Process documentation

Author: Sampsa Rannala (sampsa.rannala@gmail.com)

A rough outline of how I approached and implemented this visualization project, followed by a general commentary and thoughts on the process.

Note on how to view the project results

Interactive code execution features of Jupyter Notebooks requires user to deploy a virtual conda environment (to make sure that all python package requirements are met). Instructions for this can be found in README.md. I have also provided static files in **reports** folder for a quick look at results. The actual *.ipynb solution notebooks (separate notebook for question 1 and question 2) can be found inside notebooks folder.

Outline of project steps

1. Initial look at datasets (quality, content, potential)
   - country code data needs mapping to country name
   - Question 2 dataset has empty rows, incomplete rows that will be purged
   - both datasets have interesting KPIs that need some formatting to be interpreted as numbers by Pandas dataframe (to be done in SQL)
2. Prepare project structure (deploy git, lib folder, environment.yml etc.)
3. Git remote via DigitalOcean private VPS
   - Used for backups and versioning
   - The implementation is cross-platform, both Windows and Mac laptops used in testing and development
4. SQLite scripting to push datasets to database
   - implement SQLite db & connection
   - create tables
   - source country code mapping data from API
   - insert datasets into SQLite tables
5. Transformation of datasets to enable easy analysis
   - for Question 1 this was done in one SQL query (via Jupyter Notebook)
   - for Question 2 this was done in python backend script
6. initial profiling of datasets in Jupyter Notebooks (also some excel, Tableau work to get insight)
7. Developing solution notebooks for datasets with Altair visualization & some Pandas, ipywidgets work
   - Solution_to_question1.pynb
   - Solution_to_question2.pynb
8. Finalizing documentation, preparing HTML output (in case recipients prefer that to deploying interactive Notebook environment)

Commentary on chosen approach

SQL and SQLite database

SQL is more familiar to me than Pandas so I decided early on to build a SQLite database for the data prep stage. In retrospect, it feels like bit heavy lifting for simple CSV datasets. On the other hand, the implementation is flexible, stable, and can be leveraged for larger projects as well. I enjoyed the python backend work a lot.

Jupyter Notebook & JupyterLab

Jupyter Notebook is very commonly used in data science projects. It provides a very useful and intuitive way to interactively execute code cells and view the output, including data visualizations.

Python libraries used for interacting with and visualizing data

Based on my earlier research Altair seemed like a good fit for this project. It is based on Vega which builds on libraries such as D3. Altair gives us a pythonic API to generate Vega specification and pass it on to compatible renderer for visualization. The same Vega standard JSON can be reused in any modern Javascript-enabled platform to produce the same visual output.

My previous work with Altair was limited to a few quick tests, so this was very much an exciting learning project for me. I first intended to answer question 2, but after initial visualizations I found the dataset to be quite limited. United States has such a large playerbase that other countries seem like a tiny sideshow in comparison. I quickly moved to question 1 after getting some basic insight into the dataset with an hour of exploration in Tableau.

In the end I also returned to question 2 for a second look, and found it to be a fruitful case for showcasing some filtering interactions. But still I consider the solution to question 1 the main product of this project. Second notebook is bit more hacky.

Conclusions

In retrospect, Altair was a good choice. I was surprised at how quickly and effortlessly the visuals could be restructured and played around with. After developing functional Altair code for one project, the approach would be easy to leverage and develop further as new datasets and analytical projects come in. Given more time, I would probably experiment with other Vega-related projects to add even more interactivity and dashboard-like experience into the mix:

Vega projects

Also, I did not experiment with online sharing of notebooks or Altair visualizations. The web is of course full of opportunities to deploy analytic solutions for a wider audience.