Diagnosing Malignant versus Benign Breast Cancer Tumors via Machine Learning Techniques in High Dimensions

CS 229 Final Project

Danielle Maddix

Contact Information:
Institute of Computational and
Mathematical Engineering (ICME)
Stanford University
Email: dcmaddix@stanford.edu

Introduction

- Breast tumor classification problem
- Predictive medical procedures that are available for diagnosis, but one test may not be definitive enough and there can be error margins of either false positives or false negatives.
- An algorithm which could take into account many test diagnostics and make a prediction has potential to have a broad impact in the medical field.
- In fact, the medical literature is already becoming rich in such methods, with the potential goal of submitting patients to fewer extensive testing.
- Size of the tumor is predicative feature but not definitive enough
- Requires higher dimensional problem, which can factor into many medical diagnostic tests and measurements.

Predicting

- *Supervised Learning*: Data has the negative labels $y_i = 0$ for Benign and positive labels $y_i = 1$ for Malignant
- *Input*: Training set (x_i, y_i) consisting of n dimensional features x_i and corresponding labels y_i.
- *Output*: Classify a breast tumor with negative label 0 for benign or positive label 1 for malignant

Data

- Publicly available from the UC Irvine Machine Learning Repository Breast Cancer Wisconsin (Diagnostic) Data Set
- 569 rows, each representing a patient and the characteristics of their tumor.
- 32 columns: 1^{st} column is the patient ID, 2^{nd} column is the label
- The remaining 30 columns form the continuous feature vector x_i in the training example (x_i, y_i).
- 357 benign samples, 212 malignant samples

Features

- 30 continuous features, all of which are from the raw input data
- 10 measurable features and 3 categories for each: their mean, standard error and worst case or largest value.
- The first 10 columns of the design matrix are the measured means for each respectively, the next 10 columns are the standard error and the final 10 are the worst case or largest value, as measured for each medical image.
- 10 distinct continuous features measured:

1. radius
2. texture
3. perimeter
4. area
5. smoothness
6. compactness
7. concavity
8. concave points
9. symmetry
10. fractal dimension.

Models

Binary classification model with labels and continuous inputs: Logistic Regression, Linear and Quadratic GDA, SVM

1. *Logistic Regression*
   - Discriminative learning algorithm directly modeling the conditional probability $p(y|x)$
   - The fitting parameters $\theta \in \mathbb{R}^{n+1}$, including the intercept terms are computed via the maximum likelihood estimators and then an optimization algorithm is used to find the optimal θ.
   - Hypothesis: $h(\theta^T x) = \frac{1}{1+e^{-\theta^T x}}$
   - Optimization Algorithm is used to find the optimal θ: Newton vs Gradient Ascent
   - Newton preferred method: fewer iterations for convergence / no learning rate parameter α
   - Modification: Normalized/Scaled design matrix $X \in \mathbb{R}^{mx(n+1)}$ to avoid poor numerical properties of ill-conditioned Hessian
2. *Gaussian Discriminant Analysis (GDA)*
   - Generative learning algorithm: first builds model for $p(x|y = 1)$, the positive class of malignant tumors and also builds model for $p(x|y = 0)$, the negative class of benign tumors
   - Learns $p(y|x)$ using Baye's Rule:

$$p(y|x) = \frac{p(x|y)p(y)}{p(x|y=0)p(y=0) + p(x|y=1)p(y=1)} \quad (1)$$

   - The posterior densities are assumed to be Multivariate Gaussians with means μ_1 and μ_0, respectively
   - Linear: Covariance matrix Σ: $x|y=0 \quad \mathcal{N}(\mu_0, \Sigma)$, $x|y=1 \quad \mathcal{N}(\mu_1, \Sigma)$
   - Quadratic: Covariance matrices Σ_0 and Σ_1: $x|y=0 \quad \mathcal{N}(\mu_0, \Sigma_0)$, $x|y=1 \quad \mathcal{N}(\mu_1, \Sigma_1)$
   - Prior density $p(y)$ is Bernoulli distributed, i.e. $y \quad$ Bernouilli(ϕ)
3. *SVM*
   - Maximize functional and geometric margin
   - Optimal Margin Classifier: Solves optimization problem to maximize the distance between the points and decision boundary
   - Primal:

$$\min_{\gamma,w,b} \frac{1}{2}\|w\|^2 \quad (2)$$

$$\text{s.t. } y^{(i)}(w^T x^{(i)} + b) \geq 1, \text{ i = 1, . . ., m} \quad (3)$$

   - Liblinear-1.94: *train* and *predict* functions

Results

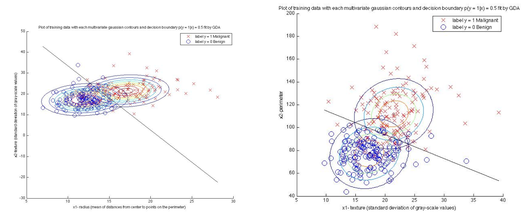

Figure 1: Linear GDA in 2D: Texture vs Radius / Perimeter vs Texture

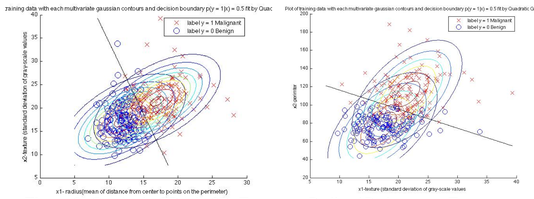

Figure 2: Quadratic GDA in 2D: Texture vs Radius / Perimeter vs Texture

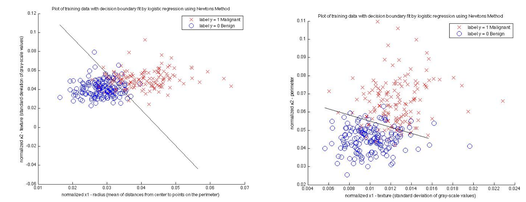

Figure 3: Logistic Regression in 2D: Texture vs Radius / Perimeter vs Texture

Linear GDA	**1**	**9**	**19**	**29**
Hold-out CV	6.47%	5.88%	2.94%	2.35%
k-fold CV	12.50%	7.32%	6.96%	4.46%
Recall	85.0%	94.87%	94.87%	92.31%
Precision	87.18%	82.22%	92.50%	97.30%

Table 1: Linear GDA

Quadratic GDA	**1**	**9**	**19**	**29**
Hold-out CV	7.65%	2.35%	2.35%	3.53%
k-fold CV	17.32%	15.89%	15.54%	12.32%
Recall	74.36%	92.31%	89.74%	84.62%
Precision	90.62%	97.30%	100.0%	100.0%

Table 2: Quadratic GDA

Logistic Regression	**1**	**9**	**19**	**29**
Hold-out CV	14.12%	9.41%	6.47%	5.29%
Number of Iterations	8	11	12	12
k-fold CV	13.39%	6.79%	5.89%	5.36%
Recall	89.74%	94.87%	97.44%	97.44%
Precision	63.64%	72.55 %	79.17%	82.61%

Table 3: Logistic Regression with Newton's Method

SVM	**15**	**25**	**30**
Hold-out CV	7.65%	4.71%	4.12%
Opt Accuracy	92.3529%	95.2941%	95.8824%
k-fold CV	11.07%	10.71%	7.32%
Recall	69.23%	89.74%	92.31%
Precision	96.43%	89.74%	90.00%

Table 4: Linear SVM

Discussion

- GDA algorithm seems to product the best results: even better than Logistic Regression and SVM which in general make less assumptions and are more robust
- This shows data is actually distributed as a multivariate Gaussian, explaining GDA's success
- Linear GDA is good approximation to data: Σ_0 and Σ_1 very close in values of elements and so in quadratic GDA appears as linear, since quadratic term $\frac{1}{2}x^T(\Sigma_0^{-1} - \Sigma_1^{-1})x$ is very small and linear term dominates
- Generally for all models, there is decrease in error measurements with increased dimensional feature space: SVM specifically does better in higher dimensional feature space, as included in the table

Future

- *Additional Algorithms*
  - *Discriminative*: Bayesian Logistic Regression/ Logistic Regression with Regularization
  - *Generative*: Multinomial Naive Bayes with Laplace smoothing, where features are discretized within certain ranges
  - *Nonlinear decision boundaries*: higher dimensional polynomials for a model with less bias
- *Feature Selection*: Find optimal subset of features relevant for tumor diagnosis to solve in smaller subspace
- *Further Datasets*:
  - 10 more biological than physical based features: comparison between which dataset would give more definitive results
  - Features to determine whether a tumor is more likely to recur or non-recur

References

[1] UCI Machine Learning Repository: Center for Machine Learning and Intelligent Systems. Breast cancer wisconsin (diagnostic) data set: wdbc.data. http://archive.ics.uci.edu/ml/machine-learning-databases/breast-cancer-wisconsin/wpbc.data, 1996.

[2] UCI Machine Learning Repository: Center for Machine Learning and Intelligent Systems. Breast cancer wisconsin (diagnostic) data set: wdbc.names. http://archive.ics.uci.edu/ml/machine-learning-databases/breast-cancer-wisconsin/wdbc.names, 1996.

[3] Andrew Ng. Lecture notes 1-5. http://cs229.stanford.edu/materials.html, 2014.