

Cette API avait pour but d'être déployé à l'aide de d'un conteneur Docker et puis d'être appelé à l'intérieur de l'application dash et récupérer les prédiction calculé depuis le modèle présent dans l'API. Par manque de temps il n'a pas été possible de faire le lien entre l'API et l'application. Cependant tout les fichier disponible permettent de crée une API déployé localement ainsi que crée une image docker pour par la suite l'implanter dans un conteneur.

Les fichier nécessaire au bon fonctionnement sont :

-requirement.txt : liste de toutes les Library nécessaire.

-regressor_global.plk : modèle de prédiction de Valeur foncière.

-Dockerfile : Document qui va permettre la création de l'image et le lancement de l'API lors du lancement du conteneur contenant l'image.

-app.py : fichier python contenant l'API

Dans le dossier templates :

-help.html

-help_en.html

-help_fr.html

-index.html

-prediction.html

Il est nécessaire d'utiliser la version 3.9.11 de python.

Pour lancer l'API en local il vous suffit (si tout les fichier sont bien présent dans le même répertoire que app.py) de lancer app.py. Rendez vous sur l'URL donné. Vous pouvez importer un fichier (csv, txt ou xlsx) contenant des données (voir « Help » pour question relative au format des données). Une fois fait le bouton prédire permet de lancer l'algorithme sur vos données. Vos prédiction sont téléchargeable au format csv avec « Exporter les prédictions ».

Pour générer une image docker, avec une invite de commande mettez-vous sur le dossier contenant tout vos fichiers. Puis lancer « docker build -t nom_de_votre_image . ». Cette commande va permettre la création de l'image en lisant le fichier Dockerfile. Cette opération peut prendre quelques minutes (modèle volumineux). Une fois votre image crée vous pouvez activer un conteneur avec votre image « docker run -p 5000 :5000 nom_de_votre_image ». Une fois votre conteneur activé, récupérer l'id du conteneur avec la commande « docker ps ». Puis optenez l'Adresse IP du conteneur avec la commande « docker inspect -f '{{range.NetworkSettings.Networks}}{{.IPAddress}}{{end}}' id_du_conteneur »

Une fois fait accédez à l'API. Si l'IP est 172.17.0.2 et que l'API est en écoute sur le port 5000, l'URL sera : **http://172.17.0.2:5000.**