Identifying Hidden/Indirect Bias in Surgical Outcomes of Colorectal Procedures

Rohith Ravindranath

Independent Research Final Report

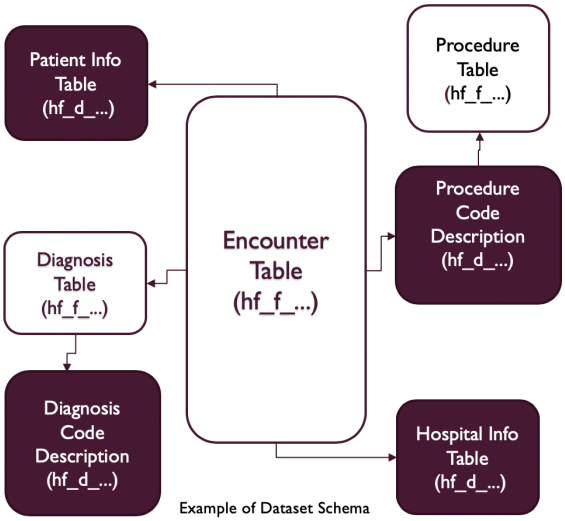
An Informative Description of the Problem to be Addressed

With the world now focused on data, it becomes imperative that hidden biases do not influence the algorithms we use to solve problems. As we develop more complex machine learning methods, we are faced with an increased risk for biased algorithms due to biases that humans cannot necessarily predict from viewing raw data. When these types of algorithms are used in real-world problems (i.e. healthcare industry), they could reinforce societal disparities in the decision-making process based on race, gender, age, and many more factors. Since many of these hidden biases arise from data, we need methods that identify them before applying them to our algorithms. With the use of a Causal Inference Model, we can capture the bias in the data as well as quantify the direct and indirect effect of a factor towards an outcome. In most cases, the biases in an algorithm are based on an indirect factor. My research project with Professor Sunil Prabhakar focused on identifying such factors (direct and indirect) for medical patients that produce a biased outcome to their treatment.

Specifically, for this research project, we are focusing on patients that have undergone colorectal procedures. We picked this subset of patients since colorectal procedures are common across all hospitals, thereby having lots of data within the Cerner dataset to work with. Another reason we picked this subset is that there are specific protocols that have been placed for doctors to follow to reduce the number of surgical infection sites (SSI). While many hospitals adhere to these protocols, many other hospitals don't, putting them at a disadvantage and significantly differ in the numbers between hospitals within the dataset. One specific example that I will be focusing on with my research is procedures done with minimally invasive (robotic/laparoscopic) techniques vs. open techniques. The former has taken over many urban health centers due to increased funding compared to hospitals in rural cities. Based on this knowledge, I hypothesize that the location of hospitals, more specifically whether they are located in an urban or rural setting, is a bias factor for patients that have undergone colorectal procedures.

The first step for this project is gathering the necessary data and understanding the type of information the data represents. All of our data comes from the Cerner database. Many hospitals use the Cerner system to maintain records of patients as electronic health records. Through this, Cerner has gathered data on hospitals and patients all over the country. The specific version of the database consists of medical records for hospitals that used the Cerner system between 2000 to 2018. It contains about 480 million medical records from about 750 hospitals around the country. This massive amount of data will give us an accurate representation of the patients nationwide, but will also pose a problem of querying and data aggregation.

The system architecture consists of a stack containing Apache Spark, Apache Hive, and Apache Hadoop in that order. The architecture is a good framework because having a distributed system with parallel processing capabilities will achieve faster computations to process big data efficiently. The raw data files are stored in the Hadoop File Distributed System (HFDS), and specific tasks can be run on Hive or Spark.

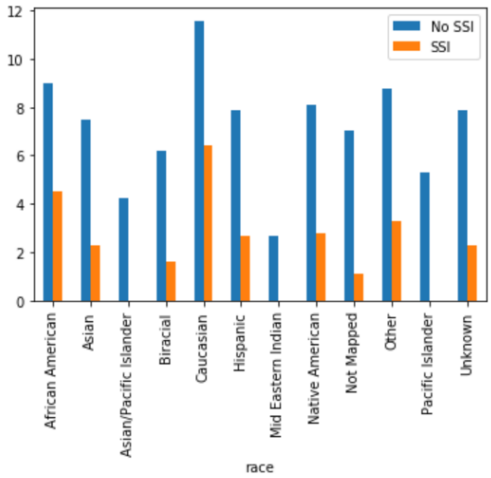


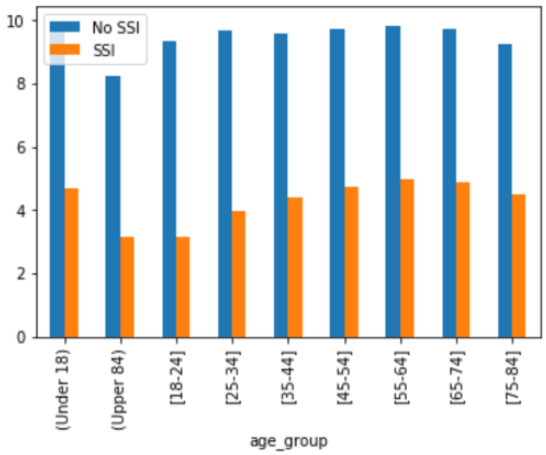
Example of Dataset Schema

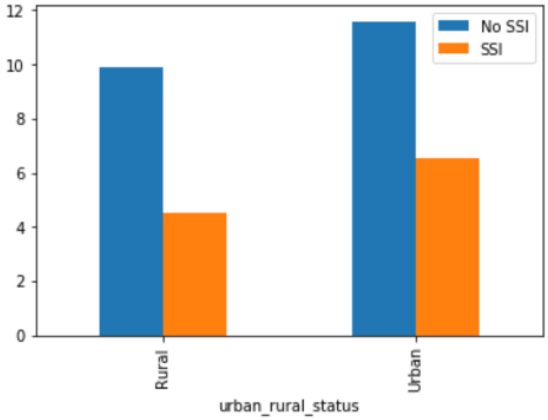
The diagram above shows an example of how the database is modularized. Notice that there are two types of tables. The first type of table ('hf_f_...') is characterized as the facts table. These tables are continually updated by daily visits of patients. Each time there is a recorded patient visit, the hospital's data is added to these tables. The other type of table ('hd_d_...') is characterized as the permanent tables. These tables hold information that is static across all encounters with the hospitals. For example, a patient's information such as age, gender, and patient id would fall under this category as they do not change for every visit.

Since we will not use all the data within the database, we chose to extract and make a separate table aggregating the necessary data from the database. This table included patient information, type of procedures, hospital information, and diagnosis information. During this, it becomes imperative to run specific processing tasks with Hive or Spark. I ran processes in Hive when the data volume was not very high and consequently ran processes on Spark when data volume was higher. I had to do multiple JOINS since Spark is capable of parallel processing. Through the new table, we were able to get subsets of data that we are concerned with.

Conducting a preliminary analysis of our data was necessary for this project since I wanted to see the distribution of data across different potential features for my causal model. I compared the distribution of every relevant feature based on the hospital's location (urban v rural). Many of the visual representations of the data were hard to visualize, so the plots below are log transformations of the raw data.







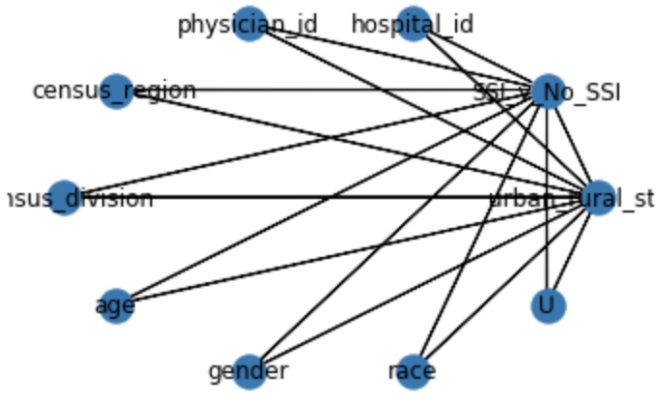
Above are some of the plots that I have constructed to understand the data better. I noticed that in the table that shows the distribution of race between the SSI features, there is no constant number across all different ethnicities. One reason could be due to where the majority of a specific ethnicity resides. However, we cannot make that assumption based on the data given. With the age group distribution, the numbers are relatively constant across the age groups, telling us that age would not be a decisive factor if patients have a surgical infection site since we do not see any discrepancies. The urban and rural distribution validates our assumption that the numbers would be higher since the hospitals located in that area are likely to see more patients. This type of preliminary analysis is helpful since I can view discrepancies that I need to be aware of during the model building and to make sure that my assumptions of the data are also correct.

During the modeling phase of this project, I used a python library called DoWhy. This library was created by Microsoft mainly for causal inference and visualizations of potentials graphs we may build. When it comes to determining the model for this graph, the main issue is not what the features should be but how the relationship is depicted.

I used a structured learning algorithm called the PC Algorithm to create a model. This model estimates the true DAG (direct acyclic graph) based on the attributes given. In the first step of the algorithm, the skeleton of the DAG is estimated. This skeleton consists of a DAG that is undirected. It also has the same edges as the true DAG with no edge orientations. The skeleton DAG is estimated through many iterations. In each iteration, the algorithm will test the constraint

for each edge to ensure there are any conditioning sets. In other words, the algorithm will try to identify if edge ‘a’ and edge ‘c’ are conditionally independent given ‘s’. If this statement is true, then we can delete the edge we are testing. Once this step is done, our skeleton DAG is complete. The second half of the algorithm is to determine which direction the edges are oriented in the DAG. By observing a triplet of nodes through each iteration, we can identify the type of triple it is from the marginal distributions. The triplets can be in the form of chains, colliders, and forks. Once this step is done, the algorithm has estimated the true DAG of the data. For this project, I used a variation of the PC algorithm to identify the outcome and treatment variable for a specific DAG. The outcome variable is the variable that will verify our hypothesis; it is the number of surgical infection sites. The treatment variable is the variable we hope to identify as the bias toward our outcome variable. For my project, the treatment variable is the location of the hospital (urban vs. rural).

After running this algorithm through our dataset and setting the treatment and outcome variable appropriately, we achieve the cause inference model below.



The next step in the research is to identify any causal relationship or effect between the treatment and outcome variables. To compute this number between two variables, we will first have to adjust for confounding variables. A confounding variable is a variable that influences both the treatment and outcome variables. We want to do this to only compute the effect of X on Y that doesn’t include any indirect or direct bias imposed by the confounding variables. Once that is complete, we use the following formula to determine the causal effect:

$$\theta = E(Y1) - E(Y0) = E(Y \mid \text{set } X = 1) - E(Y \mid \text{set } X = 0)$$

After applying this formula to our model using the DoWhy library, we estimate the causal effect between the treatment (Urban v Rural) and the outcome (SSI count) coming to about 0.02. This number tells us a couple of things. First, it tells us that the effect is not 0, thereby saying that there is some significant dependence where the result of the outcome variable is dependent on the treatment method. It also tells us that the effect between the two numbers is not big. The quantity is the first step in supporting our hypothesis. By saying that there is a causal effect between the two variables, it also alludes to the fact the distribution of the outcome variable may change when depending on the distribution of the treatment variable.

The final step in my research project was to determine how the distribution of the outcome variable changes based on the treatment variable since we have now confirmed that

there is a causal relationship between the two. I ran intervention methods through the causal graph to determine this. Usually, we observe the distribution of data through observational data and make claims about the data; however, when trying to determine how the distribution changes, we want to set a variable to a specific value. In other words, we are intervening and setting the variable to a specific value for all data and viewing how the distribution changes based. The following formula represents the intervention formula:

$$P(Y = 1|do(X = 1))$$

The do() represents the variable we are intervening and setting a specific value to. I applied this formula to our dataset twice, first, by setting our treatment variable (Urban_v_Rural) to Rural and another, by setting to Rural. We get two different distributions of the data, and we see how the outcome variable (SSI) is very different between the two.

	Original Dataset	do(Urban_v_Rural = Rural)	do(Urban_v_Rural = Urban)
0	142213	142388	142235
1	1236	1061	1214

From the table above, we see how the distribution changes when we run the intervention method on our treatment variable. More interestingly, we notice that when intervening and setting the treatment variables, the number of SSI's (0 index) increases in both interventions. This conclusion indicates some bias from the treatment variable that is affecting the outcome variable. Furthermore, we can understand this bias by quantifying it as a number. We can find the 'average causal effect' (ACE) by finding the difference between the causal effect for each of the interventions. The formula that I use is:

$$P(Y = 1|do(X = 1)) - P(Y = 1|do(X = 0))$$

From this formula, we calculated an ACE of 0.02. This number is similar to the causal effect number I acquired while building the model, which validated the code I was writing and validating our data. From all this information, I concluded that there is an underlying bias factor that affects the outcome of surgical site infections, specifically for colorectal procedures.

Describe What Success Was Achieved

My goal in this project was to determine if there was any bias toward the outcome of surgical infection sites based on where the hospitals are located. To do this, I first had to create a causal model that depicts the relationships we assume to be true. From that model, I quantified the causal effect between the treatment and outcome variable and saw how the data distribution changed based on intervention methods. From the causal effect being 0.02, we notice some hidden bias from the Urban_v_Rural feature that is somehow affecting the SSI count. We see a significant change in distribution through the intervention methods. Both of these show support to my initial hypothesis that there is some hidden bias affecting the SSI count. If there was no bias, intervention distributions would not have changed that drastically and also, the causal effect

would be much lower. Overall, there was a success in this project since I have substantial evidence to support my hypothesis.

What Challenges Were Faced And How They Were Overcome

The first challenge I encountered in this project was adapting to the big data volume I had to manage. Generally, during a class project, our projects worked with relatively low data volumes, and we wouldn't have to focus that much on reducing run time and optimizing code. However, since I was working with a huge database, I had to figure out how to optimize the tasks. I overcame this problem by understanding how the system architecture works. I notice that tasks that are run within Spark are executed in a parallel fashion while the Hive's tasks are not. Because of this, it became imperative to run specific processing tasks with Hive or Spark. I ran processes in Hive when the data volume was not very high and consequently ran processes on Spark when data volume was higher. I had to do multiple JOINS since Spark is capable of parallel processing.

Another challenge I faced was visualizing the data properly. Although I knew how to visualize the data using graphical libraries, I wasn't sure how the data should be visualized or what am I looking for in these visualizations. To figure this out, I went back to the problem I was trying to solve and decided what type of patterns I wanted to find out. Specifically, I wanted to see if there were any patterns between hospitals based on the location of any of the factors. This type of information would help me decide if there's any substance to my hypothesis or if I needed to change that. I also wanted to make sure that my assumptions about the data were correct. Those essential things were what I needed to validate in my visual analysis to continue my research. From that understanding, I was able to visualize the patterns I was looking for correctly. Lastly, my last challenge was creating the causal model.

Although I understood the algorithm from a theoretical perspective, coding the algorithm was difficult. To be more precise, understanding the algorithm's probabilities and putting it into code was challenging. I wasn't quite sure how to incorporate determining if two factors are conditionally independent based on a third factor into code. I overcame this challenge by doing the probabilities by hand and understanding the math behind it. I then realized that it is mostly summations and comparing probabilities to determine independence. Once I could translate probabilities into code, I was able to write the algorithm and create a causal model.

Systems, Languages, Tools Used, Etc.

All the code that I wrote for data processing, machine learning algorithms, and data analysis were written in Python. I mainly used libraries such as pyspark and pandas to aggregate data from the data source. I used matplotlib to create plots for my preliminary analysis. I also used dowhy and sci-kit learn as the main libraries to develop my causal model and run causal inferences on the model.

Specific, Individual Contributions Made To The Project

For this project, all of the code was written by me. I consulted my peers within the research to help debug my code and help me understand causal modeling. Specifically, I wrote code to create a cohort of data from the database to work with only the data we needed. The code

took a while to complete since I was working with big data and how to optimize my tasks between Hive and Spark. I also wrote the code during the data analysis phase of the project. I used multiple python libraries to help visualize the data and consulted my research to get input on what type of data analysis I should be looking for. I used the dowhy library to write my causal modeling code. This library helped accelerate the process since I didn't have to code everything from scratch and could use the helper functions that were in the library.

Courses Taken Which Are Most Relevant And Why

One of the classes that greatly helped with this project was CS 471, specifically the topic of Bayesian Networks. Since I had that knowledge beforehand, I had a strong foundation of graph representation and how probability plays a role in determining causality. Another class that aided me in this project was CS 242. This class broke down the steps of the data science process and explained the importance of each step. More importantly, I understood why data aggregation has such a big role in the process and also is the longest step in the process. I also learned the importance of the last step of the data science process - deriving conclusions from your results. From the project, I had to understand the model and what the numbers were inferring to conclude, which is a new skill for me.

Summary Of Technical Skills Learned

I learned SQL optimization techniques specific to distributed systems to reduce run time. One example of this is when to partition data or broadcast certain tables to reduce the runtime of tasks. Through this project, I also learned quite a lot about different causal libraries. I had to understand each one and decide which library was best for my project. Lastly, learning how to visualize data in a way to understand better what I was looking at really helped me a lot through this project. I think visual data analysis is a key tool for data scientists and something I correctly learned how to do through libraries such as matplotlib and seaborn. Overall, through this project, I understood the importance of gathering data and how important it is to validate your data before moving on to the project. I gained a lot of knowledge about Bayesian and causal graphs and how to identify causal relationships between two variables. More importantly, as an aspiring data scientist, I worked through the full data science life cycle and thus was able to see how important each step is. Through this project, I understood how large of a role probability can play in causal models, especially in understanding how the PC algorithm works. Finally, one of the most important things I learned was how to aggregate data from a large data source. This semester, I learned how to estimate a causal model from observational data and determine whether any significant causal relationships may be a bias towards the outcome variable.