TOMSK POLYTECHNIC UNIVERSITY
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский Томский политехнический университет» (ТПУ)

Инженерная школа природных ресурсов
Направление подготовки Химическая технология
Отделение химической инженерии

PYTHON ДЛЯ ЗАДАЧ ХИМИЧЕСКОЙ ТЕХНОЛОГИИ

Отчет по лабораторной работе № 2

Выполнил студент гр. 9дм21 ____________________ Шуриков М К

(Подпись)

_____ ______________ 2023 г.

Отчет принят:

Преподаватель
доцент ОХИ ИШПР, к.т.н. __________________________ В.А. Чузлов

(Подпись)

_____ ______________ 2023 г.

Томск 2023 г.

Задание 1

Формула нормализованной гауссовой функции со средним значением μ и стандартным отклонением σ:

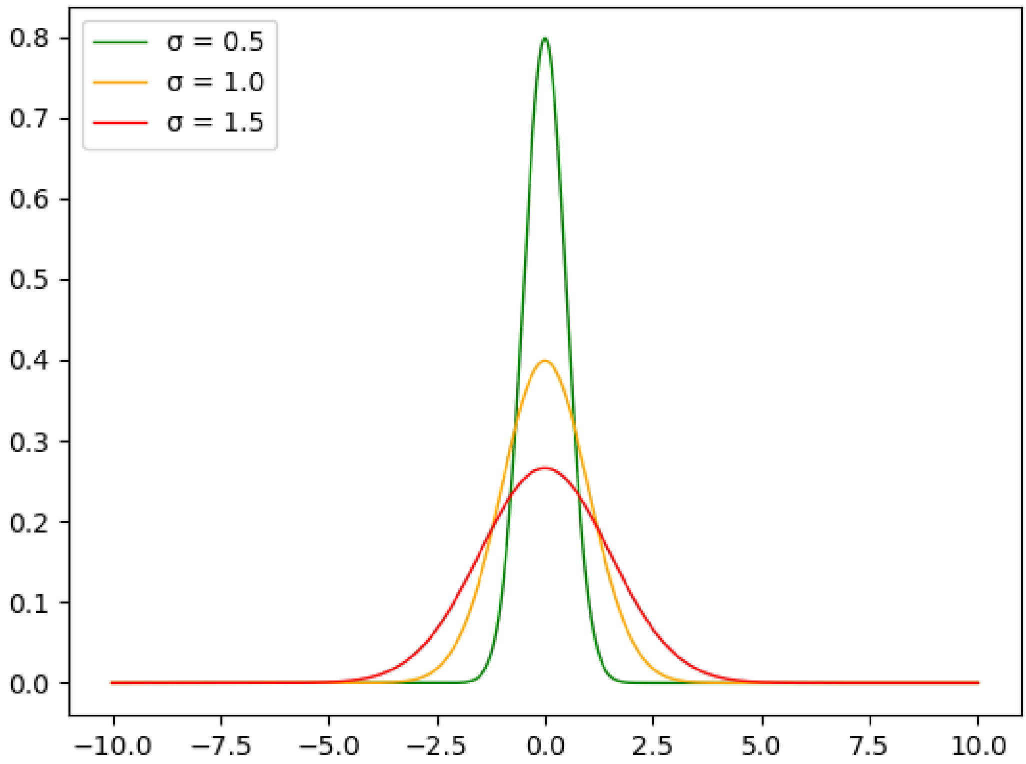
$$g(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

Необходимо написать функцию, основанную на использовании массивов NumPy для вычисления гауссовых функций при μ = 0 и σ 2 = 0.5; 1.0; 1.5. Использовать сетку из 1000 точек в интервале −10 ⩽ x ⩽ 10. Постройте графики данных функций.

Программная реализация:

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import norm

#x-axis ranges from -3 and 3 with .001 steps
x = np.arange (-10, 10, 0.001)

#plot normal distribution with mean 0 and standard deviation 0.5
plt.plot (x, norm. pdf (x, 0, 0.5), 'g', linewidth = 1, label='σ = 0.5')

plt.plot (x, norm. pdf (x, 0, 1.0), 'orange', linewidth = 1, label = 'σ = 1.0')

plt.plot (x, norm. pdf (x, 0, 1.5), 'r', linewidth = 1, label = 'σ = 1.5')

plt.legend(loc='upper left')

plt.show()
```

Ответ:



Задание 2

Уравнение Ван дер Ваальса, описывающее состояние газа, можно записать в виде следующей формулы как зависимость давления p газа от его молярного объема V и температуры T:

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

где a и b – специальные молекулярные константы, а R = 8.314 Дж / К·моль – универсальная газовая константа.

Формулу легко преобразовать для вычисления температуры по заданному давлению и объему, но ее форма, представляющая молярный объем в отношении к давлению и температуре, является кубическим уравнением:

$$pV^3 - (pb + RT)V^2 + aV - ab = 0$$

Все три корня этого уравнения ниже критической точки (T_c, p_c) являются действительными: наибольший и наименьший соответствуют молярному объему газообразной фазы и жидкой фазы соответственно. Выше критической точки, где не существует жидкая фаза, только один корень является действительным и соответствует молярному объему газа (в этой области его также называют сверхкритической жидкостью, или сверхкритической средой).

Критическая точка определяется по условию $(\partial p/\partial V)_T = (\partial^2 p/\partial V^2)_T = 0$ и для идеального газа Ван дер Ваальса выводятся формулы:

$$T_c = \frac{8a}{27Rb} \qquad p_c = \frac{a}{27b^2}$$

Для NH_3 константы Ван дер Ваальса a = 0.4225 л^2 · Па · м^6 · моль$^{-2}$ и b = 37.07 × 10^{-6} м^3 · моль$^{-1}$.

- Найти критическую точку для аммиака, затем определить молярный объем при комнатной температуре и давлении (298 К, 1 атм) и при следующих условиях (500 К, 12 МПа).
- Изотерма – это множество точек (p, V) при постоянной температуре, соответствующее уравнению состояния газа. Построить изотерму (p в зависимости от V) для аммиака при температуре 350 К, используя уравнение Ван дер Ваальса, и сравнить ее с изотермой при температуре 350 К для идеального газа, уравнение состояния которого имеет вид p = RT/V (принять значения p принадлежащими интервалу [101325; 1000000] Па, 1000 элементов).

Программная реализация:

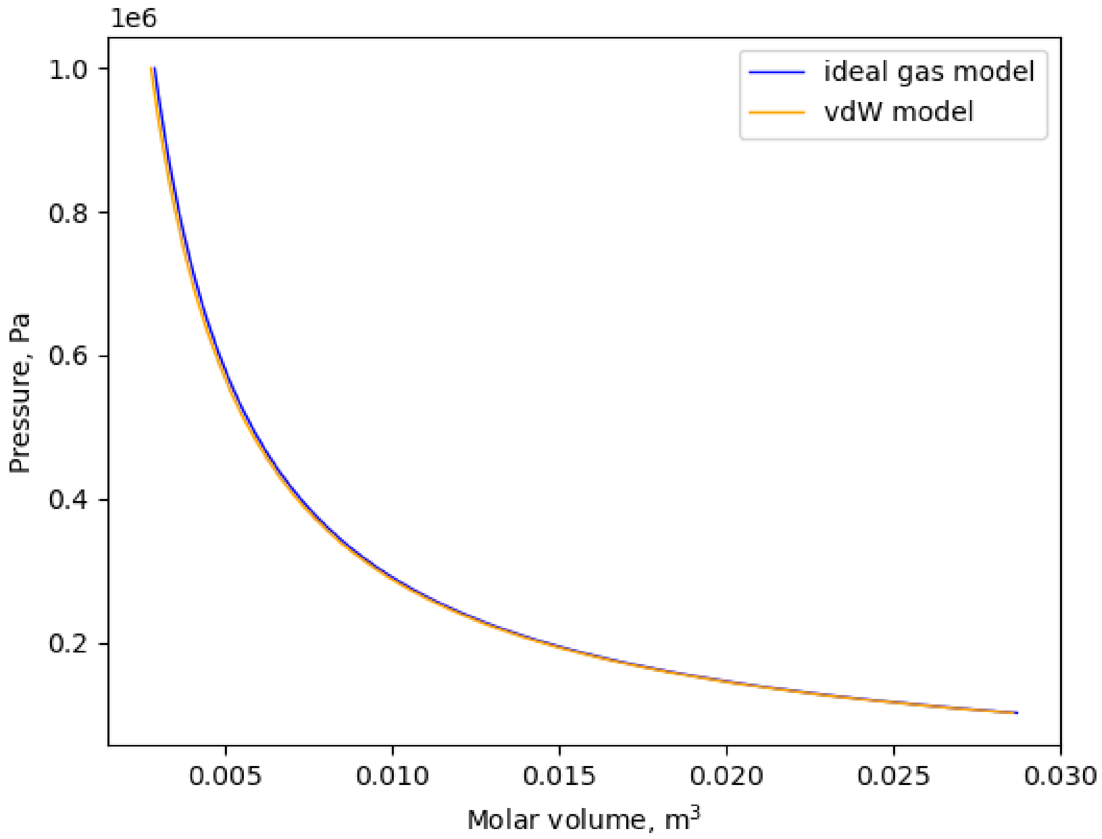
```
import numpy as np
from numpy.polynomial import Polynomial
import matplotlib.pyplot as plt

# critical point
a = 0.4225
b = 37.07e-6
Tc = (8 * a) / (27 * 8.31 * b)
Pc = a / (27 * b ** 2)
print (f'Critical point is Tc = {Tc:6.2f} K; Pc = {Pc:6.2f} Pa')
print()
```

```python
# mol volume calc
tlist = [298, 500] # K
plist = [1e5, 12e6] # Pa

def molar_volume(
        plist: list[int],
        tlist: list[int],
) -> list[float]:
    molar_v = [Polynomial([-a * b, a, -(p * b + 8.31 * t), p]) for t, p in
zip(tlist, plist)] #polynomial coefficients in here are reversed
    return molar_v

mv = molar_volume(tlist = tlist, plist = plist)

for i in mv:
    r = i.roots()
    r = np.real(r[~np.iscomplex(r)])
    print(np.max(r))
    print()

# isotherm
p = np.linspace(101325, 1000000, 1000)

T = 350

def molar_volume_vdW(
    p: float,
) -> np.polynomial.polynomial:
    x = Polynomial([-a * b, a, -(p * b + 8.31 * T), p])

    roots = x.roots()
    roots = roots[~np.iscomplex(roots)].real
    max = np.max(roots)

    return max

v = np.zeros_like(p)

for i, pi in enumerate(p):
    poly = molar_volume_vdW(pi)
    v[i] = poly

v_mk = 8.31 * T / p

fig, ax = plt.subplots()
ax.plot(v_mk,p, 'b', linewidth = 1, label = 'ideal gas model')
ax.plot(v,p, 'orange', linewidth = 1, label = 'vdW model')

ax.set_xlabel(r'Molar volume, m$^3$')
ax.set_ylabel('Pressure, Pa')
```

```
plt.legend(loc='best')
plt.show()
```

Ответ:

```
Critical point is Tc = 406.38 K; Pc = 11387221.73 Pa

0.024629586533868653

0.0002712632907744113
```



Задание 3

Закон Бугера–Ламберта–Бера связывает концентрацию вещества с в образце раствора с интенсивностью света, проходящего через этот образец It с заданной толщиной слоя вещества l при известной длине волны λ:

$$I_t = I_0 e^{-\alpha c l}$$

где I_0 — интенсивность света на входе в вещество, α — коэффициент поглощения при длине волны λ.

После проведения ряда измерений, позволяющих определить часть света, которая прошла сквозь раствор, It/I0, коэффициент поглощения α можно при помощи линейной аппроксимации:

$$y = \ln\left(\frac{I_t}{I_0}\right) = -\alpha c l$$

Несмотря на то что эта прямая проходит через начало координат (y = 0 при c = 0), мы будем выполнять подгонку для более общего линейного отношения:

$$y = mc + k$$

где m = −αl с проверкой k на приближение к нулю.

При рассмотрении образца раствора с толщиной слоя 0.8 см при измерениях были получены данные, приведенные в таблице: отношение I_t/I_0 при пяти различных концентрациях:

C, моль/л	I_t/I_0
0.4	0.891
0.6	0.841
0.8	0.783
1.0	0.744
1.2	0.692

Используя линейную аппроксимацию, определите коэффициент α.

Программная реализация:

```
import numpy as np
import matplotlib.pyplot as plt

C = [.4, .6, .8, 1.0, 1.2]
ItI0 = [0.891, 0.841, 0.783, 0.744, 0.692]
log = np.log(ItI0)
l = 0.8

fig, ax = plt.subplots()
ax.plot(C, log, 'orange', linewidth = 1, label = 'empirical')

from scipy.stats import linregress
```

```
slope, intercept, r_value, p_value, std_err = linregress(C, log)

approx = [i * slope + intercept for i in C]
ax.plot(C, approx, 'darkblue', linewidth = 1, label = 'approx')

font = {'family': 'sans',
        'color':  'darkblue',
        'weight': 'normal',
        'size': 16,
        }
plt.text(0.45, -0.35, f'y = {slope:6.4f} · x + {intercept:6.4f}', fontdict=font)
plt.legend(loc='upper right')
plt.xlabel("C")
plt.ylabel("log(It/I0)")
plt.show()
```

Ответ: