GrowthViz

GrowthViz was developed in partnership between the Health FFRDC and CDC, with feedback from leading health researchers, to support post-processing and data visualization of growthcleanr output.

The objective of this tool is to allow users to conduct post-processing and data visualization of growthcleanr output. growthcleanr is an automated method for cleaning longitudinal pediatric growth data from EHRs. It provides an environment that includes graphical user interfaces as well as interactive software development to explore data.

Contents

- Git Repository Information
- GrowthViz Purpose
- Background
- Simple Install
- Sample data and first run testing
- Docker Install

Git Repository Information

The latest code for this project should run `GrowthViz.ipynb`.

The notebook requires Python 3, Jupyter Notebook, Pandas, Matplotlib and Seaborn. Some widgets also require the Qgrid extension enabled in Jupyter. The `.csv` files in the repository are the source data required to run the notebook. Custom data should replace these files in the same format. For more details see the simple install instructions below.

GrowthViz Purpose

The objective of this tool is to allow users to conduct post-processing and data visualization of growthcleanr output. growthcleanr is an automated method for cleaning longitudinal pediatric growth data from EHRs. It is available as open source software. GrowthViz is to be used **after** a data set has been run through growthcleanr.

Background

As stated in Automated identification of implausible values in growth data from pediatric electronic health records:

> In pediatrics, evaluation of growth is fundamental, and many pediatric research studies include some aspect of growth as an outcome or other variable. The clinical growth measurements obtained in day-to-day care are susceptible to error beyond the imprecision inherent

in any anthropometric measurement. Some errors result from minor problems with measurement technique. While these errors can be important in certain analyses, they are often small and generally impossible to detect after measurements are recorded. Larger measurement technique errors can result in values that are biologically implausible and can cause problems for many analyses.

GrowthViz uses data sets that were produced by growthcleanr. The tool expects the output to be in a CSV format that is described later on in the notebook.

GrowthViz is a Juypter Notebook. It provides an environment that includes graphical user interfaces as well as interactive software development to explore data. To achieve this, GrowthViz references different software languages and packages: - Python programming language is used to import, transform, visualize and analyze the output of growthcleanr. Some of the code for the tool is directly included in this notebook. Other functions have been placed in an external file to minimize the amount of code that users see in order to let them focus on the actual data. - Data analysis is performed using NumPy and Pandas. The output of growthcleanr will be loaded into a pandas DataFrame. GrowthViz provides functions for transforming DataFrames to support calculation of some values, such as BMI, as well as supporting visualizations. It is expected that users will create views into or copies of the DataFrames built initially by this tool. Adding columns to the DataFrames created by this tool is unlikely to cause problems. Removing columns is likely to break some of the tool's functionality. - Visualization in the tool is provided by Matplotlib and Seaborn. Users may generate their own charts with these utilities.

Simple Install

Anaconda is an all-in-one package installer for setting up dependencies needed to run and view GrowthViz.

1. Install Anaconda

- Follow install instructions found here for installation.
- Opt for the Python 3.7 version
- The windows install instructions are step-by-step and will get everything set up properly for the project.

2. Download the GrowthViz project as a zip file using the "Clone or download" button on GitHub.
3. Unzip the GrowthViz zip file to have access to all of the source files for the Jupyter notebook.
4. Run the Anaconda Navigator that was installed during Step 1 (go to Start>Anaconda Navigator). This may take a while to load.
5. Before Launching the Jupyter Notebook application (shown on the home page), download one additional dependency "Qgrid". To do this:

- Click 'Environments' on the left.

- Type 'Qgrid' in the `Search Packages` text box in the top center of the screen. If it shows up with a green checkbox, proceed to Step 6.
- If it does not appear:
  - Change the 'Installed' drop down in the top center of the application to 'Not Installed' and type in 'Qgrid' in the search bar on the right.
    * If Qgrid still does not show up click 'Update Index...' button next to the search bar. This may take several minutes. Once it is done search for Qgrid again.
  - Check the box to the left of Qgrid in the list and click the green 'Apply' button in the lower right corner.
  - Confirm the installation dialog. Installation may again take several minutes.
  - Once installation is successful, click on the 'Home' in the upper left navigation panel and proceed to Step 6.

6. Click 'Launch' under the 'Jupyter Notebook' icon. This will open the Jupyter Notebook interface in your default browser.
7. Within the browser, navigate to the `GrowthViz-master` folder you downloaded and unzipped in Step 2 (likely found in your Downloads/ folder). Click on `GrowthViz.ipynb` to run the Python notebook.
8. [**Optional step for testing the notebook**] Once the notebook is open, click the 'Run' button to step through the various blocks (cells) of the document, OR click the 'Cell' dropdown in the menu bar and select 'Run all' to test the entire notebook all at once.

Sample data and first run testing

By default when you reach Step 6 of the Simple Install instructions above the notebook will use sample data loaded from the `.csv` files located in the GrowthViz-master project.

To ensure that all of the necessary example files are present, run the `check_setup.py` script.

Docker Install

Docker allows for the ability to download GrowthViz and its dependencies in an environment.

1. Download GrowthViz-Docker with the following command:

- `docker run -it -p 8888:8888 -v [data-path]/growthviz-data:/usr/src/app/growthviz-data mitre/growthviz`
- Replace the `[data-path]` with a directory path you choose on your local computer. For instance, I choose: `~/Documents` which means that a folder named `/growthviz-data` will be created in my documents folder. When I want to input my own data in to GrowthViz, I can simply drop my CSV files in this `/growthviz-data` folder.

2. View GrowthViz

- After running the above command, several lines of text will appear. Choose the third URL in this text and navigate to it in a web browser.
- The URL should be in the format: `http://X.X.X.X:8888/?token=XXX...`
- Within the browser, click on the file `GrowthViz.ipynb`. This will open a new window with the GrowthViz Jupyter Notebook.

3. Run GrowthViz

- You can choose to either click the `Run` button to step through the various blocks (cells) of the document, OR click the 'Cell' dropdown in the menu bar and select 'Run all' to test the entire notebook all at once. However, this will run with the default sample data. Step 4 will explain how to use your own data.

4. Input Your Own Dataset CSVs

- To input your own data, drop a file `[name-of-your-file.csv]` into the `/growthviz-data` folder you created in step 1.
- Then, navigate to Cells 7 and 28 and replace:
- `cleaned_obs = pd.read_csv("sample-data-cleaned.csv")` with
- `cleaned_obs = pd.read_csv("growthviz-data/[name-of-your-file.csv]")`
- Where [name-of-your-file.csv] is the input CSV file you placed in your `/growthviz-data` folder.

Output boxes   When you run all cells (see Step 8 above) `Out[#]:` boxes will appear in the notebook below the `In[#]:` code cells. These outputs are the result of the functioning code blocks on the data. The out blocks will often be interactive charts and graphs used to explore the growthcleanr data. Descriptions of each `Out[#]:` block can be found in the text sections above the `In[#]:` blocks.

Notice