Take note of the following in the preceding code block:

- The logistic regression model is initialized by calling the method LogisticRegression(solver='lbfgs', multi_class='ovr'). The attribute 'multi_class' is set to 'ovr' to create a one-vs.-rest classifier.
- The confusion matrix for a multi-class learning problem uses the `multilabel_confusion_matrix' to calculate classwise confusion matrices where the labels are binned in a one-vs.-rest manner. As an example, the first matrix is interpreted as the difference between the actual and predicted targets for class 1 against other classes.

Optimizing the Logistic Regression Model

This section surveys a few techniques to consider in optimizing/improving the performance of logistic regression models.

In the case of Bias (i.e., when the accuracy is poor with training data)

- Remove highly correlated features. Logistic regression is susceptible to degraded performance when highly correlated features are present in the dataset.
- Logistic regression will benefit from standardizing the predictors by applying feature scaling.
- Good feature engineering to remove redundant features or recombine features based on intuition into the learning problem can improve the classification model.
- Applying log transforms to normalize the dataset can boost logistic regression classification accuracy.

In the case of variance (i.e., when the accuracy is good with training data, but poor on test data)

Applying regularization (more on this in Chapter 21) is a good technique to prevent overfitting.

This chapter provides a brief overview of logistic regression for building classification models. The chapter includes practical steps for implementing a logistic regression classifier with Scikit-learn. In the next chapter, we will examine the concept of applying regularization to linear models to mitigate the problem of overfitting.

CHAPTER 21

Regularization for Linear Models

Regularization is the technique of adding a parameter, λ, to the loss function of a learning algorithm to improve its ability to generalize to new examples by reducing overfitting. The role of the extra regularization parameter is to shrink or to minimize the measure of the weights (or parameters) of the other features in the model.

Regularization is applied to linear models such as polynomial linear regression and logistic regression which are susceptible to overfitting when high-order polynomial features are added to the set of features.

How Does Regularization Work

During model building, the regularization parameter λ is calibrated to determine how much the magnitude of other features in the model is adjusted when training the model. The higher the value of the regularization, the more the magnitude of the feature weights is reduced.

If the regularization parameter is set too close to zero, it reduces the regularization effect on the feature weights of the model. At zero, the penalty the regularization term imposes is virtually non-existent, and the model is as if the regularization term was never present.

Effects of Regularization on Bias vs. Variance

The higher the value of λ (i.e., the regularization parameter), the more restricted the coefficients (or weights) of the cost function. Hence, if the value of λ is high, the model can result in a learning bias (i.e., it underfits the dataset).

E. Bisong, *Building Machine Learning and Deep Learning Models on Google Cloud Platform*,
https://doi.org/10.1007/978-1-4842-4470-8_21