Section 2.3

Cubic equations of state

Cubic equations of state are by far the most-popular equations of state. This can mainly be attributed to the their long tenure as the only practical equations of state, but also their simple functional form. We refer to such equations as cubics because they can all be written in the following form:

$$V^3+AV^2+BV+C=0$$

This form has a lot of benefits which we will discuss in part 3 of the course.

Section 2.3.1

Van der Waals equation

It is likely that most undergraduates will have encountered the van der Waals equation at some point. However, to give a high-level understanding of this equation, let us start with the ideal gas equation:

$$p=\frac{Nk_\mathrm{B}T}{V}$$

As mentioned previously, the ideal gas equation assumes that particles are infinitesmally small and experience perfectly elastic collisions. However, for most molecules, this is not the case. Firstly, molecules have volume and thus, take up space in the system, thus reducing the total amount of volume available for species to move around in. The volume of a single particle is typically denoted by the parameter b. For N particles, we reduce the available volume by Nb:

$$p=\frac{Nk_\mathrm{B}T}{V-Nb}$$

However, molecules also experience attractions between them. The impact on the pressure is two-fold. Firstly, as particles on the edge of our box are about to collide with the box surface, the attraction with molecules in the bulk will reduce their velocity. This will be proportional to the density of particles in the bulk. Similarly, particles interacting together would much rather clump together rather than venture out to the boundaries of the box, thus further reducing the pressure. This too will be proportional to the bulk density of our system. As a result, the net change to the pressure will be proportional to the density squared:

$$\Delta p\propto-\rho^2$$

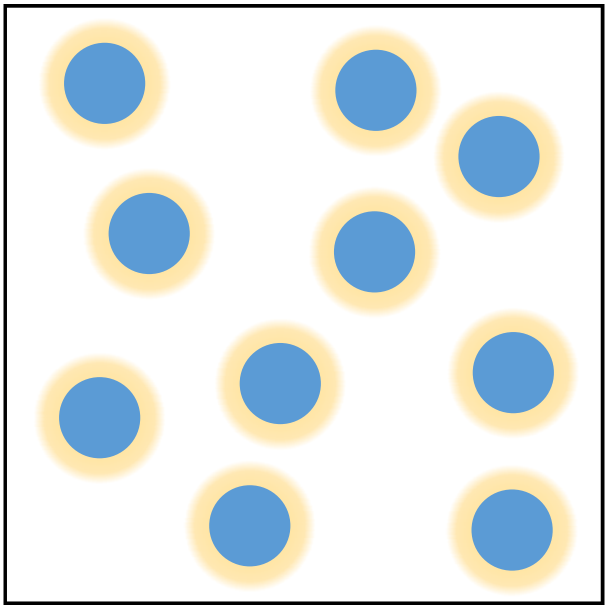
If we characterise this proportionality by a parameter a, we can write out the van der Waals equation as:

$$p = \frac{Nk_{\mathrm{B}}T}{V - Nb} - \frac{N^2 a}{V^2}$$

More commonly written out in molar form as:

$$p = \frac{RT}{v_m - b} - \frac{a}{v_m^2}$$

This was a very high-level description of the van der Waals equation; it is possible to derive it using statistical mechanics although, it is worth pointing out, van der Waals himself did not derive it this way. The equation was only ever meant to be empirical! Nevertheless, the first term can be thought of as the repulsive contribution and the second term can be thought of as the attractive contribution. Visually:



Interestingly, for all cubics we will discuss, this visual picture does not change significantly. The parameters a_m and b_m can be obtained by constraining the equation such that it must pass through the critical point of a given species:

$$a = \frac{27}{64}\frac{(RT_c)^2}{p_c}$$

$$b = \frac{1}{8}\frac{RT_c}{p_c}$$

where the subscripts c denote the critical properties of a species.

Despite its significant contributions towards the development of equations of state, the van der Waals equation is not suitable for practical applications in the liquid phase as it heavily under-estimates the saturated liquid densities:

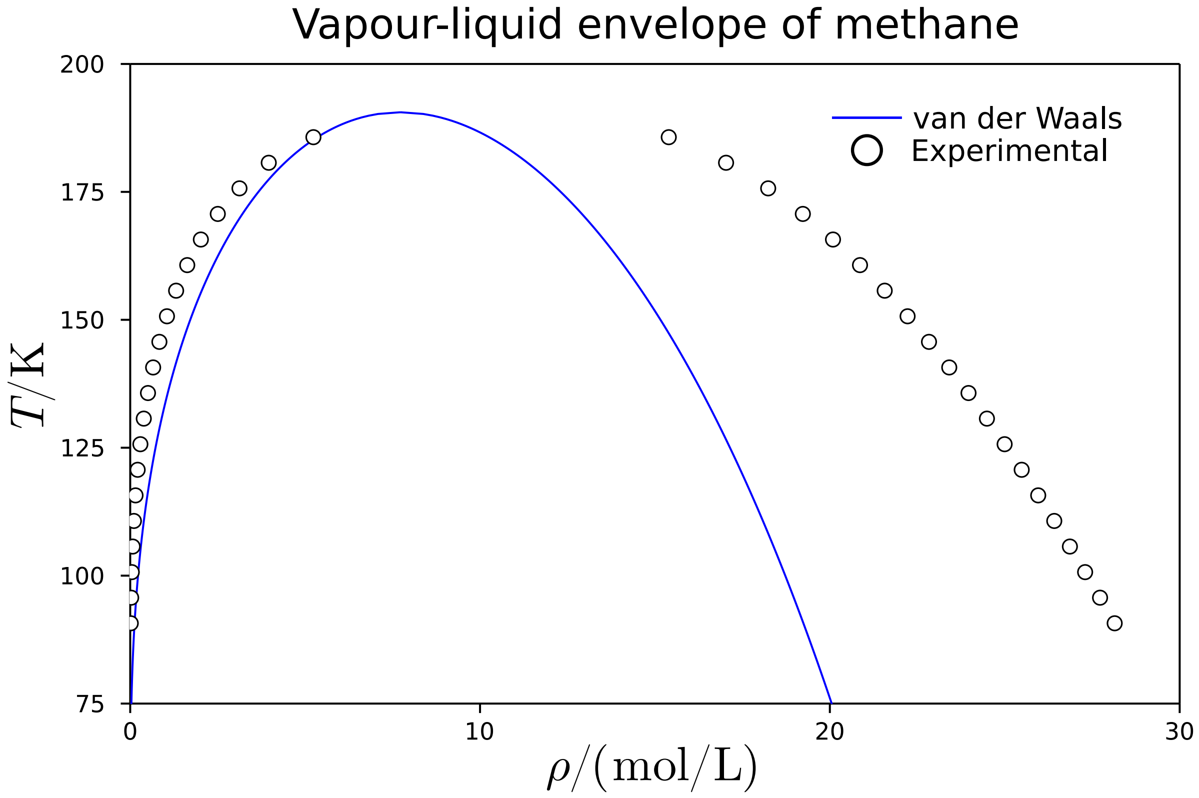
```
[(1463.37, 4.78697e-5, 0.324377), (1561.41, 4.79041e-5, 0.305764), (1664.77, 4.79387e-5, 0
```

```
begin
    species="methane"
    model = vdW([species])

    Tc,pc,Vc = crit_pure(model)

    N = 400
    T = range(0.3*Tc,Tc,length=N)

    sat = saturation_pressure.(model,T);
end
```



Even in the case of methane, which can almost be considered spherical, the van der Waals cannot reproduce the experimental data. A few modifications made by Clausius and Berthelot did come along, but none ever fully dealt with the main issues of the van der Waals equation.

Section 2.3.2

Engineering Cubics

Almost 50 years after van der Waals first derived his equation of state, the first true step towards a practically useable equation of state was that developed by Redlich and Kwong:

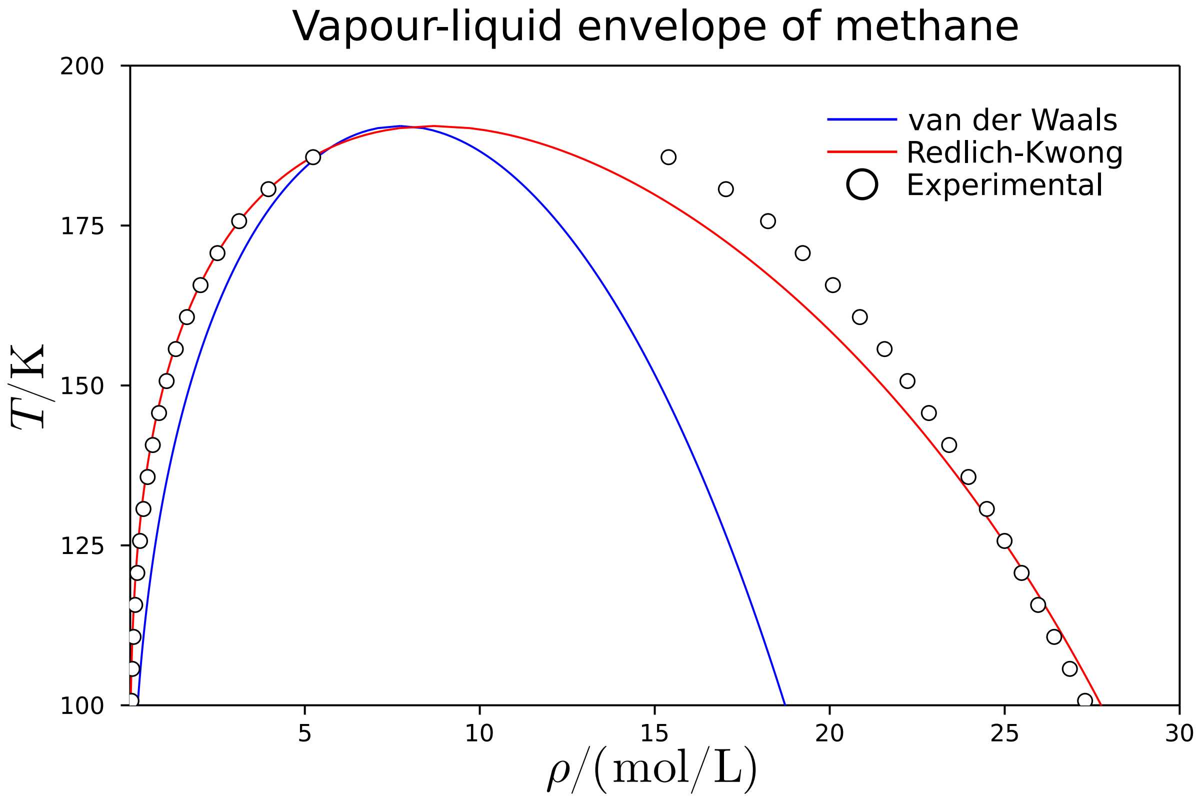
$$p = \frac{Nk_\mathrm{B}T}{V - Nb} - \frac{N^2 a}{V(V+Nb)\sqrt{T}}$$

where the parameters a and b can now be obtained from:

$$a = 0.42748 \frac{R^2 T_c^{2.5}}{p_c}$$

$$b = 0.08664 \frac{RT_c}{p_c}$$

The change made to the second term is purely empirical, with no physical meaning aside from improved modelling of gas fugacities (important in vapour-liquid calculations). In comparison to the van der Waals equation, this equation saw a substantial improvement in modelling the vapour phase:



However, the real 'game-changer' for cubic equations of state came when Soave introduced the concept of an α-function which modified the Redlich—Kwong equation in the following way:

$$p = \frac{Nk_\mathrm{B}T}{V-Nb} - \frac{N^2 a\alpha(T)}{V(V+Nb)}\ ,$$

Here, b and a are defined almost the same way, except for a_0:

$$a = 0.42748\frac{(RT_c)^2}{p_c}$$

and $\alpha(T)$, the α-function, given by:

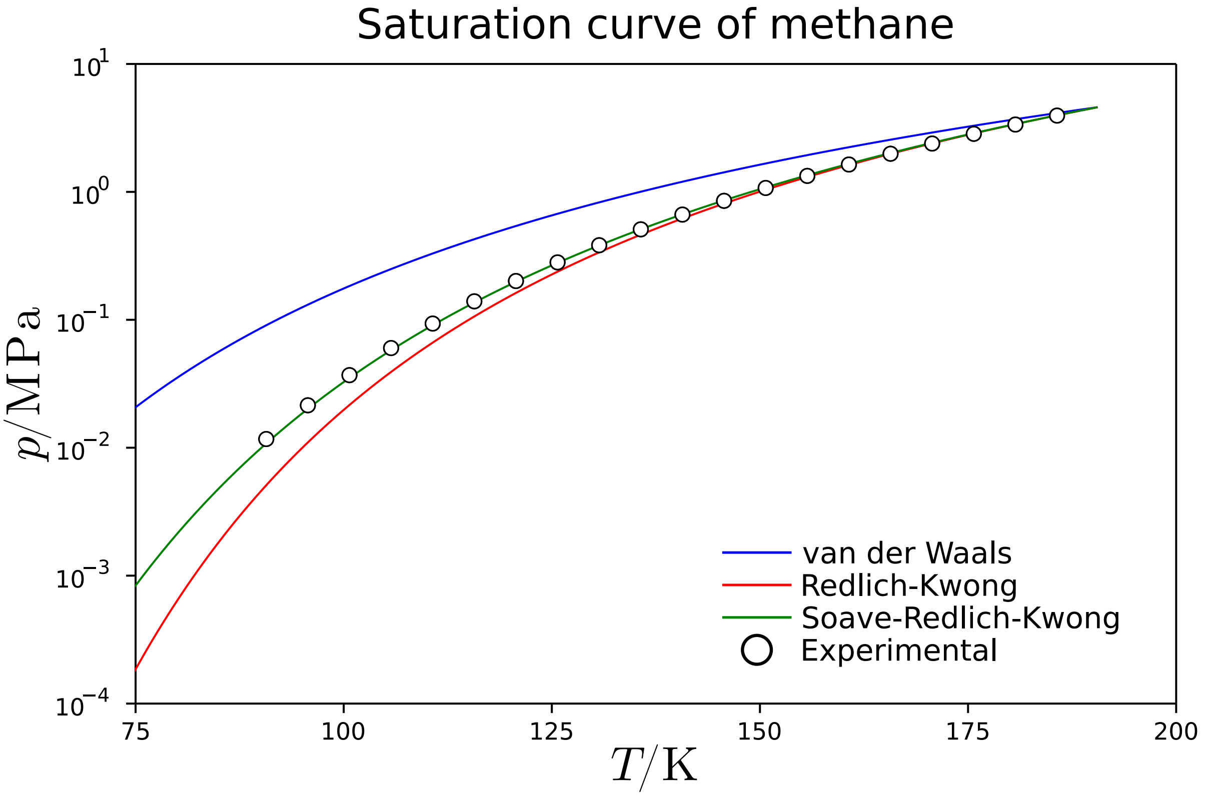
$$\alpha(T) = (1+(0.480+1.547\omega-0.176\omega^2)(1-(T/T_c))^{0.5})^2$$

where ω is the acentricity, a species-specific parameter, defined as:

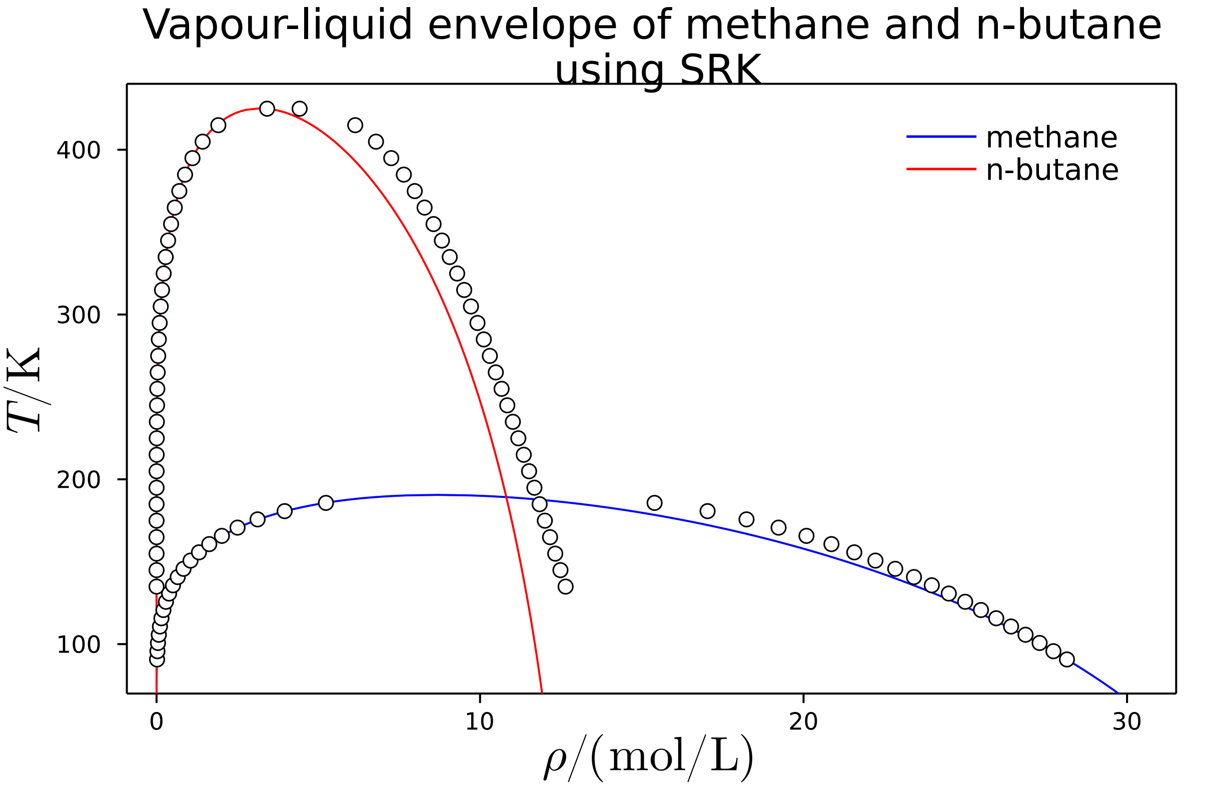
$$\omega = -\log\left(p_\mathrm{sat}/p_c\right) - 1\ \mathrm{at}\ T = 0.7T_c$$

Interestingly, the acentricity does carry some physical meaning: the more-spherical the species is, the closer its value should be to zero (such as methane or the noble gases).

The idea behind the α-function is that, if you can nail-down both the critical point (which the Redlich—Kwong equation already does) and a second point on the saturation curve around 0.7T_c, characterised by the acentricity, then, ideally, you should be able to capture the entire saturation curve. This is indeed what happens for most species:



However, even in cases like n-butane, the SRK equation isn't able to model the liquid densities accurately:



It is for this reason that Peng and Robinson (PR) developed their own cubic equation of state, also using their own α-function:

$$p = \frac{Nk_\mathrm{B}T}{V-Nb} - \frac{N^2 a\alpha(T)}{V^2+2NbV+(Nb)^2}\ ,$$

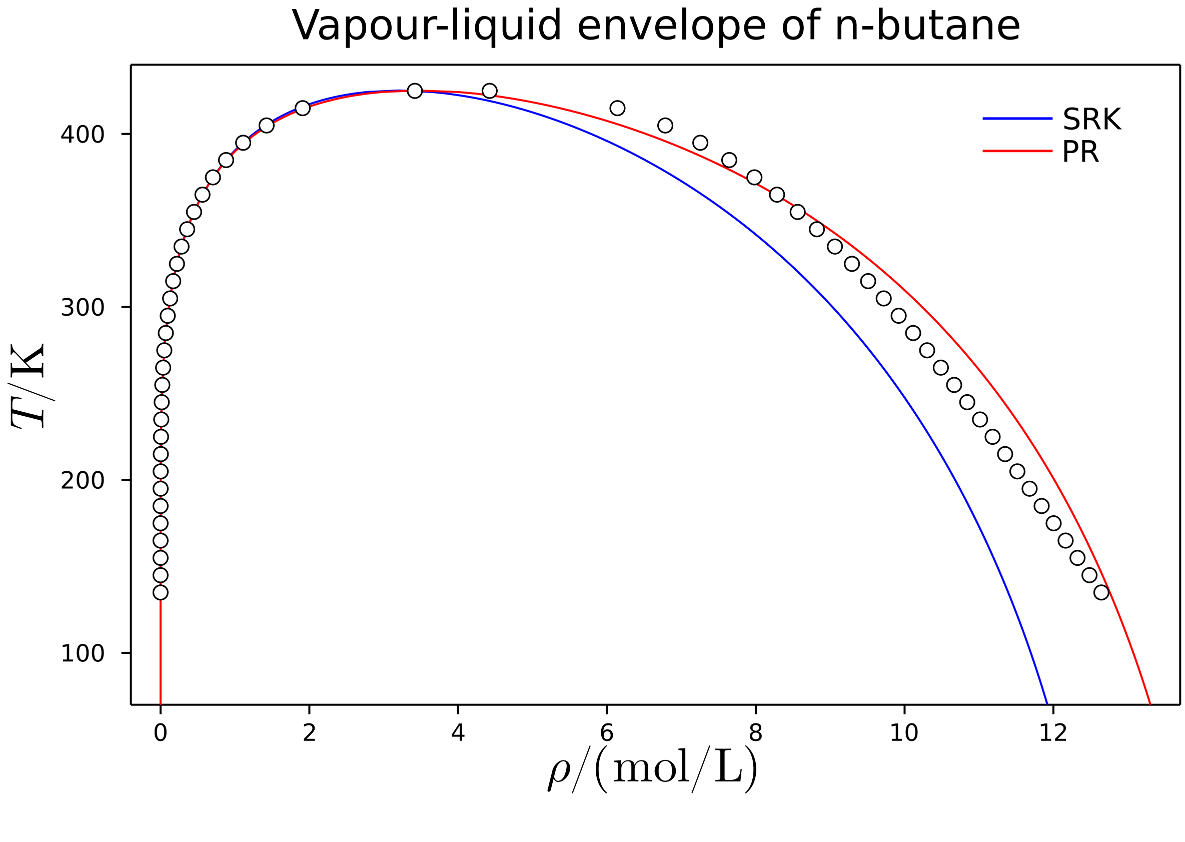
where:

$$a_0 = 0.45724\frac{(RT)^2}{p_c}$$
$$b = 0.0778\frac{RT_c}{p_c}$$

and $\alpha(T)$:

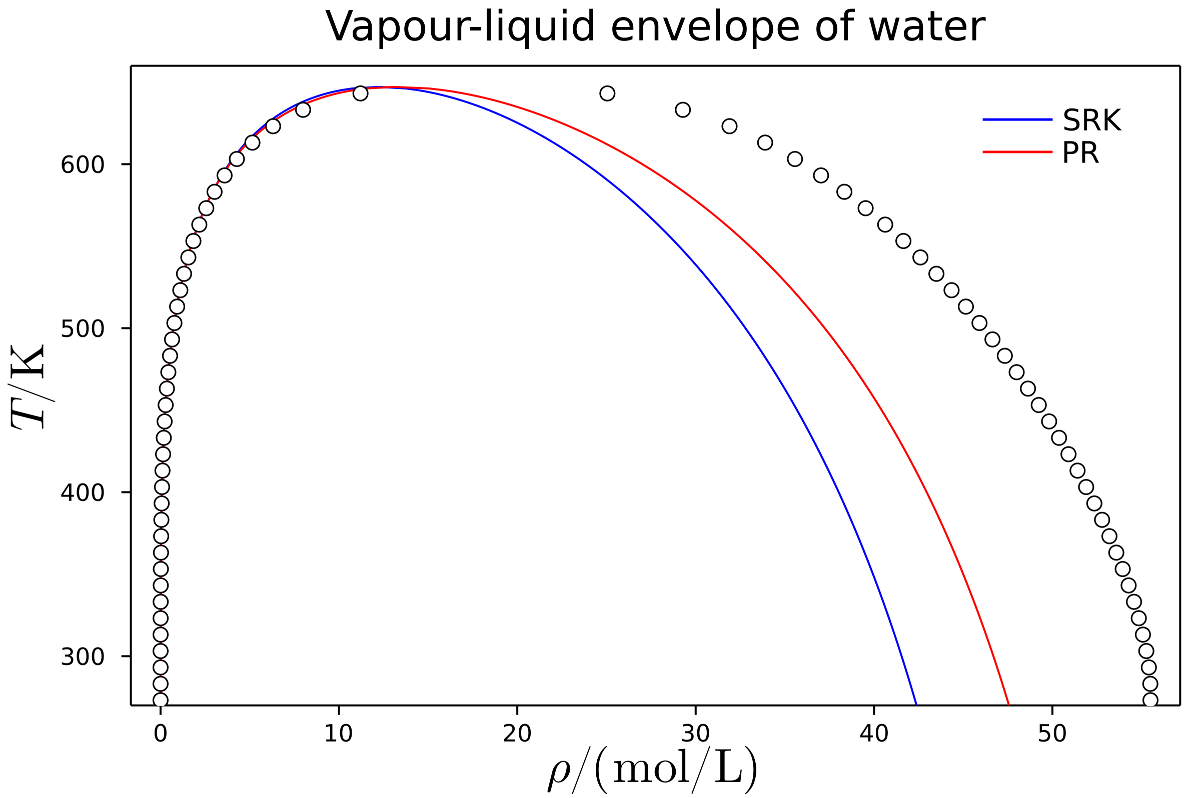
$$\alpha(T) = (1+(0.37464+1.54226\omega-0.26992\omega^2)(1-(T/T_c))^{0.5})^2$$

Indeed, using the PR equation, the improvement in modelling of liquid densities is significant:



That isn't to say SRK is no longer useful. In fact, both SRK and PR represent the industry standards for equation of state modelling as, depending on what you are trying to model, one may be more-accurate than the other.

However, it is also important to bear in mind what systems these equations of state are intended for: hydrocarbon / natural gases. If you were to try and model something like water using either of these equations the results would be disappointing:



Nevertheless, for most systems engineers are interested in, SRK and PR provide an easy way to access the full range of thermodynamic properties we might need. There have been further developments in cubic equation of state modelling (some of which will be highlighted below), including the introduction of a third (e.g., Patel-Teja) and sometimes fourth (e.g., GEOS) parameter to model a wider range of species. However, all retain the simple cubic form which can be generalised as:

$$p=\frac{Nk_{\mathrm{B}}T}{V-Nb}-\frac{N^2a\alpha(T)}{(V+r_1Nb)(V+r_2Nb)}\,,$$

where:

Equation	r_1	r_2
vdW	0	0
RK/SRK	0	1
PR	$1+\sqrt{2}$	$1-\sqrt{2}$

Meaning, if we can just write a single function to obtain the pressure in terms of the parameters r_1 and r_2, it would be compatible with all cubic equations of state we may which to use. However, if we also want to be able to obtain other properties of interest (such as heat capacities, Joule-Thomson coefficients, etc.), it is more-convenient to express our equation in terms of the Helmholtz free energy. This can be obtained by integrating the pressure in terms of the volume:

$$A=-\int pdV=-Nk_{\mathrm{B}}T\log\left(V-Nb\right)-\frac{Na}{b}\frac{\log(V+r_2Nb)-\log(V+r_1Nb)}{r_2-r_1}+c(N,T)$$

The problem here is that we re-introduce the integration constant $c(N,T)$ which, from the previous section, we know arises from the missing ideal contributions. As we know how to handle the ideal term separately, it is easier for us to focus on the residual Helmholtz free energy instead of the total (this is something we will continue to do in future sections). To obtain the residual, we deduct the ideal contribution from the total:

$$A_{\mathrm{res.}}=A-A_{\mathrm{ideal}}=-Nk_{\mathrm{B}}T\log\left(1-Nb/V\right)-\frac{Na\alpha(T)}{b}\frac{\log(V+r_2Nb)-\log(V+r_1Nb)}{r_2-r_1}$$

With this equation, we should be able to obtain any thermodynamic property for any cubic equation of state. For this reason, it will be useful for the reader to implement it themselves.

Task: Implementing a generalised function for cubics

In this exercise, we assume that we have specified a composition (moles), volume (m^3), and temperature (K) of the system and already have the parameters needed to characterise our system (a and b). To write a generalised approach for cubic equations of state, were are going to use Julia's multiple dispatch feature where we will write one function `a_res(model::CubicModel,V,T,z)` to obtain the *reduced* Helmholtz free energy ($a_{\text{res.}} = A_{\text{res.}}/(Nk_\mathrm{B}T)$) for *any* cubic and three functions to give us r_1 and r_2, (e.g., `cubic_r(model::vdWModel)=(0.,0.)`), which will be used within `a_res`, for *each* cubic.

As a first step, let us write these `cubic_r` functions:

```
cubic_r (generic function with 3 methods)
```

```julia
begin
    cubic_r(model::vdWModel) = (0.,0.) # van der Waals
    cubic_r(model::RKModel) = (0.,1.) # Redlich-Kwong and Soave-Redlich-Kwong
    cubic_r(model::PRModel) = (1+sqrt(2),1-sqrt(2)) # Peng-Robinson
end
```

Before moving on to writing the `a_res` function, there are a few key details to remember:

Firstly, as we are going to be using automatic differentiation, we need to make sure all of our variables are explicitly defined. This may seem obvious since variables like the volume and temperature are quite clearly laid out. However, for equilibrium calculations, where we require the chemical potential, the composition becomes very important. Often, in literature, as one mole is usually assumed, authors often forget to write out explicitly the composition dependence. We have ensured that, wherever a composition dependence is present, it has been written out explicitly.

For consistency, we will assume a and b are in molar units, meaning our generalised equation becomes:

$$A_{\text{res.}}=-n\bar{R}T\log\left(1-nb/V\right)-\frac{na}{b}\frac{\log(V+r_2nb)-\log(V+r_1nb)}{r_2-r_1}$$

where, for multi-component systems:

$$n=\sum_i z_i$$

The above is true when implementing any equation of state.

Secondly, in deriving our generalised equation, we neglected one case: where $r_1=r_2=0$ for vdW. Although there are some cubics where $r_1=r_2\neq 0$, for now, we will only consider vdW. Integrating again gives:

$$A_{\text{res.}}=-n\bar{R}T\log\left(1-nb/V\right)-\frac{n^2a}{V}$$

With all this in mind, we are ready to implement our own generalised cubic equation of state (remember, we are trying to obtain the reduced, residual Helmholtz free energy):

```
a_res (generic function with 1 method)
```

```julia
function a_res(model::CubicModel,V,T,z)
    n = sum(z)

    # cubic_ab will obtain a and b for any cubic equation of state
    aα,b,c = cubic_ab(model, V, T, z) # ignore c for now

    r1,r2 = cubic_r(model)

    a1 = -log(1-n*b/V)

    if r1==r2
        a2 = -n*aα/(V*R̄*T)
    else
        a2 = -aα/(b*R̄*T)*(log(V+r2*n*b)-log(V+r1*n*b))/(r2-r1)
    end
    return a1+a2
end
```

Got it!

You got the right answer!

The above is a good example of the benefits of multiple dispatch in Julia. We wrote one function which should take in any cubic model (or inputs of type `CubicModel`) and only defined the parameters which are equation-specific. This means that, if we wanted to implement another cubic equation of state in the future, all we would need to add is a new `cubic_r` function.

Nevertheless, with this equation defined, we will be able to obtain any properties of interest! How we do this is the topic of the next part of the course.

Section 2.3.3

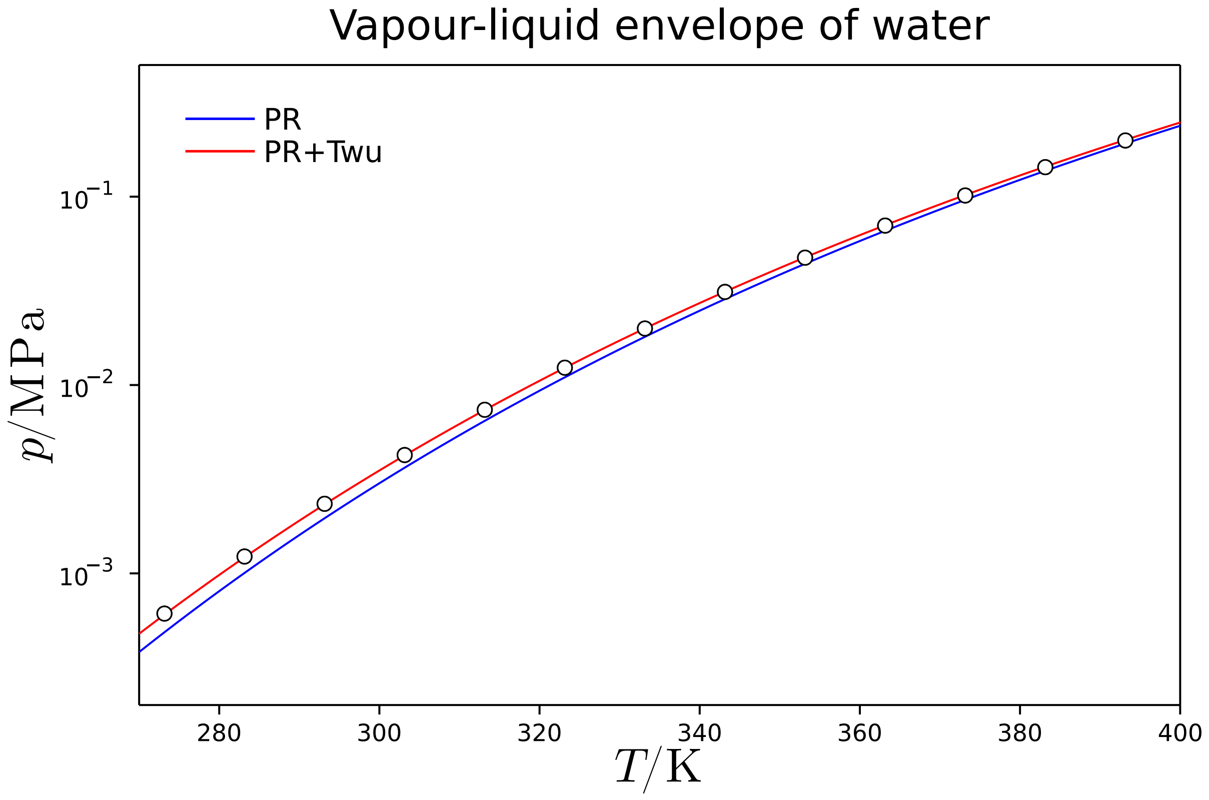
α functions

Now that we have an understanding of the advantages and disadvantages of the SRK and PR equation of state, let us go about fixing some of their failings. As we've already mentioned, the α function, introduced by Soave, greatly improved the ability to model the saturation curve of hydrocarbon / natural gas species. However, as we showed with water, this improvement is not universal. To remedy this, numerous authors have developed new α functions. To name a few and the reason for their existence:

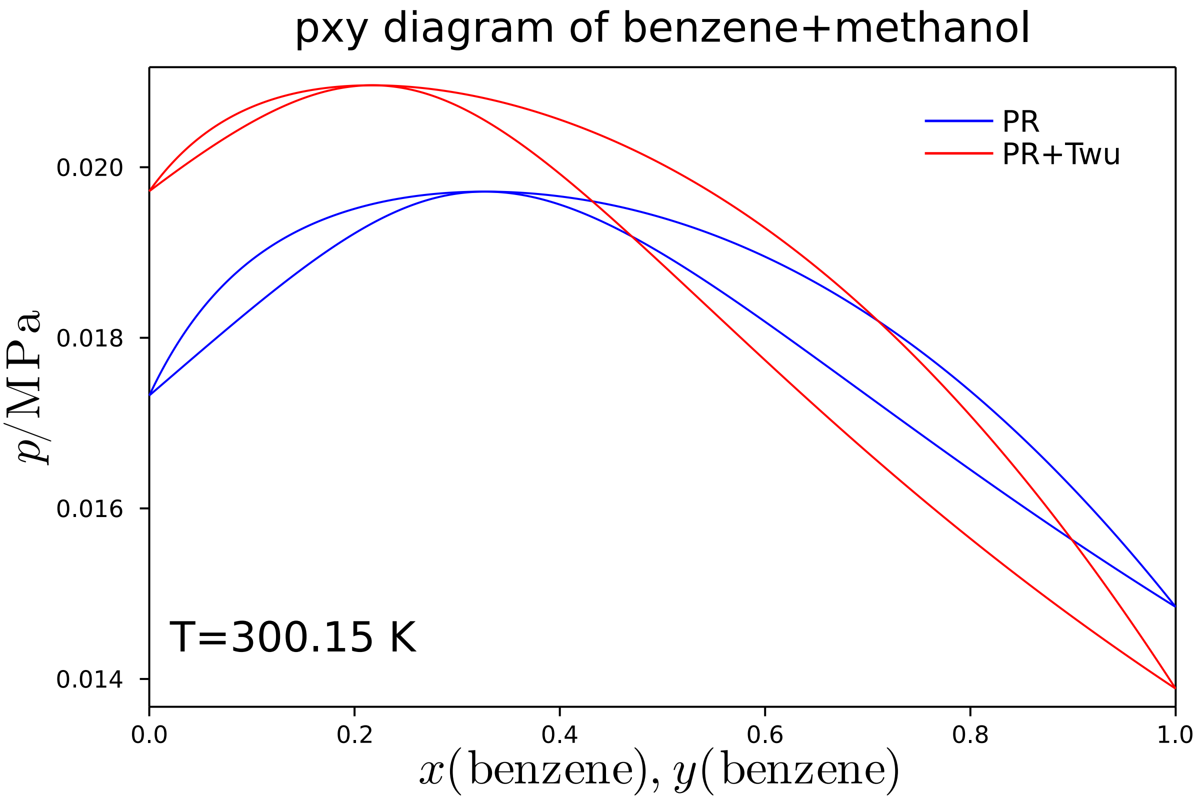
- PR-78: Just two years after they initially published their equation of state Peng and Robinson re-parameterised their α function. This version is more accurate than the original for a greater range of species.
- Boston-Matthias: Below the critical point, this α function is the same as the standard SRK and PR α function. The change Boston and Matthias made was for how the α function behaves above the critical point. We will illustrate the impact of this shortly.
- Magoulas-Tassios: A re-parameterised version of the α function using more parameters, intended to be used with a modified PR equation.
- Twu *et al.*: An α function with species-specific parameters. As the α function is now fitted for a species directly, this is by far the most accurate α function available. Intended to be used with PR.

As an example, let us compare the default PR and Twu α-functions (feel free to switch out the α function):

```
mw = PR(["water"];alpha=TwuAlpha);
```

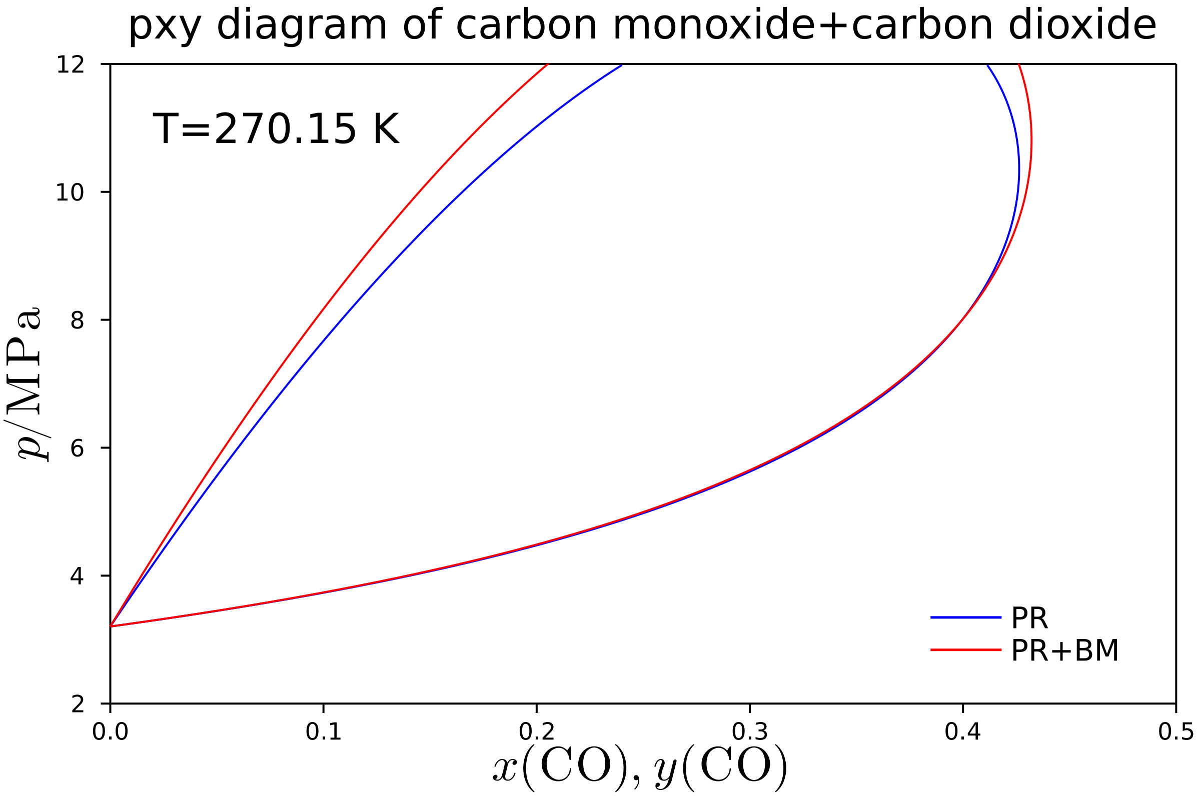


The impact of the α function is a bit more-subtle than just the saturation pressure. It can also have a large impact of vapour-liquid equilibrium properties of mixtures. Although we haven't covered how mixtures are handled within cubics, for the time being, we will only look at the impact of the α-function from a high-level perspect:



Without considering experimental data, it is clear that the α function can have a profound impact on mixture properties. The logic behind this is simple: the saturation pressure of the pure components is the end points for VLE envelope of the mixutres. If you can't get those right, then you shouldn't expect to get the points inbetween correctly.

However, the impact can go beyond this. Even when one of the components is supercritical, the α function selected still matters. Consider a mixture of carbon dioxide and carbon monoxide at a temperature where carbon monoxide is supercritical:



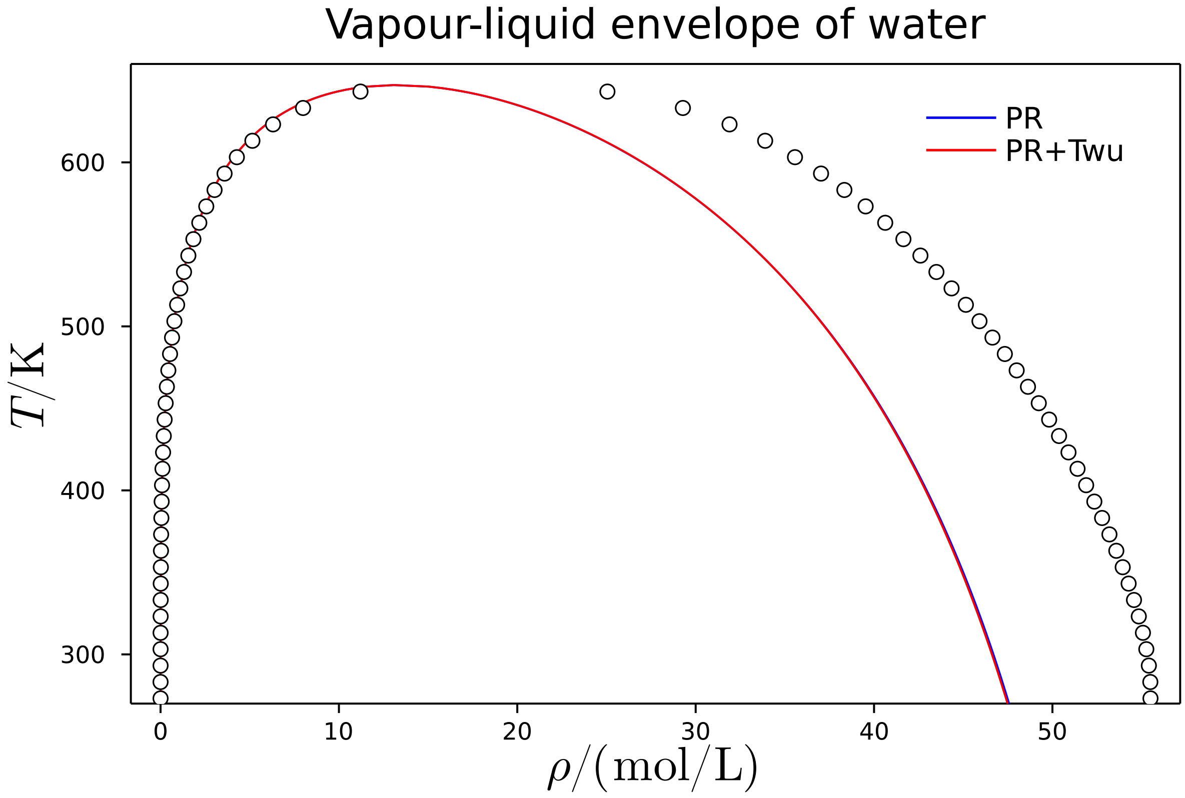
The reason for this, and why the Boston-Matthias α function should be used whenever one component is supercritical is that, the default α functions in PR and SRK both behave unphysically at temperature above the critical point ($T > 3T_c$) which is a problem when mixtures contain species, like carbon monoxide, which have low critical points.

Overall, when considering which cubic equation of state to use, which α function to use is an important question to ask. For most hydrocarbon systems, the standard SRK and PR equations should be sufficient. However, for more-complex systems, it is worth considering not only the pure saturation curves, but the mixed systems as well. In general, the safest would be to use species-specific α functions like the one developed by Twu *et al*. although the parameters may not be available for every species.

Section 2.3.4

Volume translation

We previously spent a lot of time considering the impact of the α function on the saturation curve of our species. However, what about the liquid densities? Something we know cubics struggle with for complex species like water. In short, nothing much changes:

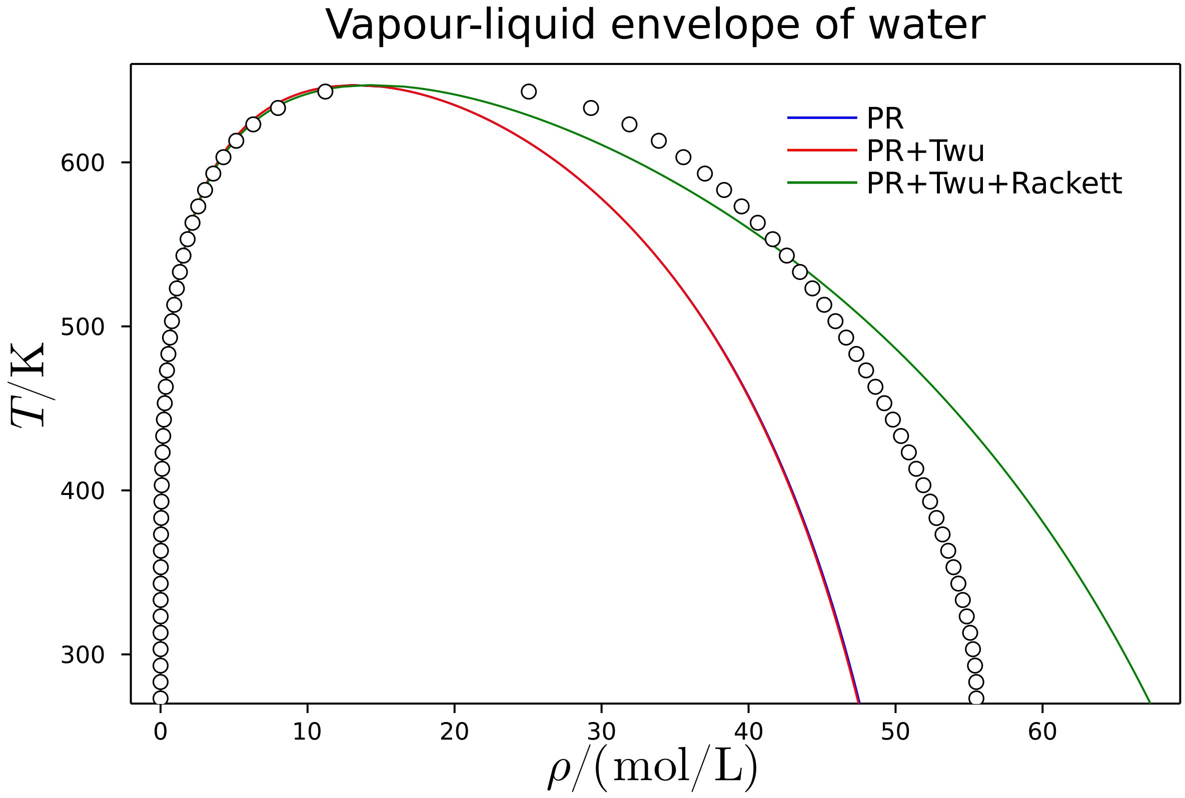


This is not ideal as, in some scenarios, we need accurate liquid densities. One very simple correction that has a minimal impact on our original equation is to introduce a volume translation. In this case, The `true` volume and the volume fed into our equation is shift slightly:

$$V_{\mathrm{eos}} = V - Nc$$

where c is our shift. The benefit of this approach is that it will not impact our saturation curve as c is just a constant; it will only improve our predictions for the liquid densities. For example, for PR, the Rackett equation provides the shift (for SRK, the Péneloux equation can be used):

```
mw4 = PR(["water"];alpha=TwuAlpha,
                   translation=RackettTranslation);
```



Although still not quite ideal for water, it is an improvement over the untranslated results.

One this to consider is the impact on our generalised equation for the cubics. Introducing the shift gives us a slightly different equation:

$$A_{\text{res.}} = -n\bar{R}T\log\left(1-n(c-b)/V\right) - \frac{na}{b}\frac{\log(V+n(c+r_2b))-\log(V+n(c+r_1b))}{r_2-r_1}$$

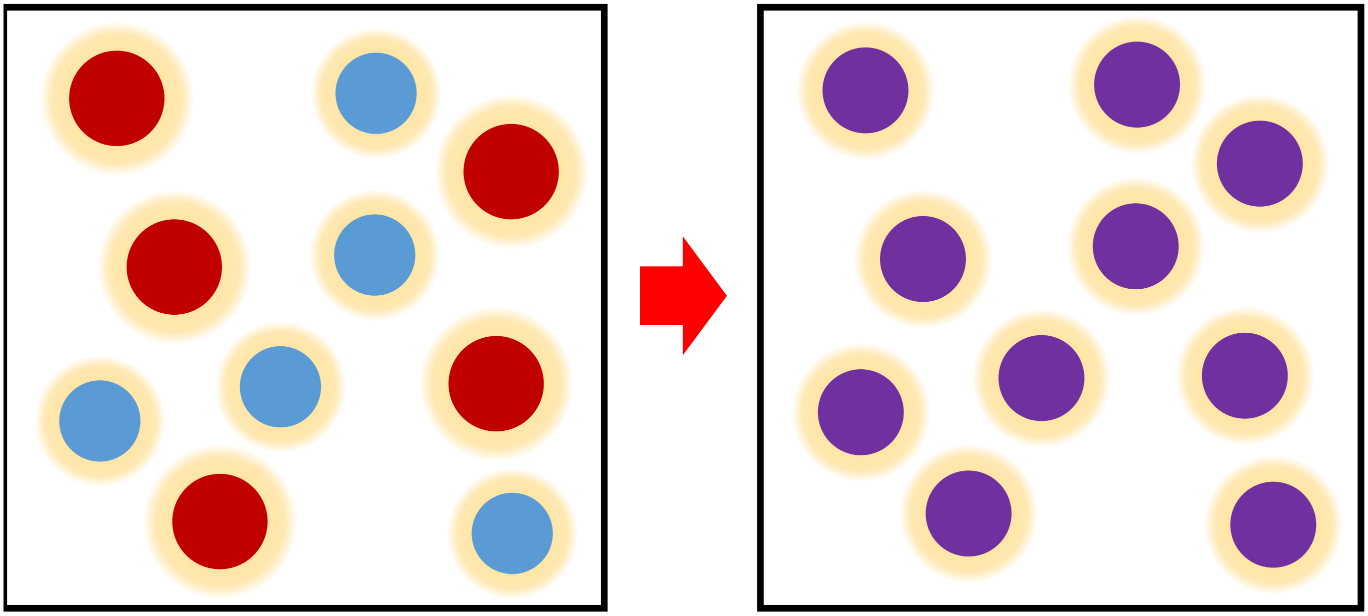
Nevertheless, using the code we previously wrote, incorporating this shift should be quite straightforward.

Generally, volume translations should only be used when we need accurate volumetric properties. If this is not the case, then one can afford to ignore the translation.

Section 2.3.5

Mixing rules

Now that we have established all the tools needed to model pure systems using cubics, we now need to consider extending them to model mixtures. Typically, we want a set of a and b parameters that characterise the mixture (we will denote this `one-fluid mixture' parameters as $\bar{a}$ and $\bar{b}$):



How can we do this? The critical points for the mixtures are generally not known, thus, using the usual definition for a and b is not an option.

Van der Waals one-fluid mixing rule

The simplest and most-widely used approximation for obtaining $\bar{a}$ and $\bar{b}$ is the van der Waals one-fluid mixing rule:

$$\bar{a} = \sum_i \sum_j x_i x_j a_{ij}$$

$$\bar{b} = \sum_i \sum_j x_i x_j b_{ij}$$

where, if $i = j$, then a_{ii} and b_{ii} are the usual parameters for a pure fluid i. If $i \neq j$, then a_{ij} and b_{ij} can be obtained using:

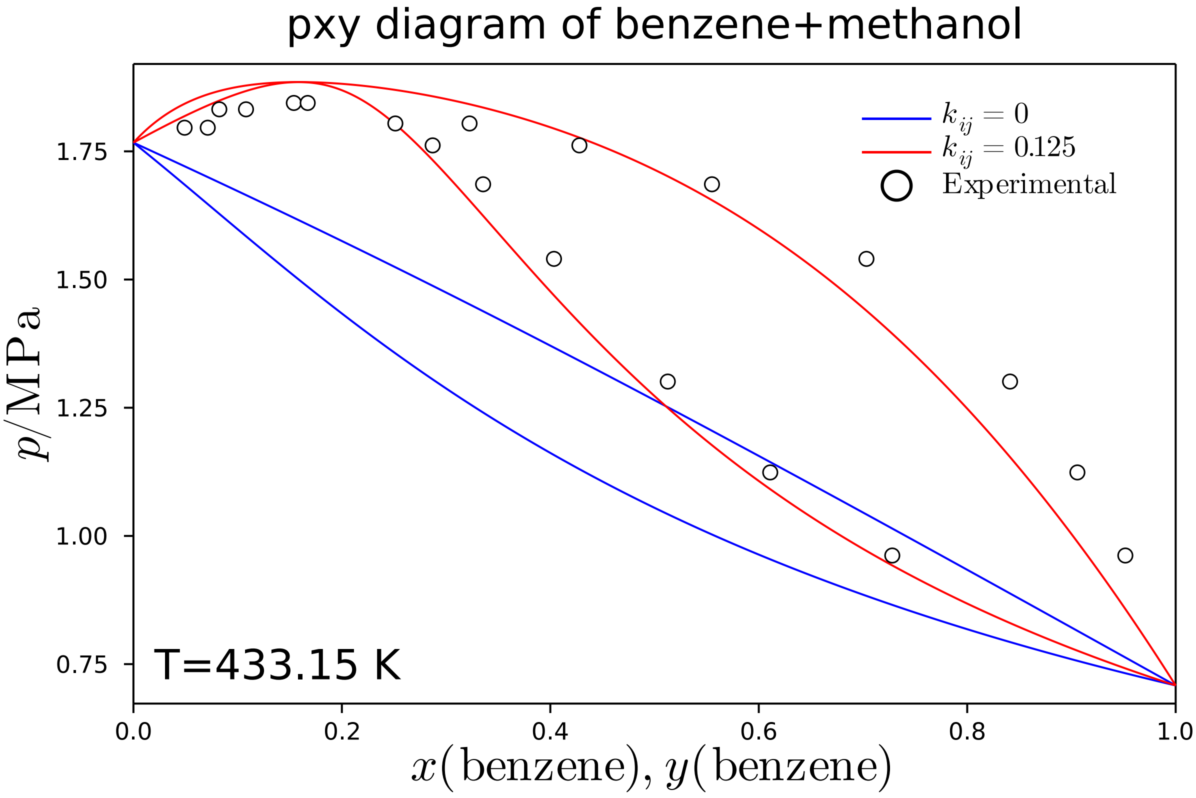
$$a_{ij} = \sqrt{a_{ii} a_{jj}}(1 - k_{ij})$$

$$b_{ij} = \frac{b_{ii} + b_{jj}}{2}(1 - l_{ij})$$

The above are known as combining rules, and k_{ij} and l_{ij} are known as binary interaction parameters. Typically, the van der Waals mixing rule will work well for mixtures of similar species (e.g. ethane+propane) but struggle with associating (e.g. water+ethanol) and/or size asymmetric (e.g. carbon dioxide+n-decane) mixtures. k_{ij} and l_{ij} are fitted against mixtures containing species i and j to account for these `non-ideal' interactions. Generally, $l_{ij} = 0$, meaning the mixing rule for $\bar{b}$ simplifies to:

$$\bar{b} = \sum_i x_i b_{ii}$$

and, in some case, k_{ij} is assigned a temperature dependence (typically $k_{ij} = k_{ij}^{(0)} + k_{ij}^{(1)} T$). We can see the impact of this parameter below:



Clearly, this binary interaction parameter can have a profound effect on the predicted equilibria for the mixture. We can see above that the equilibria goes from being almost ideal to having an azeotrope, as well as agreeing more-quantitatively with the experimental data.

There are other mixing rules similar to the van der Waals one-fluid mixing rule (e.g. Kay's rule, Rao's rule, etc.). However, all of these require binary interaction parameters to accurately model mixtures. These binary interactions can usually be found in literature, although tools like ASPEN and gPROMS have large databases available. If these are not available, it is recommended to simply fit these parameters to any available experimental data.

Naturally, there comes the limitation that we sometimes need to model systems for which there are no binary interaction parameters or experimental data available in literature.

EoS/G^E mixing rules

Having now seen the limitations of simple mixing rules in cubics, we now consider another class of mixing rules. In the previous section, we showed how effective activity coefficient-based models were for modelling equilibrium properties of mixture systems, despite being unable to model pure systems and limited to a few properties. What if we could 'borrow' this better modelling from the activity coefficent models, and use it within cubics?

The basic ideal behind G^E mixing rules is we set the excess Gibbs free energy obtained from the cubic equation of state to that obtained from activity models:

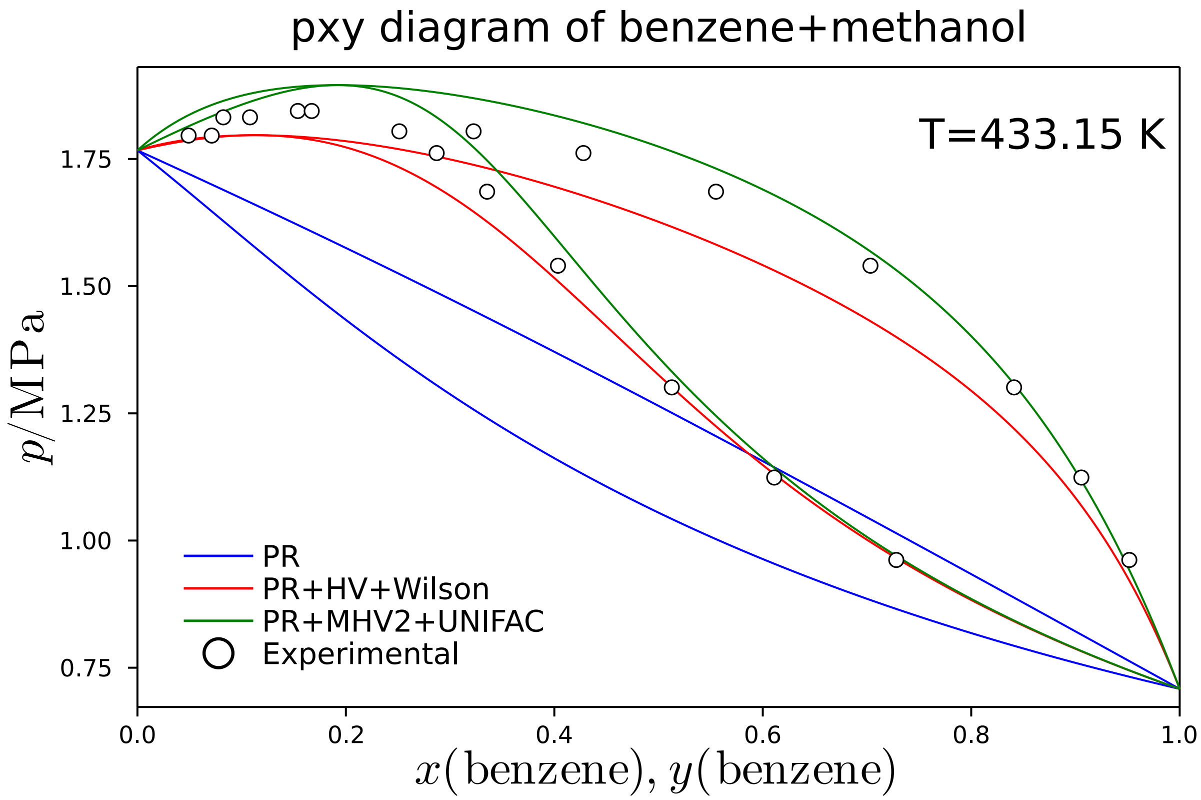
$$g^E_\text{cubic}(T,p,z)=g^E_\text{act.}(T,z)$$

The difficulty is that activity coefficient models are pressure-independent. Thus, at which pressure do we set this equality? This depends on which mixing rule we use. The first such mixing rule derived was by Huron and Vidal which took the infinite pressure limit, giving the following mixing rule:

$$\frac{\bar{a}}{\bar{b}}=\sum_i x_i\frac{a_i}{b_i}-\frac{G^E}{\lambda}$$

where G^E is obtained from the activity coefficient model and λ is equation of state-specific. Taking the opposite limit of zero pressure, the mixing rules of Michelsen and, Wong and Sandler are other alternatives (which are too complex to write here). The interesting aspect here is that there is no restriction as to which activity coefficient model can be used here (Wilson, NRTL, UNIQUAC, etc.). For example (feel free to switch out the mixing rule and activity model):

```
mix = PR(["benzene","methanol"];alpha=TwuAlpha,mixing=HVRule,activity=Wilson);
```



As we can see, the results obtained from these mixing rules are substantially better than those obtained using the simple van der Waals one-fluid mixing rule. However, the above also illustrates the big advantage of such a mixing rule. While Wilson, NRTL and UNIQUAC are all species-specific method, models like UNIFAC are group-contribution based. This means, as long as the groups for a species are available, we will be able to use this mixing rule to *predict* the mixture phase equilibria!

In terms of recommendations, if possible, it is always best to validate these mixing rules against experimental data. If binary interaction parameters have been fitted against experimental data, it is usually easier and more-trustworthy to use the simpler mixing rules. However, if one must use a G^E mixing rule, generally speaking, the Michelsen and, Wong-Sandler mixing rules are typically the most-reliable, coupled with any of the activity coefficient models (naturally, if one can use species-specific approaches, that is preferred).

Section 2.3.6

Predictive Cubics

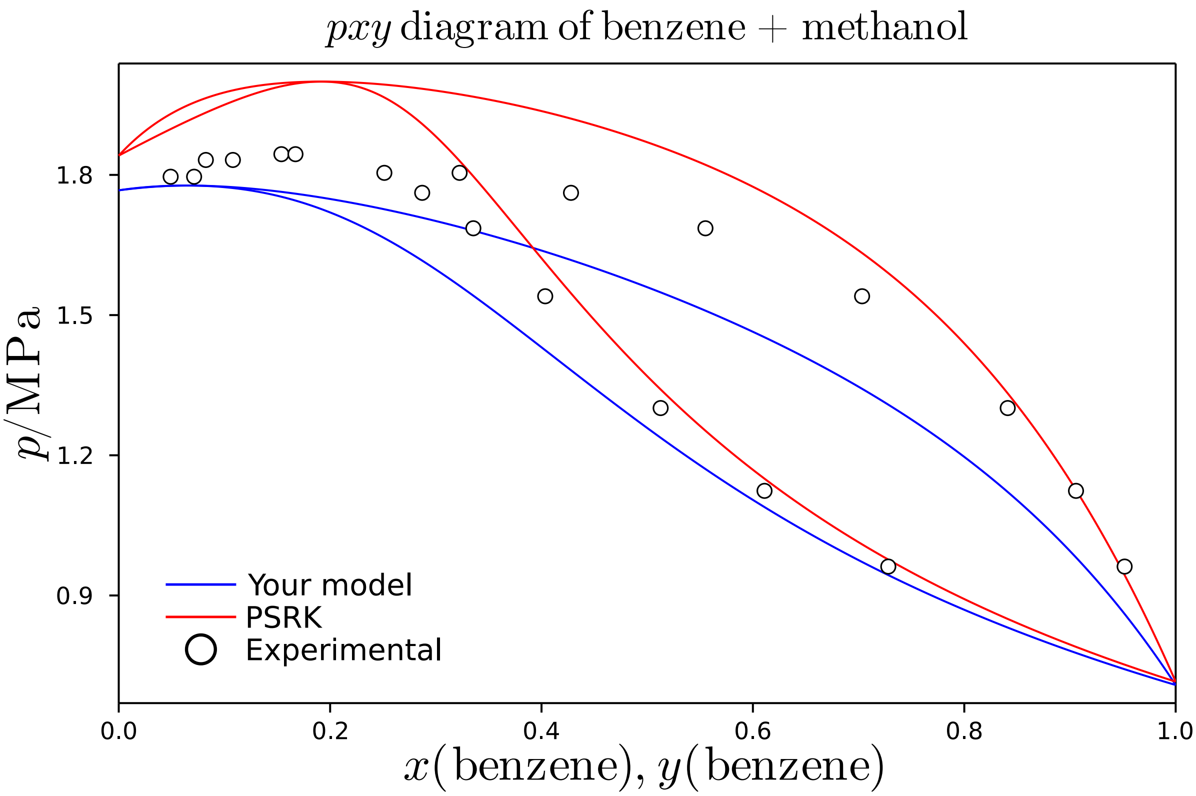
Now that we've gone through the different types of cubics, α functions, volume translation and mixing rules, it is time to bring them together:

```
m = PR(["benzene","methanol"];alpha=TwuAlpha,translation=RackettTranslation,
                               mixing=HVRule,activity=UNIFAC);
```

However, there are two cubics that make use of all the above methods to provide some of the most-accurate equations of state available:

- Predictive SRK: Combines the standard Soave α function, Peneloux volume translation, the Michelsen first order mixing rule and its own version of UNIFAC.
- Volume-translated PR: Combines Twu *et al.*'s α function, Rackett volume translation, a modified Huron-Vidal mixing rule and its own version of UNIFAC.

These equations of state are, for most mixtures of interest in industry, almost as accurate as the high-accuracy empirical models (GERG-2008). Both of these approaches are predictive for mixtures, only requiring the critical point to be used. If one can use either of these methods, they are highly recommended. Try to create your own cubic equation of state to see if it rivals their accuracy!



```
almost (generic function with 1 method)

still_missing (generic function with 2 methods)

correct (generic function with 2 methods)

not_defined (generic function with 1 method)

yays =
  [Fantastic!, Splendid!, Great!, Yay ❤, Great! 🎉, Well done!, Keep it up!, Good job!, Awe
```