**Technical Report – Assignment 1**

The goal of this report is to analyze and identify bias in machine learning projects. This experiment mainly analyzes from two perspectives: dataset bias and bias in the training process. By analyzing the data distribution and the model training process, we try to find the cause of model bias and propose corresponding solutions to ensure the fairness and accuracy of the model.

1. **Introduction**

With the widespread application of machine learning (ML) technology in various fields, fairness and bias issues have gradually become the focus of researchers and developers. Bias may exist in machine learning models in many forms, often leading to unfair decisions or predictions, especially when dealing with imbalanced classes or heterogeneous datasets. This bias not only affects the performance of the model, but may also have a serious negative impact on the actual application of the model.

In this project, our task is to identify bias in the model and analyze it from two main aspects:

**Dataset bias**: Analyze whether there are structural problems in the dataset itself that can cause model bias, such as class imbalance, uneven feature distribution, etc.

**Bias in the training process**: Analyze the bias exhibited by the model during training due to the sensitivity of the data or algorithm itself and explore how different correction methods affect model performance.

Through experiments, we hope to reveal the bias in the dataset and model training process and propose corresponding solutions to help develop more fair and robust models. In this experiment, we chose Random Forest as the baseline model and experimented with several different techniques, including oversampling, undersampling, class weight adjustment, and grid search tuning, to test the model's performance when dealing with data imbalance.

This report will detail the experimental methods, results, and the process of identifying and correcting bias in subsequent chapters.

1. **Body**
   1. **Dataset Bias Analysis**
      1. Class Imbalance

In the dataset, the ‘false’ class labels account for the majority, while the ‘true’ class labels are relatively rare. This class imbalance has a significant impact on the model’s predictive ability. Without the use of resampling techniques, the model has a high accuracy, but performs poorly in identifying the minority class (i.e., the ‘true’ class) with a low recall rate. This indicates that the model tends to predict the majority class label, resulting in a bias against the minority class.

* + 1. Feature Distribution

By analyzing the distribution of dataset features, we found that some features are significantly more densely distributed in the majority class (‘false’ class). This phenomenon makes it easier for the model to learn the relationship between these features and the majority class during training, thereby ignoring important features related to the minority class. This further exacerbates the bias of the model, especially when dealing with minority class samples.

* + 1. Impact on Model Performance

The direct result of class imbalance is reflected in the performance indicators of the model. By analyzing the precision, recall, and F1 score, we found that the model has a high precision on the majority class, but exhibits a low recall on the minority class. The following table shows the performance indicators of the model under different methods:

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Precision | Recall | F1 score |
| No balancing | 0.92 | 0.62 | 0.74 |
| Oversampling | 0.79 | 0.87 | 0.83 |
| Undersampling | 0.86 | 0.73 | 0.79 |
| Class weight adjustment | 0.88 | 0.78 | 0.82 |

* 1. **Bias analysis during training**

1. Model sensitivity to class imbalance

During the initial training process, models (such as random forests) show high sensitivity to class imbalance in the dataset. Specifically, when class weights are not adjusted or resampling techniques are used, the model tends to predict the majority class label, resulting in a lower recall rate for the minority class. The bias of the model is reflected in the significant difference in precision and recall.

1. Impact of resampling techniques

To reduce bias, we used the following resampling techniques:

Oversampling: By increasing the number of minority class samples, the recall rate of the minority class is improved, but more false positives are introduced, resulting in a decrease in precision.

Undersampling: By reducing the number of majority class samples, the balance of the model is improved, but it leads to information loss and a decrease in the overall accuracy of the model.

Class weight adjustment: By adjusting the class weights, the performance of the model on the majority and minority classes is more balanced, the difference between precision and recall is reduced, and the bias is effectively alleviated.

1. Optimization effect of grid search

We use grid search to tune the model parameters to further reduce bias. During the optimization process, the model's hyperparameters (such as the number of trees, maximum depth, etc.) are carefully adjusted to achieve the best performance balance. The results of the grid search show that the performance of the tuned model is significantly improved when dealing with minority classes, and the bias is reduced.

* 1. **Performance of the model under bias correction measures**

After applying bias correction techniques such as oversampling and class weight adjustment, the gap between the model's precision and recall is narrowed. However, even with these measures, the model still shows some sensitivity to class imbalance, indicating that the model's bias towards the majority class has not been completely eliminated.

* 1. **Additional details**

The programming language used in the experiment is Python, and common machine learning libraries such as scikit-learn and pandas are mainly used.

The random forest model was chosen because of its robustness in dealing with imbalanced data.

Both data standardization and normalization were tested in the experiment, and the results showed that normalized data can better balance precision and recall.

1. **Conclusion**

Through analysis, this experiment shows that the imbalance of the data set is the main cause of model bias, and the model's sensitivity to imbalanced data during training exacerbates this bias. By using methods such as resampling and grid search, the bias can be reduced to a certain extent, but the model still has a tendency to favor the majority class label. Therefore, in practical applications, we should continue to pay attention to the data distribution and model bias issues, and take appropriate measures to correct them.