Modeling MAUDE: Final Report

Ayesha Darekar, Robbie Goss, Vrishank Ghosh, Kris Wilson

Summary

In this final report, our team delves into the analysis and development of the MAUDE (Manufacturer and User Facility Device Experience) database, focusing on data from 2016 to 2019 (inclusive). The MAUDE database maintained by the FDA, is a comprehensive collection of medical device reports, including mandatory filings from manufacturers, importers, and device user facilities. Our primary objective is to accurately predict the 'Event Type' for each record, a critical aspect in identifying and mitigating potential safety hazards associated with medical devices. We employed extensive data cleaning and preprocessing techniques, merging various MAUDE files, addressing missing values, and creating variables of interest. Feature engineering played a significant role in our analysis, notably the creation of a variable for domestic reports and the application of Latent Semantic Analysis (LSA) to text data. The result of our analysis is a random forest classification model, augmented with LSA, which underwent rigorous optimization through cross-validation. This final report details our analysis, challenges, methodologies, and results.

Problem

Medical device errors can be catastrophic for a patient who is relying on medical device(s) for their health. When a medical device fails, it can lead to improper treatment, worsening of a patient's condition, and in severe cases death. A 2013 review article published in the *Journal of Patient Safety* estimated that 210,000-400,000 premature deaths occurred because of preventable harm to patients (James, John T. PhD). Not only is this statistic staggeringly high, but these are preventable premature deaths. Our group tackled this problem by modeling the MAUDE data given by the FDA to better understand the factors that impact the type of event (death, injury, none, other) that occurs when a medical device malfunctions. An understanding of what factors influence a specific event type can help reduce medical problems–and most importantly serious problems such as death–from occurring.

Data

The data we used is a subset of the Manufacturer and User Facility Device Experience (MAUDE) database on the FDA website. This database holds records of adverse events that concern medical devices since 1995. The following is a list of datasets from the FDA MAUDE website that we used to create a singular cleaned dataset:

- patientThru2022.txt: contains information about patients associated with reported adverse events with medical devices through 2022

- mdrfoiThru2022.txt: contains information about the reports on adverse events through 2022
- DEVICE[YEAR].txt: contains information about medical devices associated with the reported adverse events for a certain year
- patientproblemcode.txt: contains codes for patient issues that resulted because of adverse events with medical devices
- foidevproblem.txt: contains codes for medical device issues that resulted in adverse events
- deviceproblemcodes.csv: contains codes for medical device issues and their matching descriptions
- patientproblemcodes.csv: contains codes for patient issues and their matching descriptions

The dataset we are using for modeling contains records from 2016-2019. The variables of interest that are in our cleaned dataset are the following:

- MDR_REPORT_KEY: a unique identifier for the reported event
- DATE_RECEIVED: date report was received by the FDA
- REPORT_SOURCE_CODE: code that represents where the report came from
    - P: voluntary report
    - U: user facility report
    - D: distributor report
    - M: manufacturer report
- DATE_OF_EVENT: date the adverse event occurred
- REPORTER_OCCUPATION_CODE: code that represents what occupation the person who reported the event is in
- EVENT_TYPE: code that represents the type of adverse event that occurred
    - D: death
    - IN: injury
    - IL: injury
    - IJ: injury
    - M: malfunction
    - O: other
    - *: no answer provided
- REPORTER_COUNTRY_CODE: code that represents the country in which the event was reported
- PMA_PMN_NUM: the Premarket Approval (PMA) or Premarket Notification (PMN) number
- MANUFACTURER_D_NAME: name of the device's manufacturer
- DEVICE_REPORT_PRODUCT_CODE: code that represents the type of device

- COMBINATION_PRODUCT_FLAG: whether or not a combination product (product that integrates drugs, devices, and/or biological products) was used
    - Y: used
    - N: not used
- BRAND_NAME: the device's brand name
- DEVICE_PROBLEM_CODE: code that represents the issue the device had
- PATIENT_PROBLEM_CODE: code that represents the consequences of the adverse event to the patient
- PATIENT_PROBLEM_DESCRIPTION: describes the consequences of the adverse event to the patient
- DEVICE_PROBLEM_DESCRIPTION: describes the issue the device had

Methods

Data Cleaning

Our first step was to download the data from the FDA MAUDE website for each year from 2016 to 2019. The datasets we downloaded for each of these four years were patientThru2022.txt, mdrfoiThru2022.txt, DEVICE[YEAR].txt, patientproblemcode.txt, foidevproblem.txt, deviceproblemcodes.csv, and patientproblemcodes.csv, where [YEAR] corresponds to a year from 2016 to 2019.

For each of the four years, the following data cleaning steps were performed:

1. We extracted records from patientThru2022.txt and mdrfoiThru2022.text that corresponded to the year that we called patient[YEAR].csv and mdr[YEAR].csv, respectively.
2. We subsetted patientproblemcode.txt, DEVICE[YEAR].txt, patient[YEAR].csv and mdr[YEAR].csv so that these datasets only had the variables of interest mentioned in the Data section of this report.
3. We merged the subsetted patientproblemcode.txt, DEVICE[YEAR].txt, patient[YEAR].csv and mdr[YEAR].csv datasets using the MDR_REPORT_KEY variable.
4. We used deviceproblemcodes.csv and patientproblemcodes.csv to match the patient and device codes in the merged dataset to the corresponding description that was listed in these two csv files. We added the patient problem and device problem description variables to the merged dataset.
5. We saved the merged dataset as cleandata[YEAR].csv.

After these data cleaning steps were performed for each year, we vertically merged all four clean datasets into a singular, merged dataset. Preprocessing of the merged dataset included converting

DATE_OF_EVENT and DATE_RECEIVED into variables of class “date” (from character) and removal of missing values. The missing values for the event type variable (our response), denoted by an asterisk (“*”), made up well under 1% of the overall extracted data, so removal suffices.

Feature Engineering

Some variables contained extremely little information due to the sparsity of the data. One such column was REPORTER_COUNTRY_CODE. While the majority of the reports were from the United States, many countries around the world had as little as one or two reports in the MAUDE database. This inspired us to create a new variable, DOMESTIC_REPORTER, that was derived from the values of REPORTER_COUNTRY_CODE. This new variable took on a value of “Yes” if the report came from a domestic source (i.e., U.S.), and “No”, if not. If the original value of REPORTER_COUNTRY_CODE was missing, the value of DOMESTIC_REPORTER set to missing as well.

Understanding we had many columns of text data, we performed latent semantic analysis (LSA) on the text data. Due to the size of the data (the final, cleaned dataset contained over four million observations), this presented a challenge (further discussed below), and LSA was limited to the description variables (PATIENT_PROBLEM_DESCRIPTION and DEVICE_PROBLEM_DESCRIPTION) in addition to the reporter-related variables (REPORTER_OCCUPATION_CODE and DEVICE_REPORT_PRODUCT_CODE). This was done by first subsetting the data for each outcome of EVENT_TYPE present in the dataset. Of note, only four of the six possible outcomes were present in the 2016-2019 data: D (death), IN (injury), M (malfunction), O (other).

Next, the data was split into report-related variables and description variables. This was advantageous because the documents could be kept separate, so the term-document matrix (TDM) was not so large that it could not be worked with. From there, separate latent semantic analyses were performed on each combination of report and description variables for each level of EVENT_TYPE present in the data. For the report-related variables, numeric representations of the *terms* were extracted, while for the description variables, numeric representations of the *documents* were extracted. The reason for the difference lies in the structure of the data: the report-related variables were often in three-digit codes (e.g., “002”, "0HP", "OXO"), while the description-related variables were phrases of varying length. Thus, multiple report-related variables could be combined to create a single document, containing the three-digit terms present in the original data, resulting in the terms containing the relevant information. For the description variables, however, due to their varying length, the documents themselves contained the relevant information. Terms often consisted of partial or incomplete phrases such as “device”, “naturally”, and “insufficient”; in contrast, the full documents contained phrases such as “Dizziness”, and

“Difficult or Delayed Activation”. These representations were computed by multiplying the left-singular term matrix by the diagonal matrix of singular values for the report-related variables (recall the diagonal matrix of singular values represents the importance of each dimension), and multiplying the diagonal matrix of singular values by the right-singular document matrix of the description variables. Whenever possible, the full dimensions were used–sometimes this was not feasible, as the resulting TDM could contain over one thousand terms. In those cases, the first few dimensions were utilized. From there, these numeric representations were aggregated into one column, named REPORT_WEIGHT, a single number for each term-document combination.

Modeling

From there, we begin the modeling process. Many different classification models could have been fit to this data, and some were attempted but did not make the final model. In the end, we implemented a random forest classification model that applied latent semantic analysis on the PATIENT_PROBLEM_DESCRIPTION, DEVICE_PROBLEM_DESCRIPTION, REPORTER_OCCUPATION_CODE, and DEVICE_REPORT_PRODUCT_CODE variables. We had originally used a random forest classification model on its own, but applying the latent semantic analysis to these predictors before rerunning the random forest model lowered our error rate from 4.46% to 2.5%.

To create the random forest model to predict event type, we appended the variable REPORT_WEIGHT produced by LSA to our merged dataset, omitted records that contained any NA values, and dropped the PATIENT_PROBLEM_DESCRIPTION, DEVICE_PROBLEM_DESCRIPTION, MDR_REPORT_KEY, DEVICE_PROBLEM_CODE, PATIENT_PROBLEM_CODE, REPORTER_COUNTRY_CODE, REPORTER_OCCUPATION_CODE, and DEVICE_REPORT_PRODUCT_CODE variables.

We implemented 5-fold cross-validation to tune the num.trees (the number of trees that are produced by the model) parameter. We set mtry (the number of predictors that are randomly sampled at each split in a tree) equal to the square root of the total number of predictors which is 3. For each fold, we used 70% of the data for training and 30% for testing to train and test three random forest models using the ranger package. Each of the three models had 50, 100, and 200 trees set for the num.trees parameter, respectively. The importance mode for all models was set to “impurity” (Gini index for classification). This is calculated by subtracting the sum of squared probabilities of each class from one. After cross-validation was run on the three models, the mean misclassification error for each tree was calculated. The “best” model which had the lowest mean misclassification error was the model that was selected. The “best” model was trained using 70% of the dataset and the remaining 30% for testing. A confusion matrix was created for the “best” model and the misclassification error was calculated. We then created ROC curves for each combination of values for event type.

Results

The mean misclassification errors for each tree from cross-validation are listed in the following table:

	mtry	**num.trees**	**Misclassification Error**
Model 1	3	50	2.62%
Model 2	3	100	2.59%
Model 3	3	200	2.50%

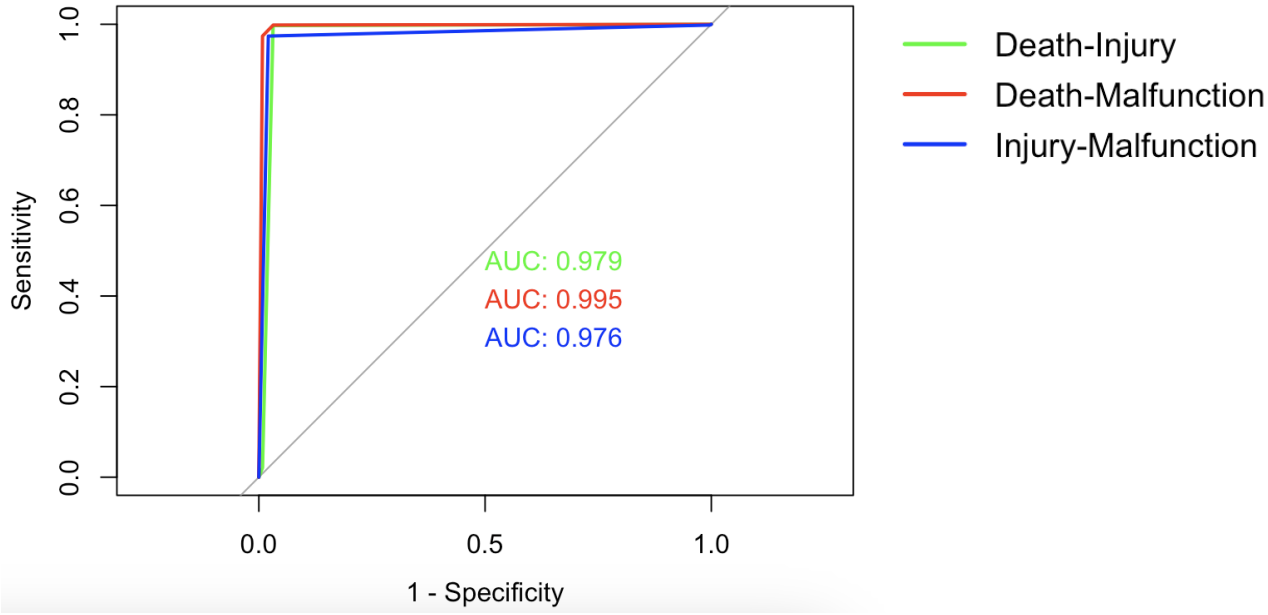
From these results, we choose model 3 (mtry = 3, num.trees = 200) as the "best" model to test and train on the whole dataset since it had the lowest misclassification error rate.

After training and testing a random forest model with mtry set to 3 and the number of trees parameter set to 200, we found the misclassification error to be 2.51%. The confusion matrix for this model is below where "D" is death, "IN" is injury, and "M" is malfunction:

	D	IN	M
D	1191	121	89
IN	29	42509	1242
M	10	915	49748

We note that the category "O" (other) is not in the confusion matrix because when we omitted records that had NA values from the cleaned, merged dataset, all records where event type was other had at least one NA value. We also note that the "IL" and "IJ" categories for event type are not in the confusion matrix because our cleaned dataset did not have records with those event types values.

The following graph shows the ROC curves for each combination of event type values:



The ROC curves and AUC values are color-coded by the combination of event type values. Each AUC value for each color combination is greater than 0.95. This information combined with the low misclassification error suggests that this random forest model performs very well at predicting event type.

Discussion

Overall Prediction

The results from our latent semantic analysis in tandem with a random forest classification model show that the type of adverse event a patient has from a device malfunction can be accurately predicted based on the factors in our data set. Our cross-validation showed a misclassification error of only 2.5%. This implies that we have the necessary information in front of us to understand what factors need to be further investigated to see how we can prevent adverse events in the future. The problem in front of us is daunting, as hundreds of thousands of premature deaths are occurring, but our results show we have the necessary information to begin tackling the biggest contributors to these deaths.

Predicting Death

Although the low misclassification error rate looks promising, we note that the distribution of categories for the event type variable is uneven. After records with NA values were omitted from our clean dataset, the death, injury, and malfunction categories had the following distribution:

D	IN	M
4661	146405	168450

Compared to the death category, the injury and malfunction categories had a lot more records. This uneven distribution in event type could have caused the model to lean towards predicting injury or malfunction simply because these categories have significantly more records. To combat this, we could sample an equal number of records from each of the death, injury, and malfunction categories and rerun the random forest model to see how well it does.

With that said, however, our model predicted these death outcomes with a relatively high accuracy even though it was an infrequent event type. Our test data contained 1,401 occurrences of death and 1,191 of these occurrences were accurately predicted to be death, resulting in a 15% error rate. We can also see that our model very infrequently predicts death when one did not occur, as this only happened 39 times. This indicates that our model still was relatively accurate in predicting when a death would or would not occur, but there is room for improvement that could come from a future model with a dataset that had a balanced number of categories for event types.

Because of the relatively high accuracy for predicting death, we have the information to predict when deaths will and won't occur, so we can further investigate to understand and take preventative measures to combat situations and events where deaths are likely to happen. This does come with a caveat, however, as the PATIENT_PROBLEM_DESCRIPTION can simply list "Death" as the description, which could mean while our LSA analysis of this description could very accurately predict this event type, this predictor variable will not be useful in understanding why the event type occurred. Therefore, our understanding of what factors result in death will come from a deeper dive into the other predictors in our data set.

Challenges and Next Steps

This dataset presented a few key challenges: the lack of information provided by the MAUDE website, the size, and the qualitative nature of the data set. To begin this process, we had to learn about and clean our data set. This was difficult because the FDA was not as descriptive about their data as we would have hoped. For example, some descriptions of our variables were insufficient or simply missing, leaving us to have to make assumptions and piece together the meaning behind multiple variables.

Next, the size of this data was relatively large even though we were only working with four years of data. Our cleaned dataset contained over 4 million observations, which made fitting

models throughout the process computationally difficult and time-consuming. This was addressed in our final model by omitting rows with at least one NA value for any given variable, which reduced the size of the data set to around 300,000 rows making the data less taxing to work with while simultaneously providing enough data to train a model. One last difficulty in this process was caused by the qualitative nature of many of our predictor variables. Many models that we wanted to fit, such as K-nearest neighbors, support vector machines, and multinomial logistic regression needed numeric representations of their predictors. Since much of our data was non-numeric, we had to try strategies such as creating dummy variables for these non-numeric predictors, but that made our already large data set even more computationally difficult to work with. In the end, we chose to use a random forest model since that better accounted for the nature of our data.

Multiple steps could be taken to expand upon the modeling we did in this project. The first of which is creating a new random forest model that uses a dataset with an even proportion of each event type. This model would likely be better at predicting when death will occur than the model we have right now, even though our current model does a decent job at predicting death with an 85% true positive rate. The next step would be to account for more of the data that the FDA has on its MAUDE website. For example, the FDA provides long-form text descriptions of the events that occurred. With that data, we could apply more informative text analysis models to more accurately predict the type of adverse events patients have. A third step that could be taken would be to further analyze our random forest model to better understand factors such as which predictors have the most influence on the model; we could fit a gradient-boosted tree or other methods that handle complex interactions between variables to gain insight into this.

References

James, John T. PhD. A New, Evidence-based Estimate of Patient Harms Associated with Hospital Care. Journal of Patient Safety 9(3):p 122-128, September 2013. | DOI: 10.1097/PTS.0b013e3182948a69

U.S. Food and Drug Administration. (10/23/2023). About Manufacturer and User Facility Device Experience (MAUDE).