MQT-6021-A21 – Travail pratique 4

Michael Morin

25 novembre 2021

Instructions

- L'équipe doit être enregistrée dans le système de gestion des équipes.
- Les communications sur le sujet du travail ne sont permises qu'à l'intérieur d'une même équipe.
- Tous les fichiers nécessaires pour reproduire votre analyse doivent être déposés dans la boîte de dépôt (voir la section Livrables).
- Le travail est noté sur **50 points**. La contribution du travail à votre note finale est telle qu'indiquée dans le plan de cours.
- Le plagiat est sanctionné par la note 0 et les mesures appropriées.
- N'oubliez pas de citer vos sources (s'il y a lieu) et d'accorder une importance particulière à la qualité de la langue et à la lisibilité.

Livrables

Ce travail pratique est constitué de deux livrables principaux :

- un rapport détaillé au format HTML, celui-ci devra être créé en utilisant R Markdown (voir les détails ci-dessous);
- l'ensemble des fichiers utilisés pour recréer votre analyse y compris les fichiers R Markdown.

Vous êtes notés à la fois sur les explications et sur le code. Vous devez appliquer de bonnes pratiques de programmation (code R le plus propre possible). Vous devez appliquer les bonnes pratiques pour l'analyse (même si ces pratiques ne sont pas toujours mentionnées sur l'énoncé). Par exemple:

- afficher un exemple de données dans le rapport;
- vérifier les valeurs manquantes après le chargement de données;
- vérifier les valeurs manquantes suite aux opérations de jointures de tables;
- revérifier les valeurs manquantes après leur traitement...

Rapport détaillé

Le travail pratique a plusieurs questions et chaque question pourrait avoir plusieurs sous-questions. Les questions sont indépendantes dans le sens où elles portent sur des données différentes.

Vous devez utiliser un R Markdown par question de façon à produire plusieurs petits rapports au format HTML (un par question). Notez que toutes les sous-questions se rapportant à une même (grande) question devraient être dans le même R Markdown et dans le même fichier HTML une fois le R Markdown compilé.

Chaque fichier HTML qui correspond au rapport devra avoir un entête contenant:

- le titre contenant le numéro de votre équipe, le numéro du travail pratique et le numéro de la question;
- le nom de chacun des membres de votre équipe et leur numéro matricule entre parenthèses;
- la date.

Cet entête doit être généré via l'entête YAML du format R Markdown.

Le rapport, pour chaque (grande) question, doit contenir une section par sous-question. Pour chaque question et sous-question, détaillez et justifiez la réponse proposée par votre équipe. L'absence de justification pour une sous-question entraînera des pénalités sur la note finale du travail.

L'utilisation de R Markdown est évaluée dans ce travail de sorte que des pénalités sont attribuées pour la non-utilisation ou la mauvaise utilisation de R Markdown.

Fichiers R Markdown et autres fichiers

Vous devez fournir tous les fichiers nécessaires pour permettre de reproduire votre analyse pour chacune des questions et des sous-questions.

Vous serez pénalisés pour les fichiers manquants puisque la note de 0 sera attribuée à une sous-question pour laquelle nous ne sommes pas en mesure de reproduire votre analyse. Ceci peut arriver pour différentes raisons, les plus fréquentes sont:

- il manque un ou plusieurs fichiers de données ou de code;
- il y a une erreur dans le code qui crée un bogue.

S'il y a un aspect aléatoire à votre analyse (peu importe la raison), SVP le mentionner. Le cas échéant, nous en tiendrons compte lors de nos tentatives de reproduction de votre analyse.

Notez que les fichiers de données que nous avons distribués pourraient être remplacés par les originaux pour tester votre code. Il ne faut pas changer le nom de ces fichiers lorsque vous les utilisez dans votre code.

Nous tenterons d'exécuter le code du R Markdown dans la séquence qu'il apparaît dans RStudio et aussi de compiler le R Markdown en HTML. Le code doit fonctionner et le R Markdown doit compiler dans cette séquence. Testez sur un environnement R vide.

Veuillez suivre la convention "EQn_TPz_Qx_nomFichier.ext" pour nommer vos fichiers où:

- EQ est la chaîne de caractères EQ suivie du numéro de votre équipe n;
- TP est la chaîne de caractères TP suivie du numéro de TP z;
- Q est le caractère Q suivi du numéro de question x;
- nomFichier peut être remplacé par un nom qui décrit sommairement le contenu du fichier, par exemple, le numéro de sous-question si c'est un fichier relié à une sous-question;
- ext est l'extension du fichier (p. ex., Rmd pour un R Markdown, csv pour un fichier CSV).

Voici un exemple de fichier nommé correctement pour le R Markdown de l'équipe 3 pour la question 2 du TP 12: "EQ3_TP12_Q2_analyse.Rmd". Pour la sous-question 1 de cette même question, l'équipe a produit un fichier de données sur les courges au format CSV. Elle l'a nommé correctement: "EQ3_TP12_Q2_1dataCourges.csv".

Des points seront retirés si la convention n'est pas respectée ou s'il y a confusion dans les fichiers.

Sur l'utilisation de R

Attention! Si vous travaillez avec la commande *setwd* dans un Rmd, c'est le moment de vous en départir. Pour que le répertoire de travail soit le répertoire courant du fichier Rmd, vous pouvez créer un fichier Rproj par question. Ensuite, ouvrez d'abord le Rproj, puis le Rmd. Ceci facilitera l'importation de données à partir d'un chemin relatif, c'est-à-dire, un chemin à partir du répertoire courant. La commande *setwd* n'a pas besoin d'être utilisée et provoque généralement des erreurs étranges.

Si votre programme génère des fichiers, il ne doit pas écraser les fichiers distribués pour une question donnée.

De plus, les fichiers de données fournis (le cas échéant) ne doivent pas être modifiés directement (pas de tri de données dans Excel, pas de changement manuel ou à l'aide de code au contenu des fichiers du sous-répertoire *data*). Ceci est vrai pour tous les travaux dans le cadre de ce cours: il convient d'éviter les traitements manuels. Si des modifications doivent être faites aux données, il faut les faire avec notre outil: R. Celui-ci permet d'automatiser le traitement de données.

Archive et structure du répertoire

Pour remettre le travail, vous devrez créer une archive zip (dossier compressé).

Placez les fichiers de chaque (grande) question i dans leur propre répertoire (aussi nommé dossier) nommé Qi. Placez tous les répertoires contenant les questions dans un répertoire nommé "EQn_TPz" où n est le numéro de votre équipe et z est le numéro du TP. Placez vos HTML compilés à la racine du répertoire. Faites le ménage des fichiers qui ne sont pas nécessaires (les retirer). La Figure 1 présente un exemple de répertoire bien structuré et de l'archive résultant de la compression de ce répertoire. Le nom du répertoire parent est

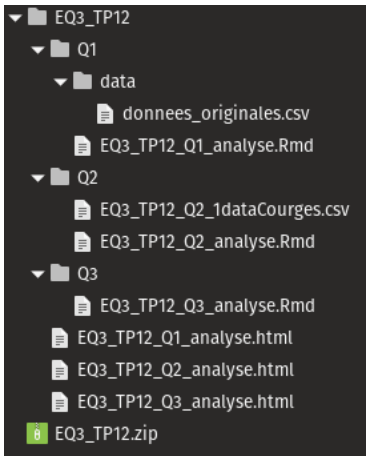

Figure 1: Exemple de répertoire bien structuré et de l'archive résultant de la compression

EQ3_TP12. Dans cet exemple, le travail pratique 12 avait 3 questions. L'équipe 3 a donc fait un répertoire par question: Q1, Q2, Q3. Dans chacun de ces répertoires, un fichier R Markdown a été créé. On remarque que chaque R Markdown a un HTML correspondant qui a été compilé par l'équipe et placé à la racine du répertoire EQ3_TP12. Chose intéressante, on remarque que des données étaient distribuées avec la question 1. Celles-ci sont disponibles dans l'archive. On remarque aussi que, pour la question 2.1, l'équipe a généré un fichier CSV nommé "EQ3_TP12_Q2_1dataCourges.csv".

Les fichiers HTML, c'est ce que nous lirons pour comprendre votre analyse. Les fichiers R Markdown, c'est ce que nous exécuterons puis compilerons en document HTML pour savoir si votre code fonctionne.

Je vous conseille de bien structurer vos répertoires dès le départ et de faire le ménage à la fin du travail pratique avant de créer l'archive. Pour faire le ménage et faire l'archive, travaillez sur une copie du répertoire (ainsi vous pourrez revenir en arrière si vous effacez quelque chose de trop). Ensuite, créez votre archive. Pour vous assurer que tout fonctionne, décompressez votre archive dans un répertoire temporaire et retestez vos fichiers R Markdown et vos scripts à partir d'un environnement R vide.

Note finale

Évitez d'utiliser des caractères spéciaux pour nommer vos fichiers et vos répertoires. Utilisez des lettres sans accents, des chiffres et le caractère _ (appelé underscore ou caractère de soulignement) pour remplacer les espaces.

Prenez bien le temps de lire l'énoncé. Notez que certaines questions sont plus courtes que d'autres.

Pour terminer, j'offre un remerciement spécial aux auxiliaires de MQT-6021 pour leur contribution à l'élaboration de ce TP!

Bon travail!

Michael Morin

1 Persistance des données (50 points)

Mise en contexte: Vous effectuez une recherche sur les compagnies aux États-Unis et au Canada. Votre superviseur a trouvé des données sur des compagnies aux États-Unis. Cette question porte sur ces données.

Tâche: Votre superviseur souhaite avoir une base de données (BD) relationnelle contenant toutes les données qu'il a "découvert" par lui-même. Ces données sont dans quatre fichiers:

- `Q1_categories.csv`
- `Q1_companies.csv`
- `Q1_locations.csv`
- `Q1_types.csv`

Il vous demande de créer une BD SQLite intègre à partir de ces fichiers. Il veut absolument que votre BD soit intègre, mais vous n'avez pas à en assurer la cohérence pour l'instant. En d'autres termes, vous devez respecter l'intégrité référentielle et l'intégrité d'entité (incluant des clés primaires uniques), mais vous n'avez pas à vous assurer que les données sont "correctes," p. ex., si la même chose est nommée de deux façons différentes dans votre BD, vous pouvez le tolérer (à part si ça contrevient à l'intégrité).

Après avoir observé rapidement les données, vous griffonnez le schéma de la Figure 2 qui représente la BD finale telle que vous envisagez de la faire. Dans ce schéma, les clés primaires sont en gras et les clés étrangères sont en italique. Le schéma représente aussi les relations que nous devrions avoir entre les tables une fois la BD construite. Finalement, certains champs représentés ne sont pas dans les données et devront être créés pour favoriser l'intégrité.

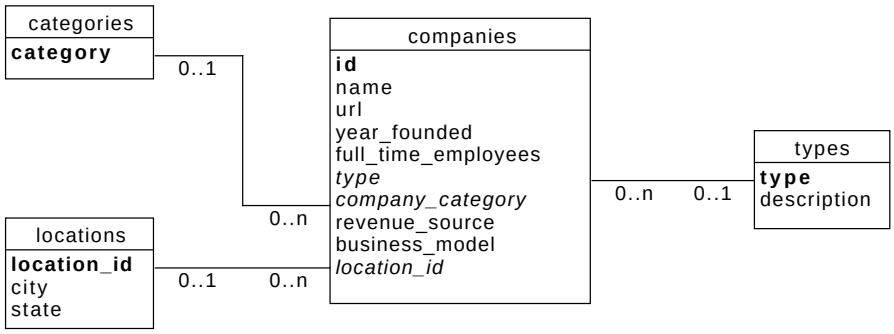

Figure 2: Diagramme entité-association de la BD

1.1 Base de données intègre: ménage pour rendre le tout intègre

Chargez les données des fichiers dans des tables de données R avec `read_csv`:

- `Q1_categories.csv` dans `data_cat`
- `Q1_companies.csv` dans `data_com`
- `Q1_locations.csv` dans `data_loc`
- `Q1_types.csv` dans `data_typ`

Utilisez les noms mentionnés pour conserver vos données.

Pour cette sous-question, votre tâche consiste à vous assurer que les données contenues dans `data_cat`, `data_com`, `data_loc` et `data_type` respectent le schéma de la Figure 2. Assurez-vous aussi que ces données relationnelles sont intègres. Pour ce faire, inspirez-vous du Tutoriel 11a.

Vous créerez la base de données plus tard, aux autres sous-questions.

1.2 Base de données: créez votre base de données

Une fois que vos données sont prêtes, créez la base de données SQLite en utilisant du code R. Vous devez stocker le contenu de `data_cat`, `data_com`, `data_loc` et `data_type` dans leur table respective nommée ainsi:

- `data_cat` va dans la table *categories*
- `data_com` va dans la table *companies*
- `data_loc` va dans la table *locations*
- `data_typ` va dansd la table *types*

À cette étape, ne créez que la BD. La configuration de la BD dans SQLite est demandée à la sous-question suivante.

1.3 Base de données: configurez la base de données

Dans DB Browser for SQLite, ajustez les relations entre les tables (clés étrangères). Assurez-vous aussi que les champs qui sont des clés primaires sont identifiés comme tels et qu'ils ne sont pas nuls. Assurez-vous aussi de préciser les relations entre vos tables à l'aide de la définition des clés étrangères.

Une fois que c'est fait, n'oubliez pas d'utiliser le bouton "Write Changes" dans DB Browser for SQLite pour sauvegarder les changements apportés à votre BD.

1.4 Requêtes sur votre nouvelle base de données

Dans R, faites les requêtes suivantes:

- toutes les compagnies (noms et url) fondées entre 1999 et 2001 ordonnées par nom;
- toutes les compagnies (noms, url, ville et année de fondation) en Orégon.

Affichez les résultats dans le rapport. Affichez aussi le nombre d'enregistrements retournés par chacune des requêtes.