Using Google Trends to Decipher Dark Figures of Crime

Nicholas Moss (upload): nm3318
Sean Liu: yhl494
Joh Euh Kang: jk5726

Introduction and Motivation

A pervasive issue facing criminal justice researchers is accurate measurements of particular acts of crime. Assault and murder, or life-changing burglary (ex. car theft), are incidences that are almost certain to be reported to police. However, there are other acts which are less likely to be reported, such as rape or theft of household items. When these incidences are not reported as frequently, we cannot ascertain a good understanding of their true occurrence rates, hence their labeling as "dark figures of crime".

In the past, researchers have relied on phone surveys of random households to get a better estimate of dark crime. Nowadays, as people are less likely to talk to strangers on the phone, this surveying method risks becoming biased and obsolete. Instead, victims of these crimes are more likely to consult the internet for help. It is reasonable that a rape victim may, for instance, conduct a google search: "my [ex-boyfriend] raped me". Or someone who had their bike stolen: "my bike was stolen". In either of these cases, the individual would not report these incidences to the police.

We are interested in evaluating the usefulness of extracting meaning from these google searches, that could correlate with the unmeasured rates of certain criminal activities. If there is high correlation, Google search data could serve to predict certain "hotbeds" of crime and thus highlight communities that may require additional governmental or other resources.

Methodology

We have written a program that extracts data from Google Trends (GT) for given keywords or strings, and outputs the results to a csv file. The program asks the user for keywords, time interval, and geographic location, then return the frequency "weights" of those search terms in that particular area. We use a package called pytrends, which allows us interact with GT search data in Python. Additionally, we use the datetime package to process the user input dates into a range that can be used by pytrends. The results of GT data are then written to a csv file.

To evaluate the usefulness of these searches for certain criminal acts, we compared the GT search frequencies to official and unofficial crime measurements. The National Crime Victimization Survey (NCVS) conducts random survey of individual households to try to estimate crime rates for rape, burglary, and theft. We used a dataset of their results from these surveys from 2010 - 2012, the most recent years for which they provide data. We also used a dataset from the Uniform Crime Reporting (UCR) group, which provides official statistics regarding criminal acts reported to the police. Here, we used datasets from years 2012 – 2017. These are accessible through their respective websites.

We used pandas to normalize these datasets and group the crime figures by metro area, which is the commonly used denominator when parsing data from GT. Specifically, we trimmed the datasets to compare three crime categories, namely: motor vehicle theft, burglary, and

rape. We used groupby to merge these datasets (on their metro area columns) to achieve one dataset which has statistics from GT, NCVS, and UCR together, indexed by metro area.

Results

We have written a program called "GTKeyword" that successfully retrieves google search data for user inputted keywords, sorted by geographic location and time period. The program is fairly user friendly in that it specifies the exact format that the user needs to input conditions for proper retrieval of GT data. However, it currently does return custom error messages. Thus, if the exact format is not followed, the errors may be slightly ambiguous. A next step would be to add exceptions and appropriate messages for all possible user input errors.

The program currently only takes up to five keywords (supports regular expressions) for which to retrieve GT search data. It uses five if/elif statements to determine the number of keywords inputted by the user. Each of these statements is followed by similar but distinct functions to process the keywords. Ideally, we would be able to process any number of keywords under the same array of functions, and thus would accommodate more than five keywords. For our application of estimating the crime figures, we did not find it necessary to input more five terms to gather all the appropriate permutations of user searches.

In the application of the program, we inputted terms that victims would be most likely to search if they were subject to either rape, burglary, or vehicle theft. We gathered these search queries for the 50 largest metro areas in the US and compared them to publicly available datasets featuring crime statistics for those same metro areas. We used functions in pandas to group and convert data from county to metro area, and to normalize figures by population.

Finally, we used matplotlib to create five scatterplots which visualize the correlation between GT searches (from potential victims) and the official (UCR) and unofficial (NCVS) crime statistics for rape, burglary, and vehicle theft in US metro areas. These include regression lines and confidence intervals for regressing UCR and NCVS on GT data. According to the visualizations, the our GTKeyword program appears to be strongest for estimating burglary rates, and weakest for estimating rape crime.

An interesting last step would be to determine r-squared values for all regressions to actually measure the strength of GT in modeling criminal activity. If there are any transgressions which are significantly correlated with GT, then search data could serve as a valuable predictor for communities which may necessitate additional social programs or resources.