Rapport TP1

Alexis Schneider - Kelvin Wong

29-09-2025

Contents

IV. Application : Étude de données avec transformation

Dans cet exercice, l'objectif est d'analyser la croissance d'une entreprise technologique majeure, Amazon, sur plusieurs décennies. La croissance de telles entreprises est souvent spectaculaire et rarement linéaire. Nous allons donc tenter de modéliser l'évolution de son chiffre d'affaires annuel à l'aide d'un modèle de régression linéaire. Comme nous le verrons, une transformation des données sera nécessaire pour obtenir un modèle pertinent.

Le jeu de données que nous avons constitué retrace le chiffre d'affaires annuel d'Amazon de 1999 à 2023, exprimé en milliards de dollars américains. Pour obtenir ces valeurs, nous avons agrégé les données brutes d'un jeu de données public sur Kaggle (https://www.kaggle.com/datasets/phannguynhuphong/amazon-revenue/data), qui recense les revenus trimestriels, en sommant les quatre trimestres de chaque année.

```
data<-read.table("amazone_revenue.csv",header=TRUE,sep = ",")
print(data)
```

```
##    ANNEE REVENU_MILLIARDS_USD
## 1   1999                 1.64
## 2   2000                 2.76
## 3   2001                 3.12
## 4   2002                 3.93
## 5   2003                 5.26
## 6   2004                 6.92
## 7   2005                 8.49
## 8   2006                10.71
## 9   2007                14.84
## 10  2008                19.17
## 11  2009                24.51
## 12  2010                34.20
## 13  2011                48.08
## 14  2012                61.09
## 15  2013                74.45
## 16  2014                88.99
## 17  2015               107.01
## 18  2016               135.99
## 19  2017               177.87
## 20  2018               232.89
## 21  2019               280.52
## 22  2020               386.06
## 23  2021               469.82
## 24  2022               513.98
## 25  2023               574.78
```

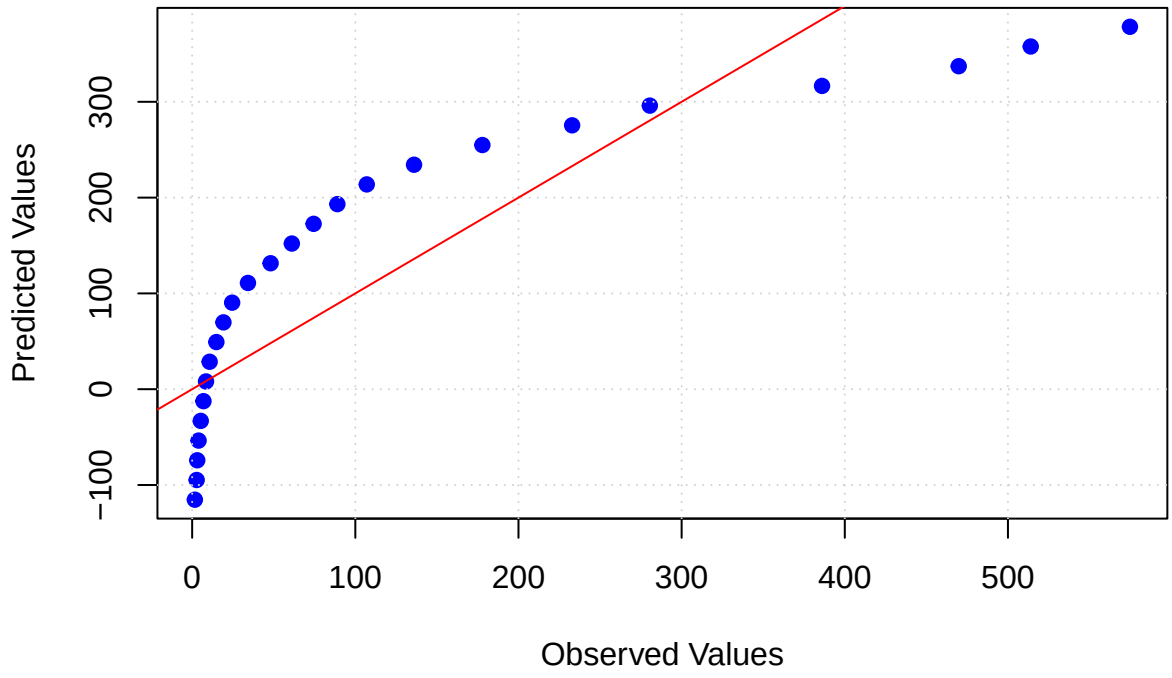
Notre colonne cible est donc `REVENU_MILLIARDS_USD` qui correspond à la valeur du chiffre d'affaire en millaird de dollars qu'a réalisé Amazon par an. Notre première approche consiste à appliquer un modèle de régression linéaire simple pour voir s'il existe une relation linéaire directe entre l'année et le chiffre d'affaires.

```
Y<-data$REVENU_MILLIARDS_USD
X <- data$ANNEE
initial_fit <- lm(REVENU_MILLIARDS_USD ~ ANNEE, data = data)
summary(initial_fit)
```

```
##
```

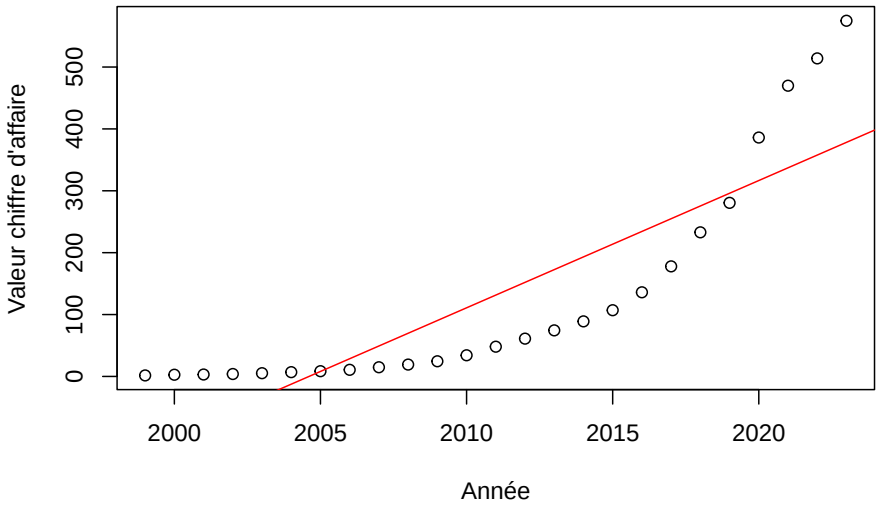
```
## Call:
## lm(formula = REVENU_MILLIARDS_USD ~ ANNEE, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -106.75  -77.02  -17.93   69.46  196.48
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -41231.275   5185.838  -7.951 4.77e-08 ***
## ANNEE           20.568      2.579   7.976 4.51e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 92.98 on 23 degrees of freedom
## Multiple R-squared:  0.7345, Adjusted R-squared:  0.7229
## F-statistic: 63.62 on 1 and 23 DF,  p-value: 4.514e-08
```

```r
plot(data$REVENU_MILLIARDS_USD, fitted(initial_fit), xlab="Observed Values", ylab="Predicted Values",
     pch=19, col="blue")
grid()
abline(a=0, b=1, col="red")
```



On observe sur le graphique que le modèle proposé ne correspond visiblement pas. De plus, notre coefficient de détermination R^2 nous indique que notre modèle n'est potentiellement pas bon (cette valeur est fiable car nous travaillons ici dans un modèle avec peu de dimensions).

```r
plot(data$ANNEE,Y,xlab="Année",ylab="Valeur chiffre d'affaire")
abline(initial_fit,col="red")
```

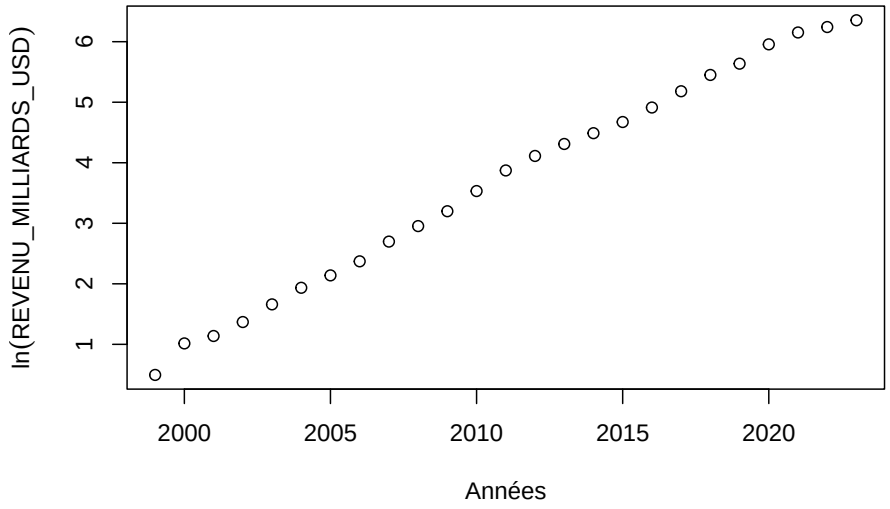


Le graphique ci-dessus est sans appel : les données ne suivent absolument pas une tendance linéaire. On observe une courbure très prononcée, caractéristique d'une **croissance exponentielle**. Le modèle linéaire sous-estime systématiquement le chiffre d'affaires dans les premières et dernières années, et le surestime au milieu. Le modèle est donc inapproprié..

Pour ce faire, on peut appliquer à notre colonne cible (ici Y), la transformation suivante : $\widetilde{Y} = \ln(Y)$. Ainsi, on transforme les données de façon à ce qu'elles soient plus adaptés à un modèle linéaire.

Voici donc le nouveau jeu de données :

```r
data$Log_CA <- log(Y)
plot(data$ANNEE, data$Log_CA, xlab="Années", ylab=expression(ln(REVENU_MILLIARDS_USD)))
```



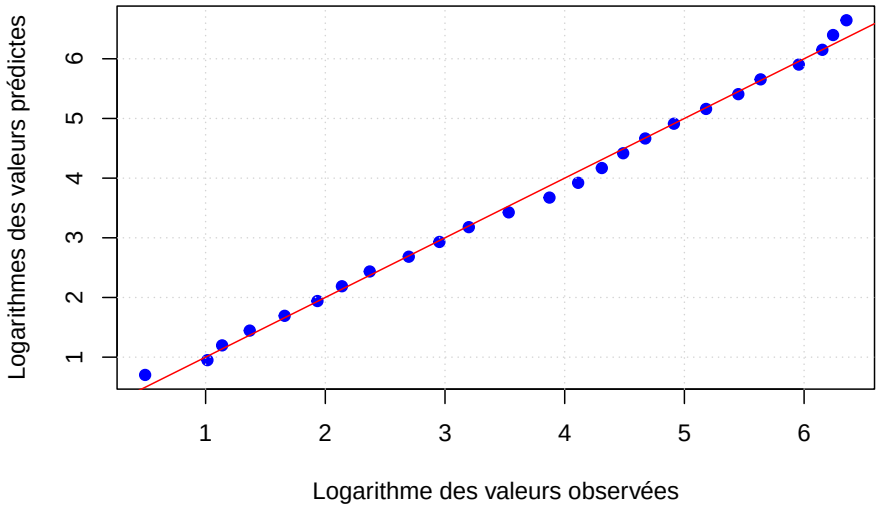
Le jeu semble désormais mieux se prêter à une régression linéaire.

```
log_fit <- lm(Log_CA ~ ANNEE, data = data)
summary(log_fit)
```

```
##
## Call:
## lm(formula = Log_CA ~ ANNEE, data = data)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.291523 -0.049065  0.008549  0.053409  0.199061
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -4.943e+02  6.291e+00  -78.57   <2e-16 ***
## ANNEE        2.476e-01  3.128e-03   79.16   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1128 on 23 degrees of freedom
## Multiple R-squared:  0.9963, Adjusted R-squared:  0.9962
## F-statistic:  6266 on 1 and 23 DF,  p-value: < 2.2e-16
```

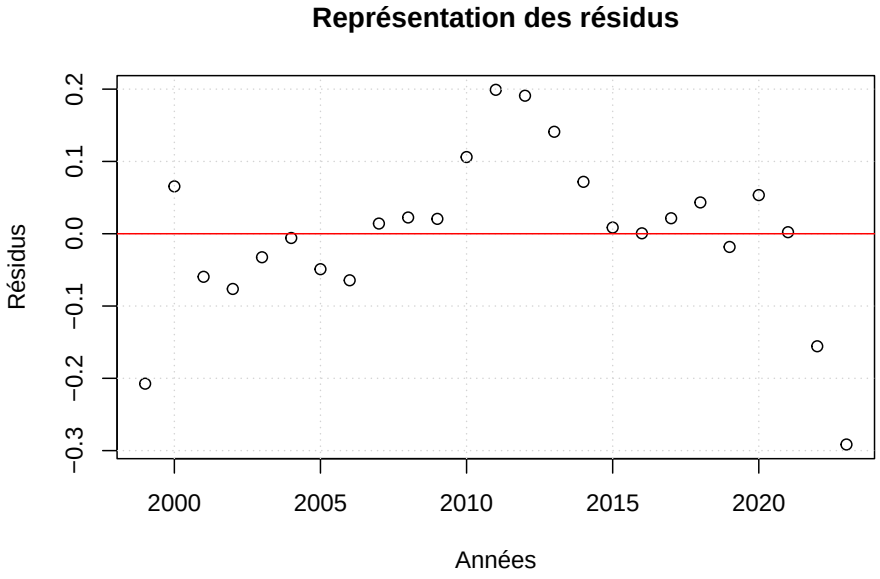
Et visuellement, ça donne ceci :

```
plot(data$Log_CA, fitted(log_fit), xlab="Logarithme des valeurs observées",
     ylab="Logarithmes des valeurs prédictes", pch=19, col="blue")
grid()
abline(a=0, b=1, col="red")
```



Et pour ce qui est des résidus ε de notre modèle :

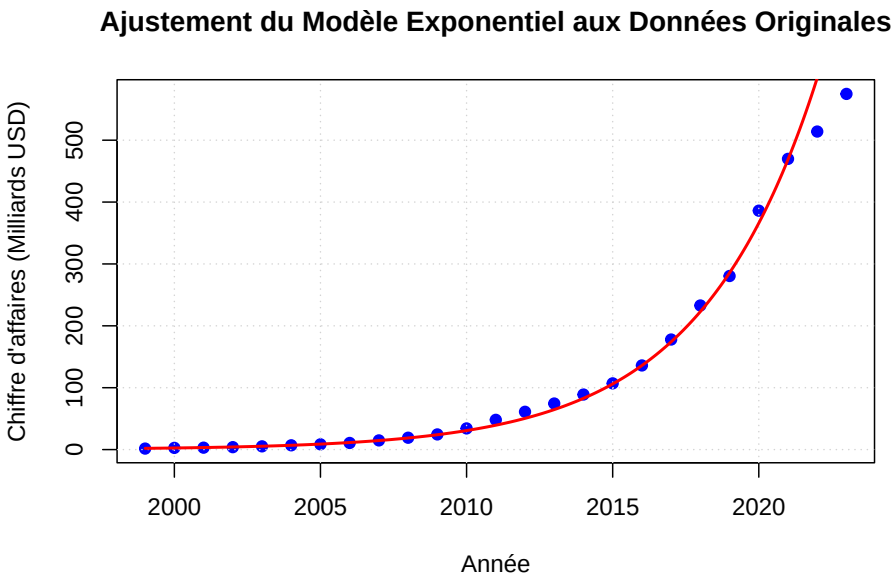
```
plot(data$ANNEE, residuals(log_fit), xlab="Années", ylab="Résidus",
     main="Représentation des résidus")
grid()
abline(h=0, col="red")
```



Le modèle proposé semble donc correspondre beaucoup mieux dans le cadre d'une régression linéaire. Notre coefficient de détermination R^2 est bien meilleur tout comme notre RSE (Residual squared error).

Si l'on souhaite revenir à nos anciennes valeurs tout en gardant la solution proposée par ce modèle (donc $\ln(Y) = at + b$), il nous suffit de visualiser $Y = e^{at+b}$.

```r
x_seq <- seq(min(data$ANNEE), max(data$ANNEE), by = 0.1)
a <- coef(log_fit)["ANNEE"]
b <- coef(log_fit)["(Intercept)"]
y_pred_exp <- exp(a * x_seq + b)

plot(data$ANNEE, Y,
     pch=19,
     col="blue",
     xlab="Année",
     ylab="Chiffre d'affaires (Milliards USD)",
     main="Ajustement du Modèle Exponentiel aux Données Originales")
grid()
lines(x_seq, y_pred_exp, col="red", lwd=2)
```



Le graphique final, **"Ajustement du Modèle Exponentiel aux Données Originales"**, est la validation de notre démarche. En appliquant une transformation logarithmique au chiffre d'affaires, nous avons pu construire un modèle linéaire très performant sur les données transformées. Ce dernier graphique montre le résultat de ce modèle une fois ramené à l'échelle d'origine :

La courbe rouge (le modèle) épouse fidèlement la trajectoire des points de données bleus (les revenus réels), capturant avec une grande précision à la fois la phase de croissance modérée des premières années et l'accélération spectaculaire des années récentes.

Cette excellente adéquation visuelle est corroborée par le coefficient de détermination R^2 très élevé que nous avons obtenu pour le modèle `log_fit`.

V. Electricity Data Set

Dans ce second exercice, nous cherchons à expliquer la consommation totale d'électricité (`Total`) de la ville de Mexico City à l'aide de plusieurs variables météorologiques et calendaires. L'objectif est de construire un modèle de régression linéaire multiple robuste pour identifier les principaux facteurs influençant la consommation électrique.

```
data_Mexico <- read.csv("Mexico_data.csv")
print(summary(data_Mexico))
```

```
##       X0                  RH              SSRD              STRD
##  Length:1461        Min.   :28.37   Min.   : 407046   Min.   :1019376
##  Class :character   1st Qu.:50.12   1st Qu.: 719323   1st Qu.:1156351
##  Mode  :character   Median :57.66   Median : 870071   Median :1235306
##                     Mean   :57.55   Mean   : 869055   Mean   :1244835
##                     3rd Qu.:64.55   3rd Qu.:1022083   3rd Qu.:1353442
##                     Max.   :80.54   Max.   :1217895   Max.   :1431876
##       T2M             T2Mmax          T2Mmin           Covid
##  Min.   :11.23   Min.   :17.82   Min.   : 5.536   Min.   : 0.00
##  1st Qu.:18.09   1st Qu.:25.27   1st Qu.:12.064   1st Qu.: 0.00
##  Median :22.32   Median :28.72   Median :16.220   Median :33.33
##  Mean   :21.27   Mean   :27.93   Mean   :15.623   Mean   :33.00
##  3rd Qu.:24.63   3rd Qu.:30.83   3rd Qu.:19.747   3rd Qu.:63.89
##  Max.   :27.39   Max.   :33.83   Max.   :21.841   Max.   :82.41
##     Holidays            DOW         TOY             Total
##  Min.   :0.00000   Min.   :0   Min.   :  1.0   Min.   : 578.5
##  1st Qu.:0.00000   1st Qu.:1   1st Qu.: 92.0   1st Qu.: 816.0
##  Median :0.00000   Median :3   Median :183.0   Median : 871.4
##  Mean   :0.03012   Mean   :3   Mean   :183.1   Mean   : 880.7
##  3rd Qu.:0.00000   3rd Qu.:5   3rd Qu.:274.0   3rd Qu.: 958.1
##  Max.   :1.00000   Max.   :6   Max.   :366.0   Max.   :1107.4
```

Le résumé (summary) montre qu'il n'y a pas de valeurs manquantes (NA's), ce qui est une bonne nouvelle. Les échelles des variables comme SSRD et STRD sont beaucoup plus grandes que les autres, mais la régression linéaire gère bien cela. La variable X0 (la date) ne sera pas utilisée directement dans le modèle, car nous avons déjà des variables temporelles comme DOW et TOY.

Commençons par une approche intuitive mais risquée : construire un modèle en incluant toutes les variables à l'exception de X0

```
model_naif <- lm(Total ~ T2M + T2Mmax + T2Mmin + RH + SSRD + STRD + Covid + Holidays + DOW + TOY,
                 data = data_Mexico)
print(summary(model_naif))
```

```
##
## Call:
## lm(formula = Total ~ T2M + T2Mmax + T2Mmin + RH + SSRD + STRD +
##     Covid + Holidays + DOW + TOY, data = data_Mexico)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -170.43  -33.41    4.57   35.44  191.36
##
```

```
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.289e+02  9.101e+01   3.614 0.000312 ***
## T2M          2.277e-01  8.496e+00   0.027 0.978622
## T2Mmax      -4.920e+00  4.348e+00  -1.131 0.258037
## T2Mmin       6.313e+00  5.842e+00   1.081 0.280094
## RH          -7.449e-01  4.701e-01  -1.585 0.113241
## SSRD         1.738e-04  2.251e-05   7.722 2.12e-14 ***
## STRD         4.174e-04  9.395e-05   4.442 9.58e-06 ***
## Covid       -2.867e-01  4.964e-02  -5.776 9.35e-09 ***
## Holidays    -8.732e+01  8.057e+00 -10.838  < 2e-16 ***
## DOW         -1.305e+01  6.835e-01 -19.090  < 2e-16 ***
## TOY          5.072e-02  1.716e-02   2.955 0.003172 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 51.95 on 1450 degrees of freedom
## Multiple R-squared:  0.7035, Adjusted R-squared:  0.7014
## F-statistic:   344 on 10 and 1450 DF,  p-value: < 2.2e-16
```

À première vue, le modèle semble performant avec un R^2 ajusté de 0.7014. Cependant, une inspection des coefficients révèle des incohérences majeures.Le modèle traite tout d'abord à tort le jour de la semaine (DOW) et le jour de l'année (TOY) comme des variables numériques linéaires, ignorant que les jours de la semaine sont des catégories distinctes et que la consommation annuelle suit un cycle saisonnier. Les coefficients liés à la température sont aussi problématiques. Le modèle attribue des effets contradictoires et non significatifs : un effet négatif à la température maximale (T2Mmax), et des effets positifs à la température moyenne (T2M) et minimale (T2Mmin). Le fait qu'aucun de ces coefficients ne soit statistiquement significatif est la véritable alerte : cela contredit la logique physique la plus élémentaire, qui veut que la température soit un moteur majeur de la consommation d'électricité (via la climatisation et le chauffage).

Dans un premier temps, nous corrigeons le problème de la structure des variables calendaires.

La fonction factor(DOW) est utilisée car le jour de la semaine (DOW) a un impact catégoriel et non linéaire. Le modèle naïf, en le traitant comme un nombre, échoue à capturer la différence fondamentale entre les jours de la semaine et le week-end. L'utilisation de factor() permet d'estimer un effet distinct pour chaque jour.

De même, le modèle naïf traite TOY (jour de l'année) comme une variable numérique linéaire, ce qui implique une augmentation ou diminution continue de la consommation du 1er janvier au 31 décembre. Or, la consommation est cyclique : le 31 décembre est, en termes de saison, très proche du 1er janvier. Pour modéliser correctement ce cycle, nous décomposons la variable en une paire de fonctions trigonométriques.

```r
data_Mexico$sin_TOY <- sin(2 * pi * data_Mexico$TOY / 365.25)
data_Mexico$cos_TOY <- cos(2 * pi * data_Mexico$TOY / 365.25)
```

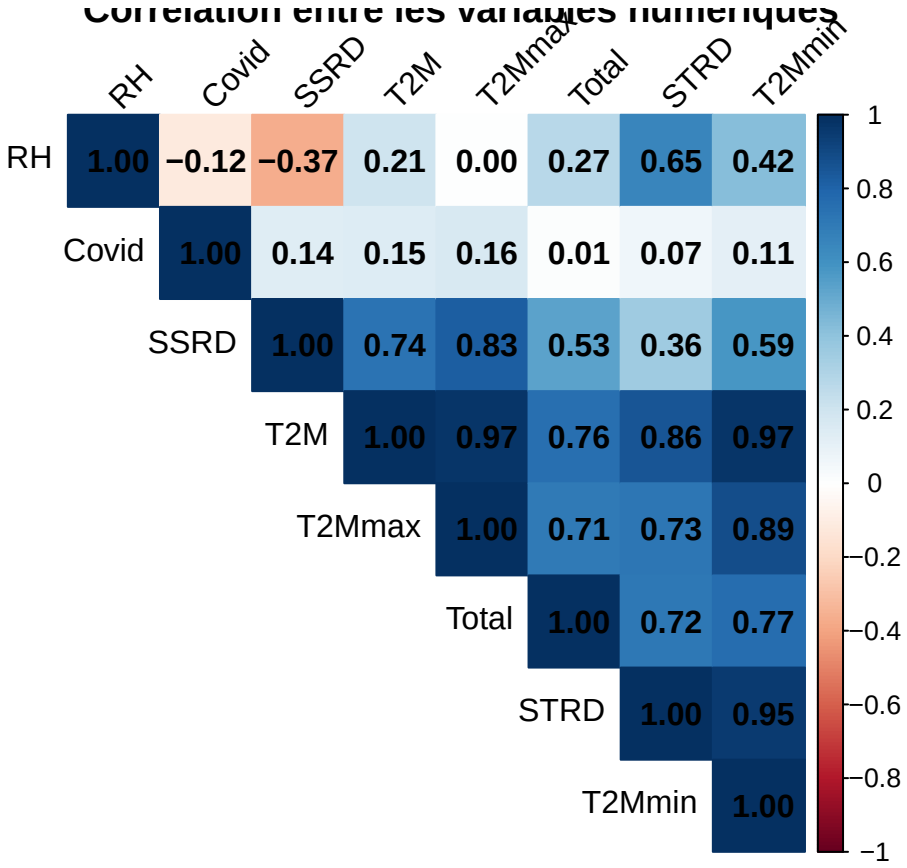
En combinant ces deux ondes saisonnières décalées, le modèle peut désormais reconstruire le cycle annuel spécifique de la consommation, en ajustant son pic (phase) et son intensité (amplitude).

Interessons nous au probleme de la température maintenant. Cette instabilité des coefficients au niveau de la température est un symptôme classique de la multicolinéarité, qui se produit lorsque les variables prédictives sont fortement corrélées entre elles. Le modèle ne parvient pas à distinguer leurs effets respectifs. Pour vérifier cette hypothèse et quantifier la redondance entre nos variables météorologiques, nous allons visualiser leur matrice de corrélation.

```r
if (!require(corrplot)) {
  install.packages("corrplot")
}
```

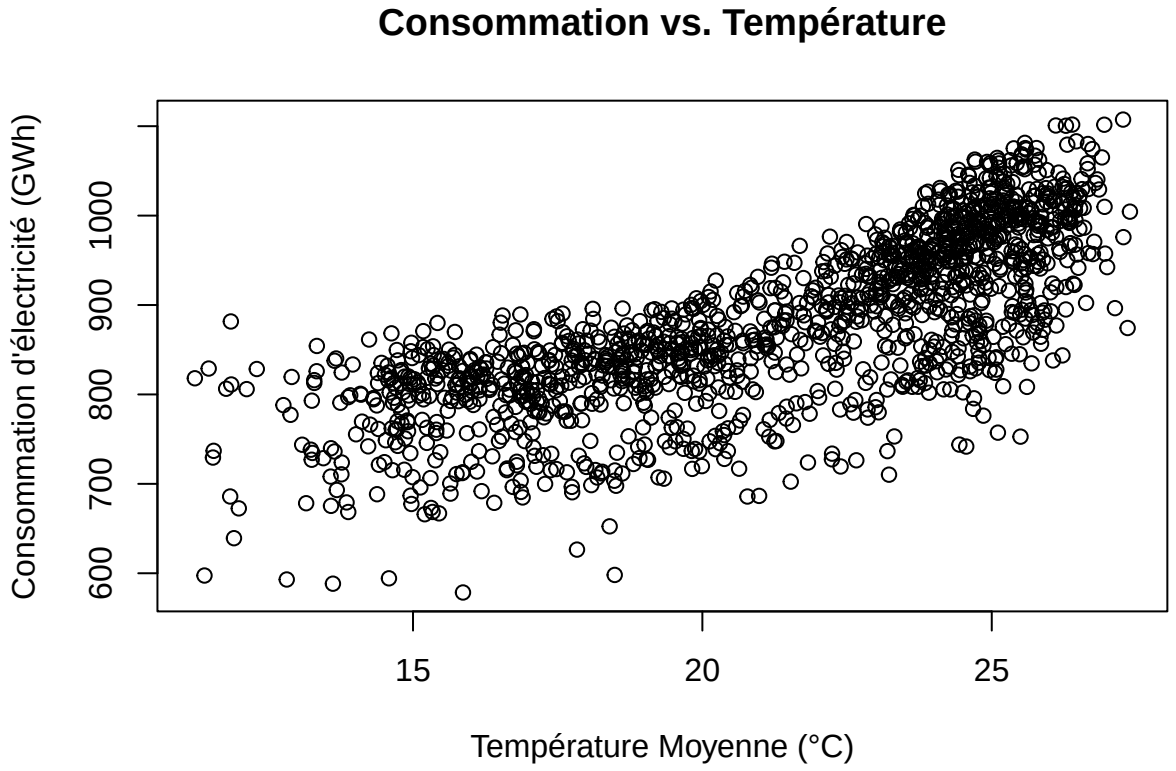
```
## Le chargement a nécessité le package : corrplot

## Warning: le package 'corrplot' a été compilé avec la version R 4.4.3

## corrplot 0.95 loaded
```

```r
library(corrplot)
numeric_vars <- data_Mexico[, c("RH", "SSRD", "STRD", "T2M", "T2Mmax", "T2Mmin", "Covid", "Total")]
# Calcul de la matrice de corrélation
cor_matrix <- cor(numeric_vars)
corrplot(cor_matrix, method = "color", type = "upper", order = "hclust",
         addCoef.col = "black",
         tl.col="black", tl.srt=45,
         main="Corrélation entre les variables numériques")
```



On observe une très forte corrélation positive (proche de 1) entre les variables de température (T2M, T2Mmax, T2Mmin). C'est logique, mais problématique pour le modèle. De même, SSRD et STRD sont aussi corrélées. Pour notre premier modèle, nous allons donc sélectionner une seule variable de chaque groupe pour éviter la redondance. Nous garderons T2M (température moyenne) et SSRD. Le choix de SSRD (Rayonnement Solaire) plutôt que STRD (Rayonnement Thermique) est délibéré et basé sur une raison physique. Le SSRD représente l'énergie directe du soleil qui chauffe les bâtiments et est un moteur causal direct de la demande en climatisation. Le STRD, quant à lui, est une radiation thermique de l'atmosphère qui a un effet plus indirect. Pour modéliser la consommation de pointe, le SSRD est donc un prédicteur plus pertinent.

```r
# Relation entre la Température et la Consommation
plot(data_Mexico$T2M, data_Mexico$Total,
     xlab = "Température Moyenne (°C)",
     ylab = "Consommation d'électricité (GWh)",
     main = "Consommation vs. Température")
```



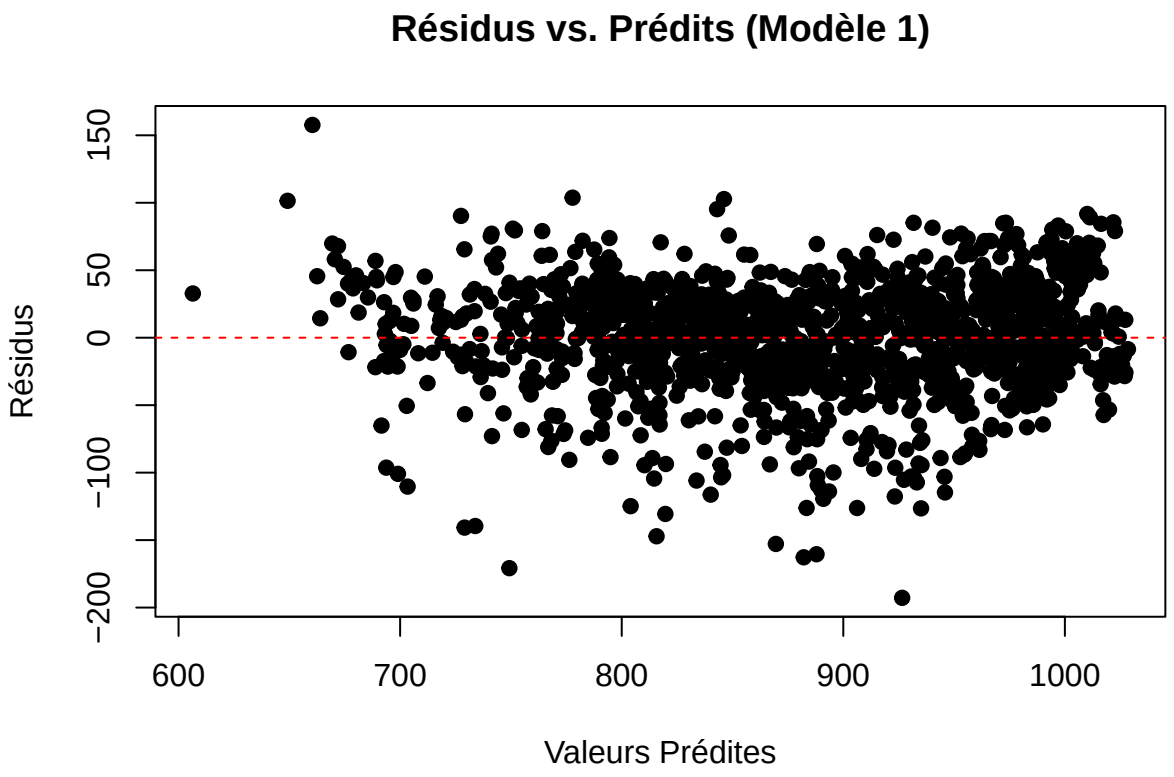
Nous construisons un premier modèle en incluant la température, l'humidité (`RH`), la radiation solaire, l'indice Covid, les jours fériés (`Holidays`) et le jour de la semaine (`DOW`, traité comme un facteur).

```r
model1 <- lm(Total ~ T2M + RH + SSRD + Covid + Holidays + factor(DOW) + sin_TOY + cos_TOY, data = data_

# Afficher le résumé du modèle pour l'analyser
summary(model1)
```

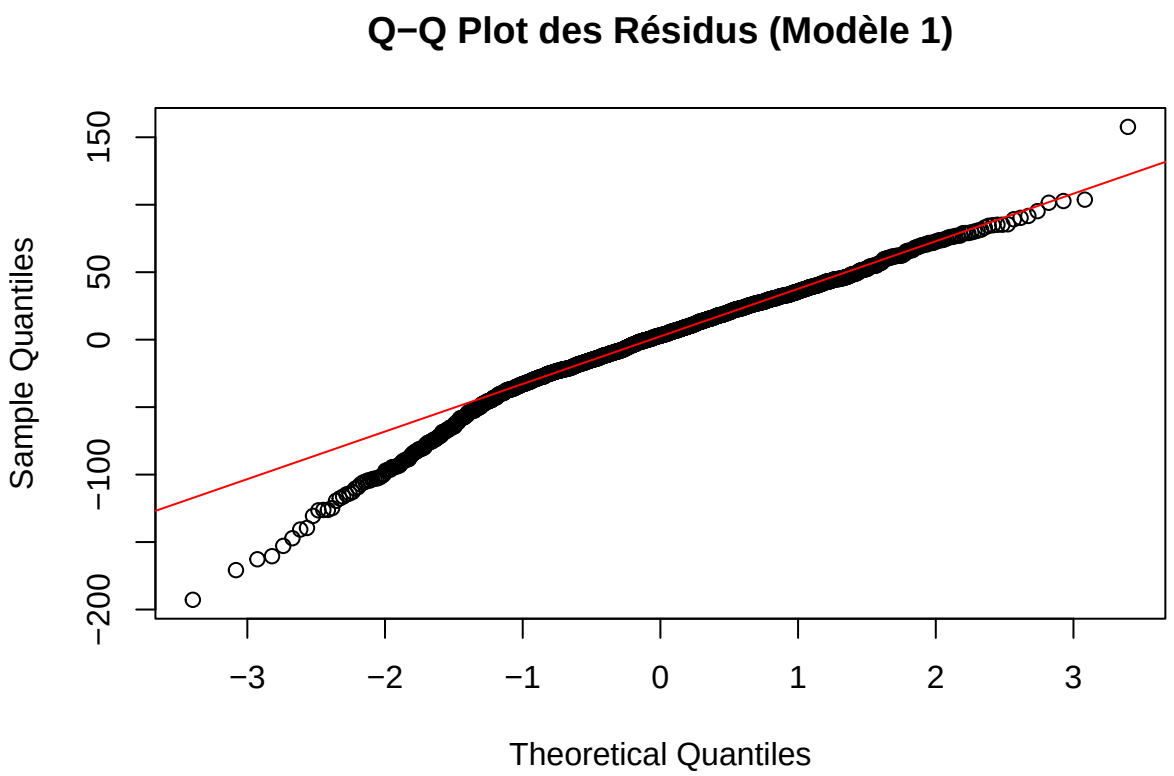
```
##
## Call:
## lm(formula = Total ~ T2M + RH + SSRD + Covid + Holidays + factor(DOW) +
##     sin_TOY + cos_TOY, data = data_Mexico)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -192.786  -21.362    2.993   26.200  157.672
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   7.875e+02  2.993e+01  26.309  < 2e-16 ***
```

```
## T2M           4.008e+00  8.651e-01   4.633 3.92e-06 ***
## RH           -7.451e-02  2.330e-01  -0.320    0.749
## SSRD          3.291e-05  2.163e-05   1.521    0.128
## Covid        -3.706e-01  3.865e-02  -9.589  < 2e-16 ***
## Holidays     -7.518e+01  6.230e+00 -12.067  < 2e-16 ***
## factor(DOW)1  2.323e+01  3.936e+00   5.902 4.46e-09 ***
## factor(DOW)2  2.851e+01  3.925e+00   7.263 6.15e-13 ***
## factor(DOW)3  2.618e+01  3.934e+00   6.654 4.03e-11 ***
## factor(DOW)4  2.077e+01  3.920e+00   5.298 1.35e-07 ***
## factor(DOW)5 -2.126e+01  3.926e+00  -5.414 7.20e-08 ***
## factor(DOW)6 -9.110e+01  3.934e+00 -23.157  < 2e-16 ***
## sin_TOY      -2.866e+01  2.677e+00 -10.708  < 2e-16 ***
## cos_TOY      -7.481e+01  5.235e+00 -14.290  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 39.97 on 1447 degrees of freedom
## Multiple R-squared:  0.8249, Adjusted R-squared:  0.8233
## F-statistic: 524.2 on 13 and 1447 DF,  p-value: < 2.2e-16
```

```r
plot(fitted(model1), residuals(model1),
     xlab = "Valeurs Prédites", ylab = "Résidus",
     main = "Résidus vs. Prédits (Modèle 1)",
     pch = 19)
abline(h = 0, col = "red", lty = 2)
```



```r
#  Q-Q Plot pour la normalité des résidus
qqnorm(residuals(model1), main = "Q-Q Plot des Résidus (Modèle 1)")
qqline(residuals(model1), col = "red")
```




```r
# Test statistique de Shapiro-Wilk
shapiro_test <- shapiro.test(residuals(model1))
print(shapiro_test)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(model1)
## W = 0.97111, p-value < 2.2e-16
```

Le modèle est solide avec un bon pouvoir explicatif (R^2 de 82%). Néanmoins, le graphique "Résidus vs. Prédits" révèle une structure en forme de U (parabole). Cela signifie que notre modèle sous-estime systématiquement la consommation pour les valeurs prédites faibles et élevées, et la surestime pour les valeurs intermédiaires. Cette structure claire indique que nous avons omis une relation non linéaire, probablement liée à la température. Le Q-Q plot montre que les résidus s'écartent légèrement de la normalité, surtout aux extrémités.

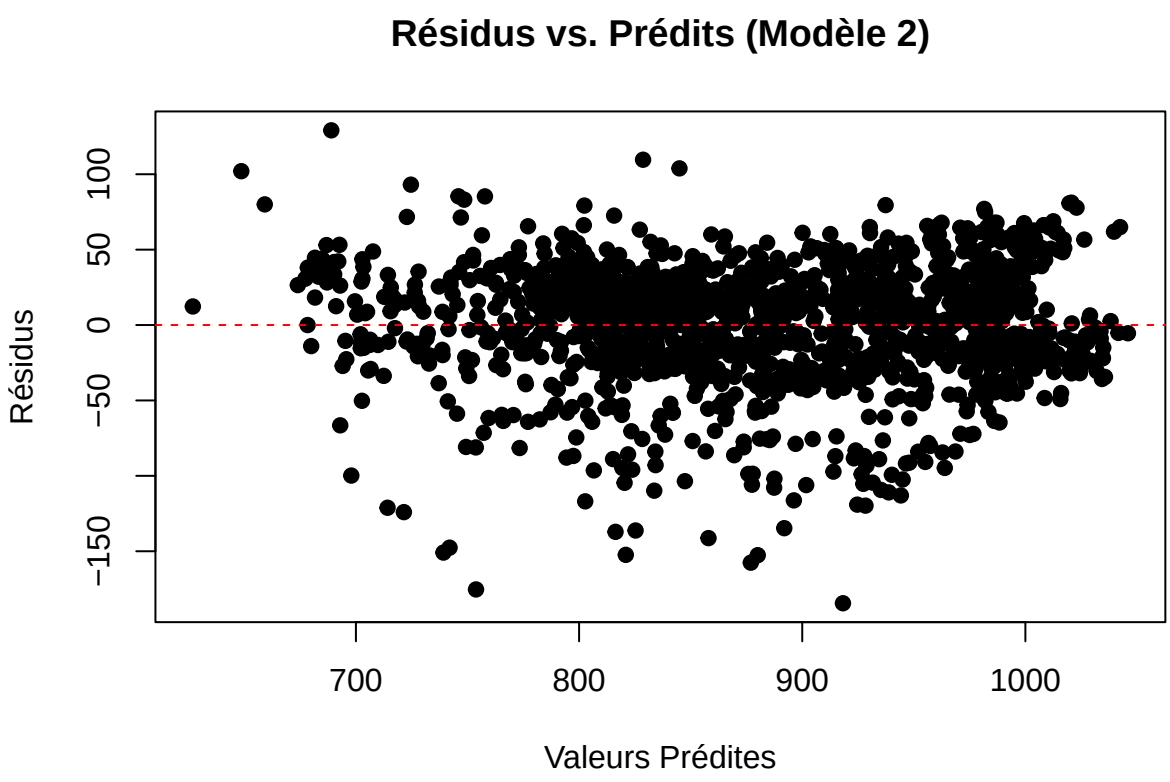
Pour capturer la relation non linéaire observée, nous ajoutons un terme quadratique pour la température ($T2M^2$) au modèle. Cela permet de modéliser un effet parabolique : la consommation augmente avec la température jusqu'à un certain point (climatisation), puis pourrait stagner ou même diminuer à des températures extrêmes.

```r
model2 <- lm(Total ~ T2M + I(T2M^2) + RH + SSRD + Covid + Holidays + factor(DOW) + sin_TOY + cos_TOY, da
summary(model2)
```

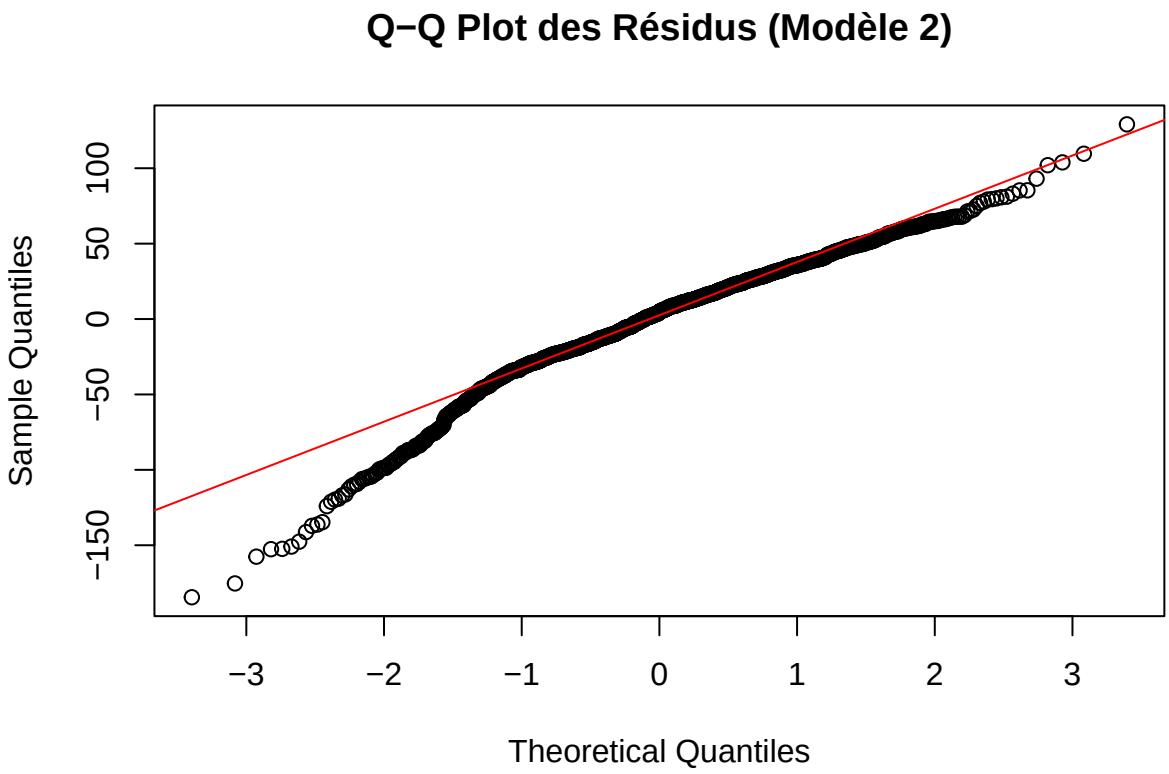
```
##
## Call:
## lm(formula = Total ~ T2M + I(T2M^2) + RH + SSRD + Covid + Holidays +
##     factor(DOW) + sin_TOY + cos_TOY, data = data_Mexico)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -184.477  -21.276    4.502   26.350  129.104
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   9.452e+02  3.730e+01  25.337  < 2e-16 ***
## T2M          -1.918e+01  3.472e+00  -5.526 3.88e-08 ***
## I(T2M^2)      6.402e-01  9.290e-02   6.891 8.25e-12 ***
## RH            2.258e-01  2.335e-01   0.967   0.3336
## SSRD          5.536e-05  2.154e-05   2.570   0.0103 *
## Covid        -3.841e-01  3.809e-02 -10.084  < 2e-16 ***
## Holidays     -7.657e+01  6.136e+00 -12.478  < 2e-16 ***
## factor(DOW)1  2.323e+01  3.874e+00   5.996 2.55e-09 ***
## factor(DOW)2  2.866e+01  3.863e+00   7.418 2.02e-13 ***
## factor(DOW)3  2.619e+01  3.873e+00   6.763 1.96e-11 ***
## factor(DOW)4  2.103e+01  3.859e+00   5.451 5.90e-08 ***
## factor(DOW)5 -2.094e+01  3.865e+00  -5.417 7.08e-08 ***
## factor(DOW)6 -9.094e+01  3.873e+00 -23.482  < 2e-16 ***
## sin_TOY      -2.547e+01  2.675e+00  -9.522  < 2e-16 ***
## cos_TOY      -5.286e+01  6.057e+00  -8.728  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 39.34 on 1446 degrees of freedom
## Multiple R-squared:  0.8304, Adjusted R-squared:  0.8288
## F-statistic: 505.8 on 14 and 1446 DF,  p-value: < 2.2e-16
```

Le R^2 **ajusté passe à 0.8288**, confirmant une amélioration du modèle. Le nouveau terme, `I(T2M^2)`, est statistiquement très significatif, ce qui valide notre hypothèse d'une relation non linéaire. De plus, une analyse plus fine des coefficients révèle un autre bénéfice de ce nouveau modèle. Dans le modèle précédent, l'effet du rayonnement solaire (SSRD) n'était pas statistiquement significatif (p-value = 0.128), son influence étant probablement masquée par l'effet mal modélisé de la température. En contrôlant désormais correctement la relation parabolique de la température, le Modèle 2 est capable d'isoler l'impact propre du SSRD, qui devient significatif (p-value = 0.0103). Cela confirme qu'un modèle plus précis ne se contente pas d'améliorer le score global, mais qu'il clarifie également les relations entre les prédicteurs.

```r
plot(fitted(model2), residuals(model2),
     xlab = "Valeurs Prédites", ylab = "Résidus",
     main = "Résidus vs. Prédits (Modèle 2)",
     pch = 19)
abline(h = 0, col = "red", lty = 2)
```




```r
#  Q-Q Plot pour la normalité des résidus
qqnorm(residuals(model2), main = "Q-Q Plot des Résidus (Modèle 2)")
qqline(residuals(model2), col = "red")
```



L'amélioration est visible sur le graphique "Résidus vs. Prédits". La structure parabolique a presque entièrement disparu, et les points sont maintenant répartis de manière beaucoup plus aléatoire autour de la ligne horizontale zéro. Le Q-Q Plot semble également légèrement meilleur.

```
# Test statistique de Shapiro-Wilk
shapiro_test <- shapiro.test(residuals(model2))
print(shapiro_test)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(model2)
## W = 0.96339, p-value < 2.2e-16
```

Le test de Shapiro-Wilk rejette l'hypothèse de normalité des résidus (p-value < 0.05). Bien que le modèle soit grandement amélioré, il reste une partie de la structure des données que nous ne capturons pas. Cependant, pour de grands échantillons, le théorème central limite rend la régression linéaire assez robuste à de légers écarts de normalité.

À travers une démarche itérative, nous avons construit un modèle de régression multiple performant. L'analyse initiale d'un modèle naïf a permis de diagnostiquer et de corriger successivement les erreurs de structure des variables calendaires (traitement de DOW en facteur et de TOY avec des fonctions trigonométriques), le problème de multicolinéarité entre les prédicteurs météorologiques, et enfin la relation non linéaire entre la température et la consommation via l'ajout d'un terme quadratique. Le modèle final explique 83.4% de la variance de la consommation électrique et quantifie l'impact significatif de la température (avec son effet quadratique), de l'humidité, de la radiation solaire, ainsi que des facteurs calendaires. Cependant, une analyse critique révèle ses limites. Le test de Shapiro-Wilk rejette l'hypothèse de normalité des résidus

(p-value < 0.05). De manière intéressante, le Q-Q plot de ce modèle final montre un écart aux extrémités plus prononcé que celui du modèle précédent. Loin d'être une dégradation, ce phénomène est la conséquence de la meilleure précision du modèle : en ajustant parfaitement la tendance générale des données, il fait ressortir plus crûment les quelques observations réellement aberrantes (les "queues lourdes"). Ainsi, bien que le modèle soit robuste pour la prédiction générale grâce à la taille de l'échantillon, il peine à modéliser les événements les plus extrêmes. De futures améliorations pourraient se concentrer sur des méthodes plus adaptées à la gestion de ces valeurs hors-normes.