**Video Script: Section 4 Video 4 big data with bigvis: binning and condensing**

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Description | Action on screen | Narration |
| 1 | Introduction  (Outcome and why it is desirable)  This should give the viewer an idea of the outcome of the task at the beginning of the videos and set the stage and expectations of the viewer. | Refer to PPT | In this video, we are going to install and use BigVis, an R package specially designed for large datasets. |
| 2 | Context(Problem/Solution)  Present the viewer with a real-world solution and how the situation would pose as a challenge. It always helps to draw the viewer's attention using a use-case. Metadata template can be used here. |  | If you have a very large dataset (e.g. hundreds of thousands of rows to millions), it makes little sense to try and plot all the data on the screen: there’s not even enough pixels! |
| 3 | Guidance (How to do it and how it works): |  | We are going to use BigVis to pre-process and summarise the data before plotting it. |
| 4 |  |  | We first need to install the package. It is not on CRAN but on github but it is easy to install it in a couple of lines. |
| 5 |  | Highlight and run in the command line:  install.packages("devtools")  devtools::install\_github("bigvis") | Open activity\_04\_04.R  Run the first 2 commands |
| 6 |  | In the editor:  install.packages("devtools")  devtools::install\_github("bigvis")  to  # install.packages("devtools")  # devtools::install\_github("bigvis") | Now that the package is installed, you can comment out these two lines; you won’t need them any more. |
| 7 |  | Highlight and run:  # loading the libraries  library(ggplot2)  library(bigvis)  # what is the data about:  ?bigvis::movies  On the screen: a description of the data. | Next we’ll load the necessary libraries (ggplot2 and bigvis) and have a look at the data we’ll be using. |
| 8 |  |  | The dataset “myMovies” contains information about films and has more than 130’000 rows. Let’s plot the distribution of movie length vs ratings. |
| 9 |  | Highlight and run:  myMovies <- subset(movies,length<300) # manually get rid of outliers.  myMoviesSummarised <- with(myMovies, condense(bin(length), bin(rating))) | Run the next 2 lines.  BigVis creates a heatmap (a two-dimensional distribution) and typically works like this:  First you put your data in bins,just as when you plot a histogram for example.  The bin function will automatically find a good bin width for your data.  You can also specify the binwidth with the parameter width.  Next you condense (or summarise) each bin.  This creates a ‘condensed’ object from a list of binned data, which only contains the location of each tile in the heatmap and its value, e.g. a count. |
| 10 |  |  | myMoviesSummarised can then be plotted quickly, since it only contains the pre-processed data necessary for the plotting. |
| 11 |  | Highlight and run:  ggplot( data = myMoviesSummarised ) +  geom\_tile(aes( x = length, y = rating, fill = .count)) +  ggtitle("Distribution of 130'000 movies along length and ratings")  A description... | Run the ggplot command. |
| 12 |  |  | Ggplot2 was able to plot the large dataset very quickly. |
| 13 |  |  |  |
| 14 |  |  |  |
| 15 |  |  |  |
| 16 | Conclusion:The video concludes by showing the viewer that the goal has been achieved, and reminding them why they should be happy about that. A PowerPoint summary slide with the key points emphasized would make it easier for the viewer to remember what was covered in the video | Back to PPT | We’ve seen how to quickly plot a very large amount of data thanks to the package BigVis, which produces a summary of the data.  BigVis is partially written in C++, which makes it process the data very fast.  In the next video, we'll see how to smooth the plot and remove outliers. |