Master Degree in Data Science and Business Informatics

Visual Analytics 2019/2020

Italian Boreholes Board

Bergantino Gianluca – 568665

Index

1. INTRO

The Italian Geothermal Database is the largest collection of Italian geothermal data. The database collects geographical and geothermal information related to boreholes (or wells) and springs spread over the Italian territory. It includes wells producing thermal fluids, such as hot water, steam or gas, geothermal or hydrocarbon exploratory wells, water wells above 30°C, unproductive wells, thermal springs and other manifestations. All available geological, drilling, hydrogeological, geophysical and thermal data have been collected from boreholes.

It is the goal of this work to present a web application based on a interactive dashboard which enables the user to explore the data and get insights on wells in an intuitive and easy way. The undergoing idea beneath the design of the dashboard is observing boreholes at two different levels of view, as aggregates and single entities: the user can get geothermal and geographical insights on aggregations of wells and select a well to dig in its geothermal and geological information.

In the following section 2 data and their preparation are presented. The implementation of the dashboard will be discussed in depth in section 3, some use cases examples will be presented in section 4. In section 5 there will be a comparison with other designs.

2. DATA PREPARATION

As mentioned above the source of data is the Italian Geothermal Database. The actual source of data are three files derived from the dataset and provided by Eugenio Trumpy, researcher at CNR Pisa. From these files only necessary information is extracted, reorganized and integrated in a new *.json* file, named *Boreholes.json*. In the following subparagraphs the semantic and syntactic analysis of the attributes, the integration and cleaning performed over the data will be illustrated.

2.1. SEMANTICS AND SYNTAX

The total number of records in *Boreholes.json* is 3806, each one referring to a well. Its attributes can be divided into four types according to their function:

- *Identity*: which include well identifiers
- *Thermal*, which refer to the temperature measurements details
- *Geological*, regarding underground composition sampling details
- *Geographical*, related to coordinates of well site

The semantics and syntax of each attribute will be discussed groupwise on the basis of this semantic division.

Each record is composed by the following categories of dimensions:

Identity (all attributes in this group are type string variables):

- *key*: unique identifier
- *name*: name

Geographical:

- *lat_wgs84*: latitude coordinate (number)
- *lon_wgs84*: longitude coordinate (number)
- *altitude*: altitude in meters (number)
- *depth*: depth in meters (number)
- *region*: region (string)
- *province*: province (string)

Geothermal:

- *temperatures*: list of temperature measurements (array of numbers)
- *depths*: list of depths of temperature measurements (array of numbers)
- *date*: list of dates of measurements (array of strings)
- *minTemp*: lowest temperature measurement calculated on *temperatures* (number)
- *maxTemp*: highest temperature measurement calculated on *temperatures* (number)
- *meanTemp*: temperature measurements mean calculated on *temperatures* (number)

Geological:

- *compositions*: list of underground layers compositions (array of strings)
- *startDepths*: list of starting depths in meters of the underground layers (array of numbers)
- *endDepths*: list of ending depths in meters of the underground layers (array of numbers)

2.2. DATA CLEANING AND TRANSFORMATION

Thermal, geological, geographical and descriptive data referring to wells were initially stored in separate files. Data were integrated into a unique file by a join over the identifier attribute *key*. All records without an identifier value were removed. The removal involved also those records which had no geographical coordinates values, since they couldn’t be represented on the map. The decimal separator of all numerical attributes was the comma and has been changed to the point, in order to be used as a number.

The computed geothermal attributes are already included in the file to speed up the fetching operations.

3. IMPLEMENTATION

The framework used for the development of the web application is Vue.js, run in Node.js runtime environment. The used packages are:

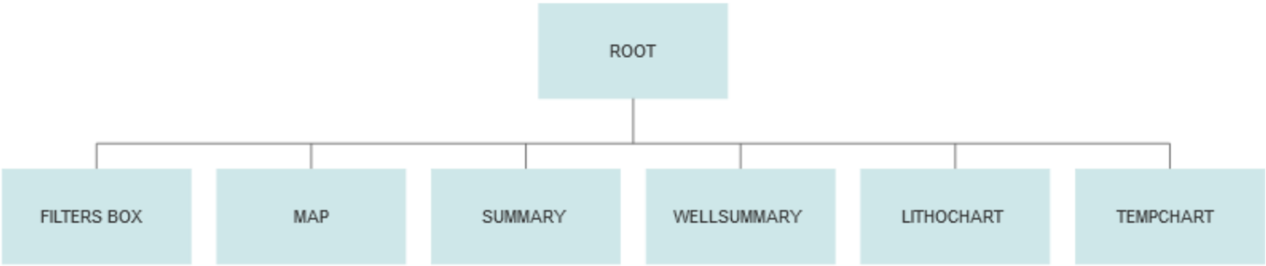
- BootstrapVue for the organization of the components layout
- VuePlotly: for map and charts
- Plotly.js: for managing interactivity on the map
- D3.js for scaling and statistical functions

The dashboard, showed in the next image, is composed with various interactive widgets: a *map* showing boreholes geographical locations, geothermal representations and descriptions; a *filter box*, which allows to filter boreholes appearing on the map by user input parameters; a *summary*, reporting metrics calculated on the boreholes; a *well summary* with metrics calculated on the selected well; a *bar plot* of the selected borehole reporting its underground layers, their thickness and geological compositions; a *scatter plot* of temperature and depth of temperature measurements of the selected well.

In the next subparagraph the implementation of each component is illustrated. In the last subparagraph layout and style choices are described.

3.1. COMPONENTS

Components are arranged in a two-levels hierarchy, as shown in the tree diagram below.



ROOT

It's the parent component and is where all components are linked. What's to be mentioned about this component are its methods.

It has the following methods:

- *apply*: applies filter values to the set of current wells;
- *reset*: resets to default value all components;
- *refreshCurrentWells*: refreshes the set of current wells and calculates new min, max and mean and count of measurements of each well over the measurements which have depth and temperature in the selected ranges in the filter box;
- *refreshMap*: refreshes the Map, by loading a new set of wells on it. Accordingly to the metric selected, it sends *minTemperature*, *maxTemperature* or *meanTemperature* of wells as input temperature values into the map;
- *colorByMin*, *colorByMax*, *colorByMean*: set the metric values to use for displaying data points on the map;
- *refreshSummary*: calculates aggregate values over the current set of wells and refreshes the summary component with these new values;
- *refreshWellSummary*: refreshes the well summary data, by calculating metric values over both those measurements satisfying the parameters set by filters and all measurements taken in the well;
- *refreshCharts*: refreshes the two charts whenever the input index of a newly selected well is sent by Map component.

FILTERS BOX

Filter box is composed with three inputs fields, accepting user defined parameters, and three buttons to manage the application of filters.

The input fields are:

- *Temperature*: it accepts two inputs of maximum three digits integers, a min and a max value; if applied, it returns only those wells which have temperature measurements in the selected range of values. Default values are 0 and 999
- *Depth*: it accepts two inputs of maximum four digits integers, a min and a max value; if applied, it returns only those wells having temperature measurements taken in the selected range of depth values. Default values are 0 and 9999
- *Thickness*: checkbox, if checked and applied, returning only those wells which have a *Thickness* plot. Default value: “Not required”, when checked: “Required”.

Four watchers on input values (*fmintemp*, *fmaxtemp*, *fmindepth*, *fmaxdepth*) are set to monitor the insertion of correct values in the input fields.

The three buttons triggers the homonymous methods when pressed:

- *Apply*: emits an event to the parent component, bringing the new filters input values, which will be used to refresh the current sets of displayed wells;
- *Cancel*: clears the currently inserted values (it does not remove current filters on the wells);
- *Reset*: triggers the emission of an event to the parent component, which resets the visualization to start conditions (where no filters are applied and all wells are displayed).

MAP

The map is the “satellite” type which is available among Mapbox tile maps. This one was chosen <u>because</u> to its clean look and its photo-realistic representation of the world.

Boreholes satisfying the input parameters are represented as circles on the map.

It is possible to zoom in and see the exact location of a borehole. Also by looking at the map at low zoom levels the representation looks like a heatmap because of wells density, giving a general view of the temperatures over the displayed set of wells.

Placing the pointer on a well opens a description window reporting its geographical coordinates, *lat_wgs84* and *lat_wgs84*, *name*, *temperature* metric value, *altitude*, *depth*, *region* and *province*.

When the set of applied filters changes, map points are refreshed and re-rendered with the new set of wells. A watcher on *wells* catches the changes on the array of points to display and refreshes the map with the new points.

When mounted the actions triggered by clicking on a point in the map are set. Clicking on a point triggers the restyling of the points. It also provokes the emission of an event together with the index value of the clicked point going to the parent component. There an event listener catches the emitted event and the index value of the clicked well and use it to refresh the charts and well summary.

Circle colors, opacities and sizes are scaled according to a temperature metric among min, max and mean. The metric value of each well is calculated on those measurements within the set intervals. It can be chosen by clicking on its homonymous button appearing right below the map.

Three scaling functions are used to define circles color, opacity and size. All are calculated on the same temperature domain, set to [0, 500]. The color scale range is set to *YlOrRd*, ranging from yellow for low values to red for high values. Opacity and size ranges are set to [0.5, 1] and to [6px, 12px]. Wells are also sorted in increasing order of the chosen metric values, so that the last points to be rendered on the visualizations are the ones with the highest values. These settings make wells with higher values more visible so that the user can to easily identify in each area of the map those wells where the highest temperatures are found. When a circle is selected, its size is increased to 24px and its opacity to 1 in order to make it always distinguishable from other circles.

SUMMARIES

The two summaries report metrics and measures calculated over both the whole group of displayed wells and the selected one.

Summary is composed with 5 measures and metrics:

- *Boreholes*: the number of currently displayed wells
- *Min*: the minimum temperature recorded among all measurements in the selected range of parameters
- *Max*: the maximum temperature recorded among all measurements in the selected range of parameters
- *Mean*: the mean temperature calculated over all measurements in the selected range of parameters
- *Count*: the total number of temperature measurements in the selected range of parameters

Similarly, the well summary shows:

- *Name*: the name of the borehole
- *Min*: the minimum temperature recorded over the set of measurements of the borehole in the selected range of parameters
- *Max*: the maximum temperature recorded over the set of measurements of the borehole in the selected range of parameters
- *Mean*: the mean temperature calculated over the set of measurements of the borehole in the selected range of parameters
- *Count*: the total number of temperature measurements of the borehole in the selected range of parameters
- all values followed by * refer to the measure calculated over all measurements of the borehole, including those one which are not within the set of parameters.

The summaries refreshes whenever the currently displayed group of wells change. Clearly, the well summary refreshes whenever a new well is selected in the map. Both components have a watcher on *info* which sets the new values displayed in the cards of the summary.

SCATTER PLOT (TEMPCHART)

The chart shows temperature measurements related to the selected well in the map. The x-axis and y-axis show *temperature* and *depth* of measurements. Each point is colored according to the method used for measuring temperature. Methods are ordinally mapped to a custom palette of colors. Colors are chosen in such a way to be very different one to another.

The legend is toggleable: clicking on a method name hides measurements labeled with that method in the chart. Placing the pointer over a point opens a description label reporting its *temperature, depth* and *date* values. Whenever the set of current wells change or another well is selected the chart refreshes.

A Watcher on *well* looks at changes on the input data. The component methods are:

- *resetScatter*: set the scatter points to default empty state.
- *refreshScatter*: refreshes the scatter with the new input data.
- *createScatterTrace*: creates a *Scatter* trace of measurements grouped by method

BAR PLOT (LITHOCHART)

The plot represents the thicknesses of the underground layers. Bar names are the depth ranges of the layers they refer to. Bars are ordinally placed on the axis from left to right based on the increasing depth values of the ranges. Placing the pointer over a bar opens a label reporting the *thickness* and geological *composition*. The chart refreshes in the same way the Scatter Plot does.

A Watcher on *well* looks at changes on the input data. The component methods are:

- *refreshBar*: refreshes the bars with the new input data;
- *resetBar*: set the bars to default empty state.

3.2. LAYOUT AND STYLE

On top of the application a fixed navbar reporting the name of the board is placed.

The components are placed in a two rows grid: the first one with *filter box¸ map* and the two *summaries* on each a column; the second one with the two *charts*.

This layout is intended to be functional and semantically designed. The map is the core of the dashboard and therefore is centrally placed; on the right side all components giving information on temperature measurements are close one to another. Filters box is placed on the left side in order to not break the continuity of the informative part of the visualization.

The color palette chosen for styling the dashboard is based on variations of gray: mildly dark (#495057) for the application background, dark (#343a40) for the components and the navbar and light for text (titles: #ced4da, values: #f8f9fa). The dark look makes the visualization less eye-stressing than a light one and more suitable to longer time usage. The contrast is modulated to be higher between the text and the rest of the application increasing text readability.

The font character used is the Avenir, Helvetica, Arial. Sans Serif. Font sizes used are 14px (values and titles) and 16px (Filters Box title, Boreholes number in Summary and Borehole name in well summary).

4. USE CASES EXAMPLES

The dashboard is designed to be open to different searches. A list of possible uses of the dashboard may be:

- Getting an image of the distribution of temperatures at flexible levels of depth over the country
- Checking the geographic positions of a wells
- Finding areas having a target temperature range
- Identifying areas where there could be new potential sources of heat such as wells where no measurements are taken or isolated groups of wells or even single ones with promising temperatures which could be nearby high heat areas.
- Visually evaluating density of wells in a specific area
- Investigating temperature measurements and underground composition assessments conducted on a well of interest

5. ALTERNATIVE DESIGN SOLUTIONS

In this part other approaches are presented for representing some of the information uncovered by the visualization.

The dashboard is designed to provide various insights on wells altogether. By just clicking on a borehole in the map, the user can get geological, geographical and geothermal information all at once. This is achieved by sacrificing the space occupied by elements such the map in the layout. Other dashboards use larger maps and menus where the user can choose the type of information that will be displayed when clicking on a well in the map. These designs follow a more detailed and specific way of exploring than the less detailed but more immediate and comprehensive one followed by the presented dashboard.

Visualizing the distribution of temperature at different levels on a geographic map can be done by means of the application of a heatmap or, for more sophisticated approximations, a contour layer of temperature measurements on top of the maps. The dashboard map instead tries to work in a dual way, both as a representation of the temperature distribution, acting similarly to a heatmap, and as boreholes navigator.

Stratigraphic underground composition and *thickness* of layers can be represented with a stacked bar. The height of each rectangle is proportional to the thickness of the layer it refers to and layers are ordinally stacked based on their depth from ground level. The dashboard bar plot represents this same information by privileging the comparison of layers thicknesses while losing the visual reference to depth a stacked bar has. The order of layers from ground level is showed by placing bars by increasing depth from left to right instead of being piled up.