

ladder_pH.R

davidharvey

Sat Feb 27 13:48:25 2016

```
# function to plot ladder diagram for weak acids and weak bases
# pka_list: list of pka values in order of most acidic to
#           least acidic; default values are for citric acid
# ph_axis: logical; defaults to FALSE but TRUE draws pH axis
# ph_limit: limits for pH axis; defaults to 0 to 14
# type: the type of ladder diagram; options are "arrow," which is the
#        default, or "strip"
# shade: choice of "color" or "gray" for shading of strip diagrams
# buffer: logical; defaults to FALSE, but TRUE will add buffer regions
# species: option to enter name of weak acid to add as title for plot;
#           defaults to NULL, which supresses main title
# labels: option to enter vector of labels for legend; defaults to
#         NULL, which uses a default legend
# locate: x-axis location of arrow or center of strip; defaults to 2,
#         which is practical lower limit; increase in steps of three
#         will separate diagrams; practical upper limit is 12
# overlay: logical; defaults to FALSE, but setting to TRUE allows for
#         adding a new ladder diagram
```

```
library(shape)
```

```
ladder_pH = function(pka_list = c(3.128, 4.761, 6.396),
                    ph_axis = FALSE,
                    ph_limit = c(0, 14),
                    type = "arrow",
                    shade = "color",
                    buffer = FALSE,
                    species = NULL,
                    labels = NULL,
                    locate = 2,
                    overlay = FALSE){

  # initial set-up; creates vector of limits for adding labels;
  # creates counter, n, for the number of alpha values;
  # sets colors for strip version of ladder diagram

  pkas = pka_list
  n = length(pkas)
  limits = c(ph_limit[1], pkas, ph_limit[2])
  if (shade == "color") {
    col.func = colorRampPalette(c("steelblue2", "lightyellow2"))
    colors = col.func(n + 1)
  } else {
    col.func = colorRampPalette(c("gray50", "gray90"))
    colors = col.func(n + 1)
  }
}
```

```

# creates default set of alpha labels if labels are not provided

if (is.null(labels) == TRUE) {
  labels = rep(0, n + 1)
  for (i in 1:(n + 1)) {
    num.protons = n - i + 1
    labels[i] = eval(substitute(expression(alpha[I]),
                                list(I = num.protons)))
  }
}

# routines for plotting the ladder diagrams for each possible set
# of options: new or overlay; arrow or strip; with or without
# pH axis, and with or without buffer regions

if (overlay == FALSE) {if (ph_axis == FALSE) {
  phax = "n"
  phlabel = "pH"
  phaxis = ""
} else {
  phax = "s"
  phlabel = ""
  phaxis = "pH"
}
plot(NULL, xlim = c(0,14), ylim = c(ph_limit[1],ph_limit[2]),
      type = "n", xaxt = "n", yaxt = phax,
      bty = "n", xlab = "", ylab = phaxis,
      xaxs = "i", yaxs = "i")
text(locate + 0.25, ph_limit[2] - (ph_limit[2] - ph_limit[1])/25,
     phlabel, pos = 4)
}

if (type == "arrow") {
  Arrows(locate, ph_limit[1], locate, ph_limit[2], lwd = 2,
        arr.type = "simple")
  segments(x0 = rep(locate - 0.3, n), y0 = pkas,
          x1 = rep(locate + 0.3, n), y1 = pkas, lwd = 2)
} else if (type == "strip") {
  for (i in 1:(n + 1)) {
    filledrectangle(mid = c(locate, (limits[i] + limits[i + 1])/2),
                  wx = 0.5, wy = limits[i + 1] - limits[i],
                  col = colors[i], lcol = "black")
  }
} else {
  return(paste(type, " is not an option.", sep = ""))
}

for (i in 1:n) {
  text(x = locate + 0.25, y = pkas[i],
       labels = pkas[i], pos = 4)
}
for (i in 1:(n + 1)){
  text(x = locate - 0.25, y = (limits[i + 1] + limits[i])/2,
       labels[i], pos = 2)
}

```

```

}
if (buffer == TRUE) {
  if (n == 1) {
    segments(x0 = locate, y0 = pkas - 1, x1 = locate, y1 = pkas + 1,
            lwd = 5, lend = "butt")
  } else { for (i in 1:n) {
    if (i %% 2 == 0){
      segments(x0 = locate + 0.05, y0 = pkas[i] - 1, x1 = locate + 0.05,
              y1 = pkas[i] + 1, lwd = 5, lend = "butt")
    } else {
      segments(x0 = locate - 0.05, y0 = pkas[i] - 1, x1 = locate - 0.05,
              y1 = pkas[i] + 1, lwd = 5, lend = "butt")
    }
  }
}
}
}
}
}
if (is.null(species) == FALSE) {
  text(x = locate - 1, y = ph_limit[2], species, pos = 2,
       srt = 90, col = "darkred")
}
}

# code to test

ladder_pH(ph_axis = TRUE, type = "arrow", buffer = TRUE, species =
"citric acid", locate = 2)
ladder_pH(type = "arrow", buffer = TRUE, locate = 5, overlay = TRUE,
pka_list = c(2.33, 4.42, 9.95), species = "glutamic acid")
ladder_pH(type = "strip", buffer = TRUE, species = "citric acid",
ph_axis = TRUE, shade = "color", locate = 8, overlay = TRUE)
ladder_pH(type = "strip", buffer = TRUE, locate = 11, overlay = TRUE,
pka_list = c(2.33, 4.42, 9.95), species = "glutamic acid",
shade = "gray")

```

