Data Visualisation

Student Number: 1260497

A project report for Data Visualisation mini project

Department of Computer Science
University of Oxford

1 Overview

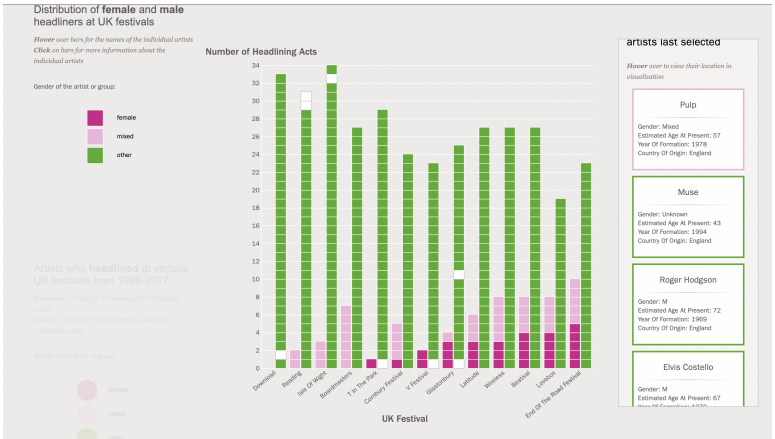

Figure 1: Summary screenshot of the visualisation

The goal of my visualisation is to explore female artists that performed at Music Festivals in the United Kingdom in the late 20th and early 21st Century. There are a large number of music festivals in the UK, each with a high number of artists performing. The more specific area that this data visualisation will explore is the Headliners at the 13 of the largest Music festivals from 1997-2017. There is a known disparity in the number of female artists chosen to headline so in order to understand who the female headliners are it will also be important to consider the context of where they are performing, and how the gender of the headliners vary across different music festivals.

The visualisation is designed for individuals who regularly attend music festivals in the UK and are familiar with the headlining artists and festival names. The user is not expected to have a background in complex visualisation techniques, so it's important that the visualisation is intuitive and visually engaging to ensure the audience remains interested throughout the views.

2 Data

2.1 Dataset URL

https://github.com/BBC-Data-Unit/music-festivals/blob/master/festival_headliners.csv

2.2 Dataset Type

The dataset type is a **multidimensional table**. The keys required to look up an item in the table are **stage_name**, **year** and **festival**.

An item in this data set represents a festival headliner, defined as an artist or group performing at a specified festival on a specified year. This means that we are making the assumption that:

1. An artist has a unique stage name.
2. A festival has a unique name.
3. An artist can only perform as a headliner once in each year at each festival

As the designer, with prior knowledge about the semantics of a "festival headliner", these are valid assumptions to make.

The combination of stage_name, festival and year as keys are always unique for each item thus satisfying the requirements for a multidimensional table key.

2.3 Data preprocessing pipeline

I used Jupyter notebook and Python to preprocess my data, using the following set of steps.

1. I removed the 'second location', 'second city', and 'year inducted' columns from my dataset due to the majority of cells in these columns being empty and their lack of relevance for my visualization purpose.

2. I replaced the blank spaces of the column names with underscores and transformed all letters to lowercase.

3. The current age of the artist was from 2017, I updated the age of the artist to reflect the current year so that the field still has the same real-world meaning.

4. For the purpose of the visualisation, we will treat the data set and divided into 'f' and 'mixed' and all others are 'other' gender, so that the gender attribute has no undefined values.

2.4 Data type

Attribute	Semantics	Semantic restrictions	Data Type	Range/Cardinality
stage_name	Stage name of artist	Two artists cannot have the same stage name	Categorical	291
festival	Name of festival	A festival is uniquely defined by it's name, hence all festivals with the same name represent the same festival	Categorical	13
year	Year that an artist headlined at a given festival	-	Ordered, Sequential	38 (ranging from 1970-2007)
gender	Gender of the artist or group	If the gender is mixed, this represents that the group has member of multiple genders	Categorical	4 ('m', 'f', 'mixed', 'm ' and nan)
birthplace	Place of birth of the artist or group	In the case of a group, the birthplace is defined as where at least one of the members of the group was born	Categorical	105
solo/band	A headliner can be described as either a single artist, or a group of artists	-	Categorical	2 ('s' or 'b') representing solo or band
ethnicity	Ethnicity of the Headliner, for a group this represents the ethnicity of at least one member	-	Categorical	18
age_present	The calculated current age at present of the artist in 2023	-	Ordinal, Sequential	52, ranging from 28 years - 80 years
formation	The year of the formation of the group, or start of the career of the solo artist	-	Ordinal, Sequential	58, ranging from 1956 - 2014
city	The city where the festival the artist is headlining at is based	There can be more than one festival in each city	Categorical	12

3 Goals and Tasks

I will first introduce the overall goal of my visualisation. I will then break down this goal into abstract tasks that address more specific questions about the BBC Festival data set. The five abstract tasks are inspired by the "Overview First, zoom and filter, details on demand" mantra proposed by Ben Shneiderman.

3.1 First task

In order to address the goal of understanding the target female headliners, it is important to allow the visualisation user to see the context of gender in the festivals that appear in the data set. In the language of the domain situation, we are looking at comparing the gender and the number of headliners across each festival. In abstract language, the action, target pair for this task is {Compare, Correlation}.

The compare action fits this task since we are addressing multiple targets; the festivals, the gender, the identities individual artists. These correspond to the 'stage_name' , 'festival' and 'gender' attributes. The target of our task has a scope of multiple attribute. The attributes our task is targeting are gender and festival and the derived frequency of the items (headliners).

3.2 Second task

The second task is to provide an overview of the gender of the individual headliners. the abstraction for this task is {Summarize, Distribution}.

We are looking at the distribution of a single attribute; the gender of the headliners. The action of summarize fits in this case because the scope is all possible headliners. The domain level task is to provide a comprehensive view of all the items in the data set.

3.3 Third task

The third task is to allow the user of the visualisation to find the identities of the female headliners. The abstraction for this task is {Lookup, Outliers}.

Outliers are data items that do not fit with the rest of the data. From prior analysis of our data set the female and mixed headliners are outliers on the gender attribute of the data set. The lookup action refers to a known target in an unknown location, the location of the female headliners should be known from the previous summarize task, hence the users will already know what they are looking for and where it is located in the data set.

3.4 Fourth task

The data should be described with artist names and how many festivals they have played at. This information should be available across my visualisation so that the meaning of different data points is clear. The fourth task in abstract language is {Annotate, Features}.

Annotate is the appropriate task because I want this information to appear temporarily. The domain level meaning of the features target is the stage_name attribute and the number of festivals the artists have played at.

3.5 Fifth task

The users of the visualisation should be able to discover in more detail who the female artists are and refer back to these discoveries across the visualisation. The abstraction for this task is {Record, Features}.

The artists selected by the user should become persistent elements of my visualisation so the record

action is appropriate. This will allow the user to view connections between different aspects of the visualisation and the selected artist. The target of features is chosen because particular attributes will be visualised.

4 Visualisation

4.1 Colour Scheme

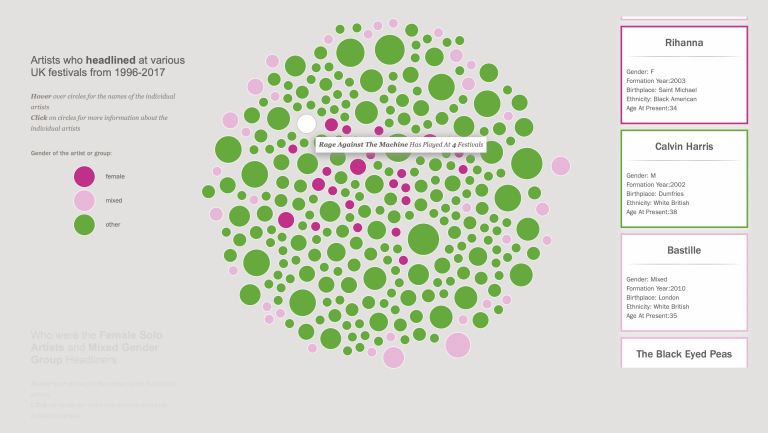

Figure 2: A screenshot demonstrating the range of colours used in my visualisation

A consistent colour palette for the attribute of gender is chosen to reduce to cognitive load on the user of the visualisation and provide intuition behind the item links between the views.

Throughout my visualisation the derived binary attribute meaning 'contains a female artist in the headliner act' and 'does not contain a female artist in the headliner act' is encoded using the hue channel. Hue is the second most effective channel after spatial region but the spatial region channels is already constrained due to other the other attribute encodings. The two hues chosen are '#d01c8b' (pink), '#4dac26' (green). These have been chosen from the ColorBrewer website to be colour blind safe. Pink and green are appropriate choices for a binary variable because they fall on opposite sides of the colour wheel.

There are two values the attribute can take within the 'female headliner' group ; 'mixed' or 'f'. There is a natural ordered implied with these values, the female 'f' headliners can be thought of semantically as 'more female'. The luminance channel is used to encode this ordering. The 'mixed' attribute value has the colour '#f1b6da', which has a luminance of 86%. The 'f' attribute value has the colour '#d01c8b', which has a lower luminance of 46%.

4.1.1 Style choices

The formatting and style has been inspired by 'The Sunlight Foundation's Data Visualization Style Guidelines' to ensure maximum accessibility and accuracy for the user. There are some minor changes made from the style guidelines with regards to formatting due to the specific layout of the visualisation. The most noticeable is the white highlighting, rather than outline highlighting. This is done to provide the 'pop out' effect in the histogram view when multiple rectangles are highlighted. The background has a higher luminance and lower saturation than the other colours in the visualisation which provides contrast.

4.2 Scrolling

Scrolling is a familiar and intuitive way of navigating through the visualisation, allowing users to rewind effortlessly to previous screens. Connections can be made between visualisations without the cognitive overload of multiple views on one screen. Scrolling also gives the user an impression of "exploring" the data set and is arguably the most engaging interactivity technique.

The order of the visualisation is as follows; first context is provided for the female headliners in the data set, then all headliners summarized and finally the female headliners are located. Scrolling allows the order of these steps to be conveyed concisely. With scrolling, the user can also revisit or reverse

previous steps with a minimal time delay and smooth transitions. A stepper would have provided the same functionality, but would have resulted in decreased usability.

There are some pitfalls with the scrollytelling idiom, which remove the intuition of scrolling. The first is 'continuous scrolling with discrete steps' and 'scroll-jacking'. These have been resolved by having a scrolling title and legend on the left hand side of the visualisation so that users are provided with visual feedback to scrolling. Another issue is an unknown length of scrolling. This is not a significant problem in this visualisation due to the short length of three views. Nevertheless, explanatory text has been added on the title screen of the visualisation.

4.3 Compare correlation

4.3.1 Visual encoding decisions

A histogram shows the number of female, mixed and other headliners for each festival. Mixed and female headliners are shown on a stacked bar for each festival. The male headliners are shown as a separate bar grouped by festival with the female and mixed headliner. The bars are placed in ascending order of percentage of number of female and mixed gender headliners. The marks are point marks, stacked on top of one another and given fixed areas to create the histogram.

4.3.2 Rational for visual design choices

A histogram is chosen to maintain the relationship of data type between the visualisation and the categorical attributes. Having each mark representing a headliner, maintains the connection between data points and their semantics. This is an appropriate choice since the data set is of medium size.

The first attribute this view is displaying is the frequency of the headliners for each festival. This is a derived attribute, calculated by summing the individual headliners for each festival, grouping them by festival and gender. The most effective channel for a magnitude attribute is position on a common scale, histograms use vertical position to show the frequency.

The second attribute this view is displaying is the festival of the headliner which is shown by the grouped bars. The abstract task is to compare correlation, and in the domain language this means the trends in gender between the festivals. Hence the most important attribute to show is the festival. Spatial region is the most effective channel for categorical variables; so bars are grouped by festival in an aligned spatial region following the effectiveness principle. The second most important attribute is the gender. Two channels are used to display this attribute, hue and spatial region. The female and not female derived attribute is displayed using the spatial region, since it is higher priority than the 'mixed' and 'f' attribute values within the female value.

The task is to compare correlations in gender distribution of across the festivals, so it is important to encode both the gender and festival attributes in such a way users can quickly discriminate between the two attributes with minimal interference. The grouped and stacked histogram ordered by percentage difference achieves.

4.3.3 Interactivity

There are two user interactions added to this view:

The first interaction is the initial scroll from the first page. On the scroll down, the bars grow upwards from their x position on the histogram. The bars with the 'f' and 'mixed' gender attribute appear first. The user can replay the transition using the scrolling interactivity.

The second interaction is that on the hover of the marks, all other marks on the bar chart with the same stage_name attribute value are highlighted.

4.3.4 Rational for Interactivity choices

The first interaction prevents 'change blindness' between the two comparisons the histogram is making. Because the bars representing the female headliners appears first, the user can clearly compare the difference in the number of female headliners, then see how they compare within the festival to the male headliners. The choice the bars appearing to 'grow' from the x position, draws the user's eyes to the differences in vertical position of the top of the histogram bars. The user does not need to rely on their working memory to recall the visualisation without the male bars as the transition can be replayed via scrolling.

The motivation behind the second interaction is to visualize the semantic link between stage_name attribute and the items in the data set. Immediate visual feedback is provided, showing links or lack of links between different bars in the histogram. Artists in the same festival are located spatially close, and require less time to traverse between. There is less cognitive load than encoding this directly into the static visualisation. The interaction addresses the task as it allows the correlation between the same headliners in the different festivals to be compared by the user.

4.4 Summarize Distribution

4.4.1 Visual encoding decisions

A point mark is used for each artist_name attribute. It is effective to do this because of the range of the stage_name attribute, 291 unique values. The area channel is used to encode the number of times the artist has been a headliner, directly proportional to the area of the point. The colour channel continues to be used for the gender attribute. The tooltip, which is provided as an encoding for the separate annotate task, also addresses this task.

4.4.2 Rational for design choices

The spatial channel is not used to encode an attribute and instead the points are grouped together into the same spatial region to convey that this is an overview of the data. This arrangement emphasises the 'pop out' effect of the colour channel.

The area channel is an effective channel for attributes with a magnitude but presents downsides with regards to its effectiveness for quantitative accuracy. In the context of my visualisation, this is a compromise that is acceptable to make in order to allow for expressiveness of the circles representing artists. To mitigate the effects of this a tool tip has been added with the number of festivals headlined at shown in text.

4.4.3 Interactivity

The circles move on the initial scroll into view. On scroll back up from the locate view, the size of the new stage_name marks grown from zero. Both of these help the user track the changes.

4.5 Locate Features

4.5.1 Visual encoding decisions

The located circle marks move closer together spatially on scroll and the filtered marks disappear. This can be interrupted and reversed by the user by scrolling.

4.5.2 Rational for design choices

The change in spatial position of the circles conveys the filter action to the viewer. The interruption of the scroll animation allows the user to connect the goal of summarizing to the locate goal, and semantically place the female headliners within the entire set of artists. The scrolling provides a quick and intuitive way of returning to the summary view. The choice to filter to address the location action rather that deriving a new quantity makes clear to the user what has been done. A dramatic change

in overall colour and size of the visualisation prevents change blindness between the summary and the filtered view.

4.6 Record Artists

4.6.1 Visual encoding decisions

Throughout the scrolling central views, clicking on a mark causes more information about the artist with stage_name attribute tied to that mark to appear on the right-hand side of the view. These boxes will be juxtaposed with the main view. The marks that have already been clicked will be recorded in the order that they are clicked, the most recently clicked mark will be at the top. The last five marks clicked on will appear in this view.

4.6.2 Independent interactivity

On hovering over the information box for the artist, the marks in the main visualisation that have the same stage_name attribute value are highlighted.

4.6.3 Rational for design choices

Once the {Record, Artist} task has been completed by the user, the view will be juxtaposed as the user backtracks and explores their previous steps. The 'eyes beat Memory' rule of thumb for visualisation state that it is more effective to have the information visible than rely on the users memory. There is not enough space to display information about all the artists at once, hence a textual history of size five is provided. This is a trade off between providing a record of selected artists and users having intuition with regards to clicking new artists without highlighting. The record task is satisfied since the annotations provide a clear timeline of which artists have been explored and clicked. This idiom provides this visual timeline for the user. Highlighting across views places the items in the three views of the main scrolling visualisation, creating links between all three views. Highlighting the selected artists on hover rather than a more permanent change in colour reduces the interference with the main hue channel in the main views of the visualisation.

4.7 Annotate Distribution

Items grouped by stage name can be annotated with their stage name, in the summary and locate views the marks are also annotated with the number of festivals that they have headline in. Linked highlighting between marks shows this the number of festivals artists have headlined at in the histogram view.

5 Credits

- Jim Vallandingham scrolling

  The 'scroller.js' file is **unchanged** from Vallandingham's project.

  Major functionality addition: I have used the structure of the 'sections.js' file but have changed all visual functionality to match my visualisation and abstracted the functionality out into separate classes to match the mini project specification which was not done in Vallandingham's code. I have also used aspects from the 'index.html' file to enable the scrolling functionality.

- Grouped bar plot **Major functionality addition**: I used this code block as a template for the grouped bar part of my code

- Rotating axis ticks **minor tweaks**: This informed how I would rotate my axis labels in the histogram.js file

- Force on scrolling This code was not directly used but provided reference on issues with scrolling and how to deal with d3 forces to provide smooth transitions.

6 Walkthrough of my visualisation

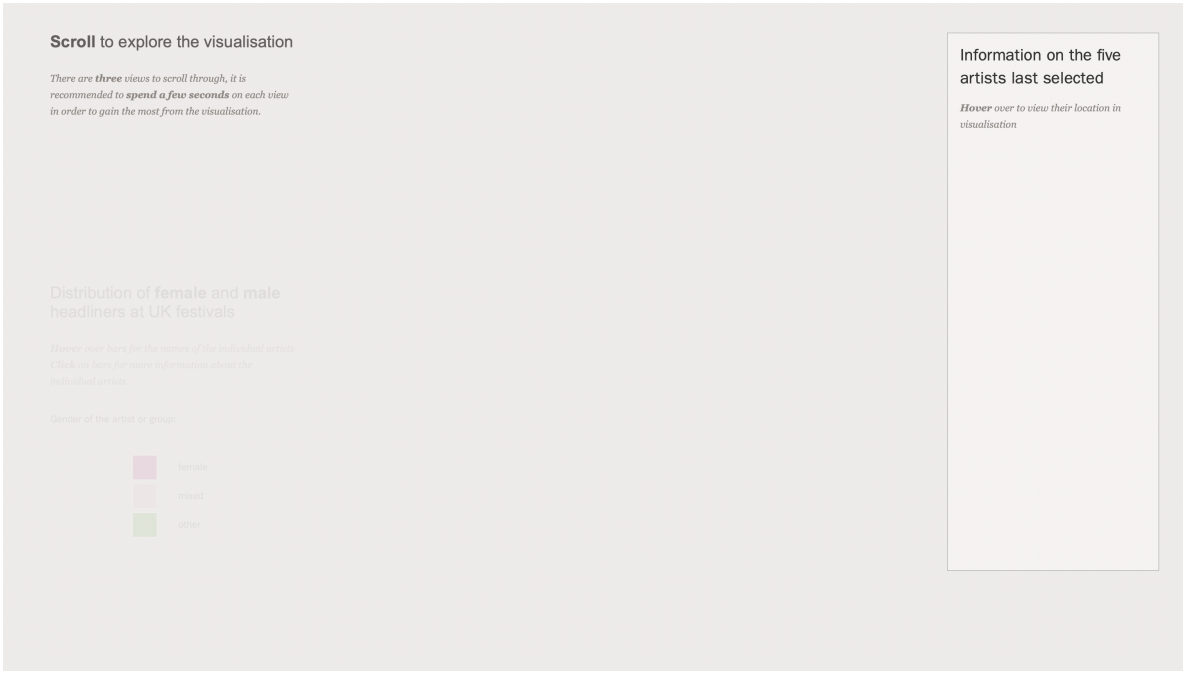

Figure 3: Title view

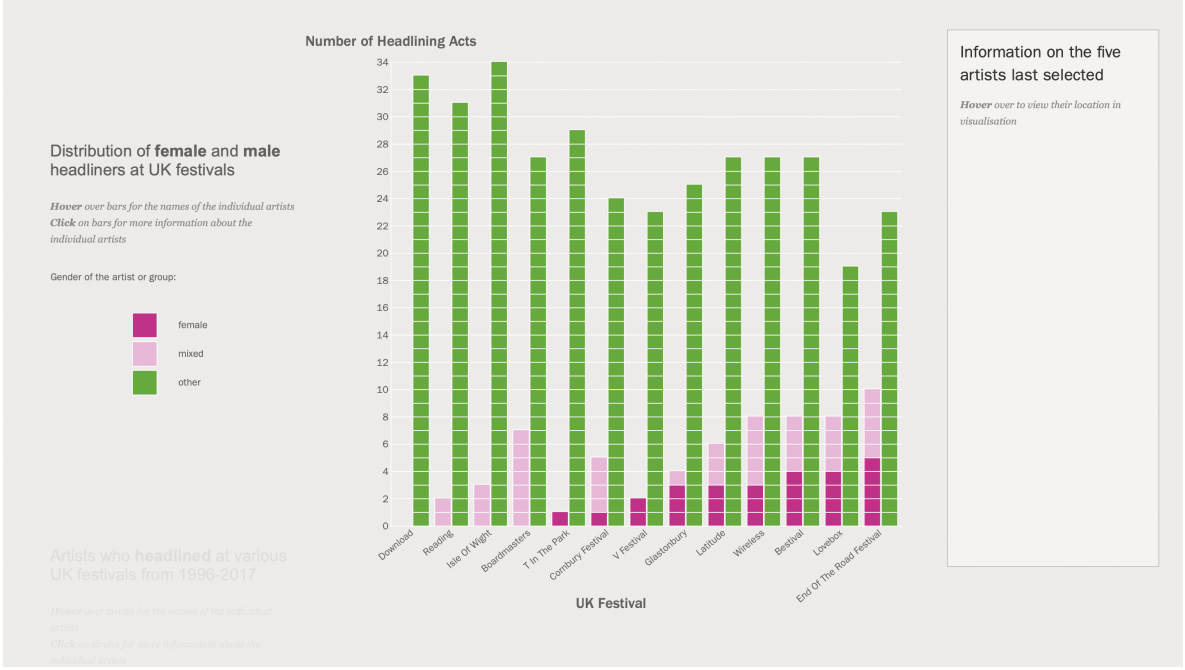

Figure 4: **Scroll down**: Histogram view

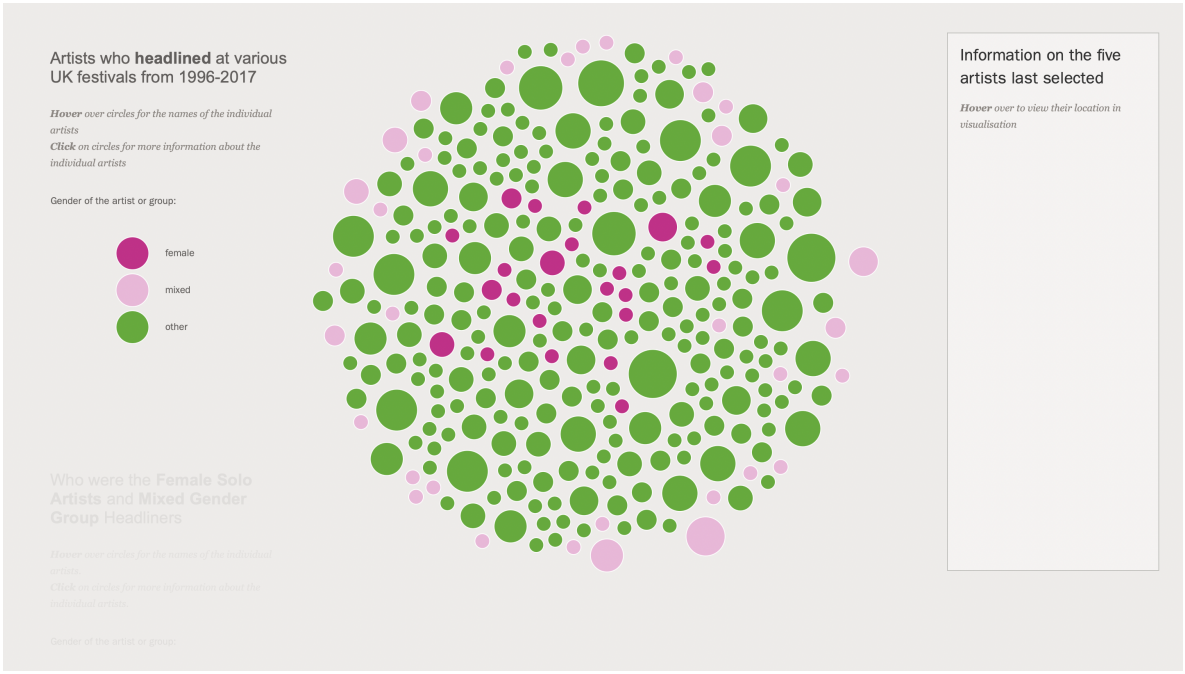

Figure 5: **Scroll down**: Summary View

Figure 6: **Scroll down**: Filter View

Figure 7: Highlighted on hover

Figure 8: More information on click

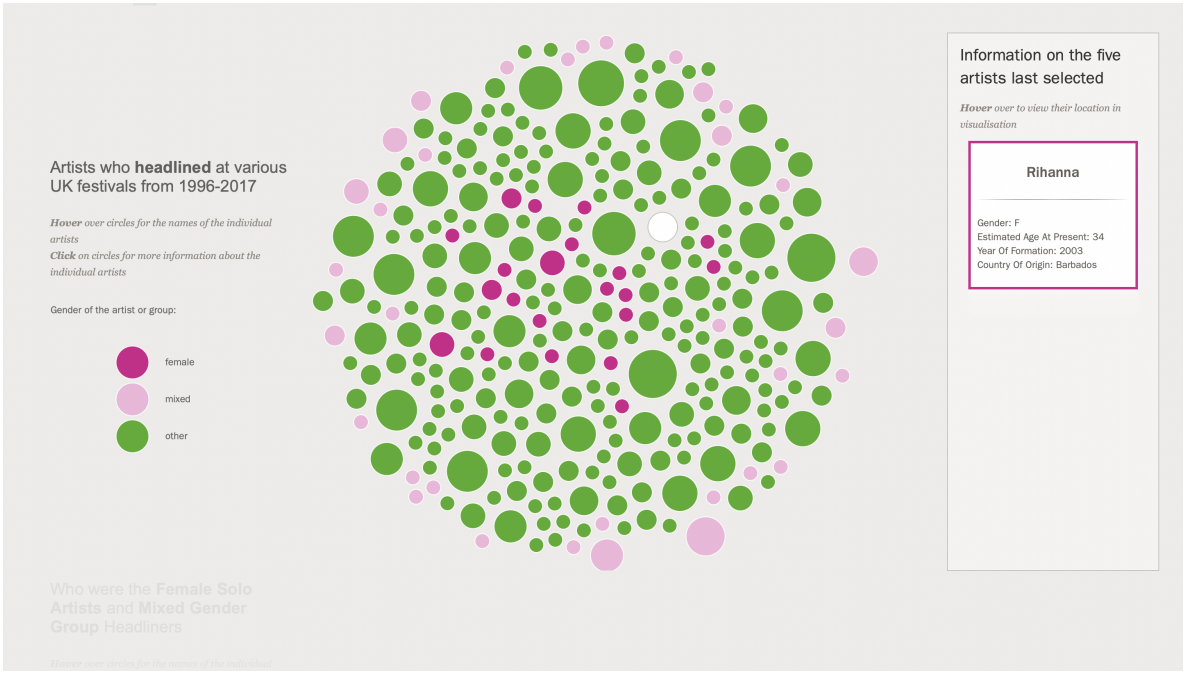

Figure 9: **Scroll back up**: Highlighting via artist information across views

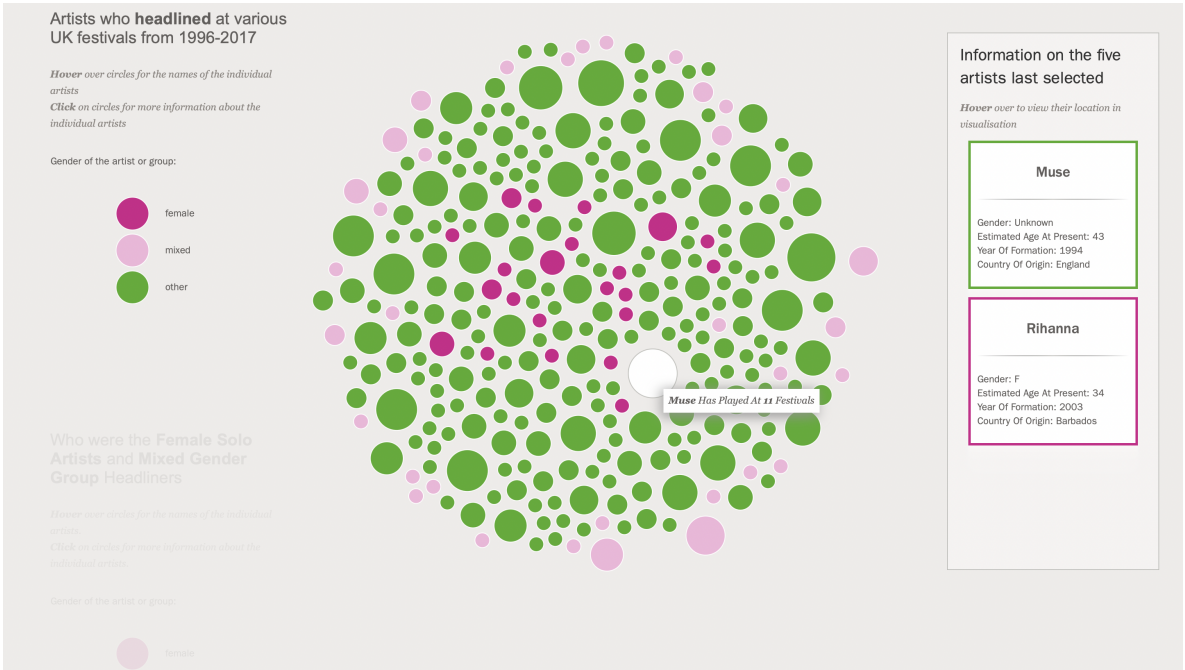

Figure 10: Annotating on hover

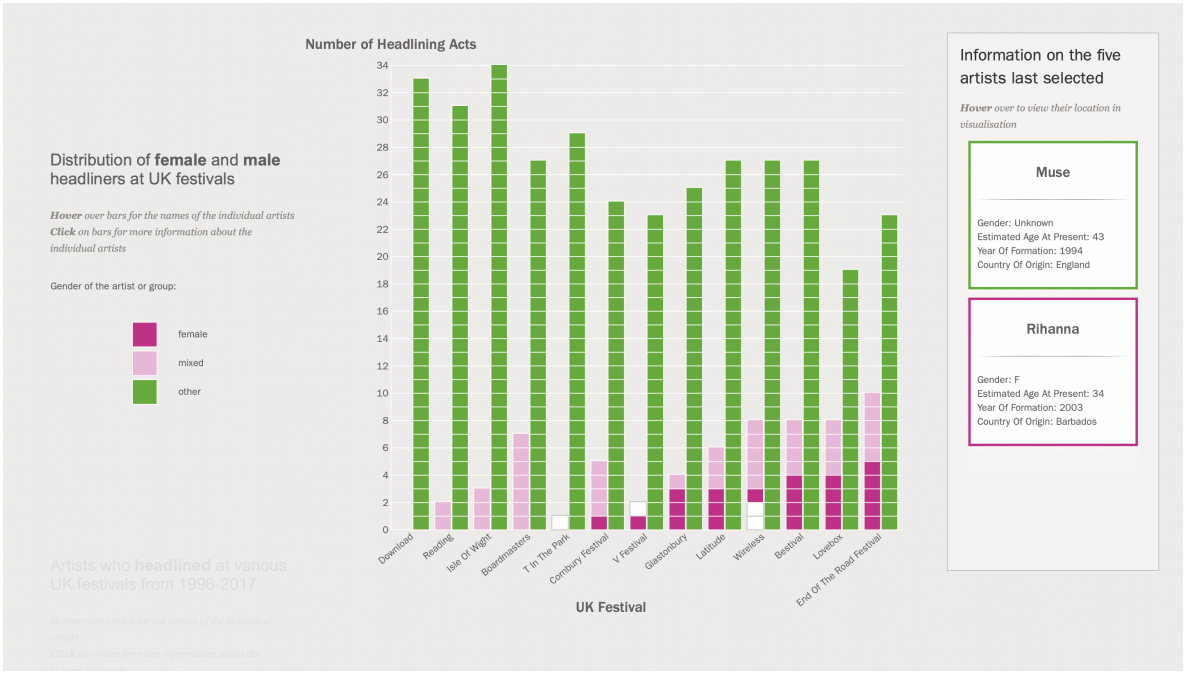

Figure 11: **Scroll back up**, multiple rectangles highlighted from hover on Rihanna artist information

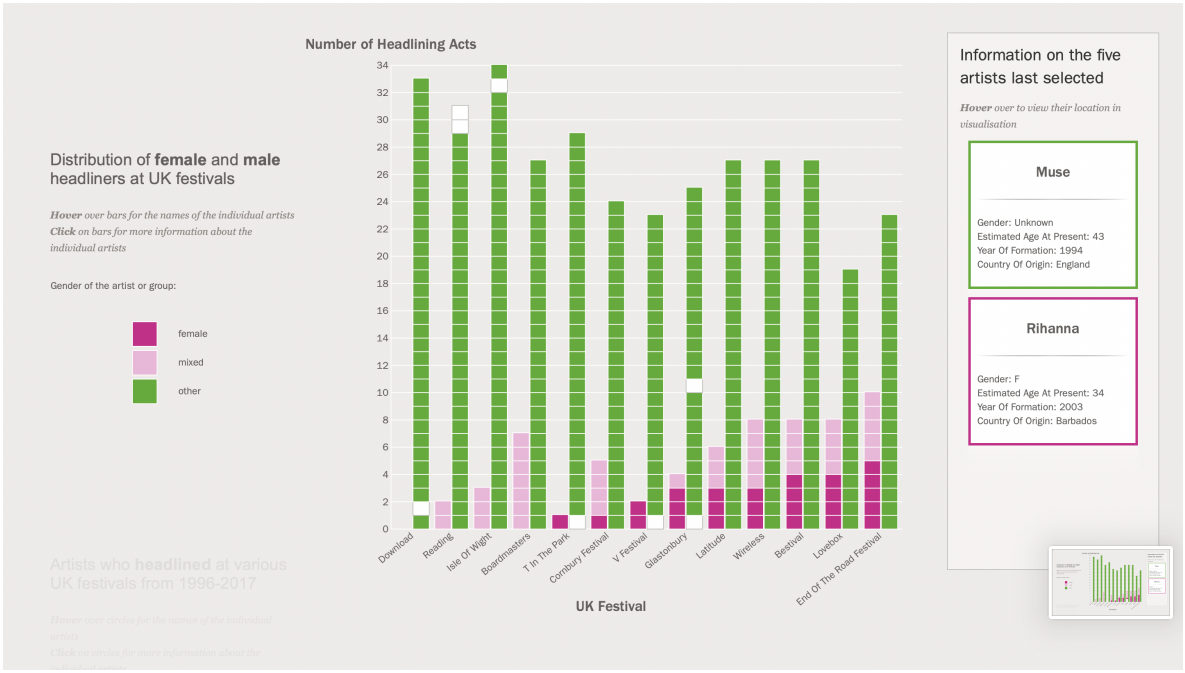

Figure 12: **Scroll back up**, multiple rectangles highlighted from hover on main chart