


<p><b>Nama:</b> <b>NUR IMAM</b></p> <p><b>NIM:</b> <b>065002300008</b></p>	 <p><b>Praktikum Probabilitas dan Statistika</b></p>	<p><b>MODUL 4</b></p> <p><b>Nama Dosen:</b> <b>Dedy Sugiarto</b></p>
<p><b>Hari/Tanggal:</b> <b>Hari, 27 MARET 2024p</b></p>		<p><b>Nama Asisten Labratorium:</b> <b>Kharisma Maulida Saara</b> <b>(064002200024)</b></p> <p><b>Tarum Widyasti Pertiwi</b> <b>(064002200027)</b></p>

## Probabilitas Peubah Acak Binom dan Poisson

### 1. Teori Singkat

Peubah acak (random variable\_ adalah variabel yang nilainya didapatkan dari nilai numerik suatu kejadian. Peubah acak juga merupakan fungsi yang memetakan set dari hasil-hasil yang mungkin dari suatu percobaan ke dalam angka. Terdapat dua jenis peubah acak yaitu diskrit dan kontinu. Peubah acak diskrit adalah jenis peubah acak di mana ruang sampelnya terdiri dari seperangkat nilai yang terbatas atau terhitung atau disebut juga dalam ruang bilangan cacah. Dua distribusi peluang peubah acak diskrit yang umum adalah distribusi binomial dan distribusi Poisson. Sedangkan peubah acak kontinu merepresentasikan hasil yang berasal dari suatu rentang nilai bilangan real.

#### Distribusi Binomial

Distribusi binomial digunakan ketika kita memiliki dua kemungkinan hasil (biasanya sukses dan gagal) dalam setiap percobaan, dan kita ingin mengetahui probabilitas jumlah keberhasilan dari sejumlah percobaan yang dilakukan.

Karakteristik Distribusi Binomial

- Setiap percobaan adalah independen.
- Setiap percobaan memiliki probabilitas keberhasilan yang sama.
- Variabel acak yang dihasilkan menggambarkan jumlah keberhasilan dalam sejumlah

percobaan yang tetap.

Jika  $X$  adalah variabel acak yang menggambarkan jumlah keberhasilan dalam  $n$  percobaan, dengan probabilitas keberhasilan  $p$  dalam setiap percobaan, maka rumus probabilitas binom adalah:

$$P(X = k) = \binom{n}{k} p^k (1 - p)^{n-k}$$

di mana:

- $n$  adalah jumlah percobaan,
- $k$  adalah jumlah keberhasilan yang diinginkan,
- $p$  adalah probabilitas keberhasilan dalam satu percobaan,
- $(1 - p)$  adalah probabilitas kegagalan dalam satu percobaan, dan
- $\binom{n}{k}$  adalah simbol kombinasi, yang menunjukkan jumlah cara yang mungkin untuk mendapatkan  $k$  keberhasilan dalam  $n$  percobaan.

### Distribusi Poisson

Distribusi Poisson digunakan untuk menggambarkan jumlah peristiwa langka yang terjadi dalam interval waktu atau ruang tertentu.

Karakteristik Distribusi Poisson

- Peristiwa terjadi secara acak dalam interval waktu atau ruang.
- Rata-rata jumlah peristiwa dalam interval waktu atau ruang tertentu adalah konstan.
- Peristiwa yang satu tidak memengaruhi peristiwa yang lain.

Jika  $X$  adalah variabel acak yang menggambarkan jumlah peristiwa yang terjadi dalam interval waktu atau ruang tertentu, dengan tingkat peristiwa  $\lambda$  per unit interval, maka rumus probabilitas Poisson adalah:

$$P(X = k) = \frac{e^{-\lambda} \lambda^k}{k!}$$

di mana:

- $\lambda$  adalah tingkat peristiwa (rata-rata jumlah peristiwa per interval),
- $k$  adalah jumlah peristiwa yang diinginkan, dan
- $e$  adalah konstanta Euler (2.71828...).

### Implementasi dalam Python

Anda dapat menggunakan berbagai paket perangkat lunak dalam Python, seperti **scipy.stats**, untuk menghitung probabilitas dari distribusi binomial dan Poisson.

## 2. Alat dan Bahan

Hardware : Laptop/PC Software: Jupyter Notebook

## 3. Elemen Kompetensi

- a. Latihan pertama – Distribusi Binomial
  1. Buka note baru pada Jupyter Notebook
  2. Implementasi manual rumus distribusi binomial

```
import math

# Fungsi untuk menghitung kombinasi (n choose k)
def n_choose_k(n, k):
    return math.factorial(n) / (math.factorial(k) * math.factorial(n - k))

# Fungsi untuk menghitung probabilitas P(X=k) dalam distribusi binomial
def binomial_probability(n, k, p):
    return n_choose_k(n, k) * (p ** k) * ((1 - p) ** (n - k))

# Parameter distribusi binomial
n = 10 # jumlah percobaan
p = 0.5 # probabilitas keberhasilan dalam satu percobaan
k = 3 # jumlah keberhasilan yang diinginkan

# Menghitung probabilitas P(X=k) untuk distribusi binomial
prob_binomial = binomial_probability(n, k, p)
print("Probabilitas P(X=k) untuk distribusi binomial:", prob_binomial)
```

## 3. Implementasi distribusi binomial dengan package scipy.stats

```
from scipy.stats import binom

# Distribusi Binomial
n = 10 # jumlah percobaan
p = 0.5 # probabilitas keberhasilan dalam satu percobaan
k = 3 # jumlah keberhasilan yang diinginkan

prob_binomial = binom.pmf(k, n, p)
print("Probabilitas P(X=k) untuk distribusi binomial:", prob_binomial)
```

## b. Latihan Kedua – Distribusi Poisson

1. Buka note baru pada Jupyter Notebook
2. Implementasi manual rumus distribusi Poisson

```
import math

# Fungsi untuk menghitung probabilitas P(X=k) dalam distribusi Poisson
def poisson_probability(lambd, k):
    return (lambd ** k) * math.exp(-lambd) / math.factorial(k)

# Parameter distribusi Poisson
lambd = 3 # rata-rata jumlah peristiwa dalam interval waktu/titik
k = 2 # jumlah peristiwa yang diinginkan

# Menghitung probabilitas P(X=k) untuk distribusi Poisson
prob_poisson = poisson_probability(lambd, k)
print("Probabilitas P(X=k) untuk distribusi Poisson:", prob_poisson)
```

### 3. Implementasi distribusi Poisson dengan package scipy.stats

```
from scipy.stats import poisson

# Distribusi Poisson
lamdb = 3 # rata-rata jumlah peristiwa dalam interval waktu/titik
k = 2 # jumlah peristiwa yang diinginkan

prob_poisson = poisson.pmf(k, lamdb)
print("Probabilitas P(X=k) untuk distribusi Poisson:", prob_poisson)
```

#### c. Latihan Ketiga – Tugas

1. Seorang penjual mengatakan bahwa di antara seluruh barang dagangannya yang dibungkus rapih, ada yang rusak sebanyak 10%. Seorang pelanggan membeli barang tersebut sebanyak 15 barang dan memilih secara acak. Jika  $X$  adalah banyaknya barang yang rusak dan mengikuti distribusi binomial.

Hitunglah:

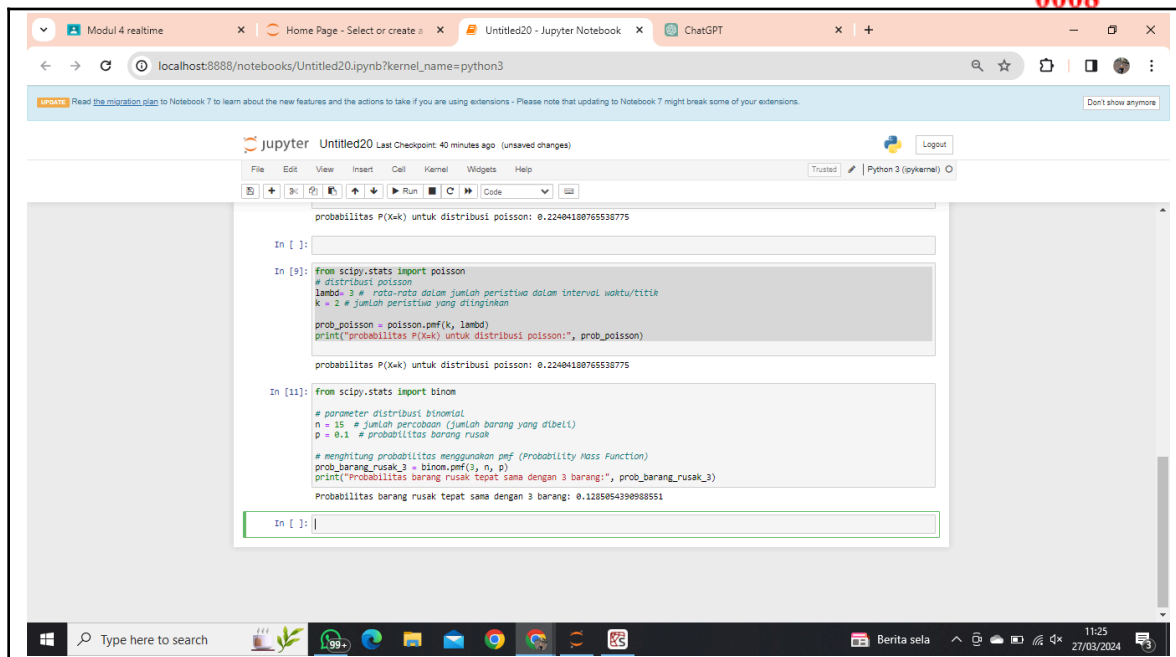
Probabilitas barang rusak tepat sama dengan 3 barang

Jawab :

R

```
> dbinom(3,15,0.1)
[1] 0.1285054
```

Output:



## Python

```
from scipy import stats
X = stats.binom(15, 0.1)
print(X.pmf(3))          # P(X = 3)
```

0.1285054390988551

Output:

```

In [9]: from scipy.stats import poisson
# distribusi poisson
lambda_3 = rata-rata dalam jumlah peristiwa dalam interval waktu/titik
k = 2 # jumlah peristiwa yang diinginkan
prob_poisson = poisson.pmf(k, lambda_3)
print("probabilitas P(X=k) untuk distribusi poisson:", prob_poisson)

probabilitas P(X=k) untuk distribusi poisson: 0.22404180765538775

In [11]: from scipy.stats import binom
# parameter distribusi binomial
n = 15 # jumlah percobaan (jumlah barang yang dibeli)
p = 0.1 # probabilitas barang rusak
# menghitung probabilitas menggunakan pmf (Probability Mass Function)
prob_barang_rusak_3 = binom.pmf(3, n, p)
print("Probabilitas barang rusak tepat sama dengan 3 barang:", prob_barang_rusak_3)

Probabilitas barang rusak tepat sama dengan 3 barang: 0.1285854390988551

In [12]: from scipy import stats
X = stats.binom(15, 0.1)
print(X.cdf(2))

0.1285854390988551

In [ ]:

```

Probabilitas barang rusak kurang dari sama dengan 2 barang

```
> pbinom(2,15,0.1)
[1] 0.8159389
```

Cara lain bisa juga dengan menjumlahkan ketiga nilai di bawah ini :

```
> dbinom(0,15,0.1)
[1] 0.2058911
> dbinom(1,15,0.1)
[1] 0.3431519
> dbinom(2,15,0.1)
[1] 0.2668959
```

Python :

```

print(X.cdf(2))          # P(X <= 2)

0.8159389308936089

```

Output:

```

In [11]: from scipy.stats import binom

# parameter distribusi binomial
n = 15 # jumlah percobaan (jumlah barang yang dibeli)
p = 0.1 # probabilitas barang rusak

# menghitung probabilitas menggunakan pmf (Probability Mass Function)
prob_barang_rusak_3 = binom.pmf(3, n, p)
print("Probabilitas barang rusak tepat sama dengan 3 barang:", prob_barang_rusak_3)

Probabilitas barang rusak tepat sama dengan 3 barang: 0.1285054390988551

In [12]: from scipy import stats
X = stats.binom(15, 0.1)
print(X.pmf(3)) #P(X=3)

0.1285054390988551

In [13]: from scipy.stats import binom

# parameter distribusi binomial
n = 15 # jumlah percobaan (jumlah barang yang dibeli)
p = 0.1 # probabilitas barang rusak

# menghitung probabilitas menggunakan cdf (Cumulative Distribution Function)
prob_barang_rusak_lebih_dari_2 = binom.cdf(2, n, p)
print("Probabilitas barang rusak kurang dari sama dengan 2 barang:", prob_barang_rusak_lebih_dari_2)

Probabilitas barang rusak kurang dari sama dengan 2 barang: 0.8159389388936089

In [ ]:

```

R:

$$P(5 < X \leq 7) = P(5 < X \leq 7)$$

$$> \text{dbinom}(6, 15, 0.1) + \text{dbinom}(7, 15, 0.1)$$

[1] 0.002216045

Output:

```

In [13]: from scipy.stats import binom

# parameter distribusi binomial
n = 15 # jumlah percobaan (jumlah barang yang dibeli)
p = 0.1 # probabilitas barang rusak

# menghitung probabilitas menggunakan cdf (Cumulative Distribution Function)
prob_barang_rusak_lebih_dari_2 = binom.cdf(2, n, p)
print("Probabilitas barang rusak kurang dari sama dengan 2 barang:", prob_barang_rusak_lebih_dari_2)

Probabilitas barang rusak kurang dari sama dengan 2 barang: 0.8159389388936089

In [14]: from scipy.stats import binom

# parameter distribusi binomial
n = 15 # jumlah percobaan (jumlah barang yang dibeli)
p = 0.1 # probabilitas barang rusak

# menghitung probabilitas menggunakan binom
prob_barang_rusak_6 = binom.pmf(6, n, p)
prob_barang_rusak_7 = binom.pmf(7, n, p)

prob_barang_rusak_6_atau_7 = prob_barang_rusak_6 + prob_barang_rusak_7
print("Probabilitas barang rusak 6 atau 7 barang:", prob_barang_rusak_6_atau_7)

Probabilitas barang rusak 6 atau 7 barang: 0.00221604519700002

In [ ]:

```



Python :

```
print(X.pmf(6)+X.pmf(7))
```

```
0.002216045197080002
```

Output:

```

In [13]: from scipy.stats import binom
# parameter distribusi binomial
n = 15 # jumlah percobaan (jumlah barang yang dibeli)
p = 0.1 # probabilitas barang rusak
# menghitung probabilitas menggunakan cdf (Cumulative Distribution Function)
prob_barang_rusak_lebih_dari_2 = binom.cdf(2, n, p)
print("Probabilitas barang rusak kurang dari sama dengan 2 barang:", prob_barang_rusak_lebih_dari_2)
Probabilitas barang rusak kurang dari sama dengan 2 barang: 0.8159389388936889

In [14]: from scipy.stats import binom
# parameter distribusi binomial
n = 15 # jumlah percobaan (jumlah barang yang dibeli)
p = 0.1 # probabilitas barang rusak
# menghitung probabilitas menggunakan binom
prob_barang_rusak_6 = binom.pmf(6, n, p)
prob_barang_rusak_7 = binom.pmf(7, n, p)
prob_barang_rusak_6_atau_7 = prob_barang_rusak_6 + prob_barang_rusak_7
print("Probabilitas barang rusak 6 atau 7 barang:", prob_barang_rusak_6_atau_7)
Probabilitas barang rusak 6 atau 7 barang: 0.002216045197080002
  
```

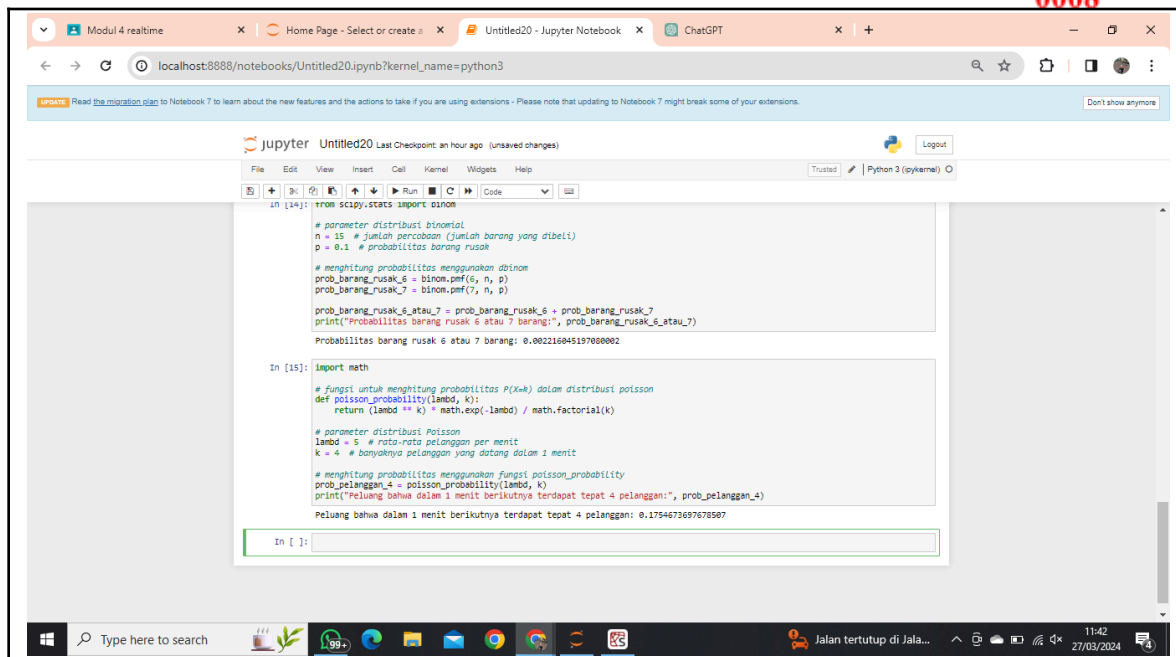
- Banyaknya pelanggan yang datang per menit pada suatu fasilitas pelayanan penukaran uang untuk lebaran diasumsikan mengikuti distribusi Poisson dengan mean ( $\lambda$ ) = 5. Hitunglah peluang bahwa dalam 1 menit berikutnya terdapat tepat 4 pelanggan yang akan datang?

Jawab :

R

```
> dpois(4,5)
[1] 0.1754674
```

Output:



Python :

```
from scipy import stats
Y = stats.poisson(5)
print(Y.pmf(4)) # P(Y = 4)
```

0.17546736976785063

Output:

The screenshot shows a Jupyter Notebook titled 'Untitled20' running on a local host. The code defines a Poisson probability function and calculates the probability of 4 customers arriving in 1 minute given a rate of 5 customers per minute.

```

prob_barang_rusak_7 = diston.pmf(x, n, p)
prob_barang_rusak_6 = prob_barang_rusak_7
print("Probabilitas barang rusak 6 atau 7 barang:", prob_barang_rusak_6_atau_7)
Probabilitas barang rusak 6 atau 7 barang: 0.00221045197080002

In [15]: import math
# fungsi untuk menghitung probabilitas P(X=k) dalam distribusi poisson
def poisson_probability(lambd, k):
    return (lambd ** k) * math.exp(-lambd) / math.factorial(k)

# parameter distribusi Poisson
lambd = 5 # rata-rata pelanggan per menit
k = 4 # banyaknya pelanggan yang datang dalam 1 menit

# menghitung probabilitas menggunakan fungsi poisson_probability
prob_pelanggan_4 = poisson_probability(lambd, k)
print("Peluang bahwa dalam 1 menit berikutnya terdapat tepat 4 pelanggan:", prob_pelanggan_4)
Peluang bahwa dalam 1 menit berikutnya terdapat tepat 4 pelanggan: 0.1754673697678507

In [16]: from scipy import stats
Y = stats.poisson(5)
print(Y.pmf(4)) # P = (Y = 4)
0.17546736976785063

In [ ]:

```

3. Hitunglah probabilitas bahwa dari 20 mahasiswa yang mengikuti ujian, tepat 15 mahasiswa lulus ujian. Probabilitas kelulusan adalah 0.7.

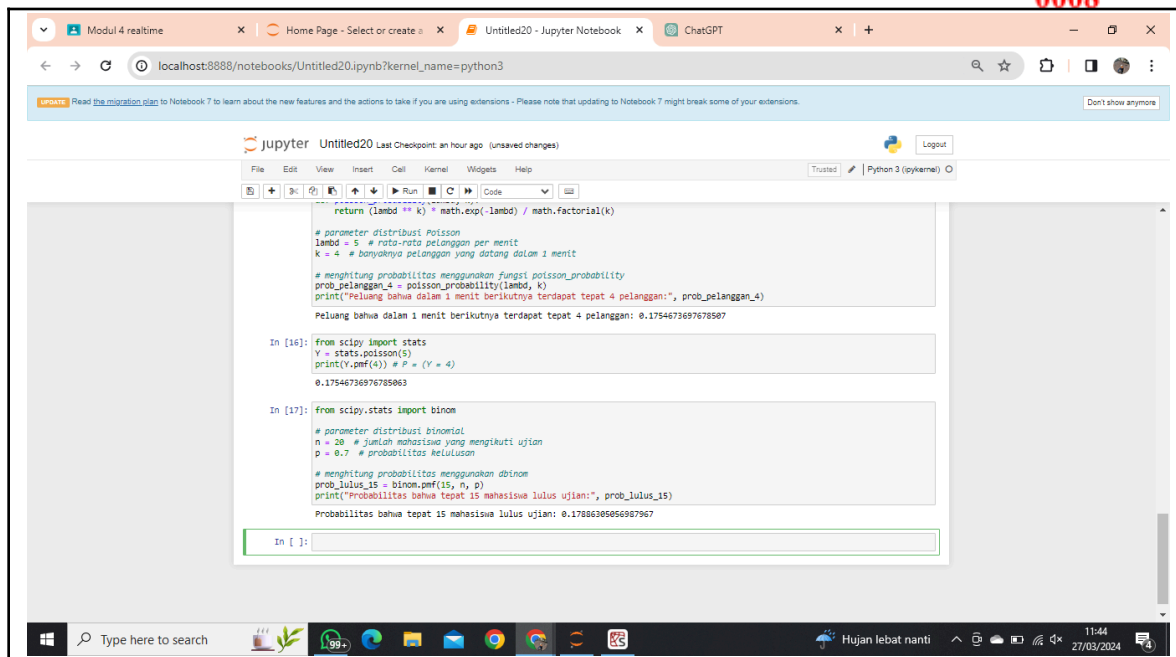
Jawab:

R

```
> dbinom(15,20,0.7)
```

```
[1] 0.1788631
```

Output:



## Python

```
from scipy import stats

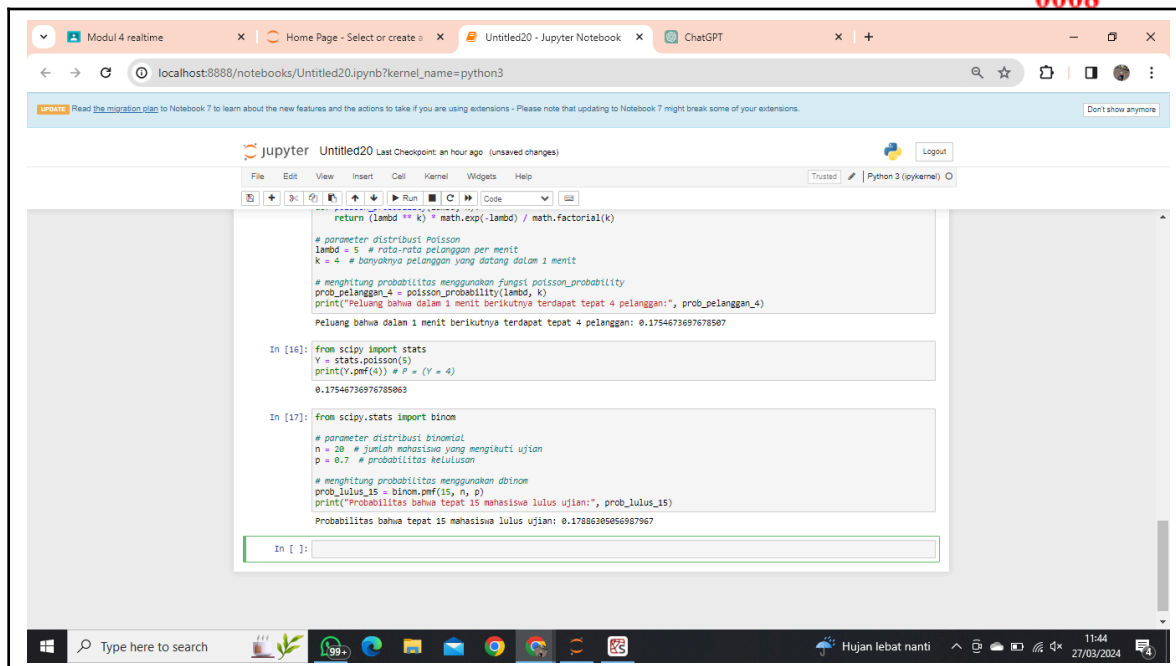
# Mendefinisikan variabel X sebagai distribusi binomial dengan 15 percobaan dan probabilitas sukses 0.1
X = stats.binom(20, 0.7)

# Menghitung probabilitas massa X sama dengan 3
pmf_15 = X.pmf(15)

# Mencetak nilai probabilitas
print(pmf_15)

0.17886305056987967
```

Output:



4. Dengan rata-rata, sebuah komputer mengalami 2 kejadian rusak dalam sebulan. Tentukan probabilitas bahwa dalam satu bulan, akan terjadi tepat 3 kejadian rusak.

Jawab:

```
> dpois(3,2)
[1] 0.180447
```

Python

```

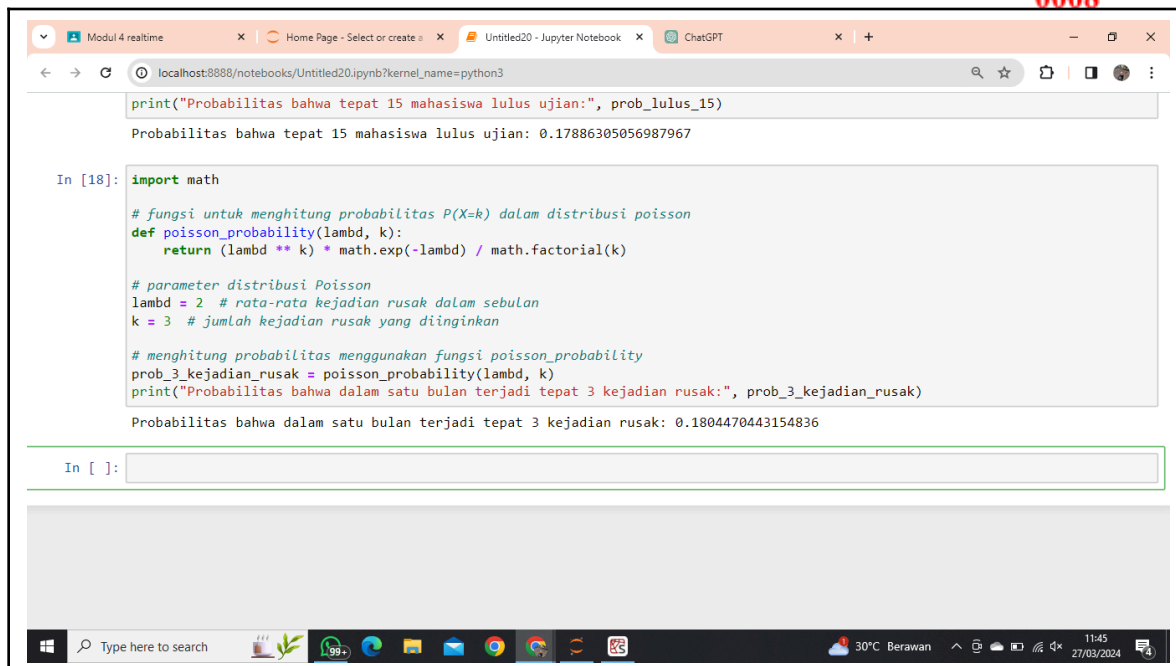
from scipy import stats
Y = stats.poisson(2)

print(Y.pmf(3))          # \ (P(Y=3)) \

0.18044704431548356

```

Output:



The screenshot shows a Jupyter Notebook running on a local host. The browser tabs include 'Modul 4 realtime', 'Home Page - Select or create', 'Untitled20 - Jupyter Notebook', and 'ChatGPT'. The address bar shows 'localhost:8888/notebooks/Untitled20.ipynb?kernel\_name=python3'. The notebook contains two code cells. The first cell has a print statement and its output. The second cell defines a Poisson probability function and calculates the probability for specific parameters.

```
print("Probabilitas bahwa tepat 15 mahasiswa lulus ujian:", prob_lulus_15)
Probabilitas bahwa tepat 15 mahasiswa lulus ujian: 0.17886305056987967

In [18]: import math

# fungsi untuk menghitung probabilitas P(X=k) dalam distribusi poisson
def poisson_probability(lambd, k):
    return (lambd ** k) * math.exp(-lambd) / math.factorial(k)

# parameter distribusi Poisson
lambd = 2 # rata-rata kejadian rusak dalam sebulan
k = 3 # jumlah kejadian rusak yang diinginkan

# menghitung probabilitas menggunakan fungsi poisson_probability
prob_3_kejadian_rusak = poisson_probability(lambd, k)
print("Probabilitas bahwa dalam satu bulan terjadi tepat 3 kejadian rusak:", prob_3_kejadian_rusak)
Probabilitas bahwa dalam satu bulan terjadi tepat 3 kejadian rusak: 0.1804470443154836

In [ ]:
```

#### 4. File Praktikum

Github Repository:

#### 5. Soal Latihan

Soal:

Seorang pengembang perangkat lunak melakukan uji coba pada 15 perangkat untuk mendeteksi kegagalan. Probabilitas bahwa perangkat mengalami kegagalan adalah 0.3. Hitunglah probabilitas bahwa tepat 5 dari 15 perangkat tersebut mengalami kegagalan.

R

Output:

Python

Output:

**6. Kesimpulan**

- a. Dalam pengerjaan praktikum Statistika, ...
- b. Kita juga dapat mengetahui...

**7. Cek List (✓)**

No	Elemen Kompetensi	Penyelesaian	
		Selesai	Tidak Selesai
1.	Latihan Pertama	...	
2.	Latihan Kedua	...	

**8. Formulir Umpan Balik**

No	Elemen Kompetensi	Waktu Pengerjaan	Kriteria
1.	Latihan Pertama	... Menit	...
2.	Latihan Kedua	... Menit	...

Keterangan:

1. Menarik
2. Baik
3. Cukup
4. Kurang