Kaggle Competition

Detection of extreme weather events

UDEM

Name:	**Sam Blouin**
Username:	https://www.kaggle.com/samblouin
Course:	6390B
PC:	20035269

REPORT

Machine Learning for Climatic Event Prediction: An Analytical Reflection

Introduction

Our machine learning pipeline aims to accurately classify events, integrating preprocessing, feature engineering, and algorithm deployment. Despite achieving high initial accuracy, the challenge of overfitting emerged, necessitating refinement in our hyperparameter tuning process.

Feature Design

The preprocessing pipeline was essential, utilizing standardization and normalization to harmonize feature scales across different algorithms, particularly for SoftLogisticRegression. We engineered features with a focus on maximizing informational value and computational efficiency.

Algorithms

Our classifier suite included SimpleDummyClassifier, SoftLogisticRegression, built from scratch and are the most significant job done in the competition despite it is not seen during the competition as we used libraries to get our best models like we got RandomForestClassifier; it was chosen for it's better result but we could have explore more. We didnt get enough good results from, by example, optula library at first so we stayed with RF.

Methodology

An 80-20 training-validation split was employed to balance learning and validation. Preprocessing, regularization and careful hyperparameter tuning were applied to optimize performance, with subsequent adjustments made to address overfitting.

Results and Analysis

The RandomForest classifier demonstrated superior performance, highlighted by its weighted F1-score. A project extension enabled us to refine our methods, resulting in improved predictive capabilities.

0.0.1 Analysis of Soft Logistic Regression Model Performance

The performance metrics for the Soft Logistic Regression model reveal a significant disparity in effectiveness across the predicted labels. Below, we delve into the implications of the precision, recall, and F1-score for each label:

Label 0 *Precision:* 0.81, *Recall:* 0.98, *F1-score:* 0.89
The model exhibits outstanding recall with decent precision for Label 0. An F1-score approaching 0.9 signifies a robust ability to correctly identify and capture the majority of instances for this class. This suggests that for Label 0, the model has learned the underlying patterns very well and can be considered reliable.

Label 1 *Precision:* 0.72, *Recall:* 0.04, *F1-score:* 0.07
The precision for Label 1, although seemingly high, is overshadowed by the extremely low recall, resulting in a negligible F1-score. Such a low recall indicates that the model is almost always overlooking the true instances of Label 1. This could be indicative of under-representation of Label 1 in the training data or insufficient model capacity to capture the characteristics of this particular class.

Label 2 *Precision:* 0.63, *Recall:* 0.13, *F1-score:* 0.21
The model's performance on Label 2 is suboptimal, characterized by moderate precision and poor recall. The resulting F1-score reflects the model's limited capability in correctly identifying Label 2 instances, suggesting difficulties in distinguishing this class from others or a lack of representative features.

Strategic Implications for Model Improvement The model's performance disparity across labels calls for a strategic evaluation of the data and model configuration. Addressing the under-performance for Labels 1 and 2 may involve investigating data imbalance and employing techniques like resampling, cost-sensitive learning, or exploring alternative feature representation. Furthermore, refining the model with a more sophisticated algorithm or an ensemble approach could enhance its discriminative power for the underperforming classes.

Conclusion and Reflections

The quest for model optimization revealed the delicate trade-off between precision tuning and overfitting. The project's extended timeline was instrumental in allowing for methodological refinements, culminating in a robust and more generalized model but late 3 hours after dead line because we canceled my search when us, said "target the deadline for any late submission" and finally changed the set up I (if you remember our last piazza private post) and the mind the next morning. The improvements are significant on Kaggle because the private score is way over '*best_baseline.csv*' Thank You for Understanding.

Statement of Contributions

I affirm that all work presented in this report is solely my own.

Table 1: Model Performance Evaluation

Model	Precision	Recall	F1-score
	SVC		
Label 0	0.98	0.76	0.86
Label 1	0.41	0.96	0.57
Label 2	0.54	0.91	0.67
Overall F1-score		0.8156	
	Random Forest (Best)		
Label 0	0.98	0.87	0.92
Label 1	0.66	0.90	0.72
Label 2	0.64	0.90	0.75
Overall F1-score		0.8814	
	Random Forest		
Label 0	0.91	0.91	0.91
Label 1	0.66	0.59	0.62
Label 2	0.64	0.64	0.64
Overall F1-score		0.8521	
	Logistic Regression		
Label 0	0.95	0.58	0.72
Label 1	0.43	0.91	0.36
Label 2	0.41	0.85	0.55
Overall F1-score		0.6756	
	XGBoost		
Label 0	0.85	0.97	0.91
Label 1	0.86	0.37	0.52
Label 2	0.76	0.36	0.49
Overall F1-score		0.8214	
	Simple Dummy		
Label 0	0.78	0.34	0.47
Label 1	0.04	0.33	0.07
Label 2	0.16	0.32	0.22
Overall F1-score		0.4122	
	Soft Logistic Regression		
Label 0	0.81	0.98	0.89
Label 1	0.72	0.04	0.07
Label 2	0.63	0.13	0.21
Overall F1-score		0.7387	
	Dummy		
Label 0	0.79	0.33	0.47
Label 1	0.04	0.32	0.07
Label 2	0.16	0.31	0.21
Overall F1-score		0.4090	
	SGD		
Label 0	0.89	0.86	0.87
Label 1	0.52	0.44	0.47
Label 2	0.52	0.64	0.58
Overall F1-score		0.8070	