Springboard - Data Science

Capstone Project 3

Causal Structure and Causal Inquiries for Health Features in the United States Population

Hiram G. Menendez

June 2023

1 Introduction

1.1 Problem

One of the goals of data science is to build mathematical models from data that can make predictions about the values of features in new, unseen data. Traditional classification algorithms such as linear regression and decision trees can accomplish this by discovering correlations among features in the data.

If these algorithms discover that there exists a correlation between two variables (or features), we cannot conclude that there is necessarily a causal link between the two features, e.g., if there is a correlation between the variables “smoking” and “heart disease”, we cannot conclude that the former “causes” the latter. In fact, when drawing conclusions solely from the data, it is equally plausible that having heart disease “causes” people to want to take up smoking. In other words, these traditional classification algorithms are not causal models.

The goal of this project is to use data about the prevalence of various health features in a sample of the U.S. population to develop causal models that might be able to discover causal links between the features, also referred to as the causal structure of the data. Another goal is to use these models to answer “causal inquiries” – probabilistic questions about the data given that we make an “intervention” on one or more of the features.

1.2 Relevance

Depending on the goals of the data scientist, knowing whether there is a causal link between features might not be useful. For example, if the goal is simply to predict the value of a feature in unseen data, then the causal structure of the data is irrelevant. For other goals, however, this may be essential. For example, other than predicting whether an individual has heart disease, the goal might be to decide what sort of treatments can lead to its prevention. We might ask ourselves whether the risk of developing heart disease decreases if the patient quits smoking. If smoking causes heart disease, then the answer would be yes. If, however, the observed correlation is not due to a causal relationship, then the answer is no. In situations similar to this, causal models would be of immense value.

1.3 Summary of Results

In this project, two different algorithms were used to build Bayesian Network models: Hill Climb Search and Mmhc (max-min hill-climbing). In a Bayesian Network, the features of the data are represented as nodes in a graph called a directed acyclic graph (DAG). Every node has an associated table CPD, or table conditional probability distributions, that contains the probabilities for each of the values of the feature conditional on each of the values of the parents in the graph.

First, the models were used as classifiers to predict whether individuals have heart disease; the results are shown in Figures 1-4 and 8-11. For Hill Climb Search, the recall and roc auc scores were, respectively, 0.81 and 0.83, while for Mmhc these scores were 0.80 and 0.83. These results show that these Bayesian Network models do a very decent job in classification tasks. The scores were then compared to ones from another, non-causal model built with the XGBoost algorithm that had scores of 0.81 and 0.84. The fact that the scores of the causal models were comparable to this non-causal model was unexpected.

Next, the two models' performance was accessed using structure scores. The results are summarized in Figure 18, which shows that, for each of the three structure scores used (Bdeu, K2, and Bic), the model built with the Mmhc algorithm scored marginally better.

The algorithms also generated the DAGs associated with the data, and they describe the causal structure of the data; these can be seen in Figures 5 and 12. We will later explain in Section 8 why there is no guarantee that these DAGs represent the correct causal structure of the data despite the fact that the algorithms do the best they can to find the graph that best represents the data.

We then proceeded to use both models to make causal inferences. Some of the results are shown in Figures 19-21 where each bar in the plot represents the difference of the probability of a feature having a value when there is an intervention and when there is not. In a medical setting, these results can be used by physicians to identify how to intervene in the lifestyle of a patient to improve his or her health. The rest of the numerous bar plots can be found in the project's jupyter notebook file.

2 Approach

2.1 Summary of Approach

We used the same dataset used in our previous project, which is described in detail in the next section. In order to simplify the model, we only used ten features from the data. This data was used as input for algorithms that built Bayesian Network models in which the features of the data are represented by DAGs which describe the possible causal structure of the data. The models also have a joint probability distribution for the features that can be fit from the data. The properties of these models will be thoroughly explained in Section 3.

Since the models can calculate probabilities, they can also be used as classifiers. In Section 4, Bayesian Network models were built and used to classify individuals in the dataset on whether they have heart disease or not. The results were then compared to the results of a non-causal model built with the XGBoost algorithm.

There is one more kind of probabilistic inquiry that can be made with these types of models: causal inquiries. In a causal inquiry, we find the probability of an event after doing an “intervention” in one or more of the features. The concept of performing an intervention will be explained in Section 3. For every value of every feature in the model, and for every value of every feature that we can intervene on, we calculated the probability of the feature of interest. For example, one of the inquiries calculated the probability that an individual has good health after intervening so that the individual has heart disease. After calculating these interventional probabilities, we took the difference between the probability without and with the intervention.

2.2 Description of the Data

The raw data used in this project consists of a survey administered by the CDC on a sample of the U.S. population called the Behavioral Risk Factor Surveillance System (BRFSS). It asks the respondents a large number of health-related questions, such as “have you ever been told by a medical practitioner that you have heart disease” or “how many days of the week, on average, do you have more than one alcoholic beverage”. Some questions, like the former, have categorical answers (e.g. “yes” or “no”) while others, like the latter, are numeric (e.g. “3 days”, “4 days”, etc.). The survey asks hundreds of questions, but we made use of only ten of them in the model in order to avoid having a complicated model.

The criteria used to choose them were the results from the computed average SHAP values of the features in the previous project for the prediction of heart disease. The average SHAP values are a measure of the “feature importance” in predicting the target feature. We chose the top ten features from the previous project that showed to be the “most important” in predicting heart disease: good health, hypertension, high cholesterol, smoker status, age category, diabetes, sodium, heavy drinker, heart disease, and sex. All of them are categorical and most data instances have values for them of either “1” (has the condition) or “2” (does not have the condition). Other less common values are numbers representing categories such as “refused to answer” or “does not know”. The possible values for the feature age category are numbers representing different age groups. The possible numeric values for the smoker feature represent different categories: smokes every day, smokes some days, former smoker, and never smoked.

3 Modeling

3.1 Introduction to Bayesian Networks

The models we built in this project are called Bayesian Networks. Each model has an associated directed acyclic graph (DAG), which is a graph made up of nodes connected by directed edges (or arrows) in such a way that, if you travel along a directed path starting from a node, you will not be able to reach that starting node (it has no cycles and is therefore called “acyclic”). When an arrow points from one node to another, the one at the beginning of the arrow is called a “parent” node and the one at the end is called a “child” node.

The graph is used to represent the existence of certain conditional independence relations in the data. Specifically, if we condition on parent variables, the child node variable will be independent of all variables that are not its descendants. In other words, once we have knowledge of the values of all of the parent variables, having knowledge of the other variables that are not descendants of the child node will not affect the value of its probability. This intuitively corresponds to the idea that only the parent variables “directly interact” with the child node.

Each node in the DAG has a table CPD (conditional probability distribution), which is a table of probabilities for each of the values of the variable conditional on each of the values of the parent nodes. Therefore, once the table CPD of each node is found, the model contains a joint probability distribution of the features.

This idea of the parent nodes “directly interacting” with the child node can also be given a causal interpretation: the variables of the parent nodes are the “causes” of the child node. This means that the random outcomes are generated by a mechanism such that, when a parent node randomly acquires a value, it sets or “causes” the child node to have a particular distribution. Therefore, if we are somehow able to intervene in the event-generating process by first manually setting the value of a parent variable before generating the random outcomes, the distribution of the child node variable will be different than the one it would have had if there was no intervention. Contrast this to how the distribution of a variable behaves if we instead intervene on another variable that does not have a “directed, connected path” to it in the DAG. In this case, after we intervene on that variable, the distribution of the variable of interest does not change.

Note, however, that the concept of conditional independence is distinct from the concept of causality. The DAG might correctly represent the set of conditional independencies in the data, but it might not necessarily represent the correct causal structure. This is because, given a set of conditional independencies, there is likely more than one DAG that can represent them. Therefore, observational data on its own is not generally capable of unambiguously determining the DAG representing the correct causal structure. This is because the concept of causality is related to how variables behave under interventions, not under observations. Therefore, when the algorithms used in this project find a DAG, it should be viewed as representing one possible causal structure that the data could have.

This is one of the reasons that it is important to supplement studies that aim to find causal connections in data with expert, domain knowledge. This can constrain the search for the graphs to exclude causal links that are known to not be there. For example, if it is known in the medical community that smoking is not a cause of diabetes, this can be manually excluded from the resulting model.

Since Bayesian Network models have a joint probability distribution for their variables, any kind of probabilistic inquiry can be made using it. The models would allow you to, for example, calculate the probability that an individual in the population is a smoker. More generally, for a given number of features, the model is able to calculate the probability of each of the different configurations of them conditional on values for others and marginalized over others. For example, the model would be able to calculate the probability that an individual is both a smoker and has heart disease given that we know that he or she has diabetes but not high cholesterol while marginalizing the rest of the features.

There is one more kind of probabilistic inquiry that can be made with these types of models: causal inquiries. Since the DAG of the trained model can be interpreted as a graph that shows causal relationships between the features, we can calculate probabilities under interventions. When an intervention is done, the underlying joint probability distribution of the model is changed so that the features that were intervened on attain a specified value with probability one. This new probability distribution will be represented by a DAG in which the probability distribution of the nodes that were intervened on is a set value that occurs with probability one, and all of the arrows pointing from the parents of these intervened features are removed. They are removed because, once we intervene on a node, their parents can no longer influence them. When doing these kinds of inquiries you also have the option of conditioning on other features.

An example of a causal inquiry would be as follows: calculate the probability that an individual has heart disease but does not have diabetes assuming that we intervene such that the individual is not a smoker but has hypertension (interventions), and given that we know that the individual is male and is young (conditioning on).

3.2 Description of the Models

The model-building process occurs in two steps. First, a graph search algorithm is used, which finds a DAG that accurately represents the conditional independencies found in the data. Second, the same data is used to fit the parameters of the table CPDs at each node. There are three main types of such algorithms: score-based, constraint-based, and hybrid.

3.2.1 Types of graph search algorithms

In a score-based algorithm, a score function is used that can compute a score for a given graph that reflects how well the joint probability distribution implied by the graph can reproduce the data. One method of calculating such a score is by, given some Bayesian prior probability on the space of all possible DAGS, calculating the posterior probability for the given graph given the data (in practice, the marginal likelihood found in the denominator in Baye’s Theorem is dropped since it is the same for all graphs). A prior probability must also be given for the parameters of the table CPDs on each node. The name of the scoring function is based on the prior that is used. Some famous score functions are Bdeu, K2, and Bic. Once a scoring score is chosen, a search method is chosen that dictates how to transverse the space of possible DAGs, scoring each graph along the way. The algorithm terminates when a graph is found that maximizes the score.

In a constraint-based approach, hypothesis tests of independence (such as, for example, the Chi-Square test for independence) are performed, one by one, on different combinations of variables. An example of such a test would be testing whether the variable diabetes is independent of the variable smoker while conditioning on the variables hypertension and age category. Since the way that nodes are connected in a graph gives rise to conditional independencies among the variables, the results of these tests inform how to build the graph.

In a hybrid approach, elements from the two other methods mentioned previously are used. First, a constraint-based approach is used to build the “skeleton” of the graph, which is the graph with the same nodes and edges as the final DAG but with all edges unoriented. Once the skeleton is found, a graph scoring function is used to decide the orientations of the edges.

We set out to use three different algorithms to build a Bayesian Network Model: Hill Climb Search, PC, and Mmhc.

3.2.2 Graph Search Algorithm Used

3.2.2.1 Hill Climb Search

This is a score-based algorithm that starts at a given DAG in the space of possible DAGs and traverses the space by doing a simple graph operation to the current graph, such as adding an edge, removing an edge, or reversing an edge. The algorithm calculates the score of each allowed step and applies the one with the largest score. The algorithm proceeds in “climbing the hill” until it reaches a graph where the score cannot be improved – the “top of the hill”.

Some of the most important parameters of the algorithm are the starting graph, the number of max iterations or “steps” in the search, the graph scoring function used, and the so-called tabu length. This last parameter is an integer “n” that tells the algorithm that the last “n” graphs found should not be revisited. The purpose of this is to promote a wider search from the starting point, and it can prevent the algorithm from accidentally settling in a local maximum. Two other important parameters are the so-called white list and black list. The former is a list of edges that are required to be in the final graph, while the latter is a list of edges that are not allowed to be in the final graph.

3.2.2.2 PC

The PC algorithm is a constraint-based algorithm. One important advantage of this algorithm is that it can produce two different types of graph outputs: a proper DAG, or a so-called class PDAG (partially directed acyclic graph). The former is a graph where only some of the edges are directed, and it is meant to represent not a single graph, but an equivalence class of graphs such that each of its members correctly represents the conditional independencies found in the data. If an edge in the class PDAG is directed, it means it is so in every member of the equivalence class. Otherwise, it is undirected. In this way, we end up with a collection of graphs such that each is a candidate for representing the causal structure of the data. The undirected edges can then be oriented using another method, such as domain knowledge.

Unfortunately, we ran into problems when using this algorithm because, regardless of the type of graph output that was chosen, the algorithm would output a graph that had one cycle and, therefore, is not a proper DAG. This meant that it could not be used as a model for making predictions. The reasons for this behavior are still unknown at the time of this writing but will be further investigated in a future project.

3.2.2.3 Mmhc

The third model is a hybrid type, which means that it uses methods from both of the previous algorithms. First, the skeleton of the graph is found with the PC algorithm, which is the graph where all of the edges are unoriented. Then, a graph score function is used to orient the edges in such a way that the score is maximized.

4 Using Bayesian Networks as Classifiers

Once a Bayesian Network model is trained, it contains a joint probability distribution with parameters fit from the data. Since we can use this to calculate probabilities, we can use these models as classifiers. For the purposes of this project, we decided to predict the class of the heart disease feature.

The first step was to split the data into a training and test set; the test set was chosen to be 20% of the data. The next step was to use a resampling technique since we are dealing with an imbalanced dataset; only about 8% of the data samples are from individuals that have heart disease. The sampling method used was random undersampling, where the entire data get resampled in such a way that the majority class gets sampled less often, ending with a final sample in which the two classes are about equal in size. This resampled data was used as input for the graph search algorithms Hill Climb Search and Mmhc. As discussed previously, we could not build a model using the PC algorithm.

The values used for the parameters starting graph, number of max iterations, graph scoring function, and tabu length were, respectively, the empty graph, one million iterations, the Bdeu score, and 100. Refer to Section 3.2.2.1 for details about the meaning of these parameters. For the black list parameter, we used a list of edges such that the features “sex” and “age category” will not have parents since it is intuitive that the other features do not “cause” them. After the graphs are found, the very same data is used for fitting the parameters of the table CPDs of each node. The parameterization used for the table CPDs was the Bdeu prior, which is the default option.

4.1 Results

For each of the models used, the results are summarized in four illustrations: the roc curve, the precision-recall curve, the classification report, and the confusion matrix. As discussed previously, we could not build a model using the PC algorithm, so we instead show the two output graphs obtained from the algorithm when using the graph output parameters “dag” and “cpdag”. We can see from them that each contains one cycle, which is not allowed in Bayesian Network models.

The results of the other two models show that they do a good job in the classification task. In particular, they both get good scores in their roc auc and recall scores. Their precision scores, on the other hand, are not very good. However, from the point of view of practical applications, this is not too much of a big problem because it is the recall score that we desire to maximize since we wish to minimize false negatives.

4.1.1 Hill Climb Search

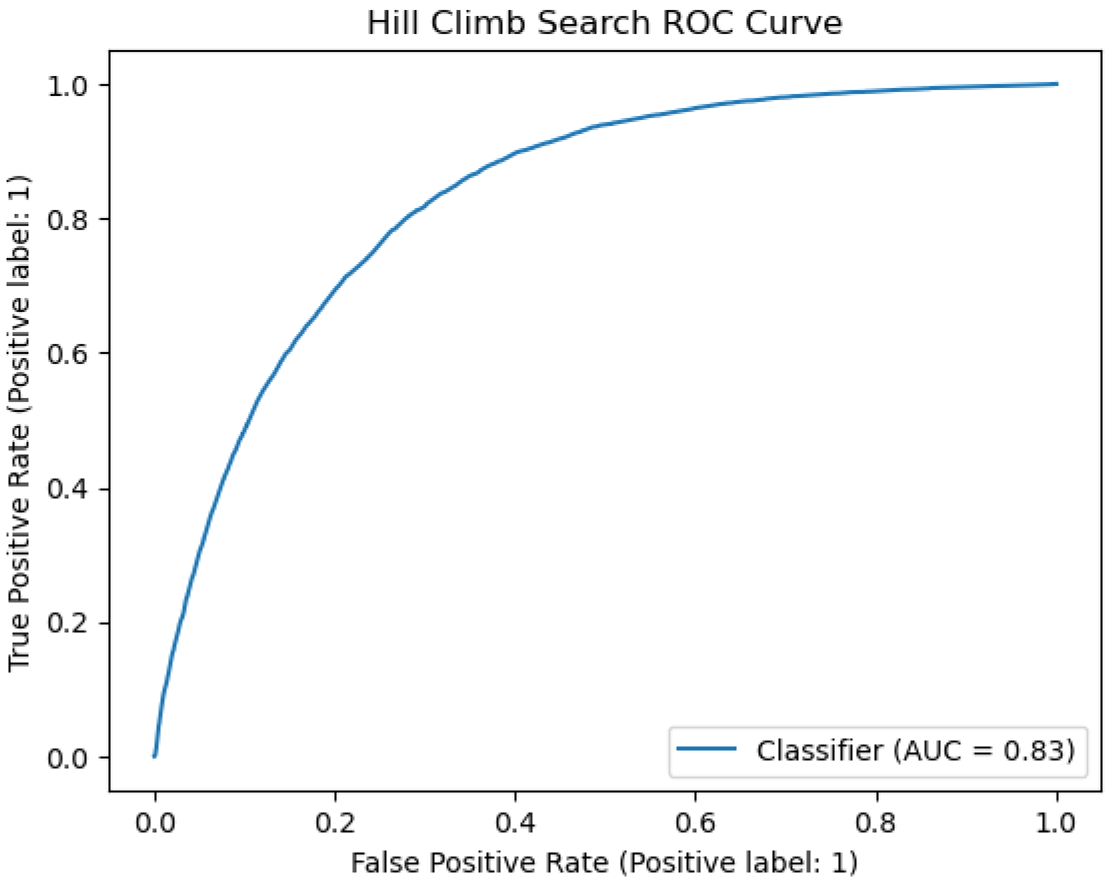


Figure 1: ROC curve for Hill Climb Search algorithm

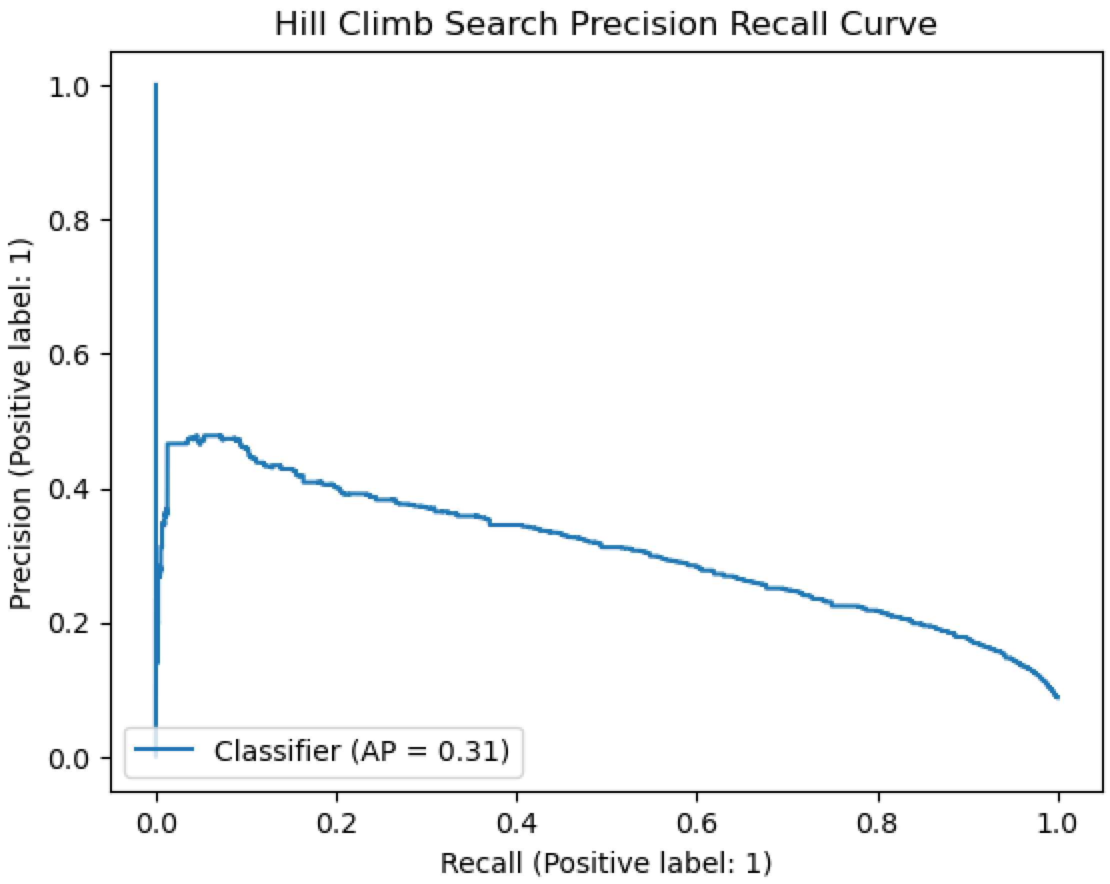
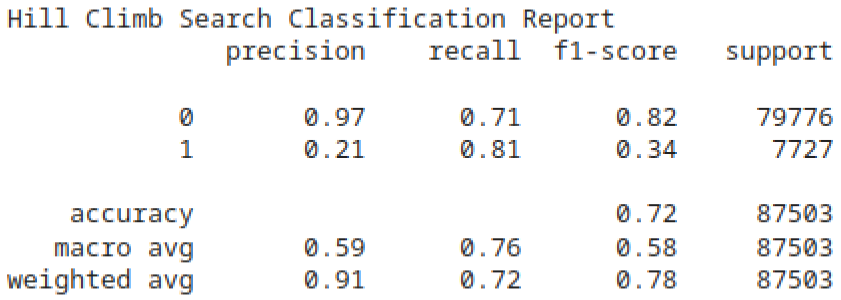


Figure 2: Hill Climb Search Precision-Recall Curve



```
Hill Climb Search Classification Report
              precision    recall  f1-score   support

           0       0.97      0.71      0.82     79776
           1       0.21      0.81      0.34      7727

    accuracy                           0.72     87503
   macro avg       0.59      0.76      0.58     87503
weighted avg       0.91      0.72      0.78     87503
```

Figure 3: Hill Climb Search Classification Report

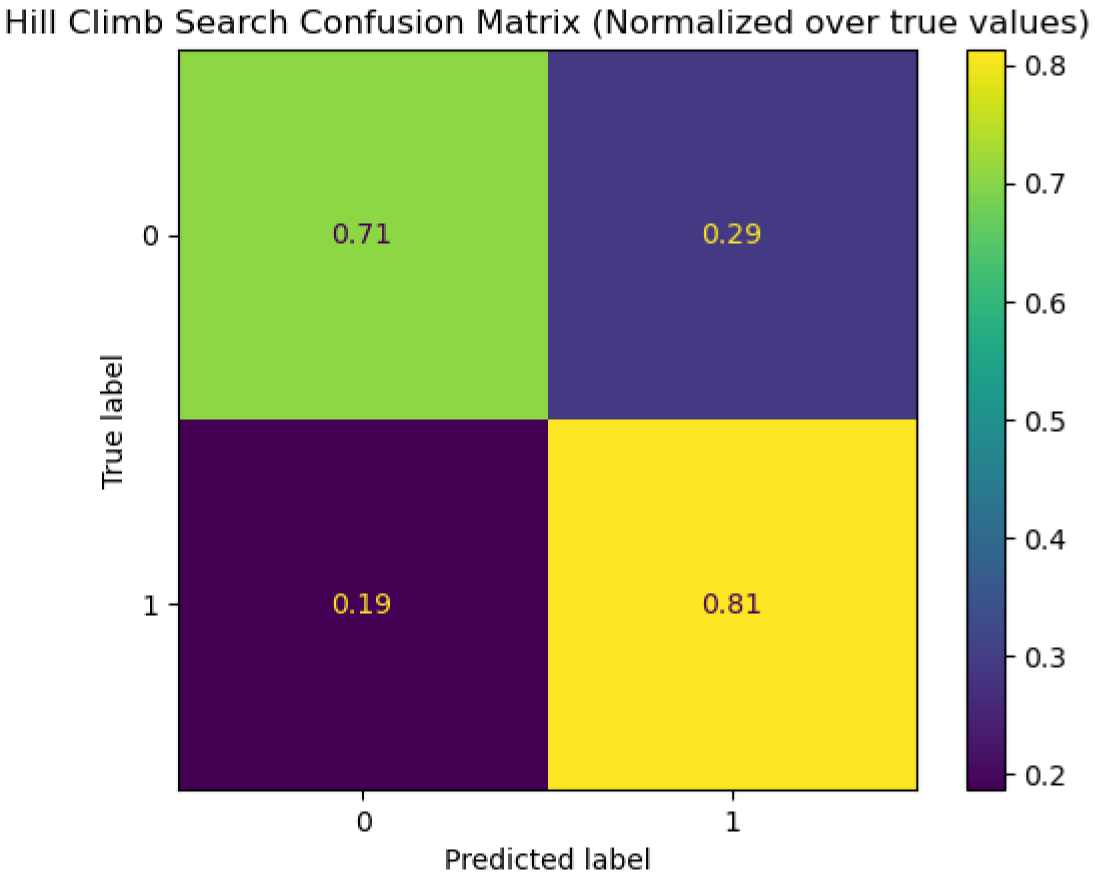


Figure 4: Hill Climb Search Confusion Matrix

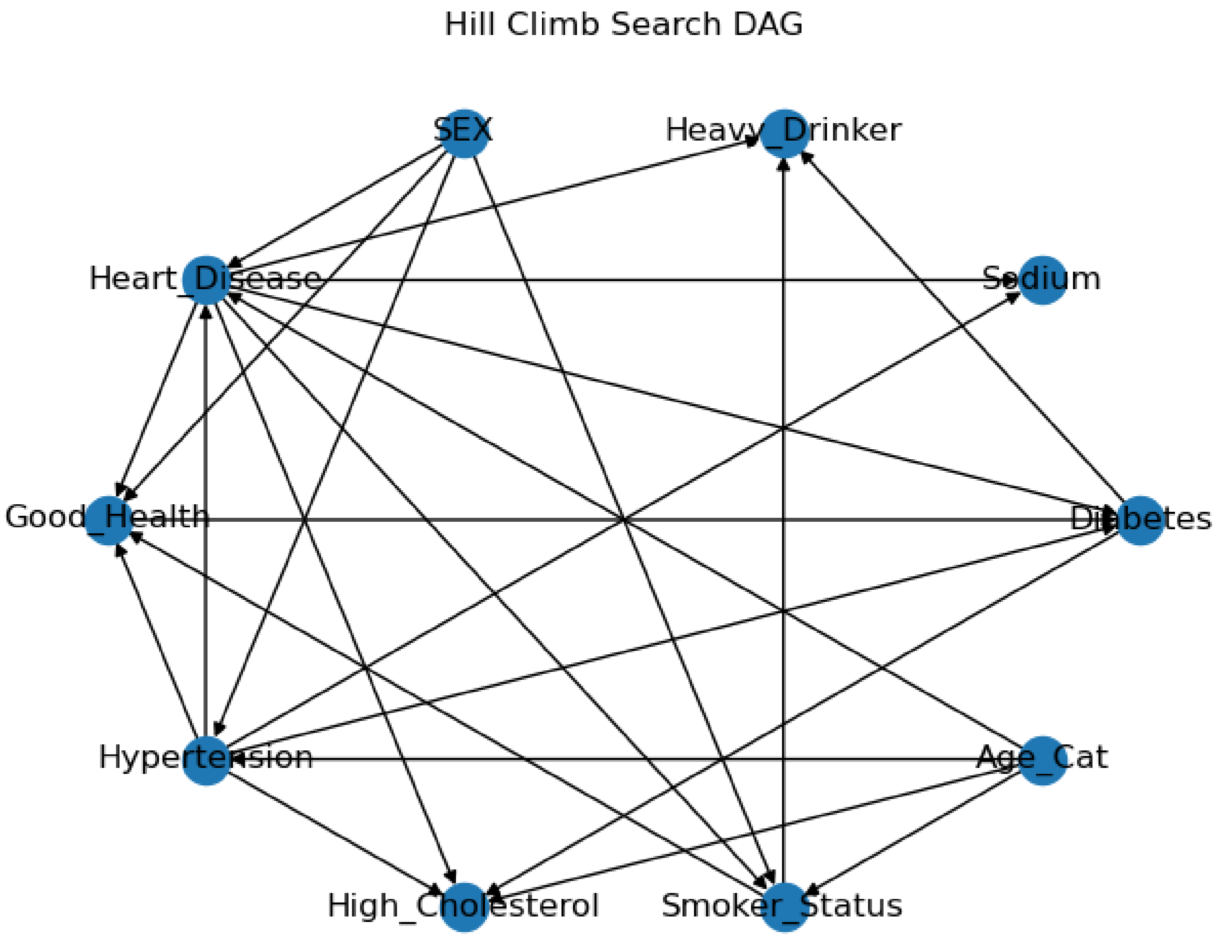


Figure 5: Hill Climn Search DAG

4.1.2 PC

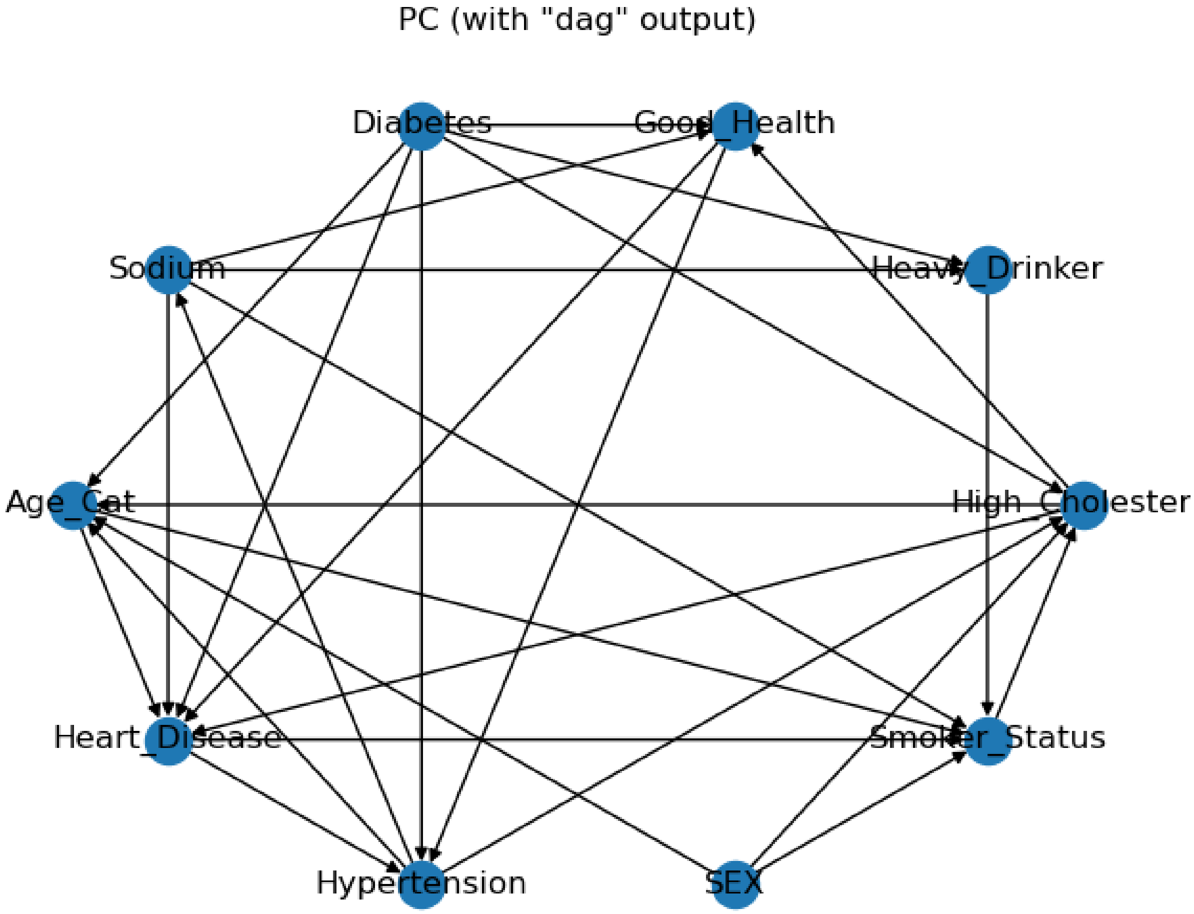


Figure 6: PC algorithm DAG (with “dag” output flag)

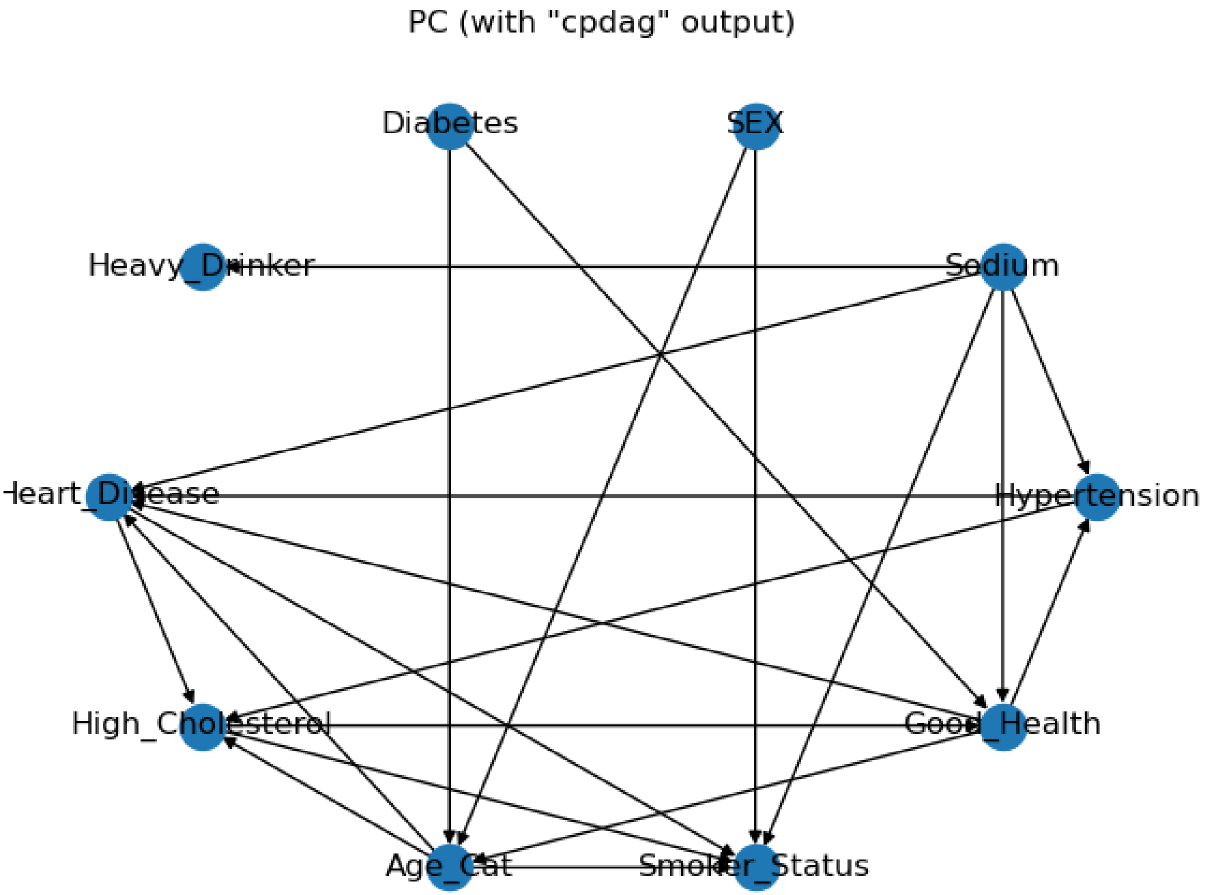


Figure 7: PC algorithm DAG (with “cpdag” output flag)

4.1.3 Max-Min Hill Climb Search

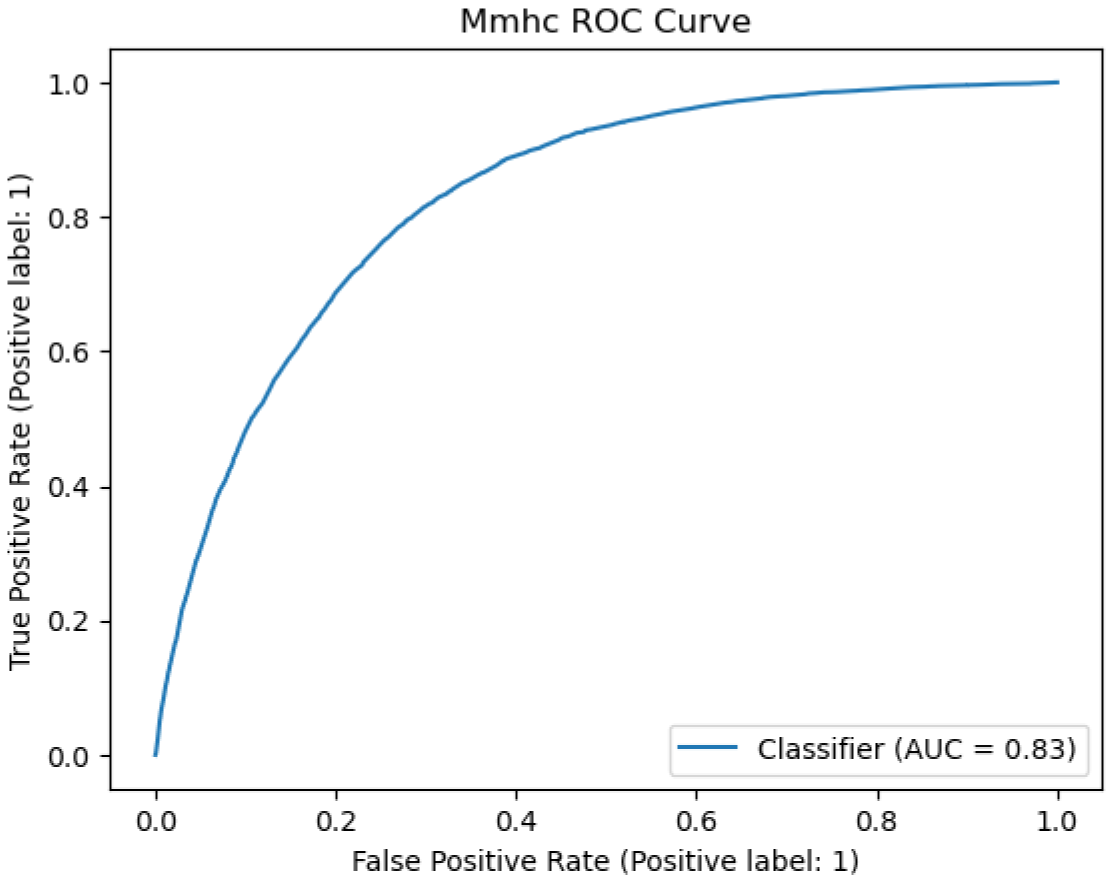


Figure 8: ROC curve for Mmhc algorithm

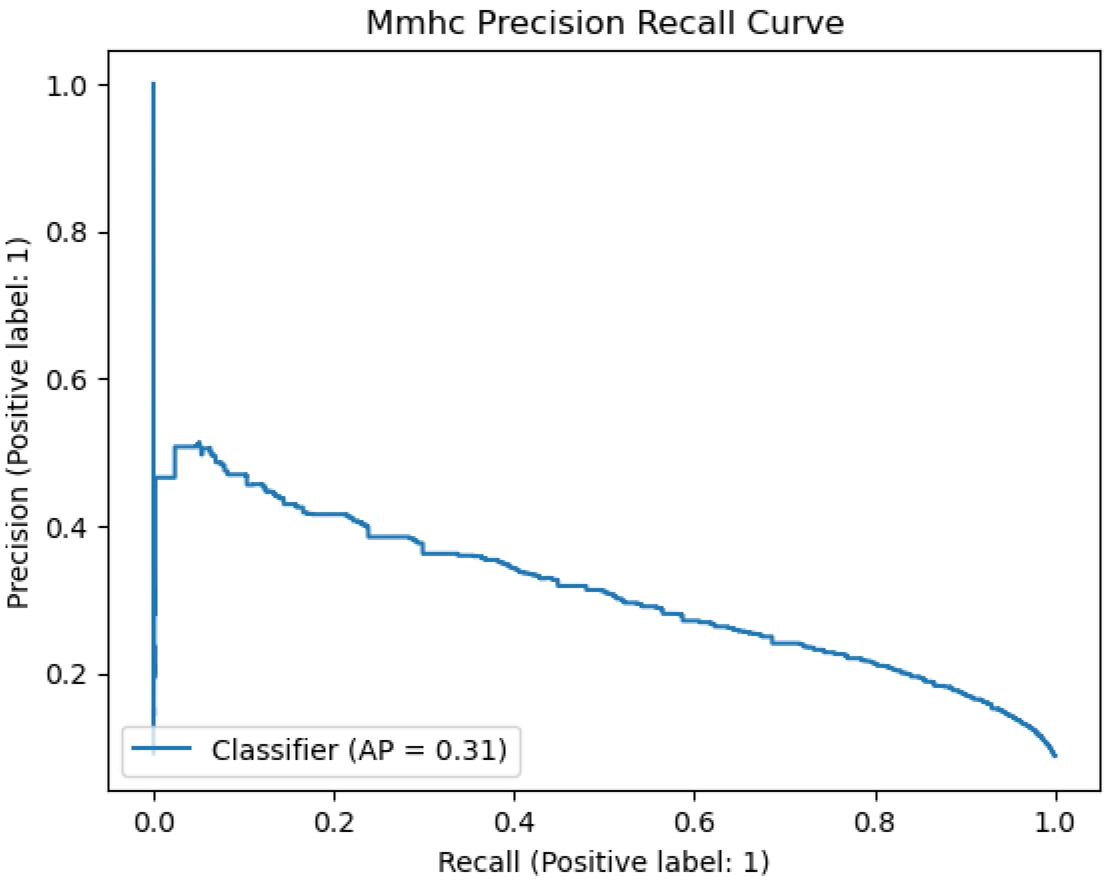
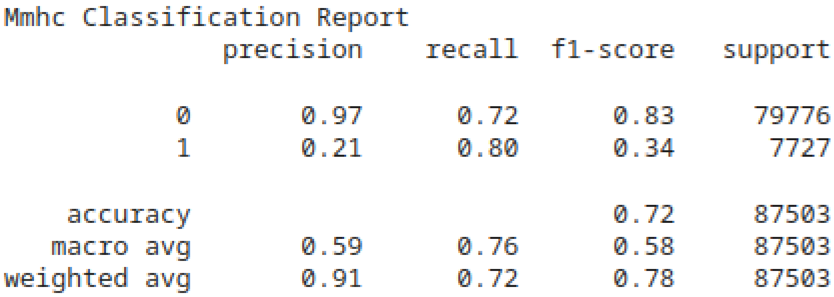


Figure 9: Precision-Recal Curve for Mmhc algorithm



```
Mmhc Classification Report
              precision    recall  f1-score   support

           0       0.97      0.72      0.83     79776
           1       0.21      0.80      0.34      7727

    accuracy                           0.72     87503
   macro avg       0.59      0.76      0.58     87503
weighted avg       0.91      0.72      0.78     87503
```

Figure 10: Classification Report for Mmhc algorithm

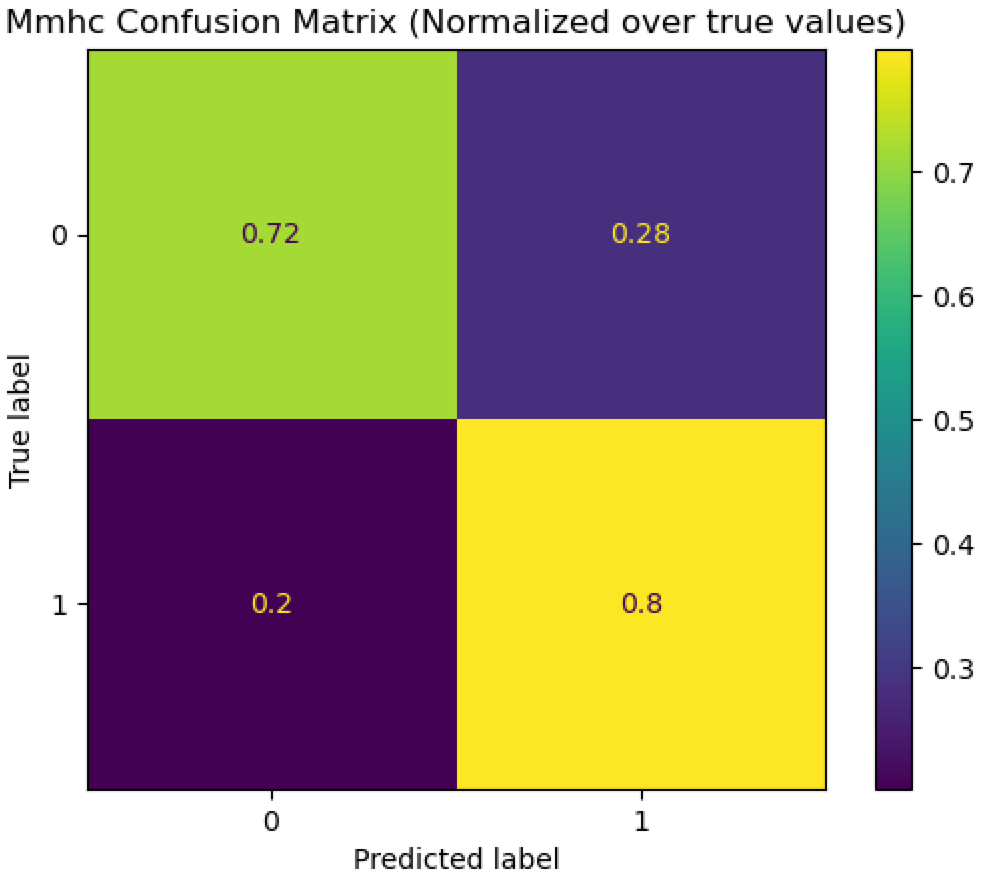


Figure 11: Confusion Matrix for Mmhc algorithm

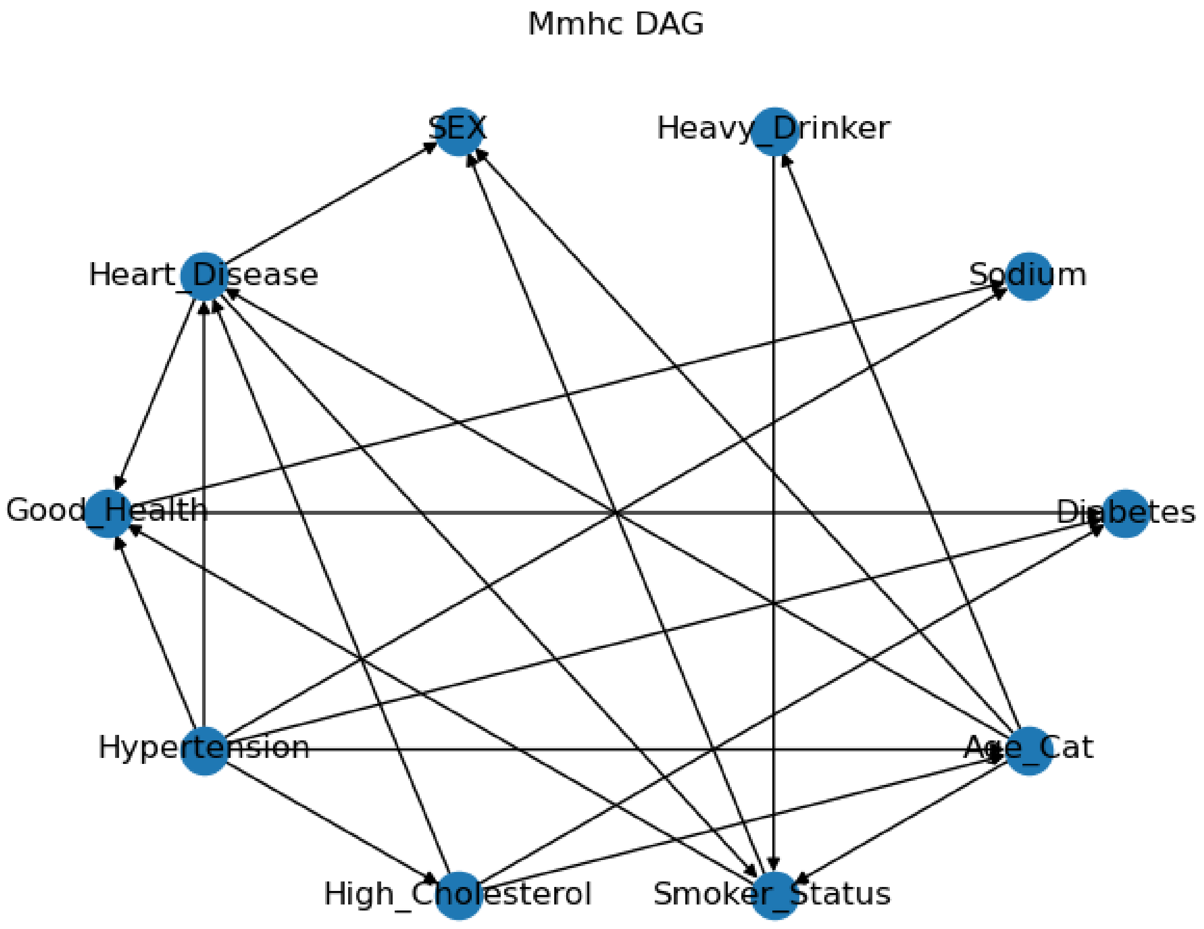


Figure 12: Mmhc algorithm Dag

4.2 Comparison with Non-Causal Model

We were interested in comparing the performance in the classification task of our built Bayesian Network models with a more traditional classification algorithm. We chose to compare it to a model built with the XGBoost algorithm because it was the one that achieved the best performance in our previous project. One noticeable result is that the performance of the Bayesian Network models was comparable with the XGBoost model, which is surprising because classification algorithms such as XGBoost rely on minimizing the classification error to build the model, while Bayesian Network models do not.

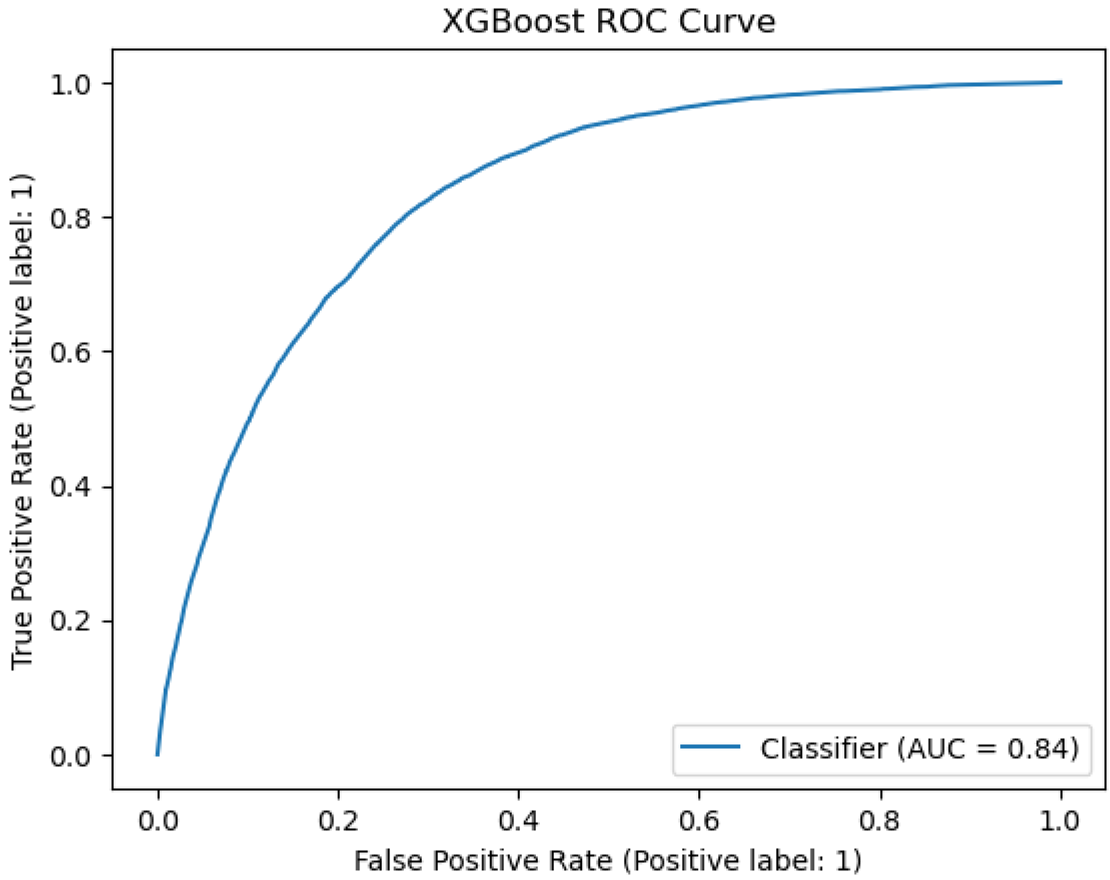


Figure 13: ROC Curve for XGBoost algorithm

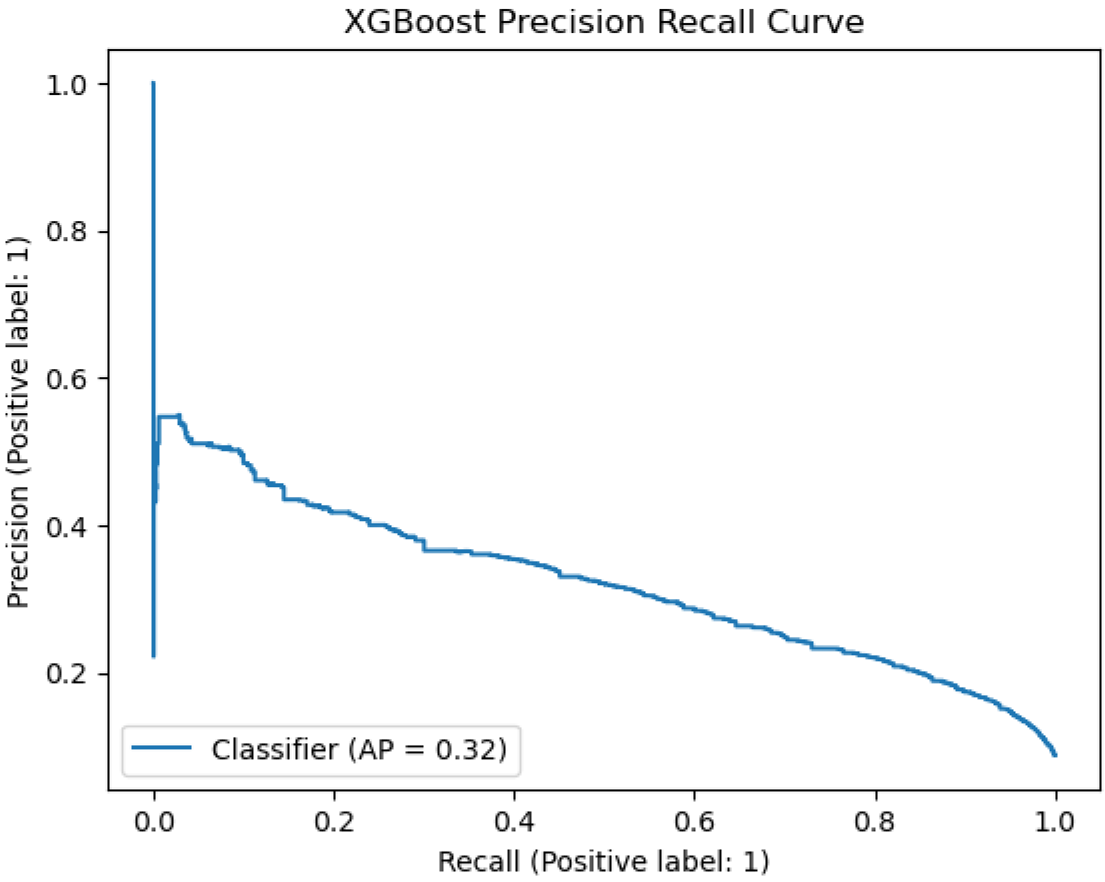
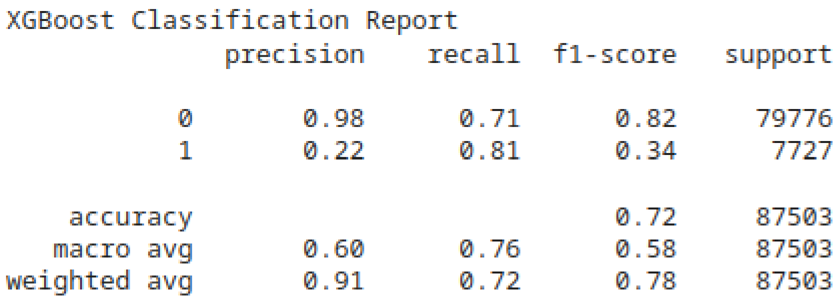


Figure 14: Precision-Recall Curve for XGBoost algorithm



```
XGBoost Classification Report
              precision    recall  f1-score   support

           0       0.98      0.71      0.82     79776
           1       0.22      0.81      0.34      7727

    accuracy                           0.72     87503
   macro avg       0.60      0.76      0.58     87503
weighted avg       0.91      0.72      0.78     87503
```

Figure 15: Classification Report for XGBoost algorithm

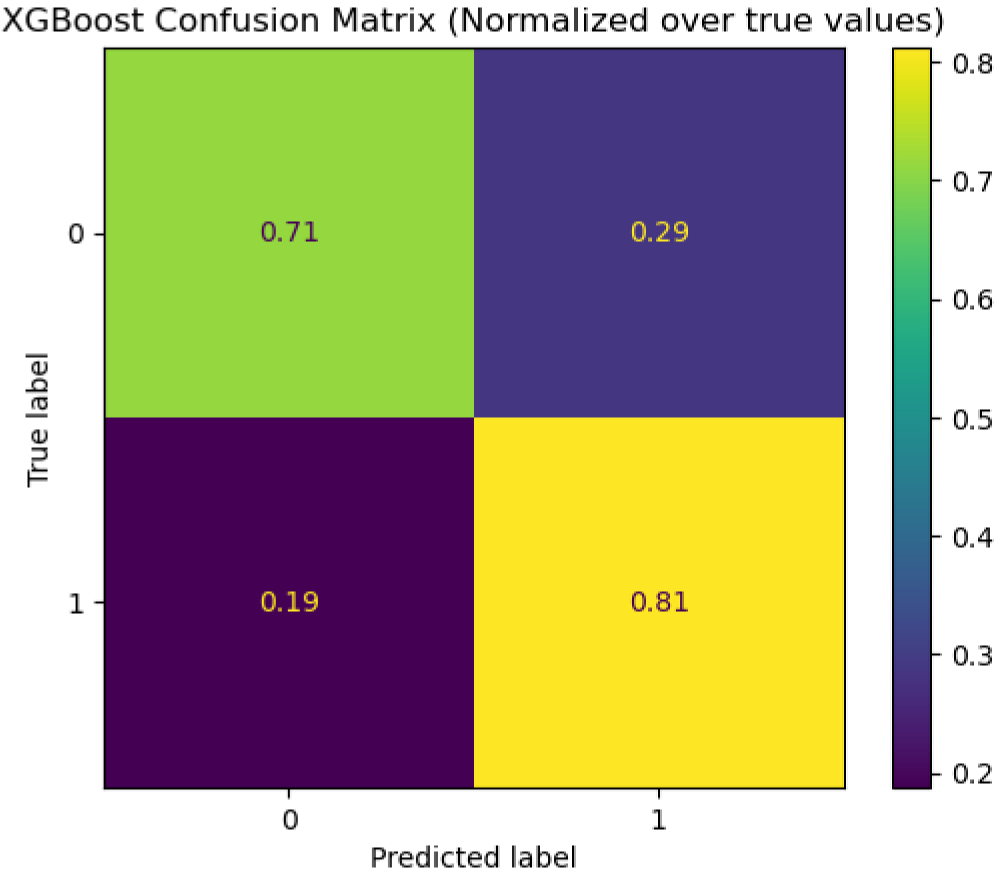
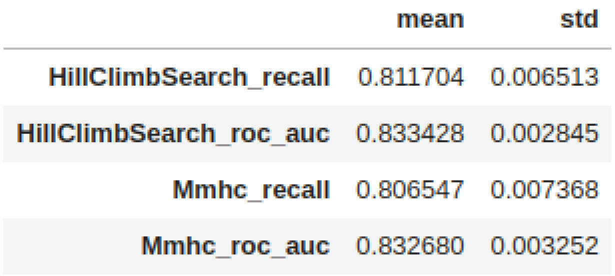


Figure 16: Confusion Matrix for XGBoost algorithm

5 Cross-Validating the Classifiers

In the next part of this project, we applied five-fold cross-validation to the models to verify whether they are overfitting for the classification task. The results are shown in the following figure.



	mean	std
HillClimbSearch_recall	0.811704	0.006513
HillClimbSearch_roc_auc	0.833428	0.002845
Mmhc_recall	0.806547	0.007368
Mmhc_roc_auc	0.832680	0.003252

Figure 17: Results of cross-validation when the models were used to classify the heart disease class.

We used two classification scores: recall and roc auc. As can be seen from the scores' low values of the standard deviation, there is no evidence to suggest that the models are overfitting in classification tasks, despite the fact that classification is not the main motivation for using Bayesian Network models.

6 Bayesian Network Structure Scores

In this part of the project, we evaluated the performance of the Bayesian Network models by using a different metric: structure scores. Unlike the metrics used in the previous chapter, it does not rely on the concept of classification. Instead, the graphs are evaluated based on how well their implied probability distribution functions can reproduce the underlying data.

The way the graphs are scored is as follows. First, a probability prior is chosen for the space of possible graphs (a uniform prior is typically used). Then, a prior is also chosen for the table CPDs at each node. Possible options for this prior are the Bdeu, K2, and Bic; the choice of prior is what gives the score its name. Once a prior is chosen, the posterior probability of the graph given the data is calculated marginalized over the parameters of the priors of each of the nodes. Since the marginal probability found in the denominator of Baye's Theorem will be the same for all graphs, it is usually dropped. The end result of this calculation is the structure score. Therefore, these structure scores are proportional to how probable is the graph to be the correct one given a prior for the space of graphs and for the parameters of the CPDs. The larger these scores, the better the model is deemed to be.

6.1 Results

The following bar graphs summarize the structure scores found for each of the two models and for each of the three structure scores. The higher the score, the better the score. However, since the scores are negative and it is the absolute values of the scores are shown on the bar graph, the lower the value shown on the bar plot, the better. These results show that, for all three structure scores, the Mmhc is considered to be the better model, but this difference is likely not to be very significant.

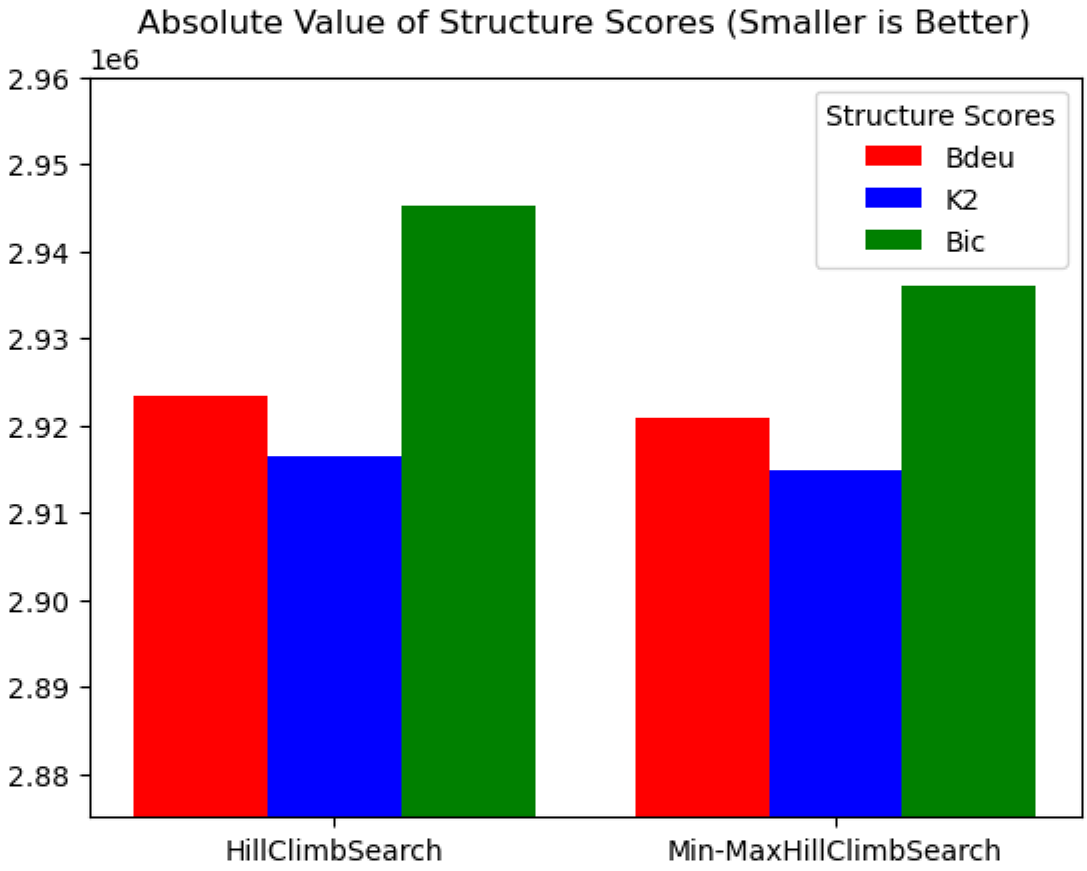


Figure 18: Structure Scores for the algorithms Hill Climb Search and Mix-Max Hill Climb Search

7 Causal Inference

In chapter 4 of this project, we used the built Bayesian Network models for a classification task. This type of task can also be performed with other non-causal models such as linear regression and decision trees. However, there is one type of task that can only be performed with causal models: causal inference.

Given a distribution of random variables, a causal inference inquires about the probability of an event given an intervention. Refer to Section 3 for an explanation of how an intervention is performed. Using this project's dataset, an example of a causal inquiry would be "what is the probability that an individual in the population has heart disease if we could somehow force the population to not smoke" It is important to note that this is fundamentally different from conditioning on a random variable; an example of this would be “what is the probability that an individual in the population has heart disease given that we already know that he or she does not smoke”.

One important aspect of intervening is that causal links between variables cannot be established from observations alone. For example, suppose that, after conditioning on the variable “high cholesterol”, it is discovered that it is positively correlated with the variable “heart disease”. We cannot conclude that this correlation is due to the fact that “high cholesterol causes heart disease” because other variables are also playing a role in the generation of random outcomes. For example, one possible explanation for this correlation is that smoking causes both an increase in cholesterol levels and heart disease (both variables have a common cause). Therefore, “high cholesterol” and “heart disease” would be correlated despite the fact that high cholesterol does not cause heart disease.

In a causal query, however, the random outcome-generating mechanism is changed and it can prevent the observation of correlations that are not due to causality. Using our previous example, if we intervene on the variable “high cholesterol” so that it attains the value “has high cholesterol” with a probability of one, then the variable “smoker” would only directly cause the variable “heart disease”. Therefore, if we compare the distribution for the variable “heart disease” when the variable “high cholesterol” is forced to attain a value of “has high cholesterol” with the distribution when we force the “high cholesterol” variable to attain a value of “does not have high cholesterol”, we would no longer see the correlation, indicating that these two variables are not causally connected.

This process of using a probability distribution fit from data to study the differences between two “theoretical” populations – one for each value of the intervened variable – is the same in principle as that of using randomized controlled trials (RCTs) to establish causality. In these RTCs, two groups are created by randomly sampling individuals from the population and assigning them randomly to two groups: a control group where the participants are instructed or “forced” to, for example, not have high cholesterol, and an experimental group were the participants are instructed to have high cholesterol (keep in mind that this example would not be ethical in practice and that the details on how to make the participants conform to such instructions are beyond the scope of this project).

In theory, there are many queries that can be made from the model because you are free to choose any number of variables to intervene on as well as the values of each of them. For the purposes of this project, we only intervened on a single variable at a time, and for every variable we intervened on we did one query per possible value of that variable. Note that it is only possible to query the probabilities of variables that are descendants of the intervened variable on the DAG.

Once a probability for a variable was obtained after doing an intervention, it was compared with the probability that would have been obtained if an intervention was not made. This difference in probability can therefore tell you which features are the most efficient to intervene on if you want to minimize the risk of developing a given health condition.

7.1 Results

The following bar plots summarize the results of three of the most important queries done. Refer to the project's jupyter notebook for the full results. The horizontal axis on this bar plot contains each intervention; it is named after the name of the feature, followed by an underscore, followed by the value of the feature. For example, High_Cholesterol_1 means that we are intervening so that the feature "high cholesterol" attains a value of 1 (meaning that the individual is forced to have high cholesterol). The height of the bars represents the differences in probability between doing and not doing the intervention. In Figure 20, for example, the height of the bar for High_Cholesteror_1 is about 0.1 which means that, according to our model, the probability of a person having hypertension increases by about 10% if we intervene to make the person have high cholesterol.

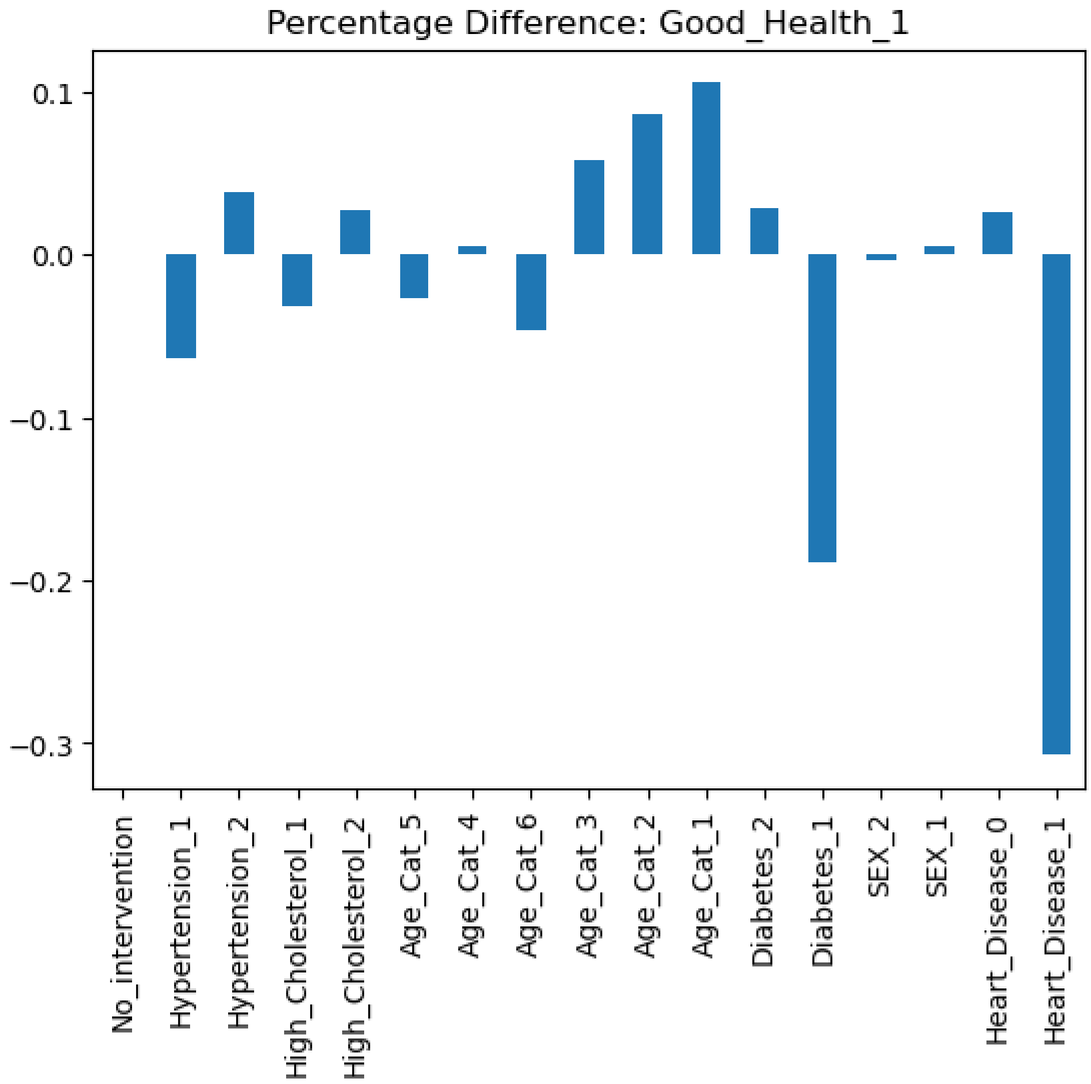


Figure 19: Bar plot showing the probability difference of having good health between intervening and not intervening for various types of interventions

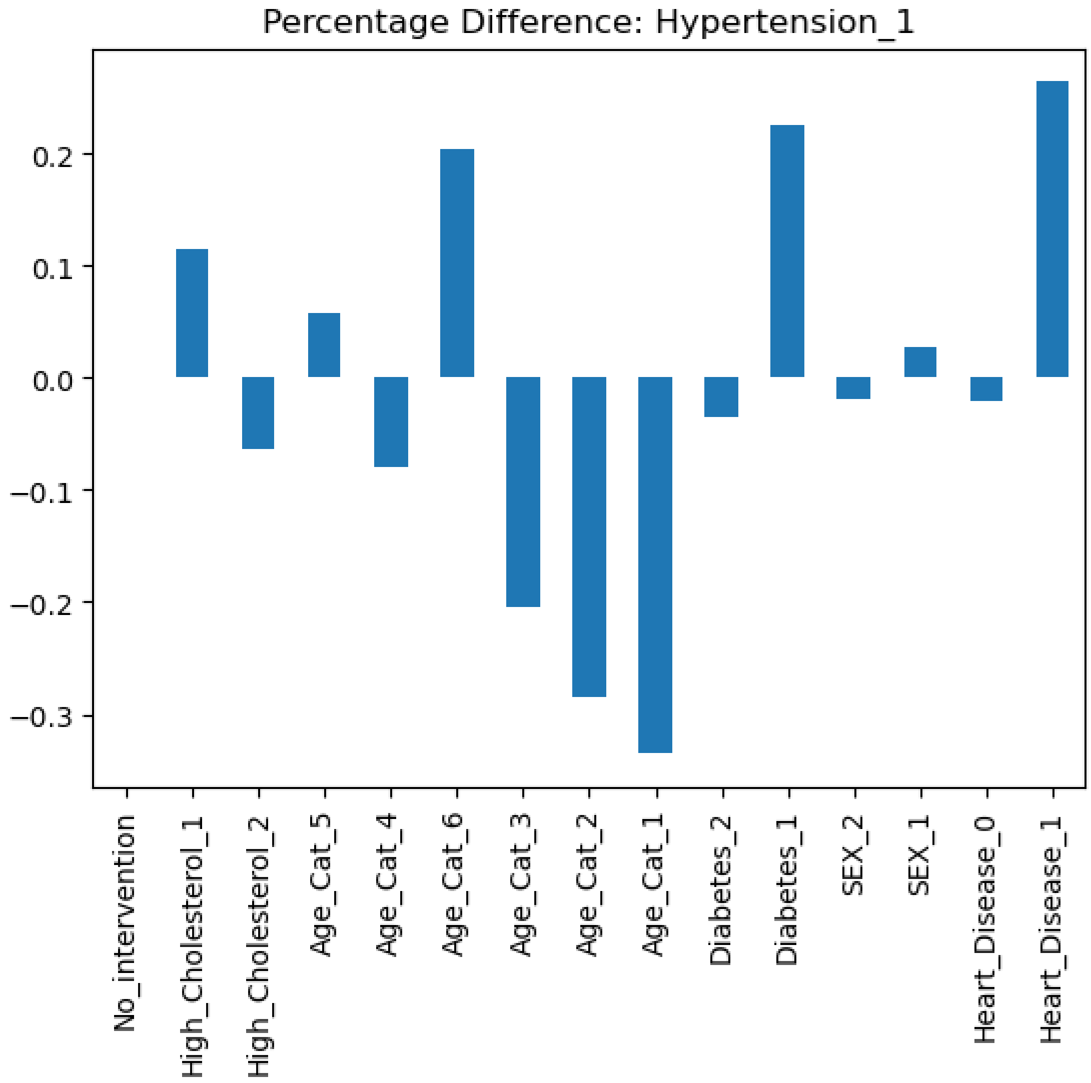


Figure 20: Bar plot showing the probability difference of having hypertension between intervening and not intervening for various types of interventions

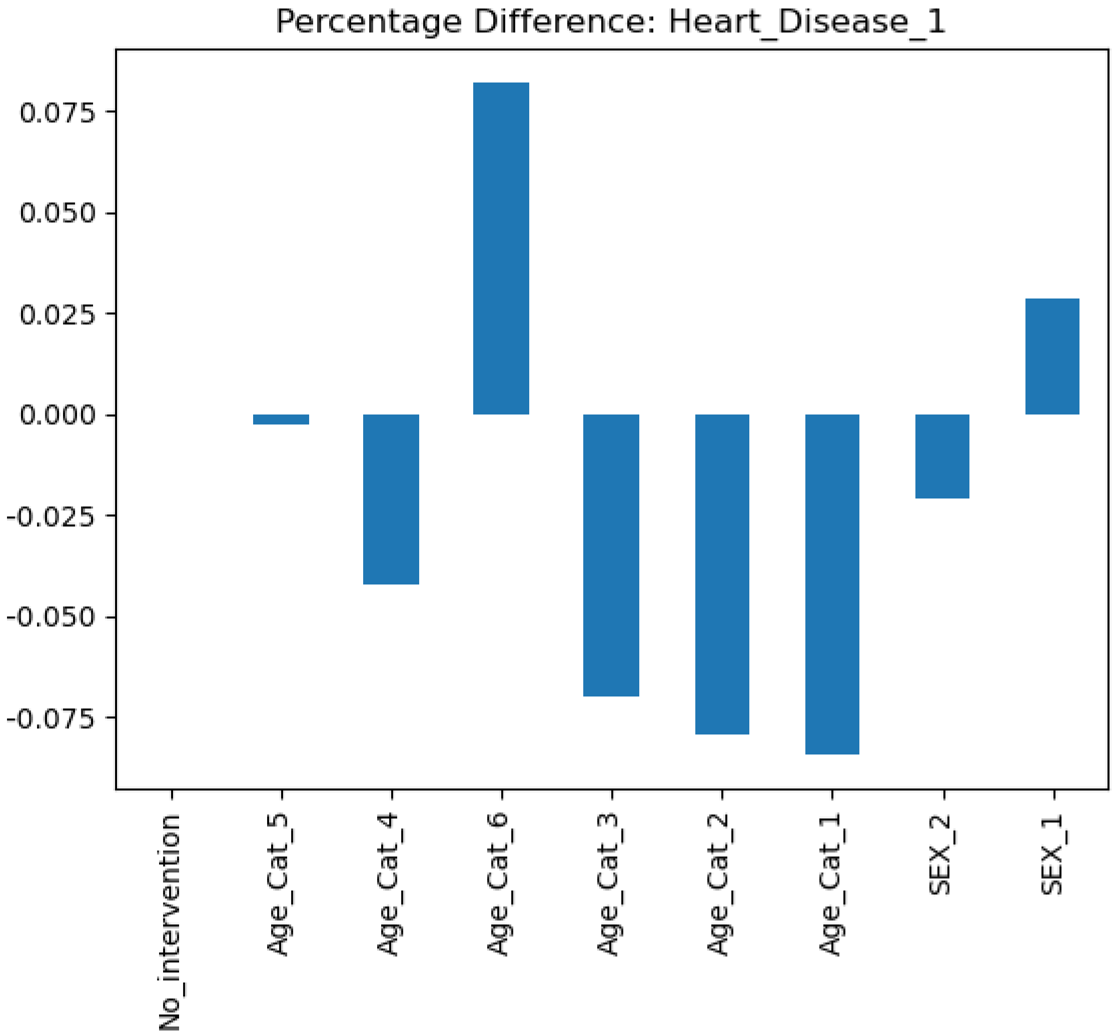


Figure 21: Bar plot showing the probability difference of having heart disease between intervening and not intervening for various types of interventions

Note that some of the interventions shown here are only possible in theory. For example, it doesn't make sense to intervene on the feature "age category". However, these theoretical values could be seen as a measure of "feature importance".

8 Conclusion and Future Work

8.1 Summary

In this project, we built three Bayesian Network models using data from the CDC about health feature incidence in the adult U.S. population.

The model built with the PC algorithm did not output a DAG, as should have been the case, so it could not be used as a model. In a future project, we will investigate the reasons for this more carefully. One possible explanation is that there might be a problem with its implementation in the pgmpy library that we used.

8.2 Discussion of Results

8.2.1 Classification

When we used our two built Bayesian Network models as classifiers for the heart disease feature, we discovered that they get good classification scores in the metrics recall and roc auc. In fact, we discovered that their performance in these two metrics was on par with a more traditional classification model that we built with the algorithm XGBoost. These results are summarized in Figures 1-4, 8-11, and 13-16. We, therefore, conclude that even though the main motivation of Bayesian Network models is not classification, they still perform well on these tasks.

There is a big difference between the criteria used for building the best possible model between standard classification algorithms (such as XGBoost) and Bayesian Network algorithms: in the former, the best model is found by minimizing a loss function that quantifies classification performance, while the latter tries to make sure that the conditional independencies found in the data are replicated as closely as possible in the final model. It should be remembered that the concept of conditional independencies is closely related to that of causality. The results, then, seem to suggest that when the algorithm builds a model for classification using the concept of causality as a guiding principle, it doesn’t seem to lead to noticeable better or worse results. The results from cross-validation also show that when Bayesian Network models are used for classification tasks, they don’t seem to do better or worse than traditional classification algorithms in terms of overfitting. Irrespective of these results, classification tasks are not the main motivation to use Bayesian Network models anyways.

8.2.2 Structure Discovery

While it seems that we don’t gain much from using Bayesian Network models for classification tasks, their real value lies in other aspects of them. One of them is that they might be able to discover the causal structure in the data. This is shown in the DAGs that were built for each of the models shown in Figures 5 and 12. These two graphs might represent the correct causal structure in the data. However, since causality is established based on how the probability distribution of the data behaves under interventions (like the interventions done in randomized control trials, for example), it can’t definitively be established from observational data alone. The graph search algorithms use instead the concept of conditional independencies to build the graphs, which is closely related to causality. In fact, a lot of the causal relationships shown in graphs 5 and 12 don’t make a lot of sense. For example, it doesn’t make sense that “sex” is a direct cause of “smoking”. Therefore, the DAGs that we get as the outputs of the algorithms should not be seen as conclusively establishing causal connections, but rather as presenting one of many possible causal structures that are consistent with the conditional independencies found in the data.

This last point highlights a few aspects of how machine learning methods should be used to discover the causal structure of data. One aspect is that, once a graph is built with a graph search algorithm, this should be seen only as “suggesting” a possible causal structure. Researchers could then use these suggestions to carry out randomized control trials to verify whether these causal connections are actually there. This is useful because a single randomized control trial is only capable of checking if there is a causal link between two pairs of variables; it is therefore unfeasible to do one randomized control trial for every pair of variables of interest. However, the graph search algorithms offer suggestions about which pairs of variables are the most worthwhile to verify through experiments.

Another aspect to consider is that the graph search algorithms can be used in conjunction with expert domain knowledge, which could lead to higher confidence in the correctness of the resulting graph in representing the causal structure of the data. This is because domain knowledge can reduce the search space of possible graphs the algorithm searches. For this project, though, we did not have access to expert knowledge. The best we could do was to rely on intuition, such as when we decided the Hill Climb Search Algorithm should not look for graphs where the variables “sex” and “age category” have parents.

For applications involving predictions, such as classification tasks, whether the resulting graph represents the correct causal structure is of no relevance because the algorithm makes sure that it represents as closely as possible the conditional independencies in the data, and this is enough to make accurate predictions. If, however, we are interested in performing causal inquiries, it is essential to have the correct graph.

8.2.3 Causal Inquiries

We also used the models to perform causal inquiries. The accuracy of these inquiries is heavily dependent on the underlying DAG being a correct representation of the causal structure of the data and, as we explained, that is not something we can be very confident about for this project. We nevertheless performed the causal inquiries as a way of showing how they are performed.

The results are summarized in figures 19-21 for three of the most important features: having good health, having hypertension, and having heart disease. As was previously stated, the value of these results is that they can tell the medical practitioner what are the appropriate features to intervene on in order to minimize the risk of a person having a particular health condition. For example, according to Figure 19, if the medical practitioner somehow makes an individual not have hypertension (Hypertension_2 on the horizontal axis), the probability of a person having good health increases by about 4%. It is also interesting to note that, according to this figure, the sorts of intervention that will have the greatest effect on good health will be those that make the patient have diabetes (Diabetes_1) or hypertension (Hypertension_1), which would decrease the individual's probability of having good health by about 20% and 30% respectively. Of course, such an intervention would not be done in practice because it leads to a negative health outcome.

Another interesting observation from these bar plots is that a lot of the interventions that would be the most beneficial cannot be done in practice. For example, according to Figure 20, the interventions that reduce the probability of developing hypertension by the greatest amounts are the age category features.

Yet another interesting observation is that, according to Figure 19, the only interventions that can be made to change the probability of developing heart disease are theoretical ones: those related to age and sex. It was also noticed that all of these interventions change the probability by a very small amount.

8.2.4 Comparing Causal Queries with Counterfactual Explanations

It is interesting to discuss the similarities between causal queries and counterfactual explanations in non-causal models. In the latter, given a data instance whose target class was predicted by a classification model, the following question could be asked: in what ways can we change the values of the features of this instance so that its predicted target class changes? For example, we might have a patient that was predicted by a model to have heart disease, and the same model could also tell us that if the patient had not been a smoker and all other features remained the same, he or she would not have been classified as having heart disease. This certainly sounds similar to the causal query "if we were to intervene to make the patient not smoke, what would the probability of the patient having heart disease be? (if the probability lowers enough, the patient could be reclassified as "not having heart disease").

The most important distinction is that counterfactual explanations involve non-causal models. In our previous example, the patient was reclassified as not having heart disease when the "smoking" feature changed because those two features are correlated. However, this should not be taken as evidence that quitting smoking will lower your chances of developing heart disease. Instead, what we can conclude from this counterfactual explanation is that "if we had observed another individual in the population that is not a smoker but otherwise has the same features as the patient, this individual would have been classified as not having heart disease". In other words, counterfactual explanations help us draw conclusions about *observations* in the population, but not about *interventions*.

By contrast, in a causal query, we calculate the probability of an individual having a feature value after first intervening in the random outcome-generating process. Using our previous example, if we intervene to make an individual not be a smoker and discover that his probability of developing heart disease diminishes, the interpretation of the result would be "if the patient had lived his or her life in the same manner up to this point except that he or she was "forced" to not smoke, the probability of the patient developing heart disease would decrease". We can also conclude in this case that smoking causes heart disease because the difference in probability is due to an intervention. Notice that this is not a conclusion about the data that was observed, but rather about a different hypothetical distribution where the random outcome-generating mechanism is changed.

This shows the immense value that causal models have in many applications. In a medical setting, we are precisely interested in the causal structure in the features of the patient because the physician will wish to make an intervention that will lead to a better health outcome for the patient. This cannot be accomplished with counterfactual explanations in a non-causal model; as the name implies, they can only offer an explanation about why a patient was categorized in a particular way.

8.3 Using the Model in a Medical Setting

Causal machine-learning models, such as Bayesian Networks, are particularly useful in the field of medicine. The reason is that the question of how to “intervene” in a patient's behavior to improve his or her health outcomes occurs frequently in this field. Note how it is not enough to simply know that, for example, “individuals who smoke are more likely to develop heart disease” because this is only a correlation that does not necessarily imply that quitting smoking will reduce the risk of heart disease. It is therefore of immense value to understand the causal links between health features so that the medical practitioner can understand the correct types of interventions to apply to the patient.

Assuming that the DAG describing the causal model is to be trusted, medical practitioners could use it to do causal queries similar to, for example, “if a patient quits smoking, how will his chances of developing heart disease change?”. One useful feature of these types of models is that they can be used to ask any kind of causal query, no matter how complex. An example of such a complex query would be “by how much would the probability of developing heart disease of a patient change if he or she quits smoking and lowers his or her cholesterol levels given that we know that the individual belongs to the young age category”. This, therefore, gives the physician a tool to discover many options on how to intervene in the behaviors of a patient in order to improve his or her health outcomes.

8.4 Future Work

One of the most pressing issues that we are interested in is discovering why the PC algorithm did not generate a DAG. If the reason for this was that there is a mistake in the algorithm’s implementation in the pgmpy library, this insight could help resolve it.

Another interest is to work on the project with more input from the medical community. As was previously explained, having expert domain knowledge is fundamental when using machine learning techniques that attempt to discover the causal structure in data. This will mainly be useful in setting restrictions for the space of graphs that will be searched by the graph search algorithms.

Another interest is to see if better models can be built with the algorithms used in this project by adjusting the parameters. One parameter we are particularly interested in is the number of features to include in the model. For this project, the choice of features was based on the previous project's analysis in determining feature importance for the prediction of having heart disease by calculating average SHAP values, and the number of features chosen to be included (ten) was, admittedly, arbitrary.