**Video Script: Section 5 Video 2 – ordering factors**

|  |  |  |  |
| --- | --- | --- | --- |
| 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’ll see how to control the order in which categorical values are plotted. |
| 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. |  | By default, categorical values are ordered by their level if they are already a factor, or in alphabetical order if no level has been defined.  This is not always desirable and there is a way to change this. |
| 3 | Guidance (How to do it and how it works): | Switch to activity\_05\_02.R in RStudio | Open activity\_05\_02.R in RStudio |
| 4 |  | Highlight and run:  commute <- data.frame(  categories = c("other","car","public transport","cycle/walk","work from home"),  proportion = c(1,64,16,14,5)  )  head(commute) | We first build a small dataframe to plot. It shows the proportion of different modes of transportation to commute to work in the UK. |
| 5 |  | highliggplot(commute) + geom\_bar(aes(x=categories,y=proportion),stat="identity") + ggtitle('default order') | We first make a bar plot directly with this data: |
| 6 |  | A description... | The categories appears in alphabetical order: car, cycle/walk, other, public transport, work from home  Not the order in which the dataset was defined. |
| 7 |  |  | Since ggplot2 uses the factor’s levels to order the categories, we need to specify the levels to control the order.  Changing the levels implies changing the data, so as a precaution, we’ll make the changes on a copy. |
| 8 |  | Highlight and run  commute$userDefined <- factor(commute$categories,c("car","public transport","cycle/walk","work from home","other"))  ggplot(commute) + geom\_bar(aes(x=userDefined,y=proportion),stat="identity") + ggtitle('user-defined order') | In this situation, we want the categories in alphabetical order, and with ‘other’ on the right handside.  We define a new column ‘userDefined’ where we specify the levels explicitly with the function factor(). |
| 9 |  | A description... | Using ‘userDefined’ instead of ‘categories’ in the aesthetics, the bars are now ordered exactly like we want. |
| 10 |  |  | You can also defined the order more programmatically, for example to order the categories in the order they appear in the dataset: |
| 11 |  | Highlight and run:  commute$originalOrder <- factor(commute$categories,as.character(commute$categories))  ggplot(commute) + geom\_bar(aes(x=originalOrder,y=proportion),stat="identity") + ggtitle('original order') | In this case, the levels are defined as the values appear in the vector commute$categories, hence respecting the original order. |
| 12 |  | A description... |  |
| 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 | In this video, we’ve seen how to control the order of categorical variables in a plot.  In the next video, we will see how to change the default colour palette when plotting categorical variables. |