CS 335 Homework 3

Last Updated: February 15, 2019

Instructions

Due Friday 2/22 at 11:55pm. Complete all exercises and problems below.

How to submit

You will edit the following files included with the homework in the directory `hw3-files`:

- `logistic_regression.ipynb`
- `logistic_regression.py`
- `sms_classify.py`

The directory includes these additional files with data, code, or documentation that you will not need to edit:

- `util.py`
- `book-data.csv`
- `sms.txt`
- `sms_readme.txt`
- `pprint_hw3.py`

Complete the requested code in these files, then follow these steps:

1. **Upload written solutions to Gradescope**. Create a single high-quality pdf with your solutions to the exercises and extra credit, if applicable. The solutions can be typed or written and scanned but the resulting pdf *must* be high quality and easily readable. Upload the pdf to Gradescope.
2. **Upload code listing to Gradescope**. Make sure you set your path to include the anaconda binary directory. On the lab computers, this is the correct command:

   ```
   $ export PATH=/anaconda/bin:$PATH
   ```

   When you are done editing and ready to submit your code listing, run the `pprint_hw3.py` script from the `hw3-files` directory:

   ```
   $ python pprint_hw3.py
   ```

   This will create a new file called `hw3-code.pdf` with a listing of all of your code and results. Open the pdf to make sure it is correct and includes all of your code and plots. You can run this multiple times if you update your code. Upload this to Gradescope.
3. **Submit a single zip file containing source code to Moodle.** Make sure your code is complete and files are included in the directory. Also include any auxiliary data or code files you created that are needed to run your code.

   Rename your code directory from `hw3-files` to `hw3-<your last name>` and zip it:

```
$ mv hw3-files hw3-sheldon
$ zip -r hw3-sheldon.zip hw3-sheldon
```

Submit the single zip file to Moodle.

Problems

Let $g(z) = \dfrac{1}{1+e^{-z}}$ be the logistic function.

Problem 1 (5 points). Show that $\frac{d}{dz}g(z) = g(z)(1-g(z))$.

Problem 2 (5 points). Show that $1 - g(z) = g(-z)$.

Problem 3 (5 points). Consider the log loss function for logistic regression simplified so there is only one training example:

$$J(\boldsymbol{\theta}) = -y \log h_{\boldsymbol{\theta}}(\mathbf{x}) - (1-y)\log(1-h_{\boldsymbol{\theta}}(\mathbf{x})), \qquad h_{\boldsymbol{\theta}}(\mathbf{x}) = g(\boldsymbol{\theta}^T\mathbf{x}) = \frac{1}{1+e^{-\boldsymbol{\theta}^T\mathbf{x}}}$$

Show that the partial derivative with respect to θ_j is:

$$\frac{\partial}{\partial\theta_j}J(\boldsymbol{\theta}) = (h_{\boldsymbol{\theta}}(\mathbf{x}) - y)x_j$$

.

Problem 4 (10 points). Logistic regression for book classification. In this problem, you will implement logistic regression for book classification. Open the jupyter notebook `logistic_regression.ipynb` and follow the instructions to complete the problem.

Problem 5 (10 points). SMS spam classification. In this problem you will use your implementation of logistic regression to create a spam classifier for SMS messages. Open the jupyter notebook `sms_classify.ipynb` and follow the instructions to complete the problem.

Problem 6 (5 points extra credit). Use your own data—either SMS or email—to create a personalized spam classifier. You can either put it in the same format as `sms.txt` and use code in `sms_classify.ipynb` to build the dictionary and features, or you can start with your own format and figure out how to use `sklearn.feature_extraction.text.CountVectorizer` to process data in your particular format.