

L'objectif de ce projet est d'étudier, en utilisant des méthodes d'apprentissage automatique, l'impact de différents critères (notes des critiques, appellation) sur le prix d'un vin.

Pour ce faire, on s'appuiera sur le site Millesima (<https://www.millesima.fr/>), qui a l'avantage de ne pas posséder de protection contre les bots. Par respect pour l'hébergeur du site, on veillera à limiter au maximum le nombre de requêtes. En particulier, on s'assurera d'avoir un code fonctionnel avant de scraper l'intégralité du site, pour éviter les répétitions.

Instructions

Ce projet est à réaliser impérativement en binôme : les projets individuels seront pénalisés. Seules les bibliothèques spécifiquement mentionnées dans le sujet sont autorisées.

Le jury donnera zéro s'il s'aperçoit lors de la soutenance que le code n'est pas maîtrisé. Ce sera en particulier le cas si le code a été produit via une IA générative.

Premier jalon : récupération des données en python

Tout projet de machine learning s'appuie sur un ensemble massif de données. L'objectif de cette première partie est la récupération de ces données, et leur stockage dans le format CSV.

Cette partie est à coder en `python`.

Étudier la page d'un vin sur le site <https://www.millesima.fr/>.

On pourra prendre comme exemple la page du millésime 2016 de Château Gloria : <https://www.millesima.fr/chateau-gloria-2016.html>, reprise dans les Figures 1, 2 et 3.

Question 1 En utilisant les bibliothèques `selenium` (dont on lira la documentation) et `Beautiful soup`, écrivez une fonction `getsoup()` qui prend en entrée l'URL de la page d'un vin, et renvoie la soupe correspondant à cette page HTML.

Ici, nous utilisons `selenium` plutôt que `requests` dans la mesure où une partie du contenu des pages de ce site est généré via un script JS. Contrairement à `requests`, `selenium` permet de lancer un navigateur et son JavaScript engine, nécessaire à l'exécution du code JavaScript.

Question 2 En haut de chaque page (cf. l'encadré rouge sur la Figure 1) est indiqué le prix du vin. Attention, on sera intéressé par le prix unitaire, et pas par le prix d'une caisse : ici, 60,33€ pour une bouteille, et pas 724,00€ pour une caisse de 12 bouteilles.

Écrivez une fonction `prix()` qui prend en entrée la soupe correspondant à la page d'un vin, et renvoie son prix unitaire. Dans l'exemple, il faudra renvoyer 60.33.

L'*appellation* d'un vin reflète sa provenance géographique : il peut s'agir d'une région entière (par exemple, *Languedoc*), d'une ville ou d'un village (par exemple, *Pauillac*), ou même d'une seule parcelle dans les cas les plus précis (par exemple, *Chambolle-Musigny Les Amoureuses*, qui correspond à une parcelle bien délimitée du village de Chambolle-Musigny).

Question 3 Écrivez une fonction `appellation()` qui attend en entrée la soupe correspondant à la page d'un vin et renvoie, sous forme de chaîne de caractères, l'appellation du vin. Cette information est disponible plus bas dans la page dans le tableau "Détails", comme indiqué en rouge sur la Figure 2. Ici, on renverra "Saint-Julien".

25€ de remise tous les 250€ d'achats* avec le code : MILL26 - J'en profite

Besoin d'aide ? TRUSTED SHOPS 4.58/5

MILLESIMA BORDEAUX

VINS BORDEAUX BOURGOGNE CHAMPAGNE PRIMEURS À L'UNITÉ OFFRES SPÉCIALES SPIRITUEUX

Rechercher

Vente de vins | Tous nos vins | Vins français | Vins de Bordeaux | Vins de Saint-Julien | Château Gloria | Château Gloria 2016

★ J. Suckling 94/100
+8 notes



Château Gloria 2016
Bordeaux - Saint-Julien - Rouge - [Détails](#)

724,00 € T.T.C.
80,33 € / unité

Conditionnement : Une caisse de 12 Bouteilles (75cl)

1 x 75CL 65,80 €	12 x 75CL 724,00 €	6 x 1.5L 724,00 €
---------------------	-----------------------	----------------------

1

En stock

FIGURE 1 – En rouge, le prix (à l'unité) du vin.

Détails

Pays	France
Région	Bordeaux
Appellation	Saint-Julien
Couleur	Rouge
En stock	Livable
Alcool	13.5
Encépagement	Cabernet Sauvignon/Merlot/Cabernet Franc/Petit Verdot
Mention qualité	AOC
Allergènes	Contient des sulfites

FIGURE 2 – Les détails du vin. On s'intéressera à son appellation, en rouge.

Dans le monde du vin, un certain nombre de critiques donnent des notes aux différentes cuvées. Il existe un grand nombre de ces critiques, mais on s'intéressera seulement dans ce sujet à trois des plus connus : Parker, J. Robinson et J. Suckling.

Les notes attribuées par ces critiques à la cuvée en question se retrouvent dans la section "Évaluation et notation" de la page du vin. Les trois notes qui nous intéressent apparaissent en rouge dans la Figure 3.

Question 4 Écrivez une fonction `parker()`, qui prend une soupe en argument et renvoie la note

Évaluations et notation

Notation Commentaire Robert Parker

				
Parker	J. Robinson	Decanter	Wine Spectator	J. Suckling
93/100	17.5/20	94/100	94/100	94/100

FIGURE 3 – Les notes attribuées par les critiques. En rouge, celles de Parker, J. Robinson et J. Suckling.

attribuée au vin par Parker. La réponse devra respecter les instructions suivantes :

- on omettra la base de la note (ici, "/100"). Dans ce cas, on renverra donc 93.
- tous les critiques ne notent pas tous les vins : en cas d'absence de note de Parker, on renverra la constante **None**.
- la note est parfois une plage (par exemple, "90-93/100"). Dans ce cas, on renverra la moyenne (dans cet exemple, 91.5).
- la note est parfois agrémentée d'un + (par exemple, "96+/100"). On ne tiendra alors pas compte du + (dans ce cas de figure, on renverra simplement 96).

On testera cette fonction sur différentes pages du site pour s'assurer qu'elle couvre tous les cas de figure évoqués.

Question 5 Implémentez les fonctions `robinson()` et `suckling()` qui reprennent la question précédente, mais pour les critiques J. Robinson et J. Suckling. On fera évidemment en sorte d'éviter toute duplication de code : il faudra factoriser un maximum de code entre ces trois fonctions.

Question 6 Écrivez une fonction `informations()` qui attend la soupe de la page d'un vin en argument, et renvoie une chaîne de caractères contenant toutes les informations de l'annonce, séparées par des virgules, dans l'ordre :

"Appellation,Parker,J.Robinson,J.Suckling,Prix"

Sur l'annonce de Château Gloria 2016, cette fonction devra donc renvoyer :

"Saint-Julien,93,17.5,94,60.33"

Question 7 Nous allons limiter nos recherches aux vins de Bordeaux, détaillés sur la page <https://www.millesima.fr/bordeaux.html?page=1> et les suivantes.

En étudiant l'URL et la structure des différentes pages de résultats (il devrait y en avoir entre 50 et 100), écrivez un script qui parcourt toutes les annonces proposées par le site et appelle `informations()` sur les soupes correspondantes.

Les résultats obtenus devront être enregistrés, à raison d'une ligne par annonce, dans un fichier CSV (comma-separated values) dont la première ligne indiquera les étiquettes (sans guillemets) des différents champs :

<code>Appellation,Robert,Robinson,Suckling,Prix</code>
--

Pour la suite du projet, on travaillera à partir de ce fichier CSV local : il ne faudra surtout pas scraper le site à chaque nouvelle session de travail !

Deuxième jalon : Nettoyage des données

Avant de pouvoir utiliser notre jeu de données dans le cadre d'un apprentissage automatique, il va nous falloir les analyser et les nettoyer. Pour cela, on utilisera la bibliothèque `pandas`.

Question 8 Commencez par importer le contenu du CSV de la partie précédente dans un `DataFrame` nommé `vins`, et vérifiez la conformité des données.

Question 9 Supprimez de `vins`, s'il y en a, toutes les lignes n'ayant pas d'appellation.

La bibliothèque d'apprentissage automatique `scikit-learn`, que nous utiliserons dans la partie suivante, ne peut manipuler que des données numériques. Il va falloir procéder à un certain nombre de conversions et de complétions.

Unicode

En fonction de la manière dont vous avez traité la première partie, il est probable que la colonne `"Prix"` contienne un caractère unicode "espace insécable" (`'\u202f'`) pour les vins coûtant plus de 1000€. Cela posera problème au moment de la conversion de cette chaîne vers un entier, il faut donc retirer ce caractère.

Question 10 En vous aidant de la documentation python concernant l'unicode, écrivez une fonction qui attend une chaîne de caractères en unicode et renvoie la sous-chaîne retrainte aux caractères ASCII : elle supprimera donc tous les caractères unicode non-ASCII. Appliquez cette fonction à chaque élément de la colonne `vins["Prix"]`, et vérifiez que le caractère `'\u202f'` n'apparaît plus.

Notes manquantes

Certains champs des colonnes `"Robert"`, `"Robinson"` et `"Suckling"` de `vins` ne contiennent pas de valeur numérique, puisque chaque critique ne note pas chaque vin. Nous allons remplir ces champs d'une manière adaptée au contexte.

Question 11 Pour chaque appellation, calculez la moyenne des notes "Robert" des vins de cette appellation – on écartera temporairement les vins n'ayant pas de note par "Robert" au moment de calculer la moyenne – et stockez le résultat dans un `DataFrame`. Dans le cas où aucune note n'existerait pour une appellation, on fera en sorte d'attribuer la note 0.

Indication : Vous aurez besoin des fonctions `groupby()` et `mean()`.

Question 12 Répétez l'opération de la Question 11 (en factorisant le code au maximum) pour "Robinson" et "Suckling".

Question 13 Modifiez `vins` en remplaçant les valeurs manquantes de chacune des colonnes "Robert", "Robinson" et "Suckling" par la note moyenne des vins de la même appellation, en faisant une jointure (via `merge()`) avec le(s) `DataFrame` obtenu(s) aux questions précédentes.

Colonne "Appellation"

On va maintenant se concentrer sur la colonne "Appellation", qui est la seule contenant des valeurs non-numériques. Pour cette colonne, on va utiliser la méthode des variables indicatrices. Cela consiste à créer une nouvelle colonne pour chaque appellation `a` possible. Une ligne possèdera un 1 dans la colonne `a` si et seulement si son `Appellation` vaut `a`, et un 0 dans toutes les autres colonnes. Un exemple minimal est donné en Figure 4.

	Appellation	...		App_Pauillac	App_Pomerol	App_Margaux	...
0	Pauillac	...	0	1	0	0	...
1	Margaux	...	1	0	0	1	...
2	Margaux	...	2	0	0	1	...
3	Pomerol	...	3	0	1	0	...

(a) Avant
(b) Après

FIGURE 4 – Illustration de l'introduction de variables indicatrices

Question 14 En utilisant la fonction `get_dummies()` de `pandas`, créez des variables indicatrices pour remplacer la colonne "Appellation".

À ce stade, on devrait obtenir un `DataFrame` ayant une trentaine de colonnes et entre 2000 et 3000 lignes n'ayant que des valeurs numériques. Nos données sont prêtes pour la phase d'apprentissage.