Project update

by
Ana, Vladimir and Maxime
Tuesday 12 December

Our research

For our research we wanted to use the *Breast Cancer Wisconsin (Original) Data Set*, but we came upon a problem when we discussed how we want to write the algorithms. The features of the data set are already a classification. More precisely, the features are assigned to one of the 10 classes, but we do not know what these classes precisely represent. Also, we should try to represent the 10 classes as a binary classification which we could achieve by the one vs. all method. However, we decide that we could better work on another data set that has features representing numerical values instead of classes. Therefore, we have chosen to work on the *Wisconsin Diagnostic Breast Cancer (WDBC)*[1], where features are computed from a digitized image of a fine needle aspirate (FNA) of a breast mass. FNA is a way to examine a small amount of tissue from the tumor. They describe the characteristics of the cell nuclei present in the image.

Important information about our dataset:

- Number of instances: 569
- Number of features: 30 real-valued input features
    - radius (mean of distances from center to points on the perimeter)
    - texture (standard deviation of gray-scale values)
    - perimeter
    - area
    - smoothness (local variation in radius lengths)
    - compactness (perimeter^2 / area - 1.0)
    - concavity (severity of concave portions of the contour)
    - concave points (number of concave portions of the contour)
    - symmetry
    - fractal dimension ("coastline approximation" - 1)
- Number of labels: 2 classes
    - malignant (class distribution of 121 malignant)
    - benign (class distribution of 357 benign)

[1] https://archive.ics.uci.edu/ml/machine-learning-databases/breast-cancer-wisconsin/wdbc.names

We still want to predict the probability that a patient has a malignant breast tumor or not.

Approach

We have loaded our data as a CSV file into python such that we can better work with the data. Furthermore, we have tried several approaches to optimize our data, two approaches that we have used is normalization and standardization. By analyzing the outcomes of both approaches, we have seen that standardization gave us better accuracy, therefore we have chosen to use standardization instead of normalization.

Classifiers

We have tested all our classifiers by 10 fold cross-validation. Furthermore, we have also optimize our parameters by comparing the accuracies of the train and test data before and after optimization. In addition, we have printed a confusion matrix, ROC, and AUC. To avoid a messy project update report, we have decided that indicate the questions we have with our code in the code itself, such that it is clear where we're still struggling with. Also we have printed a confusion matrix, ROC, and AUC for each classifier in our code.

We have trained the following classifiers:

- Logistic regression
- Linear SVM
- Non-linear SVM
- Random Forest
- Neural Network

Literature review

Street, W. N., W. H. Wolberg, and O. L. Mangasarian. 1993. Nuclear Feature Extraction for Breast Tumor Diagnosis. no. July 1993: 861–70. doi:10.1117/12.148698.

In this article, the above data set has been used to classify malignant from benign. As mentioned previously, the data set exists of digitized images. The researchers have also used the 569 images and by using ten-fold cross-validation they obtained a

accuracy of 97%. The scholars mention that a “ten-fold cross-validation accuracy of 97% was achieved using a single separating plane on three of the thirty features: mean texture, worst area and worst smoothness.”

A critical remark can be made concerning this research, because this research has been done in 1993, meaning that they used a different learning algorithm for this specific classification problem. They used the Multi-surface Method (MSM) that is known as MSM-Tree (MSM-T). This is a decision tree algorithm. We use several algorithms in our project, whereas this paper is only concerned with one algorithm. This method uses a linear programming model to iteratively place a series of separating planes in the feature space. This is a very simple algorithm that separates the data points if their labels, either malignant or benign, are linearly separable by placing a plane between the two sets of data points. If this is not possible, a plane is places that minimizes the average distance of misclassified points to the plane.

We did find some useful information that we can take into account for our own project. As discussed in class, we could restrict the number of features and only use the features that are most crucial in rightfully classifying data as malignant or benign. The scholars argue that “in order to generate a classifier which generalizes well to unseen cases, we sought to minimize not only the number of separating planes but also the number of features used. The resulting single-plane classifier separates the points based on three feature values: mean texture and extreme values of area and smoothness.” We could find out whether this three features would also be most valuable for our project. The figure shows how the researchers places a plane separating the data points only using three features.

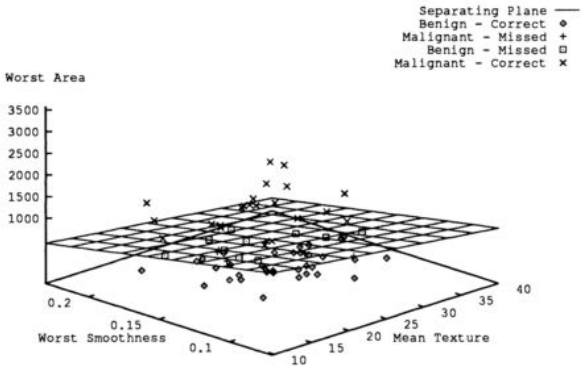

Figure 7: Separating Plane in Three Dimensions
In order to clarify the plot, only 10% of the correctly classified benign and malignant points are shown here. All of the misidentified points are shown.

Furthermore, they also used two functions in determining whether the plane is places, namely sensitivity and specificity. Sensitivity is the function that divides the correct positives by

the total positive, and specificity is the function that divides the correct negative by the total negative. These functions are important for us to analyze, because it is worst when malignant breast cancer is classified as benign, then when benign breast cancer is classified as malignant. With these two functions, we can take this important fact into consideration, meaning that we could train our algorithm in such a way that if misclassification is unavoidable, then we want the benign breast cancer to be classified as malignant instead of the other way around.

Agarap, Abien Fred. 2017. “On Breast Cancer Detection: An Application of Machine Learning Algorithms on the Wisconsin Diagnostic Dataset,” no. 1. http://arxiv.org/abs/1711.07831.

In this paper, the scholar has standardized the dataset to avoid inappropriate assignment of relevance by using *StandardScaler().fit_transform()*, which is precisely as we have done in our project. This means that we have support for our claim that it is better to use standardization instead of normalization. Furthermore, the researcher has manually assigned hyper-parameters. As he states, “in this study, the hyper-parameters set were not obtained through hyper-parameter optimization / tuning, but they were set by hand. To determine the most optimal hyper-parameters, cross validation (CV) techniques such as k-fold cross validation must be used” This is very important for our project, because it questions how we should set our parameters. Unfortunately, Agarap does not explain why he does not use hyper-parameter optimization or tuning.

Most importantly, this paper presents a comparison of six machine learning (ML) algorithms, namely GRU-SVM, Linear Regression, Multilayer Perceptron (MLP), Nearest Neighbor (NN) search, Softmax Regression, and Support Vector Machine (SVM) on *the Wisconsin Diagnostic Breast Cancer (WDBC)* dataset. For our project, we have also used some of the above ML algorithms, but not all. However, we can still compare the results in this paper with our own results. The table below represents his analysis for the different learning algorithms.

Table 2: Summary of experiment results on the ML algorithms.

Parameter	GRU-SVM	Linear Regression	MLP	L1-NN	L2-NN	Softmax Regression	SVM
Accuracy	93.75%	96.09375%	99.038449585420729%	93.567252%	94.736844%	97.65625%	96.09375%
Data points	384000	384000	512896	171	171	384000	384000
Epochs	3000	3000	3000	1	1	3000	3000
FPR	16.666667%	10.204082%	1.267042%	6.25%	9.375%	5.769231%	6.382979%
FNR	0	0	0.786157%	6.542056%	2.803738%	0	2.469136%
TPR	100%	100%	99.213843%	93.457944%	97.196262%	100%	97.530864%
TNR	83.333333%	89.795918%	98.732958%	93.75%	90.625%	94.230769%	93.617021%

For Logistic regression we obtain an accuracy of 98.2456140351 % which is very close to the softmax regression accuracy of 97.65625 %. The same holds for SVM, we obtain an accuracy of 96.4912280702% which is the (almost) the same as the SVM accuracy in the paper of 96.09375%. Nevertheless, we still need to verify whether the ML algorithms used in the paper are exactly the same as we use for our project.

Other important papers worth considering are:

Gouda I Salama, M Abdelhalim, and Magdy Abd-elghany Zeid. 2012. Breast cancer diagnosis on three different datasets using multi-classifiers. Breast Cancer (WDBC) 32, 569 (2012), 2.

Elias Zafiropoulos, Ilias Maglogiannis, and Ioannis Anagnostopoulos. 2006. A support vector machine approach to breast cancer diagnosis and prognosis. Artificial Intelligence Applications and Innovations (2006), 500–507.