1. Set the working directory as the folder containing this README file.
2. Inputs for all files listed in this README can be found in the “data_source” folder.
3. Data Wrangling - in “data_wrangling” folder

    To prepare the data for modeling, first run “batch_census_geolocate.sh” to geolocate each incident in the data set provided to us by Intterra to a census tract. Then, run “geoidmatch.R” to match the census tracts with demographic data from the American Community Survey. Next, run “Merging everything.ipynb” to merge all the demographic data with the incident data. Then, run “Aggregating and Renaming ACS Columns.ipynb” to obtain more meaningful names for the demographic variables. Then, run “cleanSVI.R” to clean the Social Vulnerability Index (SVI) data from the CDC. Finally, run “generateFinalDataset.ipynb” to complete the data wrangling phase and obtain the final dataset for modeling.

    - batch_census_geolocate.sh
        - Note: this file takes over 48 hours to run
        - **Input**: incidents_for_d2k.csv
        - **Does**: pulls geoid codes for incidents from Intterra data that had it missing
        - **Output**: stacked_geolocated_complete.csv
    - geoidmatch.R
        - Note: this file takes quite some time to run
        - **Input**: stacked_geolocated_complete.csv, nhgis0008_ds239_20185_2018_tract.csv, incidents_for_d2k.csv
        - **Does**: matches geolocated incidents with ACS demographic data on census tract
        - **Output**: temp_census_matched.csv, merged_part_3_final_pls.csv
    - Merging everything.ipynb
        - Note: this file takes quite some time to run
        - **Input**: temp_census_matched.csv, merged_part_3_final_pls.csv, merge_this.csv, merge_this_2.csv
        - **Does**: merges the outputs of geoidmatch.R (“temp_census_matched.csv” and “merged_part_3_final_pls.csv”) and adds additional ACS demographic columns
        - **Output**: complete_everything.csv
    - Aggregating and Renaming ACS Columns.ipynb
        - **Input**: complete_everything.csv
        - **Does**: aggregates overly specific demographic columns, renames all demographic columns
        - **Output**: clean_columns_everything.csv
    - cleanSVI.R
        - **Input**: SVI2018_US.csv
        - **Does**: Cleans the SVI Excel sheet for merging into dataset
        - **Output**: svi_edited.csv
    - generateFinalDataset.ipynb
        - **Input**: clean_columns_everything.csv, svi_edited.csv

        - **Does**: cleans data, aggregates rows (over census tract), merges ACS and SVI data, splits data into training and test, normalizes feature distributions, calculates target response variables, drops census tracts with troublesome coverage
        - **Output**: "trainTractsDroppedStandard.csv", "testTractsDroppedStandard.csv"
4. Exploratory Data Analysis - in "data_exp" folder

    To create plots displaying the data distribution through different dimensions and look at how the incidents were distributed temporally, spatially, and numerically, run "exploratory_plots.R".

    - exploratory_plots.R
        - **Input**: various files in "data_source" folder
        - **Does**: creates plots used for exploratory data analysis. One plot shows the daily count of incidents, separated by NFIRS group. Another plot shows the proportion of incidents by NFIRS group each day.
        - **Output**: None
5. Final Modeling - in "data_modeling" folder

    To build the Bootstrap Lasso model and view the resulting feature importance bar graphs, run "Bootstrap Lasso and ANOVA.R". To build the XGBoost model for total incidents per capita, run "xGBoostTotalIncidents.ipynb". To build the XGBoost model for medical incidents per capita, run "xGBoostMedicalIncidents.ipynb". To build the XGBoost model for fire incidents per capita, run "xGBoostFireIncidents.ipynb". All files in this folder generate various tables, graphs, and results that were formatted and inserted into the final report.

    - bootstrap lasso and anova.R
        - **Input**: trainTractsDroppedStandard.csv, testTractsDroppedStandard.csv
        - **Does**: Bootstrap lasso, ANOVA analysis and modeling, as well as various model outputs that are detailed in the report
        - **Output**: BARPLOT_inc.csv, BARPLOT_medinc.csv, BARPLOT_fireinc.csv. These are in the "table_outputs" folder. Tableau workbook that creates the bootstrap barplots is also included.
    - xGBoostTotalIncidents.ipynb, xGBoostMedicalIncidents.ipynb, and xGBoostFireIncidents.ipynb
        - **Input**: trainTractsDroppedStandard.csv, testTractsDroppedStandard.csv
        - **Does**: Trains an XGBoost regression model to predict total, medical, and fire incidents per capita, respectively. Plots figures showing the top features found to be important by each trained model.
        - **Output**: None

Packages Used & Versions

R Packages

Name	Version
dplyr	1.0.2
tidyr	1.1.2
glmnet	4.0.2
selectiveInference	1.2.5
rpart	4.1.15
rpart.plot	3.0.9
sjPlot	2.8.3
sjmisc	2.8.4
sjlabelled	1.1.4
sp	1.4.1
rgeos	0.5.2
spdep	1.1.3
rgdal	1.4.8
data.table	1.13.2
RColorBrewer	1.1.2
mclust	5.4.5
viridis	0.5.1
Rtsne	0.15
Rfast	2.0.1
plotmo	3.5.6
car	3.0.6

Python Packages

Name	Version
pandas	1.1.3
numpy	1.19.2
matplotlib	3.3.2
seaborn	0.11.0
scikit-learn	0.23.2
scipy	1.5.3
scikit-optimize	0.8.1
xgboost	1.2.1
shap	0.36.0

System Dependencies

Name	Version
Bash	5.0 (or better)
Rscript*	4.0
Python	3.8.6
R	3.6.3

*Installed by default with R, but must be added to path if on Windows

corrplot	0.84
leaps	3.1
pacman	0.5.1

Appendix:

These are files that were either used in exploratory/experimentation stages but not in our final pipeline or were lost due to unexpected termination of our AWS instance.

- partition.R (in “data_split” folder)
    - Note: This file was not used for our final model. Instead, we used the output from “generateFinalDataset.ipynb”
    - **Input**: clean_columns_everything.csv
    - **Does**: partitions the cleaned incident + census data into 80-20 training-test split
    - **Output**: train_complete.csv, test_complete.csv, train_tracts.txt, test.txt
- Variable Correlation, PCA, Exploratory Linear Models.ipynb
    - Note: this file is missing
    - **Input**: trainTracts.csv
    - **Does**: investigates correlation between variables, PCA, initial linear regression
    - **Output**: None
- Coverage Exploration and Identification of Problem Tracts
    - Note: this file is missing
    - **Input**: trainTractsStandard.csv
    - **Does**: conducts exploration by plotting the log(incidents per capita) against the number of days between the first and last incidents in a tract. Also plots the date of the last incident against the date of the first incident. Isolates the anomalous tracts into 3 trouble groups.
    - **Output**: None
- Forward and Backward Stepwise Feature Selection.ipynb
    - Note: this file is missing
    - **Input**: trainTractsCorrected.csv and testTractsCorrected.csv
    - **Does**: feature selection using forward and backward stepwise selection
    - **Output**: None
- Lasso Solution Path.ipynb
    - Note: this file is missing
    - **Input**: trainTractsStandard.csv
    - **Does**: finds lasso solution path (order in which variables enter the lasso model as lambda is gradually decreased)
    - **Output**: None