CS 445/545: Machine Learning, Winter 2020

Programming Assignment #2, A. Rhodes

Note: This assignment is **due by Monday, 2/24 at 10pm**; you will turn in the assignment by email to our grader, as instructed below.

Each student will turn in an individual assignment (so that we have something upon which to base your individual grade). However, you are encouraged to discuss and work through these problems with your instructor, TA and above all, other students in our class. Having said this, the answers and work you submit will authentically be your own.

In this homework you will use Gaussian Naïve Bayes and Logistic Regression to classify the Spambase data from the **UCI ML repository**, which can be found here: https://archive.ics.uci.edu/ml/datasets/spambase

Part I: Classification with Naïve Bayes

1. **Create training and test set:**
Split the data into a training and test set. Each of these should have about 2,300 instances, and each should have about 40% spam, 60% not-spam, to reflect the statistics of the full data set.   Since you are assuming each feature is independent of all others, here it is not necessary to standardize the features.

2. **Create probabilistic model.   (Write your own code to do this.)**
    •   Compute the prior probability for each class, 1 (spam) and 0 (not-spam) in the training data. As described in part 1, $P(1)$ should be about 0.4.

    •   For each of the 57 features, compute the mean and standard deviation in the training set of the values given each class.   If any of the features has zero standard deviation, assign it a "minimal" standard deviation (e.g., 0.0001) to avoid a divide-by-zero error in Gaussian Naïve Bayes.

3. **Run Naïve Bayes on the test data. (Write your own code to do this.)**

    •   Use the Gaussian Naïve Bayes algorithm to classify the instances in your test set, using

$$P\left(x_i \mid c_j\right) = N\left(x_i; \mu_{i,c_j}, \sigma_{i,c_j}\right), \text{ where } N\left(x; \mu, \sigma\right) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\left(\frac{(x-\mu)^2}{2\sigma^2}\right)}$$

Because a product of 58 probabilities will be very small, we will instead use the log of the product. Recall that the classification method is:

$$class_{NB}(\mathbf{x}) = \underset{class}{\mathrm{argmax}} \left[P(class) \prod_i P\left(x_i \mid class\right) \right]$$

Since $\underset{z}{\mathrm{argmax}}\, f(z) = \underset{z}{\mathrm{argmax}} \log(z)$

we have:

$$class_{NB}(\mathbf{x}) = \underset{class}{\mathrm{argmax}} \log \left[P(class) \prod_i P(x_i \mid class) \right]$$

$$= \underset{class}{\mathrm{argmax}} \left[\log P(class) + \log P(x_1 \mid class) + \ldots + \log P(x_n \mid class) \right]$$

In your report, include a short description of what you did, and your results: the accuracy, precision, and recall on the test set, as well as a confusion matrix for the test set. Write a few sentences describing your results, and answer these questions: Do you think the attributes here are independent, as assumed by Naïve Bayes? Does Naïve Bayes do well on this problem in spite of the independence assumption? Speculate on other reasons Naïve Bayes might do well or poorly on this problem.

Here is what you need to turn in:
*   Your report.
*   Your well-commented code.

How to turn it in (read carefully!):
*   Send these items in electronic format to ark2@pdx.edu (our grader) by 10pm on the due date. No hard copy please!
*   The report should be in pdf format and the code should be in plain-text format.
*   Put "MACHINE LEARNING PROGRAMMING #2" in the subject line.

If there are any questions, don't hesitate to ask me or the grader.