Important People - Exploring the Pantheon Dataset

Alper Calisir calisir.1888955@studenti.uniroma1.it

Artur Back de Luca backdeluca.1900870@studenti.uniroma1.it

Stefano Cappai cappai.1844363@studenti.uniroma1.it

Abstract

This work describes our analyses and design choices on building an tool to visualize meaningful information of relavant historical figures. This project is part of the Visual Analytics course hosted at the Sapienza University of Rome, and instructed by Professor Giuseppe Santucci and Dr. Marco Angelini.

1. Motivation

Human history is a collective process that is ever-changing and shaped by those who live at a particular point in time. Among these, there are certain figures who, by their contributions to this process, earn a spot in the long-lasting human collective memory. The lives and trajectories of these historical personalities also help us to understand the social ethos at a moment in history, and therefore essential to academic fields such as social sciences.

The Pantheon Project is an initiative that centralizes biographical data of several historical figures around the world. The project was started by MIT's Collective Learning group and collects information of the most prominent personalities presented on Wikipedia.

In this project, we intend to enhance the user's understanding of global culture and its prominent figures. To this end, we have created a dashboard that allows users to navigate fluently through the Pantheon dataset, encouraging them to explore the dataset and discovering patterns.

2. Related Work

Lots of research has been done to identify globally famous individuals. In this context, Wikipedia stands as a vast source of user-generated content, what naturally attracts many researchers and avid readers. Furthermore, many computer science related fields have utilized Wikipedia knowledge for their studies such as Gabrilovich et al. (2007); Garcia-Fernandez et al. (2011); Kinzler (2005); Milne et al. (2006). Several researchers have specifically focused on using such data to figure out important people in history. For one, the work of Eom et al. (2015) extracted the top 100 historical figures. Their intention was to investigate those 100 historical figure's spatial, temporal and gender distributions with respect to their cultural origins. Their paper Jatowt et al. (2016) looked at the historical data in Wikipedia to find the user interests and characteristics regarding the historical people in that data. They have used hyperlink structures to form graphs that represent the relationships between time and article popularity. Moreover, they have demonstrated several ways that temporal aspects of the link structure can be used to calculate importance of the person. In Andrienko et al. (2011) there is important analyses on what is geovisual analytics that puts emphasis on support for "analytical reasoning" making an important distinction between "reasoning done by human analyst" and "computational reasoning". This work also describes some challenging problems chosen by VisMaster (http://www.vismaster.eu), an European coordination action with the aim to define a roadmap for the future of visual analytics research. Lastly, in the paper of Yu et al. have presented the dataset called Pantheon 1.0. This dataset extensively captures the demographic information and measurements of the importances of historical figures.

Our work builds on top of the work conducted by Jatowt, Hidalgo and Andrienko to utilize historical data in Wikipedia.

3. Application Description

Our application aims to enable the user to visualize several aspects of historical figures and explore their similarities and differences. The dashboard does this by providing several analyses on the relevance of these personalities that are organized by their location in space and time. The application is composed by 6 parts, showcased in Figure 1.

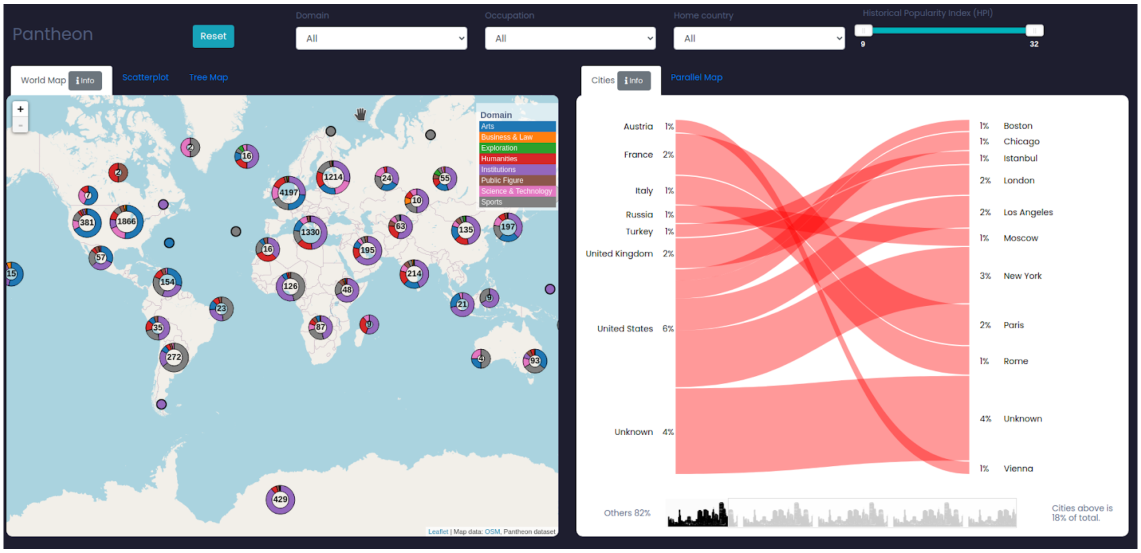

Domain: All
Occupation: All
Home country: All
Historical Popularity Index (HPI): 9 – 32

Figure 1: Overview of the application

With a large amount of data at our disposal, we provide the user the possibility of interacting with several filters, which can guide the analysis to a particular viewpoint, segmented by occupation, domain of impact, geographic location and relevance.

Naturally, once these preferences are defined, we enhance the analysis with a set of basic visualization techniques, briefly described as following:

- **World map:** located on the first left tab, it allows the user to interact with data organized geographically;
- **Scatter plot:** on the second left tab, it displays historical information reduced to three dimensions;
- **Tree map:** on the third left tab, it displays the hierarchical composition of domains and occupations ranked by its 5 most relevant personalities;
- **Cities map:** on the first right tab, it displays the most important countries and cities of birth.
- **Parallel map:** on the second right tab, it shows the summary of 4 features (continent, gender, domain, and industry);
- **Internet results:** once a point in the scatter plot is clicked, a tab on the right section displays the Wikipedia article that corresponds to the historical figure selected.

4. Implementation Details

In this section, we go through technical details involving the dataset and the implementation of each dashboard components

Dataset

We have obtained the dataset from Harvard Dataverse (Yu et al.). This dataset contains 11,341 biographies present in at least 25 languages on Wikipedia. Each entry contains a historical background such as origin, occupation, date of birth, gender. In addition, it displays summary statistics such as average page views for both English and non-English versions. Lastly, two measures of global popularity including the number of languages in which a biography is present in Wikipedia (L), and the Historical Popularity Index (HPI) a metric that combines information on L, time since birth, and page-views (2008-2013) are introduced. In their work, Yu et al. have cross-checked their indices L and HPI with Charles Murray's book (Murray) to discover that they have high correlation with the actual data. The details of these datasets can be found in Appendix.

Metrics of Fame

Authors have used 2 metrics for calculating the fame of historical characters. The first one is denoted as **L** which is "the number of different Wikipedia language editions that have an article about a historical character". The authors have also taken consideration into the languages that article was published. This shows the interest and how noted they are globally. This is important because one can be locally famous and the number of edits may be high due to population of that locality. However, in order to understand one's real impact in the world we shouldn't miss the individual's global contribution. Hence, the authors have introduced the Historical Popularity Index (**HPI**). It is a metric that takes into account of the followings:

- the individual's age in the dataset (A)
- the time elapsed since his/her birth, calculated as 2013 minus birthyear
- an L* measure that adjusts L by accounting for the concentration of pageviews among different languages (to discount characters with pageviews mostly in a few languages, see equation)
- the coefficient of variation (CV) in pageviews across time (to discount characters that have short periods of popularity)
- the number of non-English Wikipedia pageviews v^{NE} to further reduce any English bias
- lastly to dampen the recency bias of the data, HPI is adjusted for individuals known for less than 70 years

To show this work in mathematical terms we need to find 4 calculations which are L, L^*, A, v^{NE} and CV respectively.

To calculate L_i^* we first need to define H_i which is the entropy in terms of Page Views. It is calculate as

$$H_i = -\sum_j \left(\frac{v_{ij}}{\sum_i v_{ij}} ln(\frac{v_{ij}}{\sum_i v_{ij}})\right),$$

where v_{ij} is the total page views of individual i in language j. Knowing H_i we can easily calculate the L_i^*:

$$L_i^* = exp(H_i)$$

Authors have defined the A as 2013-Year of Birth and we also know that v^{NE} is equal to the total pageviews in non-English editions of Wikipedia. Lastly, only unknown remaining is that CV. It is calculated as

$$CV_i = \frac{\sigma_i}{\mu_i}$$

where σ_i is the s.d. in pageviews across all languages and μ_i is average monthly pageviews. Knowing the all the unknowns we can calculate HPI as

$$\mathbf{HPI} = \begin{cases} ln(L) + ln(L^*) + log_4(A) + ln(v^{NE}) - ln(CV) & if A \geq 70 \\ ln(L) + ln(L^*) + log_4(A) + ln(v^{NE}) - ln(CV) - \frac{70-A}{7} & if A < 70 \end{cases}$$

Data pre-processing

Since many of the visualizations used in the application rely on the birth year, we excluded those small numbers of entries in which this field was left empty. Due to the high dimensionality of our data set, we resort to reduction techniques to enable visualization in fewer dimensions.

Our intention is to capture a meaningful representation that transcends the geographical domain. In order to do so, we've experimented with the techniques presented in class, i.e. PCA (Tipping and Bishop), t-SNE (Maaten and Hinton), and MDS (Borg and Groenen), but also with UMAP (Uniform Manifold Approximation and Projection) (McInnes et al.), to reduce our data dimensionality to only two components. However, since our dataset is mostly composed of categorical features, techniques such as PCA yield limited results, since these are not suited for categories, even under numerical encoding. On the other hand, techniques such as MDS have proven to be highly costly to compute due to the large number of observations. Some of these results are in the analysis section of our data repository.

In terms of data, our selected approach consisted of using the numerical fields from the geographical coordinates (latitude and longitude) and the years of birth of these personalities, and to combine them with the corresponding domains of expertise and occupation, properly transformed via binary encoding. Additionally, since the numerical fields have different scales, we first apply a normalization scheme to assure that these will have the same contribution to the final result.

We tested the visualizations yielded by each of these methods and settled on the one that produced the most insightful separation of personalities. In t-SNE we saw the formation of clusters directly associated with occupations, which, in turn, were gathered around the same domain of expertise. Within t-SNE, using Wattenberg et al. as a guideline, we also experimented with different parameters, such as perplexity (the trade-off between local and global contributions), number of iterations and also different metrics. Nevertheless, our preferred solution is built mostly using default parameters of sklearn, such as using a euclidean distance and 1000 iterations, modifying only the perplexity, which was set to 40. This in turns makes the reduction more amenable to global contributions than the default, which is 30. However, this value is still within the recommended range proposed by the authors (5-50) (Wattenberg et al.).

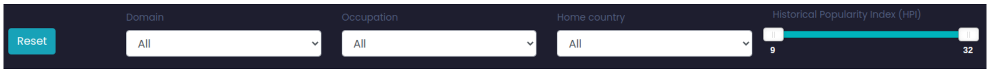
Application Components

Header

In the header portion, we have the following components:

Reset button: as the name suggests it reset all the active filter selections. This interaction is carried out by `jQuery`, that changes all the component values, and through standard functions that refresh the plot status using d3.

Selectors: there are three available selectors (domain, occupation and country), there are all managed through `jQuery`, and the functions within these events changes the inner filter reference and transmit this changes to all plots, that use this information to select the corresponding subset of data (using `d3.filter`). Furthermore, taking advantage of the hierarchical structure, the country selector exhibits two levels: country and continent. There's no additional dependencies (aside from bootstrap for visual effects) that make this possible. This is performed by an encoding and decoding scheme that detects which type of selection is being made and broadcast its value to the corresponding filter operation.



Domain
All
Occupation
All
Home country
All
Historical Popularity Index (HPI)
9
32

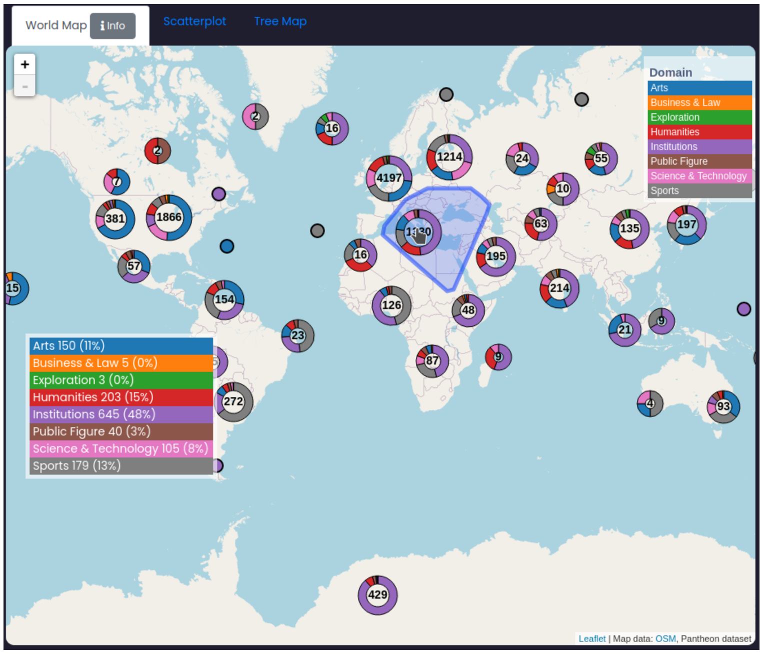
Figure 2: Detailed view of the header

Sliders: there are two sliders in the application. The first is a slider with two handles that limit the minimum and the maximum values of Historical Popularity Index (HPI) used throughout the application. Moreover, in the scatter plot, once the user clicks on a point, that triggers a proximity filter around the point of interest. At this moment, all the above filters are suspended and another slider shows up. This, in turn, controls the maximum distance from the selected point, allowing the user to select more or less data points.

All the above filters communicate with a filter object, that stores the active filter values and their corresponding fields. Once any change event is triggered this object is modified, and this change is used to filter the stored dataset using the `d3.filter` functionalities. Finally, once data is filtered, the plots are updated.

World map

Implemented in `worldMap.js`, the world map is the summarization of personalities positioned by their place of birth and segregated by their domain of expertise. The canvas of the image was built on Leaflet which allows for a better interaction of maps in JavaScript, enabling the user not only to get a distribution per continents and countries, but to go further and explore the relationships by cities as well. Moreover, the application also invites the user to make use of the world map as a filtering tool. If the user clicks on the territory of a certain country, the application automatically filters the dataset to a subset corresponding to the said country of interest. This visualization is built around a GeoJSON file, a standard file extension used to store geographical information, that is fed to Leaflet in the target format. Moreover, once the data points are placed in space, we augment the visualization scheme with a layer of d3, which is responsible for the clustering visualization of such points. One contested aspect of this is the visualization technique chosen. We have placed a donut chart inside the World Map to show the user from our 8 categorical data distributions according to the place they hover, and these are Arts, Business Law, Exploration, Humanities, Institutions, Public Figure, Science Technology and lastly Sports. Despite the strong opposition grounded on literature against the adequacy donut charts, we still opt to use them to not conflate with the numerical count placed in the center of each cluster. Being aware of the drawbacks of such technique, we provide a slight mitigation by displaying a domain distribution once the pointer hovers each cluster. And although instructors have mentioned it several times to not use pie charts, we have found a case where using it will allow users to grasp concepts better. We believe in our use case they are very easy to interpret and aesthetically pleasing.



Arts 150 (11%)
Business & Law 5 (0%)
Exploration 3 (0%)
Humanities 203 (15%)
Institutions 645 (48%)
Public Figure 40 (3%)
Science & Technology 105 (8%)
Sports 179 (13%)

Figure 3: World map chart

The above figure demonstrates how the important people are distributed by numbers inside the circle. One can quickly glimpse the colors to understand the categorical data distributions according to regions. Furthermore, if the user would like to hover the map, we show the regions and their respective numbers with accordance with the categorical data labels as you can see in the above image.

Scatter plot

The scatter plot is implemented in `scatterplot.js` and showcases the dimensionality reduction result, where each personality is represented by a dot. The visualization is further augmented by the color scheme of the points, that indicates the personalities' relevance. The user is able to zoom in and explore the clusters generated by the reduction technique, and once hovering over a particular data point, the application shows some information about that historical figure, such as birth information, occupation and overall relevance.

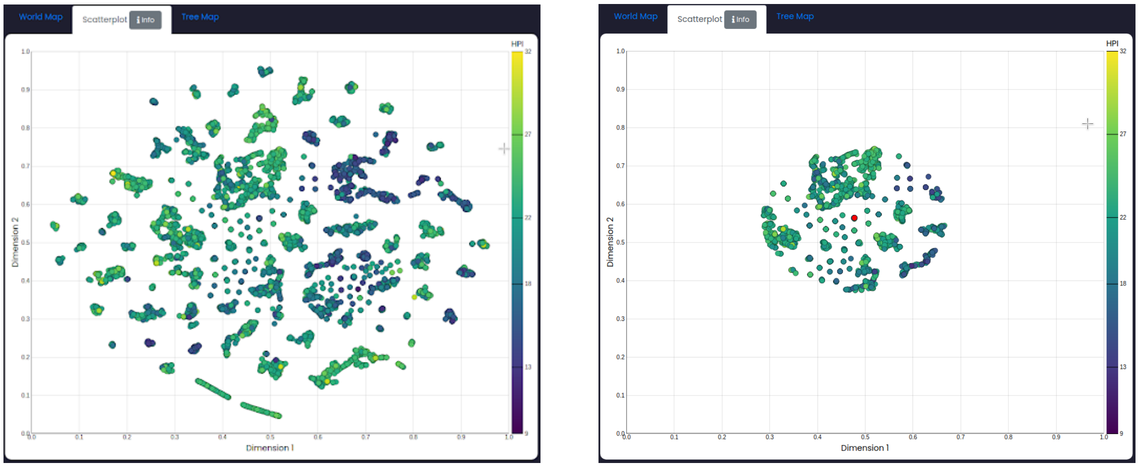
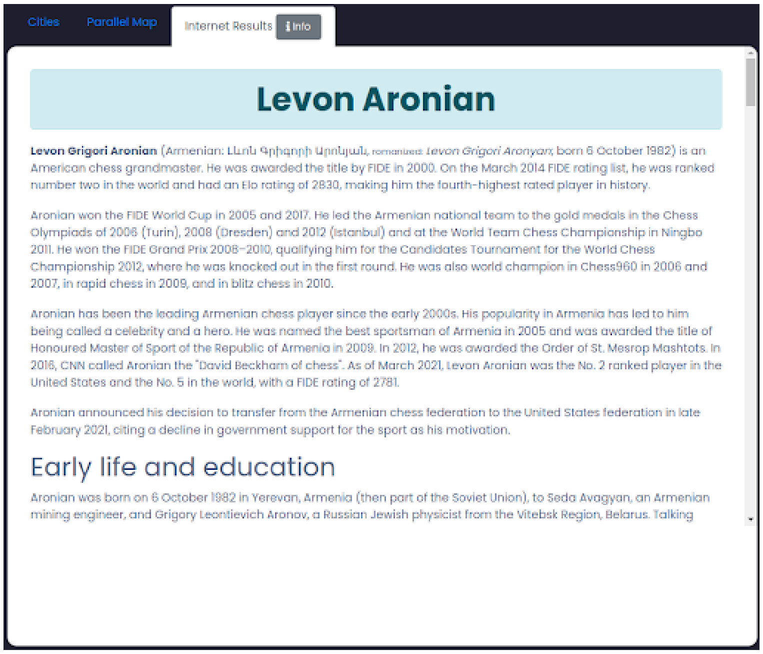


Figure 4: Scatter plot in regular condition (*left*) and under cluster-filtering (*right*)

This visualization is fed by a CSV file, that permeates all other plots except for the World plot. For this particular case, no extra preprocessing step is needed. Once the data and the visualization are loaded, the user is also given the possibility to use the points as a filtering tool. By clicking on a particular data point, we run a k-nearest neighbors routine in the background that fetches the closest points to the reference, which triggers a filtering step in d3 on the aforementioned data structure. When the data is filtered, the user may also adjust the max-distance from the reference through a slider that shows up over the Header, and once satisfied, he may clear on the Exit button in the same section.

Internet Results

When the user clicks on a particular data point within the scatter plot, that also triggers the internet results section. In this, we use the person's name to build a query and scrape the first paragraph of his/her Wikipedia page to supplement the user with biographical information of that person of interest.



Levon Aronian

Levon Grigori Aronian (Armenian: Լևոն Գրիգորի Արոնյան, romanized: *Levon Grigori Aronyan*, born 6 October 1982) is an American chess grandmaster. He was awarded the title by FIDE in 2000. On the March 2014 FIDE rating list, he was ranked number two in the world and had an Elo rating of 2830, making him the fourth-highest rated player in history.

Aronian won the FIDE World Cup in 2005 and 2017. He led the Armenian national team to the gold medals in the Chess Olympiads of 2006 (Turin), 2008 (Dresden) and 2012 (Istanbul) and at the World Team Chess Championship in Ningbo 2011. He won the FIDE Grand Prix 2008–2010, qualifying him for the Candidates Tournament for the World Chess Championship 2012, where he was knocked out in the first round. He was also world champion in Chess960 in 2006 and 2007, in rapid chess in 2009, and in blitz chess in 2010.

Aronian has been the leading Armenian chess player since the early 2000s. His popularity in Armenia has led to him being called a celebrity and a hero. He was named the best sportsman of Armenia in 2005 and was awarded the title of Honoured Master of Sport of the Republic of Armenia in 2009. In 2012, he was awarded the Order of St. Mesrop Mashtots. In 2016, CNN called Aronian the "David Beckham of chess". As of March 2021, Levon Aronian was the No. 2 ranked player in the United States and the No. 5 in the world, with a FIDE rating of 2781.

Aronian announced his decision to transfer from the Armenian chess federation to the United States federation in late February 2021, citing a decline in government support for the sport as his motivation.

Early life and education

Aronian was born on 6 October 1982 in Yerevan, Armenia (then part of the Soviet Union), to Seda Avagyan, an Armenian mining engineer, and Grigory Leontievich Aronov, a Russian Jewish physicist from the Vitebsk Region, Belarus. Talking

Figure 5: Example of the Internet Results section

This functionality was implemented in `researchPlot.js` using `jQuery`, or more specifically `ajax` (js.foundation). Finally, once the Exit button is used, this complementary section is hidden.

Tree map

Tree map is a visualization scheme composed by nested rectangles that represent hierarchical data (Ben Shneiderman). Each occupation is represented by a rectangle whose area is proportional to the sum of the HPI (Historical Popularity Index) of its 5 most prominent figures, which are shown in the bottom-most part of this tree structure. The chart is implemented using `d3.hierarchy`, and is fed by the same CSV file used in the scatter plot chart. Inside `treemap.js` the CSV file is cast to the desired hierarchical format, while the display function handles the creation of elements, as well as the transitions between parents and children.

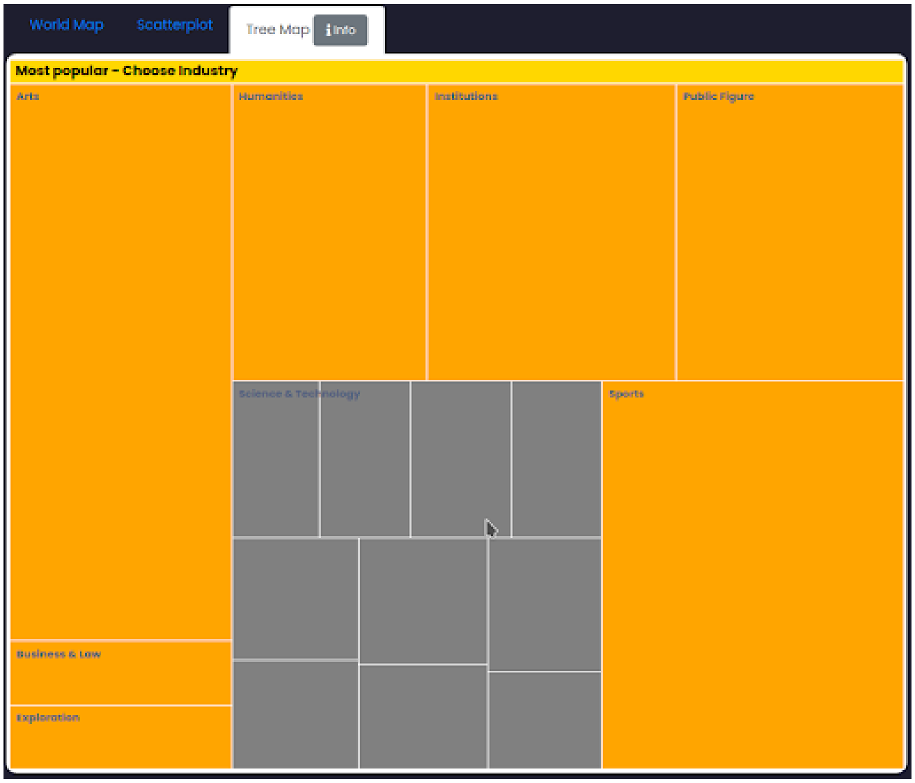


Figure 6: Tree map

Parallel map

For summarizing certain aspects of our dataset, we have chosen a Parallel set chart to visualize these fields of interest and their interactions. We have 4 categorical dimensions that represent the continent of origin, gender, domain of expertise, industry of the historical figures. In each dimension, their size correspond to the proportion of people in the dataset that share that characteristic. Moving along these dimensions, these sections become fragmented representing the relationships between the dimensions. By following paths, one can learn from general information to more specific information about these 4 lines. Moreover, since all of our charts are connected to the filters, the user may also apply certain criteria and observe its unfolding in the parallel map. Inside the `parallelmap.js`, we implemented this using the `d3.parsets` functionality (Davies), where charts are created using the parsets features. Then, a vis variable is created to populate information.

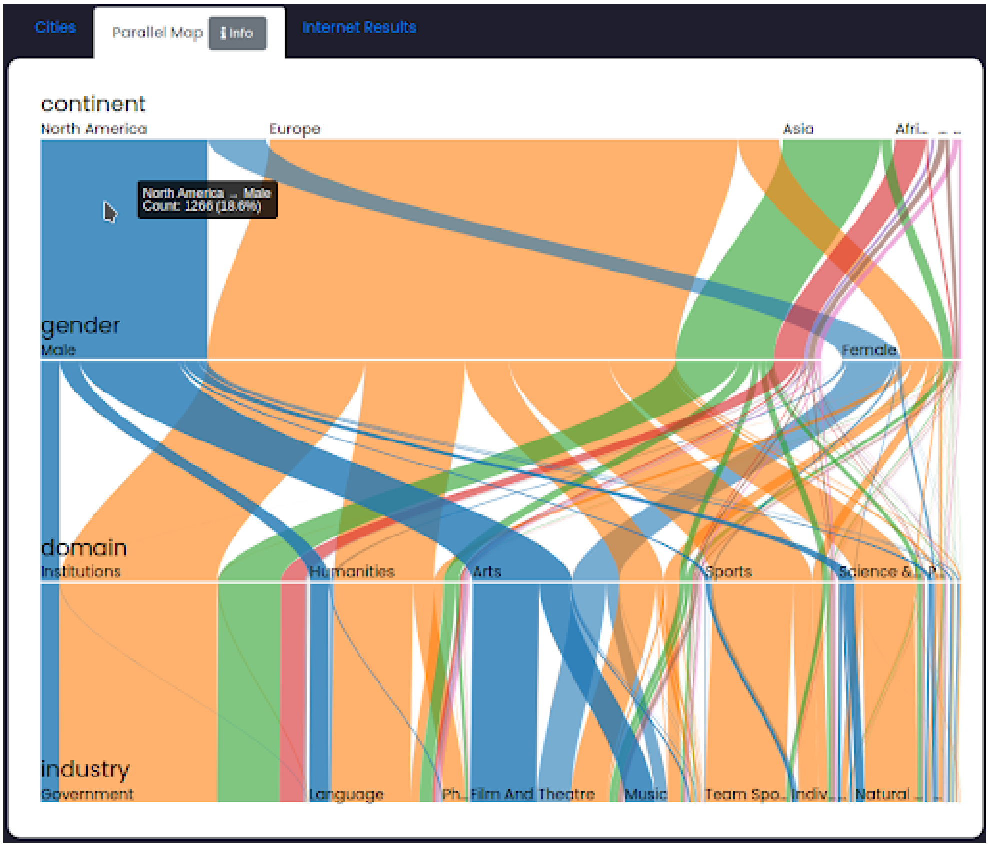
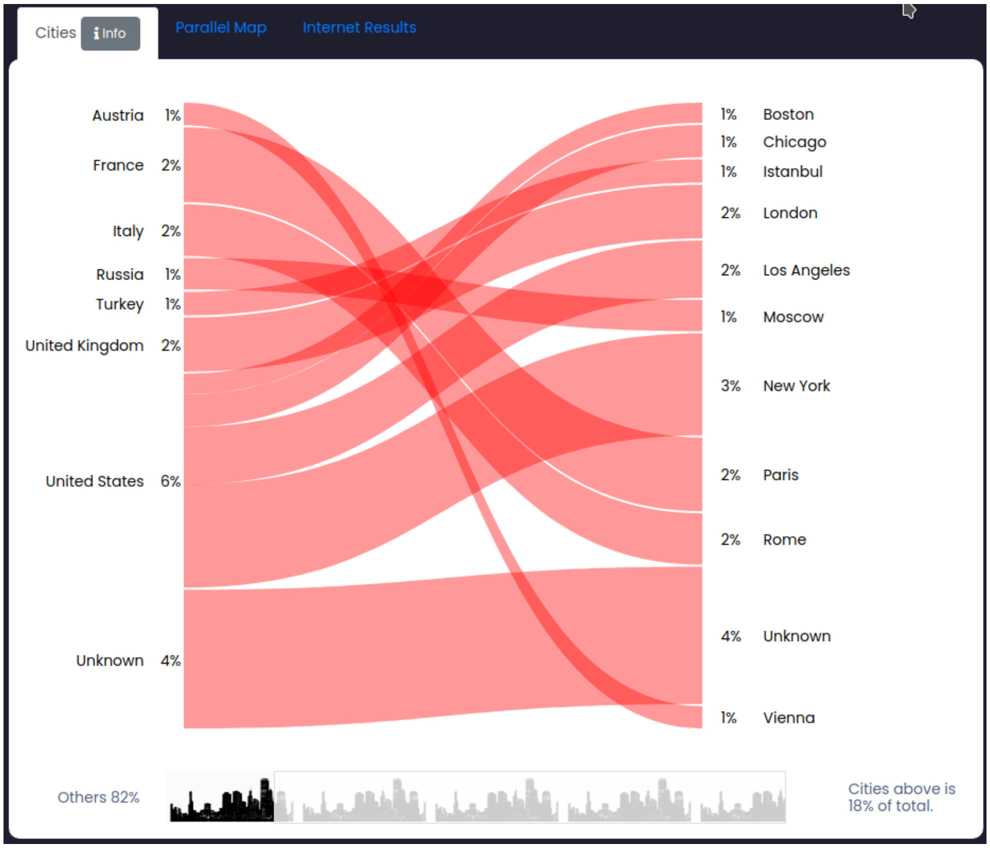


Figure 7: Parallel map chart

Cities map

In addition to the world map, the cities map complement the geographical information presented by providing a summarization of the most popular countries and cities of birth of these historical figures. The visualization is implemented in `cities.js` and built around the CSV file that is first filtered by the user's preferences and the cast a format that summarizes the number of occurrences for each city, indicating also their corresponding country. Once that is done, we build the chart using the bipartite functionality of `viz.js`, that enables the user to interact with the chart once he/she hover the pointer over each city/country.



Cities above is 18% of total.

Figure 8: Cities map

Additionally, due to a very fragmented distribution of cities, we've decided to the leave the less representative ones to the bottom portion of the graph, where we display a bar that represents the percentage of occurrences left outside the chart.

5. Discussion and Conclusion

The application here described showcases analytical and design aspects that enable us to navigate densely populated and highly dimensional problems and extract meaningful information more easily. This project was also an opportunity for us, students, to revisit some concepts and best practices seen in class and put many of these to practice.

In terms of analytics, one of the obstacles unforeseen by us, students, was the limited variety of information presented by the dataset. Despite having an assorted array of features, many of them were very correlated, which has posed a limitation on the types of analysis we could employ. During the execution, we also searched for other complementary databases that could enrich our analysis, but to no luck. Nevertheless, despite these limitations, our application is a small but representative illustration of the power of visual analytics to make sense of an overwhelming amount of information.

References

Gennady L Andrienko, Natalia Andrienko, Daniel Keim, Alan M MacEachren, and Stefan Wrobel. Challenging problems of geospatial visual analytics. *Journal of Visual Languages & Computing*, 22(4):251–256, 2011.

Ben Shneiderman. Treemaps for space-constrained visualization of hierarchies. URL `http://www.cs.umd.edu/hcil/treemap-history/`.

I. Borg and P. J. F. Groenen. *Modern Multidimensional Scaling: Theory and Applications*. Springer Series in Statistics. Springer-Verlag, 2 edition. ISBN 978-0-387-25150-9. doi: 10.1007/0-387-28981-X. URL `https://www.springer.com/gp/book/9780387251509`.

Jason Davies. jasondavies/d3-parsets. URL `https://github.com/jasondavies/d3-parsets`. original-date: 2012-04-23T10:31:47Z.

Young-Ho Eom, Pablo Aragón, David Laniado, Andreas Kaltenbrunner, Sebastiano Vigna, and Dima L Shepelyansky. Interactions of cultures and top people of wikipedia from ranking of 24 language editions. *PloS one*, 10(3):e0114825, 2015.

Evgeniy Gabrilovich, Shaul Markovitch, et al. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *IJcAI*, volume 7, pages 1606–1611, 2007.

Anne Garcia-Fernandez, Anne-Laure Ligozat, Marco Dinarelli, and Delphine Bernhard. When was it written? automatically determining publication dates. In *International symposium on string processing and information retrieval*, pages 221–236. Springer, 2011.

Adam Jatowt, Daisuke Kawai, and Katsumi Tanaka. Digital history meets wikipedia: Analyzing historical persons in wikipedia. In *2016 IEEE/ACM Joint Conference on Digital Libraries (JCDL)*, pages 17–26. IEEE, 2016.

JS Foundation js.foundation. jQuery.ajax() | jQuery API documentation. URL `https://api.jquery.com/jquery.ajax/`.

Daniel Kinzler. Wikisense-mining the wiki. Wikimania, 2005.

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. 9(86):2579–2605. ISSN 1533-7928. URL `http://jmlr.org/papers/v9/vandermaaten08a.html`.

Leland McInnes, John Healy, and James Melville. UMAP: Uniform manifold approximation and projection for dimension reduction. URL `http://arxiv.org/abs/1802.03426`.

David Milne, Olena Medelyan, and Ian H Witten. Mining domain-specific thesauri from wikipedia: A case study. In *2006 IEEE/WIC/ACM International Conference on Web Intelligence (WI 2006 Main Conference Proceedings)(WI'06)*, pages 442–448. IEEE, 2006.

Charles Murray. *Human Accomplishment: The Pursuit of Excellence in the Arts and Sciences, 800 B.C. to 1950*. HarperCollins e-books.

Michael E. Tipping and Christopher M. Bishop. Probabilistic principal component analysis. 61(3):611–622. ISSN 1369-7412. URL https://www.jstor.org/stable/2680726. Publisher: [Royal Statistical Society, Wiley].

Martin Wattenberg, Fernanda Viégas, and Ian Johnson. How to use t-SNE effectively. 1(10):10.23915/distill.00002. doi: 10.23915/distill.00002. URL http://distill.pub/2016/misread-tsne.

Amy Zhao Yu, Shahar Ronen, Kevin Hu, Tiffany Lu, and Cesar Hidalgo. Pantheon 1.0, a manually verified dataset of globally famous biographies. doi: 10.7910/DVN/28201. URL https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/28201. Publisher: Harvard Dataverse type: dataset.

Appendix

pantheon.csv

- en_curid: unique identifier for each individual biography, maps to the pageid from Wikipedia. To map to an individual's biography in Wikipedia, use the en_curid field as an input parameter to the following URL:
- name: name of the historical character (in English)
- Numlangs: number of Wikipedia language editions that each biography has a presence in (as of May 2013)
- birthcity: given birthcity of individual
- birthstate: only applicable in US, the rest of the world is NaN
- countryname: commonly accepted name of country
- countrycode: ISO 3166-1 alpha2 (based on present-day political boundaries)
- countrycode3: ISO 3166-1 alpha3 country code (based on present-day political boundaries)
- LAT: latitude
- LON: longitude
- continentName: name of continent
- birthyear: birth year of individual
- gender: male or female
- occupation: occupation of the individual
- industry: category based on an aggregation of related occupations
- domain: category based on an aggregation of related industries
- TotalPageViews: total pageviews across all Wikipedia language editions (January 2008 through December 2013)
- L_star: measure that adjusts L by accounting for the concentration of pageviews among different languages (to discount characters with pageviews mostly in a few languages
- StdDevPageViews: standard deviation of pageviews across time (January 2008 through December 2013)
- PageViewsEnglish: total pageviews in the English Wikipedia (January 2008 through December 2013)

- PageViewsNonEnglish: total pageviews in all Wikipedias except English (January 2008 through December 2013)
- AverageViews: average pageviews per language (January 2008 through December 2013)
- HPI: Historical Popularity Index

Wikilangs.tsv

- en_curid: for each individual biography
- lang: Wikipedia language code
- name: name in the language specified

Pageviews_2008-2013.tsv

- en_curid: for each individual biography
- lang: Wikipedia language code
- name: name in the language specified
- numlangs: total number of Wikipedia language editions
- countryCode3: ISO 3166-1 alpha3 country code (based on present-day political boundaries)
- birthyear: birthyear of individual
- birthcity: given birthcity of individual
- occupation: occupation of the individual
- industry: category based on an aggregation of related occupations
- domain: category based on an aggregation of related industries
- gender: male or female
- 2008-01 through 2013-12: total pageviews for the given month (denoted by the column header)