Intro to Big Data Science: Project 1

Due Date: April 23, 2019

✏ **Problem**(Maximum likelihood approach for logistic regression)

Consider the logistic regression for two-class problemp: Assuming that ($\mathbf{w} = (w_0, w_1, \ldots, w_d)^T$, and $\mathbf{x}$ includes the constant 1 in its first component)

$$Pr(y=1|\mathbf{X}=\mathbf{x}) = \frac{\exp(\mathbf{w}^T\mathbf{x})}{1+\exp(\mathbf{w}^T\mathbf{x})},$$
$$Pr(y=0|\mathbf{X}=\mathbf{x}) = \frac{1}{1+\exp(\mathbf{w}^T\mathbf{x})}.$$

By a logit transformation, $\log[p/(1-p)]$, we recover a linear regression model:

$$\log\frac{Pr(y=1|\mathbf{X}=\mathbf{x})}{Pr(y=0|\mathbf{X}=\mathbf{x})} = \mathbf{w}^T\mathbf{x}.$$

The decision boundary is the set of points for which the above quantity is zero, and this is a hyperplane defined by $\{\mathbf{w}^T\mathbf{x}=0\}$.

Remember that we have used the maximum likelihood approach to interpret the linear regression. Now we apply the same method to two-class logistic regression. Denote the probability $Pr(y=k|\mathbf{X}=\mathbf{x}) = p_k(\mathbf{x};\mathbf{w})$, $k=0$ or 1. The likelihood function is defined by

$$L(\mathbf{w}) = \prod_{i=1}^{n} p_{y_i}(\mathbf{x}_i;\mathbf{w})$$

where y_i is the label of the i-th sample $\mathbf{x}_i$. The log-likelihood is then

$$l(\mathbf{w}) = \sum_{i=1}^{n} \log p_{y_i}(\mathbf{x}_i;\mathbf{w})$$

1. Let $p_1(x;\mathbf{w}) = p(x;\mathbf{w})$ and $p_0(x;\mathbf{w}) = 1 - p(x;\mathbf{w})$. Show that the log-likelihood function can be reformulated as

$$l(\mathbf{w}) = \sum_{i=1}^{n} \left\{ y_i \mathbf{w}^T \mathbf{x}_i - \log(1 + e^{\mathbf{w}^T \mathbf{x}_i}) \right\}$$

2. Show that the maxima of the log-likelihood function must satisfy the score equations

$$\sum_{i=1}^{n} x_{ij}(y_i - p(\mathbf{x}_i;\mathbf{w})) = 0, \qquad j = 0, 1, \ldots, d. \tag{1}$$

   (Remark: In particular, the first score equation $j = 0$ gives the identity $\sum_{i=1}^{n} y_i = \sum_{i=1}^{n} p(\mathbf{x}_i;\mathbf{w})$, the expected number of class ones matches the observed number.)

3. Show that the Hessian matrix of $l(\mathbf{w})$ is

$$\frac{\partial^2 l(\mathbf{w})}{\partial \mathbf{w} \partial \mathbf{w}^T} = -\sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^T p(\mathbf{x}_i;\mathbf{w})(1 - p(\mathbf{x}_i;\mathbf{w}))$$

4. The famous Newton-Raphson method can be applied to solve the score equations (1):

   a) Initialize $\mathbf{w} = \mathbf{w}^{(0)}$;

   b) At the k-th step, $\mathbf{w}^{(k)} = \mathbf{w}^{(k-1)} - \left(\frac{\partial^2 l(\mathbf{w})}{\partial \mathbf{w} \partial \mathbf{w}^T} \right)^{-1} \Big|_{\mathbf{w}^{(k-1)}} \frac{\partial l(\mathbf{w})}{\partial \mathbf{w}} \Big|_{\mathbf{w}^{(k-1)}}$, then increase k to $k+1$;

   c) Once $|\mathbf{w}^{(k)} - \mathbf{w}^{(k-1)}| < \epsilon$, stop; otherwise, go back to b).

   Now we use matrix notations: $\mathbf{X}$ whose rows are $\mathbf{x}_i$'s, $\mathbf{y}$ is the column vector of y_i's, $\mathbf{p}$ is the column vector of $p(\mathbf{x}_i;\mathbf{w}^{(k-1)})$'s, $\mathbf{D}$ is $n \times n$ diagonal matrix with diagonal entries $p(\mathbf{x}_i;\mathbf{w}^{(k-1)})(1 - p(\mathbf{x}_i;\mathbf{w}^{(k-1)}))$. Show that under the new notations, step b) in Newton-Raphson method can be rewritten as

$$\mathbf{w}^{(k)} = \arg\min_{\mathbf{w}} (\mathbf{z} - \mathbf{X}\mathbf{w})^T \mathbf{D} (\mathbf{z} - \mathbf{X}\mathbf{w})$$

   where $\mathbf{z} = \mathbf{X}\mathbf{w}^{(\mathbf{k-1})} + \mathbf{D}^{-1}(\mathbf{y} - \mathbf{p})$. This is the so-called iteratively reweighted least square (IRLS) algorithm.

5. Write a (Python) computer program to implement the IRLS algorithm. The initial value $\mathbf{w}^{(0)}$ can be chosen in your convenience (e.g., you can choose $\mathbf{w}^{(0)} = \mathbf{0}$).

6. Apply the program you develop in the previous step to play with the South African Heart Disease data: https://sci2s.ugr.es/keel/dataset.php?cod=184. We will use the copy of the data set already partitioned by means of 5-folds cross validation. This set of data, named "saheart-5-fold", is also provided in the package of this project. Note that there are five groups of data, each of which consists of a training set (e.g. "saheart-5-1tra.dat") and a test set (e.g. "saheart-5-1tst.dat"). Train the

model by the training set in each group and test the model by the test set in the same group. Evaluate your results in terms of accuracy. (Hint: you have to do one-hot encoding for the non-numeric attributes)

7. Compare the results obtained with your own IRLS algorithm with the results computed using Python function "LogisticRegression" in the module "sklearn.linear_model". The comparison should be made based on the indices such as confusion matrix, accuracy, precision and recall.
8. (Optional) Can you improve your results? How?

You should submit your project in the form of jupyter-notebook files. The derivation parts 1-4 should also be included in jupyter-notebook file (in the form of markdown). You can package all your files and submit it online in the system.