Gabrielle Beinars
https://github.com/gbeinars

Forest Fires in the United States

Executive Summary

Background

The ability to predict what locations are more prone to fires, the most common causes of fires, or the size of a fire, can provide endless benefits. By understanding more about forest fires, preventative measures can be taken and possibly save lives, homes, forests and wildlife. Fires can be violently destructive and difficult to manage. Fires are known to destroy millions of acres of forest every year and can spread and grow quickly. Some causes of forest fires are controlled, accidental, intentional or natural. Three types of forest fires are surface, ground and crown.. Surface only impacts a small area of the forest floor and are easiest to stop. Ground fire, also referred to as underground fire, burns where there is dead and dry vegetation, spread slow, but difficult to put out because of its unpredictable pattern. Crown fires are serious, destructive and deadly.[4] This data will provide insight into forest fires with the purpose of developing a better understanding, and determine whether any factors of forest fires can be predicted, thus allowing for preventative measures to be taken.

Problem Statement

Analyzing data collected on forest wildfires can serve a variety of purposes. For example, if the cause of fires could be predicted, then preventative measures can be put in place. This could save thousands of lives, the forests, and the wildlife. Predictions of fire size can also allow for prior action to be taken.

- Are wildfires increasing over the years?
- Do certain geographic areas have more fires than others?
- Is it possible to predict the size of a fire based on certain features?
- What is the most common cause of forest fires?

Gabrielle Beinars
https://github.com/gbeinars

Scope

This project involves various exploratory analysis techniques and predictive modeling, to gain a better understanding about forest fires, and predict the fire size and location.

Method

Data Source

The dataset that will be analyzed for this project can be found at https://www.kaggle.com/rtatman/188-million-us-wildfire. This dataset includes forest fires in the US from 1992 to 2015. There are 1.88 million records, with 130 columns, where each observation is a different fire, and the total of all fire sizes span 140 million acres. The primary features to be analyzed are the discovery date, final fire size, and location (precise within 1 square mile). The data has already undergone basic error checks and duplicate records were removed. [1][6]

The publication of the data was in support of the national Fire Program Analysis (FPA) system where the records were originally collected from systems of federal, state and local level.[2] The FPA is meant to replace older systems including National Fire Management Analysis System (NFMAS), FirePro and FireBase. The FPA supports the wildland fire program and is considered important for the Forest Service and the Department of the Interior.[3]

Data Import/Cleaning

The data set was imported using Python's sqlite3 library as the data is stored in an SQL database. The table was loaded to a pandas DataFrame where it was further analyzed. Cleaning the data involved checking for missing values and removing any features necessary. Three variables involved dates that which the fire was declared contained or controlled were broken up into day of year, full date, and time. These were removed due to an abundance of missing observations, and multicollinearity displayed with the discovery variables (Figure 1), including date and day of the year the variable was discovered or confirmed. Multicollinearity can be observed with various variables that translate to similar results. For example, fire size and fire size class both have the same meaning but categorized differently, therefore one of these needed to be removed before modeling.

Gabrielle Beinars
https://github.com/gbeinars

The following variables were included in a subset of the original DataFrame to be used for further analysis: fire year, stat cause code, fire size class, latitude and longitude. A correlation heat map of the DataFrame subset is shown in Figure 2. The only variable with missing values was county, however, this is not a concern because we can look at location through the geographical coordinates.

Figure 1: Correlation Heat Map (before removing variables)

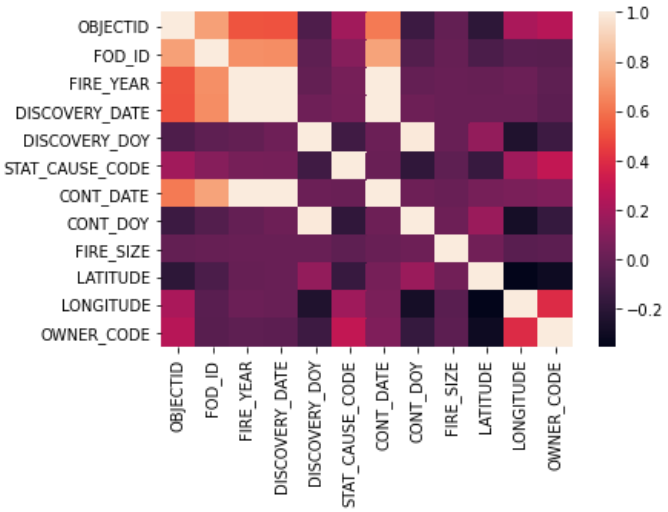
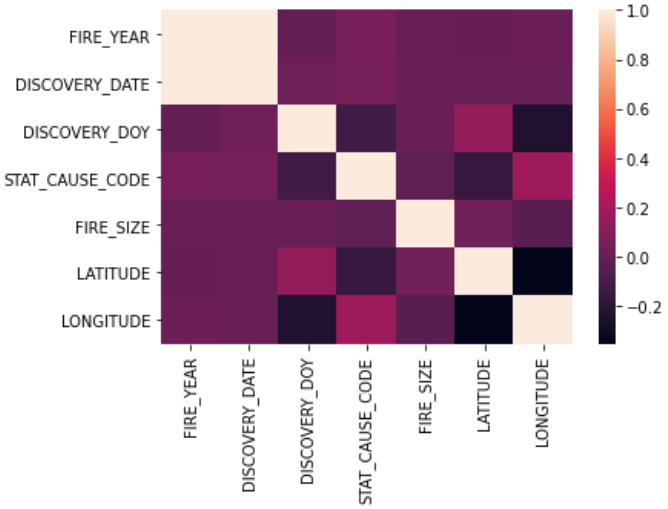


Figure 2: Correlation Heat Map (before removing variables)



Gabrielle Beinars
https://github.com/gbeinars

Exploratory Data Analysis (EDA)

After the data was cleaned, exploratory analysis began, where various visuals were created to gain a better understanding about location, fire size, and cause of forest fires. The top six causes of forest fires from 1992 to 2015 were lighting, miscellaneous, missing/undefined, arson, equipment use and debris burning (Figure 3). With missing/undefined and miscellaneous being the second and third top categories, it raises the question, what more can be done to gain further insight into these categories?

Figure 3: Horizontal Bar Chart of Fire Count by Cause

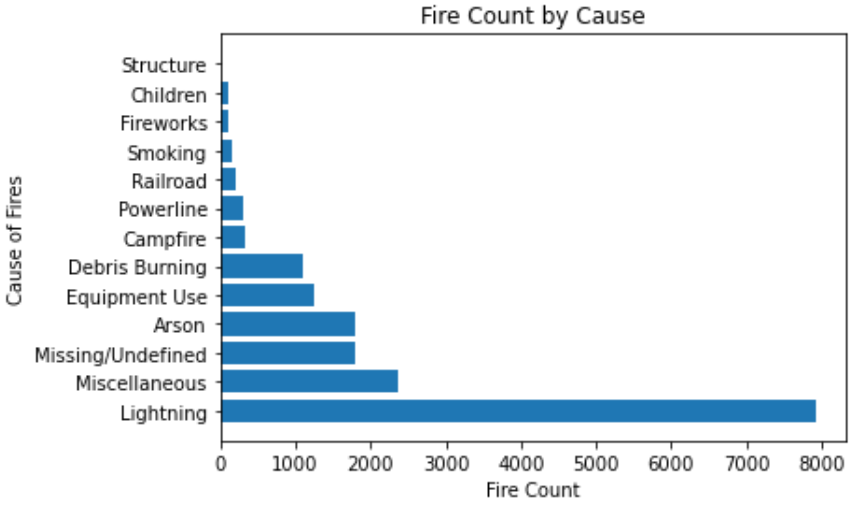
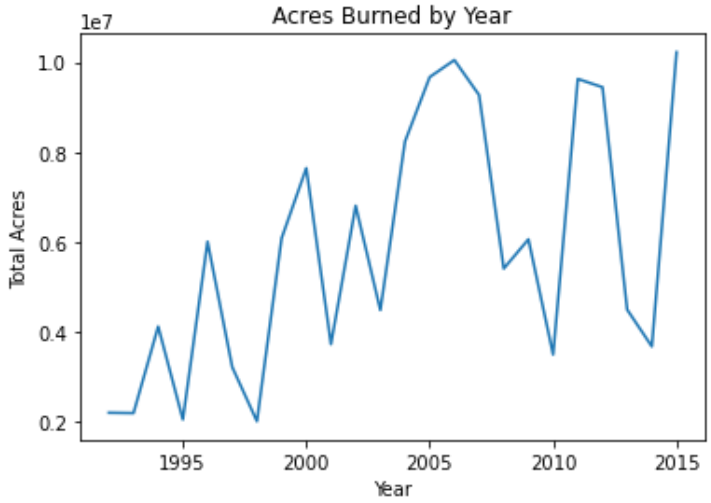


Figure 4 shows a line chart of total acres burned by year. We can see an overall increase of acres burned over time, but a drop observed around 2010 and 2014. Additional insight into current events could further explain the great increases 2006, 2012, and 2015, and the drops observed around 2010 and 2014.

Figure 4: Line Chart of Total Acres Burned by Year



Interestingly, a different pattern is observed for total fires by year (Figure 5), compared to total acres burned by year. An increase in both can be observed around 2006, implying that possibly more acres were burned due to more fires at this time, with more frequent larger fires occurring in years where we have a decrease in count, but increase in acres burned.

Figure 5: Line Chart of Total Fires per Year

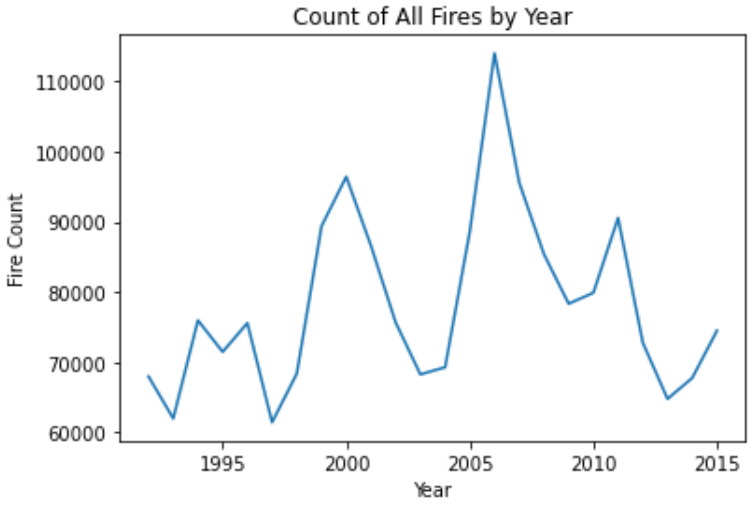
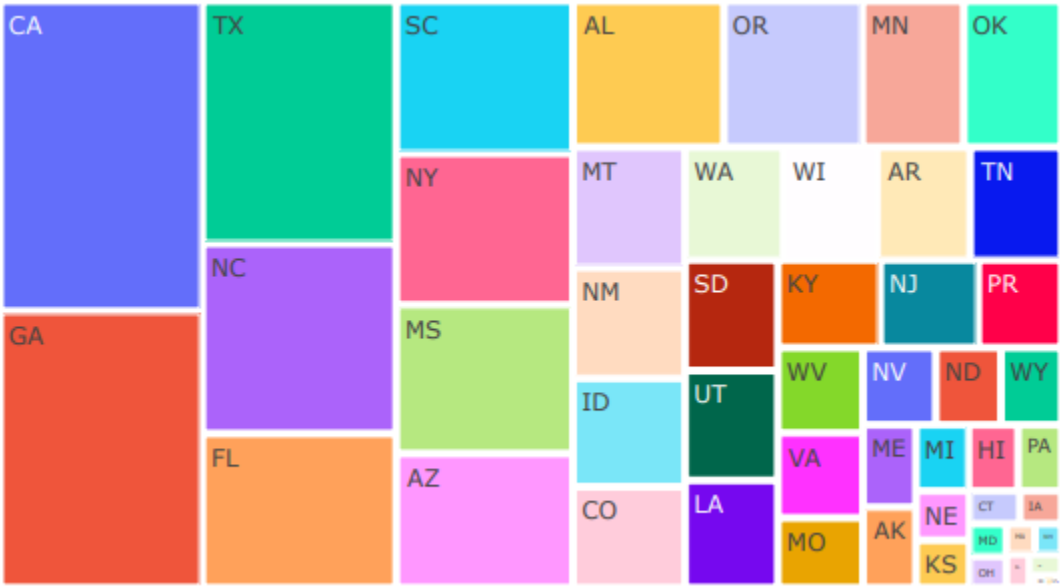


Figure 6 shows a tree map of fire count by state. It can be observed that California, Georgia, Texas, North Carolina and Florida had the most wildfires when compared with other states.

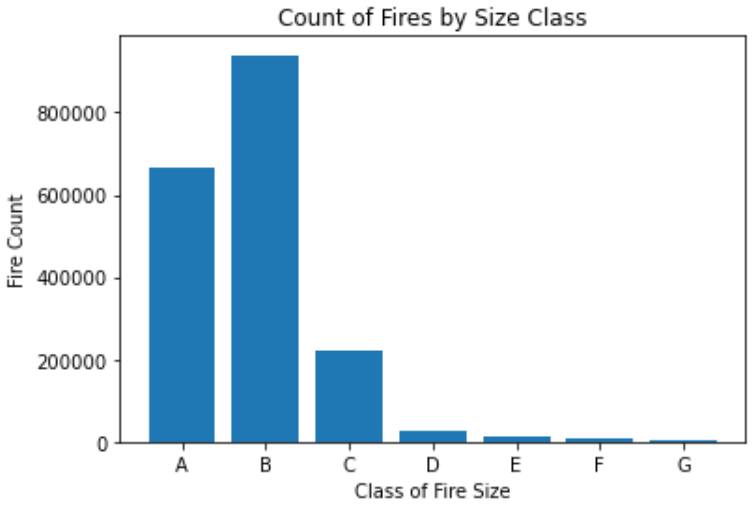
Figure 6: Tree Map of Fire Count by State



Modeling

The approach for modeling utilized various classifiers from the scikit learn package, including random forest, k-nearest neighbor (KNN), and gradient boosting. The classes used for modeling are the seven fire size classes, categorized as A through G, which correspond to sizes, 0 to 0.25 acres, 0.26 to 9.9 acres, 10 to 99.9 acres, 100 to 299 acres, 300 to 999 acres, 1000 to 4999 acres, and 5000+ acres (Figure 7). The features used for modeling are location (latitude and longitude), year and cause.

Figure 7: Bar Chart



A random forest classifier is also referred to as random decision forests, and are an ensemble learning method for classification problems. It works such that the output is the class selected by the most trees.[7] Random forest classifier is also an example of an ensemble of bagged decision trees.[11] The purpose of decision trees is to predict the value of a target variable by learning decision rules inferred from the other features. A decision tree classifier was used as it can handle multi-output problems, allows for validation, and will still perform well when assumptions are violated. [5] It is capable of handling large datasets with high dimensionality, enhances accuracy and prevents overfitting.[8] It is also important that the classes used as target variable are balanced to avoid biased.[5] While there are many advantages to using random forest, including that it can be used with classification and regression, it is known to perform better with regression.[8]

The KNN algorithm is a supervised machine learning algorithm that can be also be used for regression or classification problems. Supervised machine learning algorithms rely on the inputted data to learn a function that results in an appropriate output when introduced to unfamiliar data. For this project, it was used for classification as the output is categorical.[9]

Gradient boosting is an ensemble method that is specifically a boosting technique, meaning it combines several weak learners, where each predictor tries to improve the prior prediction by reducing the errors, resulting in improved accuracy to individual models. Rather than fitting a

predictor to the data at each iteration, it fits to the residual errors of the previous prediction, thus reducing error. With each new decision tree created, it predicts the residual that was calculated in the previous tree.[11]

Results

Figure 8 shows a confusion matrix of the results using the random forest classifier. It can be observed that classes A and B were more accurately predicted than the other classes, and this could be due to an imbalance in the fire class sizes. When creating the model, an argument referred to as 'stratify' was utilized when the data was split into test and train to handle the imbalanced target variable. The classification report results are included in the appendix, where precision, recall and F1 score are listed. These are metrics that evaluate the accuracy of the model and shows the number of times a model predicts correctly or incorrectly, by categorizing as true positive, true negative, false positive or false negative. Precision looks at the ratio of true positives compared to all positives, while recall measures the accuracy of predicting true positives. These metrics were evaluated for the random forest classifier to confirm that the classes are imbalanced, as accuracy can be ineffective for imbalanced classes. The F1 score looks at both precision and recall and represents the balance of the two metrics. The precision, recall and F1 score are broken down by class, and the results indicate highest precision and recall for fire size classes, A and B. All metrics decrease as the class gets smaller, indicating imbalance in the dataset.[14]

Figure 8: Confusion Matrix Results for Random Forest Model

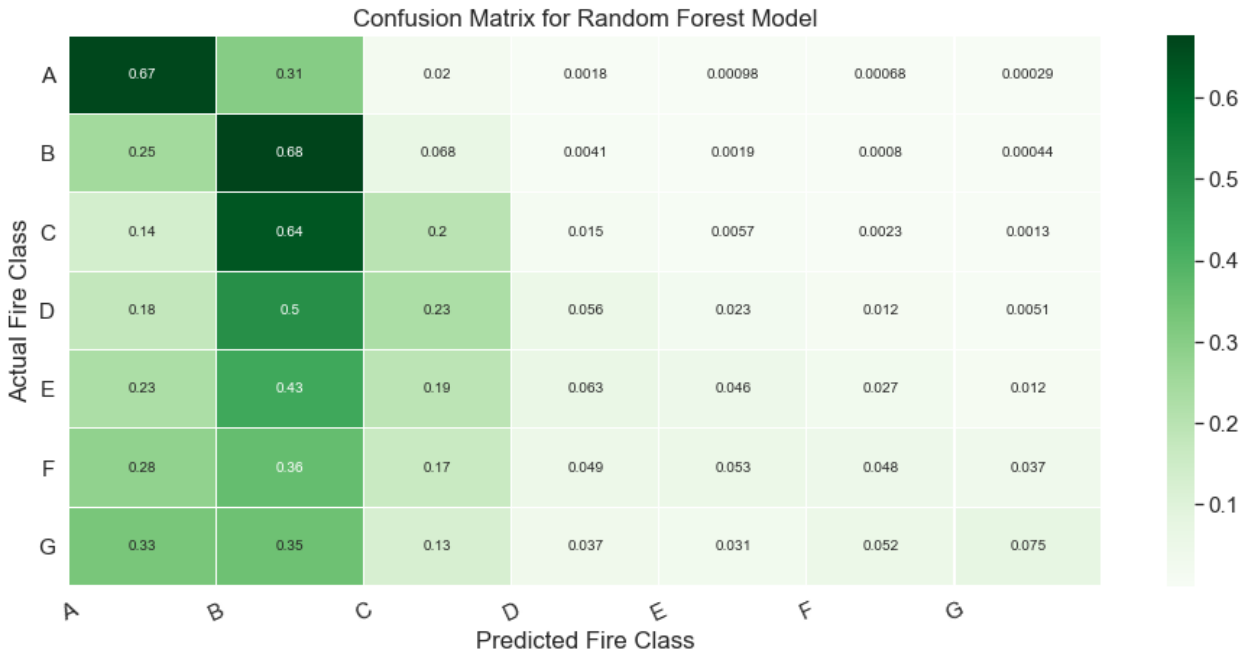


Figure 9 shows the results of KNN modeling, and how the k-value was selected. This visual was recreated for each k-value explored.

Figure 9: Visual of Training and Testing Accuracy by k-value for KNN

Table 1: Accuracy Results

Model	Accuracy
Random Forest	0.60
KNN	0.62
Gradient Boosting	0.61

As mentioned prior, bias can have an impact on random forest classification as an assumption is that the target classes are balanced. Gradient boosting is known to help reduce noise, variance and bias factors observed when using a single model.[12]

Discussion

Assumptions

Various assumptions were made for modeling and depend on the model being used. For random forest classifiers, there are two assumptions that should be met; actual values used in the feature variable such that the classifier can predict accurate results, and the predictions from each tree should have low correlations. With KNN, it is assumed that similar data points are close enough to each other and requires an appropriate k-value be used. This is determined by running several different k-values, and selecting the k that reduces error while maintaining accuracy. A disadvantage of the KNN algorithm is that it will run slower as k increases, and it does need to be ran multiple times to determine the appropriate k value.[9] It is considered non-parametric meaning that there is no assumption to the distribution of the data. [10]

As gradient boosting is an ensemble method, it should adhere to the assumptions of ensembles. No assumptions need to be made as it relies on sampling being representative. As it combines weak models, there are no separate assumptions. [13]

Limitations/Challenges

It is important to be aware when using decision trees that they can be overfitted, and pruning may need to be used to avoid this problem.[5]

Gabrielle Beinars
https://github.com/gbeinars

A challenge encountered during this project was creating a spatial map of the data with python, due to the amount of data. Various libraries for creating maps were explored, including geopandas, Folium and plotly.express, but due to the amount of data, these maps would not load.

Next Steps

Gaining more insight into the fire cause descriptions of miscellaneous and missing/undefined, could provide additional benefits and better the understanding of forest fires. Additionally, there are a variety of factors that can influence forest fires that were not evaluated as part of this project, such as weather and geographic descriptions.

Gabrielle Beinars
https://github.com/gbeinars

References

[1] *1.88 million US wildfires*. Kaggle. (n.d.). Retrieved October 16, 2021, from https://www.kaggle.com/rtatman/188-million-us-wildfires.

[2] Short, K. C. (n.d.). *Spatial wildfire occurrence data for the United States, 1992-2018 [fpa_fod_20210617] (5th edition)*. Home. Retrieved October 16, 2021, from https://www.fs.usda.gov/rds/archive/catalog/RDS-2013-0009.5.

[3] *Fire program analysis application*. Bureau of Land Management. (n.d.). Retrieved October 16, 2021, from https://www.blm.gov/policy/ib-2009-040.

[4] Rodgers, B. (2021, August 18). *Understanding forest fire safety, preparation and survival*. Prepper's Will. Retrieved October 16, 2021, from https://prepperswill.com/understanding-forest-fire-safety-preparation-and-survival/.

[5] *1.10. decision trees*. scikit. (n.d.). Retrieved October 16, 2021, from https://scikit-learn.org/stable/modules/tree.html.

[6] Short, Karen C. (2017). *Spatial wildfire occurrence data for the United States, 1992-2015 [FPAFOD20170508] (4th Edition)*. Fort Collins, CO: Forest Service Research Data Archive. https://doi.org/10.2737/RDS-2013-0009.4

[7] Wikimedia Foundation. (2021, September 13). *Random Forest*. Wikipedia. Retrieved October 23, 2021, from https://en.wikipedia.org/wiki/Random_forest.

[8] *Machine learning random forest algorithm - javatpoint*. www.javatpoint.com. (n.d.). Retrieved October 23, 2021, from https://www.javatpoint.com/machine-learning-random-forest-algorithm.

[9] Harrison, O. (2019, July 14). *Machine learning basics with the K-nearest neighbors algorithm*. Medium. Retrieved October 23, 2021, from https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761.

[10] N, S. D. (2020, August 7). *K-nearest neighbors algorithm*. Medium. Retrieved October 23, 2021, from https://medium.com/analytics-vidhya/k-nearest-neighbors-algorithm-7952234c69a4.

[11] Aliyev, V. (2020, October 7). *Gradient boosting classification explained through python*. Medium. Retrieved October 23, 2021, from https://towardsdatascience.com/gradient-boosting-classification-explained-through-python-60cc980eeb3d.

[12] Grover, P. (2019, August 1). *Gradient boosting from scratch*. Medium. Retrieved October 23, 2021, from https://blog.mlreview.com/gradient-boosting-from-scratch-1e317ae4587d.

[13] Chatterjee, D. R. (2019, August 26). *All the annoying assumptions*. Medium. Retrieved October 23, 2021, from https://towardsdatascience.com/all-the-annoying-assumptions-31b55df246c3.

[14] *Precision, recall and F1 explained (in plain English)*. DataGroomr.com. (2021, May 24). Retrieved October 23, 2021, from https://datagroomr.com/precision-recall-and-f1-explained-in-plain-english/#:~:text=Precision%20and%20recall%20%28and%20F1%20score%20as%20well%29,where%20the%20model%20correctly%20predicts%20the%20positive%20class.

Appendix

Below is a snip-it of the classification report results using the random forest classifier.

	precision	recall	f1-score	support
A	0.62	0.67	0.64	166730
B	0.63	0.68	0.65	234844
C	0.33	0.20	0.25	55019
D	0.14	0.06	0.08	7107
E	0.12	0.05	0.07	3527
F	0.12	0.05	0.07	1947
G	0.16	0.08	0.10	943
accuracy			0.60	470117
macro avg	0.30	0.25	0.27	470117
weighted avg	0.58	0.60	0.59	470117