Comparing the Value of Data Visualization Methods for Communicating Harms in Clinical Trials

Riaz Qureshi, Xiwei Chen, Carsten Goerg, Evan Mayo-Wilson, Stephanie Dickinson, Lilian Golzarri-Arroyo, Hwanhee Hong, Rachel Phillips, Victoria Cornelius, Mara McAdams DeMarco, Eliseo Guallar, Tianjing Li

Table of Contents:

Web Appendix 1
Web Appendix 2
Web Appendix 3
Web Appendix 4
Web Appendix 5
Web Appendix 6
Web Appendix 1. Barchart Visualization

Barchart Visualization: Higher-order presentation of Adverse Events in clinical trials

This script can be used to generate a Bar Chart visualization for adverse events data from a clinical trial.

The Bar Chart is a replication of that found in Chuang-Stein & Xia 2013:
Chuang-Stein C. & Xia H.A. (2013). The Practice of Pre-Marketing Safety Assessment in Drug Development. Journal of Biopharmaceutical Statistics, 23(1): 3-25. DOI: 10.1080/10543406.2013.736805

Description and generation

A bar chart is a standard bar graph that depicts the relative occurrence of harms experienced across higher classifications of adverse events (e.g., mid-level and body-level systems). Specifically, the bars present the counts of participants experiencing an adverse event to provide comparison of observed occurrence in a study and the bars can be broken down into different colors to present the occurrence by severity classification (i.e., mild, moderate, severe). This figure is produced using R procedures for bar charts and the following algorithm:

For the bar chart: A. Take all adverse events occurring over the trial and classify according to a higher-order term of choice (either mid-level or body-level system). B. Code all adverse events by severity (as determined by trial investigators) C. Generate a bar chart, by treatment group, of the counts for all harms grouped according to their higher-order term and severity.

Data file organization

The data should be set up in a long format as in the following example, with a variable for treatment, body system, severity, and the number of times of occurrence:

| Treatment | BodySystem_1 | Severity | Count |
|-----------|--------------|----------|-------|
| INT       | Mild         | x        |
| INT       | Moderate     | x        |
| INT       | Severe       | x        |
| INT       | Unclassified | x        |
| Ctrl      | Mild         | x        |
| Ctrl      | Moderate     | x        |
| Ctrl      | Severe       | x        |
| Ctrl      | Unclassified | x        |
| .          | .            | .        |
| .          | .            | .        |
| .          | .            | .        |
| INT       | Mild         | x        |
| INT       | Moderate     | x        |
| INT       | Severe       | x        |
| INT       | Unclassified | x        |
| Ctrl      | Mild         | x        |
| Ctrl      | Moderate     | x        |
| Ctrl      | Severe       | x        |
| Ctrl      | Unclassified | x        |

Source of example data

Data for this example comes from a trial of gabapentin vs. placebo (Trial 945-210) which has been revisited to analyze and publish the complete harms as part of a grant from the Restoring Invisible and Abandoned Trials (RIAT) initiative – Principal Investigator: Dr. Evan Mayo-Wilson.

Preparation of data
Visualization code

The following code generates the Bar Chart visualization. An example visualization (Trial 945-210) is included.

```r
library(ggplot2)
ggplot(Events, aes(fill=Severity, y=Count, x=Bodysystem)) +
  geom_bar(position=position_stack(reverse=TRUE), stat="identity") +
  scale_y_continuous(name="Unique count", limits=c(0, 100)) + # If any body systems appear more than 100 times, change the upper limit to an appropriate number
  ggtitle("Trial: 945-210") + # Change the title overall to match the name of the trial
  xlab("COSTART Midlevel Bodysystem") +
  scale_fill_viridis_d() +
  facet_grid(Treatment ~ .) +
  theme(axis.text.x = element_text(angle=90, hjust=1, size=8),
        axis.title.x = element_text(color = "black", size = 14, face = "bold"),
        panel.background = element_blank(),
        panel.grid.major.x = element_blank(),
        panel.grid.major.y = element_line(color = "grey", size = 1))
```

Suggested alterations to Bar Chart Visualization

The following suggestions are slight modifications to the original presentation that may be considered to improve the clarity of the bar chart visualization.

- Changing “Y-axis” title to “Frequency (count)” to be explicit that the y-axis is the number of times each
- Place bars side by side instead of in separate panels to potentially improve comparability
- Include numbers above each bar to indicate exactly how many times each occurred
- Remove body systems which only have one or two events across both arms as these are unlikely to be of interest and body systems with no events in either group are already excluded from the figure
Web Appendix 2. Dot Plot Visualization

Dot Plot Visualization: Presentation of unique preferred terms for adverse events in clinical trials

This script can be used to generate a Dot Plot visualization for adverse events data from a clinical trial.

The Volcano Plot is a replication of that found in Amit 2008:

Amit O., Heiberger R.M, & Lane P.W. (2008). Graphical approaches to the analysis of safety data from clinical trials. Pharmaceutical Statistics, 7: 20-35. DOI:10.1002/pst.254 (DOI:10.1002/pst.254)

Description and generation

A dot plot is a two-panel display of the AEs most frequently occurring in the gabapentin group of the study. The first panel will present the incidence of each AE by treatment group, and the second panel will present the relative risk of that specific event for the gabapentin group relative to the placebo arm with 95% confidence intervals. Depending on the context of the trial, the right panel could alternatively present the hazard ratios or risk differences. The primary purpose of this visualization is to highlight potential signals by providing an estimate of the treatment effect and its precision.

Data file organization

The data should be set up in a wide format as in the following example, with a variable for outcome, risk in control and intervention groups, the estimate for the measure of effect (e.g., risk ratio), and the lower and upper bounds of the 95% Confidence Interval for the estimate of effect:

| Outcome | Risk (control) | Risk (intervention) | Effect estimate | 95% CI (Lower) | 95% CI (Upper) |
|---------|----------------|---------------------|-----------------|---------------|---------------|
| PreferredTerm1 | x | x | x | x | x |
| PreferredTerm2 | x | x | x | x | x |
| PreferredTerm3 | x | x | x | x | x |
| PreferredTermn | x | x | x | x | x |

Source of example data

Data for this example comes from a trial of gabapentin vs. placebo (Trial 945-210) which has been revisited to analyze and publish the complete harms as part of a grant from the Restoring Invisible and Abandoned Trials (RIAT) initiative – Principal Investigator: Dr. Evan Mayo-Wilson.

Preparation of data

```r
# Set Working Directory to be Whichever folder contains the data
setwd("D:/Dropbox/Data/RIAT Visualizations")

# Import the comma separate value file
PreferredTerms <- read.csv("D:/Dropbox/Data/RIAT Visualizations/PreferredTerm_945-210.csv", sep="", fileEncoding = 'UTF-8-BOM')
RRSortedPT <- PreferredTerms[order(-PreferredTerms$RiskRatio ),]

# Create subset of full data with the elements that we need for each half of the plot
PTRisk <- subset(RRSortedPT, select=c(outcome,RiskPlacebo,RiskGabapentin,RiskRatio))
PTRisk$outcome <- factor(PTRisk$outcome, levels = PTRisk$outcome[order(-PTRisk$RiskRatio)])
PTRiskRatio <- subset(RRSortedPT, select=c(outcome,RiskRatio,RRLCL,RRUCL))
PTRiskRatio$outcome <- factor(PTRiskRatio$outcome, levels = PTRiskRatio$outcome[order(-PTRiskRatio$RiskRatio)])
```

Visualization code

Part 1

The first section of code generates the first half (left side) of the visualization: the group-specific risks.
# install.packages("reshape")
# install.packages("egg")

library(reshape)
ByGroup <- melt(PTRisk, id=c("outcome"))
ByGroup <- ByGroup[ByGroup$variable != "RiskRatio",]

library(ggplot2)
library(viridis)

## Loading required package: viridisLite

library(scales)

## Attaching package: 'scales'

## The following object is masked from 'package:viridis':
##    viridis_pal

left <- ggplot(ByGroup, aes(x=value, y=outcome, fill=variable)) +
   geom_dotplot(binaxis='y', stackdir='center', dotsize = 0.5) +
   ylab("MedDRA Preferred Term") +
   scale_fill_viridis_d() +
   scale_x_continuous(name="Percent", breaks = c(0,0.1,0.2,0.3), labels = c("0"="0", "0.1"="10", "0.2"="20", "0.3"="30"), limits = c(0,0.3)) +
   scale_y_discrete(limits=rev(levels(ByGroup$outcome))) +
   theme(legend.position="bottom",
         panel.background = element_blank(),
         panel.border = element_rect(color="black", fill=NA, size=1),
         panel.grid.major.y = element_line(color="black", size=0.1, linetype=3),
         axis.ticks.x = element_line(size=1, colour="black"))

Part 2

The next segment of code generates the second half (right side) of the visualization: the effect estimate and corresponding confidence interval.

right <- ggplot(PTRiskRatio, aes(x=outcome, y=RiskRatio, ymin=RRLCL, ymax=RRUCL)) +
   geom_pointrange(fatten=2) +
   geom_hline(yintercept = 1, linetype = 2, colour = "green", size = 0.75) +
   scale_y_continuous(name = "Relative Risk with 95\% CI",
                      trans = log2_trans(),
                      breaks = c(0.01,0.25,1,10,20,50,100,220)) +
   scale_x_discrete(limits = rev(levels(PTRiskRatio$outcome))) +
   coord_flip() +
   theme(legend.position="bottom",
         axis.title.y = element_blank(),
         axis.text.y = element_blank(),
         panel.background = element_blank(),
         panel.border = element_rect(color="black", fill=NA, size=1),
         panel.grid.major.y = element_line(color="black", size=0.1, linetype=3),
         axis.ticks.y = element_line(size=1, colour="black"),
         axis.ticks.x = element_line(size=1, colour="black"),
         axis.text.x = element_text(angle=90, hjust=1, size=10))

Part 3

The third section of code puts the two figures together into a single figure. An example visualization (Trial 945-210) is included.

# install.packages(ggpubr)

library(ggpubr)
DotPlot <- ggarrange(left, right, ncol=2)

## `stat_bindot()` using `bins = 30`. Pick better value with `binwidth`.

```
Suggested alterations to Dot Plot Visualization

The following suggestions are slight modifications to the original presentation that may be considered to improve the clarity of the dot plot visualization.

- Add additional panels for other effect estimates (e.g., a third panel to show the estimate and 95% CI for the risk difference)
- Selection criteria applied to the harms to reduce the number on the plot and improve legibility
Web Appendix 3. Volcano Plot Visualization

Volcano Plot Visualization: Presentation of unique preferred terms for adverse events in clinical trials

This script can be used to generate a Volcano Plot visualization for adverse events data from a clinical trial.

The Volcano Plot is a replication of that found in Zink 2013:

Zink R.C., Wolfling R., & Mann G. (2013). Summarizing the incidence of adverse events using volcano plots and time intervals. Clinical Trials, 10: 398-406. DOI:10.1177/1740774513485311 (DOI:10.1177/1740774513485311)

Description and generation

A volcano plot summarizes several characteristics that are important for understanding the relative harms associated with an intervention including: the proportional frequency of each adverse event (bubble radius) as well as the total number of adverse events experienced in both treatment groups (total bubble area), the specific treatment with greater association (colour and side of the figure), statistical significance of the association (colour saturation and position on vertical axis), and magnitude of effect (position on horizontal axis). As the treatment and statistical significance both have their own elements in the figure (i.e., side and vertical position), depending on desired context and message, the colour of bubbles could instead be used to indicate some other dimension of AE data such system organ class or event severity. Although an R-package does exist for volcano plots, we produce this figure is using the procedures for bubble plots and the following algorithm:

For the Volcano Plot: A. Take a selection of adverse events to be visualized B. Compute Fisher’s exact test p-value, total number of events, and corresponding treatment effects for events of interest C. Compute -log10(raw p-value) for the y-axis and treatment effect (risk difference, odds ratio, or risk ratio). D. Size bubbles according to total events. E. Bubbles can be colored according to the magnitude of treatment effect, blue to one side, red to the other, with grey in the middle to help de-emphasize. F. Add reference lines for -log10(0.05) for unadjusted and -log10(alpha*) for adjusted.

Data file organization

The data should be set up in a wide format as in the following example, with a variable for outcome, frequency in control and intervention groups, the estimate for the measure of effect (e.g., risk difference), and the p-value for the estimate of effect (adjusted or unadjusted):

| Outcome       | Frequency (control) | Frequency (intervention) | Effect estimate | P-value |
|---------------|---------------------|--------------------------|-----------------|---------|
| PreferredTerm1| x                   | x                        | x               | x       |
| PreferredTerm2| x                   | x                        | x               | x       |
| PreferredTerm3| x                   | x                        | x               | x       |
| PreferredTermn| x                   | x                        | x               | x       |

Source of example data

Data for this example comes from a trial of gabapentin vs. placebo (Trial 945-210) which has been revisited to analyze and publish the complete harms as part of a grant from the Restoring Invisible and Abandoned Trials (RIAT) initiative – Principal Investigator: Dr. Evan Mayo-Wilson.

Preparation of data
Set Working Directory to be Whichever folder contains the data

setwd("D:/Dropbox/Data/RIAT Visualizations")

# Import the comma separate value file

PreferredTerms <- read.csv("D:/Dropbox/Data/RIAT Visualizations/PreferredTerm_945-210.csv", sep="", fileEncoding = 'UTF-8-BOM')

PT_Data <- subset(PreferredTerms, select=c(outcome, trial_no, freq_Placebo, freq_Gabapentin, RiskDifference, RDpval))

# Some adjustments to the data to aid the visualization

PT_Data$freq_Total <- PT_Data$freq_Placebo + PT_Data$freq_Gabapentin # Total number of events to scale the size of the bubbles

PT_Data$harmful <- ifelse(PT_Data$RiskDifference < 0, 0, 1) # Create an indicator for associated with intervention (harmful = 1) or control (harmful = 0)

PT_Data$RDpval <- replace(PT_Data$RDpval, PT_Data$RDpval<=0.001, 0.001) # The purpose of this is to eliminate infinite values when taking the log of the p-value. If using adjusted p-values, change the "<=0.001" to reflect an appropriate value.

PT_Data$logPval <- -log10(PT_Data$RDpval)

Visualization code

The following code generates the Volcano Plot visualization. An example visualization (Trial 945-210) is included.

library(ggplot2)
library(dplyr)

## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##     filter, lag

## The following objects are masked from 'package:base':
##     intersect, setdiff, setequal, union

ggplot(PT_Data, aes(x=RiskDifference, y=logPval, size=freq_Total, color=as.factor(harmful), label=outcome)) +
  geom_point(alpha=(PT_Data$logPval)) +
  ggtitle("Trial: 945-210") +
  scale_size(range = c(.1, 24)) +
  scale_color_manual(breaks = c("0", "1"), values=c("deepskyblue2", "red")) +
  geom_text(size = 3, color = "black", check_overlap = TRUE) +
  geom_hline(yintercept = 1.3, linetype = 2, colour = "red") +
  scale_y_continuous(name = "-log10(Raw p-value [unadjusted])", limits = c(0,3)) + # Max of 3 equals a p-value of 0.001. If using an adjusted p-value, the max may need to be adjusted further to accommodate lower p-values
  scale_x_continuous(name = "Risk Difference for Gabapentin vs. Placebo", breaks = c(-0.1, -0.05, 0, 0.05, 0.1, 0.15, 0.2, 0.25), limits = c(-0.1,0.25)) + # change the limits to that the largest absolute effect size is contained
  theme(panel.background = element_blank(),
        legend.position = "none",
        axis.line.x = element_line(size = 1, colour = "black"),
        axis.line.y = element_line(size = 1, colour = "black"))
Suggested alterations to Volcano Plot Visualization

The following suggestions are slight modifications to the original presentation that may be considered to improve the clarity of the volcano plot visualization.

- Add label to reference line that indicates it is $p = 0.05$
- Add labels to all events that have the same underlying data and are currently hidden (e.g., ‘ggrepel’): this can get messy with too many labels
- Vertical line at Risk Difference = 0
- Label for left and right of x axis (i.e., more common in placebo or gabapentin)
- Colour can represent a different dimension of information (e.g., body systems)
- Selection criteria applied to the harms to reduce the number on the plot and improve legibility
Web Appendix 4. Heatmap and Treemap Visualization

Heatmap & Treemap Visualization: Presentation of unique preferred terms for adverse events in clinical trials

This script can be used to generate a Heatmap or a Treemap visualization for adverse events data from a clinical trial.

The Heatmap is a replication of that found in Zink 2018:

Zink R.C. et al. (2018). Sources of safety data and statistical strategies for design and analysis: Clinical Trials. Therapeutic Innovation and Regulatory Science, 52(2): 141-158. DOI:10.1177/2168479017738980 (DOI:10.1177/2168479017738980)

The Treemap is not a direct replication, but is a subtype of Heatmap that is used to present hierarchical data and has been used to present data on harms by organizations such as the Food and Drug Administration.

Description and generation

A Heatmap presents data about the expected standardized effect for harms overall and across several selected subsets/subgroups of harms. The subgroups can be any that are available, but they should be distinct such that each ‘column’ in the heatmap shows a different set of harms that may be of interest to evidence users. A standardized effect is important heatmaps as the risks can be variable because of differences in reporting but the degree of uncertainty is difficult to show in a two-dimensional field, thus the standardization aims to temper the lack of presentation of precision surrounding estimates. The method of organization that is selected for the harms in a heatmap is also important as it will change which inferences can be more easily made by looking at the figure. For example, when harms are arranged by to higher-order body systems, readers can see which specific events and body systems are most likely affected by the intervention. We will create our heatmap using a standardized risk difference – represented by colour – and arrange our preferred terms by mid-level classifications, organized from most to least events in gabapentin, to replicate the source material. Although an R-package does exist for heatmaps, we produce this figure using the procedures for tile plots and the following algorithm:

For the Heatmap: A. For all (or a selection of) preferred terms for harms occurring in the trial and compute the standardized difference for all events “overall” with the following formula: \((p_t - p_c) / (\sqrt{(p_t(1-p_t)/n_t) + (p_c(1-p_c)/n_c)})\) B. Select subgroups of interest/availability and keep only the events meeting those criteria, then calculate the standardized difference for events in each of these subgroups. Subgroup examples: i. Sex – Female, Male ii. Seriousness – Serious AEs, Non-serious AEs iii. Severity – Moderate and severe AEs, Severe AEs iv. Recurrence – Single episode AEs, Multiple episode AEs v. Relatedness to intervention – Likely related, Possibly related, Definitely related C. Merge the files for all subgroups into one single data file D. Plot the specific harms on the Y-axis with the different subgroups on the X-axis.

A Treemap is a subtype of heatmap that presents data about the expected standardized effect for harms at the level of the preferred term in boxes, organized by their mid-level classification. As with the heatmap, colour represents the standardized effect, however unlike the heatmap, the absolute count for events occurring in the intervention arm is also represented by the size of the box for each event. The size of the preferred term boxes also affects the size of the corresponding mid-level classifications, indicating which body systems are more common among gabapentin.

For the Treemap: A. For all (or a selection of) preferred terms for harms occurring in the trial and compute the standardized difference for all events “overall” with the following formula: \((p_t - p_c) / (\sqrt{(p_t(1-p_t)/n_t) + (p_c(1-p_c)/n_c)})\) B. Select subgroups of interest/availability and keep only the events meeting those criteria, then calculate the standardized difference for events in each of these subgroups. Subgroup examples: i. Sex – Female, Male ii. Seriousness – Serious AEs, Non-serious AEs iii. Severity – Moderate and severe AEs, Severe AEs iv. Recurrence – Single episode AEs, Multiple episode AEs v. Relatedness to intervention – Likely related, Possibly related, Definitely related C. Merge the files for all subgroups into one single data file D. Plot the specific harms grouped according to their corresponding midlevel terms using the Treemap function. E. Size the boxes according to number of events in the intervention arm.

Data file organization

The data should be set up in a long format as in the following example, with a variable for outcome, associated higher-order term, frequency in control and intervention groups, number of participants in control and intervention groups, risk of the outcome in control and intervention groups, and the specific subtype for which the data is being presented. The setup below is for the trial “overall” participants, however the Heatmap presents data on harms from different subgroups as well. From the trial Individual Participant Data, select only participants meeting each different subgroup that are to be visualized (independently) and recreate the same structure and estimates for those participants. Once all subgroup datasets have been created, they will be joined into a single data file for the Heatmap. The Treemap will use only data from the “Overall” subgroup.

| Outcome      | Mid_level | Freq_placebo | Freq_intervention | N_placebo | N_intervention | Risk_placebo | Risk_intervention | Subgroup |
|--------------|-----------|--------------|-------------------|-----------|----------------|--------------|-------------------|----------|
| PreferredTerm1 | MidLevel1 x | x | x | x | x | x | Overall |
| PreferredTerm2 | MidLevel2 x | x | x | x | x | x | Overall |
| PreferredTerm3 | MidLevel3 x | x | x | x | x | x | Overall |
| PreferredTerm4 | MidLevel4 x | x | x | x | x | x | Overall |
| PreferredTerm5 | MidLevel5 x | x | x | x | x | x | Overall |
| PreferredTerm6 | MidLevel6 x | x | x | x | x | x | Overall |
| PreferredTerm7 | MidLevel7 x | x | x | x | x | x | Overall |
| PreferredTermm | MidLeveln x | x | x | x | x | x | Overall |
Source of example data

Data for this example comes from a trial of gabapentin vs. placebo (Trial 945-210) which has been revisited to analyze and publish the complete harms as part of a grant from the Restoring Invisible and Abandoned Trials (RIAT) initiative – Principal Investigator: Dr. Evan Mayo-Wilson.

Preparation of data

```r
# Set Working Directory to be Whichever folder contains the data
setwd("D:/Dropbox/Data/RIAT Visualizations")

# Import the comma separate value file
PreferredTerms <- read.csv("D:/Dropbox/Data/RIAT Visualizations/PreferredTerm_945-210.csv", sep=""," fileEncoding = 'UTF-8-BOM'")

# Create subset of full data with the elements that we need for the plot
Overall <- subset(PreferredTerms, select=c(outcome,mid_level,freq_Placebo,freq_Gabapentin,N_Placebo,N_Gabapentin,RiskPlacebo,RiskGabapentin))

# Add a variable to mark which subgroup of harms is being mapped. As this is created only with a single set of harms (Overall), that is what we put. But additional subgroups of the harms could be created and put into other dataframes (e.g., data1 = moderate or severe harms, data2 = serious harms, data3 = probably or definitely associated with the intervention, data4 = harms that are recurrent, etc.) which are then merged using the Outcome as the joining variable.

Overall$subgroup <- "Overall"

# Pull in any subsets of harms with the data set up the same way and then append them in a new complete dataframe
Female <- read.csv("D:/Dropbox/Data/RIAT Visualizations/Subsets for Heatmap (Participant)/Heat_FemaleParticipant_945-210.csv", sep=""," fileEncoding = 'UTF-8-BOM'")
Male <- read.csv("D:/Dropbox/Data/RIAT Visualizations/Subsets for Heatmap (Participant)/Heat_MaleParticipant_945-210.csv", sep=""," fileEncoding = 'UTF-8-BOM'")
Severity <- read.csv("D:/Dropbox/Data/RIAT Visualizations/Subsets for Heatmap (Participant)/Heat_severityParticipant_945-210.csv", sep=""," fileEncoding = 'UTF-8-BOM'")
Serious <- read.csv("D:/Dropbox/Data/RIAT Visualizations/Subsets for Heatmap (Participant)/Heat_SeriousParticipant_945-210.csv", sep=""," fileEncoding = 'UTF-8-BOM'")
Nonrecurrent <- read.csv("D:/Dropbox/Data/RIAT Visualizations/Subsets for Heatmap (Participant)/Heat_NonrecurrentParticipant_945-210.csv", sep=""," fileEncoding = 'UTF-8-BOM'")
Recurrent <- read.csv("D:/Dropbox/Data/RIAT Visualizations/Subsets for Heatmap (Participant)/Heat_RecurrentParticipant_945-210.csv", sep=""," fileEncoding = 'UTF-8-BOM'")

#Append the subsets all to a single data file
data <- rbind(Overall,Female,Male,Severity,Serious,Nonrecurrent,Recurrent)

# Calculate the standardized difference
data$effect <- (data$RiskGabapentin - data$RiskPlacebo)/(sqrt(((data$RiskGabapentin*(1-data$RiskGabapentin))/data$N_Gabapentin)+((data$RiskPlacebo*(1-data$RiskPlacebo))/data$N_Placebo)))
data$subgroup <- factor(data$subgroup, levels = c("Overall", "Female", "Male", "Moderate or severe", "Serious", "Single episode", "Multiple episodes"))
data$mid_level <- as.factor(data$mid_level)
data$outcome <- factor(data$outcome, levels = unique(data$outcome[order(data$mid_level)]))
```

Visualization code - Heatmap

The following code generates the Heatmap visualization. An example visualization (Trial 945-210) is included.

```r
library(ggplot2)
library(viridis)

## Loading required package: viridisLite
```
Heatmap <- ggplot(data, scale = "column", aes(x=subgroup, y=outcome)) + geom_tile(aes(fill=effect, height = 1)) + scale_fill_viridis_c(name="Standardized Risk Difference (Gabapentin vs. Placebo)") + #geom_text(geom = "text", label = unique(data$mid_level), size = 6) + #geom_text(angle=90, colour="darkgray", aes(label=mid_level), position=position_dodge(width=0.9), col=gray) + #annotate(geom = "text", label = unique(data$mid_level), size = 6) + geom_text(angle=90, colour="darkgray", aes(label=mid_level), position=position_dodge(width=0.9), col=gray) + facet_grid(data$mid_level, drop = TRUE) + ggtitle("Trial: 945-210") + labs(x = "Subsets of harms", y = NULL) + theme(axis.text.x = element_text(angle=45, hjust=1, size=10), axis.title.x = element_text(color = "black", size = 14, face = "bold"), legend.position = "right", panel.background = element_blank(), panel.grid.major.x = element_blank(), panel.grid.major.y = element_blank())

plot(Heatmap)

After the Heatmap is produced, the figure should be exported and then labels for the mid-level terms can be added manually using an image editing program.

Visualization code - Treemap

The following code generates the Treemap visualization. An example visualization (Trial 945-210) is included.
# Suggested alterations to Heatmap and Treemap Visualization

The following suggestions are slight modifications to the original presentation that may be considered to improve the clarity of either visualization.

- Add secondary axes labels for higher-order classifications (manually added, or automatically added)
- Use colour scheme to better distinguish no effect (e.g., divergent with white as no effect)
- Indication that white spaces is the lack of data for a harm within that subgroup (i.e., the harm was not reported in that subgroup)
- Selection criteria applied to the harms to reduce the number on the plot and improve legibility
Web Appendix 5. Tendril Plot Visualization

Tendril Plot Visualization: Presentation of specific adverse events in clinical trials

This script can be used to generate a Tendril Plot visualization for adverse events data from a clinical trial.

The Tendril Plot is a replication of that found in Karpefors & Weatherall 2018:

Karpefors M. & Weatherall J. (2018). The Tendril Plot – a novel visual summary of the incidence, significance and temporal aspects of adverse events in clinical trials. Journal of the American Medical Informatics Association, 25(8): 1069-1073. DOI: 10.1093/jamia/ocy016

An R-package does exist specifically for producing Tendril Plots, created by the authors of the publication, and more information about how to set up the data and create this specific visualization can be found in the following documentation:

https://CRAN.R-project.org/package=Tendril

Description and generation

A tendril plot is a method of visually summarizing the timing, directionality, and magnitude of associations for adverse events after receipt of an intervention. Each “tendril” represents a preferred term for an adverse event with the coloring of each point indicating false-discovery-rate corrected p-values (Pearson’s chi-squared test for the hypothesis that the treatment arms have the same proportion of events up to that event) and the size of each point being proportional to the total number of events for that preferred term.

The path followed by the tendril presents contains the information pertaining to event timing and direction of association (i.e., intervention or comparator). The time since randomization runs along each branch with the magnitude (or length) between points being proportional to the timing between each event. The center of the figure represents the start of the study and all tendrils begin moving directly upwards, with each event shifting the direction of the tendril – clockwise for events in the placebo arm or counter-clockwise for events in the active arm – by some degree. The degree can be configured for optimal presentation and does not need to be the same for both arms, in fact it may be preferable to set the degrees as proportional to the number of participants in each arm to prevent potential bias caused by unbalanced treatment allocation. The Tendril package follows the following algorithm:

For the Tendril Plot: A. Sort the events according to time since randomization. B. Calculate the magnitudes of the vectors as the time between subsequent events. For an event occurring at the same time as the previous event, the magnitude will be zero. C. Calculate the angle of the vectors. For each vector the angle is the cumulative sum of all angles up to that event. The angle is negative (clockwise rotation) for events on the placebo arm and positive (counter-clockwise rotation) for events on the active arm. Zero-magnitude vectors will still contribute to angular changes. Thus, if 3 events, 2 on placebo and one on active, occur at the same time, the net effect is a 1 unit clockwise rotation. D. Add the vectors together cumulatively, i.e., the next vector in time starts at the end of the previous vector in time. The resulting sequence of vectors constitutes the tendril for that AE.

Data file organization

The Tendril Plot uses individual participant data as opposed to the aggregate summary data that is used by other adverse event visualizations. Consequently, the data is set up in a long format with a variable for subject ID, treatment they received, specific harm experienced, and the timing of the event (e.g., days since randomization). The full data set can be imported such that participants who do not experience any harms are still included and this will form the separate, optional dataframe from which the required subset – only participants who experience an event – will be created for the main figure.

| SubjectID | Treatment | SpecificHarm | Days |
|-----------|-----------|--------------|------|
| 1001      | Intervention | PreferredTerm1 | x    |
| 1001      | Intervention | PreferredTerm2 | x    |
| 1002      | Intervention | PreferredTerm2 | x    |
| 1003      | Comparator  | PreferredTerm3 | x    |
| 1003      | Comparator  | PreferredTerm3 | x    |
| 1004      | Intervention | PreferredTerm1 | x    |
| 1005      | Comparator  | PreferredTerm4 | x    |
| 1006      | Comparator  | PreferredTerm2 | x    |
| 1007      | Comparator  | PreferredTerm5 | x    |

Source of example data

Data for this example comes from a trial of gabapentin vs. placebo (Trial 945-210) which has been revisited to analyze and publish the complete harms as part of a grant from the Restoring Invisible and Abandoned Trials (RIAT) initiative – Principal Investigator: Dr. Evan Mayo-Wilson.
# Set Working Directory to be Whichever folder contains the data

```r
sdwd("D:/Dropbox/Data/RIAT Visualizations")
```

# Import the comma separate value file containing the trial data

```r
SubjList <- read.csv("D:/Dropbox/Data/RIAT Visualizations/Tendril_945-210.csv", sep=".", fileEncoding = 'UTF-8-BO M')
```

# Generate subset of data that is only the participants who experienced a harm

```r
t <- subset(SubjList, COSTARTpreferredterm != ")
```

## Visualization code

The following code generates the Tendril Plot visualization. An example visualization (Trial 945-210) is included.

```r
# install.packages("Tendril")
library(Tendril)

## Attaching package: 'Tendril'

## The following object is masked _by_ `.GlobalEnv`:
##
## SubjList

library(ggplot2)

harms <- tendril(mydata = t,
  rotations = 5, # set the degree to which each event pulls a tendril in a direction
  AEFreqThreshold = 3, # Change the number of occurrences required to be plotted
  Tag = "treatment_dic",
  Treatments = c("Placebo", "Gabapentin"),
  Unique.Subject.Identifier = "patient_id",
  Terms = "COSTARTpreferredterm",
  Treat = "treatment_dic",
  StartDay = "aestartday",
  SubjList = SubjList,
  SubjList.subject = "patient_id",
  SubjList.treatment = "treatment_dic",
  suppress_warnings = TRUE)

## Warning: `rename_()` is deprecated as of dplyr 0.7.0.
## Please use `rename()` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_warnings()` to see where this warning was generated.

# Default plot and coloring (each harm)
plot(harms)
Interactive visualization

By specifying "interactive = TRUE" in the call to create the plot with additional options, an interactive plot will be created in which every event can be highlighted to show the specific harm, its timing, and the treatment.
Suggested alterations to Tendril Plot Visualization

The following suggestions are slight modifications to the original presentation that may be considered to improve the clarity of the tendril plot visualization.

- Use interactive option for presentation wherever possible (e.g., website or online supplement)
- If presented as a static image, use labels to identify which harms are represented by each tendril
INTRODUCTION AND PURPOSE

Objective: To compare the characteristics and value of different approaches for visualizing harms using patient-level data from RCTs of gabapentin for neuropathic pain.

Framework: Modified ICE-T (Insight, Confidence, Essence, Time) – ICE-T: a heuristic approach to evaluating the value of multiple visualizations. The ICE-T framework comprises a survey with multiple heuristics that are assessed on a Likert scale and fall under four components: insight, time, essence, and confidence.

What not to assess: We created all visualizations using R and the same options for text font/size, colour, etc. with the purpose of focusing the evaluation of value on the contents and approaches taken by the visualizations. Please do not focus on aspects like the choice of color or symbols used in the visualizations. Similarly, for visualizations that present data on harms at a higher level (i.e., aggregated mid-level terms), do not focus on the classifications that were made.

Order of evaluations and timing: The order in which visualizations will be presented is set as chronologically when the source materials were published. Each visualization will be accompanied by a background paragraph describing its components. Our modified ICE-T has 10 questions which are to be applied to all 6 visualizations, thus the full evaluation should therefore take between 45-75 minutes. The order in which you will assess these visualizations is: (1) Dot Plot, (2) bar Chart, (3) Volcano Plot, (4) Heatmap, (5) Treemap, (6) Tendril Plot

DOMAINS

Insight: Intended to capture how a visualization supports intentional and incidental insights. Intentional insight refers to tasks or questions a person sets out to address, while incidental insight refers to serendipitous discoveries where the user may have stumbled upon an unexpected piece of knowledge. Statements 1-4 are meant to assess whether the visualization facilitates answering questions about the data.

Time: Intended to capture how a visualization facilitates faster, more efficient understanding of data with respect to both searching and browsing of data. Searching refers to a user’s deliberate task to locate particular information within a data set, while browsing refers to a user’s more casual scanning of a data set to find potentially interesting information. Statements 5-6 are meant to assess whether the visualization affords rapid parallel comprehension for efficient browsing.

Essence: Intended to capture how a visualization communicates the essence of the data set with respect to overview and context. Overview refers to a high-level view or summarization of the data set, while context refers to relevant information surrounding the data set. Statements 7-8 are meant to assess whether the visualization provides a big picture perspective of the data and whether it provides an understanding of the data beyond individual data cases (i.e., harms).

Confidence: Intended to capture how a visualization helps a user feel confident in his/her understanding of the data set with respect to the quality of the data and quality of the visualization. Confidence in the quality of the data refers to an understanding of potentially missing or erroneous data, while confidence in the quality of the visualization refers to an understanding of the accuracy of the representation of the data (e.g., does the visualization mislead?). Statements 9-10 are meant to assess whether the visualization helps to avoid making incorrect inferences and whether it helps to understand data quality.
### Modified ICE-T

Statements from the original ICE-T have been revised to fit a biomedical framework, specifically regarding the presentation of harms data collected in clinical trials.

| **Insight** | **Time** | **Essence** | **Confidence** | **How do you rate your agreement with the following statements?** |
|-------------|----------|-------------|----------------|-------------------------------------------------------------|
| The visualization facilitates answering questions about the data | The visualization affords rapid parallel comprehension for efficient browsing | The visualization provides a big picture perspective of the data | The visualization helps avoid making incorrect inferences | 1 2 3 4 5 6 7 |
| This visualization exposes data for specific events (i.e., harms at the preferred term level, e.g., dizziness, nausea).* | The visualization provides a meaningful spatial organization of the data. | The visualization provides an understanding of the data beyond individual data cases | The visualization helps understand data quality | 1 2 3 4 5 6 7 |
| This visualization exposes data for categories of events (e.g., aggregated by system such as “nervous system”).* | I can identify the key characteristics of the data** at a glance. | I can understand how the different dimensions of harms data that are presented in the visualization are related to each other (e.g., increased number of events and increased precision/significance of an estimate). | The elements of the visualization (e.g., color, shape, scale) provide a meaningful visual representation of the underlying data for multiple dimensions of harms. | 1 2 3 4 5 6 7 |
| I can perceive relationships between treatments and harms. | The visualization helps identify unusual or unexpected, yet useful data characteristics or values. | | The visualization would highlight those issues. | 1 2 3 4 5 6 7 |

* Any visualization could theoretically be created to present data at the level of either specific harms (i.e., preferred terms) or grouped harms (i.e., midlevel terms), however these statements should be considered in light of how the original visualization is created and presented, not what is theoretically possible.

** Dimensions of harms data (e.g., count of occurrence, effect by group, effect estimate, precision of estimate, statistical significance of estimate, severity, timing)

---

**Likert scale:** 1 = Strongly Disagree | 2 = Disagree | 3 = Somewhat Disagree | 4 = Neither Agree nor Disagree | 5 = Somewhat Agree | 6 = Agree | 7 = Strongly Agree