# Big Data Final Project

A report by Nora, Michaël, Andrei, Dominika and Dhriti

## Preparing and Cleaning the data

We approached the problem by firstly reading the files in a CSV format, as it's a type of file that we are the most familiar with. Next, we extracted the relevant columns of data by using csv_reader. After that, we merged the CSV files into one by applying csv_merger. To identify and ensure that there was no duplicate data, we used drop_duplicates. We further cleaned the data by removing faulty launch dates; as entries with launch date 0, for example, can be assumed invalid as that would imply they were made on 1st January 1970, following the given Unix Epoch time of the original CSV files. We also extracted the columns we were going to be working on for the rest of the project, as working with the entire file seemed inefficient. We chose to work with the pandas library, as it was what we were most comfortable with and felt we could all contribute equally if we used it.

Our data aggregation process can be separated into 3 parts (we have also recorded a more detailed process of our method in the minutes of the meeting):

We wrote code to read the csv files using Dask, then turned the files back into a pandas dataframe.After this, we converted the epoch time to datetime time to make it more understandable, extracted the relevant columns, and created a separate data frame for the columns we extracted.

## Question 1: Success percentages of completed campaigns

We began working on this question by grouping the data by months and years using a multilevel index system, but disposed of the indices once we'd arranged it chronologically. Afterwards, we identified the states of each project: successful, failed, live, etc. This would help us measure the success percentages. We created an empty results list and a *for* loop to append successful cases with their dates. What we later realised was this would be very inefficient and time consuming for processing many large files, so we changed it to a binary system where 'successful' was 1 and any other was 0, and created a new column that recorded the state of the project. From here on out the success rate can be calculated via the pandas '.groupby' function, that takes the success_rate and the categories as arguments, and the actual success rate can be calculated by the .mean function to get the final success rate with one being 100% successful.

Now having the overall success rate of all projects per month, a simple line graph can be made by using the matplotlib function .plot setting the x-axis as time and the y-axis as success rate. This gives the following graph.

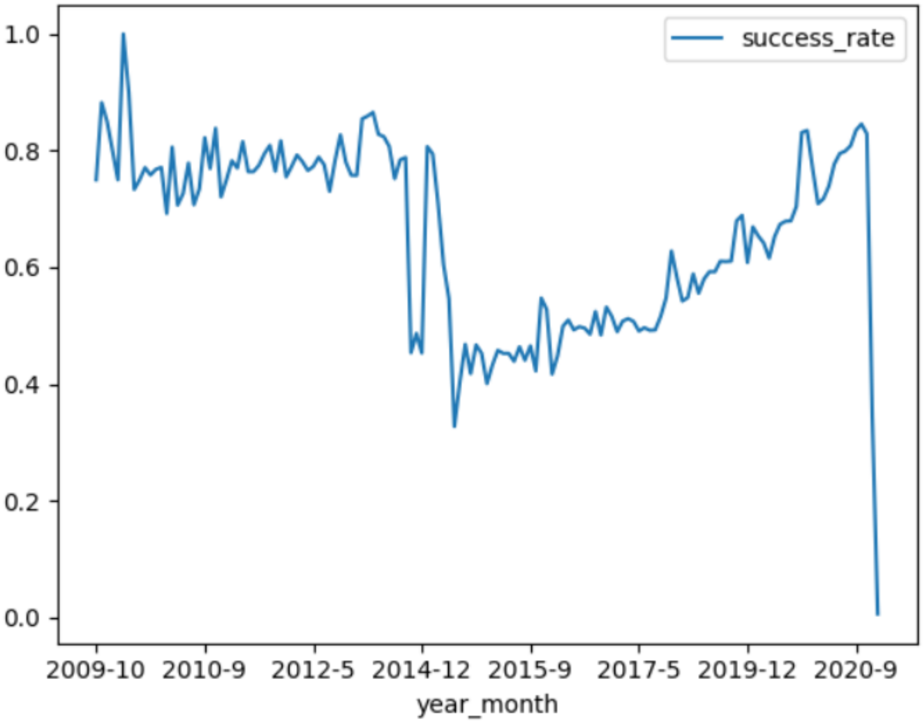
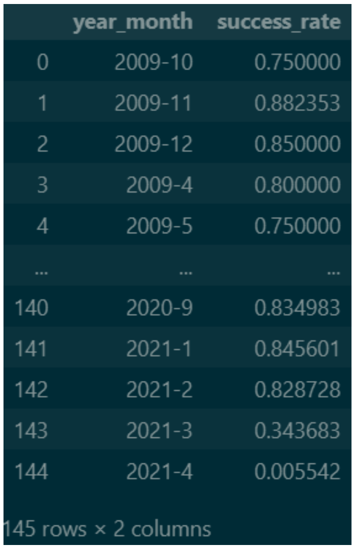


Figure 1. Line Plot for the monthly success rate of all projects



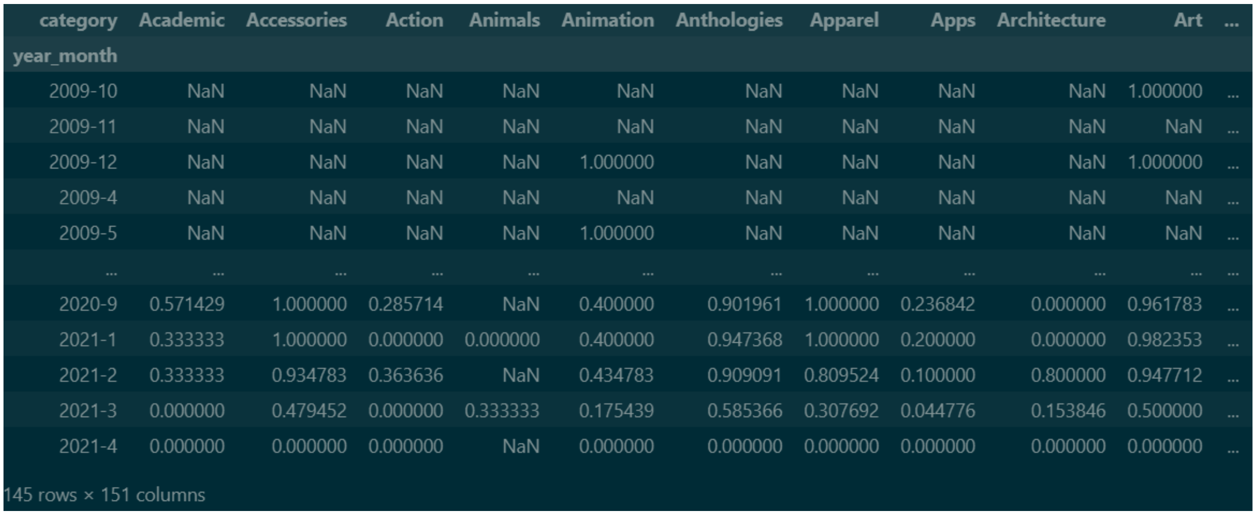
| | year_month | success_rate |
|---|---|---|
| 0 | 2009-10 | 0.750000 |
| 1 | 2009-11 | 0.882353 |
| 2 | 2009-12 | 0.850000 |
| 3 | 2009-4 | 0.800000 |
| 4 | 2009-5 | 0.750000 |
| ... | ... | ... |
| 140 | 2020-9 | 0.834983 |
| 141 | 2021-1 | 0.845601 |
| 142 | 2021-2 | 0.828728 |
| 143 | 2021-3 | 0.343683 |
| 144 | 2021-4 | 0.005542 |

145 rows × 2 columns

Figure 2. Numerical Data that shows the monthly success rate of all projects per month

**Question 2: Monthly ratios of completed campaigns per category**

For the second question, we followed a similar approach, namely using the binary system for assigning successful 1, and for failing 0. Since the category names in the dataset contained a lot of metadata about the category itself and not about the project, we applied a lambda method that makes use of a RegEx function to find the true category name and replace the old string that is in place. After cleaning and arranging the data for this question, we utilised a pivot table in order to compute the monthly success ratios of each category. We visualised the data using heat map because it allows us to show the categories in a more readable way. We noticed that using scattering or plotting makes such a large number of categories less readable. However, we decided to include two line plots in order to offer a better view of the data we computed.



| category | Academic | Accessories | Action | Animals | Animation | Anthologies | Apparel | Apps | Architecture | Art | ... |
|---|---|---|---|---|---|---|---|---|---|---|---|
| year_month | | | | | | | | | | | |
| 2009-10 | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | 1.000000 | ... |
| 2009-11 | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | ... |
| 2009-12 | NaN | NaN | NaN | NaN | 1.000000 | NaN | NaN | NaN | NaN | 1.000000 | ... |
| 2009-4 | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | ... |
| 2009-5 | NaN | NaN | NaN | NaN | 1.000000 | NaN | NaN | NaN | NaN | NaN | ... |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 2020-9 | 0.571429 | 1.000000 | 0.285714 | NaN | 0.400000 | 0.901961 | 1.000000 | 0.236842 | 0.000000 | 0.961783 | ... |
| 2021-1 | 0.333333 | 1.000000 | 0.000000 | 0.000000 | 0.400000 | 0.947368 | 1.000000 | 0.200000 | 0.000000 | 0.982353 | ... |
| 2021-2 | 0.333333 | 0.934783 | 0.363636 | NaN | 0.434783 | 0.909091 | 0.809524 | 0.100000 | 0.800000 | 0.947712 | ... |
| 2021-3 | 0.000000 | 0.479452 | 0.000000 | 0.333333 | 0.175439 | 0.585366 | 0.307692 | 0.044776 | 0.153846 | 0.500000 | ... |
| 2021-4 | 0.000000 | 0.000000 | 0.000000 | NaN | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | ... |

145 rows × 151 columns

Figure 3. Numerical Data of the Pivot Table that shows the monthly success rate for each category

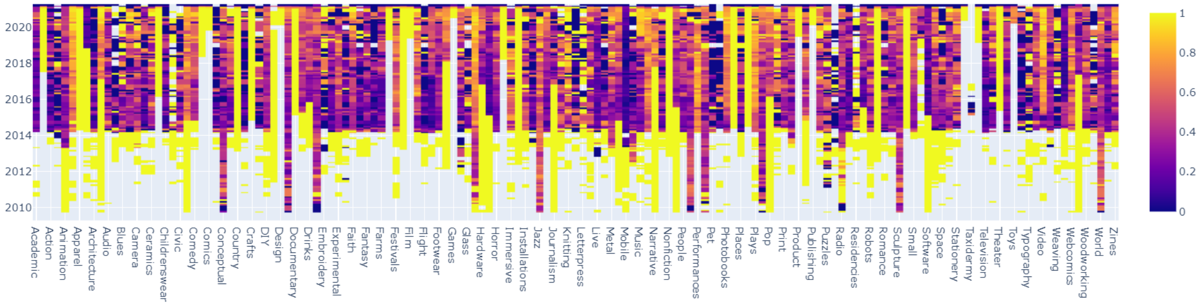


Figure 4. Heatmap created from the Pivot Table

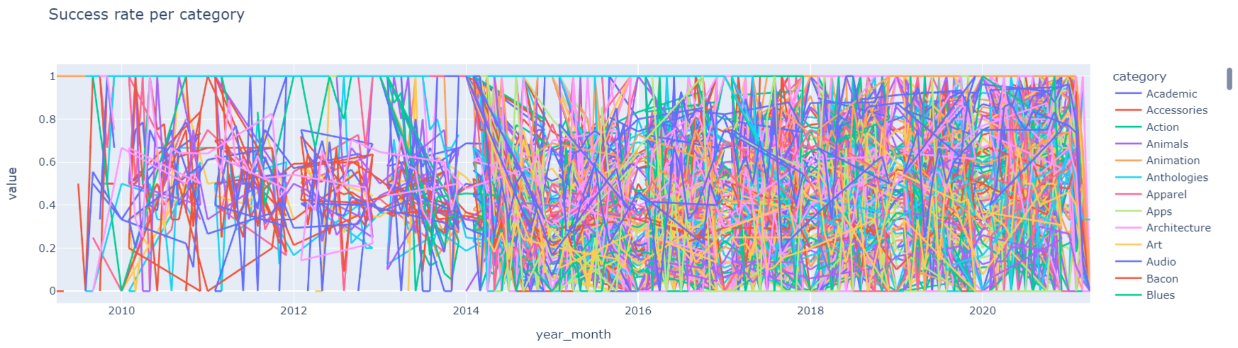


Figure 5. Line Plot generated from the Pivot Table

**Plan:**

1. Reading the csv files
2. Extract the relevant columns of data
3. Merging the csv files into one
4. Finding duplicates
5. Cleaning the data: removing faulty launch dates (it can be assumed that entries with launch date 0, for example, are not true as that would imply they were made on 1st January 1970 according to unix time)
6. Identifying the states
7. Data aggregation:
    a. Finding the success rates for the projects: **some assumptions we have made regarding the success rates:**
        i. The older data comes from scrapes where Kickstarter only retained the successful projects.
        ii. The December 2022 scrape was not complete because it only goes as far as the 12th of December so it's likely there were no projects started and completed in December (we used 'launched_at' as a way to order them chronologically, so if something was launched in December, 0 per cent of the cases would've been successful)
        iii. Originally used a for loop, but it was not time efficient
    b. Monthly ratios of completed campaigns by category
8. Graphing the data

**Minutes of the meeting:**

- January 12th:
    - Discussed the aim of the project and how to tackle the questions
    - Created a plan for the following week

- January 17th:

    - Created code for reading the csv files
    - Merging the csv files
    - Finding duplicates
    - Finding faulty values

- January 20th:
    - Created code to sort the values chronologically
    - Started using  for loop method
    - Sorted the values by category

- January 21st:
    - Visualised data
    - Worked on parallel processing and tested the cleaning process on all 62 files

- January 22nd:
    - Made the final code more readable
    - Used Plotly to visualise the data better
    - Found a different method for measuring success rate to cut down processing time
    - Compiled all snippets of code into a new notebook