Justin Hardy

Benjamin Frenkel

CS 4375.003

Dr. Karen Mazidi

ML from Scratch Assignment Reflection

I. PROGRAM OUTPUTS

The following images are outputs of our from-scratch machine learning algorithm program, performed on the given titanic data:

Logistic Regression

```
Linear Model Options:
1. Logistic Regression
2. Naive Bayes

Select a linear model option (# from above) (0 to close):
>1

Selected Logistic Regression.

Iterating weights... (0%)
Iterating weights... (5%)
Iterating weights... (10%)
Iterating weights... (15%)
Iterating weights... (20%)
Iterating weights... (25%)
Iterating weights... (30%)
Iterating weights... (35%)
Iterating weights... (40%)
Iterating weights... (45%)
Iterating weights... (50%)
Iterating weights... (55%)
Iterating weights... (60%)
Iterating weights... (65%)
Iterating weights... (70%)
Iterating weights... (75%)
Iterating weights... (80%)
Iterating weights... (85%)
Iterating weights... (90%)
Iterating weights... (95%)
Iterating weights... (100%)

Weights:

Intercept       0.999877
sex1            -2.41086

Metrics:

        Ref
Pred    0       1
0       113     35
1       18      80

Accuracy: 0.784553
Sensitivity: 0.695652
Specificity: 0.862595
Training Time: 13s

Program operations complete. Exiting...

Process finished with exit code 0
```

Naive Bayes

```
Linear Model Options:
1. Logistic Regression
2. Naive Bayes

Select a linear model option (# from above) (0 to close):
>2

Selected Naive Bayes.

Coefficients:

A-priori Probabilities:
Y
0       1
0.61    0.39

Conditional Probabilities:
age
Y       [,1]    [,2]
0       30.418  14.323
1       28.826  14.462

pclass
Y       1       2       3
0       0.17213 0.22541 0.60246
1       0.41667 0.26282 0.32051
sex
Y       0       1
0       0.15984 0.84016
1       0.67949 0.32051

Metrics:

        Ref
Pred    0       1
0       113     35
1       18      80

Accuracy: 0.78455
Sensitivity: 0.69565
Specificity: 0.8626
Training Time: 149700ns

Program operations complete. Exiting...

Process finished with exit code 0
```

II. RESULTS ANALYSIS

As it is apparent in the images of the program's outputs, we tried our best to get the metrics - that were asked for - to look as closely to the format that R outputs them in for easy comparison. Let's look over both outputs separately, and compare them to an output on the same data set, using the same target/predictors, done on R. Note that values calculated by R will be referenced, but not explicitly shown.

Logistic Regression was an interesting model to recreate 'from the ground up'. The program utilizes the technique described to us by the professor, involving performing gradient descent to calculate accurate weight values for both the target's intercept, and the predictor's estimate(s). Of course, we only had one predictor with logistic regression, so this process was much easier to perform than if we had multiple predictors. However, the main challenge came with the matrix calculations we had to perform throughout the process. Once this process is complete, we get an intercept of 0.999877 and an estimate of -2.41086 for a positive sex value.

These values in particular were close to, but not exactly the same as the values that R calculated in their use of the function, although this seemed to not make much of a difference as the predictions were all the same as R's, meaning the metrics matched as well. As an aside, the training time is largely dependent on how fast your computer can perform the x number of iterations, and will vary by computer processing speeds. Moreover, the model we created from-scratch predicted with an approximate 78.5% accuracy, which goes to show that the model could (fairly) accurately use *just* the passenger's sex to predict their survival.

Naive Bayes was, interestingly enough, far easier to implement. This is likely due to the fact that the model is generative. With this model, we simply had the program count the number of survivors/deaths in relation to the column value, and computed probabilities of survivals for each attribute value. Some notable trends we saw were that if you survived, you were more likely to be of class 1 than any other class, and likewise if you survived, you were more likely to be of sex 0 (...whatever sex that is supposed to represent). Interestingly enough, the model ended up making the exact same predictions as the Logistic Regression model, and therefore got the same metrics as it. However, it performed significantly faster than the Logistic Regression model, as it did not need to perform gradient descent on weights.

Overall, we can see that the Naive Bayes model is much faster than the Logistic Regression Model. However, we know from our studies that the Naive Bayes model may lose its accuracy in other data sets, due to its generative nature.

III. GENERATIVE VS. DISCRIMINATIVE CLASSIFIERS

Generative and discriminative classifiers are two quite different things, "Logistic regression directly estimates the parameters of P(Y|X). This is called a discriminative classifier. Naive Bayes directly estimates parameters for P(Y) and P(X|Y). This is called a generative classifier.". There are several differences between these two types of classifiers.

One difference is that Naive Bayes in general works better for smaller data sets and logistic regression does better with larger data sets. Another difference is the bias and variance. Naive Bayes has higher bias but lower variance than logistic regression. Generally, Naive Bayes is computationally faster than logistic regression, but due to its generative nature, doesn't carry over into other variations of the data set as well as logistic regression might.

IV. "REPRODUCIBLE RESEARCH IN MACHINE LEARNING"

Much like in the science community, machine learning must prioritize computational reproducibility within their published data, models, and code. Simply put, reproducible research is simply defined by research that is able to be verified and easily by other researchers. It's considered very important for your research to be producible within the science community, and the same is true for researchers in the machine learning community, as it is a derivative of both computational sciences and mathematical statistics. In an article by *Nature Methods*, it's highlighted that reproducibility in machine learning is trivial for "researchers [to] ensure the accuracy of reported results and detect biases in the models," which promote both integrable and accurate data models. Furthermore, they classify computational reproducibility via a "sliding scale, that reflects the time needed to reproduce;" notating a bronze, silver, and gold standard as various "degrees of rigor" over this concept [1].

In an article from 2012 titled *Reproducible Research for Scientific Computing*, the authors point out - similarly to the previous article - how implementation must be held to a variety of standards to promote reproducibility. Specifically, the 2012 article points out various barriers to implementing reproducibility, such as code cleanup time and preparation time, and suggested "reducing the barrier [by] developing tools that more easily capture experimental details and facilitate the communication of the environment, algorithm, data, and reasoning to collaborators and the public” [2]. Overall, taking the time to clean up code and prepare it for sharing/publishing will allow for greater reproducibility of research.

Works Cited

[1] B. J. Heil, M. M. Hoffman, F. Markowetz, S. Lee, C. S. Green, and S. C. Hicks, "Reproducibility Standards for Machine Learning in the Life Sciences," *Nature Methods*. Aug 2021, doi: https://doi.org/10.1038/s41592-021-01256-7.

[2] R. J. LeVeque, I. M. Mitchell, and V. Stodden, "Reproducible Research for Scientific Computing: Tools and Strategies for Changing the Culture," *IEEE CS and the AIP*. 2012.