**Documentation :**

**Calcul des indices de prix par la méthode des ventes répétées**

Edouard Pignède

Note : l’ensemble des scripts présentés dans cette documentation se trouve dans le dossier : « ./Github/toflit18\_data/scripts/Edouard »

1. Introduction

Le travail présenté ici permet le calcul des indices de prix par la méthode des ventes répétées pondérées pour l’ensemble des ports français, imports et exports. Il prend en entrée la base de données : *bdd\_courante.csv* que l’on peut trouver dans le fichier « ./Github/toflit18\_data/base » et retourne d’une part le csv *Index\_results.csv* qui contient les valeurs des indices de prix par année selon les paramètres considérées, et d’autre part le fichier Excel *Correlation\_matrix.xlsx* qui contient pour chaque entrée de *Index\_results.csv* , la matrice des corrélations systématiques entre les différents indexes.

Quatre scripts R sont utilisés au sein de ce travail placé dans le dossier (Indice\_ville\_scripts) :

* *Filtrage.R :* qui permet de filtrer la base de données initiale selon plusieurs paramètres qui seront définis ci-dessous.
* *Ventes\_repetees\_ponderees.R*: qui permet le calcul des indices de prix par la méthode des ventes répétées, à partir de la base de données filtrée.
* *Calcul\_index.R*: permet de mettre à jour *Index\_results.csv* et d’y rajouter de nouveaux paramètres
* *Calcul\_correlation.R*: permet, à partir d’*Index\_results.csv* de réaliser l’Excel *Correlation\_matrix.xslx*

La méthode des ventes répétées est généralement utilisée pour calculer des indices de prix immobiliers. Elle a l’avantage de pouvoir être utilisée facilement car elle nécessite uniquement des données sur la date et le prix des transactions immobilières des biens vendus au moins deux fois au cours du temps (Statistical Office of the European Communities et al. 2013). Dans cette étude, nous construisons les indices de prix en considérant que les produits vendus au cours du 18ème siècle, correspondent aux biens immobiliers.

L’indice est obtenu en considérant des régressions systématiques entre l’ensemble des transactions réalisées aux T années, selon l’équation suivante :

Avec Dnt le vecteur des variables égales à 1 lors de la revente (t), à -1 lors de la vente précédente (s) et 0 sinon, p le vecteur des prix des n produits considéré, βt le coefficient estimé et εnt le vecteur des classiques des erreurs (Statistical Office of the European Communities et al. 2013).

La première année considérée, l’indice vaut la valeur de base égale à 100 puis est égal à :

Pour chaque année où au moins une transaction a lieu.

1. Présentation des scripts
   1. *Filtrage.R*

Ce script possède une fonction principale, appelée Data\_filtrage, qui fait appel à sept autres fonctions secondaires.

La fonction Data\_filtrage prend en entrée 8 paramètres différents (avec leurs valeurs par défaut) :

* **Ville** égal au nom de la ville que l’on veut étudier entre « » 🡺 Port considéré (pour la suite on considère Marseille, Bordeaux, La Rochelle et Nantes).
* **Type** égal à « Imports » ou « Exports » (« Imports ») **🡺** Considère-t-on les imports ou les exports ?
* **Outliers** égal à T ou F (T) 🡺 Retire-t-on les outliers ? outliers d'une loi log-normale avec z-score modifié
* **Outliers\_coef** égal à un nombre positif (3,5) 🡺 Outliers\_coef utilisé pour la détection des outliers, plus il est grand plus le nombre d'outliers diminue
* **Trans\_number** égal à un entier positif (0) 🡺 Retire les produits apparaissant moins de Trans\_number fois dans la base filtrée ?
* **Prod\_problems** égal à T ou F (T) 🡺 Retire-t-on les produits avec un écart interquartile (3rd - 1st quartile) supérieur à 10 ?
* **Product\_select** égal à T ou F (F) 🡺 Conserve-t-on uniquement les produits de la sélection réalisée par Loïc ?
* **Remove\_double** égal à T ou F (T) 🡺 Rassemble-t-on les produits vendus plus de deux fois dans la même année.

Elle renvoie ensuite un dataframe composée des colonnes : year, product\_simplification, unit\_price\_metric, quantities\_metric, value, quantity\_unit\_metric, id\_prod\_simp (ID de product\_simplification), Date, id\_trans (ID de transaction), Part\_value (part du commerce totale considérée cette année-là), Part\_flux (part du flux totale considérée cette année-là).

Les fonctions secondaires suivantes sont appliquées une par une, dans l’ordre présentée ci-dessous :

1. **Read\_bdd\_courante**

Prend en entrée la ville et le type considérée. Lis le csv *bdd courante.csv*. Filtre par best\_guess\_region\_prodxpart, nom de la ville et type. Pour chaque produit conserve uniquement les ventes de la meilleure unité possible – l’unité qui apparaît dans le plus de transactions. Crée les ID trans et prod\_simp.

1. **Detect\_outliers**

Utile uniquement si Outliers = T

Prend en entrée la base de données issue de Read\_bdd\_courante et Outliers\_coef. Elle renvoie la base de données modifiée avec un nouvel attribut binaire : Outliers.

Puisque nos données sont des prix, nous considérons qu’ils suivent une loi log-normale (distribution classique pour des suites de prix). En passant au log on se ramène donc à une loi normale.

On a donc, Outliers = T lorsque le z-score modifié selon la méthode de Iglewicz et Hoaglin (1993) du log des prix est supérieur à Outliers\_coef.

Le z-score modifié est égal à :

Avec MAD égal la déviation absolue de la médiane.

Les auteurs conseillent Outliers\_coef = 3,5 qui est la valeur prise par défaut ici et qui correspond à environ 5% des valeurs extrêmes.

Plus d’informations : <https://www.itl.nist.gov/div898/handbook/eda/section3/eda35h.htm>

1. **Remove\_outliers**

Uniquement si Outliers = T.

Retire l’ensemble des observations avec Outliers = T et retire la colonne Outliers du dataframe.

1. **Remove\_double\_val**

Uniquement si Remove\_double = T.

Les produits vendus plus de deux fois la même année sont rassemblés en une seule transaction égale à la somme des anciennes transactions.

La méthode des ventes répétées ne peut considérer qu’une vente par an d’un même produit. S’il y a plusieurs transactions une même année, l’algorithme conserve la vente la plus chère. En retirant les transactions multiples, on supprime ce choix.

Cependant, on ne peut plus connaître le partenaire commerciale qui peut dépendre des transactions pour un même produit.

1. **Remove\_prod\_problem**

Uniquement si Prod\_problems = T.

Supprime les produits dont l’écart interquartile (Q3 – Q1) de la série des prix est supérieur à 10. Cela permet de retirer un petit nombre de produits défaillants. Ces produits ont souvent une erreur d’unité ou de quantité qui entraîne qu’une partie des valeurs des prix est très différentes de l’autre partie des valeurs des prix (facteur 10).

1. **Keep\_prod\_select**

Utile si prod\_select = T.

Charles Loïc a construit une base des produits ayant le plus de « sens commercial ». Cette base est stockée sur le csv *Product\_selection.csv.* Cette fonction entraine la suppression de l’ensemble des produits qui ne sont pas présents dans cette base.

1. **Keep\_trans\_number**

Uniquement si Trans\_number > 0.

Retire l’ensemble des produits qui sont vendus un nombre de fois inférieures à Trans\_number.

* 1. *Ventes\_repetees\_ponderees.R*

Ce script permet le calcul des indices par le biais de la méthode des ventes répétées. Il est composée d’une fonction principale : Filter\_calcul\_index et de deux fonctions secondaires : Calcul\_index, Plot\_index.

1. **Calcul\_index**

Cette fonction prend en entrée une base de données Data retournée par Data\_filtrage, ainsi que deux paramètres : **Ponderation** égal à T ou F (T) et **Pond\_log** égal à T ou F (F).

Elle réalise alors le calcul de l’indice par la méthode des ventes répétées en pondérant l’équation de la méthode présentée ci-dessous par la part du produit dans la valeur totale considérée (si Ponderation = T et Pond\_log = F) ou par le log de la part du produit dans la valeur totale considérée (si Ponderation = T et Pond\_log = T). Si Ponderation = F, aucune pondération n’est faite.

La fonction retourne un dataframe à deux colonnes : l’année et la valeur de l’indice.

1. **Plot\_index**

Prend en entrée l’indice renvoyée par la fonction Calcul\_index et plot l’indice sur un graphe avec en titre du graphe l’ensemble des paramètres considérées et leur valeur.

Enregistre automatiquement le graphe sous format PNG dans le dossier graphe, si celui-ci n’existe pas encore.

1. **Filter\_calcul\_index**

Prend en entrée les dix paramètres : Ville, Exports\_imports, Outliers, Outliers\_coef, Trans\_number, Prod\_problems, Product\_select, Remove\_double, Ponderation et Pond\_log.

Applique successivement la fonction Data\_filtrage, Calcul\_index et Plot\_index.

Renvoie un dataframe à quatre colonnes : l’année, la valeur de l’indice, la part du commerce considérée dans l’indice et la part du flux considérée dans l’indice.

* 1. *Calcul\_index.R*

Ce script est composé de deux fonctions : Update\_base et Add\_new\_param. A partir des scripts précédents, elles permettent de remplir le CSV Index\_results.csv avec la valeur des indices pour les différents paramètres.

1. **Update\_base**

Fonction qui ne prend aucun argument.

Elle efface le contenu du CSV Index\_results.csv et le remplace par de nouvelles valeurs pour les indices issus de la baseline (valeurs des paramètres par défaut) et pour une modification de chaque paramètre, laissant les autres valeurs par défaut.

Ce script est à utiliser lors de modifications de la bdd courante pour mettre à jour le calcul des indices avec la nouvelle base.

Remarque : le CSV Index\_results doit être fermé pour lancer la fonction.

1. **Add\_new\_param**

Ajoute les valeurs des indices du CSV Index\_results.csv pour chaque ville, imports et exports, à partir des valeurs choisies de l’ensemble des paramètres.

Si l’ensemble des paramètres choisis existe déjà dans l’Excel, ce script met à jour les valeurs de l’indice correspondant.

* 1. *Calcul\_correlation.R*

Ce script est composé d’une fonction : Calcul\_correlation\_matrix. Cette fonction ne prend pas d’arguments.

Elle permet le remplissage de l’Excel Correlation\_matrix.xlsx. Pour l’ensemble des villes du CSV Index\_results.csv, on crée deux onglets dans l’Excel : un pour l’indice Imports et un pour l’Exports. Ensuite une matrice de corrélation est calculée dans chaque onglet à partir de l’ensemble des indices calculés pour cette ville et ce type (Imports ou Exports), selon les différends paramètres considérés.

1. Présentation des Fichiers
   1. Index\_results.csv

Fichier CSV composé de 14 lignes : le port, le type (imports ou exports) et la valeur des 8 paramètres considérés pour le calcul de l’indice, l’année, la valeur de l’indice, la part du commerce considéré pour le calcul de l’indice cette année-là et la part du flux considéré pour le calcul de l’indice cette année-là.

Lors de changement de la bdd courante, ce CSV peut être mis à jour grâce à la fonction Update\_base du script *Calcul\_index.R*.

* 1. Correlation\_matrix.xlsx

Fichier Excel comportant un nombre d’onglet égal au nombre de villes considérées dans le CSV Index\_results multiplié par deux pour imports et exports. Chaque onglet a le nom de la ville et du type (imports ou exports) considéré.

Au sein de chaque onglet, on trouve une matrice de corrélation symétrique avec une diagonale égale à 1. Le nom des entrées de la matrice sont égales à la liste des noms de l’ensemble des 8 paramètres utilisés et leur valeur. La valeur est la corrélation entre les deux indices avec de tels paramètres.

Une valeur NA correspond au cas de deux indices non-comparables en raison de définition sur des années différentes. Par exemple, un filtrage trop important peut amener un indice à perdre une année d’observation. Dans ce cas, cet indice ne pourra plus être comparé avec l’ancien indice.

* 1. Product\_selection.csv

CSV issue du travail de sélection de Charles Loïc. Pour chaque port et type (Imports, Exports), on a la liste des produits commercialisés. La dernière colonne est égale à 1 si le produit est conservé lors de la sélection du paramètre Product\_select et 0 sinon.

* 1. Figure\_index

Le fichier Figure\_index contient l’ensemble des graphes des indices présents dans le CSV Index\_results.csv.

Le nom du fichier et du graphe correspond au nom et à la valeur de l’ensemble des paramètres permettant la construction du graphe. Les graphes permettent aussi d’observer la part de la valeur totale considérée dans la construction de l’indice pour chaque année.