DATA ANALYTICS

“Mix So Logic”

An Ai cocktail generator, according to your available ingredients.

Stephane ‘Lucien’ Ledan

June 6th 2023

TABLE OF CONTENT

INTRODUCTION

“Mix-ologic” is a mobile and web app project targeting both young adults and more seasoned aficionados of cocktail drinking and making, also called ‘mixology’.

The COVID pandemic, its subsequent lockdowns and restrictions worldwide have impacted many in their lifestyle, habits and routines. More than just a fab or momentary change, it seems populations adjusted their lives on several points. One of the more noticeable behavioral changes regards drinking habits (as we will explore in the following section), for better or for worse.

What the data suggests is that although a majority has returned to pre-pandemic habits, with too many newly found hobbies abandoned just as fast as they were picked up. Yet, some of these behavioral changes remain in a form or another.

A consequent amount of those who picked up cocktail mixing, by passion or curiosity, have continued in that manner. Enough so that reports such as the Bacardi Trend[4] report & CGA research[3, 5] (a leading data & insight consultancy firm) picked up on the trend and insists on its impact beyond the pandemic lockdowns & restrictions.

This project proposes an algorithm that recommends compatible ingredients with the one inputted by the user. Therefore either building a cocktail from scratch wth your available ingredients, or recommending a known recipe. To best match the list of user ingredients, a database of 4500+ cocktail recipes and their variations has been compiled.

The following pages will further explore the market trends, the extracted dataset and the process of ETL.

MARKET TREND

"For many people, pandemic lockdowns amplified aspects of creating coziness..."

- Brandy Rand, chief strategy officer, IWSR Drinks Market Analysis.[1]

"Liquor Brands Bet Thrifty Drinkers Will Keep Making At-Home Cocktails"

-Joshua Kirby, for The Wall Street Journal, Jan. 7, 2023[2]

CGA Strategy, May 2021:[3]

"47% of at-home cocktail drinkers plan to continue mixing their own drinks as often as during the pandemic period, while 8% will be making them more often. As such, the at home cocktail occasion still represents a significant opportunity for suppliers. "

*"CGA research has revealed how consumers' general alcohol consumption remains steady and **one in four (24%) now spending more on home drinking** than they did before the pandemic, there is a fresh wave of interest in recreating drinks like cocktails that might previously have only been bought out of home."* [5]

Bacardi Trends Report 2023:[4]

- The Bacardi Consumer Survey 2022 **reveals that 40% of respondents in the U.S.**, and more than **30% of those in the United Kingdom**, are choosing to make more cocktails at home in 2022 compared to 2020.
- This is **upheld by almost 30%** of respondents across the U.S., U.K., Mexico, India, and South Africa, who say **they have upped their cocktail knowledge** over the past two years.
- Google Search volume for "cocktails" went **up by 59%** between October 2021 and September 2022.

AI in cocktail making:

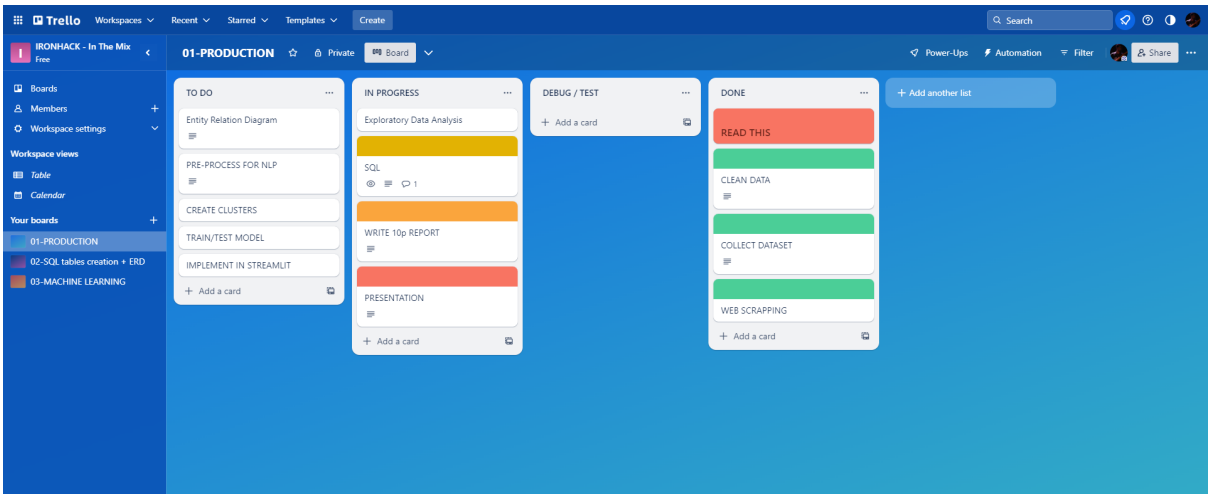
Minga Box, an Israeli robotics company has released a robot bartender, showing that there is interest in the domain of automatizing cocktail creation: https://cecilia.ai/

PRODUCTION PIPELINE

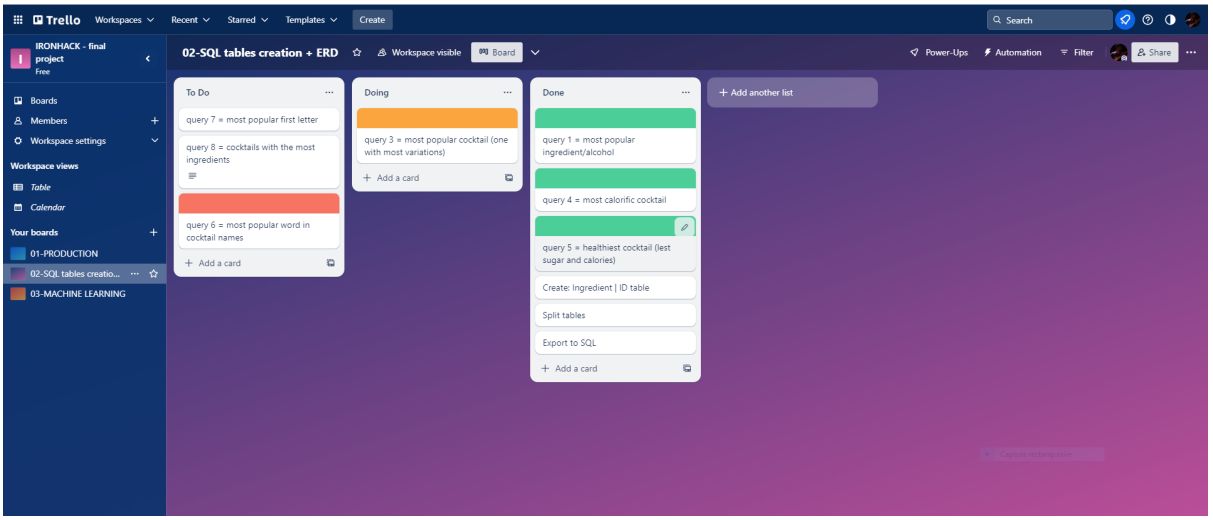
01. **Step planification with kanban tool**
02. **Evaluate data availability & best extraction manner**
03. **Data extraction (web scraping)**
04. **Data wrangling with Python**
05. **Exploring data with MySQL**
06. **Export relevant tables & ERD**
07. **Data visualization with Matplotlib & Seaborn**
08. **Pre-process data for machine learning modeling**
09. **Train & test model**
10. **Embed algorithm into StreamLit app**

PROJECT MANAGEMENT

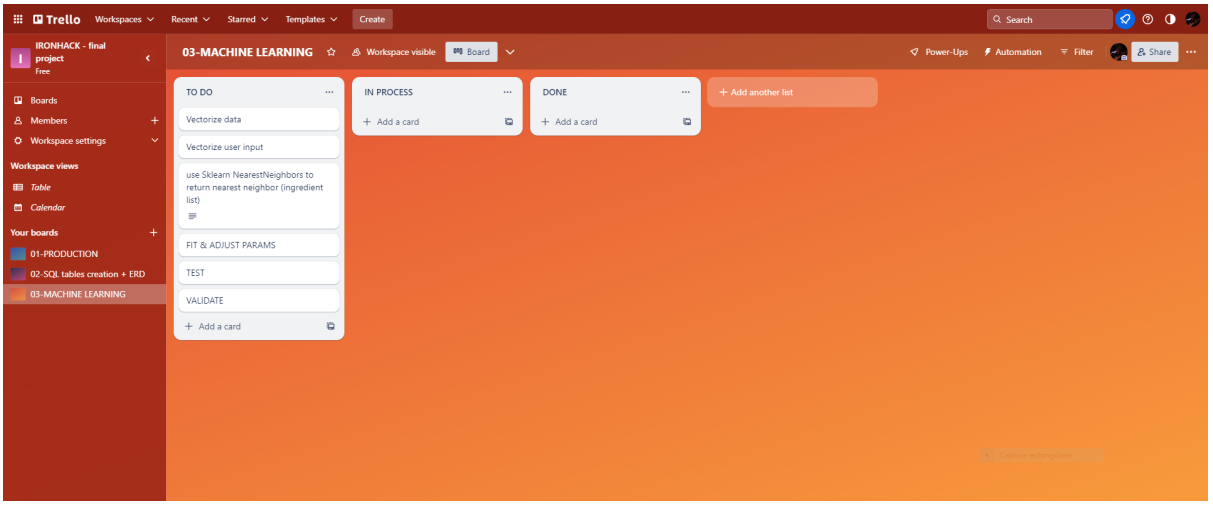
Main kanban for overall production:



SQL process kanban:



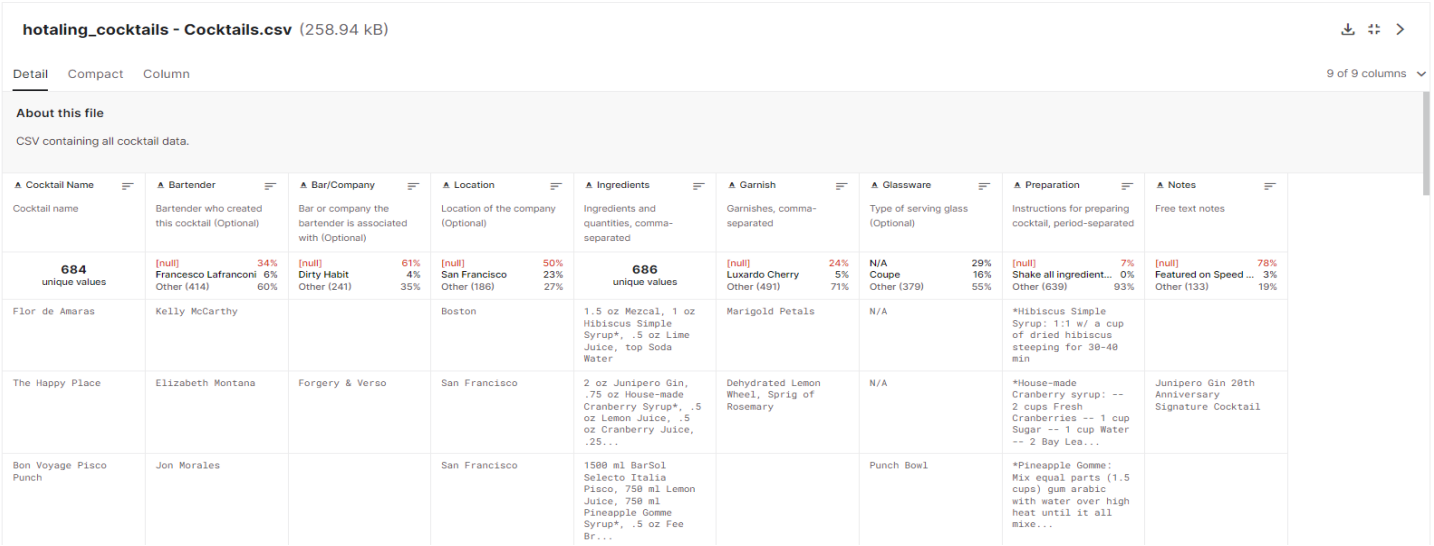
ML model kanban:



DATA EXTRACTION

The main objective is to retrieve a maximum of cocktails and their variations and specifically their ingredient list since we're looking to match these with a user input.
Other interesting data points are : description, recipe instructions, video tutorial, nutritional facts & preparation time.

After searching for existing databases, the most complete and clean one, featured on Kaggle, accounts for 600 cocktail recipes. As seen below, it also features a number of other pieces of information that are albeit interesting but none-vital to my project.



hotaling_cocktails - Cocktails.csv (258.94 kB)

Detail | Compact | Column — 9 of 9 columns

About this file

CSV containing all cocktail data.

Cocktail Name	Bartender	Bar/Company	Location	Ingredients	Garnish	Glassware	Preparation	Notes
Cocktail name	Bartender who created this cocktail (Optional)	Bar or company the bartender is associated with (Optional)	Location of the company (Optional)	Ingredients and quantities, comma-separated	Garnishes, comma-separated	Type of serving glass (Optional)	Instructions for preparing cocktail, period-separated	Free text notes
684 unique values	[null] 34% Francesco Lafranconi 6% Other (414) 60%	[null] 61% Dirty Habit 4% Other (241) 35%	[null] 50% San Francisco 23% Other (186) 27%	686 unique values	[null] 24% Luxardo Cherry 5% Other (491) 71%	N/A 29% Coupe 16% Other (379) 55%	[null] 7% Shake all ingredient... 0% Other (639) 93%	[null] 78% Featured on Speed ... 3% Other (133) 19%
Flor de Amaras	Kelly McCarthy		Boston	1.5 oz Mezcal, 1 oz Hibiscus Simple Syrup*, .5 oz Lime Juice, top Soda Water	Marigold Petals	N/A	*Hibiscus Simple Syrup: 1:1 w/ a cup of dried hibiscus steeping for 30-40 min	
The Happy Place	Elizabeth Montana	Forgery & Verso	San Francisco	2 oz Junipero Gin, .75 oz House-made Cranberry Syrup*, .5 oz Lemon Juice, .5 oz Cranberry Juice, .25...	Dehydrated Lemon Wheel, Sprig of Rosemary	N/A	*House-made Cranberry syrup: -- 2 cups Fresh Cranberries -- 1 cup Sugar -- 1 cup Water -- 2 Bay Lea...	Junipero Gin 20th Anniversary Signature Cocktail
Bon Voyage Pisco Punch	Jon Morales		San Francisco	1500 ml BarSol Selecto Italia Pisco, 750 ml Lemon Juice, 750 ml Pineapple Gomme Syrup*, .5 oz Fee Br...		Punch Bowl	*Pineapple Gomme: Mix equal parts (1.5 cups) gum arabic with water over high heat until it all mixe...	

Since the project revolves around a low or disparate list of ingredients as input, I wanted to retrieve more recipes to optimize matching and give the user more options.

Scouting the internet, I've found https://drinklab.org/ which compiles over 5000 cocktail recipes. Using the BeautifulSoup library, I've created a series of loops and functions to scrap the cocktail name, its recipe, ingredient list, description and video link.

Scrapping process :

1. Extract all urls to 'Cocktails by Letter' pages into a list.
2. Iterate in the returned url list to scrap and store locally every .html file for each individual cocktail page.
   The website loads slowly (GTMetrix score = D), to avoiding making a request for each element to scrap, I stored the files locally so that there is no website loading time.
3. Information extraction loop through the stored .html files.
   Cocktail name, description, recipe, ingredients, video link and nutrient list were extracted and stored as dataframe and then .csv file.
4. For each step, a test sequence was run on a separate notebook, and then again in the compiled notebook.

Python code extracts:

‘Cocktails by Letter’ urls extraction:

```python
cocktails_by_letter = soup.find_all('div', class_="x-text e89516-4 m1x2k-3")

url_by_letter = []

for link in cocktails_by_letter:
    anchors = link.find_all('a')
    for anchor in anchors:
        url = anchor['href']
        url_by_letter.append(url)


url_by_letter
```

Python

Individual cocktail pages urls extraction:

```python
cocktail_urls = []

for url in url_by_letter:
    base_url = url
    start_page = 1

    while True:
        url = base_url + "?_page=" + str(start_page)
        response = requests.get(url)
        soup = BeautifulSoup(response.content, 'html.parser')
        divs = soup.find_all('div', class_="col-md-3 col-sm-4 col-xs-6 pt-cv-content-item pt-cv-1-col")


        temp_urls = []

        for div in divs:
            anchor = div.find('a')
            if anchor is not None and 'href' in anchor.attrs:
                temp_urls.append(anchor["href"])

        cocktail_urls.extend(temp_urls)

        pagination = soup.find('ul', class_="pt-cv-pagination")
        if pagination is None or 'data-totalpages' not in pagination.attrs:
            # print('Next subpage')
            break

        total_pages = int(pagination['data-totalpages'])
        if start_page >= total_pages:
            # print('Next page')
            break

        start_page += 1

    # print(cocktail_urls)

```

Python

Individual cocktail pages html extraction:

```python
def get_soups(url_list):

    cocktail_htmls = []
    file_counter = 0
    file_path = 'C:/Users/User/Desktop/FORMATION/IRONHACK/PROJECTS/PROJECT-FINAL/html_soup'

    for url in tqdm_notebook(url_list):

        response = requests.get(url)

        soup = BeautifulSoup(response.content, 'lxml')

        cocktail_htmls.append(soup)

        filename = f'cocktail_{file_counter}.html'

        file_full_path = os.path.join(file_path, filename)

        with open(file_full_path, 'w', encoding='utf-8') as file:
            file.write(str(soup))

        file_counter += 1
```

Python

Cocktail information extraction:

```python
def scrap_info(x):

    extracted_info = []

    for file_name in tqdm_notebook(os.listdir(x)):
        if file_name.endswith('.html'):
            file_full_path = os.path.join(x, file_name)

            with open(file_full_path, 'r', encoding='utf-8') as file:
                soup = BeautifulSoup(file, 'html.parser')

                # name scrap
                name = soup.find('h1', class_="wprm-recipe-name wprm-block-text-bold")
                if name is not None:
                    ctl_name = name.text
                else:
                    ctl_name = None
                if ctl_name is None:
                    continue

                # description scrap
                description = soup.find('div', class_="wprm-recipe-summary wprm-block-text-normal")

                if description is not None:
                    ctl_description = description.text

                else:
                    ctl_description = "Unfortunately, we have no description for this drink... You'll have to describe it yourself!"

                # recipe scrap
                recipe = soup.find('div', class_="wprm-recipe-instruction-group")
                if recipe is not None:
                    ctl_recipe = recipe.text
                else:
                    ctl_recipe = "Woops... We couldn't retrieve the exact recipe... It's trial & error time! Just a little more fun before enjoying a nice drink!"

                # ingredients scrap
                ingredients = soup.find('div', class_="wprm-recipe-ingredient-group")
                if ingredients is not None:
                    ctl_ingredients = ingredients.text.replace('□', ' |').replace('  ', ':')
                else:
                    ctl_ingredients = "Woops... What happened ?! Something didn't work, please try again."

                # video link scrap
                video = soup.find('iframe')
                if video is not None:
                    ctl_vid = video['src']
                else:
                    ctl_vid = "There doesn't seem to be an instructional video for this cocktail. Why not make the tutorial yourself!"


                # nutrition scrap
                nutrition = soup.find('div', class_="wprm-nutrition-label-container wprm-nutrition-label-container-simple wprm-block-text-normal")
                if nutrition is not None:
                    ctl_nutrition = nutrition.text
                else:
                    ctl_nutrition = "There doesn't seem to be any nutritional facts associated to this cocktail. But we know that's not why you're here..."

                extracted_info.append([ctl_name, ctl_description, ctl_recipe, ctl_ingredients, ctl_nutrition, ctl_vid])

    df = pd.DataFrame(extracted_info, columns=['Name', 'Description', 'Recipe', 'Ingredients', 'Nutrition Facts', 'Video Link'])

    return df
```

DATA WRANGLING & CLEANING

Using BeautifulSoup.text I retrieved the relevant strings. The most obvious artifacts were dealt with directly in the extraction loops in a manner that would facilitate the cleaning:

```
# ingredients scrap
ingredients = soup.find('div', class_="wprm-recipe-ingredient-group")
if ingredients is not None:
    ctl_ingredients = ingredients.text.replace('□', ' |').replace('  ', ':')
else:
    ctl_ingredients = "Woops... What happened ?! Something didn't work, please try again."
```

The extracted raw data had the following format :

	Name	Description	Recipe	Ingredients	Nutrition Facts	Video Link
0	American Beauty Cocktail	Unfortunately, we have no description for this drink... You'll have to describe it yourself!	Woops... We couldn't retrieve the exact recipe... It's trial & error time! Just a little more fun before enjoying a nice drink!	| 1 oz: Brandy | .5 oz: Dry Vermouth | .25 oz: White Creme De Menthe | 1 oz: Orange Juice | 1 tsp: Grenadine Syrup | 1 oz: Red Port	Calories: 185kcal | Carbohydrates: 14g | Protein: 0.3g | Fat: 0.1g | Saturated Fat: 0.01g | Polyunsaturated Fat: 0.02g | Monounsaturated Fat: 0.01g | Sodium: 6mg | Potassium: 89mg | Fiber: 0.1g | Sugar: 10g | Vitamin A: 50IU | Vitamin C: 20.6mg	There doesn't seem to be an instructional video for this cocktail. Why not make the tutorial yourself!

Missing values were replaced by comments inciting the user to experiment and have a good time nonetheless since the end usage is not for professional, but recreational use.

Duplicate names of cocktails were dropped, and for ‘Ingredients’ and ‘Nutritional Facts’, the strings were split into lists, to be exploded and then loaded into specific dataframes for further EDA and manipulation.

Ingredient values to list:

```
ingredient_dict = []

for row in temp_df['ingredients']:

    ingredient_row = []
    ingredients = row.split('|')

    for ingredient in ingredients:
        ingredient = ingredient.replace(':oz',' oz')
        parts = ingredient.split(':')

        if len(parts) >= 2:
            quantity = parts[0].strip()
            ingredient_name = parts[1].strip()
            ingredient_row.append((quantity, ingredient_name))

    ingredient_dict.append(ingredient_row)

print(ingredient_dict)
```

Outputs are collapsed ···

```
ingredient_dict[0]
```

```
[('1 oz', 'Brandy'),
 ('.5 oz', 'Dry Vermouth'),
 ('.25 oz', 'White Creme De Menthe'),
 ('1 oz', 'Orange Juice'),
 ('1 tsp', 'Grenadine Syrup'),
 ('1 oz', 'Red Port')]
```

Exploding ingredients (same process for nutrients) column:

```
df_exploded_cal = unique_ctl_df.explode('nutrition_facts')
df_exploded_cal['cocktail_id'] = df_exploded_cal.index
df_exploded_cal.head(20)
```

Python

	name	description	recipe	ingredients	nutrition_facts	video_link	calories_count	cocktail_id
0	American Beauty Cocktail	Unfortunately, we have no description for this drink... You'll have to describe it yourself!	Woops... We couldn't retrieve the exact recipe... It's trial & error time! Just a little more fun before enjoying a nice drink!	[(1 oz, Brandy), (.5 oz, Dry Vermouth), (.25 oz, White Creme De Menthe), (1 oz, Orange Juice), (1 tsp, Grenadine Syrup), (1 oz, Red Port)]	(Calories, 185kcal)	There doesn't seem to be an instructional video for this cocktail. Why not make the tutorial yourself!	(Calories, 185kcal)	0
0	American Beauty Cocktail	Unfortunately, we have no description for this drink... You'll have to describe it yourself!	Woops... We couldn't retrieve the exact recipe... It's trial & error time! Just a little more fun before enjoying a nice drink!	[(1 oz, Brandy), (.5 oz, Dry Vermouth), (.25 oz, White Creme De Menthe), (1 oz, Orange Juice), (1 tsp, Grenadine Syrup), (1 oz, Red Port)]	(Carbohydrates, 14g)	There doesn't seem to be an instructional video for this cocktail. Why not make the tutorial yourself!	(Calories, 185kcal)	0
0	American Beauty Cocktail	Unfortunately, we have no description for this drink... You'll have to describe it yourself!	Woops... We couldn't retrieve the exact recipe... It's trial & error time! Just a little more fun before enjoying a nice drink!	[(1 oz, Brandy), (.5 oz, Dry Vermouth), (.25 oz, White Creme De Menthe), (1 oz, Orange Juice), (1 tsp, Grenadine Syrup), (1 oz, Red Port)]	(Protein, 0.3g)	There doesn't seem to be an instructional video for this cocktail. Why not make the tutorial yourself!	(Calories, 185kcal)	0
0	American Beauty Cocktail	Unfortunately, we have no description for this drink... You'll have to describe it yourself!	Woops... We couldn't retrieve the exact recipe... It's trial & error time! Just a little more fun before enjoying a nice drink!	[(1 oz, Brandy), (.5 oz, Dry Vermouth), (.25 oz, White Creme De Menthe), (1 oz, Orange Juice), (1 tsp, Grenadine Syrup), (1 oz, Red Port)]	(Fat, 0.1g)	There doesn't seem to be an instructional video for this cocktail. Why not make the tutorial yourself!	(Calories, 185kcal)	0
0	American Beauty Cocktail	Unfortunately, we have no description for this drink... You'll have to describe it yourself!	Woops... We couldn't retrieve the exact recipe... It's trial & error time! Just a little more fun before enjoying a nice drink!	[(1 oz, Brandy), (.5 oz, Dry Vermouth), (.25 oz, White Creme De Menthe), (1 oz, Orange Juice), (1 tsp, Grenadine Syrup), (1 oz, Red Port)]	(Saturated Fat, 0.01g)	There doesn't seem to be an instructional video for this cocktail. Why not make the tutorial yourself!	(Calories, 185kcal)	0
0	American Beauty Cocktail	Unfortunately, we have no description for this drink... You'll have to describe it yourself!	Woops... We couldn't retrieve the exact recipe... It's trial & error time! Just a little more fun before enjoying a nice drink!	[(1 oz, Brandy), (.5 oz, Dry Vermouth), (.25 oz, White Creme De Menthe), (1 oz, Orange Juice), (1 tsp, Grenadine Syrup), (1 oz, Red Port)]	(Polyunsaturated Fat, 0.02g)	There doesn't seem to be an instructional video for this cocktail. Why not make the tutorial yourself!	(Calories, 185kcal)	0

Exporting to MySQL :

These exploded columns allowed for the creation of specific dataframes. The results exported into MySQL were the following :

cocktails_id table:

	cocktail_id	cocktail	description	recipe	video_link
0	0	American Beauty Cocktail	Unfortunately, we have no description for this drink... You'll have to describe it yourself!	Woops... We couldn't retrieve the exact recipe... It's trial & error time! Just a little more fun before enjoying a nice drink!	There doesn't seem to be an instructional video for this cocktail. Why not make the tutorial yourself!
1	1	Azzuro	Unfortunately, we have no description for this drink... You'll have to describe it yourself!	Shake and strain into an ice-filled collins glass. and garnish with fruit.	There doesn't seem to be an instructional video for this cocktail. Why not make the tutorial yourself!
2	2	Apple Fairy	Unfortunately, we have no description for this drink... You'll have to describe it yourself!	For this recipe. make some Juice - Apple Juice ice cubes before you start. Then add 3 Juice - Apple Juice cubes to a cocktail glass. Add absinthe and apple Vodka. and fill with cider.	There doesn't seem to be an instructional video for this cocktail. Why not make the tutorial yourself!
3	3	Black Rose Bacardi	Unfortunately, we have no description for this drink... You'll have to describe it yourself!	Shake or stir. pour it into a coctail glass. add some crushed ice and serve.	There doesn't seem to be an instructional video for this cocktail. Why not make the tutorial yourself!

ingredients_id table:

	ingredient_id	ingredients
0	0	
1	1	Pernod
2	2	Coffee
3	3	Peach Papaya Juice
4	4	Whipping Cream

cocktail_ingredient_id table:

	cocktail_id	ingredient_id	quantity
0	0	366	1 oz
1	0	797	.5 oz
2	0	467	.25 oz
3	0	18	1 oz
4	0	457	1 tsp

cocktail_nutrient table:

	cocktail_id	name	nutrient	amount
0	0	American Beauty Cocktail	Calories	185kcal
1	0	American Beauty Cocktail	Carbohydrates	14g
2	0	American Beauty Cocktail	Protein	0.3g
3	0	American Beauty Cocktail	Fat	0.1g
4	0	American Beauty Cocktail	Saturated Fat	0.01g

word_use_count table:

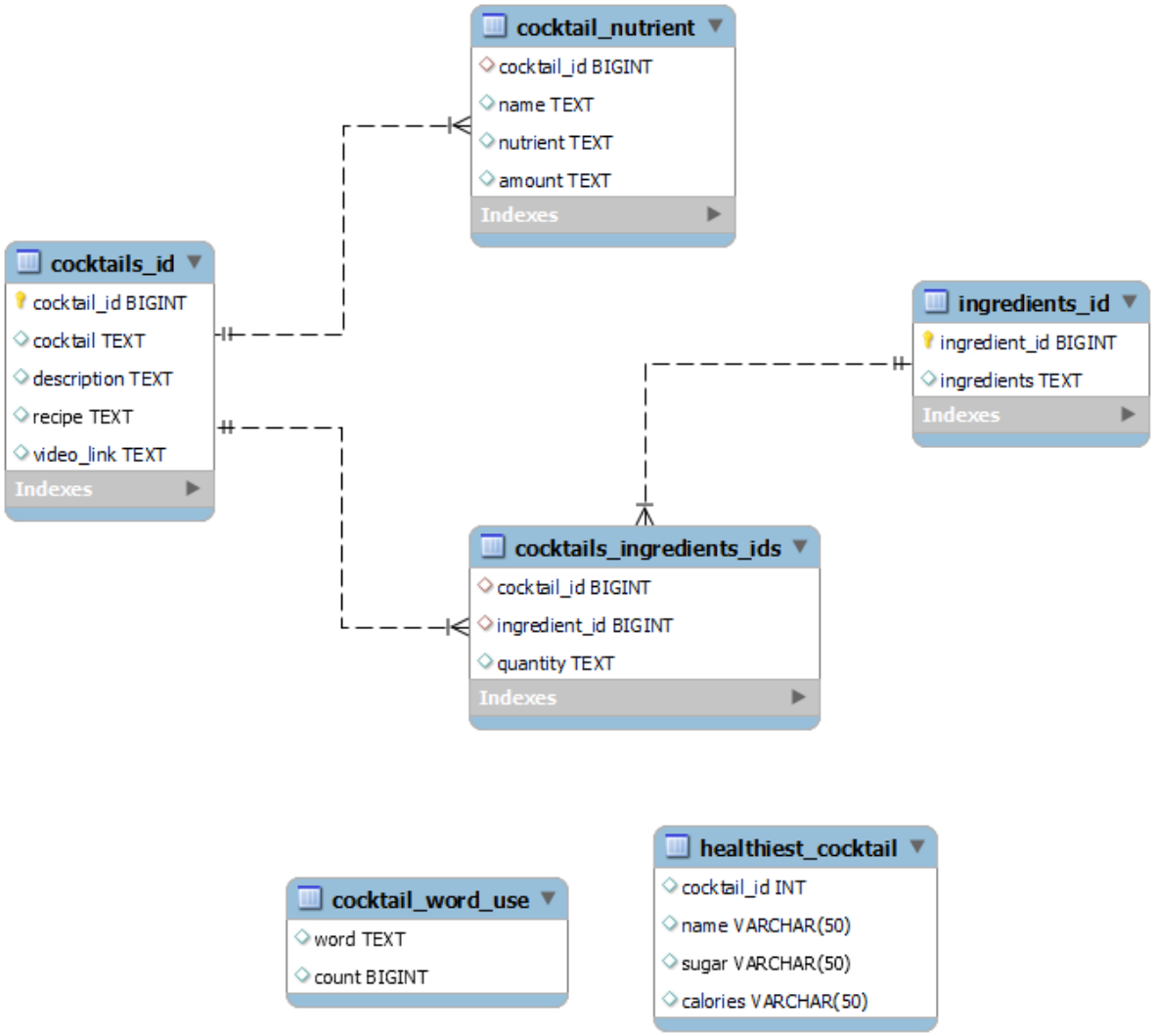
	name	cocktail_id
0	American	0
1	Beauty	0
2	Cocktail	0
3	Azzuro	1
4	Apple	2

The previous tables were imported into MySQL Workbench, where I used them to extract some “Top” lists. These views were exported into CSV and re-imported to VSCode for some plotting.

Creating those CSVs allowed me to have the liberty to plot either in Tableau or using Seaborn and Matplotlib. Due to timing issues, I’ve opted for plotting in Python.

The next section showcases the ERD and some of the queries used to create those CSVs.

ENTITY RELATIONSHIP DIAGRAM



SQL MANIPULATION

Below are some queries from which I drew insights and CSV files to further explore the data and create visualizations.

Top 20 most caloric cocktails:

```sql
SELECT name, nutrient, amount
FROM cocktail_nutrient
WHERE nutrient = 'Calories'
ORDER BY CAST(REPLACE(amount, 'kcal', '') AS DECIMAL) DESC
LIMIT 20;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

name	nutrient	amount
Apples and Oranges Martini	Calories	2036kcal
414SS Daiquiri	Calories	1552kcal
Blueberry Infused Vodka Lemonade	Calories	1529kcal
3 for a Dollar Special	Calories	1525kcal
Chocolate Sin Cocktail	Calories	1511kcal
Halloween Eyeball Jello Shots	Calories	1240kcal
The Panem-anian Soother	Calories	1136kcal

Top 20 healthiest cocktails:

```sql
CREATE TABLE IF NOT EXISTS healthiest_cocktail (
    cocktail_id INT,
    name VARCHAR(50),
    sugar VARCHAR(50),
    calories VARCHAR(50)
);

INSERT INTO healthiest_cocktail (cocktail_id, name, sugar, calories)
SELECT cn1.cocktail_id, cn1.name, cn1.amount, cn2.amount
FROM (
    SELECT name, cocktail_id , amount
    FROM cocktail_nutrient
    WHERE nutrient = 'Sugar'
    ORDER BY CAST(REPLACE(amount, 'g', '') AS DECIMAL) DESC
    /**LIMIT 100**/
) cn1
JOIN (
    SELECT cocktail_id, name, amount
    FROM cocktail_nutrient
    WHERE nutrient = 'Calories'
    ORDER BY CAST(REPLACE(amount, 'kcal', '') AS DECIMAL) DESC
    /**LIMIT 100**/
) cn2
ON cn1.name = cn2.name;

SELECT * FROM healthiest_cocktail
LIMIT 20;

SELECT * FROM healthiest_cocktail
ORDER BY CAST(REPLACE(calories, 'kcal', '') AS DECIMAL) ASC;
```

Cocktails with the most ingredients:

```sql
SELECT c.cocktail_id, c.cocktail, COUNT(i.ingredient_id) AS ingredient_count
FROM cocktails_id c
INNER JOIN cocktails_ingredients_ids i ON c.cocktail_id = i.cocktail_id
GROUP BY c.cocktail_id, c.cocktail
ORDER BY ingredient_count DESC;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

cocktail_id	cocktail	ingredient_count
2942	Jungle Juice	13
12	Burning Pine Needles	10
259	Bleeding Weasel	10
270	Blood Mary Extra Hairy	10
1353	Caribbean Smoked Torch	10
1408	Cinnamon Bloody Mary	10
2003	Eric`s Bloody Bull	10
2244	Fall Spice Cordial	10

Most popular first letter:

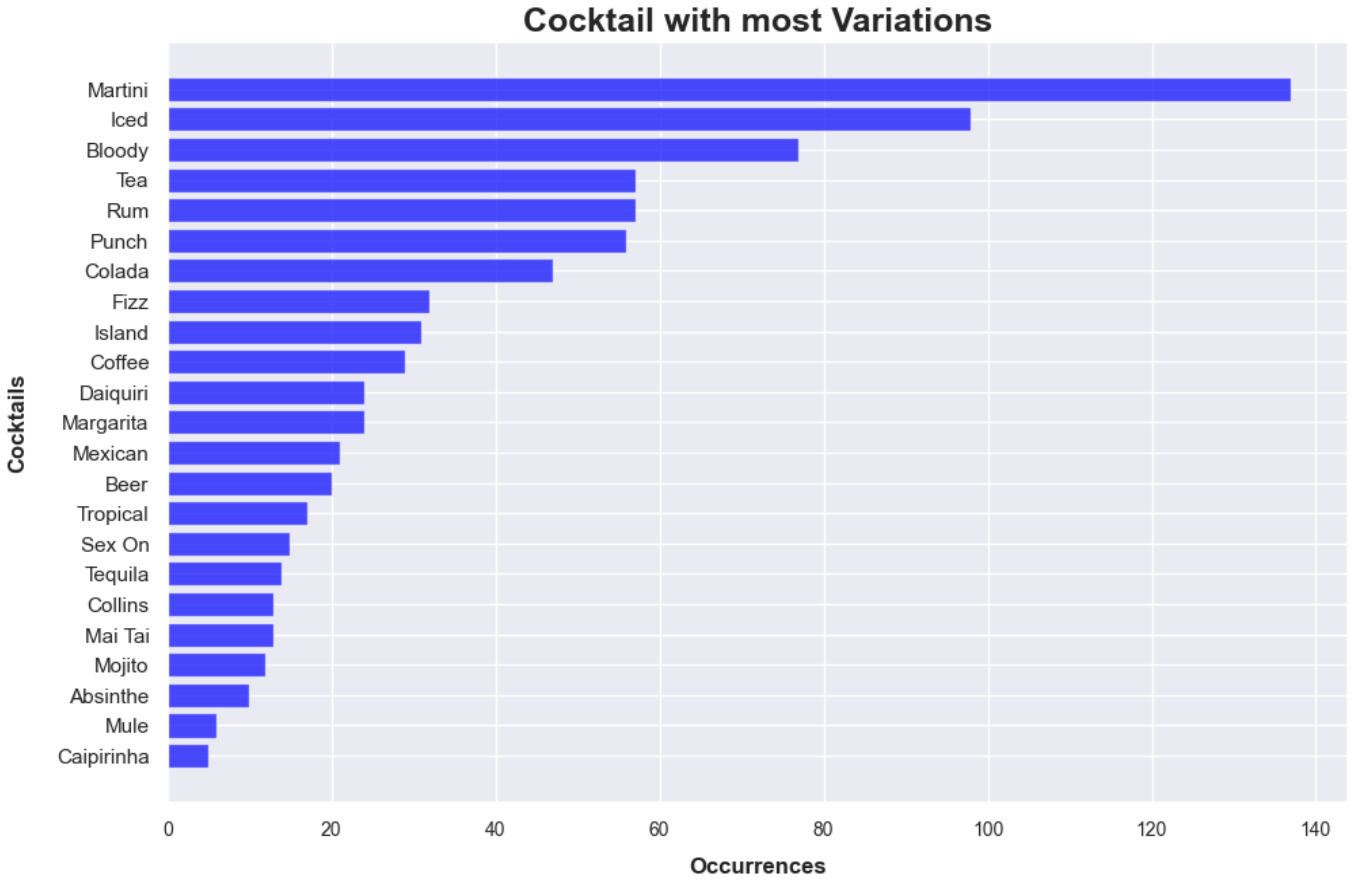
```sql
SELECT SUBSTRING(cocktail, 1, 1) AS first_letter, COUNT(*) AS count
FROM cocktails_id
GROUP BY first_letter
ORDER BY count DESC;
```

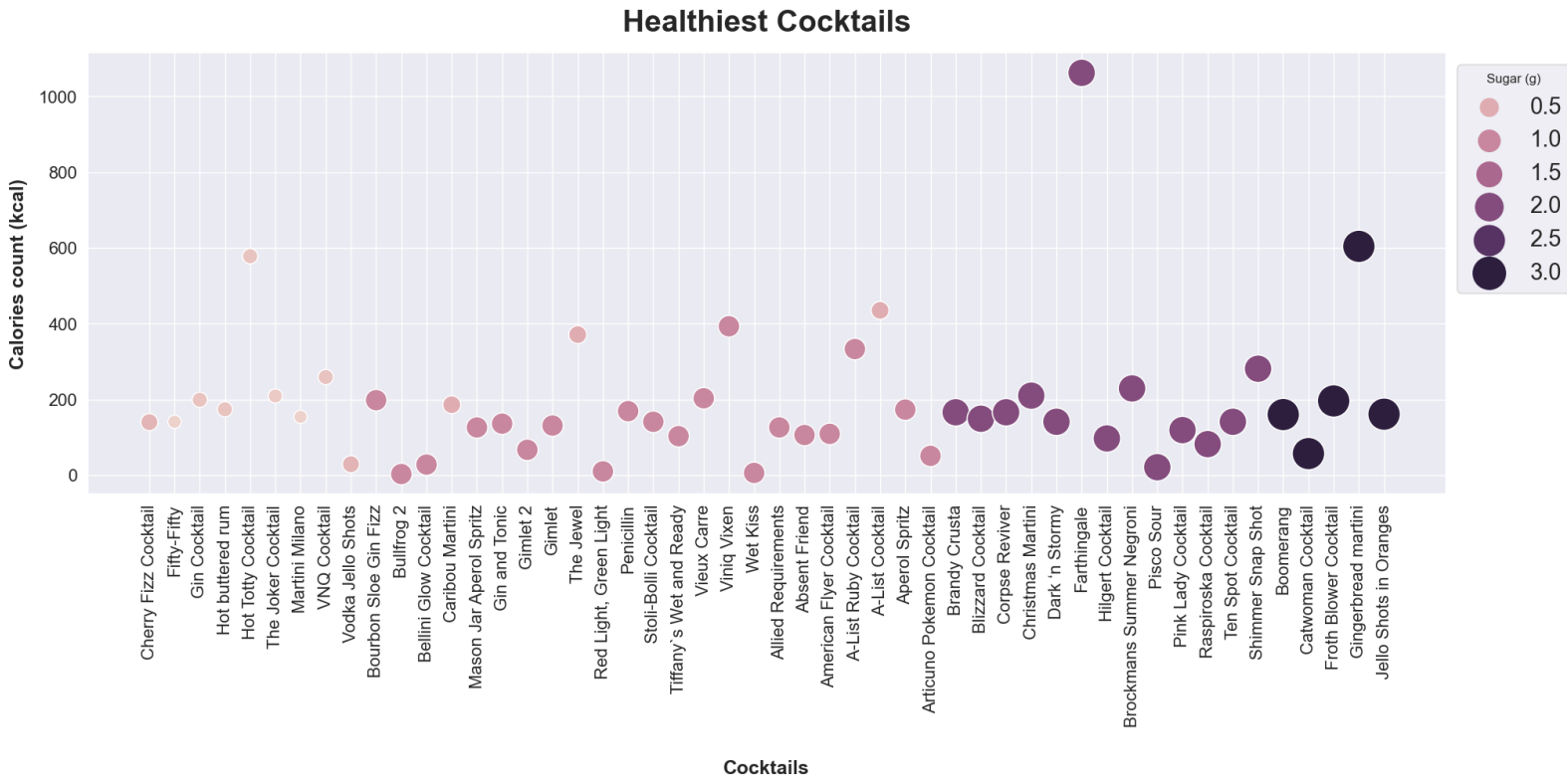
Most popular ingredients:

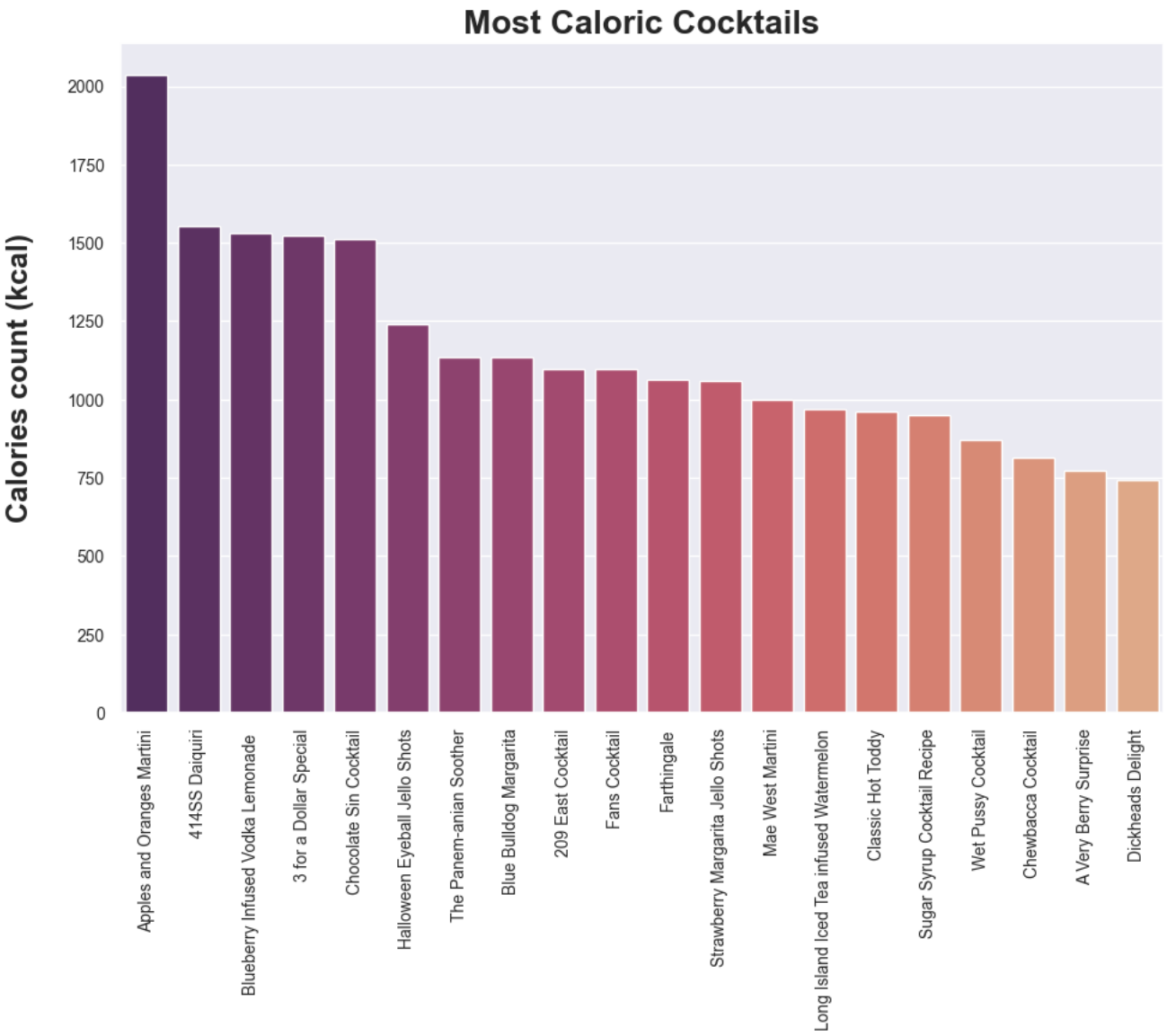
```sql
SELECT 'Juice' AS ingredient, COUNT(CASE WHEN ingredients LIKE "%Juice%" THEN 1 END) AS occurrence
FROM ingredients_id
UNION ALL
SELECT 'Gin' AS ingredient, COUNT(CASE WHEN ingredients LIKE "%Gin%" OR ingredients LIKE "%gin%" THEN 1 END) AS occurrence
FROM ingredients_id
UNION ALL
SELECT 'Soda' AS ingredient, COUNT(CASE WHEN ingredients LIKE "%Soda%" THEN 1 END) AS occurrence
FROM ingredients_id
UNION ALL
SELECT 'Tequila' AS ingredient, COUNT(CASE WHEN ingredients LIKE "%Tequila%" OR ingredients LIKE "%tequila%" THEN 1 END) AS o
FROM ingredients_id
UNION ALL
SELECT 'Cider' AS ingredient, COUNT(CASE WHEN ingredients LIKE "%Cider%" THEN 1 END) AS occurrence
FROM ingredients_id
UNION ALL
SELECT 'Port' AS ingredient, COUNT(CASE WHEN ingredients LIKE "%Port%" THEN 1 END) AS occurrence
FROM ingredients_id
UNION ALL
SELECT 'Liqueur' AS ingredient, COUNT(CASE WHEN ingredients LIKE "%Liqueur%" THEN 1 END) AS occurrence
FROM ingredients_id
UNION ALL
SELECT 'Rum' AS ingredient, COUNT(CASE WHEN ingredients LIKE "%Rum%" THEN 1 END) AS occurrence
FROM ingredients_id
UNION ALL
SELECT 'Chocolate' AS ingredient, COUNT(CASE WHEN ingredients LIKE "%Chocolate%" or "%chocolat%" THEN 1 END) AS occurrence
FROM ingredients_id
UNION ALL
SELECT 'Champagne' AS ingredient, COUNT(CASE WHEN ingredients LIKE "%Champagne%" or "%champagne%" THEN 1 END) AS occurrence
FROM ingredients_id
ORDER BY occurrence desc;
```

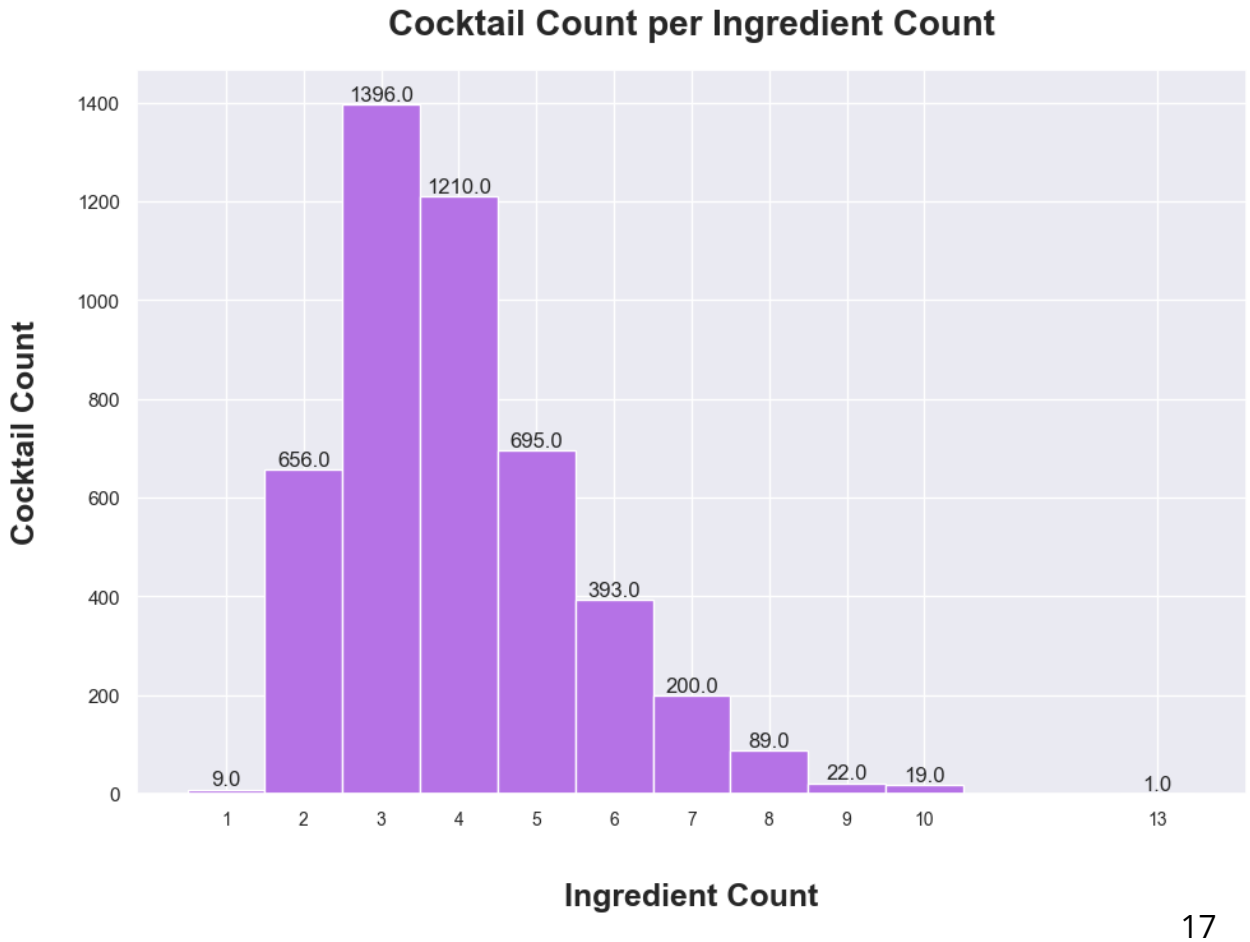
EXPLORATORY DATA ANALYSIS

The retrieved data contained few dimensions and numeric values, the exploration and interpretation were therefore limited. Nonetheless some insights can still be drawn with visualization as the plots and charts below demonstrate.

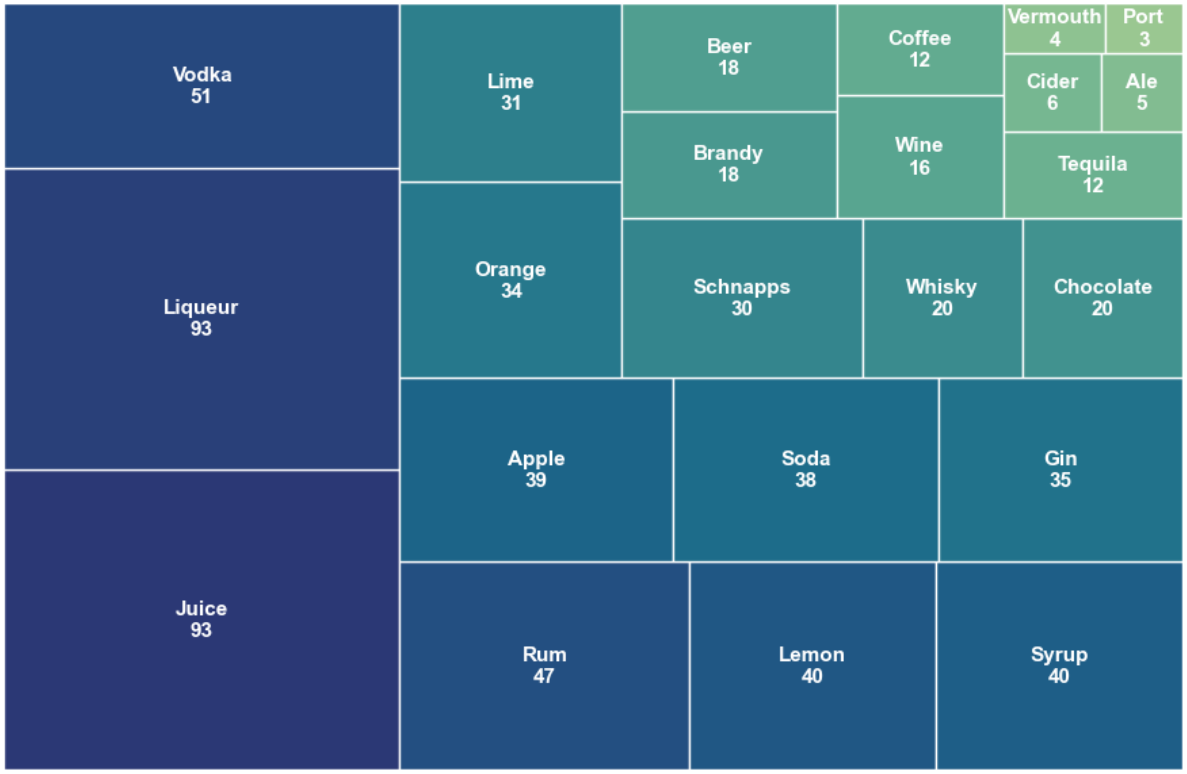




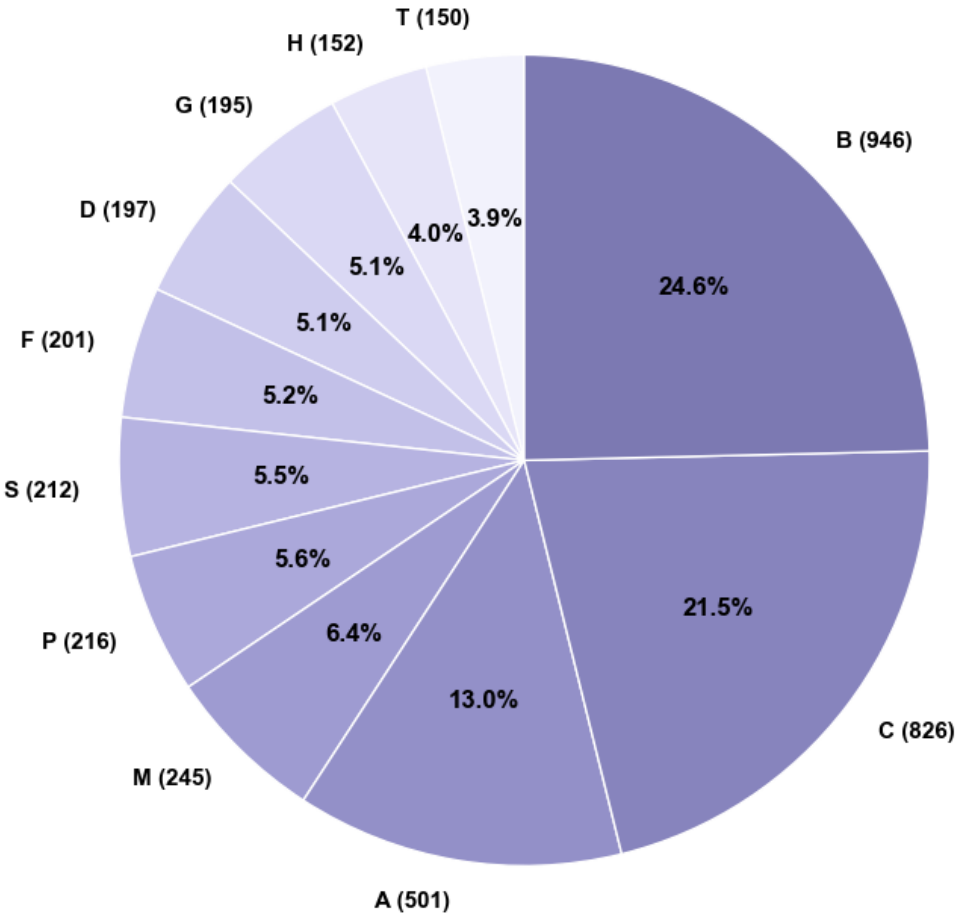

Most Caloric Cocktails
Calories count (kcal)
2000
1750
1500
1250
1000
750
500
250
0
Apples and Oranges Martini
414SS Daiquiri
Blueberry Infused Vodka Lemonade
3 for a Dollar Special
Chocolate Sin Cocktail
Halloween Eyeball Jello Shots
The Panem-anian Soother
Blue Bulldog Margarita
209 East Cocktail
Fans Cocktail
Farthingale
Strawberry Margarita Jello Shots
Mae West Martini
Long Island Iced Tea infused Watermelon
Classic Hot Toddy
Sugar Syrup Cocktail Recipe
Wet Pussy Cocktail
Chewbacca Cocktail
A Very Berry Surprise
Dickheads Delight


Cocktail Count per Ingredient Count
Cocktail Count
1400
1200
1000
800
600
400
200
0
9.0
656.0
1396.0
1210.0
695.0
393.0
200.0
89.0
22.0
19.0
1.0
1
2
3
4
5
6
7
8
9
10
13
Ingredient Count

Occurence of Ingredients



First Letter Distribution



Although the first letter distribution was out of curiosity, it sparked my interest and therefore I plotted a word cloud revealing which words are most used in cocktail names.

CONCLUSIONS

Based on our market trend analysis, we can conclude that what was observed during the COVID pandemic has resulted in a habit for a section of the population . With further market analysis, we could define exactly how to design and market a recommender app that would find its recurring users as well as its sporadic users.
The app could be developed for a specific brand, building cocktail recommendations around its main product.

The present analysis is sufficient to develop a POC and working MVP for pitching and further development.

Through EDA and visualization of the dataset, we can distinguish that a lots of cocktails have many variations, but the simplest, Martini (gin & vermouth with an olive) has the most variations as its two base ingredients are popular ingredients for mixology and the simplicity of its recipe allows for experimentation.
A related note is that most cocktails contain 3-5 different ingredients (including garnish), therefore with only one alcohol and a couple of mixers and garnish, users can craft hundreds of combinations.
The proposed algorithm is sound in that regard.

Further analysis could decipher which are the ingredients that are most capable of mixing with others, beyond their co-existence in recipes.

In the lexical field, words that sound ‘pop’ and evocative dominate the cocktail scene. Blue being an unusual color for a drink, it immediately evokes a sense of magic potion and therefore draws curiosity.
Surveys and an extensive market analysis could reveal more insights on this point.

The following top words often end in “ee” : ‘dirty’, ‘brandy’, ‘bloody’, ‘crazy’, ‘candy’, ‘monkey’, ‘tea’, ‘coffee’, ‘cranberry’... These types of sounds are light and sweet, just like ‘cherry’, the top 2nd word in cocktail names.
There is no proven correlation here, but an observation to be probed further.

Other insights include the wide range of cocktails’ caloric density. The heaviest having over 2000 kcal (Apples & Orange Martini) whilst the lightest only has 3kcal (BullFrog 2). This seems to be mainly due to the ingredients. Vodka, vermouth, gin, dry white wine are on the leaner side of alcohols, while liqueurs, syrups and juices are heavy in both sugar and calories.
The amount recommended in the recipe is also of great weight in the caloric and sugar intake. In bartending, “parts” is often synonymous with ounces (oz). Deeper analysis would provide a better understanding of the calories distribution amongst ingredients and allow for additional parameters to be integrated in the product.

REFERENCES

1. https://www.forbes.com/sites/jilliandara/2021/02/17/this-report-shares-how-the-pandemic-changed-the-way-we-consume-alcohol/

2. https://www.wsj.com/articles/liquor-brands-bet-thrifty-drinkers-will-keep-making-at-home-cocktails-11673092159

3. https://info.cga.co.uk/hubfs/Drinks/Mixed%20Drinks%20at%20Home%20Sales_Presenter%20final%20updated%202021.pdf

4. https://mpost.io/wp-content/uploads/BACARDI-Cocktail-Trends-Report-2023.pdf

5. https://cgastrategy.com/cocktails-mixed-drinks-at-home/

6. https://drinks-intel.com/latest-news/will-the-home-premise-boom-last-past-covid/

GITHUB LINK:

https://github.com/Lucien-Stephane-Ld/certification-project-DAFT410

KANBAN LINK:

https://github.com/Lucien-Stephane-Ld/certification-project-DAFT410