute is irrelevant to the performance of the model, and should be removed (along with two other attributes). Looking at the importance of each attribute is also useful in revealing which attributes contribute *most* to the performance of the model. Although these are not the full results of the feature_selection process, it’s apparent that age has a large importance.



	feature	rfc
0	Age	0.096112
1	Number of sexual partners	0.033912
2	First sexual intercourse	0.065913
3	Num of pregnancies	0.061399
4	Smokes	0.015895
5	Smokes (years)	0.024683
6	Smokes (packs/year)	0.020065
7	Hormonal Contraceptives	0.016056
8	Hormonal Contraceptives (years)	0.061133
9	IUD	0.010110
10	IUD (years)	0.016989
11	STDs	0.003390
12	STDs (number)	0.008450
13	STDs:condylomatosis	0.001102
14	STDs:cervical condylomatosis	0.000000

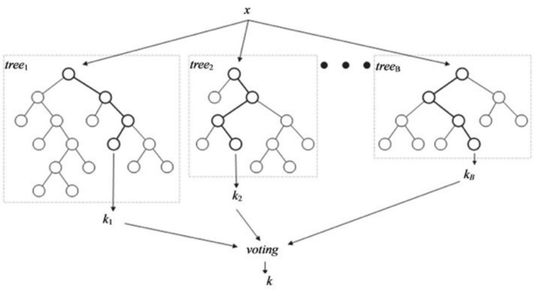
Discussion of Techniques

The supervised techniques I chose to apply to the dataset were the Random Forest Classifier, Support Vector Machine (SVM), and Gradient Boosting, as well as trying to use an Ensemble for those three classifiers. The unsupervised technique I chose to apply to the dataset was K Means Clustering.

Supervised Techniques

Random Forest Classifier

The first classifier I chose to use on the dataset was the Random Forest Classifier. Random Forest is an ensemble algorithm which creates many decision trees (a forest), and applies them to multiple subsets of the dataset, creating multiple classification results. The Random Forest Classifier uses a voting system to make its final classification prediction, with each tree voting, and chooses the class with the most votes. An alternative voting measure is using weights to assign the impact of a decision tree's result, with trees with high errors getting low weightings, and vice versa. In this voting system, trees with low error rates have a higher impact on the final classification decision.



The Random Forest Classifier splits the dataset into a training set and testing set by sampling with replacement, until approximately one-third of the data is remaining, which is used for testing the classifier. Before applying the classifier to the data, you must determine how many trees each forest should contain, and the minimum number of nodes required in order for the tree to split.

Some advantages of the Random Forest Classifier are that it's works well with noisy data and it reduces overfitting. Since the end result is an average or majority vote of multiple classification results, the classifier has a significantly lower chance of overfitting the data. Similarly, since there are multiple forests, not every forest is necessarily affected by noisy data. Some disadvantages to using a Random Forest Classifier is that they are much more complex than normal decision trees, thus are harder to understand and visualize. Additionally, because there are many more trees being created and used than in a normal decision tree, it is more computationally expensive.

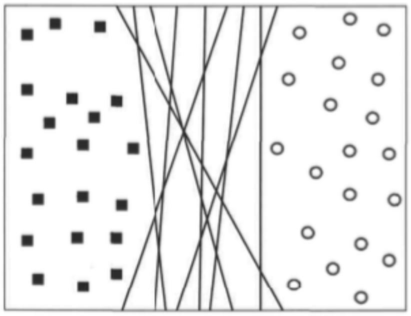
Gradient Boosting

The Gradient Boosting classifier is an ensemble classification technique which uses decision trees to make decisions on the class of the data, but can be optimized by using a loss function. Gradient Boosting comes from an idea that a weak learner (a learner in the model with a lower performance than average) can be modified to have better performance, and was preceded by Adaptive Boosting (AdaBoost), which used a weighting system to improve the performance of weak learners.

Gradient Boosting consists of three elements: a loss function, a weak learner, and an additive model. The loss function is used to determine which weak learner reduces the loss, so that the additive model can be applied. The weak learner used in a Gradient Boosting Classifier is decision trees, as they produce real values as output, and their output can be combined into another tree. Finally, the additive model is used ensure that the decision tree is moving in the right direction, by calculating the loss of each tree, and adding a tree to the model the reduces the loss.

Although Gradient Boosting classifiers usually perform well compared to other models, they are more complex and it can be harder to adjust the parameters. Additionally, since they are more complex, they can be harder to understand and analyze.

Support Vector Machine (SVM)



A Support Vector Machine is a classification technique which attempts to separate the different classes of data by finding a decision boundary which maximizes the margin. The SVM represents this boundary by using support vectors (a subset of the training data). Although there are infinitely many different hyperplanes that could be selected to separate the data, the hyperplane with the largest margin often performs better, as it leaves more room for any perturbations to the decision boundary without having an impact on the classification. Additionally, boundaries with larger margins are also less susceptible to overfitting.

Some advantages to using an SVM are that they work well with high-dimensional data, avoid the curse of dimensionality, and they still work well in cases where there are more dimensions than there are samples of data. However, a disadvantage to using SVMs is that they are harder to analyze, as they do not give out a probability score.

Unsupervised Technique

K-Means Clustering

K-Means clustering is an unsupervised approach to classifying data which tries to make clusters of similar data. Each data point is compared to randomly selected centroids, and placed in the neighbourhood of its nearest cluster (using Euclidean distance). The number of clusters (equal to the number of centroids) must be defined at the beginning of the model.

After selecting the initial centroids (usually chosen at random), the distances are computed and the data is assigned to centroid, and the centroids are recomputed multiple times until they don’t move around anymore.

K-means clustering performs well and is easy to understand the visualization of the data, However, k-means clustering doesn’t perform well when the data is of different sizes or densities, and has problems when the data contains outliers.

Results

Supervised Techniques

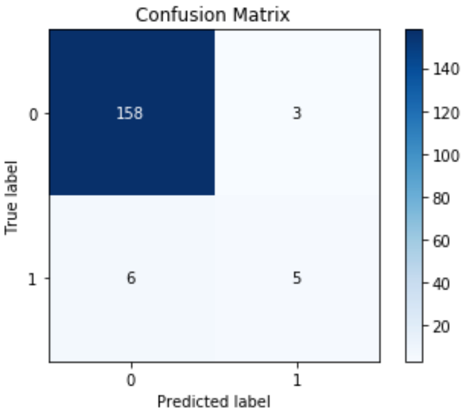
To analyze the results of the different supervised classifiers used on the test set, I will be using the following measures:

- **Precision (p):** The fraction of records that actually turns out to be positive in cases where Biopsy=true
- **Recall (r):** The fraction of positive examples correctly predicted by the classifier
- **Accuracy:** The degree of measurement error in the data
- **F_1 measure:** A combination of the precision and recall values, calculating a harmonic mean of both scores.
- **True negative rate:** The fraction of records that actually turns out to be negative in cases where Biopsy=false

Random Forest Classifier

The Random Forest Classifier ended up having an extremely high accuracy of 94.8%, after applying the preprocessing techniques discussed above. The confusion matrix for this classifier shows the higher number of values which the model correctly classified as false. Since there are not a lot of cases in which biopsy was true, the number of values correctly classified as true isn't that large. From the confusion matrix, we can calculate the following additional performance measures:

Accuracy = TP+TN/(TP+TN+FP+FN) = **94.8**
Precision (p) = TP/(TP+FP) = **38.5**
Recall (r) = TP/(TP+FN) = **45.5**
F_1 measure = 2rp/(r+p) = **41.7**
True negative = TN/(TN+FP) = **98.1**

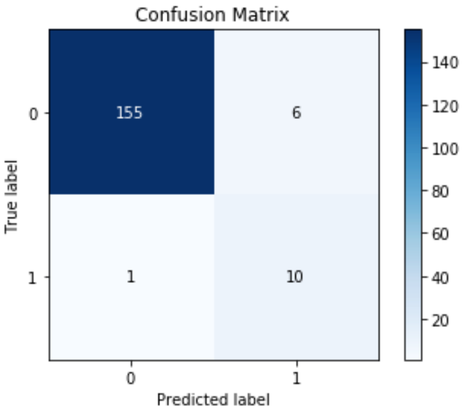


Although this model had a very high accuracy, when looking at the other performance measures calculated, it's apparent that this accuracy score is slightly skewed by the high amount True Negative value compared to the other low values in the confusion matrix. In regard to this medical data, it is useful to be able to predict when someone will not need to have a biopsy, but since the sample size of data including true values of the biopsy attribute is so small, it's hard to increase the True Positive value much more.

Gradient Boosting

The Gradient Boosting Classifier had an impressive accuracy of 95.9%, using the preprocessing techniques mentioned above, as well as playing around with the Gradient Boosting parameters. Since the Gradient Boosting Classifier is an ensemble method that uses trees, one of the parameters to adjust is the maximum depth of the trees allowed. Changing this value from 3 to 1 increased the accuracy by 0.5%. Although this classifier has the highest accuracy, it also has a very high True Negative value, skewing the accuracy slightly. However, the Gradient Boosting

Classifier has the highest True Positive value out of all models run on this dataset. From the confusion matrix, we can calculate the following performance measures:

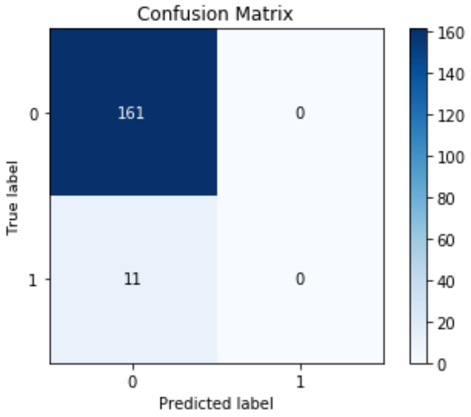


Accuracy = TP+TN/(TP+TN+FP+FN) = **95.9**
Precision (p) = TP/(TP+FP) = **62.5**
Recall (r) = TP/(TP+FN) = **90.9**
F_1 measure = 2rp/(r+p) = **74.1**
True negative = TN/(TN+FP) = **96.3**

Although the Gradient Boosting Classifier has the lowest True Negative of all classifiers, it has the largest True Positive value (although still low), which is reflective in it's high precision, recall, and F_1 scores.

Support Vector Machine

The Support Vector Machine (SVM) classifier also performed very well, with an accuracy score of 93.6%. However, just like in the Random Forest Classifier, this accuracy doesn't reflect the fraction of records that were correctly classified as true, as the True Negative value is so large. By looking at the precision, recall, and F_1 measure, we can see that the model failed to correctly classify *any* positive records.



Accuracy = TP+TN/(TP+TN+FP+FN) = **93.6**
Precision (p) = TP/(TP+FP) = **0**
Recall (r) = TP/(TP+FN) = 5/(5+6)= **0**
F_1 measure = 2rp/(r+p) = 2(0)/(0) = **0**
True negative = TN/(TN+FP) = **100**

The SVM has an impressive true negative score of correctly predicting 100% of the negative values in the test set, but has values of 0 for precision, recall, and $F_{1.}$

Unsupervised Technique

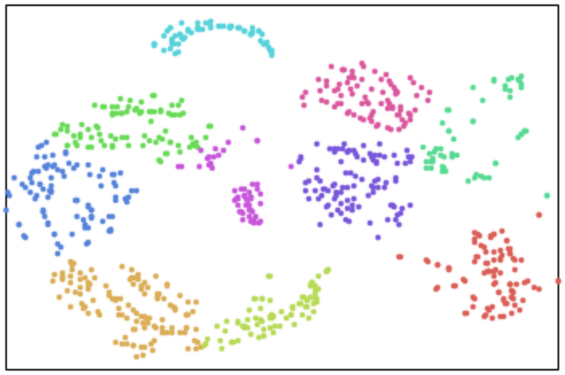
K-Means Clustering

I attempted using the unsupervised K-means clustering algorithm in two different ways:

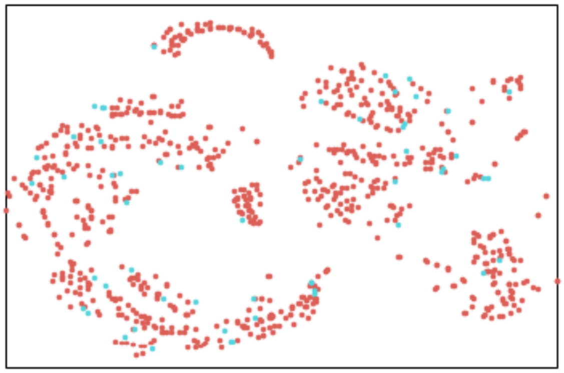
1. Specifying the number of clusters I wanted it to find and allowing the algorithm to choose those clusters on its own
2. Specifying that I wanted to show the X data (all columns except "Biopsy" and the y data (the "Biopsy" column) in the clustering

Using the K-means clustering feature in hypertools in python, one of the parameters I could tweak was how to reduce the dimensionality of the data. Although there were many options, the two illustrated below are t-distributed stochastic neighbor embedding (TSNE) and principal component analysis (PCA):

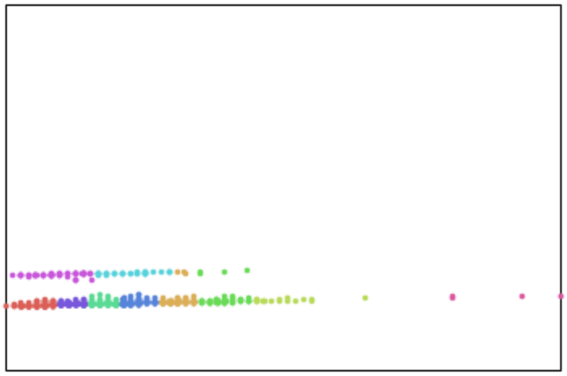
TSNE, 10 clusters



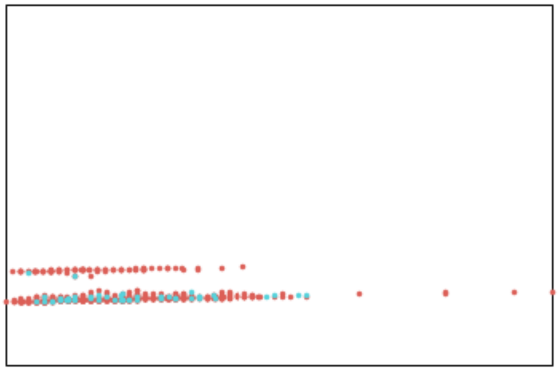
TSNE, splitting X and y



PCA, 10 clusters



PCA, splitting X & y



Looking at these four clusterings, it's apparent that the clusterings that used TSNE has split the data into clearer and more separate clusters, whereas the clusterings that used PCA are much closer together, making it harder to separate the data into different classes. However, in both cases, especially the TSNE case splitting X & y, the data points representing the y values of "Biopsy" (the blue values) are very spread out. This is a result of the dataset itself being fairly spread out, and a lack of positive "Biopsy" values.

Cumulative Discussion of Techniques

Although it can be difficult to compare the results of the supervised models with the unsupervised model, in this case, it is clear that the supervised Gradient Boosting Classifier performed better than the other three models. Not only did it have a higher overall accuracy than the Random Forest and Support Vector Machine classifiers, but it also had the highest precision, accuracy, and F_1 measures, signifying that it was better at correctly classifying the positive class. In comparing this to the unsupervised K-means clustering model, it's hard to find an exact difference in performance between the two, but the Gradient Boosting Classifier was able to clearly classify a large amount of the test set correctly, whereas the K-means cluster model struggled to separate the y values from the X values.

References

Kelwin Fernandes, Jaime S. Cardoso, and Jessica Fernandes. 'Transfer Learning with Partial Observability Applied to Cervical Cancer Screening.' Iberian Conference on Pattern Recognition and Image Analysis. Springer International Publishing, 2017.

https://www.kaggle.com/loveall/cervical-cancer-risk-classification

http://www.cancer.ca/en/cancer-information/cancer-type/cervical/cervical-cancer/?region=on

http://www.cancer.ca/en/cancer-information/cancer-type/cervical/treatment/surgery/?region=on#cone

https://www.stat.berkeley.edu/~breiman/RandomForests/cc_home.htm