Title: Machine Learning for Predicting Wildfire Occurrence

Name: Mia Oxer

Affiliation: Brown University

PhD Ecology, Evolution, and Organismal Biology

Link to Git-hub repository: https://github.com/miaoxer/widfire_final_project

Word Count: 1999

1. Introduction

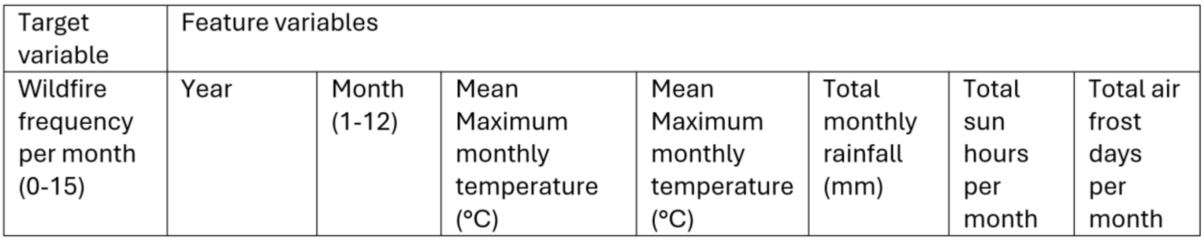
Background

Climate change is expected to increase wildfire frequency in the moorland areas of the Peak District National Park (PDNP), UK. Rising temperatures and reduced rainfall will make predicting shifts in wildfire occurrences crucial for effective management and protection[1]. Using a dataset from PDNP rangers documenting monthly wildfire frequency from 1976 to 2020, this report explores the application of machine learning to predict monthly wildfire occurrences based on climate data.

The Dataset

The analysis uses monthly wildfire frequency data from 1976 to 2020, along with publicly available climate data for Sheffield, UK, sourced from the MET Office[2]. The dataset covers 537 months, from April 1976 to December 2020, with corresponding target variable data points. A summary of the dataset variables is provided in Table 1.

Table 1: Summary of the target variable and feature variables with measurement units



Target variable	Feature variables						
Wildfire frequency per month (0-15)	Year	Month (1-12)	Mean Maximum monthly temperature (°C)	Mean Maximum monthly temperature (°C)	Total monthly rainfall (mm)	Total sun hours per month	Total air frost days per month

The ML Problem

Predicting wildfire occurrence from climate features makes this a time series classification problem. The nature of the target variable means this is an imbalanced classification problem, with a predominance of class 0 (no wildfires). Therefore, this report considers the problem as a binary classification: class 0 (no wildfires) and class 1 (wildfires). While no prior machine learning models have been applied to this dataset, research on wildfire drivers in the PDNP has been conducted (see Albertson et al., 2010; McMorrow et al., 2006; Millin-Chalabi et al., 2013).

2. EDA

Wildfire occurrence

A breakdown of wildfire frequency indicates that wildfire presence only accounts for 18.9 percent of all data (Figure 1).

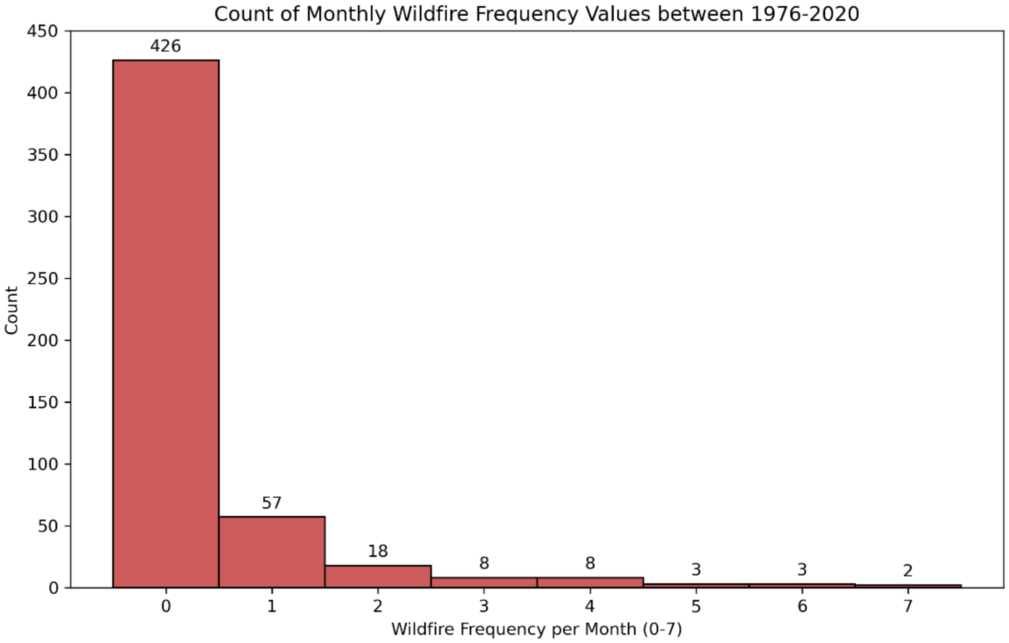


Figure 1: Wildfire frequency count between 1976-2020

Viewing the data as a binary classification, where class 1 represents the presence of one or more wildfires per month, shows an increase in the number of years with wildfires between 1976-2020 (Figure 2). This trend signals towards increasing wildfires over time, with the highest spike in 2017 where wildfires occurred across 8 months of the year. The most popular months for wildfire presence include May, followed by April and June (Figure 3).

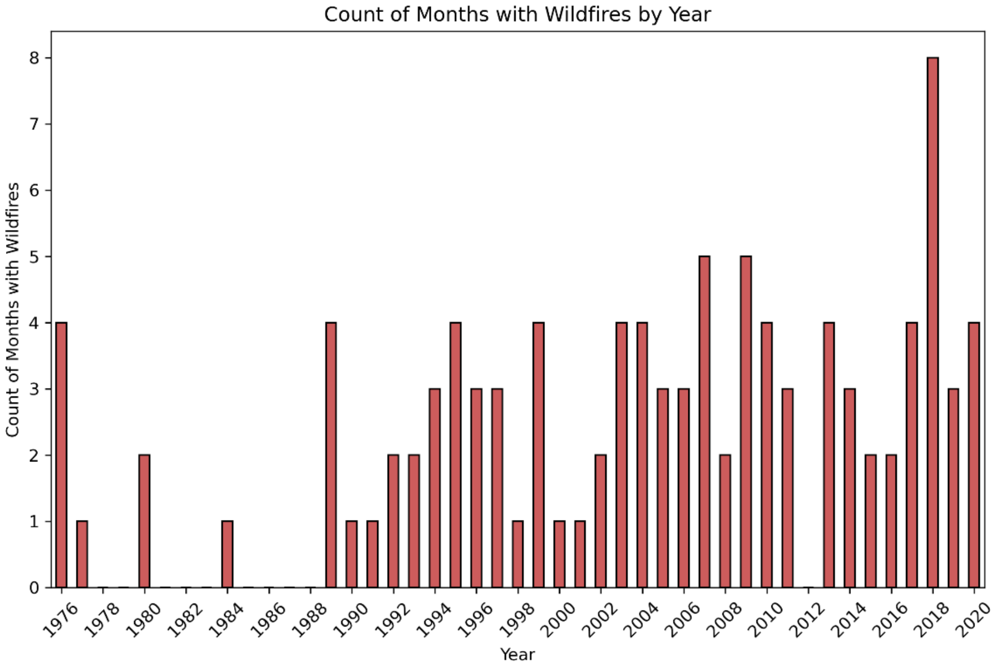


Figure 2: Count of months with wildfire occurrences by year between 1976-2020

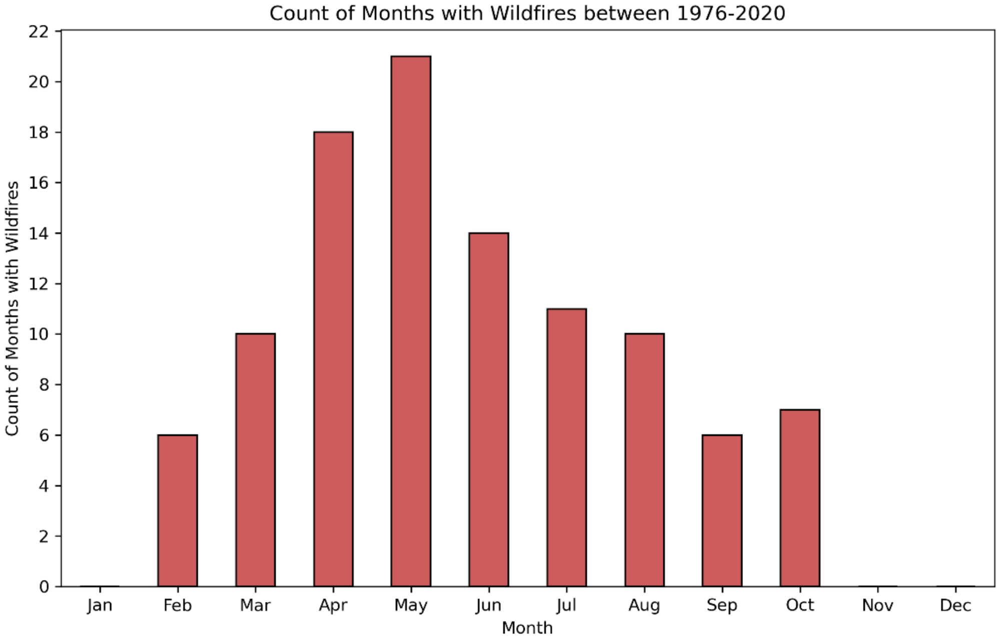


Figure 3: Count of months with wildfire occurrences by month between 1976-2020

Climate

Converting year and month variables to 'year_month' datetime, allows the plotting of mean monthly maximum and minimum temperature between April 1976 and December 2020

(Figure 4). Pearson correlation coefficients indicate a slight positive relationship between increasing temperature over time, which is statistically significant for tmin_degC, where *p=0.04*.

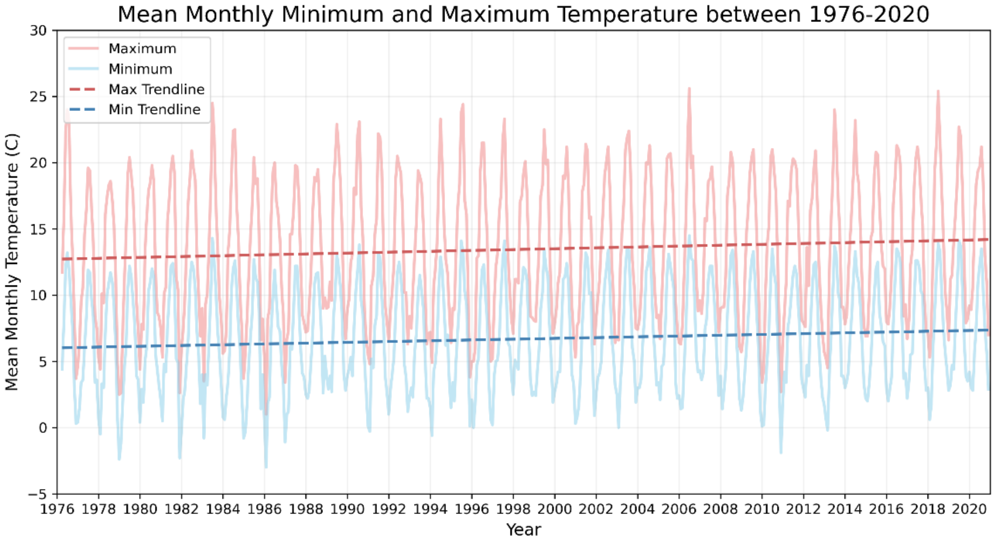


Figure 4: Mean monthly maximum and minimum temperature between 1976-2020; trendline calculated as a linear fit between variables

A clear relationship between tmax_degC and month can be seen, with peak temperatures in Sheffield, UK occurring between June, July, and August (Figure 5).

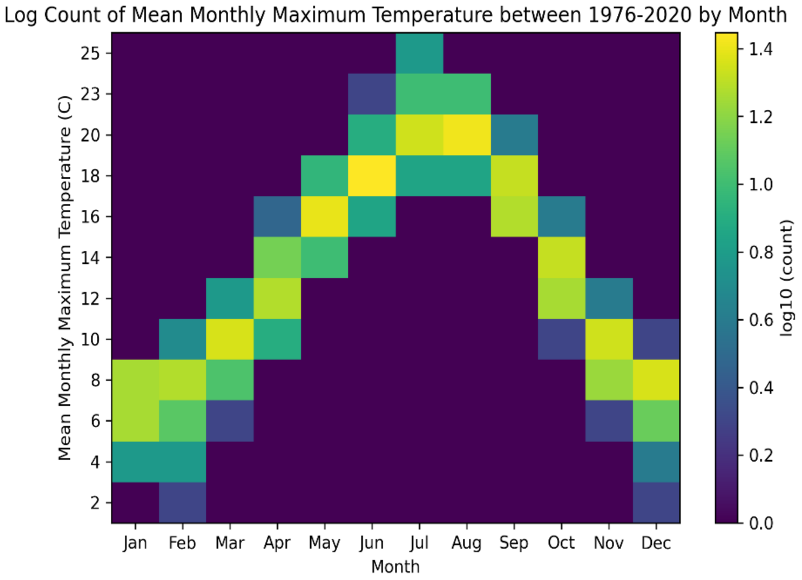


Figure 5: Log count of mean monthly maximum temperature by month between 1976-2020.

Total sun hours per month mirror this bell-shaped temperature curve, with peaks in mean sun hours occurring in May and July (Figure 6a). Small standard deviations below and above the mean indicate limited variability in these hours across the dataset. Mean air-frost days per month opposes these trends with peaks in January and February (Figure 6b).

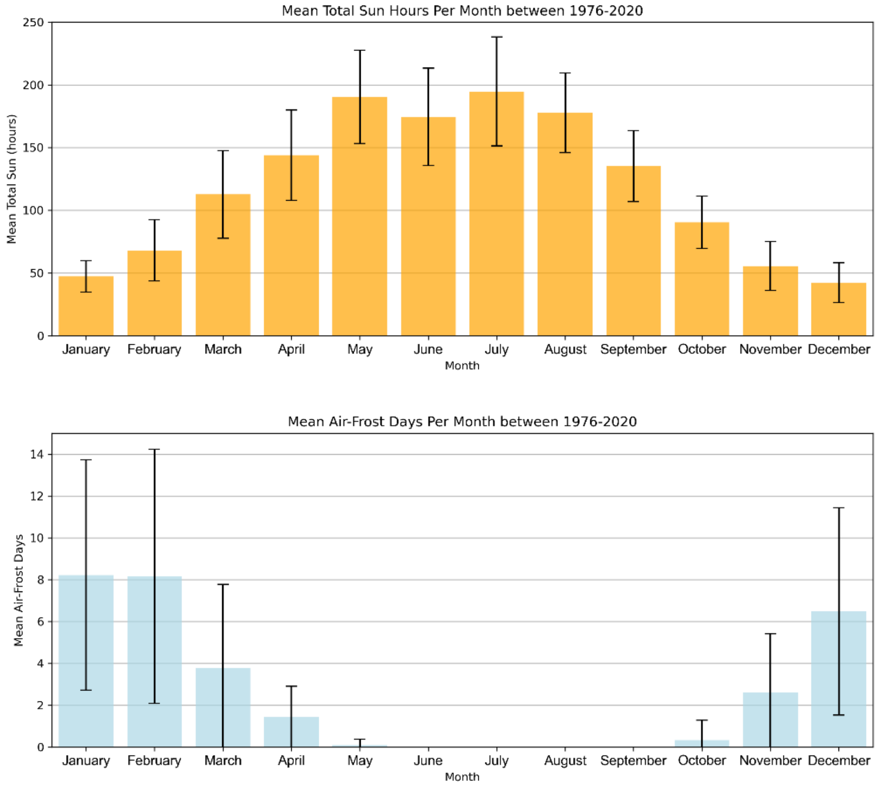


Figure 6: (a) Mean totalsun_hours per month between 1976-2020. Error bars representing mean plus and minus the standard deviation. (b) Number of mean air-frost_days (air-temperature reaching below 0°C) per month between 1976-2020

Climate and Wildfire Occurrence

Understanding the relationship between climate features and wildfire occurrence is important for determining feature engineering approaches. Specifically, a positive correlation can be found between summer temperatures and wildfire occurrence (Figure 7).

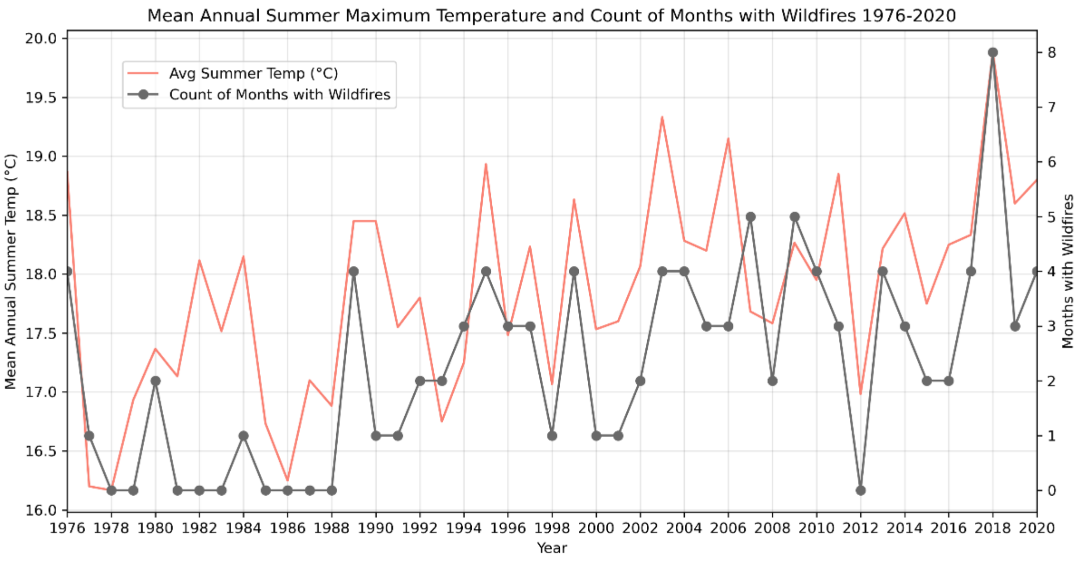


Figure 7: Average of mean monthly maximum temperature across summer months (April-September) against months with wildfires.

Plotting of wildfire occurrence with mean monthly maximum temperature and total summer rain averaged across the summer months (May, June, July, August), indicates an opposing relationship between rainfall and wildfires (Figure 8). A peak in months with wildfires in 2018 is also associated with the highest mean summer maximum temperature at 20°C.

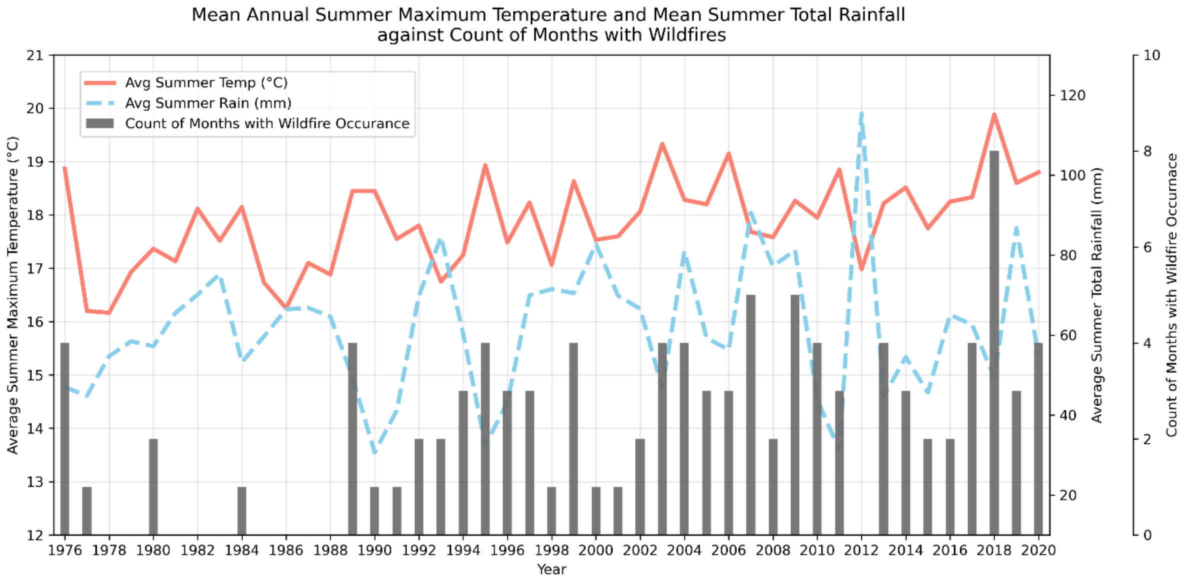


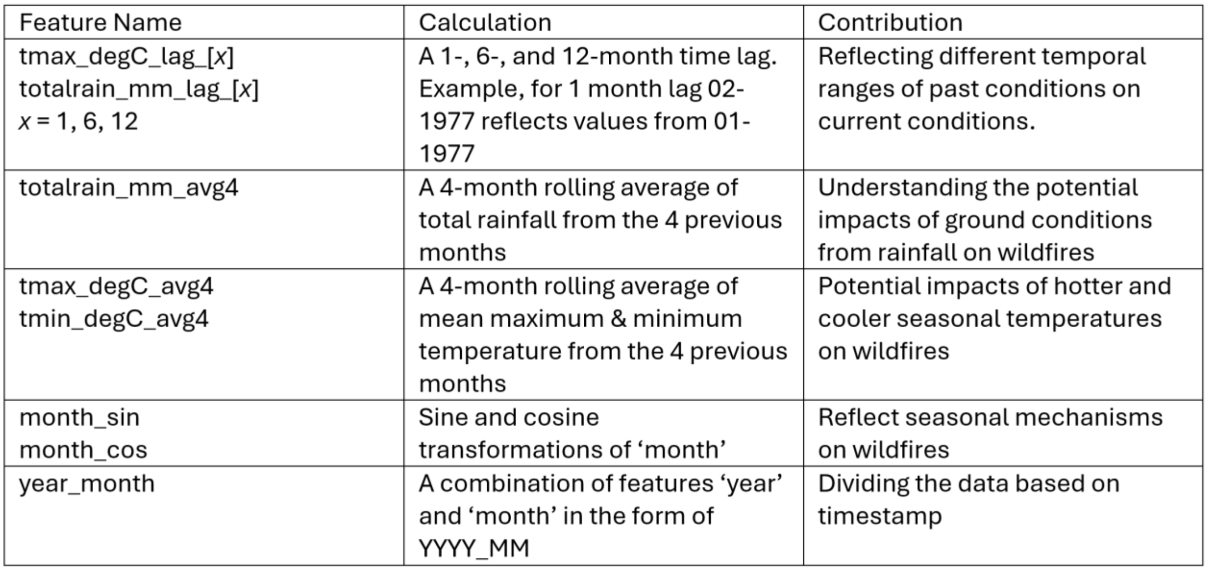
Figure 8: Count of months with wildfires against mean summer maximum temperature (°C) and mean summer total rainfall (mm)

3. Methods

Feature Generation and Selection

Before data splitting, feature generation and selection were important for improving predictability. The following features were engineered:

Table 2: Table of engineered feature names, how they were calculated, and their contribution



Feature Name	Calculation	Contribution
tmax_degC_lag_[*x*] totalrain_mm_lag_[*x*] *x* = 1, 6, 12	A 1-, 6-, and 12-month time lag. Example, for 1 month lag 02-1977 reflects values from 01-1977	Reflecting different temporal ranges of past conditions on current conditions.
totalrain_mm_avg4	A 4-month rolling average of total rainfall from the 4 previous months	Understanding the potential impacts of ground conditions from rainfall on wildfires
tmax_degC_avg4 tmin_degC_avg4	A 4-month rolling average of mean maximum & minimum temperature from the 4 previous months	Potential impacts of hotter and cooler seasonal temperatures on wildfires
month_sin month_cos	Sine and cosine transformations of ‘month’	Reflect seasonal mechanisms on wildfires
year_month	A combination of features ‘year’ and ‘month’ in the form of YYYY_MM	Dividing the data based on timestamp

The creation of lag features meant the first 12 months of the dataset were removed. Highly correlated features ($p>0.95$ or $p<-0.95$) were deleted from the dataset to reduce noise. This led to the removal of tmin_degC due to a strong positive correlation with tmax_degC (Figure 9).

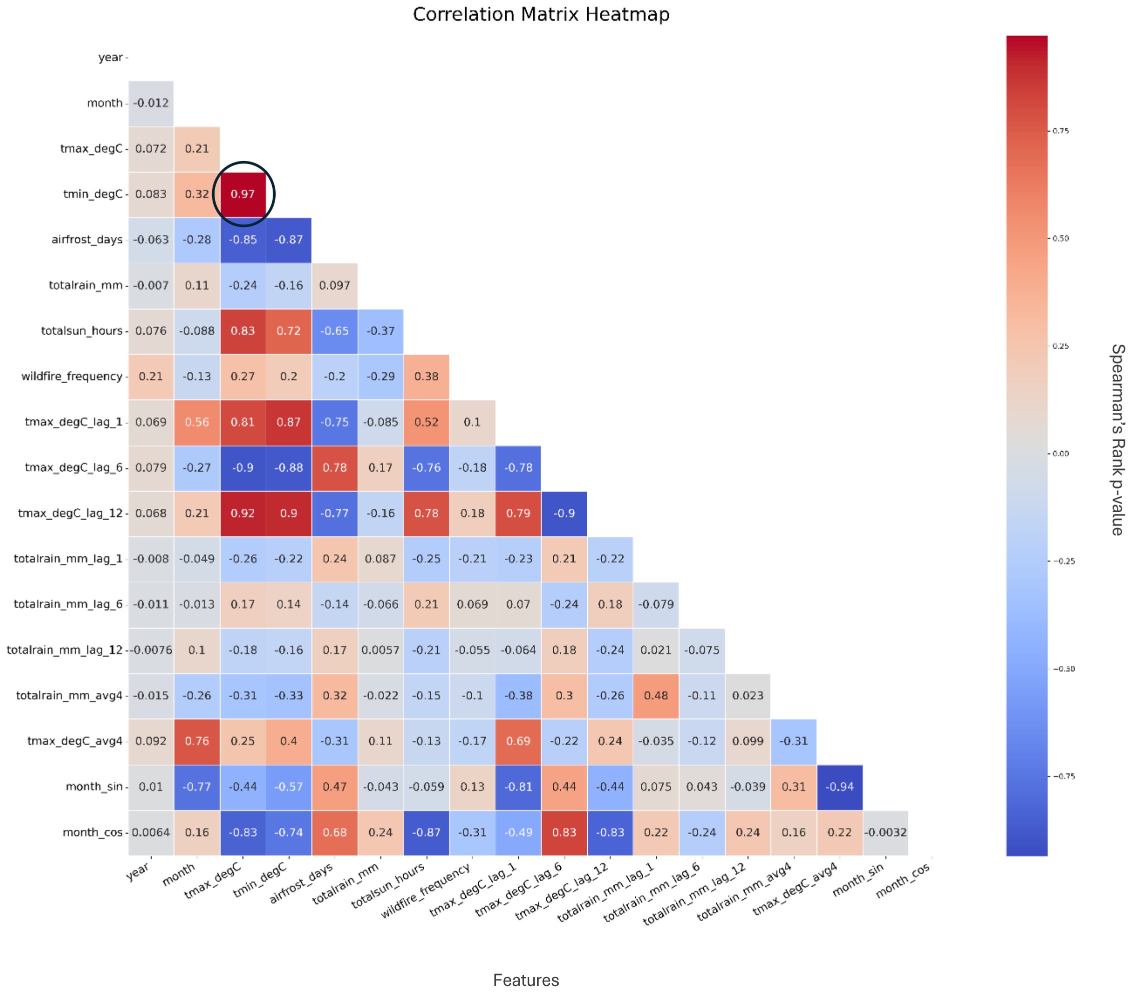


Figure 9: Feature correlation matrix using spearman's rank to account for non-linear relationships between variables.

Splitting and Cross Validation

SciKitLearn's TimeSeriesSplit was the most appropriate approach for handling timeseries data with evenly spaced observations[6]. A fold of 5 was selected to balance between the small dataset size and robustness of the model's cross-validation (Figure 10). These 5 folds were split into train, validation, and test to present more effective best model selection and validation across different time windows.

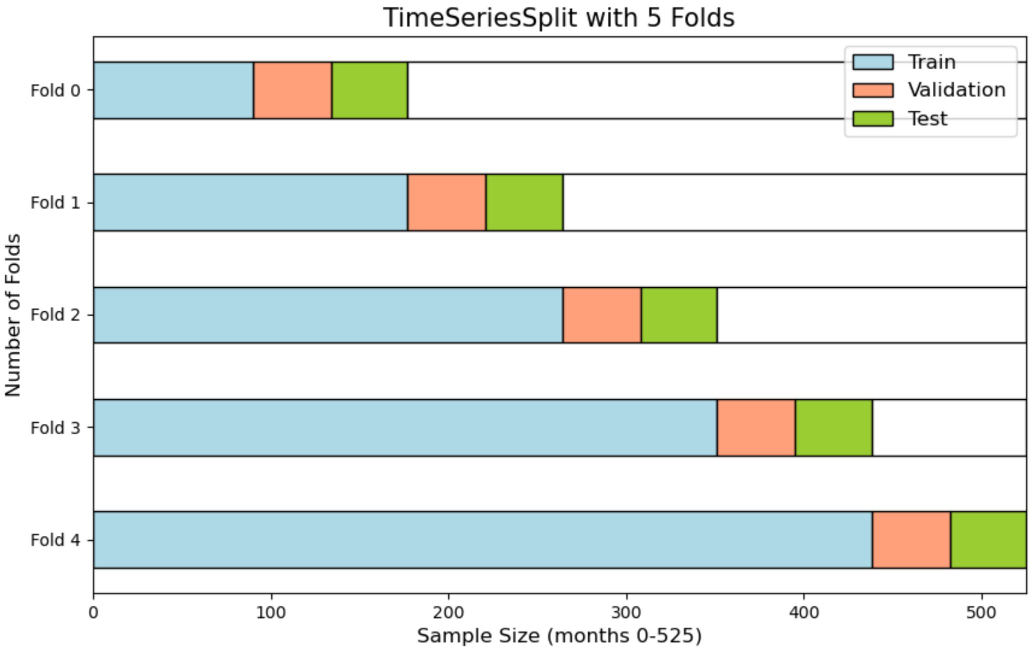


Figure 10: TimeSeriesSplit with 5 folds across the final sample size of 525 months with a final fold of 80% train, 10% test and 10% validation.

ML Pipeline and GridSearchCV

Pre-processing, model selection, and hyperparameter tuning were all integrated into the pipeline for each machine learning model. Since all selected features are continuous without defined bounds, SciKitLearn’s StandardScaler was applied separately to each training, validation, and test fold to prevent data leakage. The hyperparameters for the four selected models — Logistic Regression, Random Forest Classifier, XGBoost, and SVC — were tuned using GridSearchCV (see Table 2). Hyperparameters were adjusted based on the performance of each model. Regularization was applied to Logistic Regression and XGBoost to prevent overfitting.

Table 3: Range of hyperparameters tuned using GridSearchCV for each model

Supervised ML Model	Hyperparameter Range			Regularisation
Logistic Regression	C = Np.logspace(-2, 2, 21)	Solver = saga, sag, lbfgs, newton-cg		L2, None
Random Forest Classifier	n_estimators = 100	max_depth = 1, 2, 3, 10, 30, 100	max_features = 0.25, 0.5, 0.75, 1.0	
XGBoost	n_estimators = 500	max_depth = 3, 5, 10, 50, 100	Learning_rate = 0.01, 0.05, 0.1	Logloss, Early Stopping (15)
SVC	gamma = 0.1, 1, 10, 100, 1000	C = 1, 0.1, 0.01, 0.001, 0.0001		

The availability of a train, validation, and test set enabled for the grid_search.fit() on the training set, best model selection using the validation set, and then a final evaluation on the test sets for each of the 5 folds.

Evaluation

Several evaluation metrics were considered, but due to the imbalanced dataset and high true negatives, accuracy was not suitable. Recall is key for identifying positive cases (wildfire occurrences), while F1 balances recall and precision. F1.5 emphasizes recall more, and PR AUC balances precision and recall, with less impact from true negatives, making it ideal for imbalanced datasets. The final metrics chosen for model comparison were Recall, Precision, F1, F1.5, and PR AUC.

Uncertainties

The use of 5 folds across the data enabled the generation of 5 test results per mode, offering insights into how the model performed with more data over time. Re-running the models with different random seeds was used to consider uncertainty surrounding model non-determinism. Considering both the mean and standard deviation of overall model performance across the 5 folds allowed for quantification of uncertainty between sets.

4. Results

Baseline scores and Model Comparison

Due to the imbalanced nature of the binary classes, baseline metrics were calculated on the assumption that only the minority class would be selected. If the majority class is assumed, Recall, F1, and F1.5 become undefined. As such, the following definitions for the confusion matrix terms were used inside the metric equations:

TP = Count of class 1 (minority)

FP = Count of class 0

TN = 0

FN = 0

Baseline scores and test scores were calculated on each fold 1 – 5 for the four selected ML models. A mean and standard deviation for each of the test scores was then calculated for each of the models. The baseline scores associated with the folds 1-5 were also averaged for each model (see Table 4 and Figure 11).

Table 4: Mean and standard deviation of test prediction results for the following models for accuracy, f1, f1.5, recall, precision, and PR AUC compared alongside a mean of the baseline for the four selected models.

		Logistic Regression		Random Forest Classifier		XGBoost		SVC	
Evaluation Metric	Baseline	Score	SD	Score	SD	Score	SD	Score	SD
Accuracy	0.246	0.646	0.081	0.749	0.102	0.682	0.123	0.586	0.054
F1.5	0.501	0.558	0.161	0.469	0.267	0.342	0.281	0.522	0.163
F1	0.387	0.486	0.137	0.427	0.244	0.314	0.239	0.446	0.145
Recall	1.000	0.743	0.236	0.562	0.312	0.418	0.395	0.730	0.238
Precision	0.245	0.367	0.102	0.348	0.203	0.276	0.178	0.327	0.112
PR AUC	0.245	0.580	0.148	0.500	0.246	0.508	0.172	0.558	0.155

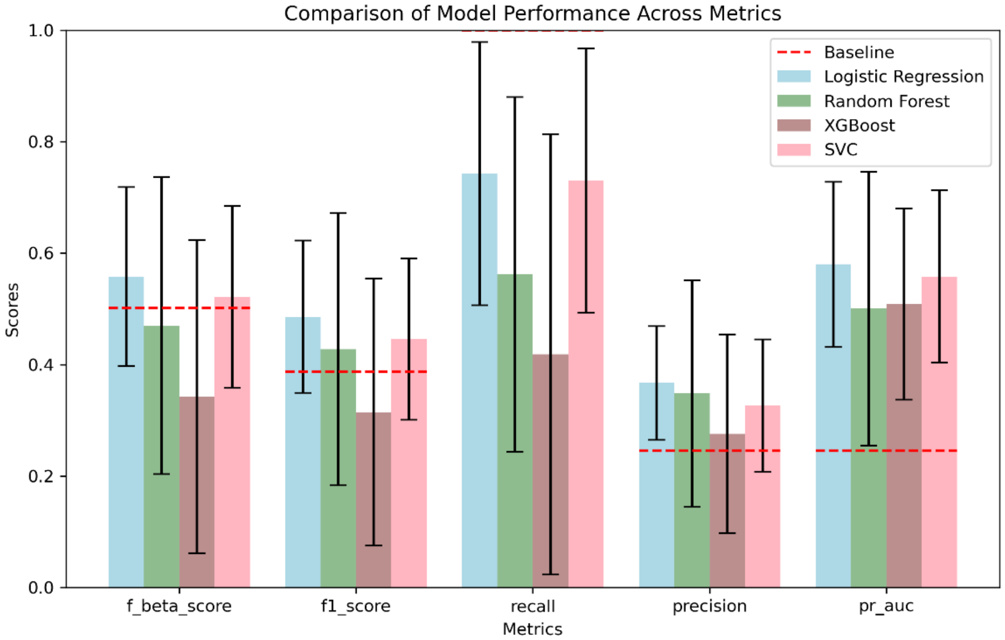


Figure 11: Test scores and associated standard deviations represented as error bars for each model based on an average of the 5 folds for each metric. Baselines shown in red.

Logistic regression and SVC performed the best, exceeding baseline scores for each metric. Across the four models, standard deviations remain high, likely in response to the nature of the TimeSeriesSplit data folds where fold 1 consistently performed less effectively than fold 5 with more training data. The highest scoring model was logistic regression under all evaluation metrics. As such, this model was selected for further global and local feature importance inspection.

Logistic Regression Inspection and Feature Importance

A confusion matrix for each fold shows how the logistic regression model improves with larger training sizes and broader temporal range (Figure 12). Each fold increases true positives, which is crucial for this analysis, while the reduction in false negatives in higher folds indicates success. Whereby, predicting no wildfires when one occurs is more problematic than false positives.

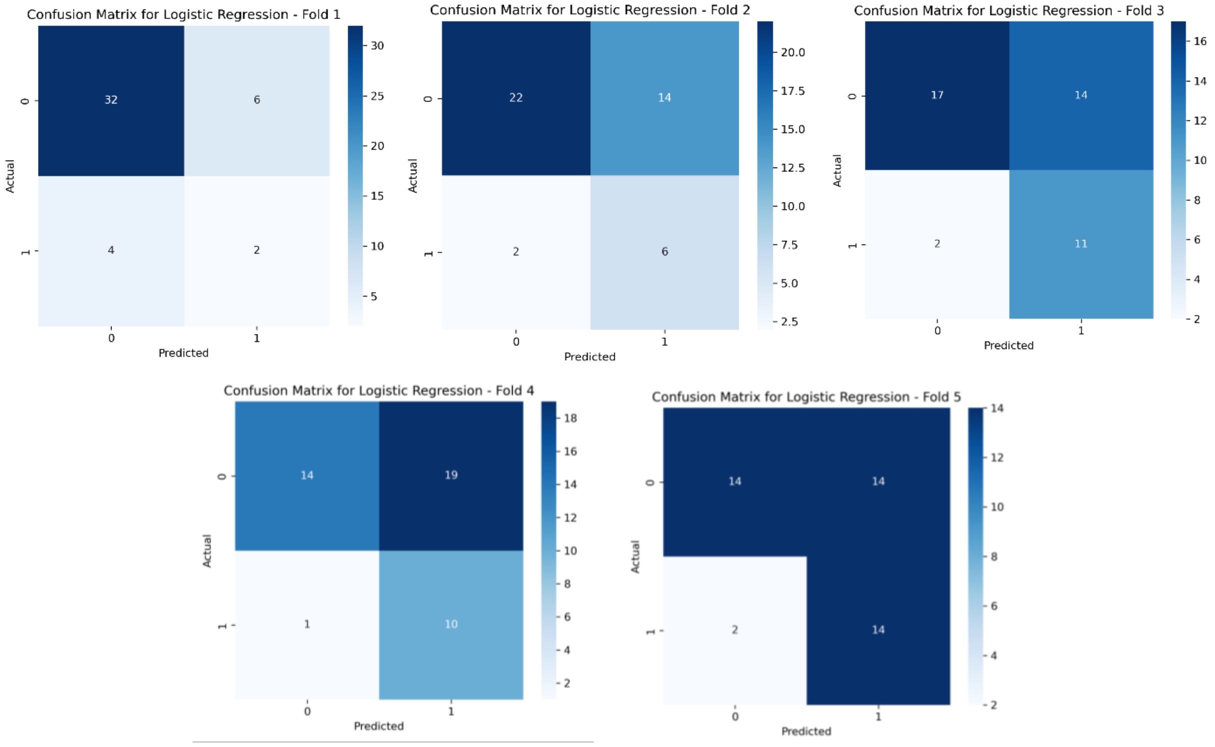


Figure 12: Confusion matrix for folds 1-5 from the logistic regression model.

Global feature importance reveals how the model uses features to make predictions. Specifically, focusing on the model output for fold 5, the coefficient magnitudes indicate year as the most significant feature, followed by totalsun_hours, and then totalrain_mm, totalrain_mm_lag_1, tmax_degC (Figure 13).

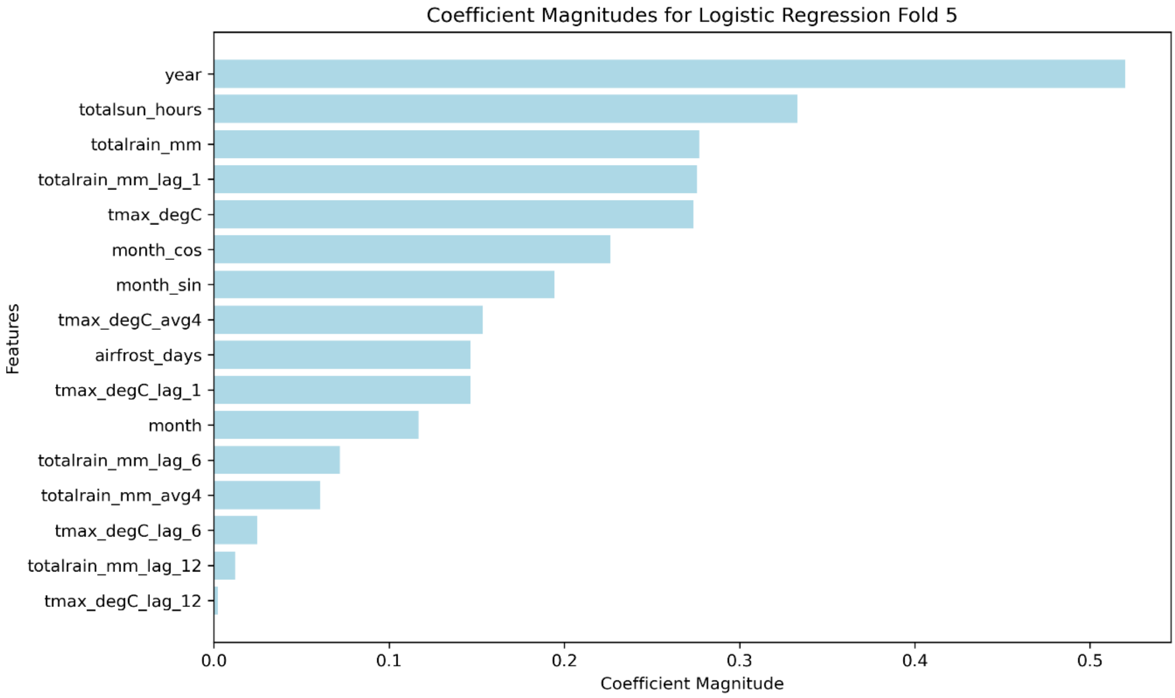


Figure 13: Coefficient magnitudes for logistic regression fold 5

This display of feature importances can also be reflected by a global SHAP analysis, whereby totalsun_hours, year, totalrain_mm and totalrain_mm_lag_1 have the most influence on SHAP values (Figure 14). Totalsun_hours impacts the SHAP value both positively and negatively, with higher sun hours increasing the probability of class 1 and lower sun hours decreasing it. Interestingly, year consistently has a positive impact, regardless of its value. As expected, high monthly totalrain_mm and totalrain_mm_lag_1 decrease wildfire probability. While temperature variables have lesser an impact, a positive relationship between a high tmax_degC and SHAP values is observed.

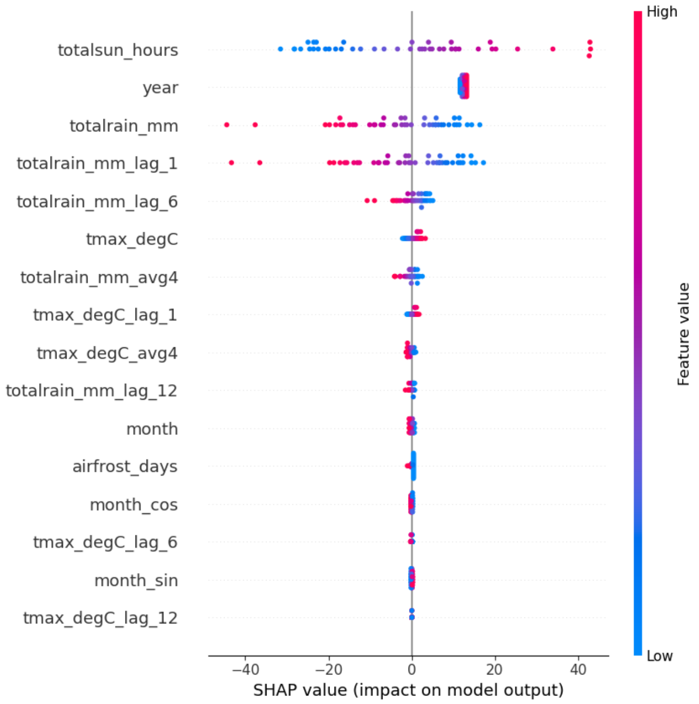


Figure 14: Global SHAP feature importance for fold 5 logistic regression

The permutation feature importance indicates a similar emphasis on the predictive power of total rain, and total sun hours (Figure 15). However, this time a recognition towards the cosine transformation of month indicates the potential influence of seasonality upon wildfire prediction. The identical mean and baseline test scores for the lower 3 features indicate they are not contributing to model predictions.

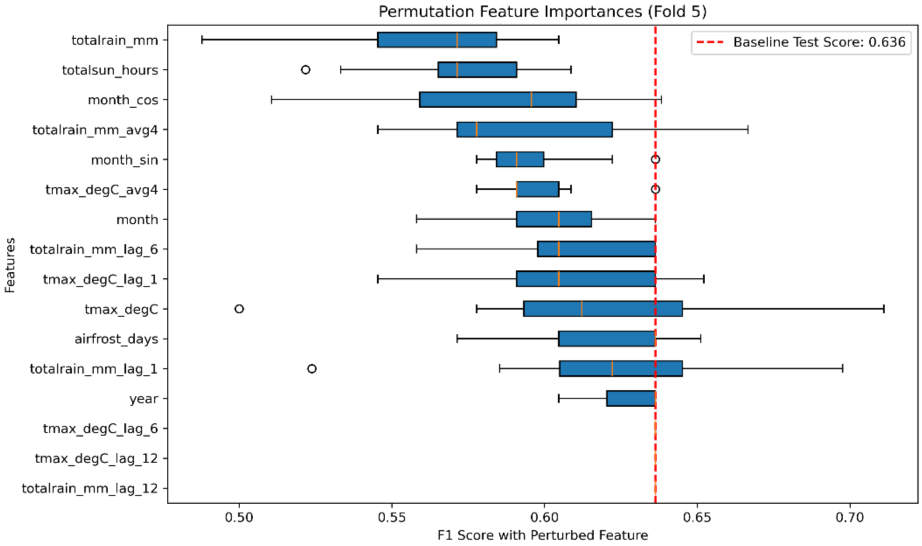
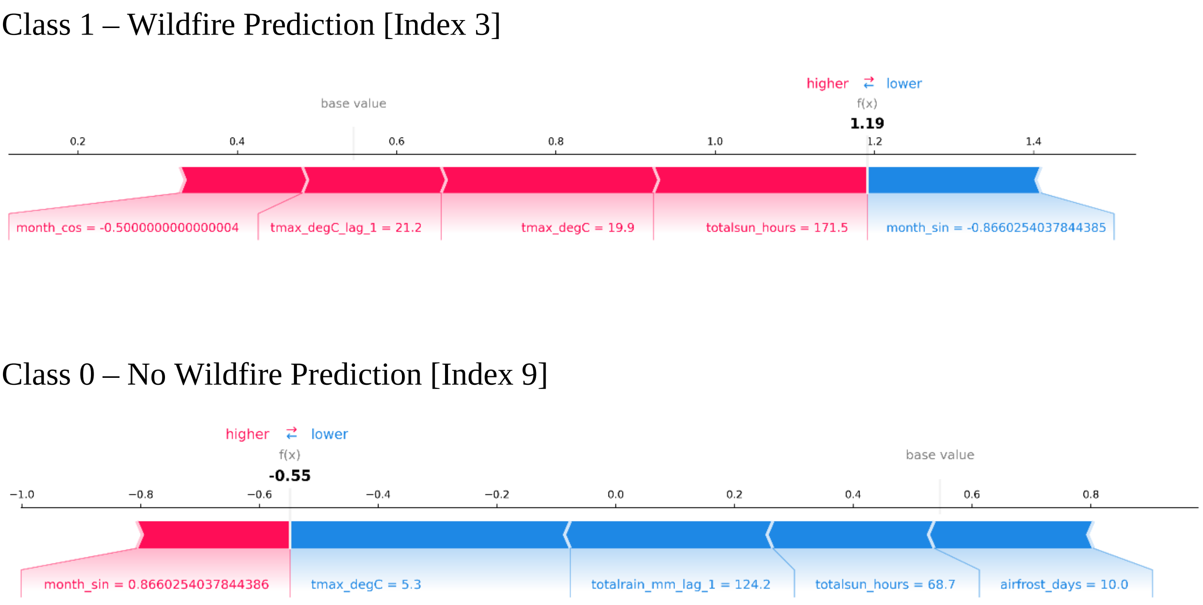


Figure 15:Permutation feature importances for logistic regression model fold 5

Local feature importance provides insight into the predictive power of individual features at the index level (Figure 16). For the top 5 features, totalsun_hours (171.5 hours) emerged as a key contributor to class 1 predictions, followed by a high tmax_degC of 19.9°C. Month_sin (-0.867) notably reduced the likelihood of a wildfire prediction. For index 9 (predicting no wildfire), a tmax_degC of 5.3°C significantly lowered the prediction, while a high total rain value of 124.2mm from the previous month also decreased wildfire likelihood. Conversely, month_sin increased the likelihood of a wildfire.


Class 1 – Wildfire Prediction [Index 3]

Class 0 – No Wildfire Prediction [Index 9]

Figure 16: SHAP local feature importance force plots for index 3 and index 9.

5. Outlook

Overall, a sample size of only 537 months very likely impacted the predictive power of the models. As such, using wildfire and climate data across a daily temporal resolution would be more effective. Additionally, some uncertainty surrounding wildfire drivers and the impact of climate on wildfire occurrence means additional data features such as visitor numbers would likely improve model performance. Logistic regression interpretability indicated that some features have little impact on the classification output. Removing less influential features, like the 6- and 12-month tmax_degC_lag features, could help reduce noise in the model.

The use of TimeSeriesSplit limited the size of the training and validation sets in the earlier folds. However, this approach was suitable for this type of data. Moving forward, more advanced algorithms designed for time series data, such as ARIMA or LSTM, could further improve model performance[7].

References

[1] Albertson, K., Aylen, J., Cavan, G., McMorrow, J., 2010. Climate change and the future occurrence of moorland wildfires in the Peak District of the UK. Clim. Res. 45, 105–118. https://doi.org/10.3354/cr00926

[2] Met Office. Historical Climate Data. URL: https://www.metoffice.gov.uk/pub/data/weather/uk/climate/stationdata/sheffielddata.txt

[3] Albertson, K., Aylen, J., Cavan, G. And Mcmorrow, J., 2009. Forecasting the outbreak of moorland wildfires in the English Peak District. Journal of environmental management, 90(8), pp. 2642-2651.

[4] McMorrow, J., Aylen, J., Albertson, K., Cavan, G., Lindley, S., Handley, J., Karooni, R. (2006). Moorland Wildfires in the Peak District National Park, Peak District Case Study, Technical Report 3. Centre for Urban and Regional Ecology, Manchester University, Manchester. Available from: http://www.snw.org.uk/tourism/downloads/Moorland_Wildfires_Final_Report.pdf

[5] Millin-Chalabi, G., McMorrow, J., Agnew, C., 2014. Detecting a moorland wildfire scar in the Peak District, UK, using synthetic aperture radar from ERS-2 and Envisat ASAR. null 35, 54–69. https://doi.org/10.1080/01431161.2013.860658

[6] Rasgo (2024) Scikit-Learn Time Series Split. Model Selection. URL: https://www.rasgoml.com/feature-engineering-tutorials/scikit-learn-time-series-split

[7] Teki, Sundeep (2023) How to Choose the Best Model for Time Series Forecasting: ARIMA, Prophet, or mSSa. URL: https://www.ikigailabs.io/blog/how-to-choose-the-best-model-for-time-series-forecasting-arima-prophet-or-mssa

[8] Scikit-Learn Developers (2024) ‘TimeSeriesSplit’ under sklearn.model_selection. URL: https://github.com/scikit-learn/scikit-learn/blob/6cccd99ae/sklearn/model_selection/_split.py#L1091

[9] Stan (2021) How to do Time Series Split using Sklean. Medium. URL: https://medium.com/@Stan_DS/timeseries-split-with-sklearn-tips-8162c83612b9