Pipeline.

The pipeline chosen for this project consists of two parts. The first part generates the data and the secund fits the regression for use in the end-user interface.

Data Model

The main philosophy behind our pipeline is versatility. At the start of the project, it was unclear to us what dataset we would be using. Three options had been presented to us for the data acquisition:

1. Pull from a rest API.
2. Pull from an SQLite database.
3. Import a .csv

At the start of the project there was some confusion amongst our team whether we ought to build for a static or a dynamic dataset. In an early stage of the project, we made the design choice to start building a pipeline that could import from the three sources. Because we were unsure if the supplied data would be static or dynamic, we made the choice to automate the generation of different datasets prior to the EDA conclusions what dataset to use for the final version.

The data models that made it to the next stage have an enumerated name structure df1 to df5.

For use in the next stage of the project the data model generated its output in .csv and stores it in an SQLite.db

Data analyses determined that df4 would be the model we’d be using for the final regression model. The other models are currently commented out to reduce system load and clutter.

Regression Model

The versatile design philosophy was at the forefront for our design of the regression file.

The EDA made it clear to us that the path ahead led to a linear regression model. We started to craft our regression model with versatile data path in and out.

For input the standard is currently set to df4 from SQLite. Import from .csv is currently commented out. With a simple filename change in the model we can use the model to run regression on our other df’s.

Our regression model’s output is versatile as well it is both stored in a python pickle format (.pkl) as in an SQLite database. Allowing for the diverse life expectancy interfaces our group is proud to present.