ETH zürich

ETH AI CENTER

INTERACTIVE MACHINE LEARNING: VISUALIZATION & EXPLAINABILITY

Project Report

Machine-learning-assisted creation of optimally located STEM programs

Clément Sicard, Kajetan Pyszkowski, Alexis Tabin
D-INFK Faculty, ETH Zurich

Table of contents

Abstract

Based on socio-economic criteria, collected thanks to open data sources, our project aims at helping the City of Chicago's Education Department create new Science, Technology, Engineering and Mathematics (STEM) programs in areas that need them the most. The point of our solution is to help public administrations fight unemployment and crime, but also to give a better access to education in science in neighborhoods that usually have the least access to it.

Introduction

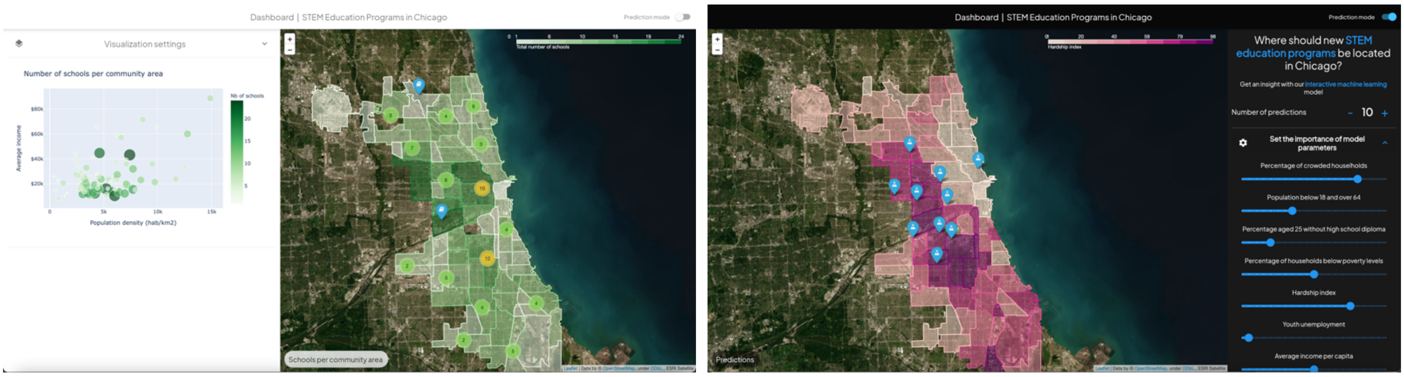
To help fight these problems, we chose the case of the City of Chicago – mainly because it is a city that collects a lot of data about its citizens. Many of them include socio-economic criteria which were of interest in our case – we will describe these sources more widely in the *Data sources and pre-processing* section.

For this purpose, we developed an interactive dashboard, designed for a qualified person in charge of educational matters for the City of Chicago. It allows to both visualize data and predict ideal locations for new STEM education programs.

Dashboard presentation

The interactive dashboard part of the solution has been designed to be used by a qualified user from a governmental department, in order to assist their decision on new locations for STEM programs in Chicago. It can be used in two distinct modes:

1. **Interactive data visualization mode**: Based on the datasets we chose (see *Data sources and pre-processing* section), we generated for each dataset both an interactive map and a collection of interactive graphs, in order to help the user gain a better understanding of the data they are given access to. Note that even though we generated the graphs to be coherent in terms of data visualization, their socio-economic meaning might not be relevant since we have no particular qualification in this domain. Our point here is more to show without pretention a proof-of-concept, assuming that some of the used datasets might not be relevant, even though they appear to, to tackle the issues we mentioned above.



Where should new STEM education programs be located in Chicago?

Get an insight with our interactive machine learning model

Number of predictions – 10 +

Dashboard | STEM Education Programs in Chicago
Prediction mode
Visualization settings
Number of schools per community area
Average income
$80k
$60k
$40k
$20k
0
5k
10k
15k
Population density (hab/km2)
Nb of schools
20
15
10
5
0
1
6
10
15
19
24
Total number of schools
Schools per community area
Dashboard | STEM Education Programs in Chicago
Prediction mode
Hardship index
Predictions
Set the importance of model parameters
Percentage of crowded households
Population below 18 and over 64
Percentage aged 25 without high school diploma
Percentage of households below poverty levels
Hardship index
Youth unemployment
Average income per capita

Figure 1: Screenshots of the dashboard in interactive data visualization mode on the left, and interactive prediction mode on the right

2. **Interactive prediction mode**: The user can use a switch at the top-right part of the screen to enable prediction mode. Here, they can interact with the model, by setting importance to the different features (here, socio-economic criteria, either from raw or from processed data) extracted from the datasets. By default, each feature has a contribution score of 100%, and the user can modify it up to 200% and down to 0%. Then all the scores are sent to the backend, normalized so that they all sum up to 1 and used to weigh the importance of their corresponding feature. Every time a slider's value is modified, the model is instantly updated, and the interactive map gets refreshed with the prediction results. The results can be viewed on top of any of the datasets that is included in the dashboard for a better comprehension of the model's results.

Tech Stack

Let's dig a bit more into technical details. The dashboard is a full-stack app, in which we built both the backend and the frontend. The course staff suggested that we use React.js, built on top of TypeScript as our frontend framework and Uvicorn (web server framework built on top of Python).

However, we instead decided to go with Flask (web server) and GeoPandas (geospatial data manipulation framework, built on top of Python) for the backend, because Flask allowed us to render complex HTML frames, whereas it was more complicated/impossible to do with Uvicorn.

For the frontend, it turned out that suggested React.js has little integration (or a too complicated one) with popular map visualization frameworks, such as Mapbox or Leaflet.js. As two of the group members were already familiar with frontend development, but with Flutter (cross-platform application framework built on top of Dart language, maintained by Google), we decided to go with the latter framework instead. It includes the main specifications of Material Design, uses components and state management in a similar fashion as React.js, so it seemed to be a reasonable equivalence for us to use.

The interactive maps are generated using Folium, which is a Leaflet.js wrapper for Python, and the interactive graphs are generated thanks to Plotly, also a Python library. Both are then served by our Flask API, and then displayed as web views inside the Flutter frontend. We chose to display every geographical data on top of satellite image tiles provided by ESRI. All the code describing these procedures can be found on our GitLab repository[1].

We decided not to use Docker, because we didn't know any efficient way to dockerize a Flutter app (in a naive way, every image build would have to re-download the Flutter SDK, which is 2.0 GB large). However, there is just a single command to run in the terminal to make it run once Flutter is installed – that appeared to be much easier.

Data sources and pre-processing

Most of our data comes from City of Chicago Open Data Portal[2]. It includes unemployment, education, average/median income and crime statistics, to cite just a few. An exhaustive list of the sources can be found in the *References* section[2] [3] [4] [5] [6] [7] [8] [9] [10] [11] [12] [13] [14].

We chose the community area division of Chicago to be our default geographical unit, mainly because most of the data had this granularity, and it arguably made no sense for some of the statistics to be too precisely located.

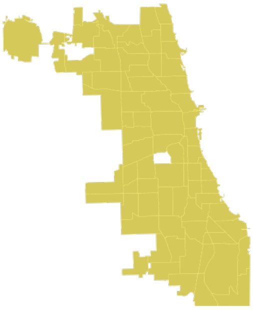


Figure 2: City of Chicago divided by community areas

Then, we grouped the different socio-economic data points by community area and aggregated each of these criteria based on the most coherent statistic. For performance reasons, we gathered the result of the transformations in a single file, `stats_by_comarea.geojson`. Its construction can be retraced by looking at the notebooks on the GitLab repository[1] of the project.

In addition to being the most common data granularity for the data we found, we also chose community-area-wide data because it helped for a better visualization and understanding of the choropleth maps we use in the frontend dashboard in both modes.

For point data, we also decided to use marker clusters, in order not to overload the map with too many data points. It was especially the case with a high-altitude satellite view over Chicago.

Our interactive geospatial inference model

All of the aforementioned data is then used to feed our model. It is largely inspired by a geospatial inference model called Geographically Weighted Regression (GWR)[15] (Fotheringham, et al., 1998). As described above in the *Dashboard description* section, the user sets interactively the weights they give to each feature. Inference is then dynamically performed, and the frontend is nicely refreshed when a change in the weights is triggered. One of the model parameters is also the number of predictions that is outputted, and it is also interactively set by the user. It defaults to 10.

A slight difference with GWR model is that our model is a two-step model: we first decide in which community area we will create a new program, then we decide where we will place a STEM program within the selected community area. This prevented a case in which many predictions appeared at the same time, and for a real-life usage it seemed coherent not to place too many new schools in the same (small) area.

The predictions have as main purpose to allow experts to choose which importance they give to each feature, closely following the human-in-the-loop model. We don't pretend to give exact and precise solutions, and as we'll discuss in the next section, our model still has some flaws.

Discussion & possible improvements

Our model seems to provide accurate predictions, especially when we cross the results with the visualization of the different datasets. However, some improvements can still be made.

For instance, the model has no knowledge of which precise area it can actually create a STEM education program – some of the predictions can be located in places that, in real life, could not host an educational program (e.g., some predictions were located in a river, or in on a landing strip at O'Hare Airport). This issue could be tackled by adding a validation procedure, right before outputting the inference result, that would check if the prediction is not in a protected area (these areas would be carefully selected beforehand, using this dataset [12]) – if the prediction is within one of these impossible areas, then we would discard this one and the next prediction will be then evaluated instead.

Other geospatial inference models could have provided also good/better results. Spatial stochastic processes, such as Gaussian processes are getting increasingly deployed, with very nice results. It could be interesting to implement one of these models and compare the predictions with the ones we have using GWR. Unfortunately, as the results were pretty convincing and seemed coherent, we decided to instead focus more on the interactive visualization part – but that could definitely be a further improvement.

Furthermore, a good thing to note about this dashboard is that even though we decided to go with the example of the City of Chicago, it is highly customizable. It can be easily adapted to any type of dataset someone qualified would find interesting to add and to base the prediction on. We can even enlarge the scope of this project by saying that the project is highly customizable also with the type of data we want to predict: here it is schools, but it could be anything else that has a point location.

Conclusion

To put it in a nutshell, visualization mode helps the qualified user quantify the selected socio-economic criteria – then prediction mode helps them make informed decisions based on the model predictions. Predictions appear to be coherent enough to provide data-driven assistance to decision-making, and we insisted on putting human-in-the-loop model at the center of our solution design.

We can conclude that even though the model is not perfect, it fulfills its role of a decision-making tool to assist public policies, and possibly not only in the case of the City of Chicago. However, we have pay attention to the ethics side, and use adapted datasets to predict socially impactful things – that is why this kind of tool need to be verified by social sciences before being used in order to make sure we take the right things into consideration.

Members contributions

• Clément:	Data pre-processing+	Front-end++	Model –
• Kajetan:	Data pre-processing++	Front-end –	Model+
• Alexis:	Data pre-processing+	Front-end –	Model++

References

[1] Code on our GitLab project repository, https://gitlab.inf.ethz.ch/COURSE-XAI-IML22/STEM-xai-iml22

[2] Open Data Portal, City of Chicago (2022) – Retrieved from https://data.cityofchicago.org/

[3] Community areas geography, City of Chicago (2018) – Retrieved from Open Data Portal, https://data.cityofchicago.org/Facilities-Geographic-Boundaries/Boundaries-Community-Areas-current-/cauq-8yn6

[4] Census Data – Selected socioeconomic indicators in Chicago, City of Chicago (2008 – 2012) – Retrieved from Open Data Portal, https://data.cityofchicago.org/Health-Human-Services/Census-Data-Selected-socioeconomic-indicators-in-C/kn9c-c2s2

[5] Crime (2001 to present), City of Chicago (2022) – Retrieved from Open Data Portal, https://data.cityofchicago.org/Public-Safety/Crimes-2001-to-Present/ijzp-q8t2

[6] Chicago Public Schools – School Locations SY2122, City of Chicago (2022) – Retrieved from Open Data Portal, https://data.cityofchicago.org/Public-Safety/Crimes-2001-to-Present/ijzp-q8t2

[7] Libraries – Locations, Contact Information, and Usual Hours of Operation, City of Chicago (2021) – Retrieved from Open Data Portal, https://data.cityofchicago.org/Education/Libraries-Locations-Contact-Information-and-Usual-/x8fc-8rcq

[8] Average City of Chicago Income Per Capita – Broken Down by Neighborhoods, Chicago Computer Classes, Chi Brander Inc, (2014), Retrieved from chicagocomputerclasses.com, https://www.chicagocomputerclasses.com/average-city-chicago-income

[9] Chicago Population Counts, City of Chicago (2021) – Retrieved from Open Data Portal, https://data.cityofchicago.org/Health-Human-Services/Chicago-Population-Counts/85cm-7uqa

[10] Chicago COVID-19 Community Vulnerability Index (CCVI) – Zip Code Only, City of Chicago (2021) – Retrieved from Open Data Portal, https://data.cityofchicago.org/Health-Human-Services/Chicago-COVID-19-Community-Vulnerability-Index-CCV/2ns9-phjk

[11] Boundaries – Wards, City of Chicago (2015) – Retrieved from Open Data Portal, https://data.cityofchicago.org/Facilities-Geographic-Boundaries/Boundaries-Wards-2015-/sp34-6z76

[12] City-Owned Land Inventory, City of Chicago (2022) – Retrieved from Open Data Portal, https://data.cityofchicago.org/Community-Economic-Development/City-Owned-Land-Inventory/aksk-kvfp

[13] Crimes, City of Chicago (2020) – Retrieved from Open Data Portal, https://data.cityofchicago.org/Public-Safety/Crimes-2020/qzdf-xmn8

[14] Crimes, City of Chicago (2021) – Retrieved from Open Data Portal, https://data.cityofchicago.org/Public-Safety/Crimes-2021/dwme-t96c

[15] Fotheringham, A. S., Charlton, M. E., & Brunsdon, C. (1998). Geographically Weighted Regression: A Natural Evolution of the Expansion Method for Spatial Data Analysis Environment and Planning A: Economy and Space, 30(11), 1905–1927 https://doi.org/10.1068/a301905