



18 mars 2021



ÉTAT D'AVANCEMENT

[Projet Informatique Individuel]

PERRIER ALBAN

BORDEAUX INP

ENSC 2A – Groupe 1

TRAVAIL RÉALISÉ

L'ensemble de mon code est disponible sur ce dépôt : <https://github.com/aperrier004/PII>

Tout d'abord, mes objectifs pour cette première partie étaient d'appréhender la réalisation d'une interface Python, ainsi que d'arriver à une application fonctionnelle qui permet l'affichage de cartes de chaleurs ainsi que de graphiques de points.

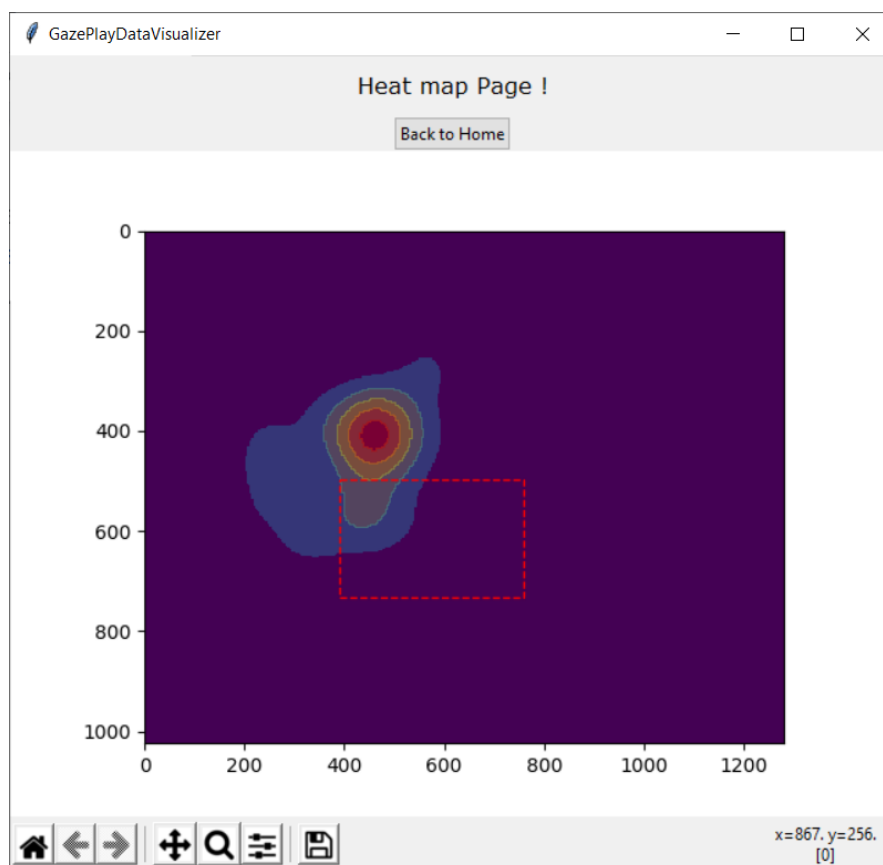
Pour commencer, j'ai d'abord eu à déterminer la librairie Python avec laquelle j'allais travailler pour l'affichage de graphiques.

J'ai choisi la librairie PyTrack (<https://pytrack-ntu.readthedocs.io>), qui permettait une analyse du regard telle que je la souhaitais, qui était libre de droit et relativement maintenue à jour. Ces trois critères étaient également des attendus de M.SCHWAB, qui encadre en même temps mon projet. Cette librairie a donc été choisie pour ces raisons, mais également pour des raisons d'environnement Python. En effet, au fur et à mesure que je trouvais des librairies, je souhaitais les tester. Cependant, n'ayant au départ aucune expérience avec Python, j'ai rencontré des difficultés à l'installation ou à la mise en place de mon environnement de travail. J'ai aussi eu des problèmes de versions pour certains packages, ce qui a été éliminatoire pour certaines librairies (dans le cas où elles n'étaient plus maintenues et présentaient des incompatibilités avec des packages qu'elles utilisent et ayant changé de versions).

Cette étape de mise en place de l'environnement terminée et le choix de la librairie validée, j'ai pu m'atteler à sa pratique avec des données d'exemples. Toutefois, cela n'a pas forcément été évident puisque les fonctions de la librairie n'étaient pas forcément bien implémentées ou documentées. En effet, j'ai eu à modifier le code source de la librairie PyTrack afin de la rendre utilisable sur mon ordinateur. De plus, les exemples d'utilisation de la documentation n'étaient pas toujours très clairs selon les cas d'usages que je voulais essayer.

Une nouvelle étape dans mon projet fut de réussir à afficher différentes fenêtres de graphique avec des données d'exemples.

Cela fait, j'ai pu m'atteler à la création d'une interface primaire. J'ai donc suivi un rapide tutoriel d'interfaces Tkinter afin de réaliser une petite fenêtre permettant de naviguer à travers différents écrans, pour afficher des graphiques et avoir quelques éléments de contrôles. Après cela, j'ai souhaité insérer les graphiques que j'avais obtenus avec la librairie directement dans mon interface, or, cela continuait de les afficher dans des fenêtres séparées. Comportement prévisible puisque les fonctions de création de ces graphiques créent à l'intérieur d'elles-mêmes la fenêtre et la figure à y insérer. Suite aux conseils de mon tuteur, j'ai donc décidé de dupliquer ces fonctions de créations de cartes de chaleurs et graphiques de points. J'ai alors pu les modifier comme je le souhaitais, en faisant en sorte qu'elles me renvoient une figure, sans qu'elles créent une fenêtre. Ainsi, j'ai pu ajouter les graphiques à mon interface.



Exemple de mon interface avec une carte de chaleur

Ce point fut particulièrement épineux puisque je n'avais pas vraiment d'expérience avec le parsing. Après de maintes comparaisons entre les fichiers d'exemples d'entrées (au format TXT, ASC,...) et les fichiers de sorties au format CSV produit par les fonctions de générations de données compatibles de ma librairie, j'ai pu comprendre quelles données étaient utiles et récupérées. Ensuite, j'ai appréhendé un fichier JSON de GazePlay, en déterminant quels champs de données je devais récupérer pour obtenir un fichier CSV complet. À partir de là, j'ai pu créer mes propres fonctions de lectures de fichiers JSON et parser les données pour les envoyer à une fonction qui s'occupe de créer le fichier au format CSV.

Exemple de données extraites du fichier JSON de GazePlay

Exemple de lignes de mon fichier CSV avec des données parser du fichier JSON

3

TRAVAIL RESTANT ET À AMÉLIORER

Pour la deuxième partie du projet, je vais d'abord commencer par résoudre mon problème d'affichage de données venant de mon fichier JSON dans l'interface. Puis, je m'attellerai à améliorer l'interface et les différents écrans pour avoir une meilleure expérience utilisateur et avoir un rendu plus utilisable. Notamment en ajoutant la possibilité de donner directement le fichier JSON depuis l'interface, plutôt que d'avoir à écrire le chemin dans le code directement comme fait actuellement.

Ces améliorations effectuées, je souhaiterais pouvoir rediscuter de la suite du projet avec M.SCHWAB afin qu'il me fasse un retour sur mes récentes avancées, et que l'on puisse déterminer des objectifs futurs à atteindre. En effet, comme spécifié dans mon cahier des charges et mon planning, la deuxième partie de mon projet est censé se consacrer à des fonctionnalités supplémentaires.

Cela me permet notamment d'être serein quant aux améliorations à réaliser sur mon application actuelle, qui devraient parfaitement rentrer dans mon planning sans pour autant accuser de retard.

Pour rappel, voici mon planning :

