**Detailed Report**

**1. Introduction**

Customer churn prediction is a critical task for businesses, especially those operating in subscription-based industries. Accurately identifying customers who are likely to leave is essential for developing retention strategies. However, churn datasets often suffer from class imbalance, where the number of customers who do not churn significantly outweighs those who do. This imbalance can lead to biased models that favor the majority class, thereby reducing the effectiveness of churn prediction.

In this study, we explored the performance of five different classification algorithms—Logistic Regression, K-Nearest Neighbors (KNN), Random Forest, Support Vector Machine (SVM), and XGBoost—on an imbalanced dataset. To address the class imbalance, we applied several class imbalance (CI) techniques, including Random Over-Sampling, Synthetic Minority Over-sampling Technique (SMOTE), and class weight adjustments. The goal was to determine how these techniques impact the performance of each algorithm, particularly in terms of recall and precision for the minority class (churned customers).

**2. Baseline Model Performance**

Before applying any class imbalance techniques, we first evaluated each model on the raw, imbalanced dataset to establish baseline performance metrics.

* **Logistic Regression:**
  + Accuracy: 80.45%
  + AUC-ROC: 0.8348
  + The model showed balanced precision between classes but struggled with recall for the minority class (churned customers), indicating that while it could identify non-churned customers well, it often failed to detect those likely to churn.
* **K-Nearest Neighbors (KNN):**
  + Accuracy: 75.12%
  + AUC-ROC: 0.7668
  + KNN had difficulty with minority class recall, resulting in lower f1-scores for churned customers. This suggests that KNN, in its baseline state, was not well-suited for handling the class imbalance.
* **Random Forest:**
  + Accuracy: 79.03%
  + AUC-ROC: 0.8171
  + Random Forest performed better than KNN but still struggled with minority class recall. The model tended to favor the majority class, leading to a drop in performance metrics for churned customers.
* **Support Vector Machine (SVM):**
  + Accuracy: 78.96%
  + AUC-ROC: 0.7942
  + SVM showed a similar trend to Random Forest, with good overall accuracy but limited effectiveness in detecting churned customers. The f1-scores for the minority class were relatively low.
* **XGBoost:**
  + Accuracy: 76.69%
  + AUC-ROC: 0.8142
  + XGBoost provided a balanced performance but still faced challenges with minority class detection. The model was slightly better at identifying churned customers compared to KNN and SVM but still exhibited a noticeable imbalance in precision and recall.

**3. Impact of Class Imbalance Techniques**

To mitigate the effects of class imbalance, we applied three CI techniques: Random Over-Sampling, SMOTE, and class weight adjustments. These techniques aim to balance the dataset, thereby improving the model’s ability to detect the minority class.

**3.1 Random Over-Sampling**

Random Over-Sampling involves increasing the number of minority class samples by randomly duplicating existing ones. This method directly addresses the class imbalance by making the dataset more balanced.

* **Logistic Regression:**
  + Accuracy: 72.64%
  + AUC-ROC: 0.8346
  + After applying Random Over-Sampling, Logistic Regression showed a significant improvement in recall for the minority class, jumping to 79%. However, this came at the expense of overall accuracy, which dropped by about 8%. The model now had a higher chance of detecting churned customers but with a notable trade-off in precision.
* **K-Nearest Neighbors (KNN):**
  + Accuracy: 68.51%
  + AUC-ROC: 0.7515
  + KNN experienced a similar shift. Recall for the minority class improved to 75%, but accuracy and precision for the majority class declined. This indicated that while KNN became better at identifying churned customers, it also became less reliable overall.
* **Random Forest:**
  + Accuracy: 77.83%
  + AUC-ROC: 0.8161
  + Random Forest benefitted from Random Over-Sampling, with an increase in minority class recall to 59%. The model’s accuracy remained relatively stable, showing that Random Forest could handle the increased sensitivity to the minority class without a significant loss in overall performance.
* **Support Vector Machine (SVM):**
  + Accuracy: 73.35%
  + AUC-ROC: 0.8029
  + SVM showed improved minority class recall (77%) after Random Over-Sampling. However, the increase in recall led to a decrease in overall accuracy and precision, similar to the trends observed with other models.
* **XGBoost:**
  + Accuracy: 74.98%
  + AUC-ROC: 0.8135
  + XGBoost saw moderate gains in minority class recall, with a slight drop in accuracy. The model maintained a balance between precision and recall, indicating that XGBoost could adapt to Random Over-Sampling without drastic performance shifts.

**3.2 SMOTE (Synthetic Minority Over-sampling Technique)**

SMOTE generates synthetic samples for the minority class by interpolating between existing minority class samples. This method not only increases the number of minority samples but also introduces variability, which can help models generalize better.

* **Logistic Regression:**
  + Accuracy: 69.79%
  + AUC-ROC: 0.7620
  + Logistic Regression benefitted from SMOTE, with an improvement in recall to 73%. The accuracy dropped slightly, but the model maintained a better balance between precision and recall compared to Random Over-Sampling.
* **K-Nearest Neighbors (KNN):**
  + Accuracy: 68.51%
  + AUC-ROC: 0.7515
  + KNN showed improvements similar to those seen with Random Over-Sampling, with increased minority class recall but a decline in overall accuracy. SMOTE allowed KNN to better detect churned customers, although at the cost of precision.
* **Random Forest:**
  + Accuracy: 77.04%
  + AUC-ROC: 0.8116
  + Random Forest’s performance improved with SMOTE, showing better recall for the minority class (64%) while maintaining overall accuracy. This indicates that Random Forest can leverage synthetic samples effectively to improve minority class detection.
* **Support Vector Machine (SVM):**
  + Accuracy: 73.85%
  + AUC-ROC: 0.8064
  + SVM also showed improvements in recall for the minority class with SMOTE, reaching 73%. The model’s accuracy remained stable, suggesting that SVM can handle the variability introduced by synthetic samples.
* **XGBoost:**
  + Accuracy: 76.55%
  + AUC-ROC: 0.8143
  + XGBoost demonstrated balanced improvements with SMOTE, showing gains in recall without significant losses in accuracy. This makes SMOTE a viable option for enhancing XGBoost’s performance on imbalanced datasets.

**3.3 Class Weight Adjustment**

Class weight adjustment involves modifying the loss function to penalize misclassification of the minority class more heavily. This technique directly addresses imbalance within the model’s learning process.

* **Logistic Regression:**
  + Accuracy: 72.71%
  + AUC-ROC: 0.8345
  + Logistic Regression with class weight adjustment showed a significant improvement in recall (79%), similar to Random Over-Sampling. The overall accuracy and precision were comparable to other CI techniques, making this a strong method for Logistic Regression.
* **K-Nearest Neighbors (KNN):**
  + **Not Applicable:** KNN does not support class weight adjustments, so this technique could not be applied to this model.
* **Random Forest:**
  + Accuracy: 78.75%
  + AUC-ROC: 0.8187
  + Random Forest benefitted moderately from class weight adjustment, with slight improvements in recall for the minority class (48%). The overall accuracy was maintained, indicating that this technique helped the model become more sensitive to the minority class without major trade-offs.
* **Support Vector Machine (SVM):**
  + Accuracy: 72.49%
  + AUC-ROC: 0.8070
  + SVM with class weight adjustment showed improved recall for the minority class, reaching 78%. The technique allowed the model to balance sensitivity to both classes while maintaining a reasonable overall accuracy.
* **XGBoost:**
  + **Not Applicable:** XGBoost does not support class weight adjustments, so this technique could not be applied to this model.

**4. Conclusion**

This study highlights the importance of addressing class imbalance in classification tasks, particularly when dealing with customer churn prediction. The results showed that different CI techniques can significantly improve the recall for the minority class, which is crucial in identifying churned customers. However, these improvements often come at the cost of overall accuracy and precision, indicating a trade-off that must be carefully managed.

* **Random Over-Sampling** is effective in boosting recall for the minority class across most models, but it can reduce overall accuracy and precision, particularly in Logistic Regression and KNN.
* **SMOTE** offers a balanced approach, providing improvements in minority class recall without drastically impacting overall model performance. It is particularly effective in models like Random Forest and SVM.
* **Class Weight Adjustment** is a robust technique for models that support it