

TP3 (version Python)

INF8808: Visualisation de données

Département de génie informatique et génie logiciel



POLYTECHNIQUE MONTREAL

Auteure: Olivia Gélinas

Chargé de lab: Hellen Vasques et Olivia Gélinas

Objectifs

L'objectif de ce travail pratique est de créer une carte de chaleur interactive en utilisant des données ouvertes en format CSV.

Avant de commencer, nous vous recommandons d'avoir effectué les lectures suivantes et d'avoir effectué les exercices suivants:

- Plotly Express

<https://plotly.com/python/plotly-express/>

- Heatmaps

Lectures :

<https://plotly.com/python/heatmaps/>

- Line Charts

<https://plotly.com/python/line-charts/>

Exercices : TP3 exercices : 1, 2, 3, 4, 5

Introduction

Une carte de chaleur (*heatmap*) est un type de visualisation de données qui affiche les données sous forme de couleurs représentées selon deux dimensions. La teinte ou intensité de chaque couleur représente la l'intensité de l'observation pour ses valeurs correspondantes en 2 dimensions.

Dans ce travail pratique, vous implémenterez une carte de chaleur interactive en utilisant des données représentant les plantations d'arbres au fil du temps dans les quartiers de Montréal. Les données ont été extraites du portail de données ouvertes de la Ville de Montréal [1]. L'ensemble de données a ensuite été modifié en une version à utiliser dans ce travail pratique. Il contient des données représentant la date et le lieu où les arbres ont été plantés à Montréal.

Pour explorer l'utilisation des tableaux de bord, nous programmerons également un graphique linéaire pour accompagner la carte de chaleur. Un graphique linéaire est un type de graphique dans lequel les données sont représentées par une ou plusieurs lignes. Il est extrêmement commun et peuvent être trouvés dans de nombreux domaines où nous aimerions voir l'évolution d'un paramètre en fonction d'un autre. Dans ce travail pratique, les données présentées sur le graphique linéaire dépendront de la sélection actuelle de la carte de chaleur.

Description

Dans ce travail pratique, vous devrez compléter le code Python à l'aide de Plotly et Dash afin d'afficher une carte de chaleur accompagnée d'un graphique linéaire pour représenter les plantations d'arbres dans les quartiers de Montréal au fil du temps.

La carte de chaleur représente le nombre d'arbres plantés dans chaque quartier de Montréal chaque année entre 2010 et 2020. Les données affichées sur le graphique linéaire dépendent de la cellule sélectionnée dans la carte de chaleur. Lorsqu'une cellule de la carte de chaleur est sélectionnée, le graphique linéaire affiche les plantations d'arbres au fil du temps pour le quartier et l'année sélectionnés. Lorsqu'aucune cellule n'est sélectionnée dans la carte de chaleur ou lorsqu'il n'y a aucune donnée à afficher pour la sélection actuelle, la zone où est situé le graphique linéaire contient un message instructif.

Les sous-sections suivantes présentent les différentes parties que vous aurez à compléter pour ce travail pratique. Pendant que vous codez, nous vous recommandons de compléter le traitement des données en premier, suivi de la mise en œuvre de la carte de chaleur. Ensuite, vous devez implémenter le graphique linéaire. Les deux parties suivantes, le thème et l'info-bulle, sont indépendantes l'une de l'autre.

Structure des fichiers

Pour terminer ce travail, vous devrez remplir les différentes sections **TODO** dans les fichiers de l'archive fournie pour le travail pratique. Les commentaires dans le code expliquent plus en détail les étapes à suivre. Les scripts à utiliser sont situés dans le répertoire **assets** de l'archive fournie pour le travail pratique.

Dans ce travail pratique, nous vous fournissons 7 fichiers Python utilisés pour accomplir la visualisation souhaitée:

- **app.py**: Ce fichier génère la structure HTML de la page web et orchestre les étapes requises pour créer la visualisation. Vous n'avez pas besoin de le modifier.
- **heatmap.py**
- **hover_template.py**
- **line_chart.py**

- `preprocess.py`
- `server.py`: Ce fichier est utilisé pour lancer l'application. Vous n'avez pas besoin de modifier il.
- `template.py`

Données

L'ensemble de données se trouve dans le répertoire `src/assets/data/` dans l'archive fournie pour le travail pratique. L'ensemble de données contient les colonnes suivantes :

- `Arrond`: ID du quartier où l'arbre a été planté.
- `Arrond_Nom`: Le nom du quartier où l'arbre a été planté.
- `Date_Plantation`: La date à laquelle l'arbre a été planté.
- `Longitude`: La longitude de l'arbre.
- `Latitude`: La latitude de l'arbre.

Prétraitement des données

Pour commencer, vous devrez prétraiter les données que nous vous fournissons. Les données contenues dans le fichier CSV sont brutes, il est donc nécessaire de réorganiser certaines parties de celles-ci afin qu'elles puissent être correctement utilisées par Plotly. Pour ce faire, vous devez compléter le fichier `preprocess.py`.

Plus précisément, vous devrez effectuer ces étapes:

1. Convertissez les dates des données dans un format compréhensible par Plotly (fonction `convert_dates`)
2. Filtrez les données par année (fonction `filter_years`). Bien que nous ayons des données sur les arbres plantés à partir du début du 20e siècle, nous souhaitons nous concentrer sur les plantations d'arbres entre 2010 et 2020 pour cette visualisation.
3. Récapitulez les données pour obtenir le nombre total d'arbres plantés par an et par quartier (fonction `summary_yearly_counts`)
4. Regroupez et restructurez les données dans un format que Plotly peut facilement lire pour générer la carte de chaleur (fonction `restructure_df`)
5. Écrivez une fonction pour obtenir des données à afficher sur le graphique linéaire, correspondant à la quantité quotidienne d'arbres plantés dans un quartier pour une année donnée (fonction `get_daily_info`)

Pour vous guider dans ces étapes, les figures 1 et 2 montrent un extrait des données après le prétraitement.

La figure 1 illustre les données dans un format qui peut être affiché dans la carte de chaleur, tandis que la figure 2 échantillonne certaines données qui peuvent être affichées dans le graphique linéaire.

| Date_Plantation | 2010-12-31 | 2011-12-31 | 2012-12-31 | 2013-12-31 | ... | 2017-12-31 | 2018-12-31 | 2019-12-31 | 2020-12-31 |
|---------------------------------------|------------|------------|------------|------------|-----|------------|------------|------------|------------|
| Arrond_Nom | | | | | ... | | | | |
| Ahuntsic - Cartierville | 437.0 | 289.0 | 327.0 | 357.0 | ... | 936.0 | 815.0 | 851.0 | 144.0 |
| Côte-des-Neiges - Notre-Dame-de-Grâce | 401.0 | 108.0 | 458.0 | 395.0 | ... | 803.0 | 623.0 | 685.0 | 79.0 |
| LaSalle | 1.0 | 147.0 | 522.0 | 0.0 | ... | 336.0 | 48.0 | 1175.0 | 0.0 |

Figure 1: Exemple de données de carte de chaleur

| Date_Plantation | Counts |
|-----------------|------------|
| 0 | 2017-05-16 |
| 1 | 2017-05-17 |
| 2 | 2017-05-18 |
| 3 | 2017-05-19 |
| 4 | 2017-05-20 |

Figure 2: Exemple de données pour le graphique linéaire pour le Sud-Ouest en 2017

Carte de chaleur

Pour cette deuxième partie, vous devrez implémenter la partie principale de la visualisation de données. Vous devrez afficher les données que vous avez prétraitées à l'étape précédente dans la carte de chaleur. Pour compléter cette partie, vous devrez modifier le fichier `heatmap.py`.

Voici l'étape principale que vous devrez accomplir pour cette partie:

1. Générez la carte de chaleur à partir des données prétraitées et affichez-la (fonction `get_figure`)

Dans la figure 3, nous pouvons voir à quoi devrait ressembler la carte de chaleur finale.

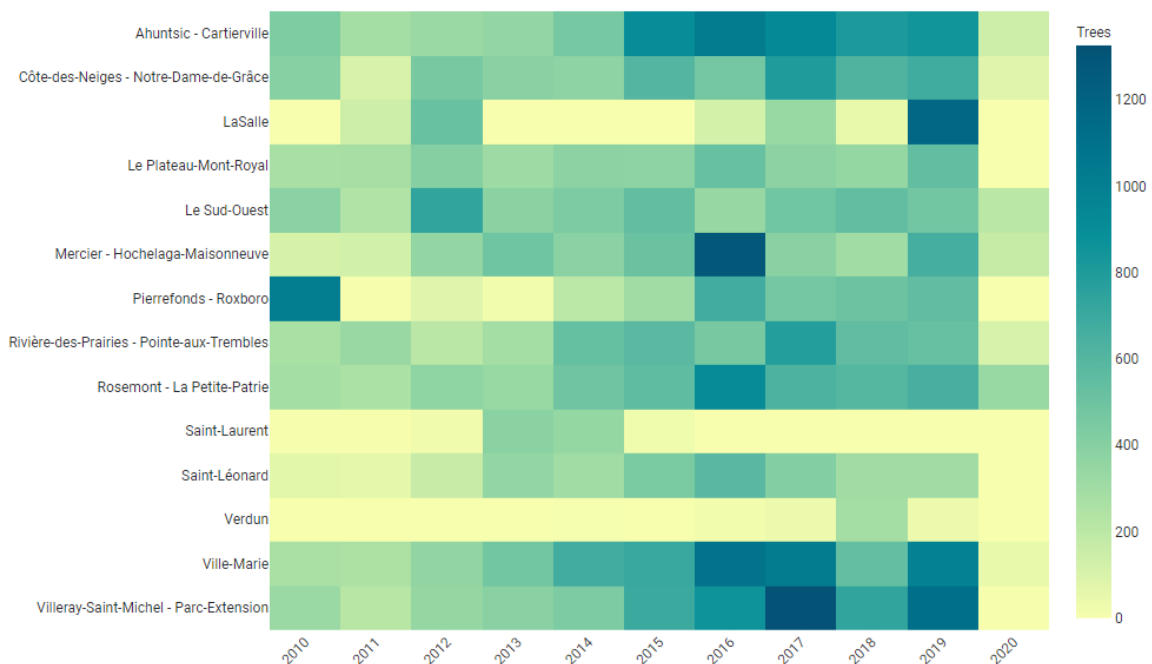


Figure 3: La carte de chaleur

Graphique linéaire

Pour cette troisième partie, vous devrez programmer le graphique linéaire qui accompagne la carte de chaleur. Tout d'abord, vous devrez implémenter la version vide du graphique linéaire, qui s'affiche lorsqu'aucune cellule n'a été sélectionnée ou lorsque la cellule sélectionnée n'a aucune donnée à afficher. Cette version vide du graphique linéaire doit contenir un message informatif et un rectangle gris en arrière-

plan. Par la suite, vous implémenterez le code pour afficher les données sur le graphique linéaire lorsqu'une cellule est sélectionnée. Lorsqu'il n'y a que des données d'un jour à afficher, assurez-vous de changer le mode d'affichage du graphique linéaire pour être sûr que seul le point unique est visible.

Plus précisément, vous devrez effectuer ces étapes:

1. Ecrivez la fonction pour afficher une figure vide lorsqu'il n'y a pas de données à afficher (fonction `get_empty_figure`)
2. Écrivez la fonction pour dessiner un rectangle gris sur la figure vide quand il n'y a pas de données à afficher (fonction `add_rectangle_shape`)
3. Lorsqu'il y a des données à afficher, générez le graphique linéaire avec les données correspondantes pour le quartier et l'année donnés (fonction `get_figure`)

Les figures suivantes donnent quelques précisions supplémentaires sur ce qui est attendu pour cette partie. Les figures 4, 5 et 6 servent d'indication pour l'aspect visuel final et le comportement des graphiques.

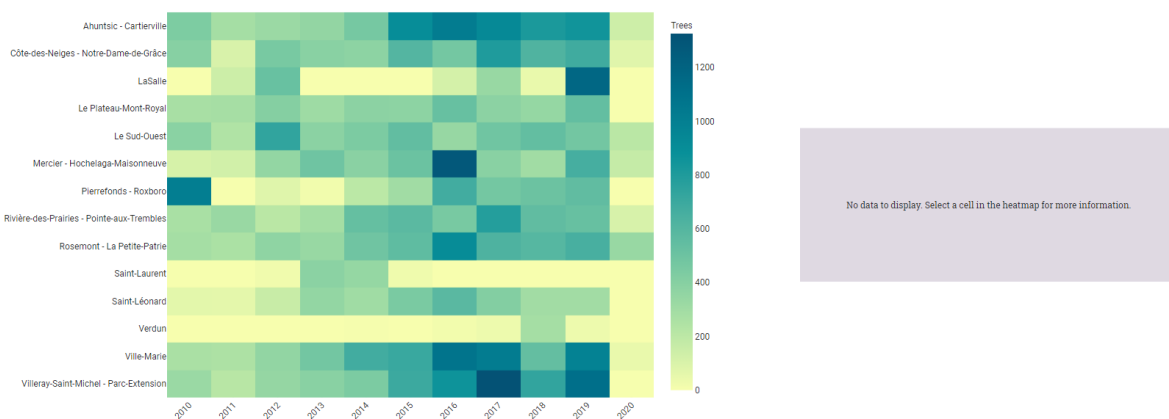


Figure 4: La carte de chaleur avec la figure vide d'accompagnement

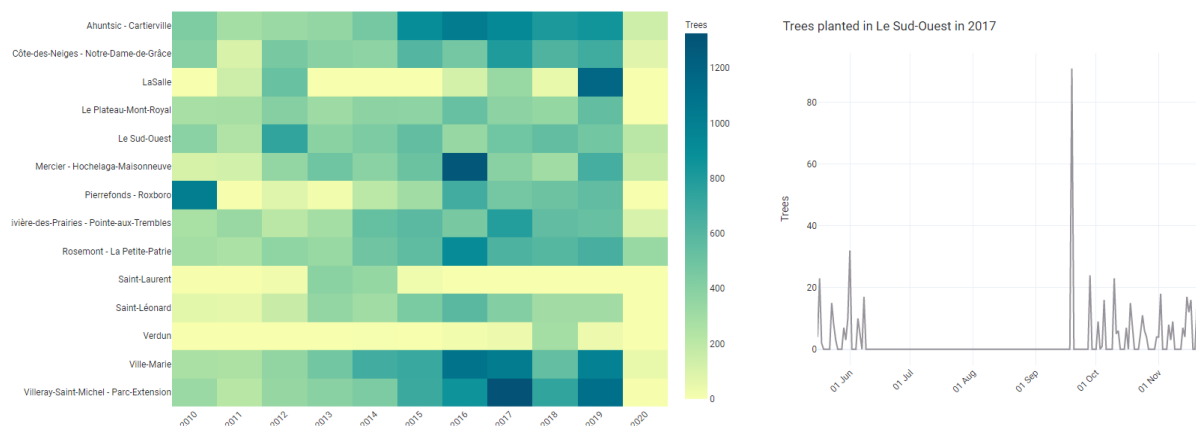


Figure 5: La carte de chaleur accompagnée d'un graphique linéaire représentant les arbres plantés quotidiennement dans le Sud Ouest en 2017

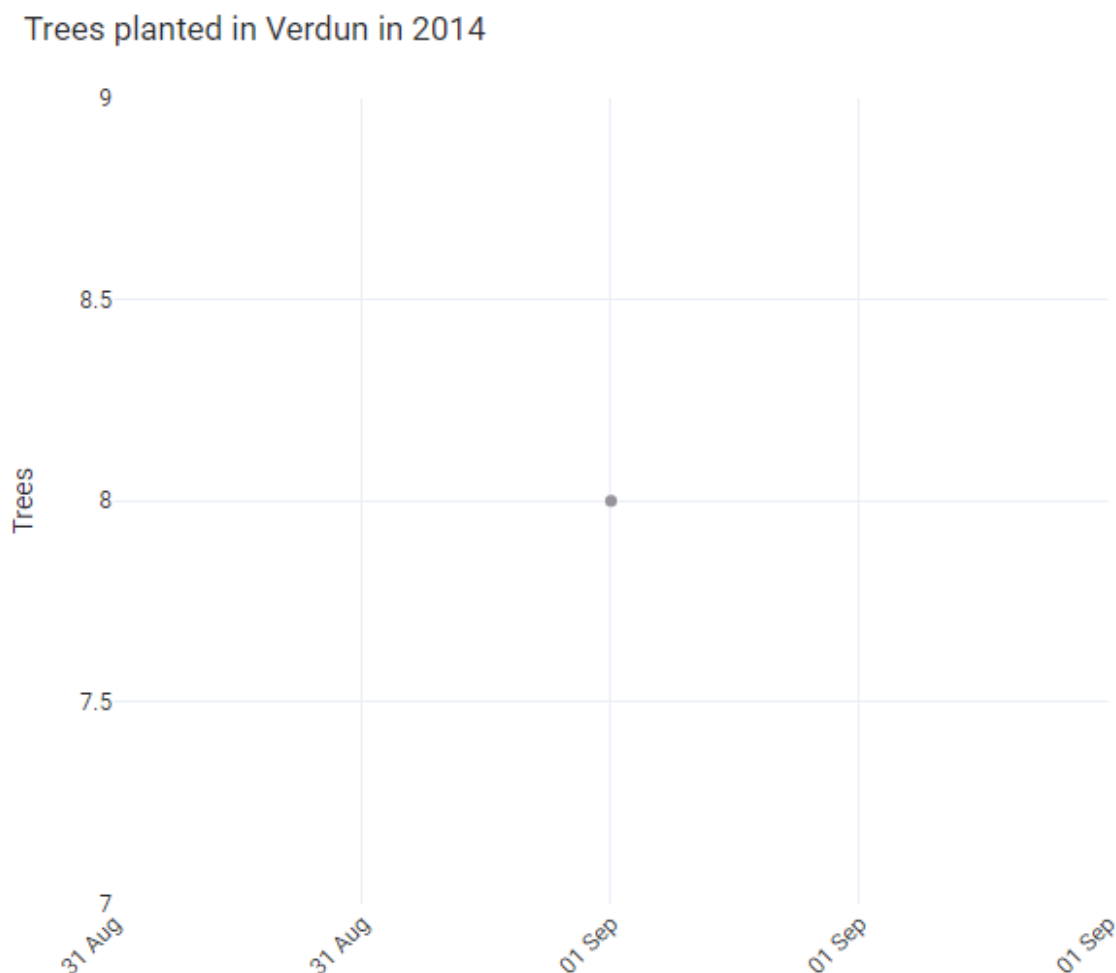


Figure 6: Le graphique linéaire lorsqu'il n'y a que des données pour un jour à afficher (ici, les données des arbres quotidiens plantés à Verdun en 2014)

Thème

Pour cette quatrième partie, vous allez créer un thème pour personnaliser l'affichage visuel du graphique. Pour l'utiliser, vous le définissez comme thème par défaut, appliqué au-dessus du thème `plotly_white` fourni par Plotly. Le code de cette partie est dans le fichier `template.py`. Vous devez remplir les fonctions `create_custom_theme` et `set_default_theme`. Assurez-vous de suivre attentivement les instructions dans les commentaires du code lors de la définition de chaque élément du thème. Les valeurs souhaitées à utiliser sont définies dans une variable au début du fichier. Assurez-vous que les couleurs dans la carte de chaleur et le graphique linéaire ressemblent à celles des figures 4, 5 et 6.

Info-bulle

Pour cette cinquième partie, vous définirez un modèle (*template*) permettant d'afficher l'info-bulle qui apparaît lorsque la souris survole une cellule de la carte de chaleur ou du graphique linéaire. Dans la carte de chaleur, l'info-bulle d'une cellule doit contenir le nom du quartier, l'année et la quantité d'arbres plantés cette année-là dans le quartier donné. Dans le graphique linéaire, l'info-bulle doit contenir la date et la quantité d'arbres plantés ce jour-là dans le quartier donné. Les commentaires dans le code donnent une description détaillée du contenu et de l'apparence de l'info-bulle. Le code de cette partie doit être écrit dans le fichier `hover_template.py` dans la fonction `get_hover_template`.

Les étapes à suivre pour compléter cette partie sont les suivantes:

1. Complétez le modèle (*template*) de survol de la carte de chaleur (fonction `get_heatmap_hover_template`)
2. Complétez le modèle (*template*) de survol du graphique linéaire (fonction `get_linechart_hover_template`)

Voir les figures 7 et 8 pour les résultats finaux attendus.

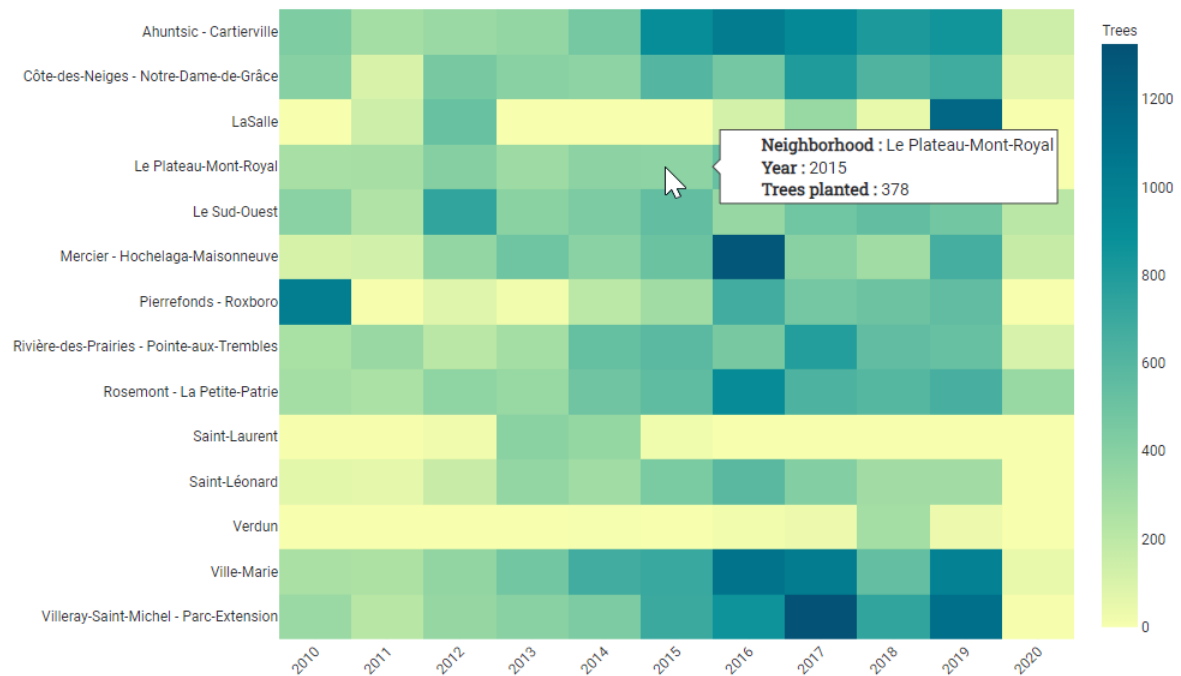


Figure 7: L'info-bulle de la carte de chaleur du Plateau Mont-Royal en 2015

Trees planted in Le Plateau-Mont-Royal in 2015

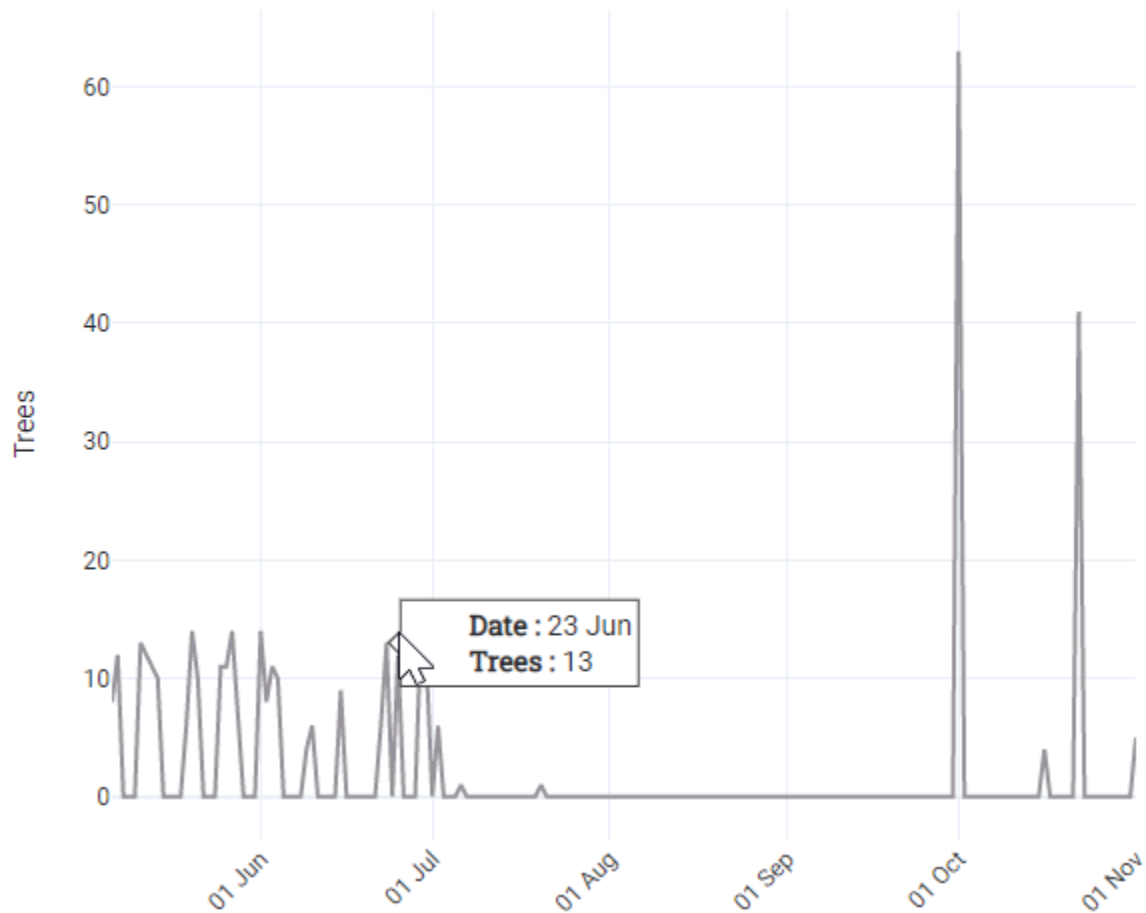


Figure 8: L'info-bulle du graphique linéaire du Plateau Mont-Royal en 2015.

Soumission

Les instructions pour la soumission sont:

1. Vous devez placer le code de votre projet dans un fichier ZIP compressé nommé **matricule1_matricule2_matricule3.zip**
2. Le travail pratique doit être soumis avant le **27 Février 23h59**.

Évaluation

Dans l'ensemble, votre travail sera évalué selon la grille suivante. Chaque section sera évaluée sur l'exactitude et la qualité du travail.

| Exigence | Points |
|---------------------------|--------|
| Prétraitement des données | 5 |
| Carte de chaleur | 3 |
| Graphique linéaire | 6 |

| Exigence | Points |
|--|-----------|
| Thème | 2 |
| Info-bulle | 3 |
| Qualité globale et clarté de la soumission | 1 |
| Total | 20 |

Références

[1] Service des grands parcs, du Mont-Royal et des sports, "arbres sur le territoire de la Ville ," Montréal : Portail de données ouvertes. Available: <http://donnees.ville.montreal.qc.ca/dataset/arbres>[Accessed 01 09 2020].