**Project Report: Predicting Road Accident Severity with Focus on Fatalities**

**Overview**

This step aimed to develop a predictive model to classify the severity of road accidents in France, focusing on improving the identification of **fatal accidents** (Class 2). The dataset provided detailed information about various factors such as road conditions, time of the accident, vehicle categories, and user demographics. The target variable was the **accident severity** (gravity), categorized as:

1. **Uninjured**
2. **Fatal**
3. **Hospitalized Injury**
4. **Minor Injury**

The primary objective was to enhance the model’s ability to predict **fatal accidents**, while maintaining a balance between **precision** (correctly predicting fatalities) and **recall** (identifying most of the actual fatalities).

**Steps Taken and Model Development Process**

**1. Data Loading and Preprocessing**

* The dataset was loaded and categorical variables such as **lighting conditions**, **atmosphere conditions**, and **collision type** were **label-encoded** for use in machine learning models.
* Missing values were handled and features like **latitude** and **longitude** were cleaned.

**2. Initial Model: Random Forest Classifier**

* A **Random Forest Classifier** was chosen as the initial model due to its robustness and ability to handle both categorical and numerical data.
* The model performed well in the majority of classes, like uninjured and minor injuries, but struggled to predict **fatal accidents** (Class 2).
* Initial performance metrics:
  + **Accuracy**: 0.63
  + **Precision for Fatalities**: ~0.30
  + **Recall for Fatalities**: ~0.16 (Many actual fatal accidents were missed)

**3. Handling Class Imbalance: SMOTE and Class Weights**

* **SMOTE (Synthetic Minority Oversampling Technique)** was applied to oversample the minority class (fatal accidents) and balance the dataset.
* Additionally, the **class weight for fatalities** was increased to give more importance to correctly predicting fatal accidents.

Results after applying SMOTE and class weights:

* **Accuracy**: 0.67
* **Precision for Fatalities**: 0.27
* **Recall for Fatalities**: 0.12 (Slight improvement)

**4. Hyperparameter Tuning: GridSearchCV**

* **GridSearchCV** was used to fine-tune key hyperparameters of the Random Forest model, including:
  + n\_estimators: Number of trees.
  + max\_depth: Maximum depth of trees.
  + min\_samples\_split: Minimum number of samples to split an internal node.
  + min\_samples\_leaf: Minimum number of samples required at a leaf node.
  + max\_features: Number of features considered for each split.

Best hyperparameters found:

* n\_estimators = 200
* max\_depth = 30
* min\_samples\_split = 5
* min\_samples\_leaf = 2
* max\_features = 'sqrt'

After tuning:

* **Precision for Fatalities**: ~0.30
* **Recall for Fatalities**: ~0.20

**5. Threshold Tuning for Fatalities**

* To improve **precision for fatalities**, threshold tuning was applied to adjust the probability threshold for classifying an accident as fatal. This allowed the model to be more confident in its predictions.

Initial results after threshold tuning:

* **Precision for Fatalities**: **1.00**
* **Recall for Fatalities**: **0.00**
* **Accuracy**: 0.99 (High accuracy due to strong performance on non-fatal cases)

The initial threshold tuning improved precision but caused the model to fail to identify any actual fatal accidents, resulting in zero recall for fatalities.

**6. Further Threshold Tuning for a Better Balance**

To address the recall issue, further threshold tuning was conducted using the **Precision-Recall Curve** to find the best threshold that balanced both precision and recall.

**Best Threshold Found**: 0.4195

**Results after further threshold tuning**:

* **Class False (Non-Fatal)**:
  + **Precision**: 0.98
  + **Recall**: 0.94
  + **F1-Score**: 0.96
  + The model continued to perform very well for non-fatal accidents.
* **Class True (Fatal)**:
  + **Precision**: 0.14
  + **Recall**: 0.32
  + **F1-Score**: 0.20
  + Recall for fatalities improved significantly from **0** to **0.35**, meaning the model was able to detect more true fatalities. However, precision remained low (0.14), indicating that many non-fatal cases were incorrectly classified as fatal.
* **Overall Metrics**:
  + **Accuracy**: 0.93
  + **Macro Average**:
    - **Precision**: 0.56
    - **Recall**: 0.65
    - **F1-Score**: 0.58
  + **Weighted Average**:
    - **Precision**: 0.96
    - **Recall**: 0.93
    - **F1-Score**: 0.94

This tuning resulted in a more balanced model that could detect fatalities more effectively, though at the expense of precision (more false positives for fatalities).

**7. Cost-Sensitive Learning Implementation**

Cost-sensitive learning was implemented by assigning different misclassification costs by adjusting the class weight parameter, which will give more importance to fatalities.

**Class-Specific Results:**

1. **Class 1 (Uninjured):**
   * Precision: 0.74
   * Recall: 0.81
   * F1-Score: 0.77
   * The model performs well in predicting uninjured cases, with strong recall (0.81) and good precision (0.74). This means the model is correctly identifying most uninjured cases while maintaining accuracy**.**
2. **Class 2 (Fatal):**
   * Precision: 0.26
   * Recall: 0.14
   * F1-Score: 0.18
   * Support: 2335 fatal cases
   * The recall for fatalities (0.14) is still low, meaning the model is missing many actual fatal accidents. However, precision (0.26) has improved compared to previous iterations, indicating that when the model predicts a fatal accident, it is slightly more likely to be correct than before.
   * The F1-score of 0.18 suggests that the balance between precision and recall is still modest but has improved with the cost-sensitive approach.
3. **Class 3 (Hospitalized Injury):**
   * Precision: 0.45
   * Recall: 0.54
   * F1-Score: 0.49
   * The model performs moderately well for hospitalized injuries, with recall and precision balanced around 0.45 to 0.54. This shows that the model is doing a reasonable job of identifying hospitalized cases.
4. **Class 4 (Minor Injury):**
   * Precision: 0.68
   * Recall: 0.59
   * F1-Score: 0.63
   * The model continues to perform reasonably well in predicting minor injuries, with a solid F1-score (0.63), indicating a good balance between precision and recall.

**Overall Model Performance:**

* Accuracy: 0.66
  + The model's overall accuracy is 66%, which reflects the majority class predictions. The model performs well for the majority classes but struggles with the minority class (fatalities).
* Macro Average:
  + Precision: 0.54
  + Recall: 0.52
  + F1-Score: 0.52
  + These averages show the model’s performance across all classes, with precision and recall being balanced around 0.50. However, this macro average reflects the model’s weaker performance on minority classes like fatalities.
* Weighted Average:
  + Precision: 0.66
  + Recall: 0.66
  + F1-Score: 0.66
  + The weighted average reflects the model’s good performance in the majority classes (uninjured and minor injuries), contributing more heavily to the overall result.

**Key Insights:**

1. **Improved Precision for Fatalities:**
   * The precision for Class 2 (Fatal) has improved to 0.26. This means that when the model predicts a fatal accident, there is a higher chance that the prediction is correct. This significantly improved over the baseline, where precision was much lower.
2. **Low Recall for Fatalities:**
   * Despite the cost-sensitive approach, the recall for Class 2 (Fatal) remains low at 0.14. This indicates that many actual fatalities are still being missed by the model. The cost-sensitive learning has not fully resolved the under-detection of fatal accidents.
3. **Hospitalized and Minor Injuries:**
   * The model performs moderately well for Class 3 (Hospitalized Injuries) and Class 4 (Minor Injuries), with recall and precision values above 0.45, resulting in reasonably good F1-scores.

**Conclusion:**

The Cost-Sensitive Learning implementation has improved precision for fatalities, but recall remains low, meaning the model is still struggling to identify a significant number of fatal accidents.

**8. Feature Engineering**

This classification report presents the performance of the Random Forest model after adding new **interaction features** to capture more subtle risk factors for fatal accidents. Here’s a breakdown of the results and a comparison with the previous model.

**Class-Specific Results:**

1. **Class 1 (Uninjured)**:
   * **Precision**: 0.74
   * **Recall**: 0.81
   * **F1-Score**: 0.77
   * The model performs well on uninjured cases, with a strong recall (0.81) and precision (0.74). These metrics are like previous results, meaning adding new features did not negatively impact the model’s ability to predict uninjured cases.
2. **Class 2 (Fatal)**:
   * **Precision**: 0.26
   * **Recall**: 0.14
   * **F1-Score**: 0.18
   * **Support**: 2335 fatal cases
   * The **recall for fatalities (0.14)** remains the same as before, meaning the model still misses many fatal accidents. However, **precision (0.26)** indicates that when the model predicts a fatal accident, it is more likely to be correct than in earlier model versions.
   * The **F1-score** of 0.18 reflects the balance between precision and recall for fatalities, indicating that performance in predicting fatal accidents remains modest despite the feature engineering.
3. **Class 3 (Hospitalized Injury)**:
   * **Precision**: 0.45
   * **Recall**: 0.54
   * **F1-Score**: 0.49
   * Like previous iterations, the model performs moderately well for hospitalized injuries. This suggests that the newly introduced interaction features did not significantly affect the model’s ability to predict this class.
4. **Class 4 (Minor Injury)**:
   * **Precision**: 0.68
   * **Recall**: 0.59
   * **F1-Score**: 0.63
   * The model performs reasonably well for minor injuries, with a good F1-score (0.63) and balanced precision and recall.

**Overall Model Performance:**

* **Accuracy**: 0.66
  + The overall accuracy remains consistent with previous iterations (66%), reflecting good performance on the majority classes but ongoing struggles with the minority class (fatalities).
* **Macro Average**:
  + **Precision**: 0.54
  + **Recall**: 0.52
  + **F1-Score**: 0.52
  + These macro averages represent the model’s performance across all classes. While precision and recall are relatively balanced, the low recall for Class 2 (Fatal) pulls down the overall score.
* **Weighted Average**:
  + **Precision**: 0.66
  + **Recall**: 0.66
  + **F1-Score**: 0.66
  + The weighted average reflects the model’s performance in the majority classes, particularly uninjured and minor injuries, where it performs better.

**Key Insights After Feature Engineering:**

1. **Impact of Feature Engineering on Fatal Accidents**:
   * Despite introducing new interaction features (such as combining lighting conditions with time, and weather with location), there is **no significant improvement in recall** for fatal accidents. The recall remains at 0.14, meaning the model continues to miss a substantial portion of actual fatal accidents.
   * However, the **precision for fatalities** (0.26) suggests that when the model does predict a fatal accident, it is doing so more accurately than before.
2. **Effect on Other Classes**:
   * The addition of interaction features did not negatively affect the model's performance on other classes (uninjured, hospitalized injuries, minor injuries). Performance on these classes remains consistent with previous iterations.

**Conclusion:**

The **Feature Engineering** step introduced additional interaction terms, but the recall for fatalities did not improve significantly. The model remains good at predicting the majority classes but struggles with correctly identifying fatal accidents

**9. Feature Engineering**

**Class-Specific Results:**

1. **Class False (Non-Fatal)**:
   * **Precision**: 0.98
   * **Recall**: 0.95
   * **F1-Score**: 0.97
   * The model performs very well in identifying non-fatal accidents, with high precision (0.98) and recall (0.95). This means the model accurately predicts most non-fatal cases, with very few false positives for fatal accidents.
   * The high **F1-score** of 0.97 indicates an excellent balance between precision and recall for non-fatal cases.
2. **Class True (Fatal)**:
   * **Precision**: 0.17
   * **Recall**: 0.37
   * **F1-Score**: 0.23
   * **Support**: 2335 fatal cases
   * The **recall for fatalities (0.37)** has increased significantly compared to previous iterations, which means the model is now identifying more actual fatal accidents. The recall improvement shows that the model is better at detecting fatal cases after lowering the threshold.
   * **Precision (0.17)** for fatalities has decreased slightly, which means more non-fatal accidents are being incorrectly classified as fatal (false positives).
   * The **F1-score (0.23)**, while modest, reflects the improvement in recall for fatalities. The model is now identifying more true fatal accidents at the expense of some false positives.

**Overall Model Performance:**

* **Accuracy**: 0.94
  + The overall accuracy remains high at 94%, primarily driven by the model’s strong performance on the majority class (non-fatal accidents). The impact of the low recall and precision for fatalities is minimal due to the class imbalance.
* **Macro Average**:
  + **Precision**: 0.58
  + **Recall**: 0.66
  + **F1-Score**: 0.60
  + The macro average gives equal weight to each class, meaning the improved recall for fatalities (Class 2) has led to a balanced average. This suggests that overall, the model is achieving a better balance between precision and recall.
* **Weighted Average**:
  + **Precision**: 0.96
  + **Recall**: 0.94
  + **F1-Score**: 0.95
  + The weighted averages remain high, reflecting the model’s dominant performance on the majority class (non-fatal cases).

**Key Insights After Threshold Adjustment:**

1. **Improved Recall for Fatalities**:
   * The **recall for Class 2 (Fatal)** increased to **0.37**, meaning that the model is now correctly identifying more fatal accidents than before. The threshold adjustment to **0.2** allows the model to be less conservative, leading to more true positives for fatalities.
2. **Precision Trade-off**:
   * **Precision for fatalities** has decreased slightly to **0.17**, indicating more non-fatal accidents are being misclassified as fatal (false positives). This is a common trade-off when recall is prioritized, as lowering the threshold makes the model less strict, leading to more predictions of fatalities, including false positives.
3. **Overall Balance**:
   * The overall balance between precision and recall for fatalities has improved, as reflected by the **F1-score (0.23)**, which has increased compared to previous attempts. This suggests the model is now better at detecting fatalities, though there is still room for improvement in precision.

**Conclusion:**

The threshold adjustment to **0.2** has significantly improved **recall for fatalities (0.37)**, meaning the model is now identifying more true fatal accidents. However, this came at the expense of some precision, resulting in more false positives. The overall model accuracy remains high at 94%, and the balance between precision and recall for fatalities has improved, as reflected by the **F1-score**. Further adjustments or advanced models may help improve performance further.