Classifying Cancer Cells Based On Geometrical Measurements

Eli Brignac, Jonathan Ma, Xiaofan Li

University of Delaware

Abstract

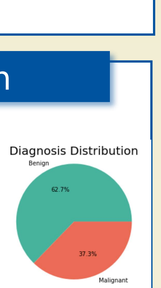
- The project aimed to predict tumor cell nature (benign/malignant) using geometrical measurements. Two significant models were created, achieving 98.83% and 97.08% accuracy respectively. Both models outperformed the previous SVM, one of which highlighted perimeter measurements as crucial in determining tumor cell malignancy.

Introduction

- We developed two MLP models that accurately predict whether a cell is benign or malignant based on geometric measurements. MLP models were chosen for their superior classification performance compared to Random Forest. The first model utilizes data from all 10 categories, while the second model focuses on a single category. This research suggests that by geometrically measuring a small biopsy, our models can aid in determining if the cells are cancerous, complementing human evaluation.

Dataset Information

Category	Attribute 1 (Mean)	Attribute 2 (SE)	Attribute 3 (Worst)
Radius	radius_mean	radius_se	radius_worst
Texture	texture_mean	texture_se	texture_worst
Perimeter	perimeter_mean	perimeter_se	perimeter_worst
Area	area_mean	area_se	area_worst
Smoothness	smoothness_mean	smoothness_se	smoothness_worst
Compactness	compactness_mean	compactness_se	compactness_worst
Concavity	concavity_mean	concavity_se	concavity_worst
Concave Points	concave points_mean	concave points_se	concave points_worst
Symmetry	symmetry_mean	symmetry_se	symmetry_worst
Fractal Dimension	fractal_dimension_mean	fractal_dimension_se	fractal_dimension_worst



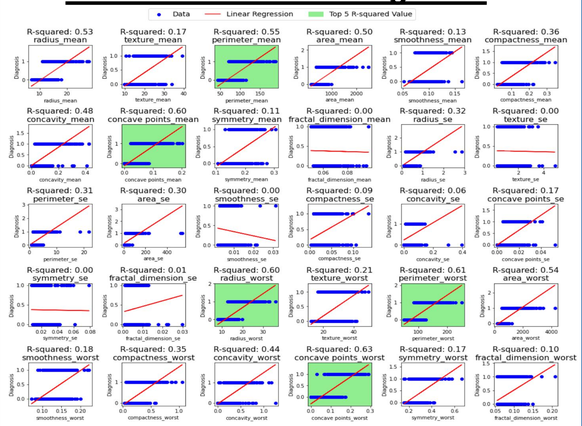
- 569 total examples
- 10 input categories each with 3 decimal attributes. 30 measurements total
- All data was normalized using StandardScalar from sklearn

Methodology

• Model 1 Best Possible Model

- Use all 30 inputs
- Be thorough with the optimization process

• Model 2 Fewest Categories



1. We first performed Linear regression between each attribute and the diagnosis
2. We looked at the 5 attributes with the largest R^2 value, these attributes are highlighted in green

radius attributes	perimeter attributes	concave attributes
radius_worst	perimeter_worst	concave points_worst
	perimeter_mean	concave points_mean

3. Seeing that Perimeter attributes and concave attributes were most prevalent, we decided to make 2 MLP models using only those attributes

• Hyperparameter Optimization

1. Fix batch size, learning rate, and random_state.
2. Generate a random tuple with a length between 2 and 6.
3. Fill the tuple with random values ranging from 10 to 100
4. Use the generated tuple as the new shape for the model
5. Train and test the model using this new shape.
6. If the accuracy of this model is better than all previous models, save its shape.
7. Repeat steps 2-6 for 100 iterations to find the shape that yields the best model.
8. Examine learning curve to identify signs of overfitting
9. If overfitting occurs, adjust the batch size or learning rate and repeat steps 3-6.

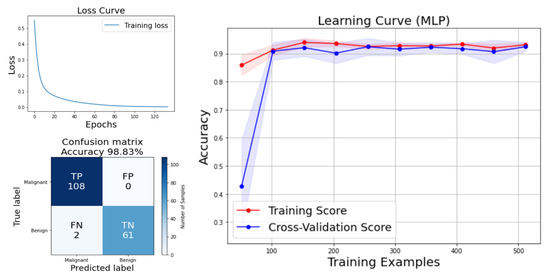
• Metrics

1. Loss curve to show the performance of a model during the training process
2. Confusion matrix to evaluate performance on the test dataset and classify TP, FP, TN, and FN predictions.
3. Learning curve to check for signs of overfitting
   - Uses 10 fold cross validation

Results

• Model 1 Best Model Possible

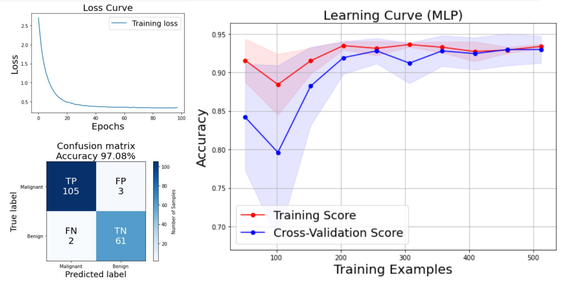
```
MLPClassifier(hidden_layer_sizes=(16,32), max_iter=1000, activation='relu',
              solver='adam', batch_size = 32, random_state = random_state)
```



- 98.83% accuracy on testing set
- No sign of overfitting
  - Small gap between the 2 curves

• Model 2.1 Perimeter Attributes

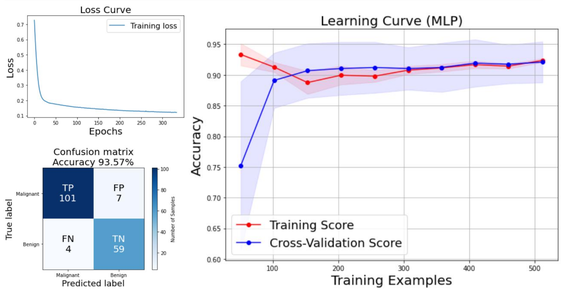
```
MLPClassifier(hidden_layer_sizes=(32, 28, 80, 86, 20, 30), max_iter=1000, activation='relu',
              solver='adam', batch_size = 32, random_state = random_state, alpha = .6)
```



- 97.08% accuracy on testing set
- No strong signs of overfitting
  - Small gap between the 2 curves
- Only 1 attribute Category
  - Used the metrics from the Perimeter category
  - Only ~1.75% drop in performance

• Model 2.2 Concave Attributes

```
MLPClassifier(hidden_layer_sizes=(39, 35, 45, 68), max_iter=1000, activation='relu',
              solver='adam', batch_size = 64, random_state = random_state)
```



- 93.57% accuracy on testing set
- No strong signs of overfitting
  - Small gap between the 2 curves
- Worse than Perimeter Model
  - This model is insignificant
  - It is also worse than the SVM models

Related Work

- Original SVM
- A previous submission used an sklearn SVM with default parameters to classify the dataset
  - Radial Basis Function Kernel (rbf kernel)
  - C = 1
- Achieved 93.86% accuracy on the testing set

- Tweaked SVM
- We tweaked their model to try to improve it
  - Kernal change:
    - rbf → linear
  - C change:
    - C = 1 → C = 5
  - Achieved 96.49% accuracy on the testing set

Conclusion

Model	Accuracy %	Inputs	Measurement Categories
Model 1	98.83	30	10
Model 2.1 (perimeter)	97.07	3	1
Model 2.2 (concave)	93.57	3	1
Original SVM	93.86	30	10
Tweaked SVM	96.49	30	10

- In the end, we accomplished our goal and were able to create 2 MLP models that perform better than both the original SVM and the Tweaked SVM.

- Our second model suggests that the Perimeter metrics are the most important geometrical measurements to look at when classifying tumor cells as benign or malignant.

References

[1] M.M Islam et al. 2020. Breast Cancer Prediction: A Comparative Study Using Machine Learning Techniques. *SN COMPUT. SCI.* 1, 290 (Sept. 2020), 290. https://doi.org/10.1007/s42979-020-00305-w

[2] Sungroh Yoon Seonwoo Min, Byunghan Lee. 2017. Deep Learning in Bioinformatics. *Briefings in Bioinformatics* 18, 5 (Sept. 2017), 851–869. https://doi.org/10.1093/bib/bbw068

[3] Zheng Rong Yang. 2004. Biological Applications of Support Vector Machines. *Briefings in Bioinformatics* 5, 4 (Dec. 2004), 328–338. https://doi.org/10.1093/bib/5.4.328