1. Dataset

Dataset is *Breast Cancer Wisconsin* from UCI machine learning repository.

Here is the attribute information of dataset: (class attribute has been moved to last column)

# Attribute	Domain
1. Sample code number	id number
2. Clump Thickness	1 - 10
3. Uniformity of Cell Size	1 - 10
4. Uniformity of Cell Shape	1 - 10
5. Marginal Adhesion	1 - 10
6. Single Epithelial Cell Size	1 - 10
7. Bare Nuclei	1 - 10
8. Bland Chromatin	1 - 10
9. Normal Nucleoli	1 - 10
10. Mitoses	1 - 10
11. Class:	(2 for benign, 4 for malignant)

For this dataset, I firstly dropped first attribute "ID number". I used next 9 attributes as the features, and last attribute as the label for each sample.

There are 699 samples, I randomly picked up 80% (560) as training data, and remaining 20% (139) as test data. Plus, there are several missing data which are represented by '?'. I replaced '?' With 0.

Besides, I reassigned label 2, 4 to 0, 1. (0 is for benign, and 1 for malignant.)

2. Logistic Regression with Newton 's method

1) I firstly create a function to calculate possibility function: $p(x_i) = \dfrac{e^{-(w^T x_i + b)}}{1 + e^{-(w^T x_i + b)}}$, then I can know the log likelihood $L(W|X) = \sum_{i=1}^{N} y_i log(p(x_i)) + (1 - y_i) log(1 - (p(x_i))$, replace $p(x_i)$ with possibility function:

$L(W|X) = -\sum_{i=1}^{N} log(1 + e^{w^T x_i + b}) + y_i(w^T x_i + b)$. Plus, I also add a L2 regularization for log likelihood $\frac{\alpha}{2} \sum_{i=1}^{K} ||w_i||^2$.

2) In order to maximize log likelihood, the first deviation should be calculated. The first deviation is $\sum_{i=1}^{N} (y_i - p(x_i))x_i + \sum_{i=1}^{N} \alpha w_i$. So I create a function to calculate this first deviation.

3) But for now, there is no closed-form solution exists. So Newton's Method as an iterative

algorithms is used. Second deviation is needed, the formula is $-\sum_{i=1}^{N}(p(x_i))(1-p(x_i)) \times x_{ij}x_{ik} + \alpha * A$ (A is a diagonal matrix). So I create a function to calculate this second deviation.

4) Based on first deviation and second deviation, the iteration in Newton's Method can be calculate: $\beta^0 - \frac{f'(\beta^0)}{f''(\beta^0)}$. So I create a function to calculate each iteration update for parameters of Newton's Method.

5) Keeping this iteration until the value of estimated parameters stop changing. At same time, I give two parameter "max iterations" and "min increment" to control this iteration. The "max iterations" is the max number of iterations, and "min increment" is the min value of estimated parameters changes for each iteration. It will also stop when any of these two parameters is reached. For the "max iterations" value and "min increment" value, I both used the default value of Sklearn, which are 100 and 10^{-4}.

6) Repeat these steps 10 times for randomly 80% training data and 20% test data, and get the average accuracy over 10 trials.

3. Result

For comparison, I also use sklearn's LogisticRegression with same data 10 times, and get the average accuracy.

For these two result, they are almost same.

For sklearn's LogisticRegression, when the solver is "newton-cg", the default penalty is l2. I also applied l2 penalty. But because of penalty coefficient are different, sometimes they have different results, but very similar.