Capstone Project1: Identifying counties that are the most fire-prone and predicting the cause of a fire wildfire.

In our statistical analysis we saw what causes more fire across the states of the USA. Now let us find out if we can predict the cause of Fire in future given the location, month and year of fire reported. I did not consider 'fire size' as a feature because the goal is to predict the cause of wildfire at its initial phase where fire_size is small.

While selecting required features I dropped the ‘Discoved_Date’ column by keeping only month data. First I will convert all non numeric features like the cause of fire and state name’ to numeric values, this is necessary for machine learning using *preprocessing.LabelEncoder.fit_transform* method.

The goal is to predict the cause of the fire = STAT_CAUSE_DESCR. Since I'll be using supervised learning I created training and test datasets. I choose features as below and I have split the data in 30% for testing, 70% for training.

Features/Predicted Variable = FIRE_YEAR,STATE,LONGITUDE,LATITUDE,MONTH
Target Variable = STAT_CAUSE_DESCR

I have tried applying Decision Tree Algorithm and Gradient Boosting Decision Tree Algorithm to the data. The prediction score is approximately 49% and 17% respectively. I have used the Random Forest Algorithm here as I think that fits my dataset better than the Decision tree classifier. The prediction score of this model is approximately 57%, with Training data accuracy 74%. We can clearly see that this model is overfitting.

An overfit model may look impressive on the training set, but will be useless in a real application. Therefore, I will try to tune hyperparameter.The standard procedure for hyperparameter optimization accounts for overfitting through cross validation. Using Scikit-Learn’s RandomizedSearchCV method, we can define a grid of hyperparameter ranges, and randomly sample from the grid, performing K-Fold CV with each combination of values.

So far I have used a Random Forest classifier with the default parameters other than for n_estimator value which I have chosen to be 60. Higher n_estimator may yield better results but I am running into timeout/memory error with higher values.To use RandomizedSearchCV, I have created a parameter grid. I tried using a wide range of values but due to limited RAM access narrowed the range of values.

Using the below random grid parameters to search for best hyperparameters.

{'bootstrap': [True],
 'max_depth': [90],
 'max_features': ['auto', 'sqrt'],
 'min_samples_leaf': [2, 4],
 'min_samples_split': [5, 10],
 'n_estimators': [60]}

Created a Random Forest Classifier model and tried to tune it by passing it to RandomizedSearchCV method along with n_iter=100, which controls the number of different combinations to try, and cv=3 which is the number of folds to use for cross validation. More iterations will cover a wider search space and more cv folds reduces the chances of overfitting, but raising each has increased the run time and at times lead to memory error.

To determine if random search yielded a better model, I again created a RandomForest model with best parameters obtained using RandomizedSearchCV method.

{'bootstrap': True,
 'max_depth': 90,
 'max_features': 'sqrt',
 'min_samples_leaf': 4,
 'min_samples_split': 5,
 'n_estimators': 60}

We achieved an unspectacular improvement in accuracy of 1% after using bet parameters found using random search.

There are a lot of different Cause classes, I wanted to put some of these together and have just 3 classes for the cause of fires. Then test again to see if the score improves.

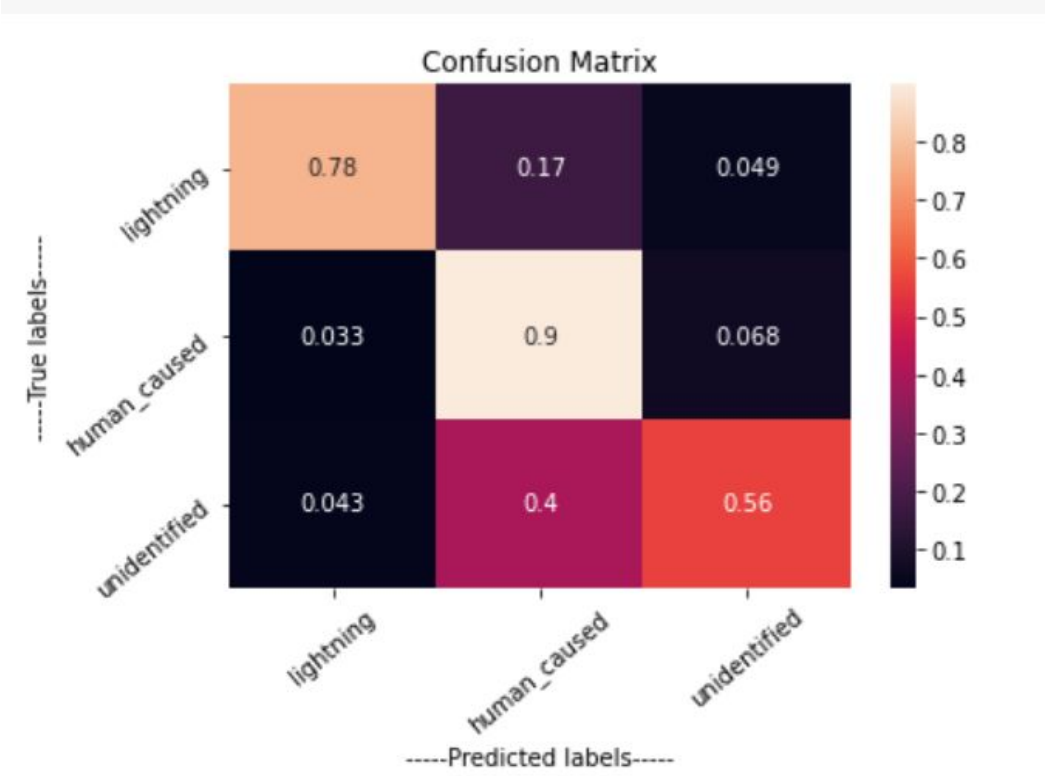
The 3 classes are: **lightning, human_caused and Unidentified/other**

where **lightning** = ['Lightning']

human_caused = ['Arson','Fireworks','Powerline','Railroad','Smoking','Children','Campfire','Equipment Use','Debris Burning','Structure']

Unidentified/other = ['Missing/Undefined','Miscellaneous']

I've now replaced STAT_CAUSE_DESCR with LABEL. So now I tried to predict LABEL.Also I will be passing best parameter values obtained by RandomSearchCV method.Reducing the number of categories improved the prediction score significantly (from around 58% to 80%). Prediction score is a good metric to measure success but we can look at other metrics like confusion matrix.



The random forest algorithm did well with the first two labels: lightning(78%) and human_caused (90%) but did not do as well with the 'unidentified' label. It labeled 40% of unidentified labels as human_caused fire.

Given that it is easier to make accurate predictions if the number of classes is reduced,I wanted to look at just one state at a time and build a system that can predict cause in that State. First I wanted to see How well the model predicts only for California.

The generic model accuracy(80%) was significantly more compared to model per state(66%). Thus I decided to not go ahead with this approach.

I wanted to check if taking a random sample of the dataset and applying a Random Forest Classifier Algorithm on this sample would yield a more efficient model. Thus I took 60% of the dataset randomly and built a classifier.Model using the subset of the dataset did not yield any better result. The accuracy is exactly the same as the model built using a full dataset.

Summary: we can predict the cause of these wildfires using the data provided, at least to an accuracy of 58% or better. Reducing the number of labels significantly improves the prediction score to 80% for the random forest algorithm.But with further tuning or a different algorithm it may be possible to reach a better score.

Below is the link to the Github repository of jupyter notebook files with Machine Learning code.

https://github.com/lasyabheemendra/Sprigboard-DatascienceProjects/blob/master/Capstone1_US-Wildfire-Prediction/US_Wildfire_ML_prediction.ipynb