Kelly Trinh

Dr. Karen Mazidi

CS 4375.004

C++ Algorithms from Scratch - Overview Document

Code Outputs

Logistic Regression:

```
Opening file titanic_project.csv.
Readling line 1
heading: "","pclass","survived","sex","age"
new length 1046
Closing titanic_project.csv file
Number of records: 1046
intercept: 1.00711
sex slope: -3.07078
accuracy: 0.784553
sensitivity: 0.695652
specificity: 0.862595
elapsed time to run this program: 9 seconds
```

Naive Bayes:

```
Number of records: 1046

prior probability, survived=no, survived=yes: 0.61 0.39

likelihood values for p(sex|survived):
    [1]    [2]
[1] 0.159836 0.840164
[2] 0.679487 0.320513

likelihood values for p(pclass|survived):
    [1]    [2]    [3]
[1] 0.172131 0.22541 0.602459
[2] 0.416667 0.262821 0.320513

means:
[1] 30.3914 28.8077

variances:
[1] 205.15 209.989

Predictions:
0.887529        0.112471
0.807452        0.192548
0.414372        0.585628
0.24422        0.75578
0.244239        0.755761
The program '/Users/kellytrinh/Desktop/school/C++ Algorithms from Scratch/naiveBayes' has exit
```

```
accuracy: 0.78455
sensitivity: 0.69565
specificity: 0.8626
elapsed time to run this program: 0 seconds
```

*Note: the elapsed time began counting after the data was read into the vectors used in the program.

Analysis

After completing the logistic regression and Naive Bayes models on the Titanic data set, we are able to see many things. With the logistic regression model, we are able to see that the intercept was 1.01, and the slope of the sex predictor was -3.07. The metrics that were calculated were accuracy, which was 0.78, sensitivity, which was 0.70, and specificity, which was 0.86. Using these metrics, we are able to see that the accuracy of this model using the C++ program was relatively high. Furthermore, we can see that the model was better at predicting negative values in comparison to positive values. This means that the model predicted more negatives than positives, and was better at predicting the negative values.

From the Naive Bayes algorithm that I wrote, we can see the likelihood values for p(sex|survived), and p(pclass|survived). The means and variances are from the age values of each passenger. From this data, we can see that the likelihood for a male to survive was 16%, and a male perishing was 84%. For women, these statistics were 68% and 32% respectively. The predictions shown at the bottom were predictions on the first 5 values in the test set. We can see that the first and second people in the test set have a higher chance of perishing in comparison to the latter three.

Also, it's important to note that, for my algorithms, the values were not identical to those used calculated in R. This may be for many reasons, such as rounding errors that occur in C++, as they are not identical to the rounding in R. Another reason for these small differences could be because the test and train sets in the R code were randomly seeded, whereas I used the first 800 values for the C++ Code.

Discriminative vs. Generative Classifiers

In this assignment, I created C++ programs for logistics regression and Naive Bayes models. These two models can be classified as discriminative and generative models respectively. Discriminative and generative classifiers are extremely different, but at their core, they both use probability to predict a new instance in a data set. In regards to discriminative classifiers, they directly estimate the parameters of P(Y|X) [1]. Discriminative classifiers are useful when the data is labeled and complex, as it helps determine where the decision boundary is between classes [2]. Discriminative classifiers generally perform better as the amount of data in a data set grows, and generative classifiers perform better on smaller data sets. Furthermore,

discriminative classifiers work well when there is an outlier in the data, as an outlier is set as a misclassified example [2]. One disadvantage of using discriminative classifiers is that they tend to overfit the data, and they are prone to misclassification [3].

On the other hand, generative classifiers estimates the parameters for P(Y) and P(X|Y) [1]. Generative classifiers focus on the probability distribution and how multiple and different features and a target variable occur at the same time [2]. This essentially means that generative classifiers learn the underlying data distribution in a data set rather than focusing on the decision boundary between different features. Generative classifiers do better on smaller data sets, and also have lower variance in comparison to discriminative models. However, disadvantages of generative models is that they have higher bias in comparison to discriminative models, and they do not perform as well when there is an outlier in the data [2].

Reproducible Research in Machine Learning

In order for researchers and scientists to confirm their findings that they research and experiment on, reproducing is a useful practice. The reproducibility of a study refers to the "ability of a researcher to duplicate the results of a prior study using the same materials as were used by the original investigation" [4]. In regards to Machine Learning, reproducibility is important for researchers in order for them to ensure the accuracy of their reports [5]. It is important that studies are reproducible within ML because it allows the findings to be "believable and informative" [4].

Due to the difficulties that are imposed when applying a standard to reproducibility, there are three tiers in regards to reproducibility in machine learning, the bronze, silver, and gold standards [5]. In the bronze standard, reproducibility can be implemented through the authors making the data and analysis public. This is the lowest standard of reproducibility in ML [5]. In the silver standard, the bronze standard must be met, and the dependencies of the analysis should be accessible, and the random components in the analysis are set to be deterministic [5]. Lastly, in the gold standard, the silver and bronze standards must be met, but also the analysis is met with a single command, meaning that the reproducibility of the study is automated [5].

There are going to be challenges when making a study reproducible in ML. Some of these challenges are the lack of records, changes in data, and inconsistencies in hyperparameters [6]. One way to fix the challenge of 'lack of records', is through complete tracking and logging

of the research done in a study. This helps with reproducibility because it allows others to see the work done during the research itself. Furthermore, we can improve the problem of changes in data by keeping a metadata repository for the data used in the study [6]. Artifact stores also help with the problem of hyperparameter inconsistencies, as they store every checkpoint in the ML model, which allows it to be replicated easier [6].

References

[1] K. Mazidi, Machine Learning Handbook, 2nd ed. Creative Commons License.

[2] S. Yıldırım, "Generative vs Discriminative classifiers in machine learning," *Medium*, 14-Nov-2020. [Online]. Available: https://towardsdatascience.com/generative-vs-discriminative-classifiers-in-machine-learning-9ee265be859e. [Accessed: 01-Mar-2023].

[3] Dr. Mazidi's Powerpoints

[4] Z. Ding, "Reproducibility," *Machine Learning Blog | ML@CMU | Carnegie Mellon University*, 24-Aug-2020. [Online]. Available: https://blog.ml.cmu.edu/2020/08/31/5-reproducibility/. [Accessed: 04-Mar-2023].

[5] B. J. Heil, M. M. Hoffman, F. Markowetz, S.-I. Lee, C. S. Greene, and S. C. Hicks, "Reproducibility standards for machine learning in the Life Sciences," *Nature News*, 30-Aug-2021. [Online]. Available: https://www.nature.com/articles/s41592-021-01256-7. [Accessed: 04-Mar-2023].

[6] E. Onose, "How to solve reproducibility in ML," *neptune.ai*, 26-Jan-2023. [Online]. Available: https://neptune.ai/blog/how-to-solve-reproducibility-in-ml. [Accessed: 04-Mar-2023].