

# **PROJET PYTHON**

## **WEB SCRAPING**

### **DU SITE D'AMAZON**

**HYESU\_KIM**  
**UNIVERSITÉ PARIS SACLAY**  
**MASTER 2 IMSD 2018/2019**

A decorative wavy line in white and yellow on the left side of the slide.

**\_ INTRODUCTION**

**\_ DATA SCRAPING**

**\_ DATA PROCESSING**

**\_ DATA ANALYSIS**

**\_ CONCLUSION**

A decorative wavy line in yellow and white on the left side of the image.

# **PARTE 1\_** **INTRODUCTION**

# POURQUOI LE SITE D'AMAZON?

- C'est un site internet d'e-commerce qui offre non seulement les détails du produit, mais aussi les derniers prix du marché, les vendeurs disponibles pour des codes PIN particuliers, les évaluations, etc.
- « Amazon prices » (les prix Amazon) indiquent approximativement le prix actuel en ligne d'un article. Les données sont donc une bonne référence pour comparer le même produit sur d'autres sites d'e-commerce.
- Aussi c'est parce que leur interface est bien structurée en fonction de données très variées comme par exemple, le prix de produit neuf et d'occasion, les critiques, les classements (les étoiles), etc.

# POURQUOI LES DONNÉES DES LIVRES ?

- Les livres sont un produit symbolique pour Amazon. Grâce à (ou à cause de) l'Amazon, l'industrie du livre a été totalement changée.
- L'industrie qui était très traditionnelle avant l'arrivée de l'Amazon, a été rapidement digitalisée et tous les livres et leurs données sont tous maintenant enregistrés sur le site d'Amazon.
- C'est une des raisons de son impact énorme sur les consommateurs, les vendeurs, les auteurs, et les intéressés dans l'industrie, online ou offline.
- De plus, je suis intéressée par la littérature française en tant que étudiante étrangère en France car connaître la culture du pays où j'habite est très important.

# QUELLES SONT LES DONNÉES À SCRAPER?

- Parmi les informations sur la page d'un livre, j'ai choisi neuf variables différentes afin de pouvoir connaître **l'évolution du prix d'un livre neuf** en fonction de différentes caractéristiques du livre et **la relation entre les variables**.
- Les données choisies sont :
  - Le titre du livre (variable de catégorie)
  - Le nom d'auteur(variable de catégorie)
  - Le nom de la maison d'édition (variable de catégorie)
  - Le nom de la collection (variable de catégorie)
  - La date de publication
  - Le nombre d'avis (variable continue)
  - Les étoiles (variable continue)
  - Prix du livre neuf (variable continue)

A decorative wavy line in white and yellow runs vertically along the left side of the image.

# **PARTE 2\_ DATA SCRAPING**

# LES ÉTAPES DU SCRAPING 1/4

I. Récupérer l'url de toutes les pages du genre, littérature française.

```
In [3]: pages = []

for i in range(7, 8): # change to range 1-76 for entire pages
    page_url = "https://www.amazon.fr/s/ref=sr_pg_{0}?fst=as%3Aoff&rh=n%3A301061%2Cn%3A%21301130%2Cn%3A301132%2Cn%3A302038%2Cp_n_b"
    pages.append(page_url)

print(pages)

['https://www.amazon.fr/s/ref=sr_pg_7?fst=as%3Aoff&rh=n%3A301061%2Cn%3A%21301130%2Cn%3A301132%2Cn%3A302038%2Cp_n_binding_brows
e-bin%3A492481011&page=7&bbn=302038&ie=UTF8&qid=1540392876']
```



# LES ÉTAPES DU SCRAPING 2/4

2. Récupérer l'url de chaque page d'un livre (produit) en tirant le code ASIN (Amazon Standard Identification Number).

```
In [4]: asin_pattern = re.compile(r"(?<=/dp/)(\w{10})") # to extract the ASIN (Amazon Standard Identification Number)
books_url = []

for page in pages:
    resp = session.get(page, headers= headers).content # Giving User-Agent will help to be considered as a real user
    html_amazon = BeautifulSoup(resp,"html.parser")
    books = html_amazon.find_all('a', attrs={"class", "a-link-normal s-access-detail-page s-color-twister-title-link a-text-normal"

    for book in books:
        url = book.get('href') # get the href
        asin = re.search(asin_pattern, url) # search the asin pattern
        product_url = "https://www.amazon.fr/dp/" + asin.group(1) # write an adress with asin code
        books_url.append(product_url) # append it in the list of books_url

print(books_url)
```

```
['https://www.amazon.fr/dp/2013949766', 'https://www.amazon.fr/dp/2266255126', 'https://www.amazon.fr/dp/2266163744', 'http
s://www.amazon.fr/dp/2253108618', 'https://www.amazon.fr/dp/2350305368', 'https://www.amazon.fr/dp/2081412144', 'https://www.a
mazon.fr/dp/2253152846', 'https://www.amazon.fr/dp/2253004227', 'https://www.amazon.fr/dp/2253095060', 'https://www.amazon.fr/
dp/225310907X', 'https://www.amazon.fr/dp/2266276298', 'https://www.amazon.fr/dp/226627628X', 'https://www.amazon.fr/dp/226622
6061', 'https://www.amazon.fr/dp/2266275143', 'https://www.amazon.fr/dp/2070410854', 'https://www.amazon.fr/dp/2070368106']
```

# LES ÉTAPES DU SCRAPING 3/4

3. Récupérer toutes les variables choisies sur la page du produit en utilisant l'expression régulière.

```
Virginie Grimaldi
<li><b>Broché:</b> 240 pages</li>
<li><b>Moyenne des commentaires client :</b>
<span class="dpProductDetail225310079X">
<span class="a-declarative" data-a-popover="{\"closeButton\":\"false\", \"max-width\":\"700\", \"position\":\"triggerBottom\", \"u
rl\":\"/review/widgets/average-customer-review/popover/ref=acr_dpproductdetail_popover?ie=UTF8&asin=225310079X&a
mp;contextId=dpProductDetail&ref=acr_dpproductdetail_popover\"}" data-action="a-popover">
<a class="a-popover-trigger a-declarative" href="javascript:void(0)">
<a class="a-link-normal a-text-normal" href="https://www.amazon.fr/product-reviews/225310079X/ref=acr_dpproductdet
ail_text?ie=UTF8&showViewpoints=1">
<i class="a-icon a-icon-star a-star-4-5"><span class="a-icon-alt">4.7 étoiles sur 5</span></i>
</a>
<i class="a-icon a-icon-popover"></i></a>
</span>
<span class="a-letter-space"></span>
<span class="a-size-small">
<a class="a-link-normal" href="https://www.amazon.fr/product-reviews/225310079X/ref=acr_dpproductdetail_text?ie=UT
F8&showViewpoints=1">
  15 commentaires client
</a>
</span>
</span>
</li>
4.7
15
HERE Editeur : Le Livre de Poche (31 octobre 2018)
Le Livre de Poche
collection: Littérature
31 octobre 2018
5.0
```

# LES ÉTAPES DU SCRAPING 4/4

4. Retransformer les données en DataFrame de package 'pandas' et écrire un fichier csv avec cette DataFrame.

```
In [9]: columns = {'book_names': book_names, 'author_names': author_names, 'editor_names': editor_names, 'collections': collections,
                  'publication_dates': publication_dates, 'page_numbers': page_numbers, 'stars_counts': stars_counts,
                  'comments_counts': comments_counts, 'prices_new': prices_new }
```

```
df = pd.DataFrame(columns)
print(df)
```

```
          book_names \
0  Bibliocollège - Nouvelles réalistes, Maupassant
1      Entre mes mains le bonheur se faufile
2      Bel-Ami à 1,99 euros
3  La mort du roi Tsongor - Prix Goncourt des Lyc...
4  Marivaux : La Dispute ; La Fausse suivante ; L...
5      Le Mariage de Figaro
6      Métaphysique des tubes
7      Germinal
8      Et tu n'es pas revenu
9      Le ventre de l'Atlantique
10     L'Instant présent
11     Central Park
12     Marie d'en haut
13     La Fille de Brooklyn
14     Pierre et Jean
15    Le Ravissement de Lol V. Stein
```

# LES DIFFICULTÉS RENCONTRÉES 1/3

- **La difficulté** : trouver les expressions régulières qui marchent toujours pour extraire les données spécifique comme le nom de l'éditeur, le nombre de pages, etc.
- **La raison** : Amazon modifie la place et la façon de mettre des données afin de sécuriser ces dernières. J'ai souvent rencontré soit une erreur disant qu'il n'avait pas trouvé les patterns soit des données inattendues.
- **La solution** : « Trial and Error » - j'ai beaucoup modifié et testé l'expression régulière pour extraire certaines données.
- **Le résultat** : j'ai pu récupérer toutes les données précises donc j'ai besoin avec les expressions régulières.

# LES DIFFICULTÉS RENCONTRÉES 2/3

- **La difficulté** : lorsque j'ai exécuté le code avec 'requests', plusieurs pages affichaient des listes vides et ce cas s'est produit plusieurs fois. C'est l'étape 2 qui fait du parsing la page html en forme de soup (BeautifulSoup) n'a pas fonctionné du tout.
- **La raison** : En cherchant sur Google, j'ai trouvé une réponse : « Si le site que vous voulez 'parser' est dynamiquement créé en Javascript, le package 'requests' n'est pas opérationnel. » Le site d'Amazon est le cas. - référence : <https://stackoverflow.com/questions/45537642/webscraping-with-beautifulsoup-getting-empty-list/45537682>
- **La solution** : utiliser Selenium + WebDriver (Chrome) au lieu de 'requests'
- **Le résultat** : ça a fonctionné un certain temps puis des erreurs sont apparues comme 'timeout', 'handshake failed', 'list index out of range', etc. Aussi, l'exécution du programme a pris plus du temps que le code avec 'requests'.

# LES DIFFICULTÉS RENCONTRÉES 3/3

- **La difficulté** : Comme mentionné dans la deuxième difficulté , Selenium et Chromedriver ne fonctionnaient pas toujours.
- **La raison** : Honnêtement, je ne connaissais pas la raison exacte. Peut-être j'avais exécuté trop de code, donc Amazon m'a bloqué ? De toutes façons, le code marchait bien (après avoir eu un message d'erreur de 'list index out of range', il a récupéré les données quand je l'ai relancé).
- **La solution 1** : chercher un autre moyen - revenir 'requests' + 'time.sleep'
- **La solution 2** : enlever tous les cookies et changer le 'header' à chaque lancement du code afin de ne pas être connu par Amazon
- **Le résultat** : ça a bien marché et j'ai récupéré les données de toutes les pages (1-75) pourtant à partir des pages 16-20, il n'a pas fourni quelques noms d'auteurs ou noms d'éditeurs. Il a parfois doublé une ligne. J'estime que c'est Amazon qui est à l'origine du dysfonctionnement parce que si ça avait été un problème de code, je n'aurais même pas réussi à avoir le DataFrame et à créer le fichier csv à la fin du code.

# AUTRES PETITES DIFFICULTÉS...

- L'installation de Selenium et ChromeDriver était compliquée.
- Il m'a donné des erreurs si j'exécutais le code pour plus de 6 pages. D'ailleurs, j'ai lu qu'Amazon interdit de faire plus de 5,000 requêtes en moyenne.
  - *Donc, j'ai relancé le code pour tous les 5 pages et ça a marché.*
- Lorsque j'utilisais le code avec Selenium et si je ne bougeais pas le curseur, le code s'arrêtait et me demandait de mettre les caractères indiqués pour savoir si j'étais un robot ou pas.
  - *J'étais toujours à côté de mon pc et essayais de bouger le curseur pour lui rassurer que je n'étais pas un robot.*
- L'ordre des livres sur des pages changent dynamiquement
  - *J'ai vite lancé du code une page après une page mais il y avait des délais quand même..*

A decorative wavy line in yellow and white on the left side of the image.

# **PARTE 3\_** **DATA** **PROCESSING**



# LES DONNÉES PROPRES 1/2

- Comme les données manquantes n'étaient pas si nombreuses et le problème ne venait pas du code, j'ai décidé de compléter manuellement les données en comparant les détails de chaque produit.
  - J'ai inscrit, manuellement, 2 noms d'éditeurs et 10 noms d'auteurs.

| book_names                                 | author_names      | editor_names      | collections      | publication_dates |
|--|-------------------|-------------------|------------------|-------------------|
| Gargantua                                  | Francois Rabelais | Points            | POINTS           | 1 janvier 1997    |
| On ne badine pas avec l'amour à 1,55 euros | Alfred de MUSSET  | Pocket            | Classiques       | 23 juin 2005      |
| Le Livre de ma mère                        | Albert Cohen      | Gallimard         | Folio            | 25 avril 1974     |
| Lambeaux                                   | Charles Juliet    | Gallimard         | Folio            | 11 avril 1997     |
| La dynastie des Forsyte 1: Le propriétaire | John Galsworthy   | Archipoche        | Romans étrangers | 12 septembre 2018 |
| L'Enfant                                   | Jules Vallès      | Le Livre de Poche | Classiques       | 1 juillet 1972    |

\*L'observation en jaune est la valeur qui a été bien récupérée

\*\*L'observation en vert est la valeur qui a été complétée par moi

\*\*\*L'observation en bleu est la valeur doublé

# LES DONNÉES PROPRES 2/2

- Après, j'ai réuni tous les fichiers csv dans un fichier et enlevé les lignes qui avaient des valeurs redondantes en utilisant Python.
  - 75 doublons ont ainsi été supprimés.

```
In [3]: ## Merge all the csv files while getting rid of duplicates

csv_files = glob.glob("../data/*.csv") # find all the csv files in data directory
df = pd.concat((pd.read_csv(f, encoding = "utf-8-sig", engine= "python") for f in csv_files)) # concatenate all the csv files in
df_deduplicated = df.drop_duplicates() # drop the duplicated rows

save_path = "C:/Users/kimi/Desktop/IMSD/Cours/Python/data/"
df_deduplicated.to_csv(save_path+"merged_data.csv", sep=',', encoding= "utf-8-sig" , index=False) # write one combined csv file
```

- Au final j'ai obtenue 1.125 lignes et 9 variables (1200 lignes avant du traitement).

# LES DONNÉES RÉCUPÉRÉES

```
data_path = "../data/merged_data.csv"
data = pd.read_csv(data_path, encoding = "utf-8-sig", engine= "python")
# important to have "utf-8-sig" encoding in order to keep French characters with accent such as é, à, ç, etc.

print(data.shape) # 1,125 observation X 9 columns
print(data.info()) # 5 variables in object dtype, 3 in float64 dtype, and 1 in int64 dtype

data.head()
```

|   | book_names                | author_names        | editor_names | collections          | publication_dates | page_numbers | stars_counts | comments_counts | prices_new |
|---|---------------------------|---------------------|--------------|----------------------|-------------------|--------------|--------------|-----------------|------------|
| 0 | Demain                    | Guillaume MUSSO     | Pocket       | BEST                 | 5 janvier 2017    | 544.0        | 4.4          | 538             | 8.1        |
| 1 | Désolée, je suis attendue | Agnès MARTIN-LUGAND | Pocket       | BEST                 | 6 avril 2017      | 416.0        | 4.1          | 245             | 7.5        |
| 2 | Belle du Seigneur         | Albert Cohen        | Gallimard    | Folio                | 12 février 1998   | 1109.0       | 4.1          | 115             | 12.6       |
| 3 | Tous les matins du monde  | Pascal Quignard     | Gallimard    | Folio                | 16 novembre 1993  | 116.0        | 3.9          | 28              | 6.6        |
| 4 | La cicatrice              | Bruce Lowery        | J'AI LU      | LITTERATURE GENERALE | 21 octobre 1999   | 125.0        | 4.3          | 41              | 4.0        |

# LE NETTOYAGE DES DONNÉES 1/3

## - LES VALEURS MANQUANTES

```
# 17 missing values in collections and 5 of them in page numbers
# 1. For continuous variable: 'page_numbers'
mean_pages = round(np.mean(data['page_numbers']))
data['page_numbers'].fillna(mean_pages, inplace=True) # replace the NaN value by mean of page
numbers

# 2. For categorical variables: 'collection_names'
no_collection = "no collection"
data['collections'].fillna(no_collection, inplace = True) # replace the NaN value with "no collection"
```

|                   |    |
|-------------------|----|
| book_names        | 0  |
| author_names      | 0  |
| editor_names      | 0  |
| collections       | 17 |
| publication_dates | 0  |
| page_numbers      | 5  |
| stars_counts      | 0  |
| comments_counts   | 0  |
| prices_new        | 0  |
| dtype: int64      |    |

- Les données manquantes en 'page\_numbers' ont été remplacées par la moyenne du nombre de pages.
- Normalement, les données catégoriques sont remplacées par la valeur la plus fréquente pourtant dans ces données, cette information est manquante car il n'y a tout simplement aucune information sur la collection pour les livres spécifiques. Par conséquent, je les ai remplacées par 'no\_collection'.

# LA NETTOYAGE DES DONNÉES 2/3

## - *LE FORMAT DES VARIABLES*

```
# 3.2.1 String to datetime: 'publication_dates'
publication_dates_parsed = []
for date in data['publication_dates']:
    date_parsed = dateparser.parse(date, settings={'PREFER_DAY_OF_MONTH': 'first'})
    publication_dates_parsed.append(date_parsed)

data['publication_datetime'] = publication_dates_parsed # add a new column 'publication_datetime'
pd.to_datetime(data['publication_datetime']) # convert the data type from string to datetime
data.drop(['publication_dates'], inplace=True, axis=1) # drop the old column

# 3.2.2 Float to int: 'page_numbers'
data['page_numbers'] = data['page_numbers'].astype('int64')
```

- Comme la date de la publication était écrite en type String et ce n'était pas bien organisé, j'ai utilisé un package, 'dateparser', qui permet de faire le string lisible pour transformer en format de datetime.
- Les pages sont toujours comptées en nombre entier, donc j'ai changé le type de 'float64' à 'int64'.

# LA NETTOYAGE DES DONNÉES 3/3


## - *LES DONNÉES CARACTÉRISTIQUES*

```
## 3.3 character variables: 'book_names', 'author_names', 'editor_names', 'collections'
```

```
data['book_names'] = data['book_names'].str.lower() # 'str' allows to lowercase the string  
data['author_names'] = data['author_names'].str.lower()  
data['editor_names'] = data['editor_names'].str.lower()  
data['collections'] = data['collections'].str.lower()
```

```
data.tail()
```

- Afin de grouper les données d'un même livre, même auteur, même éditeur, ou même collection, tous les caractères doivent être écrits en miniscule.

A decorative wavy line in yellow and white on the left side of the slide.

# **PARTE 4\_ DATA ANALYSIS**

# ANALYSE UNIVARIÉE

## - VARIABLES CONTINUES

### I. La tendance centrale

- Le remplacement des valeurs manquantes par la moyenne (321), n'a pas affecté la moyenne et l'écart standard la variable « page\_numbers »
- Comme il y a des livres qui n'ont encore eu ni commentaires ni étoiles, le valeur minimum des variables « stars\_counts » et « comments\_counts » est 0.
- La plupart des livres ont un prix inférieur à 10 euros.

|       | page_numbers | stars_counts | comments_counts | prices_new  |
|-------|--------------|--------------|-----------------|-------------|
| count | 1125.000000  | 1125.000000  | 1125.000000     | 1125.000000 |
| mean  | 321.100444   | 3.964267     | 62.456889       | 6.851280    |
| std   | 197.768534   | 0.800847     | 109.170274      | 3.049983    |
| min   | 16.000000    | 0.000000     | 0.000000        | 1.500000    |
| 25%   | 186.000000   | 3.800000     | 11.000000       | 5.500000    |
| 50%   | 275.000000   | 4.100000     | 26.000000       | 6.950000    |
| 75%   | 416.000000   | 4.300000     | 65.000000       | 8.000000    |
| max   | 1691.000000  | 5.000000     | 1290.000000     | 62.500000   |



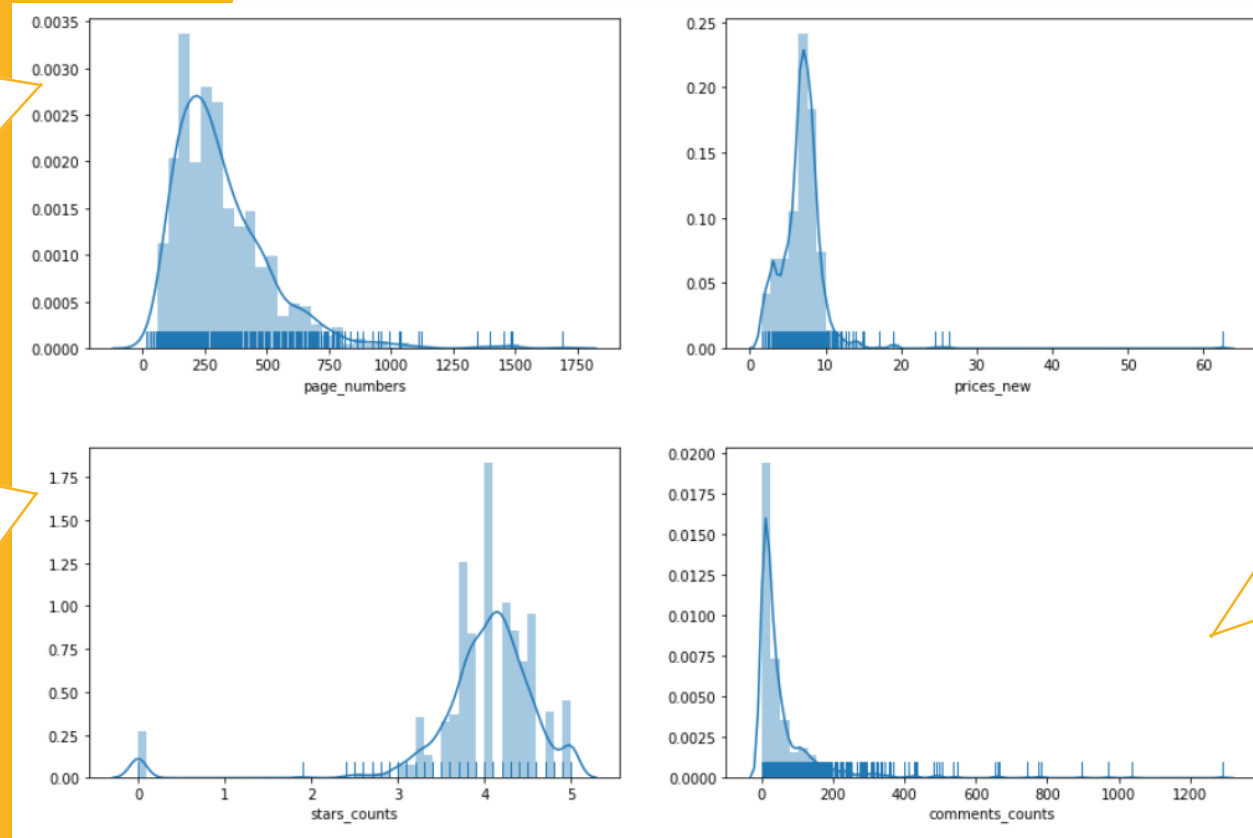
# ANALYSE UNIVARIÉE

## - VARIABLES CONTINUES

### 2. l'Histogramme

- Sauf la variable « stars\_counts », les distributions des variables continues sont étalées à droite.

La distribution du nombre de pages est concentrée entre 200 et 300, pourtant à cause des livres qui ont plus de 500 pages, l'écart standard est important (197 pages).



La plupart des livres qui ont des commentaires ont relativement de bonnes références, autour du 4 étoiles. En revanche, les livres qui ont 0 étoiles sont des livres qui n'ont pas encore ni commentaires ni classements, mais pas de mauvaises avis.

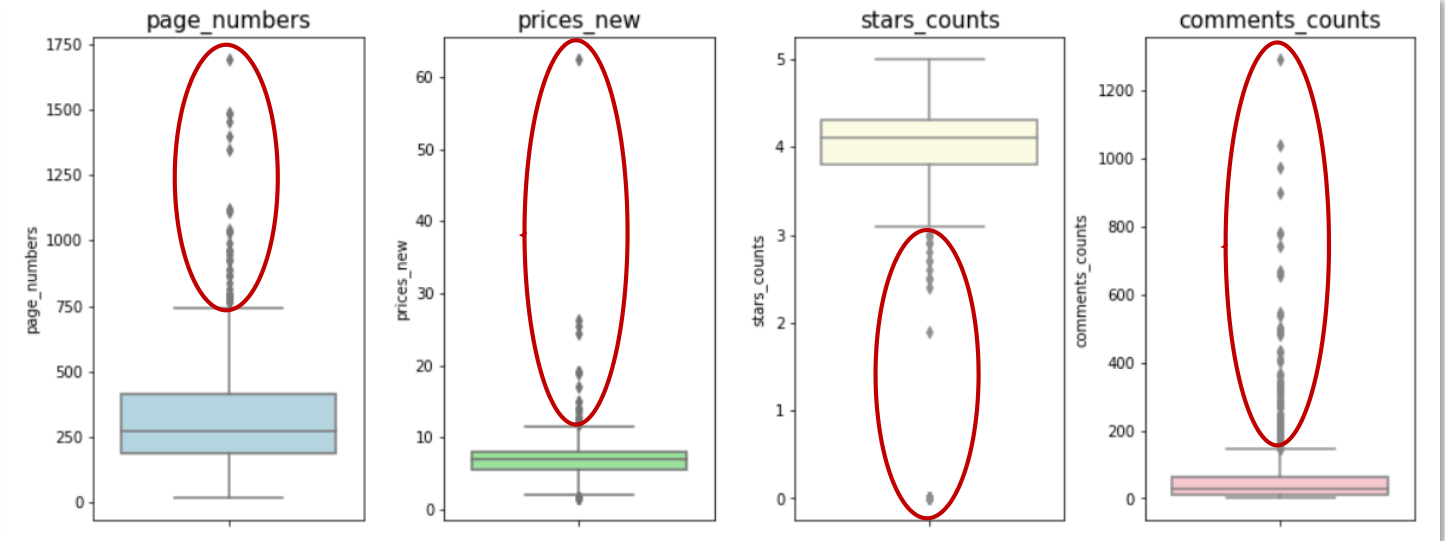
La majorité des livres ont un nombre de commentaires inférieur à 100, exceptionnellement certains ont plus de 500 commentaires.

# ANALYSE UNIVARIÉE

## - VARIABLES CONTINUES

### 3. Le boxplot

- Comme on l'a vu dans l'histogramme, la majeure partie des valeurs de chaque variable continue est concentrée dans la marge étroite.
- Dans toutes les variables, nous observons que les valeurs aberrantes ne sont pas si nombreuses mais assez extrêmes comme par exemple, les commentaires.

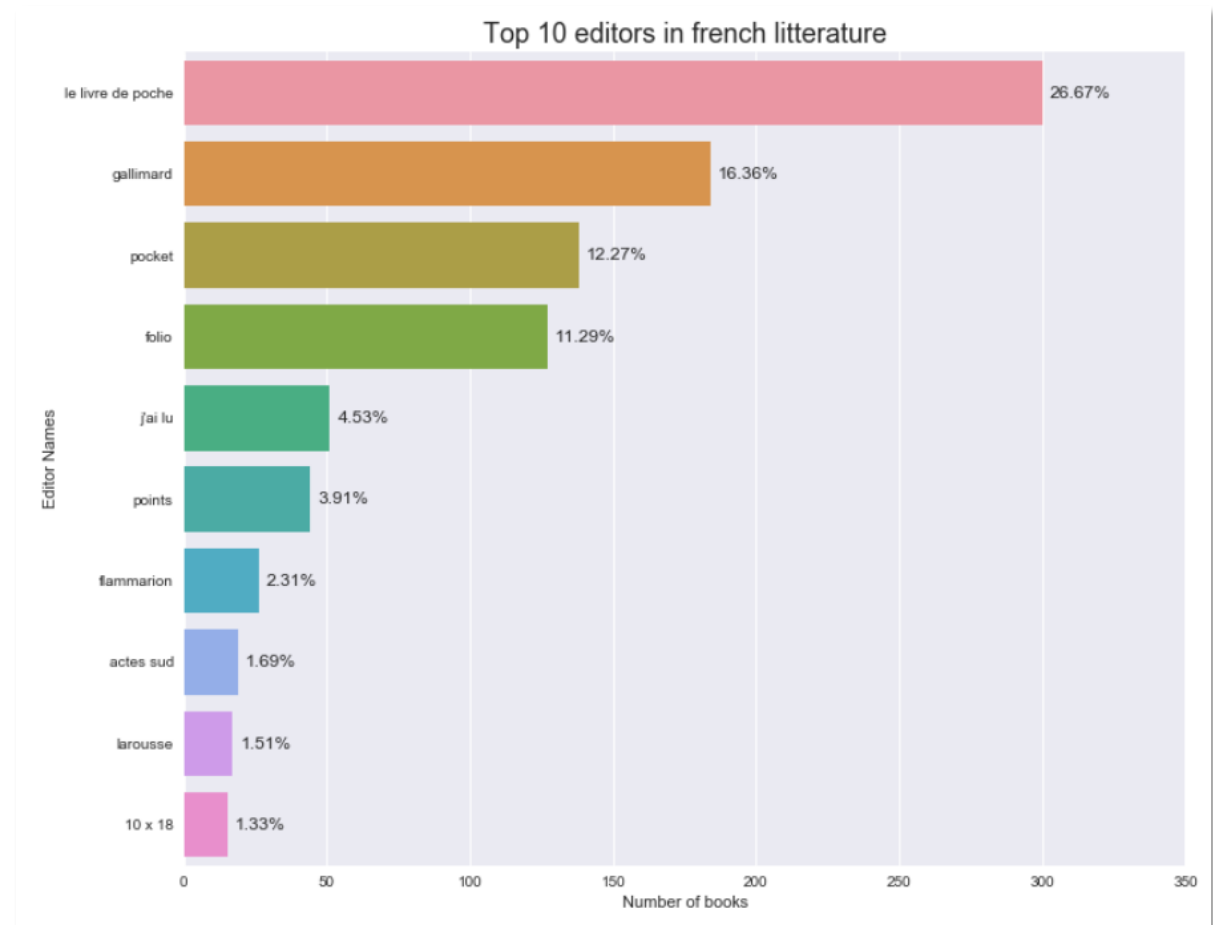


# ANALYSE UNIVARIÉE

## - VARIABLES DE CATÉGORIES

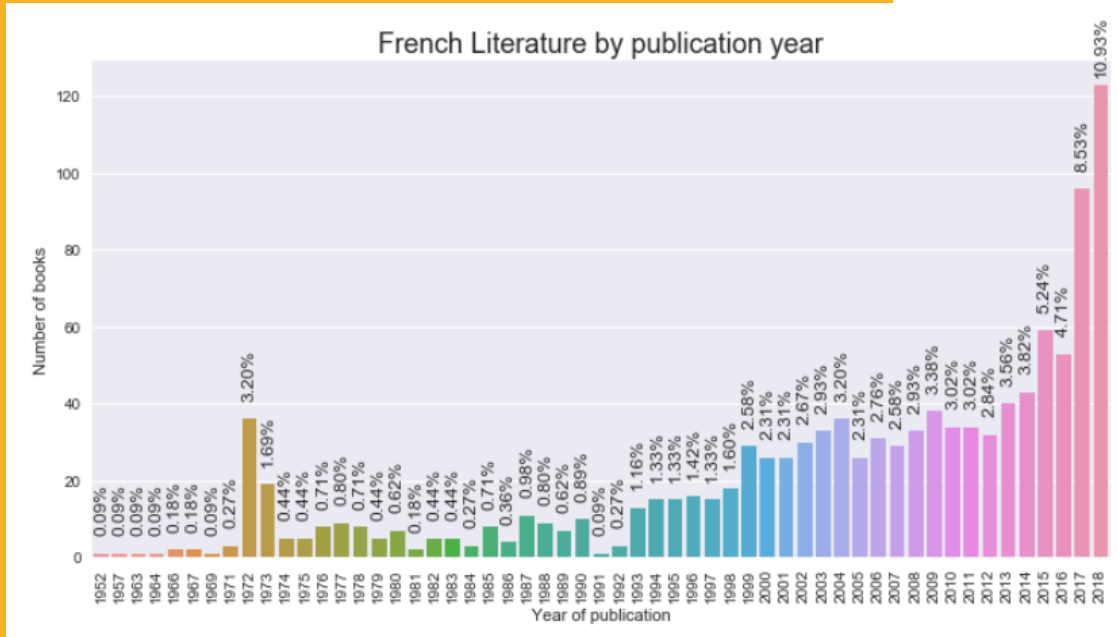
### Le countplot/barplot

- Concernant la caractéristique des variables catégories dans ces données, il y a trop de catégories différentes dans chaque variable.
- J'ai donc fait un countplot/barplot qui montre le top 10 des valeurs pour chaque variable catégorie.
- Par exemple, 26.7 % des livres ont le nom d'éditeur, 'le livre de poche'.



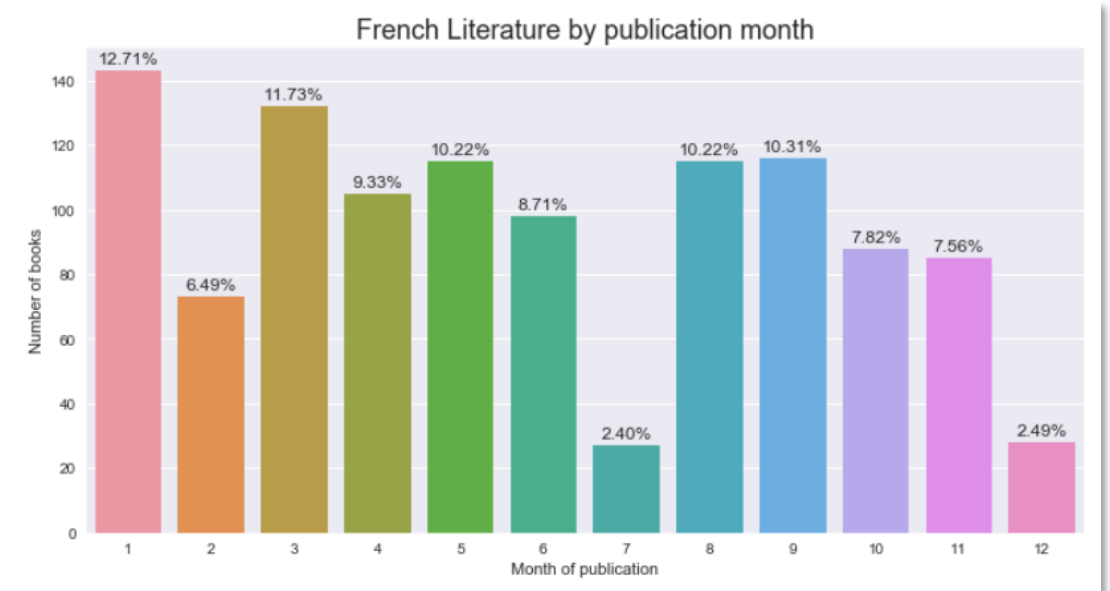
# ANALYSE UNIVARIÉE

## - VARIABLES DE TEMPS



La distribution de la variable d'année est étalée à droite. La moitié des livres sont publiés depuis 10 ans.

On n'observe pas de grand-chose mais c'est intéressant de savoir que les mois de juillet et de décembre (les mois de vacances) sont les mois qui ont le moins de nombre des livres publiés.

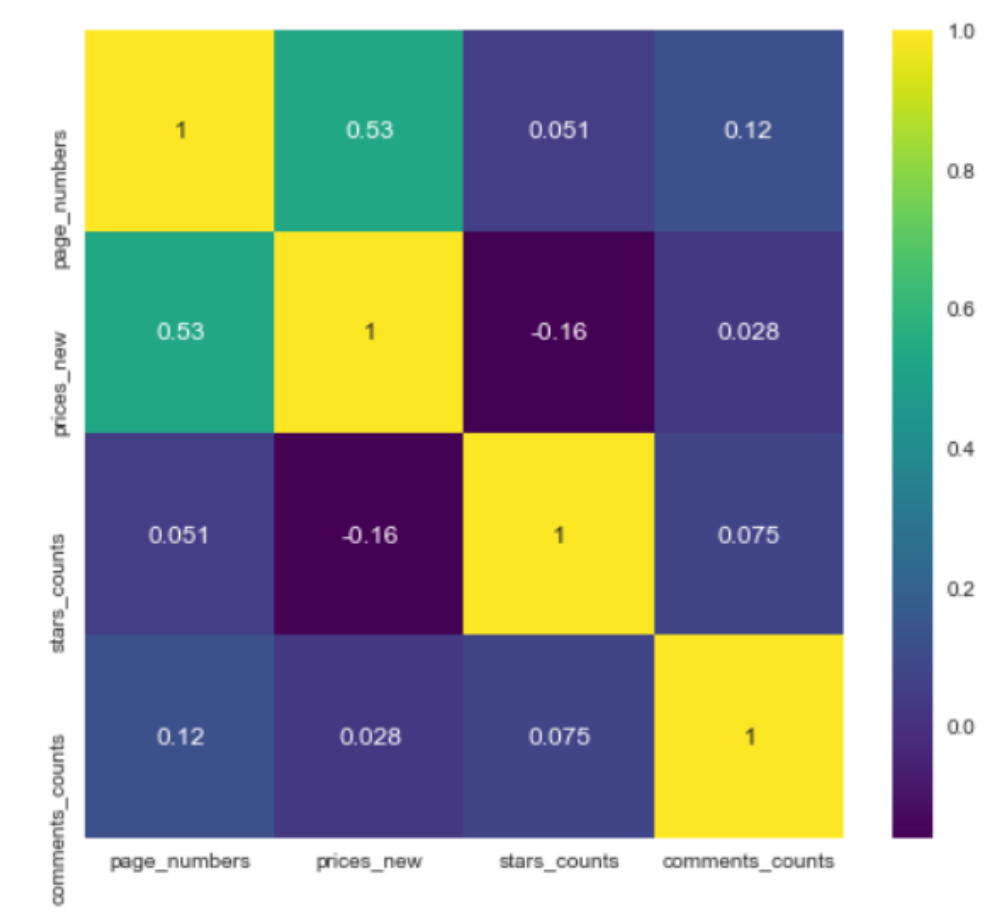


# ANALYSE BIVARIÉE

## - VARIABLES CONTINUES X CONTINUES

### La corrélation – Heatmap

- Nous observons une corrélation assez forte entre «prices\_new» et «page\_numbers» (0.53).
- Cette corrélation est logique car le plus un livre a de pages plus son prix augmente (relation linéaire positive).
- Par contre, il n'y a pas une vraie corrélation entre «prices\_new» et «stars\_counts» (-0.16) ou «comments\_counts» (0.028).
- Les étoiles étant données après fixation du prix du livre. Ces dernières n'ont pas aucune influence sur le prix d'un livre neuf.



# ANALYSE BIVARIÉE

## - VARIABLES CONTINUES X CATÉGORIES

- Les prix des livres sont assez similaires d'un mois à sur l'autre.
  - Chaque mois, la majorité des livres ont des prix inférieurs à 10 euros.
  - Les médianes des prix sont presque semblables d'un mois sur l'autre à l'exception de celles du mois juillet dont le prix médian est le plus bas.
  - Aussi les dispersions des prix de livres par mois ne sont pas larges.



# ANALYSE BIVARIÉE

## - VARIABLES CONTINUES X CATÉGORIES

- Les prix des livres du top 10 des auteurs sont assez variés.
  - Le prix moyen des livres des auteurs classiques est moins élevé que celui des auteurs contemporains.
  - Les livres d'auteurs contemporains comme Guillaume MUSSO, Françoise BOURDIN, et Marc LEVY ont les prix très définis entre 7 et 8 euros.
  - Les livres de Victor HUGO ont la plus grande dispersion de prix.

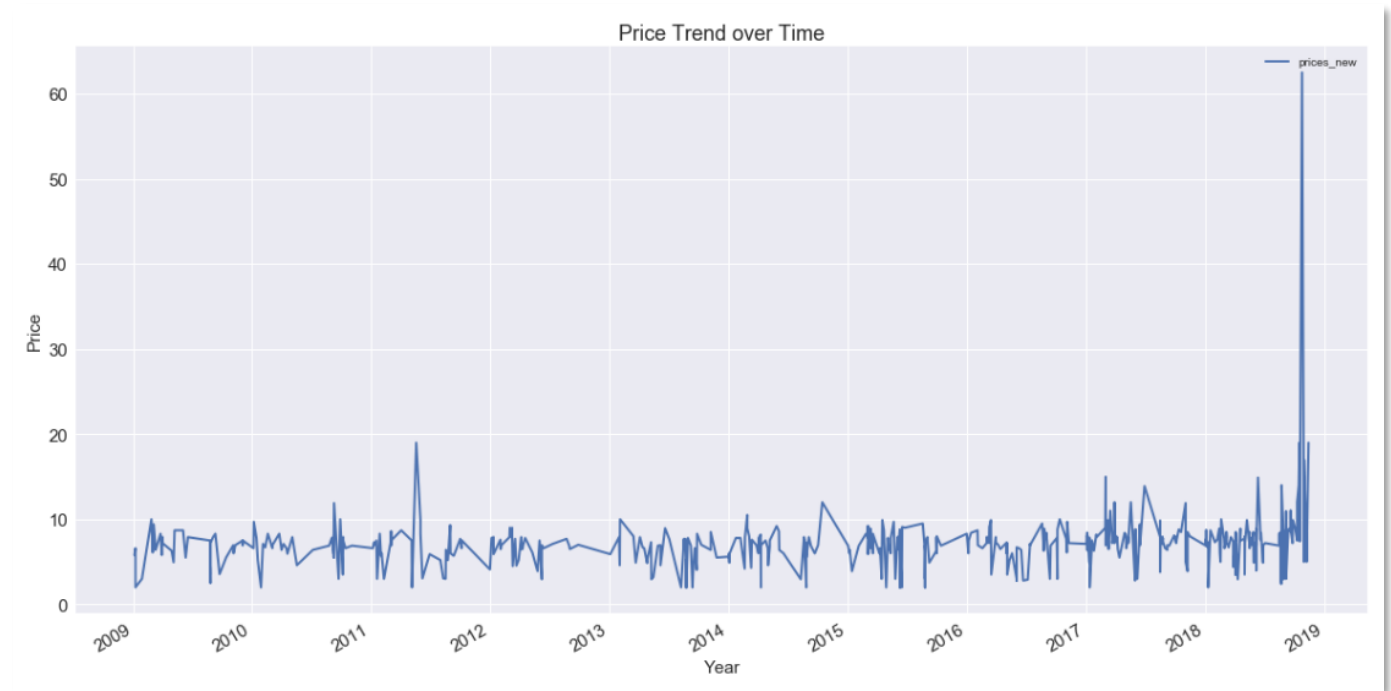


# LE SÉRIE TEMPORELLE PLOT

## - *LA TENDANCE DU PRIX DANS LE TEMPS*

### Le série temporelle plot

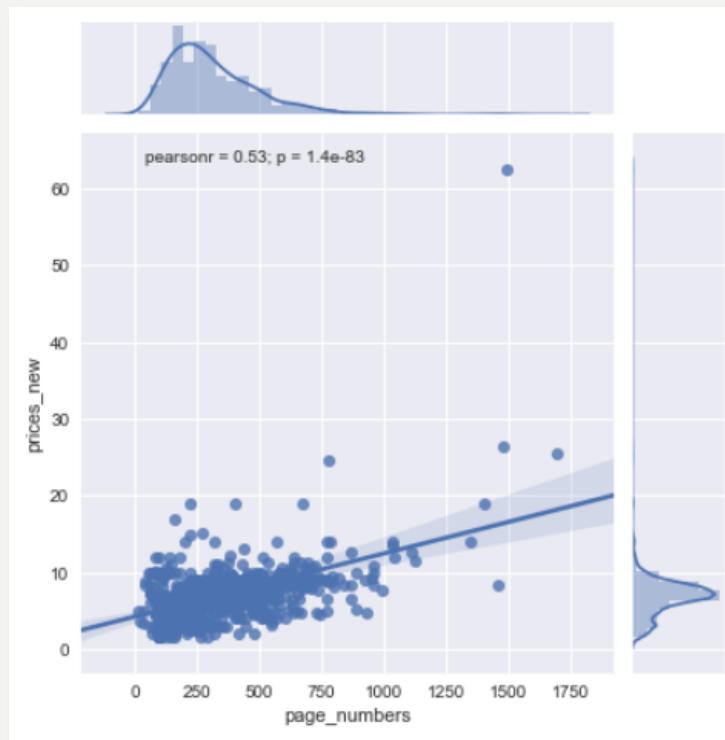
- De la moitié des livres publiés depuis 10 ans, on ne regarde que 10 dernières années afin d'étudier la tendance du prix dans le temps.
- Comme on a vu sur le boxplot de prix par mois de publication, on ne voit pas un pattern remarquable sur le plot.





# RÉGRESSION LINÉAIRE SIMPLE

## - PRIX *X* NOMBRE DE PAGES D'UN LIVRE




OLS Regression Results

|                   |                  |                     |           |       |        |        |
|-------------------|------------------|---------------------|-----------|-------|--------|--------|
| Dep. Variable:    | prices_new       | R-squared:          | 0.795     |       |        |        |
| Model:            | OLS              | Adj. R-squared:     | 0.795     |       |        |        |
| Method:           | Least Squares    | F-statistic:        | 4358      |       |        |        |
| Date:             | Fri, 09 Nov 2018 | Prob (F-statistic): | 0.00      |       |        |        |
| Time:             | 02:33:41         | Log-Likelihood:     | -2971.6   |       |        |        |
| No. Observations: | 1125             | AIC:                | 5945.     |       |        |        |
| Df Residuals:     | 1124             | BIC:                | 5950.     |       |        |        |
| Df Model:         | 1                |                     |           |       |        |        |
| Covariance Type:  | nonrobust        |                     |           |       |        |        |
|                   | coef             | std err             | t         | P> t  | [0.025 | 0.975] |
| page_numbers      | 0.0177           | 0.000               | 66.015    | 0.000 | 0.017  | 0.018  |
| Omnibus:          | 334.843          | Durbin-Watson:      | 1.839     |       |        |        |
| Prob(Omnibus):    | 0.000            | Jarque-Bera (JB):   | 10539.288 |       |        |        |
| Skew:             | 0.712            | Prob(JB):           | 0.00      |       |        |        |
| Kurtosis:         | 17.927           | Cond. No.           | 1.00      |       |        |        |

- Ce modèle explique 79.5% de la variance des prix des livres neufs (R carré)
- Toute augmentation d'une page du livre est associée à une hausse de 0.0177 euros du prix du livre avec une marge d'erreur de 5% toute chose égale par ailleurs.

# L'ANALYSE DES DONNÉES EN RÉSUMÉ

- Parmi des variables, le nombre de pages est la variable la plus influente sur le prix d'un livre neuf.
- Avec les variables suivantes, le prix d'un livre neuf ne semble pas avoir de relation avec : son titre, le nom de la maison d'édition, le nom de la collection, la date de publication, le nombre d'avis, le nombre d'étoiles.
- Pour les quelques auteurs contemporains, les Stars auteurs, il me semble que leurs livres ont des prix déterminés.
- Je pense que c'est parce que le prix d'un livre neuf varie peu en fonction de sa popularité, de la tendance, ou de la période.

A decorative wavy line in yellow and white on the left side of the image.

# **PARTE 5\_** **CONCLUSIÓN**

# LE BILAN DU PROJET

## C'est possible !

- Scraper les données sur le site d'Amazon est très difficile (pour un débutant) mais c'est possible !
- L'important est de mieux connaître la restriction imposée aux robots par le site en avance après c'est « TRY AND RETRY UNTIL IT WORKS ! »
- Avoir une bonne qualité de base de données est très important. Dans le projet, je n'ai pas pu récupérer une base de données « propres » à cause du bug, je l'ai donc traitée manuellement (12 observations), mais il vaut mieux trouver une manière qui offre les données « propres ».

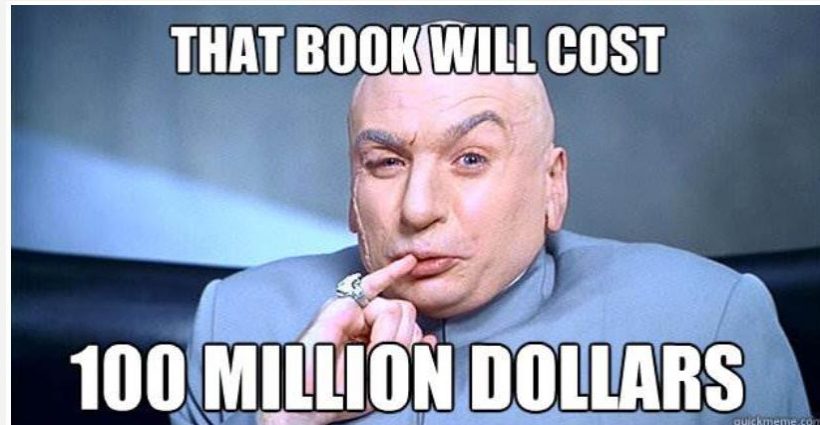
## La suite ?

- L'étape suivante possible peut être de scraper et d'analyser le prix d'un livre d'occasion afin d'en savoir plus sur l'évolution du prix d'un livre. Car le prix d'un livre neuf est déjà fixé quand il est publié alors le prix reste plutôt stable. Pourtant le prix d'un livre d'occasion peut être indépendant des commentaires, et de son classement.
- Avec les commentaires, on pourrait faire le Text Mining, l'analyse du sentiment des lecteurs pour connaître la relation entre les commentaires et les étoiles données.

# LA POSSIBLE UTILISATION DE L'ANALYSE

- Normalement le prix d'un livre est défini par plusieurs facteurs. Si vous voulez publier votre livre et l'éditer, l'éditeur fera faire, pour vous, toutes les étapes de l'impression, de la publication et la vente de votre livre bien qu'il vous demande des droits sur votre livre et vous paie des royalties.

**Mais si vous passez à l'auto-publication... ?**



- Vous n'avez aucune idée du prix correct pour votre livre.
- Dans ce cas-là, cette analyse peut vous inspirer en connaissant le prix des autres livres et ainsi fixer le prix de votre livre pour qu'il corresponde au prix du marché.