

Rapport : Modèles Linéaires Généralisés & Choix de Modèles

Sophie ROBERT OKADA

2024-06-30

1. Introduction

Dans le cadre du projet sur les données météorologiques à Bâle, nous cherchons à prédire s'il pleuvra le lendemain à partir de mesures journalières recueillies entre 2010 et 2018. La variable à prédire "pluie.demain" Y_i est de type binaire où $Y_i \in \{0, 1\}$, et les variables explicatives comprennent des données de température, d'humidité, de pression, de nébulosité, de vitesse et direction des vents, de précipitations et d'ensoleillement.

Pour prédire correctement notre variable d'intérêt, nous appliquerons un modèle linéaire généralisé particulièrement adapté pour gérer une réponse binaire. Un modèle linéaire généralisé est défini par le choix de la famille de lois aléatoires L et par la fonction de lien g , où $g(E[Y_i]) = \beta_0 + \beta_1 x_i$. Pour la variable dépendante Y_i , représentant la pluie le lendemain, nous appliquerons deux approches principales : la régression logistique dans un premier temps, suivi d'un modèle de régression probit dans une deuxième temps, afin de comparer leurs efficacités et précisions dans la prédiction des jours de pluie survenant le lendemain.

1.1 Chargement des packages utiles

Pour réaliser l'analyse, nous commencerons tout d'abord par charger les packages utiles suivants:

```
rm(list = ls())
invisible(suppressMessages(lapply(c("readr", "corrplot", "forecast",
  "car", "stats", "ggplot2", "kableExtra", "tidyverse", "gridExtra",
  "grid", "dplyr", "viridis", "summarytools", "reshape2", "knitr",
  "MASS", "ROCR"), library, character.only = TRUE)))
```

1.2 Chargement & Visualisation des données

Nous allons ensuite charger les données d'entraînement "meteo.train", puis nous allons exclure la première variable nommée X qui est juste l'indice du nombre de données.

```
meteo_train <- read_csv("C:/Documents/Dauphine/Module 2/Modèle Linéaire Généralisé/Projet/meteo.train.csv",
  show_col_types = FALSE, name_repair = "minimal")
meteo_train <- meteo_train[, -1]
```

Comme les noms des variables semblent difficiles à lire, et pour obtenir plus de clarté dans notre analyse, nous commencerons par renommer les variables. Les variables renommées sont présentées dans le tableau ci-dessous.

Table 1: Nouveaux Noms de variables meteo_train

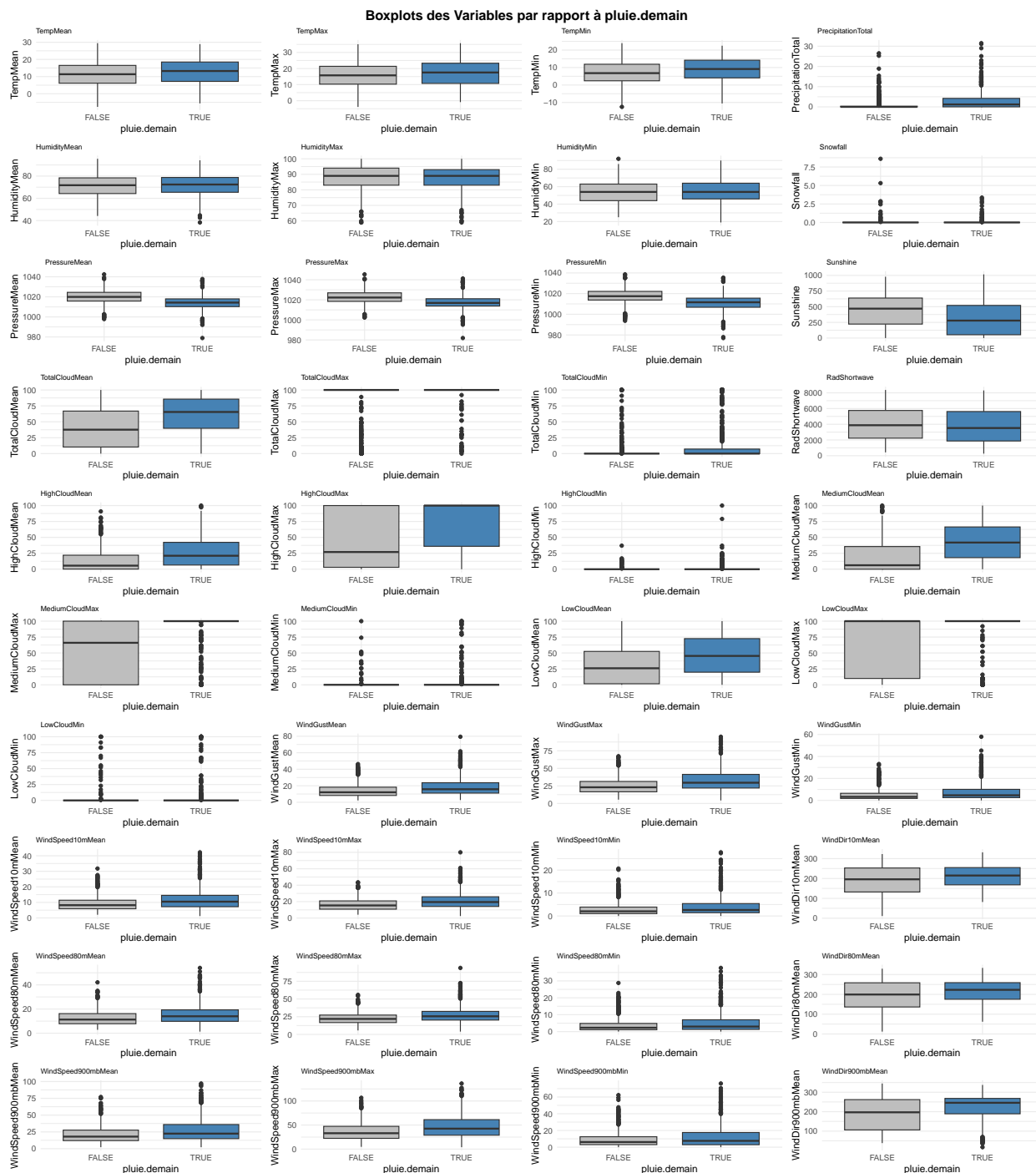
NewNames	OldNames
Year	Year
Month	Month
Day	Day
Hour	Hour
Minute	Minute
TempMean	Temperature.daily.mean..2.m.above.gnd.
HumidityMean	Relative.Humidity.daily.mean..2.m.above.gnd.
PressureMean	Mean.Sea.Level.Pressure.daily.mean..MSL.
PrecipitationTotal	Total.Precipitation.daily.sum..sfc.
Snowfall	Snowfall.amount.raw.daily.sum..sfc.
TotalCloudMean	Total.Cloud.Cover.daily.mean..sfc.
HighCloudMean	High.Cloud.Cover.daily.mean..high.cld.lay.
MediumCloudMean	Medium.Cloud.Cover.daily.mean..mid.cld.lay.
LowCloudMean	Low.Cloud.Cover.daily.mean..low.cld.lay.
Sunshine	Sunshine.Duration.daily.sum..sfc.
RadShortwave	Shortwave.Radiation.daily.sum..sfc.
WindSpeed10mMean	Wind.Speed.daily.mean..10.m.above.gnd.
WindDir10mMean	Wind.Direction.daily.mean..10.m.above.gnd.
WindSpeed80mMean	Wind.Speed.daily.mean..80.m.above.gnd.
WindDir80mMean	Wind.Direction.daily.mean..80.m.above.gnd.
WindSpeed900mbMean	Wind.Speed.daily.mean..900.mb.
WindDir900mbMean	Wind.Direction.daily.mean..900.mb.
WindGustMean	Wind.Gust.daily.mean..sfc.
TempMax	Temperature.daily.max..2.m.above.gnd.
TempMin	Temperature.daily.min..2.m.above.gnd.
HumidityMax	Relative.Humidity.daily.max..2.m.above.gnd.
HumidityMin	Relative.Humidity.daily.min..2.m.above.gnd.
PressureMax	Mean.Sea.Level.Pressure.daily.max..MSL.
PressureMin	Mean.Sea.Level.Pressure.daily.min..MSL.
TotalCloudMax	Total.Cloud.Cover.daily.max..sfc.
TotalCloudMin	Total.Cloud.Cover.daily.min..sfc.
HighCloudMax	High.Cloud.Cover.daily.max..high.cld.lay.
HighCloudMin	High.Cloud.Cover.daily.min..high.cld.lay.
MediumCloudMax	Medium.Cloud.Cover.daily.max..mid.cld.lay.
MediumCloudMin	Medium.Cloud.Cover.daily.min..mid.cld.lay.
LowCloudMax	Low.Cloud.Cover.daily.max..low.cld.lay.
LowCloudMin	Low.Cloud.Cover.daily.min..low.cld.lay.
WindSpeed10mMax	Wind.Speed.daily.max..10.m.above.gnd.
WindSpeed10mMin	Wind.Speed.daily.min..10.m.above.gnd.
WindSpeed80mMax	Wind.Speed.daily.max..80.m.above.gnd.
WindSpeed80mMin	Wind.Speed.daily.min..80.m.above.gnd.
WindSpeed900mbMax	Wind.Speed.daily.max..900.mb.
WindSpeed900mbMin	Wind.Speed.daily.min..900.mb.
WindGustMax	Wind.Gust.daily.max..sfc.
WindGustMin	Wind.Gust.daily.min..sfc.
pluie.demain	pluie.demain

Dans le summary des données présenté en annexe 1 de ce report, *Summary des données*, et dans le rapport html, nous remarquons qu’il y a 1180 lignes et pas de valeurs manquantes. Les variables “Hour” et “Minute” étant semblables pour toutes les données, nous excluons ces variables.

2. Représentation Graphique et Corrélation

2.1 Représentation Graphique - BoxPlots

Nous allons regarder maintenant les différentes variables numériques par rapport à la variable binaire “pluie.demain” à l’aide de box plots.



Sur les box-plots précédents, on notera pour les variables suivantes de:

- Températures (TempMean, TempMax & TempMin) que les médianes semblent très légèrement plus

élevées les jours de pluie comparés aux jours sans pluie bien que les différences ne soient pas très marquées. Les variables “température” semblent être des prédicteurs très moyens de la pluie du lendemain.

- Précipitations (PrecipitationTotal & Snowfall) que la présence de pluie est légèrement plus élevée les jours de pluie le lendemain. La variable Snowfall ne semble pas être un bon prédicteur car la plupart des valeurs sont à zéros dans les deux catégories.
- Humidité (HumidityMean, HumidityMax & HumidityMin) que l’humidité moyenne et minimale sont légèrement plus élevées les jours de pluie le lendemain et peuvent être des bons prédicteurs. La variable HumidityMax semble être un moins bon indicateur des conditions de pluie car la médiane est similaire entre les jours avec et sans pluie et les valeurs maximales sont plus élevées les jours sans pluie.
- Pression (PressureMean, PressureMax & PressureMin) que les valeurs sont plus faibles les jours de pluie, indiquant des conditions de pression favorisant la pluie.
- Ensoleillement (Sunshine) qu’il y a une réduction des minutes d’ensoleillement les jours de pluie.
- Nébulosité (TotalCloudMean, TotalCloudMax, TotalCloudMin, HighCloudMean, HighCloudMax, HighCloudMin, MediumCloudMean, MediumCloudMax, MediumCloudMin, LowCloudMean, LowCloudMax, LowCloudMin) que les variables se terminant par “Mean” et “Max” montrent une nébulosité plus élevée les jours de pluie le lendemain et semblent être de bons prédicteurs. Les variables se terminant par “Min”, quant à elles, montrent beaucoup moins de différences significatives entre les jours avec et sans pluie et seront moins utiles à la prédictions
- Rayonnement solaire (RadShortwave) que la variable est légèrement réduite les jours de pluie ce qui peut s’expliquer par une couverture nuageuse plus épaisse. Cette variable peut être un bon prédicteur.
- Rafales de Vent (WindGustMean, WindGustMax, WindGustMin) qu’il y a des différences notables entre les jours sans et avec pluie. Les valeurs semblent plus élevées les jours de pluie indiquant des conditions de vent plus fort.
- Vitesse et direction du vent (WindSpeed10mMean, WindSpeed10mMax, WindSpeed10mMin, WindDir10mMean, WindSpeed80mMean, WindSpeed80mMax, WindSpeed80mMin, WindDir80mMean, WindSpeed900mbMean, WindSpeed900mbMax, WindSpeed900mbMin, WindDir900mbMean,) que ces variables semblent être de bons prédicteurs de pluie le lendemain car elles semblent toutes plus élevées les jours de pluie le lendemain.

Pour nous assurer de ces résultats, nous allons voir maintenant les corrélations entre les variables.

2.2 Corrélations entre les variables

2.2.1 Matrice de Corrélation

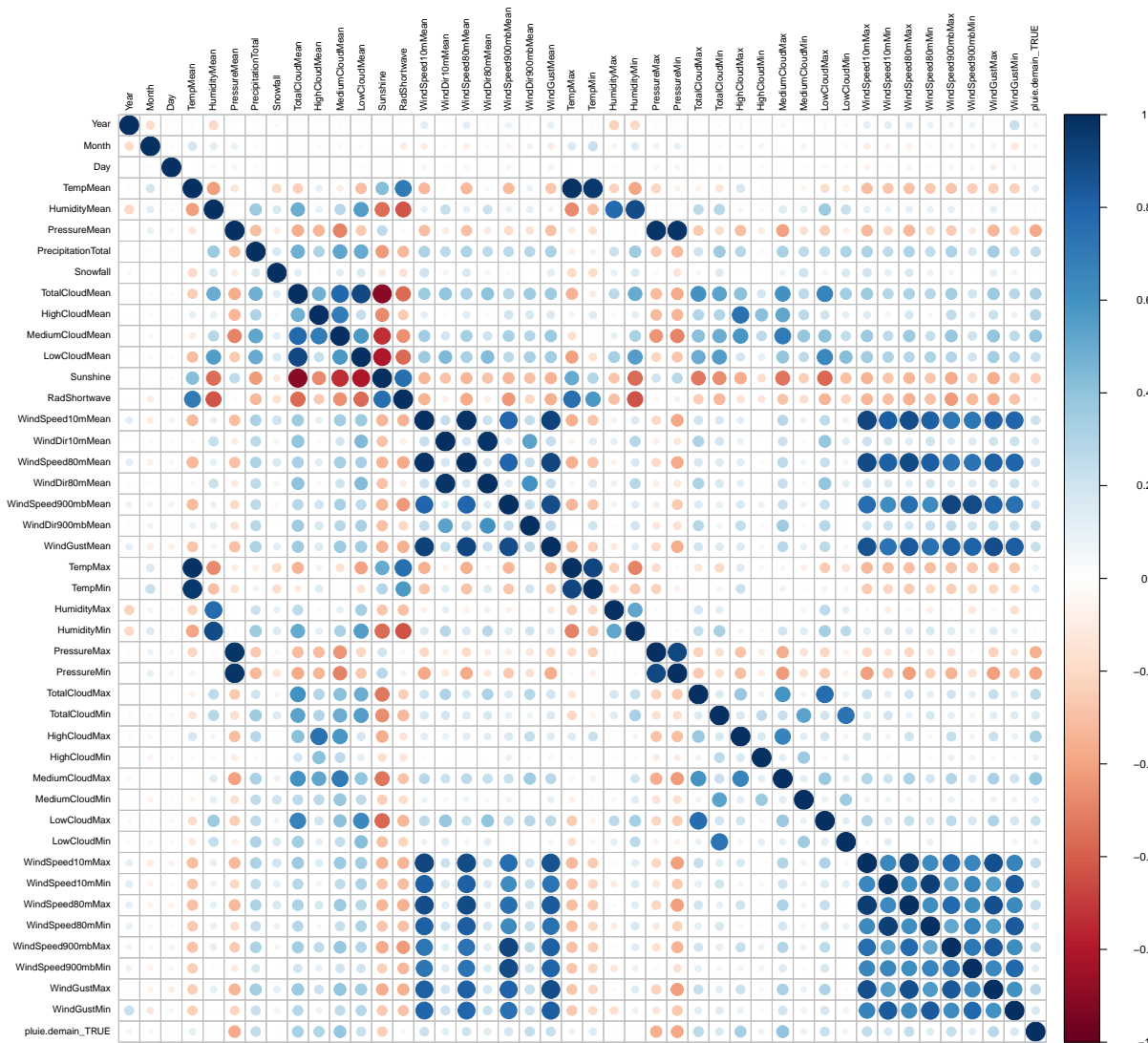
La matrice de corrélation présentée ci-dessous permet de visualiser les corrélations entre les différentes variables ainsi que leur corrélation avec le variable pluie.demain = “TRUE”.

On observe ici que les variables comportant “Mean”, “Max” & “Min” sont fortement corrélées entre elles. Pour éviter les problèmes de colinéarité, il sera nécessaire de sélectionner parmi ces 3 types de variables celle qui présente la plus forte corrélation avec la variable “pluie.demain”. On pourra donc ainsi inclure la variable la plus pertinente dans notre modèle.

On remarque également des fortes corrélations entre les variables de pression et de nébulosité totale indiquant que les pressions plus basses sont souvent associés à une couverture nuageuse plus élevée.

Comme constaté avec les boxplots précédemment, on note que la nébulosité, l’humidité, la pression, et l’ensoleillement sont assez corrélés avec les jours pluvieux le lendemain. La température montre une corrélation plus assez modérée avec pluie.demain = “TRUE”. Les tombées de neige montrent une corrélation faible avec pluie.demain = “TRUE” ce qui confirme ce que nous avons vu précédemment.

Correlation Matrix

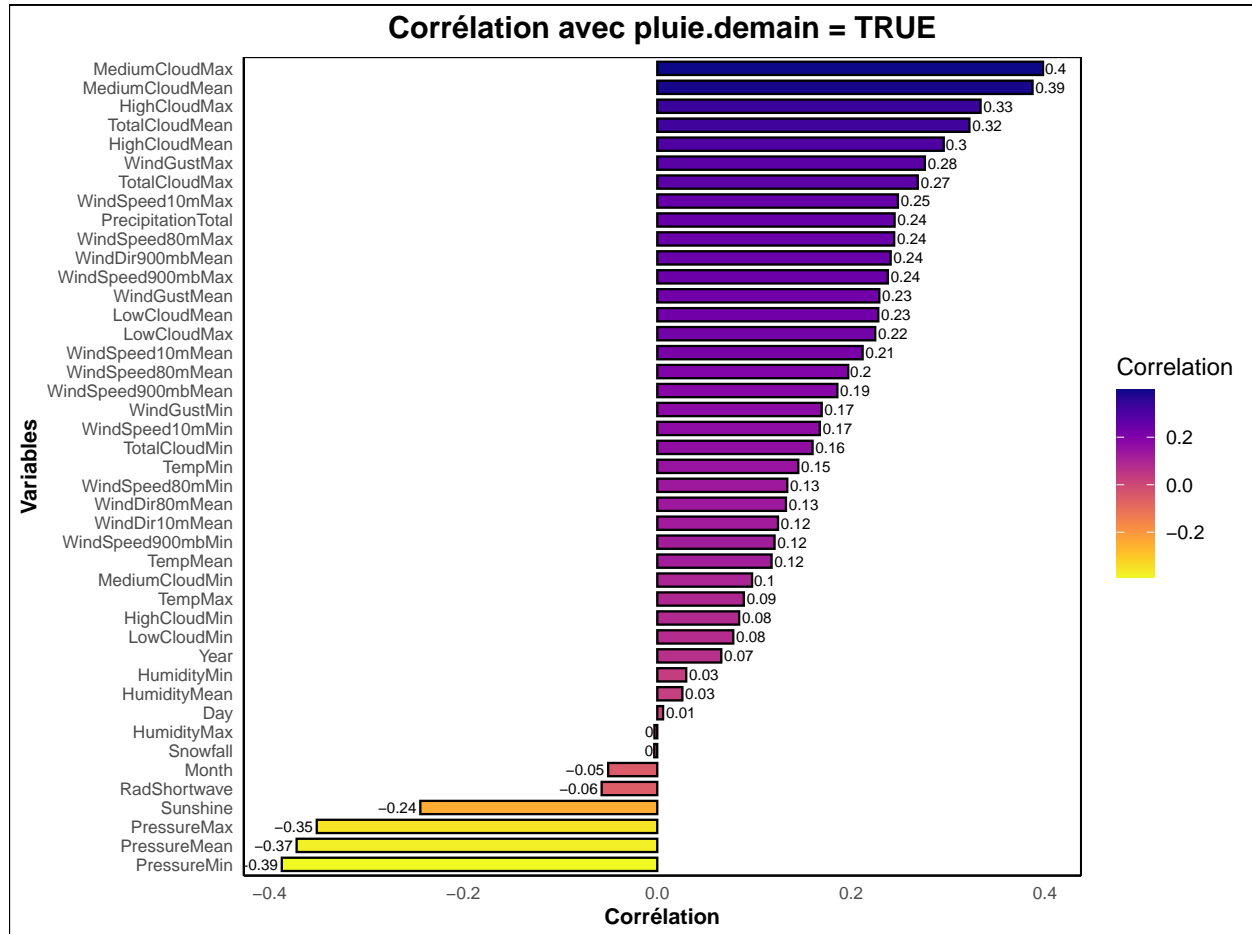


2.2.2 Corrélation avec pluie.demain = TRUE

Dans cette partie nous allons faire un zoom sur les variables corrélées avec “pluie.demain” = TRUE. On remarque que les groupes de variables suivantes ont un impact “pluie.demain” = TRUE:

- **Nébulosité:** les variables liées à la couverture nuageuse, surtout TotalCloud, HighCloud & MediumCloud, montrent des corrélations positives avec les jours de pluie. Cela confirme que ces variables sont de bons indicateurs pour la prédiction.
- **Vent:** Les variables des rafales de vent ont une corrélation forte avec la variable réponse, en particulier “WindGustMax”. Les vitesses “Max” des vents à différentes altitudes montrent également de corrélation positive. On peut également utiliser ces variables comme prédicteurs.

- Pression: Les 3 variables de pression sont négativement corrélées avec “pluie.demain” = TRUE. On confirme ici que les pressions plus basses sont associées à des conditions pluvieuses.
- Ensoleillement: la corrélation avec les jours de pluie est négative ici. Cela fait de cet variable un bon prédicteur.
- Humidité, Température et autres variables: Les corrélations avec ces variables semblent faibles indiquant qu’elles ne sont pas les principaux prédicteurs de la pluie le lendemain.



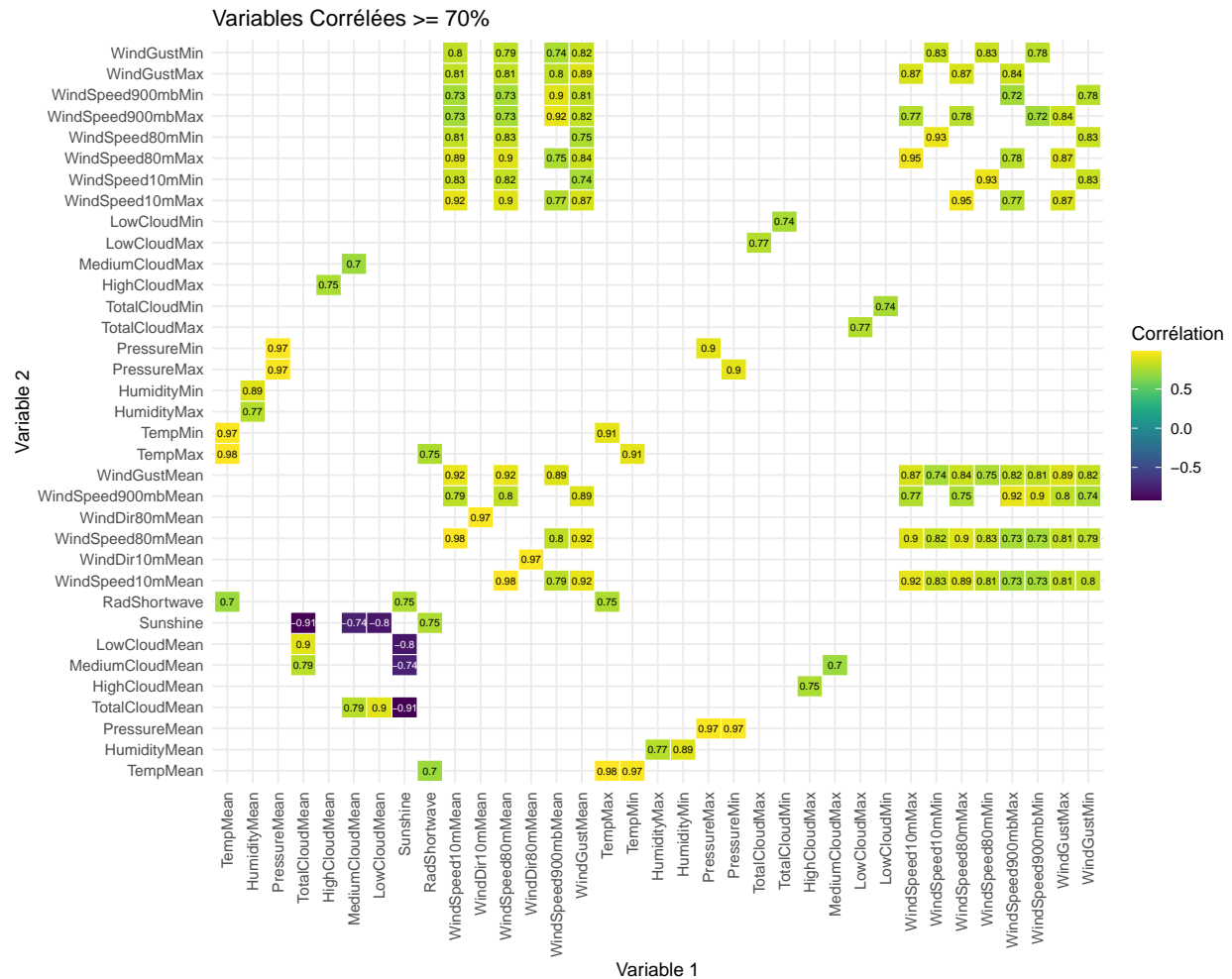
2.2.3 Variables corrélées entre elles

Pour identifier plus précisément les variables fortement corrélées entre elles, nous avons construit une matrice de corrélation, ci-dessous, qui met uniquement en évidence les variables présentant des corrélations positives ou négatives égale ou supérieure à 70%.

Comme précédemment, nous voyons que les variables d’un même groupe ayant un “Mean”, “Max” et “Min” sont fortement corrélées entre elles. Par exemple, la variable de “PressureMean” est corrélée à 97% avec les variables “PressureMax” et “PressureMin”. Il va falloir choisir l’une des 3 variables pour notre modèle.

Concernant les corrélations entre les groupes de variables, nous remarquons que la variable “Sunshine” est fortement corrélée négativement aux variables de couvertures nuageuses “TotalcloudMean”, “MediumCloudMean” et “LowCloudMean” indiquant que des jours ensoleillés sont associés à une faible couverture nuageuse. Cette variable aussi est très corrélée positivement à la variable “Radshorwave” car les jours avec plus d’ensoleillement tendent à recevoir plus de rayonnement solaire.

Nous remarquerons aussi avec ce graph que les variables de nébulosité sont très corrélées positivement entre elles et aussi que les variables de rafales de vent sont très corrélées positivement avec les variables indiquant les vitesses du vent.



2.2.4 Variables retenues

A la suite de notre étude graphique et corrélation et dans le but de pour construire un modèle manuel, nous allons sélectionner les variables suivantes:

- MediumCloudMax : Cette variable la plus corrélée avec la variable “pluie.demain = TRUE” et est peu corrélée avec les autres variables y compris “Sunshine”. On ne prendra pas “MediumCloudMean” car ces deux variables sont corrélées à 70% ensemble.
- HighCloudMax: Cette variable est corrélée avec la variable “pluie.demain = TRUE” et est peu corrélée avec les autres variables y compris “Sunshine”. On ne prendra pas “HighCloudMean” car ces deux variables sont corrélées à 75% ensemble.
- TotalCloudMax: Cette variable est corrélée avec la variable “pluie.demain = TRUE” et est peu corrélée avec les autres variables y compris “Sunshine”. On ne prendra pas “LowCloudMax” car ces deux variables sont corrélées à 77% ensemble.

- WindGustMax: Cette variable est corrélée avec la variable “pluie.demain = TRUE”, peu corrélée avec Sunshine et très corrélée avec les variables “WindSpeed”. Cela nous permet de prendre uniquement une variable.
- TempMin: Cette variable est la plus corrélée des variables de Température avec “pluie.demain”. Nous avons sélectionné une seule variable car elle est très corrélée avec “TempMax” et “TempMean”.
- PressureMin: Cette variable est la plus corrélée négativement avec “pluie.demain = TRUE”. Nous avons sélectionné une seule variable car elle est très corrélée avec “PressureMax” et “PressureMean”.
- WindDir900mbMean: Cette variable est la plus corrélée des variables de direction du vent avec “pluie.demain = TRUE” et n’est pas corrélée à plus de 70% avec les autres variables.
- Sunshine: Cette variable est corrélée négativement avec “pluie.demain = TRUE”. Elle n’a pas de corrélation avec les variables “CloudMax” mais avec les variables “CloudMean”. Elle a une corrélation positive à 75% avec la variable RadShortwave qui est donc exclus.
- PrecipitationTotal: Cette variable a une corrélation positive avec “pluie.demain = TRUE” et n’a pas de corrélation à plus de 70% avec les autres variables.

Notre modèle manuel prendra en compte toutes ces variables.

3. Choix du modèle

Pour prédire notre variable réponse “pluie.demain”, qui est de type binaire et qui suit une Loi de Bernoulli, nous allons nous intéresser au modèle linéaire généralisé utilisant la fonction de lien “logit” dans un premier temps et la fonction de lien “probit” dans un deuxième temps.

3.1 Choix du modèle Logistique

Le modèle logistique est défini par: $Y_i \sim \text{Bernoulli}(p_i)$ $\ln\left(\frac{p_i}{1-p_i}\right) = \beta_0 + \beta_1 x_i$ ce qu’on peut aussi écrire en $P(Y_i = 1) = \frac{e^{\beta_0 + \beta_1 x_i}}{1 + e^{\beta_0 + \beta_1 x_i}}$.

Les coefficients β sont estimés en maximisant la vraisemblance et représentent le changement du logarithme de cotes, dit logit, pour une unité de changement dans la variable dépendante en tenant constantes toutes les autres variables.

Dans les summary présenté pour chaque modèle, on calculera également le rapport de cotes (odds ratio) qui a pour formule e^{β_i} . Cette notion est utilisée pour mesurer l’impact d’une covariable sur la probabilité d’un évènement. Un rapport de cotes > 1 indique que la probabilité augmente, et < 1 qu’elle diminue. On calculera également la probabilité correspondante $p_i = P(Y_i = 1) = \left(\frac{\exp(\beta_i)}{1 + \exp(\beta_i)}\right)$ pour avoir une meilleure lisibilité.

3.1.1 Modèle initial

Nous commencerons par regarder un model_initial, qui prend en compte toutes les variables encore présente dans “meteo_train”.

```
model_initial <- glm(pluie.demain ~ ., family = binomial, data = meteo_train)
```

Ci dessous est présenté, le summary du model_initial avec les coefficients, les odds ratio, la probabilité et la significativité. Le modèle a atteint la convergence vers l’estimateur du maximum de vraisemblance après cinq itérations du scoring de Fisher.

	Estimate	Std..Error	z.value	p.value	Odds.Ratio	Probability....	Significance
(Intercept)	-76.461	70.610	-1.083	0.279	0.000	0.000	
Year	0.069	0.035	1.985	0.047	1.072	51.730	*
Month	-0.019	0.025	-0.746	0.456	0.982	49.535	
Day	0.012	0.008	1.445	0.148	1.012	50.295	
TempMean	0.183	0.164	1.116	0.264	1.201	54.562	
HumidityMean	0.020	0.032	0.612	0.540	1.020	50.496	
PressureMean	0.512	0.139	3.675	0.000	1.669	62.536	***
PrecipitationTotal	0.026	0.028	0.922	0.356	1.026	50.647	
Snowfall	-0.285	0.234	-1.220	0.223	0.752	42.916	
TotalCloudMean	0.012	0.012	1.039	0.299	1.013	50.312	
HighCloudMean	-0.003	0.007	-0.477	0.633	0.997	49.919	
MediumCloudMean	0.006	0.007	0.835	0.404	1.006	50.140	
LowCloudMean	-0.004	0.008	-0.535	0.593	0.996	49.892	
Sunshine	0.000	0.001	0.556	0.578	1.000	50.012	
RadShortwave	0.000	0.000	0.297	0.766	1.000	50.001	
WindSpeed10mMean	-0.046	0.097	-0.478	0.632	0.955	48.840	
WindDir10mMean	0.006	0.006	0.977	0.328	1.006	50.141	
WindSpeed80mMean	-0.094	0.069	-1.357	0.175	0.910	47.644	
WindDir80mMean	-0.009	0.006	-1.592	0.111	0.991	49.763	
WindSpeed900mbMean	0.018	0.026	0.707	0.479	1.019	50.459	
WindDir900mbMean	0.005	0.001	3.723	0.000	1.005	50.135	***
WindGustMean	0.018	0.037	0.483	0.629	1.018	50.445	
TempMax	-0.011	0.096	-0.119	0.905	0.989	49.714	
TempMin	-0.130	0.086	-1.509	0.131	0.878	46.749	
HumidityMax	0.000	0.021	0.003	0.997	1.000	50.002	
HumidityMin	-0.007	0.019	-0.370	0.711	0.993	49.828	
PressureMax	-0.259	0.075	-3.449	0.001	0.772	43.568	***
PressureMin	-0.321	0.076	-4.234	0.000	0.726	42.052	***
TotalCloudMax	0.003	0.005	0.701	0.483	1.003	50.085	
TotalCloudMin	0.008	0.006	1.243	0.214	1.008	50.195	
HighCloudMax	0.003	0.003	1.186	0.236	1.003	50.086	
HighCloudMin	0.006	0.021	0.294	0.769	1.006	50.154	
MediumCloudMax	0.006	0.003	1.946	0.052	1.006	50.154	.
MediumCloudMin	-0.005	0.009	-0.560	0.576	0.995	49.868	
LowCloudMax	0.003	0.003	0.867	0.386	1.003	50.074	
LowCloudMin	0.000	0.007	0.017	0.986	1.000	50.003	
WindSpeed10mMax	0.056	0.034	1.620	0.105	1.057	51.397	
WindSpeed10mMin	0.169	0.064	2.635	0.008	1.184	54.216	**
WindSpeed80mMax	0.004	0.028	0.138	0.890	1.004	50.098	
WindSpeed80mMin	-0.053	0.042	-1.257	0.209	0.948	48.674	
WindSpeed900mbMax	-0.013	0.012	-1.106	0.269	0.987	49.665	
WindSpeed900mbMin	-0.004	0.019	-0.212	0.832	0.996	49.899	
WindGustMax	0.023	0.017	1.319	0.187	1.023	50.570	
WindGustMin	0.005	0.028	0.182	0.855	1.005	50.128	

Statistic	Value	DF
Null Deviance	1635.417	1179
Residual Deviance	1232.696	1136

Statistic	Value	DF
Number of Fisher Scoring Iterations	5.000	NA
AIC	1320.696	NA

On remarque dans ce `model_initial` qu'il y a uniquement 7 coefficients statistiquement significatifs. Le modèle semble assez complexe et semble avoir beaucoup de colinéarité.

- Colinéarité : En utilisant la fonction `VIF` du package `{car}`, qui mesure à quel point la variance d'un coefficient de régression est augmentée en raison de la colinéarité, nous allons pouvoir étudier quelles sont les variables ayant une colinéarité élevée. Un VIF supérieur à 7 est souvent utilisé comme indicateur d'une colinéarité sévère qui nécessite une attention. Les résultats détaillés sont présentés dans la version html du document. Parmi les vif les plus importants, nous avons "TempMean" avec un vif de 258 et surtout "PressureMean" avec un vif de 189. Cette dernière variable est pourtant statistiquement significative avec un `odd ratio` > 1 . Sur les 43 variables du modèle, 16 ont un vif inférieur à 7, 3 ont un vif supérieur à 100 et 24 ont un vif entre 7 et 100. Ce modèle a bien de nombreuses colinéarités.
- Table deviance: Nous utiliserons tout d'abord, la fonction `anova(model_initial, test = "LRT")` présentée dans la version html du report. Pour chaque ajout d'un prédicteur dans le modèle, la fonction calcule la deviance du modèle et effectue un test du rapport de vraisemblance entre le modèle avec le prédicteur et sans le prédicteur. Une p-value inférieure à 0.05 indique que le prédicteur améliore significativement l'ajustement du modèle. Dans le `model_initial`, 17 variables réduisent la deviance du modèle. Ces variables apportent une contribution significative et utile à la prédiction de la variable réponse. Parmi les plus significatives on retrouve les variables: "PressureMin", "PressureMax", "PrecipitationTotal", "TotalCloudMean", "WindDir900mbMean", "HumidityMean", "WindSpeed10mMean", "TempMean". Nous pouvons faire également un test de deviance
- Test de deviance: Nous utiliserons ensuite un test de deviance pour comparer notre modèle noté génériquement M_k dans un premier temps à un modèle nul M_0 dans lequel la variable réponse Y_i , "pluie.demain" dans notre cas, est iid et il existe un p pour tout i , $p_i = p$. Ce modèle est équivalent à dire que $\beta_1 = \beta_2 = \dots = \beta_k = 0$ et le degré de liberté est égal à 1. Dans un deuxième temps on comparera notre modèle à un modèle saturé M_{sat} dans lequel il n'y a aucune structure aux p_i et n degrés de liberté. Les trois modèles sont imbriqués comme ceci: $M_0 \subset M_k \subset M_{sat}$. Nous ferons ensuite des tests du χ^2 de rapport de vraisemblance.

```
pvalMkinitial <- pchisq(1635.4 - 1232.7, 1179 - 1136, lower = F)
cat("La p-valeur pour la différence de déviance est :", format(pvalMkinitial,
  scientific = TRUE))
```

```
## La p-valeur pour la différence de déviance est : 6.130973e-60
```

On testera d'abord M_0 contre M_k en utilisant la statistique de test suivante: $D_0 - D_k = -2 \ln \left(\frac{\text{vraisemblance } M_0}{\text{vraisemblance } M_k} \right)$ où D_0 est la déviance nulle et D_k est la déviance résiduelle. Sous l'hypothèse que M_0 est le vrai modèle on a $D_0 - D_k \sim \chi^2(k)$. La déviance nulle du `model_initial` est de 1635.4 avec 1179 degrés de liberté et la déviance résiduelle est de 1232.7 pour 1136 degrés de liberté. Avec une p-valeur inférieure à 0.05, le `model_initial` avec plus de variable est meilleur en termes d'ajustement par rapport au modèle M_0 qui n'inclut que l'intercept.

```
pvalMkMsatinitial <- pchisq(1232.7, 1136, lower = F)
cat("La p-valeur est:", sprintf("%.3f", pvalMkMsatinitial))
```

```
## La p-valeur est: 0.023
```

On testera ensuite M_k contre M_{sat} en utilisant la statistique de test suivante: $D_k = -2 \ln \left(\frac{\text{vraisemblance de } M_k}{\text{vraisemblance de } M_{sat}} \right)$ où D_k est la déviance résiduelle. Sous l'hypothèse que M_k est le vrai modèle, on a $D_k \sim \chi^2_{n-k-1}$. La déviance résiduelle du model_initial est de 1232.7 pour 1136 degrés de liberté. On remarquera que la p-valeur est de 0.023 et donc inférieur à 0.05. On rejette donc notre modèle initial car on lui préfère le modèle saturé. Notre modèle n'est pas suffisant.

- Prédiction: Malgré le fait que le model_initial soit rejeté, nous allons tout de même calculer les prédictions pour ce modèle pour pouvoir comparer avec les autres modèles. On cherche ici à faire une prédiction binaire. Si notre prédiction est $\tilde{Y} \sim \text{Bernoulli}(\tilde{p})$ avec $\tilde{p} = \frac{e^{\beta \tilde{X}}}{1+e^{\beta \tilde{X}}}$, on va souvent prédire 1 si $\tilde{p} \geq \frac{1}{2}$ et 0 si $\tilde{p} < \frac{1}{2}$. Voici la table de décision ci-dessous avec un seuil de décision fixé à 0.5 pour comparer les modèles. Nous avons des FPR (False Positive Rate) = $P(\tilde{Y} = 1 | Y = 0) = 158/579 = 27\%$, une spécificité $421/579 = 0.72 = 1-\text{FPR}$ et sensibilité TPR (True Positive Rate) = $P(\tilde{Y} = 1 | Y = 1) = 464/601 = 77\%$. En tout le model_initial donne 75% de bonnes prédictions.

Table 3: Table de Décision - Model_initial

	FALSE	TRUE
FALSE	421	137
TRUE	158	464

Bonnes prédictions: 75.00%

Nous pouvons aussi peut définir un seuil s optimal. On pourrait mettre des poids pour pénaliser les mauvaises prédictions mais nous ne le ferons pas dans ce rapport. Le seuil optimal reste donc proche de 0.5 et est de 0.49. Une fausse prédiction est enlevée. Nous avons aussi calculé la courbe ROC (receiving operator characteristic) avec en abscisse les FPR et en ordonnée les TPR. On voit que la courbe est au-dessus de la droite d'équation $y = x$, notre model_initial fait donc mieux que l'aléatoire et l'AUC (Area Under Curve) qui mesure la qualité de la classification est de 82%, ce qui est un très bon score puisqu'il est au-dessus de la classification aléatoire (AUC=50%).

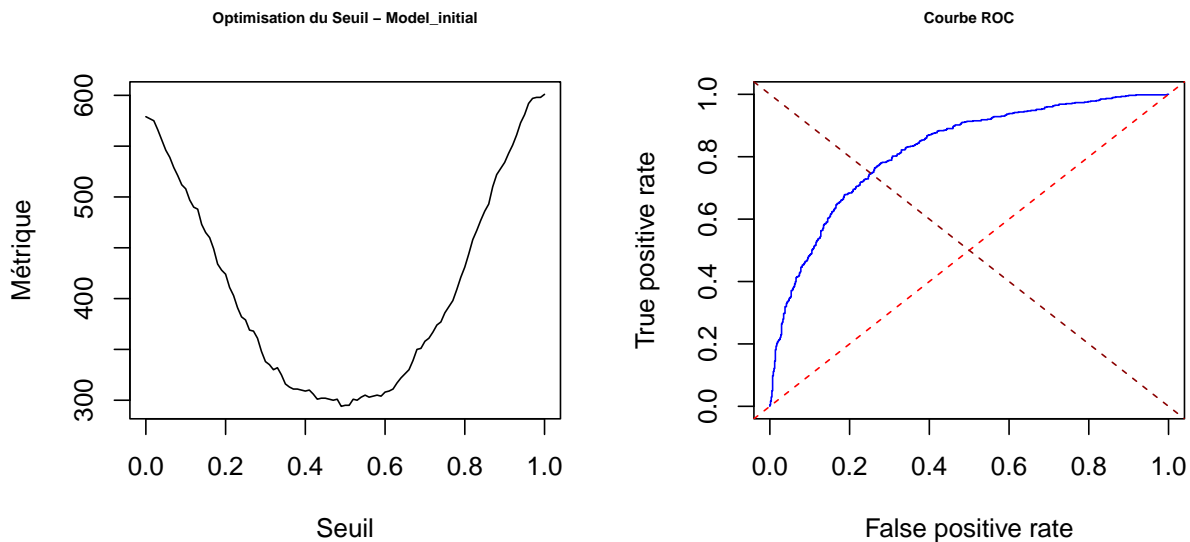


Table 4: Seuil, Coût 0.5, Coût Seuil Optimal, AUC

Metric	Value
Meilleur seuil	0.49
Coût pour le seuil de 0.5	295
Coût pour le meilleur seuil	294
AUC - model_initial	0.82

3.1.2 Modèle Manuel

Nous continuerons par regarder un `model_manuel`, qui prend uniquement en compte les variables de la partie graphique et corrélation.

```
model_manuel <- glm(pluie.demain ~ MediumCloudMax + HighCloudMax +
  TotalCloudMax + WindGustMax + TempMin + PressureMin + WindDir900mbMean +
  Sunshine + PrecipitationTotal, family = binomial, data = meteo_train)
```

Ci dessous est présenté, le summary du `model_manuel` avec les coefficients, les odds ratio, la probabilité et la significativité. Le modèle a atteint la convergence vers l'estimateur du maximum de vraisemblance après 4 itérations du scoring de Fisher.

	Estimate	Std..Error	z.value	p.value	Odds.Ratio	Probability....	Significance
(Intercept)	67.758	10.413	6.507	0.000	2.671794e+29	100.000	***
MediumCloudMax	0.008	0.003	2.975	0.003	1.008000e+00	50.193	**
HighCloudMax	0.004	0.002	2.067	0.039	1.004000e+00	50.107	*
TotalCloudMax	0.005	0.003	1.374	0.170	1.005000e+00	50.118	
WindGustMax	0.019	0.006	3.214	0.001	1.019000e+00	50.467	**
TempMin	0.048	0.013	3.630	0.000	1.050000e+00	51.208	***
PressureMin	-0.069	0.010	-6.792	0.000	9.330000e-01	48.267	***
WindDir900mbMean	0.002	0.001	2.154	0.031	1.002000e+00	50.055	*
Sunshine	0.000	0.000	-0.463	0.644	1.000000e+00	49.996	
PrecipitationTotal	0.015	0.022	0.673	0.501	1.015000e+00	50.363	

Statistic	Value	DF
Null Deviance	1635.417	1179
Residual Deviance	1303.084	1170
Number of Fisher Scoring Iterations	4.000	NA
AIC	1323.084	NA

On remarque dans ce `model_manuel` qu'il y a 6 coefficients statistiquement significatifs sur 9 variables. Le modèle est moins complexe mais à un AIC un peu moins bon.

- Colinéarité : En utilisant la fonction VIF décrite plus haut. On remarque que tous les vifs sont inférieurs à 2.32. Ce modèle n'a pas de colinéarité. Les résultats détaillés sont présentés dans le version html du document.
- Table deviance: Nous utiliserons tout d'abord, la fonction `anova(model_manuel, test = "LRT")` présentée dans la version html du report. Dans le `model_manuel`, 7 variables réduisent la deviance du le modèle. "MediumCloudMax" est la variable qui réduit le plus la deviance. "Sunshine" et "PrecipitationTotal" ne contribuent pas beaucoup à la réduction de la deviance et sont non statistiquement

significatives. On utilise maintenant la fonction `anova(model_initial, model_manuel, test = "LRT")` qui compare les 2 modèles. La p-valeur étant de très petite (0.0002431), nous concluons donc que le `model_initial` fournit un ajustement meilleur que le `model_manuel`.

- Test de deviance: Nous utiliserons ensuite un test de deviance pour comparer notre modèle noté génériquement M_k dans un premier temps à un modèle null M_0 . Nous ferons ensuite des tests du χ^2 de rapport de vraisemblance.

```
pvalMoMkmanuel <- pchisq(1635.4 - 1303.084, 1179 - 1170, lower = F)
cat("La p-valeur pour la différence de déviance est :", format(pvalMoMkmanuel,
  scientific = TRUE))
```

```
## La p-valeur pour la différence de déviance est : 3.579865e-66
```

La déviance null du `modele_manuel` est de 1635.4 avec 1179 degrés de liberté et la déviance résiduelle est de 1303.084 pour 1170 degrés de liberté. Avec une p-valeur inférieur à 0.05, le `model_manuel` avec plus de variables est meilleur en termes d'ajustement par rapport au modèle M_0 .

```
pvalMkMsatmanuel <- pchisq(1303.084, 1170, lower = F)
cat("La p-valeur est:", sprintf("%.3f", pvalMkMsatmanuel))
```

```
## La p-valeur est: 0.004
```

On testera ensuite M_k contre M_{sat} . La déviance résiduelle du `model_manuel` est de 1303.084 pour 1170 degrés de liberté. On remarquera que la p-valeur est de 0.004 et donc inférieur à 0.05. On rejette donc notre `model_manuel` car on lui préfère le modèle saturé. Notre modèle n'est pas suffisant.

- Prédiction: Malgré le fait que le `model_manuel` soit rejeté, nous allons tout de même calculer les prédictions pour ce modèle pour pouvoir comparer avec les autre modèle. Voici la table de décision ci-dessous avec un seuil de décision fixé à 0.5 pour comparer les modèles. Nous avons des FPR (False Positive Rate) = $P(\tilde{Y} = 1 \mid Y = 0) = 183/579 = 31\%$, un spécificité $396/579 = 0.68 = 1 - \text{FPR}$ et sensibilité TPR (True Positive Rate) = $P(\tilde{Y} = 1 \mid Y = 1) = 472/601 = 78\%$. En tout le `modele_manuel` donne 73.6% de bonnes prédictions. Ce modèle a un meilleur TPR que le `model_initial` mais le pourcentage de bonnes prédictions est un peu plus faible de 1.4%.

Table 6: Table de Décision - Model_manuel

	FALSE	TRUE
FALSE	396	129
TRUE	183	472

```
## Bonnes prédictions: 73.56%
```

Nous pouvons aussi peut définir un seuil s optimal. Le seuil optimal reste donc proche de 0.5 et est de 0.51. Une fausse prédiction est enlevée. Nous avons aussi calculé la courbe ROC (receiving operator characteristic) avec en abscisse les FPR et en ordonnée les TPR. On voit que la courbe est au dessus de la droite d'équation $y = x$, notre `modele_step_backward` fait donc mieux que l'aléatoire et l'AUC (Area Under Curve) qui mesure la qualité de la classification est de 79%, ce qui est un très bon score puisqu'il est au-dessous de la classification aléatoire (AUC=50%). L'AUC est un peu moins bon que pour le `model_initial` de -3%.

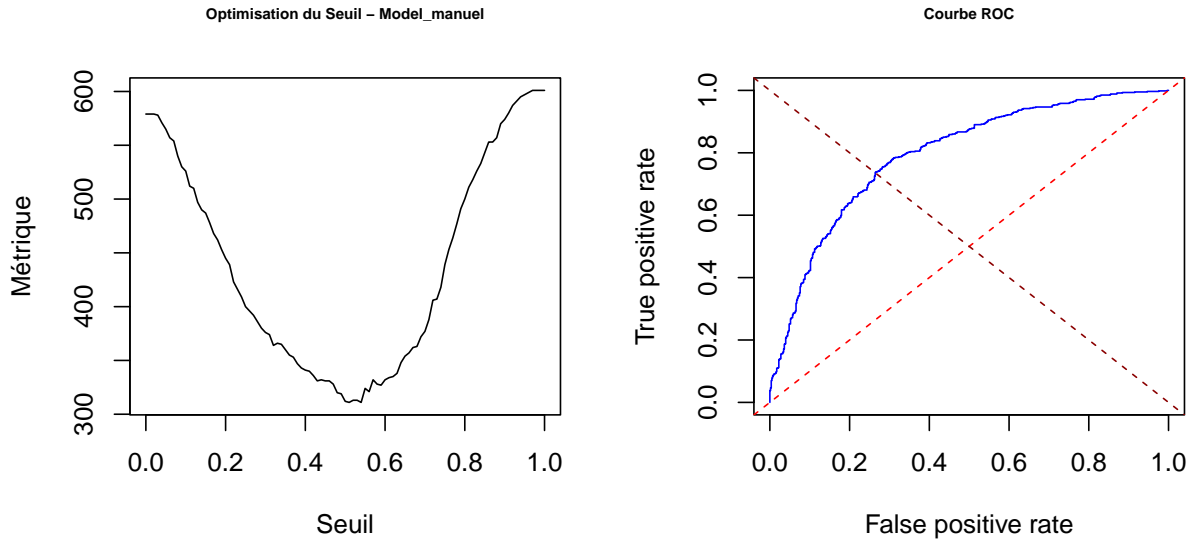


Table 7: Seuil, Coût 0.5, Coût Seuil Optimal, AUC

Metric	Value
Meilleur seuil	0.51
Coût pour le seuil de 0.5	312
Coût pour le meilleur seuil	311
AUC - model_manuel	0.79

3.1.3 Modèle automatique - Step Backward

Nous continuerons par regarder un `model_step_backward` automatique, car les précédents modèles ne semblent pas totalement bien capter les ajustements de la variable réponse “pluie.demain”. La fonction `step` sans précision génère directement la version “both”. On privilégie la fonction `step` “backward” à la version “forward”. Quant à la la version avec “both”, elle donne le même modèle que la version “backward”. Comme on commence avec un modèle complexe et riches en variables, on va réaliser le `step backward` à partir du `model_initial` et cela permet de simplifier progressivement tout en capturant les interactions importantes dès le départ. On présentera tous les résultats uniquement dans le html.

Le meilleur modèle donné par le fonction `step(model_initial, direction = “backward”)` est le modèle présenté ci-dessous avec les coefficients, les odds ratio, la probabilité et la significativité. Le modèle a atteint la convergence vers l’estimateur du maximum de vraisemblance après 4 itérations du scoring de Fisher.

```
model_step_backward <- glm(pluie.demain ~ Year + TempMean + PressureMean +
  Snowfall + MediumCloudMean + WindSpeed80mMean + WindDir80mMean +
  WindDir900mbMean + TempMin + PressureMax + PressureMin +
  TotalCloudMax + TotalCloudMin + MediumCloudMax + WindSpeed10mMax +
  WindSpeed10mMin + WindGustMax, family = binomial, data = meteo_train)
```

	Estimate	Std..Error	z.value	p.value	Odds.Ratio	Probability....	Significance
(Intercept)	-68.943	62.792	-1.098	0.272	0.000	0.000	
Year	0.066	0.031	2.134	0.033	1.068	51.646	*
TempMean	0.147	0.050	2.943	0.003	1.158	53.662	**
PressureMean	0.482	0.131	3.673	0.000	1.619	61.820	***
Snowfall	-0.316	0.215	-1.470	0.142	0.729	42.155	
MediumCloudMean	0.011	0.004	2.662	0.008	1.011	50.270	**
WindSpeed80mMean	-0.115	0.030	-3.824	0.000	0.892	47.140	***
WindDir80mMean	-0.003	0.002	-1.763	0.078	0.997	49.933	.
WindDir900mbMean	0.005	0.001	3.557	0.000	1.005	50.115	***
TempMin	-0.103	0.054	-1.897	0.058	0.902	47.432	.
PressureMax	-0.242	0.071	-3.428	0.001	0.785	43.976	***
PressureMin	-0.306	0.072	-4.276	0.000	0.736	42.409	***
TotalCloudMax	0.008	0.004	2.383	0.017	1.008	50.209	*
TotalCloudMin	0.008	0.004	2.031	0.042	1.008	50.196	*
MediumCloudMax	0.006	0.003	2.331	0.020	1.006	50.156	*
WindSpeed10mMax	0.060	0.023	2.620	0.009	1.061	51.491	**
WindSpeed10mMin	0.111	0.036	3.081	0.002	1.118	52.779	**
WindGustMax	0.024	0.011	2.180	0.029	1.024	50.592	*

Statistic	Value	DF
Null Deviance	1635.417	1179
Residual Deviance	1246.847	1162
Number of Fisher Scoring Iterations	4.000	NA
AIC	1282.847	NA

On remarque dans ce `model_step_backward` qu'il y a 14 coefficients statistiquement significatifs sur 17 variables. L'intercept n'est plus significatif comme dans le `modele_manuel`. Le modèle est moins complexe et à un meilleur AIC à 1282.8. La variable `Snowfall` n'est pas significative.

- Colinéarité : En utilisant la fonction VIF décrite plus haut. On remarque que certains VIF sont très importants comme ceux des variables de Pressure. Celui de "PressureMean" est à 169. Ceux des variables de températures sont entre 20 et 47. Il y a 3 variables "Pressure" et 2 variables "Temp" dans ce modèle. Comme elles sont très corrélées entre elles nous en enlèveront certaines dans un prochain `model_step_backward_corr`. Les résultats détaillés sont présentés dans la version html du document. Malgré les collinéarités nous allons poursuivre pour voir ce que cela donne.
- Table deviance: Nous utiliserons tout d'abord, la fonction `anova(model_step_backward, test = "LRT")` présentée dans la version html du report. Dans le `model_step_backward`, 13 variables réduisent la deviance du modèle. "PressureMean" est la variable qui réduit le plus la deviance avec 168.15. On gardera cette variable pour le `model_step_backward_corr`. "WindDir80mMean" et "Snowfall" ne contribuent pas beaucoup à la réduction de la deviance et sont non statistiquement significatives. On utilise maintenant la fonction `anova(model_initial, model_step_backward, test = "LRT")` qui compare les 2 modèles. La p-valeur étant de très grande (0.9709), nous concluons donc qu'il n'y a pas de différence significative entre les deux modèles en termes de qualité de l'ajustement. Néanmoins, comme le `model_step_backward` est plus simple, il est préférable de choisir ce modèle qui facilite l'interprétation et réduit le surajustement.
- Test de deviance: Nous utiliserons ensuite un test de deviance pour comparer notre modèle noté génériquement M_k dans un premier temps à un modèle null M_0 . Nous ferons ensuite des tests du χ^2 de rapport de vraisemblance.

```
pvalMkstepback <- pchisq(1635.4 - 1246.85, 1179 - 1162, lower = F)
cat("La p-valeur pour la différence de déviance est :", format(pvalMkstepback,
  scientific = TRUE))
```

```
## La p-valeur pour la différence de déviance est : 4.574864e-72
```

La déviance null du modele_manuel est de 1635.4 avec 1179 degrés de liberté et la déviance résiduelle est de 1246.85 pour 1162 degrés de liberté. Avec une p-valeur inférieur à 0.05, le model_manuel avec plus de variables est meilleur en termes d'ajustement par rapport au modèle modèle M_0 .

```
pvalMkMsatstepback <- pchisq(1246.85, 1162, lower = F)
cat("La p-valeur est:", sprintf("%.3f", pvalMkMsatstepback))
```

```
## La p-valeur est: 0.042
```

On testera ensuite M_k contre M_{sat} . La déviance résiduelle du model_step_backward est de 1246.85 pour 1162 degrés de liberté. On remarquera que la p-valeur est de 0.042 et est toujours inférieure à 0.05 mais se rapproche du seuil de significativité. On rejette donc notre model_step_backward car on lui préfère le modèle saturé. Notre modèle n'est pas encore suffisant.

- Prédiction: Malgré le fait que le model_step_backward soit rejeté, nous allons tout de même calculer les prédictions pour ce modèle pour pouvoir comparer avec les autre modèle. Voici la table de décision ci-dessous avec un seuil de décision fixé à 0.5 pour comparer les modèles. Nous avons des FPR (False Positive Rate) = $P(\tilde{Y} = 1 \mid Y = 0) = 162/579 = 27\%$, un spécificité $417/579 = 0.72 = 1 - \text{FPR}$ et sensivité TPR (True Positive Rate) = $P(\tilde{Y} = 1 \mid Y = 1) = 468/601 = 78\%$. En tout le model_step_backward donne 75% de bonnes prédictions. Ce modèle a un meilleur TPR que le model_initial de 1% et le poucentage de bonnes prédictions est le même.

Table 9: Table de Décision - Model_step_backward

	FALSE	TRUE
FALSE	417	133
TRUE	162	468

```
## Bonnes prédictions: 75.00%
```

Nous pouvons aussi peut définir un seuil s optimal. Le seuil optimal reste donc proche de 0.5 et est de 0.52. 10 fausses prédictions sont enlevées. Nous avons aussi calculé la courbe ROC (receiving operator characteristic) avec en abscisse les FPR et en ordonnée les TPR. On voit que la courbe est au dessus de la droite d'équation $y = x$, notre modele_manuel fait donc mieux que l'aléatoire et l'AUC (Area Under Curve) qui mesure la qualité de la classification est de 82%, ce qui est un très bon score puisqu'il est au-dessous de la classification aléatoire (AUC=50%). L'AUC est identique au model_initial.

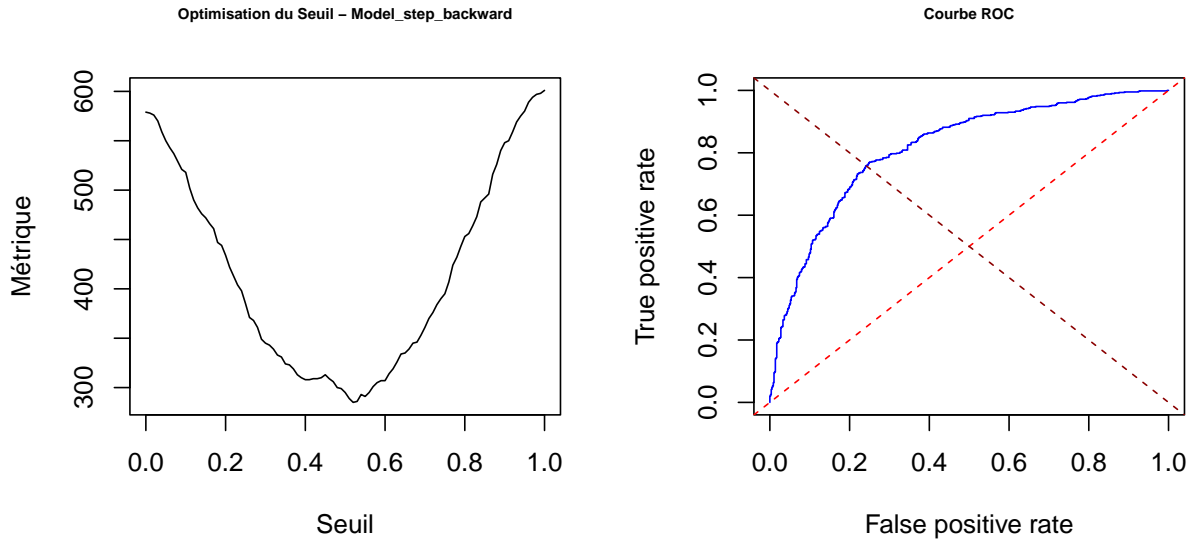


Table 10: Seuil, Coût 0.5, Coût Seuil Optimal, AUC

Metric	Value
Meilleur seuil	0.52
Coût pour le seuil de 0.5	295
Coût pour le meilleur seuil	285
AUC - model_step_backward	0.82

3.1.4 Modèle - Step Backward avec correction VIF

Nous continuerons par la correction du `model_step_backward` automatique, car il reste trop de colinéarité dans ce modèle, nous aurons donc ensuite le `model_step_backward_corr`. On retire d’abord, les variables “PressureMax” et “PressureMin”, pour garder seulement “PressureMean” car cette variable explique mieux la deviance. En faisant cela le vif de cette variable descend de 169 à 1.25. On enlève ensuite la variable “TempMin” et on garde la variable “TempMean” car celle-ci explique plus la deviance. On passe d’un vif de 24 sur cette variable à 1.35. On enlèvera aussi la variable “WindSpeed80mMean” ayant un vif de 9.9. Tous les vif sont maintenant en dessous de 5.

Nous terminerons avec le modèle corrigé `model_step_backward_corr` présenté ci-dessous avec les coefficients, les odds ratio, la probabilité et la significativité. Le modèle a atteint la convergence vers l’estimateur du maximum de vraisemblance après 4 itérations du scoring de Fisher.

```
model_step_backward_corr <- glm(pluie.demain ~ Year + TempMean +
  PressureMean + Snowfall + MediumCloudMean + WindDir80mMean +
  WindDir900mbMean + TotalCloudMax + TotalCloudMin + MediumCloudMax +
  WindSpeed10mMax + WindSpeed10mMin + WindGustMax, family = binomial,
  data = meteo_train)
```

	Estimate	Std..Error	z.value	p.value	Odds.Ratio	Probability...	Significance
(Intercept)	-80.718	61.510	-1.312	0.189	0.000	0.000	
Year	0.068	0.030	2.259	0.024	1.071	51.706	*
TempMean	0.056	0.012	4.854	0.000	1.057	51.397	***
PressureMean	-0.059	0.011	-5.385	0.000	0.943	48.528	***
Snowfall	-0.240	0.176	-1.362	0.173	0.787	44.036	
MediumCloudMean	0.007	0.004	1.807	0.071	1.007	50.174	.
WindDir80mMean	-0.003	0.001	-2.005	0.045	0.997	49.928	*
WindDir900mbMean	0.004	0.001	3.102	0.002	1.004	50.097	**
TotalCloudMax	0.008	0.003	2.219	0.026	1.008	50.190	*
TotalCloudMin	0.007	0.004	1.889	0.059	1.007	50.177	.
MediumCloudMax	0.007	0.003	2.838	0.005	1.007	50.186	**
WindSpeed10mMax	0.008	0.017	0.475	0.635	1.008	50.208	
WindSpeed10mMin	0.004	0.026	0.141	0.888	1.004	50.090	
WindGustMax	0.021	0.010	2.063	0.039	1.021	50.526	*

Statistic	Value	DF
Null Deviance	1635.417	1179
Residual Deviance	1288.715	1166
Number of Fisher Scoring Iterations	4.000	NA
AIC	1316.715	NA

On remarque dans ce `model_step_backward_corr` que les coefficients sont moins statistiquement significatifs que dans `model_step_backward`. L'intercept n'est pas significatif comme dans le `model_step_backward`. Le modèle est moins complexe et à un moins bon AIC à 1316.7. La variable `Snowfall` n'est toujours pas significative.

- Colinéarité : En utilisant la fonction VIF décrite plus haut, nous avons exclu toute colinéarité du `model_step_backward_corr`. Les résultats détaillés sont présentés dans le version html du document.
- Table deviance: Nous utiliserons tout d'abord, la fonction `anova(model_step_backward_corr, test = "LRT")` présentée dans la version html du report. Dans le `model_step_backward_corr`, 9 variables réduisent la deviance du modèle. "PressureMean" est la variable qui réduit le plus la deviance avec 168.15. "TotalCloudMin", "WindSpeed10mMin" et "Snowfall" ne contribuent pas beaucoup à la réduction de la deviance et sont non statistiquement significatives. On utilise maintenant la fonction `anova(model_initial, model_step_backward_corr, test = "LRT")` qui compare les 2 modèles. La p-valeur étant de très faible (0.002717), nous concluons donc qu'il y a une différence significative entre les deux modèles en termes de qualité de l'ajustement. Le `model_step_backward_corr` est plus simple mais il n'est préférable de choisir ce modèle par rapport au `model_initial`. On utilise maintenant la fonction `anova(model_step_backward, model_step_backward_corr, test = "LRT")`. La p-valeur étant de très faible (1.777e-08), nous concluons donc qu'il y a une différence significative entre les deux modèles en termes de qualité de l'ajustement. Le `model_step_backward` ajuste significativement mieux les données que le `model_step_backward_corr`.
- Test de deviance: Nous utiliserons ensuite un test de deviance pour comparer notre modèle noté génériquement M_k dans un premier temps à un modèle null M_0 . Nous ferons ensuite des tests du χ^2 de rapport de vraisemblance.

```
pvalMoMkstepbackcorr <- pchisq(1635.4 - 1288.715, 1179 - 1166,
  lower = F)
cat("La p-valeur pour la différence de déviance est :", format(pvalMoMkstepbackcorr,
  scientific = TRUE))
```

```
## La p-valeur pour la différence de déviance est : 3.863549e-66
```

La déviance null du modele_manuel est de 1635.4 avec 1179 degrés de liberté et la déviance résiduelle est de 1288.71 pour 1166 degrés de liberté. Avec une p-valeur inférieure à 0.05, le model_manuel avec plus de variables est meilleur en termes d'ajustement par rapport au modèle modèle M_0 .

```
pvalMkMsatstepbackcorr <- pchisq(1288.715, 1166, lower = F)
cat("La p-valeur est:", sprintf("%.3f", pvalMkMsatstepbackcorr))
```

```
## La p-valeur est: 0.007
```

On testera ensuite M_k contre M_{sat} . La déviance résiduelle du model_step_backward_corr est de 1288.71 pour 1166 degrés de liberté. On remarquera que la p-valeur est de 0.007 et est toujours inférieure à 0.05. On rejette donc notre model_step_backward_corr car on lui préfère le modèle saturé. Notre modèle n'est pas encore suffisant.

- Prédiction: Malgré le fait que le model_step_backward_corr soit rejeté, nous allons tout de même calculer les prédictions pour ce modèle pour pouvoir comparer avec les autres modèles. Voici la table de décision ci-dessous avec un seuil de décision fixé à 0.5 pour comparer les modèles. Nous avons des FPR (False Positive Rate) = $P(\tilde{Y} = 1 \mid Y = 0) = 183/579 = 31\%$, une spécificité $396/579 = 0.68 = 1 - \text{FPR}$ et sensibilité TPR (True Positive Rate) = $P(\tilde{Y} = 1 \mid Y = 1) = 465/601 = 77\%$. En tout le model_step_backward_corr donne 72.97% de bonnes prédictions. Ce modèle a un moins bon TPR que le model_step_backward de -1% et le pourcentage de bonnes prédictions est le moins bon de tous les modèles.

Table 12: Table de Décision - Model_step_backward_corr

	FALSE	TRUE
FALSE	396	136
TRUE	183	465

```
## Bonnes prédictions: 72.97%
```

Nous pouvons aussi définir un seuil s optimal. Le seuil optimal reste donc proche de 0.5 et est de 0.58. 12 fausses prédictions sont enlevées. Nous avons aussi calculé la courbe ROC (receiving operator characteristic) avec en abscisse les FPR et en ordonnée les TPR. On voit que la courbe est au-dessus de la droite d'équation $y = x$, notre model_step_backward_corr fait donc mieux que l'aléatoire et l'AUC (Area Under Curve) qui mesure la qualité de la classification est de 80%, ce qui est un très bon score puisqu'il est au-dessus de la classification aléatoire (AUC=50%). L'AUC est inférieur au model_step_backward.

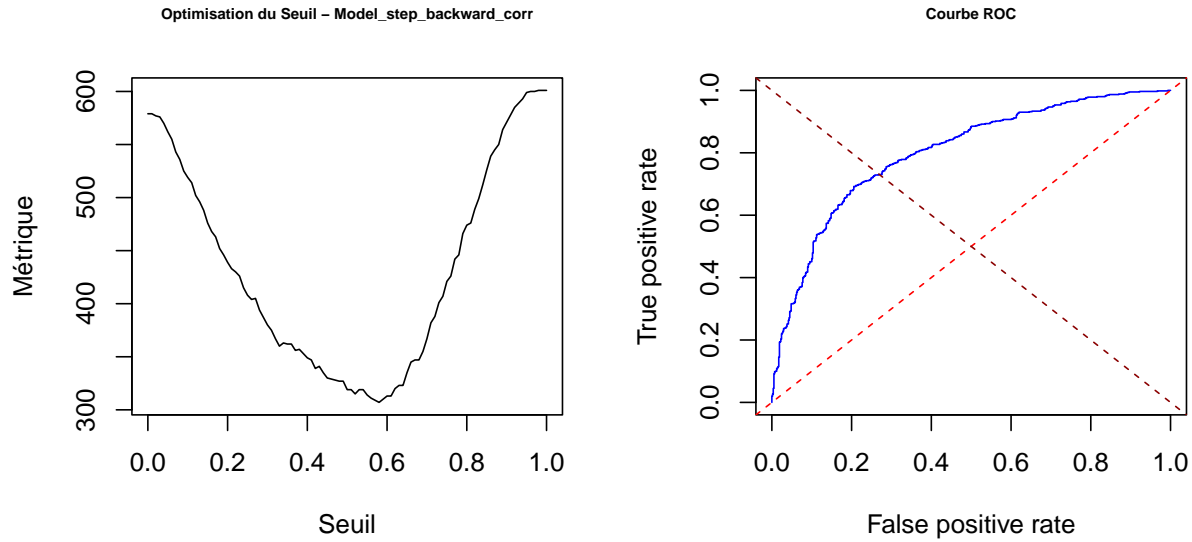


Table 13: Seuil, Coût 0.5, Coût Seuil Optimal, AUC

Metric	Value
Meilleur seuil	0.58
Coût pour le seuil de 0.5	319
Coût pour le meilleur seuil	307
AUC - model_step_backward_corr	0.80

3.1.5 Conclusion choix de modèle logistitique

Nous avons vu que malgré ses défauts le `model_step_backward` reste le meilleur des 4 modèles en terme d'AIC, de deviance, et d'AUC. Nous verrons cela plus loin dans la cross validation des modèles.

3.2 Choix du modèle Probit

Nous allons étudier maintenant les modèles probit. Un modèle probit est un modèle de régression utilisé pour les variables dépendantes binaires. Il est principalement utilisé pour estimer la probabilité qu'un événement se produise. Le modèle probit se définit comme suit $Y_i \sim \text{Bernoulli}(p_i)$ avec $p_i = \Phi(\beta_0 + \beta_1 x_i)$ où Φ est la fonction de répartition de la loi $\mathcal{N}(0, 1)$. Les modèles logit et probit donnent des résultats très proches. Nous allons ici étudier le `model_initial` puis un `model_step` provenant du `model_initial` version probit.

3.2.1 Modèle initial Probit

Nous commencerons par regarder un `model_initial` en fonction de lien probit, qui prend en compte toutes les variables encore présente dans "meteo_train".

```
model_initial_probit <- glm(pluie.demain ~ ., family = binomial(link = "probit"),
  data = meteo_train)
```

Ci dessous est présenté, le summary du model_initial_probit avec les coefficients et la significativité. Le modèle a atteint la convergence vers l'estimateur du maximum de vraisemblance après cinq itérations du scoring de Fisher.

	Estimate	Std..Error	z.value	p.value	Significance
(Intercept)	-44.760	41.334	-1.083	0.279	
Year	0.040	0.020	1.971	0.049	*
Month	-0.012	0.015	-0.857	0.391	
Day	0.007	0.005	1.368	0.171	
TempMean	0.114	0.095	1.203	0.229	
HumidityMean	0.009	0.019	0.498	0.618	
PressureMean	0.288	0.079	3.622	0.000	***
PrecipitationTotal	0.015	0.016	0.946	0.344	
Snowfall	-0.170	0.129	-1.316	0.188	
TotalCloudMean	0.007	0.007	1.050	0.294	
HighCloudMean	-0.002	0.004	-0.408	0.683	
MediumCloudMean	0.003	0.004	0.774	0.439	
LowCloudMean	-0.003	0.005	-0.582	0.560	
Sunshine	0.000	0.001	0.596	0.551	
RadShortwave	0.000	0.000	0.294	0.769	
WindSpeed10mMean	-0.031	0.057	-0.537	0.591	
WindDir10mMean	0.003	0.003	1.003	0.316	
WindSpeed80mMean	-0.056	0.041	-1.384	0.166	
WindDir80mMean	-0.006	0.003	-1.644	0.100	
WindSpeed900mbMean	0.012	0.015	0.821	0.412	
WindDir900mbMean	0.003	0.001	3.702	0.000	***
WindGustMean	0.012	0.021	0.575	0.565	
TempMax	-0.015	0.055	-0.271	0.787	
TempMin	-0.075	0.050	-1.493	0.136	
HumidityMax	0.001	0.012	0.080	0.936	
HumidityMin	-0.002	0.011	-0.197	0.843	
PressureMax	-0.146	0.043	-3.389	0.001	***
PressureMin	-0.180	0.043	-4.201	0.000	***
TotalCloudMax	0.002	0.003	0.576	0.565	
TotalCloudMin	0.004	0.004	1.138	0.255	
HighCloudMax	0.002	0.002	1.202	0.229	
HighCloudMin	0.001	0.011	0.051	0.959	
MediumCloudMax	0.004	0.002	2.066	0.039	*
MediumCloudMin	-0.002	0.005	-0.367	0.714	
LowCloudMax	0.002	0.002	0.894	0.371	
LowCloudMin	0.001	0.004	0.127	0.899	
WindSpeed10mMax	0.036	0.020	1.760	0.078	.
WindSpeed10mMin	0.095	0.037	2.549	0.011	*
WindSpeed80mMax	0.001	0.017	0.081	0.935	
WindSpeed80mMin	-0.030	0.025	-1.207	0.227	
WindSpeed900mbMax	-0.008	0.007	-1.188	0.235	
WindSpeed900mbMin	-0.004	0.011	-0.404	0.686	
WindGustMax	0.013	0.010	1.330	0.183	
WindGustMin	0.004	0.016	0.253	0.800	

Statistic	Value	DF
Null Deviance	1635.417	1179
Residual Deviance	1236.123	1136
Number of Fisher Scoring Iterations	5.000	NA
AIC	1324.123	NA

On remarque dans ce `model_initial_probit` que l'AIC est moins bon que dans le `model_initial` (1324 au lieu de 1320) ainsi que la résiduelle déviance (1236 au lieu de 1232).

- Colinéarité : En utilisant la fonction VIF nous avons parmi les vif les plus importants "TempMean" avec un vif de 251 (`model_initial_logit` = 258) et surtout "PressureMean" avec un vif de 191 (`model_initial_logit` = 189). Sur les 43 variables du modèle, 16 ont un vif inférieur à 7, 3 ont un vif supérieur à 100 et 24 ont un vif entre 7 et 100. Ce modèle a toujours de nombreuses colinéarité en probit.
- Table deviance: Nous utiliserons tout d'abord, la fonction `anova(model_initial_probit, test = "LRT")` présentée dans la version html du report. Dans le `model_initial_probit`, 18 variables réduisent la deviance du le modèle. Ces variables apportent une contribution significative et utile à la prédiction de la variable réponse. Parmi les plus significatives on retrouve les variables: "PressureMin", "PressureMean", "PrecipitationTotal", "TotalCloudMean", "WindDir900mbMean", "HumidityMean", "WindSpeed10mMean", "TempMean".
- Test de deviance: Nous utiliserons ensuite un test de deviance pour comparer notre modèle noté génériquement M_k dans un premier temps à un modèle null M_0 . Nous ferons ensuite des tests du χ^2 de rapport de vraisemblance.

```
pvalMoMkiniprobit <- pchisq(1635.4 - 1236.123, 1179 - 1136, lower = F)
cat("La p-valeur pour la différence de déviance est :", format(pvalMoMkiniprobit,
  scientific = TRUE))
```

```
## La p-valeur pour la différence de déviance est : 2.852602e-59
```

La déviance null du `modele_manuel` est de 1635.4 avec 1179 degrés de liberté et la déviance résiduelle est de 1236.123 pour 1136 degrés de liberté. Avec une p-valeur inférieur à 0.05, le `model_manuel` avec plus de variables est meilleur en termes d'ajustement par rapport au modèle M_0 .

```
pvalMkMsatiniprobit <- pchisq(1236.123, 1136, lower = F)
cat("La p-valeur est:", sprintf("%.3f", pvalMkMsatiniprobit))
```

```
## La p-valeur est: 0.020
```

On testera ensuite M_k contre M_{sat} . La déviance résiduelle du `model_initial_probit` est de 1236.123 pour 1136 degrés de liberté. On remarquera que la p-valeur est de 0.02 et est toujours inférieure à 0.05. On rejette donc notre `model_initial_probit` car on lui préfère le modèle saturé. Notre modèle n'est pas encore suffisant et a p-valeur moins important que le `model_initial`

- Prédiction: Malgré le fait que le `model_initial_probit` soit rejeté, nous allons tout de même calculer les prédictions pour ce modèle pour pouvoir comparer avec les autre modèle. Voici la table de décision ci-dessous avec un seuil de décision fixé à 0.5 pour comparer les modèles. Nous avons des FPR (False Positive Rate) = $P(\tilde{Y} = 1 | Y = 0) = 161/579 = 27\%$, un spécificité $418/579 = 0.72 = 1 - \text{FPR}$ et sensibilité TPR (True Positive Rate) = $P(\tilde{Y} = 1 | Y = 1) = 464/601 = 77\%$. En tout le `model_initial_probit` donne 74.75% de bonnes prédictions. Ce modèle donne des prédictions moins bonnes que le `model_initial`.

Table 15: Table de Décision - Model_initial_probit

	FALSE	TRUE
FALSE	418	137
TRUE	161	464

Bonnes prédictions: 74.75%

Nous pouvons aussi peut définir un seuil s optimal. Le seuil optimal reste donc proche de 0.5 et est de 0.51. 2 fausses prédictions sont enlevées. Nous avons aussi calculé la courbe ROC (receiving operator characteristic) avec en abscisse les FPR et en ordonnée les TPR. On voit que la courbe est au dessus de la droite d'équation $y = x$, notre modele_initial_probit fait donc mieux que l'aléatoire et l'AUC (Area Under Curve) qui mesure la qualité de la classification est de 82%, ce qui est un très bon score puisqu'il est au-dessous de la classification aléatoire (AUC=50%). L'AUC est égale au model_initial.

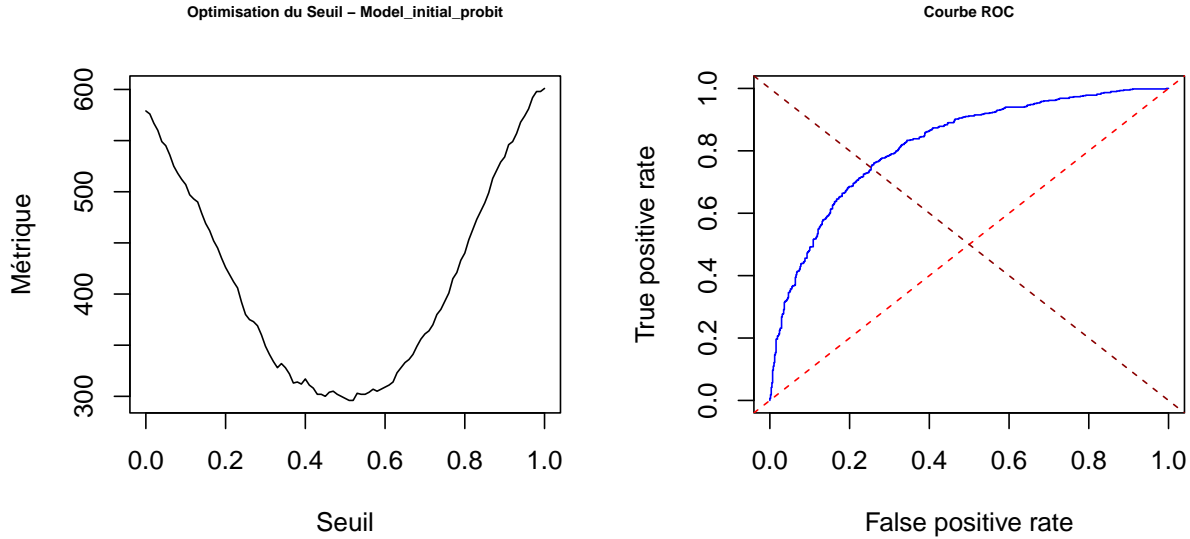


Table 16: Seuil, Coût 0.5, Coût Seuil Optimal, AUC

Metric	Value
Meilleur seuil	0.51
Coût pour le seuil de 0.5	298
Coût pour le meilleur seuil	296
AUC - model_initial_probit	0.82

3.2.2 Modèle initial Step Probit

Nous continuerons par regarder un model_step en fonction de lien probit pour exclure certaines variables du modèle model_initial_probit.

Le modèle probit donné par la fonction step est le suivant:

```
model_step_probit <- glm(pluie.demain ~ Year + TempMean + PressureMean +
  Snowfall + TotalCloudMean + WindSpeed80mMean + WindDir80mMean +
  WindDir900mbMean + TempMin + PressureMax + PressureMin +
  TotalCloudMin + HighCloudMax + MediumCloudMax + LowCloudMax +
  WindSpeed10mMax + WindSpeed10mMin + WindGustMax, family = binomial(link = "probit"),
  data = meteo_train)
```

Ci dessous est présenté, le summary du model_step_probit avec les coefficients et la significativité. Le modèle a atteint la convergence vers l'estimateur du maximum de vraisemblance après cinq itérations du scoring de Fisher.

	Estimate	Std..Error	z.value	p.value	Significance
(Intercept)	-41.803	36.929	-1.132	0.258	
Year	0.040	0.018	2.208	0.027	*
TempMean	0.093	0.032	2.941	0.003	**
PressureMean	0.259	0.074	3.483	0.000	***
Snowfall	-0.170	0.119	-1.428	0.153	
TotalCloudMean	0.004	0.003	1.701	0.089	.
WindSpeed80mMean	-0.066	0.018	-3.742	0.000	***
WindDir80mMean	-0.002	0.001	-2.297	0.022	*
WindDir900mbMean	0.003	0.001	3.530	0.000	***
TempMin	-0.068	0.034	-2.023	0.043	*
PressureMax	-0.134	0.041	-3.305	0.001	***
PressureMin	-0.165	0.040	-4.089	0.000	***
TotalCloudMin	0.005	0.002	1.953	0.051	.
HighCloudMax	0.002	0.001	1.553	0.120	
MediumCloudMax	0.005	0.001	3.237	0.001	**
LowCloudMax	0.002	0.002	1.561	0.119	
WindSpeed10mMax	0.035	0.013	2.623	0.009	**
WindSpeed10mMin	0.061	0.021	2.906	0.004	**
WindGustMax	0.014	0.006	2.200	0.028	*

Statistic	Value	DF
Null Deviance	1635.417	1179
Residual Deviance	1248.701	1161
Number of Fisher Scoring Iterations	5.000	NA
AIC	1286.701	NA

On remarque que le model_step_probit est différent du model_step_backward en termes de variables. L'AIC reste tout de même l'un des meilleurs parmi les modèles avec 1286.

- Colinéarité : En utilisant la fonction VIF nous avons parmi les vif les plus importants "PressureMean avec un vif de 169. Ce modèle a toujours de nombreuses colinéarité en probit sur les variables "Pressure" et "Temp". Il faudrait retirer ses variables ou leurs variables corrélées.
- Table deviance: Nous utiliserons tout d'abord, la fonction anova(model_step_probit, test = "LRT") présentée dans la version html du report. Dans le model_step_probit, 14 variables réduisent la deviance du le modèle. Ces variables apportent une contribution significative et utile à la prédiction de la variable réponse. Parmi les plus significatives on retrouve les variables: "PressureMin", "PressureMean", "TotalCloudMean", "WindDir900mbMean", "WindSpeed10mMax", "TempMean",

“MediumCloudMax”, “HighCloudMax”, “WindSpeed80mMean”. Ensuite nous utiliserons la fonction `anova(model_initial_probit, model_step_probit, test = “LRT”)`. La p-valeur est égale à 0.9813 et nous remarquons que la différence entre les deux modèles n’est pas statistiquement significative. On pourra préférer le `model_step_probit` car il a moins de variables.

- Test de deviance: Nous utiliserons ensuite un test de deviance pour comparer notre modèle noté génériquement M_k dans un premier temps à un modèle null M_0 . Nous ferons ensuite des tests du χ^2 de rapport de vraisemblance.

```
pvalMkMstepprobit <- pchisq(1635.4 - 1248.701, 1179 - 1161,
  lower = F)
cat("La p-valeur pour la différence de déviance est :", format(pvalMkMstepprobit,
  scientific = TRUE))
```

```
## La p-valeur pour la différence de déviance est : 5.405733e-71
```

La déviance null du `modele_manuel` est de 1635.4 avec 1179 degrés de liberté et la déviance résiduelle est de 1248.701 pour 1161 degrés de liberté. Avec une p-valeur inférieur à 0.05, le `model_manuel` avec plus de variables est meilleur en termes d’ajustement par rapport au modèle M_0 .

```
pvalMkMsatstepprobit <- pchisq(1248.701, 1161, lower = F)
cat("La p-valeur est:", sprintf("%.3f", pvalMkMsatstepprobit))
```

```
## La p-valeur est: 0.037
```

On testera ensuite M_k contre M_{sat} . La déviance résiduelle du `model_initial_probit` est de 1248.701 pour 1161 degrés de liberté. On remarquera que la p-valeur est de 0.037 et est toujours inférieure à 0.05. On rejette donc notre `model_step_probit` car on lui préfère le modèle saturé. Notre modèle n’est pas encore suffisant et a p-valeur moins important que le `model_step_backward`.

- Prédiction: Malgré le fait que le `model_step_probit` soit rejeté, nous allons tout de même calculer les prédictions pour ce modèle pour pouvoir comparer avec les autre modèle. Voici la table de décision ci-dessous avec un seuil de décision fixé à 0.5 pour comparer les modèles. Nous avons des FPR (False Positive Rate) = $P(\tilde{Y} = 1 \mid Y = 0) = 167/579 = 29\%$, un spécificité $412/579 = 0.71 = 1 - \text{FPR}$ et sensibilité TPR (True Positive Rate) = $P(\tilde{Y} = 1 \mid Y = 1) = 465/601 = 77\%$. En tout le `model_step_probit` donne 74.32% de bonnes prédictions. Ce modèle donne des prédictions un peu moins bonnes que le `model_initial_probit` et le `model_step_backward`.

Table 18: Table de Décision - `Model_step_probit`

	FALSE	TRUE
FALSE	412	136
TRUE	167	465

```
## Bonnes prédictions: 74.32%
```

Nous pouvons aussi peut définir un seuil s optimal. Le seuil optimal reste donc proche de 0.5 et est de 0.46. 8 fausses prédictions sont enlevées. Nous avons aussi calculé la courbe ROC (receiving operator characteristic) avec en abscisse les FPR et en ordonnée les TPR. On voit que la courbe est au dessus de la droite d’équation $y = x$, notre `modele_step_probit` fait donc mieux que l’aléatoire et l’AUC (Area Under Curve) qui mesure la

qualité de la classification est de 82%, ce qui est un très bon score puisqu'il est au-dessous de la classification aléatoire (AUC=50%). L'AUC est égale au `model_initial_probit`.

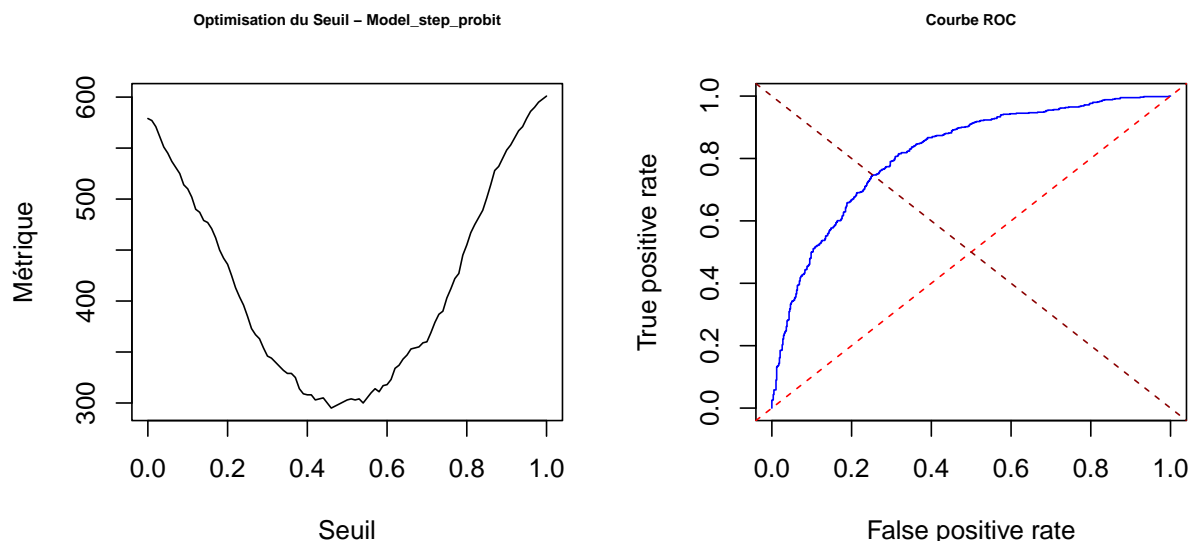


Table 19: Seuil, Coût 0.5, Coût Seuil Optimal, AUC

Metric	Value
Meilleur seuil	0.46
Coût pour le seuil de 0.5	303
Coût pour le meilleur seuil	295
AUC - <code>model_step_probit</code>	0.82

3.2.3 Conclusion choix de modèle probit

Nous avons vu que malgré ses défauts le `model_step_probit` reste le meilleur des 2 modèles en terme d'AIC, de deviance, et d'AUC. Nous verrons cela plus loin dans la cross validation des modèles.

4. Cross Validation des modèles

Pour faire la cross validation des modèles nous choisirons, la cross validation k-fold qui est une méthode de validation des modèles qui permet de mieux estimer la performance d'un modèle en utilisant des échantillons répétés de l'ensemble des données. Nous choisissons $k = 10$ qui est le nombre de fold. Pour chaque fold i de 1 à 10, on va entraîner le modèle, faire des prédictions sur le fold de test pour chaque modèle et ensuite calculer la proportion de bonnes prédictions ("Mean Accuracy") et calculer l'erreur moyenne ("Mean Error"). Les résultats sont présentés ci-dessous:

Model	Mean_Accuracy	Mean_Error
<code>model_initial</code>	0.7252072	0.3598578
<code>model_manuel</code>	0.7311159	0.3743294
<code>model_step_backward</code>	0.7452907	0.3564279
<code>model_step_backward_corr</code>	0.7215905	0.3706786

Model	Mean_Accuracy	Mean_Error
model_initial_probit	0.7243586	0.3619018
model_step_probit	0.7333543	0.3590594

Nous pouvons constater qu'à chaque run le `model_step_backward` à l'erreur la plus petite et le pourcentage de bonnes prédictions le plus élevé pour tous les modèles évalués. On sélectionnera donc ce modèle pour faire nos prédictions sur nos données "meteo_test" malgré ses inconvénients.

5. Prédictions sur meteo_test

Nous commençons par charger le fichier "meteo_test" en appliquant le même traitement que pour "meteo_train" sur la variable X. Nous allons également renommer les variables pour avoir les mêmes noms entre les deux fichiers.

```
meteo_test <- read_csv("C:/Documents/Dauphine/Module 2/Modèle Linéaire Généralisé/Projet/meteo.test.csv",
  show_col_types = FALSE, name_repair = "minimal")
meteo_test <- meteo_test[, -1]
```

En regardant le summary de "meteo_test" on s'aperçoit que les variables "Hour" et "Minute" sont semblables à celles de "meteo_train" avec une valeur constante à 0. Nous allons donc aussi exclure ces variables.

Nous continuerons en effectuant les prédictions avec le modèle sélectionné: `model_step_backward`.

```
pred_test = predict(model_step_backward, meteo_test, type = "response")
pluie.demain = (pred_test >= 0.5)
```

Nous allons ensuite imprimer nos résultats dans la table "meteo_test" et sortir un fichier de prédictions nommé : "Prediction_Sophie_ROBERT_OKADA.csv".

```
predpluie <- cbind(meteo_test, pluie.demain)
View(predpluie)
write.table(predpluie, "Prediction_Sophie_ROBERT_OKADA.csv",
  sep = ";", col.names = TRUE)
```

6. Conclusion du Projet

Après une longue recherche sur le meilleur modèle, il n'a pas été simple de choisir parmi les modèles considérés. Le `model_step_backward` semble être le meilleur parmi ceux testés, mais il présente des défauts. Il aurait peut-être été judicieux de considérer le `model_manuel`, plus simple, bien que ses prédictions soient moins précises et que son erreur soit plus importante. Cela aurait également permis de réduire les risques d'overfitting.

En conclusion, la prévision météorologique reste une tâche complexe en raison des interactions multiples entre les variables. Pour analyser ces variables, une analyse en composantes principales (ACP) aurait pu être plus pertinente dans le contexte de la prévision météorologique car cela nous aurait permis de réduire la dimensionnalité et d'améliorer la performance en réduisant la complexité du modèle.

7. Bibliographie

- Cours Robin RYDER - Modèles Linéaires Généralisés
- Cours Vincent Rivoiard - Choix de Modèles

8. Annexes



8.1 Annexe 1 - Summary des données



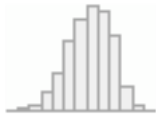
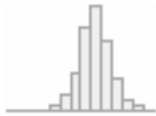



Summary Meteo_Train

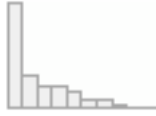
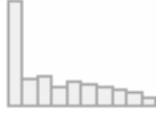
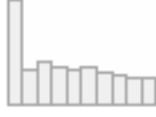
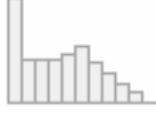
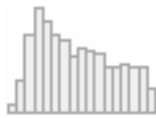
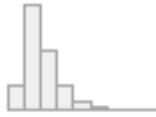
meteo_train

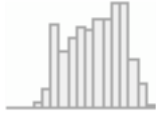
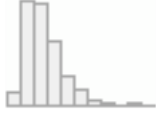
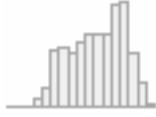
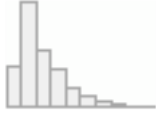
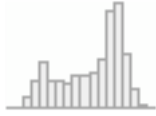
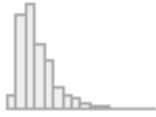
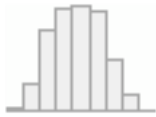
Dimensions: 1180 x 44


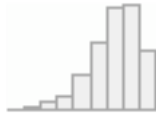
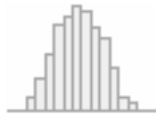
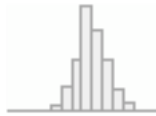
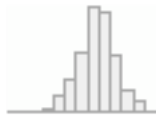


Duplicates: 0








No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Missing
1	Year [numeric]	Mean (sd) : 2013.9 (2.4) min < med < max: 2010 < 2014 < 2018 IQR (CV) : 4 (0)	2010 : 86 (7.3%) 2011 : 153 (13.0%) 2012 : 149 (12.6%) 2013 : 144 (12.2%) 2014 : 147 (12.5%) 2015 : 143 (12.1%) 2016 : 143 (12.1%) 2017 : 143 (12.1%) 2018 : 72 (6.1%)		0 (0.0%)
2	Month [numeric]	Mean (sd) : 6.4 (3.4) min < med < max: 1 < 6 < 12 IQR (CV) : 6 (0.5)	12 distinct values		0 (0.0%)

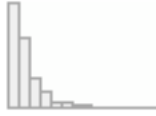
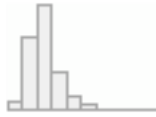
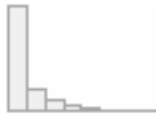
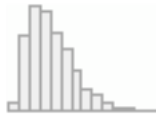
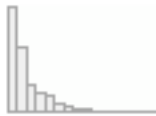
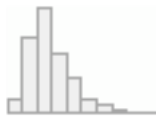
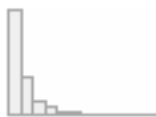
No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Missing
3	Day [numeric]	Mean (sd) : 15.8 (8.8) min < med < max: 1 < 16 < 31 IQR (CV) : 15 (0.6)	31 distinct values		0 (0.0%)
4	TempMean [numeric]	Mean (sd) : 12.2 (7) min < med < max: -7.6 < 12.1 < 29.4 IQR (CV) : 10.8 (0.6)	961 distinct values		0 (0.0%)
5	HumidityMean [numeric]	Mean (sd) : 71.4 (9.5) min < med < max: 38.3 < 72.2 < 95.5 IQR (CV) : 13.8 (0.1)	646 distinct values		0 (0.0%)
6	PressureMean [numeric]	Mean (sd) : 1017 (7.9) min < med < max: 978.9 < 1017 < 1042.4 IQR (CV) : 9.6 (0)	967 distinct values		0 (0.0%)
7	PrecipitationTotal [numeric]	Mean (sd) : 2.1 (4.1) min < med < max: 0 < 0.1 < 31.5 IQR (CV) : 2.3 (1.9)	148 distinct values		0 (0.0%)
8	Snowfall [numeric]	Mean (sd) : 0 (0.4) min < med < max: 0 < 0 < 8.6 IQR (CV) : 0 (8)	27 distinct values		0 (0.0%)
9	TotalCloudMean [numeric]	Mean (sd) : 50.8 (31.7) min < med < max: 0 < 51.7 < 100 IQR (CV) : 54.7 (0.6)	1007 distinct values		0 (0.0%)


No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Missing
10	HighCloudMean [numeric]	Mean (sd) : 20.3 (21.8) min < med < max: 0 < 11.9 < 100 IQR (CV) : 31.6 (1.1)	690 distinct values		0 (0.0%)
11	MediumCloudMean [numeric]	Mean (sd) : 31.5 (29.8) min < med < max: 0 < 25 < 100 IQR (CV) : 52.4 (0.9)	746 distinct values		0 (0.0%)
12	LowCloudMean [numeric]	Mean (sd) : 39.3 (31.1) min < med < max: 0 < 36.4 < 100 IQR (CV) : 56.3 (0.8)	811 distinct values		0 (0.0%)
13	Sunshine [numeric]	Mean (sd) : 373.1 (275.1) min < med < max: 0 < 366.8 < 1015.8 IQR (CV) : 473.5 (0.7)	1022 distinct values		0 (0.0%)
14	RadShortwave [numeric]	Mean (sd) : 3984.6 (2153.1) min < med < max: 265.2 < 3675.3 < 8363.3 IQR (CV) : 3627.4 (0.5)	1093 distinct values		0 (0.0%)
15	WindSpeed10mMean [numeric]	Mean (sd) : 10.7 (6.1) min < med < max: 1.3 < 9.2 < 42.2 IQR (CV) : 6.6 (0.6)	829 distinct values		0 (0.0%)

No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Missing
16	WindDir10mMean [numeric]	Mean (sd) : 201.8 (61.6) min < med < max: 11.2 < 206.4 < 331.7 IQR (CV) : 101.8 (0.3)	1139 distinct values		0 (0.0%)
17	WindSpeed80mMean [numeric]	Mean (sd) : 14.3 (7.9) min < med < max: 1.3 < 12.4 < 54 IQR (CV) : 8.9 (0.6)	930 distinct values		0 (0.0%)
18	WindDir80mMean [numeric]	Mean (sd) : 206.2 (62.3) min < med < max: 12.2 < 213.8 < 333.4 IQR (CV) : 101.6 (0.3)	1151 distinct values		0 (0.0%)
19	WindSpeed900mbMean [numeric]	Mean (sd) : 24.6 (16) min < med < max: 2.2 < 19.6 < 97.1 IQR (CV) : 19.1 (0.7)	1041 distinct values		0 (0.0%)
20	WindDir900mbMean [numeric]	Mean (sd) : 206.2 (74.5) min < med < max: 17.4 < 233.5 < 344.8 IQR (CV) : 121.9 (0.4)	1144 distinct values		0 (0.0%)
21	WindGustMean [numeric]	Mean (sd) : 16.7 (10.3) min < med < max: 2.2 < 14.1 < 79.4 IQR (CV) : 11.7 (0.6)	929 distinct values		0 (0.0%)
22	TempMax [numeric]	Mean (sd) : 16.5 (7.7) min < med < max: -3.8 < 16.5 < 35.8 IQR (CV) : 11.8 (0.5)	961 distinct values		0 (0.0%)

No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Missing
23	TempMin [numeric]	Mean (sd) : 8.1 (6.3) min < med < max: -12.5 < 8 < 23.9 IQR (CV) : 9.7 (0.8)	927 distinct values		0 (0.0%)
24	HumidityMax [numeric]	Mean (sd) : 87.7 (7.6) min < med < max: 59 < 89 < 100 IQR (CV) : 11 (0.1)	41 distinct values		0 (0.0%)
25	HumidityMin [numeric]	Mean (sd) : 54 (12.5) min < med < max: 19 < 54 < 92 IQR (CV) : 18 (0.2)	66 distinct values		0 (0.0%)
26	PressureMax [numeric]	Mean (sd) : 1019.9 (7.6) min < med < max: 981.9 < 1019.5 < 1045.4 IQR (CV) : 9.3 (0)	320 distinct values		0 (0.0%)
27	PressureMin [numeric]	Mean (sd) : 1014.2 (8.5) min < med < max: 977 < 1014.6 < 1038.6 IQR (CV) : 10.2 (0)	352 distinct values		0 (0.0%)
28	TotalCloudMax [numeric]	Mean (sd) : 88.2 (29.5) min < med < max: 0 < 100 < 100 IQR (CV) : 0 (0.3)	79 distinct values		0 (0.0%)
29	TotalCloudMin [numeric]	Mean (sd) : 8.7 (23) min < med < max: 0 < 0 < 100 IQR (CV) : 2.4 (2.6)	107 distinct values		0 (0.0%)

No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Missing
30	HighCloudMax [numeric]	Mean (sd) : 60.2 (43) min < med < max: 0 < 97 < 100 IQR (CV) : 85 (0.7)	91 distinct values		0 (0.0%)
31	HighCloudMin [numeric]	Mean (sd) : 0.9 (5.1) min < med < max: 0 < 0 < 100 IQR (CV) : 0 (5.4)	28 distinct values		0 (0.0%)
32	MediumCloudMax [numeric]	Mean (sd) : 70.9 (42.6) min < med < max: 0 < 100 < 100 IQR (CV) : 77.2 (0.6)	76 distinct values		0 (0.0%)
33	MediumCloudMin [numeric]	Mean (sd) : 2.1 (11.3) min < med < max: 0 < 0 < 100 IQR (CV) : 0 (5.4)	42 distinct values		0 (0.0%)
34	LowCloudMax [numeric]	Mean (sd) : 80 (38.4) min < med < max: 0 < 100 < 100 IQR (CV) : 0 (0.5)	47 distinct values		0 (0.0%)
35	LowCloudMin [numeric]	Mean (sd) : 3.9 (16.8) min < med < max: 0 < 0 < 100 IQR (CV) : 0 (4.3)	46 distinct values		0 (0.0%)
36	WindSpeed10mMax [numeric]	Mean (sd) : 19.1 (9.4) min < med < max: 2.5 < 17.4 < 80 IQR (CV) : 11.1 (0.5)	773 distinct values		0 (0.0%)

No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Missing
37	WindSpeed10mMin [numeric]	Mean (sd) : 3.6 (3.8) min < med < max: 0 < 2.4 < 27.7 IQR (CV) : 3.3 (1.1)	247 distinct values		0 (0.0%)
38	WindSpeed80mMax [numeric]	Mean (sd) : 25.4 (10.9) min < med < max: 4 < 23.9 < 93.8 IQR (CV) : 11.6 (0.4)	889 distinct values		0 (0.0%)
39	WindSpeed80mMin [numeric]	Mean (sd) : 4.7 (5.7) min < med < max: 0 < 2.6 < 37.7 IQR (CV) : 4.7 (1.2)	316 distinct values		0 (0.0%)
40	WindSpeed900mbMax [numeric]	Mean (sd) : 41.8 (22.5) min < med < max: 4 < 37.1 < 136.2 IQR (CV) : 29.8 (0.5)	1021 distinct values		0 (0.0%)
41	WindSpeed900mbMin [numeric]	Mean (sd) : 11.1 (11.9) min < med < max: 0 < 6.7 < 76.1 IQR (CV) : 12.3 (1.1)	565 distinct values		0 (0.0%)
42	WindGustMax [numeric]	Mean (sd) : 29.3 (14.9) min < med < max: 4.3 < 26.1 < 95 IQR (CV) : 18 (0.5)	185 distinct values		0 (0.0%)
43	WindGustMin [numeric]	Mean (sd) : 6.5 (7.1) min < med < max: 0 < 4 < 58 IQR (CV) : 6.1 (1.1)	96 distinct values		0 (0.0%)

No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Missing
44	pluie.demain [logical]	1. FALSE 2. TRUE	579 (49.1%) 601 (50.9%)		0 (0.0%)

8.2 Annexe 2 - Lien GitHub

Le lien vers le projet sur GitHub est le suivant:

Voir le projet sur GitHub