Making Your Graphs Come To Life

Monika Huhn, Jesper Havsol, Daniel Goude, Volker Schnecke, Christoffer Stedt, Martin Karpefors

monika.huhn@ astrazeneca.com

2nd of July, 2015
Introduction - AstraZeneca

Pharmaceutical company with research spanning from discovery to late stage clinical research

Advanced Analytics Centre

- ~25 people
- Different skills coming together to provide novel solutions to data-driven clinical research problems in drug development, through applied data science.

2nd of July, 2015: Making Your Graphs Come To Life
Introduction - Making Your Graphs Come To Life

Animation application using D3 and R Shiny, in use at AstraZeneca to explore clinical data.

Structure:
- Motivation
- Tools
- Implementation
- Demo
- Conclusions
Clinical Data

- Many different variables
- Recorded over time

2nd of July, 2015: Making Your Graphs Come To Life
Technical hurdle:
Data not easily and directly accessible for medical experts.
Solution

**Interactivity:**
Applications that enable easy data exploration.
Benefits & risks of animated data

Inspired by Hans Roslings famous Gapminder example

Benefits:
- Additional variable can be displayed in a meaningful way
- Allows spotting of temporal patterns

Risks:
- Can distract from data
- User has to be able to follow the individual data points over time
- Can it be interactive enough for exploratory research?

→ We tried to take these points into account in the design
Motivation

Tools

Implementation

Demo

Conclusions
• Web Application Framework for R

• Developed by RStudio team

• Allows web development using R but can also be extended using HTML/css/javascript

shiny.rstudio.com
• **Javascript library** developed by Mike Bostock
• Manipulate documents based on **data** by **binding** to DOM elements
• Beautiful visualizations when used in connection with **svg** elements
• Very advanced **transition** capabilities
• Advantage: absolute **flexibility**
• Disadvantage: longer learning process

d3js.org
Implementation
Combination of tools with different strengths

Pure javascript implementation run into trouble with data manipulations → use different tools and focus on their strengths:

- **R**: Data manipulation
- **Shiny**: Easily creating a web page frame with reactive elements
- **DB**: Beautiful graphics and transitions
Implementation

Main question: How to connect D3 and Shiny?

Server side:
Data preparation using R

Send data using handler function

Client side:
D3 animation updated

2nd of July, 2015: Making Your Graphs Come To Life

AstraZeneca
2nd of July, 2015: Making Your Graphs Come To Life
Animation

Choose x-axis variable:
- fasting_plasma_glucose

Choose y-axis variable:
- hemoglobin_HbA1c_percent

Choose color variable:
- gender

Choose size variable:
- weight
- day
- age
- daily_insulin_dose
- weight
- bmi
- hemoglobin_HbA1c_percent
- fasting_plasma_glucose

Start Animation

2nd of July, 2015: Making Your Graphs Come To Life
Follow individuals and filter

- Click marker to trace
- Click legend to filter

2nd of July, 2015: Making Your Graphs Come To Life
Conclusions

- Great potential in connecting the different tools
- Interactive features enable non-technical users to explore data
- Animation conveys a more powerful message than static plots could
- Enables temporal pattern detection
  → supported findings about relation between HbA1c and weight change that are currently being published
References and further reading:

AstraZeneca (http://www.astrazeneca.com/)

Gapminder (http://www.gapminder.org/)


R Shiny (http://shiny.rstudio.com/)

D3 (http://d3js.org/)