Modèle linéaire généralisé et Choix de modèles

Rachid Rahmani

Plan

1. Objectif du projet

L'objectif du projet consiste à analyser un jeu de données qui contient des informations sur les conditions météorologiques à Bale afin de prédire s'il pleuvra le lendemain.
On cherche à expliquer/prédire la variable d'intérêt 'pluie.demain' à partir des valeurs moyennes, minimale et maximale de :

- Température (°C) : *Temperature.daily.xxx..2.m.above.gnd.*
- Humidite relative (pourcentage) : *Relative.Humidity.daily.xxx..2.m.above.gnd.*
- Presion (hPa) : *Mean.Sea.Level.Pressure.daily.xxx..MSL.*
- Nébulosité (pourcentage) : *Total.Cloud.Cover.daily.xxx..sfc.*
- Nébulosité forte, moyenne et faible : *High.Cloud.Cover.daily.xxx..high.cld.lay. , Medium.Cloud.Cover.daily.xxx..mid.cld.lay., Low.Cloud.Cover.daily.xxx..low.cld.lay.*
- Vitesse (en km/h) et direction (en degrés) du vent à 10 m d'altitude, 80 m d'altitude, et à l'altitude ou la pression vaut 900 hPa : *Wind.Speed.daily.xxx..10.m.above.gnd., Wind.Speed.daily.xxx..900.mb., Wind.Speed.daily.xxx..80.m.above.gnd.,* Wind.Direction.daily.mean..10.m.above.gnd., Wind.Direction.daily.mean..80.m.above.gnd., Wind.Direction.daily.mean..900.mb.
- Rafales de vent à 10 m : *Wind.Gust.daily.xxx..sfc.*

Ainsi qu'aux valeurs totales sur la journée de :

- Précipitations (mm) : *Total.Precipitation.daily.sum..sfc.*
- Neige (cm) : *Snowfall.amount.raw.daily.sum..sfc.*
- Minutes d'ensoleillement : *Shortwave.Radiation.daily.sum..sfc.*
- Rayonnement solaire (W/m2) : *Sunshine.Duration.daily.sum..sfc.*

On a à notre disposition deux jeux de données :

- Un jeu d'entrainement qui nous permettra de sélectionner et d'entrainer notre modèle.
- Un jeu de test qui sera utilisé pour évaluer notre modèle.

2. Exploration préalable

2.1 Chargement et visualisation des données

Après avoir chargé nos données, on visualise les premières lignes et on affiche un résumé des données.

```
> summary(meteo.train$pluie.demain)
   Mode   FALSE    TRUE
logical     579     601
```

- On voit que notre variable d'intérêt 'pluie.demain' est encodé sous forme de booléen.
- Les variables 'Hour' et 'Minute' ont les mêmes valeurs pour toutes les observations, elles n'apportent donc aucune information.
- La variable 'Day' ne peut pas aussi apporter d'information sur la pluie le lendemain ou pas. De même pour la première colonne 'X' qui semble être unique pour toutes les lignes.

On ignorera donc pour la suite de notre analyse les co-variables suivantes :

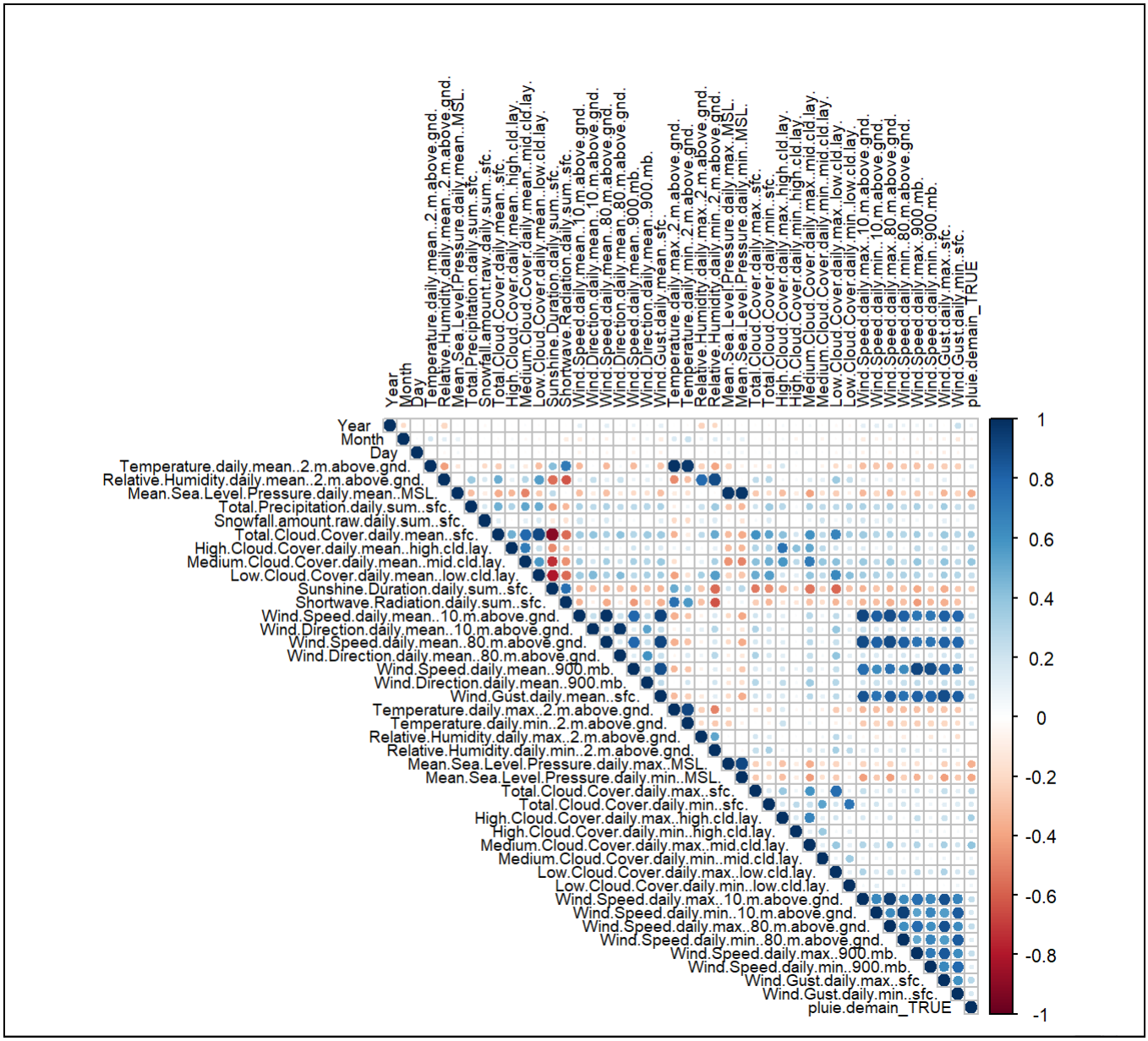
Column1	Year	Month	Day	Hour	Minute

2.2 Etude des corrélations

Avant de commencer notre modélisation on affiche la corrélation via un corrplot après avoir encodé notre variable pluie.demain sous forme d'entier qui prend la valeur 0 ou 1 selon que pluie.demain soit False ou True.

L'objectif de cette étude des corrélations est de :

- Comprendre les relations entre les variables.
- Détecter la multi colinéarité qui résulterait de deux ou plusieurs variables fortement corrélées entre elles car cela peut poser des problèmes en rendant les estimations des coefficients instables et difficiles à interpréter.



Du fait de la taille des intitulés et du nombre de covariables, le corrélogramme est difficilement exploitable graphiquement. En complément une fonction est implémentée pour identifier les variables fortement corrélées : On affiche ci-dessous les variables dont la corrélation est supérieure en valeur absolu à 85%

col1	col2	abs correl
Wind.Speed.daily.mean..10.m.above.gnd.	Wind.Speed.daily.mean..80.m.above.gnd.	98%
Temperature.daily.mean..2.m.above.gnd.	Temperature.daily.max..2.m.above.gnd.	98%
Wind.Direction.daily.mean..10.m.above.gnd.	Wind.Direction.daily.mean..80.m.above.gnd.	97%
Mean.Sea.Level.Pressure.daily.mean..MSL.	Mean.Sea.Level.Pressure.daily.min..MSL.	97%
Mean.Sea.Level.Pressure.daily.mean..MSL.	Mean.Sea.Level.Pressure.daily.max..MSL.	97%
Temperature.daily.mean..2.m.above.gnd.	Temperature.daily.min..2.m.above.gnd.	97%
Wind.Speed.daily.max..10.m.above.gnd.	Wind.Speed.daily.max..80.m.above.gnd.	95%

Wind.Speed.daily.min..10.m.above.gnd.	Wind.Speed.daily.min..80.m.above.gnd.	93%
Wind.Speed.daily.mean..10.m.above.gnd.	Wind.Gust.daily.mean..sfc.	92%
Wind.Speed.daily.mean..80.m.above.gnd.	Wind.Gust.daily.mean..sfc.	92%
Wind.Speed.daily.mean..10.m.above.gnd.	Wind.Speed.daily.max..10.m.above.gnd.	92%
Wind.Speed.daily.mean..900.mb.	Wind.Speed.daily.max..900.mb.	92%
Temperature.daily.max..2.m.above.gnd.	Temperature.daily.min..2.m.above.gnd.	91%
Total.Cloud.Cover.daily.mean..sfc.	Sunshine.Duration.daily.sum..sfc.	91%
Mean.Sea.Level.Pressure.daily.max..MSL.	Mean.Sea.Level.Pressure.daily.min..MSL.	90%
Total.Cloud.Cover.daily.mean..sfc.	Low.Cloud.Cover.daily.mean..low.cld.lay.	90%
Wind.Speed.daily.mean..80.m.above.gnd.	Wind.Speed.daily.max..10.m.above.gnd.	90%
Wind.Speed.daily.mean..80.m.above.gnd.	Wind.Speed.daily.max..80.m.above.gnd.	90%
Wind.Speed.daily.mean..900.mb.	Wind.Speed.daily.min..900.mb.	90%
Relative.Humidity.daily.mean..2.m.above.gnd.	Relative.Humidity.daily.min..2.m.above.gnd.	89%
Wind.Speed.daily.mean..900.mb.	Wind.Gust.daily.mean..sfc.	89%
Wind.Gust.daily.mean..sfc.	Wind.Gust.daily.max..sfc.	89%
Wind.Speed.daily.mean..10.m.above.gnd.	Wind.Speed.daily.max..80.m.above.gnd.	89%
Wind.Speed.daily.max..80.m.above.gnd.	Wind.Gust.daily.max..sfc.	87%
Wind.Speed.daily.max..10.m.above.gnd.	Wind.Gust.daily.max..sfc.	87%
Wind.Gust.daily.mean..sfc.	Wind.Speed.daily.max..10.m.above.gnd.	87%

La variable ‘Mean.Sea.Level.Pressure.daily.max..MSL.’ est fortement corrélee à:

- ‘Mean.Sea.Level.Pressure.daily.min..MSL.’ : correl 90%
- ‘Mean.Sea.Level.Pressure.daily.mean..MSL.’ : correl 97%

La variable ‘Relative.Humidity.daily.mean..2.m.above.gnd.’ est fortement corrélee à:

- ‘Relative.Humidity.daily.min..2.m.above.gnd.’ : correl 89%

La variable ‘Temperature.daily.max..2.m.above.gnd.’ est fortement corrélee à:

- ‘Temperature.daily.min..2.m.above.gnd.’ : correl 91%
- ‘Temperature.daily.mean..2.m.above.gnd.’ : correl 98%

La variable ‘Total.Cloud.Cover.daily.mean..sfc.’ est fortement corrélee à:

- ‘Sunshine.Duration.daily.sum..sfc.’ : correl 91%
- ‘Low.Cloud.Cover.daily.mean..low.cld.lay.’ : correl 90%

La variable ‘Wind.Direction.daily.mean..10.m.above.gnd.’ est fortement corrélee à:

- ‘Wind.Direction.daily.mean..80.m.above.gnd.’ : correl 97%

La variable ‘Wind.Gust.daily.mean..sfc.’ est fortement corrélée à :

- ‘Wind.Gust.daily.max..sfc.’ : correl 89%

- ‘Wind.Speed.daily.max..10.m.above.gnd.’: correl 87%
- ‘Wind.Speed.daily.mean..10.m.above.gnd.’: correl 92%
- ‘Wind.Speed.daily.mean..80.m.above.gnd.’: correl 92%
- ‘Wind.Speed.daily.mean..900.mb.’: correl 89%

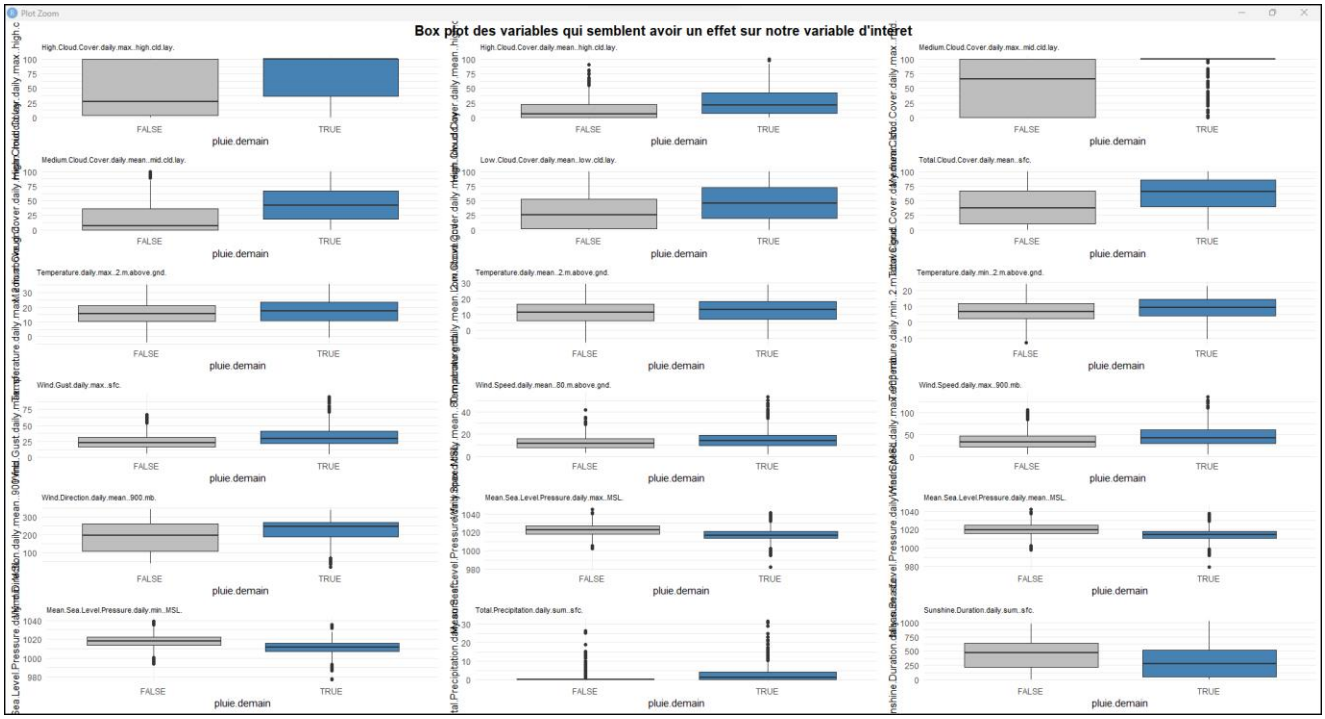
La variable ‘Wind.Speed.daily.max..900.mb.’ est fortement corrélée à :
- ‘Wind.Gust.daily.max..sfc.’ : correl 92%
- ‘Wind.Speed.daily.min..900.mb.’: correl 90%

La variable ‘Wind.Speed.daily.min..10.m.above.gnd.’ est fortement corrélée à :
- ‘Wind.Speed.daily.min..80.m.above.gnd.’ : correl 93%

Ces informations seront utiles pour la suite afin de réduire le nombre de variables que l’on considèrera dans notre régression.

2.3 BoxPlot

Nous avons représenté graphiquement l’influence de chacune des covariables sur notre variable d’intérêt ‘pluie.demain’. Pour ne pas encombrer le rapport nous n’afficherons les box plot que pour les variables qui visuellement semblent avoir un effet significatif sur la variable ‘pluie.demain’. Les conclusions de cette analyse graphique seront utiles pour la suite afin de sélectionner les variables que l’on retiendra dans la construction de notre modèle.



En conclusion il s’agit des variables suivantes :

- "High.Cloud.Cover.daily.max..high.cld.lay."

- "High.Cloud.Cover.daily.mean..high.cld.lay."
- "Medium.Cloud.Cover.daily.max..mid.cld.lay."
- "Medium.Cloud.Cover.daily.mean..mid.cld.lay."
- "Low.Cloud.Cover.daily.mean..low.cld.lay."
- "Total.Cloud.Cover.daily.mean..sfc."
- "Temperature.daily.max..2.m.above.gnd."
- "Temperature.daily.mean..2.m.above.gnd."
- "Temperature.daily.min..2.m.above.gnd."
- "Wind.Gust.daily.max..sfc."
- "Wind.Speed.daily.mean..80.m.above.gnd."
- "Wind.Speed.daily.max..900.mb."
- "Wind.Direction.daily.mean..900.mb."
- "Mean.Sea.Level.Pressure.daily.max..MSL."
- "Mean.Sea.Level.Pressure.daily.mean..MSL."
- "Mean.Sea.Level.Pressure.daily.min..MSL."
- "Total.Precipitation.daily.sum..sfc."
- "Sunshine.Duration.daily.sum..sfc."

2.4 Conclusion

Pour conclure en se basant sur une interprétation purement visuelle on pourrait considérer un modèle qui ne considèrerait que les variables suivantes pour expliquer/prédire la variable 'pluie.demain':

- "High.Cloud.Cover.daily.max..high.cld.lay."
- "High.Cloud.Cover.daily.mean..high.cld.lay."
- "Medium.Cloud.Cover.daily.max..mid.cld.lay."
- "Medium.Cloud.Cover.daily.mean..mid.cld.lay."
- "Low.Cloud.Cover.daily.mean..low.cld.lay."
- "Total.Cloud.Cover.daily.mean..sfc."
- "Temperature.daily.mean..2.m.above.gnd."
- "Wind.Gust.daily.max..sfc."
- "Wind.Speed.daily.mean..80.m.above.gnd."
- "Wind.Speed.daily.max..900.mb."
- "Wind.Direction.daily.mean..900.mb."
- "Mean.Sea.Level.Pressure.daily.max..MSL."
- "Total.Precipitation.daily.sum..sfc."
- "Sunshine.Duration.daily.sum..sfc."

3. Construction du modèle

Notre variable d'intérêt 'pluie.demain' est de type binaire, elle suit donc une loi de Bernoulli de paramètre p_i.On utilisera un modèle linéaire généralisé GLM pour la modéliser avec comme fonction de lien la fonction logit.

$$\ln\left(\frac{p_i}{1-p_i}\right) = \beta.X_i$$

Le vecteur de paramètres β : $[\beta_0, \ldots., \beta_k]$ est estimé en maximisant la vraisemblance au vu des n observations de notre variable d'intérêt 'pluie.demain'.

Sous R, on utilise la fonction glm avec le paramètre familiy=binomial.

On commence par construire un premier modèle qui contient l'intégralité des covariables à l'exception des colonnes : X, Day, Hour, Minute et Year

```
model_0 = glm(pluie.demain ~  . -X -Day -Hour - Minute - Year, data = meteo.train, family = binomial)
```

On utilisera deux approches pour construire notre modèle :

- Une première approche base sur le choix automatique de modèle qui utilise la fonction step
- Une approche manuelle

3.1 Choix de modèle : Utilisation de la fonction Step

Trois méthodes automatiques sont à notre disposition via l'utilisation de la fonction R step:

- Méthode ascendante (forward selection) : A chaque pas, une variable est ajoutée au modèle, celle qui a l'apport le plus important (que l'on peut mesurer pour un test par celle qui a la plus petite p-valeur)
- Méthode descendante (backward selection) : A chaque pas, une variable est enlevée au modèle, celle qui le plus fort impact (que l'on peut mesurer pour un test par celle qui a la plus grande p-valeur)
- Méthode progressive (stepwise selection) : C'est la même méthode que la méthode ascendante, à l'exception qu'à chaque étape, on peut remettre en cause une variable présente dans le modèle selon la méthode descendante.

	model_step_forward	model_step_backward	model_step_mixed
#parametres	42	18	18
AIC	1,322.54	1,285.60	1,285.60
BIC	1,535.62	1,376.91	1,376.91

L'approche forward ne semble pas satisfaisante, toutes les covariables sont retenus et les critères d'information AIC et BIC sont tous les deux supérieurs à celui des modèles backward et mixed.
Les méthodes backward et mixed donnent le même modèle ou 18 paramètres sont retenus.

Une analyse de variance pour comparer notre modele complet au modele model_step_backward/model_step_mixed indique qu'il n'y a pas de differences significatives entre nos deux modèles. Autrement dit, notre second modèle explique tout autant de variance que notre premier modèle, tout en étant plus parcimonieux.

```
> anova(model_0, model_step_backward)
Analysis of Deviance Table
....
  Resid. Df Resid. Dev  Df Deviance Pr(>Chi)
1      1138     1238.5
2      1162     1249.6 -24  -11.051   0.9886
```

Analyse de la déviance de notre modèle M_k

On utilise un test de déviance pour comparer notre modèle M_k à deux modèles extrêmes :

- Le modèle nul M_{null} dans lequel la variable 'pluie.demain' est iid c'est-à-dire $p_i = p$, ce modele est equivalent à $\beta_1 = \beta_2 = \cdots = \beta_k = 0$ et qui a donc un seul degré de liberté.

  La statistique de test sera $D_0 - D_k = -2ln\frac{vraisemblance\ M_{null}}{vraisemblance\ M_k}$ et sous l'hypothése que M_0est le vrai modèle on a $D_0 - D_k \sim Khi_k^2$

  Avec D_0 appelé Déviance Nulle et D_kDéviance Residuelle.

- Le modèle saturé $M_{saturé}$ dans lequel il n'y a aucune structure au $p_i$et qui a donc n degrés de liberté, n étant le nombre d'observation de 'pluie.demain'.

  La statistique de test appelé Déviance Résiduelle sera $D_k = -2ln\frac{vraisemblance\ M_k}{vraisemblance\ M_{saturé}}$

  et sous l'hypothése que $M_k$est le vrai modèle on a $D_k \sim Khi_{n-k-1}^2$

On voit que ces trois modèles sont imbriqués : $M_{null} \subset M_k \subset M_{saturé}$

Analyse de la déviance

Analyse de la déviance de notre modèle model_step_forward

La sortie summary indique :

```
    Null deviance: 1635.4  on 1179  degrees of freedom
Residual deviance: 1238.5  on 1138  degrees of freedom
```

(i) Le test de notre modèle par rapport au modèle sans covariable donne

```
> # Le test par rapport au modèle sans covariable donne
> pchisq(1635.4 - 1238.5, 1179 - 1138, lower = F)
[1] 8.515678e-60
```

La p-value étant très faible on rejette le modèle nul.

(ii) Le test de notre modèle par rapport au modèle saturé donne

```
> # Le test par rapport au modèle saturé donne
> pchisq(1238.5, 1138, lower = F)
[1] 0.01962949
```

La p-value étant faible on rejette notre modèle et on prefere le modèle saturé. Autrement dit, notre modèle n'est pas suffisant.

Analyse de la déviance de notre modèle model_step_backward/model_step_mixed
La sortie summary indique :

```
    Null deviance: 1635.4  on 1179  degrees of freedom
Residual deviance: 1249.6  on 1162  degrees of freedom
AIC: 1285.6
```

(i) Le test de notre modèle par rapport au modèle sans covariable donne

```
> # Le test par rapport au modèle sans covariable donne
> pchisq(1635.4 - 1249.6, 1179 - 1162, lower = F)
[1] 1.716014e-71
```

La p-value étant très faible on rejette le modèle nul.

(ii) Le test de notre modèle par rapport au modèle saturé donne

```
> # Le test par rapport au modèle saturé donne
> pchisq(1249.6, 1162, lower = F)
[1] 0.036959
```

La p-value étant faible on rejette notre modèle et on préfère le modèle saturé. Autrement dit, notre modèle n'est pas suffisant.

En conclusion l'approche de selection avec la fonction Step n'est pas satisfaisante, même dans le cas ou toutes les covariables ont été retenus : Il y a potentiellement deux raisons ou qu'il reste de la variation à expliquer ou que les hypothèses du modèles ne sont pas vérifiés.

Je suspecte de la multi colinéarité je repars du model initial model_0 auquel je vais appliqué itérativement la fonction VIF pour détecter et enlever itérativement toutes les variables avec un VIF au-dessus de 7. Après plusieurs incréments j'obtiens le modèle suivant :

```
model_0_vif = glm(
  pluie.demain ~  . - X - Day - Hour - Minute - Year
  - Temperature.daily.mean..2.m.above.gnd.
  - Mean.Sea.Level.Pressure.daily.mean..MSL.
  - Wind.Speed.daily.mean..10.m.above.gnd.
  - Wind.Speed.daily.mean..900.mb.
  - Wind.Direction.daily.mean..80.m.above.gnd.
  - Total.Cloud.Cover.daily.mean..sfc.
  - Wind.Speed.daily.mean..80.m.above.gnd.
  - Temperature.daily.max..2.m.above.gnd.
  - Relative.Humidity.daily.mean..2.m.above.gnd.
  - Wind.Gust.daily.mean..sfc.
  - Wind.Speed.daily.max..10.m.above.gnd.
  - Wind.Speed.daily.min..10.m.above.gnd.
  - Sunshine.Duration.daily.sum..sfc.,
  data = meteo.train,
  family = binomial
)
```

Je pars de ce modèle pour utiliser la fonction step de sélection de modèle.

	model_step_vif_cleaned_forward	model_step_backward	model_step_mixed
#parametres	28	15	15
AIC	1332.7	1312.1	1312.1

Analyse de la déviance de notre modèle model_step_vif_cleaned_forward

La sortie summary indique :

```
    Null deviance: 1635.4  on 1179  degrees of freedom
Residual deviance: 1274.7  on 1151  degrees of freedom
```

(i) Le test de notre modèle par rapport au modèle sans covariable donne

```
> # Le test par rapport au modèle sans covariable donne
> pchisq(1635.4 - 1274.7, 1179 - 1151, lower = F)
[1] 1.747878e-59
```

La p-value étant très faible on rejette le modèle nul.

(ii) Le test de notre modèle par rapport au modèle saturé donne

```
> # Le test par rapport au modèle saturé donne
> pchisq(1274.7, 1151, lower = F)
[1] 0.006122457
```

La p-value étant faible on rejette notre modèle et on préfère le modèle saturé. Autrement dit, notre modèle n'est pas suffisant.

Analyse de la déviance de notre modèle model_step_backward/model_step_mixed

Comme précédemment l'approche backward et mixed donne le même modele

La sortie summary indique :

```
    Null deviance: 1635.4  on 1179  degrees of freedom
Residual deviance: 1282.1  on 1165  degrees of freedom
AIC: 1312.1
```

(i) Le test de notre modèle par rapport au modèle sans covariable donne

```
> # Le test par rapport au modèle sans covariable donne
> pchisq(1635.4 - 1282.1, 1179 - 1165, lower = F)
[1] 8.35888e-67
```

La p-value étant très faible on rejette le modèle nul.

(ii) Le test de notre modèle par rapport au modèle saturé donne

```
> pchisq(1282.1, 1165, lower = F)
[1] 0.009068516
```

La p-value étant faible on rejette notre modèle et on préfère le modèle saturé. Autrement dit, notre modèle n'est pas suffisant.

3.2 Choix de modèle : Approche manuelle

On se base dans ce cas-là sur les conclusions de notre analyse graphique, on crée le modèle suivant

```
model_manuel = glm(
  pluie.demain ~  High.Cloud.Cover.daily.max..high.cld.lay.
  + High.Cloud.Cover.daily.mean..high.cld.lay.
  + Medium.Cloud.Cover.daily.max..mid.cld.lay.
  + Medium.Cloud.Cover.daily.mean..mid.cld.lay.
  + Low.Cloud.Cover.daily.mean..low.cld.lay.
  + Total.Cloud.Cover.daily.mean..sfc.
  + Temperature.daily.mean..2.m.above.gnd.
  + Wind.Gust.daily.max..sfc.
  + Wind.Speed.daily.mean..80.m.above.gnd.
  + Wind.Speed.daily.max..900.mb.
  + Wind.Direction.daily.mean..900.mb.
  + Mean.Sea.Level.Pressure.daily.max..MSL.
  + Total.Precipitation.daily.sum..sfc.
  + Sunshine.Duration.daily.sum..sfc.,
  data = meteo.train,
  family = binomial
)
```

Analyse de la déviance de notre modèle model_ manuel

La sortie summary indique :

```
    Null deviance: 1635.4  on 1179  degrees of freedom
Residual deviance: 1300.2  on 1165  degrees of freedom
AIC: 1330.2
```

(i) Le test de notre modèle par rapport au modèle sans covariable donne

```
> # Le test par rapport au modèle sans covariable donne
> pchisq(1635.4 - 1300.2, 1179 - 1165, lower = F)
[1] 5.20344e-63
```

La p-value étant très faible on rejette le modèle nul.

(ii) Le test de notre modèle par rapport au modèle saturé donne

```
> # Le test par rapport au modèle saturé donne
> pchisq(1300.2, 1165, lower = F)
[1] 0.003333542
```

La p-value étant faible on rejette notre modèle et on préfère le modèle saturé. Autrement dit, notre modèle n'est pas suffisant.

Anova

On va maintenant faire un test Anova pour comparer notre modèle au modèle complet :

```
> anova(model_manuel, model_0, test="LRT")
…
  Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
1      1165     1300.2
2      1138     1238.5 27   61.678 0.0001583 ***
```

Le modèle complet explique mieux les données que notre modèle qui s'est basé sur nos conclusions de l'analyse graphique.

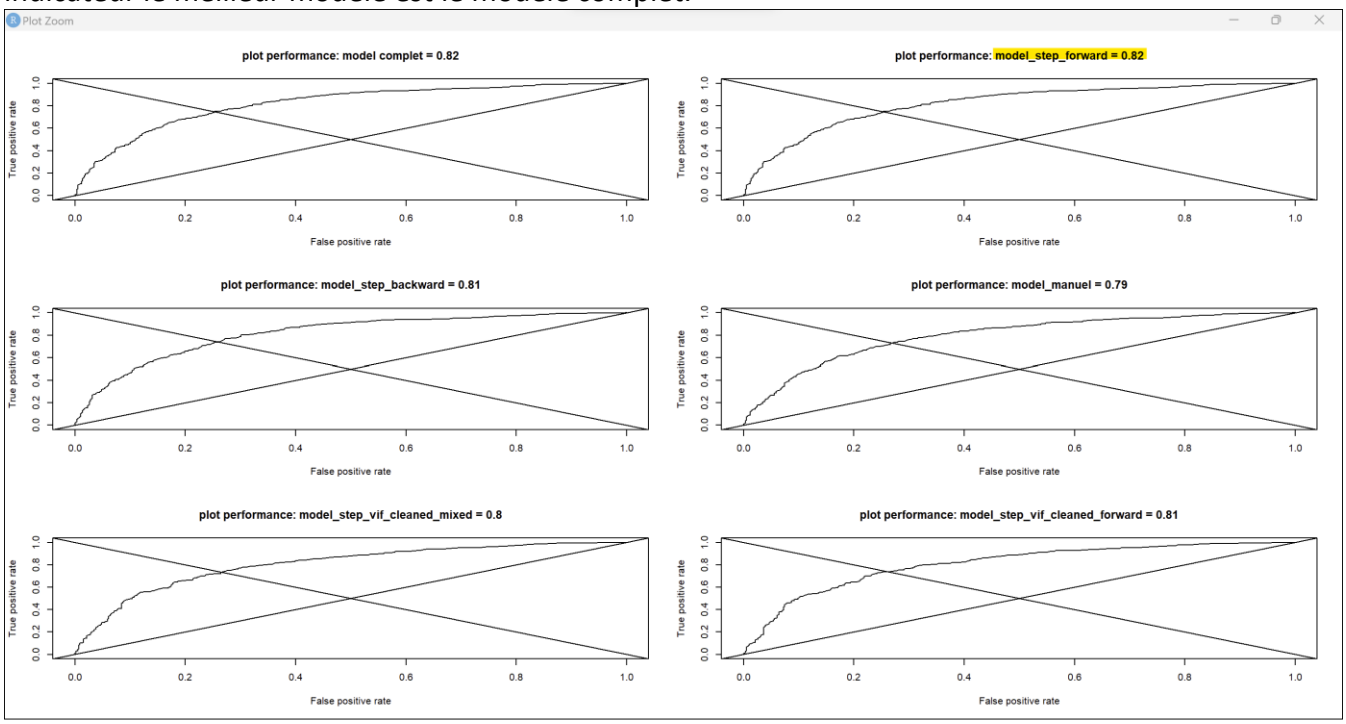
4. Sélection du modèle

On a construit six modèles qui ont un taux de prédiction avec un seuil de décision fixé à 0.5 comparable.

```
> mean(pred_model_0 == meteo.train$pluie.demain)
[1] 0.7474576
> mean(pred_model_0 == meteo.train$pluie.demain)
[1] 0.7474576
> mean(pred_model_step_forward == meteo.train$pluie.demain)
[1] 0.7474576
> mean(pred_model_step_backward == meteo.train$pluie.demain)
[1] 0.7423729
> mean(pred_model_manuel == meteo.train$pluie.demain)
[1] 0.7305085
> mean(pred_model_step_vif_cleaned_mixed == meteo.train$pluie.demain)
[1] 0.7364407
> mean(pred_model_step_vif_cleaned_forward == meteo.train$pluie.demain)
[1] 0.7338983
```

Courbe ROC de nos modèles

On affiche ci-dessous les courbes ROC pour chacun des modèles construit. La courbe ROC permet de mesurer la performance de chacun des modèles qu'on a construit en affichant le taux de vrais positifs en fonction du taux de faux positifs.
Pour chacun des modèles construit l'AUC est supérieur à 0.5, c'est une bonne nouvelle, les modèles qu'on a défini font mieux qu'une classification aléatoire et sur la base de cet indicateur le meilleur modèle est le modèle complet.



Validation K Fold – fonction de lien logit et progit

On va utiliser une validation croisée k-fold avec k égale à 10 sur chacun des modèles défini précédemment en utilisant cette fois la fonction de lien logit et progit pour déterminer le meilleur modèle.

Cette technique consiste à

- Séparer les données en 2 ensembles, un ensemble d'entrainement et un ensemble de test.
- Obtenir nos paramètres β_ià partir de notre ensemble d'entrainement.
- Utiliser les paramètres $\beta_i$pour faire de la prédiction sur notre ensemble de test
- Mesurer la qualité de la prédiction, plusieurs mesures de qualité sont possibles
    - Une perte 0-1 ou on fait une prédiction binaire
    - Une perte L^1ou on mesure $|y_i - p_i|$
    - Une perte L^2ou on mesure $(y_i - p_i)^2$

```
# Validation croisée k-fold
k = 10
index = sample(1:k, nrow(meteo.train), replace=T)

res.model_0.logistique = rep(NA, k)
….
res.model_step_vif_cleaned_forward.probit = rep(NA, k)


ComputeReg = function(reg_model, link_function, i) {
  reg = glm(formula(reg_model),
          family = binomial(link = link_function),
          data = meteo.train[index != i, ])

  pred.reg = predict(reg, newdata=meteo.train[index == i, ],
                     type="response")


  erreur_l1 = mean(abs(pred.reg - meteo.train[index==i, "pluie.demain"]), na.rm = T)

  return(erreur_l1)
}


for(i in 1:k){

  res.model_0.logistique[i] = ComputeReg(model_0, "logit", i)
  res.model_0.probit[i] = ComputeReg(model_0, "probit", i)
…
```

Les résultats de cette validation K fold ou j'ai utilisé l'erreur L1 sont présentés ci-dessous :

model	erreur L1
res.model_0.logistique	0.3649124
res.model_0.probit	0.3669295
res.model_step_forward.logistique	0.3649124
res.model_step_forward.probit	0.3669295
res.model_step_backward.logistique	0.3600312
res.model_step_backward.probit	0.3623495
res.model_manuel.logistique	0.3761442
res.model_manuel.probit	0.3772884
res.model_step_vif_cleaned_mixed.logistique	0.3699203
res.model_step_vif_cleaned_mixed.probit	0.3713642
res.model_step_vif_cleaned_forward.logistique	0.3712959
res.model_step_vif_cleaned_forward.probit	0.3729108

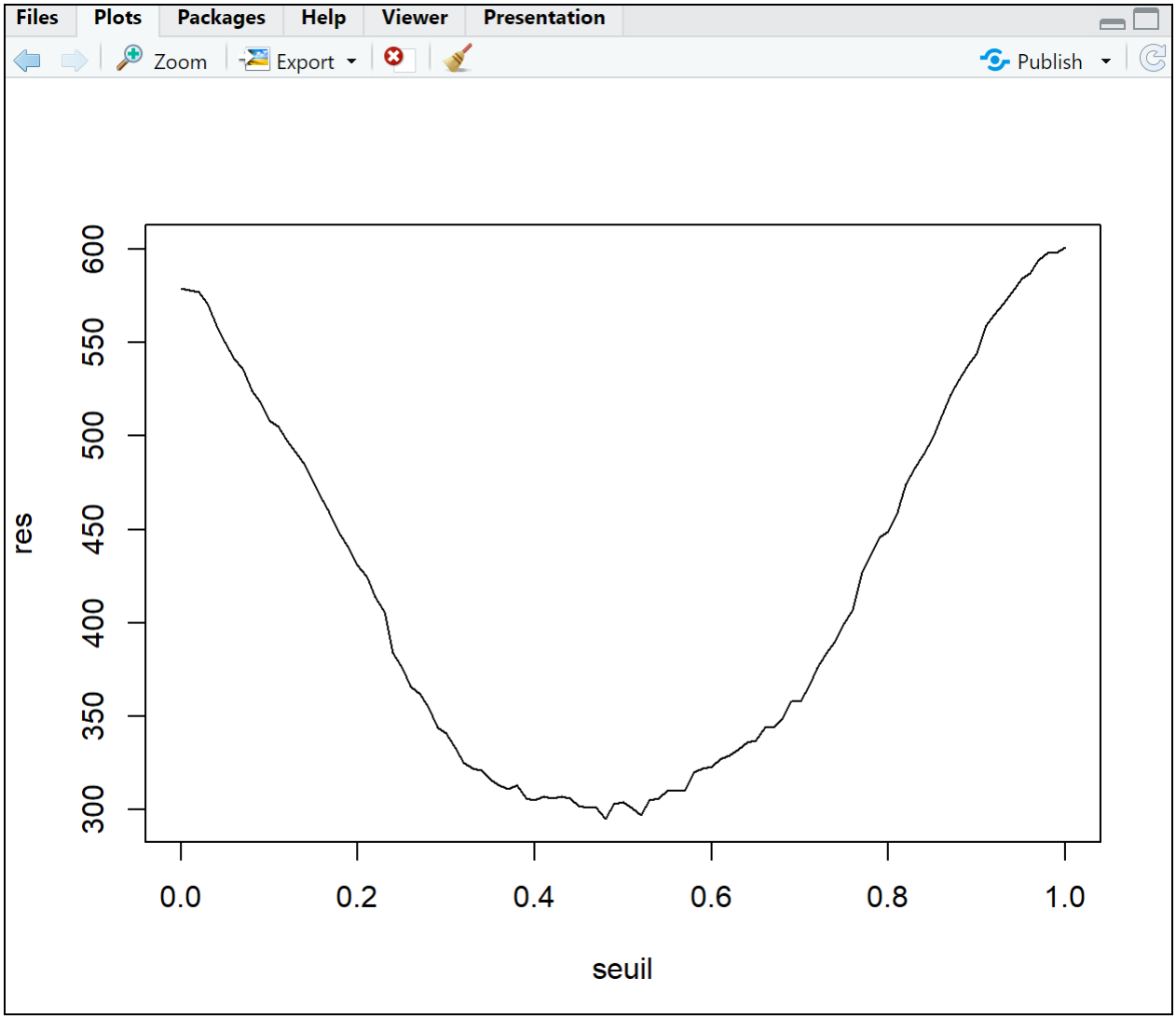
Le modèle le plus performant est le modele `model_step_backward` avec comme fonction de lien la fonction logit.

Détermination du seuil optimal

On va maintenant déterminer le seuil optimal sur le modèle précédemment sélectionné.
Selon le contexte on aurait pu faire le choix de pénaliser plus les faux positifs ou les faux négatifs, ceci étant on considérera à égalité les faux positifs et les faux négatifs.

Le seuil optimal est à 0.48

```
> seuil[which.min(res)]
[1] 0.48
```



5. Generation des predictions

A l'aide du modèle et du seuil de décision sélectionné précédemment on va générer sur le jeu de données de test nos prédictions.
Les prédictions sont disponibles dans le fichier 'prediction_RAHMANI_Rachid.csv'.

```
# Generation des predictions
pred.meteo.test = predict(model_step_backward, newdata = meteo.test, type = "response")
pluie.demain = (pred.meteo.test>=seuil[which.min(res)])
prediction <- cbind(meteo.test, pluie.demain)
View(prediction)
write.table(prediction,"prediction_RAHMANI_Rachid.csv",sep=";",col.names=TRUE)
```

6. Conclusion

Après avoir chargé les données on a fait une exploration dont la conclusion a été reprise pour la création de notre modèle manuel.
Plusieurs modèles ont été considéré, et pour aucun d'entre eux le test déviance résiduelle n'a permis d'accepter le modèle en comparaison du modèle saturé.
Cela signifie qu'aucun des modèles n'est satisfaisant : soit parce que les hypothèses du modèle ne sont pas vérifiées ou soit parce qu'il manque des covariables.
On voit toute la complexité de la prédiction météorologique ou même le modèle complet qui contient un très grand nombre de variables n'est pas satisfaisant lorsqu'on le compare au modèle saturé.
On a ensuite utilisé la validation croisée pour comparer les différent modèles et on a choisi comme 'meilleur modèle' celui qui minimise l'erreur de prédiction au sens L1 . A partir ce meilleur modèle après avoir déterminé le seuil optimal de décision, on a fait de la prédiction sur notre jeu de test.