

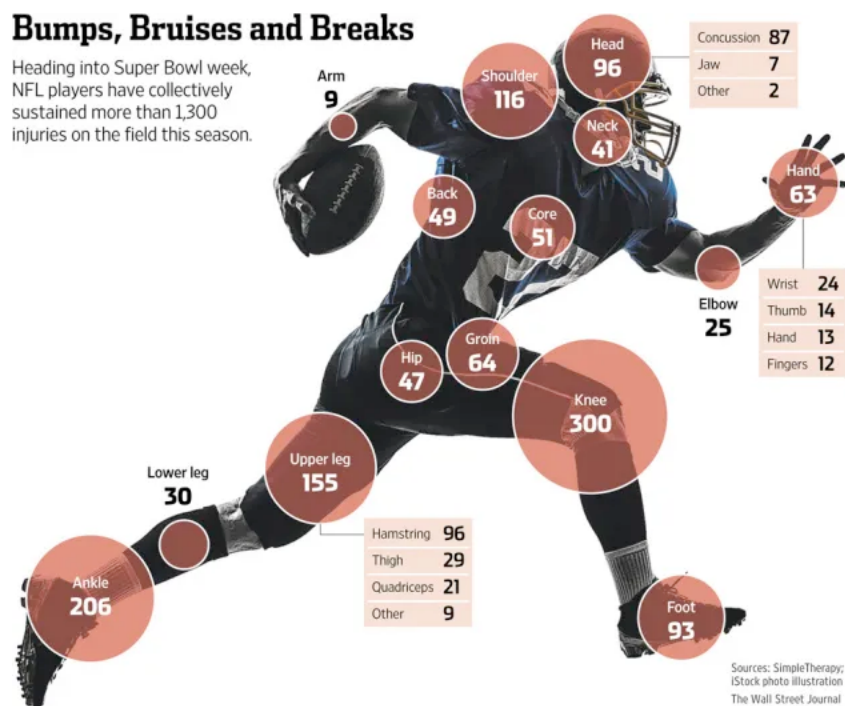
Exercise 3

(20 points)

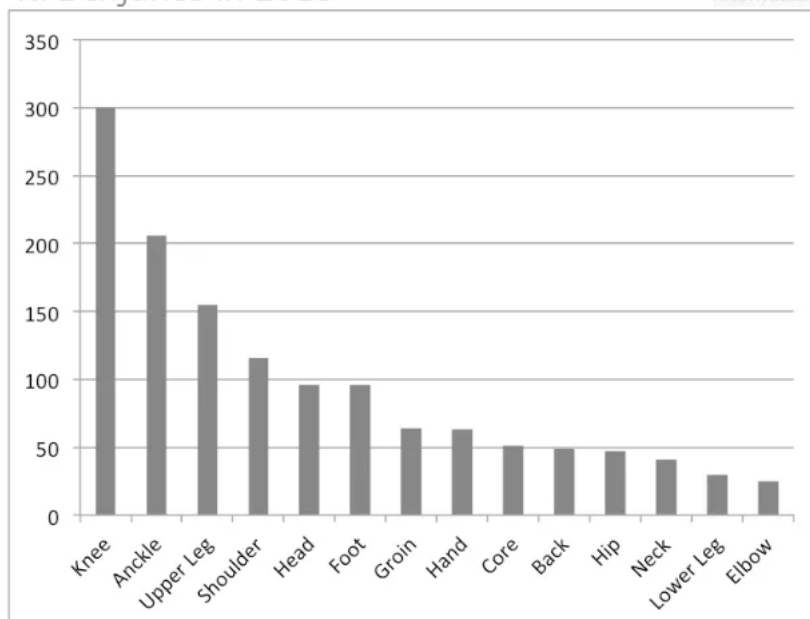
Due: 6.5.2024 10:00AM

Task 1 (10 points) – The following two visualizations show the same data about NFL injuries. Answer the following questions:

- Which one focuses on *expressiveness*, and which one on *effectiveness*?
- What are the different marks and visual variables used in both visualizations?
- How do the different marks and variables affect the tasks that the visualizations support?

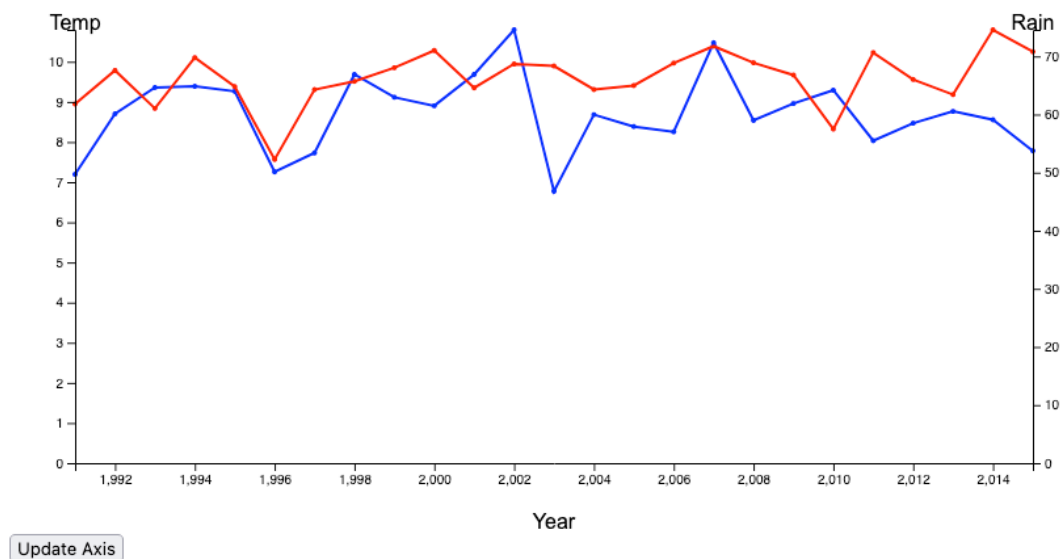


NFL Injuries in 2013



Task 2: Create a Line-Chart with D3.js, and test different scales!

Yearly Rain and Temperature in Germany from 1991 to 2015



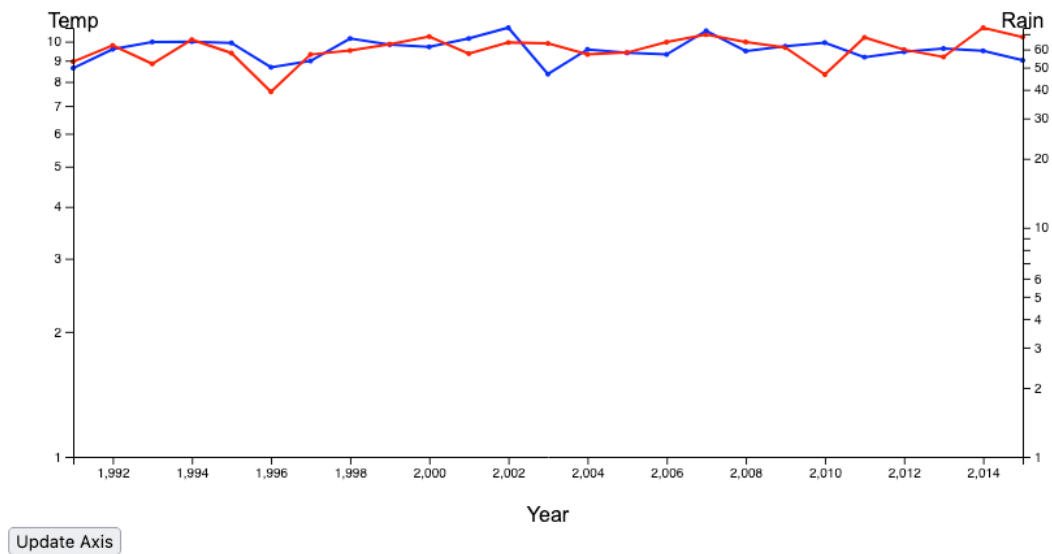
Goal of this exercise is to implement a changing line-chart with D3.js! Attached to this exercise you will find a folder called *linechart* which contains an unfinished implementation of a line-chart. Your task is to finish the implementation such that opening the *index.html* shows a line-chart as depicted in figure above. The figure shows trends (averaged by year) of the temperature (red line) and rainfall (blue line) in Germany from 1991 to 2015. To finish the implementation, follow the steps described as comments within the dedicated file. Each comment starting with *TASK* indicates a position you have to add code. Within the folder *linechart* there are 4 files:

- **index.html**
The main entry point of the visualization. When your implementation is finished, opening this file with a browser should show a *linechart*. Currently, opening the *index.html* will show a heading but no visualization.
- **index.js**
The main JavaScript entry point which implements a line-chart.
- **index.css**
Implements CSS Rules for specific elements.
- **data.js**
Reflects the dataset we want to visualize. It represents monthly rainfall and temperature of Germany from 1991-2015. Attribute 'pr' is Rainfall in (mm) while 'tas' is the average temperature.

Hints: Be sure to use d3.js instead of the raw JavaScript methods. Apply concepts such as method-chaining and anonymous functions whenever possible.

The exercise contains a button that, if you complete the exercise correctly, changes the scale of the visualization between Linear and Logarithmic. While the image above represents how the visualization looks using a linear scale, this is how it should look like using a Logarithmic scale:

Yearly Rain and Temperature in Germany from 1991 to 2015



Task 2a (1 point) – Import d3 in index.html

Task 2b (3 points) – Follow the steps in index.js to create a line chart

Task 2c (3 points) – Get the visualization to change when the user presses the “Update Axis” button

Task 2d (3 points) – Answer the following questions:

- What differences do you notice between the linear and the logarithmic scale?
- Can you think about cases where a logarithmic scale would be more informative than a linear one?

Submission: Zipped folder including all files of the programming exercise (index.html, index.js, index.css, data.js) and a PDF/text file with the answers to the theoretical questions.

Please find yourself in Groups of **2 Students**. Only 1 member of the group must submit the exercise in ILIAS.