

# AK4183 Model Risiko II dan Simulasi

## Tugas 01 - 31 Agustus 2022

Oleh: **Venansius Ryan Tjahjono** NIM. **20821026**

Tugas pertama terdiri dari tiga soal yang meliputi cara pembangkitan sampel acak, mengecek/mengobservasi apakah sampel acak sudah cukup 'baik', dan simulasi perhitungan peluang atau sifat statistik.

### 1. Simulasi kejadian acak dengan cara membangkitkan sampel acak

#### **Studi kasus:**

Misalkan  $X$  adalah suatu peubah acak risiko yang menyatakan besar klaim asuransi umum per tahun. Diasumsikan  $X$  mengikut distribusi Eksponensial dengan parameter 3 (ratus juta rupiah). Karena  $X \sim \mathcal{E}(3)$ , fungsi peluang dari  $X$  diberikan oleh

$$f_X(x) = 3e^{-3x}$$

dengan  $x > 0$ . Untuk melakukan simulasi nilai-nilai klaim (acak) yang mungkin terjadi pada satu tahun kedepan, dilakukan dengan **metode transformasi invers** yang diberikan oleh algoritma berikut.

1. Tentukan fungsi distribusi dari  $X$ , katakan  $F_X(x)$ , dan cari  $F_X^{-1}(x)$ .
2. Bangkitkan  $u \sim U(0, 1)$ .
3. Hitung  $F_X^{-1}(u)$ , lalu simpan nilainya.
4. Ulangi langkah 2-3 sebanyak  $M$  iterasi untuk mendapatkan sampel acak berukuran  $M$ .

**Ilustrasi hasil simulasi lima sampel acak berdistribusi Eksponensial dengan parameter 3.**

Saat  $X \sim \mathcal{E}(3)$ , bisa diperoleh  $F_X(x) = 1 - e^{-3x}$  dan  $F_X^{-1}(x) = -\frac{\ln(1-x)}{3}$ .

```
In [29]: # import library NumPy untuk melakukan operasi bilangan dan pembangkitan bilangan acak un
import numpy as np

# memasang banyak sampel sebanyak 5
M = 5

# melakukan inisiasi array kosong untuk menampung hasil bangkitan
randomized_expo = []
for i in range(M):
    u = np.random.rand()
    x = -np.log(1-u)/3
    randomized_expo.append(x)

# mengecek hasil simulasi
randomized_expo
```

```
Out[29]: [0.40219685219933105,
0.8090998584913339,
0.4242764235347023,
0.41547627012025173,
0.09099594638296127]
```

Hasil bangkitan dan visualisasi sampel acak  $X \sim \mathcal{E}(3)$  berukuran 5000 dengan metode transformasi invers

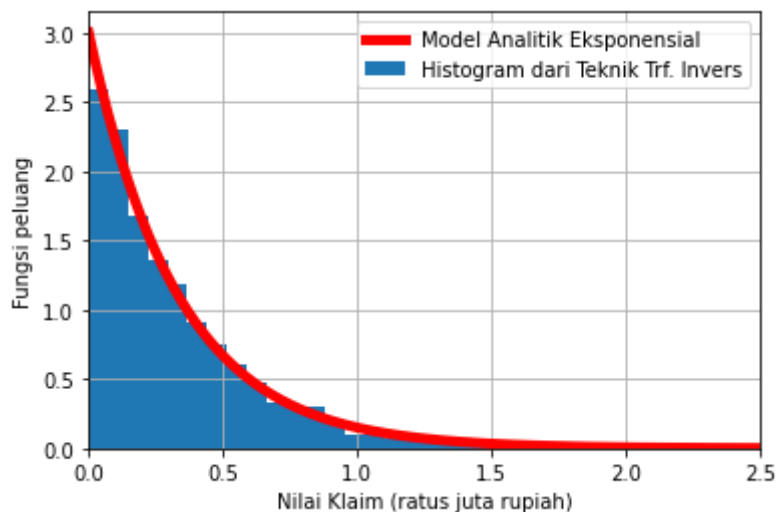
```
In [56]: # import library matplotlib untuk melakukan visualisasi
import matplotlib.pyplot as plt

M = 5000
randomized_expo = []
for i in range(M):
    u = np.random.rand()
    x = -np.log(1-u)/3
    randomized_expo.append(x)

# menyiapkan data untuk plot fungsi peluang
x_lb = np.min(randomized_expo)
x_ub = np.max(randomized_expo)
y_pdf = [3*np.exp(-3*x_pdf) for x_pdf in np.arange(x_lb, x_ub, 0.0001)]

# visualisasi hasil bangkitan dari metode transformasi invers dan model analitik
plt.hist(randomized_expo, bins=40, density=True, label='Histogram dari Teknik Trf. Invers')
plt.plot(np.arange(x_lb, x_ub, 0.0001), y_pdf, linewidth=5, color='r', label='Model Anali')
plt.xlim(0, 2.5)
plt.xlabel('Nilai Klaim (ratus juta rupiah)')
plt.ylabel('Fungsi peluang')
plt.grid(True)
plt.legend()
```

Out[56]: <matplotlib.legend.Legend at 0x1996decb9a0>



## 2. Melakukan pengecekan keacakan dari suatu sampel acak

Cara untuk mengecek apakah sampel acak cukup 'baik' adalah dengan uji hipotesis. Salah satu uji hipotesis yang bisa dilakukan untuk menguji keacakan (*randomness*) adalah **uji Wald–Wolfowitz** atau *runs test*. Uji ini diterapkan oleh Bradley (1968) sebagai salah satu cara menguji apakah suatu sampel bisa dibidang acak atau tidak. Uji Wald–Wolfowitz masih relatif populer digunakan untuk menguji keacakan pada beberapa penelitian terbaru (lih. Hoffman, 2019; dan Juhasz dkk., 2020).

Uji ini dimulai dengan menghitung median dari sampel acak (nominal) terlebih dahulu. Jika sampel **melebihi mediannya**, maka diberi label "+". Sedangkan bila sampel **kurang dari median**, akan diberi label "-". Kemudian, dihitung nilai  $n_1$  dan  $n_2$  yang masing-masing menyatakan banyak label "+" dan "-" pada sampel acak.

Setelah itu, dievaluasi nilai statistik uji  $Z$  yang didefinisikan sebagai

$$Z = \frac{R - \bar{R}}{s_R}$$

dengan

$$\bar{R} = \frac{2n_1n_2}{n_1 + n_2} + 1$$

dan

$$s_R^2 = \frac{2n_1n_2(2n_1n_2 - n_1 - n_2)}{(n_1 + n_2)^2(n_1 + n_2 - 1)}.$$

Untuk hipotesis nol dan alternatif dari uji Wald–Wolfowitz masing-masing diberikan oleh:

$H_0$ : sampel memiliki 'keacakan yang baik',

$H_1$ : sampel memiliki 'keacakan yang tidak baik'.

Hipotesis nol ditolak pada saat  $|Z| > Z_{1-\frac{\alpha}{2}}$  atau saat  $p\text{-value} < \alpha$ . Untuk melakukan pengujian uji Wald–Wolfowitz, digunakan library `statsmodels`.

```
In [57]: # import library statsmodels untuk uji Wald-Wolfowitz
from statsmodels.sandbox.stats.runs import runtest_1samp as wald_wolfowitz

# setup parameter banyaknya sampel acak yang akan dibangkitkan dengan metode transformasi
M = 5000
sampel_acak = []
for i in range(M):
    u = np.random.rand()
    x = -np.log(1-u)/3
    sampel_acak.append(x)

# pengujian Wald-Wolfowitz
uji = np.round(wald_wolfowitz(sampel_acak), 5)
z_test = uji[0]
p_val = uji[1]

print('Nilai statistik Z\t=', z_test)
print('p-value \t\t=', p_val)
print('')
```

```
Nilai statistik Z      = 0.66636
p-value               = 0.50518
```

$H_0$  gagal ditolak, sehingga sampel sudah memiliki keacakan yang baik

### 3. Simulasi untuk menentukan ekspektasi dari studi kasus pada No. 1

Saat  $X \sim \mathcal{E}(3)$ , secara analitik, ekspektasi  $X$  diberikan oleh

$$\mathbb{E}[X] = \int_0^{\infty} x (3e^{-3x}) dx = \frac{1}{3}.$$

Untuk menentukan  $\mathbb{E}[X]$  secara numerik, dilakukan simulasi melalui fungsi momen, sehingga

$$\mathbb{E}[X] = \frac{1}{M} \sum_{i=1}^M X_i$$

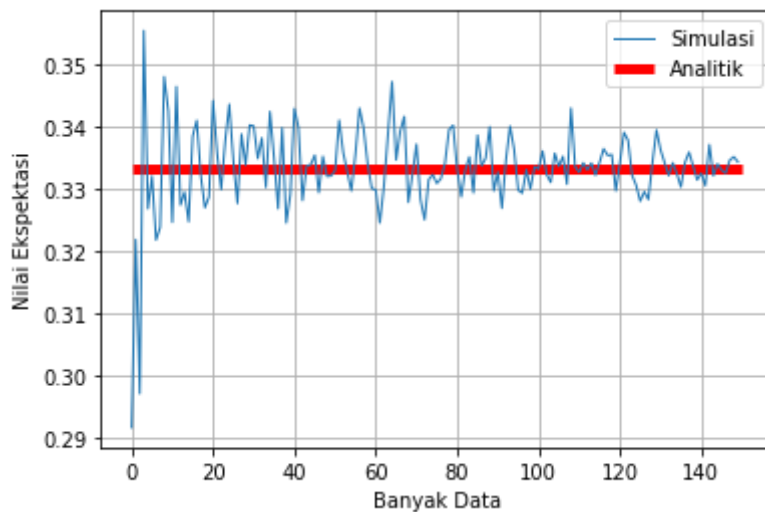
dengan  $M$  adalah banyaknya sampel yang dibangkitkan.

```
In [116... # setup parameter banyaknya sampel acak yang akan dibangkitkan dengan metode transformasi
M = 100
EX = []
for it in range(150):
    sampel_acak = []
    for i in range(M):
        u = np.random.rand()
        x = -np.log(1-u)/3
        sampel_acak.append(x)

    # menghitung nilai ekspektasi secara numerik
    EX.append(np.mean(sampel_acak))
    M = M + 100

plt.hlines(1/3, xmin=0, xmax=len(EX), color='r', linewidth=5, label='Analitik')
plt.plot(range(len(EX)), EX, linewidth=1, label='Simulasi')
plt.grid(True)
plt.legend()
plt.xlabel('Banyak Data')
plt.ylabel('Nilai Ekspektasi')
```

Out[116... Text(0, 0.5, 'Nilai Ekspektasi')



Pada hasil simulasi ini, bisa diperoleh suatu kesimpulan bahwa pada saat sampel berukuran cukup besar, maka nilai ekspektasi (simulasi) akan konvergen menuju nilai ekspektasi analitik.

In [ ]: