

PYTHON PROJECT

Cette note explicative a pour but de détailler l'ensemble de notre projet. Nous reviendrons notamment sur son champ d'application, sur les objectifs poursuivis et sur l'utilité de l'outil développé. Plus globalement, nous expliquerons le fonctionnement de l'outil, des différents logiciels et packages utilisés et nous commenterons les différents choix que nous avons fait. Dans une note de synthèse, nous expliciterons les limites de notre outil et les potentiels développements supplémentaires que nous aurions pu mettre en œuvre.

Réalisé par :

SEZESTRE Emilien

LAMON Océane

PEDROT Emma

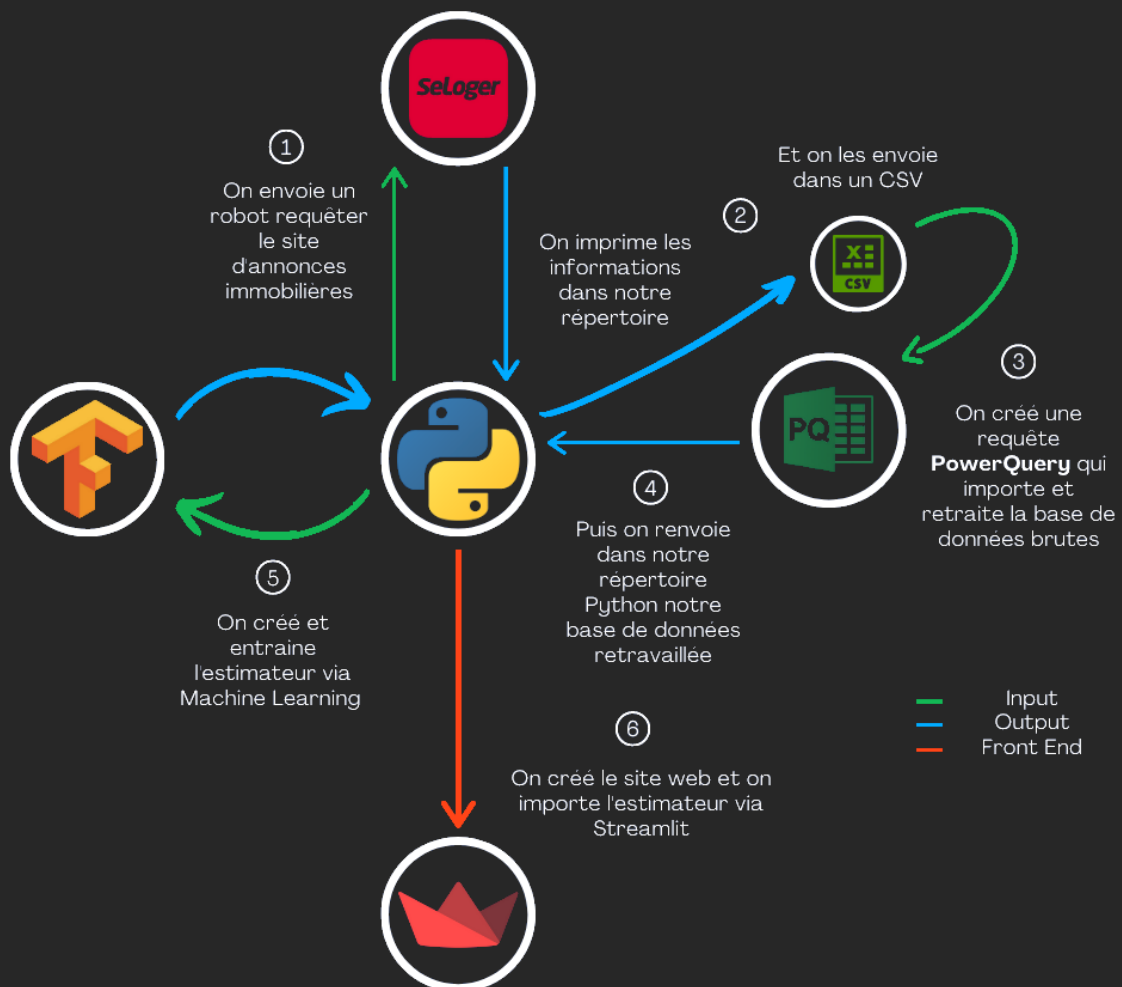
Professeur :

GAILLOTY Axel-Cleris

Sommaire

I.	Introduction	3
II.	Déterminant du prix d'une maison.....	4
III.	Champs de notre étude	5
IV.	Extraction et traitement des données	7
a.	Section 1	7
b.	Section 2.....	7
c.	Section 3.....	8
d.	Section 4.....	8
e.	Section 5.....	9
f.	Section 6.....	10
V.	Construction de l'estimateur via Machine Learning	12
VI.	Partie Front-End	13
a.	Seuils sur les saisies chiffrées	14
b.	Recueil pertinent des distances	15
VII.	Conclusion, Limites et ouvertures.....	16
VIII.	Bibliographie.....	17
IX.	Annexes.....	18

Infographie :



I. Introduction

L'acquisition d'un bien immobilier est perçue par le plus grand nombre comme une finalité. Gage de stabilité et de contrôle dans son espace de vie, il s'agit d'un investissement qui permet de se construire un actif à long terme. La possession d'un bien immobilier présente également une sécurité financière une fois l'hypothèque remboursée, et offre la possibilité de léguer cet investissement à ses descendants voire d'être source d'une potentielle plus-value à la revente sous certaines circonstances favorables.

Néanmoins, la décision de devenir propriétaire ou de vendre son bien ne doit pas se faire sans avoir au préalable étudié les caractéristiques et évolutions du marché de l'immobilier. Ces investissements représentent en effet des sommes conséquentes pour lesquelles les individus s'endettent pendant plusieurs décennies, il est donc nécessaire de s'intéresser de plus près à la conjoncture actuelle du marché pour estimer le bon moment avant d'acheter ou de vendre. Ainsi, un marché en tension fera le bonheur des propriétaires cherchant à vendre dans la mesure où les prix seront artificiellement gonflés par rapport à la réelle valeur des biens, tandis que de potentiels acheteurs préféreront attendre que l'écart entre offre et demande se resserre pour investir.

En définitive, il est intéressant de se pencher sur ce qui fait la valeur d'un bien immobilier. Cela peut permettre aux vendeurs de se faire une idée du prix de leur bien compte tenu de ses caractéristiques et de fixer ce dernier en conséquence, tandis que les acheteurs peuvent confronter leur projet avec leur budget d'achat et revoir leurs attentes à la hausse ou à la baisse en fonction des prix pratiqués sur le marché.

Notre objectif sera donc de construire un estimateur de prix pour des individus souhaitant acheter ou vendre un bien immobilier. Pour ce faire, nous allons dans un premier temps explorer les déterminants du prix d'un bien immobilier pour en extraire les indicateurs pertinents à prendre en compte. Par la suite, nous confronteront les pré-requis en termes de données nécessaires ainsi obtenus avec les sources d'informations disponibles pour délimiter notre champ d'action. Par la suite, le développement de notre outil se fera en trois phases : extraction des données via Web-Scraping et retraitement avec Power-Query pour nettoyer et standardiser ces dernières, création d'un modèle de machine learning pour l'estimation du prix d'un bien en fonction de caractéristiques données, puis mise en place d'une interface homme-machine sur laquelle les utilisateurs pourront réaliser leurs estimations de façon interactive et intuitive. Enfin, nous reviendrons sur notre cahier des charges initial pour conclure quant à notre réalisation et exposer les limites de notre projet.

II. Déterminant du prix d'une maison

Cette partie aura pour but de lister les éléments influençant le prix d'un bien immobilier. Pour cela nous sommes notamment servis du site "Zillow", qui est le site concernant le marché immobilier le plus visité des Etats-Unis [1]. Ainsi, l'article que nous allons utiliser ici se nomme "Valuing a House: What is it really Worth"[2] qui revient sur les critères principaux utilisés pour mesurer le prix d'une maison. Pour chaque critère nous estimons si l'information est trouvable et implémentable dans notre estimateur.

Ainsi, le premier critère est le prix du marché, c'est-à-dire le prix moyen auquel se vendent et s'achètent les biens dans une période et un marché donné. Ce prix varie en fonction de différents facteurs, comme la quantité de bien demandé par rapport à l'offre (en France, le marché est en tension : trop de demande pour pas assez d'offres, ce qui fait gonfler le prix). Ainsi, en fonction du moment que l'on choisit pour vendre un bien immobilier la structure de marché peut varier (liées à des changements d'ordres macro (exemple : hausse des taux d'intérêt, quantité de biens disponible, période de l'année)).

Ce facteur est aisément implémentable dans notre modèle car à l'aide de notre méthode de récupération des données (web-scraping), nous aurons une image relativement représentative du marché à l'instant T.

Le second critère qui détermine le prix d'un bien sont d'ordre matériel, nous avons listé ci-dessous, une liste d'éléments faisant gonfler ou baisser le prix d'un logement :

- Localisation : Selon le quartier ou la proximité avec le centre urbains, les transports en commun, ou avec une zone industrielle/zone bruyante, la valeur du logement en sera grandement influencée.
- La taille : Toute chose égale par ailleurs, plus une maison est grande, plus elle coûte cher, on pourra donc prendre en compte par exemple : Nombre de chambre, nombre de salle de bain, nombre d'étage, nombre de m² habitable, la taille du jardin, s'il y a un garage, une piscine, un sous-sol et des étages.
- Caractéristiques stylistiques : en fonction du degré de modernisation ou de finition du bâtiment
- L'âge : En fonction de l'âge d'un bien immobilier, la valeur changera également car cela implique potentiellement des coûts de rénovation supplémentaires, les nouvelles maisons sont également dotées de plus de choses (matériaux de construction, système de chauffage, isolation...) pouvant faire augmenter le prix d'une maison.
- La taxe foncière qui est un impôt local (le taux varie selon le département [3]), dont la base imposable est calculée en prenant en compte 50% de la valeur du bien pour les propriétés bâties [4], peut également faire diminuer l'envie d'acheter un bien trop spacieux.

Concernant la localisation, son implémentation dans l'outil sera plus complexe. En effet, les adresses sont rarement précisément écrites sur les sites d'annonces (exemple : LocService.fr [5]), la plupart du temps, ces adresses sont vagues et s'arrêtent à la Ville, exemple Angers. De plus, les adresses "précises" sont pour la plupart écrites dans la description de l'offre et on ne peut donc pas les extraire facilement. Cependant si nous avons accès à une adresse précise, nous pourrions récupérer les coordonnées GPS (Latitude/Longitude), via un site comme par exemple "GPS Coordinate Converter" [6]. A partir de ces coordonnées GPS, nous pourrions utiliser le package Geopy ou Shapely pour construire des zones, exemple : Quartier Monplaisir, Angers Centre, Périphérie et ainsi de suite, afin de savoir dans quel type de quartier se trouve le bien immobilier. Nous pourrions également calculer la proximité avec les transports en commun en allant récupérer les coordonnées GPS de l'ensemble des arrêts de tram. Ensuite nous pourrions utiliser la formule d'Haversine pour calculer la distance entre 2 coordonnées GPS.

Cependant, si l'on s'intéresse à la taille du bien, c'est une information plus facilement trouvable et traitable automatiquement. Cependant le nombre de pièces, de chambres, de salle de bain est une information moins commune sur les sites d'annonces. Cependant sur un site comme SeLoger.fr [8], cette information est disponible. De plus, si on s'intéresse à des appartements, la mention "T1/T2/T3", nous donne déjà une bonne estimation du nombre de pièces, de chambres et de salle de bain.

Les caractéristiques stylistiques quant à elles sont probablement introuvables, cependant ce n'est pas un problème car l'impact sur le prix du logement est plus faible que le nombre de m² ou la localisation.



L'âge du bien immobilier est une information plus importante et aussi plus probablement marqué dans le titre de l'offre, comme sur SeLoger.com, avec par exemple la mention d'"Appartement neuf" ou "Projet en construction". Nous pourrions donc estimer si un logement est neuf ou non. Pour finir, les taxes foncières ne nous concernent pas car notre zone géographique est limitée à Angers et la périphérie, pour information, elles sont de 21.2% à Angers.

Pour finir, nous pouvons donc dire qu'il y a de nombreux éléments influençant le prix d'une maison, cependant, selon nous les variables les plus importantes pour estimer le prix d'une maison seront disponible et facilement extractible, comme par exemple, le nombre de m², le nombre de pièce. Le seul élément plus complexe à traiter et tout aussi important est la localisation.

III. Champs de notre étude

Dans cette seconde partie, nous allons délimiter le champ de notre projet, qu'ils soient d'ordre géospatiales ou pratiques. Ainsi, comme nous l'avons précédemment expliqué, géographiquement, nous nous limitons à la ville d'Angers et ses alentours. Nous avons choisi de réaliser cette décomposition par ville pour différentes raisons. Tout d'abord, il fallait selon nous cadrer le sujet pour éviter de nous disperser, car plus la zone choisie est grande, plus la volumétrie nécessaire à la création d'un estimateur est importante. De plus, les prix de l'immobilier varient énormément selon la ville, cela est lié à la structure de marché. En effet, en fonction de l'attractivité (liés par exemple à l'emploi, aux clusters industriels, à la sécurité, coûts de la vie...) du territoire l'ensemble des prix d'une zone vont en être impactés. Il existe des comparateurs de prix au m² moyen selon la ville qui montre clairement une disparité, à Angers par exemple, le prix moyen au m² est de 2286 € alors qu'à Nantes il est de 3459 € [9]. De plus, en termes de modélisation statistique, la localisation est une donnée difficilement ajoutable dans un modèle. En effet, il aurait fallu créer une variable par Ville, codé 0 et 1 selon si le bien se situe dans la ville, ce qui aurait fait trop de variable.

Concernant les sources de données, nous allons nous limiter à une seule car en fonction du site que nous choisissons, les informations disponibles ne sont pas les mêmes et les retraitements seront donc différents, c'est pour cette raison que nous avons choisi une seule source qui sera le site internet "SeLoger.com". Ce site nous communique de nombreuses informations et surtout les informations les plus déterminantes du prix.

236 000 € 2 865 € le m²

À partir de 1184 € / mois



Appartement neuf

4 pièces • 82 m² • Étage 1/-

Grand Pigeon-Deux Croix-Banchais à Angers (49100)

Autres logements disponibles : 4 pièces

[A voir sur SeLoger neuf →](#)

294 000 € 5 042 € le m²

À partir de 1472 € / mois



Appartement neuf

3 pièces • 2 chambres • 58 m² • Jardin

Monplaisir à Angers (49100)

Autres logements disponibles : 4 pièces • 6 pièces

[A voir sur SeLoger neuf →](#)



187 250 € 2 432 € le m²

À partir de 1001 € / mois

Appartement

5 pièces • 2 chambres • 77 m² • Étage 3/- • Balcon • Box

Centre à Angers (49000)

La capture d'écran à gauche provient du site internet SeLoger. Comme vous pouvez le remarquer, de nombreuses informations sont récupérables : le prix, le type d'appartement (Neuf ou pas), le nombre de pièces, le nombre de m², la présence d'un jardin ou d'un balcon, etc...

Il y a également le quartier où se trouve le logement, qui est une information intéressante, mais cela reste peu précis.

Nous avons également décidé de travailler sur 2 types de biens immobiliers, les maisons et les appartements. De plus nous nous intéressons qu'aux Maisons achetables et aux appartements en locations (Cependant nous aurions aussi pu réaliser les maisons et appartement achetable et les maisons/appartement disponible à la location, nous avons opéré cette distinction afin de créer des estimateurs de meilleure qualité). Ainsi nous aurons 2 estimateurs distincts car ces 2 biens ne représentent pas le même marché. De plus, certaines caractéristiques des logements ne se retrouvent pas dans les maisons et inversement.

L'idée est de créer en Front End, un site où la personne pourra dans un premier temps sélectionner son type de bien, ici maison ou appartement. L'objectif de cette double utilisation est avant tout de montrer que notre outil est déclinable en de nombreuses combinaisons (choix de la ville ou choix du type de bien).

IV. Extraction et traitement des données

Cette première partie concerne le code qui nous permettra d'extraire et de retraiter la base de données que nous utiliserons pour entraîner un modèle en machine learning. Elle se divise en 5 sections distinctes :

- Section 1 : Utilisation d'un web scraper (selenium) afin de récupérer le code source des pages internet
- Section 2 : Extraction des informations qui nous intéresse du code source
- Section 3 : Rangement des informations dans un fichier csv
- Section 4 : Exécution d'une requête Power Query afin de retraiter l'ensemble de nos données.
- Section 5 : Ajout des variables de localisation via le calcul des distances de Haversine.

Nous allons donc détailler ces différentes sections et expliquer nos choix en termes de développement. Dans une dernière partie, nous présenterons un dictionnaire des données.

a. Section 1

Pour la première partie, nous avons réalisé une boucle qui permet d'aller chercher dynamiquement l'ensemble des pages que l'on souhaite récupérer. Pour cela nous sommes allés chercher le site internet SeLogger.com qui est un site assez complet, quand on le compare à "LocSevices.com" (nous avons fait une première version qui allait chercher des informations chez LocServices.com [disponible ici](#). Mais nous nous sommes rendu compte que les informations présentes n'étaient pas suffisantes).

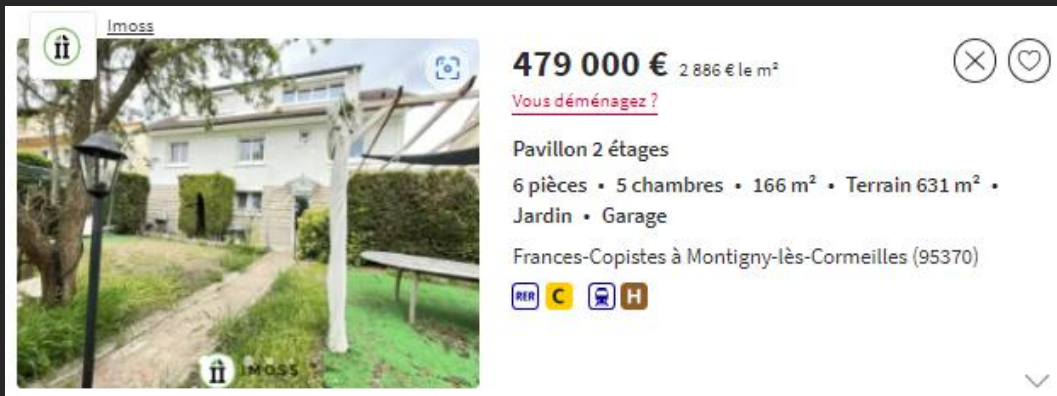
Au départ, nous voulions utiliser plusieurs sites pour faire notre étude, mais nous avons des craintes. Notamment vis à vis des doublons mais aussi car cela aurait causé des problèmes dans notre base de données lorsque certaines informations sont disponibles sur un site et non sur l'autre. Donc notre unique source de données proviendra de "SeLogger.com".

Pour ce faire nous avons donc utilisé le package sélénium qui facilite les opérations de scraping. Comme nous le disions en introduction, nous avons réalisé une boucle afin d'aller chercher plusieurs page Html en même temps, cela est possible grâce à l'URL de SeLogger qui se finit par : `"/?LISTING-LISTpg=1"`. Si l'on remplace le 1 par 2, on passe à la seconde page du site et ainsi de suite. Nous sommes donc allés compter le nombre de sites sur la page puis nous avons créé un array qui stockait les numéros afin de changer de page à chaque boucle et de s'arrêter lorsque l'on a récupéré toutes les pages.

Point d'amélioration possible : - Aller chercher ce nombre de page automatiquement via le nombre de maisons pour notre recherche (information disponible), chaque page affiche 25 maisons donc pour 400 maisons divisé par 25, on aura théoriquement 16 pages.

b. Section 2

Dans cette seconde section, nous allons extraire les informations de nos différents codes sources et les ranger dans un dictionnaire qui est un objet python permettant de ranger plusieurs array (= listes) dans un seul objet. Pour les informations que nous souhaitons extraire, il y a 2 méthodes :



Imoss

479 000 € 2 886 € le m²

[Vous déménagez ?](#)

Pavillon 2 étages

6 pièces • 5 chambres • 166 m² • Terrain 631 m² •
Jardin • Garage

Frances-Copistes à Montigny-lès-Cormeilles (95370)

REI C R H

Le prix, le type de bien et la localisation sont simplement rangés dans des classes assez simples d'accès.

Exemple : `<div data-test="sl.price-label" class="sc-bhlBdH cbvitC">479 000 €</div>`

Cependant, les classes sont spéciales, en effet, elles changent régulièrement (probablement pour des raisons de sécurité) donc il faut manuellement les modifier lorsqu'elles changent. Dans le code « MainAppartementSeLogger.py » nous avons utilisé les attributs « data-test= »sl.price-label », afin d'éviter de devoir réécrire manuellement l'ensemble des classes.

Pour les pièces, chambres, nombre de m², l'extraction est plus complexe car certaines maisons ont des garages et d'autres non. De plus les éléments se trouvent tout dans une seule et même classe :

Exemple : `<ul data-test="sl.tagsLine" class="sc-chAAoq gTPsvx">6 pièces5 chambres166 m²Terrain 631 m²JardinGarage`

Ainsi nous avons réalisé un système qui nous permet d'aller chercher si par exemple le mot "pièces" est présent dans la classe "sc-chAAoq gTPsvx", si c'est le cas il ira chercher le mot "6 pièces" sinon il ira écrire "N" dans la liste afin de spécifier qu'il n'y a pas de pièces dans notre array.

Dans l'ensemble, notre programme fonctionne sous la forme d'une double boucle, la première permet d'aller chercher chaque page html car elles sont rangées dans des documents séparés. La seconde boucle permet, pour chaque maison d'aller extraire l'élément "ul" et d'enregistrer chaque "tag" présent.

c. Section 3

La troisième section permet de sauvegarder les données dans un fichier csv. La seule difficulté réside dans le fait que nos données sont rangées dans un dictionnaire et qu'il faut mettre à la suite dans notre fichier csv (au lieu d'avoir une feuille/un fichier par csv). Pour cela il faudra encore une fois mettre en place une boucle qui s'exécutera en fonction du nombre de pages.

d. Section 4

Dans cette quatrième section nous allons traiter nos données. Ce retraitement aurait pu être réalisé via l'utilisation du package "pandas" de Python. Cependant, nous avons préféré utiliser une requête Power Query pour plusieurs raisons : Tout d'abord, Power Query est un outil spécialement désigné pour faire du retraitement de données, cela nous permet de le faire plus rapidement. Nous voulions également

expérimenter si faire appel à un outil extérieur à Python pour le retraitement pouvait être automatisé via Python.

Une requête Power Query se compose d'un ensemble d'étapes qui vont se lancer simultanément (à la manière d'une macro) par exemple : importation des données, modification du nom d'une colonne, transformation des "N" en "O", etc...

Pour lancer la requête Power Query à partir de Python, nous avons déposé cette requête dans le répertoire python que nous utilisons. Puis, à l'aide de "pyautogui" qui permet d'effectuer de faire des choses à la place d'un humain comme appuyer sur les touches d'un clavier et de "win32com.client" qui permet d'ouvrir des applications Microsoft comme Excel. Ensuite, nous ouvrons notre fichier de requête, puis nous ordonnons à Python d'appuyer sur le raccourci "Alt+F5" qui permet d'actualiser (de lancer) une requête.

A la fin de cette section, nous nous retrouvons avec les données actualisées et retirées mais dans le fichier de requête.

Cependant si nous voulons utiliser nos données, il est préférable de copier ces nouvelles données au lieu d'utiliser le fichier de base car il peut y avoir certains conflits entre Python et Excel (comme la suppression de la requête).

e. Section 5

Cette dernière partie permet de calculer la distance de chaque maison par rapport à des points d'intérêt dans la ville d'angers. Cela permettra de rajouter 4 colonnes à notre base de données : la distance par rapport au centre, au campus Belle Beille, au campus Saint Serge et la plus petite distance par rapport aux arrêts de tram, ligne A et B.

Pour cela nous utiliserons les coordonnées GPS que nous avons récupérées via Google Maps et la formule de Haversine qui permet de calculer la distance entre 2 points d'une sphère [10] (dans notre cas, les coordonnées GPS).

Cependant, la localisation des maisons est rarement très précise (on aura par exemple le quartier, donc une approximation de la localisation). De plus, ce processus n'est pas automatisé donc nous devons aller chercher les localisations manuellement. Dans notre cas, cela n'est pas un problème car nous n'avions que 18 localisations mais dans le cadre d'un projet de plus grosse ampleur, il faudrait trouver un moyen d'automatiser cette tâche.

Ainsi pour cette partie nous reprenons les données présentes dans le fichier de requête et nous utilisons un fichier des coordonnées GPS qui est déjà présent dans le répertoire. Ce fichier compile l'ensemble des coordonnées GPS qui nous intéresse et est [disponible ici](#). Pour la proximité avec les transports en commun, nous avons seulement conservé les arrêts de tram (ligne A et B) car prendre l'ensemble des arrêts de bus aurait représenté un travail de collecte des données trop important. Cependant, il est parfaitement possible de collecter l'ensemble de ces données.

Pour les points d'intérêts (centre et campus), nous calculons simplement la distance entre la maison et les différents points d'intérêts.

Concernant la proximité avec les arrêts de trams, nous avons mis en place une double boucle qui dans un premier temps, saisi les informations pour la *xième* maison, puis dans un second temps calcul la distance entre cette maison et l'ensemble des arrêts de tram. Chacune de ces distances sont rangés dans un array, puis nous sélectionnons la valeur minimale de cet array, cette valeur est ensuite stockée dans notre fichier Excel.

f. Section 6

A la suite de cette étape nous créons un fichier final qui comporte l'ensemble des données que nous utiliserons pour entraîner notre modèle machine learning afin de déterminer le prix d'une maison.

Cette dernière section a pour objectif de vous présenter le dictionnaire des données, pour les données que nous avons recueillies et que nous allons utiliser :

NOM	MAISON	APPARTEMENT	DEFINITION	TYPE_DONNEES
prix_maison	O	O	Prix du bien immobilier où Coût de la location par mois.	Monétaire (float)
type_bien	O	O	Données brutes de SeLoger permettant d'avoir une information générale	Caractère (string)
Localisation	O	O	Données brutes de localisation, information sur le quartier de résidence	Caractère (string)
Latitude, Longitude	O	O	Coordonnée GPS du bien immobilier	Coordonnées (float)
Neuf	O	N	= 1 Biens dit "neuf", lorsque le type de maison est "Maison neuve"/"Projet de construction"	Binaire (bool)
Maison	O	N	= 1 Tout ce qui est une maison, cela exclut les projets en construction, les villas, hôtel, manoir et pavillon	Binaire (bool)
Villa	O	N	= 1 Si le bien est une villa	Binaire (bool)
Meublé	N	O	= 1 Si l'appartement est meublé	Binaire (bool)
Projet en construction	O	N	= 1 Si le bien est un projet en construction	Binaire (bool)
Nombre_étages	O	N	Nombre d'étage de la propriété, va de 0 à 4	Entier (integer)

Nombre_pièces	O	O	Nombre de pièce dans le bien immobilier (or chambres, va de 1 à 14)	Entier (integer)
Nombre_chambres	O	O	Nombre de chambre dans le bien immobilier (va de 0 à 9)	Entier (integer)
Surface_maison	O	O	Surface en m² du bien (va de 100 à 500)	Entier (integer)
Surface_Terrain	O	N	Surface en m² du terrain du bien (va de 0 à 27000)	Entier (integer)
Présence_Jardin	O	O	= 1 S'il y a un jardin	Binaire (bool)
Présence_Box	O	O	= 1 S'il y a un box pour garer sa voiture	Binaire (bool)
Viager	O	N	= 1 Si le bien est en viager	Binaire (bool)
Viager_Occupe	O	N	= 1 Si le bien est en viager occupée par une personne	Binaire (bool)
Viager_Libre	O	N	= 1 Si le bien est en viager libre	Binaire (bool)
Présence_Garage	O	O	= 1 S'il y a un garage pour garer sa voiture	Binaire (bool)
Présence_Piscine	O	O	= 1 S'il y a une piscine	Binaire (bool)
Présence_Parking	O	O	= 1 S'il y a un parking pour garer sa voiture	Binaire (bool)
Numéro_d'Etage	N	O	= Numéro d'étage où se trouve l'appartement	Entier (integer)
Nombre_Balcon	N	O	= Nombre de balcon (va de 0 à 2)	Entier (integer)
Présence_Terrasse	N	O	= 1 S'il y a une terrasse pour l'appartement	Binaire (bool)

Présence_Ascenseur	N	O	= 1 S'il y a un ascenseur dans l'immeuble	Binaire (bool)
Distance_minimale_tram	O	O	Représente la distance entre le bien immobilier et l'arrêt de tram le plus proche (en mètre)	Distance (float)
Distance_centre	O	O	Représente la distance entre le bien immobilier et le centre d'Angers (place Ralliement) (en mètre)	Distance (float)
Distance_SaintSerge	O	O	Représente la distance entre le bien immobilier et le campus Saint-Serge (en mètre)	Distance (float)
Distance_BelleBeille	O	O	Représente la distance entre le bien immobilier et le campus Belle-Beille (en mètre)	Distance (float)

Comme vous pouvez le voir il y a donc 2 bases de données, ainsi certaines informations seront disponibles dans un fichier et non dans l'autre et inversement.

V. Construction de l'estimateur via Machine Learning

Avec cette quatrième partie, nous allons utiliser le machine learning afin de créer un estimateur qui nous permettra par la suite de prédire le prix d'une maison selon certaines caractéristiques.

Pour trouver quel modèle utiliser, nous avons réalisé plusieurs tests :

- Modèle de régression multiple
- Modèle de machine learning random forest
- Modèle de machine learning gradientboostingregressor

Comme nous l'avons constaté dans les parties précédentes, nous avons un ensemble de 26 caractéristiques. Cela fait un peu trop, et augmente le risque d'erreurs, nous avons donc décidé de conserver, pour les maisons, les 15 variables suivantes : neuf, maison, projet_en_construction, nombre_etage, nombre_pieces, nombres_chambre, surface_maison, surface_terrain, présence_jardin, présence_box, présence_garage, présence_parking, distance_minimale_tram, distance_centre, distance_BelleBeille.

Et pour les appartements, les variables suivants : Meublé, Nombre de pièces, Nombres de chambres, Surface de la maison, Présence d'un Jardin, Présence d'un Box, Présence d'un Garage, Présence d'une

Piscine, Présence d'un Parking, Numéro d'Etage, Balcon, Terrasse, Ascenseur, Distance_minimale_tram, Distance_centre, Distance_SaintSerge, Distance_BelleBeille.

Nous allons utiliser l'erreur quadratique moyenne comme indicateur pour choisir le meilleur modèle. Cet indicateur est une mesure de la distance moyenne entre la valeur prédite par notre modèle et la réalité [12]. En effet, plus cet indicateur sera faible, plus notre modèle sera en adéquation avec nos données.

Concernant le modèle avec le machine learning, nous obtenons une erreur quadratique moyenne de 0.1518, pour le modèle avec random forest nous avons un résultat moyen de 0.1042 et pour finir une erreur moyenne de 0.0895 pour le dernier. Compte tenu des résultats nous devrions conserver le modèle avec gradientboostingregressor.

Nous avons néanmoins effectué une boucle en fin de programme nous permettant de retenir le modèle minimisant le RMSE. Ce modèle est ensuite enregistré dans un fichier "pickle". Cette extension de fichier permet de sérialiser, c'est-à-dire de convertir des objets python en chaîne binaire [11] et de désérialiser, c'est-à-dire de convertir des chaînes binaires en objet python. Pour faire simple, cette extension va nous permettre de sauvegarder puis de réutiliser notre modèle de machine learning, notamment pour l'utiliser sur notre interface streamlit.

VI. Partie Front-End

Après avoir récolté et stocké les données puis créé un modèle de machine learning à des fins de prédiction, nous devons à présent mettre au point une IHM qui permettra à l'utilisateur d'interagir avec ces différentes composantes.

En effet, l'objectif du premier modèle précédemment estimé est de permettre à un utilisateur d'estimer le prix d'un bien immobilier au regard de ses caractéristiques. Cela peut notamment l'aider à se situer avant d'engager des démarches plus poussées pour une quelconque vente.

Dans le même temps, il peut être intéressant d'offrir aux utilisateurs un moyen d'estimer le loyer mensuel d'un appartement qu'ils comptent mettre en location. Il s'agit du second modèle que nous avons estimé dans l'étape précédente.

Nous avons mis au point une interface permettant de faire tourner chaque modèle de ML sur de nouvelles données. En effet, les modèles ont précédemment été entraînés par les valeurs de nos jeux de données, il est temps de le mettre à profit en produisant des estimations.

Pour ce faire, il est nécessaire que l'utilisateur indique les caractéristiques du bien immobilier nécessaires pour faire fonctionner le modèle, autrement dit l'ensemble des variables qui composent ce dernier. En définitive, l'interface va permettre de récolter les données entrées par l'utilisateur, et sollicitera le modèle pour afficher l'estimation qui aura été produite.

Après avoir chargé en mémoire les modèles à l'aide de la librairie pickle, nous avons eu recours à streamlit pour construire l'interface. Nous lui donnons dans un premier temps un titre ainsi qu'une petite description pour brièvement expliquer son intérêt. Par la suite, nous avons choisi de diviser en plusieurs catégories les variables pertinentes, et de structurer les parties de sorte à rendre le questionnaire plus digeste et intuitif. Cela permet de ne pas rendre l'individu confus ou ennuyé face à l'enchaînement des questions, qui serait alors plus enclin à ne pas solliciter l'interface. Enfin, l'utilisateur peut valider ses réponses et obtenir une estimation.

Nous avons donc regroupé nos variables dans les cinq parties suivantes pour la partie Vente : Type de bien (Neuf, Maison, Projet_en_construction), Caractéristiques du bien (nombre_etage, nombre_pieces, nombres_chambre), Surface (surface_maison, surface_terrain), Équipements et aménagements

(présence_jardin, présence_box, présence_garage, présence_parking) et enfin Emplacement (distance_minimale_tram, distance_centre, distance_BelleBeille). Pour chacune de ces catégories, nous avons procédé à des agencements "en colonnes" afin de disposer plusieurs questions sur la même ligne et ainsi rendre l'interface plus agréable.

De la même manière, nos variables ont été regroupées en quatre catégories pour la partie location : Caractéristiques du bien (Meublé, Numéro d'étage), Surface et pièces (Nombre de pièces, Nombres de chambres, Surface de la maison), Commodités (Présence d'un Jardin, Présence d'un Box, Présence d'un Garage, Présence d'une Piscine, Présence d'un Parking, Ascenseur, Balcon, Terrasse) et enfin Emplacement (Distance_minimale_tram, Distance_centre, Distance_SaintSerge, Distance_BelleBeille)

Il est également primordial de choisir un widget adapté à la variable que l'on souhaite récupérer, et ce pour éviter au maximum les erreurs de saisie. Par exemple, le widget `st.radio` permet une sélection entre un faible nombre de modalités qui s'excluent mutuellement, le widget `st.checkbox` fonctionne sur le même principe mais est propre aux questions fermées (oui/non), tandis que le widget `st.selectbox` permet de choisir entre un plus grand nombre de modalités sans surcharger l'interface au moyen d'une liste. Enfin, certains widgets sont spécifiques aux valeurs numériques, comme le widget `st.number_input` qui comporte une fonctionnalité de saisie manuelle chiffrée ou bien le widget `st.slider` avec la possibilité de déplacer un curseur pour sélectionner une valeur.

Nous pouvons nous intéresser à certaines spécificités de notre code pour illustrer nos propos.

a. Seuils sur les saisies chiffrées

Pour limiter encore davantage les erreurs, nous pouvons contraindre les utilisateurs à ne pas saisir des entrées incohérentes. Cela s'illustre dans les parties II. et III. du questionnaire pour la partie Vente, par exemple.

Dans un premier temps, il est nécessaire d'indiquer le nombre de pièces ainsi que le nombre de chambres. Cela implique trois évidences : 1/ le nombre de pièces ne peut être nul. 2/ le nombre de chambres ne peut pas être supérieur ou égal au nombre de pièces. 3/ Dès lors qu'il y a plus d'une pièce, il y a nécessairement une chambre. Au regard de ces observations, nous avons construit notre code de sorte à fixer dans un premier temps un seuil minimal de 1 pour le nombre de pièces. Puis, une fois le nombre de pièces récupérées, nous avons créé une variable égale au nombre de pièces moins une, cette variable faisant office de seuil maximal lors de la saisie du nombre de chambres. Enfin, si le nombre de pièces est strictement supérieur à 1, le seuil minimal pour le nombre de chambres est fixé à 1.

```
pieces = st.number_input("Nombre de pièces", step=1, min_value=1)
max_chambres = pieces - 1 #on s'assure de ne pas pouvoir mettre plus ou = de
chambres que de pièces
if pieces ==1:
chambres = st.number_input("Nombre de chambres", step=1, min_value=0,
max_value=max_chambres)
elif pieces > 1:
chambres = st.number_input("Nombre de chambres", step=1, min_value=1,
max_value=max_chambres)
```

Sur le même principe, nous avons imposé une restriction similaire dans le cadre des variables récoltant les surfaces du bien et du terrain. Ces surfaces ne peuvent être nulles (nous avons fixé un seuil minimal à 20

pour une maison à vendre), et le bien ne peut en tout logique pas être plus grand que le terrain. Nous avons donc fixé un seuil minimal à 20 pour le bien, et un seuil minimal à la valeur de la surface du bien pour le terrain.

```
surface_m=st.number_input("Superficie du bien", min_value=20, step=1)
ste_min=surface_m #on s'assure que le terrain soit au moins aussi grand que
la maison
surface_te=st.number_input("Superficie du terrain", min_value=ste_min,
step=1)
```

b. Recueil pertinent des distances

La dernière partie du questionnaire doit permettre à l'utilisateur d'indiquer la distance entre son bien et 1/ l'arrêt de tram le plus proche, 2/ le centre-ville et 3/ le campus Belle Beille (4/ le campus St Serge dans la partie location). Dans le jeu de données, ces informations sont répertoriées en mètres, mais en fonction de l'emplacement du bien certaines distances peuvent être relativement élevées. Pour pallier ce problème, nous avons décidé de laisser le choix à l'utilisateur de faire une sélection en mètres ou en kilomètres. En fonction de son choix, le widget de sélection s'adapte en termes de valeurs, des bornes, et du pas de sélection.

Ainsi, si l'utilisateur choisit une sélection en mètres, il peut indiquer une valeur entre 0 et 1000m avec un pas de 50m. S'il opte pour une sélection en kilomètres, il peut choisir une valeur entre 0 et 10km avec un pas de 0.5km.

Le choix se fait au moyen d'un widget `st.radio` (m ou km) pour chaque distance à récupérer, dont la réponse sélectionnée à pour effet de fixer les paramètres du slider concerné. Enfin, si l'utilisateur a choisi une sélection en kilomètres, la valeur retenue est multipliée par 1000 afin de correspondre au standard du jeu de données.

```
unit_centre = st.radio("centre", ["m", "km"], label_visibility="collapsed")

if unit_centre == "m":
    distance_centre = st.slider("Distance du centre (m)", 0, 1000,
500,
    step=50, label_visibility="collapsed")
else:
    distance_centre = st.slider("Distance du centre (km)", 0.0, 10.0, 5.0,
step=0.5, label_visibility="collapsed")
    distance_centre=distance_centre*1000
```

En définitive, nous obtenons deux questionnaires bien distincts présents sur la même interface qui permettent de récolter les variables nécessaires au fonctionnement du modèle concerné, ainsi que d'un bouton en fin de page qui permet d'afficher les résultats de l'estimation.

Estimateur de la valeur de son bien immobilier (achat/location)

Sur cette page, vous serez en mesure d'obtenir une estimation de la valeur à la vente ou à la location de votre bien à partir de ses caractéristiques principales.

Vous souhaitez :

☒ Estimer le prix d'une maison à la vente
☐ Estimer le prix d'un appartement à la location

I. Type de bien

Je possède une :

☒ Maison
☐ Villa

Votre bien est-il neuf ?

☒ Oui
☐ Non

☐ En construction

II. Caractéristiques du bien

Nombre d'étages :

Nombre de pièces :

Nombre de chambres :

III. Surface

Surface du bien :

Surface du terrain :

IV. Équipements et aménagements

Votre bien dispose-t-il d'un(e) :

☐ Jardin
☐ Garage
☐ Place de parking
☐ Box de stockage

V. Emplacement

Quelle est la distance qui sépare votre bien :

De l'arrêt de tram le plus proche ?

m
 km

Du centre-ville ?

m
 km

Du campus de Belle-Belle ?

m
 km

Estimateur de la valeur de son bien immobilier (achat/location)

Sur cette page, vous serez en mesure d'obtenir une estimation de la valeur à la vente ou à la location de votre bien à partir de ses caractéristiques principales.

Vous souhaitez :

☐ Estimer le prix d'une maison à la vente
☒ Estimer le prix d'un appartement à la location

II. Caractéristiques du bien

Le bien est-il loué meublé ?

☒ Oui
☐ Non

Votre bien se situe à l'étage n° :

III. Surface et pièces

Surface du bien :

Nombre de pièces :

Nombre de chambres :

III. Commodités

L'appartement/l'immeuble dispose-t-il d'un(e) :

☐ Balcon
☐ Ascenseur
☐ Terrasse
☐ Jardin
☐ Piscine
☐ Place de parking
☐ Garage
☐ Box de stockage

IV. Emplacement

Quelle est la distance qui sépare le logement :

De l'arrêt de tram le plus proche ?

m
 km

Du centre-ville ?

m
 km

Du campus de Saint Serge ?

m
 km

Du campus de Belle-Belle ?

m
 km

VII. Conclusion, Limites et ouvertures

Arrivés au terme de notre projet, cette conclusion nous permettra de revenir sur les différents choses réalisés au cours du projets, nous reviendrons également sur les limites et ouvertures possibles.

Dans un premier temps, le but de notre étude était de réaliser un site internet ayant pour fonction d'estimer ou de comparer le prix de sa maison à celui du marché, autrement dit de réaliser une expertise de son bien immobilier. Nous avons uniquement choisi 2 types de marché : la location d'appartement et l'achat de maison. De plus, nous nous sommes concentrés sur la ville d'Angers et ses alentours à cause de la grande disparité des prix selon la région.

Ainsi, notre projet se décompose en 3 parties. La première permet d'extraire des données du site « SeLogger », via une méthode appelée « webscraping », puis le retraitement de ces mêmes données.

La seconde partie a pour objectif de créer un modèle via des méthodes de machine learning. Puis nous comparons ces différents modèles via le RMSE, puis nous choisissons le modèle minimisant ce paramètre pour enregistrer ce modèle dans un fichier joblib (permettant d'enregistrer des informations python dans un format binaire pour le réutiliser).

Pour finir, la troisième partie est un site internet (hébergé localement) qui permet à un utilisateur lambda d'estimer le prix de sa maison ou la mensualité de sa location.

Nous allons désormais parler des différentes limites de notre projet. Tout d'abord, il y a de nombreux problèmes liés au webscraping. Tout d'abord, suite à une mise à jour de Chrome, nous n'arrivons plus à utiliser le package Selenium, le problème n'est pas résolu à ce jour.

De plus, l'utilisation de la boucle que nous avons créée pour récupérer l'ensemble des pages associées à un type de bien ne fonctionne pas car l'administrateur du site nous détecte et exclut du site car nous envoyons trop d'information au serveur. Nous avons essayé de corriger ce problème en ajoutant du délai entre chaque instruction mais nous n'avons pas réussi à corriger ce problème.

Outre cela, il existe une différence de méthode entre les 2 codes permettant de récupérer les données de « SeLoger ». En effet dans le code permettant de récupérer les informations des maisons nous avons utilisé les « class » alors que dans celui des appartements, nous nous sommes servis des « attributs ». Le code des appartements est meilleur car celui des classes change au cours du temps, ce qui induit que nous devons retrouver les classes pour réécrire le code lorsque les classes changent.

Une autre de ces limites concerne la récupération des coordonnées GPS des maisons, des arrêts de tram etc.... Cette opération est manuelle et peu précise. En effet l'adresse que nous récupérons est souvent peu précise. De plus cette opération est entièrement manuelle ce qui est fastidieux à réaliser. Nous aurions pu utiliser une API comme celle de Google Maps mais nous n'avons pas le temps et cela ne rentre pas dans le cadre du cours.

L'une des fonctionnalités que nous aurions pu ajouter est la réalisation de statistiques descriptives en plus de l'estimateur afin de donner une image à un instant T du marché. À l'image d'un tableau de bord Power BI, il y aurait eu le nombre de biens immobiliers sur le marché, le prix moyen, le type de biens disponibles, une carte avec la position des biens etc...

Une autre fonctionnalité que nous pourrions implémenter est l'utilisation des localisations, pour créer des zones géographiques afin de déterminer le type de quartier dans lequel se trouve le bien : banlieue, périphérie, centre, zones industrielles etc...

VIII. Bibliographie

- [1] « Corporate - About », Zillow. <https://www.zillow.com/z/corp/about/> (consulté le 10 septembre 2023).
- [2] « Valuing a House: How to Determine Market Value on A Home | Zillow ». <https://www.zillow.com/learn/how-to-value-a-house/> (consulté le 10 septembre 2023).
- [3] J.-V. Semeraro, « Taxe foncière : ce qu'elle vous coûte, département par département », Capital.fr, 21 octobre 2021. <https://www.capital.fr/votre-argent/taxe-fonciere-ce-quelle-vous-coute-departement-par-departement-1417920> (consulté le 10 septembre 2023).
- [4] « Taxes foncières | collectivites-locales.gouv.fr ». <https://www.collectivites-locales.gouv.fr/finances-locales/taxes-foncieres> (consulté le 10 septembre 2023).
- [5] « Location d'appartement à Angers de particulier à particulier ». <https://www.locservice.fr/maine-et-loire-49/location-appartement-angers.html> (consulté le 10 septembre 2023).
- [6] « GPS coordinate converter ». <https://www.gps-coordinates.net/gps-coordinates-converter> (consulté le 10 septembre 2023).

- [7] « Calcul de distance avec latitudes et longitudes ». <http://villemin.gerard.free.fr/aGeograp/Distance.htm> (consulté le 10 septembre 2023).
- [8] « 1 381 annonces de ventes et de viagers de maisons ou d'appartements à Angers (49000), Seloger.com ». https://www.seloger.com/list.htm?projects=2,5&types=2,1&natures=1,2,4&places=%5b%7B%22inseeCodes%22:%5b490007%5d%7D%5d&sort=a_px&mandatorycommodities=0&enterprise=0&qsversion=1.0&m=search_refine-redirection-search_results (consulté le 10 septembre 2023).
- [9] « Prix de l'immobilier : Classement des villes en région Île-de-France ». <https://www.bien-dans-ma-ville.fr/classement-ville-prix-m2-ile-de-france/> (consulté le 10 septembre 2023).
- [10] « Formule de haversine », Wikipédia. 5 mai 2021. Consulté le : 10 septembre 2023. [En ligne]. Disponible sur : https://fr.wikipedia.org/w/index.php?title=Formule_de_haversine&oldid=182613177
- [11] « pickle — Python object serialization », Python documentation. <https://docs.python.org/3/library/pickle.html> (consulté le 10 septembre 2023).
- [12] Zach, « How to Interpret Root Mean Square Error (RMSE) », Statology, 10 mai 2021. <https://www.statology.org/how-to-interpret-rmse/> (consulté le 10 septembre 2023).
- [13] « Create an app - Streamlit Docs ». <https://docs.streamlit.io/> (consulté le 10 septembre 2023).

IX. Annexes

Code pour Scraping LocServices : <https://drive.google.com/file/d/1GAto1Tw2qZPBJ9bqtRMfwO-itENWSL6O/view?usp=sharing>

Fichier Excel des coordonnées GPS : https://docs.google.com/spreadsheets/d/1-tKU-xtjnZ9AvTz6xw8qbEydkJ1OT_cT/edit?usp=sharing&oid=118087692705218831279&rtpof=true&sd=true