Examen_R_&_Maths

Sous la direction de M. Henri LAUDE

Claude REN

24/01/2021

Contents

Introduction

Dans ce dossier, nous allons parcourir et faire une évaluation des travaux de mes camarades en R et Maths, pour cela nous allons introduire les 5 critères d'évaluation choisis en R puis en Maths :

- Reproductibilité : il est facile à partir des documents et des explications fournis dans leur étude de reproduire leur expérience ;
- Didactique : la présentation du package est facile à comprendre ;
- Visuel : la mise en page est propre, agréable à lire ou pas ;
- Difficulté : le package est complexe et difficile à prendre en main ;
- Utilité : le package choisi est utile.

En maths nos critères seront :

- Visuel : la mise en page est propre, agréable à lire ou pas ;
- Difficulté : les formules et notions sont complexes ;
- Didactique : les formules et notions sont bien expliquées ;
- Originalité : le papier choisi est original ;
- Analyse : le niveau de compréhension du papier par les auteurs.

1. Travaux R

1.1 Le premier - dabr

1.1.1 Synthèse de la présentation

Le premier des douzes travaux que nous allons analyser est celui de Corentin BRETONNIERE et Antoine SERREAU sur le package dabr. Dans leur présentation, ils nous expliquent comment installer le package et présente quelques unes des fonctions qui peuvent être utile. Le package dabr est utilisé pour manipuler des bases de données, et pour leur présentation ils passent par mariadb pour convertir leur fichier .csv en une database SQL.

1.1.2 Explication du code

```
query <- paste(unlist(lapply(c(), trimws)), collapse = " ")
delete.MariaDBConnection <- function(conn, ..., quiet = FALSE) {query}
dabr::delete(conn, query, quiet = FALSE)
```

Le chunk choisi concerne la fonction **delete** car je trouve qu'il y a un manque d'explications de la part des auteurs ici qui ne précisent que la finalité du chunk.

De mon point de vue, la première ligne par exemple peut s'avérer difficile à comprendre, la fonction *trimws*() supprime les espaces des chaînes de caractères ce dernier est appliquée au vecteur c() par la fonction *lapply*() qui permet d'appliquer une fonction à chaque élément d'un vecteur. La fonction *unlist*() permet de transformer une liste en vecteur et ensuite *paste*() concatène le vecteur.A la ligne suivante, {query} indique seulement que l'on remplace les "..." par query.

1.1.3 Evaluation

- Visuel : D'un point de vue visuel ce n'était pas une belle présentation, les résultats affichants les tableaux sont trop long, très peu de textes ;
- Reproductibilité : Nous avons les informations sur la database utilisée, malheureusement elle n'est pas mise à disposition sur leur Github respectif. Cependant, l'installation du package et de mariadb sont expliquées de manière assez détaillé.
- Difficulté : Le package semble assez simple, difficile d'aller plus loin en terme de difficulté ;
- Didactique : La finalité de chaque chunk est donnée, sans plus d'explications ;
- Utilité : Le package à l'air intéressant car on peut y relier directement une base de donnée SQL, cependant le nombre de fonctionsà disposition semble assez légère.

1.1.4 Conclusion

En conclusion, le package présenté était relativement simple et dans ce sens il est dommage qu'il n'y ait pas eu plus d'explications sur les quelques codes présent, même si la majorité est plutôt explicite, et en terme de visuel de ne pas avoir accordé plus de temps à la mise en page.

1.2 Le deuxième - Data.table

1.2.1 Synthèse de la présentation

Le deuxième des douzes travaux que nous allons analyser est celui de Claire MAZZUCATO et Adrien JUPITER sur le package Data.table. Ils commencent par nous présenter le package, comment l'installer et son usage. Le package Data.table est utilisé pour manipuler des bases de données, il est présenté comme une alternative à data.frame qui est l'outil par défaut de R, le dataset est aussi présent dans R et est celui de mtcars. Au cours de la présentation ils vont montrer la différence entre data.frame et data.table.

1.2.2 Explication du code

```
dt1 <- mtcars_dt[5:25,.(carname, mpg, cyl)]
dt2 <- mtcars_dt[1:10, .(carname, gear)]
dt3 <- mtcars_dt[2:12, .(carname, disp)]

# Inner Join
merge(dt1, dt2, by='carname')
#> <returns 6 rows>
# Left Join
merge(dt1, dt2, by='carname', all.x = T)
#> <returns 21 rows>
# Outer Join
merge(dt1, dt2, by='carname', all = T)
#> <returns 25 rows>

dcast.data.table(mtcars_dt, cyl ~ carb, fun.aggregate = mean, value.var = 'mpg')
```

Dans les premières lignes, ils ont créé des sous ensembles du dataset pour l'utilisation de la fonction *merge*(). Ici on voit l'utilisation de 3 types de fusions : inner join, left join et outer join, pour fusionner deux tables il faut avoir une colonne en commun, dans cette exemple la colonne commune est carname. Le premier Inner join indique qu'on veut fusionner dt1 et dt2 en gardant seulement les lignes non vide des deux tables simultanément, le deuxième left join garde les lignes de la première table non vide et le dernier prends toute les lignes non vides des deux tables.

La dernière ligne correspond à la création d'une table dynamique montrant la moyenne de kilométrage (paramètre fun.aggregate = mean), en fonction du type de cylindrés et du type de carburant avec le ~.

1.2.3 Evaluation

- Visuel : Le visuel est plutôt propre, mais il manque un peu de numérotations et un sommaire pour s'y retrouver ;
- Reproductibilité : Le dataset est donné, l'installation est expliqué et les codes sont donnés ;
- Difficulté : Le package semble très populaire et complet, les fonctions présentés sont assez simple ;
- Didactique : Les chunks de codes sont bien expliqués, de plus les fonctions sont assez simple à comprendre ;

- Utilité : Le package est souvent utilisé et utile pour la manipulation de dataset, c'est un package très utile qui permet d'avoir une alternative de meilleur qualité que data.frame.

1.2.4 Conclusion

En conclusion, le package présenté était relativement simple et connu de beaucoup, le travail est propre visuellement mais difficile de savoir facile quelles informations on peut y trouver. Au niveau didactique et en terme de reproductibilité le package est bien présenté.

1.3 Le troisième - Prophet

1.3.1 Synthèse de la présentation

Le troisième des douzes travaux que nous allons analyser est celui de Lucas BILLAUD sur le package Prophet qui a été développé dans le but d'étudier et de faire de la prédiction sur les séries temporelles. Le package excelle dans le cas où la data connait des cycles saisonnier, il est aussi efficace contre les données manquantes et celles aberrantes. Le choix a été fait de faire la présentation en anglais, et les dataset présenté contient le nombre de personne allant sur la page wikipédia de Peyton Manning et le second de R. L'auteur nous introduit le package avant de nous expliquer les aspects mathématiques derrière le package.

1.3.2 Explication du code

```
#we specify growth = linear but you can also not specify (default setting)
m <- prophet(df, growth = "linear")

#now we specify growth to logistic
m <- prophet(df, growth = "logistic")
#the carrying capacity is usually set using data expertise about the market size
df$cap <- 8.5

#now we specify seasonality
m <- prophet(df, changepoint.prior.scale = 0.5)

#now we specify holidays
m <- prophet(df,holidays =  holidays)

#we use the same process for the forecasting part
future <- make_future_dataframe(m, periods = 1826)
forecast <- predict(m,future)
```

Ici nous avons regroupé les parties de codes que nous avons jugé les plus intéressantes. En effet en terme de difficulté syntaxtique c'est plutôt inexsitant, cependant ce que ces lignes représentent sont important. L'auteur a très bien amené ces quelques lignes de code en introduisant les aspects mathématiques de l'équation utilisé.

L'équation est la suivante : $y(t) = g(t) + s(t) + h(t) + e(t)$

- $g(t)$: représente l'évolution (growth), c'est là qu'on choisit le modèle, linéaire, logarithmique etc..
- $s(t)$: représente des récurrences, d'où le terme saisonnier.
- $h(t)$: représente les vacances, ou périodes qui sont connus pour être impactant, l'exemple de la présentation est le superbowl.
- $e(t)$: représente l'erreur.

Les lignes de codes représentes les 3 premiers termes, et comment les modifier pour trouver le bon modèle.

1.3.3 Evaluation

- Visuel : Le rapport est très propre, il manque un sommaire mais sinon le reste est qualitatif et surtout la construction est logique ;
- Reproductibilité : Les csv sont données, les lignes de codes y sont bien indiqués ce qui le rend reproductible ;
- Difficulté : Le package est en concurrence avec Arima, l'auteur a choisi ici d'en faire une introduction plutôt que d'un guide avancé ;
- Didactique : Les chunks de codes sont bien expliqués, de plus les fonctions sont assez simple à comprendre ;
- Utilité : Le package semble avoir une belle utilité sur les séries temporelles, et notamment celle avec des récurrences saisonnier.

1.3.4 Conclusion

En conclusion, le package est très intéressant et est très bien présenté de façon didactique, malheureusement le guide n'est pas très avancé et n'introduit que la base du package.

1.4 Le quatrième - Shiny

1.4.1 Synthèse de la présentation

Le quatrième des douzes travaux que nous allons analyser est celui de William ROBACHE et Rindra LUTZ sur le package Shiny qui a été développé par RStudio qui permet de créer des applications dynamique web. Pour que cela marche, il faut créer l'UI (user interface) qui comme son nom l'indique sera l'interface de l'utilisateur et le serveur qui contiendra les commandes données par ce dernier. A travers la création d'une application simple qui permet de visualiser les x premières lignes d'un ensemble de données.

1.4.2 Explication du code

```
# Définition de l'interface utilisateur de notre application
ui <- shinyUI(fluidPage(

  # Le titre de votre application
  titlePanel("Aperçu d'un dataset"),

  #Indiquer le 'layout' de votre application :
  #autrement dit le squelette visuel de l'application
  sidebarLayout(
    #Composants de la région gauche de l'application
    sidebarPanel(
      #Ici, nous insérons un champ pour entrer un chiffre,
      #ainsi qu'un menu déroulant
      textInput(inputId = "lignes",
                label = "Combien de lignes voulez-vous voir ? ",
                value = 10),
      selectInput(inputId = "labs",
                  label = "Dataset",
                  choice = c("cars","rock","beaver1", "sleep")
      )
    ),

    #Ici, nous indiquons l'élement qui sera présent dans la fenêtre principale :
    #l'element "dataset", qui est un graph
    mainPanel(
      tableOutput("dataset")
    )
  )))

server <- shinyServer(function(input, output) {
  #On retrouve ici l'élement "dataset", qui communique avec ui par 'output'.
  output$dataset <- renderTable({
    #Nous imprimons les éléments d'après les données en entrée :
    #ces derniers sont appellés avec `input` puis le nom de la composante
    #de ui (ici 'labs' et 'lignes')
```

```
    if(input$labs == "cars"){
      print(head(cars, input$lignes))
    } else if(input$labs == "rock"){
      print(head(rock, input$lignes))
    } else if(input$labs == "sleep"){
      print(head(sleep, input$lignes))
    } else {
      print(head(beaver1, input$lignes))
    }
  })
})
```

La présentation du package étant courte, nous avons décidé de reprendre tout le code présent sur le document. Les portions de codes sont bien détaillés c'est pourquoi nous jugeons que l'explication du code serait inutile, le seul point ambigü c'est peut être le fait de ne pas avoir explicitement dit que cars, rock, beaver1 et sleep sont des datasets. Un point qui est plutôt dommage c'est qu'il n'y a pas les résultats des codes car même si le code est bien expliqué ça aurait été bien de voir le résultat produit par cette portion de code.

1.4.3 Evaluation

- Visuel : Le rapport est propre, il manque un sommaire mais comme il est court cela ne pose pas trop de problème et il y a une logique dans sa contruction ;
- Reproductibilité : Les datasets utilisés sont des datas déjà intégrés dans R ;
- Difficulté : Le package à l'air complet mais pour l'utilisé complètement il semble nécessaire de connaître un peu de html ou du CSS. Malheureusement les auteurs n'ont qu'introduit le package ;
- Didactique : Les chunks de codes sont bien expliqués, de plus les fonctions sont assez simple à comprendre ;
- Utilité : Il est difficile de voir les possibilités du package à partir de juste ce document, mais il est toujours utile d'avoir un outil comme celui-ci.

1.4.4 Conclusion

En conclusion, le package semble intéressant et même si ce dernier a été bien expliqué, il a seulement été légèrement introduit ici ce qui est dommageable. De plus de ne pas avoir pu observer de résultats l'est d'autant plus.

1.5 Le cinquième - RandomForest

1.5.1 Synthèse de la présentation

Le cinquième des douzes travaux que nous allons analyser est celui de Thomas MASSE sur le package RandomForest, comme son nom l'indique le travail en question traite le sujet des arbres décisionnelles et donc de la prédiction. L'auteur commence par nous introduire ce qu'est une forêt aléatoire, nettoie ses données puis applique les fonction du package. Le set de données utilisé est celui du jeu vidéo PUBG, le rappot est très long et une grande partie est dédiée à l'analyse et le nettoyage des données. Une grande quantité de travail a dû être nécessaire pour écrire ce rapport, nous nous devions de l'analyser.

1.5.2 Explication du code

```
solo <- randomForest(winPlacePerc ~ ., data = head(trainsetSolo, 10000), na.action = na.omit, importanc

varImpPlot(solo)

predictionSolo <- predict(solo, testsetSolo)

testsetSolo$winPlacePerc <- predictionSolo
```

Bien que la présentation soit longue, la plupart du code concerne le nettoyage des données ce qui n'est pas directement lié au package en lui-même et donc nous avons décidé de ne choisir que quelques lignes de codes. La plus importante est évidemment la première qui contient $randomForest()$, c'est elle qui entraîne le modèle et par conséquent est celle la plus intéressante à comprendre.

- winPlacePerc correspond ici à la variable que l'on veut prédire, en fonction de celles sélectionnées qui correspondent aux 10000 premilères lignes du dataset solo.

En terme de résultats nous pouvons lire :

- Le nombre d'arbres créé par défaut la fonction va créer 500 arbres.
- On peut aussi avoir la main sur le nombre de variables utilisé à chaque branche dont la valeur par défaut est la racine carré du nombres de prédicteurs mais que l'on peut modifier avec mtry =.
- Et le résidu qui est calculé à partir de la différence entre la valeur à prédire et celle prédite toutes deux au carré.

La fonction $VarImplot()$ nous permet d'observer les prédicteurs les plus importants, l'auteur nous montre aussi comment trouver le mtry optimal en traçant l'évolution du résidu en fonction des différentes valeurs de mtry tout simplement.

Et pour finir les dernières lignes nous montrent, après avoir faire le même procédé de nettoyage des données pour créer le dataset à tester, comment tout simplement utiliser la fonction $predict()$ et comme obtenir le résultat.

1.5.3 Evaluation

- Visuel : Le rapport est long, beaucoup de tableaux de résultats et la mise en page reste assez minimaliste ;
- Reproductibilité : Le dataset est connu, toutes les étapes de nettoyages sont données ce qui rend l'étude facilement reproductible ;
- Difficulté : Le package n'est pas difficile, mais il y a des notions et une méthodologie à connaître ;
- Didactique : La présentation est détaillée, parfois ça peut rendre les choses plus complexes mais dans le cas présent le travail est très didactique notamment concernant les étapes préliminaires de présentation de la data, nettoyage, choix des valeurs d'entraînement ;
- Utilité : Les forêts aléatoires sont des modèles de prédictions très simples mais qui peuvent être très efficace, maîtriser un package tel que RandomForest est un plus. Cependant il existe un détail important à ne pas oublier, et c'est l'overfitting qui est plutôt présent dans les algorithmes de ce type.

1.5.4 Conclusion

En conclusion, le package est extrêmement bien introduit par l'auteur, plus que le package, il nous montre une vraie méthodologie que l'on peut reproduire pour d'autres datasets. Etant donné l'effort fourni on peut comprendre que la mise en page ait été simpliste.

2. Travaux Maths

2.1 Le sixième - Algèbre tropicale

2.1.1 Synthèse de la présentation

Le sixième des douzes travaux que nous allons analyser est celui de Marion DANYACH sur un papier de Dominique CASTELLA et Stephane GAUBERT intitulé "Algèbre de groupe en caractéristique 1 et distances invariantes sur un groupe fini". Le papier traite de l'algèbre tropicale, qui re-définit l'algèbre que nous avons appris depuis toujours, elle trouve son utilité dans SQL par exemple par sa caractérisque qui fait qu'un script idempotent ne modifie pas la base sur laquelle on l'applique. L'auteur de l'analyse commence par introduire ce qu'est l'algèbre linéaire, quelques définitions et applications utiles avant de rentrer plus en détail sur le papier en lui-même. Le papier lui est très étoffé, beaucoup de notions y est introduit ce qui rend la compréhension difficile.

2.1.2 Explication des formules

Les équations retenues proviennent de la proposition n°20 :

$$ef = e + f \tag{1}$$

La proposition nous dit que si e et f sont idempotents centraux (IC) alors nous avons $e + f$ idempotent si et seulement si (1). Par définition ef est idempotent si e et f sont IC et on a $ef \geq e + f$, or on a $(e + f)^2 = e + f + ef$ ce qui implique que $ef = e + f$.

La seconde partie de la proposition dit que si e est idempotent central alors il est irréductible s'il vérifie : $e = fg$ où f et g sont idempotents centraux et donc $e = f$ et $e = g$. Cette partie n'a pas été expliquée par l'auteur, et de notre côté la démonstration nécessite une bonne compréhension des autres notions et nous n'avons pas réussi à réellement comprendre le résultat.

2.1.3 Evaluation

- Visuel : Le visuel est plutôt propre, dommage que le texte ne soit pas justifiée. ;
- Difficulté : les formules et notions ne sont pas très compliqué, cependant étant un type d'algèbre il faut pouvoir s'y habituer ;
- Didactique : La notion d'algèbre tropicale est bien introduite, et beaucoup d'annexes sont données pour mieux appréhender les différentes notions cependant les formules choisis sont expliquées de façon assez superficiel ;
- Originalité : le papier choisi est original, c'est pourquoi nous avons voulu le lire ;
- Analyse : L'analyse donne l'impression que la compréhension est assez légère, mais avec autant de nouvelles notions cela peut se comprendre.

2.1.4 Conclusion

En conclusion, seule une courte partie du papier a été étudié dans l'analyse, beaucoup de notions assez avancées n'ont pas été traité et la petite partie analysée est faite de façon assez superficielle. Nous avons eu beaucoup de mal à corriger l'explication donné par l'auteur sur la proposition choisie ce qui montre la difficulté du papier.

2.2 Le septième - R-INLA

2.2.1 Synthèse de la présentation

Le septième des douzes travaux que nous allons analyser est celui de Jiayue LIU sur un papier de Pr. Paula MORAGA intitulé "Geospatial Health Data : Modeling and Visualisation with R-INLA and Shiny". Ce papier traite de l'analyse des modèles spatio-temporels avec R-INLA, et son application sur les cas de VIH en Ohio, Amérique au cours des années. Mais avant ça, l'auteur nous introduit les fondamentaux mathématiques derrière ces modèles et notamment la pobabilité et l'inférence bayésienne. Il passe ensuite à l'exemple et l'application du modèle INLA sur R avec les différents packages nécessaires.

2.2.2 Explication des formules

$$p(B \mid A) = \frac{p(A \mid B) * p(B)}{p(A)} \tag{2}$$

$$\pi(\mathbf{y}) = \int_{\theta \in \Theta} \pi(\mathbf{y} \mid \theta)\pi(\theta)d\theta \tag{3}$$

La première formule correspond au Théorème de Bayes, la différence par rapport à la probabilité classique est qu'il y a une part de subjectivité sur la valeur de l'*a priori* ($p(B)$). Dans le cas où l'a priori a une valeur fixe, et que la prédiction n'a pas trop de résultats possibles, cette formule est la plus adaptée mais souvent ce ne sera pas le cas d'où l'utilité de la seconde formule. La différence est qu'à la place des probalités fixes, il y a des distributions ce qui explique la forme de $\pi(\mathbf{y})$ qui représente une aire.

2.2.3 Evaluation

- Visuel : Le visuel est propre, dommage que les résultats ne soient calqué sur la map à la fin ;
- Difficulté : les formules et notions choisis ne sont pas compliquées, cependant dans l'ouvrage de Pr. Paula MORAGA on peut y trouver des informations plus détaillées ;
- Didactique : La présentation est bien faite, les notions nécessaire à la présentation sont expliquées et des documents annexes ont été fournis pour plus de détails, ce qui est le cas de l'exemple aussi ;
- Originalité : Le sujet traité est plutôt ordinaire ;
- Analyse : L'analyse est plutôt bonne, les notions fondamentaux sont bien comprises et les packages utilisés semblent être maîtrisés.

2.2.4 Conclusion

En conclusion, nous avons trouvé que le rapport était bien écrite, agréable à la lecture et didactique. Il y a une bonne répartition entre la partie théorique et la partie application qui était intéressante. L'auteur aurait pu entrer plus en détail mais ça aurait alourdi la lecture.

2.3 Le huitième - Arbres de décisions

2.3.1 Synthèse de la présentation

Le huitième des douzes travaux que nous allons analyser est celui de Antoine SERREAU, Benjamin GUIGON et Corentin BRETONNIERE sur le sujet des arbres de décisions. Les auteurs reviennent sur l'aspect mathématique des deux types d'arbres : régression et classification, notamment la notion de pureté et coût du noeud. Ils utilisent un exemple inspiré du travail de Christophe Chesnau intitulé "Introduction aux arbres de décisions" pour nous expliqué les bases à l'aide du dataset Iris.

2.3.2 Explication des formules

$$G_i = 1 - \sum_{k=1}^{n} p_{i,k}^2 \tag{4}$$

$$J(k) = (\frac{m_{gauche}}{m})G_{gauche} + (\frac{m_{droite}}{m})G_{droite} \tag{5}$$

La première formule permet de calculer la pureté d'un noeud, un noeud est dit pur si tous les individus associés sont de la même classe et que la valeur est 0. La formule est donc basée sur la probabilité d'avoir un individu d'une classe k parmi la population au noeud i.

La seconde formule concerne le coût du noeud, celle-ci inclu la formule de la pureté d'un noeud ce qui implique que plus les noeuds sont purs plus le coût est faible.

2.3.3 Evaluation

- Visuel : Le visuel est propre, dommage que les chunks de code R n'ont pas été intégré directement à l'aide d'un fichier rmarkdown ;
- Difficulté : Les notions indroduit sont basiques et ne font pas l'objet de difficultés, dommage de ne pas avoir développé la partie sur les arbres de classifications notamment la partie sur les optimum locaux ;
- Didactique : Le travail est clair et concis, avec les notions qui sont plutôt bien expliquées ;
- Originalité : Le sujet traité est plutôt ordinaire ;
- Analyse : L'analyse est plutôt bonne, les notions fondamentaux sont bien comprises.

2.3.4 Conclusion

En conclusion, nous avons trouvé que le rapport était bien écrite, agréable à la lecture et didactique. Les points d'améliorations qui auraient pu rendre leur devoir meilleur sont l'utilisation de Rmarkdown et non juste de Latex pour les chunks de code et aller un peu plus sur les notions, surtout que les arbres sont assez populaires bien documentés.

2.4 Le neuvième - Cryptographie

2.4.1 Synthèse de la présentation

Le neuvième des douzes travaux que nous allons analyser est celui de Marko ARSIC et William ROBACHE sur la cryptographie et la théorie des nombres. La cryptographie qui est la science de crypter des informations dans le but d'assurer la confidentialité entre l'émetteur et le destinataire était une science ominiprésent dans le domaine militaire, notamment les messages dits codés. Dans le rapport, les auteurs commencent par introduire le cryptage classique à clé symétrique avant de nous détailler celles à clé asymétrique qui sont des versions évoluées. Ce sont ces dernières qui sont utilisées dans le domaine bancaire par exemple.

2.4.2 Explication des formules

$$f_D(M) = f_D[f_C(m)] = m \tag{6}$$

Cette formule résume à elle seule le fonctionnement d'une clé de cryptage asymétrique en effet on y voit m le message, ou l'information à garder secrète, f_C la fonction de cryptage et f_D la fonction de décryptage. La fonction de cryptage est une fonction à sens unique avec trappe, ce qui implique qu'en connaissant la fonction il est difficile de retrouver le message et la trappe qui est la fonction f_D ajoute une protection en plus.

2.4.3 Evaluation

- Visuel : Le visuel est propre excepté le fait qu'il n'y ait pas de sommaire, ni de séparation entre les titres et les textes ;
- Difficulté : Les notions indroduit sont basiques, ils auraient pu augmenter encore un peu la difficulté ;
- Didactique : Les notions sont bien amenés, pas de soucis à ce niveau là ;
- Originalité : Le sujet traité est différent de ceux proposé par la promotion, ça change un peu ;
- Analyse : Les notions de cryptographie introduit dans leur travail est maîtrisé.

2.4.4 Conclusion

En conclusion, nous avons trouvé que le rapport était bien écrite, agréable à la lecture et didactique. Nous aurions aimé que les auteurs aillent un peu plus loin et de voir un exemple concret, par exemple la création d'un petit message crypté.

2.5 Le dixième - Régression linéaire sur variable fonctionelle

2.5.1 Synthèse de la présentation

Le dixième des douzes travaux que nous allons analyser est celui de Maxime ALLAKERE HORMO sur la regression sur une variable fonctionnelle inspiré par la thèse de Laurent DELSOL. L'auteur commence par nous expliquer ce qu'est une variable fonctionnelle, une variable est dite fonctionnelle lorsque celle-ci appartient à un espace de dimension infinie. C'est de ça que vient son utilité croissante, car en effet avec l'abondance de données mesurées sur des échelles de plus en plus précises, il fallait absolument développé des outils permettant de suivre au mieux cette évolution.

2.5.2 Explication des formules

$$Y = aX + b + \epsilon,\ X(\mu, \gamma^2)\, et\, N(0, \sigma^2) \tag{7}$$

$$Y = aX + b + \epsilon,\ E(\epsilon|X) = 0 \tag{8}$$

$$Y = r(X) + \epsilon,\ r \in C(\mathbb{R})\, et\, R(\epsilon \mid X) = 0 \tag{9}$$

Ces 3 équations sont des modèles de régression, la (7) est une équation paramétrique c'est à dire que le nombre de paramètres est fixe par rapport à la taille de l'échantillon, l'équation (9) elle est non-paramétrique et donc le nombre de paramètres peut augmenter avec la taille de l'échantillon. L'équation (8) elle a la particularité d'être soit paramétrique soit non paramétrique suivant les conditions énoncés par l'auteur.

En ce qui concerne le test de structure, l'hypothèse nulle ou alternative est validée selon la valeur du résidu que l'on obtient.

2.5.3 Evaluation

- Visuel : Le visuel fait très compact, il manque une introduction pour pouvoir voir le plan plus clairement ;
- Difficulté : La notion de régression linéaire sur variable fonctionnelle à l'air plutôt compliqué, cependant seule la 1ère couche a été abordée ici ;
- Didactique : L'auteur a fait un effort pour vulgariser au mieux les notions abordées et il y a pas mal d'exemples d'applications, dommage qu'il n'y ait pas eu des graphiques pour illustrer un peu plus les propos ;
- Originalité : Le sujet traité est différent de la simple régression linéaire, le sujet à l'air très intéressant et en adéquation avec l'évolution de la data ;
- Analyse : Au vue de l'effort de vulgarisation, la compréhension des notions présentes sur le document est correcte.

2.5.4 Conclusion

En conclusion, le travail est de bonne qualité, il est difficile d'analyser une thèse entière pour en tirer des informations qui sont à porter de tous, d'ailleurs aucun autre élève ne s'est lancé dans l'étude d'une thèse. Cependant, l'aspect du rendu donne l'impression que ça va être lourd à lire.

3. Auto-critique

3.1 R - Ggplot2 & StatsBombR

Lors de ce rapport, nous avons analysé 5 travaux sur R, mais pour choisir ces 5 travaux nous en avons lu beaucoup plus. De façon objective, le visuel des deux packages présentés sont de bonnes qualités, malheureusement il n'y a pas de sommaire donc nous nous rendons compte que ça peut rendre la lecture beaucoup plus complexe surtout que beaucoup de fonctions différentes ont été abordées.

Les rapports sont écrits de sortes que les utilisateurs puissent obtenir des résultats similaires avec les datasets de leur choix. L'aspect didactique du travail a été soigné et un réel effort pour expliquer toutes les lignes de codes à été fourni. Il était important de basé le travail sur des exemples concret car nous ne voulions pas juste donner une liste exhaustive des fonctions disponibles dans les packages. De plus, nous estimions que cela rendrait la lecture plus agréable et faciliterait la compréhension de certaines des fonctions.

Nous pensons qu'à travers nos travaux l'utilisateur est capable de travailler avec les packages respectifs, surtout que le niveau de difficulté est assez faible. A noter que peu de travaux ont été rendu en anglais et même si le package ggplot2 n'était pas un choix très original, le second sur StatsBombR apporte un peu plus d'originalité.

3.2 Maths - Prédictions

En ce qui concerne les travaux en Maths, il m'a semblé que les consignes initiales n'ont été respecté que par très peu de groupes, par exemple nous n'avons lu aucune "critique" des papiers de recherche ou de comparaison entre les papiers. Comparer notre travail aux autres groupes seraient donc étranges, cependant, il est important de pointer ce que nous avons trouvé de bons ou moins bons. Par exemple, notre rapport n'avait pas pour but d'être didactique, ce que nous trouvons dommage au vue de certain travaux en maths que nous avons trouvé très intéressant.

La différence se voit surtout sur l'analyse du papier en lui-même, nous trouvons dommageable que très peu de personnes ont vraiment analysé leur papier choisi, et forcément il n'y a pas eu de comparaison entre les papiers choisis dans leur groupe non plus.

Un point dont nous sommes assez déçu est l'explication des formules mathématiques, cela va avec le fait que notre travail n'avait pas pour but d'être didactique, mais nous trouvons que ça restait beaucoup trop léger et que ça n'avait pas de plus value dans notre analyse. De plus nous n'avons pas bien lié ces formules avec les sujets du papier qui abordait le thème de la prédiction.

En résumé, même si de notre point de vue les consignes ont été respecté de notre côté, il aurait été intéressant de s'imprégner des formats un peu plus didactique des autres travaux pour rendre le dossier plus agréable à lire.