sk_logisticRegression

April 15, 2019

1 Logistic Regression

Logistic regression fits a logistic model to data and makes predictions about the probability of an event (between 0 and 1).

```
In [1]: from sklearn import datasets
        from sklearn import metrics
        from sklearn.linear_model import LogisticRegression
```

2 Iris flowers Dataset

```
In [2]: dataset = datasets.load_iris()
        dataset.feature_names
```

```
Out[2]: ['sepal length (cm)',
         'sepal width (cm)',
         'petal length (cm)',
         'petal width (cm)']
```

3 MODEL

```
In [3]: # creating a model
        model = LogisticRegression(random_state=0, solver='lbfgs',multi_class='auto', max_iter=
        model.fit(dataset.data, dataset.target)
```

```
Out[3]: LogisticRegression(C=1.0, class_weight=None, dual=False, fit_intercept=True,
                   intercept_scaling=1, max_iter=1000, multi_class='auto',
                   n_jobs=None, penalty='l2', random_state=0, solver='lbfgs',
                   tol=0.0001, verbose=0, warm_start=False)
```

4 lbfgs solver

Limited-memory BFGS is an optimization algorithm in the family of quasi-Newton methods that approximates the Broyden–Fletcher–Goldfarb–Shanno algorithm using a limited amount of computer memory. It is a popular algorithm for parameter estimation in machine learning.

5 Prediction/Classification

```
In [4]: expected = dataset.target
        predicted = model.predict(dataset.data)
        print(metrics.classification_report(expected, predicted))
        print(metrics.confusion_matrix(expected, predicted))
```

	precision	recall	f1-score	support
0	1.00	1.00	1.00	50
1	0.98	0.94	0.96	50
2	0.94	0.98	0.96	50
micro avg	0.97	0.97	0.97	150
macro avg	0.97	0.97	0.97	150
weighted avg	0.97	0.97	0.97	150

```
[[50  0  0]
 [ 0 47  3]
 [ 0  1 49]]
```

6 Classification Report variables

6.1 Precision

The precision is intuitively the ability of the classifier not to label as positive a sample that is negative. ### Precision = truePositive/(truePositive+falsePositive)

6.2 Recall

The recall is intuitively the ability of the classifier to find all the positive samples. ### Recall = truePositive / (truePositive + falseNegative)

6.3 F1-score

The F1 score can be interpreted as a weighted average of the precision and recall, where an F1 score reaches its best value at 1 and worst score at 0. The relative contribution of precision and recall to the F1 score are equal. ### F1 = 2 * (precision * recall) / (precision + recall)

6.4 Summary

The f1-score gives you the harmonic mean of precision and recall. The scores corresponding to every class will tell you the accuracy of the classifier in classifying the data points in that particular class compared to all other classes.The support is the number of samples of the true response that lie in that class.

6.5 References

1. https://machinelearningmastery.com/get-your-hands-dirty-with-scikit-learn-now/

2. https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html#sklearn

3. https://scikit-learn.org/stable/modules/generated/sklearn.metrics.precision_recall_fscore_support.html

4. https://scikit-learn.org/stable/modules/generated/sklearn.metrics.f1_score.html

5. https://en.wikipedia.org/wiki/Limited-memory_BFGS