Université Claude Bernard Lyon I

Master Data Science

Modèles de régression

Régression des données d'extraction de puits de pétrole au Canada

Auteurs :
Bruno Dumas
Jules Sauvinet

Professeur :
François Wahl

14 janvier 2017

Résumé

L'objectif est d'effectuer des régressions des valeurs d'extractions des puits de pétrole du Canada afin de pouvoir prédire les futures données d'extraction. Les courbes des valeurs des données d'extraction donnent une classification de la qualité des puits. L'étiquetage de la qualité de chaque puits à été effectué par des experts. L'objectif de ce travail est de mettre en place une classification automatique de ces puits : La démarche proposée est décrite ci-dessous. L'idée est de remplacer ces courbes par des fonctions paramétriques. Pour cela, plusieurs régressions vont être envisagées afin d'avoir la meilleure prédiction/classification possible.

Table des matières

1 Régressions polynomiales

Une façon simple est d'ajuster un polynôme de degré faible sur chacune des courbes et de voir si les coefficients présentent des clusters, c'est à dire des groupes de points distincts quand on les regarde dans l'espace. On essaiera des polynômes de degré 0, 1, 2, 3, et 4. On présentera les courbes de production simulées obtenues, comme dans la figure ci-dessous.

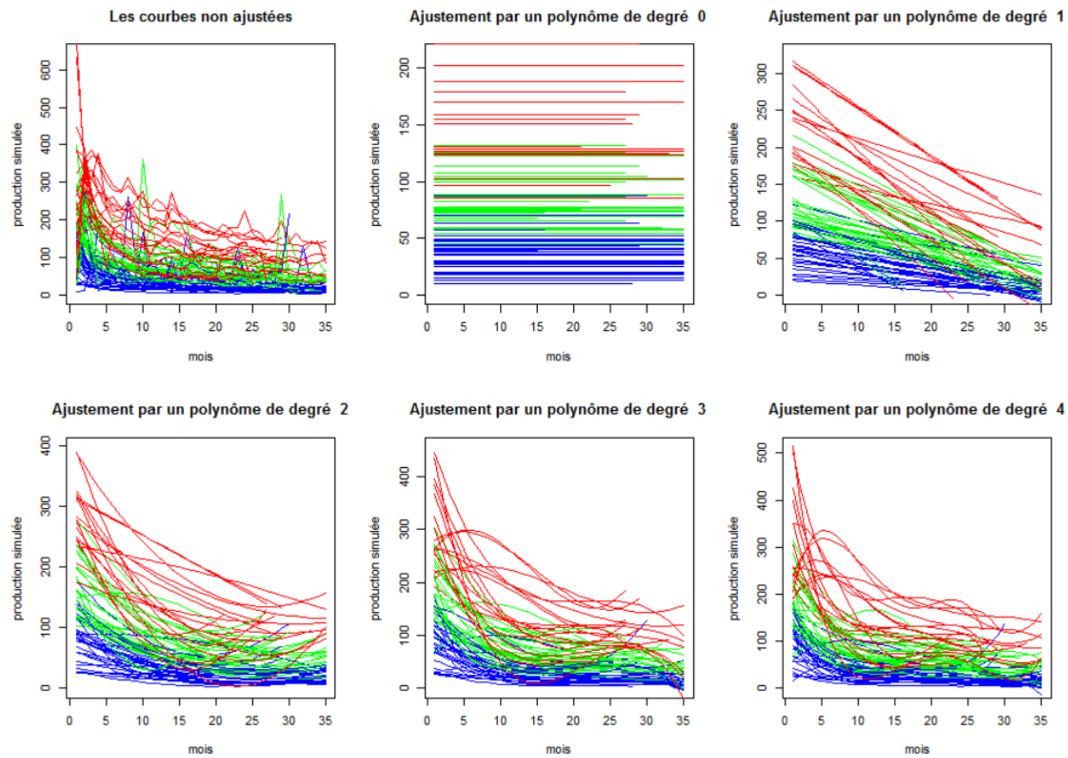

FIGURE 1 – Courbes non régressées, et courbes de régressions polynomiales de degrés 0,1,2,3 et 4

La moyenne des R-Squared ajusted pour les 75 courbes pour les 5 types de régression sont respectivement :
0.0000000, 0.4853609, 0.6622269, 0.7202006 et 0.7557633.

On voit que plus le degré du polynome augmente, plus la régression est de qualité au critère du R-Squared (les valeurs prédites se rapprochent des valeurs des observations).

Si on se réfère à la figure ci-dessus, on observe que la plupart des simulations présente une remontée au bout de quelques mois, que certaines simulations sont concaves au lieu d'être convexes, voire que certaines d'entre elles pourraient avoir des valeurs négatives (autour des 35 mois d'exploitation).

2 Régressions exponentielles

Une idée simple pour corriger ces défauts est d'utiliser une autre forme paramétrique pour les simulations. Une suggestion immédiate pour qui a un peu l'habitude de ces courbes est une forme exponentielle du style : oú y est la production, t le mois, et $k0$ et $k1$ deux paramètres à déterminer.

Régression exponentielle $log(y) = ax + b$

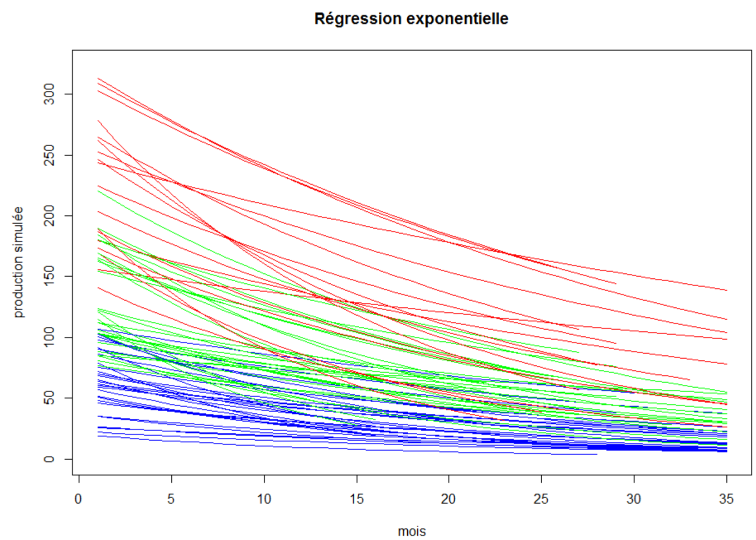

FIGURE 2 – Courbes ajustées avec une fonction exponentielle et l'outil lm de R

Régression exponentielle $y = k_0 e^{-k_1 x}$

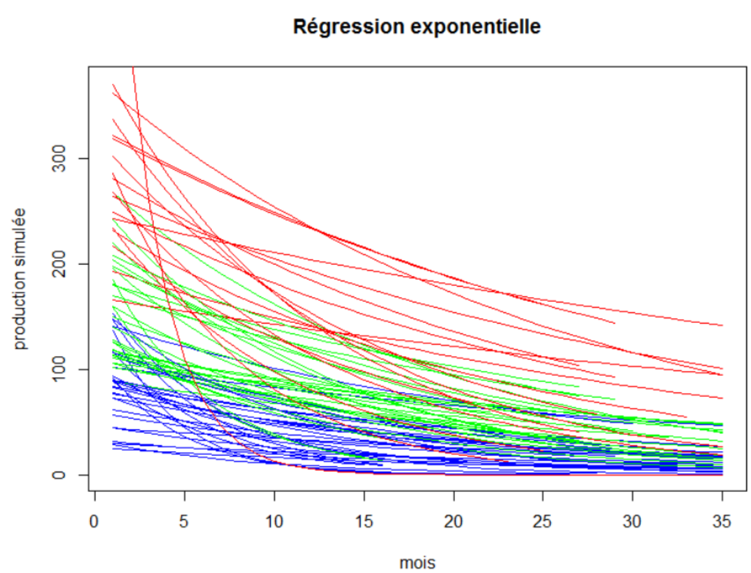

FIGURE 3 – Courbes ajustées avec une fonction exponentielle et l'outil nls de R

La régression $log(y) = ax + b$ faite avec lm est mieux, on s'intéressera donc à celle-ci.

La moyenne du "ajusted R-Squared" pour les 75 régressions exponentielles est cette fois-ci de 0.6754964, donc de moins bonne qualié à priori que les régressions polynomiales de degré supérieures ou égale à 3.
Néanmoins, quand on s'intéresse au détail des R-Squared, on trouve de très bonne régressions avec un R-Squared très bon comme des régressions avec un R-Squared très mauvais. Cela est probablement dû à certaines valeurs "outliers" qui ne permettent pas à la fonction exponentielle de s'adapter aux courbes.
La partie 5 et le lissage des courbes permettra de pallier à ce problème et de probablement montrer que la régression exponentielle est adaptée si un travail d'atténuation des "pics" est fait au préalable.

10 premiers R-Squared :
0.8011383, 0.4182548, 0.7403922, 0.7585257, 0.4186529, 0.6888943, 0.3297291, 0.818889, 0.7745934 et 0.7527809.

Autres possibilités
Nous avons pensé a un ajustement par fonction inverse ou avec une gaussienne.

3 Courbes hautes et basses à 95%

Quelles sont les incertitudes sur les régressions des points 2 ? Plus concrètement, on vous demande de tracer pour un exemple de chaque type de courbe, la courbe haute (à 95%) et la courbe basse (toujours à 95%)

Intervalles de confiance avec nls

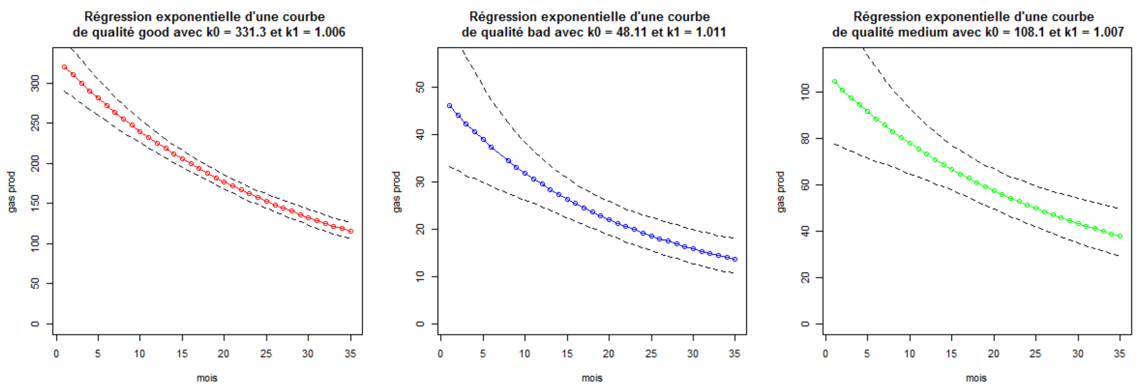

FIGURE 4 – Courbes hautes et basses à 95% pour un exemple de chaque classe de courbes

Intervalles de confiance avec lm

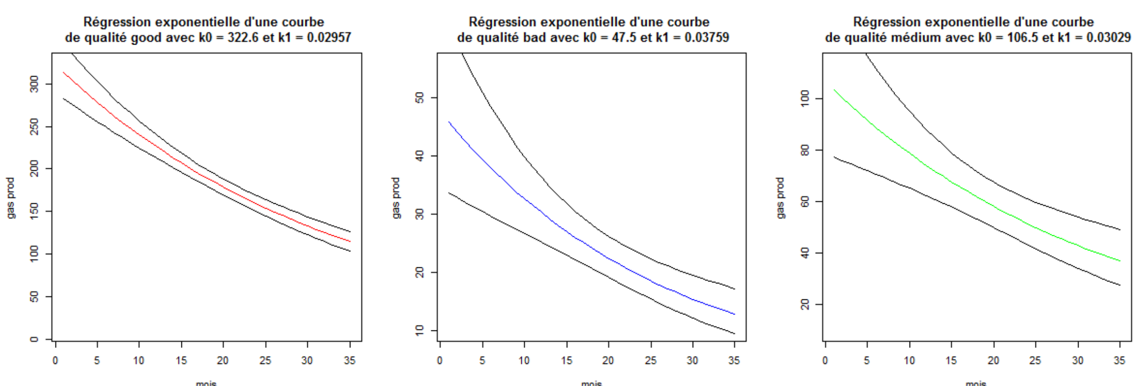

FIGURE 5 – Courbes hautes et basses à 95% pour un exemple de chaque classe de courbes

Comme le montre les figures ci-dessus, il y a davantage d'incertitude sur les valeurs des courbes de qualité 'bad' et 'médium'. Cela induit que les valeurs de ces courbes suivent des tendances plus compliquées á ajuster. En effet, les valeurs d'extraction pour ces puits sont probablement plus imprévisible et suivent parfois quelques oscillations. En outre, les incertitudes aux valeurs d'extraction pour les premiers et derniers mois sont également plus fortes.

4 Reclassement avec régression logistique

En examinant le graphe k1 fonction de k0, on se rend compte que certaines courbes classées 'Good' par les experts donnent l'impression d'être plutôt 'medium', tandis que certaines 'bad' pourraient être aussi 'medium'.
Avez-vous des suggestions sur 5 courbes au plus qui pourraient être mal classées ?
Justifiez vos choix (i.e. une façon de faire est d'effectuer une régression logistique dont le y est la classe prédite par l'expert et les x sont les coefficients $k0$ et $k1$, et d'examiner comment la régression est améliorée en changeant la classe d'un point).

Clustering des courbes en fonction de $k0$ et $k1$ avant reclassement

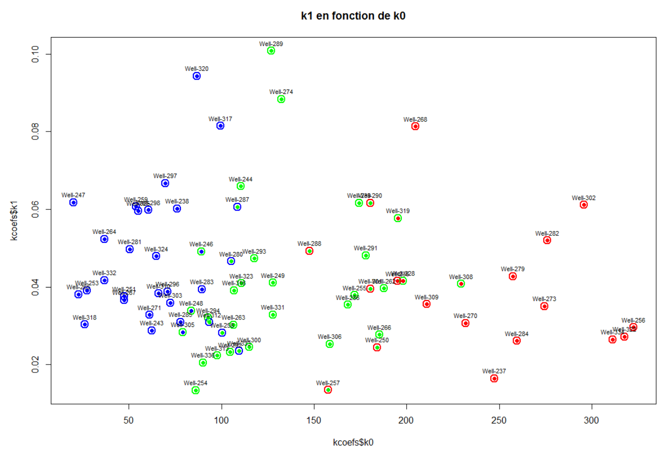

FIGURE 6 – Coefficients $k1$ en fonction de $k0$ et classes des courbes

La couleur du cercle extérieur est la classe de départ (observée) et le cercle intérieur est sa classe prédite.
16 courbes sont mal classées d'après les prédictions et l'AIC de la régression est de 74.63465.

On change la classe de 5 puits.
On se limite au reclassement 5 puits car on ne veut pas trop modifier le modèle de départ et laisser de la souplesse au classifieur si celui-ci doit traiter la classification de nouveaux puits á l'avenir.

Aprés interprétations graphiques et des tests de prédiction de classe avec ou sans changement de classe des 16 puits mal classés, on détermine les 5 puits qui réajuste mieux de modéle et améliore le classifieur :
Les courbes des puits '$Well-288$' de good à médium, '$Well-333$' de bad à médium, '$Well-246$' de médium à bad, '$Well-257$' de good à médium, et '$Well-258$' de bad à médium.

Clustering des courbes en fonction de $k0$ et $k1$ après reclassement

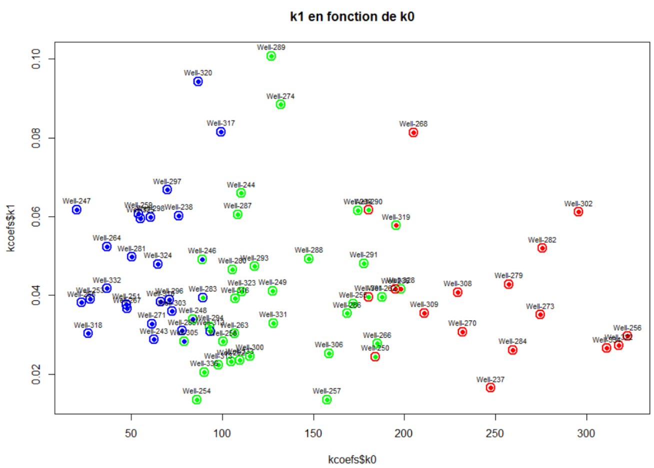

FIGURE 7 – Coefficients $k1$ en fonction de $k0$ et classes des courbes

L'AIC de la régression est de 52.52519 et il n'y a plus que 10 courbes mal classées pour 5 de changées sur les 16 de départ. Soit 1 courbe qui dont la prédiction de classe par le classifieur s'est accordée sur sa vraie classe.

5 Gestion des spikes et lissage des courbes

Régression polynomiale de degré 3 avec smooth

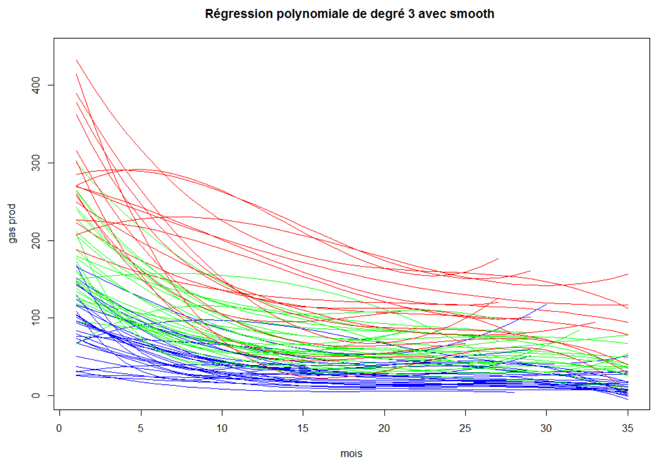

FIGURE 8 – Régression polynomiale de degré 3 avec courbes lissées au préalable avec loess

On obtient une moyenne de R-Squared de 0.9802724 cette fois-ci après lissage des courbes. La régression polynomiale devient ainsi très performante. Il reste à savoir si l'on peut se permettre l'approximation faite par le lissage des courbes et si les valeurs induites par les spikes avaient une pertinence intransigible.
Toutefois, la régression avec smooth ne règle pas les valeurs qui remontent, les simulations concaves au lieu d'être convexes, et les potentielles valeurs négatives autour des 35 mois.

Régression exponentielle avec lissage des spikes

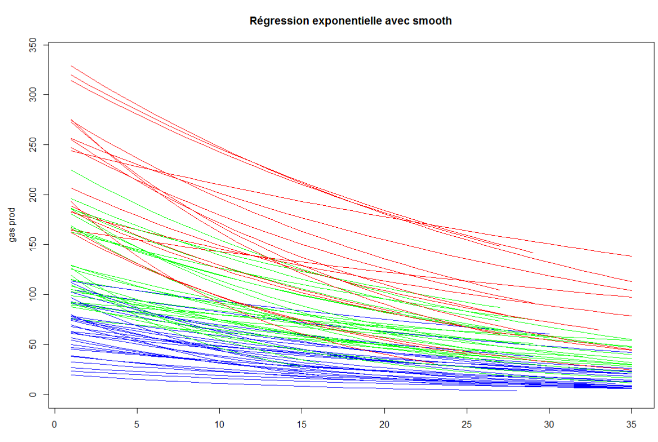

FIGURE 9 – Régression exponentielle lm avec courbes lissée au préalable avec loess

On obtient une moyenne de R-Squared de 0.7569813 cette fois-ci après lissage des courbes. La régression est sensiblement améliorée avec le lissage des courbes mais reste moins efficace que la régression polynomiale de degré 3.

Reclassemement après lissage des courbes et ajustement exponentiel

On effectue á nouveau une régression logistique á partir des courbes lissées comme á la question 4. On obtient cette toujours 16 courbes mal classées, un AIC de 85 et le graphique suivant :

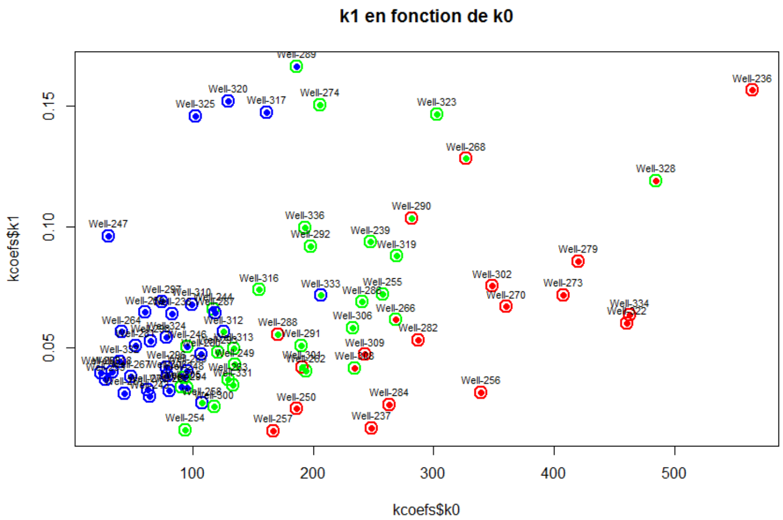

FIGURE 10 – Clustering sur modèle exponentiel avec courbes lissées

On se limite comme dans la partie 3 á seulement 5 reclassement de courbes. Aprés reclassement des puits '$Well-290$', '$Well-333$', '$Well-312$', '$Well-288$', '$Well-258$', on obtient un AIC de 56 et plus que 8 courbes mal classées en faisant une nouvelle régression logistique, soit 3 courbes dont la classe prédite a cette fois-ci été égale á celle observée.

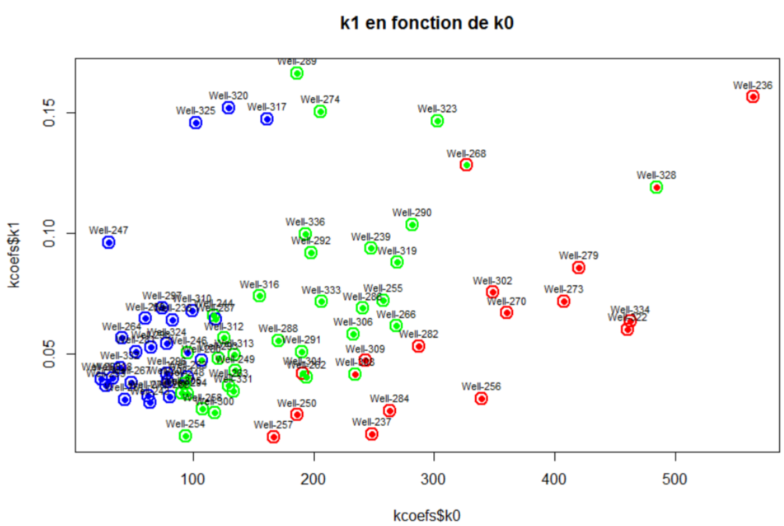

FIGURE 11 – Clustering sur modèle exponentiel avec courbes lissée aprés reclassement

Reclassemement après lissage des courbes et ajustement polynomial

On effectue une régression logistique á partir des courbes lissées et ajuster par un polynôme de degré 3 cette fois-ci. On obtient cette toujours 12 courbes mal classées, un AIC de 78.25367 et le graphique de dépendance des 4 coefficients entre eux suivant :

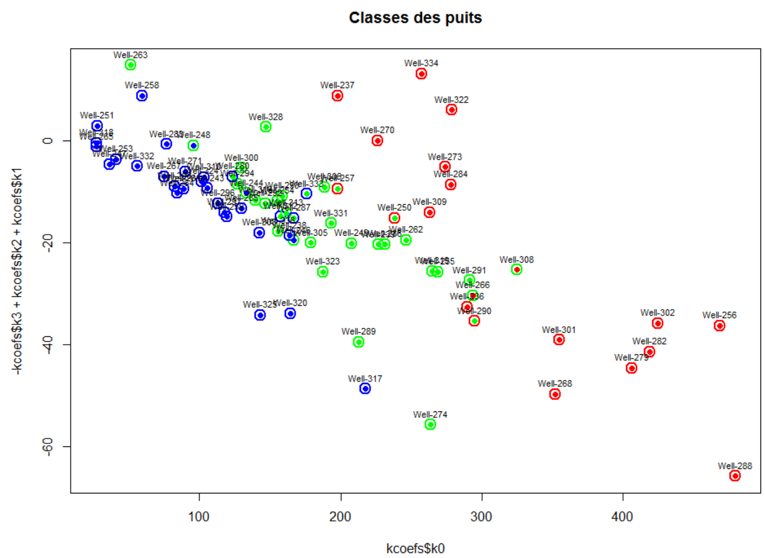

Figure 12 – Clustering sur modèle polynomial de degré 3 avec courbes lissées

Aprés reclassement des puits ’$Well-257$’, ’$Well-250$’, ’$Well-333$’, ’$Well-266$’ on obtient un AIC de 64.16384 et plus que 6 courbes mal classées en faisant une nouvelle régression logistique, soit 2 courbes dont la classe prédite a cette fois-ci été égale á celle observée.

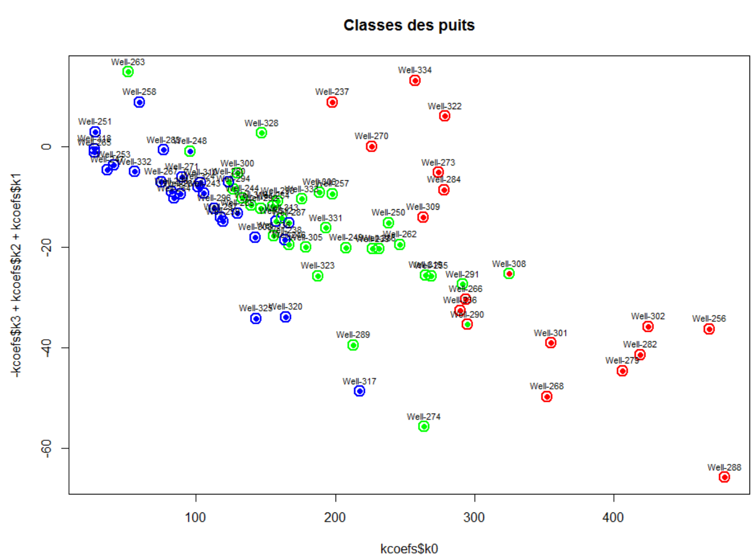

Figure 13 – Clustering sur modèle polynomial de degré 3 avec courbes lissées et reclassement

Acknowledgments

Nos remerciements vont à François Wahl pour l'enseignement de son cours "Modèles de régression" á l'Université Claude Bernard Lyon I et son accompagnement durant ce projet.

A Code de la question 1

```r
#récupération des données du fichier EXCEL
perl <- 'C:\\Strawberry\\perl\\bin\\perl.exe'
datapuits = read.xls(file.path("data/FW_Donnees_Puits.xlsx"), perl=perl)

#on transpose les données pour avoir les puits en colonne et les mois en ligne
puits = setNames(data.frame(t(datapuits[,-1])), datapuits[,1])
puits

##################################################
##   QUESTION   1   -   Régressions polynomiales##
##################################################

par(mfrow=c(2,3))
seq1 <- -1:4 #on va tracer 6 graphiques
seq2 <- 1:75 #les 75 puits
r2 <- numeric(5) #la moyenne du rsquared
for (j in seq1){
  r2j = 0.0 # le rsquared d'une regression
  for (i in seq2){
    mois <- 1:35 #l'abscisse
    v <- as.vector(puits[,i]) #on récupère les données du puits i

    col="black" #la couleur en fonction de la classification de qualité
    if (v[36] == "Good"){col = "red"}else if (v[36] == "medium"){col = "green"}else {col = "blue"}

    v <- as.numeric(v[1:35])

    #on plot pas les 0 qui sont des ND
    nd <- which(v %in% 0)
    v <- v[v != 0]
    mois <- mois[!mois %in%  nd]

    #tracé de la figure 1 : les données de production
    if (j==-1){
      if (i==1){
        plot(mois,v,type="l",col=col, ylab="production simulée", main="Les courbes non ajustées",ylim=c(0,max(v)+10))
      }
      else
        lines(mois,v,type="l",col=col)
    }
    #tracé de la figure 2 : les courbes de production obtenues avec des polynômes de degré 0, 1, 2, 3, et 4
    else{
      if (j==0){
        fit2 <- lm(v ~ 1)
      }else {
        fit2 <- lm(v ~ poly(mois, j, raw=TRUE))
      }
      if (i==1){
        plot(mois, predict(fit2), type="l",col=col, ylab="production simulée", lwd=1, main=paste("Ajustement par un polynôme de c
      }
      lines(mois, predict(fit2), col=col, lwd=1)
    }
    if (j >= 0) {r2j=r2j+summary(fit2)$adj.r.squared}
  }
  #on calcule la moyenne des R-Squared pour chaque type de régression
  if (j >= 0){
    r2[j+1]=r2j/i
  }
}
```

FIGURE 14 – Code de la question 1 des régressions polynomiales

B Code de la question 2

```r
########################################################
##   QUESTION   2    -    Régression exponentielle ##
########################################################

#Avec lm
rse2=0
r2e1 <- numeric(1)
for (i in seq2){
  mois <- 1:35
  v <- as.vector(puits[,i])

  col="black"
  if (v[36] == "Good"){
    col = "red"
  }else if (v[36] == "medium"){
    col = "green"
  }else
    col = "blue"

  v <- as.numeric(v[1:35])

  nd <- which(v %in% 0)
  v <- v[v != 0]
  mois <- mois[!mois %in%  nd]

  expfit <- lm(log(v) ~ mois)

  if (i==3){
    #residus=v-exp(expfit$fitted.values)
    #plot(v,residus, main=paste("Graphique des résidus de la régression exponentielle"), xlab="production simulée", ylab="rési

    plot(mois,exp(predict(expfit)),type="l",col=col, ylab="production simulée",
         main=paste("Régression exponentielle"),ylim=c(0,max(exp(predict(expfit)))+10))

  }
  else {
    lines(mois,exp(predict(expfit)),type="l",col=col)
  }
  rse2 = rse2 + exp(sigma(expfit))
}
rse2 = rse2 / 75
rse2
summary(expfit)
```

FIGURE 15 – Code de la question 2 des régressions exponentielles, partie 1

```r
#Avec nls
rse = 0
seq2 <- 1:75
r2e2 <- numeric(1)
for (i in seq2){
  mois <- 1:35
  v <- as.vector(puits[,i])

  col="black"
  if (v[36] == "Good"){
    col = "red"
  }else if (v[36] == "medium"){
    col = "green"
  }else {
    col = "blue"
  }

  v <- as.numeric(v[1:35])

  #on plot pas les 0 qui sont des ND (d'apres moi)
  nd <- which(v %in% 0)
  v <- v[v != 0]
  mois <- mois[!mois %in%  nd]
  df <- data.frame(mois, v)
  df$lv <- df$v

  k0start=-300
  k1start=0.1

  m <- nls(lv ~ k0*exp(-k1*mois), start=c(k0=k0start, k1=k1start), df)
  summary(m)

  if (i==1){
    plot(df$mois,predict(m),type="l",col=col, ylab="production simulée", xlab="mois", main=paste("Régression exponentielle"),y
  }
  lines(df$mois,predict(m),type="l",col=col)

  rse = rse + sigma(m)
}
rse = rse / 75
rse
summary(m)
```

Figure 16 – Code de la question 2 des régressions exponentielles, partie 2

C Code de la question 3

```
#récupération des données du fichier EXCEL
perl <- 'C:\\Strawberry\\perl\\bin\\perl.exe'
datapuits = read.xls(file.path("data/FW_Donnees_Puits.xlsx"), perl=perl)

#on transpose les données pour avoir les puits en colonne et les mois en ligne
puits = setNames(data.frame(t(datapuits[,-1])), datapuits[,1])
puits

##################################################
##   QUESTION   1   -   Régressions polynomiales##
##################################################

par(mfrow=c(2,3))
seq1 <- -1:4 #on va tracer 6 graphiques
seq2 <- 1:75 #les 75 puits
r2 <- numeric(5) #la moyenne du rsquared
for (j in seq1){
  r2j = 0.0 # le rsquared d'une regression
  for (i in seq2){
    mois <- 1:35 #l'abscisse
    v <- as.vector(puits[,i]) #on récupère les données du puits i

    col="black" #la couleur en fonction de la classification de qualité
    if (v[36] == "Good"){col = "red"}else if (v[36] == "medium"){col = "green"}else {col = "blue"}

    v <- as.numeric(v[1:35])

    #on plot pas les 0 qui sont des ND
    nd <- which(v %in% 0)
    v <- v[v != 0]
    mois <- mois[!mois %in%  nd]

    #tracé de la figure 1 : les données de production
    if (j==-1){
      if (i==1){
        plot(mois,v,type="l",col=col, ylab="production simulée", main="Les courbes non ajustées",ylim=c(0,max(v)+10))
      }
      else
        lines(mois,v,type="l",col=col)
    }
    #tracé de la figure 2 : les courbes de production obtenues avec des polynômes de degré 0, 1, 2, 3, et 4
    else{
      if (j==0){
        fit2 <- lm(v ~ 1)
      }else {
        fit2 <- lm(v ~ poly(mois, j, raw=TRUE))
      }
      if (i==1){
        plot(mois, predict(fit2), type="l",col=col, ylab="production simulée", lwd=1, main=paste("Ajustement par un polynôme de
      }
      lines(mois, predict(fit2), col=col, lwd=1)
    }
    if (j >= 0) {r2j=r2j+summary(fit2)$adj.r.squared}
  }
  #on calcule la moyenne des R-Squared pour chaque type de régression
  if (j >= 0){
    r2[j+1]=r2j/i
  }
}
```

FIGURE 17 – Code de la question 3 des intervalles de confiance

D Code de la question 4

```
#récupération des données du fichier EXCEL
perl <- 'C:\\Strawberry\\perl\\bin\\perl.exe'
datapuits = read.xls(file.path("data/FW_Donnees_Puits.xlsx"), perl=perl)

#on transpose les données pour avoir les puits en colonne et les mois en ligne
puits = setNames(data.frame(t(datapuits[,-1])), datapuits[,1])
puits

##################################################
##   QUESTION   1   -   Régressions polynomiales##
##################################################

par(mfrow=c(2,3))
seq1 <- -1:4 #on va tracer 6 graphiques
seq2 <- 1:75 #les 75 puits
r2 <- numeric(5) #la moyenne du rsquared
for (j in seq1){
  r2j = 0.0 # le rsquared d'une regression
  for (i in seq2){
    mois <- 1:35 #l'abscisse
    v <- as.vector(puits[,i]) #on récupère les données du puits i

    col="black" #la couleur en fonction de la classification de qualité
    if (v[36] == "Good"){col = "red"}else if (v[36] == "medium"){col = "green"}else {col = "blue"}

    v <- as.numeric(v[1:35])

    #on plot pas les 0 qui sont des ND
    nd <- which(v %in% 0)
    v <- v[v != 0]
    mois <- mois[!mois %in%  nd]

    #tracé de la figure 1 : les données de production
    if (j==-1){
      if (i==1){
        plot(mois,v,type="l",col=col, ylab="production simulée", main="Les courbes non ajustées",ylim=c(0,max(v)+10))
      }
      else
        lines(mois,v,type="l",col=col)
    }
    #tracé de la figure 2 : les courbes de production obtenues avec des polynômes de degré 0, 1, 2, 3, et 4
    else{
      if (j==0){
        fit2 <- lm(v ~ 1)
      }else {
        fit2 <- lm(v ~ poly(mois, j, raw=TRUE))
      }
      if (i==1){
        plot(mois, predict(fit2), type="l",col=col, ylab="production simulée", lwd=1, main=paste("Ajustement par un polynôme de
      }
      lines(mois, predict(fit2), col=col, lwd=1)
    }
    if (j >= 0) {r2j=r2j+summary(fit2)$adj.r.squared}
  }
  #on calcule la moyenne des R-Squared pour chaque type de régression
  if (j >= 0){
    r2[j+1]=r2j/i
  }
}
```

Figure 18 – Code de la question 4 des reclassements de courbes après régression logistique

E Code de la question 5

```r
#récupération des données du fichier EXCEL
perl <- 'C:\\Strawberry\\perl\\bin\\perl.exe'
datapuits = read.xls(file.path("data/FW_Donnees_Puits.xlsx"), perl=perl)

#on transpose les données pour avoir les puits en colonne et les mois en ligne
puits = setNames(data.frame(t(datapuits[,-1])), datapuits[,1])
puits

##################################################
##   QUESTION   1   -   Régressions polynomiales##
##################################################

par(mfrow=c(2,3))
seq1 <- -1:4 #on va tracer 6 graphiques
seq2 <- 1:75 #les 75 puits
r2 <- numeric(5) #la moyenne du rsquared
for (j in seq1){
  r2j = 0.0 # le rsquared d'une regression
  for (i in seq2){
    mois <- 1:35 #l'abscisse
    v <- as.vector(puits[,i]) #on récupère les données du puits i

    col="black" #la couleur en fonction de la classification de qualité
    if (v[36] == "Good"){col = "red"}else if (v[36] == "medium"){col = "green"}else {col = "blue"}

    v <- as.numeric(v[1:35])

    #on plot pas les 0 qui sont des ND
    nd <- which(v %in% 0)
    v <- v[v != 0]
    mois <- mois[!mois %in%  nd]

    #tracé de la figure 1 : les données de production
    if (j==-1){
      if (i==1){
        plot(mois,v,type="l",col=col, ylab="production simulée", main="Les courbes non ajustées",ylim=c(0,max(v)+10))
      }
      else
        lines(mois,v,type="l",col=col)
    }
    #tracé de la figure 2 : les courbes de production obtenues avec des polynômes de degré 0, 1, 2, 3, et 4
    else{
      if (j==0){
        fit2 <- lm(v ~ 1)
      }else {
        fit2 <- lm(v ~ poly(mois, j, raw=TRUE))
      }
      if (i==1){
        plot(mois, predict(fit2), type="l",col=col, ylab="production simulée", lwd=1, main=paste("Ajustement par un polynôme de
      }
      lines(mois, predict(fit2), col=col, lwd=1)
    }
    if (j >= 0) {r2j=r2j+summary(fit2)$adj.r.squared}
  }
  #on calcule la moyenne des R-Squared pour chaque type de régression
  if (j >= 0){
    r2[j+1]=r2j/i
  }
}
```

Figure 19 – Code de la question 5 des lissages de spikes puis des reclassements de courbes