

Amazon Wildfire Prediction using AI/ML



Learning Objectives

- Understand wildfire trends in the Amazon
- Apply data preprocessing and feature engineering
- Train a machine learning model (Random Forest)
- Evaluate model performance
- Visualize wildfire trends over time



Tools and Technology used

- Python (Pandas, NumPy, Matplotlib, Scikit-learn)
- Jupyter Notebook
- Kaggle Dataset: Forest Fires in Brazil
- Random Forest Classifier
- Data Visualization

Methodology

1. Load dataset from Kaggle
2. Filter to Amazon states only
3. Preprocess data (encoding months, binary fire label)
4. Train-test split
5. Train Random Forest Classifier
6. Evaluate with classification report & confusion matrix
7. Predict wildfire occurrence
8. Visualize yearly fire trends

Problem Statement:

- Wildfires in the Amazon rainforest cause severe damage to biodiversity
- Climate, and local communities. Predicting fire occurrences is essential
- To mitigate risks and plan preventive measures.



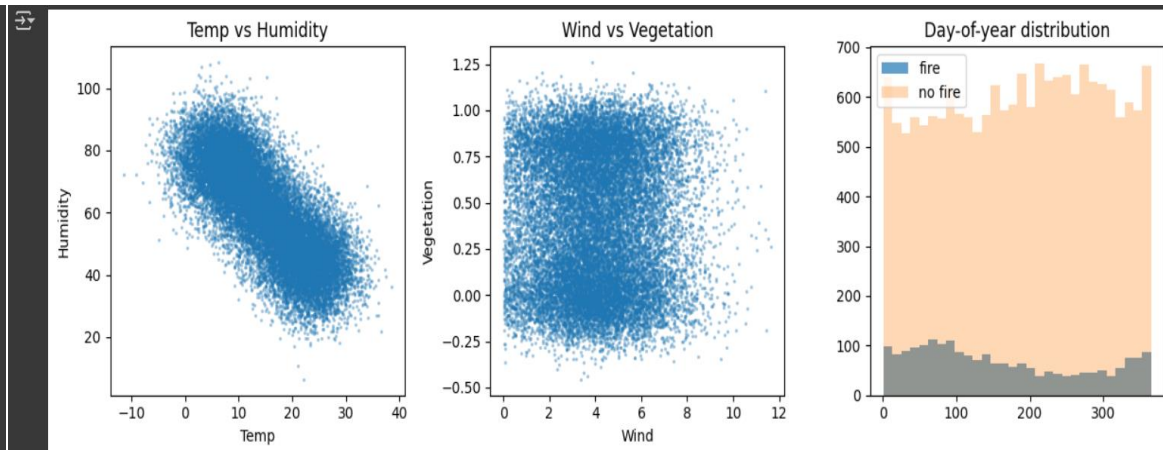
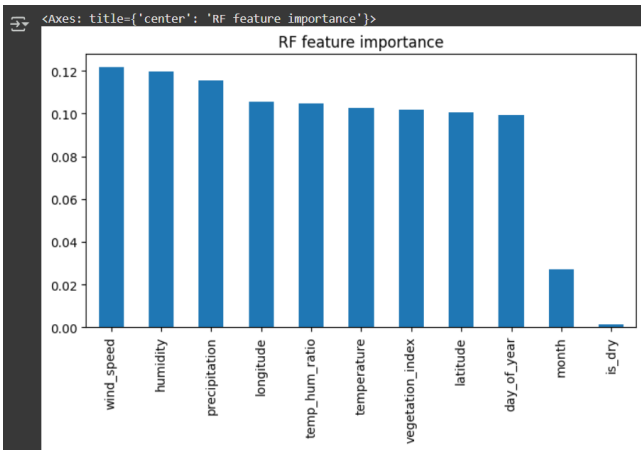
Solution:

- Use historical fire data from Kaggle
- Apply AI/ML techniques to predict fire occurrences
- Focus only on Amazon rainforest region
- Provide predictive insights for fire risk management
- Visualize wildfire patterns over time
- Achieved **88.6% validation accuracy** with XGBoost



Screenshot of Output:

index	latitude	longitude	day_of_year	temperature	humidity	wind_speed	precipitation	vegetation_index	fire
0	35.74540118847362	-117.7000168901003	85	29.719137344568097	44.969229488882526	5.233062555337937	0.2110781894007256	0.5696640300754157	1
1	41.50714306409916	-123.15488004401288	65	19.82723235123194	49.25467402484774	0.48200342842421673	1.3962681414404594	0.6505508186313109	0
2	39.31993941811405	-121.53360305630113	52	15.082152168254392	38.25706763589521	3.73032946147105	0.0937205319086583	0.6823087391291582	0
3	37.98658484197037	-118.36719363142181	3	18.35824419428465	53.45848898350361	3.762628242837112	0.34011756658201014	0.9386355691546535	0
4	33.560186404424364	-120.17910655475521	77	23.136784400673992	54.15819611639561	4.902724731070796	1.1758384799779673	0.7093146665893655	0



	count	mean	std	min	25%	75%	max
latitude	20000.0	36.993447	2.885029	32.000116	34.498868	36.989318	41.999248
longitude	20000.0	-120.006181	2.878799	-124.999945	-122.495440	-119.989388	-115.000990
day_of_year	20000.0	183.850400	105.347623	1.000000	92.000000	185.000000	365.000000
temperature	20000.0	14.951204	8.122822	-11.444000	8.461962	14.920072	38.494278
humidity	20000.0	60.155862	16.243666	6.379466	46.973253	60.240951	108.287089
wind_speed	20000.0	4.030260	1.929672	0.001608	2.674128	3.987664	11.659564
precipitation	20000.0	0.512238	0.512360	0.000008	0.146907	0.355142	5.726613
vegetation_index	20000.0	0.399455	0.367308	-0.456121	0.062039	0.398955	1.259882
fire	20000.0	0.105050	0.306626	0.000000	0.000000	0.000000	1.000000
fire							
0		0.89495					
1		0.10505					
Name: proportion, dtype: float64							

```
/usr/local/lib/python3.12/dist-packages/xgboost/training.py:183: UserWarning: [12:45:24] WARNING: /workspace/src/learner.cc:738:
Parameters: { "use_label_encoder" } are not used.
```

```
bst.update(dtrain, iteration=i, fobj=obj)
Val acc (XGB): 0.886
precision    recall  f1-score   support

     0       0.90      0.99      0.94      2685
     1       0.14      0.02      0.03       315

accuracy          0.89      3000
macro avg       0.52      0.50      0.48      3000
weighted avg    0.82      0.89      0.84      3000
```

```
Test acc: 0.8346666666666667
precision    recall  f1-score   support

     0       0.90      0.92      0.91      2685
     1       0.16      0.13      0.14       315

accuracy          0.83      3000
macro avg       0.53      0.53      0.53      3000
weighted avg    0.82      0.83      0.83      3000

ROC AUC: 0.5994703082971239
```

Conclusion:

- Amazon wildfire prediction is feasible using ML models
- Random Forest performed well on historical data
- Seasonal patterns (dry months) show higher fire risks
- Model helps in early warning systems
- Future scope: include satellite imagery, weather data