Cervical Cancer Risk Factors

Aprenentatge Automàtic I

GCED

UNIVERSITAT POLITÈCNICA DE CATALUNYA

Ramon Ventura Navarro - 21785256R

Contents

1 Introduction

Cervical cancer [1] is a serious worldwide health issue that affects many women's well-being, potentially resulting in loss of life. The appearance of cervical cancer can be influenced by various external factors, such as age and medical history. Exploring the relationship between these factors and cervical cancer can provide valuable insights for early detection, prevention, and intervention.

For this reason, this study aims to explore the relationship between cervical cancer and demographic, behavioral, and clinical factors [Soc] of a group of individuals using different methods, in order to predict the likelihood of someone having cancer, or better said, testing positive on the diagnostic tests, based on these factors.

The main objectives of this study are:

- Identify the most important external factors associated with cervical cancer.
- Create accurate models able to classify samples in testing positive or not on cervical cancer.
- Compare different models to see which one performs best in predicting cervical cancer.

By achieving these objectives, we hope to contribute to expanding the knowledge about the risk factors for cervical cancer and provide useful information for preventing cancer.

1.1 Dataset information

To conduct this study, a dataset [Rep] with features covering demographic information, habits, and historic medical records of 858 patients will be used. It was collected at the Hospital Universitario de Caracas in Caracas, Venezuela. Several patients decided not to answer some of the questions because of privacy concerns which will be represented as missing values.

There's a total amount of 858 instances and 36 attributes (both numerical and categorical). Target variables are `Hinselmann`, `Schiller`, `Citology` and `Biopsy`, which are all diagnostic procedures used in the evaluation and detection of cervical cancer.

Dataset characteristics:	Multivariate	**Number of Instances:**	858	**Area:**	Life
Attribute Characteristics:	Integer, Real	**Number of Attributes:**	36	**Data Donated:**	2017-03-03
Associated Tasks:	Classification	**Missing Values?**	Yes	**Number of Web Hits:**	196578

Source: Kelwin Fernandes (kafc_at_inesctec_dot_pt) - INESC TEC & FEUP, Porto, Portugal. Jaime S. Cardoso - INESC TEC & FEUP, Porto, Portugal. Jessica Fernandes - Universidad Central de Venezuela, Caracas, Venezuela

[1] This topic was chosen jointly with my partner (medicine student) when the project was presented. For this reason, it is possible that conclusions about risk factors may not be the most accurate or extensive.

2 Data Exploration Process

Before proceeding with the pre-processing of the dataset, the first step will be performing a basic inspection and description with the purpose of knowing the data we will be treating.

2.1 Basic inspection and description of the dataset

After inspecting the variables we will be working with, we see that we count with both numerical and categorical variables. However, all of the categorical variables have two levels and act as binary variables. Additionally, their format is not the adequate as they are float numbers:

- `Smokes`
- `Hormonal Contraceptives`
- `IUD`
- `STDs`
- `STDs:condylomatosis`
- `STDs:cervical condylomatosis`
- `STDs:vaginal condylomatosis`
- `STDs:vulvo-perineal condylomatosis`
- `STDs:syphilis`
- `STDs:pelvic inflammatory-disease`
- `STDs:genital herpes`
- `STDs:molluscum contagiosum`
- `STDs:AIDS`
- `STDs:HIV`
- `STDs:Hepatitis B`
- `STDs:HPV`
- `Dx:Cancer`
- `Dx:CIN`
- `Dx:HPV`
- `Dx`
- `Hinselmann`(target)
- `Schiller`(target)
- `Cytology`(target)
- `Biopsy`(target)

As it has been previously said, we find 4 target variables: `Hinselmann`, `Schiller`, `Citology` and `Biopsy`, located in the last 4 columns of the dataset. These are 4 different diagnostic tests that have two possible values (binary variables, positive/negative), therefore this is a classification problem. Originally, the idea of this project was to study the 4 target variable cases separately but, ultimately, it has been decided to study a single case. Therefore, these variables will be deleted from the dataframe, and we will create a new variable named `Cancer` that will exclusively focus on whether the patient has tested positive for cancer or not based on the information gathered from the 4 variables.

Note that our new target variable `Cancer` specifically indicates whether a patient has been diagnosed with cancer or not **through testing positive on any of the diagnostic procedures**. The variables `Dx:Cancer` and `Dx` are clinically different and they are not the target on this dataset. The first provides the diagnostic for cancer in general but may not directly record its actual presence, while the second respectively records the same for cervical cancer specifically. To make it clearer, `Dx:Cancer` tells if the patient has any type of cancer and `Dx` tells if the patient has cervical cancer, however, due to the way data is registered, this is not a necessary condition. A patient could still have cervical cancer and it not being registered in these two columns but instead on `Hinselmann`, `Schiller`, `Citology` or `Biopsy`. That's why throughout this study, we will refer to having cervical cancer as to testing positive on the diagnostic tests, or what is the same, having a positive value on the new `Cancer`.

Finally, we can also see in a quick exploration that we have lots of missing values. To address this issue and guarantee a correct format of our data before modeling, a pre-processing is needed.

2.2 Pre-processing

The steps we will be following are: (1) dealing with missing values, (2) finding outliers, (3) treatment of mixed data types, (4) normalization and (5) ending the pre-processing.

1. Dealing with missing values: The easiest way to deal with missing values would be deleting the involved rows or columns. However, in our case, the amount of missing values is significantly big and this would cause the loss of lots of relevant data. If we did so, our dataset would reduce its instances from 858 to 59. After taking a closer look we conclude the responsible of this are specifically 2 variables, which are NaN for almost every observation. Therefore, the best approach might be deleting each instance containing missing values but without taking into account `STDs:Time since first diagnosis` and `STDs:Time since last diagnosis`, which will be directly removed from the dataset. This decision has been taken without the supervision of an expert on the subject, nevertheless, we consider they are not relevant in the study and `STDs` already exists, which would still capture some of their information.

   Once these two columns are removed from the dataset we can see how now we have no missing values and maintain a total of 668 instances and 34 features, which is not such a big loss of values.

2. Finding outliers: In order to find outliers, we have plotted the histogram and box-plot of every numerical variable. After doing so we conclude that values that fall really far from the box-plot in `Age`, `Number of sexual partners` and `Smokes(packs/year)` will have to be deleted. Additionally, the rest of variables excepting `Age, First sexual intercourse` and `Num of pregnancies` present trouble in interpretation and visualization of the plots, as lots of values are 0. This occurs due to the fact that people who don't smoke, haven't had an IUD and do not suffer from an STD are the vast majority.

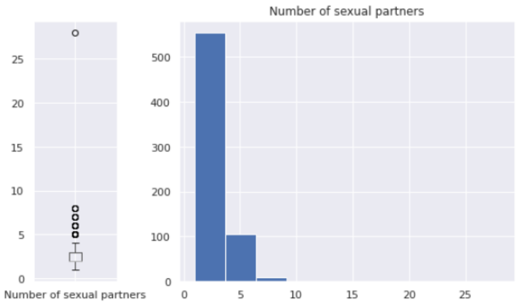

Figure 1: Histogram and box-plot of `Number of sexual partners`.

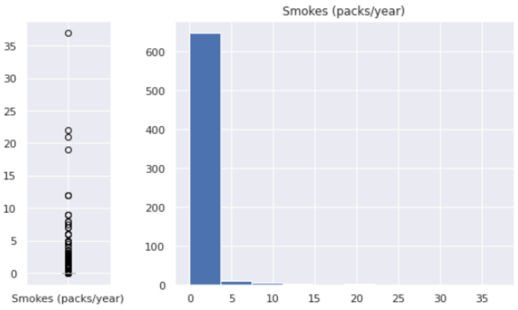

Figure 2: Histogram and box-plot of `Smokes (packs/year)`.

Observations mentioned (total of 6) are deleted, and so are variables whose values are mostly equal to 0. A possible solution besides deletion would be creating new discrete features considering values = 0 and > 0 as two different levels. However, `Smokes`, `Hormonal Contraceptives`, `IUD`, and `STDs` already exist and capture the presence or absence of any risk factor, so there's no point on doing so.

Regarding categorical attributes, we haven't found any proof that there exist any outliers as they only take binary values and there doesn't seem to be any incoherence.

3. Treatment of mixed data types: All of the features are in data types `int64` and `float64` regardless them being numerical or categorical. In order to assure they have the appropriate format, categorical variables have been encoded into 0s and 1s (integers) which will ease the modeling phase. After checking that all variables where in their correct data type, columns `STDs:cervical condylomatosis` and `STDs:AIDS` stand out to us as they only take one value (0). Therefore, we conclude these two variables take no effect on the study we are realizing and need to be removed. This step leaves us with a dataframe consisting of 662 instances and 22 variables to work with.

4. Normalization: In order to not have variables of ranges too far away, data has been normalized. It's worth mentioning that although all variables seem to be discrete, not all of them truly are. Before normalizing, as we have seen, some of these variables (like `Num of pregnancies`) have values equal to 0, which can't be deleted to help normalize. For that reason, we wouldn't be able to simply apply a Box-Cox transformation. However, we will add a small non-zero value to avoid the logarithm of 0. This measure is a common approach when trying to apply Box-Cox in this kind of situation. After that we will scale out data through Min-Max scaling, which will send values to range $[0, 1]$.

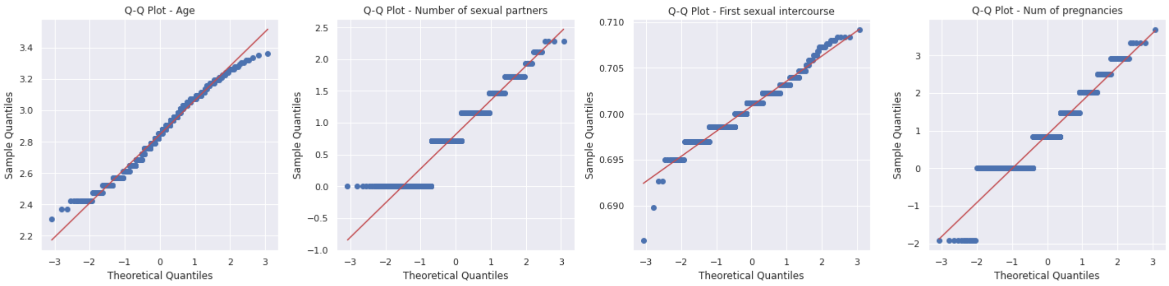

Figure 3: Numerical variables' Q-Q plots.

By looking at the Q-Q plots, the histogram with the overlapped Gaussian and the kernel density estimate (KDE), we can see that now our data resembles a normal distribution.

5. Ending the pre-processing: Before ending the pre-processing, it is noticeable that as this dataset is related to disease risks there's a high imbalance in our data. There's a significantly little amount of people that test positive on any test, less than a 15%. This is reflected in our data, where some variables don't seem to explain much because few people with cancer also suffer from other diseases. In this last step of the pre-processing we will focus on finding which categorical variables don't have enough data for them to be considered relevant. That is, which features do not have enough samples of patients with and without cancer for both cases (positive/negative in that disease).

After checking so, 6 STDs appear to not contribute any information to the prediction of cancer and, therefore, can be considered as candidates for deletion. Even though, again, the decision of deleting parameters should be consulted with an expert, we will proceed removing them as they are clearly statistically insufficient. Additionally, note that for example variable `STDs` is still relevant and we are definitely not concluding that STDs have no impact on the prediction of cervical cancer.

After ending the pre-processing, we are left with a dataframe containing 662 samples and a total of 16 numerical and categorical attributes. On the next section we will cover the visualization of this data and we will extract some conclusions based on it.

2.3 Data visualization

Before beginning with the modeling phase, it seems reasonable to visualize the relationships between variables. The reason of doing so is to try to gather information about their importance and influence.

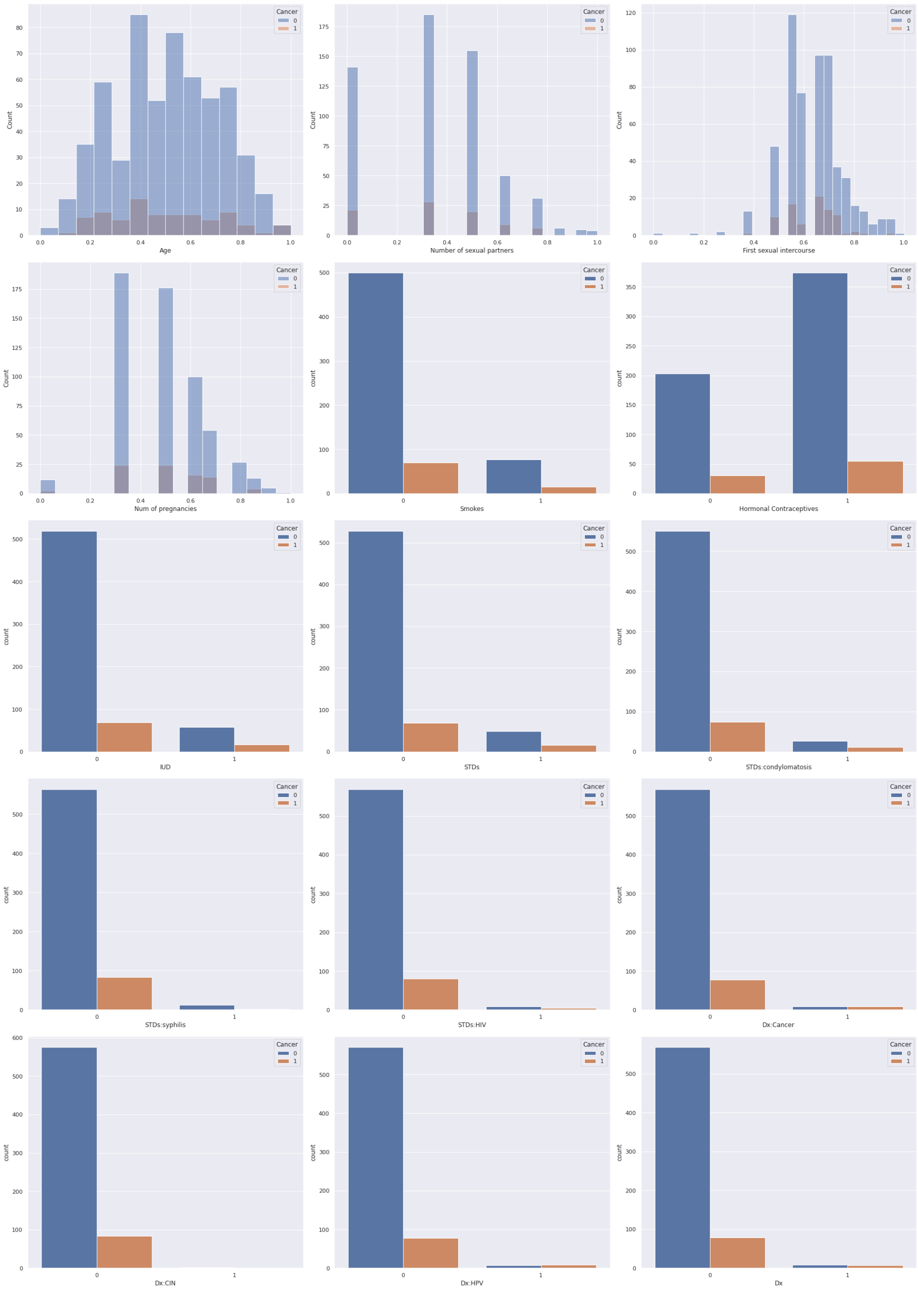

Figure 4: Histograms filtered by `Cancer`

After taking a look at Figure 4, we can see the data imbalance we were talking about previously. Some of the categorical variables don't seem to have lots of samples and this makes us think that prediction won't be easy. Even though variables as for example `Dx:CIN` seem to not have samples when they're positive, they actually do, although they are really small amounts. For that reason, these variables can't be deleted but are probably going to be insufficient when trying to make good predictions.

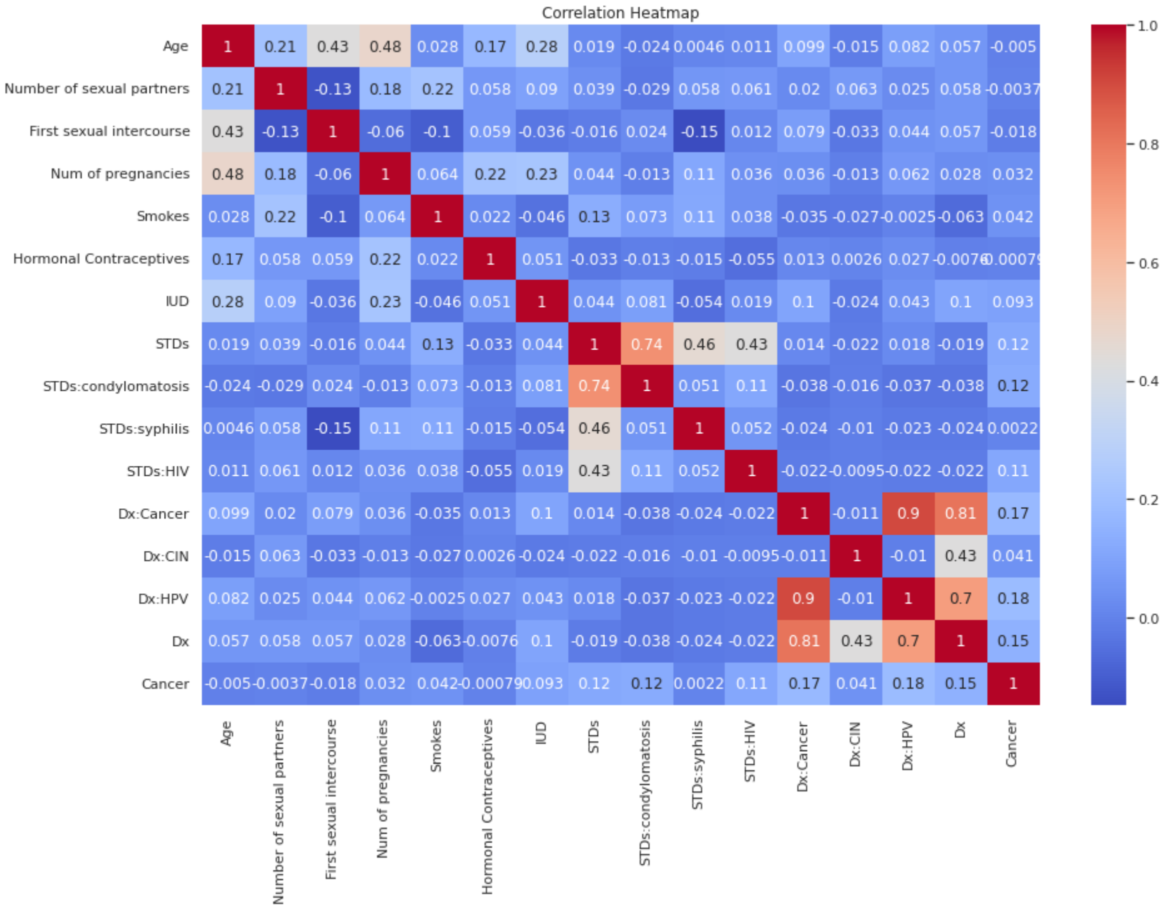

Figure 5: Correlation Matrix

We know the presence of strong correlations between the target variable and other predictor variables can be beneficial for prediction. When checking the correlation matrix in Figure 5, we see that's not the case in our study. However, it's important to note that correlation alone does not guarantee good predictions as it doesn't take into account interactions. Moreover, we can see that there exists a correlation between STDs, diagnosis of cancer, CIN (presence of abnormal cells), HPV (human papillomavirus) and of cervical cancer. As a reminder, note that Dx stands for diagnosis and that `Dx:Cancer`, `Dx` and `Cancer` are clinically different as it was mentioned during the Data Exploration Process section.

Finally, when taking a look at the pairplot [2] (on the numerical variables), we don't seem to reach any new conclusions or relevant information.

[2] The resultant pairplot is included in the appendix file along with this report.

3 Model Proposals

After familiarizing ourselves with the data, we are now prepared to develop a model that will try to predict the results on the diagnostic tests in patients for cervical cancer. The metrics evaluated for each model are going to be:

1. Recall for class 1: True positive rate
2. F1 score for class 1: Harmonic mean of precision and recall for class 1
3. Accuracy: Overall correctness considering all classes
4. F1 Macro: Average F1 score across all classes
5. Precision Macro: Average precision across all classes
6. Recall Macro: Average recall across all classes

Given the nature of this particular case, accurate classification of each observation becomes crucial. This is due to the fact that working with health risk factors requires extreme care to avoid misdiagnosing a patient as its life may be put at risk. Therefore, our main objective will be to maximize both recall and F1 score for class 1 in order to minimize false negatives. Accuracy is also an important factor, but as we will see in a few moments, in some scenarios a high accuracy does not imply a good classification due to data imbalance.

As this is a **classification problem**, we will only work with methods seen in class [UPC23] regarding Linear Classification as well as Decision Trees and Random Forests. Specifically the linear classification models considered are: LDA, QDA, *k*-NN, Gaussian Naive Bayes (and variations) and Logistic Regression.

Additionally, feature selection techniques such as Lasso Regression and results obtained from Decision Trees and Random Forests will be used. Therefore, in this study, three distinct dataframes will be analyzed, each containing a different number of variables.

The first dataframe, referred to as `DF1` in the code, will consist of the resulting dataframe after the pre-processing. The second dataframe, `DF2`, will correspond to the dataframe obtained after a first feature selection using Lasso Regression on `DF1`. Lastly, the last dataframe, `DF3`, will represent the produced dataframe after performing feature selection through Decision Trees and Random Forests on `DF2`. However, the same methods and metrics will be studied on all three dataframes to ensure consistent and comparable results. This comparison will aim to evaluate the effectiveness of the feature selection techniques in improving the prediction metrics.

In the following sections, we will thoroughly describe the size of the respective dataframes, the optimal parameters for each of the mentioned models, and lastly the results, metrics and conclusions obtained.

Nevertheless, the resampling protocols followed for each of the three scenarios are the same. Data will be split in 3 sets: train, validation and test. This protocol was chosen because we count with an amount of samples large enough. As we are dealing with a highly imbalanced dataset, it is generally better to allocate a larger portion to the training set compared to the validation and test sets. This should allow the model to learn from the minority class and improve the understanding of its patterns and characteristics. Even though it's important to have samples from the minority class in both the validation and test sets, reducing the training set too much could negatively affect the model's ability to learn from the imbalanced data.

For that reason, we will maintain a relatively larger training set to provide sufficient exposure to the minority class. Each of the partitions has been stratified in order to ensure a fair evaluation of the model's generalization across all classes. This setting will keep the same proportion of positive and negatives cases of cancer in each partition. If we were unlucky, we could have the case of not having any positive sample in a partition so this feature is essential to avoid that.

First of all, data has been split in 80%-20% for training and test sets, then the resultant 25% of the training partition has been assigned to the validation set. The size of the splits are approximately:

1. Train: 60%
2. Validation: 20%
3. Test: 20%

Moreover, to address the issue concerning data imbalance, the function `RandomUnderSampler` from `imbalanced-learn` package has been used in order to reduce the amount of samples belonging to the majority class. After trying different sampling strategies, a reduction to the 70% of the original size of the majority class seems to be the optimal. Reducing it by a larger proportion turns out to be an aggressive approach as it results a in significant loss of information due to the reduced amount of representative samples for the minority class.

This way we end up with a more balanced data, nonetheless, in the following sections we will discover whether or not after applying these measures has a positive impact on predicting the presence of cervical cancer on women. It's also worth noting that the introduction of a random process may affect by a little the results obtained on each computation of the methods.

3.1 Results obtained for DF1

Before finding out how do models perform, it's important to clarify that the mentioned metrics have been computed using the validation set. Validation sets are used during model development to assess the performance and select the best model. They help to prevent overfitting and provide an estimate of the model's performance on unseen data. On the other hand, testing sets are used as an independent evaluation of the final chosen model's generalization capabilities. These provide an unbiased estimate of how the model will perform on new and unseen data. For this reason, once we chose our best model based on the metrics on the validation set, we will compute them again for the test set.

Let's also provide a brief explanation on what does every model do:

1. Linear Discriminant Analysis (LDA): LDA is a statistical method that aims to find a linear combination of features to discriminate between classes. It assumes that the data follows a Gaussian distribution and models the class-conditional densities to make predictions.
2. Quadratic Discriminant Analysis (QDA): QDA is similar to LDA, but it relaxes the assumption of equal covariance matrices across classes. It allows for more flexible decision boundaries by considering class-specific covariances.
3. k-Nearest Neighbors (k-NN): k-NN is a non-parametric algorithm that classifies new instances based on the majority class among its k nearest neighbors in the feature space. It does not make any assumptions about the underlying data distribution.
4. Gaussian Naive Bayes: Gaussian Naive Bayes is a probabilistic algorithm that assumes the features are conditionally independent given the class labels. It models the class-conditional densities using Gaussian distributions and applies Bayes' theorem for classification. We will study different variations of it.
5. Logistic Regression: Logistic Regression models the relationship between the features and the binary outcome using the logistic function. It estimates the probabilities of class membership and applies a decision threshold for classification. As our target variable is a binary variable, it seems appropriate to apply this model.
6. Decision Trees and Random Forests: Decision Trees create a hierarchical structure of binary decisions based on features to classify instances. Random Forests combine multiple decision trees and aggregate their predictions to improve performance and reduce overfitting. We will also study different variations of them.

This first dataframe (`DF1`)is the resulting after the pre-processing and it consists of 662 samples and 16 features. All of the methods have been tested on two train-validation-test partitions, with and without undersampling. Results on the undersampling case are better for almost every model, so from now on we will only consider this scenario. Next, let's discuss the chosen parameters for each model and check the metrics.

We've found the following metrics and optimal parameters:

1. QDA: reg = 0.01
2. *k*-NN: k = 5
3. Logistic Regression: C = 10.0
4. Decision Tree - Best: criterion = entropy, max_depth = None, max_features = None, min_samples_leaf = 4, min_samples_split = 3
5. Random Forests - Best: class_weight = balanced_subsample, max_depth = None, min_samples_leaf = 6, min_samples_split = 4, n_estimators = 200
6. Extra Trees - Best: class_weight = balanced, max_depth = 100, min_samples_leaf = 4, min_samples_split = 4, n_estimators = 150

Model	**Recall class 1**	**F1 class 1**	**Accuracy**	**F1 Macro**	**Precision Macro**	**Recall Macro**
KNN-5	0.706	0.316	0.609	0.521	0.568	0.65
extra_trees-best	0.471	0.39	0.812	0.64	0.625	0.666
QDA-0.01	0.353	0.324	0.812	0.608	0.601	0.616
Combined-NB-tuned	0.353	0.364	0.842	0.637	0.64	0.633
DT-default	0.353	0.387	0.857	0.653	0.668	0.642
LDA	0.235	0.211	0.774	0.539	0.537	0.544
Gaussian-NB-only-categorical	0.176	0.24	0.857	0.581	0.631	0.567
DT-best	0.118	0.154	0.835	0.531	0.551	0.529
RF-default	0.118	0.19	0.872	0.561	0.692	0.55
RF-balance	0.118	0.19	0.872	0.561	0.692	0.55
RF-best	0.118	0.118	0.774	0.494	0.494	0.494
extra_trees	0.118	0.167	0.85	0.542	0.583	0.537
Gaussian-NB	0.059	0.1	0.865	0.513	0.605	0.521
Combined-NB	0.059	0.105	0.872	0.518	0.689	0.525
LogReg-10.0	0.059	0.111	0.88	0.523	0.939	0.529
Gaussian-NB-only-numerical	0.0	0.0	0.872	0.466	0.436	0.5

Table 1: Validation Metrics for DF1

As we can see, in this first case, KNN-5 seems to be the better model. The performance metrics obtained are concerning and might indicate there are issues with the classification models. The most critical metric in this context is the recall for class 1, which measures the ability of the models to correctly identify instances of cancer. A recall value of 0.706 for KNN-5 suggests that approximately 30% of cancer cases are being misclassified. Similarly, the F1 score for class 1 is only 0.316, reflecting a low balance between precision and recall for cancer instances. The overall accuracy of 0.609 indicates that the models are making correct predictions for only a little over half of the cases, which is far from desirable in a cancer-related problem. In this kind of context, it's vital to prioritize high recall values to minimize the risk of false negatives and ensure accurate detection.

We will continue proposing new models on DF2 after performing a feature selection, in order to try to improve this results. Our intuition tells us there might be a problem either with the models or the data.

predicted	**0**	**1**
target		
0	69	47
1	5	12

Table 2: Confusion Matrix for KNN-5 on DF1

3.2 Results obtained for DF2

In this next section we will reveal whether feature selection through Lasso Regression improves our models or not. After performing Lasso with alpha = 0.001, we conclude that variables `Age, Number of sexual partners, First sexual intercourse, Smokes, Hormonal Contraceptives, IUD, STDs:condylomatosis, STDs:HIV, Dx:Cancer, Dx:CIN, Dx:HPV` and `Cancer` are going to be the variables selected. Therefore, out new dataframe consists of 662 observations and 12 attributes.

We've found this time the following metrics and optimal parameters:

1. QDA: reg = 0.01
2. *k*-NN: k = 5
3. Logistic Regression: C = 10.0
4. Decision Tree - Best: criterion = gini, max_depth = 20, max_features = auto, min_samples_leaf = 1, min_samples_split = 1
5. Random Forests - Best: class_weight = balanced_subsample, max_depth = 100, min_samples_leaf = 6, min_samples_split = 4, n_estimators = 200
6. Extra Trees - Best: class_weight = balanced_subsample, max_depth = None, min_samples_leaf = 2, min_samples_split = 6, n_estimators = 150

Model	Recall class 1	F1 class 1	Accuracy	F1 Macro	Precision Macro	Recall Macro
KNN-5	0.706	0.3	0.579	0.499	0.56	0.633
LDA	0.353	0.316	0.805	0.601	0.594	0.612
QDA-0.01	0.353	0.375	0.85	0.645	0.653	0.638
DT-best	0.235	0.242	0.812	0.568	0.569	0.566
Combined-NB-tuned	0.176	0.24	0.857	0.581	0.631	0.567
DT-default	0.176	0.171	0.782	0.523	0.522	0.524
RF-best	0.176	0.158	0.759	0.509	0.509	0.511
extra_trees-best	0.176	0.171	0.782	0.523	0.522	0.524
RF-default	0.118	0.174	0.857	0.548	0.608	0.542
RF-balance	0.118	0.174	0.857	0.548	0.608	0.542
extra_trees	0.118	0.148	0.827	0.526	0.539	0.524
Gaussian-NB	0.059	0.1	0.865	0.513	0.605	0.521
Gaussian-NB-only-categorical	0.059	0.1	0.865	0.513	0.605	0.521
Combined-NB	0.059	0.111	0.88	0.523	0.939	0.529
LogReg-10.0	0.059	0.105	0.872	0.518	0.689	0.525
Gaussian-NB-only-numerical	0.0	0.0	0.872	0.466	0.436	0.5

Table 3: Validation Metrics for DF2

In the new dataframe it is evident that the best model, based on the provided metrics, is again the KNN-5 classifier. Although the overall accuracy of the model is relatively lower at 0.579, it still holds the same performance in terms of recall for class 1 with a value of 0.706. The rest of the metrics are pretty similar than previously which make us reach the same conclusions. The KNN-5 model shows potential and could be further optimized to improve its overall performance, otherwise, we would still consider it insufficient. Besides from these conclusions, we can also state that after removing four variables through Lasso regression, our best model is quite the same. This means the deleted features did not have much relevance. Similarly, in this next step, we will try to perform another feature selection using Decision Trees and Random Forests in order to remove more irrelevant factors.

predicted	**0**	**1**
target		
0	65	51
1	5	12

Table 4: Confusion Matrix for KNN-5 on DF2

3.3 Results obtained for DF3

Finally, we will proceed to compute the same metrics for the same methods on a new dataframe DF3 containing `Age, First sexual intercourse, Number of sexual partners, Hormonal Contraceptives, STDs:condylomatosis, IUD` and `Smokes`, which seem to be the most relevant variables after this new feature extraction. Our last dataframe consists of 662 observations and 8 attributes.

We've found this last time the following metrics and optimal parameters:

1. QDA: reg = 0.01
2. *k*-NN: k = 5
3. Logistic Regression: C = 10.0
4. Decision Tree - Best: criterion = gini, max_depth = 15, max_features = auto, min_samples_leaf = 1, min_samples_split = 1
5. Random Forests - Best: class_weight = balanced_subsample, max_depth = 100, min_samples_leaf = 6, min_samples_split = 6, n_estimators = 200
6. Extra Trees - Best: class_weight = balanced, max_depth = None, min_samples_leaf = 2, min_samples_split = 6, n_estimators = 150

Model	**Recall class 1**	**F1 class 1**	**Accuracy**	**F1 Macro**	**Precision Macro**	**Recall Macro**
KNN-5	0.706	0.289	0.556	0.483	0.554	0.62
QDA-0.01	0.471	0.34	0.767	0.599	0.59	0.64
Gaussian-NB	0.412	0.35	0.805	0.617	0.607	0.637
DT-default	0.294	0.244	0.767	0.553	0.549	0.565
LDA	0.176	0.188	0.805	0.538	0.541	0.537
Combined-NB-tuned	0.118	0.182	0.865	0.554	0.641	0.546
RF-best	0.118	0.125	0.789	0.503	0.503	0.503
extra_trees	0.118	0.148	0.827	0.526	0.539	0.524
DT-best	0.059	0.087	0.842	0.5	0.52	0.508
RF-default	0.059	0.083	0.835	0.496	0.508	0.504
RF-balance	0.059	0.091	0.85	0.504	0.537	0.512
extra_trees-best	0.059	0.057	0.752	0.457	0.458	0.456
Gaussian-NB-only-numerical	0.0	0.0	0.872	0.466	0.436	0.5
Gaussian-NB-only-categorical	0.0	0.0	0.872	0.466	0.436	0.5
Combined-NB	0.0	0.0	0.872	0.466	0.436	0.5
LogReg-10.0	0.0	0.0	0.872	0.466	0.436	0.5

Table 5: Validation Metrics for DF3

In this last dataframe, it becomes apparent that the KNN-5 model once more emerges as the best performing model. We can see the metrics are quite similar than before, and specially recall of class 1 still remains the same. However, the deletion of 3 more variables affected a little on the accuracy. At this point of the project, we still are not satisfied with the proposed models so we have tried to perform a third feature extraction, as well as fitting a new model.

predicted	**0**	**1**
target		
0	63	54
1	5	12

Table 6: Confusion Matrix for KNN-5 on DF3

3.4 Extra: 3rd feature selection using DT and RF & usage of the XGBoost method

When trying to perform a third feature selection through Decision Trees and Random Forests, removing two more variables (`Smokes` and `IUD`), we are left with a dataframe with 7 variables. At this point metrics seem to perform even worse, even though Extra Trees improves considerably. So we reach the conclusion that no more features can be deleted as they are all somewhat relevant. Consequently, `Age, Number of sexual partners, First sexual intercourse, Smokes, Hormonal Contraceptives, IUD` and `STDs:condylomatosis` have all proven to be significantly relevant factors in the diagnosis of cervical cancer on women.

Model	**Recall class 1**	**F1 class 1**	**Accuracy**	**F1 Macro**	**Precision Macro**	**Recall Macro**
xgboost	0.353	0.169	0.556	0.433	0.486	0.47

Table 7: Validation Metrics for XGBoost on DF3

On Table 7, located above, we can see the metrics for a new extra method not seen in class that has been applied on DF3. This method specifically addresses the class imbalance. If we take a close look on the recall for class 1 and the accuracy we can see that we are not in the same situation as we generally are. Due to the imbalance, accuracy has been generally a very high value for most of the methods, while recall for class 1 has mostly taken low values. However, we can see that after fitting this model that tries to address this imbalance, we can see accuracy and recall for class 1 are not that polarized. Nevertheless, its predictions are far from good and do not improve the best ones found until now.

4 Model Selection and Evaluation

The final step is going to decide which is the best method found from all of the presented. In other words, we have to decide which is the best model based on its performance on the validation set. Then we will proceed to evaluate its performance on the independent test set.

It becomes really clear that KNN-5 is the best choice as it drastically outperforms every other model in all the presented scenarios. Remember that even though some other models have better metrics in general, recall for class 1 has to be maximized in this context. Also, we have seen that DF3 is the best of the three proposed dataframes, as we seem to have simplified its factors without losing information. That being said, let's evaluate the model on the test set of DF3 to provide a more unbiased assessment of the model's performance and generalization ability.

Below, the confusion matrices and prediction metrics for both the validation and test sets:

predicted **target**	**0**	**1**
0	62	54
1	5	12

(a) Confusion Matrix on the Validation Set

predicted **target**	**0**	**1**
0	74	42
1	9	8

(b) Confusion Matrix on the Test Set

Table 8: Confusion Matrices

Model	**Recall class 1**	**F1 class 1**	**Accuracy**	**F1 Macro**	**Precision Macro**	**Recall Macro**
KNN-5-val	0.706	0.289	0.556	0.483	0.554	0.62
KNN-5-test	0.471	0.239	0.617	0.491	0.526	0.554

Table 9: Metrics for validation and test sets

The KNN-5 model demonstrates different performance when evaluated on the validation and test sets. In terms of class 1 recall, we can see the model achieves a higher value in the validation set, indicating a lower performance in detecting cancer in unseen data. This metric has dropped to 0.471 in the new predictions. Additionally, the F1 score for class 1 still remains too low suggesting the model is sacrificing the detection of true positives for overall precision, which is a bit higher for the test set. Specifically the accuracy is 0.617, which tells us that the model makes correct predictions for more than 60% of the cases. The F1 macro, precision macro, and recall macro indicate that, in theory, the model's performance is somewhat balanced across all classes.

Even though we have an accuracy of 60%, we are again facing the problems of high imbalance in our data. We can see how, after trying to address this issue through multiple tools, we still don't have a good relation of recall for class 1 and accuracy. On the test set, we are just predicting the 47% of the true positive cases of cancer, unlike the 70% we predict on the validation set. This makes us consider our model as not sufficient for cervical cancer predictions, because as we have reiterated during this study, in cancer-related problems it's important to maximize true positives and minimize false negatives, which we are failing to do.

It's also worth noting that, when computing the metrics on the training partition for decision trees, we were able to see that data was predicted perfectly. This made us think model was overfitting, so we tried to optimize the hyperparamers. However, we still weren't able to make good predictions on the validation partitions. For this reason, we considered the possibility that, due to the fact of training data not being representative of the overall population, the model had adapted too much to it and because of that, we were obtaining those bad results on our validation set.

Moreover, when working with random forests where Out-Of-Bag error was introduced, even though we had both good OOB metrics and actual validation accuracy metrics, the detection of true positives wasn't maximized, so accuracy was won at the cost of recall of class 1 which is, again, not the desired outcome. Lastly, we tried to address the imbalance in our target variable by including balance weights but still got no decent results.

In addition, we also tried to apply some dimensionality reduction algorithms [3] to find out how our data behaved even though we didn't continue modeling using the features extracted from these techniques. By doing so we reduced the dimensionality and improved the correlations significantly, but we lost the relation with the variables and the target which is our main goal. For this reason, we didn't continue with this specific approach.

After all, we have tried different methods and tools to try to predict on such an imbalanced dataset. Nonetheless, even though the chosen model KNN-5 is the best we have found, we can't say we have found a good model as it's not robust and reliable. Furthermore, we also can't state we have properly addressed the data imbalance through the mentioned tools throughout this project.

[3] Heatmap comparison of dimensionality reduction can be found in the appendix file.

5 Conclusions

In conclusion, this project aimed to develop a predictive model for accurately identifying instances of cervical cancer. We firstly pre-processed the data and applied the necessary transformations to the target variable. Afterwards, we explored various machine learning algorithms and evaluated their performance using validation and test partitions, while trying to address the presented imbalance in our data. These algorithms went from linear classifiers to decision trees and random forests.

After careful analysis of the results, we have finally found that the KNN-5 model consistently outperformed other models but is far from being good, robust and reliable. This final model would need a further improvement to minimize false negatives and maximize true positives, as life of individuals might be put at risk. The model under consideration had an accuracy of 0.617, recall for class 1 of 0.471 and F1 score for class 1 of 0.239 on the test partition, corroborating our conclusions and implying poor generalization on unseen data.

We can also state that, after the pre-processing of the original data and several feature selection procedures, `Age, Number of sexual partners, First sexual intercourse, Smokes, Hormonal Contraceptives, IUD` and `STDs:condylomatosis` have seemed to be the most relevant factors when trying to predict this kind of cancer. However, this conclusion should not be interpreted as a fact, as we would need the help of an expert to really decide which aspects are important when determining the presence or absence of cervical cancer, and results might not have been accurate.

Finally, as a possible extension of this project, if the data could actually be properly balanced, it would be interesting to find a workaround to address this issue and truly find a strong association between the factors presented in the dataset and cervical cancer on women. This way we could help in the early detection, prevention and intervention of this disease.

6 Bibliography

Bibliography

[UPC23] UPC. *Notes from the AA1 course*. El Racó de la FIB, 2023.

[Rep] UCI Machine Learning Repository. *Cervical cancer (Risk Factors) Data Set*. URL: `https://archive.ics.uci.edu/ml/datasets/Cervical+cancer+%28Risk+Factors%29`. (March 3, 2017).

[Soc] American Cancer Society. *Risk Factors for Cervical Cancer*. URL: `https://www.cancer.org/cancer/types/cervical-cancer/causes-risks-prevention/risk-factors.html`. (January 3, 2020).