Analyzing the Effects of Anonymization Techniques on Health Data

Natalia Valencia
Erick Gonzalez-Vega
Sharon Beulah Perumandla
Ananya Kaza

CAP-5768

Abstract

This paper explores how anonymizing data to preserve privacy affects the data's integrity. For any analysis of anonymized data to provide insightful interpretations, the anonymized data must remain as close to the original as possible. For the sake of having the modified data as close to the original, many organizations end up de-identifying the data instead of anonymizing it. De-identified data removes, generalizes, or replaces all explicit identifiers such as, SSNs, names, addresses, telephone numbers, etc.. Even if identifiers have been removed, replaced, or generalized, an adversary can still use background knowledge and cross-correlation with other datasets to re-identify individual data records. Anonymized data on the other hand, refers to data that can't be manipulated or linked to identify an individual. This paper demonstrates that out of various anonymization techniques, resampling proves to be the most convenient technique when it comes to being able to analyze merged datasets. With other forms of anonymization, one is limited to using the same dataset that's being anonymized. In cases where one dataset is not enough to draw insightful conclusions from, merging two or more datasets can make the data more robust, reducing the chances of coming up with one dimensional conclusions. This paper introduces multiple tests in which resampling of explicit and quasi-identifiers is done in order to analyze the data's integrity .

Introduction

Personal health identifiable information is any information concerning a person's health or treatment that is used to identify them. The HIPAA (Health Insurance Portability and Accountability Act) Privacy Rule provides federal protections for personal health information held by entities covered by HIPAA and gives patients a list of rights which they can enforce. The Privacy Rule also allows for the dissemination of personal health information needed for patient care and other purposes. Organizations release and receive medical data with all explicit identifiers removed, replaced, or generalized on the assumption that the confidentiality of a patient is maintained because the resulting data looks anonymous. However, the remaining data can be used in conjunction with other datasets to re-identify individuals. Direct identifiers explicitly identify a person. HIPAA considers these 18 identifiers as direct/explicit identifiers: (1) names, (2) addresses (all geographic subdivisions smaller than state, including street address, city county, and zip code), (3) all elements (except years) of dates related to an individual (including birthdate, admission date, discharge date, date of death, and exact age if over 89), (4) telephone numbers, (5) fax numbers, (6) email addresses, (7) social security numbers, (8) medical record numbers, (9) health plan beneficiary numbers, (10) account numbers, (11) certificate or license numbers, (12) vehicle identifiers and serial numbers, including license plate numbers, (13) device identifiers and serial numbers, (14) web URLs, (15) Internet Protocol (IP) addresses, (16) finger or voice prints, (17) photographic images – not limited to images of the face, (18) any other characteristic that can uniquely identify the individual.

Indirect, or quasi-identifiers, refer to information that can be combined with additional data to identify an individual. These include: gender, age, race, ethnicity, etc.. Many data releases only de-identify the former, while providing the latter in clear text.

Confidentiality of personal identifiable information protects an organization's interest, while privacy protects an individual's autonomy. In medical usage, both confidentiality and privacy mean the same thing, they are interchangeable. In order to preserve privacy, the data can be modified in numerous ways, depending on the availability of the data. In other words, how the data owners are releasing the data. Is the data being made fully available to the public modified? Was noise added to the data? (1)

Noise addition works by adding or multiplying a stochastic or randomized number to confidential quantitative attributes. Were masking techniques applied? (2) Masking replaces data elements with similar-looking proxy data using characters that preserve the format requirements, (i.e., replacing parts of an email address with *** to preserve anonymity). Or k-anonymity? (3) K-anonymity combines sets of data with similar attributes, making values indistinguishable from each other. Or was the data removed? Or replaced with synthetic data? (4) Feature removal is the easiest way to protect data integrity, although that does not guarantee the protection of privacy. (5) Synthetic data is artificially generated data that mimics the structure and statistical properties of the data gathered. If the data remains under the control of the owner, then perturbations can be added. (6) Perturbations falsify the data by introducing an element of error in order to retain confidentiality.

There are many ways to anonymize data, each technique having its pros and cons. Choosing the appropriate technique or techniques depends on the types of data (health, finance, economic, geospatial, etc.) being used, whether the data has been de-identified, or if it's in its raw form, etc. There's a lot of literature out there that talks about de-identifying raw data, as if that were the only solution to guaranteeing privacy and confidentiality. This paper provides an example to demonstrate that this is not the case. This example is used as a starting point for investigating how to better protect de-identified data from being re-identified. Different anonymization techniques are sampled in order to best assess their performance before selecting the one that best protects anonymity while preserving as much of the original data as possible. From there, a thorough analysis is conducted on this particular technique, comparing and contrasting the before and after results on identifiable information.

An appendix is included at the end of the report which contains supplementary material that provides the experiments used for preliminary analysis.

Approaches

Tools

The programming language used was Python. The experiments were done in Python version 3.9.7. The libraries used to achieve the results were: (1) NumPy[1], (2) Pandas[2], (3) SciPy[3], (4) Statsmodels[4], (5) Matplotlib[5], (6) seaborn[6], (7) anonympy[7], (8) association_metrics[8], (9) scikit-learn[9], and (10) prince[10]. The Pandas library is used for data manipulation and analysis. NumPy is used for working with arrays and matrices. SciPy provides utility functions for optimization, integration, interpolation, eigenvalue problems, algebraic equations, differential equations, statistics, etc. . Statsmodels allows users to explore data, estimate statistical models, and perform statistical tests. Matplotlib is a comprehensive library used for creating static, animated, and interactive visualizations. Seaborn is a data visualization tool based on Matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphs. Association metrics is a module used to measure the degree of association between variables built on top of SciPy and NumPy. GeoPandas is an open-source project used to make working with geospatial data in Python easier. Scikit-learn (or sklearn) is a free software machine learning library for Python. Prince is a library for doing factor analysis. The goal is to provide an efficient implementation for each algorithm provided within the library.

Techniques

For preprocessing, data cleaning, preliminary analysis, and testing, Python 3.9.7 was used in conjunction with Jupyter Notebook, along with the following techniques: (1 and 2) Pandas and NumPy were used for data preprocessing and data cleaning. (3) SciPy was used to run all relationship tests, including T-tests, ANOVAs, and linear regressions. (4) Statsmodels was used to calculate effect sizes and run power analyses for T-tests and ANOVAs. (5 and 6) Matplotlib was used in conjunction with Seaborn to create the heatmaps and graphs used to explain the results. (7) From anonympy.pandas , dfAnonymizer was utilized to apply resampling and tokenization to the PII columns. (8) Association metrics was used to calculate Cramer's V for categorical columns. The Cramer's V method measures the degree of association between columns based on Pearson's chi square, which is then fitted to the data provided. (9) From sklearn, preprocessing was used to apply encoded labels to the categorical columns to then compute Pearson's correlation on the encoded dataframe. (10) From prince, Factor Analysis of Mixed Data (FAMD) was used to apply dimensionality reduction to the original data after data preprocessing and cleaning. (11)

Data

The primary dataset used in this project was the 2019 New York State De-Identified Hospital Inpatient Dataset[11], an 867MB large dataset comprising over 2 million hospital cases across 56 counties. The data contains a number of different features, including location data in the form of hospital name, county, and zip code; demographic data in regard to age, sex, and race; and diagnosis/finacial data such as CCSR diagnosis codes, types of insurance used, and total charges/costs. In addition to the hospital data, four supplemental datasets were used in the subsequent analyses, the first of these supplemental datasets being the Baby Names Dataset[12], from which the most common baby names of all New York counties in 2019 were used for re-identification attempts. The second supplemental dataset came from the 2010 Census Percent Urban and Rural Classification webpage[13], specifically the "Percent Urban and Rural in 2010 by State and County" file. As of the time of this writing, this dataset, while over a decade old, is unfortunately the most up-to-date record of how much of a location's area and population is considered to be urban or rural, though it is set to be updated in early 2023, allowing more accurate analysis using such data to be performed then. From this dataset, percent urban/rural split for population was extracted for each New York county. The final two datasets consisted of census population estimate data from the American Community Survey (ACS), table B01001[14] in the ACS was used to find population data for the different sexes and age groups across each NY county, while table B02001[15] was used for population data corresponding to different races across each NY county.

Results and Discussion

How does one know when to anonymize data? As mentioned earlier in the report, anonymization provides confidentiality and privacy. But what if an organization promises to anonymize data, and de-identifies it instead? This is clearly not anonymization. And to top it off, the organization fails to follow HIPAA guidelines and does not de-identify all geographic subdivisions smaller than state. How much privacy and confidentiality is guaranteed then? To demonstrate how de-identified data can be re-identified, an example is provided below

Example:

The 2019 New York State De-Identified Hospital Inpatient Dataset includes features like age, gender,race, ethnicity, date, facility name, hospital service area, hospital county, and zip code. Out of these features, facility name, hospital service area, hospital county, zip code, and date are explicit identifiers. Age, race, gender, and ethnicity are quasi-identifiers. Date has been generalized to year, zip code has been shortened to three digits, age has been grouped into 0-17, 18-29, 30-49, 50-69, and 70-older, race and ethnicity have provided an "other" option in order to retain the privacy of the individual. No de-identifying technique was applied to the hospital service area, facility name, and hospital county. Unfortunately, when looking at other datasets, facility name and hospital service area were not available . The only feature from other datasets that matched with itself in the 2019 New York State De-Identified Hospital Inpatient Dataset was county. As far as datasets are concerned, ones that are small and don't contain a lot of values are the best to use with the 2019 New York State De-Identified Hospital Inpatient Dataset to re-identify individuals. Merging of the 2019 New York State De-Identified Hospital Inpatient Dataset with the Baby Names: Beginning 2007 | New York State Dataset on county and filtered by the "newborn" type of admission, and then aggregating it by county occurrences produced these results: (1) it narrowed down the names of newborn babies to counties and (2) it allowed for the possible re-identification of a newborn's name.

Hospital County	
Albany	7
Allegany	1
Bronx	15
Broome	375
Cattaraugus	1
Chautauqua	1
Chemung	1
Clinton	2
Erie	77
Genesee	1
Jefferson	4
Kings	7515
Madison	420
Manhattan	9277
Monroe	1
Nassau	1066
Niagara	1
Orange	1
Queens	3867
Richmond	1981
Schenectady	10
St Lawrence	157
Suffolk	41
Tompkins	231
Ulster	2
Westchester	553

Inpatient newborn babies are shown for the year 2019 by county. For reproducibility, Allegany was chosen as the county to look into, as it's one of 8 counties with only one entry. Once the county is chosen, the dataframe is then filtered once again by said county. The resulting picture demonstrates how easy it is to re-identify an individual.

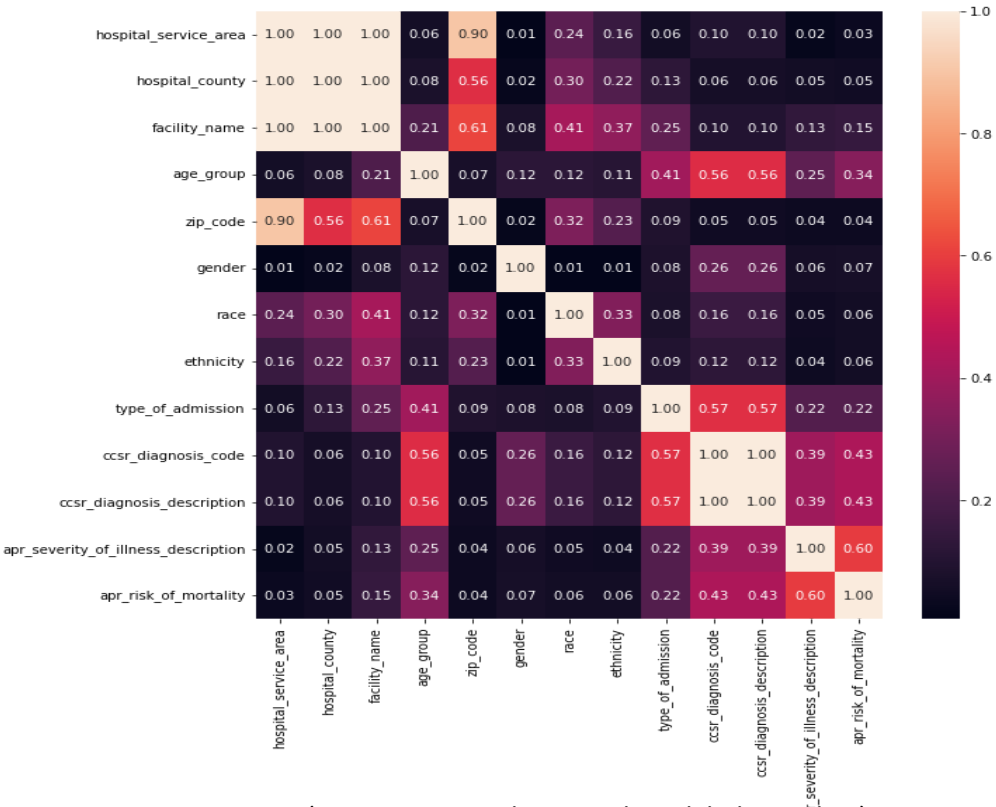
	Year	First Name	Hospital County	Gender
6062	2019	ABIGAIL	Allegany	F

It is easy to see that the inpatient newborn baby's name is Abigail. Clearly, this is a violation of privacy. De-identification is therefore not a good solution when it comes to preserving confidentiality and privacy in health related data.

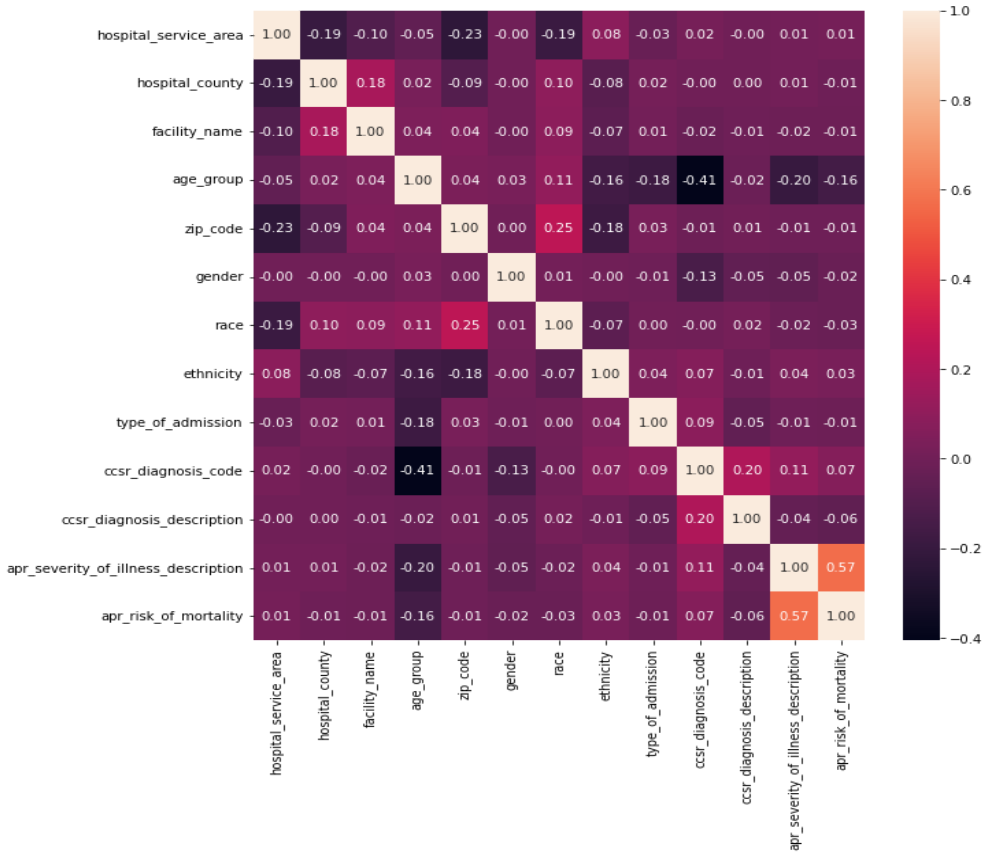
Data anonymization can take many forms. This paper looks at various techniques before settling with resampling. The data anonymization techniques looked at include: (1) label encoding, (2) dimensionality reduction, and (3) tokenization. These techniques provide minimal modification to the original dataset in regards to dealing with categorical values, as well as provide consistency, throughout all experiments.

Label Encoding

Even though label encoding is primarily used for label normalization and transforming non-numerical labels to numerical labels, it can be used as a data anonymization technique. When data is anonymized, it is no longer identifiable as personal data. Label encoding is perfect for when there's a need to normalize multivariate datasets. The dataset is radically simplified to numerical values between 1 to however many variables appear within a column. This removes the risk of re-identification because the values are no longer leaking information that can be used to identify an individual. No particular order is followed in label encoding, nor are values repeated, so that removes biases and imbalance from the data. Although label encoding provides true anonymity, the integrity of the data is highly compromised. As can be seen from the images below, all positive correlation is lost between hospital service area, hospital county, faculty name, and zip code. In the original data, race had a correlation of 0.41 with facility name, meaning that it could possibly be used in conjunction with facility name to re-identify a specific individual. In the label encoded data, race and facility name have no correlation at all. In Figure 1a, age group has a correlation of 0.56 with both ccsr diagnosis code and ccsr diagnosis description. In Figure 1b, age group has two completely different correlations with ccsr diagnosis code and ccsr diagnosis description. There is no need to look further into how the correlations between categories differ between one figure and the other because it is clear that the integrity of the data has been lost. This is great for anonymization, but not great for making conclusions based on incorrect results.



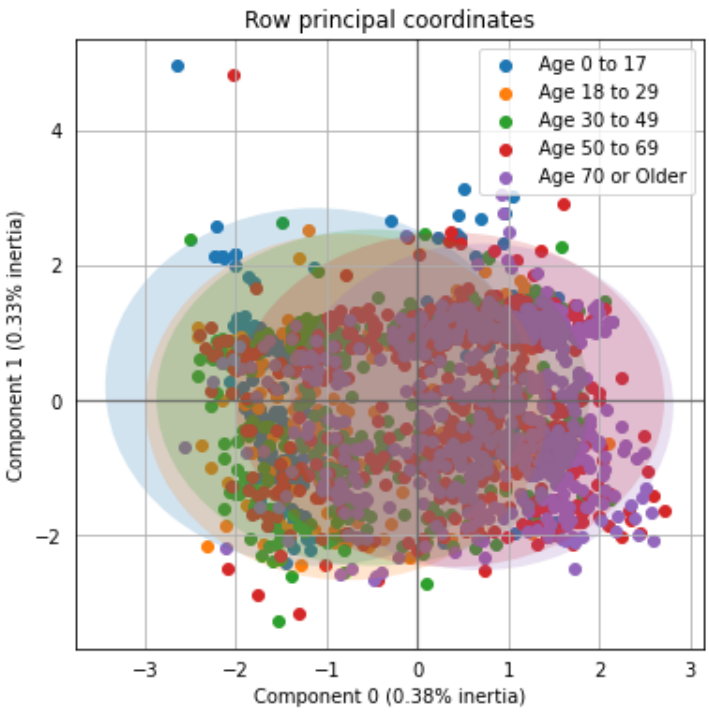
(Figure 1a: Correlation without label encoding)



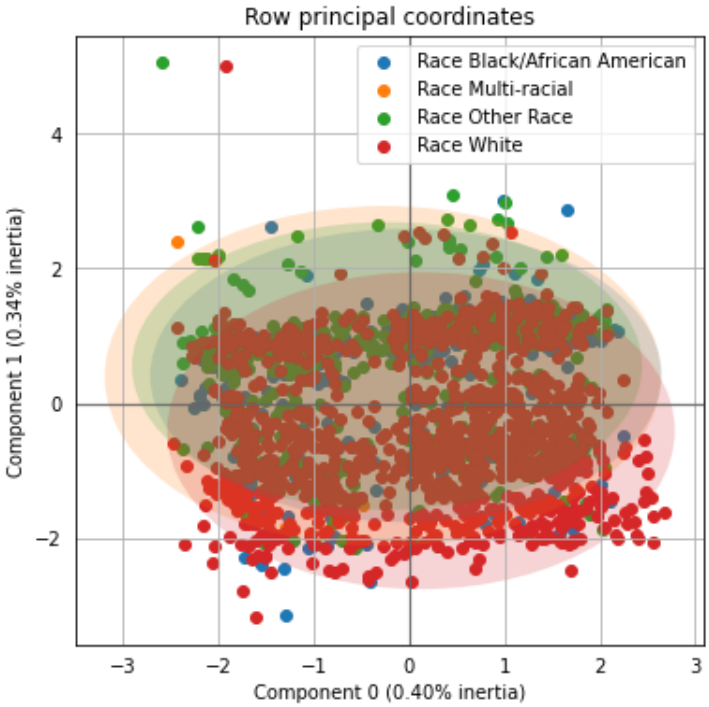
(Figure 1b: Correlation with label encoding)

Dimensionality Reduction

Dimensionality reduction classifies as another data anonymization technique. With dimensionality reduction, high-dimensional data is reduced into a low-dimensional representation of the original data. Principal Component Analysis is primarily used to reduce the data into two principal components. Unfortunately, PCA works only for numerical data. As the dataset used in this report contains a combination of numerical and nominal attributes, Factor Analysis of Mixed Data was used to better portray the data. The problem with imbalanced datasets is that even if one uses dimensionality reduction as a means of retaining anonymity, the imbalanced categories will still be present in the solution, as can be seen from the images below. The reason why race and age are being looked at is because these categories are the most imbalanced categories in the dataset. In both cases, one can deduce that age group 0-17 and race multi-racial are the least represented in their corresponding category. An adversary can use this information to draw valuable conclusions about particular groups of individuals that can lead to re-identification.



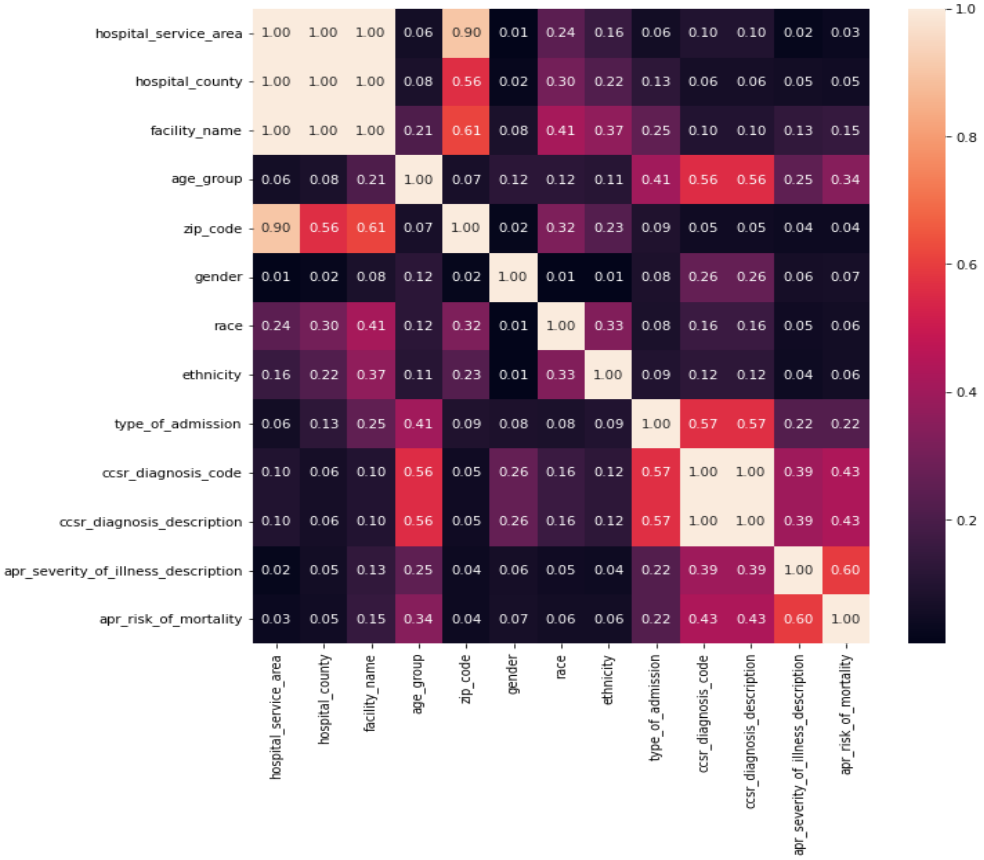
(Figure 2a: FAMD with regard to Age Group)



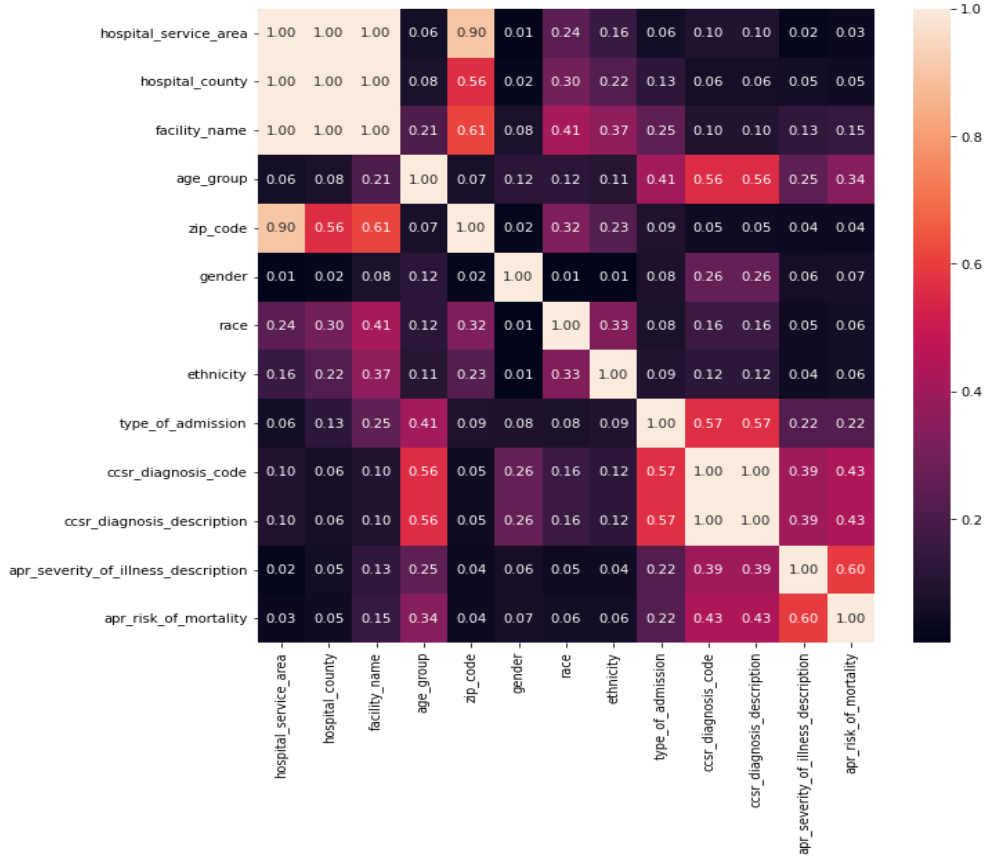
(Figure 2b: FAMD with regard to Race)

Tokenization

Unlike label encoding and dimensionality reduction, tokenization is a well known data anonymization technique. It can be used as a stand-alone solution or it can be used with a combination of other well known anonymization techniques like resampling and noise addition. Tokenization substitutes a sensitive data element with a non-sensitive equivalent that has no intrinsic or exploitable meaning or value. Tokenization as a stand-alone technique does not work well in preserving anonymity if the dataset has many non-unique values. As can be seen from the images below, although the integrity of the data is 100% guaranteed, one can still group correlated values together and decode tokenized values based on the frequency in which the values appear in the dataset. This can lead to attempts to re-identify certain individuals. The experiment first tokenized hospital service area, hospital county, facility name, age group, zip code, gender, and race. This produced the same heatmap as the original. Each column was then individually tokenized and the results still remained the same. As long as there are non-unique values per column, the result will always guarantee data integrity.



(Figure 3a: Correlation without Tokenization)

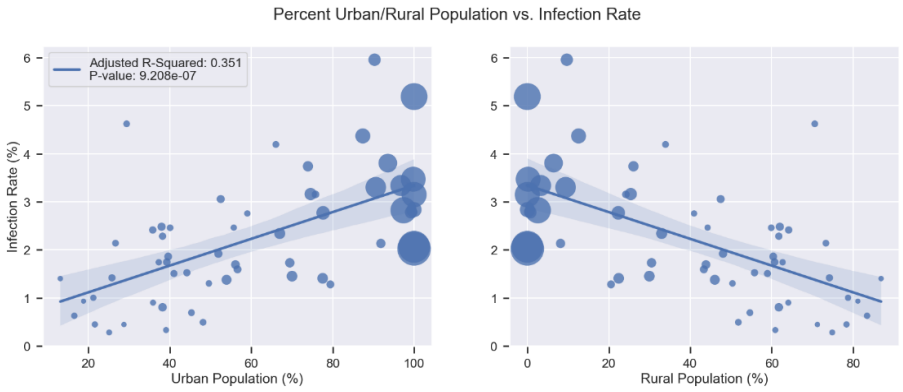


(Figure 3b: Correlation with Tokenization)

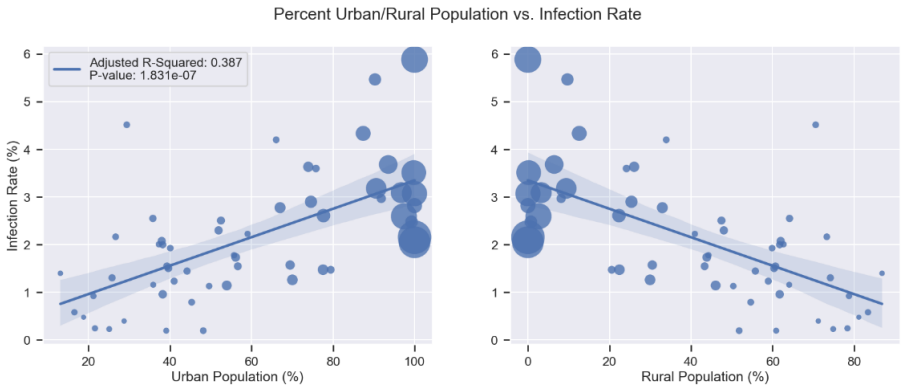
Resampling

Resampling is an anonymization technique that aims to reproduce the distributions of different features via sampling with repetition. To test the effects of resampling on different identifiers found in the data, correlational tests were performed to establish possible relationships between infection rate and identifiers such as county, sex, age, and race. Infection rate was calculated using the number of infection cases for each class in each county, and dividing it by the respective total population for that class and county. For categorical features, a combination of ANOVA and T-tests were used to see if any variation in infection rate between different classes was significant, as well as power analyses to check the validity of significant p-values. For numerical features, linear regression was used to assess if there is a correlation between the two attributes. The primary purpose was not to prove the presence of a relationship between the tested features, or lack thereof, but rather to see if the act of resampling causes a noticeable change in how the data would be interpreted if anonymization were to be performed.

Running resampling on the counties shows little to no difference in the results, illustrating one of the scenarios where resampling retains data integrity to a high degree (Figures 4a and 4b). In both cases, percent urban/rural population and total population (represented as dot size) show a correlation to infection rate, likely due to urban areas tending to have higher population densities that then lead to an increase in vectors for infection to spread. It is possible that resampling did not affect this analysis very much because, after filtering out non-infection cases, no features other than counties from the inpatient dataset were used. The same cannot be said for the other infection rate analyses done, all of which incorporate two features from the inpatient data, a combination of county and age group, sex, or race.

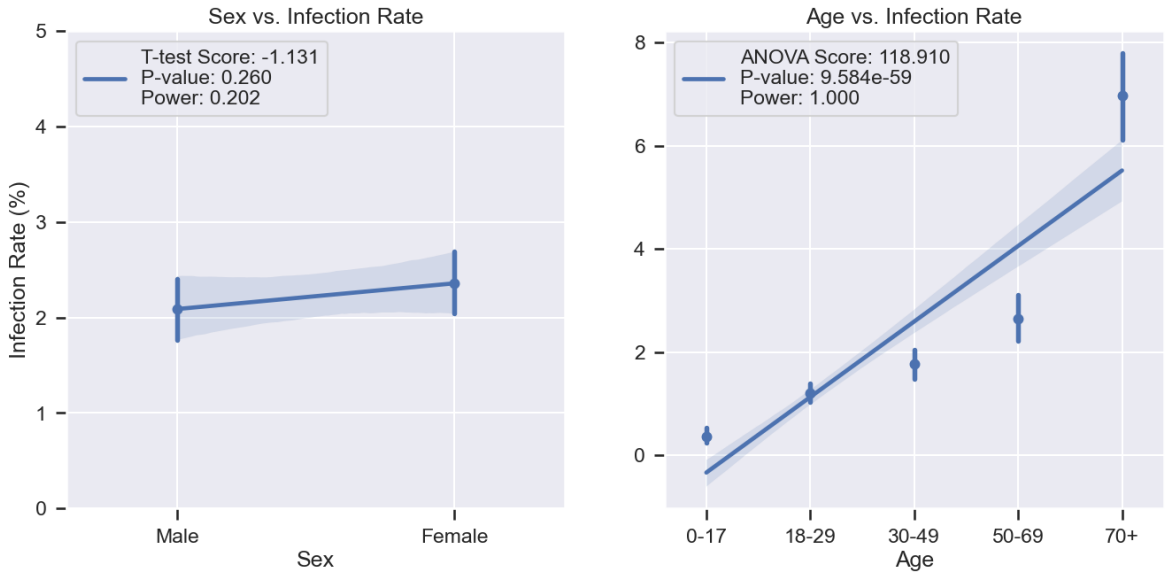


(Figure 4a: Urban Population Percent vs. Infection Rate)

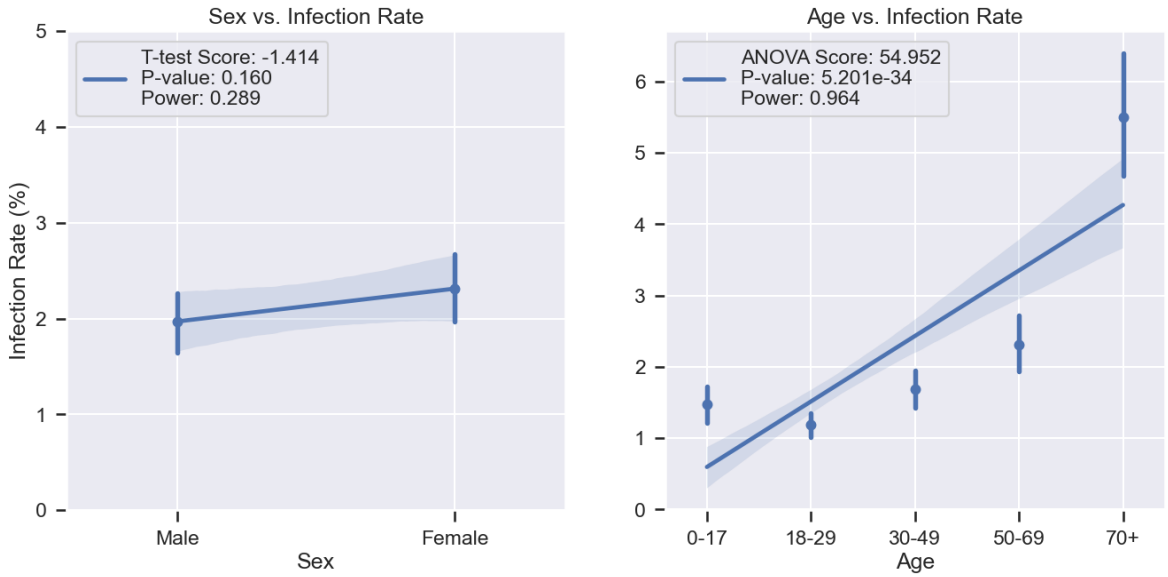


(Figure 4b: Urban Population Percent vs. Infection Rate with resampling)

Both before and after resampling, no significant relationship was found between sex and infection rate, which follows the expected outcome, however the same could not be said about age group (Figures 5a and 5b). It stands to reason that as people get older, immune systems become steadily worse, and are more susceptible to infection as a result. The only notable difference in the data post-resampling regards the 0-17 age group, where said age group's average infection rate increased to the point that it overtakes the 18-29 age group. This deviation likely stems from the fact that after resampling, the number of infections for the 0-17 age group came out to be higher than for the 18-29 age group, which was not the case prior to resampling.



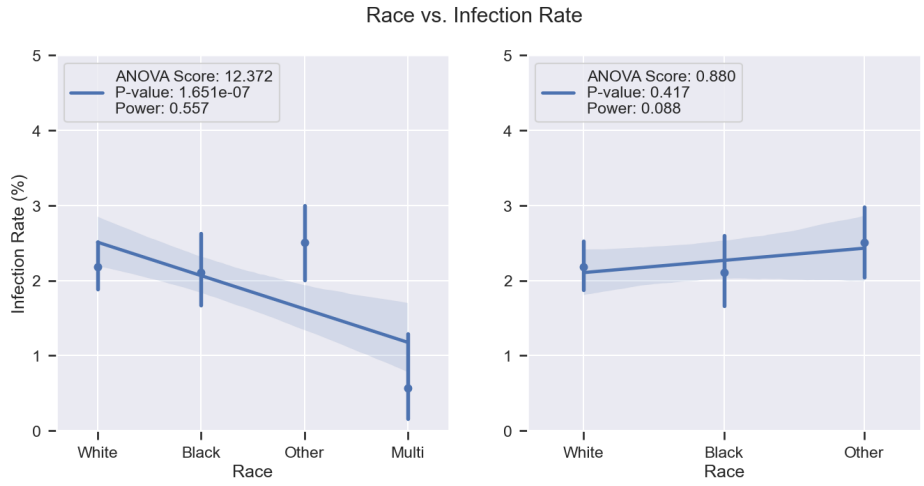
(Figure 5a: Sex/Age vs. Infection Rate)



(Figure 5b: Sex/Age vs. Infection Rate with resampling)

When comparing the infection rates of different races, there were some peculiar results even prior to resampling, namely that the average infection rate of the multi-racial population was significantly lower than for any of the other races (Figure 6a). Upon closer inspection of the data, the reason for such a phenomenon became clear. Many of the counties in the data have very low populations of multi-racial people, leading to some counties having no infection cases for those people, resulting in those counties having infection rates of zero. This likely skewed the average infection rate for the multi-racial group down, causing ANOVA tests to return abnormally low p-values. When the multi-racial class is removed from the analysis, all signs of a significant relationship between race and infection rate go away.

Looking at the same analysis of race vs. infection rate after resampling, there is an extreme deviation from the original data, with the multi-racial population having an average infection rate near zero, and the black and "other" populations showing significantly higher infection rates than before (Figure 6b). The low infection rate for multi-racial people likely comes from the fact that the multi-racial group had very little representation in the original data to begin with, while the number of infections for the black and "other" groups were inflated due to resampling's tendency to oversample the minority class for each county, leading to infection rates are many times higher than what the non-anonymized data showed.



(Figure 6a: Race vs. Infection Rate)

(Figure 6b: Race vs. Infection Rate with resampling)

Conclusion

Despite the necessity for data to be anonymized, or at least de-identified, before being published, doing so without sacrificing data integrity is only possible in a limited number of circumstances. Label encoding, dimensionality reduction, and tokenization are valid techniques for anonymization, but they don't always produce the desired results nor do they make it possible to compare or merge the anonymized features to supplementary datasets, vastly limiting the possible insights that could be drawn from them. Resampling on the other hand, due to its nature of keeping class label names the same, still allows for transformations and correlations with other datasets, but should only be done to balanced features in order to prevent the oversampling or undersampling of certain classes, limiting its utility. Another possible solution, federated learning[16], promises to keep data private by only analyzing/training with non-anonymized data local machines, then merging the resulting models. Such a solution is not only extremely difficult to implement across the number of databases that would need to be included to achieve meaningful results, but also doesn't solve the primary problem of making those insights possible to be discovered through a public source. Thus, when a publisher is anonymizing a dataset containing quasi/explicit identifiers, it is vital that they strike a balance between how much raw information to keep for the sake of data integrity and how much to obfuscate for the sake of privacy.

Data Sources

2019 De-Identified New York Hospital Inpatient Data
(https://health.data.ny.gov/Health/Hospital-Inpatient-Discharges-SPARCS-De-Identified/4ny4-j5zv)

Baby Names: Beginning 2007 | New York State
(https://health.data.ny.gov/Health/Baby-Names-Beginning-2007/jxy9-yhdk)

Percent Urban/Rural areas of New York Counties
(https://www.census.gov/programs-surveys/geography/guidance/geo-areas/urban-rural/2010-urban-rural.html)

New York County Populations by Age/Sex
(https://data.census.gov/table?q=B01001&g=0400000US36$0500000&tid=ACSDT1Y2019.B01001&moe=false)

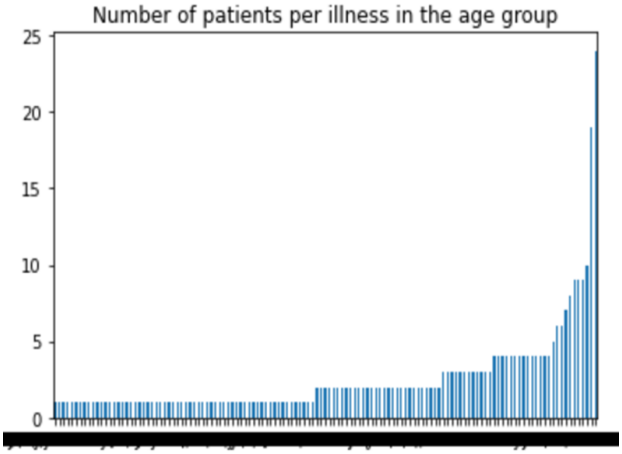
New York County Populations by Race
(https://data.census.gov/table?q=B02001&g=0400000US36$0500000&tid=ACSDT1Y2019.B02001&moe=false)

Appendix

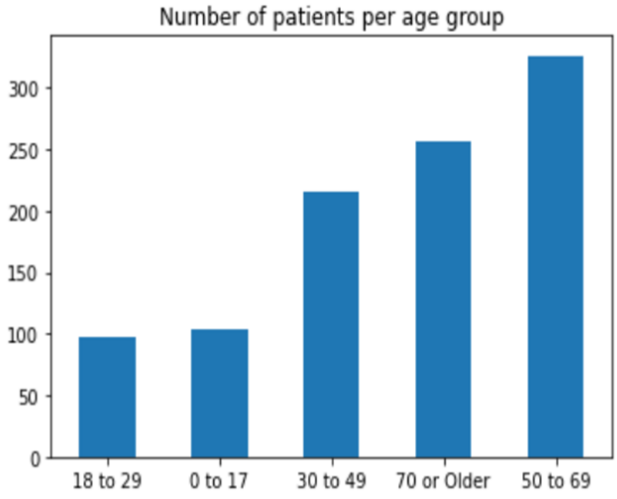
These experiments were used as a preliminary stepping stone for a more advanced and comprehensive analysis.

Correlation

In order to examine the data more efficiently, first start using the column header as an argument. The .copy() method in pandas copies data from a parent data frame to a new data frame. Data can be sorted unilaterally or in steps. The data analysis procedure is made simple and transparent by the step-by-step sorting of the data. To determine the age group with the most patients, which is unquestionably the age group most impacted, data is first sorted using the initial sorting rule, in this case, the mode of the age group. Then, a new data frame is generated using only information from the most affected age group. The next step is to call the second sorting, which involves determining the diseases' modes. This gives you the name of the highest-ranking column. The x-axis of the graph depicts the diseases, and the y-axis the number of patients. As a result, the below graph (7a) depicts **Septicemia & Disseminated infection**, the disease that affects the greatest number of people. In the following graph (7b) the age range 50 to 69 shown is being most susceptible to the disease Septicemia & Disseminated infection.

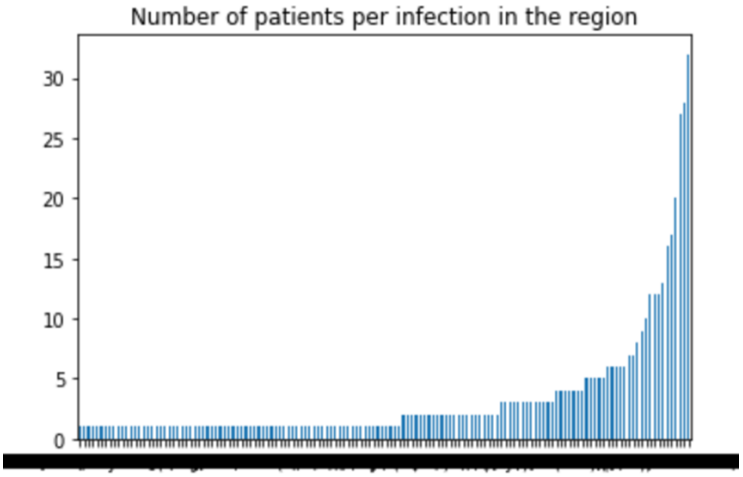


(Figure 7a: Diseases vs. number of patients)

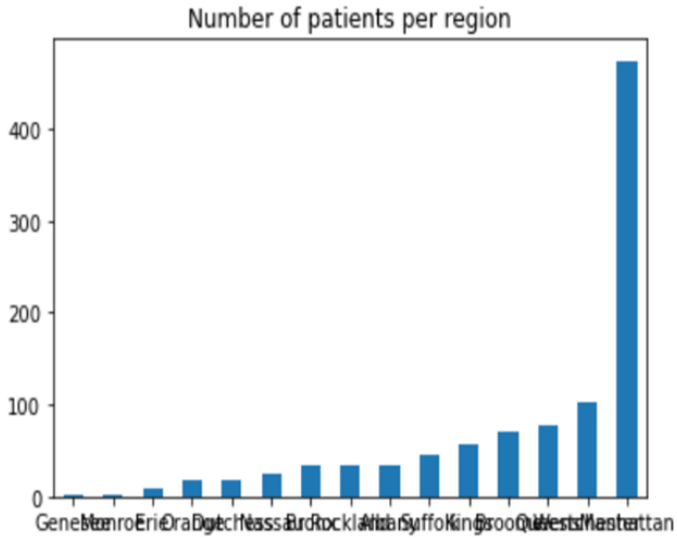


(Figure 7a: Age group vs. number of patients)

As in the previous case, determining the area most affected entails a two-step sorting phase. First, a new data frame is created and the data that simply includes hospital county, the region, and illness is copied over. The most affected area is then determined using the hospital county's mode. The mode in the disease's column can then be used to identify the disease that primarily affects this area. In the graph (8a) the x-axis depicts the region, and the y-axis depicts the number of patients. In the following graph (8b) it shows the most severely affected region is **Manhattan** (which shows the overall number of patients by region).



(Figure 8a: region vs. number of patients)



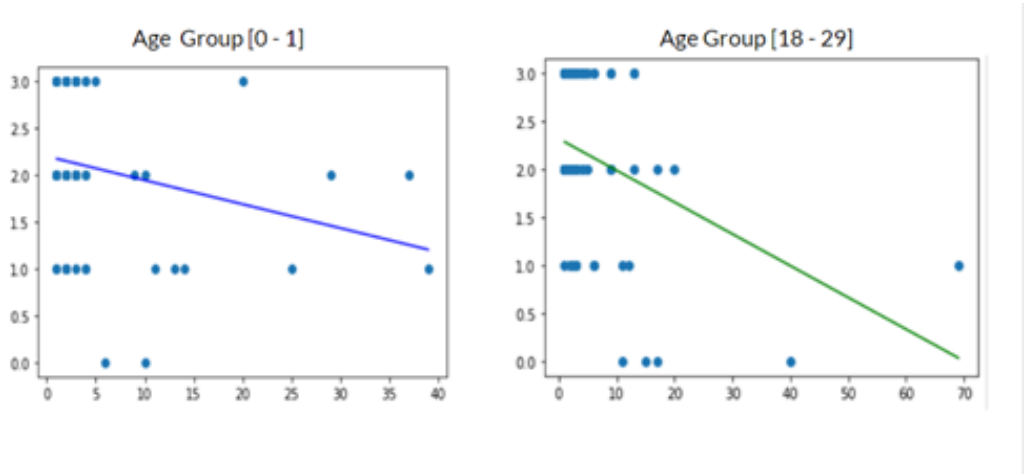
(Figure 8b: region vs. number of patients)

In order to determine the individual medications, a new data frame made up of the medicine, the ailment it treats, and the length of time a patient is on medication is taken from the parent data frame using the copy method. Data is then sorted by eliminating duplicate entries depending on the medicines column. The unique medications, diseases they treat, and duration are included in the generated data set. By iterating through the data frame and printing the results row by row, the results can then be displayed. There were 210 different medications that each treated a different condition at the time of analysis. The result is :

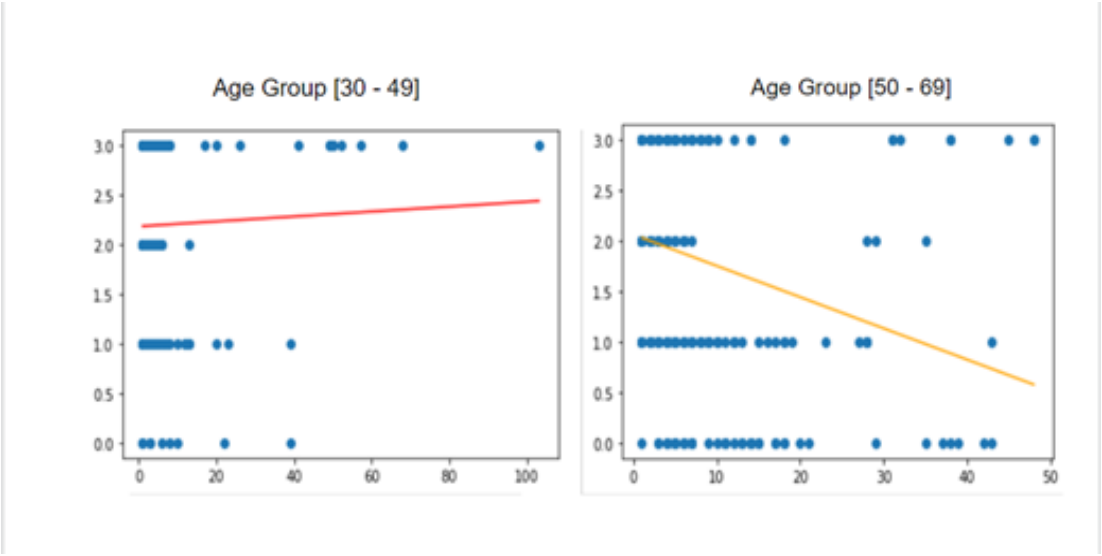
```
The drug  INF008 cures the disease VIRAL INFECTION in 2 days
The drug  GEN002 cures the disease ACUTE AND UNSPECIFIED RENAL FAILURE in 3 days
The drug  GEN006 cures the disease OTHER SPECIFIED AND UNSPECIFIED DISEASES OF KIDNEY AND URETERS in 7 days
The drug  RSP009 cures the disease ASTHMA in 1 days
The drug  MBD012 cures the disease SUICIDAL IDEATION/ATTEMPT/INTENTIONAL SELF-HARM in 3 days
The drug  END003 cures the disease DIABETES MELLITUS WITH COMPLICATION in 5 days
The drug  DIG012 cures the disease INTESTINAL OBSTRUCTION AND ILEUS in 9 days
The drug  RSP002 cures the disease PNEUMONIA (EXCEPT THAT CAUSED BY TUBERCULOSIS) in 3 days
The drug  CIR012 cures the disease NONSPECIFIC CHEST PAIN in 1 days
The drug  SKN001 cures the disease SKIN AND SUBCUTANEOUS TISSUE INFECTIONS in 4 days
The drug  SYM002 cures the disease FEVER in 2 days
The drug  MUS006 cures the disease OSTEOARTHRITIS in 1 days
The drug  INJ037 cures the disease COMPLICATION OF OTHER SURGICAL OR MEDICAL CARE, INJURY, INITIAL ENCOUNTER in 7 days
The drug  MUS011 cures the disease SPONDYLOPATHIES/SPONDYLOARTHROPATHY (INCLUDING INFECTIVE) in 1 days
The drug  INJ035 cures the disease COMPLICATION OF INTERNAL ORTHOPEDIC DEVICE OR IMPLANT, INITIAL ENCOUNTER in 3 days
The drug  MBD001 cures the disease SCHIZOPHRENIA SPECTRUM AND OTHER PSYCHOTIC DISORDERS in 37 days
The drug  MBD017 cures the disease ALCOHOL-RELATED DISORDERS in 5 days
The drug  BLD006 cures the disease COAGULATION AND HEMORRHAGIC DISORDERS in 6 days
The drug  CIR008 cures the disease HYPERTENSION WITH COMPLICATIONS AND SECONDARY HYPERTENSION in 1 days
```

Feature Removal

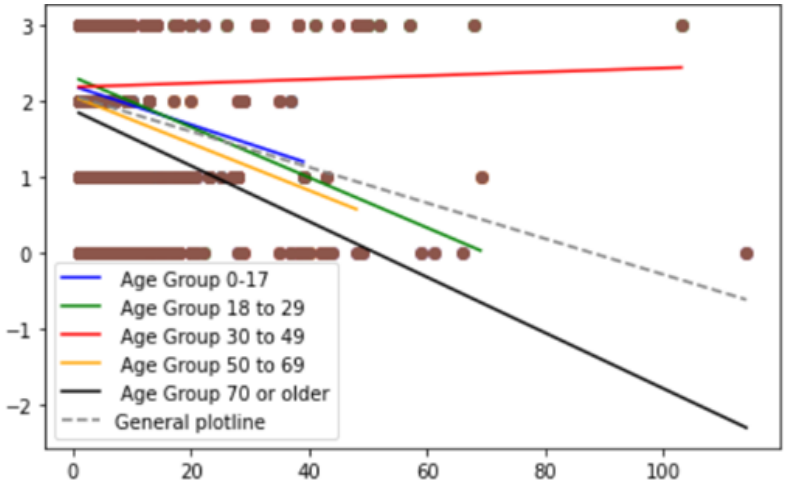
To achieve feature removal on a particular column and to eliminate that column from the data set is achieved as follows: First the required data is extracted from the API and then the extracted data is represented in the JSON format. The following functions are used for few conversions, mydataset.json() function converts the data into the JSON format, pd.read_json() function converts the data into a python dataframe for further analysis, pd.Series is used to classify the different age groups into corresponding groups such as '0-17', '18-29', '30-49', '50-69', '70+'. Correlation is performed in comparison with Age to Length of stay of the patient and Severity of the disease. The effect of a patient's age on the cure period depending on the severity of disease. Individual age group correlations can be visualized and compared against each other. All the age groups except the age group 30 to 49 have a negative correlation. This implies that for all the other age groups, if the disease is very severe, it takes a shorter time to cure. In the special case of the age group 30 to 49 years, the increase in severity leads to an increase in the cure period.



(Figure 9a: total no.of Patients vs Age)

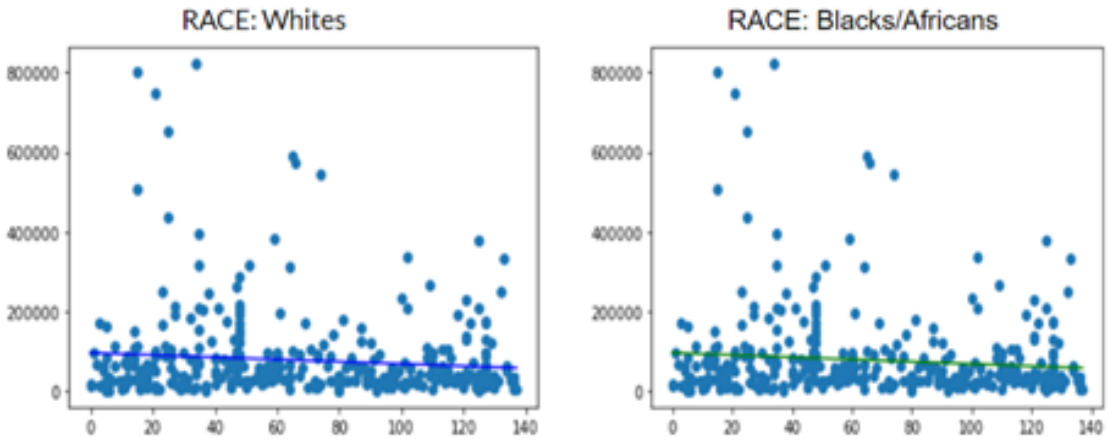


(Figure 9b: Total no.of Patients vs Age)

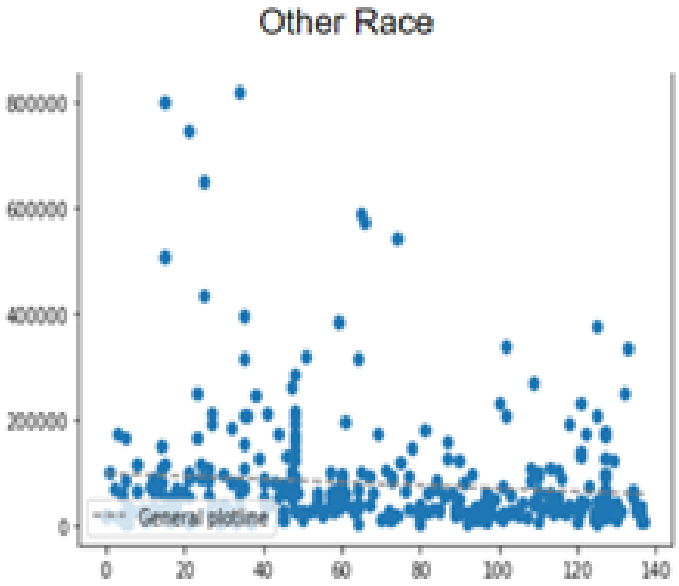


(Figure 9c: Age vs Severity)

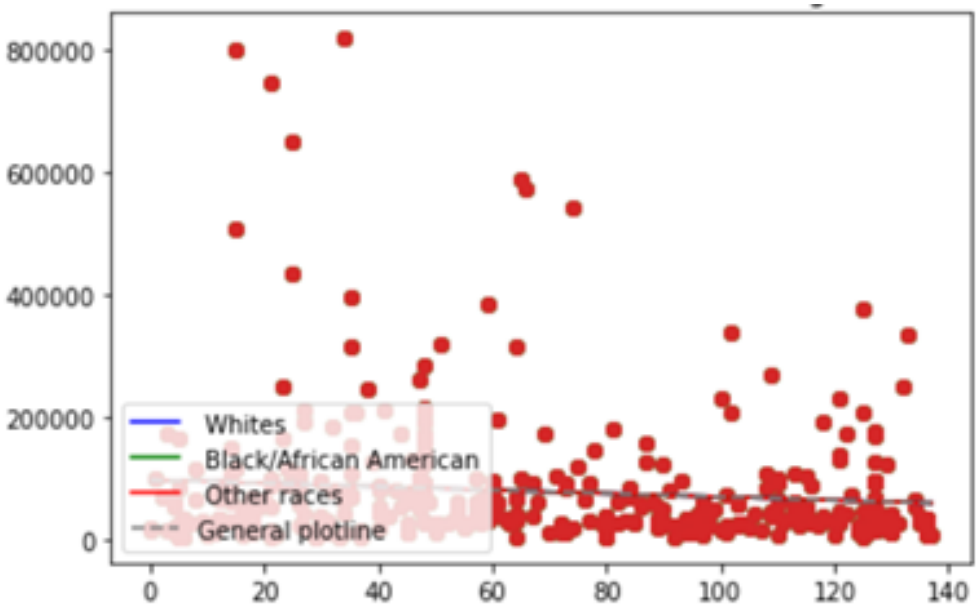
The next analysis is to see the effects of the race column removal. In this analysis the fit_transform method is used to compute the mean and standard deviation for a given feature to be used further for scaling. Here it is compared if the Race variable has any relation with the Total Charges column for the same disease. From the data, correlations can be drawn to view the relationship between the different races as far as race is concerned. From the plotted data, the correlation line of best fit for all the races and the general correlation line all lie at the same line. This means that race does not affect the total charges of patients with the same disease. Since it has been achieved that the Race variable has no relation to the Total Charges charged for a patient, feature removal can be performed here.



(Figure 10a: Total no.of Patients vs Race)



(Figure 10b: Total no.of Patients vs Race)



(Figure 10c: Patient's Race vs Total Charges)

Contributions

Natalia Valencia: Abstract, Introduction, Approaches, the intro and example inside results and discussion, Label Encoding, Dimensionality Reduction, Tokenization.

Erick Gonzalez-Vega: Data, Resampling, Conclusion, Data Sources, Website.

Sharon Beulah - Correlation.

Ananya Kaza - Feature Removal.