Advanced Statistics for Machine Learning

Exam 2024

Samd Guizani

(A23 – SPOC)

Table of contents

A preliminary note on R script 3

I. Exercise 1 4

1. Question 1 4

2. Question 2 6

a. Model involving all the explanatory variables 6

b. Model with constraint 7

c. Models' comparison 8

II. Exercise 2 10

1. Question 1 10

a. Dataset description and preprocessing 10

b. Exploratory data analysis 10

c. Candidate models development 11

d. Model performance evaluation and comparison 15

e. Conclusion 16

2. Question 2 17

a. Dataset description and preprocessing 17

b. Exploratory data analysis 18

c. Candidate models development 20

d. Model performance evaluation and comparison 23

e. Conclusion 25

III. References 26

A PRELIMINARY NOTE ON R SCRIPT

The R script used to solve the exercises is provided. It is divided into 3 sections covering Exercise 1, Exercise 2 question 1 and Exercise 2 question 2. The execution time of the 2 first sections is quick and can easily be rerun. However, the 3rd section (Exercise 2, question 2) requires several minutes to be executed. Hence, the output models as well as the datasets have been saved and provided in a separate folder (Outputs), should the reader be interested in exploring details or replicating some part of this work.

I. EXERCISE 1

1. QUESTION 1

This problem is a constrained least square optimization of a multiple linear regression (MLR). So, the aim is to find a vector of coefficients $\hat{\beta}_c$ such that:

$$\hat{\beta}_c = \min_{F(\beta)=0} J(\beta) = \min_{R.\beta - r = 0} \frac{1}{2} \|Y - X\beta\|^2$$

N.B.: (1/2) coefficient in J(β) is introduced to avoid carrying multiplication by 2 when later taking the derivative.

Where *R* is a q x (p+1) matrix and *r* is a vector of size q, q being the number of constraints and also the rank of *R*. So, *R* and *r* define a set of q linear constraints on the coefficients of the vector *β* (*R* matrix holds the weights on the constrained coefficients and *r* vector holds the weighted sum to reach).

First, let us check the existence and uniqueness of a solution:

- The set $K = \{\beta \in \mathbb{R}^{p+1} | R.\beta = r\}$ is a **closed set** as it is defined by equality constraints.
- ***J(β)* is continuous** and ***α*-convex** as it is a squared norm.

We conclude that there exists at least a minimum of *J* on *K*.

Moreover:

- *K* is a **convex set**, indeed considering 2 points β_1 and β_2 in *K* i.e. $R.\beta_1 = R.\beta_2 = r$ and considering any point $\beta_\theta = \theta\beta_1 + (1-\theta)\beta_2, \theta \in [0,1]$, we get:
  $R.\beta_\theta = R(\theta\beta_1 + (1-\theta)\beta_2) = \theta R.\beta_1 + (1-\theta)R.\beta_2 = \theta r + (1-\theta)r = r$
  So, β_θ belongs to *K*.
- ***J(β)* is strictly convex** (being *α*-convex)

We conclude that there exists at most one minimum of *J* on *K*.

To find the minimum, since J and the linear constraints are differentiable and the constraints are regular, we use the Lagrange multipliers to find the minimum of *J*:

$$\mathcal{L}(\beta, \lambda) = J(\beta) + \langle \lambda, F(\beta) \rangle \; with \; F(\beta) = R.\beta - r$$

Where λ is a vector of size q.

The derivatives of *J* and *F* are:

$$\nabla J(\beta) = (X'X).\beta - X'Y$$

$$\nabla F(\beta) = R'$$

Then:

$$\nabla \mathcal{L}(\beta, \lambda) = \nabla J(\beta) + \nabla F(\beta).\lambda = (X'X).\beta - X'Y + R'.\lambda$$

To minimize *J(β)*, we must find the critical point of the Lagrangian, such that:

$$\begin{cases} \nabla \mathcal{L}(\hat{\beta}_c, \lambda) = (X'X).\hat{\beta}_c - X'Y + R'.\lambda = 0 \\ F(\hat{\beta}_c) = R\hat{\beta}_c - r = 0 \end{cases}$$

From the 1st equation, we deduce an expression of $\hat{\beta}_c$:

$$\begin{aligned} &(X'X).\hat{\beta}_c - X'Y + R'.\lambda = 0 \\ &\Leftrightarrow (X'X).\hat{\beta}_c = X'Y - R'.\lambda \\ &Assuming \; (X'X) \; is \; invertible: \\ &\Leftrightarrow \hat{\beta}_c = (X'X)^{-1}X'Y - (X'X)^{-1}R'.\lambda \\ &\Leftrightarrow \hat{\beta}_c = \hat{\beta} - (X'X)^{-1}R'.\lambda \end{aligned}$$

$\hat{\beta} = (X'X)^{-1}X'Y$ being the unconstrained solution of the minimization problem.

Substituting $\hat{\beta}_c$ in the 2nd equation, we deduce an expression of λ:

$$\begin{aligned} &R\hat{\beta}_c - r = 0 \\ &\Leftrightarrow R\left[\hat{\beta} - (X'X)^{-1}R'.\lambda\right] = r \end{aligned}$$

$$\Leftrightarrow R(X'X)^{-1}R'.\lambda = R\hat{\beta} - r$$
$$\Leftrightarrow \lambda = (R(X'X)^{-1}R')^{-1}(R\hat{\beta} - r)$$

Finally, injecting the expression of λ in the expression of $\hat{\beta}_c$, we obtain:

$$\hat{\beta}_c = \hat{\beta} + (X'X)^{-1}R'.(R(X'X)^{-1}R')^{-1}(r - R\hat{\beta})$$

2.QUESTION 2

In the Ozone dataset, two variables are categorical: *pluie* (with levels “Pluie” and “Sec”) and *vent* (with levels "Est," “Nord,” “Ouest,” and “Sud”). To enable comparison between the standard R lm() function and our multiple linear regression implementation, categorical variables were preprocessed using one-hot encoding, converting each variable to L-1 binary variables (where L is the number of original levels). The first six observations are shown below, before and after encoding.

Before pre-processing:

	maxO3	T9	T12	T15	Ne9	Ne12	Ne15	Vx9	Vx12	Vx15	maxO3v	vent	pluie
601	87	15.6	18.5	18.4	4	4	8	0.6946	-1.7101	-0.6946	84	Nord	Sec
602	82	17.0	18.4	17.7	5	5	7	-4.3301	-4.0000	-3.0000	87	Nord	Sec
603	92	15.3	17.6	19.5	2	5	4	2.9544	1.8794	0.5209	82	Est	Sec
604	114	16.2	19.7	22.5	1	1	0	0.9848	0.3473	-0.1736	92	Nord	Sec
605	94	17.4	20.5	20.4	8	8	7	-0.5000	-2.9544	-4.3301	114	Ouest	Sec
606	80	17.7	19.8	18.3	6	6	7	-5.6382	-5.0000	-6.0000	94	Ouest	Pluie

After pre-processing:

	(Intercept)	T9	T12	T15	Ne9	Ne12	Ne15	Vx9	Vx12	Vx15	maxO3v	ventNord	ventOuest	ventSud	pluieSec
601	1	15.6	18.5	18.4	4	4	8	0.6946	-1.7101	-0.6946	84	1	0	0	1
602	1	17.0	18.4	17.7	5	5	7	-4.3301	-4.0000	-3.0000	87	1	0	0	1
603	1	15.3	17.6	19.5	2	5	4	2.9544	1.8794	0.5209	82	0	0	0	1
604	1	16.2	19.7	22.5	1	1	0	0.9848	0.3473	-0.1736	92	1	0	0	1
605	1	17.4	20.5	20.4	8	8	7	-0.5000	-2.9544	-4.3301	114	0	1	0	1
606	1	17.7	19.8	18.3	6	6	7	-5.6382	-5.0000	-6.0000	94	0	1	0	0

a. Model involving all the explanatory variables

Applying R software lm() function, we obtain a model predicting maxO3 from all the explanatory variables. The summary is:

```
> LM = lm(formula = maxO3 ~ ., data = Dataset_ozone)
> summary(LM)

Call:
lm(formula = maxO3 ~ ., data = Dataset_ozone)

Residuals:
    Min      1Q  Median      3Q     Max
-51.814  -8.695  -1.020   7.891  40.046

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 16.26536   15.94398   1.020   0.3102
T9           0.03917    1.16496   0.034   0.9732
T12          1.97257    1.47570   1.337   0.1844
T15          0.45031    1.18707   0.379   0.7053
Ne9         -2.10975    0.95985  -2.198   0.0303 *
Ne12        -0.60559    1.42634  -0.425   0.6721
Ne15        -0.01718    1.03589  -0.017   0.9868
Vx9          0.48261    0.98762   0.489   0.6262
Vx12         0.51379    1.24717   0.412   0.6813
Vx15         0.72662    0.95198   0.763   0.4471
maxO3v       0.34438    0.06699   5.141 1.42e-06 ***
ventNord     0.53956    6.69459   0.081   0.9359
ventOuest    5.53632    8.24792   0.671   0.5037
ventSud      5.42028    7.16180   0.757   0.4510
pluieSec     3.24713    3.48251   0.932   0.3534
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 14.51 on 97 degrees of freedom
Multiple R-squared:  0.7686,      Adjusted R-squared:  0.7352
F-statistic: 23.01 on 14 and 97 DF,  p-value: < 2.2e-16
```

We can calculate $\hat{\beta}$ (with their 95% confidence intervals), the unconstrained least square solution, as following:

```
> hbeta = solve(t(X) %*% X) %*% t(X) %*% Y

> print(hbeta_ci)
            Est. Coef.  Std. Error Lower Bound Upper Bound
(Intercept) 16.26535597 15.94398012 -15.3790310  47.9097430
T9           0.03916979  1.16495679  -2.2729470   2.3512865
T12          1.97257424  1.47570493  -0.9562915   4.9014400
T15          0.45030800  1.18707252  -1.9057023   2.8063183
Ne9         -2.10975486  0.95985471  -4.0148008  -0.2047090
Ne12        -0.60559218  1.42633808  -3.4364784   2.2252941
Ne15        -0.01717804  1.03589370  -2.0731403   2.0387842
Vx9          0.48260889  0.98762405  -1.4775515   2.4427692
Vx12         0.51379495  1.24716744  -1.9614872   2.9890771
Vx15         0.72662334  0.95197628  -1.1627861   2.6160327
maxO3v       0.34437835  0.06699148   0.2114188   0.4773379
ventNord     0.53956395  6.69459345 -12.7473509  13.8264788
ventOuest    5.53631722  8.24792304 -10.8335269  21.9061613
ventSud      5.42028442  7.16180048  -8.7939071  19.6344759
pluieSec     3.24713025  3.48251475  -3.6646975  10.1589580
```

We confirm that the calculated coefficients are identical to lm() function results.

b. Model with constraint

If we apply a constraint on the variables T9, T12 and T15 such that $\beta_{T9} + \beta_{T12} + \beta_{T15} = 0$, we obtain the following $\hat{\beta}_c$ solution:

```
> hbeta_c =
+   hbeta +
+   solve(t(X) %*% X) %*%
+   t(R) %*%
+   solve(R %*% solve(t(X) %*% X) %*% t(R)) %*%
+   (r - R %*% hbeta)

> print(hbeta_c_ci)
             Est. Coef. Std. Error Lower Bound Upper Bound
(Intercept) 59.42206714  17.016295  25.6494312  93.1947031
T9          -1.87215990   1.243306  -4.3397785   0.5954587
T12          1.17078960   1.574954  -1.9550576   4.2966368
T15          0.70137031   1.266909  -1.8130939   3.2158346
Ne9         -2.97625939   1.024410  -5.0094295  -0.9430893
Ne12        -1.37664117   1.522267  -4.3979192   1.6446369
Ne15         0.08932548   1.105563  -2.1049109   2.2835618
Vx9         -0.16837184   1.054047  -2.2603631   1.9236195
Vx12         0.53454939   1.331046  -2.1072083   3.1763071
Vx15         0.77110593   1.016002  -1.2453760   2.7875879
maxO3v       0.46895526   0.071497   0.3270535   0.6108570
ventNord    -5.14140434   7.144840 -19.3219330   9.0391244
ventOuest    3.94210314   8.802639 -13.5286978  21.4129040
ventSud      6.50590711   7.643469  -8.6642624  21.6760766
pluieSec     5.28638433   3.716732  -2.0902997  12.6630683
```

c. Models' comparison

The 2 models can be compared using statistical metrics as presented in the following table:

	Unconstrained MLR	**Constrained MLR**
R^2	0.7686	0.7364
Adjusted R^2	0.7352	0.6983
Residuals Standard Error	14.51	15.48

Table 1 – Comparison of unconstrained vs. constrained MLR performance

As expected, the constrained model shows lower R^2/Adjusted R^2 and higher Residual Standard Error. The unconstrained model minimizes residuals, producing coefficients $\hat{\beta}$ that fit the observed data as closely as possible, resulting in the lowest Residual Standard Error and highest R^2/Adjusted R^2. In contrast, constraining coefficients for T9, T12, and T15 produces a set $\hat{\beta}_c$ with predictions further from observed data, leading to higher Residual Standard Error and lower R^2/Adjusted R^2.

On the following figure, the 2 models can be graphically compared by plotting the predicted response of each model against the actual response.

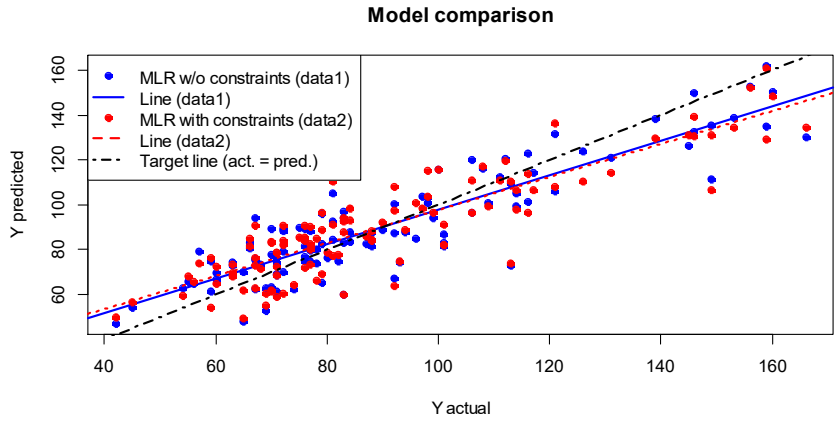

Figure 1 – Comparative plot of constrained vs. unconstrained MLR models

It is to be mentioned that the constrained and unconstrained models are very close, despite the restrictions applied to the coefficients of the variables T9, T12 and T15. This may be explained by the correlations that exist among the explanatory variables which can be evaluated through the correlation matrix of the explanatory variables:

```
> # Correlation matrix of explanatory variables
> print(round(cor(X[,(1:p)+1]), 2))
             T9   T12   T15   Ne9  Ne12  Ne15   Vx9  Vx12  Vx15 maxO3v ventNord ventOuest ventSud pluieSec
T9         1.00  0.88  0.85 -0.48 -0.47 -0.33  0.25  0.22  0.17   0.58    -0.19     -0.09     0.26     0.38
T12        0.88  1.00  0.95 -0.58 -0.66 -0.46  0.43  0.31  0.27   0.56    -0.22     -0.10     0.26     0.44
T15        0.85  0.95  1.00 -0.59 -0.65 -0.57  0.45  0.34  0.29   0.57    -0.20     -0.09     0.23     0.42
Ne9       -0.48 -0.58 -0.59  1.00  0.79  0.55 -0.50 -0.53 -0.49  -0.28    -0.11      0.32    -0.08    -0.39
Ne12      -0.47 -0.66 -0.65  0.79  1.00  0.71 -0.49 -0.51 -0.43  -0.36    -0.16      0.37    -0.11    -0.42
Ne15      -0.33 -0.46 -0.57  0.55  0.71  1.00 -0.40 -0.43 -0.38  -0.31    -0.18      0.29     0.02    -0.29
Vx9        0.25  0.43  0.45 -0.50 -0.49 -0.40  1.00  0.75  0.68   0.34    -0.02     -0.39     0.20     0.42
Vx12       0.22  0.31  0.34 -0.53 -0.51 -0.43  0.75  1.00  0.84   0.22     0.22     -0.64     0.12     0.30
Vx15       0.17  0.27  0.29 -0.49 -0.43 -0.38  0.68  0.84  1.00   0.19     0.20     -0.53     0.04     0.21
maxO3v     0.58  0.56  0.57 -0.28 -0.36 -0.31  0.34  0.22  0.19   1.00    -0.01     -0.06     0.12     0.38
ventNord  -0.19 -0.22 -0.20 -0.11 -0.16 -0.18 -0.02  0.22  0.20  -0.01     1.00     -0.56    -0.30     0.08
ventOuest -0.09 -0.10 -0.09  0.32  0.37  0.29 -0.39 -0.64 -0.53  -0.06    -0.56      1.00    -0.43    -0.25
ventSud    0.26  0.26  0.23 -0.08 -0.11  0.02  0.20  0.12  0.04   0.12    -0.30     -0.43     1.00     0.14
pluieSec   0.38  0.44  0.42 -0.39 -0.42 -0.29  0.42  0.30  0.21   0.38     0.08     -0.25     0.14     1.00
```

Noticeably, variables T9, T12 and T15 are correlated negatively with Ne9, Ne12 and Ne15 and positively with maxO3v. Hence, the constraint on the coefficients of T9, T12 and T15 get “distributed” on other variables. This finally leads to a different model (refer to section 2a and 2b where the coefficients of the 2 models are reported) which overall is closely comparable to the unconstrained solution.

II. EXERCISE 2

Two datasets are provided, each with one response variable and multiple explanatory variables. The goal is to build and compare models for each dataset to predict the response variable and select the best one.

The approach is as follows:

- Step 1: Split each dataset into training and test sets.
- Step 2: Perform exploratory data analysis on the training set.
- Step 3: Develop and fine-tune models using the training set, creating a list of candidates.
- Step 4: Evaluate candidate models on the test set using statistical metrics to identify the best model.

1. QUESTION 1

a. Dataset description and preprocessing

The dataset has 77 observations, 200 numeric explanatory variables and a binary categorical response variable (coded -1/1). Hence the goal is to develop a classification model.

No preprocessing is applied to the data. The observations have been randomly split between a training set (70% of the observations) and a test set (30% of the observations).

b. Exploratory data analysis

Linear correlation coefficients of each explanatory variable against the response variable (transformed to a numeric value -1 or +1) have been computed and plotted on a bar chart (Figure 2). We notice that the 6 first variables are the most correlated to the response variable. Hence, the developed models will integrate variable selection strategies to obtain sparse, easy to explain and robust-to-noise models.

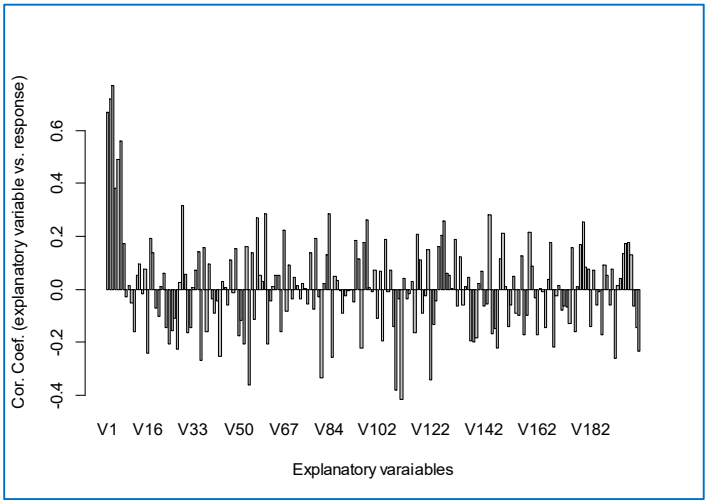

Figure 2 – Correlation coefficients of explanatory variables vs. response variable

c. Candidate models development

Several models' options have been considered:

- Generalized Linear Model, associated with penalty approach to perform variable selection
- CART decision trees, associated with pruning to achieve variable selection
- Random Forest
- Random Forest for variable selection, followed by CART tree fitting on the selected variables

Generalized Linear Regression (using R package "glmnet")

The response is a binary categorical variable. Therefore, using Multiple Linear Regression, which considers a numeric continuous response variable, is not appropriate. A more suited approach is to use Generalized Linear Models and specifically a binary **Logistic Regression** [1] that models the probability of an observation being in class (+1) as:

$$p(Y = +1|X = x) = \frac{e^{\beta_0 + \beta.x}}{1 + e^{\beta_0 + \beta.x}}$$

An alternative writing is called "log-odds":

$$log\frac{p(Y=+1|X=x)}{p(Y=-1|X=x)} = \beta_0 + \beta.x$$

The objective is to find set of coefficients (β_0, β) that minimizes the classification error of training set observations.

When the number of variables exceeds the number of observations, fitting a reliable model can be very difficult and unreliable. Therefore, a regularized approach, using LASSO, has been chosen to develop a sparse model. The tuning of the hyperparameter λ has been done through cross-validation. The value that minimizes cross-validation error is 0.09965 and yielded the selection of 5 explanatory variables (Figure 3), which are V1, V2, V3, V5 and V6. These variables are among the most correlated with the response variable.

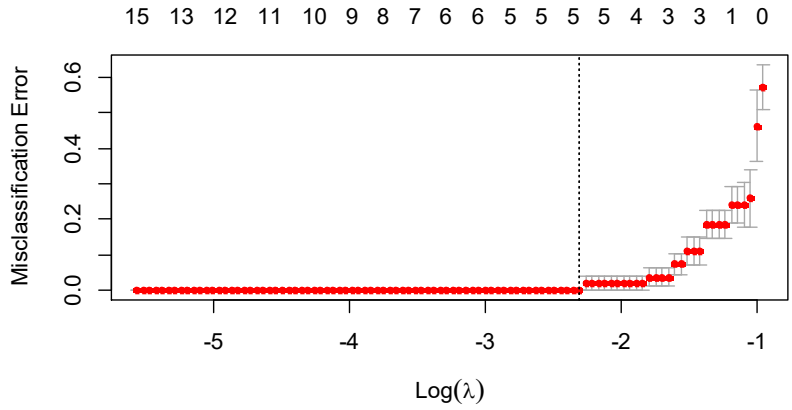

Figure 3 – Logistic Regression misclassification error as function of λ hyperparameter, using a LASSO regularization

CART decision tree (using R package "rpart")

The principle is to build a maximal decision tree, for which the leaves will be pure and then prune it thanks to tuning the hyperparameter cp. The tuning is performed by cross-validation (the tuned value is 0.0769).

CART allows also to provide a ranking of the explanatory variables' importance which yielded the top 4 variables V2, V3, V1 and V6. The decision tree can also be plotted to explain how new observations would be classified (Figure 4).

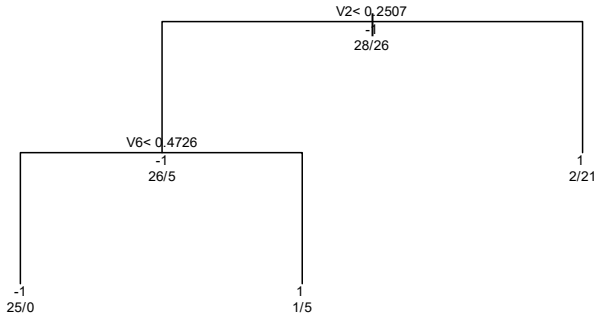

Figure 4 – CART tree for classification

We can note that V1 and V3 are not actually represented on the tree. The reason is that these variables are used in the surrogate splits, which makes it possible to predict the class of an observation for which V2 or V6 values would be missing.

Random forest (using R package "randomForest")

Random Forrest builds a large number of decision trees. Each tree is trained on a randomly selected set of observations, taken with replacement from the training set. In addition, only a randomly selected subset of the variables contributes to developing each tree. This approach increases the variety of the trees which improves the reliability and robustness of the predicted class.

In this case, 500 trees have been developed, each of them using 14 variables. The out-of-bag error (i.e. comparing the actual class and the predicted class of observations not used to train a given tree) is 7.41%. A confusion matrix can also be calculated, showing the split of misclassification in term of "False Negative" and "False Positive":

```
Confusion matrix:
   -1  1 class.error
-1 28  0   0.0000000
1   4 22   0.1538462
```

Random Forest also provides a ranking of the variable importance which can be plotted as follow (Figure 5). Here again, we observe that the top 4 variables V3, V1, V2 and V6 are among the most correlated to the response variable.

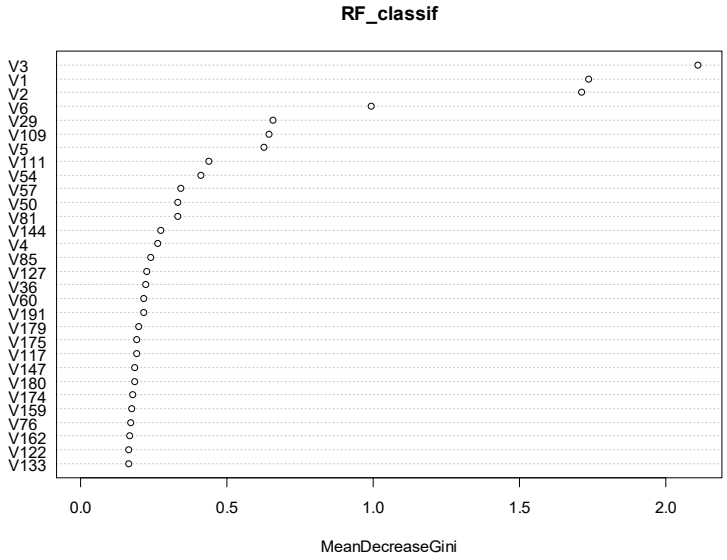

Figure 5 – Random Forest variable importance plot

Variable selection by Random Forest and CART modelling on selected variables (using R package "VSURF")

One drawback of Random Forest is their lack of explainability. Hence, an alternative approach is to use Random Forest for variable selection and then build a CART model using only the selected variables. This variable selection strategy through Random Forest is implemented in "VSURF" R package and relies on 3 steps:

- Elimination: aiming to reduce the number of variable thanks to thresholds established based on the mean and standard deviation of variable importance scores.
- Interpretation: aiming to further reduce the number of variables to facilitate interpretation of the link between explanatory and response variables.
- Prediction: aiming to reduce the number of variables to the minimum required to achieve a reliable prediction.

In this case (Figure 6), 20 variables have been selected after Elimination step, which have been reduced to 5 at the Interpretation step (V3, V2, V1, V6, V5) and to 4 after the Prediction step (V3, V2, V1, V5).

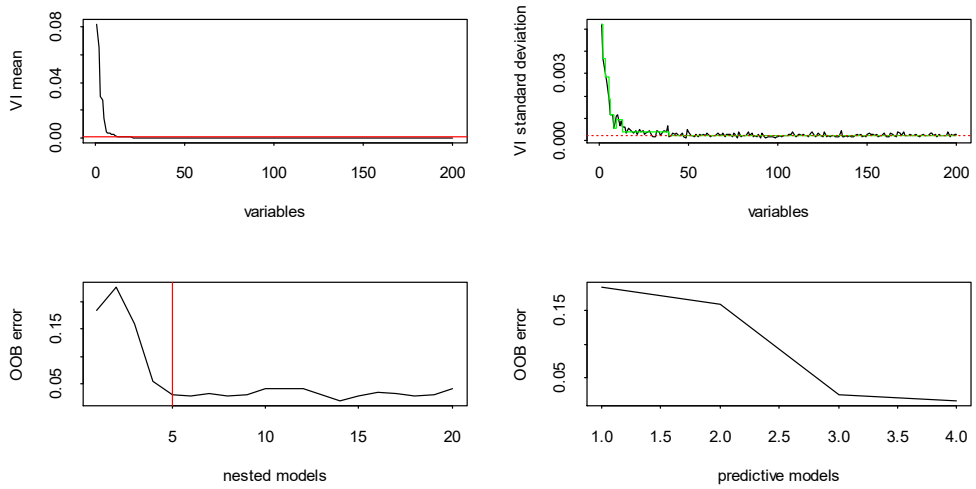

Figure 6 – Output of VSURF variable selection (5 variables kept after Interpretation step, 4 variables kept after Prediction step)

Using the Interpretation or the Prediction steps' selected variables, 2 CART trees can be developed (Figure 7):

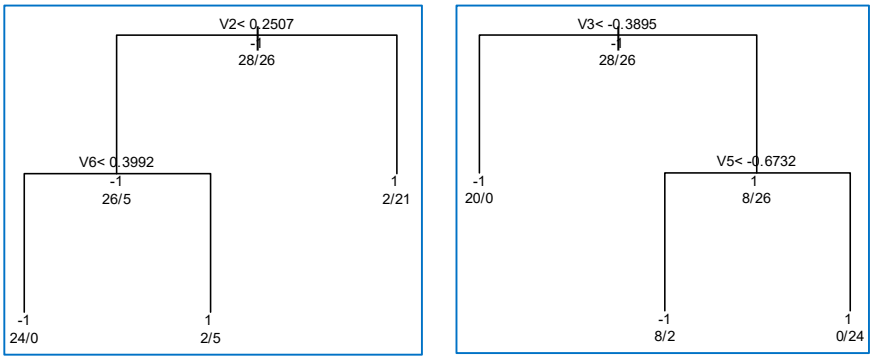

Figure 7 – CART trees based on Random Forest variable selection (left: post-Interpretation step, right: port-prediction step)

It can be noted that the tree developed based on the variables selected at Interpretation step is essentially the same as the one developed through CART (after applying cross-validation pruning) while the tree based on Prediction step variables uses the surrogate split variables.

d. Model performance evaluation and comparison

The performance of the candidate models is assessed on the test set observations, using 2 metrics:

- The test set **classification error** = fraction of misclassified observations in the test set.
- The test set **confusion matrix** [2], which shows, for each class, the count of correctly and wrongly classified observations.

The following table summarizes the performance measured for the candidate models.

Model	Tuned hyperparameters	Test set classification error (%)	Test set confusion matrix
Logistic regression	λ= 0.09965	0	Reference Prediction -1 1 -1 12 0 1 0 11
CART (with pruning)	cp = 0.07692	8.7	Reference Prediction -1 1 -1 10 0 1 2 11
Random Forest	n trees = 500 n variables/tree = 14	8.7	Reference Prediction -1 1 -1 11 1 1 1 10
VSURF + CART (Interpretation step)	---	8.7	Reference Prediction -1 1 -1 10 0 1 2 11
VSURF + CART (Prediction step)	---	4.3	Reference Prediction -1 1 -1 12 1 1 0 10

Table 2 – Performance metrics of classification models on test set

e. Conclusion

All the tested classification algorithms selected a sparse subset of explanatory variables required for predicting the class of an observation. These variables are among the most correlated to the response variable, as observed through the exploratory data analysis.

However, they differ in terms of performance. Based on their evaluation on a test set, the best model is the Logistic Regression which has 0 prediction error. The second best is the model combining variable selection through Random Forest (VSURF prediction variables) followed by developing a CART decision tree. It has a 4.3% error

rate on the test set, with only 1 misclassified observation. The rest of the models showed an error rate of 8.7% with slight variations in the misclassified observations.

2.QUESTION 2

In this question the response is a numeric continuous variable. Hence, the objective is to develop regression models.

Two modelling strategies have been attempted. As presented in section Dataset description and preprocessing, 6 dataframes are provided, containing the same variables collected from 6 different geographical locations:

- **Approach 1** consists in merging the 6 dataframes and developing common models. To keep the information about stations, a categorical variable 'station' has therefore been created in the combined dataset. However, this approach encounters difficulties with missing explanatory variables on 2 of the locations.
- **Approach 2** consists in developing local models for each of the locations, considering only the variables available at this location. The inconvenience of this approach is that it lacks generalization and robustness, expected when using larger datasets.

a. Dataset description and preprocessing

PM10 dataset is available in VSURF R package. A description of its content can be found in VSURF documentation [3]. It is a collection of 6 dataframes corresponding to pollution records from 6 stations located in Normandy (France), identified as "jus", "gui", "gcm", "rep", "hri" and "ail".

The variables are:

- PM10 Daily concentration of PM10, in µg/m3
- NO, NO2, SO2 Daily mean concentration of NO, NO2 , SO2, in µg/m3
- T.min, T.max, T.moy Daily minimum, maximum and mean temperature, in degree Celsius

- DV.maxvv, DV.dom Daily maximum speed and dominant wind direction (0 degree is north)
- VV.max, VV.moy Daily maximum and mean wind speed, in m/s
- PL.som Daily rainfall, in mm
- HR.min, HR.max, HR.moy Daily minimum, maximum and mean relative humidity, in %
- PA.moy Daily mean air pressure, in hPa
- GTrouen, GTlehavre Daily temperature gradient, in degree Celsius

An important fact to consider is the missing data:

- The 6 dataframes have in total 90 observations with a missing response variable PM10. They have been discarded, as they are randomly occurring and represent a small fraction of the total number of observations.
- Some observations have partly missing values for the explanatory variables. They were kept in the dataset. However, some modelling approaches (e.g. Multiple Linear Regression), discard them while others (e.g. CART) can still use them.
- A more challenging issue lies in the fact that 2 of the stations, “gcm” and “ail”, respectively miss records of NO/NO2 and NO/NO2/SO2. This may hinder the models’ performance for the concerned stations.

No preprocessing was applied to the data before modelling. The data have been split into a training and a testing sets (approximately 30% of the observations). A difference in the way data is split has been implemented for the 2 approaches:

- **Approach 1**, the random split between train and test sets has been stratified by station. The goal is to obtain a comparable number of observations from each station within each set.
- **Approach 2**, a random split is applied to each dataframe.

b. Exploratory data analysis

Exploratory data analysis was performed on the combined dataset with all 6 stations.

To visualize the distribution of PM10 values vs. station, a boxplot has been plotted (Figure 8). The level of PM10 is dependent on the station. This visual observation has been confirmed by an analysis of variance of PM10 vs. station factor. Therefore, the station is likely to be an important variable to predict PM10.

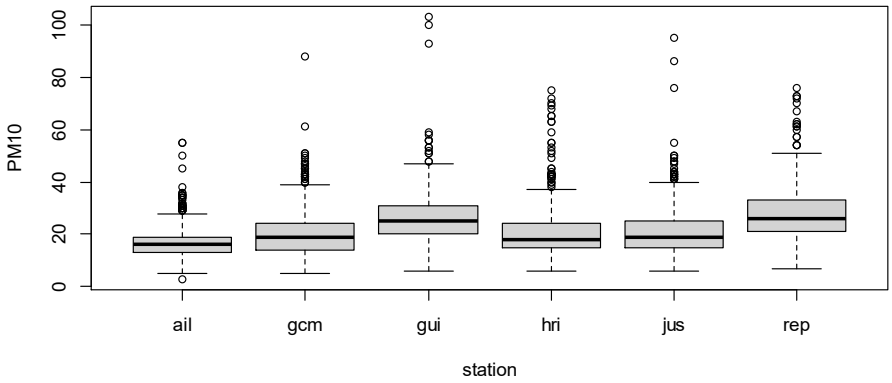

Figure 8 – Boxplot of PM10 vs. station factor

To evaluate the correlation pattern between the numerical variables, a heatmap has been used (Figure 9). Several insights can be mentioned from the correlations study:

- PM10 is positively and highly correlated with NO/NO2/SO2, temperature gradients (GTlehavre and GTrouen) and atmospheric pressure. PM10 is negatively correlated with some wind variables.
- Among the explanatory variables:
    - NO and NO2 are tightly and positively correlated. Same observation can be made about GTlehavre and GTrouen.
    - Temperature variables, relative humidity and wind variables form 3 “clusters” within which the variables are positively correlated.

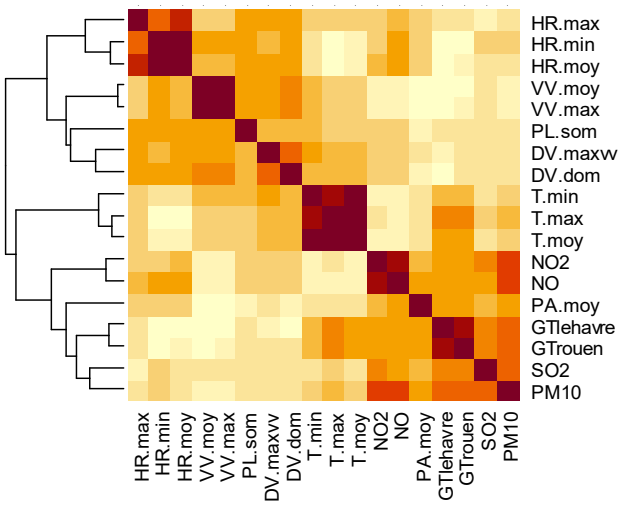

Figure 9 – Correlation heatmap of numeric variables

c. Candidate models development

Approach 1: Combined station data modelling

Linear models

3 linear models have been developed:

1. **Linear model 1**: Multiple linear regression of PM10 vs all the numeric variables. It yields an adjusted R^2 of 0.5783. This model has some limitations. Firstly, due to the absence of measurement of SO2/NO/NO2 for "gcm" and "ail" stations, 1761 observations, more than a third of the training set, have been discarded[1]. Secondly, the residuals do not distribute normally, which can be a problem to interpret the individual predictors' p-values. Logarithm transformation of PM10 improves the distribution of residuals and makes them closer to a Gaussian distribution (Figure 10). However, it was decided to use untransformed response to facilitate comparison with all other models.

[1] N.B.: Although not presented in this document, as an attempt to resolve the issue of missing values for "gcm" and "ail" stations, we also tried to build linear models to predict NO/NO2 and SO2 using the other explanatory variables. These models would be used to impute the missing data. However, the performance of the models was insufficient.

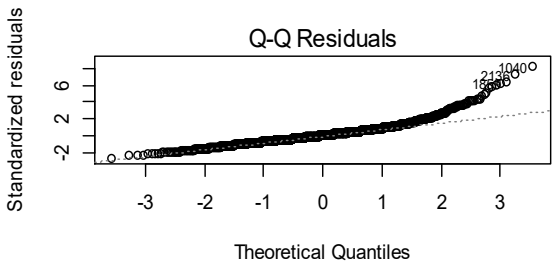

Figure 10 – QQ plot of standardized residuals

2. **Linear model 2**: Multiple linear regression of PM10 vs "station" (as a categorical factor) and all the numeric variables except NO/NO2/SO2 (due to missingness for "gcm" and "ail" stations). In this model, the pair-wise crossed terms for all variables have been included. The adjusted R^2 is 0.5420. The residuals were not normally distributed as well (but the issue can be corrected by transforming the response)
3. **Linear model 3**: The 3rd linear model is obtained by backward reduction of the previous model. Its adjusted R^2 is 0.5446. And here again, residuals were not normally distributed (possibly corrected by a transformation).

The 3 linear models do not seem to provide a good prediction capability. They all have a residual standard error of approximately 6.5 g/m^3. Nevertheless, they will be evaluated using the test set, as a baseline to compare with non-linear models.

Non-linear models

3 modelling approaches have been developed:

1. **CART decision tree**: maximal tree is first created and then pruned using a cross-validation approach. A main advantage of CART algorithm is that it can handle missing values in the explanatory variables. Moreover, CART reports a variable importance score. In this case, the top 6 variables are NO2, NO, station, SO2, GTrouen and GTlehavre, confirming the observations made in the exploratory data analysis.
2. **Random Forest**: 500 trees, based on 6 variables each have been created using randomly selected observations taken from the training set. Random forest implementation in R requires the data has no missing values, but a very

basic imputation tools is provided (the na.roughfix argument replaces missing values by median or mode). Random Forest can also provide a variable importance score (Figure 11), which confirms the findings from CART variable importance scores.

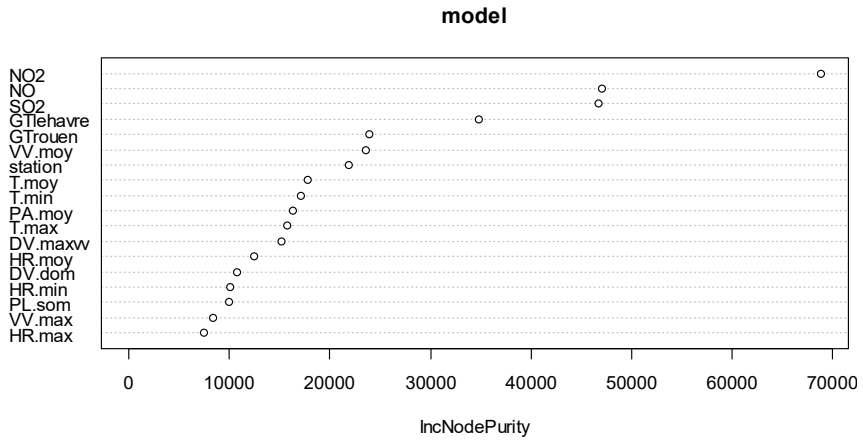

Figure 11 – Random Forest variable importance plot

3. **Random Forest for variable selection + CART on selected variables**: VSURF package was used to achieve a variable selection through Random Forest variable importance scores. After reduction, 15 variables are chosen (including NO2/NO, SO2, GTlehavre/GTrouen and station). Then these variables are used to build a CART decision tree which turned out to be very close to the tree developed using CART algorithm with pruning.

Approach 2: Separate 'station' data modelling

For each station's training set, 3 models were fitted:

1. **Linear model**: including pair-wise crossed terms, followed by backward model reduction.
2. **CART**: pruned through cross-validation.
3. **Random Forest**: since it showed the best performance on stations combined dataset.

The strategy combining Random Forest (variables selection) followed by CART was abandoned as it did not yield a significant reduction in the number of variables for the stations combined dataset.

d. Model performance evaluation and comparison

To compare the performance of the models, the test set predicted values of each model were compared to the actual response using the root mean squared statistics[2]. Also, plots of actual vs. predicted values have been produced to visualize the goodness of fit.

Approach 1: Combined station data modelling

The following table summarizes the results:

Model	Test set RMSE
Linear model 1	6.59
Linear model 2	6.89
Linear model 3	6.92
CART	6.88
Random Forest (RF)	**5.25**
RF variable selection + CART	6.85

Table 3 – Approach 1, Model performance comparison

The best model in this case is the Random Forest and the plot of test set actual vs. predicted values is shown hereafter. A reasonably good agreement can be expected for PM10 ranging between 10 and 40 µg/m^3 (Figure 12). However, the model will tend to underestimate the high PM10 values (> 50 µg/m^3).

[2] Root mean squared error, $RMSE = \sqrt{\frac{(y_i - \hat{y}_i)^2}{n}}$

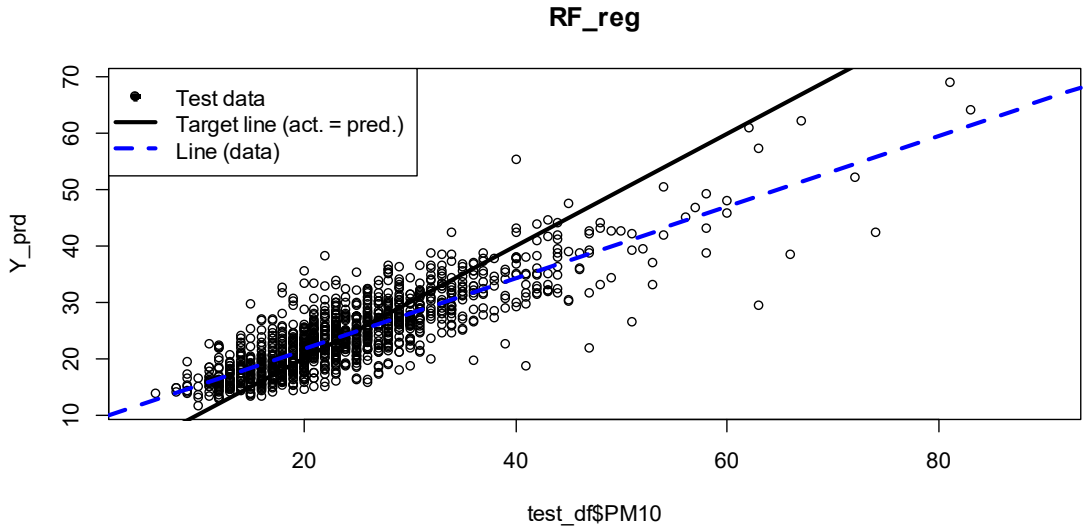

Figure 12 – Approach 1, Random Forest model – Test set actual vs. predicted

Approach 2: Separate station data modelling

The model performance for each station is summarized in the table below.

	Test set RMSE		
Station	**Linear model**	**CART**	**Random Forest**
jus	**5.67**	14.84	5.68
gui	**5.51**	40.05	6.89
gcm	6.72	18.45	**6.24**
rep	6.29	17.84	**5.95**
hri	**7.93***	11.17	**5.94**
ail	5.02	11.11	**4.64**

* This value might be inflated by 1 data point too highly predicted. Hence, Linear model performance would need to be re-evaluated after deciding whether this unusual data point should be removed (based on guidance from a subject matter expert).

Random Forest algorithm is likely to be the best choice for separately modelling each station's data. It is particularly useful for stations missing NO/NO2/SO2 measurements. Linear models could alternatively be used for stations having those measurements available.

e. Conclusion

Regression models can reasonably be developed to predict PM10 based on a set of meteorological and pollution explanatory variables measurements performed at 6 different geographical locations. Two modelling approaches have been developed and they have pros and cons:

- **Approach 1** has the benefit of building a common model, valid across stations. The best performing model is the Random Forest, despite being less explainable than other algorithms (Linear models or CART).
- **Approach 2** builds a separate model for each station, which can be a better fit locally than using a global model. Random Forest models were found to be the best performing but sometimes being closely comparable to simpler and more explainable Linear models.

III. REFERENCES

1. https://glmnet.stanford.edu/articles/glmnet.html#quick-start
2. https://developer.ibm.com/tutorials/awb-confusion-matrix-r/
3. VSURF.pdf

Declaration on use of AI tools: In application of DSTI Assessment Policies, I would like to inform the reader that ChatGPT tool has been occasionally used to support this work. Its use was solely limited to debugging R script errors or suggesting script starters or improvements. No AI tool has been used to develop the mathematical and statistical approaches or the reasoning to solve the exercises.