TUGAS PERTEMUAN 4

Disusun Untuk Memenuhi Tugas Mata Kuliah Machine Learning

Dosen Pengampu : Estiyan Dwippriyoko,S.Kom., MT.

Disusun Oleh :

NAMA : MILADYNA FAUZIA

NPM : 41155050210023

KELAS : IF-A1

FAKULTAS TEKNIK PROGRAM STUDI INFORMATIKA

UNIVERSITAS LANGLANGBUANA

2024

1. Lakukan praktik dari https://youtu.be/Sj1ybuDDf9I?si=hCajHe1zasTQ9HGY , buat screenshot dengan nama kalian pada coding, kumpulkan dalam bentuk pdf, dari kegiatan ini:

   1.1. Pengenalan Bayes Theorem | Teori Bayes | Conditional Probability

   Classification Task dengan Naive Bayes

   Bayes' Theorem

   Bayes' theorem menawarkan suatu formula untuk menghitung nilai probability dari suatu event dengan memanfaatkan pengetahuan sebelumnya dari kondisi terkait; atau sering kali dikenal dengan istilah conditional probability.

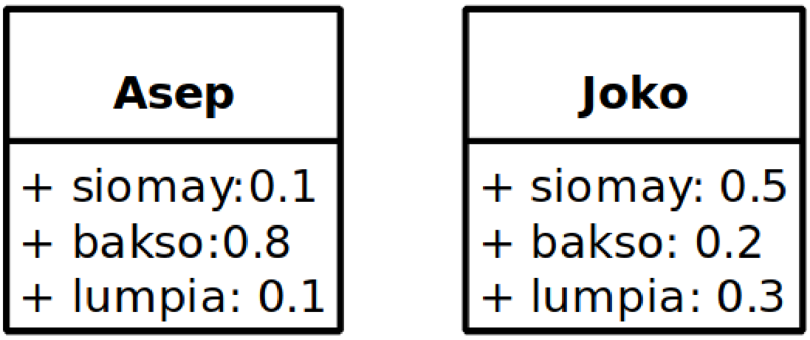
   $$P(A|B) = \frac{P(B|A) \times P(A)}{P(B)}$$

   $$P(y|B) = \frac{P(X|y) \times P(y)}{P(X)}$$

   $$Posterior = \frac{Likelihood \ \times Prior}{Evidence}$$

   1.2. Pengenalan Naive Bayes Classification

   Studi Kasus 1



   Misi: Lakukan prediksi siapa pelanggan yang melakukan pemesanan dengan diketahui pesanannya adalah lumpia dan bakso.

   1.3. Pengenalan Prior Probability

   Prior Probability: $P(y)$

   - $P(Asep) = 0{,}5$
   - $P(Joko) = 0{,}5$

1.4. Pengenalan Likelihood

Likelihood: $P(X|y)$

- Asep:

  $P(lumpia, bakso|Asep) = (0.1 \times 0.8)$

  $= 0.08$

- Joko

  $P(lumpia, bakso|Joko) = (0.3 \times 0.2)$

  $= 0.06$

1.5. Pengenalan Evidence | Normalizer

Evidence atau Normalizer : $P(X)$

$$Evidence \ = \sum (Likelihood \times Prior)$$

$$P(lumpia, bakso) = (0.08 \times 0.5) + (0.06 \times 0.5$$

$$= 0.07$$

1.6. Pengenalan Posterior Probability

Posterior Probability: $P(y|X)$

- Formula:

  $$Posterior = \frac{Likelihood \ \times Prior}{Evidence}$$

- Asep:

  $P(Asep|lumpia, bakso = (0.08 \times 0.5)$

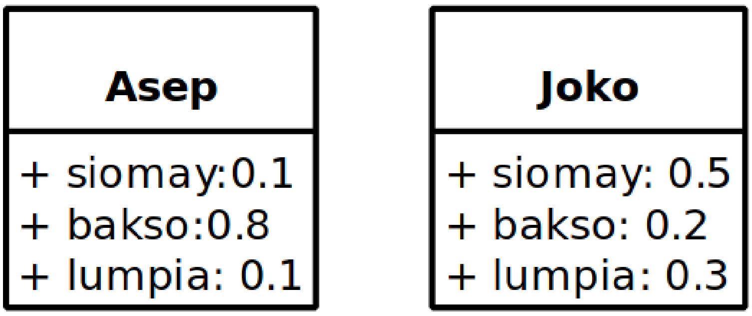
  $= 0.57$

- Joko

  $P(Joko|lumpia, bakso = (0.06 \times 0.5)$

  $= 0.43$

Studi Kasus 2



Misi: Lakukan prediksi siapa pelanggan yang melakukan pemesanan dengan diketahui pesanannya adalah siomay dan bakso.

Posterior Probability: $P(y|X)$(kasus 2)

- Pesanan: siomay, bakso
- Evidence : $P(X)$

  $$P(siomay, bakso) = (0.1 \times 0.8 \times 0.5) + (0.5 \times 0.2 \times 0.5$$
  $$= 0.09$$
- Asep :

  $$P(Asep|siomay, bakso = \frac{(0.1 \times 0.8)\ \times 0.5}{0.09}$$
  $$= 0.444$$
- Joko :

  $$P(Asep|siomay, bakso = \frac{(0.5 \times 0.2)\ \times 0.5}{0.09}$$
  $$= 0.555$$

Mengapa disebut Naïve ?

- Karena sewaktu kita mendefinisikan Likelihood $P(lumpia, bakso|Asep)$,
- kita mengasumsikan conditionally $P(lumpia|Asep)$ independent terhadap $P(bakso|Asep)$ demikian sebaliknya.
- Sehingga dapat diformulasikan sebagai berikut:

  $P(lumpia, bakso|Asep) = P(lumpia|Asep) \times P(bakso|Asep)$

Dataset: Breast Cancer Wisconsin (Diagnostic)

Load Dataset

```
jupyter Miladyna Fauzia Last Checkpoint: 4 minutes ago
File Edit View Run Kernel Settings Help                                   Trusted
Code                                          JupyterLab   Python 3 (ipykernel)

[1]: from sklearn.datasets import load_breast_cancer

     print(load_breast_cancer().DESCR)

     .. _breast_cancer_dataset:

     Breast cancer wisconsin (diagnostic) dataset
     --------------------------------------------

     **Data Set Characteristics:**

     :Number of Instances: 569

     :Number of Attributes: 30 numeric, predictive attributes and the class

     :Attribute Information:
         - radius (mean of distances from center to points on the perimeter)
         - texture (standard deviation of gray-scale values)
         - perimeter
         - area
         - smoothness (local variation in radius lengths)
         - compactness (perimeter^2 / area - 1.0)
         - concavity (severity of concave portions of the contour)
         - concave points (number of concave portions of the contour)
         - symmetry
         - fractal dimension ("coastline approximation" - 1)

         The mean, standard error, and "worst" or largest (mean of the three
         worst/largest values) of these features were computed for each image,
         resulting in 30 features.  For instance, field 0 is Mean Radius, field
         10 is Radius SE, field 20 is Worst Radius.

         - class:
                 - WDBC-Malignant
```

```
[3]: load_breast_cancer?
     #X, y = load_breast_cancer(return_X_y=True)
     #X.shape

     Signature: load_breast_cancer(*, return_X_y=False, as_frame=False)
     Docstring:
     Load and return the breast cancer wisconsin dataset (classification).

     The breast cancer dataset is a classic and very easy binary classification
     dataset.

     =================   ==============
     Classes                          2
     Samples per class    212(M),357(B)
     Samples total                  569
     Dimensionality                  30
     Features             real, positive
     =================   ==============

     The copy of UCI ML Breast Cancer Wisconsin (Diagnostic) dataset is
     downloaded from:
     https://archive.ics.uci.edu/dataset/17/breast+cancer+wisconsin+diagnostic

     Read more in the :ref:`User Guide <breast_cancer_dataset>`.

     Parameters
     ----------
     return_X_y : bool, default=False
         If True, returns ``(data, target)`` instead of a Bunch object.
         See below for more information about the `data` and `target` object.

         .. versionadded:: 0.18

     as_frame : bool, default=False
```

```
[5]: #load_breast_cancer?
     X, y = load_breast_cancer(return_X_y=True)
     X.shape

[5]: (569, 30)
```

Training & Testing Set

```
[7]: from sklearn.model_selection import train_test_split

     X_train, X_test, y_train, y_test = train_test_split(X,
                                                         y,
                                                         test_size=0.2,
                                                         random_state=0)

     print(f'X_train shape {X_train.shape}')
     print(f'X_test shape {X_test.shape}')

     X_train shape (455, 30)
     X_test shape (114, 30)
```

Naive Bayes dengan Scikit Learn

```
[9]: from sklearn.naive_bayes import GaussianNB
     from sklearn.metrics import accuracy_score

     model = GaussianNB()
     model.fit(X_train, y_train)

     y_pred = model.predict(X_test)
     accuracy_score(y_test, y_pred)

[9]: 0.9298245614035088

[11]: model.score(X_test, y_test)

[11]: 0.9298245614035088
```