UNIVERSITÉ PARIS-DAUPHINE – PSL
CENTRE DE RECHERCHE EN MATHÉMATIQUES DE LA DÉCISION

THYROID CANCER RECURRENCE

AXELLE MERIC

COURSE SUPERVISOR:

QUENTIN GUIBERT

ASSISTANT PROFESSOR (MAÎTRE DE CONFÉRENCES) - CEREMADE

DATA VISUALISATION WITH R
ACADEMIC YEAR 2025/2026

Contents

1 Introduction

The aim of this project is to study Thyroid Cancer Recurrence given medical information of the patient. This project of data visualisation is based on a previous project done in statistical learning course of Master 1 [7] [6]. It was done in `Python` and we will use some of the developed model to focus here on the visualisation part.

1.1 Dataset Presentation

In this project we use a dataset from [5]. It consists of 383 patients who had previously had thyroid cancer. There is exactly one continuous feature, which is `Age`. The 15 other features are categorical ones, as shown in Figure 1. The target variable is Recurred, which indicates whether or not the cancer recurred for each patient (Figure 2). The problem can be formulated as a supervised binary classification task, where the goal is to predict the recurrence of thyroid cancer using a set of clinical and pathological features.

Table 1: Description and modalities of the features

Feature	**Modalities**
`Age`	Continuous variable (e.g., 15 to 82 years)
`Gender`	F (Female), M (Male)
`Smoking`	No, Yes
`Hx Smoking`	No, Yes
`Hx Radiotherapy`	No, Yes
`Thyroid Function`	Euthyroid, Clinical Hyperthyroidism, Clinical Hypothyroidism, Subclinical Hyperthyroidism, Subclinical Hypothyroidism
`Physical Examination`	Normal, Diffuse goiter, Single nodular goiter-left, Single nodular goiter-right, Multinodular goiter
`Adenopathy`	No, Right, Left, Bilateral, Posterior, Extensive
`Pathology`	Papillary, Micropapillary, Follicular, Hurthle cell
`Focality`	Uni-Focal, Multi-Focal
`Risk`	Low, Intermediate, High
`T`	T1a, T1b, T2, T3a, T3b, T4a, T4b
`N`	N0, N1a, N1b
`M`	M0, M1
`Stage`	I, II, III, IVA, IVB
`Response`	Excellent, Indeterminate, Biochemical Incomplete, Structural Incomplete

Table 2: Description of the target variable

Target Variable	**Description**
`Recurred`	Indicates whether the cancer recurred after initial treatment (No, Yes). This is the variable to be predicted.

1.2 Features Analysis

Before building any model, the first step is to explore the features and understand the structure of the dataset. In this section, we conduct a variable analysis to get an overview of the main features and their behavior.

1.2.1 Target Variables

We begin our study with the target variable `Recurred` which has two modalities: `Yes` and `No`. Knowing that the patient previously had cancer, it takes the value `Yes` if the cancer recurred after initial treatment and `No` otherwise. One can see in the distribution (Figure 1) is unbalanced.

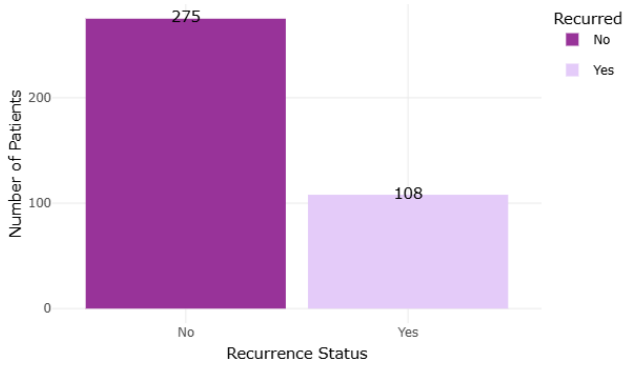


Figure 1: Distribution of `Recurred`

1.2.2 Analysis of other relevant features

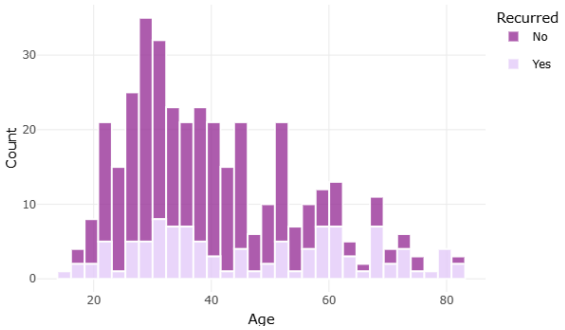


Figure 2: Age Distribution by `Recurrence` status

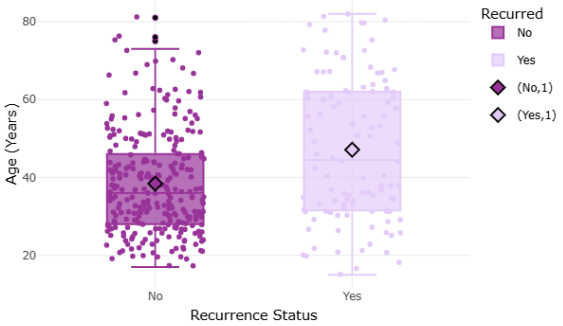
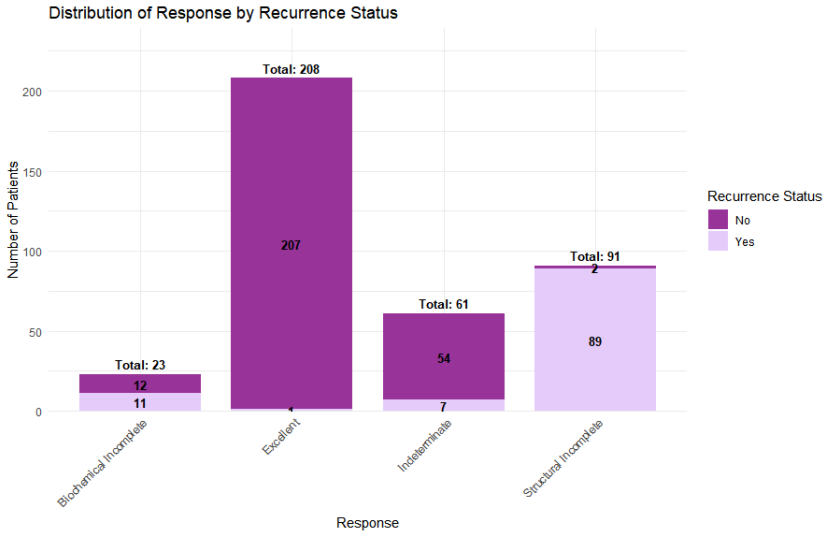


Figure 3: `Age` Distribution vs `Recurrence` status

`Age` It is the only numerical feature. It ranges from 15 to 82. The mean (40.87) is slightly higher that the median (37.00). It is consistent with the peak of values that we observe around the age 31 in figure 2. Furthermore, as we draw the boxplots for the two modalities (Figure 3), we observe a larger interquantile range for the recurrent case. It can be partly explained by the fact that there are less values so it is less precise and it also indicates a greater variability in the age profile of patients who relapse. The black diamond represents the mean. The mean of the recurrent modality is higher than the mean of the other modality. This trend is confirmed by the median represented by

the horizontal line in the boxplot, which is clearly elevated for the `Yes` group compared to the `No` group, suggesting that an advanced age is a significant risk factor for cancer recurrence. Finally, we can observe distinct outliers in the `No` group, representing older patients who, despite their age, did not experience recurrence.

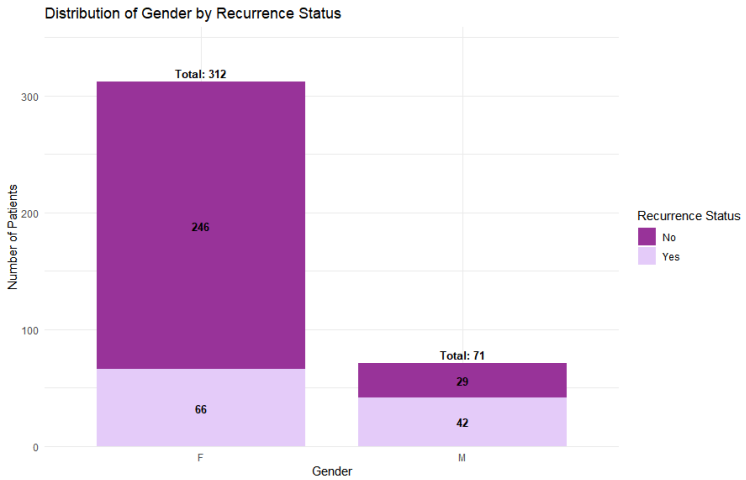
`Response` For all other features, the categorical ones, we print the following plots: a distribution among the recurrence status and a table with the precise number and percentage in each couple of modalities. Figure 4 presents such plots for the feature `Response`. In this plot, one can see that on one hand, a patient which `Response` was "Excellent" will almost surely not have a recurrent cancer (99.5%). Whereas on the other hand, a patient which `Response` was "Structural Incomplete" will almost surely have a recurrent cancer (98.7%). For the other two modalities, it is not as much telling, but we can conjecture that this feature will help with the classification.



Detailed Percentages

Category	No	Yes	Total
Biochemical Incomplete	12 (52.2%)	11 (47.8%)	23
Excellent	207 (99.5%)	1 (0.5%)	208
Indeterminate	54 (88.5%)	7 (11.5%)	61
Structural Incomplete	2 (2.2%)	89 (97.8%)	91

Figure 4: Characteristics of the feature `Response`



Detailed Percentages

Category	No	Yes	Total
F	246 (78.8%)	66 (21.2%)	312
M	29 (40.8%)	42 (59.2%)	71

Figure 5: Characteristics of the feature `Gender`

`Gender` Figure 5 is an other example, but with a different interpretation. Here, we do not observe modalities with a clear link like we saw for the feature `Response`. However, one can still conclude that more Female had a first Thyroid cancer. And that in percent, women tend to have less recurrence than men: only 21.2% for female against 59.2% for male. This is why it is important to draw the table with percentage, as one could be misled by the numbers alone. We cannot assume anything clearly from this feature at this point and will wait for the model to show if the gender has a role in the recurrence.
In such cases, it is interesting to give an another version of the barplot: a version in which each bar has the same height and we do not see the real number of patients but their proportions. The feature `Gender` is a relevant example. In this version, we can visually identify on the barplot that male patient are more susceptible to have a thyroid cancer recurrence (Figure 6).

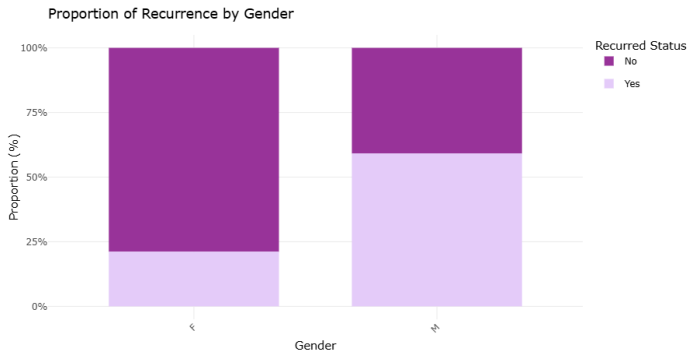


Figure 6: Proportioned version of the barplot for the feature `Gender`

2 Preprocessing

2.1 Data Engineering

2.1.1 Correlation

For this project, we will use models that could require variable selection. Therefore we perform correlations tests using the method of Cramer's V with the `rcompanion` package.

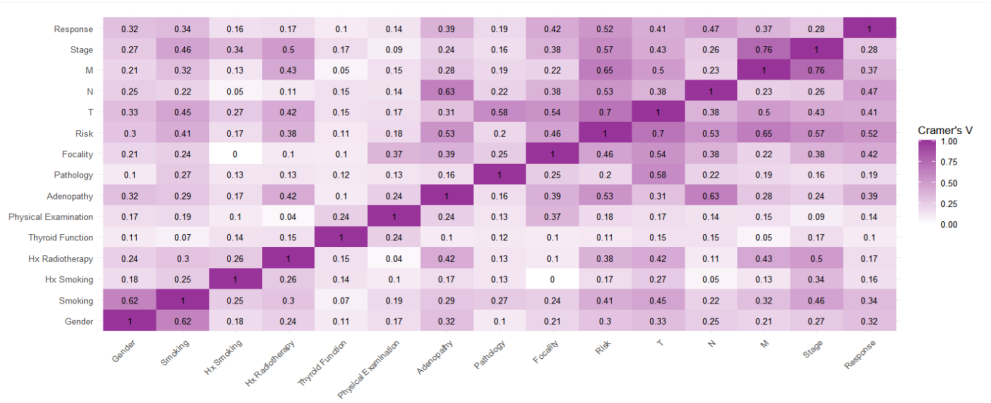


Figure 7: Correlation matrix on categorical features using the method of Cramer's V

We then fix a threshold of 0.7 above which we consider two features to be strongly correlated. This allows us to identify three couples of features:

- `Risk` and `Recurred` are strongly correlated with a Cramer's V of 0.74;
- `M` and `Stage` are strongly correlated with a Cramer's V of 0.76;
- `Response` and `Recurred` are strongly correlated with a Cramer's V of 0.90.

Response will be removed from the dataset as it creates data leakage. Moreover, either `M` or `Stage` need to be removed as it would create noise in models to which some methods are sensitive. On a medical point of view it is more interesting to keep `M, N` and `T` and to let go of `Stage` in order to avoid multicolinearity. However, even if `Risk` and `Recurred` are strongly correlated, we keep it as it is a feature given by doctors before knowing the result. The correlation especially means that the doctors are doing a good job at estimating the risk of cancer recurrence. Thus, it as it is a very good indicator of the classification of our patients.

2.1.2 Duplicates checking

We identify several lines that are identical, we will consider them corresponding to two patients having the same characteristics as it is highly probable to have such occurrences in real life.

2.1.3 Missing values

When it comes to missing values, we must decide whether to remove the affected rows or to replace the missing entries with a mean, a median, or another deterministic value. However, this dataset is complete which simplifies the consideration.

2.1.4 Outliers

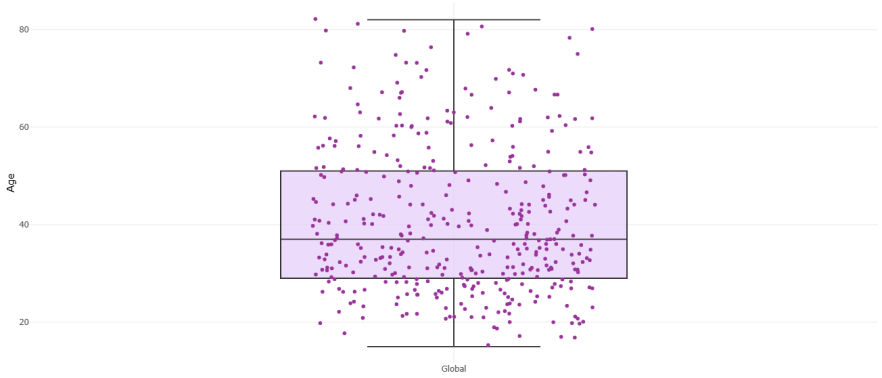


Figure 8: Outlier detection for the feature `Age`

As there is only one numerical features, the only possible source of outliers is the feature `Age`. Outliers are extreme observations that deviate from the general pattern of the data, and we use boxplots as it provides a simple way to visualize and detect these unusual values. Figure 8 presents the boxplot of the feature `Age`. Every observation seems to be located between the whiskers so no significant outliers. To verify this intuition we perform a test with the values of Inter-Quantile Range. It confirms that there are no outliers in the dataset.

2.2 Data Sampling

From the previous subsection, we conclude that will consider two datasets:

- the complete one (`data`) we will be using for models which do not require variable selection,
- and an other one with variable selection (`data_selected`).

We sample our dataset with a 80%/20% train/test split using stratification on the target variable as it is unbalanced (seen in Figure 1). Our train set consist of 306 lines.

2.3 Features Engineering

One-hot encoding In the section we create binary features which are the one-hot encoding of every categorical features as they will be useful for the models built in this project. For the features with two modalities, we create a binary version in which one of the modalities is replaced by 1 and the other by 0. Regarding multi-modal features, here is an example of one-hot encoding for the feature `Risk`.

Table 3: Example of one-hot encoding for the variable `Risk`

Initial Modalities	`Risk_High_Binary`	`Risk_Intermediate_Binary`
High	1	0
Intermediate	0	1
Low	0	0

This feature has three initial modalities so we create two new features `Risk_Intermediate_Binary` and `Risk_High_Binary`. We do not need to create a third feature `Risk_Low_Binary` as we know it corresponds to the case where both other binary features are equal to 0.
It is then important to delete the initial features in order to keep only the binary ones. We do this on our two training sets and we obtain `train_data_Binary` and `train_data_selected_Binary`. And then we repeat the same code on the test sets.

Normalisation of the numerical feature As all other features have now values 0 and 1, it is crucial to normalise the feature `Age`, especially for models which will compute distances.

3 Choice of Metrics

The metrics that are explained in this section separate the observations and their predictions in four classes.

- A **True-Positive** corresponds to a patient which had cancer recurrence and the model classified it as recurrent.
- A **True-Negative** corresponds to a patient which did not have cancer recurrence and the model did not classify it as recurrent.
- A **False-Negative** corresponds to a patient which had cancer recurrence and the model did not classify it as recurrent.

- A **False-Positive** corresponds to a patient which did not have cancer recurrence and the model did classify it as recurrent.

In this precise medical context, a patient with a False-Positive outcome would benefit from enhanced follow-up care, which is generally beneficial. However, they may also receive a more aggressive treatment that could lead to unnecessary side effects, which could be avoided if the patient had been correctly identified. On the other side, a patient with a False-Negative would present high risk of cancer recurrence but will not benefit from this extra care which could lead the patient in the worst case scenario to deaf.
This considerations lead us to choose the following metrics to compare our models:

- **Accuracy:** the proportion of total predictions (both Positive and Negative) that are correct. It is defined as:
$$\text{Accuracy} = \frac{\text{True-Positive} + \text{True-Negative}}{\text{Total Predictions}}$$
- **Recall:** the proportion of actual positive cases that were correctly identified. It is defined as:
$$\text{Recall} = \frac{\text{True-Positive}}{\text{True-Positive} + \text{False Negatives}}$$
- **AUC (Area Under the ROC Curve):** serves as a robust metric for summarizing the performance of a classification model across all possible thresholds. AUC quantifies the model's overall ability to discriminate between positive and negative classes. AUC values range from 0 to 1, where 0 indicates that all predictions are incorrect, and 1 indicates that all predictions are correct.

Since the classes are unbalanced, the accuracy can be misleading due to the under representation of one class. This is why we also compute the AUC, with the goal of achieving the highest possible AUC value which allows us to know if the model has sufficient discriminatory power to make it worth choosing a threshold.

4 Model Evaluation

In this section, we will built three different models. The performance results will be given in the next section.

4.1 Model 1: K Nearest Neighbors (KNN)

The first model we will implement uses the method of K Nearest Neighbors. This is an algorithm based on distances that will classify a new observation according to the class of its nearest neighbors. As it computes distances, we will use the binary version of our dataset and the version with variable selection as it is sensitive to noises.
We find a optimal parameter K of 15, then we train our model. We obtain the following confusion matrix and ROC curve presented in the Figure 9.
With this first model, 68 patients of the test set have been correctly classified, 8 patients are False-Negative (had recurrence but not classified so) and only one patient is False-Positive (no recurrence but classified as recurrent). The accuracy is equal to 0.88 and the recall is equal to 0.64. It indicates good performance in the majority of cases, however as False-Negative could have very dangerous consequences, we will try other models to reduce this number.

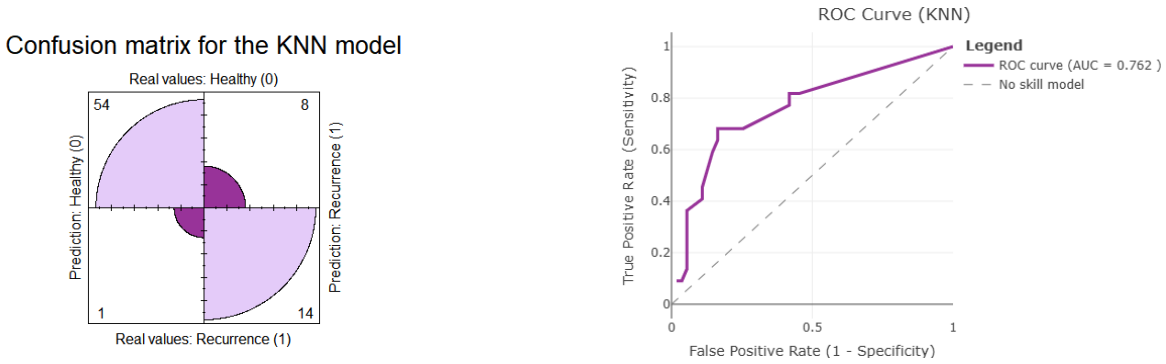


Figure 9: Confusion matrix and ROC curve obtained with the KNN method

4.2 Model 2: Logistic Regression with Lasso Regularisation (LR)

The Logistic regression method is a generalisation of linear models. In this we also add an L1 penalty, a Lasso regularisation. It will do variable selection by choosing an optimal coefficient for each feature and setting some of them to 0. For this reasons, we us the initial training set.
We find a optimal parameter $\lambda = 0.017$, then we train our model. We obtain the following confusion matrix and ROC curve presented in the Figure 10.

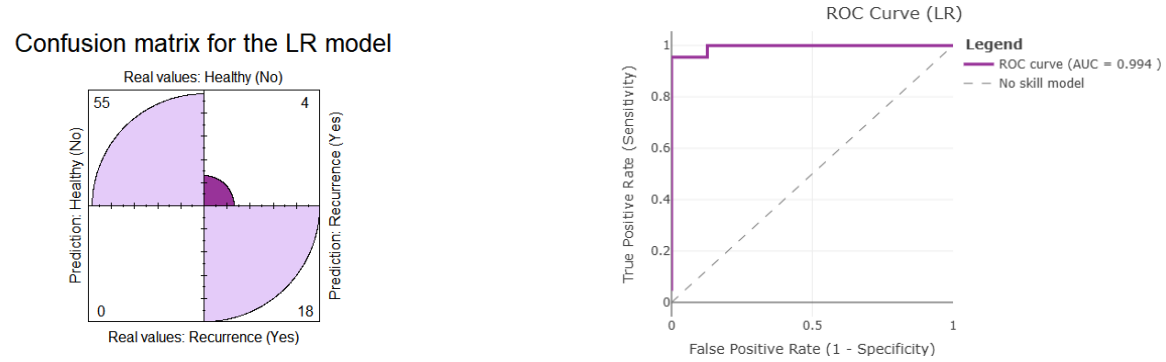


Figure 10: Confusion matrix and ROC curve obtained with the LR method

With this second model, there is no False-Positive and only 4 False-Negative. 73 patients have been correctly identified. The accuracy is 0.95 and the recall 0.81. This model shows better performances than the previous one. Only 6% of the patients have been wrongly classified, which is still considerable for the incurred risk. The ROC curve shows very good performances according to the North-West-Rule.
As this is a glm model, it enables us to interpret the effect of each selected feature on the probability of cancer recurrence. Figure **??** presents the coefficients that remained non null after the L1 regularisation. The other features not listed in this figure are assumed to have negligible impact on the classification. As conjectured in the Preproceesing section, the feature `Response` is the most significant feature. An `Excellent` or `Intermediate Response` reduces the risk of recurrence as the coefficient are both negative, whereas the `Structural Incomplete Response` increases the risk as the coefficient is positive. The same way we can observe the impact of the other mentioned features.

Table 4: Relevant coefficient of the LR model with Lasso regularisation

Feature	Coefficient
`Response_Excellent`	-1.33244375
`Response_Structural_Incomplete`	1.18263183
`Risk_Low`	-0.55136080
`Response_Indeterminate`	-0.46324889
`Stage_II`	0.27560477
`N_N1b`	0.20491250
`Age`	0.10117990
`Pathology_Hurthel_cell`	-0.05975526
`Thyroid_Function_Euthyroid`	0.04787256

4.3 Model 3: Random Forest (RF)

This last method, the Random Forest, is a bagging method which train several decision trees on independent sub-datasets. The final classification is the class which has the majority among the decision trees. This method requires neither variable selection nor one-hot encoding.

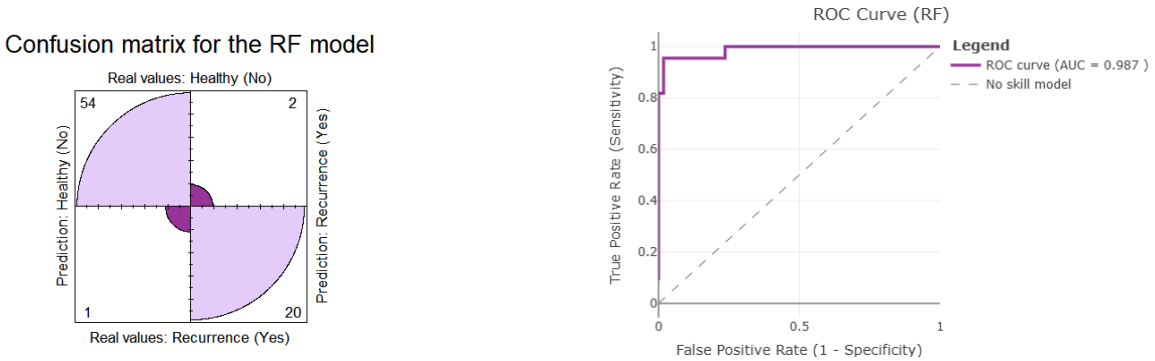


Figure 11: Confusion matrix and ROC curve obtained with the RF method

This last model shows better performances than the two others in the confusion matrix. Only three patients were misclassified: 1 False-Positive and 2 False-Negative. The accuracy is 0.96 and the recall 0.91. The ROC curve also shows great performances according to the North-West-Rule and an AUC of 0.99.

5 Performance comparison

The performance metrics of the three implemented models have been summarised in the following Table 5. The Accuracy and the Recall are maximal for the RF model, and the AUC is optimal for the LR with Lasso regularisation.

Table 5: Comparison of model performance based on accuracy, recall, and AUC on the test set

Model	**Accuracy**	**Recall**	**AUC**
K-Nearest Neighbors	0.883	0.636	0.762
Logistic Regression	0.948	0.818	0.994
Random Forest	0.961	0.909	0.987

6 Conclusion

Throughout this project, we developed three supervised learning models to predict the recurrence of thyroid cancer based on clinical and pathological features. We focused on three key performance metrics: Accuracy, Recall and AUC.

The Random Forest showed best performances for two of the three metrics. It still has a very good performance for the AUC with a value of 0.987. The Logistic Regression with Lasso Regularisation also performed very well on the three metrics with an optimal AUC. The K-Nearest Neighbors had the weakest results, especially for a Recall of 0.636 which was presented as a very important metric regarding this clinical setting. All values are presented in Table 5

The Random Forest and the Logistic Regression both show very strong performances. Even if the Random Forest slightly outperforms the Logistic Regression, the choice will be made to select the Logistic Regression as it enables medical expert to understand the role of each factor in the final decision of the model. Indeed as the Random Forest is a black box, it is more complicated to understand how the model performs. Whereas the Logistic Regression is efficient, transparent and understandable which are qualities that are appreciated in a clinical environment.

One limitation of this dataset was that some modalities of some categorical feature did not have any values which limits the role of such modalities in the classification. It will be especially limiting if a new patient classify so as the model will not be able to use this information.

Finally, in order to help the medical team in there detection, a web-based application was developed in R shiny. This interface acts as a decision support tool, estimating the probability of recurrence based on our Lasso Logistic Regression model. It provides clinicians with an immediate risk assessment using the patient's specific parameters. The tool is available at the following link:

`https://axellemeric.shinyapps.io/axelle_meric/`

The complete code of this project and the R shiny application can be found on the following GitHub repository:

`https://github.com/AxelleMeric/Thyroid-Cancer-Recurrence_Data-Visualisation`

References

[1] Google. Use of Generative AI to help wih bug in the code in R and HTML and to implement an R shiny app, however always under supervision and has been entirely mastered.

[2] Kaggle. *Kaggle*. Kaggle page dedicated to the dataset, however no project done in R. URL: https://www.kaggle.com/datasets/joebeachcapital/differentiated-thyroid-cancer-recurrence/code?datasetId=4324904&language=Python&outputs=Visualization.

[3] Katia Meziani. *Regression and Classification methods Course Notes*. Course material from Université Dauphine-PSL / M2 ISF. Lecture notes used in the Regression and Classification methods course. Academic Year 2025-2026.

[4] *RDocumentation*. URL: https://www.rdocumentation.org/.

[5] Aidin Tarokhian Shiva Borzooei. *Differentiated Thyroid Cancer Recurrence*. 2023. DOI: 10.24432/C5632J. URL: https://archive.ics.uci.edu/dataset/915.

[6] *Thyroid Cancer Recurrence Project of M1 Statisticla learning class, Python code.ipynb at main · AxelleMeric/Thyroid-Cancer-Recurrence_Data-Visualisation*. en. URL: https://github.com/AxelleMeric/Thyroid-Cancer-Recurrence_Data-Visualisation/blob/main/Python%20code.ipynb.

[7] *Thyroid Cancer Recurrence Project of M1 Statisticla learning class, Report of Python code at main · AxelleMeric/Thyroid-Cancer-Recurrence_Data-Visualisation*. en. URL: https://github.com/AxelleMeric/Thyroid-Cancer-Recurrence_Data-Visualisation/blob/main/Project%20Report%20in%20Python.pdf.