POLLUANT NO

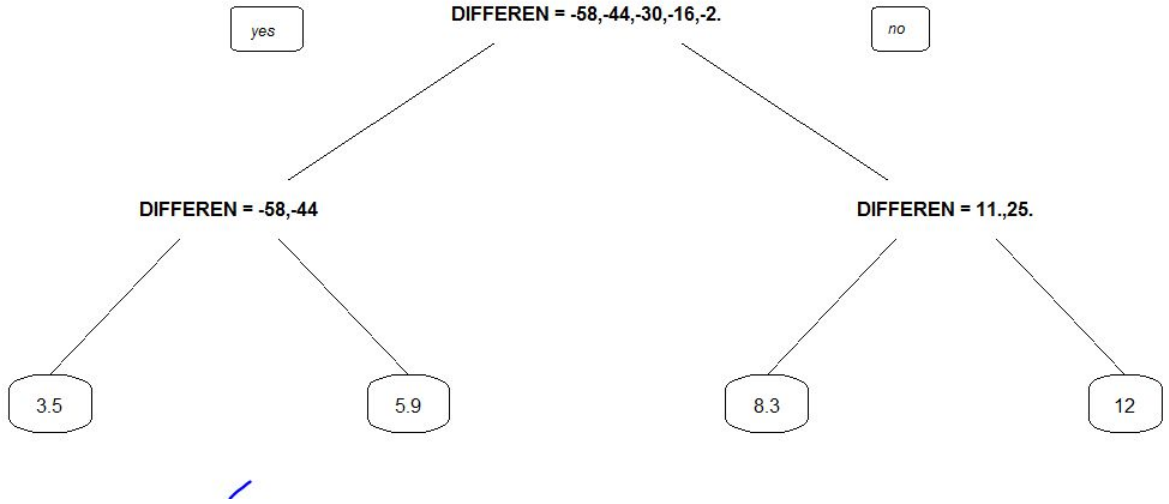
Predictif Number of patients

```
library(rpart)
library(e1071)
library(rpart.plot)
library(caret)
library(Metrics)

#setting the tree control parameters
fitControl <- trainControl(method = "cv", number = 5)
cartGrid <- expand.grid(.cp=(1:50)*0.01)

#decision tree
tree_model <- train(NUMBEROFPATIENTS ~ ., data = dataset, method = "rpart", trControl = fitControl, tuneGrid = cartGrid)
print(tree_model)

main_tree <- rpart(NUMBEROFPATIENTS ~ ., data = dataset, control = rpart.control(cp=0.01))
prp(main_tree)
pre_score <- predict(main_tree, type = "vector")
rmse(newPatientsDataNo$NUMBEROFPATIENTS, pre_score)
```

Pour faire du prédictif, nous avons besoin des packages suivants : rpart, e1071, rpart.plot, caret et Metrics.
La fonction train() définit une grille de paramètres pour un certain nombre de routines de classification et de régression. Elle correspond à chaque modèle et calcule une mesure de performance basée sur le rééchantillonnage. La fonction trainControl() contrôle les nuances de calcul de la fonction train().
La fonction expand.grid crée un cadre de données à partir de toutes les combinaisons des vecteurs ou des facteurs fournis.
Ensuite, nous faisons un modèle d'arbre avec la fonction train() prenant en paramètres NUMBEROFPATIENTS (c'est la valeur dont nous voulons faire la prédiction) et le data : dataset. Ce dernier contient ici le jeu de données newPatientsDataNO (fusion des données des patients atteints de dyspnée et du polluant NO).

```
> print(tree_model)
CART

51 samples
10 predictors

No pre-processing
Resampling: Cross-Validated (5 fold)
Summary of sample sizes: 41, 41, 41, 41, 40
Resampling results across tuning parameters:

  cp    RMSE      Rsquared
  0.01  3.273963  0.032298915
  0.02  3.292442  0.071884315
  0.03  3.280420  0.071884315
  0.04  3.280420  0.071884315
  0.05  3.248108  0.078288595
  0.06  3.248108  0.078288595
  0.07  3.214871  0.072679407
  0.08  3.122272  0.003233272
  0.09  3.122272  0.003233272
  0.10  3.089782  0.004856115
```

Nous affichons ce modèle.
Un arbre de classification peut être créé à l'aide de la fonction rpart(). C'est ce que nous faisons et nous le stockons dans la variable main_tree.



Voici l'affichage de notre arbre.
Pour le nœud intermédiaire, un cas concerne le nœud enfant gauche si et seulement si la condition est satisfaite. La classe prédite est donnée sous chaque noeud feuille.
Ici, les différentes catégories représentent le nombre de patients. Cela dépend du facteur "difference".

```
> rmse(newPatientsDataNo$NUMBEROFPATIENTS, pre_score)
[1] 1.055069
```

RMSE donne l'écart type de l'erreur de prédiction du modèle. Une valeur plus petite indique une meilleure performance du modèle.
Ici, l'écarte type de l'erreur est environ égal à 1.05. Nous pouvons donc en déduire que notre modèle est performant.

```
library(randomForest)
library(foreach)

#set tuning parameters
control <- trainControl(method = "cv", number = 5)


#random forest model
rf_model <- train(NUMBEROFPATIENTS ~ ., data = dataset, method = "parRF", trControl = control)

#check optimal parameters
print(rf_model)
```

Nous avons maintenant besoin des packages suivants : randomForest et foreach.

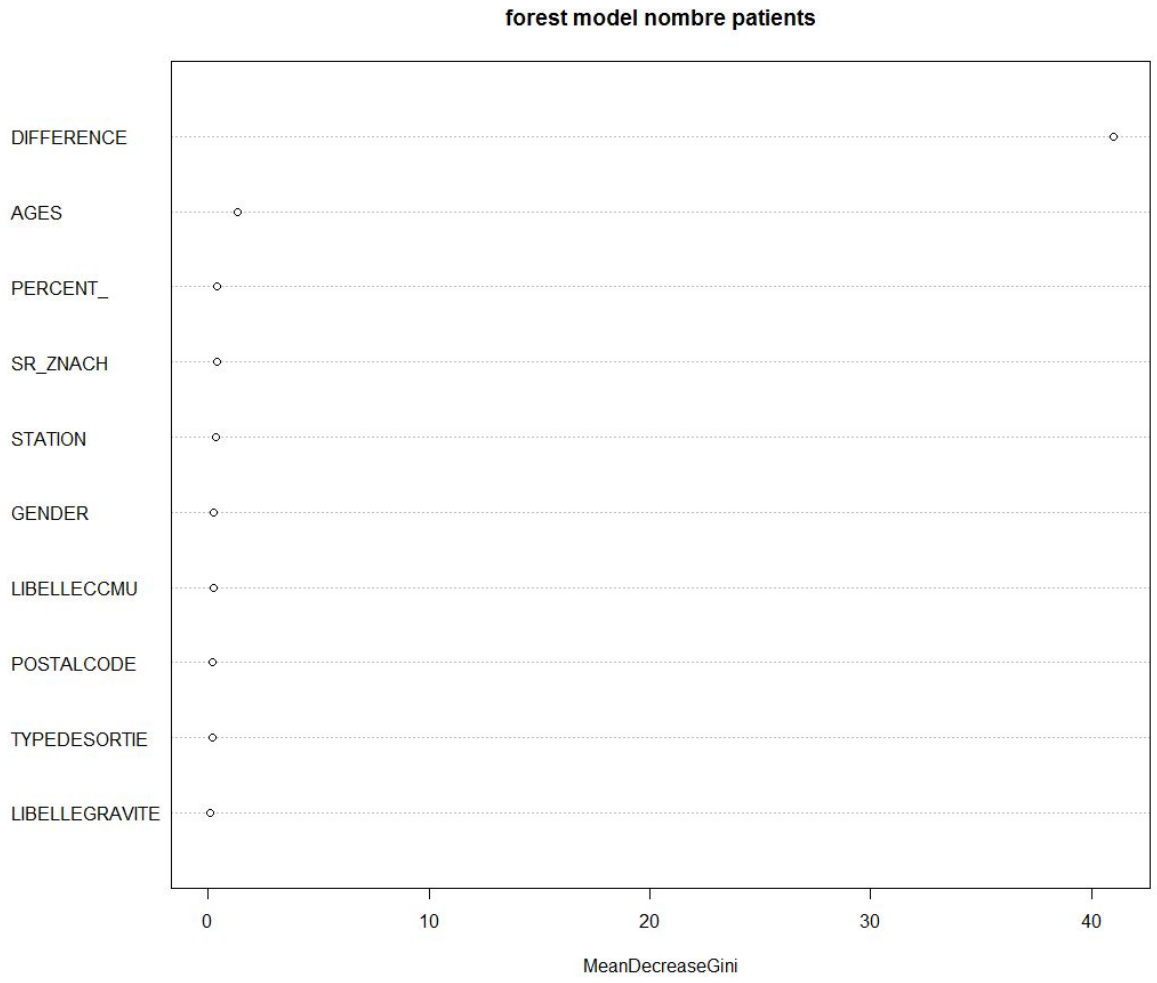
Les forêts aléatoires ou les forêts de décision aléatoire sont une méthode d'apprentissage d'ensemble pour la classification qui fonctionnent en construisant une multitude d'arbres de décision.

```
> print(rf_model)
Parallel Random Forest

51 samples
10 predictors

No pre-processing
Resampling: Cross-Validated (5 fold)
Summary of sample sizes: 42, 41, 41, 41, 39
Resampling results across tuning parameters:

  mtry  RMSE      Rsquared
   2    2.858527  0.3442063
  31    2.046240  0.6468595
  60    1.514366  0.8053893

RMSE was used to select the optimal model using  the smallest value.
The final value used for the model was mtry = 60.
```

Ici, la valeur optimale pour mtry est 60.

```
forest_model <- randomForest(as.factor(NUMBEROFPATIENTS) ~ ., data = dataset, mtry = 60, ntree = 1000)
print(forest_model)
varImpPlot(forest_model, main="forest model nombre de patients")
```

varImplot permet d'obtenir un graphique avec un score d'importance variable sur l'axe des X et le nom de la variable sur l'axe des Y.



Les études incluses dans le random forest seront généralement identifiées par ordre chronologique sur le côté gauche par auteur et date. Il n'y a aucune importance accordée à la position verticale assumée par une étude particulière.
La partie graphique de la parcelle forestière sera du côté droit et indiquera la différence entre les groupes témoins et témoins dans les études. La distance horizontale d'une boîte à partir de l'axe des y montre la différence entre le test et le contrôl.
Les lignes horizontales minces - parfois appelées whiskers - émergeant de la boîte indiquent la grandeur de l'intervalle de confiance. Plus les lignes sont longues, plus l'intervalle de confiance est large, et moins les données sont fiables. Plus les lignes sont courtes, plus l'intervalle de confiance est étroit et plus les données sont fiables.
Si la zone ou les moustaches de l'intervalle de confiance passent à travers l'axe y sans effet, les données de l'étude sont statistiquement insignifiantes.
La signification des données de l'étude, ou la puissance, est indiquée par le poids (taille) de la boîte. Des données plus significatives, telles que celles provenant d'études avec une plus grande taille d'échantillon et des intervalles de confiance plus petits, sont indiquées par une boîte de plus grande taille que les données provenant d'études moins significatives, et elles contribuent au résultat combiné à un degré plus important.
Le “parc forestier” est capable de démontrer la mesure dans laquelle les données provenant d'études multiples observant le même effet se chevauchent. Les résultats qui ne parviennent pas à se chevaucher sont qualifiés d'hétérogènes et sont appelés l'hétérogénéité des

données: ces données sont moins concluantes. Si les résultats sont similaires entre différentes études, les données sont dites homogènes et la tendance est que ces données sont plus concluantes.
L'hétérogénéité est indiquée par I2. Une hétérogénéité inférieure à 50% est appelée faible et indique une plus grande similitude entre les données d'étude qu'une valeur I2 supérieure à 50%, ce qui indique une plus grande divergence.

GINI : L'importance de GINI mesure le gain de pureté moyen par splits d'une variable donnée. Si la variable est utile, elle a tendance à diviser les noeuds marqués mélangés en noeuds de classe unique. Le fractionnement par une variable permutée n'a tendance à augmenter ni à diminuer les puretés du noeud. Permutant une variable utile, ont tendance à donner une diminution relativement importante du gain moyen de gini. L'importance de GINI est étroitement liée à la fonction de décision locale, que la forêt aléatoire utilise pour sélectionner la meilleure division disponible. Un faible Gini (c'est-à-dire une diminution plus élevée dans Gini) signifie qu'une variable prédictive particulière joue un rôle plus important dans le partage des données dans les classes définies.

Ici, nous pouvons constater que pour le nombre de patients, la variable DIFFERENCE joue un rôle plus important dans le partage des données.

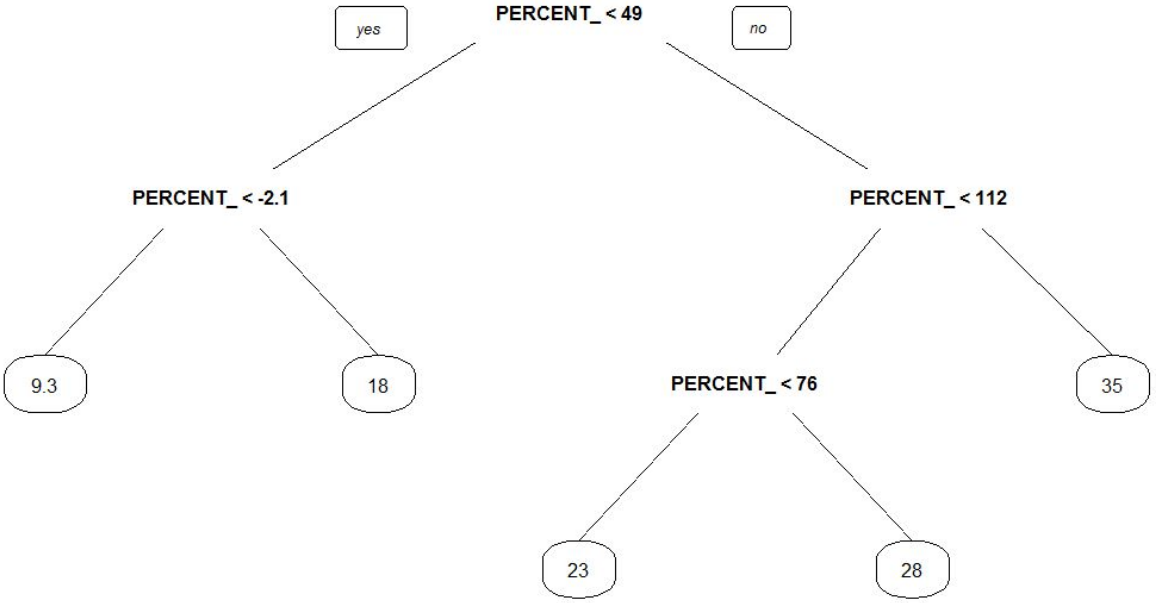
Prédictif Pollution NO

On reprend les mêmes algorithmes

```
#decision tree
tree_model <- train(SR_ZNACH ~ ., data = dataset, method = "rpart", trControl = fitControl, tuneGrid = cartGrid)
print(tree_model)

main_tree <- rpart(SR_ZNACH ~ ., data = dataset, control = rpart.control(cp=0.01))
prp(main_tree)
pre_score <- predict(main_tree, type = "vector")
rmse(dataset$SR_ZNACH, pre_score)
```

```
> rmse(dataset$SR_ZNACH, pre_score)
[1] 2.656961
```

On constate ici que l'écart - type est bas et donc notre modèle est performant.

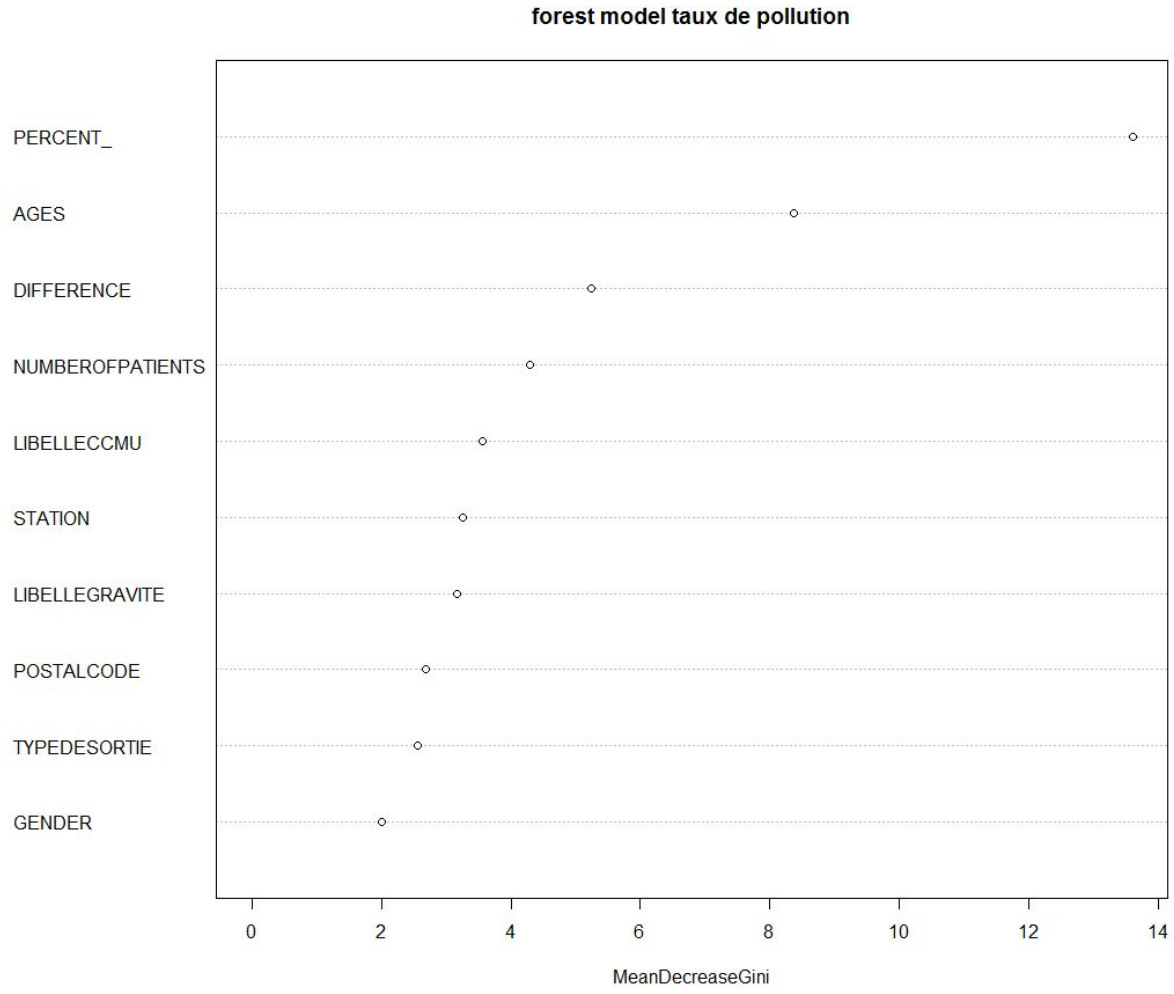


Pour le nœud intermédiaire, un cas concerne le nœud enfant gauche si et seulement si la condition est satisfaite. La classe prédite est donnée sous chaque noeud feuille.
Ici, les différentes catégories représentent le taux de pollution moyen pour le polluant NO. Cela dépend du facteur “percent”.

```
> print(rf_model)
Parallel Random Forest

51 samples
10 predictors

No pre-processing
Resampling: Cross-Validated (5 fold)
Summary of sample sizes: 41, 41, 41, 41, 40
Resampling results across tuning parameters:

  mtry  RMSE      Rsquared
   2    8.375591  0.1603169
  31    3.238589  0.9176103
  60    1.554734  0.9783273

RMSE was used to select the optimal model using  the smallest value.
The final value used for the model was mtry = 60.
```

La valeur optimale pour mtry est ici 60.

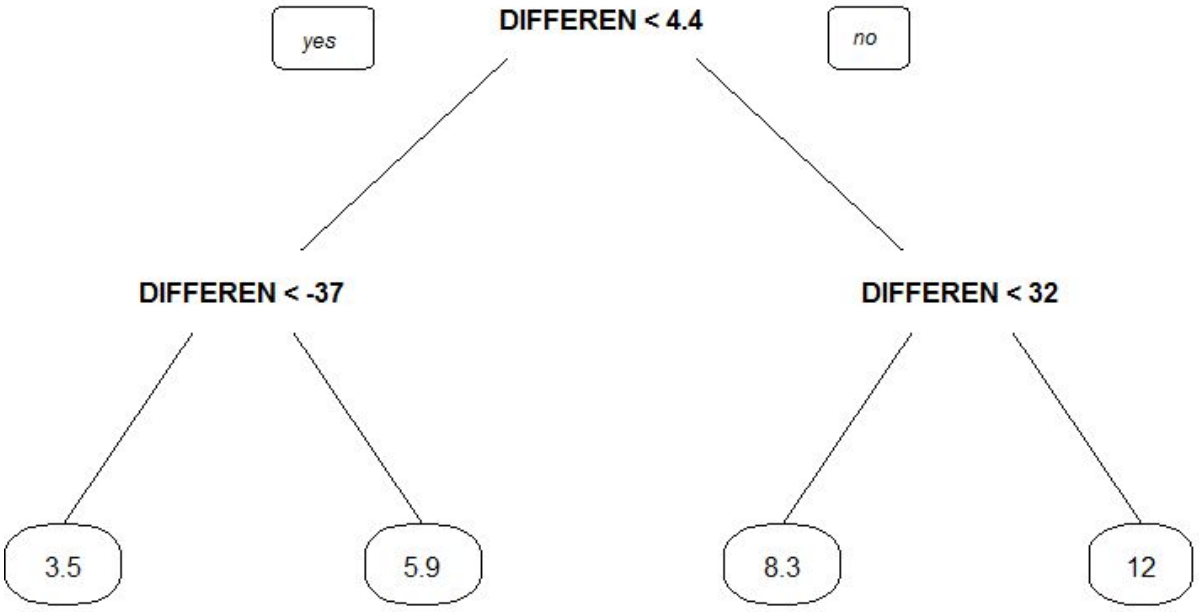
```
forest_model <- randomForest(as.factor(SR_ZNACH) ~ ., data = dataset, mtry = 60, ntree = 1000)
print(forest_model)
varImpPlot(forest_model, main="forest model taux de pollution")
```



Ici, nous pouvons voir que la variable PERCENT joue un rôle plus important dans le partage des données en ce qui concerne le taux de pollution. L'age des patients joue aussi un rôle important.

POLLUANT NO2

Predictif Number of patients

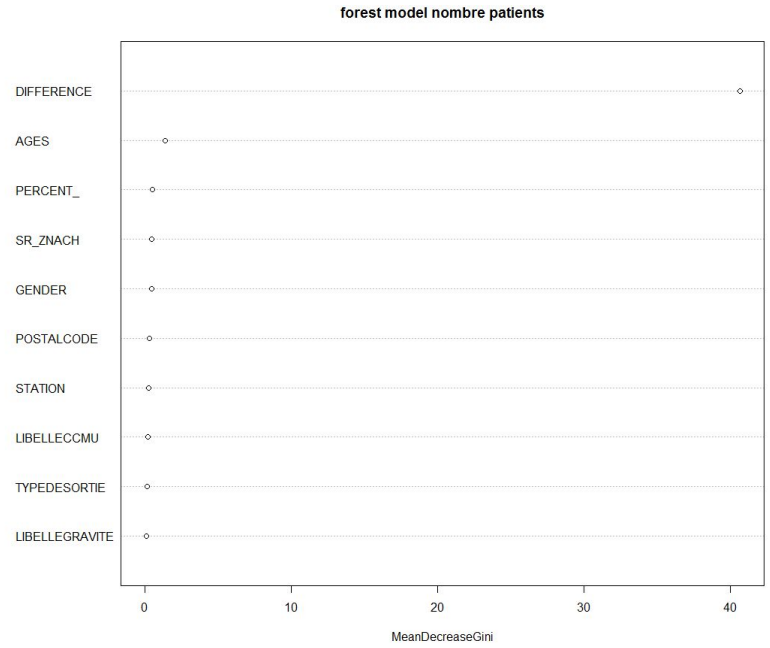



```
> rmse(newPatientsDataNo2$NUMBEROFPATIENTS, pre_score)
[1] 1.055069
```

L'écart-type est bas. Notre modèle est donc performant.

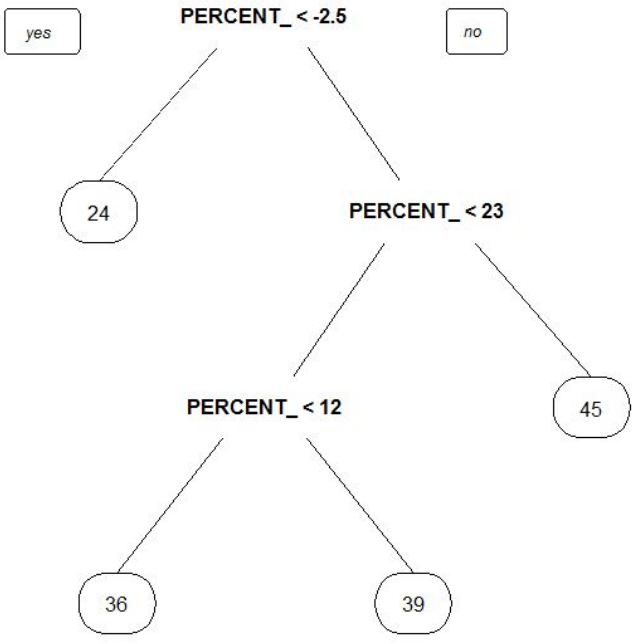
```
RMSE was used to select the optimal model using  the smallest value.
The final value used for the model was mtry = 52.
```

La valeur optimale pour mtry est ici 52.



Ici, nous pouvons constater que pour le nombre de patients, la variable DIFFERENCE joue un rôle plus important dans le partage des données.

Predictif Pollution NO2

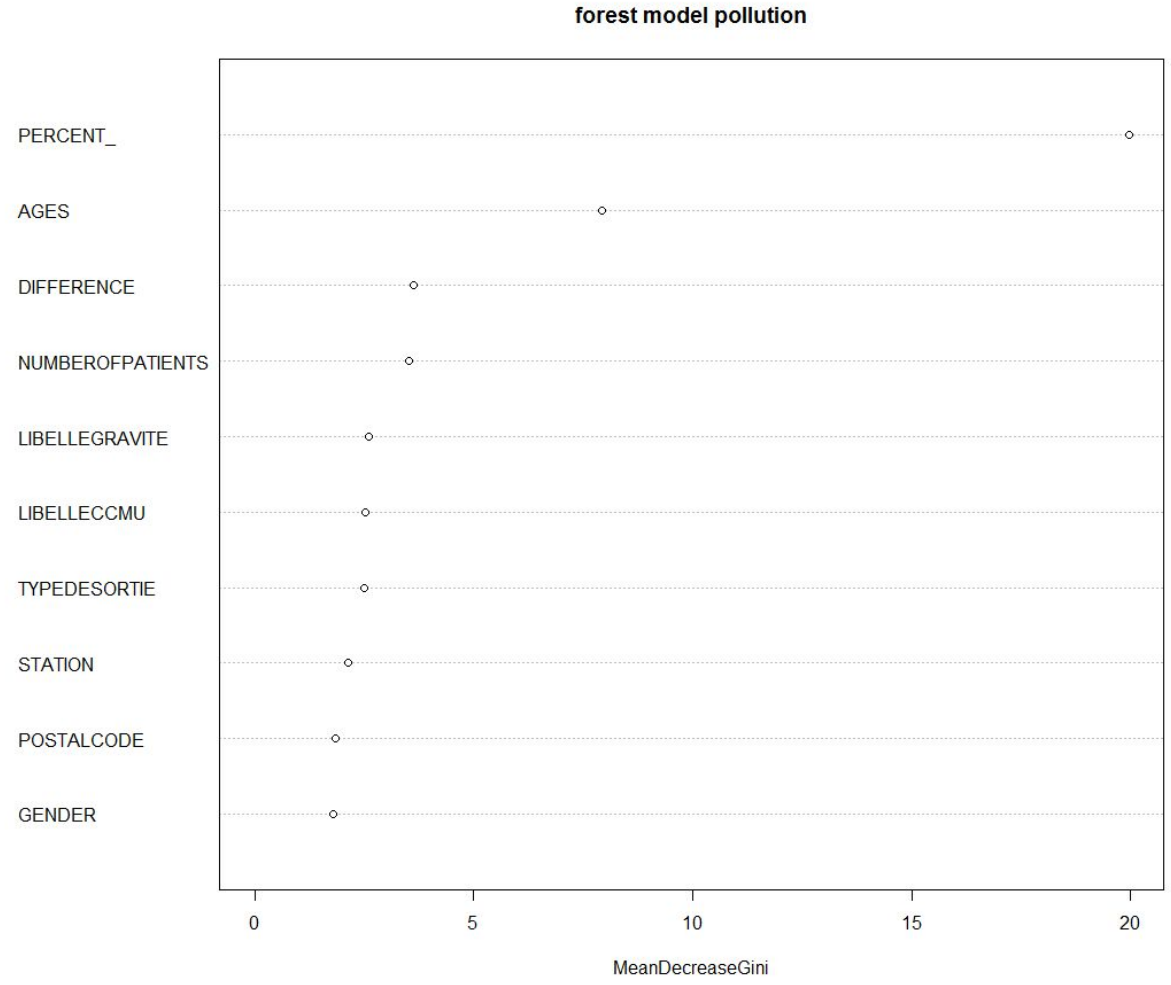



```
> rmse(dataset$SR_ZNACH, pre_score)
[1] 3.99827
```

L'écart-type est bas : notre modèle est performant.

```
RMSE was used to select the optimal model using  the smallest value.
The final value used for the model was mtry = 52.
```

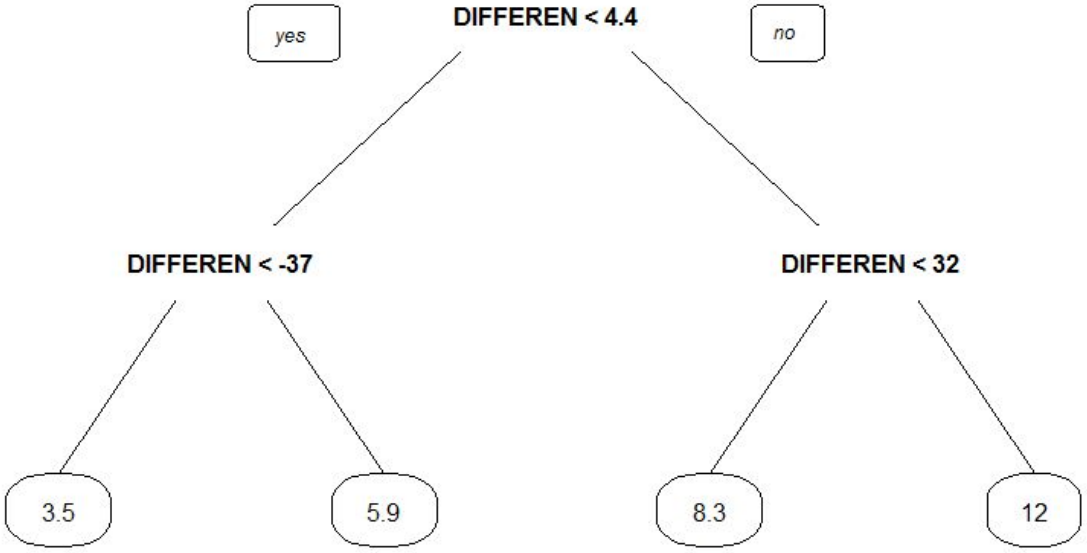
La valeur optimale pour mtry est ici 52.



Ici, nous pouvons constater que pour le nombre de patients, la variable PERCENT joue un rôle plus important dans le partage des données.

POLLUANT NOX

Predictif Number of patients

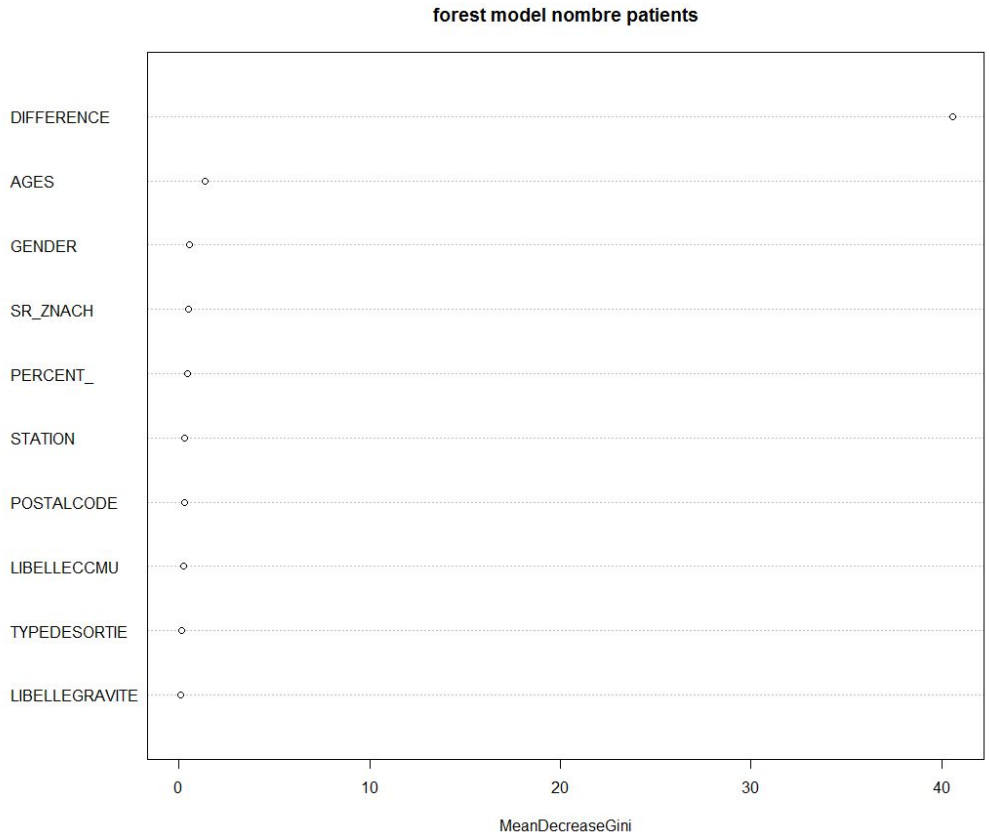



```
> rmse(newPatientsDataNox$NUMBEROFPATIENTS, pre_score)
[1] 1.055069
```

L'écart-type est bas. Notre modèle est donc performant.

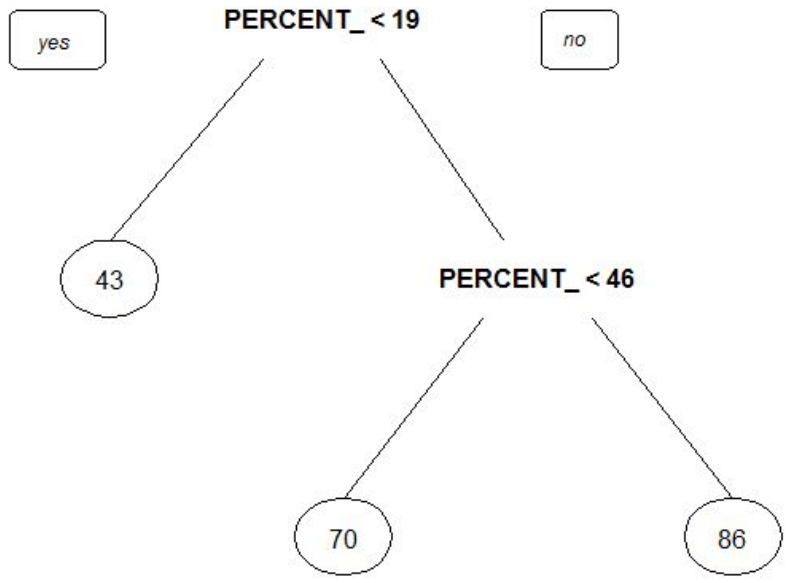
```
RMSE was used to select the optimal model using  the smallest value.
The final value used for the model was mtry = 52.
```

La valeur optimale pour mtry est ici 52.



Ici, nous pouvons constater que pour le nombre de patients, la variable DIFFERENCE joue un rôle plus important dans le partage des données.

Predictif Pollution NOX

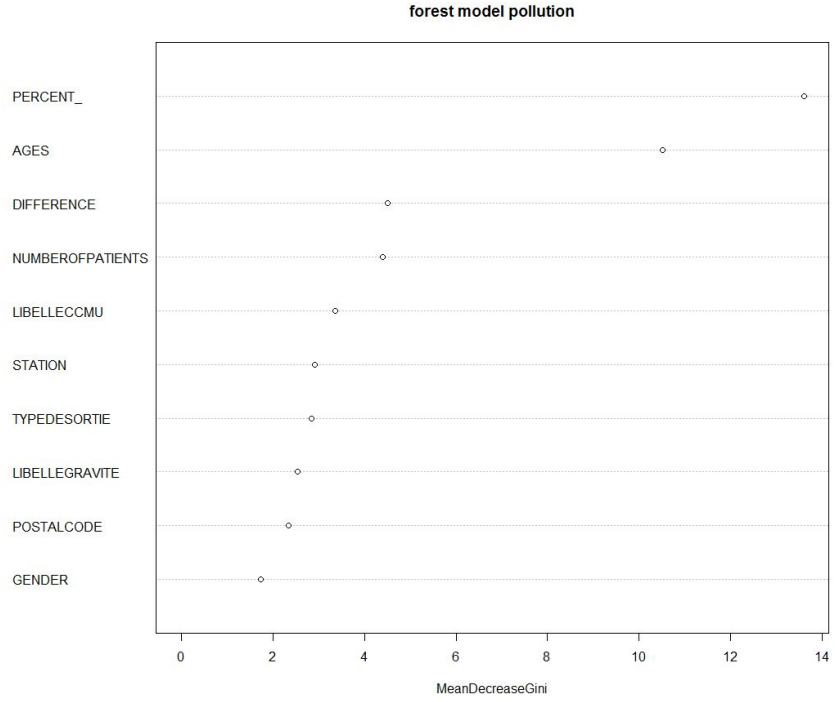



```
> rmse(dataset$SR_ZNACH, pre_score)
[1] 7.8341
```

L’écart-type est bas. Notre modèle est donc performant.

```
RMSE was used to select the optimal model using  the smallest value.
The final value used for the model was mtry = 52.
```

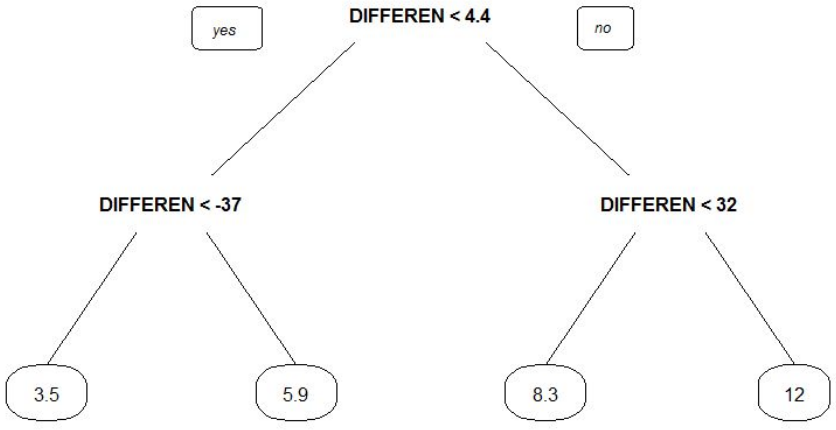
La valeur optimale pour mtry est ici 52.



Ici, nous pouvons constater que pour le nombre de patients, la variable PERCENT joue un rôle plus important dans le partage des données.

POLLUANT O3

Predictif Number of patients

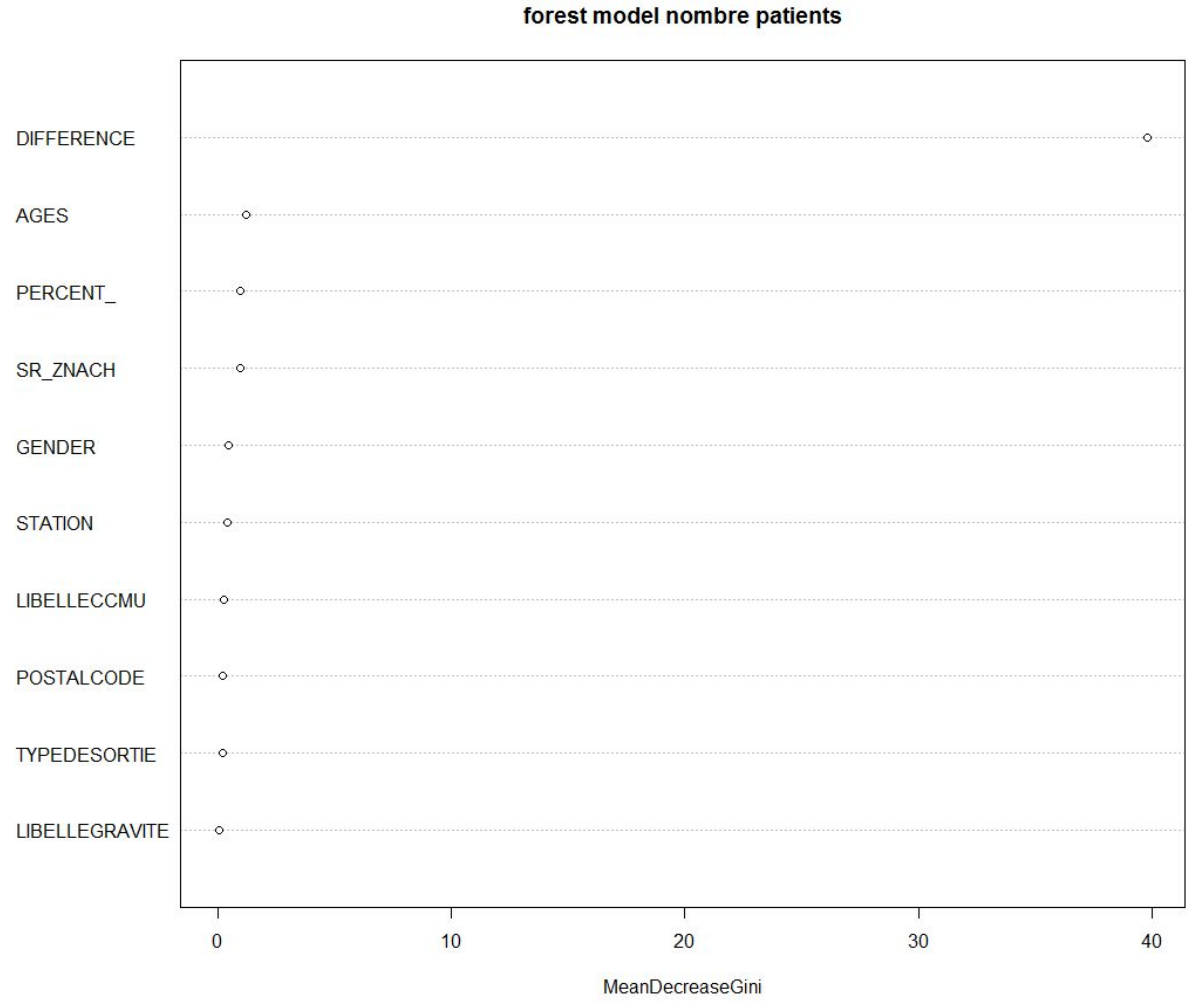



```
> rmse(newPatientsDataO3$NUMBEROFPATIENTS, pre_score)
[1] 1.055069
```

L’écart-type est bas. Notre modèle est donc performant.

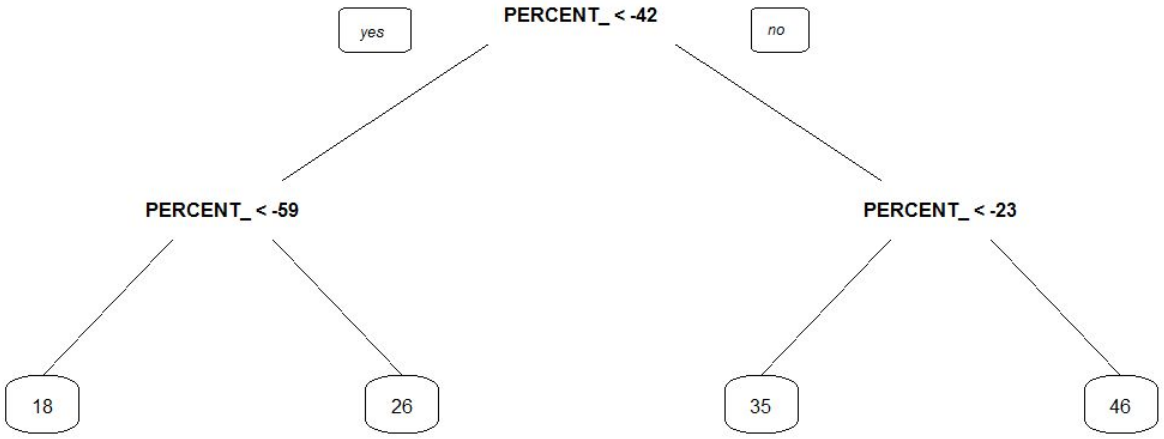
```
RMSE was used to select the optimal model using  the smallest value.
The final value used for the model was mtry = 52.
```

mtry a ici pour valeur optimale 52.



Ici, nous pouvons constater que pour le nombre de patients, la variable DIFFERENCE joue un rôle plus important dans le partage des données.

Predictif Pollution O3

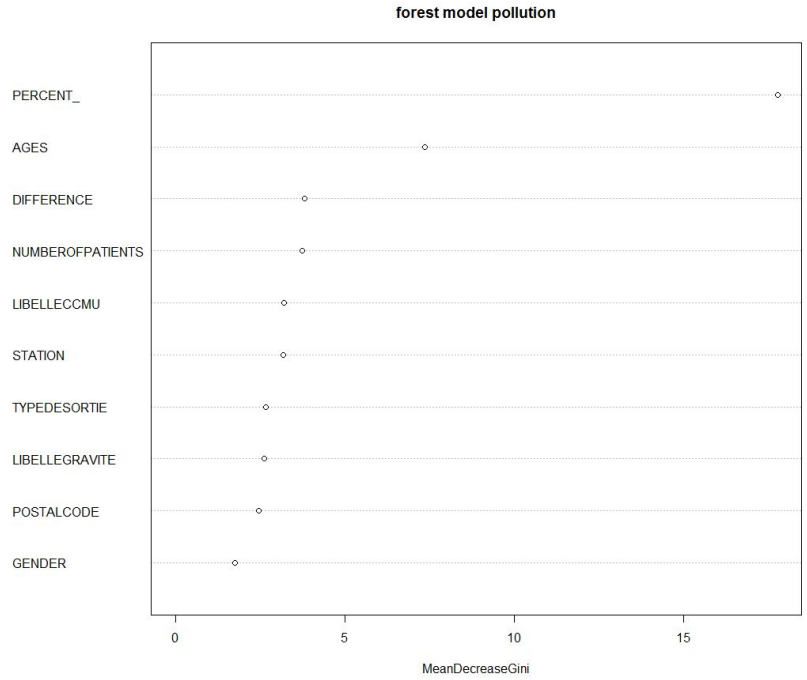



```
> rmse(dataset$SR_ZNACH, pre_score)
[1] 3.019277
```

L'écart-type est bas. Notre modèle est donc performant.

```
RMSE was used to select the optimal model using  the smallest value.
The final value used for the model was mtry = 52.
```

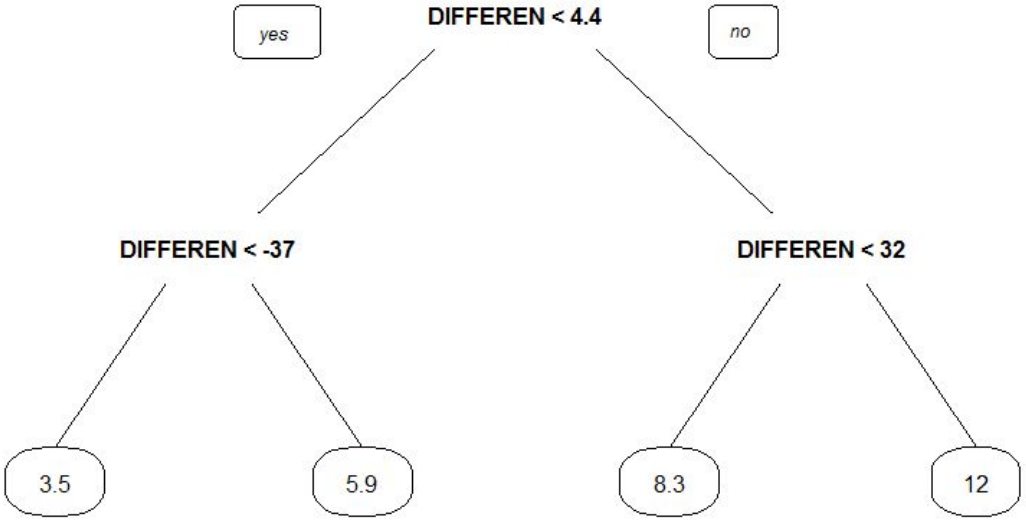
La valeur optimale pour mtry est ici 52.



Ici, nous pouvons constater que pour le nombre de patients, la variable PERCENT joue un rôle plus important dans le partage des données.

POLLUANT PM10

Predictif Number of patients

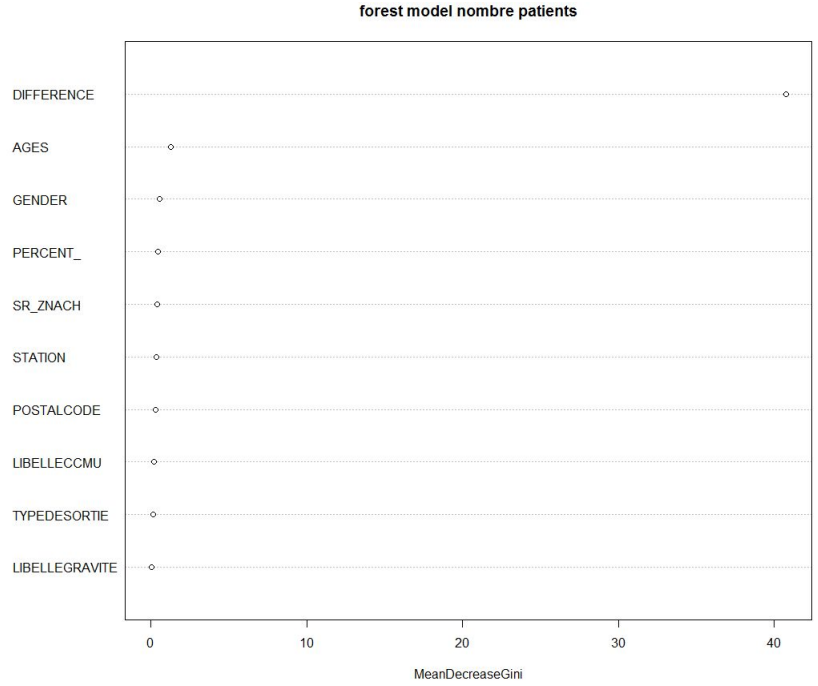



```
> rmse(dataset$NUMBEROFPATIENTS, pre_score)
[1] 1.055069
```

L'écart-type est bas. Notre modèle est donc performant.

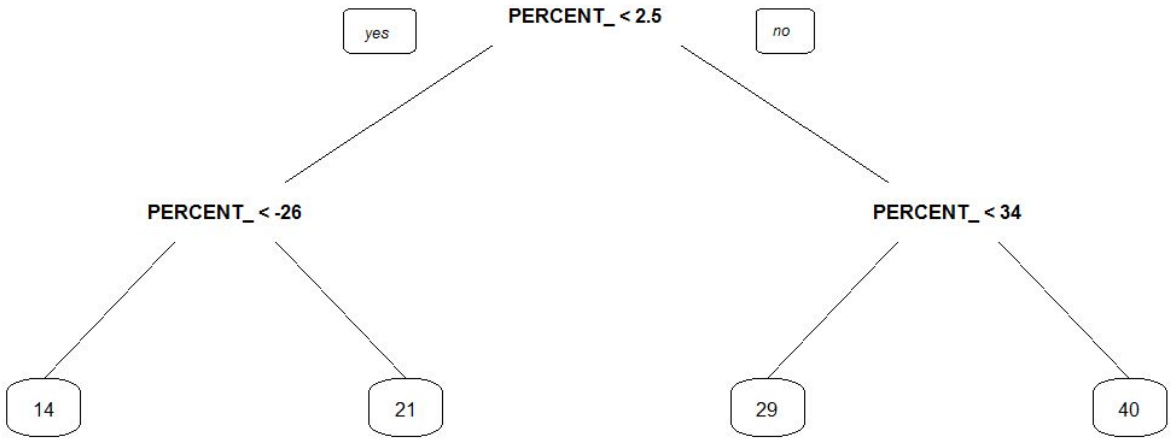
```
RMSE was used to select the optimal model using  the smallest value.
The final value used for the model was mtry = 52.
```

La valeur optimale pour mtry est ici 52.



Ici, nous pouvons constater que pour le nombre de patients, la variable DIFFERENCE joue un rôle plus important dans le partage des données.

Predictif Pollution PM10

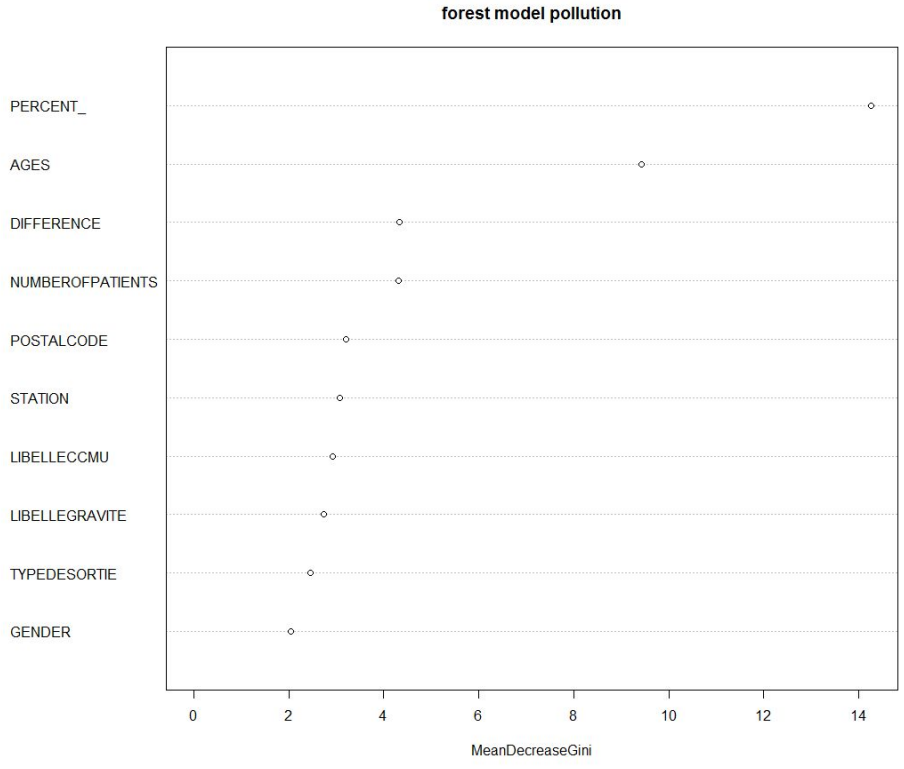



```
> rmse(dataset$SR_ZNACH, pre_score)
[1] 5.37224
```

L'écart-type est bas. Notre modèle est donc performant.

```
RMSE was used to select the optimal model using  the smallest value.
The final value used for the model was mtry = 52.
```

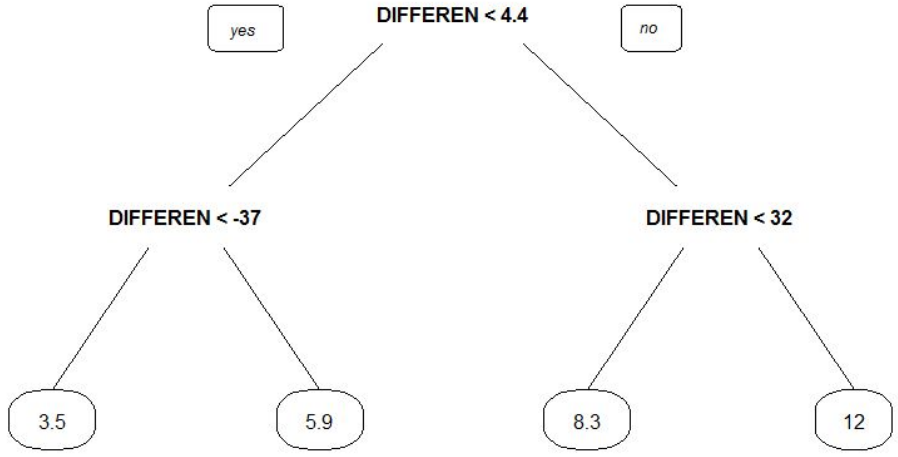
La valeur optimale pour mtry est ici 52.



Ici, nous pouvons constater que pour le nombre de patients, la variable PERCENT joue un rôle plus important dans le partage des données.

POLLUANT PM25

Predictif Number of patients

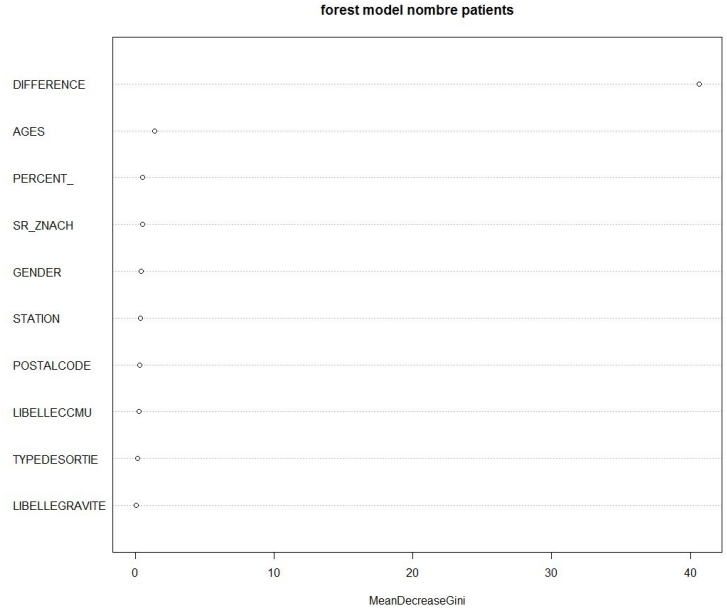



```
> rmse(dataset$NUMBEROFPATIENTS, pre_score)
[1] 1.055069
```

L'écart-type est bas. Notre modèle est donc performant.

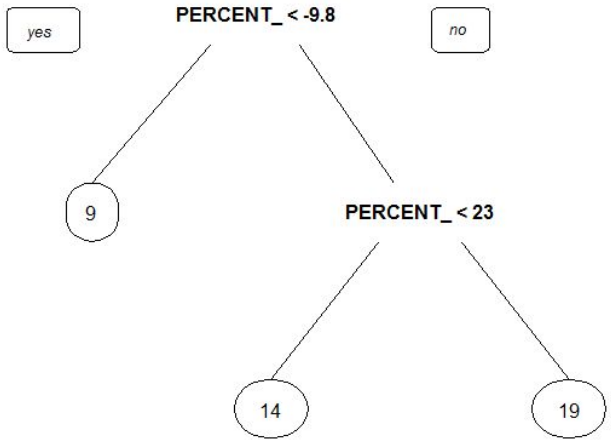
```
RMSE was used to select the optimal model using  the smallest value.
The final value used for the model was mtry = 52.
```

La valeur optimale pour mtry est ici 52.



Ici, nous pouvons constater que pour le nombre de patients, la variable DIFFERENCE joue un rôle plus important dans le partage des données.

Predictif Pollution PM25

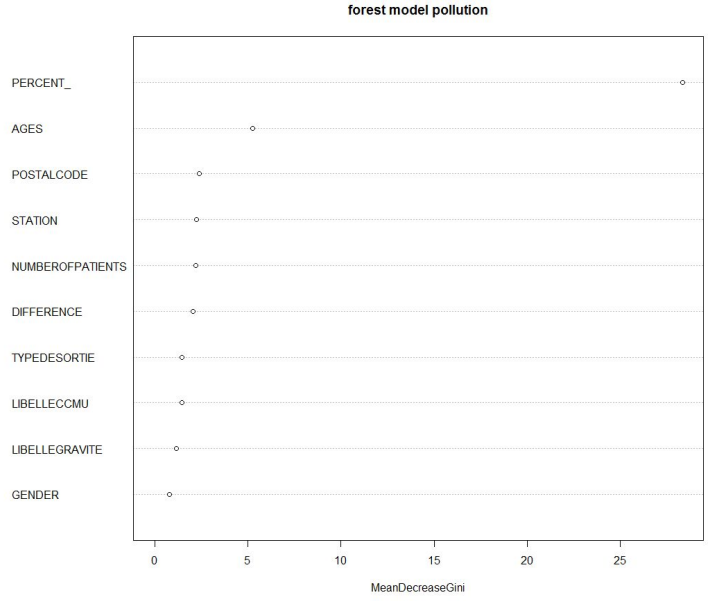



```
> rmse(dataset$SR_ZNACH, pre_score)
[1] 1.738791
```

L'écart-type est bas. Notre modèle est donc performant.

```
RMSE was used to select the optimal model using  the smallest value.
The final value used for the model was mtry = 52.
```

La valeur optimale pour mtry est ici 52.



Ici, nous pouvons constater que pour le nombre de patients, la variable PERCENT joue un rôle plus important dans le partage des données.

Conclusion :

Nous avons dans ce document vu les différentes prédictions pour le nombre de patients et le taux de pollution pour chaque polluant.

Nous constatons que pour les patients, la prédiction s'établie en fonction de la variable DIFFERENCE et pour le taux de pollution pour un polluant, elle s'établie en fonction de la variable PERCENT.
Chaque arbre nous donne des classes de prédiction pour l'année à venir en fonction de ces variables.
Les écart-type correspondant à ces modèles sont faibles : nous pouvons en déduire que ces modèles sont performants.