Assignment 05
Logistic Regression
EECE-6822 Machine Learning

Cris Ababei
Electrical and Computer Engr., Marquette University

1. Objective
The objective of this activity include: (1) to run several code examples that present logistic regression, and (2) modify some of the provided code to investigate logistic regression and regularization on the Pima Indians Diabetes dataset.

2. Prerequisite Readings
Murphy
- Logistic regreasion: Murphy 10-10.2.4, 10.3-10.3.3

Abu-Mostafa
- Logistic regression: 3.3

Geron
- Ch.4: Last part on Logistic regression.

Raschka
- Ch.3: ML classifiers using SciKit-Learn

3. Code Examples

Example 1: Logistic Regression - SciKit-Learn

Open your google colab and upload the notebook provided for this lecture (first read ch.4 from Aurelian Geron's book). It showcases logistic regression on the Iris flowers dataset.
logreg_sklearn.ipynb

NOTES:
--(1)This example is essentially provided by Kevin P. Murphy (our main textbook on theory) on github, for ch.10:
https://github.com/probml/pyprobml/tree/master/notebooks/book1/10
If you scroll down you will find the link to the Jupiter notebook:
https://colab.research.google.com/github/probml/pyprobml/blob/master/notebooks/book1/10/logreg_sklearn.ipynb
--(2)The example itself was developed from Aurelian Geron's book.
[*B3-Geron] Aurelien Geron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, O'Reilly, 2022. The original code can be found at (see last part on logistic regression):
https://github.com/ageron/handson-ml3/blob/main/04_training_linear_models.ipynb

Example 2: ML Classifiers - SciKit-Learn

This code example is from ch.3 from (works also with the Iris dataset!):
[*B3-Raschka] Sebastian Raschka, Yuxi Liu, and Vahid Mirjalili, Learning with PyTorch and Scikit-Learn: Develop machine learning and deep learning models with Python, Packt Publishing, 2022.
Again, the source code of this book is in this GitHub repository:
https://github.com/rasbt/machine-learning-book

Follow the details presented in Assignment #2 on how to run code from Raschka book. We will look at the notebook:
https://github.com/rasbt/machine-learning-book/blob/main/ch03/ch03.ipynb
NOTES:
--(1)In my case, I found the easiest to actually run directly the Python code corresponding to this notebook in Anaconda Spyder:
EECE6822\Book_Code_Raschka\machine-learning-book-main\ch03\ch03.py
--(2)This example does not cover multinomial/softmax regression.

At this time, you should run the notebook or run directly the Python code from the beginning of it until the Section titled "**Maximum margin classification with support vector machines**". You should also first read the chapter from the book itself, before running the code!
Essentially, this example will show you how to: (1) train a perceptron model (and apply to 2 features and 3 classes), (2) implement logistic regression by changing Adaline code from Ch.2, and (3) use SciKit-Learn's classes for logistic regression!

Example 3: Logistic Regression - mlm tutorials and code examples

In this part, we will look at several tutorials from **machinelearningmastery (mlm)**. Read the following tutorials and run the code where applicable:

a--Logistic Regression Tutorial for Machine Learning
https://machinelearningmastery.com/logistic-regression-tutorial-for-machine-learning/

b--A Gentle Introduction to Logistic Regression With Maximum Likelihood Estimation
https://machinelearningmastery.com/logistic-regression-with-maximum-likelihood-estimation/

c--How To Implement Logistic Regression From Scratch in Python ← logistic regression algorithm on the Pima Indians Diabetes dataset
https://machinelearningmastery.com/implement-logistic-regression-stochastic-gradient-descent-scratch-python/

d--Cost-Sensitive Logistic Regression for Imbalanced Classification
https://machinelearningmastery.com/cost-sensitive-logistic-regression/

e--Multinomial Logistic Regression With Python ← synthetic dataset with 10 columns and 3 classes
https://machinelearningmastery.com/multinomial-logistic-regression-with-python/

f--Making Predictions with Logistic Regression in PyTorch
https://machinelearningmastery.com/making-predictions-with-logistic-regression-in-pytorch/

4. Assignment

The objective is to investigate logistic regression for Pima Indians Diabetes dataset. A second objective is to investigate how regularization strength parameter "C" impacts logistic regression result.

--(a)Create a new Jupyter notebook (or - recommended - work directly as a Python program) that starts with the code that imports the Pima Indians Diabetes dataset and keeps only the two most important features out of the ten input feature columns.
You could re-use code to load the dataset from this example tutorial (from Example 3 above):
https://machinelearningmastery.com/implement-logistic-regression-stochastic-gradient-descent-scratch-python/
You will again split the dataset into 80/20 train/test portions. Also, do not forget to normalize all columns with input features!

NOTE: Diabetes dataset was also discussed (in even more details) in the tutorial example that we did in Assignment 1. You can re-use code for loading the dataset from that tutorial as well: https://machinelearningmastery.com/tutorial-first-neural-network-python-keras/

--(b)Then, add code to identify and keep only the two most important features. Detection of the two most important features should be done similarly to how you did it in the previous two assignments (but for a different dataset). Basically, that was based on the first part of the example linear_regression_demo.ipynb from Lecture 2.

--(c)Then, add in the code from Section "Training a logistic regression model with scikit-learn" from Example 3 above. It is the one that makes use of LogisticRegression class. Create a plot similar to the one in Figure 3.7 from Raschka book where this code Example 3 is taken from. The figure will show two classes only though (not 3), and, they are Yes-diabetes (1) and No-diabetes (0).

--(d)In the second part of this assignment, you must use the code from section "Tackling overfitting via regularization" from the same code Example 3 – to study how performance of the logistic regression changes when the strength parameter "C" varies. You must generate a plot similar to the one in Figure 3.9 from Raschka book. You must also plot prediction accuracy on the test portion of the dataset vs. parameter "C", for each value of "C" investigated in the code from the above section, inside the for loop:

```
for c in np.arange(-5,5)
```

--(e)Present a discussion of your findings.

5. Deliverables

You must write (typed) a report and upload it as a PDF file on D2L. The report should be named "**hw5_report_LastName.pdf**". You should also create a .zip archive with all your code and implementations of all parts of the assignment. Upload also this archive .zip file with the name "**hw5_implementation_code_LastName.zip**" to D2L. Hence, your D2L should contain two items: the report and the .zip file. **Do not include the report inside the .zip and upload only the .zip. They should be two separate items!**

The report should include the following sections and subsections. Make sure section titles are in bold font and pages are numbered.

1) **Title + course info + your name**
2) **Summary.** Describe in one paragraph what the objective of the assignment is.
3) **Logistic regression.** Write a short paragraph to describe what logistic regression is.
4) **Description of Experiments and Discussion.** Describe the experiments you did. All tables and figures should be numbered and should have captions. All plots in all figures should have axes labels and titles. Present a meaningful discussion with the interpretation of the results you obtained. Explain if you expected the results or not; discuss the intuition behind it.
5) **Conclusion.** Present your conclusions; highlight what are your main takeaways that you learned from this assignment. Describe what issues you encountered and how you solved them.
6) **References.** Include all references that you used, as a numbered list. Cite them in the report itself; do not just list them! If your report has References that are not cited in the report, points will be deducted!
7) If your report has References that are not cited in the report, points will be deducted!