SIS 10.1-7

AUTHOR
k.wodehouse

PUBLISHED
March 6, 2025

I'll start by bringing in the data given in the question as a dataframe because thats super easy to work with. I grabbed the molecular weight from NIST.

```
import numpy as np
import pandas as pd
from scipy.optimize import fsolve
from scipy.constants import R

df = pd.read_csv('data.txt', sep='\t', index_col=0)
components =  list(df.index)
df['pvap'] *= 1e5
df
```

	zif	pvap	density	mw
compound				
n-butane	0.1	242800.0	0.575	58.1222
n-hexane	0.9	20000.0	0.655	86.1754

I think what we need to do is

1. guess a pressure
2. calculate V and L
3. calculate the volume of liquid in the tank
4. calculate the pressure for the vapor using ideal EOS
5. if it matches the guess, thats the answer
6. otherwise guess another pressure

```
volume = 30.0 / 1000 # liters -> m^3
T = 298.0

def get_v(V, Ptotal):
    sum_thingy = 0
    for component in components:
        zif, pvap = df.loc[component]['zif'], df.loc[component]['pvap']
        ki = pvap / Ptotal
        sum_thingy += (zif )/(1 + V*(ki - 1))
    return (1 - sum_thingy)

def flash(Ptotal):
    V = fsolve(get_v, 0.5, (Ptotal), maxfev=9999)[0]
    L = 1 - V

    liquid_volume = 0
    for component in components:
        zif, pvap, mw, density = df.loc[component]['zif'], df.loc[component]['pvap'], df.
        ki = pvap / Ptotal
        xi = zif / (1 + V*(ki - 1))
        liquid_volume += L * xi * mw / density / 1e6
    vapor_volume = volume - liquid_volume

    pressure = 0
    for component in components:
        zif, pvap, mw = df.loc[component]['zif'], df.loc[component]['pvap'], df.loc[compo
        ki = pvap / Ptotal
        yi = zif * ki/ (1 + V*(ki - 1))
        N = V * yi
        pressure += N * R * T / vapor_volume
    return abs(Ptotal - pressure)

pressure = fsolve(flash, 1e5, maxfev=99999)[0]
V = fsolve(get_v, 0.9, (pressure))[0]
print(f'pressure: {pressure*1e-5:.4f} bar')

pressure_ig = R*T/volume
print(f'P (ideal gas in 30L): {pressure_ig*1e-5:.4f} bar')
```

```
pressure: 0.2616 bar
P (ideal gas in 30L): 0.8259 bar

/var/folders/f9/38nkw6517lgf362dv55hy0xc0000gn/T/ipykernel_28164/2401709798.py:13:
RuntimeWarning: The iteration is not making good progress, as measured by the
  improvement from the last ten iterations.
  V = fsolve(get_v, 0.5, (Ptotal), maxfev=9999)[0]
```

realistically, the liquid volume is probably negligible. let's get the compositions now

```
print('----- liquid composition -----')
xis = []
for component in components:
        zif, pvap = df.loc[component]['zif'], df.loc[component]['pvap']
        ki = pvap / pressure
        xi = zif / (1 + V*(ki - 1))
        xis.append(xi)
        print(f'{component}: {xi:.3f}')

yis = []
print('----- vapor composition -----')
for component in components:
        zif, pvap = df.loc[component]['zif'], df.loc[component]['pvap']
        ki = pvap / pressure
        yi = zif*ki / (1 + V*(ki - 1))
        yis.append(yi)
        print(f'{component}: {yi:.3f}')

results = pd.DataFrame({'component':components, 'x':xis, 'y':yis}).set_index('component')
```

```
----- liquid composition -----
n-butane: 0.028
n-hexane: 0.972
----- vapor composition -----
n-butane: 0.257
n-hexane: 0.743
```

now we can check these numbers to make sure they're right.

- we know x and y should sum to 1
- we know our calculated pressure should be $\sum x_i P_i^{\text{vap}}$
- our liquid and vapor volumes must add up to 30 liters

```
sumy = results['y'].sum()
sumx = results['x'].sum()
print(f'sum y: {sumy:.4f}\nsum x: {sumx:.4f}\n')

calcp = np.dot(results['x'], df['pvap'])
print(f'pressure: {calcp*1e-5:.4f} bar\n')

liquid_volume = 0
L = 1 - V
for component in components:
    zif, pvap, mw, density = df.loc[component]['zif'], df.loc[component]['pvap'], df.loc[
    ki = pvap / pressure
    xi = zif / (1 + V*(ki - 1))
    liquid_volume += L * xi * mw / density / 1e6
print(f'liquid volume: {liquid_volume*1e3:.4f} L')

vapor_volume = 0
for component in components:
    zif, pvap, mw, density = df.loc[component]['zif'], df.loc[component]['pvap'], df.loc[
    ki = pvap / pressure
    yi = zif * ki/ (1 + V*(ki - 1))
    vapor_volume += V * yi * R * T / pressure
print(f'vapor volume: {vapor_volume*1e3:.4f} L')

print(f'volume sum: {(liquid_volume + vapor_volume)*1e3:.2f}')
```

```
sum y: 1.0000
sum x: 1.0000

pressure: 0.2616 bar

liquid volume: 0.0894 L
vapor volume: 29.9106 L
volume sum: 30.00
```

boom! they check out.

```
# filler text
```