CS229 Final Report

Identification of disease in corn/maize based on physiological traits

davidu@stanford.edu

Introduction

The department of plant pathology and plant-microbe biology at Cornell is studying maize/corn resistance to disease. (a genetic mechanism that provides plants with the natural ability to defend against pathogen attack) They have collected plant physiological traits such as leaf architecture, productivity, etc. Researchers working on this department would like to create a model to predict if a plant is healthy or not. This project is part of a large umbrella of food security related projects that aim to make contributions to increase crop productivity. Also, by studying disease resistance as a natural way to fight against pathogens, it will be addressing some environmental issues related to the indiscriminate use of pesticides, which is harmful to the environment and to humans.

This project is a Stanford-Cornell collaboration where Stanford provides machine learning knowledge and implementation and Cornell the datasets, subject matter experts, and guidance.

Dataset and Features

The dataset contains 19 features, one indicator of the plant being healthy or not, and 2519 samples. The dataset was split into 80% training data, 10% validation, and 10% test. The dataset was pre-processed and features that lacked data in more than 50% of the dataset were deleted, samples where one of the rows were empty were also deleted. The dataset was reduced to 10 features and 1011 samples, dates were converted to unix epoch timestamps.

In order to get an intuition of relations between features a couple of them were selected to be plot against, a correlation matrix (using Pearson correlation) was generated, and PCA was used to plot the features in a three-dimensional space.

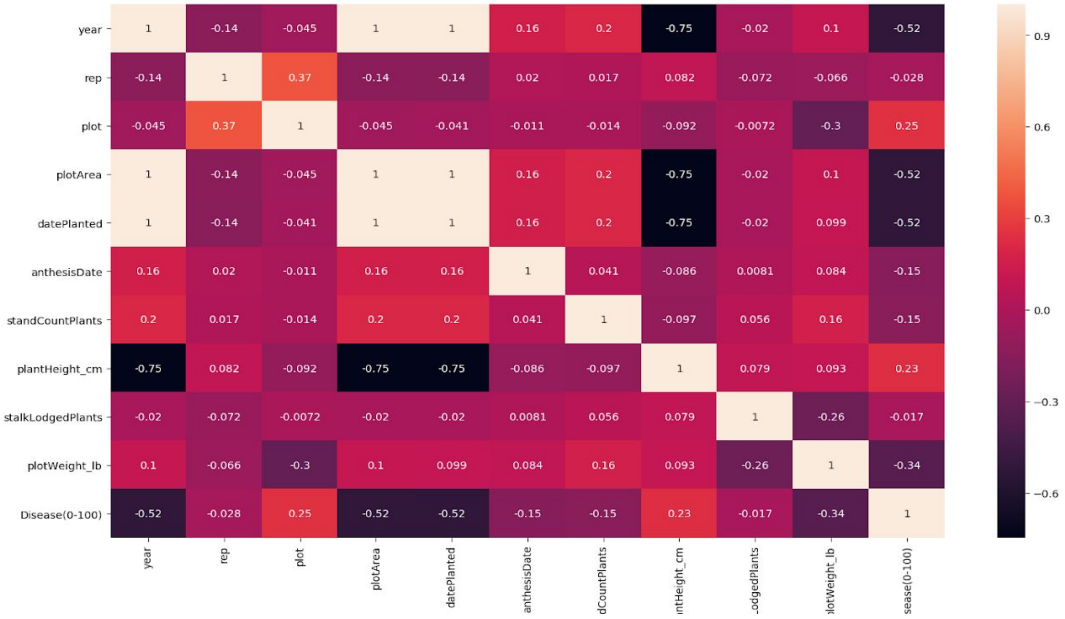

Figure 1: correlation matrix.

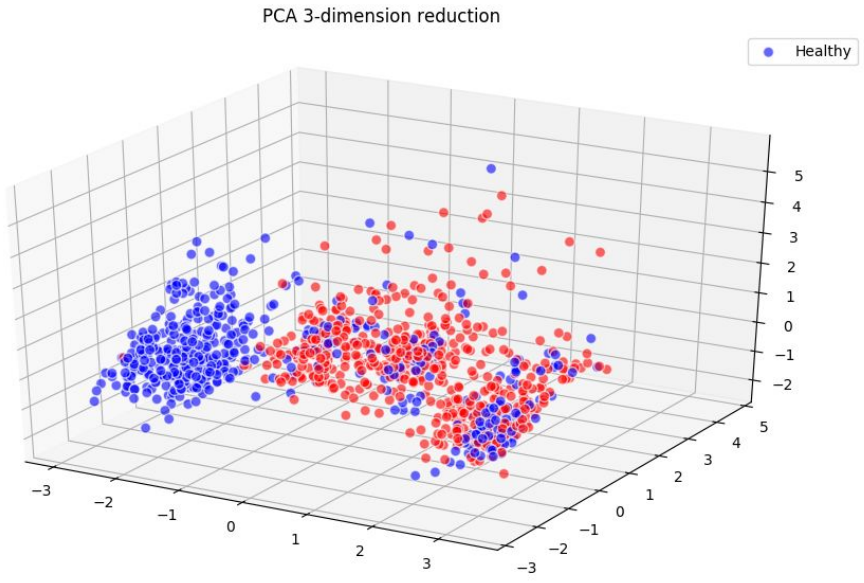

Figure 2: PCA 3-dimensions

Methods

Different supervised learning classification methods were tested and their effectiveness was measured using a confusion matrix and the F1 [Sasaki 2007] score.

The following methods were used:

Logistic regression

Logistic regression it's a classification method that uses the logistic function as its hypothesis function, maximizes the log likelihood between predictions and the true classification values of each example using gradient ascent on each theta parameter until the model converges and then it can be used to predict an output.

$$h_\theta(x) = g(\theta^T x) = \frac{1}{1+e^{-\theta^T x}}$$ Logistic function

$$\sum_{i=1}^{n} y^{(i)} \log h(x^{(i)}) + (1 - y^{(i)}) \log(1 - h(x^{(i)}))$$ Loss function

Support Vector Machines

A support vector machine (SVM) creates a hyper-plane in a high or infinite dimensional space, which can be used for classification, regression or other tasks. Intuitively, a good separation is achieved by the hyper-plane that has the largest distance to the nearest training data points of any class also known as a functional margin, since in general the larger the margin the lower the generalization error of the classifier. [Scikit-learn] Formally, SVMs can be defined as a dual optimization problem where an algorithm such as Sequential Minimal Optimization (SMO) [Platt 1998] can solve it and find the optimal set of support vectors.

Dual optimization problem

$$max_\alpha W(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} y^{(i)} y^{(j)} \alpha_i \alpha_j < x^{(i)}, x^{(j)} >$$

$$s.t.\ 0 \leq a_i \leq C,\ i = 1, ..., n$$

$$\sum_{i=1}^{n} \alpha_i y^{(i)} = 0$$

Random Forest

Random forests use multiple decision trees and a voting mechanism to determine if an example belongs or not to a class [Ho 1995]. Decision trees tend to overfit and random forests can prevent that by relying on the predictions of all the decisions trees used underneath. Concretely, at prediction time, prediction of unseen samples x' can be made by averaging the predictions from all the individual regression trees.

$$f' = \frac{1}{B} \sum_{b=1}^{B} f_b(x')$$

Artificial neural network

Artificial neural networks use layers of computations (matrix multiplication, sum, and application of an activation function on its output) named neurons which start with randomized weights and biases which overtime converge on optimal values for a given task (regression, classification) by using the back-propagation algorithm which transfers the error from a prediction back to all the weights at every layer of the network. In particular, for the classification problem, the network output layer uses sigmoid function. The Loss function used for forward and back-propagation [Werbos 1975] can be switched depending on the task.

$$z = w^T x + b$$ Neuron computation, weights 'w' and biases 'b'

$$a = RELU(z)$$ Hidden layer activation function

$$output = sigmoid(a)$$ Output layer activation function

Results and discussion

Logistic regression was used as a baseline but performed the lowest regarding F1 score compared to all other methods. SVM performed the best with the highest F1 score.

$$F1_{score} = 2.\frac{precision\ x\ recall}{precision + recall}$$

<table>
<tr><th>Method</th><th>F1 score</th><th>Confusion matrix</th></tr>
<tr><td>Logistic regression</td><td>82.70%</td><td>
<table>
<tr><td>Pred/Actual</td><td>Healthy</td><td>Unhealthy</td></tr>
<tr><td>Healthy</td><td>48</td><td>13</td></tr>
<tr><td>Unhealthy</td><td>10</td><td>55</td></tr>
</table>
</td></tr>
<tr><td>Random Forest</td><td>89.36%</td><td>
<table>
<tr><td>Pred/Actual</td><td>Healthy</td><td>Unhealthy</td></tr>
<tr><td>Healthy</td><td>48</td><td>13</td></tr>
<tr><td>Unhealthy</td><td>2</td><td>63</td></tr>
</table>
</td></tr>
<tr><td>Artificial Neural Network</td><td>89.85%</td><td>
<table>
<tr><td>Pred/Actual</td><td>Healthy</td><td>Unhealthy</td></tr>
<tr><td>Healthy</td><td>47</td><td>14</td></tr>
<tr><td>Unhealthy</td><td>1</td><td>64</td></tr>
</table>
</td></tr>
<tr><td>Support Vector Machines</td><td>90.27%</td><td>
<table>
<tr><td>Pred/Actual</td><td>Healthy</td><td>Unhealthy</td></tr>
<tr><td>Healthy</td><td>47</td><td>14</td></tr>
<tr><td>Unhealthy</td><td>0</td><td>65</td></tr>
</table>
</td></tr>
</table>

Conclusion/Future work

For the small size of the dataset many methods provided a good F1 score, SVMs performed the best. My expectation was for neural networks to not perform as good as SVMs given deep learning often requires large datasets. In my tests I used a shallow neural network of 2 layers and its F1 score was almost the same as SVMs.

The future work on this project involves classifying images of healthy and unhealthy corn plants with a convolutional neural network and then compare its performance with the results of this project.

Contributions

I'm the only contributor for this project on the machine learning part but I had support from two researchers: Danilo Moreta and Chloe Siegel under Professor Rebecca Nelson from Cornell who are subject matter experts.

Code:

https://github.com/daviduvalle/cs229

References

[Sasaki 2007] The truth of the F-measure, https://www.toyota-ti.ac.jp/Lab/Denshi/COIN/people/yutaka.sasaki/F-measure-YS-26Oct07.pdf
[Scikit-learn] Documentation on SVMs, https://scikit-learn.org/stable/modules/svm.html
[Platt 1998] Sequential Minimal Optimization, https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/tr-98-14.pdf
[Ho 1995] Random Decision Forests, https://web.archive.org/web/20160417030218/http://ect.bell-labs.com/who/tkh/publications/papers/odt.pdf
[Werbos 1975] *Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences*