Smoker classification based on biological data

Riccardo Santi, Michele Garbin, Delì Lin

May 2025

1 Introduction and exploratory Data Analysis

In clinical and research contexts, reliably determining whether a patient is a smoker is of interest both for the prevention and management of numerous chronic diseases and for ensuring the reliability of responses in surveys and investigations. Indeed, smoking status can be a sensitive question for someone, and a person might lie to appear healthier.

A possible solution is to use only the patient's biological data to predict whether they are a smoker. Specifically, we will consider the following explanatory variables:

- **Age**: Patient's age.
- **Height/Weight/Waist**: Body measurements (height in cm, weight in kg, waist circumference in cm). Waist size indicates abdominal fat.
- **Eyesight**: Visual acuity (measured in tenths) for each eye (e.g. 10/10 = normal vision)
- **Hearing**: Binary measure (normal/abnormal) of hearing threshold for each ear
- **Blood Pressure**:
  - *Systolic*: Pressure when heart beats (top number)
  - *Diastolic*: Pressure between heartbeats (bottom number)
- **Fasting Blood Sugar**: Blood glucose level after overnight fasting (mg/dL). High values suggest diabetes risk.
- **Cholesterol Panel**:
  - *Total*: All cholesterol in blood
  - *HDL*: "Good" cholesterol that protects arteries
  - *LDL*: "Bad" cholesterol that clogs arteries
  - *Triglycerides*: Blood fats linked to heart disease
- **Hemoglobin**: Oxygen-carrying protein in red blood cells (g/dL). Low levels indicate anemia.
- **Urine Protein**: Marker of kidney health (categorized levels). High levels suggest kidney damage.

- **Serum Creatinine**: Waste product indicating kidney function (mg/dL). Higher values mean poorer kidney filtration.
- **Liver Enzymes**:
  - *AST/ALT*: Indicators of liver inflammation
  - *GGT*: Marker of alcohol/tobacco effects on liver
- **Dental Caries**: Presence of tooth cavities (yes/no)

Our objective is to accurately determine a patient's smoking status:

- **smoking**: 1 or "yes" indicates a smoker, while 0 or "no" indicates a non-smoker.

1.1 Overview of the data set

The working dataset contains N adult observations and $p = 15$ biological variables drawn from routine clinical examinations[1].

The target variable `Smoking` is binary (1 = current smoker, 0 = non-smoker). No missing data. The sample size of smokers is 14,318, and non-smokers is 24,666.

The dataset is freely accessible online and can be found on Kaggle.

The project was developed collaboratively by the three authors. All code and implementation details can be found in the GitHub repository available at the following link

1.2 Correlation among continuous features

Certain variables exhibit strong correlations. Notably, `relaxation` correlates with `systolic` (blood pressure), `LDL`, and `cholesterol`. Additionally, `ALT` and `AST` (liver enzymes) show high correlation, as do `waist circumference` and `weight`. These groups of variables likely contain redundant information.

1.3 Distributional Comparisons (Boxplots)

Figure 1 reveals that smokers tend to exhibit:

1. Younger age distributions.
2. Higher median values for both height (cm) and hemoglobin - variables previously identified as correlated - with both distributions showing slight left-skewness across groups.
3. Elevated median values for waist circumference and weight, suggesting smokers' habits may contribute to weight gain.
4. Numerous upper-tail outliers present across nearly all variables.

[1] E.g., liver-function enzymes, lipid profile, renal markers, anthropometrics, and categorical descriptors such as hearing loss and dental caries.

5. Moderately higher median relaxation values among smokers.

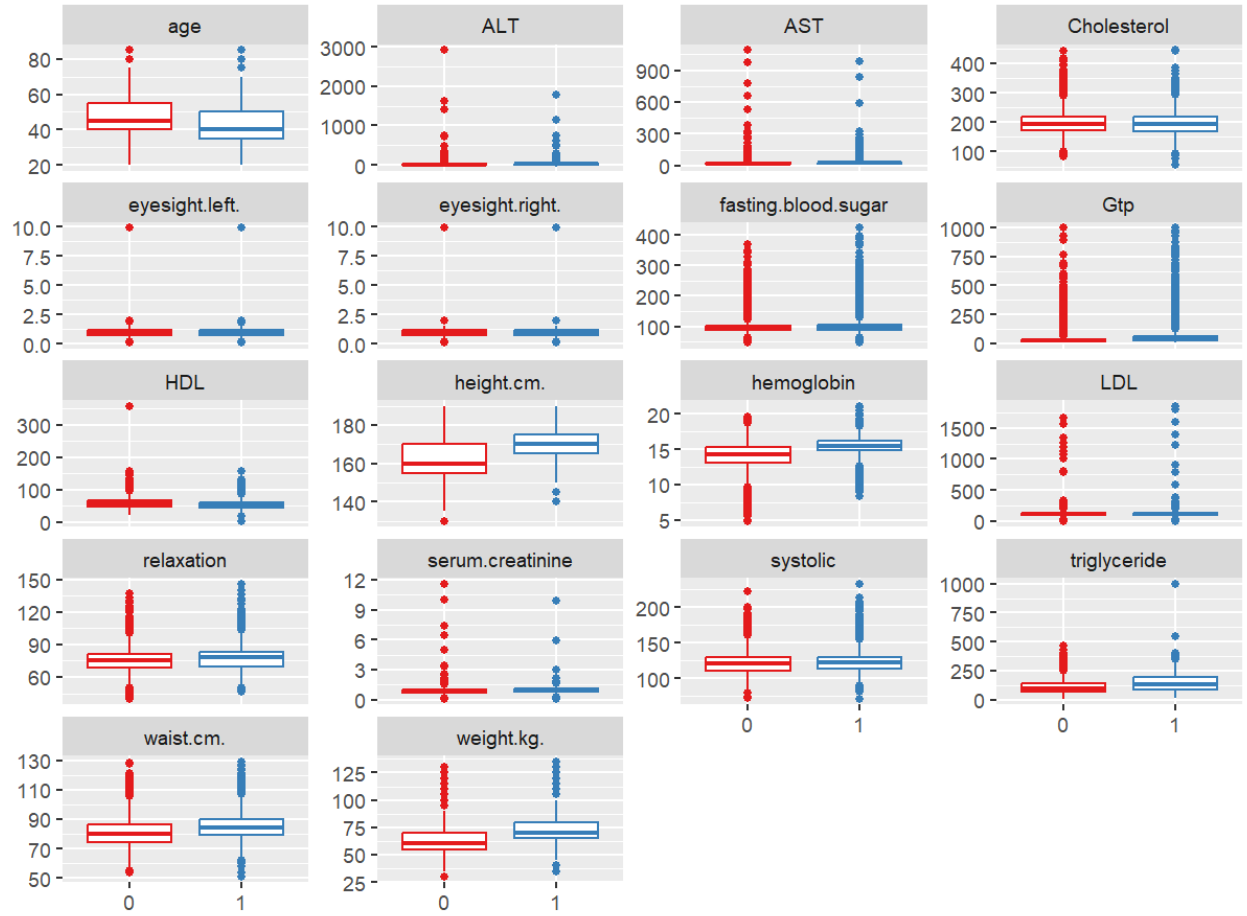

Figure 1: Boxplots of continuous variables by smoking status.

1.4 Association between smoking and categorical variables

Bar plots visualize categorical variable distributions between smokers (n=14,318) and non-smokers (n=24,666). Proportional frequencies are used to enable fair comparisons given the substantial group size imbalance, reducing potential bias in distribution assessments.

Renal marker (Urine Protein). For Protein Levels from 1 to 4 it seems that higher the level and higher is the level of smokers, then for the level 5 of urine protein the percentage of non smokers gets higher and the pattern is repeated in levels 5-6, caused by the low numerosity of subjects for these levels of protein.

```
> table(data$Urine.protein)
```

```
    1     2     3     4     5     6
36836  1236   667   182    58     5
```

Dental caries. In the class of smokers, the proportion of people having dental caries is slightly higher, suggesting that tobacco use may be associated with an increased risk of dental caries.

Hearing impairment. The histograms do not reveal substantial differences in relative frequencies. These two variables can be combined into a single dichotomous variable (1 if hearing loss is observed, regardless of whether it affects the left or right ear).

1.5 Association between smoking and continuous variables

We present here some of the most relevant scatterplots. Additional scatterplots can be generated using the provided source code.

- **Height vs. Hemoglobin.** Smokers tend to have greater height and hemoglobin values. These variables also exhibit moderate correlation, potentially because taller individuals may have a larger blood volume, requiring higher hemoglobin levels to meet oxygen demands, and smoking itself can increase hemoglobin concentrations (via carbon monoxide exposure, which stimulates red blood cell production).
- **Waist vs. Weight.** Smokers exhibit a higher values of waist circumferences and weights, reinforcing the "central obesity" signal seen earlier.

2 Stepwise and penalized regression

The simplest way to tackle problems with a binary response is to use logistic regression. Instead of predicting the outcome directly, logistic regression estimates the probability that a given input belongs to a certain class. It uses the *logit function*, which transforms probabilities (ranging from 0 to 1) into values that can be modeled as a linear combination of the predictors. The main formula is:

$$\log\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 x_1 + \cdots + \beta_k x_k$$

where p denotes the probability that a person is a smoker. Once fitted, logistic regression allows us to compute the predicted probability for each observation and make classifications, for example, by using a cutoff of 0.5.

2.1 Stepwise variable selection

As a first strategy to reduce model complexity and potentially improve predictive performance, we adopted stepwise selection methods. These procedures add or remove predictors based on the *Akaike Information Criterion* (AIC). The goal is to balance model fit and complexity: a lower AIC indicates a better trade-off between goodness-of-fit and the number of parameters.

We explored four standard approaches:

- **Backward selection**: starts from the full model including all predictors and iteratively removes the variable that leads to the greatest reduction in AIC when excluded, until no further improvement can be achieved
- **Forward selection**: starts from the null model containing only the intercept, and iteratively adds the variable that leads to the greatest reduction in AIC, until no additional variable improves the model
- **Hybrid selection (starting from the full model)**: allows both inclusion and exclusion of variables at each step, starting from the full model. Variables can be added or removed depending on which operation reduces the AIC the most
- **Hybrid selection (starting from the null model)**: similar to the previous approach, but starting from the null model. At each step, variables can be added or removed based on the impact on AIC

All four procedures led to the same final model, excluding the following variables: `AST`, `eyesight.right.`, and `LDL`. The performance of the fitted model on the test set is summarized in table 1.

Table 1: Performance metrics for the Stepwise logistic regression model

(a) Confusion Matrix

Prediction	**yes**	**no**
yes	2349	1404
no	1831	5921

(b) Metrics

Metric	**Value**
Sensitivity	0.5620
Specificity	0.8083
AUC (Test set)	0.802

2.2 Ridge regression

Ridge regression is a regularization technique used to prevent overfitting in regression models, especially when multicollinearity is present or when the number of predictors is large in comparison to the number of observations. In the context of logistic regression, it introduces a penalty term proportional to the square of the L2 norm of the coefficients. This encourages smaller coefficient values and can improve generalization on unseen data.

In our analysis, we applied ridge logistic regression using a 100-fold cross-validation to identify an appropriate regularization strength (lambda) based on predictive performance on the training set.

The function returns two commonly used values:

- `lambda.min`: the value of λ that minimizes the cross-validated error
- `lambda.1se`: the largest λ such that the cross-validated error is within one standard error of the minimum

Instead of choosing `lambda.min`, which minimizes the average cross-validation error, we selected `lambda.1se`. This decision was motivated by the fact that `lambda.min` not only minimized the error, but also happened to be the smallest lambda value considered in the cross-validation grid, as it is shown in figure 2. As such, the resulting model was very close to the unregularized solution obtained via stepwise selection. To place more emphasis on regularization and avoid overfitting, we opted for `lambda.1se`, which corresponds to a stronger penalty and leads to a more parsimonious model.

Table 2 reports the performance metrics obtained using ridge logistic regression with the regularization parameter set to `lambda.1se`. We selected `lambda.1se` to promote greater penalization and reduce model complexity compared to `lambda.min`, which corresponded to the minimum tested value and closely resembled the stepwise solution. The results show a slight deterioration in performance compared to the stepwise logistic regression, particularly in terms of sensitivity.

Table 2: Performance metrics for the Ridge logistic regression model ($\lambda =$ `lambda.1se`)

(a) Confusion Matrix

Prediction	**0**	**1**
0	6027	1991
1	1298	2189

(b) Metrics

Metric	**Value**
Sensitivity	0.5237
Specificity	0.8228
AUC (Test)	0.799

2.3 Lasso regression

Lasso regression is a regularization technique that, like ridge regression, aims to prevent overfitting by adding a penalty term to the loss function. However, unlike ridge regression, lasso applies an L1 penalty. This key difference allows lasso to shrink some coefficients exactly to zero, effectively performing variable selection as part of the estimation process.

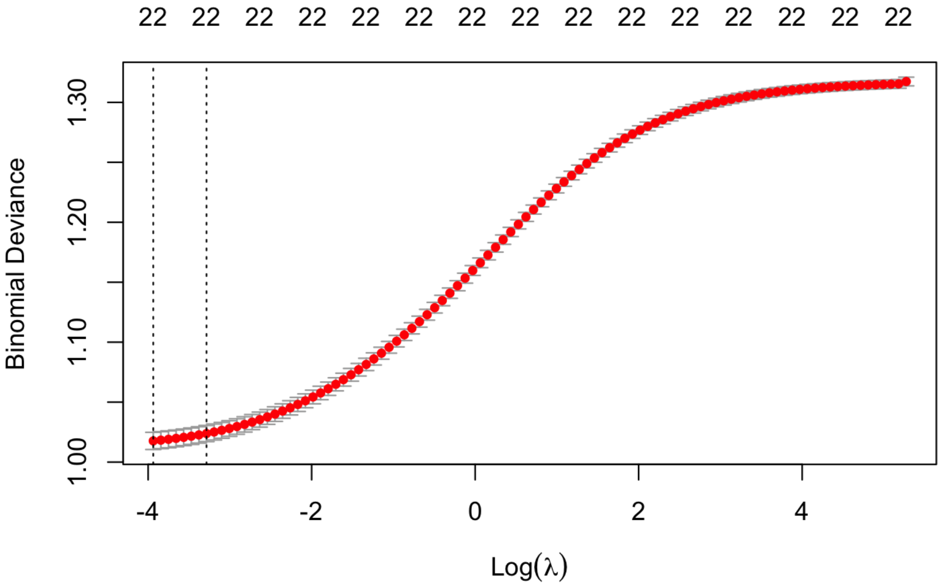

Figure 2: Plot of the cross-validated error as a function of the regularization parameter λ for the Ridge model.

To ensure consistency with our previous approach using ridge regression, we adopted the same methodology for the lasso model. Here are reported the values of lambda used for cross-validation, the associated error (figure 3), and the metrics computed on the selected model (table 3). The results are similar to those obtained with ridge regression.

Table 3: Performance metrics for the Lasso logistic regression model ($\lambda =$ `lambda.1se`)

(a) Confusion Matrix

Prediction	**0**	**1**
0	5962	1945
1	1363	2235

(b) Metrics

Metric	**Value**
Sensitivity	0.5347
Specificity	0.8139
AUC (Test)	0.799

Table 4 contains the estimated coefficients obtained through Lasso logistic regression for $\lambda =$ `lambda.1se`. Lasso performs variable selection by shrinking some coefficients exactly to zero, which are marked with a dot (.). In this case, the following variables were excluded from the model: `age`, `waist.cm.`, `eyesight.left.`, `eyesight.right.`, `hearing.left.`, `hearing.right.`, `relaxation`, `Urine.protein`, and `AST`.

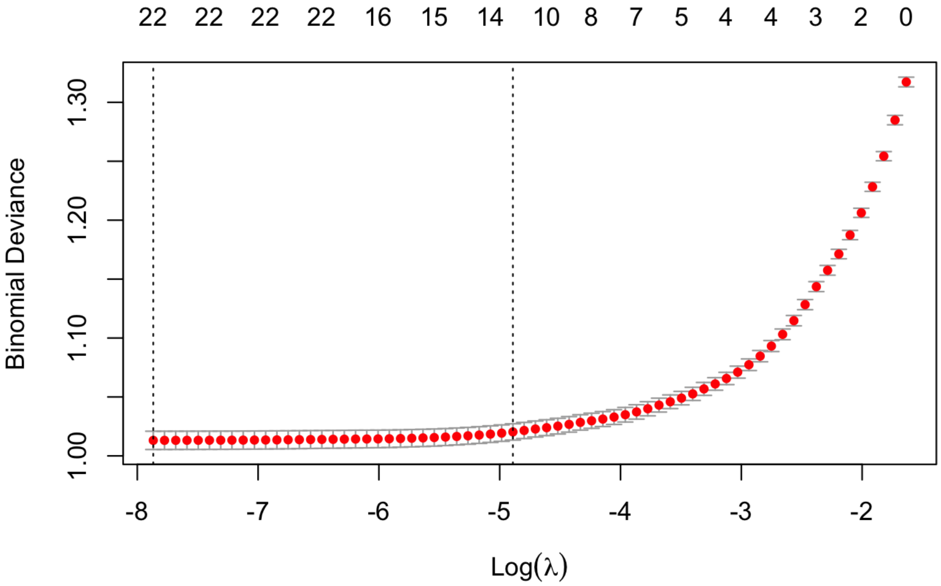

Figure 3: Plot of the cross-validated error as a function of the regularization parameter λ for the Lasso model.

2.4 Elastic net logistic regression

The `glmnet` package allows for regularized logistic regression by specifying `family = "binomial"`. In this case, the model estimates the coefficients by minimizing the following penalized loss function:

$$\mathcal{L}(\beta) = -\frac{1}{n}\sum_{i=1}^{n}\left[y_i \log(p_i) + (1-y_i)\log(1-p_i)\right] + \lambda\left[\alpha\sum_{j=1}^{p}|\beta_j| + (1-\alpha)\sum_{j=1}^{p}\beta_j^2\right], \tag{1}$$

where $p_i = \frac{1}{1+\exp(-x_i^\top \beta)}$ is the predicted probability for observation i, λ is the regularization parameter, and $\alpha \in [0,1]$ controls the balance between Lasso (L1) and Ridge (L2) penalties.

To explore the trade-off between lasso and ridge regularization, we considered 20 different values of α evenly spaced between 0 and 1. For each value of α, we used 50-fold cross-validation to determine the optimal regularization strength λ, specifically selecting `lambda.1se` to favor model simplicity and penalization. Then, for each fitted model, we computed four key performance metrics on the test set: accuracy, sensitivity, specificity, and AUC.

Regarding the performance metrics, we observed that higher values of alpha (closer to 1) tended to yield better results in terms of *accuracy* and *sensitivity.* Conversely, values of α closer to 0 generally led to higher *specificity* and *AUC*. However, $\alpha = 0.8421$ stood out as one of the values associated with the highest AUC, while also providing good performance across the other metrics. For this reason, we selected it as the final

Table 4: Estimated coefficients under the Lasso model with $\lambda =$ `lambda.1se`

```
                       lambda.1se
(Intercept)         -17.309745984
age                   .
height.cm.            0.065976787
weight.kg.           -0.002441706
waist.cm.             .
eyesight.left.        .
eyesight.right.       .
hearing.left.         .
hearing.right.        .
systolic             -0.004164588
relaxation            .
fasting.blood.sugar   0.002907499
Cholesterol          -0.000129743
triglyceride          0.003473012
HDL                  -0.004409955
LDL                  -0.003626599
hemoglobin            0.415145519
Urine.protein         .
serum.creatinine      0.054834214
AST                   .
ALT                  -0.006633506
Gtp                   0.007955240
dental.caries         0.353674868
```

value of α to fit and evaluate the elastic net model (using a 100-fold cross-validation).

Table 5: Performance metrics for the Elastic Net logistic regression model ($\alpha = 0.8421$, $\lambda =$ `lambda.1se`)

(a) Confusion Matrix

Prediction	**yes**	**no**
yes	2244	1359
no	1936	5966

(b) Metrics

Metric	**Value**
Sensitivity	0.5368
Specificity	0.8145
AUC (Test)	0.798

2.5 Summary of regression results

The problem we are addressing lies in the medical domain. Our primary interest is to correctly identify individuals who are actually smokers. In this context, false positives are less problematic; the model should aim to avoid false negatives, since failing to identify a smoker could lead to an underestimated health risk.

For this reason, our priority is to maximize sensitivity. While penalized regressions tend to yield higher specificity values, the best-performing model in this section is the stepwise logistic regression, as it achieves the highest sensitivity and the highest AUC as well.

3 Discriminant Analysis

For the discriminant analysis we chose not to include categorical or discrete variables, because by their very nature they violate the normality assumption required by this technique. In our case the biological variables are approximately normal, which justifies a comparison of Linear Discriminant Analysis (LDA) with classification trees and logistic regression. All continuous predictors were standardized so that differences in scale do not bias the discriminant function. Model performance was evaluated using a dedicated validation set rather than cross-validation primarily because LDA has no hyperparameters requiring optimization. Additionally, preliminary 10-fold cross-validation revealed minimal variation in key performance metrics. We also examined the discriminant coefficients, giving below a brief theoretical description.

Discriminant coefficients Discriminant coefficients arise naturally when we classify observations that are assumed to originate from different multivariate normal distributions. Suppose two populations share the same covariance matrix Σ and have prior probabilities (p_1, p_2). The decision regions are

$$
\begin{aligned}
R_1: \ \exp\Big[-\tfrac{1}{2}(x-\mu_1)^\top\Sigma^{-1}(x-\mu_1) + \tfrac{1}{2}(x-\mu_2)^\top\Sigma^{-1}(x-\mu_2)\Big] \ &\geq \ \frac{p_2}{p_1}, \\
R_2: \ \exp\Big[-\tfrac{1}{2}(x-\mu_1)^\top\Sigma^{-1}(x-\mu_1) + \tfrac{1}{2}(x-\mu_2)^\top\Sigma^{-1}(x-\mu_2)\Big] \ &< \ \frac{p_2}{p_1},
\end{aligned}
$$

which are equivalent to

$$
\begin{aligned}
R_1: (\mu_1-\mu_2)^\top\Sigma^{-1}x - \tfrac{1}{2}(\mu_1-\mu_2)^\top\Sigma^{-1}(\mu_1+\mu_2) \ &\geq \ \ln\Big(\tfrac{p_2}{p_1}\Big), \\
R_2: (\mu_1-\mu_2)^\top\Sigma^{-1}x - \tfrac{1}{2}(\mu_1-\mu_2)^\top\Sigma^{-1}(\mu_1+\mu_2) \ &< \ \ln\Big(\tfrac{p_2}{p_1}\Big).
\end{aligned}
$$

Replacing the theoretical quantities with their sample estimates and assuming equal priors ($p_1 = p_2$), we assign x_0 to population 1 if

$$
(\bar{x}_1-\bar{x}_2)^\top S_{\text{pooled}}^{-1}x_0 \ \geq \ \tfrac{1}{2}(\bar{x}_1-\bar{x}_2)^\top S_{\text{pooled}}^{-1}(\bar{x}_1+\bar{x}_2).
$$

Define

$$
\begin{aligned}
&\hat{y} = (\bar{x}_1-\bar{x}_2)^\top S_{\text{pooled}}^{-1}x \ = \ \hat{a}^\top x, \\
&\hat{m} = \tfrac{1}{2}(\bar{x}_1-\bar{x}_2)^\top S_{\text{pooled}}^{-1}(\bar{x}_1+\bar{x}_2) \ = \ \tfrac{1}{2}(\bar{y}_1+\bar{y}_2), \\
&\bar{y}_1 = \hat{a}^\top\bar{x}_1, \qquad \bar{y}_2 = \hat{a}^\top\bar{x}_2.
\end{aligned}
$$

The rule is therefore $\hat{y} \geq \hat{m}$.

The standardised coefficients $\hat{a}_1^*, \hat{a}_2^*, \ldots, \hat{a}_p^*$ lie in $[-1, 1]$. Setting $\hat{a}_1^* = 1$ and expressing the others as multiples of $\hat{a}_1^*$ makes it easy to gauge the relative discriminatory power of $X_2, \ldots, X_p$ with respect to X_1.

3.1 Linear Discriminant Analysis (LDA)

Three LDA models were fitted:

```
X_train  : original training set after removing variables that violate
           normality by construction
X_train2 : within each cluster of highly correlated predictors,one variable
           was removed
X_train3 : starting from X_train2, variables whose absolute coefficients
           in model$scaling were smaller than 0.1 were discarded
```

Table 6: Coefficients of linear discriminant functions

lda1		lda2		lda3	
age	0.00616	age	0.0126		
height.cm.	0.61345	height.cm.	0.6206	height.cm.	0.6531
weight.kg.	-0.17304	weight.kg.	-0.1451	weight.kg.	-0.1576
waist.cm.	0.04266				
systolic	-0.12003	systolic	-0.0698		
relaxation	0.06623				
fasting.blood.sugar	0.05771	fasting.blood.sugar	0.0629		
Cholesterol	-0.18486				
triglyceride	0.33046	triglyceride	0.2646	triglyceride	0.3042
HDL	-0.00435	HDL	-0.0737		
LDL	0.04028	LDL	-0.0925		
hemoglobin	0.52786	hemoglobin	0.5273	hemoglobin	0.5234
serum.creatinine	0.03145	serum.creatinine	0.0315		
AST	-0.06119	AST	-0.0890		
ALT	-0.04462				
Gtp	0.32894	Gtp	0.3343	Gtp	0.2981

Note that removing certain variables changes the coefficient estimates of the remaining ones.

The model `lda3` therefore retains only those predictors whose variation most strongly influences classification. The third LDA model coefficients provide a interpretable framework for understanding how key physiological markers contribute to classifying smoking status, offering a critical tool for scenarios where self-reported data may be unreliable. Height (cm) and hemoglobin levels show strong positive associations with smoking (coefficients: 0.6531 and 0.5234, respectively), suggesting taller individuals and those with higher hemoglobin are more likely to smoke—a potential link to respiratory demands or hematological adaptations. Conversely, higher weight (kg) correlates negatively (-0.1576), implying lower smoking probability in heavier individuals, possibly due to metabolic or lifestyle factors. Triglycerides and Gtp (0.3042 and 0.2981) show moderate positive ties, aligning with known cardiovascular risks of smoking. While validation is essential, the model's structure underscores the utility of physiological proxies for smoking assessment.

Perfomances metrics evaluated on the validation set gave:

Table 7: Performance metrics for LDA models on validation set

Model	**Sensitivity**	**Specificity**	**AUC**
LDA_full	0.578	0.792	0.8017
LDA_drop4	0.570	0.793	0.8002
LDA_drop11	0.568	0.796	0.7959

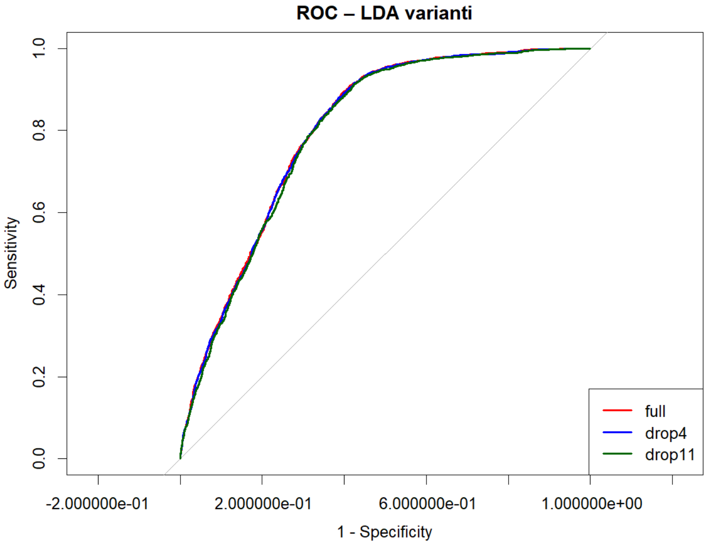

Figure 4: ROC Curve Comparison

As specified in the introduction, as we intend to focus on the sensitivity, here the best model seems to be the linear discriminant analysis that considers all the variables.

3.2 Quadratic Discriminant Analysis(QDA)

The training process using the QDA Model was the same, using the same three training sets used in LDA. The metrics obtained on the validation set are:

Table 8: QDA Model Performance Metrics

Model	**Sensitivity**	**Specificity**	**AUC**
QDA_drop11	0.367	0.885	0.79
QDA_drop4	0.345	0.900	0.790
QDA_full	0.292	0.926	0.79

We observe that QDA exhibits very low sensitivity, indicating few true smokers were identified. The best-performing model in terms of sensibility uses fewer predictors. The high specificity suggests most non-smokers were correctly classified. Performance differences between `QDAdrop11` and `LDAfull` are evident in their confusion matrices:

Table 9: Comparison confusion matrices between LDA_Full and QDA_drop11

QDA_drop11

Prediction	**0**	**1**
0	3832	1608
1	496	934

LDA_Full

Prediction	**0**	**1**
0	3428	1073
1	900	1469

The discriminant function explains QDA's sensitivity issue:

$$\delta_k(x) = \log \pi_k - \frac{1}{2}(x - \mu_k)^\top \Sigma_k^{-1}(x - \mu_k) - \frac{1}{2}\log|\Sigma_k|$$

QDA penalizes classes with larger covariance determinants. In our case, smokers (class 1) have higher variance:

$$-\log|\hat{\Sigma}_1| = 6.714$$
$$-\log|\hat{\Sigma}_0| = 8.782$$

The larger penalty term $\left(-\frac{1}{2}\log|\Sigma_k|\right)$ for smokers reduces their discriminant scores.

IMPORTANT NOTE: we emphasize that in this case it could be limiting to look at the discriminant function since its estimation depends on the quality of the data. As can be seen in figure 4, an option would be to adjust the threshold and use the posterior probabilities estimated by the model for classification.

The final model choosen in this category is the LDA with all predictors. Training the model combining the train and then validation sets yields the following metrics on the test set:

Table 10: Performance metrics for the default model

(a) Confusion Matrix

Prediction	**0**	**1**
0	5867	1742
1	1458	2438

(b) Metrics

Metric	**Value**
Sensitivity	0.583
Specificity	0.801
AUC (estimated)	0.692

4 Classification tree

In this section, we fit a recursive partitioning classification tree, tuning its complexity parameter via cross-validation. We decided to choose the pruning parameter based on sensitivity (to prioritize detecting smokers); though this is not a crucial choice: adjusting the final probability threshold for predictions can alter sensitivity, specificity, or accuracy post-estimation, providing flexibility in balancing false positives and negatives. Cross-validation used the default probability cutoff of 0.5.
Maximizing sensitivity aligns with the clinical requirement to detect as many smokers as possible, even at the expense of more false alarms.

4.1 Decision Tree Classification

Decision tree classification is a supervised, non-parametric method that splits the feature space into regions to assign one of two class labels. Starting with all data at the root, it recursively chooses a feature j and threshold s to partition a region into two (those with $x_j \leq s$ and $x_j > s$), selecting each split by maximizing the reduction in impurity (e.g., Gini impurity or entropy). Once splitting stops, each leaf's class-probability estimate is the proportion of training points of each class in that leaf; new points are classified according to the leaf they fall into. To control overfitting, trees are pruned by increasing a complexity parameter (cp), which penalizes larger trees (trading flexibility for lower variance).

4.2 Hyperparameter Grid Search

We searched over complexity parameter (`cp`) values in the grid:

$$cp \in \{0.0001, 0.00015, \ldots, 0.005\}.$$

This grid balances fine resolution at low `cp` with computational tractability.

The `cp` acts as a regularization term in the recursive partitioning model, penalizing the complexity of the tree to prevent overfitting. It determines the minimum reduction in impurity required for a split to be made. In class, we referred to this parameter as α, but we called it `cp` according to the notation in the *rpart* library, that was used for fitting the model. A smaller `cp` allows the tree to grow deeper, potentially capturing more intricate patterns, while a larger `cp` results in simpler, pruned trees.

4.3 Other choices

We used the Gini index to determine the optimal splits during tree construction. Other regularization parameters, such as `minsplit`, `minbucket`, and `maxdepth`, were left at their default values and were not as important as `cp` in controlling the tree growth and model performance.

4.4 Results of Cross-Validation

A 5-fold cross-validation was performed to choose the `cp` that maximizes the sensitivity. Figure 5 shows the sensitivity, specificity, and accuracy as functions of `cp`. A dashed vertical line marks the chosen `cp`(0.00365).

As shown in Figure 5, optimizing for sensitivity has little negative impact on specificity. Conversely, if we hypothetically maximize for specificity, sensitivity tends to decrease more substantially.

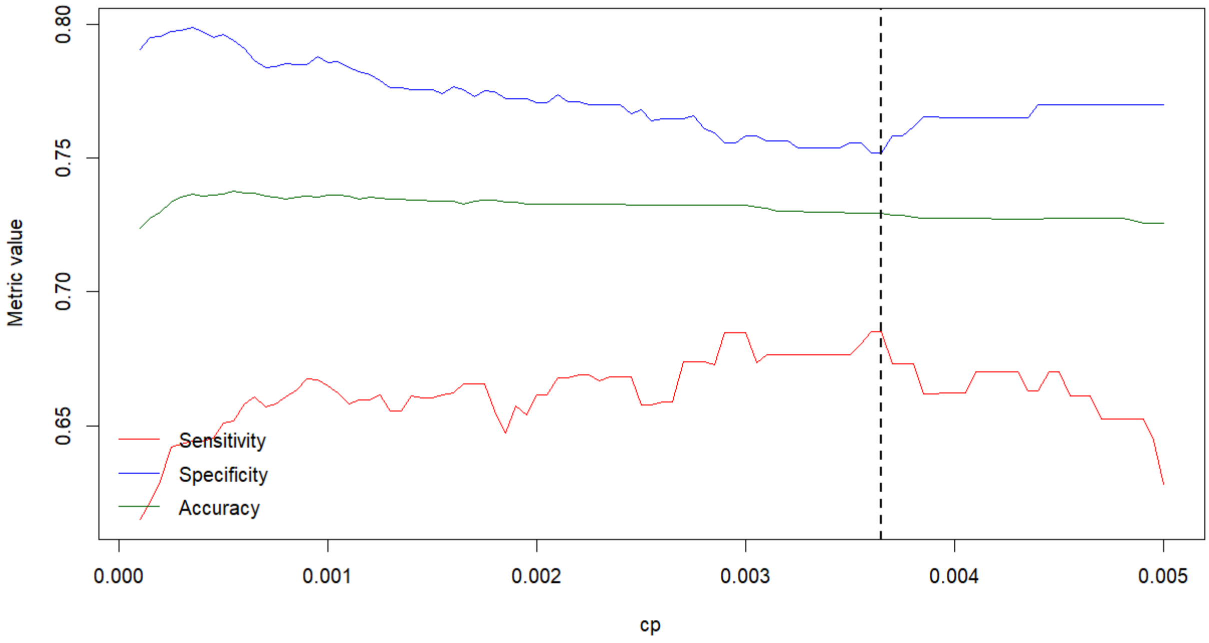

Figure 5: Sensitivity, Specificity, and Accuracy vs. Complexity Parameter during cross-validation

4.5 Final Model and Test-Set Evaluation

We re-fit the `rpart` model on the full training data using `cp=0.00365`. Predictions on the completely held-out test set yielded the confusion matrix in Table 11.

Additionally, for threshold-independent considerations, we computed the ROC curve and area under the curve (AUC = 0.7744), shown in Figure 6.

For simplicity, the confusion matrix was computed with the default threshold; the specificity is substantially bigger: we obtain a sensitivity of 68.23% and specificity of 74.02% on test data. Overall accuracy is 71.92%, and the AUC of 0.7745 indicates decent discriminative ability.

5 Random Forest

Random forests improve single decision trees by averaging an ensemble of T unpruned trees. Two key mechanisms introduce diversity among trees:

- **Bootstrap sampling:** each tree is built on a bootstrap sample of the training data.
- **Feature subsetting:** at each split, only a random subset of m features (out of d) is considered.

In terms of bias, each tree is grown deep (unpruned), yielding low bias at the individual-tree level. Decision trees tend to produce high-variance predictions, so averaging many trees yields substantial benefits. Since

Table 11: Confusion Matrix and Key Performance Metrics on Test Set for the tree (Positive = yes)

	Reference = yes	Reference = no
Prediction = yes	2852	1903
Prediction = no	1328	5422
Performance Metrics		
Accuracy	0.7192	
Sensitivity	0.6823	
Specificity	0.7402	

each tree in the ensemble is drawn from the same distribution, the expected value of their average prediction matches the expectation of any single tree, leaving bias unchanged. Therefore, the method's effectiveness lies entirely in its ability to reduce variance while maintaining the original bias. However, reducing the number of features m considered at each split can slightly increase each tree's bias by limiting split options. Overall, random forests maintain low bias while dramatically reducing variance:

$$\mathrm{Var}_{\mathrm{RF}} = \rho\,\sigma^2 \;+\; \frac{1-\rho}{T}\,\sigma^2,$$

where σ^2 is the variance of a single tree's prediction and ρ is the average correlation between tree outputs.

Predictions of different trees are aggregated by majority vote:

$$\hat{p}_{\mathrm{RF}}(\mathbf{x}) = \arg\max_{k} \sum_{t=1}^{T} \mathbf{1}\big(\hat{p}_t(\mathbf{x}) = k\big),$$

where each tree t casts one vote for its predicted class $\hat{p}_t(\mathbf{x})$ ($k \in \{0, 1\}$ in our case).
For more details, see Breiman (2001).

5.1 Implementation and results

We trained a random forest with 1000 trees with the *randomForest* package, using the default feature subset size $m = \lfloor\sqrt{p}\rfloor$ and growing each tree to purity (minimum node size of one).

For thresholding, we applied the default probability cutoff of 0.5 to assign class labels.

The confusion matrix for the Random Forest model on the test set is shown in Table 12.

Table 12: Confusion Matrix and Key Performance Metrics on Test Set for the random forest(Positive = yes)

	Reference = yes	Reference = no
Prediction = yes	3053	1180
Prediction = no	1127	6145
Performance Metrics		
Accuracy	0.7995	
Sensitivity	0.7304	
Specificity	0.8389	

Random Forest outperforms the single decision tree across all metrics, achieving an accuracy of 79.95%, sensitivity of 73.04%, and specificity of 83.89%. Remarkably, the random forest achieves higher sensitivity than any other method. Moreover, regardless of where you set the decision threshold, its ROC curve and its AUC clearly demonstrate a superior sensitivity–specificity trade-off compared to alternative approaches.Additionally, the area under the ROC curve (AUC) for the Random Forest is 0.883, as shown in Figure 6. This AUC indicates excellent discriminative ability and substantial improvement over the single tree (AUC = 0.774).

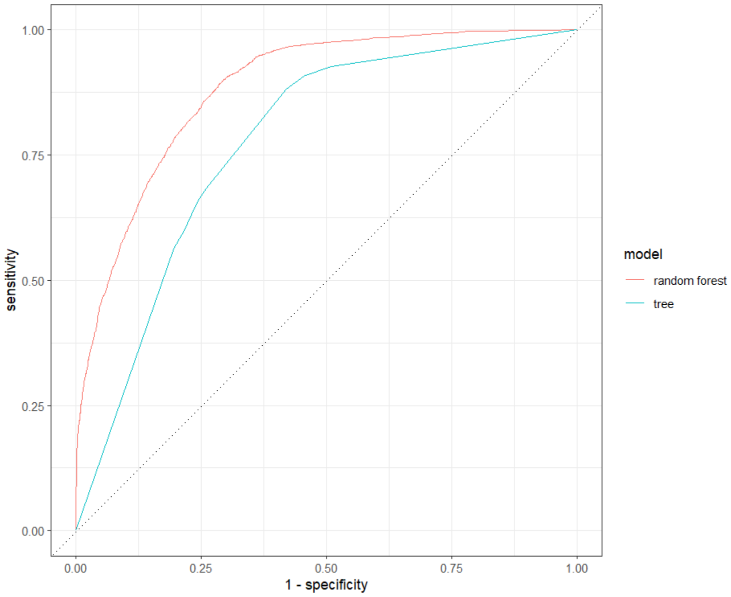

Figure 6: ROC Curve for Final Tree on Test Data

While the computational cost of training and predicting with Random Forests is higher than that of a single tree, the improvement in predictive performance justifies this additional complexity.

References

Breiman L. (2001). Random forests. *Machine learning*, **45**, 5–32.