109550198 卜銳凱

Data Mining
HW3
Report

1. Explain your implementation which get the best performance in detail.

   In this code, the KNN approach yielded the best performance in outlier detection. It kicked off by loading the training and test datasets from files in CSV format. Then, it separated the features and the target variable, subsequently scaling the features with StandardScaler to have a mean of 0 and a standard deviation of 1. This is useful because KNN is sensitive to the scale of the data. The training data is then split using an 80-20 split into train and validation sets such that model evaluation is done on unseen data during training.

   The outlier detection ability of the model is evaluated by marking a fraction of the validation data as outliers. This is done by dividing the validation data into normal and outlier subsets and then combining them. A KNN model is instantiated using n_neighbors=1 and is trained on data from the training split. The model is evaluated based on its performance in computing the distances to the nearest neighbors in the validation set and using those distances to calculate the AUC score.

   The model would then be retrained on the full scale of training data, and test data would be used to make the prediction by computing distances to the nearest neighbors. These distances would be normalized to probabilities taken between 0 and 1, and the resulting file would be saved in a CSV for submission.

   Here, the KNN model performs well, as it is effective in detecting outliers using local density estimation, is very flexible for various kinds of data since it does not assume the underlying distribution of the data and is sensitive to deviations from the normal data with n_neighbors=1. These aspects, taken together, explain a high performance for the KNN model in outlier identification, as displayed in the AUC.

2. Explain the rationale for using auc score instead of F1 score for binary classification in this homework.

   The reason for opting for the AUC score over the F1 score for binary classification in this homework is the reason that directly relates to the task in the question: outlier detection. The AUC score is particularly advantageous when employed to evaluate outlier detection models since it is a threshold-independent measure of how the model differentiates between classes. This threshold-independent measure rigorously evaluates the overall discriminatory power of the model—an optimal measure for tasks such as outlier detection, in which we are interested in identifying rare events (outliers) among most normal events. An F1 score, on the other hand, is a trade-off between precision and recall

based on a fixed threshold, which is usually set at 0.5. It is very useful to combine precision and recall into a single measure, but depending on a single threshold is a limitation—especially when the optimal threshold value is not transparent or changes from context to context. Additionally, the F1 score gives an equal emphasis on the balance between precision (the ability to produce positive predictions accurately) and recall (the ability of the classifier to detect all positive instances). This is not ideal for outlier detection since the model needs to capture the ability to do proper instance ranking. More so, in the case of outlier detection, we may have cases where there are imbalanced datasets, considering the low prevalence of outlier cases compared to the normal number of data points. The AUC score, on the other hand, is less influenced by the imbalance of classes, providing a more reliable performance measure, despite this class imbalance. More generally, since the AUC evaluates model performance across all thresholds and is less sensitive to class imbalance, it is a more appropriate and informative metric when evaluating the performance of model is outlier detection, hence its application to this homework.

3. Discuss the difference between semi-supervised learning and unsupervised learning.

   Basically, what differs these from unsupervised learning is the way they treat labeled and unlabeled data. Unsupervised learning describes modeling strategies where the goal is to describe the hidden structure or explanatory factors of the data without prior information about the relationship between input features and target responses. It's associated with the tasks that can be performed on data without any supervision. Most common of these are clustering, dimension reduction, and association rule learning. These have found applications in diverse sectors, for tasks like customer segmentation, anomaly detection, or compression. The biggest advantage is, of course, that unsupervised learning doesn't need labeled data, which is hardest and most expensive to obtain. On the other hand, interpretation of the uncovered patterns is usually far from trivial, and there is no accuracy to be measured since there are no available labels.

   Semi-supervised learning, on the other hand, is intermediate between unsupervised and supervised learning, in which a few labeled examples are available for training to classify or cluster data. In general lines of semi-supervised learning, usually a small subset of data is labeled because it is more costly to label data instances. Self-training, co-training, and graph-based methods are the common approaches in semi-supervised learning, which utilize the information from unlabeled data to enhance the performance of the models. Some of the common applications of semi-supervised learning span text classification, image classification, speech recognition, and bioinformatics. The main advantage of semi-supervised learning is that it provides improvement in model performance with a reduced number of labeled examples, thereby reducing the cost and effort in labeling more data. However, it remains sensitive to the quality of the unlabeled instances, and these methods are often quite complex to implement and tune.

   In summary, while unsupervised learning focuses on finding hidden patterns in completely unlabeled data, semi-supervised learning aims at using both labeled and

unlabeled data to improve learning performance, often outperforming purely unsupervised methods, particularly when labeled data is scarce.