Women's Reproductive Health and How Supervised Machine Learning Goes In Deep

Alex Biskis
alexandrasbiskis@lewisu.edu
Victoria Griffin
victoriakgriffin@lewisu.edu
DATA-51000-002 Fall 1
Data Mining and Analytics
Lewis University

I. INTRODUCTION

In the field of data mining, it is only natural for supervised machine learning (SML) concepts, and the most common algorithms used in the profession, to come up. These algorithms aid in organizing data, solving complex problems, and, in the end, attaining the answers that are highly sought after. The steps taken to prepare the selected dataset for analysis and the classification processes via machine learning will be discussed throughout the entirety of this paper. The purpose of this assignment was to select a dataset capable of classification, apply preprocessing to said dataset, use cross-validation with at least three SML algorithms to tune hyperparameters, and finally test the tuned models and compare algorithmic performance metrics, such as accuracy, precision, recall, and F1-score, between the three selected models. After computing performance metrics, visuals of the results must be created.

The dataset for this paper was found from the University of California at Irvine's (UCI) machine learning repository [1]. Within the parameters given, the dataset is of appropriate age (within the past decade), relevance, and size. The dataset was originally donated by the Hospital Universitario de Caracas in Venezuela, and comprises demographic information, habits, and historic medical data from 858 patients that were screened for cervical cancer. The dataset was first published in 2017 by Fernandes et al., titled *'Transfer Learning with Partial Observability Applied to Cervical Cancer Screening'*, the same year the dataset was donated to the UCI's repository for public use [2].

There were several good choices for possible datasets to use for this assignment, but in the end this dataset was chosen due to the unique and varied personal connections to cervical cancer the group members each had. One group member works closely with cancer research as part of their career, and the other group member has personally experienced a high-dysplasia of cervical tissue which warranted preventative procedures as it related to cervical cancer. As an added comment, this dataset also aligns with the group members concentration of bioinformatics and computational biology.

At its most basic form, machine learning is a method by which mathematical algorithms learn through experience to provide a way to measure how well it has 'learned' something and how well it may perform in the future on similar data. To put this into perspective using the cervical cancer dataset from UCI, SML models will be used to see to what degree they can correctly identify patients who have been known to have a cervical cancer diagnosis in the dataset. The idea behind this is to potentially take these tuned models and apply them onto patients who do not yet have a diagnosis but may be at risk based on their demographic information, habits, and historical medical data that the models were trained upon. Ideally, SML can provide a quick analysis that offers accurate and conclusive information on if the algorithm used is trained properly. SML is primarily used for datasets that have features as well as a class label, such as a cancer diagnosis or non-cancer diagnosis. Hyperparameters that are used by SML models are often seen to be tuned through what is called a cross-validation process. A 10-fold cross-validation is customary in the field as it enables a sufficient amount of variation in training/testing dataset splitting, with each fold being selected as the testing fold at least once [6]. Once tuned, SML models with the selected hyperparameters from the cross-validation are then applied onto the single training/testing dataset that is split randomly into two respective sets. Usually, 80% of the original dataset is used for training and learning purposes, while the remaining 20% is used for testing how well the algorithm learned (i.e. performance metrics). Cutting edge applications of SML include facial recognition, medical diagnosis, stock price predictions, housing market predictions and more.

There are many different SML algorithms. Specifically, this paper will investigate the K-Nearest Neighbors (KNN), Random Forest (RF), and Support Vector Machine (SVM) algorithms [3][4][5]. Specific cross-validation metrics this paper uses to tune hyperparameters include classification accuracy (CA) and ROC area under the curve (AUC). After cross-validation and hyperparameter tuning the confusion matrix is further analyzed to assess performance metrics such as accuracy, precision, recall, and F1-score, between the three selected models. More details about this will be discussed in section III, methodology, and section IV, results and discussion.

The sections of this report cover different stages of the analysis. Within Section II, the dataset is described more in detail with information ranging from the types of variables, distributions of variables, and other significant information.

Section III explains the methodology of the analysis in detail and the tools utilized to carry it out. Section IV dives into the results from the tuned hyperparameters of each SML model and discusses the findings. Section V wraps up the analysis with final thoughts and future recommendations/implications.

II. DATA DESCRIPTION

The size of the cervical cancer dataset is approximately 19 kB[1]. The dataset was collected and published in 2017. The original dataset contains 858 observations with 36 attributes. Due to the large number of observations and attributes, some were excluded, due to either large amounts of missing data or irrelevance. After cleaning the data, only 726 observations (patients) and 28 attributes remained. For the reasons stated above, the following attributes were eliminated: 'STDs: Time since first diagnosis', 'STDs: Time since last diagnosis', 'STDs: Cervical condylomatosis', 'STDs: AIDS', 'Hinselmann', 'Schiller', 'Cytology', and 'Biopsy'. All in all, 132 patients were excluded from the dataset. Table 1 shows the attribute, data type, and gives an example with a more detailed description of the cleaned dataset. Table 2 shows the descriptive statistics of the nominal variables in the cleaned dataset, while Fig 1 in Appendix A shows the histograms.

Table 1: Summary of Cervical Cancer Dataset

Attribute	Type	Example	Description
Age	Nominal	18, 24, 35, etc.	Age of patient in years
Number of Sexual Partners	Nominal	2, 3, 4, etc.	The total number of sexual partners
First Sexual Intercourse	Nominal	17, 18, 20, etc.	Age of first sexual intercourse
Number of Pregnancies	Nominal	1, 2, 3, etc.	Total number of pregnancies
Smokes	Categorical	0 (no) or 1 (yes)	Does the patient smoke
Smokes (years)	Nominal	5, 10, 12, etc.	Total years patient has smoked
Smokes (packs/year)	Nominal	30, 42, etc.	How many packs/year patient smokes
Hormonal Contraceptives	Categorical	0 (no) or 1 (yes)	Does the patient use hormonal contraceptives
Hormonal Contraceptives (years)	Nominal	5, 10, 12, etc.	Total years patient has used hormonal contraceptives
IUD	Categorical	0 (no) or 1 (yes)	Does the patient use an IUD
IUD (years)	Nominal	5, 10, 12, etc.	Total years patient has used an IUD
STDs	Categorical	0 (no) or 1 (yes)	Does the patient currently have an STD
STDs (number)	Nominal	1, 2, 3, etc.	Total number of current STDs
STDs:condylomatosis	Categorical	0 (no) or 1 (yes)	Does the patient have this STD
STDs:vaginal condylomatosis	Categorical	0 (no) or 1 (yes)	Does the patient have this STD
STDs:vulvo-perineal condylomatosis	Categorical	0 (no) or 1 (yes)	Does the patient have this STD
STDs:syphilis	Categorical	0 (no) or 1 (yes)	Does the patient have this STD
STDs:pelvic inflammatory disease	Categorical	0 (no) or 1 (yes)	Does the patient have this STD
STDs:genital herpes	Categorical	0 (no) or 1 (yes)	Does the patient have this STD
STDs:molluscum contagiosum	Categorical	0 (no) or 1 (yes)	Does the patient have this STD
STDs:HIV	Categorical	0 (no) or 1 (yes)	Does the patient have this STD
STDs:Hepatitis B	Categorical	0 (no) or 1 (yes)	Does the patient have this STD
STDs:HPV	Categorical	0 (no) or 1 (yes)	Does the patient have this STD
STDs: Number of diagnosis	Nominal	1, 2, 3, etc.	Total number of STDs patient has ever had
Dx:Cancer	Categorical	0 (no) or 1 (yes)	Does the patient have a history of cancer
Dx:CIN	Categorical	0 (no) or 1 (yes)	Does the patient have a CIN diagnosis
Dx:HPV	Categorical	0 (no) or 1 (yes)	Does the patient have a HPV diagnosis
Dx (Target Variable)	Categorical	0 (no) or 1 (yes)	Does the patient have a cervical cancer diagnosis

Table 2: Statistical Description of the Dataset

Attribute	Mean	Standard Deviation	Min/Max
Age	27.58	8.73	13/84
Number of Sexual Partners	2.51	1.62	1/28
First Sexual Intercourse	17.09	2.83	10/32
Number of Pregnancies	2.31	1.42	0/11
Smokes (years)	1.24	4.18	0/37
Smokes (packs/year)	0.46	2.31	0/37
Hormonal Contraceptives (years)	2.23	3.65	0/22
IUD (years)	0.51	1.95	0/19
STDs (number)	0.15	0.54	0/4
STDs: Number of diagnosis	0.08	0.302	0/3

III. METHODOLOGY

The first step taken was a preliminary investigational/statistical analysis of the dataset using Python version 3.9.8 throughout the entire process. Other various libraries were utilized throughout the analysis as well, such as Pandas version 1.5.0, Numpy version 1.23.3, and Sklearn version 1.12.0. Sample and background information of the dataset were taken using Pandas to gain a look into the variables, data types, and how many missing values there were. It was found through the pandas.info() function that the results were saying that there were no missing values even though the sample taken showed there were missing values present. The issue was that when the dataset was compiled, the researchers used a '?' symbol to denote a missing value instead of a NaN value. This issue was remedied by replacing all '?' symbols with actual missing NaN values.

The second step taken was preprocessing the dataset. The UCI machine learning repository gave an in-depth background of variable information to know that 26 out of 36 variables were boolean, and 10 out of 36 variables were integer values. The issue that arose is that boolean variables sometimes do not offer any valuable information. For example, if a boolean variable's observations are all either '0' for 'no' or '1' for 'yes', then that variable does not have any uniqueness and does not offer any value to the analysis. Two boolean variables, 'STDs:cervical condylomatosis' and 'STDs:AIDS', fit this criterion and were removed. The following step in preprocessing was to start addressing the missing values throughout the dataset. This was done in a few different ways such as deleting the columns that had missing values, deleting the rows that had missing values, and imputation (mean) methodology. Two columns, 'STDs: Time since first diagnosis' and 'STDs: Time since last diagnosis', were seen to have a missing value percentage of about 92%. Given the sensitive nature of the questionnaire given to patients, it was understandable that the rate of missing values for these two variables were high. However, this also meant that these two variables needed to be removed in order to find all rows that have at least one missing value, otherwise the number of rows returned would be very high and not representative of what was actually going on. Therefore, the two aforementioned variables were removed.

Next, boolean variables were once again revisited, but this time from the perspective of the rows. Using Pandas, rows whose boolean variable data points had at least one missing value were removed, for a total of 132 patients being excluded from the analysis. Next, integer variables were imputed using Sklearn's SimpleImputer() class. After this, the whole dataset was once again reassessed for missing values, at which point none were found. Finally, the integer variables were standardized due to the statistical analysis showing that none were normally distributed.

The third step taken was applying Sklearn's KFold() 10-fold cross-validation to Sklearn's KNN, RF, and SVM algorithms. This crucial step was done to be able to tune selected hyperparameters for each respective algorithm. It is common practice in the field of data science to use a 10-fold cross-validation of the dataset used. For simplicity sake, only ROC AUC and CA were used to compare the three different SML algorithms when tuning.

For the KNN algorithm, the hyperparameters chosen to tune were as follows: number of nearest neighbors and type of distance calculation. As seen in Table 3, after applying the 10-fold cross validation, the value of k=12 for number of nearest neighbors and the Manhattan distance calculation were selected, as denoted in yellow.

Table 3. KNN Hyperparameters After 10-Fold Cross-Validation

k	Distance	ROC AUC	CA	k	Distance	ROC AUC	CA
3	manhattan	0.772	0.977	3	euclidean	0.669	0.973
6	manhattan	0.806	0.973	6	euclidean	0.711	0.973
8	manhattan	0.900	0.972	8	euclidean	0.738	0.972
12	**manhattan**	**0.902**	**0.972**	12	euclidean	0.760	0.972

For the RF algorithm, the hyperparameters chosen to tune were as follows: number of trees in the forest, minimum number of samples required to split an internal node, and minimum number of samples required to be at a leaf node. Table 4 shows all of the hyperparameter combinations used. After applying the 10-fold cross-validation, the value of n=10 for minimum number of trees, n=10 for minimum number of samples required to split an internal node, and n=5 for minimum number of samples required to be a leaf node were selected, as denoted in yellow.

Table 4. Random Forest Hyperparameters After 10-Fold Cross-Validation

Number of trees	Minimum number of samples to split node	Min number of samples to be a node	ROC AUC	CA	Number of trees	Minimum number of samples to split node	Min number of samples to be a node	ROC AUC	CA	Number of trees	Minimum number of samples to split node	Min number of samples to be a node	ROC AUC	CA
10	5	5	0.861	0.975	**10**	**10**	**5**	**0.927**	**0.981**	10	5	10	0.749	0.974
25	5	5	0.872	0.979	25	10	5	0.936	0.978	25	5	10	0.878	0.974
100	5	5	0.939	0.974	100	10	5	0.912	0.971	100	5	10	0.854	0.974

For the SVM algorithm, the hyperparameters chosen to tune were as follows: regularization parameter and kernel. As seen in Table 5, after applying the 10-fold cross validation, the value of C=1.0 for regularization parameter and the 'Linear' kernel were selected, as denoted in yellow.

Table 5. SVM Hyperparameters After 10-Fold Cross-Validation

C	Kernel	ROC AUC	CA	C	Kernel	ROC AUC	CA
0.5	'Linear'	0.997	0.994	0.5	'poly'	0.971	0.972
1.0	**'Linear'**	**0.998**	**0.993**	1.0	'poly'	0.977	0.973
1.5	'Linear'	1.0	0.994	1.5	'poly'	0.978	0.976

Finally, the last step taken was applying the tuned models to the dataset. The dataset was split using Sklearn's train_test_split() class. The split followed an 80% training and 20% testing split. A confusion matrix and classification report which included accuracy, precision, recall, and F1-score, were generated for each tuned model using Sklearn's confusion_matrix() and classification_report() functions respectively. The confusion matrices followed a general template

as laid out in Table 6. The classification reports, and subsequent performance metrics, for each algorithm were calculated using the confusion matrices and the performance formulas as seen in Table 7. These performance metrics were then used to compare the three tuned models. The methodology schema is laid out in Fig 2.

Table 6. Two Class 2 x 2 Confusion Matrix Explanation

		Predicted Label	
		Cancer (1)	No Cancer (0)
True Label	Cancer (1)	True Positive (TP)	False Negative (FN)
	No Cancer (0)	False Positive (FP)	True Negative (TN)

Table 7. Performance Metrics Formulas

Accuracy (CA)	**Precision**	**Recall**	**F1-Score**
$\frac{TP+TN}{TP+FP+TN+FN}$	$\frac{TP}{TP+FP}$	$\frac{TP}{TP+FN}$	$\frac{TP}{TP+\frac{1}{2}(FP+FN)}$

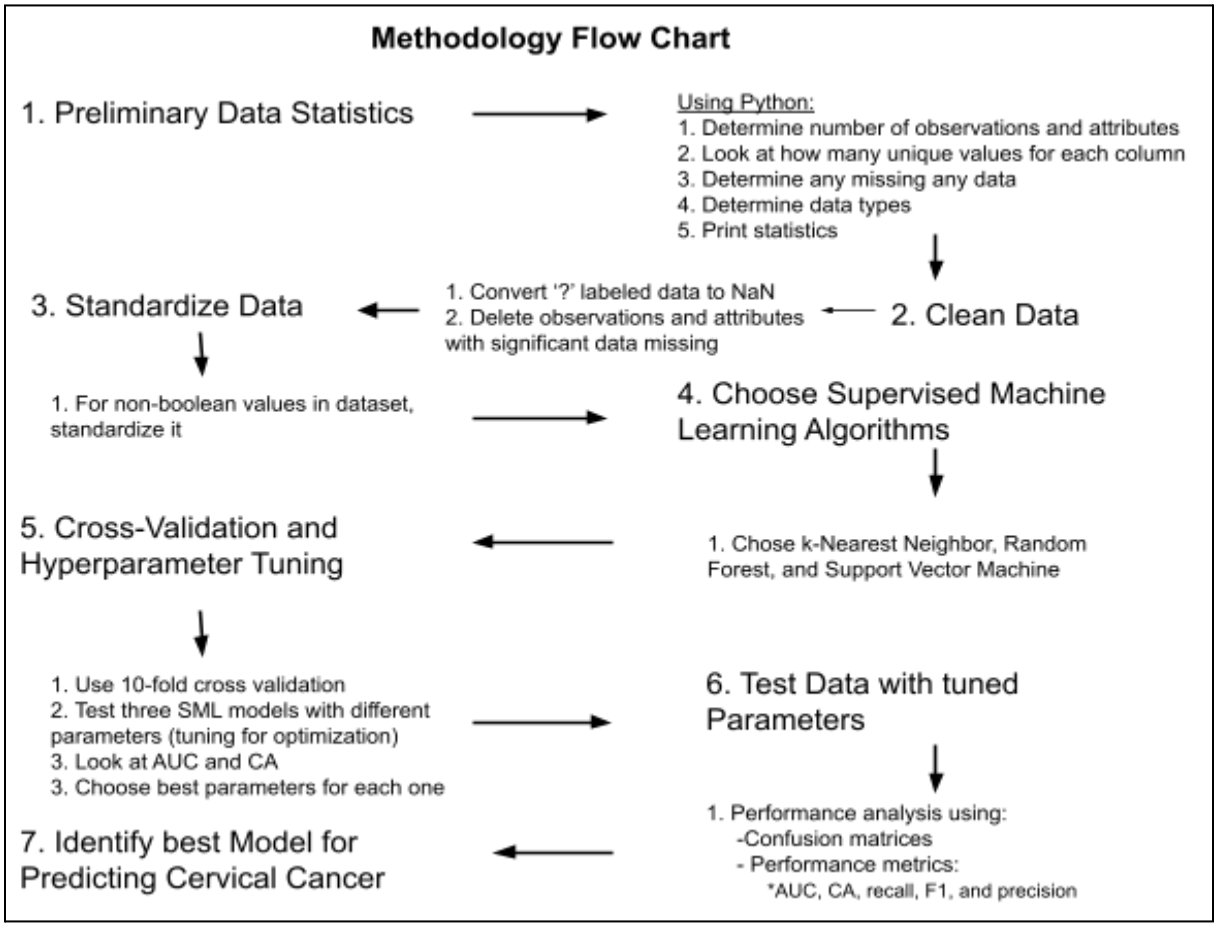

Fig 2: Methodology Schema Layout

IV. RESULTS AND DISCUSSION

The hyperparameters selected and subsequent selected models with tuned hyperparameters after the 10-fold cross-validation, for the KNN, RF, and SVM algorithms are summarized in Table 3, Table 4, and Table 5 respectively, found in Section III Methodology. For the sake of further analysis and review, the tuned hyperparameters used for KNN were k=12 and Manhattan distance calculation. For RF, the tuned hyperparameters were n=10 for minimum number of trees, n=10 for minimum number of samples required to split an internal node, and n=5 for minimum number of samples required to be a leaf node were selected. For SVM, the tuned hyperparameters were C=1.0 for the regularization parameter and the 'Linear' for kernel.

The confusion matrices constructed followed the template matrix laid out in the previously mentioned Table 6. The individual confusion matrices for the selected models with tuned hyperparameters can be observed in Fig 3 in Appendix A. From these confusion matrices, a classification report for each class label (0 and 1) was constructed using the aforementioned Table 7 performance formulas, and can be seen in Table 8 and Table 9.

Table 8. Classification Report (Target 0 - No Cervical Cancer)

SML Model	**Accuracy**	**F1**	**Precision**	**Recall**
KNN	0.979	0.990	0.979	1.000
RF	0.993	0.993	0.986	1.000
SVM	**0.993**	**0.996**	**0.993**	**1.000**

Table 9. Classification Report (Target 1 - Cervical Cancer)

SML Model	**CA**	**F1**	**Precision**	**Recall**
KNN	0.979	0.000	0.000	0.000
RF	0.993	0.500	1.000	0.333
SVM	**0.993**	**0.800**	**1.000**	**0.667**

The tuned KNN model was the worst performing model at predicting if a patient did not have cervical cancer (Target 0), and unsurprisingly was the worst model at predicting if the patient had cancer (Target 1), as seen in Table 9 under metrics F1, Precision, and Recall, as well. This could be due to the small number of target labels who had cervical cancer in the dataset. Only 19 patients were diagnosed with cervical cancer out of 726 total patients. Using a 80/20 testing/training split, this would provide the model with very few positive class labels to work with, putting extra emphasis on the models ability to classify true positives, especially if it gets one wrong.

On the other hand, the simplicity of the KNN algorithm and the fact that it struggles to calculate distances in high dimensions could also explain why it ran so poorly. The way the KNN algorithm classifies new data points is based on the distance between the new data point and the nearest training data in the variables space that is already classified. Based on the tuned hyperparameters, such as the k-value, the KNN algorithm then groups the new data point into the class label who had the highest number of bordering neighbors around said data point. All in all, The KNN model as a whole was not a suitable method for predicting cervical cancer on this dataset.

The tuned RF model was second to SVM in terms of performance metrics of classifying if a patient did or did not have cervical cancer, as observed in Table 8 and Table 9. RF as an algorithm is an improvement upon single decision trees by incorporating many decision trees, having each one make a prediction output, and the class label with the most votes from each tree then becomes the main prediction for the forest. This makes RFs very versatile on continuous and categorical data. This versatility is ultimately the 'double-edged sword' of RFs, and could quite possibly be the reason why it was second to SVM in this analysis. In other words, RF was too generalizable and may have overlooked key aspects of the dataset. Nevertheless, even though it was not the best method, the RF model still performed well in this case.

The tuned SVM model was best at predicting whether a patient had cervical cancer or not based on the performance metrics seen in Table 8 and Table 9. The SVM algorithm operates by utilizing and dividing the variable space with a hyplane based on the class label. Through maximizing the margins between the hyperplane and the next closest data point from either class label, the SVM algorithm ensures a greater probability of correctly classifying new data points. This approach proved to be very beneficial in a binary class label situation, such as the one seen in this analysis.

IV. CONCLUSION

As stated in the introduction, the purpose of this assignment was to select a dataset capable of classification, apply preprocessing to said dataset, use cross-validation with at least three SML algorithms to tune hyperparameters, and finally compare performance metrics, such as accuracy, precision, recall, and F1-score, between the three selected tuned models. After computing performance metrics, visuals of the results were created. The statistical and visual results will aid in ultimately distinguishing which model was the top performer.

Having successfully tuned the chosen SML models (i.e KNN, RF, and SVM) to the best of the group members abilities, the group is confident that they selected SVM as the best approach to addressing the issue of predicting if patients did or did not have a cervical cancer diagnosis. On the other hand, the group feels they provided ample reasoning as to why KNN and RF performed worse than SVM, and why they may not have been suitable to use on this dataset.

In terms of real-world analysis, a model like this could aid an organization or hospital in selecting and contacting patients that would have the highest chance of having cervical cancer, and get them in for a screening, or closely follow women who have a very high risk of developing it in the near future. This model would not be as useful for women who have already been diagnosed (although that data could be used to update our model regularly). For preventative care, this would be priceless in saving possibly thousands of women's lives each year, by catching cervical cancer early. However, it is important to note that the algorithms used in this analysis had an extensive list of hyperparameters, only a few 'high-impact' hyperparameters were used. That is to say that if these models were to be used in a medical setting, there would have to be efforts made in using many more hyperparameters and effectively tuning each one as well.

REFERENCES

[1] Cervical cancer (Risk Factors) Data Set, UCI Machine Learning Repository, March 2017. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/Cervical+cancer+%28Risk+Factors%29

[2] K. Fernandes, J. Cardoso, and J. Fernandes. 'Transfer Learning with Partial Observability Applied to Cervical Cancer Screening.' *Iberian Conference on Pattern Recognition and Image Analysis*, 2017 [Online] http://www.inescporto.pt/~jsc/publications/conferences/2017KelwinIBPRIA.pdf [Accessed October 10, 2022]

[3] E. Fix, J. Hodges, 'Discriminatory analysis-nonparametric discrimination: consistency properties', *University of California, Berkely,* 1951[Online] https://apps.dtic.mil/sti/pdfs/ADA800276.pdf [Accessed October 10, 2022]

[4] T.K. Ho, 'Random Decision Forest, '*Proceedings of the 3rd International Conference on Document Analysis and Recognition, Montreal*, 14-16 August 1995 [Online]. Available: https://ieeexplore.ieee.org/document/598994

[5] C. Cortes, V. Vapnik, 'Support-Vector Networks,' *AT&T Bell Labs,* 1995 [Online] http://image.diku.dk/imagecanon/material/cortes_vapnik95.pdf [Accessed October 10, 2022]

[6] S.C. Larson, 'The shrinkage of the coefficient of multiple correlation'. *Journal of Educational Psychology, 22*(1), 45–55, 1931[Online]. Available: https://doi.org/10.1037/h0072400

APPENDIX A

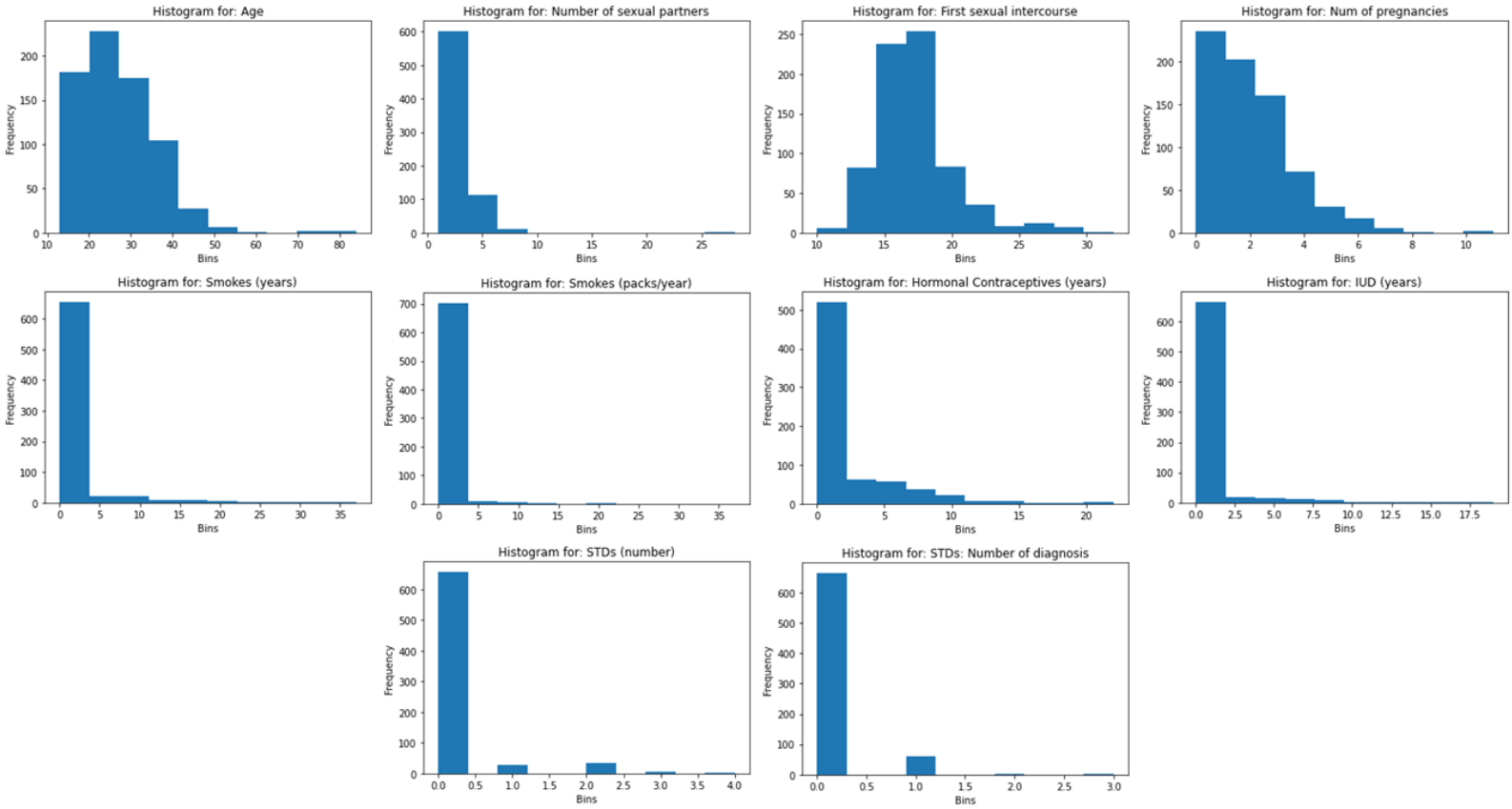

Fig 1: Histograms of Numerical Attributes

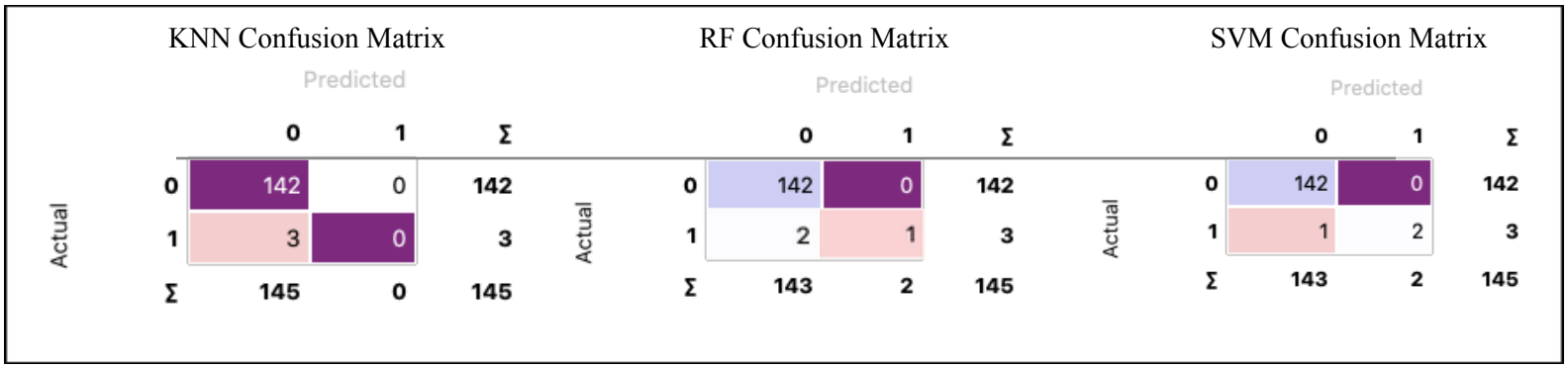

Fig 3: Confusion Matrices for SML Models