C4: hypothesis testing - assignment

February 6, 2025

1 PHAS0029 Session 4: Hypothesis testing

Updated: 28/01/2025

This notebook contains your tasks for Session 4. Please make sure you go through the theory notebook before attempting the tasks here.

If you cannot see the green boxes around task instructions, make sure you click on 'Not trusted' on top left of the notebook.

Remember to use text cells to describe your reasoning and results, and comments to annotate the code. You can cut and paste code, equations and images from the theory notebook if you want, as long as your notebook makes it clear where you have pasted material and where it came from.

All headings have been removed from this notebook to give you a chance to structure it yourself. This should be a good preparation for your final assignment where you will be creating a notebook from scratch. You may remove this text if you wish.

2 1. Test significance of n-sized sample (Student's t distribution)

[3]:
```
#usual imports
import numpy as np
import matplotlib.pyplot as plt
import scipy.stats as sci
```

2.0.1 Task 1

In this part we test a hypothesis that a n-sized sample is a statistically significant sample drawn from a normal population. The data saved in the file hypothesis1.txt contains six different measurements of molecule velocities (x components only) in a gas.

Briefly explain which distribution you expect the data to follow, including the mean value.

load the data and plot it on a histogram

comment on the shape of the histogram: does it resemble your expected distribution?

I expect the data to follow the Maxwell Boltzmann distribution, which for lower speeds has a sharp peak and a long tail to the left. The mean would be to the right of the peak. The histogram below follows the expected result.

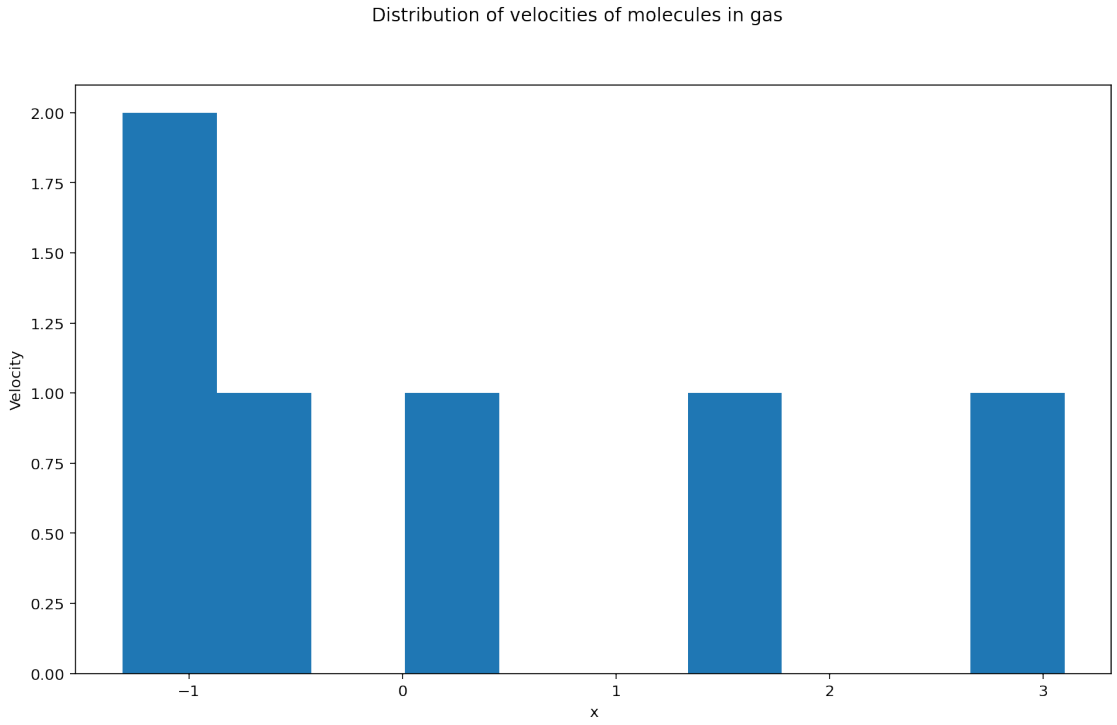
[4]:
```
#loading data
velocity = np.loadtxt('hypothesis1.txt',delimiter=",",unpack=True)
print(velocity)

#plotting data onto histogram
#Some of the code was taken and adapted from 'W3 schools'
plt.figure()
plt.hist(velocity)
plt.xlabel("x")
plt.ylabel("Velocity")
title_label=('Distribution of velocities of molecules in gas')
plt.suptitle(title_label)
```

```
[-1.3116076  -0.59263884 -1.19837675  0.1815155   3.10168358  1.38371709]
```

[4]:
```
Text(0.5, 0.98, 'Distribution of velocities of molecules in gas')
```

[4]:



2.0.2 Task 2

Write a function called ‘tstat which takes a one-dimensional array of data as an input and returns the Student’s t statistics. Do not use the scipy package at this point, you can find appropriate equations in the theory notebook or your lecture notes from PHAS0028.

[10]:
```
#some code was taken and adapted using ai

def tstat(velocity):
    """perform a t-test to an array of values"""
    n = len(velocity)
    mean = np.mean(velocity)
    sd = np.sqrt(np.sum((velocity - mean) ** 2) / (n-1))
    se = sd/np.sqrt(n)
    t = abs(mean)/se
    return t
```

[11]:
```
%run -i c6-checkpoint1.py
```

```
************
 Well done! All test passed. You can move to the next part of the task.
************
```

2.0.3 Task 3

Write a function called p_value which returns p-value for Student's t distribution given t statistics, no. of degrees of freedom and number of tails as inputs. You may use the SciPy package for calculation of the integrals. Make sure your function only works for 1 or 2 tails and returns a value error if any other input is given. HINT: you may see how to raise an error in a number of online tutorials, e.g. <a https://victoria.dev/blog/do-i-raise-or-return-errors-in-python/>here).

[7]:
```
#some code was taken and adapted using ai

def p_value(t,k,tails=2):
    """Returns a probabillity value for the t-test"""
    if tails not in (1,2):
        raise ValueError('Number of tails must be 1 or 2')

    p=sci.t.sf(abs(t),k)
    if tails == 2:
        p*= 2


    return p
```

[8]:
```
%run -i c6-checkpoint2.py
```

```
************
 Well done! All test passed. You can move to the next part of the task.
************
```

2.0.4 Task 5

Calculate p-value for the loaded data and plot the probabiity distribution clearly showing area used for calculation in p-value. What is the conclusion of the test? HINT: use the fill_between method from matplotlib to shade the correct areas. It takes three arguments/boundaries of the shaded area: array of x values, array containing the corresponding gaussian values and bottom border (0).

[0]:
```
```

3 2. Test if sample matches distribution (Chi squared distribution for Poissonian data)

3.0.1 Task 1

Load the data in file hypothesis2.txt into a variable named obs_list. Calculate the sample mean m_obs and the expected occurrences (probabilities) for that mean: exp_obs. Hint 1: You will need to create a dummy variable holding the number of radioactive counts. Hint 2: Use the scipy.stats.poisson.pmf method to calculate probabilities of a Poisson distribution.

[9]:
```
# YOUR CODE HERE
obs_list
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[9], line 2
      1 # YOUR CODE HERE
----> 2 obs_list

NameError: name 'obs_list' is not defined
```

[0]:
```
%run -i c6-checkpoint3.py
```

3.0.2 Task 2

What can you say about the accuracy of the last bin? Calculate the correct value and replace it in the exp_obs. Hint: Is it possible to have more than 17 counts? What should be the sum of exp_obs if the model was fully correct?

YOUR ANSWER HERE

[0]:
```
# YOUR CODE HERE
```

[0]:
```
%run -i c6-checkpoint4.py
```

3.0.3 Task 3

Plot the expected occurrences with their errors and observed occurrences on the same figure.

[0]:
```
# YOUR CODE HERE
```

3.0.4 Task 4

Considering the properties of Poisson distribution, explain why it is necessary to avoid bins w

The cell below will print a table of your values. HINT: Use np.concatenate to combine the rows of both arrays.

[0]:
```
fin_counts = ["<=4", "5", "6", "7", "8", "9-10", ">=11"] # new combined list of␣
 ↪x values
# YOUR CODE HERE
```

[0]:
```
%run -i c6-checkpoint5.py
```

[0]:
```
#print your bins in a neatly formatted table
print("Bin","\t\t","O_i","\t\t","E_i")
print("---------------------------------------------------------------------------")
for i in range(len(fin_counts)):
    print(fin_counts[i],
       "\t","\t",
       "{:}".format(fin_obs[i]),

       "\t","\t",
       "{:.2f}".format(fin_exp[i])
      )
```

3.0.5 Task 5

Calculate one-tailed p-value for the observed χ^2 statistic. What conclusion can be drawn about the null hypothesis?

[0]:
```
# YOUR CODE HERE
```