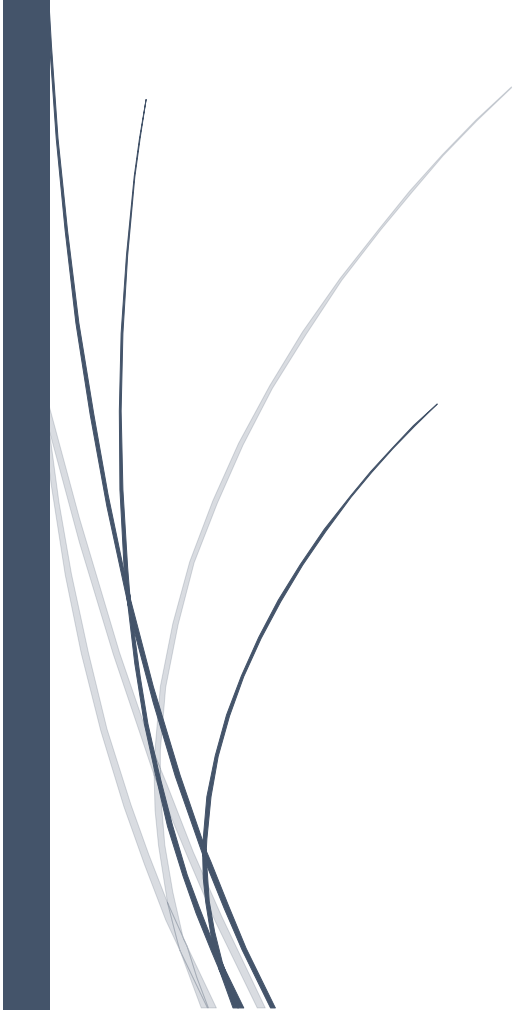
13/12/2018

Synthèse du stage M1 – Détection d'événements inhabituels sur les données SurSaUD



Santé Publique France

Table des matières

Contenu du fichier ZIP

Le fichier ZIP contient les éléments suivants :

- Le script de traitement des données : « donnees_historique.R »
- Le script de création de séries temporelles (un exemple) : « serie_temporelle.R »
- Un dossier data contenant les données SurSaUD (période 2010-2017), quatre fichiers RDS contenant le total des actes par semaine/dep, semaine/reg, jour/dep et jour/reg
- Un fichier Excel contenant la liste des regroupements syndromiques « liste_codes_RS.xslx »
- Un fichier Excel contenant un récapitulatif des tables SQL : « tables_historique_bdd.xslx »
- Le mémoire du M1

Données

Contenu des données

L'ensemble des données est contenu dans le fichier RDS appelé « data_sursaud_histo.rds ».

Ce fichier contient 75 056 993 lignes et 5 colonnes :

- Identifiant_etablissement
- Identifiant_classe_age
- Identifiant_regroupement_syndromique
- Date_de_passage
- Nombre de passage

Le fichier contient l'ensemble des passages pour les deux réseaux, OSCOUR et SOS Médecins, durant la période janvier 2010 à décembre 2017, pour une quarantaine de regroupements syndromiques. Ces regroupements syndromiques sont listés dans le fichier « liste_codes_RS.xslx ».

Traitement des données

Tout le processus de traitement des données se situe dans le script « donnees_historique.R ».

Préparation des données

Avant d'exploiter les données, ces dernières ont besoin d'être transformées pour la suite.

1ère étape : Le code départemental de la Corse est ré-écrit de manière propre, c'est-à-dire « 2A » ou « 2B ».

2ème étape : Pour pouvoir travailler sur les régions, on rajoute un code régional. Grâce aux codes départementaux, on peut retrouver le code régional selon les normes de l'INSEE : https://www.insee.fr/fr/information/3363419

Ces codes régionaux se retrouvent grâce aux codes départementaux et on p

3ème étape : Les données étant journalières et, pour pouvoir travailler en semaine, il est nécessaire de récupérer la semaine courante. Pour cela, on calcule le premier lundi de la semaine courante grâce à une fonction « prevM » dont le code peut être retrouvé ici : https://stackoverflow.com/questions/31459651/converting-date-to-monday-of-that-week

Agrégation des données

Après transformation des données, les données sont agrégées selon quatre niveaux différents :

1. Par jour et par département
2. Par jour et par région
3. Par semaine et par département
4. Par semaine et par département

Pour chaque niveau, on somme tous les actes/passages afin d'obtenir le nombre total pour une classe d'âge, une zone géographique (département ou région) et une date donnés pour tous les regroupements syndromiques.

Dans les données, dans la colonne « Identifiant_regroupement_syndromique », il existe deux libellés importants : « TOUS_T_DRP » et « TOUS_T_DSO ».
Ces deux libellés sont l'ensemble des actes/passages pour une classe d'âge, une zone géographique et une date donnés pour les réseaux OSCOUR et SOS Médecins, respectivement. (il s'agit donc d'une somme de tous les regroupements syndromiques).

Pour des raisons de facilité et pour la suite, ces totaux seront extraits des données et stockés dans des fichiers .RDS à part. Cela permettra de faire des vérifications.

Ces fichiers ont tous la même structure :

- Code départemental ou régional
- Identifiant de la classe d'âge
- Date de passage
- Le total pour le réseau OSCOUR (« TOUS_T_DRP »)
- Le total pour le réseau SOS Médecins (« TOUS_T_DSO »)

Ces différents totaux se trouvent dans 4 fichiers RDS :

Nom fichier	Nombre de lignes
total_actes_codes_jour_dep.rds	2 176 530
total_actes_codes_jour_reg.rds	423 671
total_actes_codes_sem_dep.rds	315 475
total_actes_codes_sem_reg.rds	61 265

Tableau 1 : Totaux des regroupements syndromiques avec le nombre de lignes

Ces totaux seront remis dans les données mais sur une différente forme c'est-à-dire sous forme de colonne pour faciliter le calcul de proportion et donner une meilleure visibilité.

Les données agrégées auront donc la forme suivante :

- Code départemental ou code régional
- Identifiant de la classe d'âge
- Date de passage ou premier lundi (données journalières ou hebdomadaires)
- Identifiant du regroupement syndromique
- Nombre de passage
- Source qui vaut SAU pour OSCOUR et SOS pour SOS Médecins
- Total des actes codés c'est-à-dire les totaux présentés précédemment.

IMPORTANT

Lors de l'agrégation au niveau régional, il existe des départements qui n'ont pas de code régional selon l'INSEE. Pour ces départements, le code régional 0 leur a été attribué.

Stockage des données

Les données seront stockées dans des tables SQL.
Il a été décidé de créer une table selon :

- Les deux réseaux : OSCOUR et SOS Médecins
- Les niveaux géographiques souhaités : France métropolitaine, DOM-TOM et France entière
- La temporalité : données journalières ou hebdomadaires

Ceci donne un total de 24 tables en ajoutant quatre tables pour les totaux donc 28.

Tables	Total
SAU_JOUR_DOM_DEP	1 177 189
SAU_JOUR_DOM_REG	1 173 149
SAU_JOUR_FRANCE_DEP	27 426 393
SAU_JOUR_FRANCE_REG	8 053 781
SAU_JOUR_METRO_DEP	26 249 204
SAU_JOUR_METRO_REG	6 880 632
SAU_SEM_DOM_DEP	313 931
SAU_SEM_DOM_REG	310 733
SAU_SEM_FRANCE_DEP	6 703 586
SAU_SEM_FRANCE_REG	1 556 213
SAU_SEM_METRO_DEP	6 389 655
SAU_SEM_METRO_REG	1 245 480
SOS_JOUR_DOM_DEP	157 407
SOS_JOUR_DOM_REG	157 407
SOS_JOUR_FRANCE_DEP	7 894 615
SOS_JOUR_FRANCE_REG	4 172 342
SOS_JOUR_METRO_DEP	7 737 208
SOS_JOUR_METRO_REG	4 014 935
SOS_SEM_DOM_DEP	50 312
SOS_SEM_DOM_REG	50 312
SOS_SEM_FRANCE_DEP	2 341 028
SOS_SEM_FRANCE_REG	928 327
SOS_SEM_METRO_DEP	2 290 716
SOS_SEM_METRO_REG	878 015
TOTAL ACTES_CODES	
JOUR_DEP	2 176 530
JOUR_REG	423 671
SEM_DEP	315 475
SEM_REG	61 265

Ce résumé des tables se trouve aussi dans le fichier « tables_historique_bdd.xslx ».

Une fois les tables SQL créées, le script alimente les tables avec du calcul parallèle pour aller plus vite.

IMPORTANT

Selon les capacités de l'ordinateur, le script peut prendre plusieurs heures à quelques jours (deux jours) pour l'alimentation des tables. Il est donc préférable, pour vérifier dans un premier temps, de ne pas ajouter toutes les données directement. C'est-à-dire qu'il est préférable de rajouter, par exemple, les données journalières et par département puis journalières et par région, etc.

Points importants

Le script ne permet que d'ajouter de nouvelles données et donc ne peut pas modifier des données. C'est-à-dire qu'en cas de doublon de ligne dans une table, le script retournera une erreur. Si cela arrive, il faudra alors vider toutes les tables que l'on souhaite alimenter.

Selon l'interface de la base de données, il se peut que cette dernière ne soit pas exacte, notamment sur le nombre de lignes affichées. Il est arrivé plusieurs fois qu'une table n'affichait pas correctement le nombre de lignes qu'elle contenait : par exemple, une différence d'environ 2 millions de lignes.

Pour vérifier si les tables contiennent bien le nombre de lignes, il suffit de les relire avec R et d'extraire le nombre de lignes.

Création des séries temporelles avec borne supérieur d'intervalles de prédiction et alarmes

Une fois les données agrégées et stockées, il est maintenant possible de créer des séries temporelles. De plus, il est possible d'appliquer des algorithmes de détection d'événements inhabituels qui ont été développés dans le package « surveillance » de R.

Le lien : https://cran.r-project.org/web/packages/surveillance/index.html

Quatre algorithmes sont utilisés pour la détection d'événements inhabituels : RKI3, Bayes 2, Farrington et EARS C2.
Ces algorithmes ont été conçus pour des données hebdomadaires mais il est possible de les faire tourner sur des données journalières.

Une description détaillées de ces algorithmes a été faite et se situe dans le document : « M1MSP_MEMOIRE_JOHANN_KUHN.pdf »

Le script montrant comment créer une série temporelle avec affichage de la borne supérieure d'intervalles de prédiction ainsi que les alarmes se trouve dans le script appelé « serie_temporelle.R ».

La méthode pour créer une série temporelle est à peu près identique pour les quatre algorithmes et peut se résumer de la façon suivante :

1ère étape : Extraction des données depuis la base de données
À cette étape, on choisit alors la zone géographique (France métropolitaine, DOM-TOM, France entière), le réseau ainsi que le type de données (hebdomadaires ou journalières)

2ème étape : Choix du regroupement syndromique, de la région ou du département et de la classe d'âge et calcul des proportions

3ème étape : Création d'un objet STS et d'un objet Disprog
L'objet STS est nécessaire pour être converti en objet Disprog. L'objet Disprog est utilisé par trois algorithmes : RKI3, Bayes 2 et Farrington alors que EARS C2 n'utilise que des objets STS.

L'objet STS a pour arguments :

- Les cas observés, ici les proportions
- Le point de départ : dans le cadre d'un travail hebdomadaire, ce sera la première année de 2010
- La fréquence : dans un travail hebdomadaire, la fréquence vaut 52 (car 52 semaines)
- Les dates

4ème étape : Calcul de la période que l'on veut étudier (par exemple 3 ans)

5ème étape : Application de l'algorithme souhaité
L'algorithme calculera alors les bornes supérieures des intervalles de prédiction ainsi que les alarmes à partir des données qu'on lui donne.

6ème étape : Création des séries temporelles
On crée d'abord la série temporelle puis on rajoute les bornes supérieures des intervalles de prédiction ainsi que les alarmes

Exemple de graphe

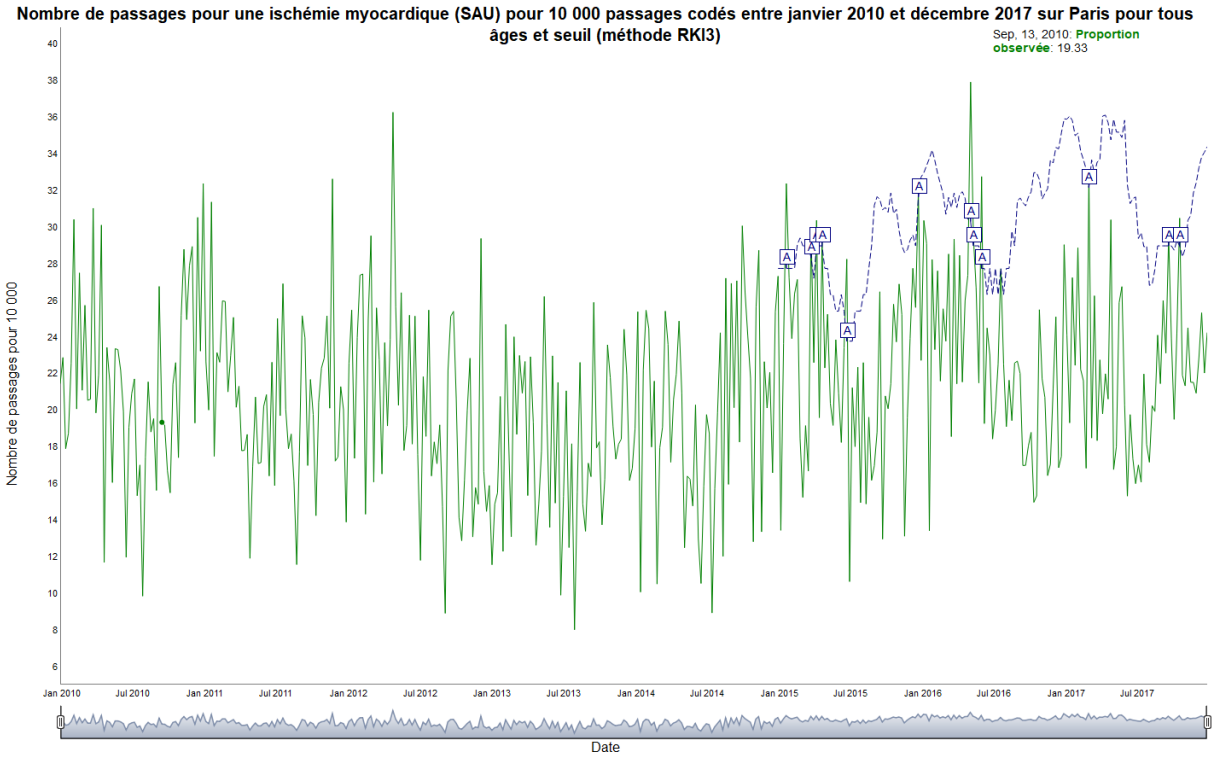


Figure 1 : Série temporelle avec borne sup des intervalles de prédiction ainsi que les alarmes

Points à ajouter

Le script de traitement des données n'est pas parfait car il n'est pas capable de modifier des données si celles-ci sont mises à jour. De plus, il n'est pas complètement automatique.

Selon la machine utilisée, l'alimentation des tables peut être très longue et peut faire saturer l'ordinateur. Il est donc préférable, dans une période de test, de ne pas lancer le script de traitement des données d'un coup mais de choisir les données que l'on veut ajouter aux tables. (par exemple, les données journalières et par département prennent environ deux jours pour être stockées dans des tables)

Pour les algorithmes utilisés, il faut faire attention aux données passés qui sont utilisés comme historique pour calculer les alarmes.
L'algorithme de Farrington, par exemple, prend, par défaut, trois ans d'historique (exemple : si on veut prédire pour l'année 2015, il va aller chercher des données en 2012). Il faut donc faire attention à la période que l'on veut étudier, même s'il est possible de modifier dans la fonction la période de l'historique.

Pour certains départements ou régions, il se peut qu'il n'y ait pas de total pour l'ensemble des regroupements syndromiques pour une date donnée. Par exemple, pour une semaine, il n'y a eu aucune donnée.
Ce cas n'est pas géré par les scripts. De plus, il faudra se poser la question s'il faudra laisser ce trou lorsqu'on utilisera les algorithmes car le calcul d'alarme peut en être faussé.

Un autre point que le script peut ne pas gérer et qu'il faudra vérifier. Imaginons que, pour un regroupement syndromique et une date donnés, il n'y a pas de données pour ce regroupement syndromique mais il existe bien un total. Il faudrait savoir si le script crée un « trou » ou s'il arrive à mettre la valeur 0 pour cette semaine, ce qui reviendrait à un calcul de proportion égal à 0.