multi_linear_reg_alex

alexandre

2023-11-29

Importation et modification des données

On va également enlever les données aberantes qui compromettent la regression lineaire.

```
library(corrplot)
```

```
## corrplot 0.92 loaded
```

```
library(factoextra)
```

```
## Le chargement a nécessité le package : ggplot2
```

```
## Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WBa
```

```
library(FactoMineR)
library(coefplot)
library(ggplot2)
library(gridExtra)
library(ggfortify)
```

```
## Registered S3 methods overwritten by 'ggfortify':
##   method         from
##   autoplot.acf   useful
##   fortify.acf    useful
##   fortify.kmeans useful
##   fortify.ts     useful
```

```
library(plotly)
```

```
##
## Attachement du package : 'plotly'
```

```
## L'objet suivant est masqué depuis 'package:ggplot2':
##
##     last_plot
```

```
## L'objet suivant est masqué depuis 'package:stats':
##
##     filter
```

```
## L'objet suivant est masqué depuis 'package:graphics':
##
##     layout
```

```
library(ellipse)
```

```
##
## Attachement du package : 'ellipse'

## L'objet suivant est masqué depuis 'package:graphics':
##
##     pairs
```

```
library(leaps)
library(MASS)
```

```
##
## Attachement du package : 'MASS'

## L'objet suivant est masqué depuis 'package:plotly':
##
##     select
```

```
library(corrplot)
library(glmnet)
```

```
## Le chargement a nécessité le package : Matrix

## Loaded glmnet 4.1-8
```

```
library(coefplot)
library(ggplot2)
library(gridExtra)
library(ggfortify)
library(plotly)
library(reshape2)
Data<-read.csv('Data-projetmodIA-2324.csv')

data_quant=Data[,c("nox_kg","so2_kg","pm10_kg","pm25_kg","co_kg","c6h6_kg","nh3_kg","ges_teqco2","ch4_t
data_quant_scaled <- scale(log(data_quant))
data_scaled_df <- as.data.frame(data_quant_scaled)
```

```
enlever_donnee_aber <- function(data_frame,columns) {
# Définir le facteur d'échelle interquartile (IQR)
iqr_factor <- 1.5
# Appliquer la règle des quantiles pour chaque colonne
for (col in columns) {
# Calculer les quantiles
q1 <- quantile(data_frame[[col]], 0.15)
q3 <- quantile(data_frame[[col]], 0.85)
```

```
# Calculer l'IQR
iqr <- q3 - q1
# Calculer les limites
lower_limit <- q1 - iqr_factor * iqr
upper_limit <- q3 + iqr_factor * iqr
# Supprimer les outliers
data_frame <- data_frame[data_frame[[col]] >= lower_limit & data_frame[[col]] <= upper_limit, ]
}
return(data_frame)
}
```

```
data_scaled_df=enlever_donnee_aber(data_scaled_df,colnames(data_scaled_df))
```

Modèle linéaire additif expliquant le gaz à effet de serre en fonction de tous les autres polluants

On a le modèle additif suivant que l'on peut ajuster sur R de la façon suivante

$$ges_i = \theta_0 + \theta_1 nox_i + \theta_2 so2_i + \theta_3 pm10_i + \theta_4 pm25_i + \theta_5 co_i + \theta_6 c6h6_i + \theta_7 nh3_i + \theta_8 ch4_i + \theta_9 co2_i + \theta_1 0no2_i + \epsilon$$

```
mod_ges=lm(formula=ges_teqco2~.,data=data_scaled_df)
summary(mod_ges)
```

```
##
## Call:
## lm(formula = ges_teqco2 ~ ., data = data_scaled_df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.55525 -0.09027 -0.01230  0.06137  0.88949
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.008201   0.004786  -1.714 0.086926 .
## nox_kg       0.267759   0.030125   8.888  < 2e-16 ***
## so2_kg       0.036316   0.009534   3.809 0.000148 ***
## pm10_kg     -0.271206   0.030295  -8.952  < 2e-16 ***
## pm25_kg      0.157159   0.037777   4.160 3.47e-05 ***
## co_kg       -0.009881   0.051616  -0.191 0.848228
## c6h6_kg      0.001898   0.042850   0.044 0.964676
## nh3_kg      -0.279798   0.029280  -9.556  < 2e-16 ***
## ch4_t        0.244085   0.012347  19.769  < 2e-16 ***
## co2_t        0.604444   0.031403  19.248  < 2e-16 ***
## n2o_t        0.393794   0.030976  12.713  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1479 on 948 degrees of freedom
## Multiple R-squared:  0.9715, Adjusted R-squared:  0.9712
## F-statistic:  3235 on 10 and 948 DF,  p-value: < 2.2e-16
```

Il est à noter que le test de nullité pour certaines variables telles que c6h6_kg et co_kg présente une p-value supérieure à 0,05. Cela pourrait suggérer la possibilité de les exclure du modèle afin de le simplifier.

Selection des variables explicatives

Nous allons maintenant simplifier le modèle en selctionnant les variables explicatives pertinentes

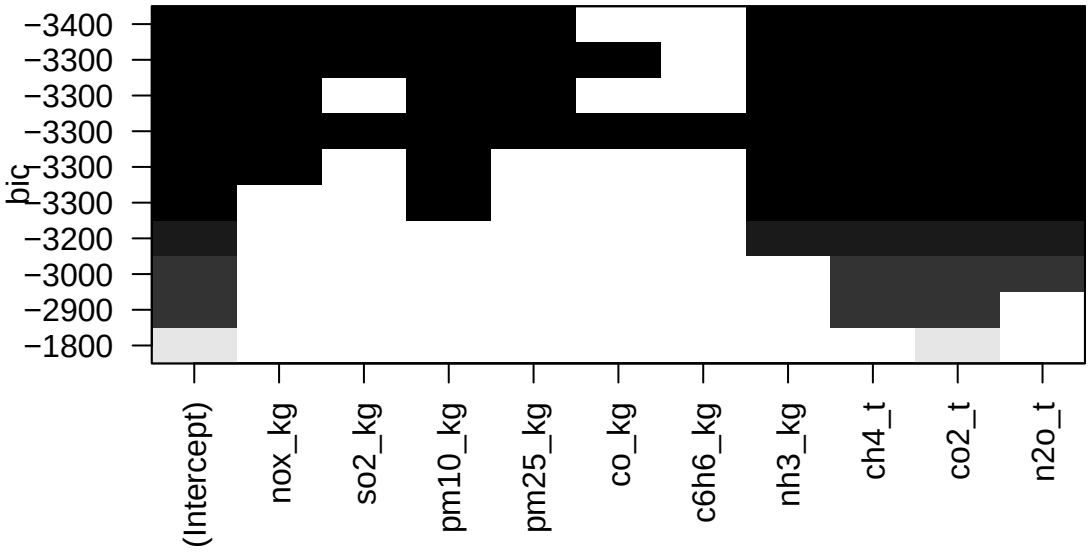
```
library(leaps)
```

Avec la méthode backward

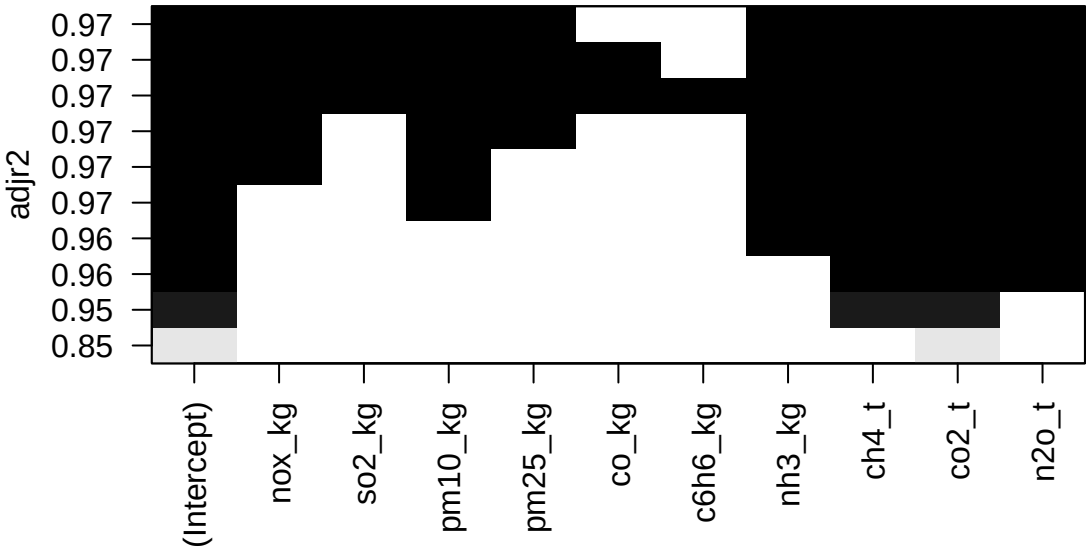
```
choixb<-regsubsets(ges_teqco2~.,data=data_scaled_df,nbest=1,nvmax=10,method="backward")
summary(choixb)
```

```
## Subset selection object
## Call: regsubsets.formula(ges_teqco2 ~ ., data = data_scaled_df, nbest = 1,
##     nvmax = 10, method = "backward")
## 10 Variables  (and intercept)
##         Forced in Forced out
## nox_kg      FALSE      FALSE
## so2_kg      FALSE      FALSE
## pm10_kg     FALSE      FALSE
## pm25_kg     FALSE      FALSE
## co_kg       FALSE      FALSE
## c6h6_kg     FALSE      FALSE
## nh3_kg      FALSE      FALSE
## ch4_t       FALSE      FALSE
## co2_t       FALSE      FALSE
## n2o_t       FALSE      FALSE
## 1 subsets of each size up to 10
## Selection Algorithm: backward
##           nox_kg so2_kg pm10_kg pm25_kg co_kg c6h6_kg nh3_kg ch4_t co2_t n2o_t
## 1  ( 1 )  " "    " "    " "     " "     " "   " "     " "    " "   "*"   " "
## 2  ( 1 )  " "    " "    " "     " "     " "   " "     " "    "*"   "*"   " "
## 3  ( 1 )  " "    " "    " "     " "     " "   " "     " "    "*"   "*"   "*"
## 4  ( 1 )  " "    " "    " "     " "     " "   " "     "*"    "*"   "*"   "*"
## 5  ( 1 )  " "    " "    "*"     " "     " "   " "     "*"    "*"   "*"   "*"
## 6  ( 1 )  "*"    " "    "*"     " "     " "   " "     "*"    "*"   "*"   "*"
## 7  ( 1 )  "*"    " "    "*"     "*"     " "   " "     "*"    "*"   "*"   "*"
## 8  ( 1 )  "*"    "*"    "*"     "*"     " "   " "     "*"    "*"   "*"   "*"
## 9  ( 1 )  "*"    "*"    "*"     "*"     "*"   " "     "*"    "*"   "*"   "*"
## 10  ( 1 ) "*"    "*"    "*"     "*"     "*"   "*"     "*"    "*"   "*"   "*"
```

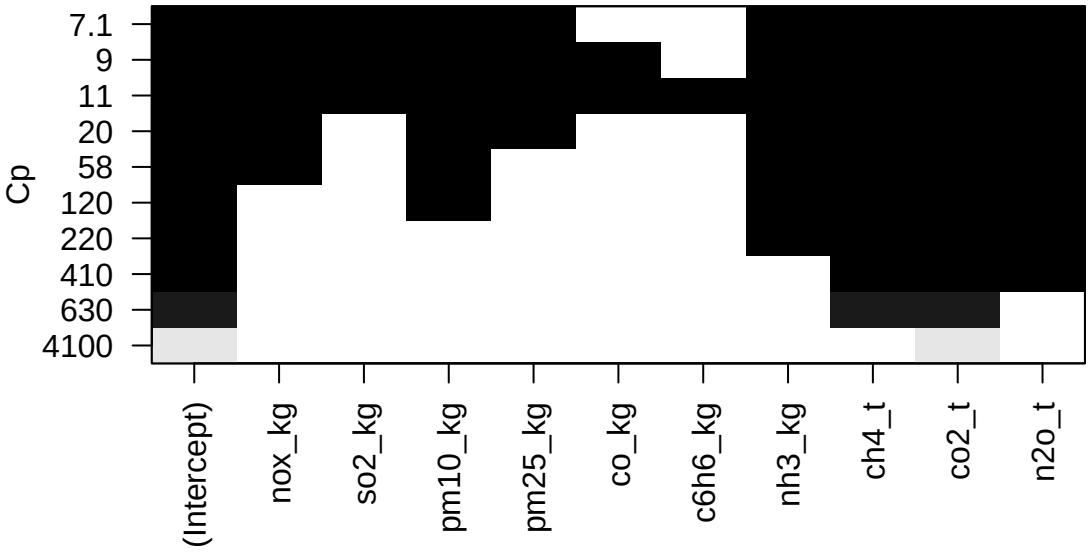
```
plot(choixb,scale="bic")
```




```
plot(choixb,scale="adjr2")
```




```
plot(choixb,scale="Cp")
```



En utilisant la méthode Backward, tous les critères conduisent à la même sélection de variables, celle pour laquelle nous avions formulé l'hypothèse précédemment lors des tests de nullité.

Voici le modèle simplifié :

$ges_i = \theta_0 + \theta_1 nox_i + \theta_2 so2_i + \theta_3 pm10_i + \theta_4 pm25_i + \theta_5 nh3_i + \theta_6 ch4_i + \theta_7 co2_i + \theta_8 no2_i + \epsilon$

Avec la méthode forward

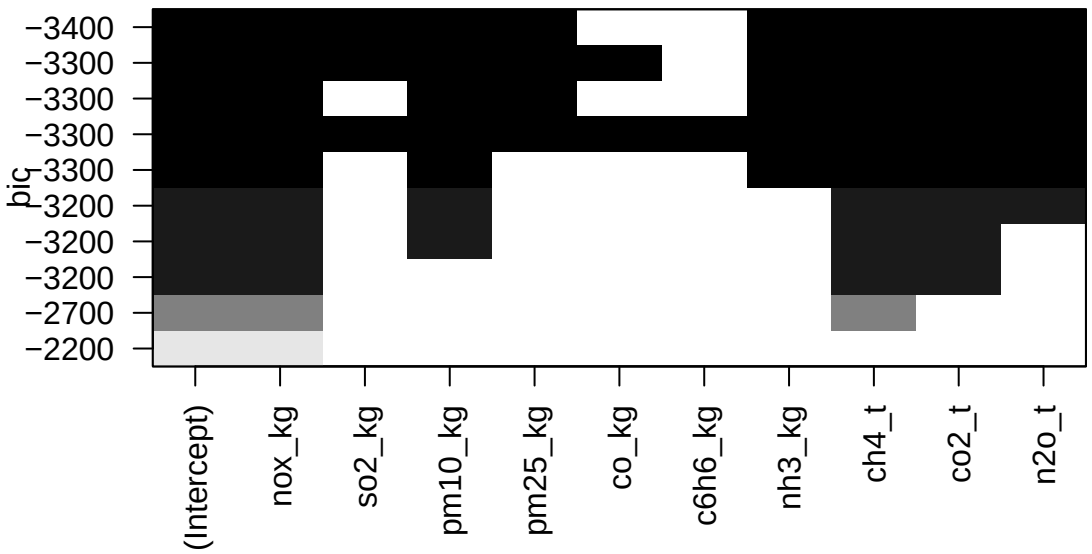
Nous allons maintenant effectuer la même selection mais cette fois-ci avec la méthode pour vérifier la simplification possible du modèle.

```
choixf<-regsubsets(ges_teqco2~.,data=data_scaled_df,nbest=1,nvmax=10,method="forward")
summary(choixf)
```

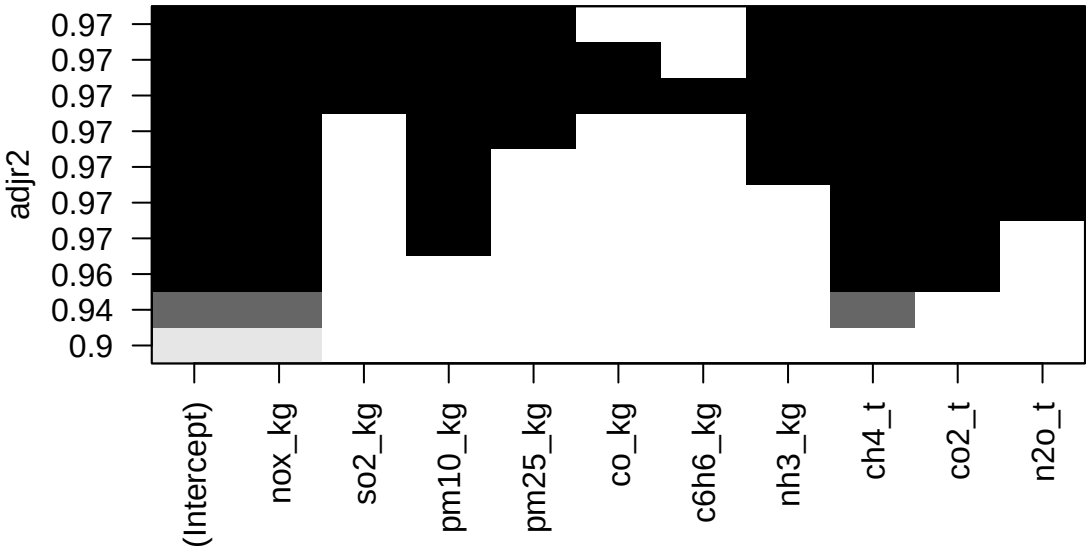
```
## Subset selection object
## Call: regsubsets.formula(ges_teqco2 ~ ., data = data_scaled_df, nbest = 1,
##     nvmax = 10, method = "forward")
## 10 Variables  (and intercept)
##         Forced in Forced out
## nox_kg      FALSE      FALSE
## so2_kg      FALSE      FALSE
## pm10_kg     FALSE      FALSE
## pm25_kg     FALSE      FALSE
## co_kg       FALSE      FALSE
```

```
## c6h6_kg     FALSE      FALSE
## nh3_kg      FALSE      FALSE
## ch4_t       FALSE      FALSE
## co2_t       FALSE      FALSE
## n2o_t       FALSE      FALSE
## 1 subsets of each size up to 10
## Selection Algorithm: forward
##           nox_kg so2_kg pm10_kg pm25_kg co_kg c6h6_kg nh3_kg ch4_t co2_t n2o_t
## 1  ( 1 )  "*"    " "    " "     " "     " "   " "     " "    " "   " "   " "
## 2  ( 1 )  "*"    " "    " "     " "     " "   " "     " "    "*"   " "   " "
## 3  ( 1 )  "*"    " "    " "     " "     " "   " "     " "    "*"   "*"   " "
## 4  ( 1 )  "*"    " "    "*"     " "     " "   " "     " "    "*"   "*"   " "
## 5  ( 1 )  "*"    " "    "*"     " "     " "   " "     " "    "*"   "*"   "*"
## 6  ( 1 )  "*"    " "    "*"     " "     " "   " "     "*"    "*"   "*"   "*"
## 7  ( 1 )  "*"    " "    "*"     "*"     " "   " "     "*"    "*"   "*"   "*"
## 8  ( 1 )  "*"    "*"    "*"     "*"     " "   " "     "*"    "*"   "*"   "*"
## 9  ( 1 )  "*"    "*"    "*"     "*"     "*"   " "     "*"    "*"   "*"   "*"
## 10  ( 1 ) "*"    "*"    "*"     "*"     "*"   "*"     "*"    "*"   "*"   "*"
```

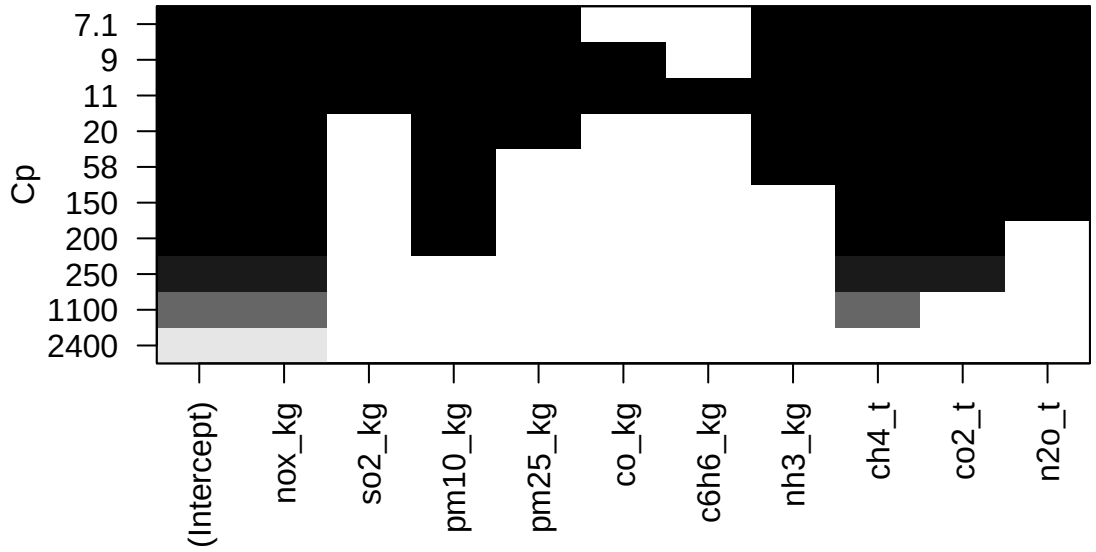
```
plot(choixf,scale="bic")
```




```
plot(choixf,scale="adjr2")
```




```
plot(choixf,scale="Cp")
```



Tous les critères nous donnent le même résultat que la méthode backward pour simplifier le modèle ie retirer les variables Co et C6h6. Nous devons maintenant valider ce sous modèle.

Validation des sous modèles

Nous allons maintenant valider si le sous modèle convient.

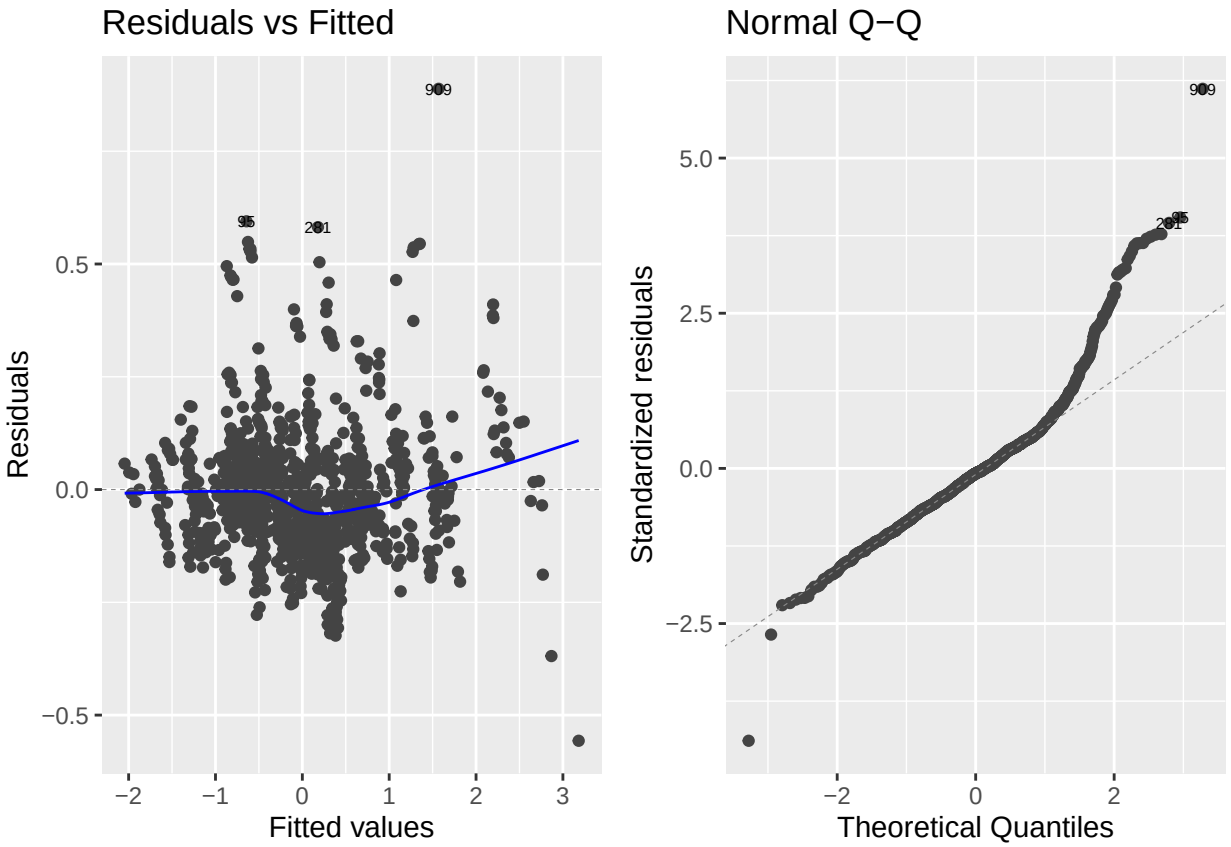
```
reg_simpl=lm(formula=ges_teqco2~nox_kg+so2_kg+pm10_kg+pm25_kg+nh3_kg+ch4_t+co2_t+n2o_t,data=data_scaled
anova(reg_simpl,mod_ges)
```

```
## Analysis of Variance Table
##
## Model 1: ges_teqco2 ~ nox_kg + so2_kg + pm10_kg + pm25_kg + nh3_kg + ch4_t +
##     co2_t + n2o_t
## Model 2: ges_teqco2 ~ nox_kg + so2_kg + pm10_kg + pm25_kg + co_kg + c6h6_kg +
##     nh3_kg + ch4_t + co2_t + n2o_t
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1    950 20.737
## 2    948 20.735  2 0.0024644 0.0563 0.9452
```

En effectuant un test de Fisher de sous model on obtient une pvaleur >0.05 donc on ne rejette pas H0 et on peut simplifier le modèle additif en un sous modèle:

$ges_i = \theta_0 + \theta_1 nox_i + \theta_2 so2_i + \theta_3 pm10_i + \theta_4 pm25_i + \theta_5 nh3_i + \theta_6 ch4_i + \theta_7 co2_i + \theta_8 no2_i + \epsilon$

```
autoplot(reg_simpl,which=c(1,2),label.size=2)
```



Nous faisons ensuite un autoplot, afin de pouvoir vérifier les différentes hypothèses d'un modèle linéaire. Premièrement, les ϵ_i doivent être centré en 0; Quand on regarde le premier graphe, on remarque que les résidus $\hat{\epsilon_i}$ semblent centrés en 0. La deuxième hypothèse nous dit que tous les ϵ_i ont la même variance. Or, tous les individus semblent contenu dans un tube, nous indiquant que cette hypothèse semble vérifier. Ensuite la troisième hypothèse est l'indépendance entre les ϵ_i et Y_i. Dans le premier graphe, il n'y a pas de forme particulière, et les ϵ_i et Y_i semblent donc indépendants. La dernière hypothèse est celle de la normalité des Y_i. En regardant le Q-Q plot, les quantiles empiriques sont plutôt proches des théoriques. Ainsi, les 4 hypothèses sont vérifiées, le modèle linéaire est donc adapter pour réprésenter ces données.

Régréssion régularisé

Nous allons maintenant effectuer une régression régularisée. Cette méthode consiste à changer la fonction à minimiser pour trouver notre estimateurs des paramètres $\hat{\theta}$. Le but de cette méthode est d'obtenir un estimateur certes biaisé, mais qui a une variance plus petite. Il faudrait résoudre $\hat{\theta} = \text{argmin}_{\theta \in \mathbb{R}_k}(\|Y - X\theta\|^2 - \lambda pen(\theta))$. La fonction $\theta \mapsto \text{pen}(\theta)$ dépend du type de régression régularisée. Nous allons voir la régression de Ridge, de Lasso et Elastic Net.

Ridge Commençons par faire une régréssion Ridge. Cette méthode consiste à définir $pen(\theta) = \|\theta\|_2^2$. On commence par calculer la valeurs des coefficients de $\hat{\theta}$ minimisant la fonction pour différentes valeurs de λ, réprésentées dans la figure 1 . Ensuite, nous faisons une validation croisé afin de trouver le λ optimal. En regardant la figure de droite de 1, on remarque que le λ retenu est 0.001380384 . La droite rouge dans la figure de droite est tracé au niveau du lambda optimal, et nous permet de récupérer les coefficients du $\hat{\theta}$ final.

```
## Scale for x is already present.
## Adding another scale for x, which will replace the existing scale.
```

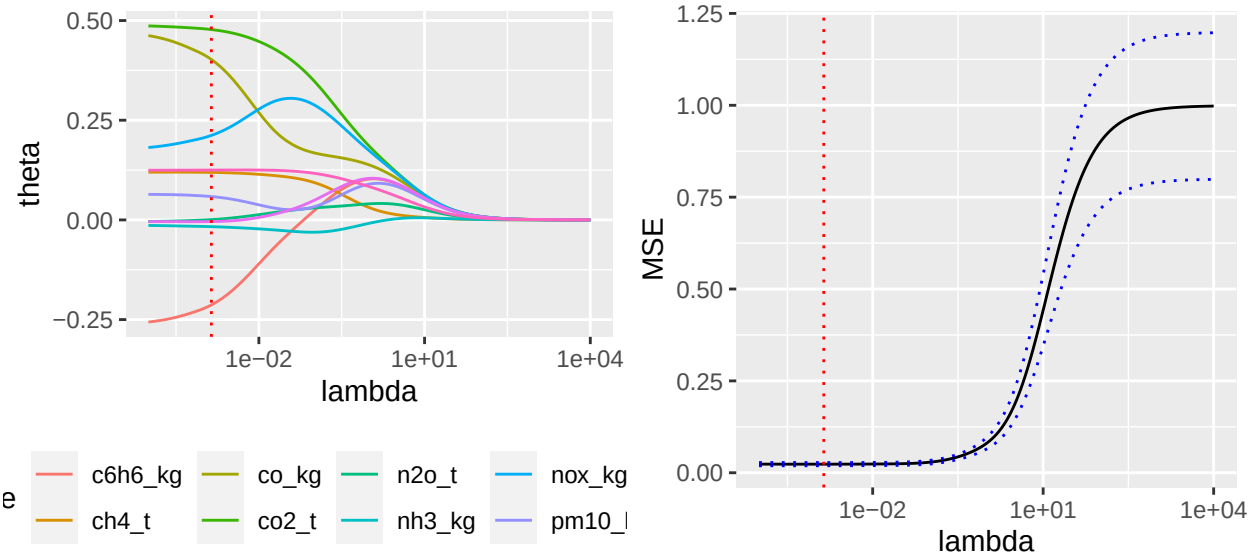


Figure 1: Régularisation Ridge

Lasso On effectue exactement la même procédure, mais avec une régression de Lasso, qui correspond à : $pen(\theta) = \|\theta\|_1$

```
## Scale for x is already present.
## Adding another scale for x, which will replace the existing scale.
```

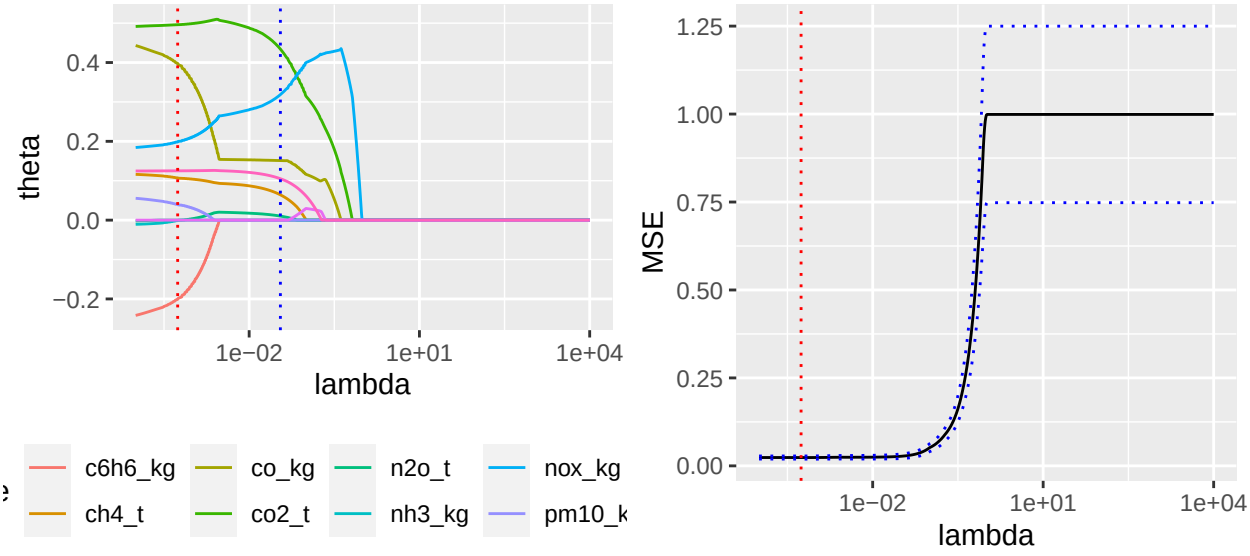
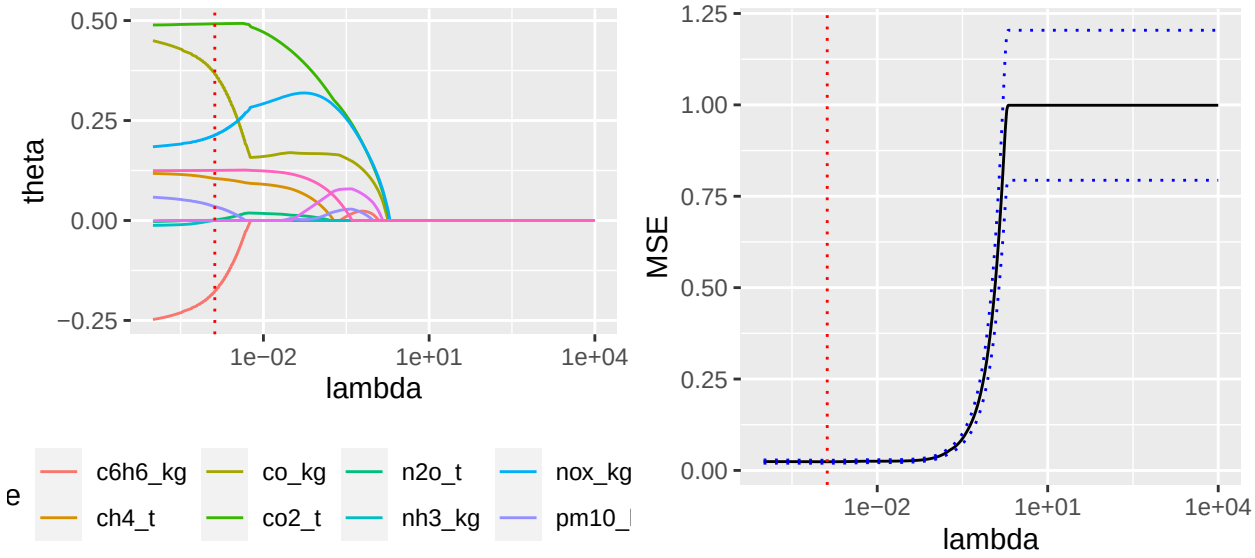


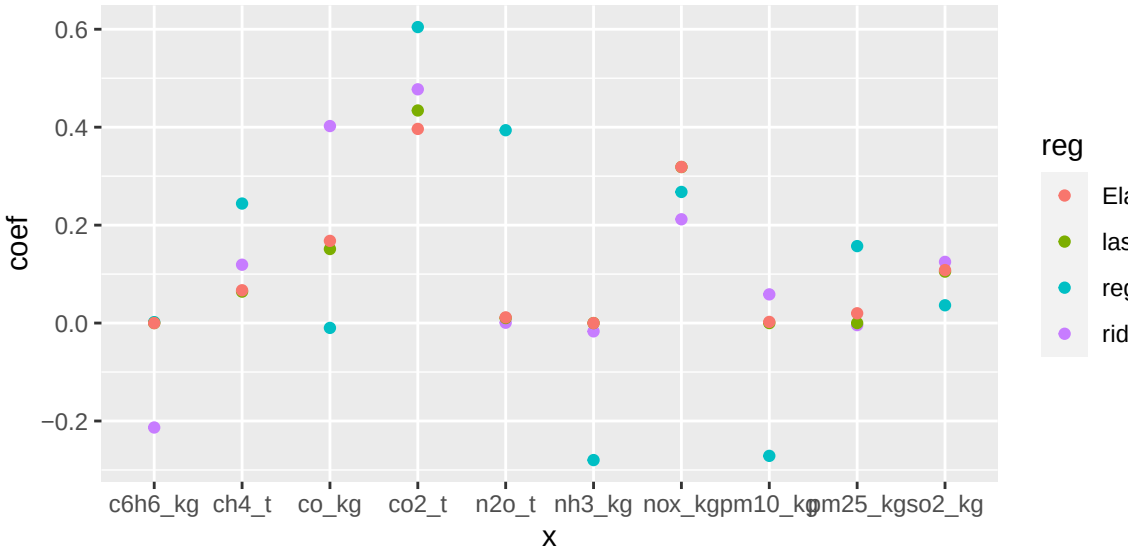
Figure 2: Régularisation Lasso

Notre λ optimal est: 0.0005495409 .



Elastic Net

Le λ optimal pour Elastic Net est de : 0.001318257 .



Analyse des résultats

Pour finir avec cette partie, regardons les valeurs des différents coefficients obtenus à l'aide des méthodes de régréssions. Le premier point à noter, c'est que les trois régressions régularisés donnent des résultats proches, contrairement à la régression linéaire. Précédemment, nous avions vu que les coefficients de c6h6_kg et co_kg, ce que confirme la figure **??**. En effet le point bleu associé à ces variables sont proches de 0. Les régréssions régularisées annuleraient elles aussi le coefficient associé à c6h6_kg, mais pas celui de co_kg. Ces régularisations, surtout ElasticNet et Lasso, ont trouvé des valeurs proches de 0 plutôt les coefficients associés aux variables : n2o_t, nh3_kg, pm10_kg et pm25_kg.