Functions for Generating and Plotting Titration Curves

David T. Harvey

2016-02-10

Gathered here are functions for generating and plotting normal, first derivative, and second derivative titration curves, including acid-base titrations, complexation titrations, redox titrations, and precipitation titrations. This document provides a basic introduction to the functions and worked examples.

Acid-Base Titrations

The following 10 functions are available for the titration of acids and bases.

function	analyte(s)	titrant
sa_sb	monoprotic strong acid	strong base
sb_sa	monoprotic strong base	strong acid
wa_sb	monoprotic weak acid	strong base
wb_sa	monoprotic weak base	strong acid
diwa_sb	diprotic weak acid	strong base
diwb_sa	diprotic weak base	strong acid
triwa_sb	triprotic weak acid	strong base
triwb_sa	triprotic weak base	strong acid
wamix_sb	mixture of two monoprotic weak acids	strong base
wbmix_sa	mixture of two monoprotic weak bases	strong acid

Each function uses the general approach outlined in *Principles of Quantitative Chemical Analysis* by Robert de Levie (McGraw-Hill, 1997) in which a single master equation is used to calculate the progress of a titration. For example, the function `wa_sb` calculates the volume of strong base needed to achieve a particular pH using this equation

$$\text{volume} = V_a \times \left\{ \frac{C_a \times \alpha - \Delta}{C_b + \Delta} \right\}$$

where V_a and C_a are, respectively, the initial volume and the initial concentration of the weak acid analyte, C_b is the initial concentration of the strong base titrant, Δ is defined as $\left[\mathrm{H}^+\right] - \left[\mathrm{OH}^-\right]$, and α, which is defined as

$$\alpha = \frac{K_a}{\left[\mathrm{H}^+\right] + K_a}$$

is the fraction of weak acid present in its conjugate weak base form.

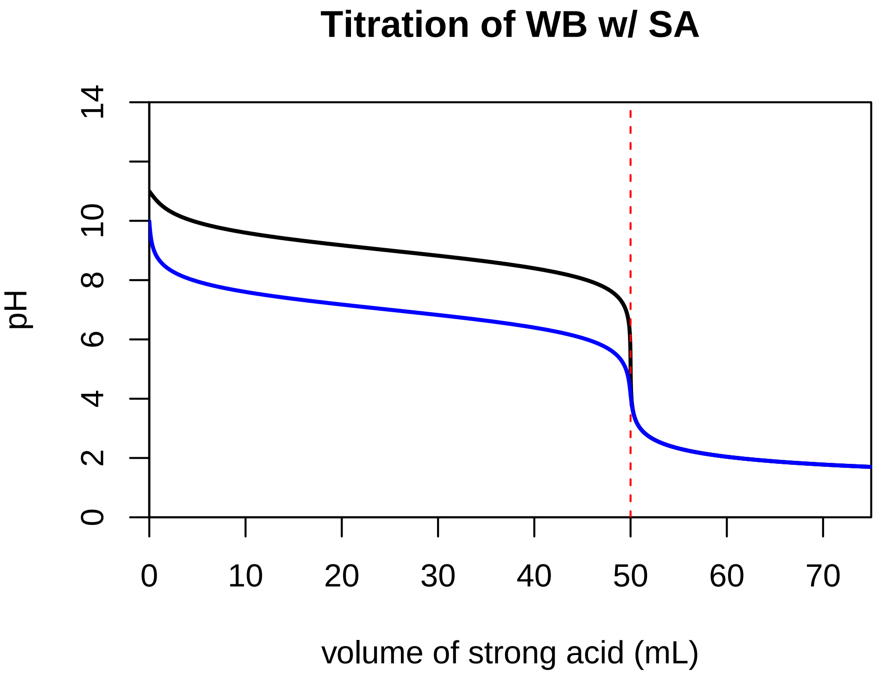
The function calculates the volume of titrant needed to achieve a particular pH over a range that extends from a pH of 1 to a pH equal to pK_w. Because some of the calculated volumes are negative—equivalent to adding a strong acid to achieve a pH less than that of the original solution—and some of the calculated volumes are quite large, prior to returning results the function removes all negative volumes and all volumes greater than twice the volume of the titration curve's last equivalence point.

The function's arguments, each of which has a default value, are intuitive; a representative example of a function is shown here for the titration of a weak base with a strong acid

```
wb_sa = function(conc.acid = 0.1, conc.base = 0.1, pka = 9, pkw = 14,
  vol.base = 50, plot = TRUE, eqpt = FALSE, overlay = FALSE, ...)
```

Note that the equilibrium constant for a weak base is provided using the pK_a value for the base's conjugate weak acid, and that you can adjust pK_w when using a solvent other than water. The option `plot` creates a plot of the titration curve with axes fixed to display the pH from 0 to 14 and to display the volume of titrant from 0 to 1.5× the titration curve's final equivalence point. The option `eqpt` adds a vertical line at the equivalence point and the option `overlay` allows for displaying two or more titration curves on the same plot; you can pass other plot options, such as color and a main title.

```
wb_sa(eqpt = TRUE, main = "Titration of WB w/ SA")
wb_sa(pka = 7, col = "blue", overlay = TRUE)
```



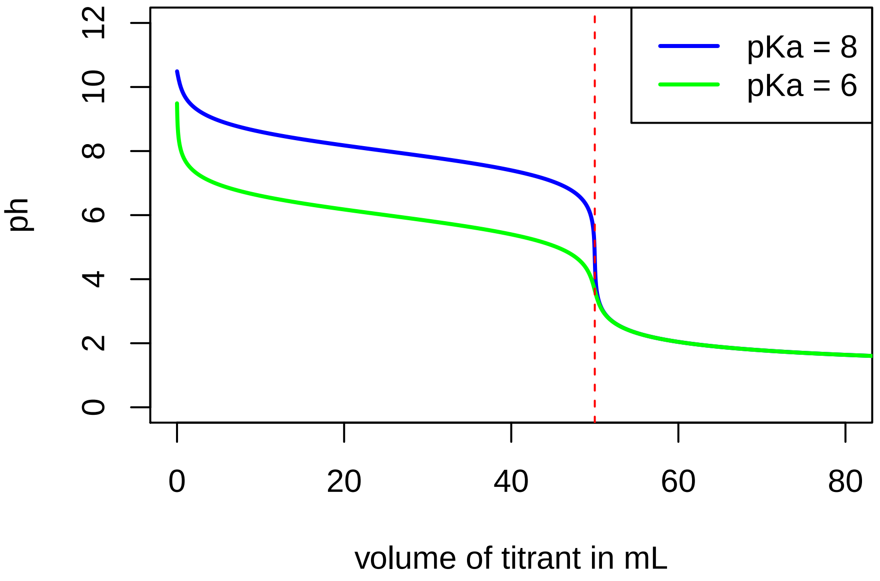
When the function is assigned to an object, it returns to the object a data frame with volumes in the first column and pH values in the second column; we can use the function `head` to examine the object's first six values.

```
wb1 = wb_sa(pka = 8)
wb2 = wb_sa(pka = 6)
head(wb1)
```

```
##        volume    ph
## 1 0.006739371 10.49
## 2 0.013979345 10.48
## 3 0.021226851 10.47
## 4 0.028485659 10.46
## 5 0.035759545 10.45
## 6 0.043052291 10.44
```

We can use this object to prepare a customized plot, as demonstrated here where we use the `plot` command's options to adjust the limits of the x-axis and the y-axis from their default values. We also add a line for the equivalence point and a legend.

```
plot(wb1, ylim = c(0, 12), xlim = c(0, 80), type = "l", col = "blue",
     lwd = 2, xlab = "volume of titrant in mL")
lines(wb2, col = "green", lwd = 2)
abline(v = 50, col = "red", lty = 2)
legend(x = "topright", legend = c("pKa = 8", "pKa = 6"),
       col = c("blue", "green"), lty = 1, lwd = 2)
```



Complexation Titrations

The function `metal_edta` calculates and plots the titration curve for the analysis of a metal ion using EDTA as the titrant. The function uses the general approach outlined in *Principles of Quantitative Analysis* by Robert de Levie (McGraw-Hill, 1997) in which a single master equation

$$\text{volume} = V_M \times \left\{ \frac{C_M - M}{\alpha_{MY} C_Y + M} \right\}$$

is used to calculate the progress of a titration, where V_M and C_M are, respectively, the initial volume and the initial concentration of the metal ion, C_Y is the initial concentration of the EDTA titrant, α_{MY}, which is defined as

$$\alpha_{MY} = \frac{K_{f,cond} M}{1 + K_{f,cond} M}$$

is the fraction of metal bound with EDTA, and M is the concentration of metal in solution not bound with EDTA. The conditional formation constant, $K_{f,cond}$ is calculated from the formation constant, K_f, for the metal-EDTA complex, the fraction of EDTA, α_Y, present in its fully unprotonated form, and the fraction of metal, α_M, not bound by an auxiliary complexing agent.

$$K_{f,cond} = \alpha_Y \alpha_M K_f$$

The value of α_Y is a function of EDTA's acid-base dissociation constants and the solution's pH, and is calculated by the function. Values of α_M are found in tables and entered into the function as one of its arguments; for example, Table 9.12 in *Analytical Chemistry 2.1* provides values for several metal ions in the presence of a variety of concentrations of NH_3.

The function calculates the volume of titrant needed to achieve a particular pM over a range that extends from 0 to logKf + 2. Because some of the calculated volumes are negative—equivalent to adding analyte instead of titrant to achieve a pM less than that for the initial solution—and some of the calculated volumes are quite large, prior to returning results the function removes all negative volumes and all volumes greater than twice the volume of the titration curve's equivalence point.

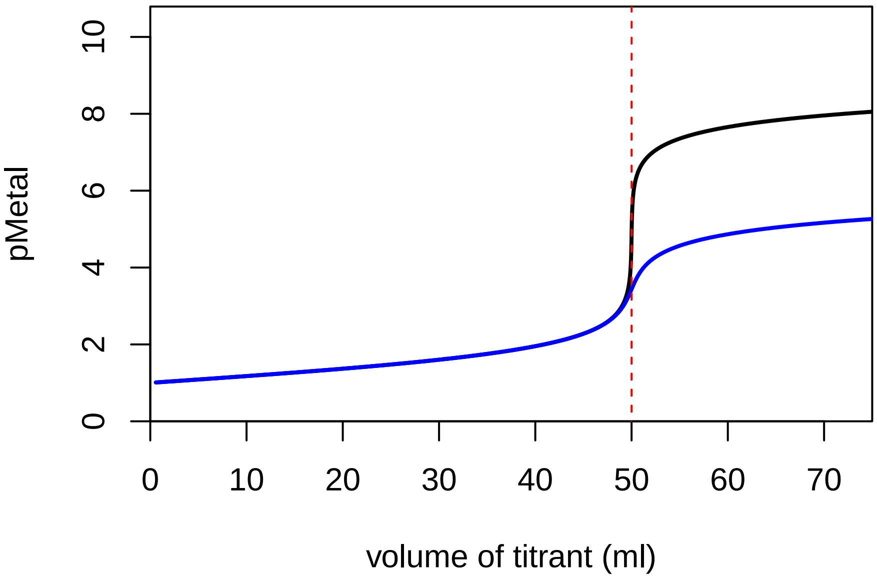
The function's arguments, each of which has a default value, are intuitive

```
metal_edta = function(conc.metal = 0.1, conc.edta = 0.1, vol.metal = 50,
  ph = 10, logkf = 8.79, alpha.metal = 1, eqpt = TRUE, overlay = FALSE, ...)
```

and include, through `alpha.metal`, the ability to consider the presence of an auxiliary complexing agent.

As shown below, the function produces a plot of the titration curve with options to overlay two or more titration curves and to add a marker for the equivalence point. The limits for the x-axis and for the y-axis are fixed, but you can pass along other plot options.

```
metal_edta(eqpt = TRUE)
metal_edta(logkf = 6, col = "blue", overlay = TRUE)
```



Redox Titrations

The function `redox_titration` calculates and plots the titration curve for the analysis of a reducing agent using an oxidizing agent as the titrant. The function uses the general approach outlined in *Principles of Quantitative Analysis* by Robert de Levie (McGraw-Hill, 1997) in which a single master equation

$$\text{volume} = V_a \times \left\{ \frac{n_a C_a \alpha_a}{n_t C_t \alpha_t} \right\}$$

is used to calculate the progress of a titration, where V_a and C_a are, respectively, the initial volume and the initial concentration of the analyte, C_t is the initial concentration of the titrant, n_a and n_t are, respectively, the number of electrons in the analyte's and the titrant's half-reaction, α_a is the fraction of the analyte in its final, oxidized form, and α_t is the fraction of the titrant in its final reduced form; the latter two values are given by

$$\alpha_a = \frac{k_a^{n_a}}{h^{n_a} + k_a^{n_a}}$$

$$\alpha_t = \frac{h^{n_t}}{h^{n_t} + k_t^{n_t}}$$

where k_a and k_t are functions of the analyte's and the titrant's standard state or formal reduction potential

$$k_a = 10^{-E_a^o/0.05916}$$

$$k_t = 10^{-E_t^o/0.05916}$$

adjusted, as needed, for other species that participate in the redox reaction (see last section of this document for a worked example), and where h is a function of the solution's potential

$$h = 10^{-E/0.05916}$$

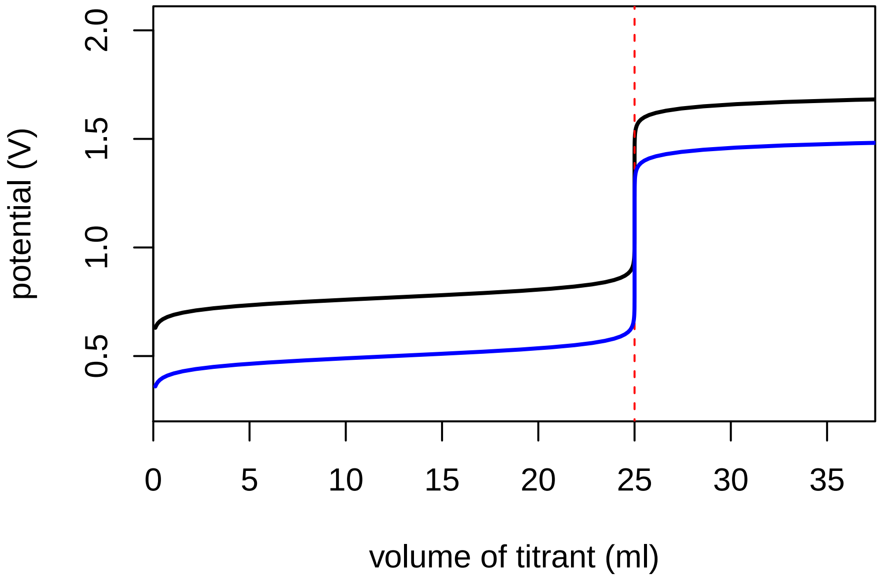
The function calculates the volume of titrant needed to achieve a particular potential over a range from -3 to $+3$ volts (vs. SHE). Because some of the calculated volumes are negative—equivalent to adding analyte instead of titrant to achieve a potential that is more negative (or less positive) than the original potential—and some of the calculated volumes are quite large, prior to returning results the function removes all negative volumes and all volumes greater than twice the volume of the titration curve's equivalence point.

The function's arguments, each of which has a default value, are intuitive

```
redox_titration = function(conc.analyte = 0.01, vol.analyte = 25, pot.analyte = 0.77,
  elec.analyte = 1, conc.titrant = 0.01, pot.titrant = 1.7, elec.titrant = 1,
  eqpt = FALSE, overlay = FALSE, ...)
```

As shown below, the functions produce a plot of the titration curve, with options to overlay two or more titration curves and to add a marker for the equivalence point. The limits for the x-axis and for the y-axis are fixed, but you can pass along other plot options.

```r
redox_titration(eqpt = TRUE)
redox_titration(pot.analyte = 0.5, pot.titrant = 1.5, col = "blue", overlay = TRUE)
```



Precipitation Titrations

The following functions are available for precipitation titrations:

- ppt_analyte: displays pAnalyte as a function of the volume of titrant when titrating a single analyte
- ppt_titrant: displays pTitrant as a function of the volume of titrant when titrating a single analyte
- ppt_mixture: displays pTitrant as a function of the volume of titrant when titrating a mixture of two analytes

These functions use the general approach outlined in *Principles of Quantitative Analysis* by Robert de Levie (McGraw-Hill, 1997) in which a single master equation is used to calculate the progress of a titration. For example, the function `ppt_analyte` calculates the volume of titrant needed to achieve a particular pAnalyte using this equation

$$\text{volume} = V_a \times \left\{ \frac{C_a - A + T}{C_t + A - T} \right\}$$

where V_a and C_a are, respectively, the initial volume and the initial concentration of the analyte, C_t is the initial concentration of the titrant, and A and T are the concentrations of analyte and of titrant, respectively, as controlled by the precipitate's solubility. Note that this equation assumes a 1:1 stoichiometry between the analyte and the titrant.

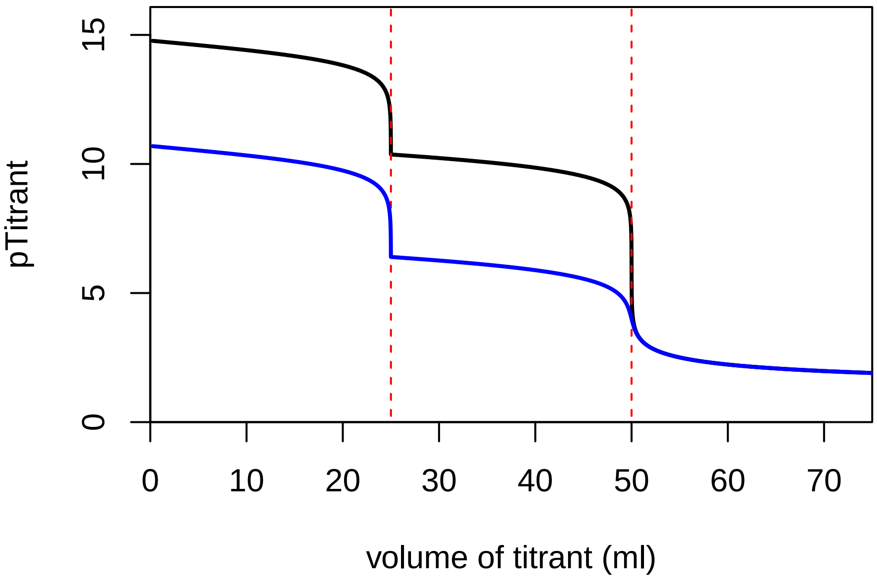
The function calculates the volume of titrant needed to achieve a particular pAnalyte over a range that extends from 1 to logKsp. Because some of the calculated volumes are negative—equivalent to adding analyte instead of titrant to achieve a pAnalyte that is smaller or a pTitrant that is larger than is present initially—and some of the calculated volumes are quite large, prior to returning results the function removes all negative volumes and all volumes greater than twice the volume of the titration curve's equivalence point.

The function's arguments, each of which has a default value, are intuitive; a representative example is shown here for the titration of a mixture of analytes

```
ppt_mixture = function(conc.analyte1 = 0.05, conc.analyte2 = 0.05,
  vol.analyte = 25, conc.titrant = 0.05, pksp1 = 16.08, pksp2 = 11.97,
  eqpt = FALSE, overlay = FALSE, ...)
```

As shown below, the functions produce a plot of the titration curve, with options to overlay two or more titration curves and to add a marker for the equivalence point. The limits for the x-axis and the y-axis are fixed, but you can pass along other plot options.

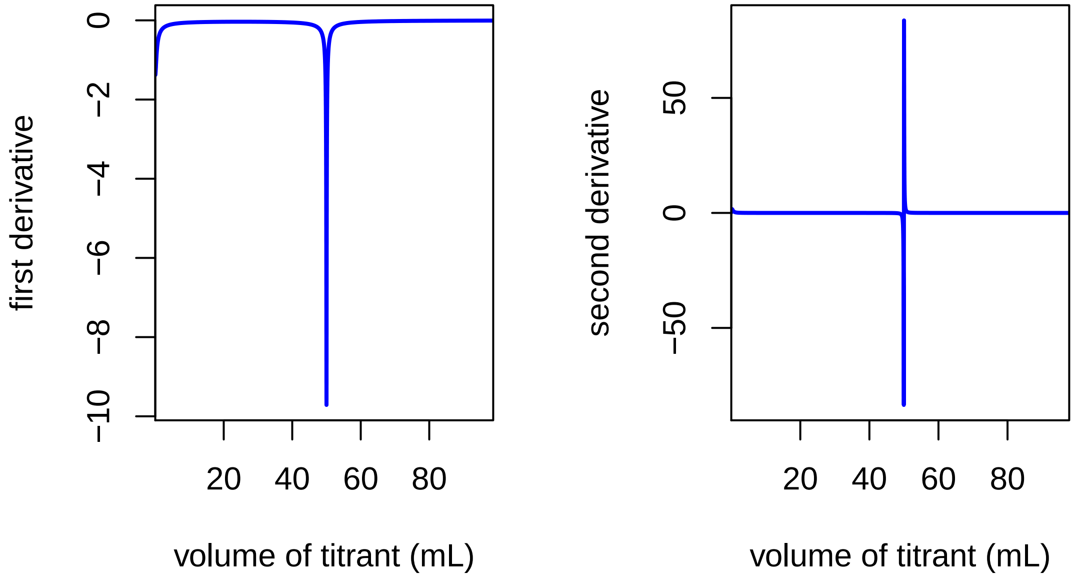
```r
ppt_mixture(eqpt = TRUE)
ppt_mixture(pksp1 = 12, pksp2 = 8, col = "blue", overlay = TRUE)
```



Deriviative Titration Curves

The function `derivative` accepts an object created by any of the titration functions described above and returns a plot of the titration curve's first derivative and its second derivative, and a data frame that contains the values used to prepare the two titration curves.

```r
wbd = derivative(wb1)
```



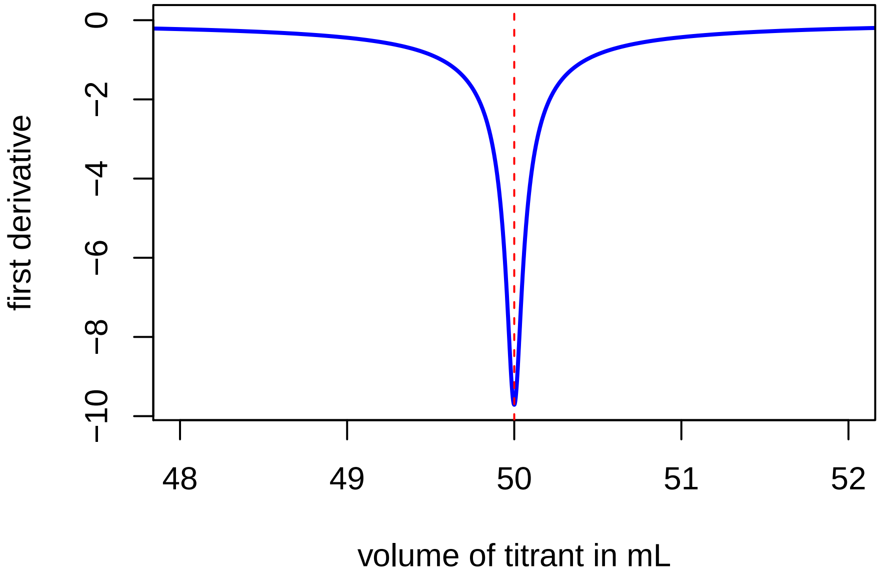
As shown below, the object returned by `derivative` consists of two data frames, which, in this case, are identified as `wbd$first_deriv` and `wbd$second_deriv`.

```r
str(wbd)
```

```
## List of 2
##  $ first_deriv :'data.frame':    901 obs. of  2 variables:
##   ..$ x1: num [1:901] 0.0104 0.0176 0.0249 0.0321 0.0394 ...
##   ..$ y1: num [1:901] -1.38 -1.38 -1.38 -1.37 -1.37 ...
##  $ second_deriv:'data.frame':    900 obs. of  2 variables:
##   ..$ x2: num [1:900] 0.014 0.0212 0.0285 0.0358 0.0431 ...
##   ..$ y2: num [1:900] 0.198 0.296 0.393 0.488 0.581 ...
```

You can plot a customized derivative titration curve by sending its object to the `plot` function with options of your choosing; this is particularly useful, as shown below, if you wish to highlight the portion of the titration curve that contains the equivalence point.

```r
plot(wbd$first_deriv, xlim = c(48, 52), col = "blue", type = "l", lwd = 2,
     xlab = "volume of titrant in mL", ylab = "first derivative")
abline(v = 50, col = "red", lty = 2)
```



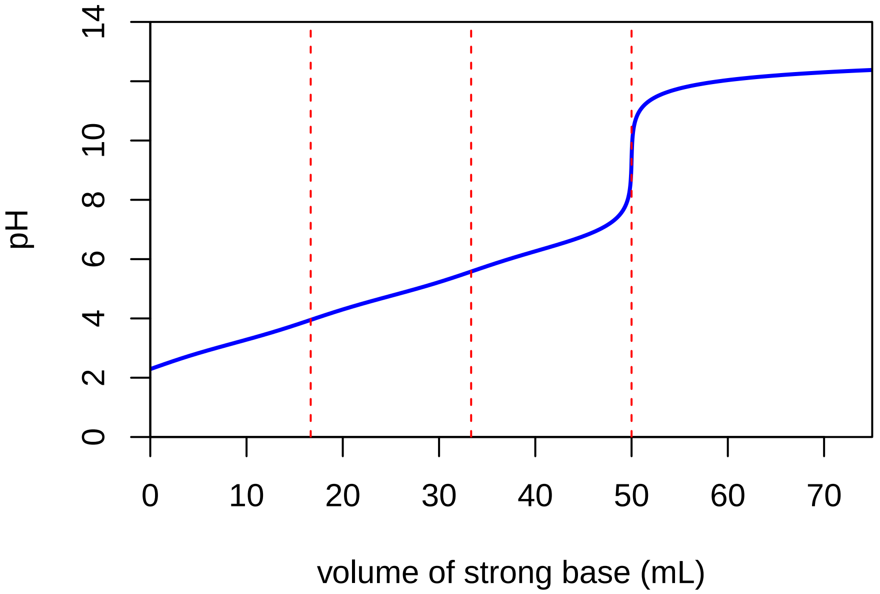
Additional Examples

Gathered here are several worked examples, each adapted from an end-of-chapter problem in Chapter 9 of *Analytical Chemistry 2.1*.

Problem 9.1e. Plot the titration curve for the analysis of 50.0 mL of 0.0400 M citric acid using 0.120 M NaOH as a titrant.

Citric acid is a triprotic weak acid with pK_a values of 3.128, 4.761, and 6.396. To calculate the titration curve, we use the function `triwa_sb`, entering only those values that differ from the function's default values.

```
triwa_sb(conc.acid = 0.0400, conc.base = 0.120, pka1 = 3.128, pka2 = 4.761,
         pka3 = 6.396, col = "blue", eqpt = TRUE)
```

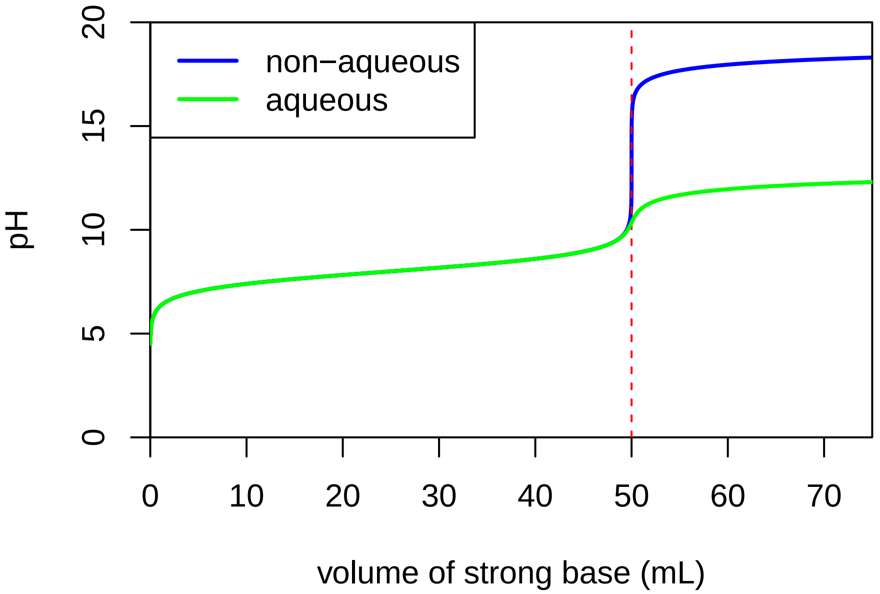


Note that because the first two pK_a values are not sufficiently different from each other, the titration curve shows just one distinct equivalence point for the titration of citric acid's third proton.

Problem 9.8. Plot the titration curve for the analysis of a 50.0 mL solution of a 0.100 M monoprotic weak acid (p$K_a = 8.0$) with 0.10 M strong base in a non-aqueous solvent for which $K_s = 10^{-20}$. You may assume that the change in solvent does not affect the weak acid's pK_a. Compare your titration curve to the titration curve when using water as the solvent.

For an analyte that is a monoprotic weak acid, we use the function `wa_sb`, entering only those values that differ from the function's default values.

```
wa_sb(pka = 8, pkw = 20, col = "blue", eqpt = TRUE)
wa_sb(pka = 8, col = "green", overlay = TRUE)
legend(x = "topleft", legend = c("non-aqueous", "aqueous"),
       col = c("blue", "green"), lty = 1, lwd = 2)
```

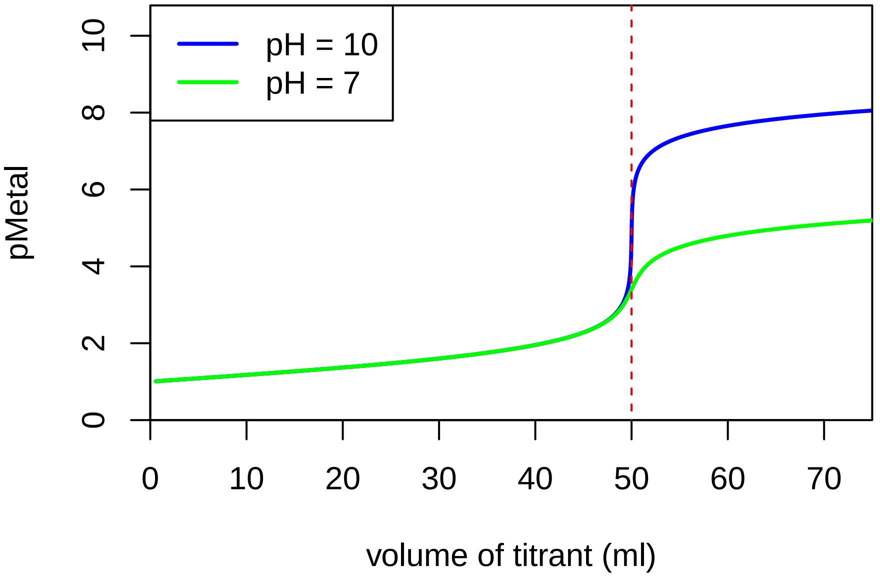


Note that the change in pH at the equivalence point is much greater for the non-aqueous solvent.

Problem 9.31. Plot the titration curve for the analysis of 50.0 mL of 0.100 Mg^{2+} using 0.100 M EDTA as the titrant at a pH of 7 and at a pH of 10.

For a complexation titration of a metal ion using EDTA as a titrant, we use the function `metal_edta`, entering only those values that differ from the function's default values (note that the default value for logkf is for the Mg^{2+}-EDTA complex).

```
metal_edta(col = "blue", eqpt = TRUE)
metal_edta(ph =  7, col = "green", overlay = TRUE)
legend(x = "topleft", legend = c("pH = 10", "pH = 7"),
       col = c("blue", "green"), lty = 1, lwd =2)
```

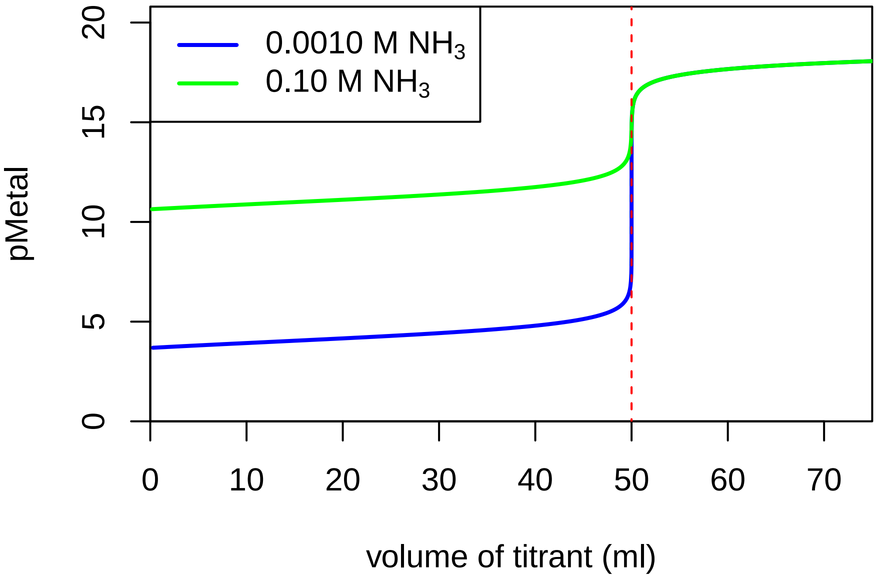


Note that the change in pM at the equivalence point is much greater at a pH of 10 because a greater fraction of the EDTA is in its fully unprotonated form at this pH.

Problem 9.32. Plot the titration curve for the analysis of 25.0 mL of 0.0500 M Cu^{2+} with 0.025 M EDTA at a pH of 10 in the presence of 0.0010 M NH_3 and in the presence of 0.10 M NH_3.

In this case, NH_3 serves as an auxiliary complexing agent, so we need to find values for α_M; from Table 9.12 in *Analytical Chemistry 2.1*, we find that the values are 0.00415 when the concentration of NH_3 is 0.0010 M and 4.63×10^{-10} when the concentration of NH_3 is 0.10 M. The value for $\log K_f$ is 18.80

```
metal_edta(conc.metal = 0.0500, conc.edta = 0.025, vol.metal = 25.0,
           alpha.metal = 0.00415, logkf = 18.80, col = "blue", eqpt = TRUE)
metal_edta(conc.metal = 0.0500, conc.edta = 0.0250, vol.metal = 25.0,
           alpha.metal = 4.63e-10, logkf = 18.80, col = "green", overlay = TRUE)
legend(x = "topleft",
       legend = c(expression(paste("0.0010 M N", H[3])),
                  expression(paste("0.10 M N", H[3]))),
       col = c("blue", "green"), lty = 1, lwd = 2)
```



Note that the change in pM at the equivalence point is much greater for the smaller concentration of NH_3; this is because a greater fraction of Cu^{2+} remains free from complexation before the equivalence point.

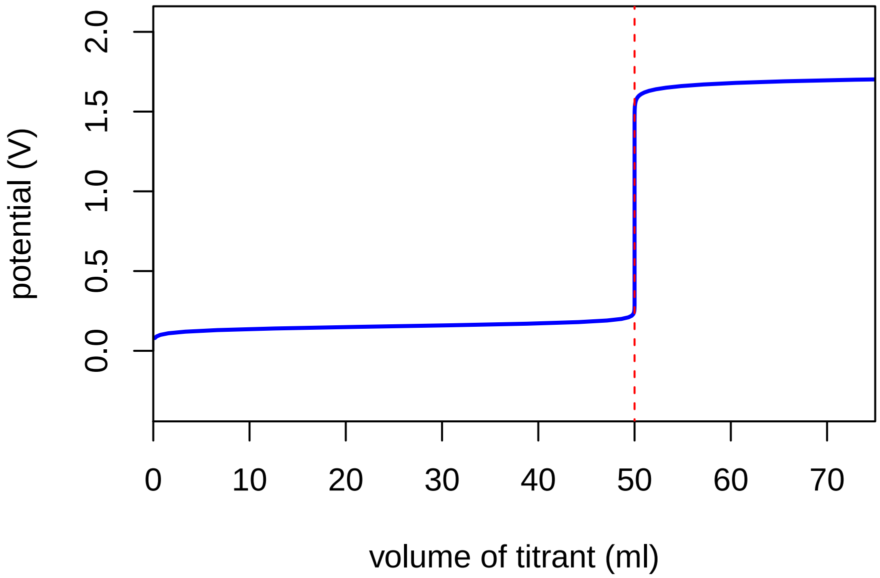
Problem 43b. Plot the titration curve for the analysis of 25.0 mL of 0.0100 M Sn^{2+} using 0.0100 M Ce^{4+} as the titrant.

For a redox titration we use the function `redox_titration`, entering only those values that differ from the function's default value. To determine the number of electrons in the analyte's and the titrant's half-reactions, we need to know the reaction's overall stoichiometry, which is

$$Sn^{2+} + 2Ce^{4+} \rightarrow Sn^{4+} + 2Ce^{3+}$$

From this, we know that Sn^{2+} undergoes a two-electron oxidation and that Ce^{4+} undergoes a one-electron oxidation. The standard state reduction potential for Sn^{4+}/Sn^{2+} is 0.154 V and the standard state reduction potential for Ce^{4+}/Ce^{3+} is 1.72 V.

```
redox_titration(pot.analyte = 0.154, elec.analyte = 2, pot.titrant = 1.72,
                col = "blue", eqpt = TRUE)
```



Problem 43c. Plot the titration curve for the analysis of 25.0 mL of 0.0100 M Fe^{2+} using 0.0100 M MnO_4^- as the titrant at pH values of 0.0, 1.0, and 2.0.

To determine the number of electrons in the analyte's and the titrant's half-reactions, we need to know the reaction's overall stoichiometry, which is

$$5Fe^{2+} + MnO_4^- + 8H^+ \rightarrow 5Fe^{3+} + Mn^{2+} + 4H_2O$$

From this, we know that Fe^{2+} undergoes a one-electron oxidation and that MnO_4^- undergoes a five-electron reduction. The standard state reduction potential for Fe^{3+}/Fe^{2+} is 0.771 V; however, because its half-reaction

$$MnO_4^- + 8H^+ + 5e^- \rightarrow Mn^{2+} + 4H_2O$$

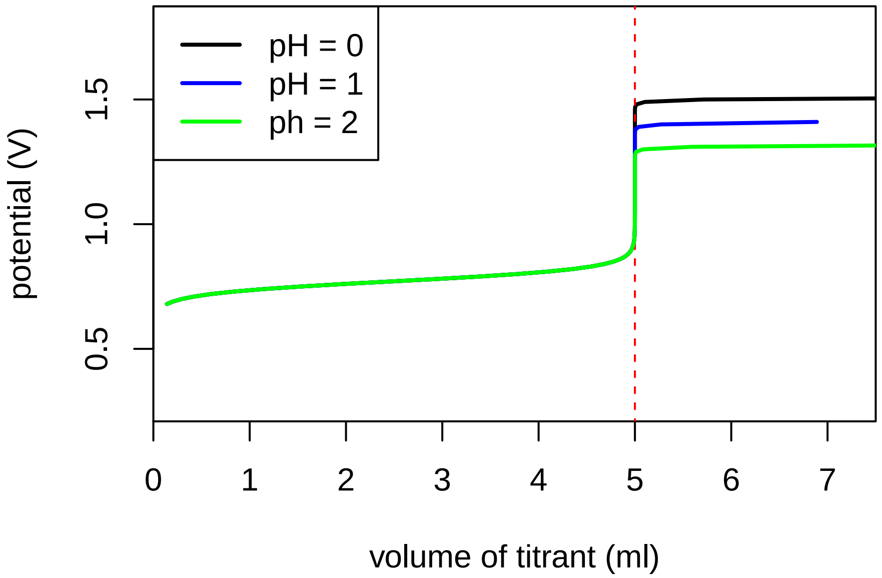
includes H^+, we must replace the standard state potential of 1.51 V for MnO_4^-/Mn^{2+} with a formal potential that corrects for the non-standard state concentrations for H^+ of 0.10 M and 0.010 M when the pH is 1.0 and 2.0, respectively. To do this we, we write out the Nernst equation for the half-reaction and substitute in the standard state concentrations for MnO_4^- and for Mn^{2+}, and the non-standard state concentration for H^+; thus, for a pH of 1.0, the formal potential is

$$E = 1.51 - \frac{0.05916}{5}\log\frac{\left[Mn^{2+}\right]}{\left[MnO_4^-\right]\left[H^+\right]^8} = \frac{0.05916}{5}\log\frac{[1]}{[1]\,[0.10]^8} = 1.415$$

For a pH of 2, the formal potential is 1.321.

Now we can plot the titration curves.

```
redox_titration(pot.analyte = 0.771, pot.titrant = 1.51, elec.titrant = 5,
                col = "black", eqpt = TRUE)
redox_titration(pot.analyte = 0.771, pot.titrant = 1.415, elec.titrant = 5,
                col = "blue", overlay =TRUE)
redox_titration(pot.analyte = 0.771, pot.titrant = 1.321, elec.titrant = 5,
                col = "green", overlay = TRUE)
legend(x = "topleft", legend = c("pH = 0", "pH = 1", "ph = 2"),
       col = c("black", "blue", "green"), lty = 1, lwd =  2)
```



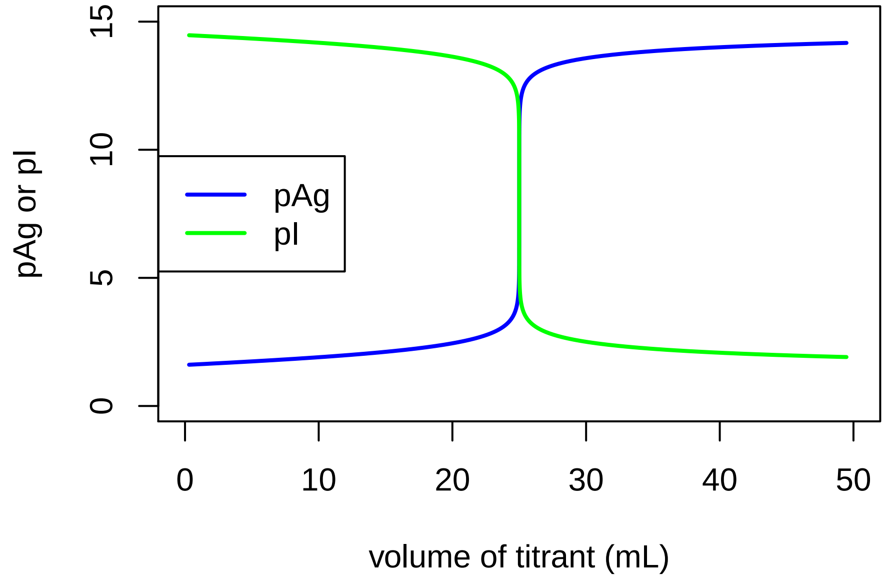
Note that the change in potential at the equivalence point decreases with an increase in pH because the potential for the titrant's half-reaction decreases with an increase in pH.

Problem 9.55. Plot the titration curve for the analysis of 50.0 mL of 0.0250 M KI using 0.0500 M $AgNO_3$ as the titrant. Prepare the titration curve so that both pAg and pI are plotted on the y-axis.

For a precipitation titration we use the function `ppt_analyte` when we monitor the analyte's concentration and `ppt_titrant` when we monitor the titrant, entering only those values that differ from the function's default values (note that the default value for pksp is for AgI). So that we have control over the y-axis label, we will save the titration curves in separate objects and plot them outside of the titration functions.

```
p.a = ppt_analyte(eqpt = TRUE)
p.t = ppt_titrant(overlay = TRUE)
```

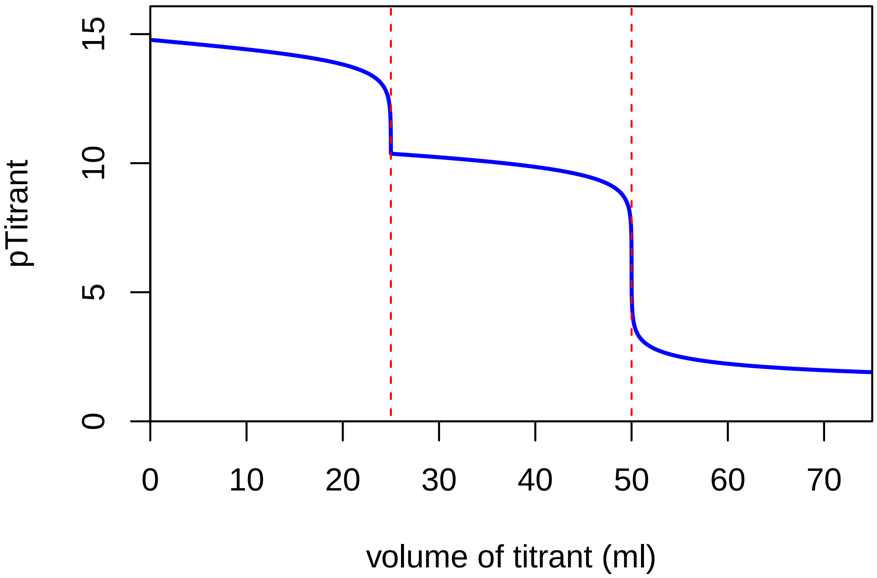
```
plot(p.a, col = "blue", type = "l", lwd = 2, xlim = c(0,50), ylim = c(0,15),
     xlab = "volume of titrant (mL)", ylab = "pAg or pI")
lines(p.t, col = "green", lwd = 2)
legend(x = "left", legend = c("pAg", "pI"), col = c("blue", "green"),
       lty = 1, lwd = 2)
```



Problem 9.56. Plot the titration curve for the analysis of 25.0 mL of a mixture that is 00500 M KI and 0.0500 KSCN using 0.0500 M $AgNO_3$ as the titrant.

For this problem we use the `ppt_mixture` function, for which the default values for pksp1 and for pksp2 are for AgI and for AgSCN.

```
ppt_mixture(col = "blue", eqpt = TRUE)
```


pTitrant
volume of titrant (ml)