Software Engineering Project

Visualization of high-dimensional data

Kiljanek Adrian

Zdziebło Arkadiusz

Stasica Krystian

Ryczek Bartosz

Creation date:	18.10.2021
Modification date:	01.12.2021
Version:	1.2.0

Supervisor: Jakub Nalepa, Ph D.

Content

Document history 2

Project topic 3

Project description 4

Use case modeling 4

Class diagram 6

Dictionary 6

References 7

Document history

Version	Authors	Description
1.0.0	All	Basic project concept.
1.0.1	BR	Correction in project description + dictionary
1.0.2	AZ, BR	Extension of project description
1.0.3	All	Project description correction
1.1.0	AK, AZ, BR	UCM
1.1.1	BR	UCM correction
1.2.0	BR, KS	Class diagram

1. Project topic

The project is an application to visualize high-dimensional data.

High-dimensional means that the number of dimensions is staggeringly high and the number of features can exceed the number of observations. This kind of data has multiple dimensions therefore it cannot be visualized with a simple 2-dimensional or 3-dimensional graph as a bar plot. Such data can be visualized using: radar plot, parallel coordinates plot, scatter plot matrix.

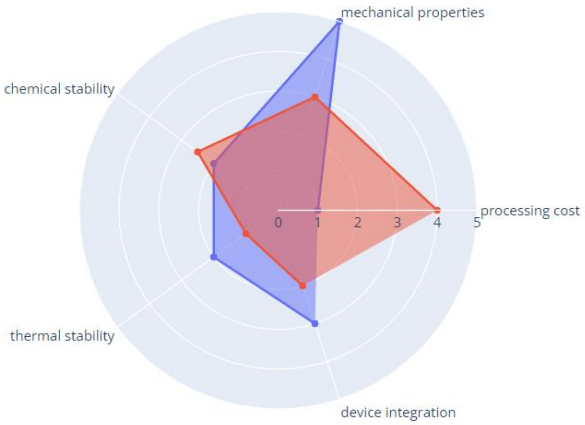


Fig.1 Radar plot[3]

High dimensional data is used in a variety of fields. It is mainly used in healthcare data sets where the number of features for a given observation can be massive (i.e. blood pressure, resting heart rate, immune system status, surgery history, height, weight, existing conditions, etc.). It can be also used in game statistics, object description, financial data and many more.

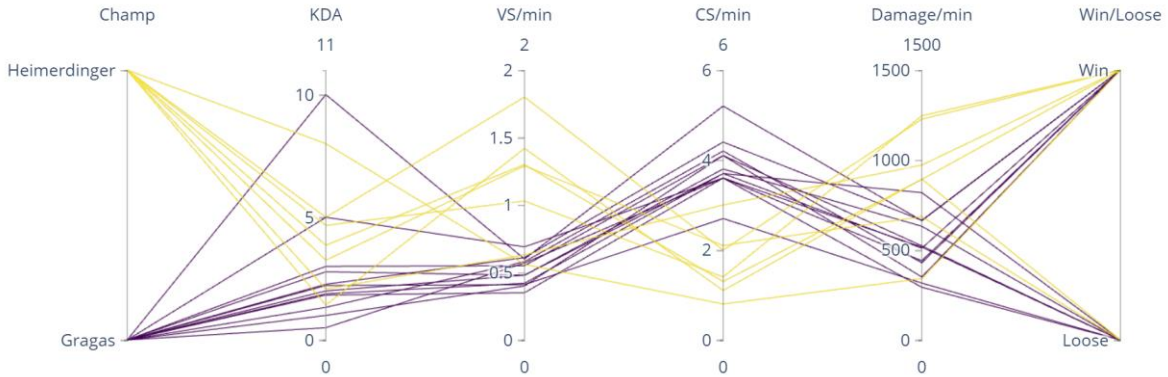


Fig.2 Exemplary parallel coordinates plot presenting game statistics comparison of two champions

High-dimensional data graphs are used to visualize the data so it is simple to read, allows for easy comparison, finding similarities/differences as well as anomalies in the data.

2. Project description

Project main feature is based on data visualisation.

The data consists of parameter's names and numerical values of the features of certain observations.

Application will have a simple GUI consisting of: buttons for input data, result of visualization and way for user to export the visualization. The visualization will be primarily exported as an .html file. Where the user will be able to manipulate the visualization and export it as an .png file. The scope of manipulation contains the zoom in and zoom out options as well as changing the order parameters and highlighting chosen objects.

When the data will be inconsistent or in case of lacking parts of data, there will be a pop up window which will let the user choose how to deal with the problem. For example to skip the missing fields or not include the object with missing values in solution at all. Program will also be able to tell the user which parts of data were lacking.

The program should support multiple input formats or databases, for example: '.txt', '.xls', '.xlsm', '.csv'.

3. Use case modeling

The diagram includes one actor (user), nine use cases (visualising, providing input, verifying input, correcting input, saving data to data structure, choosing type of visualization, displaying plot, manipulating output, exporting output), seven include dependencies and three extend dependencies and the associations between the performing actor and use cases.

Main case of the application is visualizing data, which converts numerical data to the plot.

User provides input, which is verified and in case of some inconsistencies in the data, the user can correct it. After optional corrections data is saved in data structure. Then the user choses type of graph. Users can also choose the path to where the .html file containing visualization will be saved.

The output consists of a .html file. The file presents visualization of the graph and makes it possible to zoom, highlight specific objects, change order of parameters and export it as a .png file.

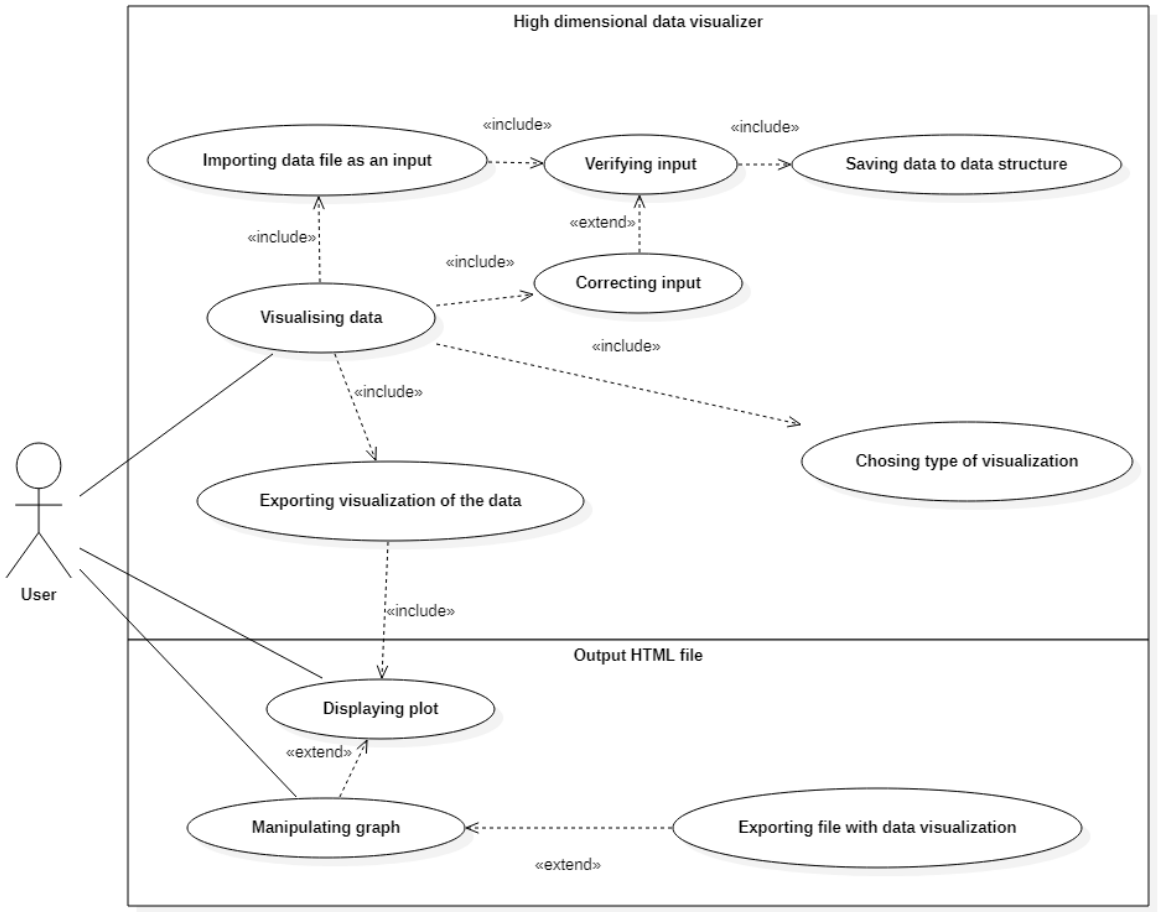
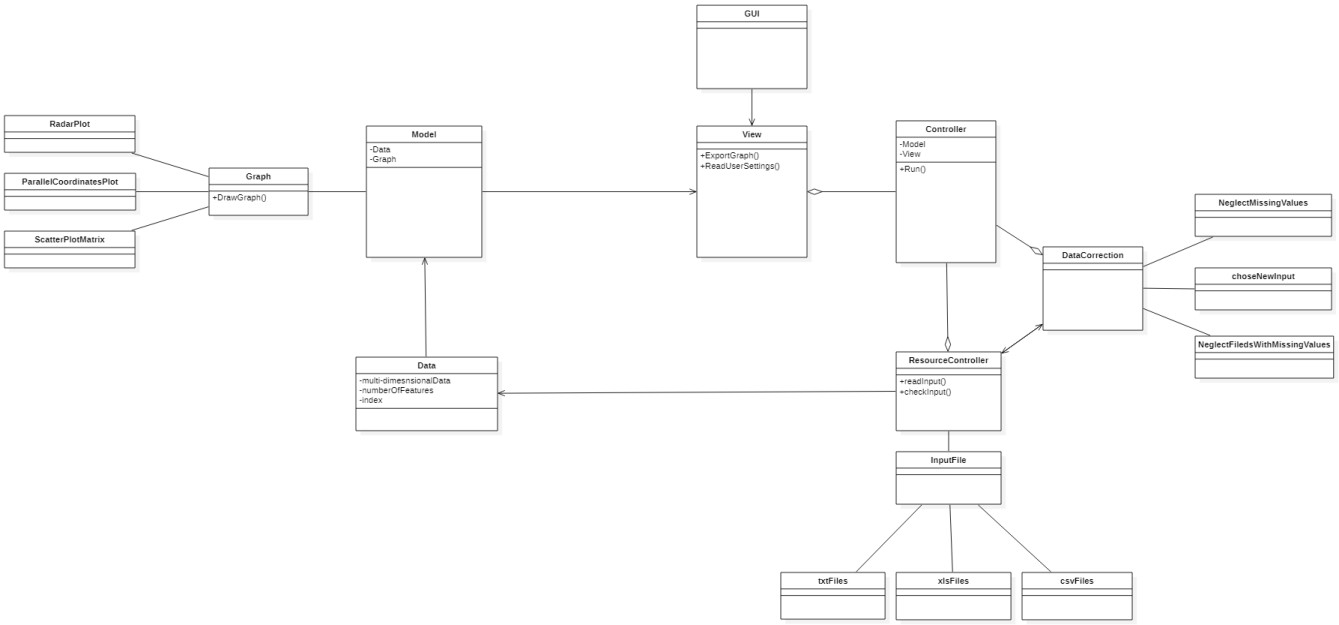


Fig.3 Use Case Diagram of a High-dimensional Data Visualiser use-case model.

Correcting input use case specification. The user will be able to:

- chose to not visualize observation with lacking data
- chose to not visualize only the lacking feature
- chose to reload the data set

4. Class diagram



The application is based on the Model-View-Controller pattern:

- View class is responsible for graphical user interface and reading data provided by user
- Model class is responsible for storing data and drawing graphs
- Controller combines model and view and add data correction

Additional descriptions:

- DataCorrection, Graph and InputFile are template classes, which will work like an interface. Those classes will be a base for more specific classes handling for example specific file format or creating one type of a graph.
- ResourceController is class responsible for initializing and changing data. It will input the data to a data structure as well as check its correctness and make corrections if necessary.

5. Dictionary

Feature - also referred to as “variables” or “attributes”, represents a measurable piece of data that can be used for analysis
High-dimensional data - data where number of features or covariates can be larger than the number of independent samples.
Observation - also referred as “object”, one occurrence of something you’re measuring
Parallel coordinates[1] - way of visualizing high-dimensional datasets. Plot consists of a vertical bar representing a variable. Values are plotted as a series of lines connected across each axis.
Radar chart[2] - also known as “spider plot”, a chart that consists of a sequence of equi-angular spokes, called radii, with each spoke representing one of the variables.

6. References

[1] https://en.wikipedia.org/wiki/Parallel_coordinates
[2] https://en.wikipedia.org/wiki/Radar_chart
[3] https://plotly.com/python/radar-chart/