Pareto charts

A Pareto chart is a bar chart (with the bars ordered) combined with a line chart that shows the cumulative total or frequency of the bar values.

It is very useful for seeing at a glance what proportion of the data is represented by the first n bars.

It allows us to visualise how ranked quantities contribute to the whole

It is useful for revealing something like the 80-20 rule—e.g. 80% of the accidents are due to 20% of the possible reasons.

The data for the example below came from the Road Safety Authority (RSA) of Ireland (<https://www.rsa.ie/en/RSA/Road-Safety/RSA-Statistics/>) from the report on its web site on Vehicle factors in fatal accidents on Irish roads 2012 to 2018.

The visualisation shows the disproportionate involvement of private cars in fatal road accidents in the period.

There is a little bit of data preparation required for the Pareto chart. We need to represent the cumulative frequency of occurrence of the category values we wish to plot.

# load packages  
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

library(ggplot2)

## Warning: package 'ggplot2' was built under R version 4.0.4

fatal\_acc <- read.csv("RSA\_vehicle\_fatal\_accidents-2012-2018xlsx.csv")  
# str gives the type of the data container (a data.frame), the variables and types  
str(fatal\_acc)

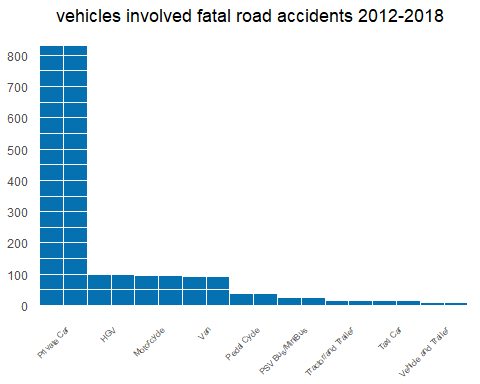
## 'data.frame': 9 obs. of 2 variables:  
## $ ï..vehicle: chr "Private Car" "HGV" "Motorcycle" "Van" ...  
## $ n : int 831 103 96 94 37 27 17 16 9

# If we want to plot this as a pareto chart we need to represent the cumulative frequency of occurrence in ascending order  
  
fatal\_acc\_df\_cumulative <- fatal\_acc %>%  
 mutate(relative\_freq = n/sum(n), cumulative\_freq = cumsum(relative\_freq))  
  
colnames(fatal\_acc\_df\_cumulative) <-c("vehicle","n","relative\_freq","cumulative\_freq")  
head(fatal\_acc\_df\_cumulative)

## vehicle n relative\_freq cumulative\_freq  
## 1 Private Car 831 0.67560976 0.6756098  
## 2 HGV 103 0.08373984 0.7593496  
## 3 Motorcycle 96 0.07804878 0.8373984  
## 4 Van 94 0.07642276 0.9138211  
## 5 Pedal Cycle 37 0.03008130 0.9439024  
## 6 PSV Bus/MiniBus 27 0.02195122 0.9658537

The first plot is a standard bar chart of the data. It’s fine. However, we are after a chart that shows us the breakdown of values per vehicle type - but also shows us the cumulative contribution of the top ranked categories to the whole.

library(scales)  
  
# plot  
p <-   
 ggplot(fatal\_acc\_df\_cumulative, aes(x = reorder(vehicle, -n), y = n)) +  
 geom\_col(width = 1, fill = "#0571b0", color="white") +   
   
 scale\_y\_continuous(breaks=seq(0,1000, by =100)) +   
   
ggtitle("vehicles involved fatal road accidents 2012-2018 ") +   
   
 #labs(x = "", y = "Number of vehicles") +  
 theme(plot.title = element\_text(hjust = 0.5),  
 axis.line.y = element\_blank(),  
 axis.ticks.y = element\_blank(),  
 axis.line.x = element\_blank(),  
 axis.ticks.x = element\_blank(),  
 axis.title.y = element\_blank(),  
 axis.title.x = element\_blank(),  
 axis.text.x = element\_text(size = 6, angle = 45, vjust = 1, hjust = 1),  
 panel.background=element\_blank(),  
 panel.grid.major.y = element\_line(size = 0.1, linetype = 'solid',colour = "white"),  
 panel.ontop = TRUE  
   
 )  
   
  
p



This is a Pareto chart of the data.

The Pareto chart adds a line that runs from the top of the first bar with subsequent points indicating the cumulative percentage of each bar.

This allows us to immediately see and state that 91% of fatal accidents in the period were caused by the first four categories of motorised vehicles.

The line giving the cumulative percentages is a minor enhancement to the bar plot – but offers major gains in being able to interpret and make statements about contributions of the largest categories to the whole.

The main points to observe in the ggplot code are

Four geoms are used. geom\_col is used to represent the bars. geom\_line is used to represent the Pareto line. geom\_point is used for the grey points on the line. geom\_text is used to label the point positions. Note how these geoms have their own aes functions. This overrides the aes settings in the ggplot function. geom\_line, geom\_point andgeom\_text have their own aes function because their y values represent cumulative totals.

By default ggplot would plot the line directly on top of (invisible) points vertically aligned with x-axis breaks. E.g. the line would start at the x value indicated by the value Private Car. However, it is typical for Pareto charts to start at the right most edge of the bar. To achieve that I have had to nudge the (invisible) points underpinning the line using position=position\_nudge(x = 0.5, y = 0) As the geom\_point and geom\_text elements follow the line, I have done the same with their positioning.

library(scales)  
  
total\_accidents <- sum(fatal\_acc\_df\_cumulative$n)  
  
# plot  
p <-   
 ggplot(fatal\_acc\_df\_cumulative, aes(x = reorder(vehicle, -n), y = n)) +  
 geom\_col(width = 1, fill = "#0571b0", color="white") +   
   
 scale\_y\_continuous(breaks=seq(0,1200, by =100)) +   
   
#scale\_y\_continuous(sec.axis = sec\_axis(~(./total\_accidents)\*100, name = "Percentage")) +   
   
 # the pareto line   
 geom\_line(aes(x = reorder(vehicle, -n), y = cumulative\_freq\*total\_accidents), position=position\_nudge(x = 0.5, y = 0), group = 1, col="#ca0020") + # NB: Must use "group = 1"  
 # points on the pareto line   
 geom\_point(aes(x = reorder(vehicle, -n), y = cumulative\_freq\*total\_accidents), position=position\_nudge(x = 0.5, y = 0), size =1, col="grey") +  
 geom\_text(aes(x = reorder(vehicle, -n), y = cumulative\_freq\*total\_accidents, label = sprintf("%1.2i%%", round(cumulative\_freq\*100,0))), size = 3, nudge\_x = 0.4,nudge\_y = 80) +  
   
 labs(x = "", y = "Number of vehicles") +  
 theme(plot.title = element\_text(hjust = 0.5),  
 axis.line.y = element\_blank(),  
 axis.ticks.y = element\_blank(),  
 axis.line.x = element\_blank(),  
 axis.ticks.x = element\_blank(),  
 axis.text.x = element\_text(size = 6, angle = 45, vjust = 1, hjust = 1),  
 panel.background=element\_blank(),  
 panel.grid.major.y = element\_line(size = 0.1, linetype = 'solid',colour = "white"),  
 panel.ontop = TRUE  
   
 )  
   
  
p

