LEPL1109 - Statistics and Data Sciences
HACKATHON 4 - Clustering: What is it all about?

December 9, 2022

Lastname	Firstname	Noma
Barbason	Romain	22142000
Dubois	Brieuc	23752000
Jadin	Guillaume	10581800
Shafiei	Tania	17451800
Villette	Emile	10992000
Jeanmenne	Nicolas	48741900

Please, read carefully the following guidelines:

- Answer in English, with complete sentences and correct grammar. Feel free to use grammar checker tools such as LanguageTools free and open-source plugin;
- Do not modify questions, and input all answers inside `\begin {answer}...\end {answer}` environments;
- Each question should be followed by an answer;
- Clearly cite every source of information (even for pictures!);
- For bonus material (additional figures, code, very long equations, etc.), use Appendices;
- Whenever possible, use the `.pdf` format when you export your images: this usually makes your report look prettier[1];
- Do not forget to also submit your completed notebook on Moodle.

Contents

[1] This is because `.pdf` is a vector format, meaning that it keeps a perfect description of your image, while `.png` and other standard formats use compression. In other words, this means you can zoom as much as you want on your figure without decreasing image resolution. For simple plots, vector formats can also save a lot of memory space. On the other hand, we recommend using `.png` when you are plotting many data points: large scatter plots, heatmap, etc.

Context and objectives

The objective of this hackathon is threefold: (1) extract meaningful information from a dataset, (2) observe relationship(s) (if any) between features and eventual underlying groups (clusters), and (3) develop an unsupervised clustering tool and exploit the associated data.
To this end, you will use a synthtic dataset (available on **Moodle**) inspired from a real dataset from Kaggle. Given a couple of features, you should be able to **create Pokémon clusters based on spatial coordinates, temporal informations and other prodived features. Then exploit the content of these different clusters to determine the likeliness of capturing a given Pokémon for some input requests such as time and position**.

Figure 1: Context illustration.

Nowadays, mobile games are gaining more and more attention, sometimes maybe too much. This is especially true for the well known Pokémon GO.

For those who do not know: Pokémon GO is a mobile-game in which players have to capture as many Pokémon as possible. Pokémon are creatures that randomly spawn (i.e., appear) at different positions and times, but some locations are more likely to have Pokémon appearing: shopping malls, city centers, parks, and so on. Once a Pokémon has spawned (i.e. appeared), the players have to physically go to the same place as the Pokémon to hopefully capture it.

As a casual Pokémon GO player and a proficient data scientist, you would like to increase your level by leveraging some data-related techniques. To this end, you found a Pokémon GO spawning versus localization dataset that you will use to, hopefully, achieve your goals (see above).

Questions and Answers

1 Data Preprocessing

Question 1.1: Removing unnecessary features

Can you already, a priori, detect that some features are useless?

1. if yes, list those (useless) features and explain your choice;
2. if not, then explain why it is better to wait.

Generally speaking, is it a good idea to remove a feature based on a priori knowledge, or it doesn't alter the final outcome?

Expected answer length: 2-4 lines.

Answer to 1.1:

We have chosen to remove the following features : *id* and *num* because id doesn't add any value to the dataset while num is the same information than *name*, represented as an integer.

Generally speaking removing features on a priori knowledge can be beneficial as it can make the model simpler to learn and to interpret but it can also strongly impact the performance of the model and therefore the final outcome. In general, it is necessary to consider the impact on the final outcome for each feature removed.

Question 1.2: Handling missing data

Given the dataset and the amount / type of missing information, what strategy do you propose to follow regarding missing data (NaNs)? You can choose one or many of the following:

1. drop features (column) with missing information;
2. drop samples (row) with missing information;
3. replace missing information with interpolation / extrapolation / simple substitution / ...

Expected answer length: 4-8 lines.

Answer to 1.2:

First of all, there is 18 854 samples (row) out of 923 461 where there is one or more feature missing. All missing value are related to the feature `appear_duration`. We decided to drop every sample with missing information as it represent only 2 % of the whole dataset. Thus it should not divert the performance of a potential model. Moreover compared to the strategy of dropping column we still have the `appear_duration` feature for the remaining 98 %. We have also decided to **not** interpolate, extrapolate or substitute missing information because it is not relevant in this case and could even result to worst performance.

Question 1.3: New features

What features have you added? If a particular manipulation has been applied, please explain.
Expected answer length: 2-4 lines.

Answer to 1.3:

We added the following features : `time`, `sin_time` and `cos_time`. These new features allow us to make a cyclical feature encoding of the initial date feature. Any model can then make a better usage of the information in the date because the main advantage of a cyclical encoding is to allow the model to understand the relationship between different dates.

2 Data Visualization

Question 2.1: Features visualization

Based on what you have seen in your notebook and whatever other visualization you will try, you can already get an idea of which features seem to contain discriminative information, i.e., which features are likely to be more important for the clustering than others.
Justify which features you think would be interesting or not to keep in order to realize the required task. Feel free to try and add your own data visualization to highlight or not their importance.
Expected answer length: 2-8 lines and 0-2 image(s).

Answer to 2.1:

Since the goal of the hackaton is to predict which Pokémon is the most likely to appear in function of the time and geographical position, we think that a few features contain discriminative information. Firstly, as shown in Figure 2, the amounts of Pokémons that appear vary greatly depending on the hour of the day (it spikes around midnight and noon). This seems to be one major factor. Furthermore, the type of the Pokémon is also a very important metric to determine if it is likely to appear at a certain position (see Figure 3, which shows the type proportions in the data set). We included two basic charts here, but there would of course be a plethora of other comparisons that we could think of.

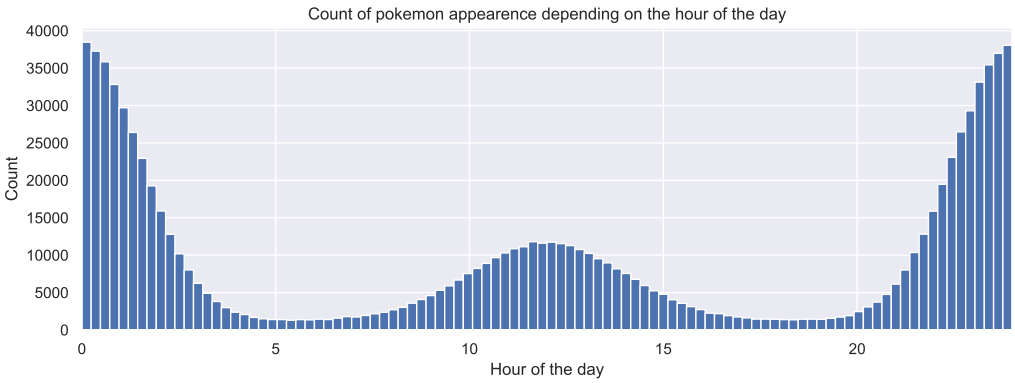

Figure 2: Bar chart of hour vs number of appearances

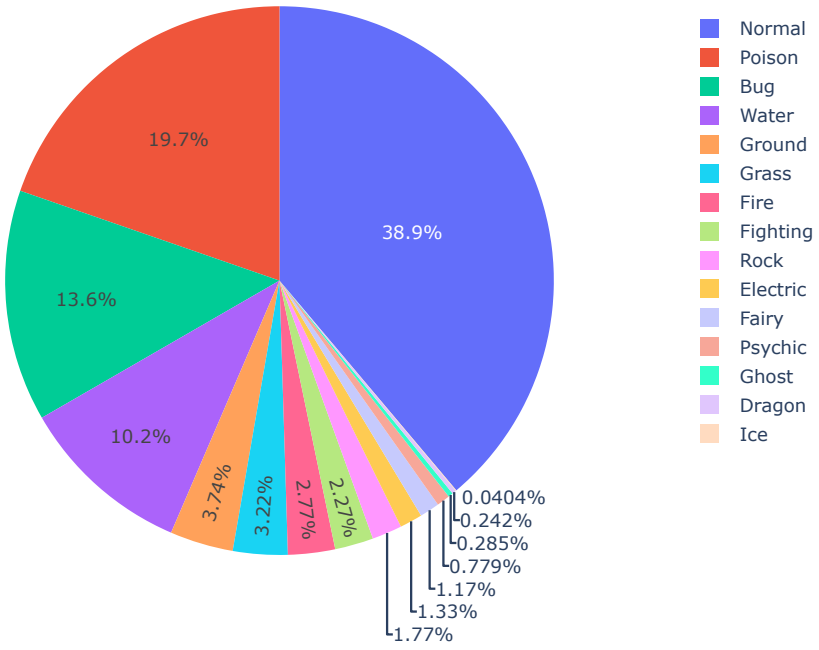

Figure 3: Pie chart of Pokémon appearances by type

Question 2.2: Spatial features visualization

Based on the maps from your notebook, what can you infer about the spawn locations of the Pokémon? Is there a link between their types and the land affectation? If yes, explain.
Expected answer length: 2-4 lines.

Answer to 2.2:

Yes, there seems to be a clear link between link and land affectation. When isolating the legend, it is immediately visible. For instance, water Pokémons almost exclusively appear in permanent water bodies, normal ones in build-up areas, bug types outside of built-up areas, and so forth. This helps to reasonably infer that this assumption is correct.

Question 2.3: Spatial clutering

Based on the maps above, i.e., on the spatial features only, how do you think a clustering will perform according to the number of clusters? In other words, what do you think will happen with 1, 15 (= number of types), 100+ clusters?
Expected answer length: 2-4 lines.

Answer to 2.3:

With one cluster, it would simply be "normal" (cf. Figure 3). 15 clusters, we think, would be a more appropriate amount, since the answer in 2.2 underlines the link between type and land affectation. 100+ clusters would probably show diminishing returns and be too much. We think the sweet spot lies somewhere between 15 and $\sim$ 80.

Question 2.4: Feature importance visualization

Based on the biplot graph you generated in your Jupyter Notebook, do all features have the same importance? If no, which features are less important and why? You can use all other graphs from the visualization part to justify your answer.
Expected answer length: 2-6 lines + 0-2 image(s).

Answer to 2.4:

No, not all features have the same importance as pointed out by Figure 4. The `cos_time`, `latitude` and `longitude` features appear to be the most important. This tends to confirm the trend drawn out in the previous questions wherein we infered that position (i.e. soil type) and time have a great influence on the Pokémon appearances.

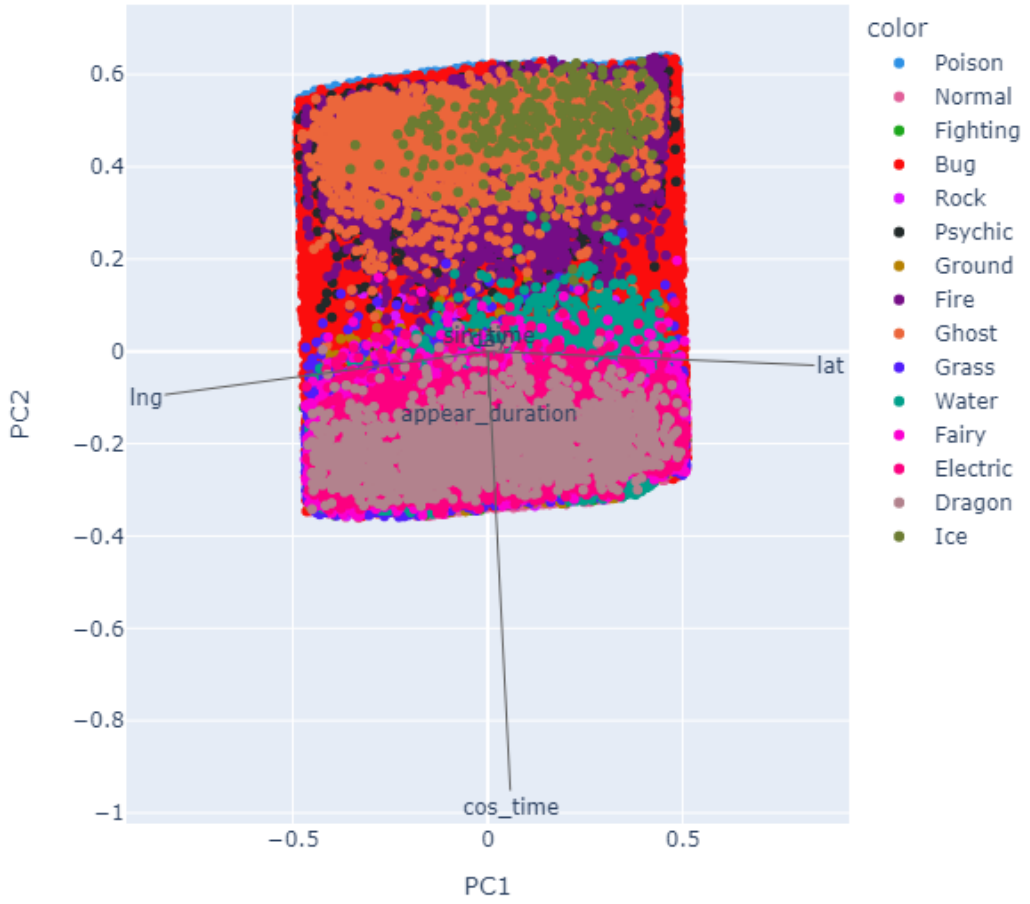

Figure 4: PCA analysis of types

3 Clustering

Question 3.1: Number of clusters

Accounting for all features (i.e., spatial **and** temporal coordinates), what do you think is the ideal number of clusters? What will happen if too many or even too few clusters are chosen?
Expected answer length: 2-6 lines + 0-2 image(s).

Answer to 3.1:

During our testing, we found that the number of clusters with the highest average accuracy is 15 (0.5352 accuracy score, with a standard deviation of 0.0323), which is a compromise. If too many clusters are chosen, then it leads to "overfitting", where K-Means tries to find clusters where there shouldn't be any (cf. Figure 5, the pink/light blue clusters) [1]. The logic applies similarly when too few clusters are chosen, i.e. the clustering algorithm groups clearly distinct clusters together.

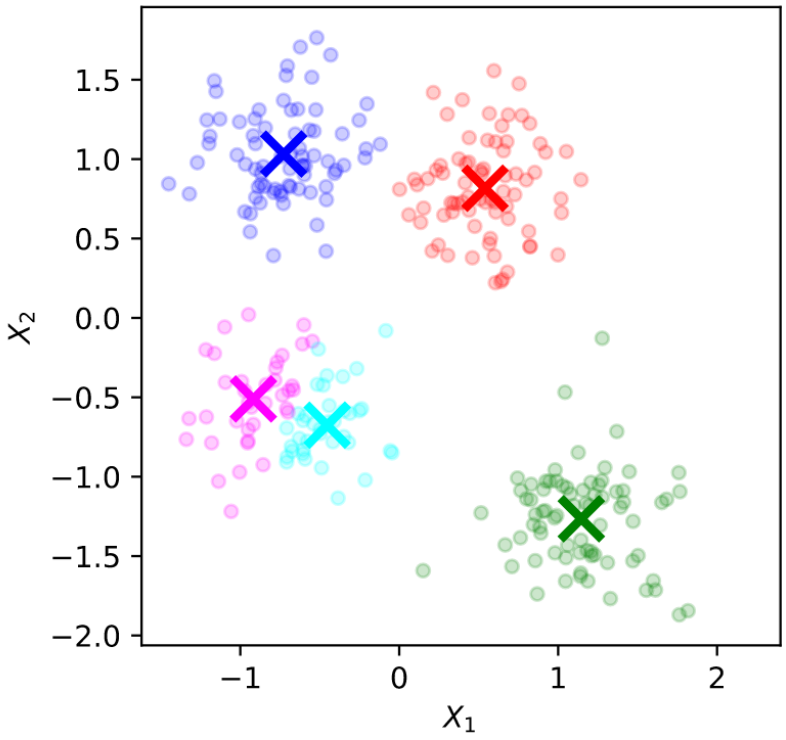

Figure 5: Example of overfitting [1]

Question 3.2: Cluster composition

Do you think the naive approach will give the best results? Justify briefly.
What do you think would be the best way to estimate the Pokémon you encounter? Explain.
Expected answer length: 2-4 lines.

Answer to 3.2:

We think that there aren't many cases where the naive approach works. For us, the best way would be to select relevant data points with the most representative information possible. The more verbose, the more accurate we think the prediction will be.

Question 3.3: Your clustering solution

Describe here your clustering solution (how many clusters, which method of sampling the Pokémon, other important choices that have been made, etc.). Justify your choices with the help of the metric.
Expected answer length: 4-8 lines + 0-2 image(s).

Answer to 3.3:

For each cluster, we choose the most frequent Pokémon (meaning that the remaining ones have a probability of 0 to be chosen). This means that for K clusters, we observe a maximum of K different Pokémons. The number of clusters is chosen by trying with different values and evaluating the precision and standard deviation (cf. Figure 6). However, as explained in answer 3.1, we need to choose a relevant number of clusters for the K-Means algorithm. So the value we chose is a trade-off.

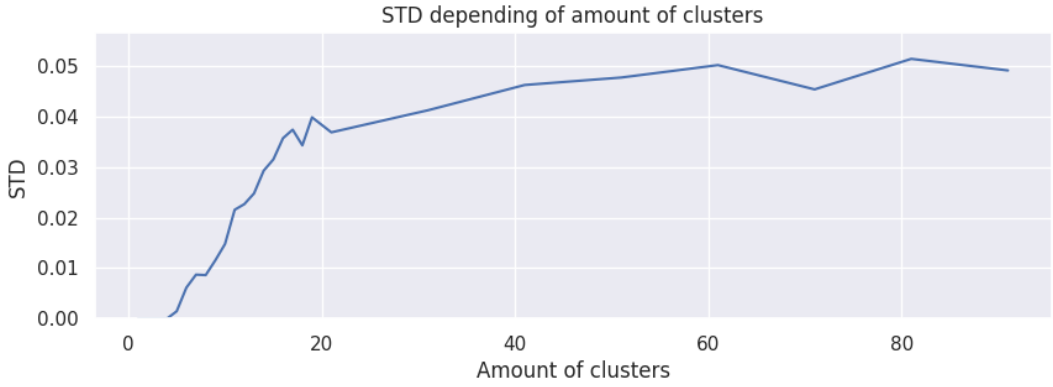

Figure 6: Clustering accuracy and std deviation

Question 3.4: Comparing models - BONUS

Compare how your model performs when predicting the Pokémon types, based on `output="type"` versus `output="name"`. I.e., does predicting Pokémon types based on the Pokémon name performs better than directly predicting the types?
Expected answer length: 2-6 lines + 0-4 image(s).

Answer to 3.4:

Please fill this space with your answer.

References

[1] Laurent Jacques. Statistics and data science, part iii: Unsupervised learning., 2022.

[2] Laurent Jacques and Thomas Feuillen. The importance of phase in complex compressive sensing, 2020.

A Demo

A.1 Interesting questions

Question Demo:

Can you show me what I can do?

Answer to Demo:

This is how I answer to Demo. I can cite [2] content that I use or refer to A. I can also reference images such as in Figure 7, or equations with (1) or Equation 1.

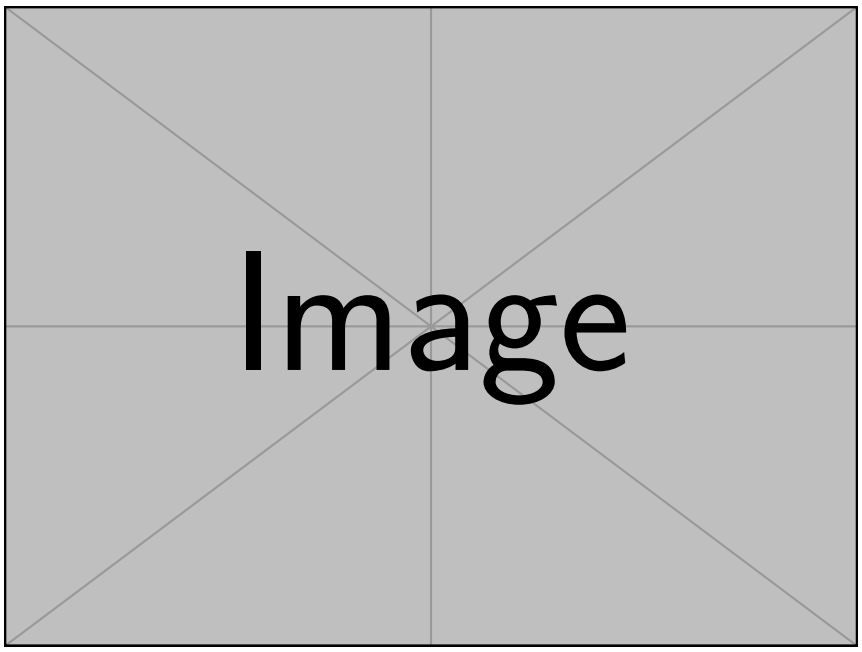

Figure 7: Demo caption.

$$E = mc^2 \tag{1}$$

If you wish to present code samples, you can either use the Listing 1 format or use inline code `import numpy as np; x = np.arange(10)` if this better suits your needs. However, we recommend putting your code in the Appendices.

```
1  import numpy as np
2
3  x = np.arange(10)
```

Listing 1: My super code.

Note: syntax highlighting for code is provided by the `minted` package. If you are not using Overleaf, you might need to install some requirements before it can work.