Cervical Cancer Classification and Causal Inference

CHAN, Shun Hin

E-mail: *shchanai@connect.ust.hk*
SID: 20100545

and

FONG, Ki Wai

E-mail: *kwfongab@connect.ust.hk*
SID: 20271186

30th May 2021

1 Introduction

Cervical cancer is one of the most common cancers in women (WHO, 2021). Those who found abnormal cells in their cervix would accept cervical biopsy to determine if they have the cancer. We use the cervical cancer dataset collected from UCI Machine Learning Repository (Cervical cancer (Risk Factors) Data Set, 2017) with 36 variables for our analysis. Several researches have been done on the same dataset; Choudhury, Wesabi and Won (2018) applies different machine learning models on predicting the Biopsy results using over-/under-sampled dataset; Razali, Mostafa, Mustapha, Wahab and Ibrahim (2020) applies deep learning models on predicting Biopsy results using over-sampled dataset. Predictions is useful for two reasons. If we can make accurate predictions, we can save costs for conducting biopsies. Predictions can also give us insight for risk management, say understanding the risk of being diagnosed with the cancer. It is also beneficial to health insurance sectors since they often need to provide health checking service for any new policies. Having accurate predictions can reduce costs for providing health services.

Both prediction accuracy and associations or causal relationships among variables are important. However, the relationships are more interesting since we are often more interested about the root causes of the Biopsy results so that we can adopt preventive measures. Bayesian network is a good model for this particular type of problems. Bayesian network provides a good visualization of the relationships among variables, and can also be used to evaluate the strength of the relationships and conditional probabilities. Several similar researches have been done; Park, Chang and Nam (2018) uses Bayesian networks to predict post-stroke outcomes with related risk factors; Muibideen and Prasad (2020) uses Bayesian network to predict cardiovascular disease and discover the relationship among attributes. To learn the underlying structure based on the data, we follow Friedman and Koller (2001) who uses a Markov chain monte carlo method on the variable (or node) ordering, then sample graphs that are consistent with the order. Their method is more efficient than a local search method called MC^3 proposed by Madigan, York and Allard (1995), which conducts local movements by adding, deleting, or reversing an arc. Once graphs are sampled, we can conduct analysis on the features, say the strength of the connections between two variables, by estimating the probability of the existence of an arc empirically, as known as text features, using methods mentioned in Suter and Kuipers (2021). We also use likelihood weighted sampling to estimate the conditional probabilities

so that we can evaluate the relative risks given a set of variables.

In this project, we will first try to replicate the results using models similar to Choudhury et al. (2018) with an additional random forest model. Then, we will apply Bayesian networks for prediction and causal inference for figuring out important relationships among variables. Section 2 describes the dataset used in this project, Section 3 discusses the methodology used for data cleaning, predictions and causal inference, and the package used for implementation, Section 4 gives the main results for both prediction and causal inference, Section 5 summarizes the findings in this report.

2 Dataset

The cervical cancer dataset (Cervical cancer (Risk Factors) Data Set, 2017) contains demographic information, habits and historical medical records of 858 patients. The description of the variables plus the percentage of missing data of each variable are shown in Table 1. The variable `STDs: Time since first diagnosis` and `STDs:Time since last diagnosis` contain 91.7% of missing values, thus we delete both variables from our analysis. As for other variables, we perform the imputation to the missing values using the modes of the variables for simplicity.

3 Methodology

If we treat `Biopsy` as the sole target variable as we let the models learn, then given there are 858 patients, only 55 ($< 7\%$) has cancerous cells found during the cervical biopsy (and thus their `Biopsy` are 1). The vast difference between positive cases and negative cases would make the resulting learned model skewed towards the majority class (Danquah (2020)). This is undesirable if we wish to use the model to help us predict if a patient has cervical cancer, which should still be rare by nature. Therefore, we used over-/under-sampling to manipulate our data set for training purpose.

3.1 Selected existing learning models

First, we split the whole data set ($N = 858$) into the training and test sets in $8 : 2$ ratio, then standardize both sets according to the distribution of the training set. Next, instead of manipulating the whole data set, we manipulate the standardized training set ($N = 686$) to avoid data leakage when we train the model. We over-sample the positive cases using SMOTE (synthetic minority over-sampling technique), under-sample the negative cases using Near Miss proposed by Mani and Zhang (2003), or use SMOTE on positive cases followed by Near Miss on negative cases to even out the number of positive and negative cases. In particular, using SMOTE to synthesize data follows this formula (Lemaître, Nogueira and Aridas (2017)):

$$x_s = x_i + t \cdot d(x_i, x_{i,nn}), \tag{1}$$

where x_i is the selected sample from the minority class, $x_{i,nn}$ is one of the k-nearest neighbors ($k = 5$) of x_i randomly selected, $d(a, b) \geq 0$ is the distance between a and b, $t \in [0, 1)$ is randomly selected, and x_s is the synthesized sample. We used Near Miss instead of Tomek Linking like in Choudhury et al. (2018) so that we can control the ratio between positive and negative cases. Also in contrast to Choudhury et al. (2018) that they did not specify the new positive data count after using SMOTE, we only SMOTE-ed the positive data to have the same count as negative data in the training set, or SMOTE-ed the positive data and Near Miss-ed the negative data in the training set so their counts $(+_{\text{SMOTE}}, -_{\text{NM}})$ are the geometric mean of the original positive count $(+_{686})$ and the negative count $(-_{686})$ rounded up to the nearest integer:

$$+_{\text{SMOTE}} = -_{\text{NM}} = \lceil \sqrt{+_{686} \times -_{686}} \rceil. \tag{2}$$

Attribute	Type	Missing percentage
Age	Integer	0.0%
Number of sexual partners	Integer	3.0%
First sexual intercourse (age)	Integer	0.8%
Num of pregnancies	Integer	6.5%
Smokes	Boolean	1.5%
Smokes (years)	Boolean	1.5%
Smokes (packs/year)	Boolean	1.5%
Hormonal Contraceptives	Boolean	12.6%
Hormonal Contraceptives (years)	Integer	12.6%
IUD	Boolean	13.6%
IUD (years)	Integer	13.6%
STDs	Boolean	12.2%
STDs (number)	Integer	12.2%
STDs:condylomatosis	Boolean	12.2%
STDs:cervical condylomatosis	Boolean	12.2%
STDs:vaginal condylomatosis	Boolean	12.2%
STDs:vulvo-perineal condylomatosis	Boolean	12.2%
STDs:syphilis	Boolean	12.2%
STDs:pelvic inflammatory disease	Boolean	12.2%
STDs:genital herpes	Boolean	12.2%
STDs:molluscum contagiosum	Boolean	12.2%
STDs:AIDS	Boolean	12.2%
STDs:HIV	Boolean	12.2%
STDs:Hepatitis B	Boolean	12.2%
STDs:HPV	Boolean	12.2%
STDs: Number of diagnosis	Integer	0.0%
STDs: Time since first diagnosis	Integer	91.7%
STDs: Time since last diagnosis	Integer	91.7%
Dx:Cancer	Boolean	0.0%
Dx:CIN	Boolean	0.0%
Dx:HPV	Boolean	0.0%
Dx	Boolean	0.0%
Hinselmann: target variable	Boolean	0.0%
Schiller: target variable	Boolean	0.0%
Cytology: target variable	Boolean	0.0%
Biopsy: target variable	Boolean	0.0%

Table 1: The description of 36 variables included in the cervical cancer dataset.

After the above procedures, we tried Gaussian Naive Bayes (GNB), decision trees (Tree), logistic regressions (LR) with L_2 penalty by default, support vector machines (SVM) and k-Nearest Neighbors (kNN) with $k = 5$ on the processed data sets so we would compare our results with the results Choudhury et al. (2018) obtained. However, to quantify the learning of the existing models, instead of listing the (weighted) average of accuracy, specificity, precision, and the F_1 score, we also consider the weighted F_2 score that we weigh recall more than precision, and AUC (area under the ROC curve) of the existing learning models we mentioned above. As for recall, since the weighted recall equals overall accuracy, we refer "recall" in the tables in the result section as the positive case recall. The specificity, precision, the F_1 and F_2 scores are weighted by class size in the test set because we want to select the model that is decent in predicting both positive and negative cases. Since the AUC of a model close to 1 and far away from 0 implies high recall (high probability to find out a patient actually have abnormal cervical cells) and low fall-out (low probability to falsely claim the

patient has abnormal cervical cells), comparing the AUC between the learning models would help decide using which one to detect the anomaly in the patient's cervix.

Apart from the above supervised learning models that Choudhury et al. (2018) used, we also tried random forest classifiers. We followed the same procedure as above until we need to determine the hyper-parameters that give the highest recall on the standardized training set. We tune the hyper-parameters: `max_features`, the maximum number of features to consider when looking for the best split in the forest, `n_estimators`, the number of trees in the forest, `max_depth`, the maximum depth of a tree, and `min_samples_split`, the minimum number of samples required to split an internal node, by performing a grid search stratified 7-fold cross validation on the above 4 hyper-parameters regarding the recall score of the SMOTE-ed and Near Miss-ed standardized training set. After we obtained the highest recall score and its respective hyper-parameter combinations, we fit the same model using the same standardized training set with(out) over-/under-sampling using SMOTE and/or Near Miss. Then we compare the metrics (overall accuracy, weighted specificity, weighted precision, positive recall, weighted F_1 score, weighted F_2 score and AUC) on the test set.

The model training and evaluation in this part are coded in Python 3.7, with many usages of the libraries `scikit-learn` and `imbalanced-learn`.

3.2 Bayesian Networks

In the previous part, we have mentioned various machine learning models for predicting the cervical biopsy outcome. However, the results are difficult to be interpreted. Instead of doing plain prediction, Bayesian networks can give interpretable predictions. They can also be used to explain the relationships among variables.

The problem we are considering is to model the conditional dependencies of a set of variables V. We will first introduce the definition of Bayesian networks.

Definition 3.1 (Bayesian Network)**.** A *Bayesian Network* is a probabilistic graphical model that represents the conditional dependencies of a set of n variables $V = (X_1, \ldots, X_n)$ using a directed acyclic graph (DAG). Each node in the graph represents a variable. Let $X_1, X_2 \in V$, a (directed) *arc* $X_1 \to X_2$ indicates conditional dependence. A parent of the node X_i, denoted by Π_i, is a set of nodes X_j such that there exists arcs that $X_j \to X_i$. A children of a node X_i is a node X_j such that $X_i \to X_j$. A set of nodes V' is called non-descendent of X_i if V' does not contain children, or children of children of X_i. A Bayesian network satisfies the conditional independence, i.e., a nodes X_i is independent to its non-descendent conditional to its parent set Π_i. Then, a Bayesian network can represent a factorization of a joint distribution

$$P(X_1, \ldots, X_n) = \prod_{i=1}^{n} P(X_i|\Pi_i). \tag{3}$$

A structure G of the Bayesian network is defined as a set of nodes V and a set of arcs E, denoted by the pair $G = (V, E)$.

Definition 3.2 (Directed acyclic graph)**.** A *Directed acyclic graph* is a directed graph with no directed cycles. A directed graph is a graph with nodes connected with directed edges only. A set of nodes $\{X_1, \ldots, X_k\}$ is said to be forming a cycle if there exists a path such that X_j can return to X_j for all $j = 1, \ldots, k$.

Definition 3.3 (Causal Bayesian Network)**.** A *causal Bayesian network* is a Bayesian network built using causal relationship. Arcs are interpreted as indicating causal relationships between nodes.

Note that not all Bayesian networks are causal Bayesian networks, but we are often interested in causal Bayesian networks.

3.3 Structural Learning

In most of the cases, the underlying structure of a set of variables V is unknown. What we are interested is to recover the structure G given a set of data D. Consider a simple case with all variables taken discrete values, say, $X_i \in \chi_i := \{x_{i1}, \ldots, x_{in_i}\}$, where n_i is the number of possible states of X_i. Given a set of data D, which contains N cases, each case can be represented by a vector $D_\ell \in \bigtimes_{i=1}^{n} \chi_i$, $\ell = 1, 2, \ldots, N$. With this, Π_i can take $K_i = \prod_{i=1}^{r_i} \chi_{(i)}$ configurations, where $\chi_{(i)}$ is the i-th parent of X_i and $r_i = |\Pi_i|$. We further let $\theta_{ijk} = P(X_i = j|\Pi = k)$, given a structure G and assuming non-informative prior on G, i.e., $P(G) \propto 1$, and assume that the prior distribution of $\theta_{i*k} := (\theta_{i1k}, \ldots, \theta_{in_ik})$ is Dirichlet with concentration parameter $\alpha_{i*k} = (\alpha_{i1k}, \ldots, \alpha_{in_ik})$, the posterior distribution of G given D can be expressed as

$$P(G|D) \propto P(D|G)P(G) \propto \prod_{i=1}^{n} P(X_i|\Pi_G), \tag{4}$$

where

$$P(X_i|\Pi_G) = \prod_{i=1}^{n} \left[\prod_{i=1}^{n_i} \frac{\Gamma(\alpha_{i.k})}{\Gamma(\alpha_{i.k} + N_{i.k})} \prod_{j=1}^{r_i} \frac{\Gamma(N_{ijk} + \alpha_{ijk})}{\Gamma(\alpha_{ijk})} \right],$$

where N_{ijk} is the number of cases in D such that $X_i = j$ and $\Pi_i = k$, and $N_{i.k} := \sum_{j=1}^{n_i} N_{ijk}$ is the number of cases in D such that $\Pi_i = k$. Having $P(G|D)$, we can conduct structural learning given the dataset D, and Π_G is a set of all parents of all nodes in G. The score

$$\log P(G|D) = \sum_{i=1}^{n} \left[\log \frac{\Gamma(\alpha_{i.k})}{\Gamma(\alpha_{i.k} + N_{i.k})} + \sum_{j=1}^{q_i} \prod_{j=1}^{r_i} \frac{\Gamma(N_{ijk} + \alpha_{ijk})}{\Gamma(\alpha_{ijk})} \right] \tag{5}$$

is known to be the *Bayesian Dirichlet equvalent uniform* (BDeu) score.

There are many ways to learn the structure, say we can do exhaustive search, or we can do optimization. In this project, we adopt the method proposed by Friedman and Koller (2001), who treats the DAG's as random and uses the partial ordering of nodes to sample DAG's using a MCMC method.

Definition 3.4 (Partial node ordering)**.** A *partial node ordering* of a DAG G over the node set V of n nodes is a set of ordering $\prec_1, \ldots, \prec_n$ such that if $X_i \in \Pi_j$ then $i \prec j$, i.e., node X_i ordered before X_j whenever X_i is a parent of X_j.

Under Bayesian framework, we assume that the underlying graph of a set of variables is random; alternatively, the adjacency matrix of a graph G is a random matrix. Thus, given a set of data, we can conduct MCMC sampling on the adjacency matrix and thus obtain a set of sampled graphs. Friedman and Koller (2001) proposed a MCMC sampling method by sampling the node ordering. Here is the algorithm. A MCMC step m, suppose that we have a node ordering $\prec^{(m)} = (\prec_1^{(m)}, \ldots, \prec_n^{(m)})$. We propose a candidate $\prec'$ by swapping two elements in $\prec^{(m)}$, and accept it with probability

$$\min \left\{ 1, \frac{P(\prec' |D)}{P(\prec^{(m)} |D)} \right\},$$

where

$$P(\prec |D) = \prod_{i=1}^{n} \sum_{\Pi \in \Pi_i^{\prec}} P(X_i|\Pi_i), \tag{6}$$

where $\Pi_i^{\prec}$ is a set of parents of X_i which is consistent to the order $\prec$. After the above mcmc procedure, we can sample a graph, G_m, according to the terms in the posterior probability in Equation 6. Thus, we can obtain a time series of graphs $\{G_1, \ldots, G_M\}$ if we conduct the order mcmc for M iterations.

3.3.1 Prediction

In order to predict the values of a variable X_i given a set of evidence $E = e$, we can estimate the conditional probability $P(X_i = x|E = e)$ for $X_i \in \chi_i$, and predict X_i as $\arg\max_{x \in \chi_i} P(X_i = x|E = e)$. This can be done using the sampled graphs $\{G_1, \ldots, G_M\}$. Note that when we conduct mcmc, usually we need to remove the first $B < M$ iterations since the time series has not converged before the burn-in period. There are mainly two approaches for estimating

(1) Maximum a posteriori (MAP). We use the graph that has the highest posterior score $P(G|D)$. Then, we can estimate the probability $P(X_i = x|E = e)$ using the MAP graph.

(2) Model averaging. For each graph in the sampled chain, we estimate the probability $P(X_i = x|E = e)$. Then, we take average of these probability.

It remains to find a method to estimate $P(X_i = x|E = e)$ from the graph. More generally, we estimate $P(Y = y|E = e)$, where $Y = y$ can be any event as a function of V. A method is the *Likelihood-Weighted (LW) sampling*.

Algorithm 1: Likelihood-Weighted (LW) sampling

Result: Write here the result
Input: A graph G, an event $Y = y$ and an evidence with k vairables, $E = e \equiv (e_1, \ldots, e_k)$;
Initialize $w \leftarrow 1$;
for $i = 1, 2, \ldots, n$ **do**
  if $X_i \in E$ **then**
    Set $x_i \leftarrow e_i$;
    Update $w \leftarrow w \times P(e_i|\Pi_i)$, where Π_i is the parent set of X_i in the graph G ;
  else
    Sample x_i from $P(X_i|\Pi_i)$;
  end
end
return $(x_1, \ldots, x_n), w$

We conduct Algorithm 1 for L times (e.g., $L = 10^6$), and thus we obtain the output $(x_1^\ell, x_2^\ell, \ldots, x_n^\ell), w_\ell$ for $\ell = 1, 2, \ldots, L$. Then, the estimate of the probability $P(Y = y|E = e)$ is

$$\hat{P}(Y = y|E = e) = \frac{\sum_{\ell=1}^{L} w_\ell I\{y = y_\ell\}}{\sum_{\ell=1}^{L} w_\ell}, \tag{7}$$

where y_ℓ is the configuration of Y in the ℓ-th sample.

3.3.2 Estimating Edge Features

In order to evaluate the *strength* of a causal relationship which represented by an arc, we estimate the probability of the existence of a particular arc, $X_i \to X_j$, using the formula

$$\hat{P}(X_i \to X_j|D) = \frac{1}{M - B} \sum_{m=B+1}^{M} 1_{X_i \to X_j}(G_m), \tag{8}$$

where $1_{X_i \to X_j}(G_m) = 1$ if the arc $X_i \to X_j$ exists in the graph G_m, and is equal to zero otherwise. We refer this as the probability of *edge feature*. Estimating probability of edge feature is important, since we can use this to explain the strength of a causal relationship. To evaluate the precision of the estimated edge features, we can conduct Bootstrap. We conduct bootstrap 1000 times, each time we sample 50 estimated edge features probabilities with replacement. Then, we obtain 1000 bootstrapped means of the edge features. Then, we get the 2.5^{th} and 97.5^{th} percentile to form the credible interval of the average edge feature.

3.4 R packages

The R package `BiDAG` contains functions for implementation of a collection of MCMC methods for Bayesian structure learning of DAGs for both continuous and discrete data (Suter & Kuipers, 2021). `orderMCMC` function in the package is used for order MCMC structural learning.

The R package `bnlearn` contains functions for building network, structural learning and scoring (Scutari, 2010). `empty.graph()` function is used for initialize a network, `bn.fit()` function is used to estimate the parameters in the Bayesian network, `cpquery()` function is used for calculating conditional probabilities.

4 Results

4.1 Selected existing learning models

Figure 1 shows the boxplots of the AUC by first splitting the whole data set into the training and test sets in $8:2$ ratio, then over-sampling, under-sampling, and doing both on the training set from left to right for 100 times.

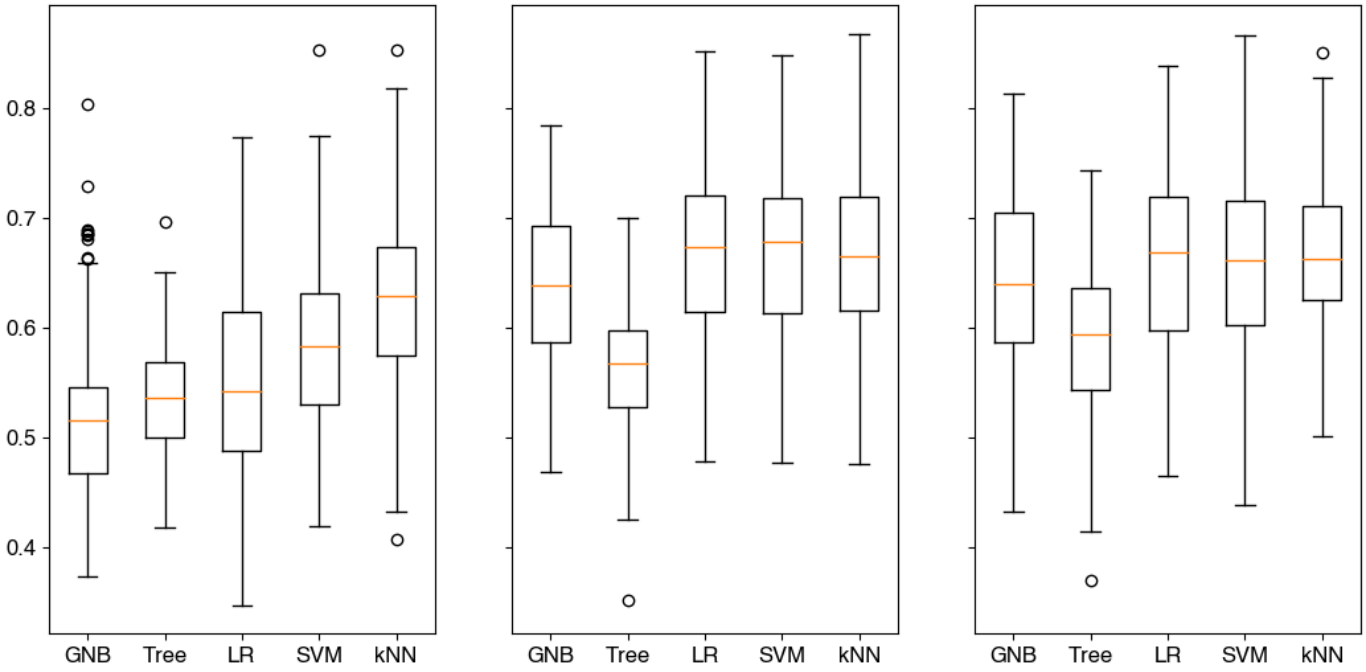


Figure 1: The boxplots of the AUC under the ROC curve of GNB, Tree, LR, SVM and kNN after we repeat the workflow in Section 4.1 for 100 times. From left to right: over-sampling, under-sampling, and doing both.

We find the decision tree is not as outstanding, and the Gaussian Naive Bayes is not as not suitable as Choudhury et al. (2018) claimed. While the average AUC of GNB is not the best, the AUC of the decision tree is even worse than GNB on average when we used the (SMOTE-ed and) Near Miss-ed standardized training set. Using the same training sets, the average AUC of logistic regression, SVM and kNN are very close between 0.661 and 0.671.

Since the boxplots about the AUC scores cannot imply a lot, we refer to several model metrics to decide which would be the best for imbalanced data set like the cervical cancer risk factor data set. For comparisons with Choudhury et al. (2018), the overall accuracy, positive case recall, weighted specificity, weighted precision, weighted F_1 score and weighted F_2 score (all in %), of the selected models regarding the standardized test set without over-/under-sampling are all presented as follows from Table 2 to 4. Those weighted metrics are weighted with positive or negative class size.

	GNB	Tree	LR	SVM	kNN
Accuracy	9.4	87.0	76.8	83.3	76.9
Recall	97.2	15.9	35.4	26.1	42.8
Specificity	91.2	20.7	38.1	30.1	45.1
Precision	89.7	89.0	89.4	89.4	90.1
F_1 score	6.9	87.9	82.0	86.0	82.2
F_2 score	5.6	87.3	78.5	84.2	78.6

Table 2: Metrics about the selected models after learning the SMOTE-ed training set for 100 times.

	GNB	Tree	LR	SVM	kNN
Accuracy	63.5	34.5	44.5	53.8	60.1
Recall	63.9	82.0	76.2	70.2	63.7
Specificity	63.8	78.7	74.0	69.0	63.4
Precision	90.9	90.5	90.7	90.8	90.6
F_1 score	72.7	44.7	55.8	64.4	70.0
F_2 score	65.7	35.5	46.5	56.0	62.5

Table 3: Metrics about the selected models after learning the Near Miss-ed training set for 100 times.

	GNB	Tree	LR	SVM	kNN
Accuracy	70.8	60.5	51.1	58.9	59.7
Recall	56.1	57.7	68.2	63.0	67.4
Specificity	57.1	57.8	67.0	62.7	66.9
Precision	90.7	90.0	90.4	90.5	91.0
F_1 score	78.2	70.3	62.2	69.0	69.7
F_2 score	72.8	63.0	53.4	61.3	62.0

Table 4: Metrics about the selected models after learning the SMOTE-ed and Near Miss-ed the training set for 100 times.

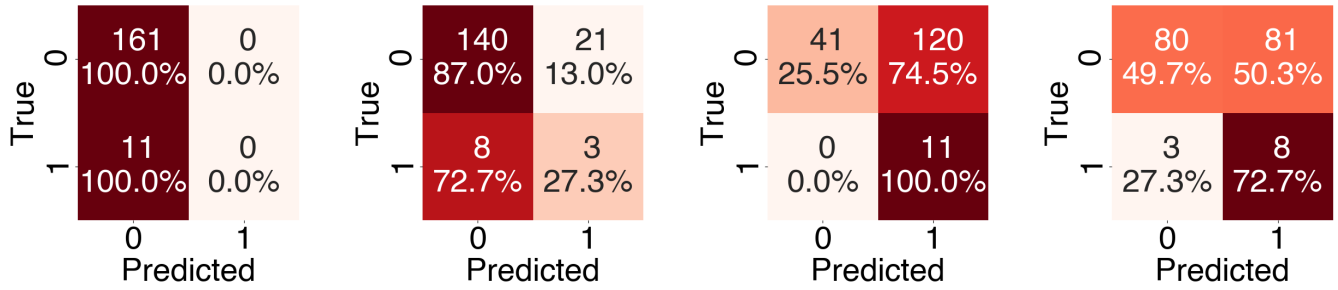
From Table 2, if we SMOTE-ed the training set first, GNB focused on identifying the positive cases and brought a lot of false alarms, thus the positive recall, weighted specificity and precision is high but the overall accuracy is low. The other 4 models do not do so as much as GNB, so their recall and specificity are lower but the accuracy is a lot higher, in which kNN gives the highest recall, specificity, precision and AUC score. If we Near Miss-ed the training set first, GNB is also okay in terms of the metrics found in Table 3, while the decision tree becomes the model that found the most positive cases with the highest positive case recall while also giving a lot of false alarms (thus the overall accuracy is the lowest).

From Table 4, if we first SMOTE-ed then Near Miss-ed the standardized training set before letting the models learn the data, then GNB still ranks 1$^{\text{st}}$ in overall accuracy, weighted F_1 score and weighted F_2 score but its recall and specificity, despite > 0.5, is the lowest. If we consider accuracy, recall, specificity and precision at the same time, kNN ($k = 5$) might have the most all-rounded performance. Combining with the AUC score in Figure 1, once performing SMOTE on the positive cases and Near Miss on the negative cases in the standardized training set, we would suggest using kNN to do the prediction on the Biopsy result.

The difference between the observations by Choudhury et al. (2018), who suggested GNB is the worst among the 5 models, and our observations is probably because we use Near Miss to fix the case ratio instead of letting Tomek Links to delete certain negative data with certain randomness

from using SMOTE to synthesize the positive data. Different training–test split (ours are around 80 : 20 while Choudhury et al. (2018) was around 88 : 12) might also explain the difference between the performance of the models.

As for the random forest, after first by splitting the whole data set into the training set and the test set, the hyper-parameters that returns the best recall to the training set are `max_features` $= \log_2$ of the number of features of the input rounded up to the nearest integer (i.e. $\lceil \log_2 28 \rceil$ after cleansing and imputation), `n_estimators` $= 10$, `max_depth` $= 5$, and `min_samples_split` $= 2$. Using this set of hyper-parameters, the tuned random forest produces the confusion matrices annotated with empirical values plus true-label-wise percentage values instead of considering the whole test set under all the situations as in Figure 2.



(a) Nothing (b) SMOTE (c) Near Miss (d) Both

Figure 2: The confusion matrices of the random forest about the test set ($N = 172$) when the training set is over-/under-sampled as indicated above. The percentages are true-label-wise.

The metrics (in %) when the training set is either over-sampled by SMOTE, under-sampled by Near Miss, both or none are shown in Table 5.

Over-/Under-sampling?	nothing	SMOTE	Near miss	both
Overall accuracy	93.6	83.1	30.2	51.2
Weighted precision	0.0	12.5	8.4	9.0
Positive recall	0.0	27.3	100.0	72.7
Weighted specificity	6.4	31.1	95.2	71.3
Weighted F_1 score	0.0	17.1	15.5	16.0
Weighted F_2 score	0.0	22.1	31.4	30.1
Overall AUC	53.4	56.2	78.2	73.9

Table 5: The metrics (in %) of the random forest on the test set ($N = 172$) with(out) over-/under-sampling the training set.

Now that with(out) SMOTE on the positive cases and/or Near Miss on the negative cases, the precision weighted by class size of all 4 situations are low. The precision of the situation having Near Miss is lower than only having SMOTE because the forest identified more than half of 11 positive cases while giving many false alarms, as illustrated in Figure 2. We considered it would be more important not to give false negative predictions to the cell anomaly in the cervix, so we also calculated the F_2 score that weigh recall more than precision for our own reference.

Given the precision is low in all the situations, simply performing SMOTE before training the forest might be the best way to provide the prediction with its accuracy $> 80\%$, although the accuracy is mainly on the negative cases. Performing Near Miss (with or without SMOTE beforehand) gives decent metrics in positive recall and weighted specificity, but the data treatment also triggered many

false alarms, which lowers the overall accuracy a lot from simply performing SMOTE on the training set, and hinders the credibility of the forest.

4.2 Prediction and Causal Inference Using Bayesian Network

Note that, though, we can use the Bayesian network to do prediction, the power of the Bayesian network is more on interpretation of the relationship among variables, and in some interesting cases, we can interpret some connections as (probabilistic) causal relationship. Section 4.2.1 will show some prediction results and Section 4.2.2 will show the causal analysis of the dataset.

4.2.1 Prediction

We conduct 20 replications on the order MCMC using the SMOTE over-sampled cervical cancer dataset. Here is the procedure on the prediction and evaluation: In each replication,

- We first discretize the continuous variables in the dataset using 0, 25, 50, 75 and 100-th percentile of each variable as break points. We discretize the variables because learning a network with mixture types (discrete and continuous) is complicated, it would be more interpretable and simple by working at pure type of variables.
- 80% of the dataset is used to learn the Bayesian network and 20% of data is used as test set.
- The MAP graph is used to conduct the prediction. Let e_i be the i-th case in the test set. The probability $P(\text{Biopsy} = 1|E = e)$ is estimated using LW sampling.

For each replication, ROC curves is plotted by changing the threshold c in the classification rule that assigning Biopsy $= 1$ whenever $P(\text{Biopsy} = 1|E = e) > c$. All ROC curves are shown in Figure 3. The curves are generally in good shape. Figure 4 shows the boxplot of the distribution of AUC when we first over-sample the dataset using SMOTE, accuracy, sensitivity, specificity, precision and F-measure of the predictions in 20 replications. Except AUC, all other measures use the cut-off $c = 0.5$. The median AUC is 87.9%, the median accuracy is 81.8%, the median sensitivity is 73.3%, the median specificity is 90%, the median precision is 84% and the median F-measure is 72.4%. The specificity is higher than the sensitivity indicates that the model performs better in predicting negative cases, nevertheless, the accuracy is still high.

Figure 5 shows the true positive rates and specificity of training set (denoted by `train_TPR` and `train_Specificity` in the graph) and test set (denoted by `test_TPR` and `train_Specificity` in the graph), showing that the model has good balance between false positive and false negative (for example, choosing $c \approx 0.375$ can allow the test sensitivity and specificity to be equal and is equal to around 0.75). We can further conduct cross-validation for choosing another c in order to improve the sensitivity of the model, but we will not do it in this project since this is not our main interest for the use of Bayesian networks. The AUC is comparable to the performances of other models mentioned in Section 4.1. Instead of plain prediction, we could further analyze the edge features of the learnt network, which will be discussed more detailedly in the next sections.

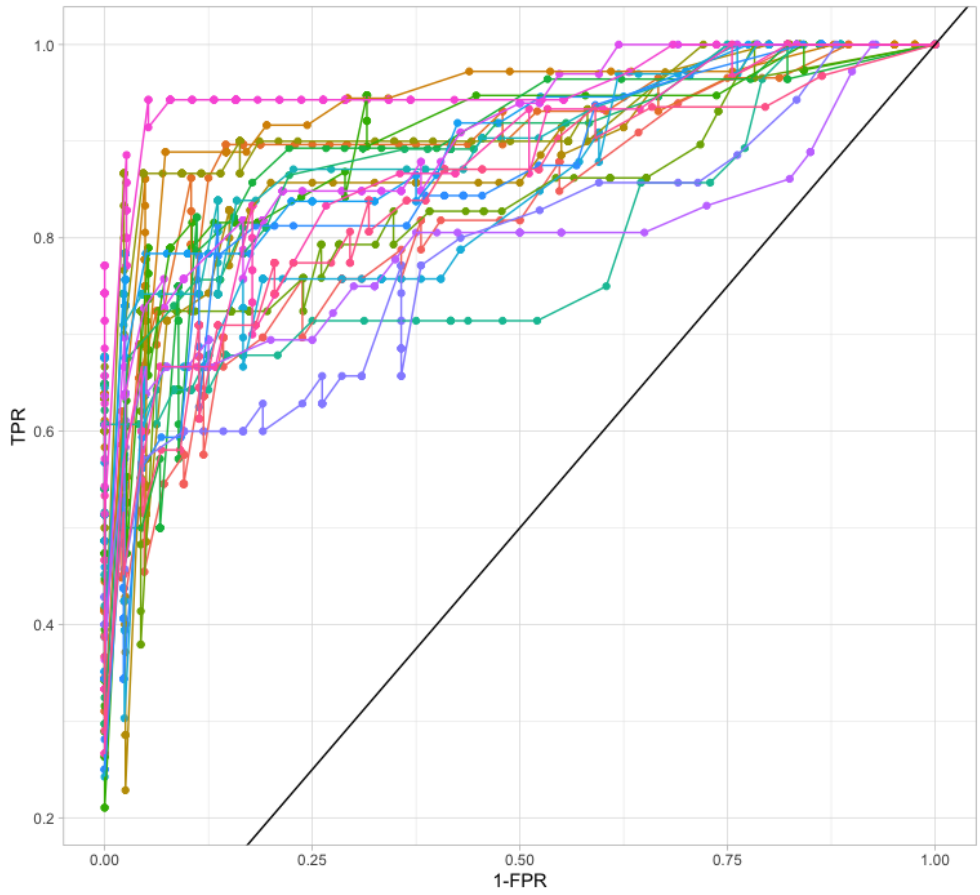


Figure 3: ROC curves of 20 replications. Colors indicate different replications.

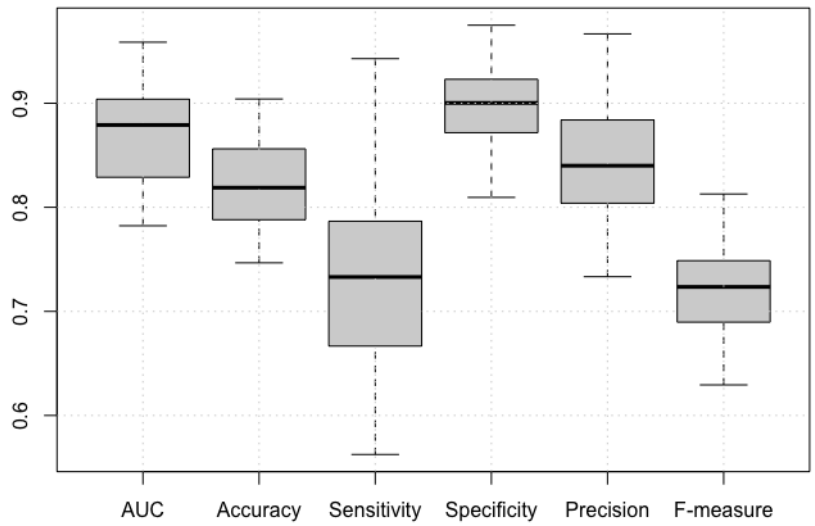


Figure 4: Boxplot of the AUCs of the ROC curves of 20 replications with SMOTE conducted before train test partition.

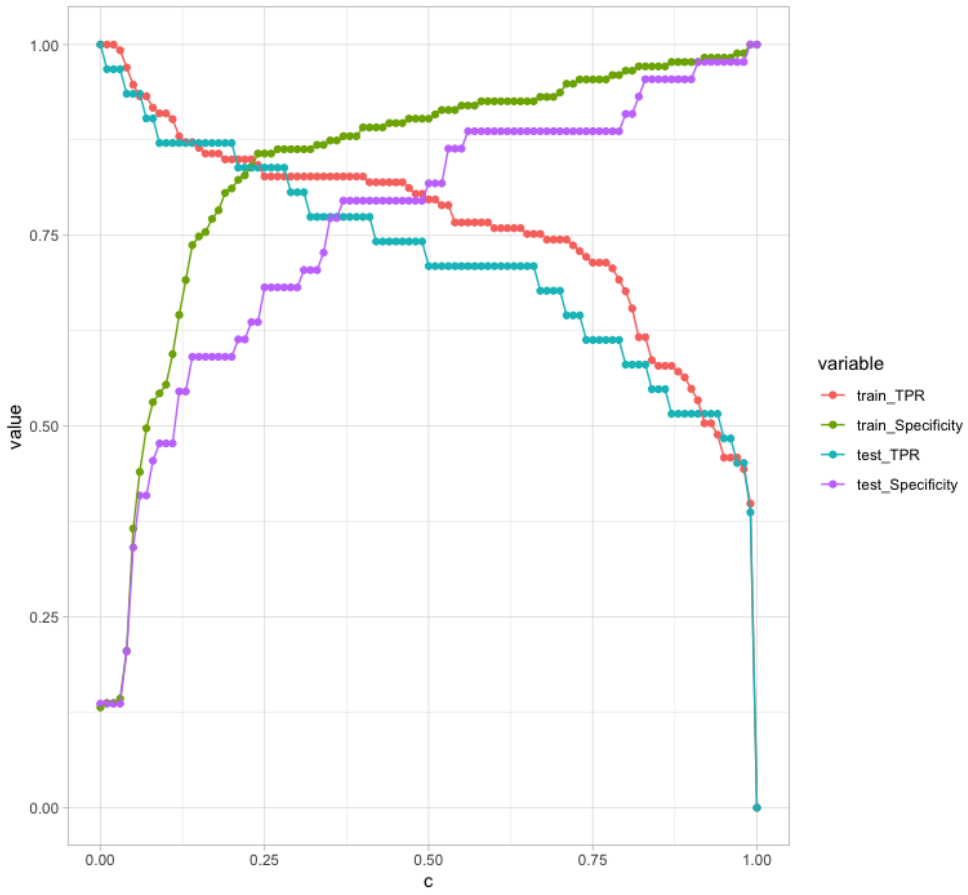


Figure 5: True positive rate and specificity of training and test sets using different cut-offs.

4.2.2 Causal Inference

In this section, we will interpret the learnt network using the original (non-over-sampled), discretized dataset. The discretization is done manually instead of using the quantiles mechanically since we want the partition to be meaningful. For example, using quantile discretization, the variable `Age` would be discretized into four classes: [14,22], (22,29], (29,35] and (35,52]. Obviously the fourth class is way to wide. The discretization rules are described in Table 6.

To ensure the stability of the algorithm, we conduct the order MCMC for 50 times and the their time series of BDeu scores defined in Equation 5 are plotted in Figure 6. The time series are stationary after 100 iterations and all chains are moving around $-10,400$, which shows that the order MCMC structural learning is stable. We pick the graph that has the highest BDeu score out of the samples in 50 replications to conduct analysis. The maximum score graph is shown in Figure 7.

Before we start interpret the network, we first note that not all arcs in the network can be explained causally. It is expected since there are a lot of variables and we cannot ensure every arcs contain casual meaning. For example, the arc `STDs.Hepatitis.B`→`Smokes` does not make any causal sense. Also note that, there exists other graphs that have the same dependence structure, i.e., same factorization of the unconditional distribution, but with different arc directions (that is called *equivalent class* of a graph), and thus we need to interpret the result carefully. For example, considering three variables X, Y and Z. The triple $X \to Y \leftarrow Z$ and The triple $X \to Y \to Z$ have same dependence structure that X and Z are independent conditional to Y. Here are some observations from the MAP graph shown in Figure 7.

(i) Before we interpret our main results, we can see that the graph successfully put all relationships that are trivial together; for example, `Smokes`, `Smokes years` and `Smokes packs year` are connected, `Hormonal Contraceptives years` and `Hormonal Contraceptives` are also connected,

and the sexual transmitted diseases (STD) related variables are also clustered together. As such, we are confidence that the graph can make causal sense for most of the arcs, so that we have confidence to interpret the relationships we are interested.

(ii) The only parent of the variable `Biopsy` is `Dx.HPV`. Thus, the only factor controlling the Biopsy result is the diagnosis of the HPV, meaning that conditional to the diagnosis result, the Biopsy is independent to all other non-descendants, except the variable `STDs.genital.herpes`, which is the children of `Biopsy`. It could also indicate that the Biopsy is also useful for screening genital herpes. Here, we support that HPV is a risk factor for the cervical cancer, which is well-known in the literature.

(iii) `Smoke` and `IUD` (Intrauterine Device, a device that put inside the uterus of the patient for birth control) have no directed edges to `Biopsy`, `STDs.HPV`, `Dx.HPV` and `Biopsy`, which indicates that smoking and the use of IUD are not risk factors for the HPV and cervical cancer. However, Smokes have direct impact to `Dx`, the diagnosis of cancer, including CIN (Cervical intraepithelial neoplasia, the abnormal growth of cells on the surface of the cervix).

(iv) There is a directed edge between `IUD` and `Dx.Cancer`, meaning that using IUD would be a risk factor of cancer.

(v) We also see that there is a directed edge from `Dx.HPV` to `Dx.Cancer`, meaning that having HPV may also associated with the risk of having other cancers.

However, using a single graph may not be credible. It is more reliable if we can put the information of all graphs sampled in an order MCMC run. We select the sampled graphs from the replication containing the MAP graph using Equation 8 for each possible edge $X_i \to X_j$. To visualize $n(n-1) = 29(28) = 406$ pairs of possible edge features effectively, we plot the estimated edge features in the heatmap in 8. To show that the estimates are precise enough, we can calculate the 95% credible interval for each of the estimated edge features using Bootstrap. To visualize it, we order the non-zero average estimated edge features and plot their credible intervals in Figure 9. We can see that the intervals are quite narrow, so that we can conclude that the estimations are quite precise. Again, we can observe clusters in Figure 8 for some groups of trivially related variables, showing that the results make logic sense.

Go back to the observation we discussed using the maximum score graph, we can use the probability of edge features to evaluate the strength of the causal relationship, and hence, estimate the probability conditional to the risk factors to see how the risk factors affect our target variables.

(i) Is HPV a risk factor of cervical cancer? Relationship between `Biopsy` and `Dx.HPV`. The average of $\hat{P}(\text{Dx.HPV} \to \text{Biopsy}|D)$ is 0.3126, which shows a moderate causal relationship between two variables. It is well-known that HPV is a risk factor of cervical cancer. Furthermore, we estimate the probabilities $P(\text{Biopsy} = 1|\text{Dx.HPV} = x)$, $x = 0, 1$, to justify our result. We have $P(\text{Biopsy} = 1|\text{Dx.HPV} = 0) = 0.05846$ and $P(\text{Biopsy} = 1|\text{Dx.HPV} = 1) = 0.33486$, showing that `Dx.HPV` greatly increases the risk of having cervical cancer by 5.728 times.

(ii) Are smoking and IUD risk factors to the HPV diagnosis? The average of $\hat{P}(\text{Smokes} \to \text{Biopsy}|D)$ and $\hat{P}(\text{IUD} \to \text{Biopsy}|D)$ are both zero, indicating that smoking and the use of IUD are not risk factors to cervical cancer.

(iii) Is IUD a risk factor for diagnosing cancers (not restricted to cervical cancer)? The average of $\hat{P}(\text{IUD} \to \text{Dx.Cancer}|D)$ is 0.82518, showing that the use of IUD indeed is a strong risk factor to diagnosing cancer. More specifically, in order to draw conclusion that whether or not the risk factors have positive or negative effects, we use the graph that has maximum BDeu score to estimate the probability $P(\text{Dx.Cancer} = 1|\text{IUD} = x)$ for $x = 0, 1$ using LW sampling. We have $P(\text{Dx.Cancer} = 1|\text{IUD} = 0) = 0.018$ while $P(\text{Dx.Cancer} = 1|\text{IUD} = 1) = 0.046$, which indicates that the use of IUD increases risk of having cancer by 2.556 times.

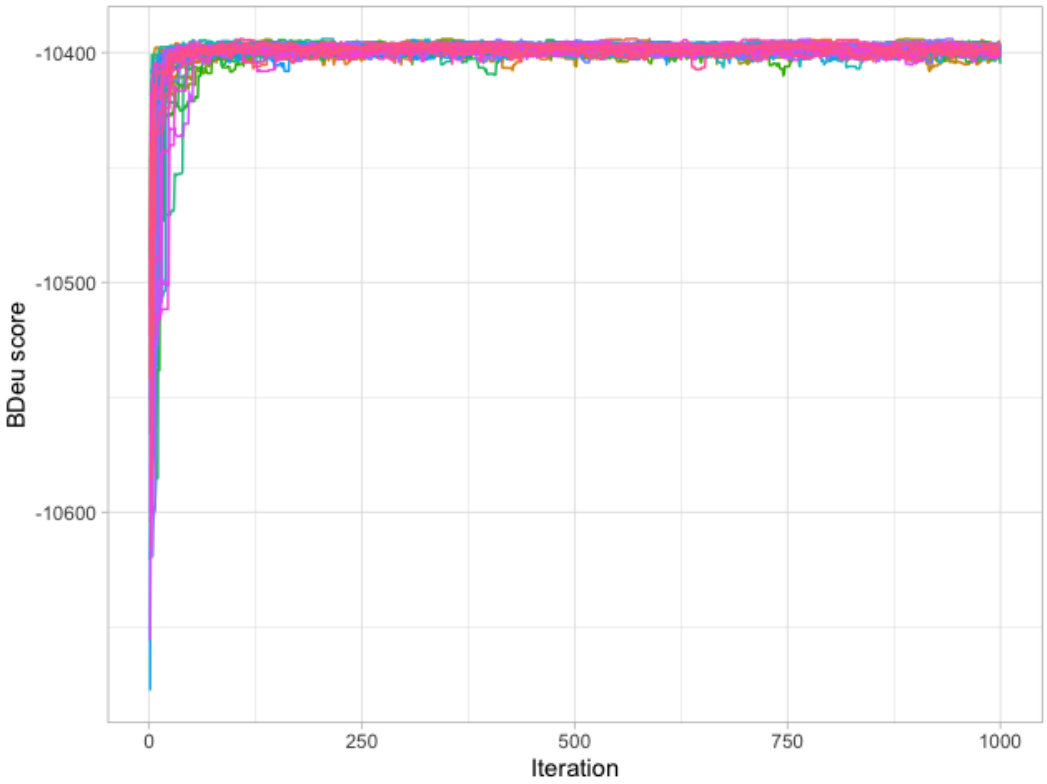


Figure 6: The time series plots of the BDeu scores of 50 replications.

(iv) Is HPV a risk factor to other cancers? The average of $\hat{P}(\text{Dx.HPV} \to \text{Dx.Cancer}|D)$ is 0.8601, showing that it is a strong risk factor. We estimate $P(\text{Dx.Cancer}|\text{Dx.HPV} = 0) = 0.00246$ and $P(\text{Dx.Cancer}|\text{Dx.HPV} = 1) = 0.86858$. The risk of having cancer is 352.47 times for a patient who has HPV. The impact of HPV to cancer is extremely substantial.

Variables	Classes
Age	(0,10] (10,15] (15,20] (20,25] (25,30] (30,35] (35,40] (40,45] (45,50] (50,70] (70,90]
Number of sexual partners	1 2 3 4 5 6 (6,8] (8,28]
First sexual intercourse	(0,12] (12,14] 15 16 17 18 (18,20] (20,22] (22,24] (24,26] (26,28] (28,32]
Num of pregnancies	0 1 2 3 4 5 6 7 (8,11]
Smokes (years)	0 1 2 3 4 5 [6,7] [8,9] 10 (10,13] (13,15] (15,20] (20,24] (24,28] (28,37]
Smokes (packs/year)	0 1 2 3 4 5 6 [7,8] (8,15] (15,37]
Hormonal Contraceptives (years)	0 1 2 3 4 5 6 7 8 9 10 (10,12] (12,14] (14,16] (16,20] (20,30]
IUD (years)	0 1 2 3 (3,5] (5,7] (7,9] (9,12] (12,19]

Table 6: The manual discretization rule of the contionuous variables.

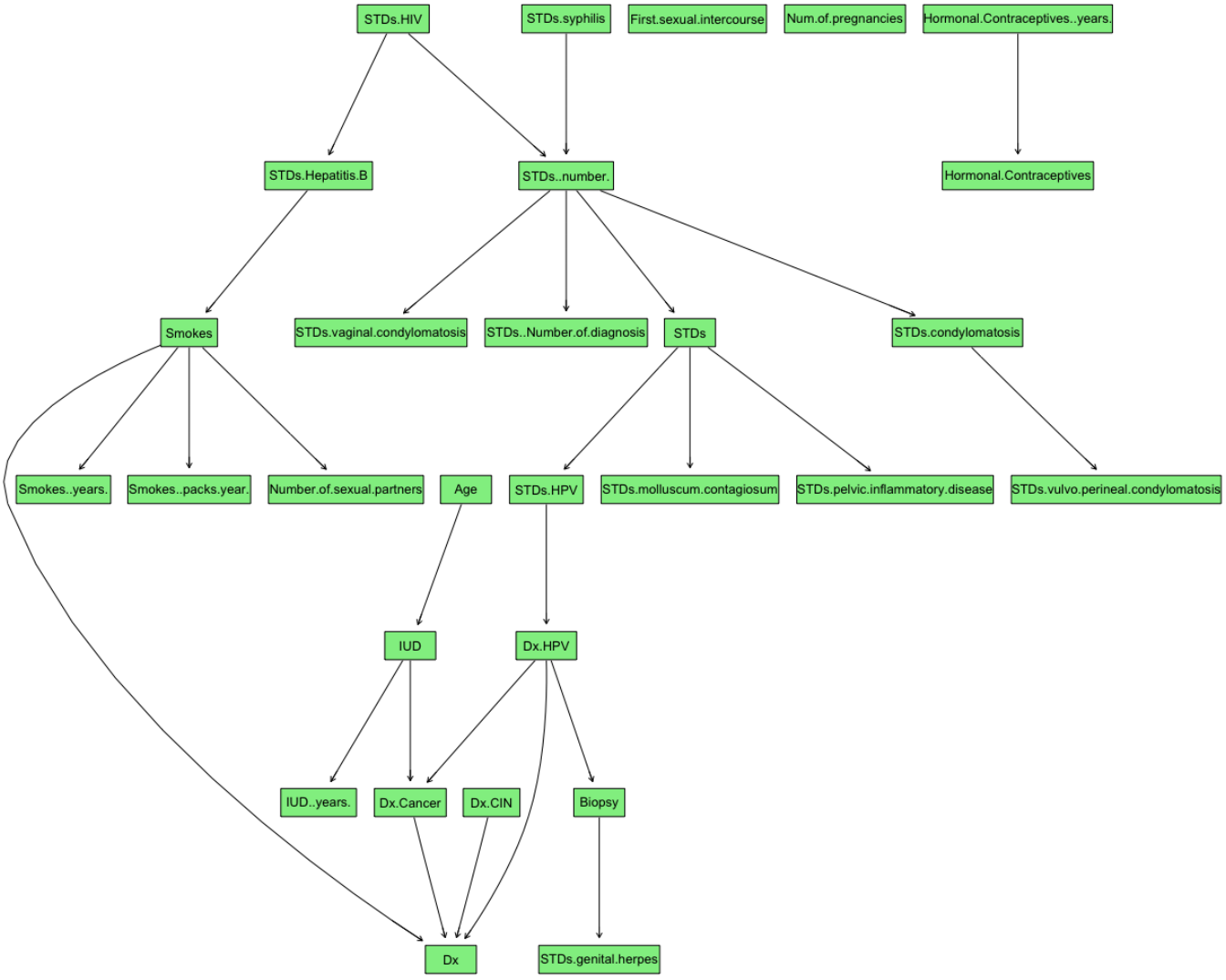


Figure 7: The graph that has the highest BDeu out of the sampled graphs of 50 replications.

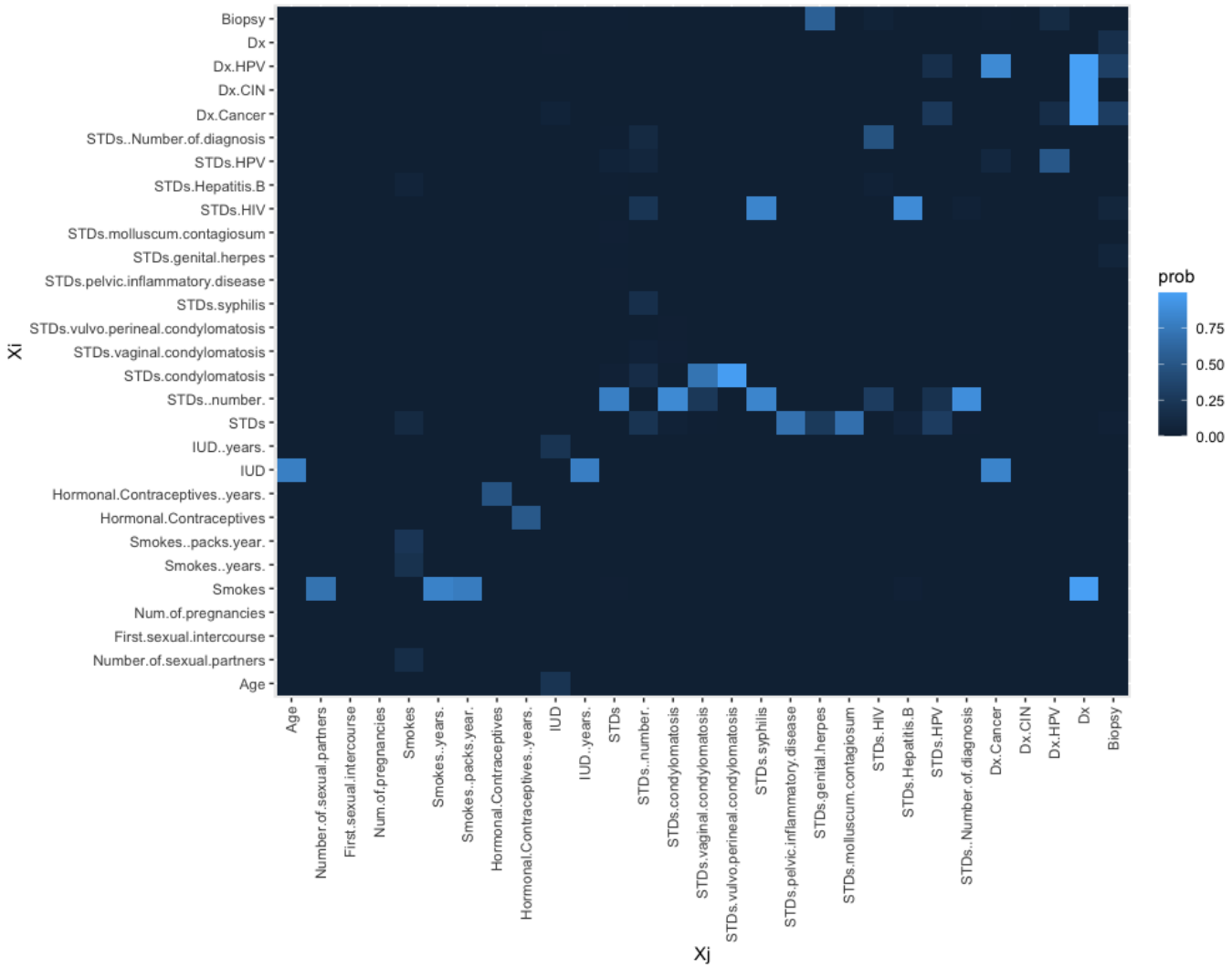


Figure 8: The heatmap of the average estimated posterior probability $P(X_i \to X_j|D)$ of the edge features using 50 replications. The y-axis is X_i and the x-axis is X_j.

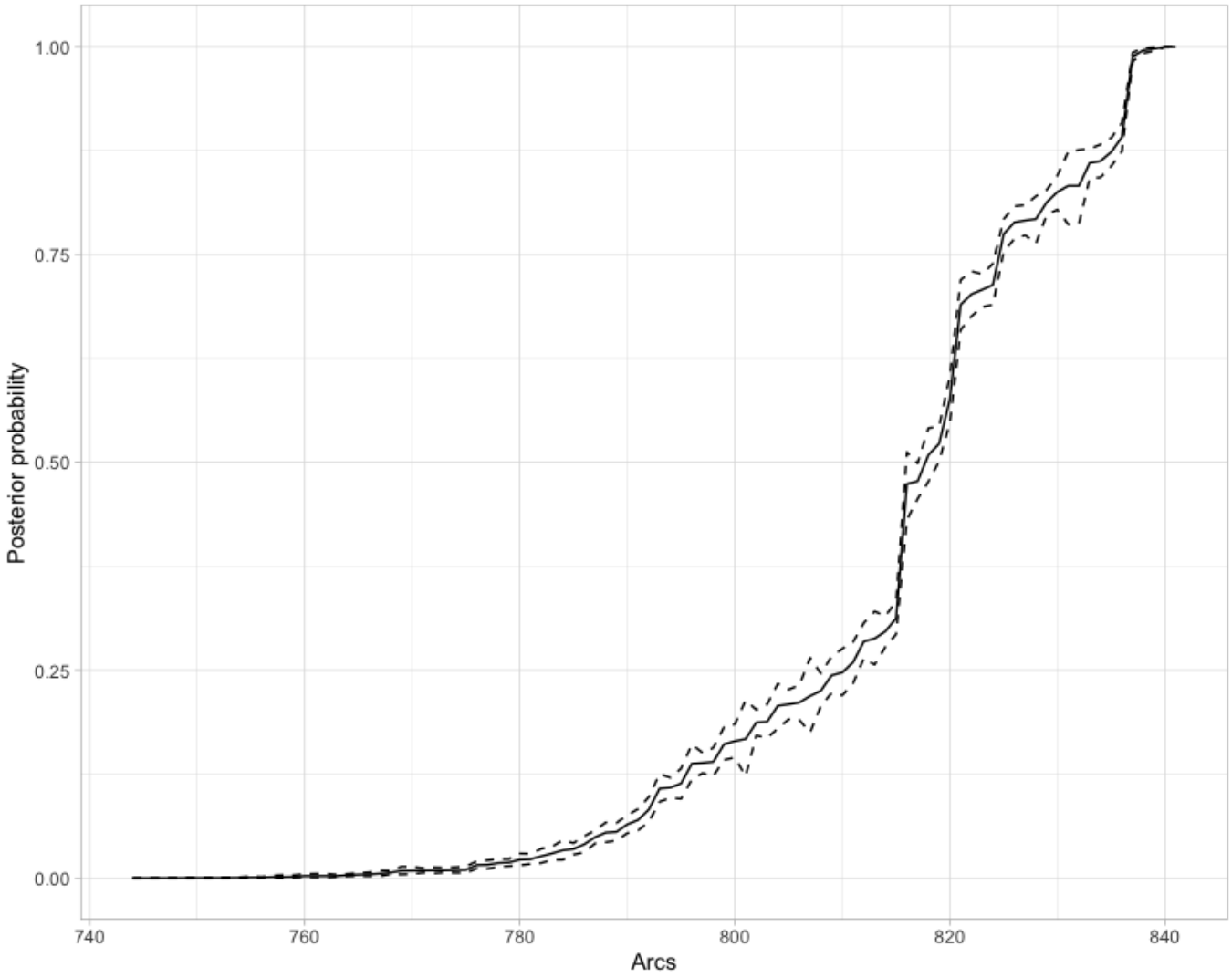


Figure 9: The ordered non-zero edge features with the credible intervals.

5 Conclusion

We tried straightforward prediction on Biopsy result using classification models with standardized training set with different over-/under-sampling treatments. We compared the (weighted) metrics and/or the confusion matrices of different models from differently treated data set. If we over-sample the positive cases then under-sample the negative cases before training our models, we would suggest using 5-Nearest Neighbors to do the prediction as we consider the accuracy, recall, specificity, precision and the area under the ROC curve at the same time. We also tried planting random forests with hyper-parameters tuned by grid searching 7-fold cross validation. For random forest, we suggest only over-sample the standardized training set before planting because the corresponding forest detects certain positive cases while maintaining high accuracy and not giving a lot of false alarms.

We also fit Bayesian networks using the manual discretized version of dataset. We found that the prediction has similar performances as using the other machine learning model mentioned in this paper. The difference is we can also interpret the network instead of predicting only. The sampled networks show correct clustering for some trivial relationships which gives us confidence to interpret some other relationships using these samples. We found that the only factor directly cause Biopsy results is the HPV, which is a well-known result. We also found that the Biopsy results may also be useful for screening genital herpes. Smoking has no direct effect to the Biopsy result. Relationships other than Biopsy also discovered. use of IUD is a risk factor that increase risk for the diagnoses of cancers. HPV is a risk factor for diagnosing cancers, not restricted to cervical cancers.

Contributions of the members

Shun Hin Chan: Section 1-2, 3.2, 3.3, 4.2, 5 (Bayesian networks related);
Ki Wai Fong: Section 1-2, 3, 3.1, 4.1, 5 (Prediction models related).

References

Cervical cancer (risk factors) data set. (2017). *UCI Machine Learning Repository*. Retrieved from https://archive.ics.uci.edu/ml/datasets/Cervical+cancer+(Risk+Factors)

Choudhury, A., Wesabi, Y. M. S. A. & Won, D. (2018). Classification of cervical cancer dataset. *CoRR*, *abs/1812.10383*. arXiv: 1812.10383. Retrieved from http://arxiv.org/abs/1812.10383

Danquah, R. A. (2020). Handling Imbalanced Data: A Case Study for Binary Classification Problems. https://doi.org/10.6084/m9.figshare.13082573.v2

Friedman, N. & Koller, D. (2001). Being bayesian about network structure: A bayesian approach to structure discovery in bayesian networks. *Mach Learn*, *50*. https://doi.org/10.1023/A:1020249912095

Lemaître, G., Nogueira, F. & Aridas, C. K. (2017). Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning. *Journal of Machine Learning Research*, *18*(17), 1–5. Retrieved from http://jmlr.org/papers/v18/16-365.html

Madigan, D., York, J. & Allard, D. (1995). Bayesian graphical models for discrete data. *International Statistical Review / Revue Internationale de Statistique*, *63*(2), 215–232. Retrieved from http://www.jstor.org/stable/1403615

Mani, I. & Zhang, I. (2003). Knn approach to unbalanced data distributions: A case study involving information extraction. In *Proceedings of workshop on learning from imbalanced datasets* (Vol. 126). ICML United States.

Muibideen, M. & Prasad, R. (2020). A fast algorithm for heart disease prediction using bayesian network model. arXiv: 2012.09429 [cs.LG]

Park, E., Chang, H.-j. & Nam, H. S. (2018). A bayesian network model for predicting post-stroke outcomes with available risk factors. *Frontiers in Neurology*, *9*, 699. https://doi.org/10.3389/fneur.2018.00699

Razali, N., Mostafa, S. A., Mustapha, A., Wahab, M. H. A. & Ibrahim, N. A. (2020). Risk factors of cervical cancer using classification in data mining. *Journal of Physics: Conference Series*, *1529*, 022102. https://doi.org/10.1088/1742-6596/1529/2/022102

Scutari, M. (2010). Learning bayesian networks with the bnlearn r package. *Journal of Statistical Software*, *35*(3), 1–12. Retrieved from http://www.jstatsoft.org/v35/i03/

Suter, P. & Kuipers, J. (2021). *Bidag: Bayesian inference for directed acyclic graphs*. R package version 2.0.0. Retrieved from https://CRAN.R-project.org/package=BiDAG

WHO. (2021). Cervical cancer. Retrieved from https://www.who.int/health-topics/cervical-cancer