



Implementasi Bayes Decision Rule berbasis Maximum Posterior Probability and Minimum Risk

- Nama : Ahmad Naufal Hakim
- NIM : 13517055
- Link code : <https://github.com/ahmadnaufalhakim/IF5181-Bayes-Decision-Rule>
- Link video : https://www.youtube.com/watch?v=53mht_llagE

Deskripsi Masalah

Klasifikasi apakah suatu ikan merupakan ikan salmon atau ikan seabass menggunakan bantuan data histogram jumlah ikan salmon dan seabass terhadap warna kulitnya.

Deskripsi Solusi atau Implementasi

Untuk menyelesaikan permasalahan di atas, akan diimplementasikan Bayes Decision Rule berbasis Maximum Posterior Probability dan Minimum Risk dengan menggunakan bahasa Python.

Data yang digunakan berasal dari histogram dalam buku *Pattern Classification, 2nd Ed.* oleh Richard O. Duda (2001) di halaman 6 (figure 1.3).

Dari histogram tersebut, dibuat sebuah file csv dengan 3 kolom (lightness, countSalmon, dan countSeaBass), lalu dilakukan perhitungan probabilitas dan implementasi Bayes Decision Rule menggunakan bahasa Python.

1. Import file csv yang berisikan lightness, countSalmon, dan countSeaBass

```
import pandas as pd
data = pd.read_csv("lightness_fish_classification.csv")
data.head()
```



	lightness	countSalmon	countSeaBass
0	0.5-1.0	2	0

```
n_salmon = sum(data['countSalmon'])
print('Jumlah ikan salmon:', n_salmon)
```

```
n_seabass = sum(data['countSeaBass'])
print('Jumlah ikan seabass:', n_seabass)
```

```
↳ Jumlah ikan salmon: 74
   Jumlah ikan seabass: 57
```

X menyatakan sebuah range lightness dari ikan:

- $x[0] = '0.5-1.0'$
- $x[1] = '1.0-1.5'$
- dst.

```
x = [lightness for lightness in data['lightness']]
x
```

```
↳ ['0.5-1.0',
    '1.0-1.5',
    '1.5-2.0',
    '2.0-2.5',
    '2.5-3.0',
    '3.0-3.5',
    '3.5-4.0',
    '4.0-4.5',
    '4.5-5.0',
    '5.0-5.5',
    '5.5-6.0',
    '6.0-6.5',
    '6.5-7.0',
    '7.0-7.5',
    '7.5-8.0',
    '8.0-8.5',
    '8.5-9.0',
    '9.0-9.5',
    '9.5-10.0']
```

▼ 2. Mencari nilai $P(\text{kelas})$ dan $P(x \mid \text{kelas})$.

$P(\text{kelas})$ di mana kelas = **[salmon, seabass]**

- $P(\text{salmon}) = n_1 / (n_1 + n_2)$
- $P(\text{seabass}) = n_2 / (n_1 + n_2)$

```
n_fishes = n_salmon + n_seabass
```

```
P_prior = {
    'salmon': n_salmon / n_fishes,
    'seabass': n_seabass / n_fishes
}
```

```
P_prior
```

```
{'salmon': 0.5648854961832062, 'seabass': 0.4351145038167939}
```

$$P(x | \text{kelas}) = P(x, \text{kelas}) / N_{\text{kelas}}$$

```
P_conditional = {
    row['lightness']: {
        'salmon': row['countSalmon'] / n_salmon,
        'seabass': row['countSeaBass'] / n_seabass
    } for index, row in data.iterrows()
}
```

```
P_conditional
```

```
{'0.5-1.0': {'salmon': 0.02702702702702703, 'seabass': 0.0},
 '1.0-1.5': {'salmon': 0.10810810810810811, 'seabass': 0.0},
 '1.5-2.0': {'salmon': 0.05405405405405406, 'seabass': 0.0},
 '2.0-2.5': {'salmon': 0.0945945945945946, 'seabass': 0.0},
 '2.5-3.0': {'salmon': 0.13513513513513514, 'seabass': 0.0},
 '3.0-3.5': {'salmon': 0.16216216216216217, 'seabass': 0.0},
 '3.5-4.0': {'salmon': 0.10810810810810811, 'seabass': 0.017543859649122806},
 '4.0-4.5': {'salmon': 0.08108108108108109, 'seabass': 0.0},
 '4.5-5.0': {'salmon': 0.10810810810810811, 'seabass': 0.05263157894736842},
 '5.0-5.5': {'salmon': 0.05405405405405406, 'seabass': 0.05263157894736842},
 '5.5-6.0': {'salmon': 0.04054054054054054, 'seabass': 0.07017543859649122},
 '6.0-6.5': {'salmon': 0.013513513513513514, 'seabass': 0.10526315789473684},
 '6.5-7.0': {'salmon': 0.013513513513513514, 'seabass': 0.08771929824561403},
 '7.0-7.5': {'salmon': 0.0, 'seabass': 0.21052631578947367},
 '7.5-8.0': {'salmon': 0.0, 'seabass': 0.15789473684210525},
 '8.0-8.5': {'salmon': 0.0, 'seabass': 0.15789473684210525},
 '8.5-9.0': {'salmon': 0.0, 'seabass': 0.03508771929824561},
 '9.0-9.5': {'salmon': 0.0, 'seabass': 0.017543859649122806},
 '9.5-10.0': {'salmon': 0.0, 'seabass': 0.03508771929824561}}
```

3. Implementasi Bayes Decision Rule berbasis *Maximum Posterior Probability*

Nilai probabilitas posterior untuk suatu kelas w_i diberikan data x bisa didapatkan dengan menggunakan formula perhitungan sebagai berikut:

$$P(w_i | x) = \text{likelihood} * \text{prior} / \text{evidence}$$

atau

$$P(w_i | x) = (P(x | w_i) * P(w_i)) / P(x)$$

Karena nilai $P(w_i | x)$ akan dibandingkan antar semua kelas w yang ada dan nilai $P(x)$ untuk semua kelas bernilai sama, maka nilai $P(x)$ bisa kita abaikan dalam perhitungan.

```
P_posterior = {
    'salmon': {
        row['lightness']: P_conditional[row['lightness']]['salmon'] * P_prior['salmon']
    },
    'seabass': {
        row['lightness']: P_conditional[row['lightness']]['seabass'] * P_prior['seabass']
    }
}

for key1, value1 in P_posterior.items() :
    print(key1)
    for key2, value2 in value1.items() :
        print('\t', key2, ':', value2)
```



salmon

```
0.5-1.0 : 0.015267175572519087
1.0-1.5 : 0.06106870229007635
1.5-2.0 : 0.030534351145038174
2.0-2.5 : 0.0534351145038168
2.5-3.0 : 0.07633587786259544
3.0-3.5 : 0.09160305343511452
3.5-4.0 : 0.06106870229007635
4.0-4.5 : 0.04580152671755726
4.5-5.0 : 0.06106870229007635
5.0-5.5 : 0.030534351145038174
5.5-6.0 : 0.02200076335877862
```

Setelah mendapatkan nilai probabilitas posterior untuk kelas salmon dan seabass, maka kita bisa menentukan *decision class* yang akan diambil untuk setiap interval yang ada berdasarkan nilai probabilitas tersebut.

```
x = 0.5 - 1.0
```

for interval in x :

```
decision = 'salmon' if max(P_posterior['salmon'][interval], P_posterior['seabas
print('Ikan dengan nilai lightness', interval, 'akan diklasifikasikan sebagai i
```

```
↳ Ikan dengan nilai lightness 0.5-1.0 akan diklasifikasikan sebagai ikan salmon
Ikan dengan nilai lightness 1.0-1.5 akan diklasifikasikan sebagai ikan salmon
Ikan dengan nilai lightness 1.5-2.0 akan diklasifikasikan sebagai ikan salmon
Ikan dengan nilai lightness 2.0-2.5 akan diklasifikasikan sebagai ikan salmon
Ikan dengan nilai lightness 2.5-3.0 akan diklasifikasikan sebagai ikan salmon
Ikan dengan nilai lightness 3.0-3.5 akan diklasifikasikan sebagai ikan salmon
Ikan dengan nilai lightness 3.5-4.0 akan diklasifikasikan sebagai ikan salmon
Ikan dengan nilai lightness 4.0-4.5 akan diklasifikasikan sebagai ikan salmon
Ikan dengan nilai lightness 4.5-5.0 akan diklasifikasikan sebagai ikan salmon
Ikan dengan nilai lightness 5.0-5.5 akan diklasifikasikan sebagai ikan salmon
Ikan dengan nilai lightness 5.5-6.0 akan diklasifikasikan sebagai ikan seabass
Ikan dengan nilai lightness 6.0-6.5 akan diklasifikasikan sebagai ikan seabass
Ikan dengan nilai lightness 6.5-7.0 akan diklasifikasikan sebagai ikan seabass
Ikan dengan nilai lightness 7.0-7.5 akan diklasifikasikan sebagai ikan seabass
Ikan dengan nilai lightness 7.5-8.0 akan diklasifikasikan sebagai ikan seabass
Ikan dengan nilai lightness 8.0-8.5 akan diklasifikasikan sebagai ikan seabass
Ikan dengan nilai lightness 8.5-9.0 akan diklasifikasikan sebagai ikan seabass
Ikan dengan nilai lightness 9.0-9.5 akan diklasifikasikan sebagai ikan seabass
Ikan dengan nilai lightness 9.5-10.0 akan diklasifikasikan sebagai ikan seabass
```

▼ 4. Implementasi Bayes Decision Rule berbasis *Minimum Risk*

Dalam Bayes Decision Rule berbasis *Minimum Risk*, didefinisikan nilai alpha adalah action yang diambil untuk menentukan kelas suatu data berdasarkan nilai *loss* ketika mengambil action tersebut.

Contoh: untuk **mengambil** atau **meninggalkan** mata kuliah berdasarkan klasifikasi *good*, *fair*, atau *bad* pada sebuah mata kuliah

Dalam kasus ini, nilai alpha ada dua, yaitu menentukan jika suatu ikan adalah ikan salmon (**decide salmon**) atau ikan seabass (**decide seabass**). Misalkan aksi 'decide salmon' dan kelas salmon diidentifikasi sebagai angka 0, serta aksi 'decide seabass' dan kelas seabass diidentifikasi sebagai angka 1, maka nilai *loss function* bisa direpresentasikan sebagai sebuah matriks dengan dimensi 2x2.

Dimensi kolom menyatakan kelas ikan dan dimensi baris menyatakan aksi yang diambil.

```
loss = [[0, 1], [1, 0]]
for row in loss :
    print(row)
```

```
☞ [0, 1]
   [1, 0]
```

Nilai risk dari sebuah aksi α_i yang diambil diberikan data x bisa dihitung menggunakan formula perhitungan berikut ini:

$$R(\alpha_i | x) = \sum (\lambda(\alpha_i | w_j) * P(w_i | x)) \text{ untuk setiap kelas } j \text{ yang ada}$$

Setelah mendapatkan nilai risk untuk masing-masing aksi dan data, maka untuk setiap data bisa diambil aksi yang memiliki nilai risk yang lebih kecil.

Namun pada kasus ini, jumlah aksi dan kelas adalah sebanyak 2, sehingga bisa menggunakan Bayes Decision Rule berbasis Minimum Risk:

Jika $P(x | w_1)/P(x | w_2) > (\lambda_{12} - \lambda_{22})/(\lambda_{21} - \lambda_{11}) * P(w_2)/P(w_1)$, maka ambil aksi α_1 , kalau tidak ambil aksi α_2

- Nilai $P(x | w_1)/P(x | w_2)$ disebut *Likelihood Ratio* (LR)
- Nilai $(\lambda_{12} - \lambda_{22})/(\lambda_{21} - \lambda_{11}) * P(w_2)/P(w_1)$ disebut *Threshold*

Karena nilai $P(x | w_2)$ mungkin sama dengan nol, maka nilainya akan diganti menjadi $1e-323$ agar tidak terjadi pembagian oleh nilai nol

```
def likelihood_ratio(data) :
    try :
        return P_conditional[data]['salmon']/P_conditional[data]['seabass']
    except ZeroDivisionError :
        return P_conditional[data]['salmon']/1e-323
```

```
threshold = ((loss[0][1] - loss[1][1]) * P_prior['seabass']) / ((loss[1][0] - los
threshold
```

```
☞
```

```

0 7700700700700700
for interval in x :
    action = ''
    if likelihood_ratio(interval) > threshold :
        action = 'decide salmon'
    else :
        action = 'decide seabass'
    print('Untuk data ikan dengan interval lightness', interval, ', karena nilai Li

```

```

↳ Untuk data ikan dengan interval lightness 0.5-1.0 , karena nilai Likelihood Ratio > Thr
Untuk data ikan dengan interval lightness 1.0-1.5 , karena nilai Likelihood Ratio > Thr
Untuk data ikan dengan interval lightness 1.5-2.0 , karena nilai Likelihood Ratio > Thr
Untuk data ikan dengan interval lightness 2.0-2.5 , karena nilai Likelihood Ratio > Thr
Untuk data ikan dengan interval lightness 2.5-3.0 , karena nilai Likelihood Ratio > Thr
Untuk data ikan dengan interval lightness 3.0-3.5 , karena nilai Likelihood Ratio > Thr
Untuk data ikan dengan interval lightness 3.5-4.0 , karena nilai Likelihood Ratio > Thr
Untuk data ikan dengan interval lightness 4.0-4.5 , karena nilai Likelihood Ratio > Thr
Untuk data ikan dengan interval lightness 4.5-5.0 , karena nilai Likelihood Ratio > Thr
Untuk data ikan dengan interval lightness 5.0-5.5 , karena nilai Likelihood Ratio > Thr
Untuk data ikan dengan interval lightness 5.5-6.0 , karena nilai Likelihood Ratio < Thr
Untuk data ikan dengan interval lightness 6.0-6.5 , karena nilai Likelihood Ratio < Thr
Untuk data ikan dengan interval lightness 6.5-7.0 , karena nilai Likelihood Ratio < Thr
Untuk data ikan dengan interval lightness 7.0-7.5 , karena nilai Likelihood Ratio < Thr
Untuk data ikan dengan interval lightness 7.5-8.0 , karena nilai Likelihood Ratio < Thr
Untuk data ikan dengan interval lightness 8.0-8.5 , karena nilai Likelihood Ratio < Thr
Untuk data ikan dengan interval lightness 8.5-9.0 , karena nilai Likelihood Ratio < Thr
Untuk data ikan dengan interval lightness 9.0-9.5 , karena nilai Likelihood Ratio < Thr
Untuk data ikan dengan interval lightness 9.5-10.0 , karena nilai Likelihood Ratio < Th

```