Machine Learning

A.Y 2019/2020

A Machine Learning application in Healthcare: Cervical Cancer Biopsia response classification

Cannistraci Irene

1603090

March 2, 2020

Contents

1 Introduction

Eleven years ago, cervical cancer ranked as the third most common cancer among women worldwide. However, in 42 low-resource countries, it was the most common cancer in women. There were over 500,000 new cases in 2018[1]. Usually all cervical cancers are associated with human papilloma viruses (HPV), however the majority of women with HPV do not develop cervical cancer. Women become susceptible to developing cervical cancer following HPV infection, but other environmental factors are required for the cancer to develop. This is the reason why cervical cancer is a significant cause of mortality in many low-income countries and the existence of several diagnosis methods allow to create automated methods that can help people to prevent cancer and to receive focused treatments. Unfortunately in developing countries resources are limited and patients usually do not do routine screening due to financial problem, so the prediction of the individual patient's risk and the best screening strategy during her diagnosis becomes a fundamental problem. The aim of this project is to detect cervical cancer based on the Biopsia response.

2 Preprocessing and Feature Engineering

"Torture the data, and it will confess to anything." — Ronald Coase

Feature engineering and preprocessing are important steps since machine learning models are as good or as bad as the data we have. That's the reason why during this phase the dataset is inspected in order to check if there were wrong data such as missing values, special characters and so on.. This is very important since wrong data may produce misleading results and we want to select only features that would contribute most to the quality of the model.

2.1 Dataset description

I chose the dataset (see Table 1) from the UCI Machine Learning Repository[2]. It focuses on the prediction of indicators/diagnosis of cervical cancer and it was collected at 'Hospital Universitario de Caracas' in Caracas, Venezuela. The dataset comprises demographic information, habits, and historic medical records of 858 patients. Several patients decided not to answer some of the questions because of privacy concerns. The dataset is composed by **858** instances and **36** attributes. Target variables are four: *Hinselmann*, *Schiller*, *Citology*, and *Biopsy*.

- **Hinselmann** 0: 823, 1: 35
  - Hans Hinselmann introduces colposcopy for early diagnosis of cervical cancer in the nineteen twenties, so I interpreted this feature as *Colposcopy*: a procedure in which a lighted, magnifying instrument called a colposcope is used to examine the cervix, vagina, and vulva.
- **Schiller** 0: 784, 1: 74
  - *Schiller's test* or Schiller's Iodine test is a medical test in which iodine solution is applied to the cervix in order to diagnose cervical cancer. The iodine colors healthy cells brown while abnormal cells remain unstained, usually appearing white or yellow.
- **Citology** 0: 814, 1: 44
  - *Cytology* is the exam of a single cell type, as often found in fluid specimens. It's mainly used to diagnose or screen for cancer.
- **Biopsy** 0: 803, 1: 55
  - *Biopsy* is the removal of cells or tissues for examination by a pathologist. The pathologist may study the tissue under a microscope or perform other tests on the cells or tissue. There are many different types of biopsy procedures.

I found two different papers that study and apply different models to the same dataset that I chose, so I used them to understand more about data and to compare my results with theirs. Both papers use **Biopsy** as target feature since it is the most recommended in literature, so I used Biopsy too.

Name	**Type**	**Description**	**Missing**
Age	int	Age	No
Number of sexual partners	int	Number of sexual partners	Yes
First sexual intercourse (age)	int	Age of first sexual intercourse	Yes
Number of pregnancies	int	Number of pregnancies	Yes
Smokes	bool	If she is a smoker or not	Yes
Smokes (years)	float	How long have she has been smoking	Yes
Smokes (packs/year)	float	How many packs per year	Yes
Hormonal Contraceptives	bool	If she takes hormonal contraceptives or not	Yes
Hormonal Contraceptives (years)	float	How many years she has been taking the contraceptives	Yes
IUD	bool	If she use Intra-Uterine devices or not	Yes
IUD (years)	float	How many years she has been taking the IUD	Yes
STDs	bool	If she had/has Sexual Transmitted Diseases	Yes
STDs (number)	int	How many STD she had/has	Yes
STDs:condylomatosis	bool	If she contracted condylomatosis or not	Yes
STDs:cervical condylomatosis	bool	If she contracted cervical condylomatosis or not	Yes
STDs:vaginal condylomatosis	bool	If she contracted vaginal condylomatosis or not	Yes
STDs:vulvo-perineal condylomatosis	bool	If she contracted vulvo-perineal condylomatosis or not	Yes
STDs:syphilis	bool	If she contracted syphilis or not	Yes
STDs:pelvic inflammatory disease	bool	If she contracted pelvic inflammatory disease or not	Yes
STDs:genital herpes	bool	If she contracted genital herpes or not	Yes
STDs:molluscum contagiosum	bool	If she contracted molluscum contagiosum or not	Yes
STDs:AIDS	bool	If she contracted AIDS or not	Yes
STDs:HIV	bool	If she contracted HIV or not	Yes
STDs:Hepatitis B	bool	If she contracted Hepatitis B or not	Yes
STDs:HPV	bool	If she contracted HPV or not	Yes
STDs: Number of diagnosis	int	Number of diagnosis	Yes
STDs: Time since first diagnosis	int	Time since first diagnosis	Yes
STDs: Time since last diagnosis	int	Time since last diagnosis	Yes
Dx	bool	Diagnosis[5]	No
Dx:Cancer	bool	Cancer diagnosed	No
Dx:CIN	bool	If abnormal cells are found on the surface of the cervix[6]	No
Dx:HPV	bool	Human Papilloma Virus diagnosed	No
Hinselmann	**bool**	**Colposcopy[7] response**	**No**
Schiller	**bool**	**Schiller's test[8] response**	**No**
Citology	**bool**	**Citology[9] response**	**No**
Biopsy	**bool**	**Biopsy[10] response**	**No**

Table 1: Dataset description.

2.2 Adjusting data

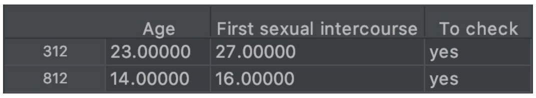
The hardest part of this project was to understand the dataset; there is no documentation and it is a medical dataset, so there are only medical names and most of them were unknown to me. The only dataset description was a list of features with the corresponding type that was according to my idea not totally correct. So the first thing that I did was to search on the internet the meaning of each feature and try to assign the correct type to each one.

For example, let's analyze the features about 'smoke':

- Smokes, type: bool
- Smokes (years), type: bool
- Smokes (packs/year), type: bool

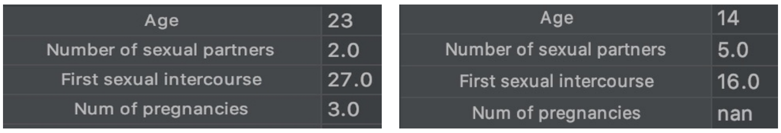
How can *smokes (years)* and *Smokes (packs/year)* be boolean? In my opinion there are both float, so I had to convert all data to the correct type.

Then I noticed other two things: the first was that missing values where indicated with a question mark, so I converted all the ’*?*’ with ’*None*’, the second one was that some data were completely wrong and it’s justifiable since data were collected by human. For example two patients had an Age that was lower than the age at which they had the first sexual intercourse (see Figure 1), so I added a new column *To check* in order to filter instances that contains wrong data. I isolated two specific case, the instances *312* and *812*. Unfortunately there isn’t a specific way to understand what the mistake is (see Figure 2), I think *Age* and *First sexual intercourse* were wrongly swapped, so I decide to swapped them again.



	Age	First sexual intercourse	To check
312	23.00000	27.00000	yes
812	14.00000	16.00000	yes

Figure 1: Wrong data.



Age	23
Number of sexual partners	2.0
First sexual intercourse	27.0
Num of pregnancies	3.0

Age	14
Number of sexual partners	5.0
First sexual intercourse	16.0
Num of pregnancies	nan

Figure 2: Instances 312 and 812.

2.3 Missing data

As I previously said this dataset suffers of missing data caused by privacy. Some patients decided to not respond at some questions, in particular (see Figure 3) 787 patiens over 858 (92%) decided to not provide data about how long they had been diagnosed with a sexually transmitted disease and when it was the last diagnosis (*STDs: Time since last diagnosis*, *STDs: Time since first diagnosis*), so I decided to drop these features.

Other attributes suffers of missing data, but since it is a low percentile I decided to proceed as follows.
First of all I decided to maintain in the dataset only instances with at least 25 non-null attributes over 34 (where 4 are the target variables), so if an instances has more than 5 null values, it is dropped from the dataset. In this way 105 instances were deleted, then I decided to impute missing values as follows:

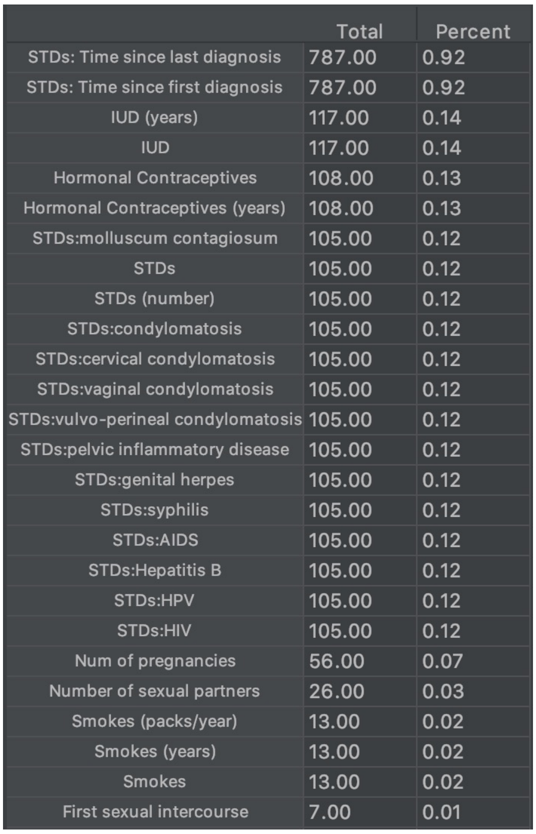
- For what concern *Smokes*, *Smokes (years)* and *Smokes (packs/year)* I substituted the None values of *Smokes* with its mode (0) and then I replaced None values of *Smokes (years)* and *Smokes (packs/year)* with the mean.

- For what concern *Hormonal Contraceptives* and *Hormonal Contraceptives (years)* I substituted the None values of *Hormonal Contraceptives* with its mode (1) and then I replaced None values of *Hormonal Contraceptives* with the mean.

- For what concern *IUD* and *IUD (years)* I substituted the None values of *IUD* with its mode (0) and then I replaced None values of *IUD (years)* with the mean.

- For what concern *Num of pregnancies*, *Number of sexual partners* and *First sexual intercourse* I substituted the None values with the mean and then I checked if substituted values were coherent (e.g First sexual intercourse can’t be higher than Age and Num of pregnancies can’t be a number higher than 0 if First sexual intercourse and Number of sexual partners is 0).

- For what concern all the features about *STDs ’Sexual Transmitted Diseases’* I substituted None values of boolean type with the mode and then integer values (*STDs (Number)* and *STDs (Number of diagnosis)*) looking at the result of other imputation (e.g if others are all 0, STDs (Number) must be 0 too).



	Total	Percent
STDs: Time since last diagnosis	787.00	0.92
STDs: Time since first diagnosis	787.00	0.92
IUD (years)	117.00	0.14
IUD	117.00	0.14
Hormonal Contraceptives	108.00	0.13
Hormonal Contraceptives (years)	108.00	0.13
STDs:molluscum contagiosum	105.00	0.12
STDs	105.00	0.12
STDs (number)	105.00	0.12
STDs:condylomatosis	105.00	0.12
STDs:cervical condylomatosis	105.00	0.12
STDs:vaginal condylomatosis	105.00	0.12
STDs:vulvo-perineal condylomatosis	105.00	0.12
STDs:pelvic inflammatory disease	105.00	0.12
STDs:genital herpes	105.00	0.12
STDs:syphilis	105.00	0.12
STDs:AIDS	105.00	0.12
STDs:Hepatitis B	105.00	0.12
STDs:HPV	105.00	0.12
STDs:HIV	105.00	0.12
Num of pregnancies	56.00	0.07
Number of sexual partners	26.00	0.03
Smokes (packs/year)	13.00	0.02
Smokes (years)	13.00	0.02
Smokes	13.00	0.02
First sexual intercourse	7.00	0.01

Figure 3: Missing data.

After filling missing values, I analyzed remaining features and I noticed that:

- *STDs:cervical condylomatosis* assumes always the same value: **858/858** instances have value **0**.
- *STDs:AIDS* assumes always the same value: **858/858** instances have value **0**.

So I decided to drop both the features. After this operation the dataset has **858** instances and **32** features.

2.4 Features correlation

In the broadest sense **correlation** is any statistical association, though it commonly refers to the degree to which a pair of variables are linearly related. It is a way to understand the relationship between multiple variables and attributes in your dataset. Using Correlation, you can get some insights such as:

- One or multiple attributes depend on another attribute or a cause for another attribute.
- One or multiple attributes are associated with other attributes.

In order to check correlation in the dataset I used the Panda's method who implement the **Pearson correlation**: the Pearson correlation coefficient, r, can take a range of values from +1 to -1. A value of 0 indicates that there is *no correlation* between the two variables. A value greater than 0 indicates a *positive correlation* that is, as the value of one variable increases, so does the value of the other variable. A value less than 0 indicates a *negative correlation* that is, as the value of one variable increases, the value of the other variable decreases[11]. This is shown in the diagram below:

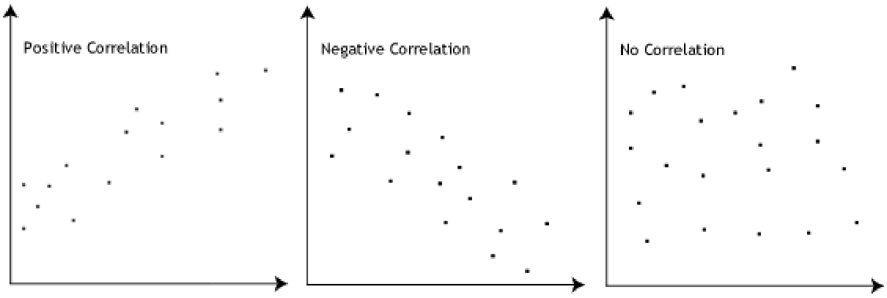


Figure 4: Pearson Correlation.

If the dataset has perfectly positive or negative attributes then there is a high chance that the performance of the model will be impacted by a problem called **Multicollinearity** that can lead to skewed or misleading results. Luckily, decision trees and boosted trees algorithms are immune to multicollinearity by nature. When they decide to split, the tree will choose only one of the perfectly correlated features. However, other algorithms like *Logistic Regression* are not immune to that problem, so I have to check if the dataset is affected by this problem and fix it before training the model. There are several ways to deal with multicollinearity, the common one is to delete or eliminate one of the *perfectly correlated* features, where perfectly correlated means +1 or -1 but you can choice to move your range like for example interpreting it as a score between 0.9 and 1 (and between -0.9 and -1).

Analyzing the correlation matrix (see Figure 5) I noticed that:

- *STDs:condylomatosis* and *STDs:vulvo-perineal condylomatosis* have a correlation of **0.99**
- *STDs (number)* and *STDs Number of diagnosis* have a correlation of **0.90**
- *STDs* and *STDs Number of diagnosis* have a correlation of **0.91**
- *STDs* and *STDs (number)* have a correlation of **0.92**

So I dropped features *STDs*, *STDs Number of diagnosis* and *STDs:vulvo-perineal condylomatosis*.

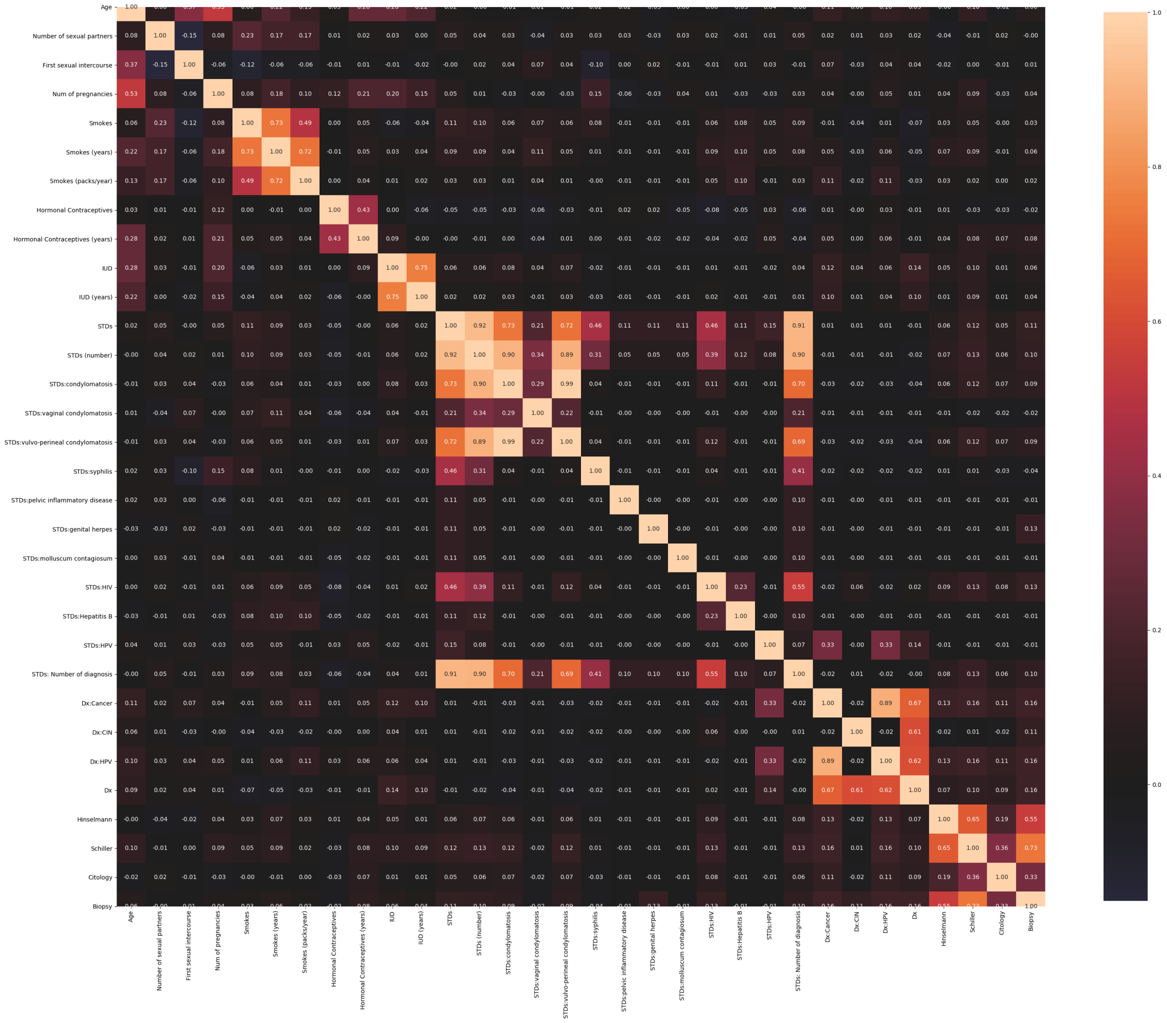


Figure 5: Correlation Matrix.

2.5 Feature transformation

Transformation of data improve the accuracy of the algorithm, main reasons of applying this operation are:

- Certain algorithms require a specific format for data
- Certain algorithms suffer for unbalanced scaling of features
- Certain algorithms perform better or converge faster when features are on a relatively similar scale and/or close to normally distributed (e.g *logistic regression*, *nearest neighbors*, *neural networks* and *support vector machines*)

We have to apply **normalization** to our data so that all features are placed on equal standing; Scaling and standardizing can help features arrive in more digestible form for these algorithms[12]. I tried different scalers and and the best choice was to scale features to lie between a given minimum and maximum value (between zero and one) using *MinMaxScaler*; the motivation to use this scaling include robustness to very small standard deviations of features and preserving zero entries in sparse data.

2.6 Target feature distribution

Notice how **imbalanced** is the dataset: most of the response to the four different tests are negative (no cancer). If we use this dataframe as the base for our predictive models and analysis we might get a lot of errors and our algorithms will probably overfit since it will assume that most patient are cancer free. But the aim is to *detect patterns that give signs of cancer*, so we have to deal with imbalanced data and find strategies to avoid overfitting.

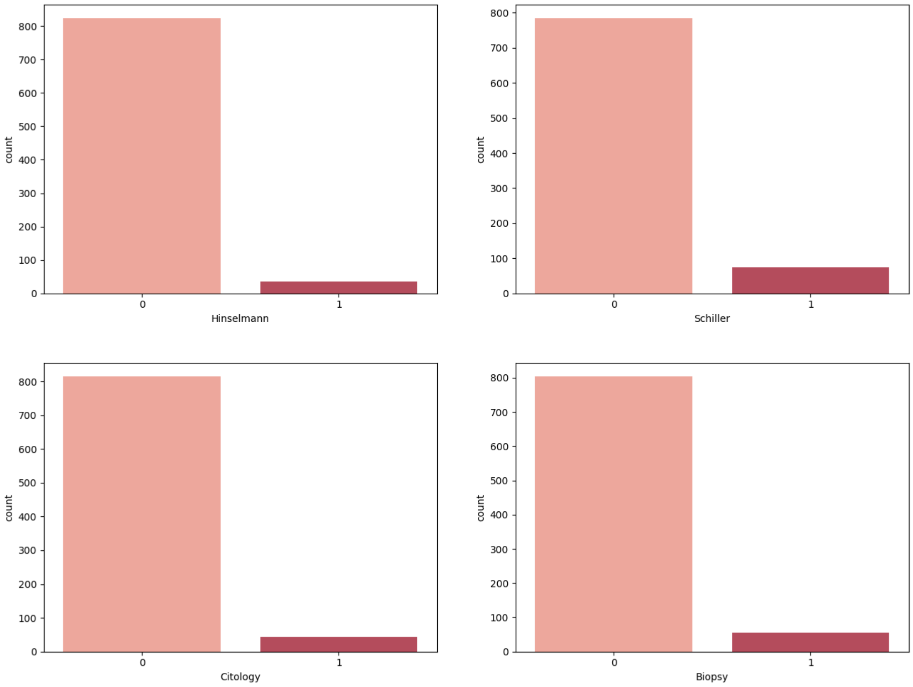
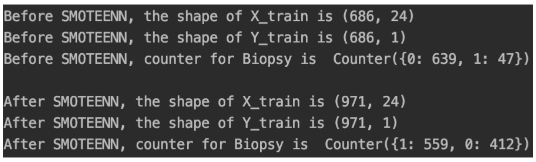


Figure 6: Target features distribution.

2.7 Sampling the dataset

Class imbalance is when each class does not make up an equal portion of your data-set and it is possible to reach an accuracy of 90% by simply predicting the most relevant class every time, but this provides a useless classifier for this project. A way to fix imbalanced data-sets is simply to balance them, either by oversampling instances of the minority class or undersampling instances of the majority class: **Sampling**. I tried both oversampling and undersampling and the one that performed better was **SMOTEEN**[15]: it combines oversampling and undersampling using SMOTE and Edited Nearest Neighbours.



```
Before SMOTEENN, the shape of X_train is (686, 24)
Before SMOTEENN, the shape of Y_train is (686, 1)
Before SMOTEENN, counter for Biopsy is  Counter({0: 639, 1: 47})

After SMOTEENN, the shape of X_train is (971, 24)
After SMOTEENN, the shape of Y_train is (971, 1)
After SMOTEENN, counter for Biopsy is  Counter({1: 559, 0: 412})
```

Figure 7: Before and after SMOTE.

2.8 Evaluation metrics

The following are the metrics I used to evaluate and compare models. I did not use Accuracy since it is widely not recommended when your dataset is imbalaced.

- **Sensitivity**, also called the true positive rate, the recall, or probability of detection in some fields, measures the proportion of actual positives that are correctly identified as such (e.g. the percentage of sick people who are correctly identified as having the condition).

$$Sensitivity = \frac{TP}{TP + FN}$$

- **Specificity**, also called the true negative rate, measures the proportion of actual negatives that are correctly identified as such (e.g. the percentage of healthy people who are correctly identified as not having the condition).

$$Specificity = \frac{TN}{TN + FP}$$

- **Precision**, also called true positive rate, is the fraction of the positive predictions that are actually positive.

$$Precision = \frac{TP}{TP + FP}$$

- **F1-score**, also called F-score or F-measure, is the harmonic mean of the precision and recall, where an F1 score reaches its best value at 1 (perfect precision and recall) and worst at 0.

$$F - score = 2 * \frac{Precision * Recall}{Precision + Recall}$$

- **AUC - ROC**, is a performance measurement for classification problem at various thresholds settings. ROC is a probability curve and AUC represents degree or measure of separability. It tells how much model is capable of distinguishing between classes. Higher the AUC, better the model is at predicting 0s as 0s and 1s as 1s. By analogy, Higher the AUC, better the model is at distinguishing between patients with cancer and no cancer.

3 Classification

3.1 The models

In this project I used 5 different models: *Logistic Regression*, *Support Vector Machine*, *Decision Trees*, *K-Nearest Neighbors* and *Multi Layer Perceptron*. I divided the dataset into 20% Training set and 80% Test set and then performed a Grid-search to find the optimal hyperparameters and a K-Fold Cross-Validation where K=5.

3.1.1 Logistic Regression

Logistic Regression is a Machine Learning algorithm which is used for the classification problems, it is a predictive analysis algorithm and based on the concept of probability. Hyperparameter tuning was performed on *solver*, *penalty* and *c*; others are the default ones.

LR best parameters: solver=liblinear, penalty=l1, C=100

Performance:

- F-measure train: 0.6, F-measure test: 0.5
- sensitivity (TP) train: 0.53, sensitivity (TP) test: 0.67
- Specificity: 0.71
- Precision: 0.54
- AUC: 0.67
- TN: 117, FP: 47, FN: 3, TP: 5

3.1.2 Support Vector Machine

A Support Vector Machine (SVM) is a discriminative classifier formally defined by a separating hyperplane. Hyperparameter tuning was performed on *C*, *kernel* and *gamma*; others are the default ones.

SVM best parameters: C=1000, kernel=rbf, gamma=scale

Performance:

- F-measure train: 0.89, F-measure test: 0.48
- sensitivity (TP) train: 0.86, sensitivity (TP) test: 0.56
- Specificity: 0.75
- Precision: 0.51
- AUC: 0.51
- TN: 123, FP: 41, FN: 5, TP: 3

3.1.3 Decision Tree

A Decision Tree is a simple representation for classifying examples. It is a Supervised Machine Learning where the data is continuously split according to a certain parameter. Hyperparameter tuning was performed on: *criterion* and *splitter*; while others are the default ones.

Dtree best parameters: criterion=entropy, splitter=random

Performance:

- F-measure train: 0.94, F-measure test: 0.57
- sensitivity (TP) train: 0.93, sensitivity (TP) test: 0.73
- Specificity:0.73
- Precision: 0.56
- AUC: 0.73
- TN: 136, FP: 28, FN: 3, TP: 5

3.1.4 K-Nearest Neighbors

In k-NN classification sn object is classified by a plurality vote of its neighbors, with the object being assigned to the class most common among its k nearest neighbors (k is a positive integer, typically small). Hyperparameter tuning was performed on *weights*, *n_neighbors* and *algorithm*; others are the default ones.

KNN best parameters: weights=distance, n_neighbors=5, algorithm=auto

Performance:

- F-measure train: 0.95, F-measure test: 0.6
- sensitivity (TP) train: 0.93, sensitivity (TP) test: 0.76
- Specificity: 0.78
- Precision: 0.6
- AUC: 0.81
- TN: 125, FP: 39, FN: 2, TP: 6

3.1.5 Multilayer Perceptron

A MLP is a deep, artificial neural network composed of more than one perceptron. The input layer to receive the signal, the output layer that makes a decision or prediction about the input and in between those two, an arbitrary number of hidden layers that are the true computational engine of the MLP. Hyperparameter tuning was performed on *hidden_layer_sizes*, *activation* and *solver*; others are the default ones.

KNN best parameters: hidden_layer_sizes=28, activation=tanh, solver=lbfgs

Performance:

- F-measure train: 0.95, F-measure test: 0.54
- Sensitivity (TP) train: 0.94, Sensitivity (TP) test: 0.71

- Specificity: 0.79
- Precision: 0.55
- AUC: 0.83
- TN: 130, FP: 34, FN: 3, TP: 5

4 Evaluation

Unfortunately models suffers of overfitting, they all have very good results in training phase but not soo god in testing phase. I mainly analyzed two metrics: **AUC** and **Sensitivity**.

By looking at the **ROC** and at the **AUC**, that explain how good is the model at distinguish between patients with cancer and patients without cancer, it's possible to say:

- *Support Vector Machine* classifier is the worst model since it makes a random guess, much like a human would do (AUC=0.51).
- *K-Nearest Neighbors* classifier and *Multilayer Perceptron* classifier are the two best models since they have a value near to 1 that is the maximum value that can be achieved (respectively 0.81 and 0.83).

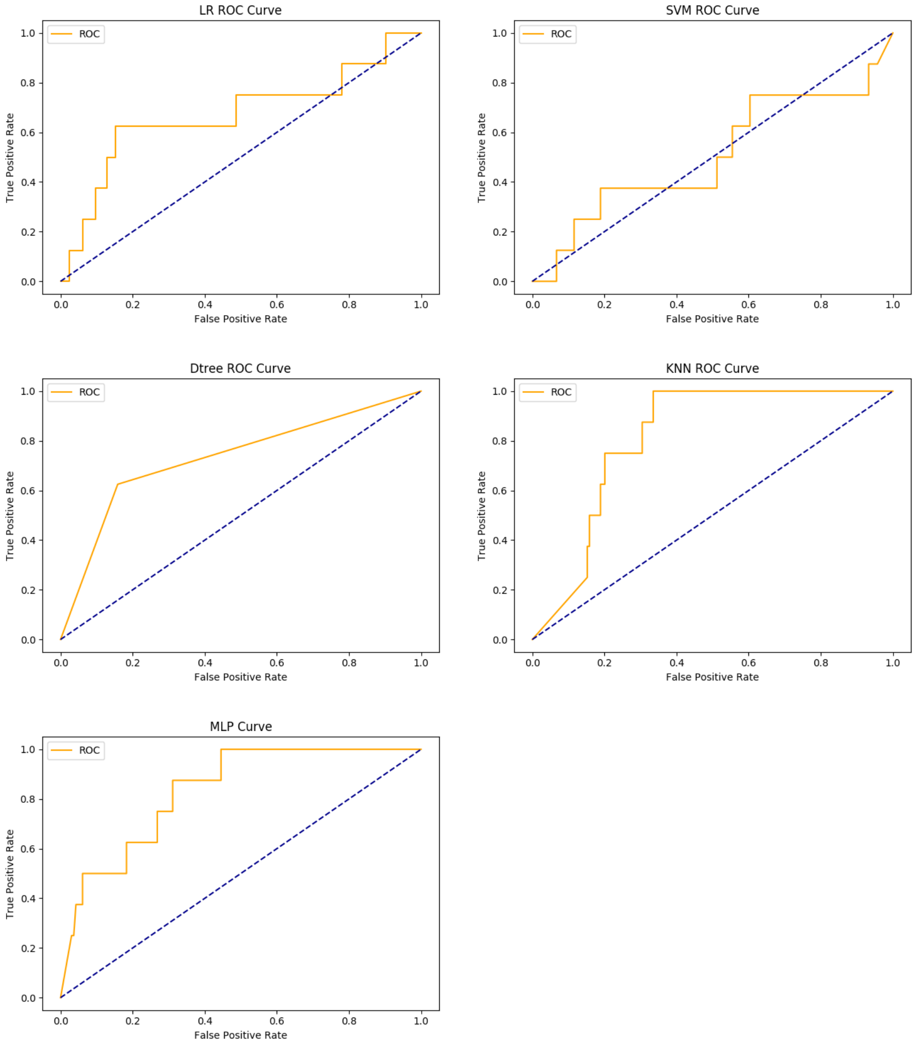


Figure 8: Single ROC curves.

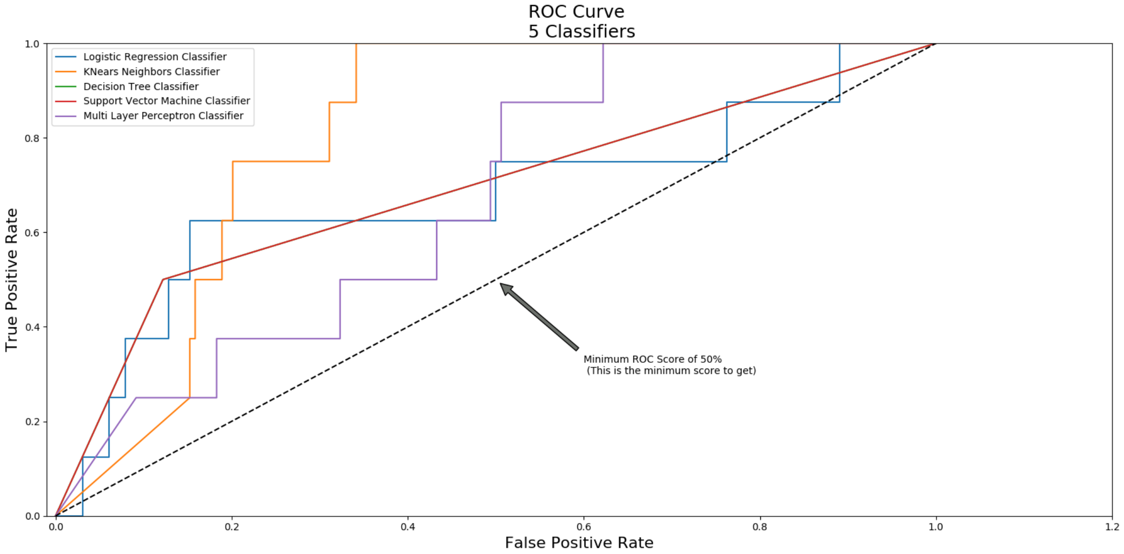


Figure 9: Multiple ROC curves.

Another value that is important for this classification task is **sensitivity**; a high sensitivity is clearly important where the test is used to identify a serious but treatable disease. The error in detecting positive cancer patient as negative cancer patient is very dangerous and may lead to death as a result of staying without necessary medical procedures. The model with the highest sensitivity were (again) the *K-Nearest Neighbors* and the *Multilayer Perceptron* with values rispectively of 0.76 and 0.71.

The recap of all the performance of the 5 classifiers looking at table 2:

	F-score	**AUC**	**Sensitivity**	**Precision**	**Specificity**
LR	0.50	0.67	0.67	0.54	0.71
SVM	0.48	0.51	0.56	0.51	0.75
DTree	0.57	0.73	0.73	0.56	0.73
KNN	0.6	0.81	0.76	0.6	0.78
MLP	0.54	0.83	0.71	0.55	0.79

Table 2: Models performance.

By looking at this table we can conclude that *Decision Tree*, *k-Nearest Neighbors* and *Multilayer Perceptron* are the three best classifier even if results could be better.

Since the two papers[4][3] that I referred to for the development of my project used the Decision Tree Classifier, I compared mine results with theirs (see Table 3) and it turned out that my results were better than [3] but worst than [3]. This is a god result for me since mine is just a simple project and the Dtree classifier is not the model that achieved the best results.

	Paper[3]	**Paper[4]**	**Mine**
Sensitivity	0.43	1.0	0.73
Precision	0.43	0.95	0.56
F-score	0.4	0.97	0.57

Table 3: Comparing performance with papers.

5 Conclusion

The dataset was truly imbalanced, almost all features have values 0 and only few target variables had a positive result for the Biopsy test (and for all the other target features). The use of different metrics and the implementation of sampling techniques on the training set helped a lot, but models still oversampled the test set. Next step could be to try a *Cost-Sensitive Learning* that takes the misclassification costs into consideration in order to minimize the total cost and also try a *Multi-label* classification with the sum of all the four target features. There are several type of classification that could be done using this dataset, but the hardest part is to to 'adjust it', maybe collecting more data.

References

[1] Estimates of incidence and mortality of cervical cancer in 2018: a worldwide analysis
https://www.thelancet.com/action/showPdf?pii=S2214-109X%2819%2930482-6

[2] UCI Machine Learning repository: Cervical cancer (Risk Factors) Data Set
https://archive.ics.uci.edu/ml/datasets/Cervical+cancer+%28Risk+Factors%29

[3] Hayder K. Fatlawi: Enhanced Classification Model for Cervical Cancer Dataset based on Cost Sensitive Classifier
https://www.researchgate.net/publication/325710505_Enhanced_Classification_Model_for_Cervical_Cancer_Dataset_based_on_Cost_Sensitive_Classifier

[4] Y. M. S. Al-Wesabi, Avishek Choudhury, Daehan Won: Classification of Cervical Cancer Dataset
https://arxiv.org/pdf/1812.10383.pdf

[5] MedicineNet: MedTerms Medical Dictionary - Medical Definition of DX
https://www.medicinenet.com/script/main/art.asp?articlekey=33829

[6] National Cancer Institute: NCI Dictionary of Cancer Terms - definition of CIN
https://www.cancer.gov/publications/dictionaries/cancer-terms/def/cin

[7] National Cancer Institute: NCI Dictionary of Cancer Terms - definition of Colposcopy
https://www.cancer.gov/publications/dictionaries/cancer-terms/def/colposcopy

[8] National Cancer Institute: NCI Dictionary of Cancer Terms - definition of Schiller test
https://www.cancer.gov/publications/dictionaries/cancer-terms/def/schiller-test

[9] MedicineNet: MedTerms Medical Dictionary - Medical Definition of Citology
https://www.medicinenet.com/script/main/art.asp?articlekey=80540

[10] National Cancer Institute: NCI Dictionary of Cancer Terms - definition of Biopsy
https://www.cancer.gov/publications/dictionaries/cancer-terms/def/biopsy

[11] Pearson Correlation
https://statistics.laerd.com/statistical-guides/pearson-correlation-coefficient-statistical-guide.php

[12] Paola Velardi: Pratical ML, feature engineering
https://twiki.di.uniroma1.it/pub/ApprAuto/WebHome/2b.FeatureEngineering.pptx

[13] Scikit-learn: Preprocessing data
https://scikit-learn.org/stable/modules/preprocessing.html

[14] Abdul Ghaaliq Lalkhen, MB ChB FRCA, Anthony McCluskey, BSc MB ChB FRCA. Clinical tests: sensitivity and specificity
https://academic.oup.com/bjaed/article/8/6/221/406440

[15] SMOTEENN
https://imbalanced-learn.readthedocs.io/en/stable/generated/imblearn.combine.SMOTEENN.html