

Modul Praktikum Kecerdasan Buatan



Rolly Maulana Awangga
0410118609

Applied Bachelor of Informatics Engineering
Program Studi D4 Teknik Informatika

Applied Bachelor Program of Informatics Engineering
Politeknik Pos Indonesia

Bandung 2019

‘Jika Kamu tidak dapat menahan lelahnya belajar,
Maka kamu harus sanggup menahan perihnya Kebodohan.’
Imam Syafi’i

Acknowledgements

Pertama-tama kami panjatkan puji dan syukur kepada Allah SWT yang telah memberikan rahmat dan hidayah-Nya sehingga Buku Pedoman Tingkat Akhir ini dapat diselesaikan.

Abstract

Buku Pedoman ini dibuat dengan tujuan memberikan acuan, bagi mahasiswa Tingkat Akhir dan dosen Pembimbing. Pada intinya buku ini menjelaskan secara lengkap tentang Standar pengerjaan Intership dan Tugas Akhir di Program Studi D4 Teknik Informatika, dan juga mengatur mekanisme, teknik penulisan, serta penilaiannya. Dengan demikian diharapkan semua pihak yang terlibat dalam aktivitas Bimbingan Mahasiswa Tingkat Akhir berjalan lancar dan sesuai dengan standar.

Contents

1	Mengenal Kecerdasan Buatan dan Scikit-Learn	1
1.1	Teori	1
1.2	Instalasi	2
1.3	Penanganan Error	2
1.4	andi muh aslam/1164064	2
1.4.1	sejarah dan perkembangan kecerdasan buatan	2
1.5	Instalasi	4
1.5.1	instalasi Library Scikit dari Anaconda	4
1.5.1.1	Mencoba Loading an example dataset	4
1.6	Aip Suprpto Munari/1164063	5
1.6.1	Teori	5
1.6.2	Instalasi	7
1.6.2.1	Instalasi Library Scikit dari Anaconda	7
1.6.2.2	Mencoba Loading an example Dataset	8
1.6.2.3	Learning and Predicting	8
1.6.3	Mencoba Model Persistance, menjelaskan maksud dari tulisan tersebut dan mengartikan per baris	9
1.6.4	Mencoba Conventions, menjelaskan maksud dari tulisan tersebut dan mengartikan per baris	10
1.7	Penanganan Error	14
1.7.0.1	Model Presistence	16
1.7.0.2	Conventions	18
1.7.1	Penanganan eror	24
1.7.1.1	ScreenShoot Eror	24
1.7.1.2	Tuliskan Kode Error dan Jenis Erornya	24
1.7.1.3	Solusi Pemecahan Masalah Error	25

2	Related Works	26
2.1	Aip Suprpto Munari/1164063	26
2.1.1	Teori	26
2.1.2	Binary Classification	26
2.1.3	Supervised Learning, Unsupervised Learning, Dan Classtering	26
2.1.4	Evaluasi Dan Akurasi	28
2.1.5	Confusion Matrix	28
2.1.6	Cara Kerja K-Fold Cross Validation	29
2.1.7	Decision Tree	30
2.1.8	Gain Dan Entropi	31
2.2	Same Topics	32
2.2.1	Topic 1	32
2.2.2	Topic 2	32
2.3	Same Method	32
2.3.1	Method 1	32
2.3.2	Method 2	33
3	Methods	34
3.1	The data	34
3.2	Method 1	34
3.3	Method 2	34
4	Experiment and Result	35
4.1	Experiment	35
4.2	Result	35
5	Conclusion	36
5.1	Conclusion of Problems	36
5.2	Conclusion of Method	36
5.3	Conclusion of Experiment	36
5.4	Conclusion of Result	36
6	Discussion	37
7	Discussion	38
8	Discussion	39

9 Discussion	40
10 Discussion	41
11 Discussion	42
12 Discussion	43
13 Discussion	44
14 Discussion	45
A Form Penilaian Jurnal	46
B FAQ	49
Bibliography	51

List of Figures

1.1	Langkah 1 instalasi anaconda..	4
1.2	Langkah 2 instalasi anaconda.	4
1.3	Langkah 3 instalasi anaconda.	5
1.4	Langkah 4 instalasi anaconda.	5
1.5	Langkah 1 dataset.	6
1.6	Langkah 2 dataset.	6
1.7	Langkah 3 dataset.	7
1.8	Langkah 4 dataset.	7
1.9	Langkah 5 dataset.	8
1.10	Hasil Pengujian Classifier	9
1.11	Hasil Pengujian Classifier	10
1.12	Pickle Pada Python	10
1.13	Pengujian Classifier Pickle	10
1.14	Penggunaan Joblib	11
1.15	Deklarasi Numpy	11
1.16	Contoh Type Casting	11
1.17	Menggunakan FitTransform	12
1.18	Regresi Yang Dilempar	12
1.19	Refitting dan Memperbaharui Parameter	13
1.20	MultiClass Classifier	14
1.21	MultiClass Classifier biner 2D	14
1.22	MultiLabel Classifier	14
1.23	Error Import	15
1.24	Instal Library Joblib	15
1.25	Berhasil Import Library Joblib	15
1.26	Download Anaconda.	16
1.27	Langkah pertama instalasi anaconda.	17
1.28	Langkah kedua instalasi anaconda.	18

1.29	Langkah ketiga instalasi anaconda.	19
1.30	Langkah terakhir instalasi anaconda.	20
1.31	Langkah pertama instalasi scikit pada CMD.	20
1.32	Langkah ketiga instalasi conda scikit pada CMD.	21
1.33	Langkah kedua pilih y.	21
1.34	Langkah cek version yang diinstall.	22
1.35	Hasil Tampilan 1.	22
1.36	Hasil Tampilan 2.	23
1.37	Hasil Tampilan 3.	23
1.38	Hasil Tampilan Error.	24
1.39	Hasil Tampilan Uji coba perintah joblib.	25
2.1	Binary Classification.	27
2.2	Supervised Learning.	28
2.3	Unsupervised Learning.	29
2.4	Clusterring.	30
2.5	Evaluasi Dan Akurasi.	31
2.6	K-Fold Cross Validation.	31
2.7	Decision Tree.	32
2.8	Gain Dan Entropi.	33
A.1	Form nilai bagian 1.	47
A.2	form nilai bagian 2.	48

Chapter 1

Mengenai Kecerdasan Buatan dan Scikit-Learn

Buku umum yang digunakan adalah [4] dan untuk sebelum UTS menggunakan buku *Python Artificial Intelligence Projects for Beginners*[2]. Dengan praktek menggunakan python 3 dan editor anaconda dan library python scikit-learn. Tujuan pembelajaran pada pertemuan pertama antara lain:

1. Mengerti definisi kecerdasan buatan, sejarah kecerdasan buatan, perkembangan dan penggunaan di perusahaan
2. Memahami cara instalasi dan pemakaian sci-kit learn
3. Memahami cara penggunaan variabel explorer di spyder

Tugas dengan cara dikumpulkan dengan pull request ke github dengan menggunakan latex pada repo yang dibuat oleh asisten riset.

1.1 Teori

Praktek teori penunjang yang dikerjakan :

1. Buat Resume Definisi, Sejarah dan perkembangan Kecerdasan Buatan, dengan bahasa yang mudah dipahami dan dimengerti. Buatan sendiri bebas plagiat[hari ke 1](10)
2. Buat Resume mengenai definisi supervised learning, klasifikasi, regresi dan unsupervised learning. Data set, training set dan testing set.[hari ke 1](10)

1.2 Instalasi

Membuka <https://scikit-learn.org/stable/tutorial/basic/tutorial.html>. Dengan menggunakan bahasa yang mudah dimengerti dan bebas plagiat. Dan wajib skreensut dari komputer sendiri.

1. Instalasi library scikit dari anaconda, mencoba kompilasi dan uji coba ambil contoh kode dan lihat variabel explorer[hari ke 1](10)
2. Mencoba Loading an example dataset, menjelaskan maksud dari tulisan tersebut dan mengartikan per baris[hari ke 1](10)
3. Mencoba Learning and predicting, menjelaskan maksud dari tulisan tersebut dan mengartikan per baris[hari ke 2](10)
4. mencoba Model persistence, menjelaskan maksud dari tulisan tersebut dan mengartikan per baris[hari ke 2](10)
5. Mencoba Conventions, menjelaskan maksud dari tulisan tersebut dan mengartikan per baris[hari ke 2](10)

1.3 Penanganan Error

Dari percobaan yang dilakukan di atas, apabila mendapatkan error maka:

1. skreensut error[hari ke 2](10)
2. Tuliskan kode eror dan jenis errornya [hari ke 2](10)
3. Solusi pemecahan masalah error tersebut[hari ke 2](10)

1.4 andi muh aslam/1164064

1.4.1 sejarah dan perkembangan kecerdasan buatan

1. didefinisikan kecerdasan yang ditunjukkan oleh suatu entitas buatan. Umumnya dianggap komputer. Kecerdasan Buatan (Artificial Intelligence atau AI) didefinisikan sebagai kecerdasan yang ditunjukkan oleh suatu entitas buatan. Sistem seperti ini umumnya dianggap komputer. Kecerdasan dimasukkan ke dalam mesin (komputer) agar dapat melakukan pekerjaan seperti yang dapat dilakukan manusia. Kecerdasan Buatan (Artificial Intelligence atau AI)

didefinisikan sebagai kecerdasan yang ditunjukkan oleh suatu entitas buatan. Sistem seperti ini umumnya di anggap komputer. Kecerdasan diciptakan dan dimasukkan melakukan pekerjaan seperti yang dapat dilakukan manusia.

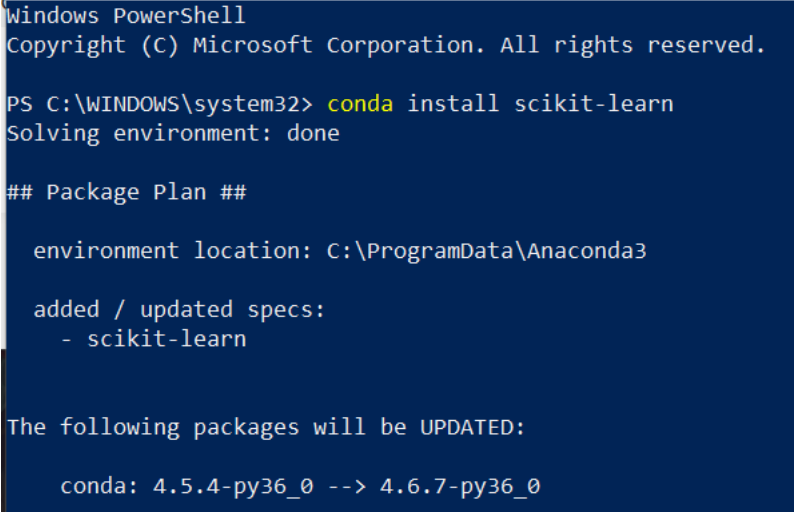
2. Sejarah dan perkembangan kecerdasan buatan terjadi pada musim panas tahun 1956 tercatat adanya seminar mengenai AI di Darmouth College. Seminar pada waktu itu dihadiri oleh sejumlah pakar komputer dan membahas potensi komputer dalam meniru kepandaian manusia. Akan tetapi perkembangan yang sering terjadi semenjak diciptakannya LISP, yaitu bahasa kecerdasan buatan yang dibuat tahun 1960 oleh John McCarthy. Istilah pada kecerdasan buatan atau Artificial Intelligence diambil dari Marvin Minsky dari MIT. Dia menulis karya ilmiah berjudul Step towards Artificial Intelligence, The Institute of radio Engineers Proceedings 49, January 1961[3].
3. Supervised learning merupakan sebuah pendekatan dimana sudah terdapat data yang dilatih, dan terdapat variable yang ditargetkan sehingga tujuan dari pendekatan ini adalah mengelompokkan suatu data ke data yang sudah ada. Sedangkan unsupervised learning tidak memiliki data latih, sehingga dari data yang ada, kita mengelompokkan data tersebut menjadi 2 bagian atau 3 bagian dan seterusnya.
4. Klasifikasi adalah salah satu topik utama dalam data mining atau machine learning. Klasifikasi yaitu suatu pengelompokan data dimana data yang digunakan tersebut mempunyai kelas label atau target.
5. Regresi adalah Supervised learning tidak hanya mempelajari classifier, tetapi juga mempelajari fungsi yang dapat memprediksi suatu nilai numerik. Contoh, ketika diberi foto seseorang, kita ingin memprediksi umur, tinggi, dan berat orang yang ada pada foto tersebut.
6. Data set adalah cabang aplikasi dari Artificial Intelligence/Kecerdasan Buatan yang fokus pada pengembangan sebuah sistem yang mampu belajar sendiri tanpa harus berulang kali di program oleh manusia.
7. Training set yaitu jika pasangan objek, dan kelas yang menunjuk pada objek tersebut adalah suatu contoh yang telah diberi label akan menghasilkan suatu algoritma pembelajaran.

Testing set digunakan untuk mengukur sejauh mana classifier berhasil melakukan klasifikasi dengan benar[1].

1.5 Instalasi

1.5.1 instalasi Library Scikit dari Anaconda

1. Download aplikasi Anaconda terlebih dahulu



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\WINDOWS\system32> conda install scikit-learn
Solving environment: done

## Package Plan ##

  environment location: C:\ProgramData\Anaconda3

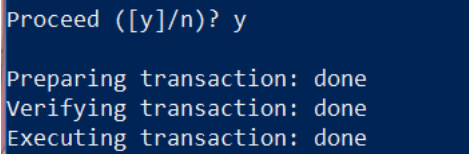
  added / updated specs:
    - scikit-learn

The following packages will be UPDATED:

  conda: 4.5.4-py36_0 --> 4.6.7-py36_0
```

Figure 1.1: Langkah 1 instalasi anaconda..

2. Proceed install anaconda



```
Proceed ([y]/n)? y

Preparing transaction: done
Verifying transaction: done
Executing transaction: done
```

Figure 1.2: Langkah 2 instalasi anaconda.

3. install scikit-learn

4. perintah print

1.5.1.1 Mencoba Loading an example dataset

1. Masuk Python terlebih dahulu
2. `from sklearn import datasets`(pada baris ini merupakan sebuah perintah untuk mengimport sebuah datasets dari file sklearn).

```

PS C:\WINDOWS\system32> pip install -U scikit-learn
Collecting scikit-learn
  Downloading https://files.pythonhosted.org/packages/ee/c8/c89ebdc0d7dbba6e6fd222daabd257da3c28a967
cp36-cp36m-win32.whl (4.3MB)
    100% |████████████████████████████████████████| 4.3MB 463kB/s
Requirement not upgraded as not directly required: numpy>=1.8.2 in c:\programdata\anaconda3\lib\site
Requirement not upgraded as not directly required: scipy>=0.13.3 in c:\programdata\anaconda3\lib\site
distributed 1.21.8 requires msgpack, which is not installed.
Installing collected packages: scikit-learn
  Found existing installation: scikit-learn 0.19.1
    Uninstalling scikit-learn-0.19.1:
      Successfully uninstalled scikit-learn-0.19.1
Successfully installed scikit-learn-0.20.2
You are using pip version 10.0.1, however version 19.0.3 is available.
You should consider upgrading via the 'python -m pip install --upgrade pip' command.

```

Figure 1.3: Langkah 3 instalasi anaconda.

```

PS C:\WINDOWS\system32> python
Python 3.6.5 [Anaconda, Inc.] (default, Mar 29 2018, 13:23:52) [MSC v.1900 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> print('Andi')
Andi

```

Figure 1.4: Langkah 4 instalasi anaconda.

3. iris datasets.load_iris()(pada baris kedua ini dimana iris merupakan suatu variabel yang berfungsi untuk mengambil data pada datasets dengan perintah .load_iris).
4. digits datasets.load_digits()(pada baris ketiga ini dimana digits merupakan suatu variabel yang berfungsi untuk mengambil data pada datasets dengan perintah .load_digits)
5. print(digits.data)(pada baris keempat ini merupakan perintah yang berfungsi untuk memanggil atau menampilkan variabel digits.data) `##### HEAD`

1.6 Aip Suprpto Munari/1164063

1.6.1 Teori

- (a) Definisi, sejarah, dan perkembangan kecerdasan buatan.

Definisi kecerdasan buatan adalah ilmu pengetahuan yang dapat membuat komputer untuk meniru kecerdasan manusia yang berhubungan dengan penangkapan, pemodelan, dan penyimpanan kecerdasan manusia dalam

```
C:\Users\Aslam>python
Python 3.6.5 |Anaconda, Inc.| (default, Mar 29 2018, 13:23:52) [MSC v.1900 32 bit (Intel)]
Type "help", "copyright", "credits" or "license" for more information.
```

Figure 1.5: Langkah 1 dataset.

```
>>> from sklearn import datasets
```

Figure 1.6: Langkah 2 dataset.

sebuah sistem teknologi. Contohnya yaitu melakukan analisa penalaran untuk mengambil suatu kesimpulan atau penerjemahan atau keputusan dari satu bahasa satu ke bahasa lain.

Sejarah dan perkembangan kecerdasan buatan terjadi pada musim panas tahun 1956 tercatat adanya seminar mengenai AI di Darmouth College. Seminar pada waktu itu dihadiri oleh sejumlah pakar komputer dan membahas potensi komputer dalam meniru kepandaian manusia. Akan tetapi perkembangan yang sering terjadi semenjak diciptakannya LISP, yaitu bahasa kecerdasan buatan yang dibuat tahun 1960 oleh John McCarthy. Istilah pada kecerdasan buatan atau Artificial Intelligence diambil dari Marvin Minsky dari MIT. Dia menulis karya ilmiah berjudul Step towards Artificial Intelligence, The Institute of radio Engineers Proceedings 49, January 1961[?].

- (b) Definisi supervised learning, klasifikasi, regresi, dan unsupervised learning. Data set, training set dan testing set.

Supervised learning merupakan sebuah pendekatan dimana sudah terdapat data yang dilatih, dan terdapat variable yang ditargetkan sehingga tujuan dari pendekatan ini adalah mengelompokkan suatu data ke data yang sudah ada. Sedangkan unsupervised learning tidak memiliki data latih, sehingga dari data yang ada, kita mengelompokkan data tersebut menjadi 2 bagian atau 3 bagian dan seterusnya.

Klasifikasi merupakan salah satu topik utama dalam data mining atau machine learning. Klasifikasi yaitu suatu pengelompokan data dimana data tersebut digunakan untuk mempunyai kelas label atau target.

Regresi adalah Supervised learning tidak hanya mempelajari classifier, tetapi juga mempelajari fungsi yang dapat memprediksi suatu nilai numerik. Contoh, ketika diberi foto seseorang, kita ingin memprediksi umur, tinggi, dan berat orang yang ada pada foto tersebut.

```
>>> iris = datasets.load_iris()
```

Figure 1.7: Langkah 3 dataset.

```
>>> digits = datasets.load_digits()
```

Figure 1.8: Langkah 4 dataset.

Data set adalah cabang aplikasi dari Artificial Intelligence/Kecerdasan Buatan yang fokus pada pengembangan sebuah sistem yang mampu belajar sendiri tanpa harus berulang kali di program oleh manusia.

Training set yaitu jika pasangan objek, dan kelas yang menunjuk pada objek tersebut adalah suatu contoh yang telah diberi label akan menghasilkan suatu algoritma pembelajaran.

Testing set digunakan untuk mengukur sejauh mana classifier berhasil melakukan klasifikasi dengan benar[?].

1.6.2 Instalasi

1.6.2.1 Instalasi Library Scikit dari Anaconda

- (a) Download aplikasi Anaconda terlebih dahulu. Lihat pada gambar 1.1
- (b) Install aplikasi Anaconda yang sudah di download tadi. Lihat pada gambar 1.2
- (c) Centang Keduanya lalu tekan tombol install. Lihat pada gambar 1.3
- (d) Setelah itu tunggu sampai proses instalasi selesai lalu jika sudah tekan tombol finish. Lihat pada gambar 1.4
- (e) Lalu buka command prompt anda dan tuliskan perintah berikut ini untuk mengecek apakah aplikasinya sudah terinstall. Lihat pada gambar 1.5
- (f) Kemudian ketikkan perintah pip install -U scikit-learn seperti gambar berikut. Lihat pada gambar 1.6
- (g) Lalu jika sudah ketikkan juga perintah conda install scikit-learn. Lihat pada gambar 1.7
- (h) dan setelah itu pilih y. Lihat pada gambar 1.8
- (i) Hasil version yang diinstall. Lihat pada gambar 1.9


```

>>> print(digits.data)
[[ 0.  0.  5. ... 0.  0.  0.]
 [ 0.  0.  0. ... 10.  0.  0.]
 [ 0.  0.  0. ... 16.  9.  0.]
 ...
 [ 0.  0.  1. ... 6.  0.  0.]
 [ 0.  0.  2. ... 12.  0.  0.]
 [ 0.  0. 10. ... 12.  1.  0.]]
>>>

```

Figure 1.9: Langkah 5 dataset.

1.6.2.2 Mencoba Loading an example Dataset

- `from sklearn import datasets`(pada baris ini merupakan sebuah perintah untuk mengimport class datasets dari packaged sklearn).
- `iris = datasets.load_iris()`(pada baris kedua ini dimana iris merupakan suatu estimator/parameter yang berfungsi untuk mengambil data pada item datasets.load_iris).
- `digits = datasets.load_digits()`(pada baris ketiga ini dimana digits merupakan suatu estimator/parameter yang berfungsi untuk mengambil data pada item datasets.load_digits).
- `print(digits.data)`(pada baris keempat ini merupakan perintah yang berfungsi untuk menampilkan estimator/parameter yang dipanggil pada item digits.data dan menampilkan outputannya) Lihat gambar 1.10.
- `digits.target`(barisan ini untuk mengambil target pada estimator/parameter digits dan menampilkan outputannya) Lihat gambar 1.11.
- `digits.images[0]`(barisan ini untuk mengambil images[0] pada estimator/parameter digits dan menampilkan outputannya) Lihat gambar 1.12.

=====

78cdb6514db9716f252ef2024a7df6097aace611

1.6.2.3 Learning and Predicting

- `from sklearn import svm`(pada baris ini merupakan sebuah perintah untuk mengimport class svm dari packaged sklearn).

```
svm.SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
```

Figure 1.10: Hasil Pengujian Classifier

- `clf = svm.SVC(gamma=0.001, C=100.)`(pada baris kedua ini `clf` sebagai estimator/parameter, `svm.SVC` sebagai class, `gamma` sebagai parameter untuk menetapkan nilai secara manual)
- `clf.fit(digits.data[:-1], digits.target[:-1])`(pada baris ketiga ini `clf` sebagai estimator/parameter, `fit` sebagai metode, `digits.data` sebagai item, `[:-1]` sebagai syntax pythonnya dan menampilkan outputannya) Lihat gambar 1.13.
- `clf.predict(digits.data[-1:])`

iiiiiii HEAD

1.6.3 Mencoba Model Persistance, menjelaskan maksud dari tulisan tersebut dan mengartikan per baris

1. Pada Python Shell ketikan "from sklearn import svm" artinya akan mengimport sebuah Support Vector Machine(SVM) yang merupakan algoritma classification yang akan diambil dari Scikit-Learn.
2. Kemudian, lanjutkan dengan "from sklearn import datasets" yang artinya akan mengambil package datasets dari Scikit-Learn.
3. ketikan, `clf = svm.SVC(gamma='scale')` berfungsi untuk mendeklarasikan suatu value yang bernama `clf` yang berisi `gamma`. Parameter `gamma` menentukan seberapa jauh pengaruh dari satu contoh training.
4. Ketikan, `X, y = iris.data, iris.target`, artinya `X` sebagai data iris, dan `y` merupakan larik target.
5. Ketikan, `clf.fit(X, y)` berfungsi untuk melakukan pengujian classifier. hasilnya seperti ini

Dari gambar diatas dapat dijelaskan bahwa akan mengimport Pickle dari Python. Pickle digunakan untuk serialisasi dan de-serialisasi struktur objek Python. Objek apa pun dengan Python dapat di-Pickle sehingga dapat disimpan di disk.

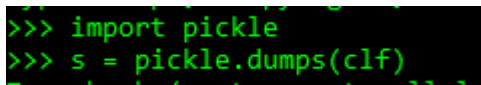
6. 

Figure 1.11: Hasil Pengujian Classifier

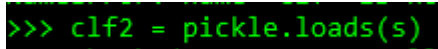


Figure 1.12: Pickle Pada Python

kemudian menyimpan data objek ke file CLF sebelumnya dengan menggunakan function `pickle.dumps(clf)`.

7. Setelah mengetikan fungsi fungsi diatas, selanjutnya ketikan "`clf2 = pickle.loads(s)`" yang artinya `pickle.loads` digunakan untuk memuat data pickle dari string byte. "S" dalam loads mengacu pada fakta bahwa dalam Python 2, data dimuat dari string.

Pada gambar diatas dilakukan pengujian nilai baru dengan menggunakan "`cf2.predict(X[0:1])`" dengan target asumsinya (0,1) hasilnya berbentuk array.

9. Dalam kasus khusus scikit-learn, mungkin lebih menarik untuk menggunakan `joblib` (dump dan load) untuk menggantikan `Pickle`, yang lebih efisien pada data besar tetapi hanya bisa di `Pickle` ke disk dan tidak ke string. untuk menggunakan `Joblib` pertama ketikan "`from joblib import dump , load`" yang artinya akan Merekonstruksi objek Python dari file yang sudah ada.

`dump(clf, 'filename.joblib')` akan merekontruksi file CLF yang tadi sudah dideklarasikan. `clf = load('filename.joblib')` untuk mereload model yang sudah di `Pickle`

1.6.4 Mencoba Conventions, menjelaskan maksud dari tulisan tersebut dan mengartikan per baris

1. `Import numpy as np`, digunakan untuk mengimport Numpy sebagai np.
`From sklearn import randomprojection` artinya modul yang mengimplemen-

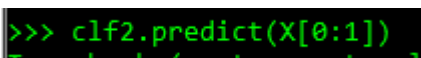
8. 

Figure 1.13: Pengujian Classifier Pickle

```
C:\Users\Asus-PC>python
Python 3.6.5 [Anaconda, Inc.] (default, Mar 29 2018, 13:23:52) [MSC v.1900 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> from joblib import dump, load
>>> dump(clf, 'filename.joblib')
```

Figure 1.14: Penggunaan Joblib

```
>>> import numpy as np
>>> from sklearn import random_projection
```

Figure 1.15: Deklarasi Numpy

tasikan cara sederhana dan efisien secara komputasi untuk mengurangi dimensi data dengan memperdagangkan sejumlah akurasi yang terkendali (sebagai varian tambahan) untuk waktu pemrosesan yang lebih cepat dan ukuran model yang lebih kecil.

Pada gambar diatas dapat dijelaskan bahwa :

`rng = np.random.RandomState(0)`, digunakan untuk menginisialisasikan random number generator.

`X = rng.rand(10, 2000)` artinya akan merandom value antara 10 sampai 2000.

`X = np.array(X, dtype='float32')` Array numpy terdiri dari buffer memori "mentah" yang diartikan sebagai array melalui "views". Anda dapat menganggap semua array numpy sebagai tampilan. Mendeklarasikan X sebagai float32.

3. Dalam contoh ini, X adalah float32, yang dilemparkan ke float64 oleh `fittransform(X)`.

4. Target regresi dilemparkan ke float64 dan target klasifikasi dipertahankan.

`list(clf.predict(irisdata[:3]))`, akan memprediksi 3 data dari iris.

`clf.fit irisdata, iristargetnames[iristarget]` menguji classifier dengan ada target-

```
>>> rng = np.random.RandomState(0)
>>> X = rng.rand(10, 2000)
>>> X = np.array(X, dtype='float32')
>>> X.dtype
dtype('float32')
```

2.

Figure 1.16: Contoh Type Casting

```
>>> transformer = random_projection.GaussianRandomProjection()
```

Figure 1.17: Menggunakan FitTransform

```
C:\Users\Asus-PC>python
Python 3.6.5 |Anaconda, Inc.| (default, Mar 29 2018, 13:23:52) [MSC v.1900 32 bit
Intel)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> from sklearn import datasets
>>> from sklearn.svm import SVC
>>> iris = datasets.load_iris()
>>> clf = SVC(gamma='scale')
>>> clf.fit(iris.data, iris.target)
SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
    decision_function_shape='ovr', degree=3, gamma='scale', kernel='rbf',
    max_iter=-1, probability=False, random_state=None, shrinking=True,
    tol=0.001, verbose=False)
>>> list(clf.predict(iris.data[:3]))
[0, 0, 0]
>>> clf.fit(iris.data, iris.target_names[iris.target])
SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
    decision_function_shape='ovr', degree=3, gamma='scale', kernel='rbf',
    max_iter=-1, probability=False, random_state=None, shrinking=True,
    tol=0.001, verbose=False)
>>> list(clf.predict(iris.data[:3]))
['setosa', 'setosa', 'setosa']
```

Figure 1.18: Regresi Yang Dilempar

nya yaitu irisnya sendiri.

`list(clf.predict(irisdata[:3]))`, setelah diuji maka akan muncul datanya seperti dibawah ini

Di sini, prediksi pertama () mengembalikan array integer, karena `iristarget` (array integer) yang digunakan sesuai. Prediksi kedua () mengembalikan array string, karena `iristargetnames` cocok.

5. Refitting dan Memperbaharui Parameter

`y = rngbinomial(1, 0.5, 100)` , random value dengan angka binomial atau suku dua untuk `y`

`clfsetparams(kernel='linear')fit(X, y)` mengubahn kernel default menjadi linear
`clfsetparams(kernel='rbf', gamma='scale')fit(X, y)` Di sini, kernel default rbf pertama kali diubah menjadi linear melalui

`SVCsetparams()` setelah estimator dibuat, dan diubah kembali ke rbf untuk

```

>>> import numpy as np
>>> from sklearn.svm import SVC
>>> rng = np.random.RandomState(0)
>>> X = rng.rand(100, 10)
>>> y = rng.binomial(1, 0.5, 100)
>>> X_test = rng.rand(5,10)
>>> clf = SVC()
>>> clf.set_params(kernel='linear').fit(X,y)
SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
    decision_function_shape='ovr', degree=3, gamma='auto_deprecated',
    kernel='linear', max_iter=-1, probability=False, random_state=None,
    shrinking=True, tol=0.001, verbose=False)
>>> clf.predict(X_test)
array([1, 0, 1, 1, 0])
>>> clf.set_params(kernel='rbf', gamma='scale').fit(X,y)
SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
    decision_function_shape='ovr', degree=3, gamma='scale', kernel='rbf',
    max_iter=-1, probability=False, random_state=None, shrinking=True,
    tol=0.001, verbose=False)
>>> clf.predict(X_test)
array([1, 0, 1, 1, 0])

```

Figure 1.19: Refitting dan Memperbaharui Parameter

mereparasi estimator dan membuat prediksi kedua.

6. MultiClass VS MultiLabel Classifier

from sklearn.multiclass import OneVsRestClassifier ,adalah ketika kita ingin melakukan klasifikasi multiclass atau multilabel dan baik untuk menggunakan OneVsRestClassifier per kelas. Untuk setiap classifier, kelas tersebut dipasang terhadap semua kelas lainnya. (Ini cukup jelas dan itu berarti bahwa masalah klasifikasi multiclass / multilabel dipecah menjadi beberapa masalah klasifikasi biner).

from sklearn.preprocessing import LabelBinarizer ,adalah kelas utilitas untuk membantu membuat matriks indikator label dari daftar label multi-kelas

Dalam gambar dibawah, classifier cocok pada array 1d label multiclass dan oleh karena itu metode predict () memberikan prediksi multiclass yang sesuai.

7. Di sini, classifier cocok () pada representasi label biner 2d dari y, menggunakan LabelBinarizer. Dalam hal ini predict () mengembalikan array 2d yang mewakili prediksi multilabel yang sesuai.
8. from sklearn.preprocessing import MultiLabelBinarizer , artinya Transformasi antara iterable dari iterables dan format multilabel.

```

>>> from sklearn.svm import SVC
>>> from sklearn.multiclass import OneVsRestClassifier
>>> from sklearn.preprocessing import LabelBinarizer
>>> X = [[1, 2], [2, 4], [4, 5], [3, 2], [3, 11]]
... y = [0, 0, 1, 1, 2]
      File "<stdin>", line 2
        y = [0, 0, 1, 1, 2]
          ^
SyntaxError: invalid syntax
>>> X = [[1, 2], [2, 4], [4, 5], [3, 2], [3, 11]]
>>> y = [0, 0, 1, 1, 2]
>>> classif = OneVsRestClassifier(estimator=SVC(gamma='scale',
... random_state=0))
>>> classif.fit(X, y).predict(X)
array([0, 0, 1, 1, 2])
>>>

```

Figure 1.20: MultiClass Classifier

```

>>> classif.fit(X, y).predict(X)
array([0, 0, 1, 1, 2])

```

Figure 1.21: MultiClass Classifier biner 2D

Dalam hal ini, penggolongnya sesuai pada setiap instance yang diberi beberapa label. MultiLabelBinarizer digunakan untuk membuat binarize array 2d dari multilabel agar sesuai. Hasilnya, predict () mengembalikan array 2d dengan beberapa label yang diprediksi untuk setiap instance.

1.7 Penanganan Error

HARI KEDUA

```

>>> from sklearn.preprocessing import MultiLabelBinarizer
>>> y = [[0, 1], [0, 2], [1, 3], [0, 2, 3], [2, 4]]
>>> y = MultiLabelBinarizer().fit_transform(y)
>>> classif.fit(X, y).predict(X)
array([[1, 0, 1, 0, 0],
       [1, 0, 1, 0, 0],
       [0, 0, 1, 1, 0],
       [1, 0, 1, 0, 0],
       [0, 0, 1, 0, 1]])

```

Figure 1.22: MultiLabel Classifier

```

ameError: name 'clf' is not defined
>> from joblib import dump, load
>>

```

Figure 1.23: Error Import

```

C:\Users\Asus-PC>pip install joblib
Collecting joblib
  Downloading https://files.pythonhosted.org/packages/cd/c1/50a758e8247561e58cb8730
  100% | 286kB 1.8MB/s
distributed 1.21.8 requires msgpack, which is not installed.
Installing collected packages: joblib
Successfully installed joblib-0.13.2
You are using pip version 10.0.1, however version 19.0.3 is available.
You should consider upgrading via the 'python -m pip install --upgrade pip' command

```

Figure 1.24: Instal Library Joblib

1. Berikut ini merupakan eror yang ditemui pada saat melakukan percobaan skrip.
2. Pada gambar eror diatas, kode erornya adalah "ImportError: No Module Named" artinya mengalami masalah saat mengimpor modul yang ditentukan.
3. Solusinya bisa dilakukan seperti berikut :
eror diatas terjadi dikarenakan Library Joblib belum terinstal pada PC. Maka dari itu sekarang kita harus menginstalnya dulu.
4. Buka CMD, kemudian ketikan "pip install joblib" tunggu sampai instalasi berhasil seperti gambar berikut.
5. Apabila sudah terinstall, dapat dilakukan lagi import library joblib, maka akan berhasil seperti dibawah berikut

=====

```

C:\Users\Asus-PC>python
Python 3.6.5 [Anaconda, Inc.] (default, Mar 29 2018, 13:23:52) [MSC v.1900 32 bit (
Type "help", "copyright", "credits" or "license" for more information.
>>> from joblib import dump, load
>>> dump(clf, 'filename.joblib')

```

Figure 1.25: Berhasil Import Library Joblib

Anaconda installer archive			
Filename	Size	Last Modified	MD5
Anaconda2-2018.12-Linux-ppc64le.sh	289.7M	2018-12-21 13:14:33	d50ce6eb037f72edfe8f94f90d61aca6
Anaconda2-2018.12-Linux-x86.sh	518.6M	2018-12-21 13:13:15	7d26c7551af6802eb83ecd34282056d7
Anaconda2-2018.12-Linux-x86_64.sh	628.2M	2018-12-21 13:13:10	84f39388da2c747477cf14cb02721b93
Anaconda2-2018.12-MacOSX-x86_64.pkg	640.7M	2018-12-21 13:14:30	c2bfeef310714501a59fd58166e6393d
Anaconda2-2018.12-MacOSX-x86_64.sh	547.1M	2018-12-21 13:14:31	f4d8b10e9a754884fb96e68e0b276a
Anaconda2-2018.12-Windows-x86.exe	458.6M	2018-12-21 13:16:27	f123fda0ec8928bb7d55d1ca72c0d784
Anaconda2-2018.12-Windows-x86_64.exe	560.6M	2018-12-21 13:16:17	10ff4176a94fcff86e6253b0cc82c782
Anaconda3-2018.12-Linux-ppc64le.sh	313.6M	2018-12-21 13:13:03	a775fb6d6c441b899ff2327bd9dadcd
Anaconda3-2018.12-Linux-x86.sh	542.7M	2018-12-21 13:13:14	4c9922d1547128b866c6b9cf750c03c7
Anaconda3-2018.12-Linux-x86_64.sh	652.5M	2018-12-21 13:13:06	c9af603d89656bc89680889ef1f92623

Figure 1.26: Download Anaconda.

1.7.0.1 Model Persistence

- `from sklearn import svm`(pada baris ini merupakan sebuah perintah untuk mengimport class svm dari packaged sklearn).

- `from sklearn import datasets`(pada baris ini merupakan sebuah perintah untuk mengimport class datasets dari packaged sklearn).

- `clf = svm.SVC(gamma='scale')`

(pada baris ketiga ini clf sebagai estimator/parameter, svm.SVC sebagai class, gamma sebagai parameter untuk menetapkan nilai secara manual dengan nilai scale).

- `iris = datasets.load_iris()`

(pada baris keempat ini iris sebagai estimator/parameter, datasets.load_iris() sebagai item dari suatu nilai).

- `X, y = iris.data, iris.target`

(pada baris kelima ini X, y sebagai estimator/parameter, iris.data, iris.target sebagai item dari 2 nilai yang ada).

- `clf.fit(X, y)`

(pada baris keenam ini clf sebagai estimator/parameter dengan menggunakan metode fit untuk memanggil estimator X, y dengan outputannya)

- `import pickle`

(pickle merupakan sebuah class yang di import).

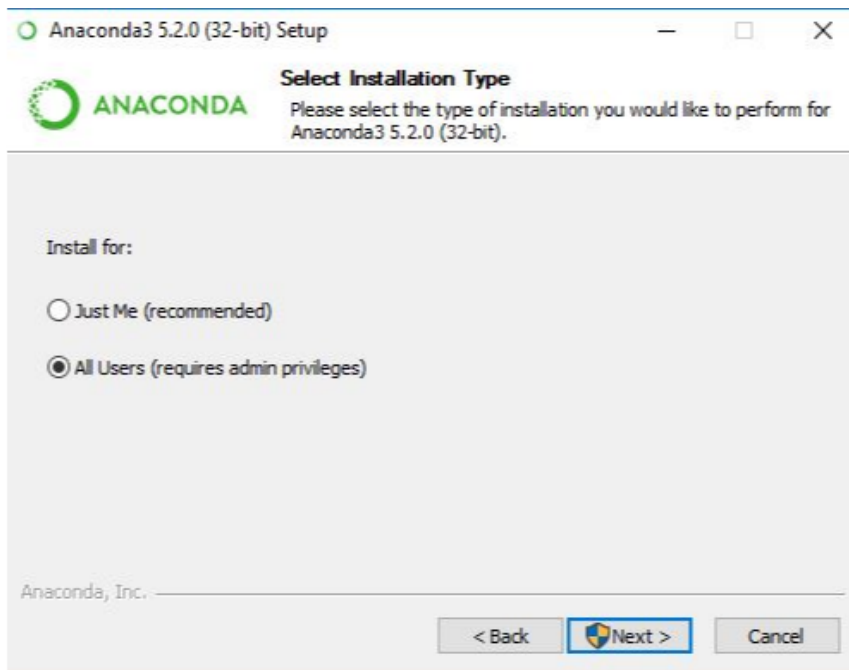


Figure 1.27: Langkah pertama instalasi anaconda.

- `s = pickle.dumps(clf)`
(pada baris ini `s` sebagai estimator/parameter dengan `pickle.dumps` merupakan suatu nilai/item dari estimator/parameter `clf`)
- `clf2 = pickle.loads(s)`
(pada baris ini `clf2` sebagai estimator/parameter, `pickle.loads` sebagai suatu item, dan `s` sebagai estimator/parameter yang dipanggil)
- `clf2.predict(X[0:1])`
(pada baris ini `clf2.predict` sebagai suatu item dengan menggunakan metode `predict` untuk menentukan suatu nilai dari (`X[0:1]`))
- `y[0]`
(pada estimator/parameter `y` berapapun angka yang diganti nilainya akan selalu konstan yaitu 0)
- `from joblib import dump, load`
(pada baris berikut ini merupakan sebuah perintah untuk mengimport class `dump`, `load` dari packaged `joblib`).

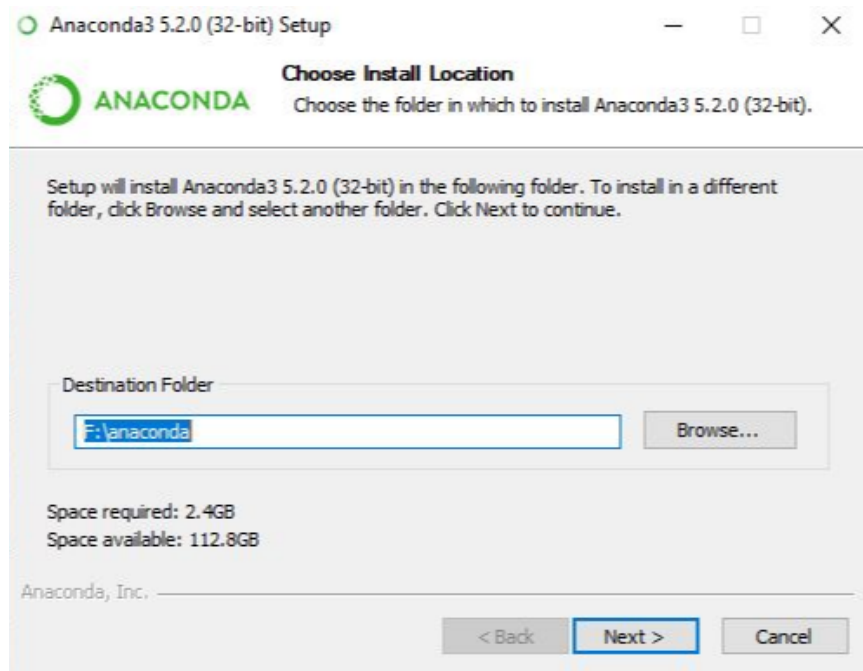


Figure 1.28: Langkah kedua instalasi anaconda.

- `dump(clf, 'filename.joblib')`
(pada baris berikutnya `dump` di sini sebagai class yang didalamnya terdapat nilai dari suatu item `clf` dan data `joblib`).
- `clf = load('filename.joblib')`
(pada baris terakhir `clf` sebagai estimato/parameter dengan suatu nilai `load` berfungsi untuk mengulang data sebelumnya)
- dari ketiga baris akhir tersebut jika di jalankan aau dituliskan perintah seperti itu maka akan menampilkan tampilan eror

1.7.0.2 Conventions

1. Type Casting

- `from sklearn import svm`
(pada baris ini merupakan sebuah perintah untuk mengimport class `svm` dari packaged `sklearn`).
- `from sklearn import random_projection`
(pada baris ini merupakan sebuah perintah untuk mengimport class `random_projection` dari packaged `sklearn`).

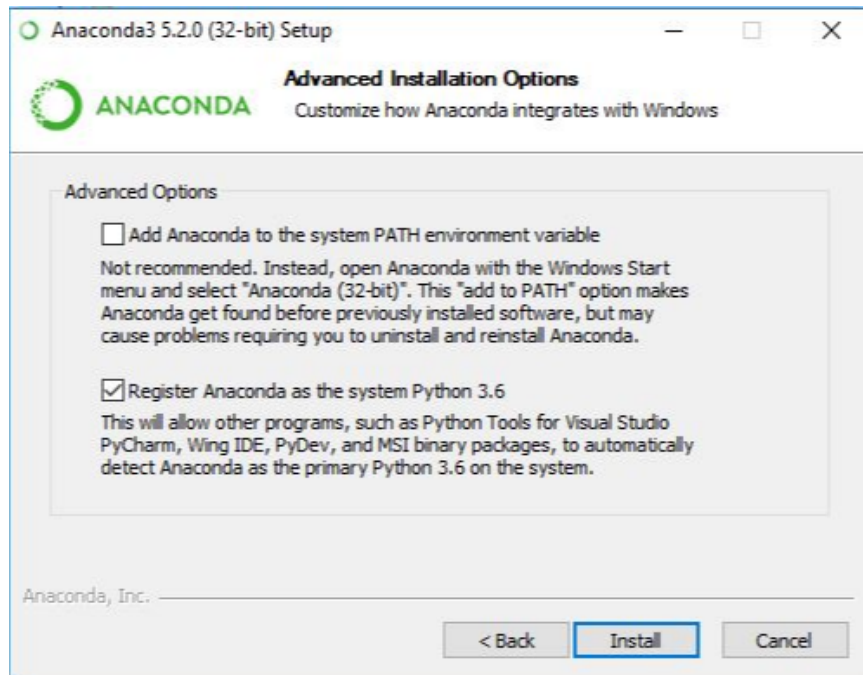


Figure 1.29: Langkah ketiga instalasi anaconda.

- `rng = np.random.RandomState(0)`
(rng sebagai estimator/parameter dengan nilai suatu itemnya yaitu `np.random.RandomState(0)`)
- `X = rng.rand(10, 2000)`
(X sebagai estimator/parameter dengan nilai item `rng.rand()`)
- `X = np.array(X, dtype='float32')`
(X sebagai estimator/parameter dengan nilai item `np.array()`)
- `X.dtype`
(`X.dtype` sebagai item pemanggil)
- `transformer = random_projection.GaussianRandomProjection()`
(transformer sebagai estimator/parameter dengan memanggil class `random_projection`)
- `X_new = transformer.fit_transform(X)`
(`X_new` di sini sebagai estimator/parameter dan menggunakan metode `fit`)
- `X_new.dtype`
(`X_new.dtype` sebagai item)
- `from sklearn import datasets`

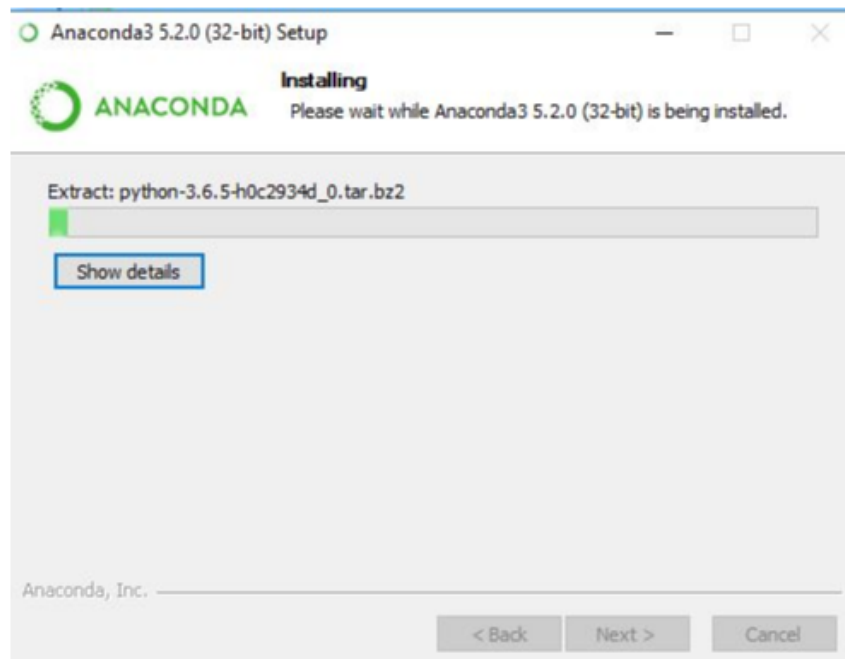


Figure 1.30: Langkah terakhir instalasi anaconda.

```
C:\Users\Asus-PC>pip install -U scikit-learn
Collecting scikit-learn
  Downloading https://files.pythonhosted.org/packages/ee/c8/c89ebdc8d7dbba6e6fd22daab257da3c28a967d07c35284272b2e1cefd/scikit_learn-0.20.2-cp36-cp36m-win32.whl (4.3MB)
    100% |#####| 4.3MB 660kB/s
Requirement not upgraded as not directly required: numpy>=1.8.2 in c:\programdata\anaconda3\lib\site-packages (from scikit-learn) (1.14.3)
Requirement not upgraded as not directly required: scipy>=0.13.3 in c:\programdata\anaconda3\lib\site-packages (from scikit-learn) (1.1.0)
Installing collected packages: scikit-learn
  Found existing installation: scikit-learn 0.19.1
  Uninstalling scikit-learn-0.19.1:
    Successfully uninstalled scikit-learn-0.19.1
Successfully installed scikit-learn-0.20.2
You are using pip version 19.0.1, however version 19.0.3 is available.
You should consider upgrading via the 'python -m pip install --upgrade pip' command.
```

Figure 1.31: Langkah pertama instalasi scikit pada CMD.

(pada baris ini merupakan sebuah perintah untuk mengimport class datasets dari packaged sklearn).

- `from sklearn.svm import SVC`

(pada baris ini merupakan sebuah perintah untuk mengimport class SVC dari packaged sklearn.svm).

- `iris = datasets.load_iris()`

(iris sebagai estimator/parameter dengan item datasets.load_iris()).

- `clf = SVC(gamma='scale')`

(clf sebagai estimator/parameter dengan nilai class SVC pada parameter gamma sebagai set penilaian).

- `clf.fit(iris.data, iris.target)`

(estimator/parameter clf menggunakan metode fit dengan itemnya)

```

C:\Users\Asus-PC>conda install scikit-learn
Solving environment: done

## Package Plan ##

  environment location: C:\ProgramData\Anaconda3

  added / updated specs:
    - scikit-learn

The following packages will be downloaded:

  package-----|-----build
  conda-4.6.7      |      py36_0      1.7 MB

The following packages will be UPDATED:

  conda: 4.5.4-py36_0 --> 4.6.7-py36_0

```

Figure 1.32: Langkah ketiga instalasi conda scikit pada CMD.

```

Proceed ([y]/n)? y
Downloading and Extracting Packages
conda-4.6.7      | 1.7 MB | ##### | 100%
Preparing transaction: done
Verifying transaction: done
Executing transaction: done

```

Figure 1.33: Langkah kedua pilih y.

- `list(clf.predict(iris.data[:3]))`
(menambahkan item list dengan metode predict)
- `clf.fit(iris.data, iris.target_names[iris.target])`
(estimator/parameter clf menggunakan metode fit dengan itemnya)
- `list(clf.predict(iris.data[:3]))`(menambahkan item list dengan metode pr

2. Refitting and Updating Parameters

- `import numpy as np`
(pada baris ini merupakan sebuah perintah untuk mengimport class svm dari np).
- `from sklearn.svm import SVC`
(pada baris ini merupakan sebuah perintah untuk mengimport class SVC dari packaged sklearn.svm).
- `rng = np.random.RandomState(0)`
(rng sebagai estimator/parameter dengan nilai suatu itemnya yaitu np.random.RandomS

```

C:\Users\Asus-PC>conda --version
conda 4.6.7

C:\Users\Asus-PC>python --version
Python 3.6.5 :: Anaconda, Inc.

C:\Users\Asus-PC>

```

Figure 1.34: Langkah cek version yang diinstall.

```

C:\Users\Asus-PC>python
Python 3.6.5 [Anaconda, Inc.] (default, Mar 29 2018, 13:23:52) [MSC v.1900 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> from sklearn.metrics import confusion_matrix
>>> y_true = [2, 0, 2, 2, 0, 1]
>>> y_pred = [0, 0, 2, 2, 0, 2]
>>> confusion_matrix(y_true, y_pred)
array([[2, 0, 0],
       [0, 0, 1],
       [1, 0, 2]], dtype=int64)
>>> y_true = ["cow", "goat", "cow", "goat", "horse"]
>>> y_pred = ["goat", "goat", "cow", "goat", "cow"]
>>> confusion_matrix(y_true, y_pred, labels=["cow", "goat", "horse"])
array([[1, 1, 0],
       [0, 2, 0],
       [1, 0, 0]], dtype=int64)
>>> ab, fa, fb, aa = confusion_matrix([0, 1, 0, 1], [1, 1, 1, 0]).ravel()
>>> (ab, fa, fb, aa)
(0, 2, 1, 1)
>>>

```

Figure 1.35: Hasil Tampilan 1.

- `X = rng.rand(100, 10)`
(X sebagai estimator/parameter dengan nilai item `rng.rand`).
- `y = rng.binomial(1, 0.5, 100)`
(y sebagai estimator/parameter dengan nilai item `rng.binomial`).
- `X_test = rng.rand(5, 10)`
(X_test sebagai estimator/parameter dengan nilai item `rng.rand`).
- `clf = SVC()`
(clf sebagai estimator/parameter dan class SVC)
- `clf.set_params(kernel='linear').fit(X, y)`
(set_params sebagai item)
- `clf.predict(X_test)`
(menggunakan metode predict)
- `clf.set_params(kernel='rbf', gamma='scale').fit(X, y)`
- `clf.predict(X_test)`

```
>>> y_true = ["cow", "goat", "cow", "goat", "horse"]
>>> y_pred = ["goat", "goat", "cow", "goat", "cow"]
>>> confusion_matrix(y_true, y_pred, labels=["cow", "goat", "horse"])
array([[1, 1, 0],
       [0, 2, 0],
       [1, 0, 0]], dtype=int64)
```

Figure 1.36: Hasil Tampilan 2.

```
>>> ab, fa, fb, aa = confusion_matrix([0, 1, 0, 1], [1, 1, 1, 0]).ravel()
>>> (ab, fa, fb, aa)
(0, 2, 1, 1)
```

Figure 1.37: Hasil Tampilan 3.

3. Multiclass vs. Multilabel Fitting

- `from sklearn.svm import SVC`
(pada baris ini merupakan sebuah perintah untuk mengimport class SVC dari packaged sklearn.svm).
- `from sklearn.multiclass import OneVsRestClassifier`
(pada baris ini merupakan sebuah perintah untuk mengimport class OneVsRestClassifier dari packaged sklearn.multiclass).
- `from sklearn.preprocessing import LabelBinarizer`
(pada baris ini merupakan sebuah perintah untuk mengimport class LabelBinarizer dari packaged sklearn.preprocessing).
- `X = [[1, 2], [2, 4], [4, 5], [3, 2], [3, 1]]`
- `y = [0, 0, 1, 1, 2]`
- `classif = OneVsRestClassifier(estimator=SVC(gamma='scale', random_state=0))`
- `classif.fit(X, y).predict(X)`
- `y = LabelBinarizer().fit_transform(y)`
- `classif.fit(X, y).predict(X)`
- `from sklearn.preprocessing import MultiLabelBinarizer`
- `y = [[0, 1], [0, 2], [1, 3], [0, 2, 3], [2, 4]]`
- `y = MultiLabelBinarizer().fit_transform(y)`
- `classif.fit(X, y).predict(X)`


```

>>> from joblib import dump, load
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ModuleNotFoundError: No module named 'joblib'
>>> dump(clf, 'filename.joblib')
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'dump' is not defined
>>> clf = load('filename.joblib')
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'load' is not defined

```

Figure 1.38: Hasil Tampilan Error.

1.7.1 Penanganan eror

1.7.1.1 ScreenShoot Error

1.7.1.2 Tuliskan Kode Error dan Jenis Erornya

- from joblib import dump, load

(Kode baris pertama)

```

Traceback(most recent call last):
  File "<stdin>", line 1, in<module>
ModuleNotFoundError: No module named 'joblib'

```

(Errornya)

- dump(clf, 'filename.joblib')

(Kode baris kedua)

```

Traceback(most recent call last):
  File "<stdin>", line 1, in<module>
NameError: name 'dump' is not defined

```

(Errornya)

- `clf = load('filename.joblib')`

(Kode baris ketiga)

```
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'load' is not defined
```

(Errornya)

1.7.1.3 Solusi Pemecahan Masalah Error

1. Pada masalah error sebelumnya itu dikarenakan kita belum mempunyai packaged joblib. Jadi solusinya yaitu dengan cara menginstall terlebih dahulu packaged joblibnya setelah itu baru perintah tersebut dapat dijalankan

```
>>> from joblib import dump, load
>>> dump(clf, 'filename.joblib')
['filename.joblib']
>>> clf = load('filename.joblib')
>>>
```

Figure 1.39: Hasil Tampilan Uji coba perintah joblib.

lllllll 78cdb6514db9716f252ef2024a7df6097aace611

Chapter 2

Related Works

Your related works, and your purpose and contribution which must be different as below.

2.1 Aip Suprpto Munari/1164063

2.1.1 Teori

2.1.2 Binary Classification

1. Binary Classification atau diartikan kedalam bahasa indonesia yaitu Klasifikasi Biner adalah tugas dalam mengkalrifikasikan elemen-elemen dari himpunan yang diberikan kedalam dua kelompok berdasarkan aturan klarifikasi. Pada ummnya klarifikasi biner akan jatuh ke dalam domain Supervised Learning dan dimana kasus khusus hanya memiliki dua kelas. Beberapa contoh yang meliputi Binary Classification adalah

- Deteksi Transaksi Penipuan Kartu Kredit
- Diagnosa medis
- Deteksi Spam

Untuk contoh Binary Classification dapat dilihat pada gambar ??

2.1.3 Supervised Learning, Unsupervised Learning, Dan Classter-ing

1. Supervised Learning merupakan sebuah pendekatan yang dimana sudah adanya sdata yang dilatih dan telah terdapat variabel yang telah ditargetkan sehingga

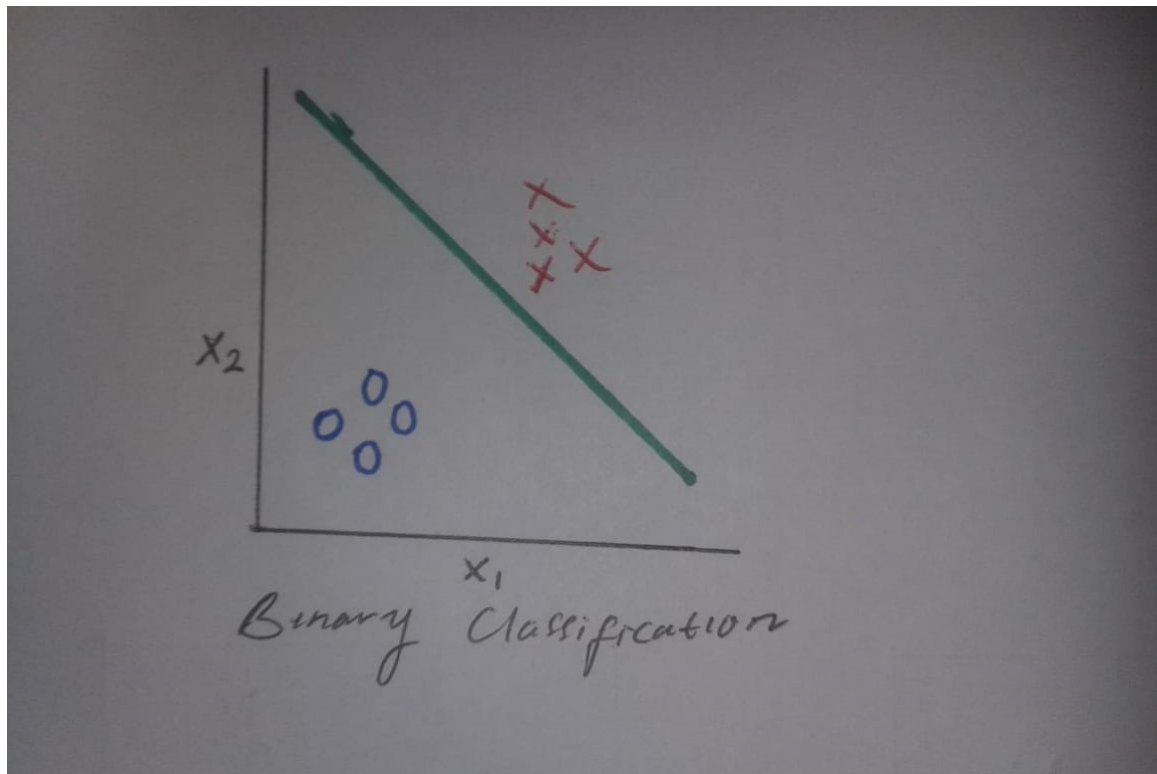


Figure 2.1: Binary Classification.

bertujuan untuk mengelompokkan suatu data ke data yang sudah ada. Contoh dalam Supervised Learning yaitu ketika anda memiliki sejumlah buku yang telah dilabel dengan urutan kategori tertentu. Ketika anda akan membeli sebuah buku baru, maka harus diidentifikasi isi dari buku tersebut dan memasukkannya kedalam kategori tertentu. Ketika anda membeli sebuah buku tersebut maka anda telah menerapkan sebuah logika fuzzy. Ilustrasi Supervised Learning dapat dilihat pada gambar ??.

2. Unsupervised Learning merupakan sebuah data yang belum ditentukan variabelnya jadi hanya berupa data saja. Dalam sebuah kasus Unsupervised Learning adalah anggap saja anda belum pernah membeli buku sama sekali dan pada suatu hari anda telah membeli buku dengan sangat banyak dalam kategori yang berbeda. Sehingga buku tersebut belum di kategorikan dan hanya berupa data buku saja. Ilustrasi Unsupervised Learning dapat dilihat pada gambar ??.
3. Clustering merupakan sebuah proses untuk mengklasifikasikan sebuah data dalam satu parameter. Dalam kasus ini dapat dijelaskan ada beberapa orang yang memiliki kekuatan tubuh yang sehat dan kekuatan tubuh yang lemah.

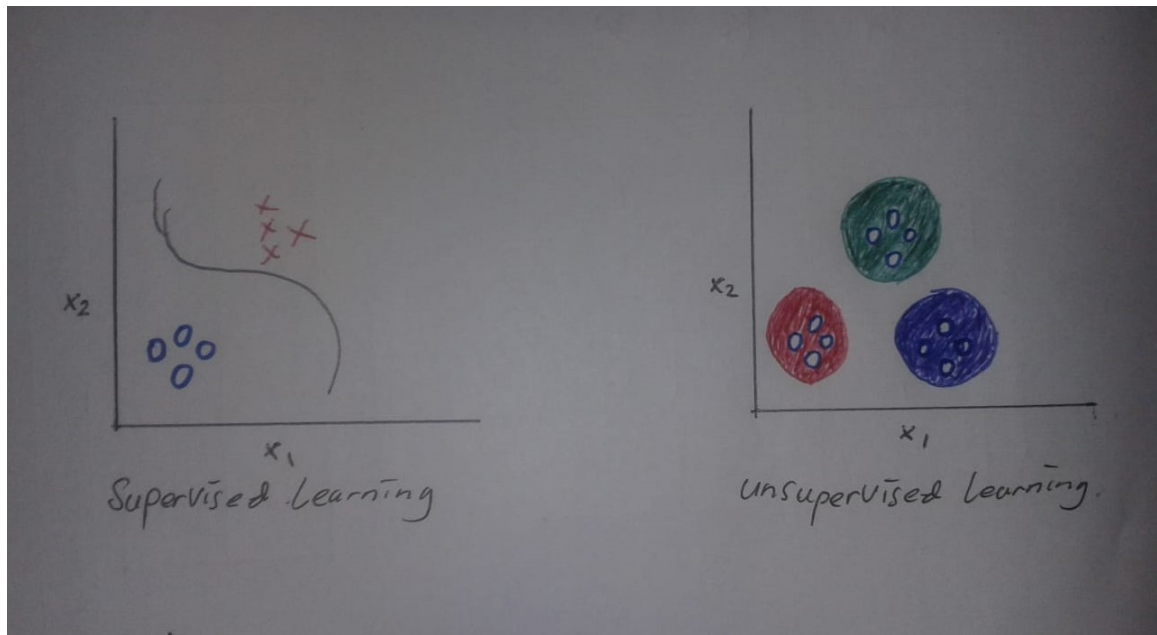


Figure 2.2: Supervised Learning.

Parameter bagi orang yang memiliki tubuh yang kuat adalah orang yang terlihat bugar dan sehat maka dengan orang yang memiliki parameter adalah orang yang memiliki kekuatan tubuh yang kuat dan untuk kekuatan tubuh yang lemah adalah sebaliknya. Ilustrasi gambar dapat di lihat di gambar ??

2.1.4 Evaluasi Dan Akurasi

1. Evaluasi dan akurasi adalah bagaimana cara kita dapat mengevaluasi seberapa baik model melakukan pekerjaannya dengan cara mengukur akurasi. Akurasi akan didefinisikan sebagai presentase kasus yang telah diklasifikasikan dengan benar. Kita dapat melakukan analisis kesalahan yang telah di buat oleh model. Dalam tabel tersebut baris true mangga dan true anggur menunjukkan kasus apakah itu objek mangga atau anggur. Kolom telah di prediksi dan dibuat oleh model. Ada 20 mangga yang di prediksi benar dan ada 5 anggur yang di prediksi salah. Ilustrasi dapat di lihat pada gambar ??

2.1.5 Confusion Matrix

1. Ada beberapa cara untuk membuat dan membaca confusion matrix antara lain
 - Tentukan pokok permasalahan serta atributnya
 - Buat Decision Tree

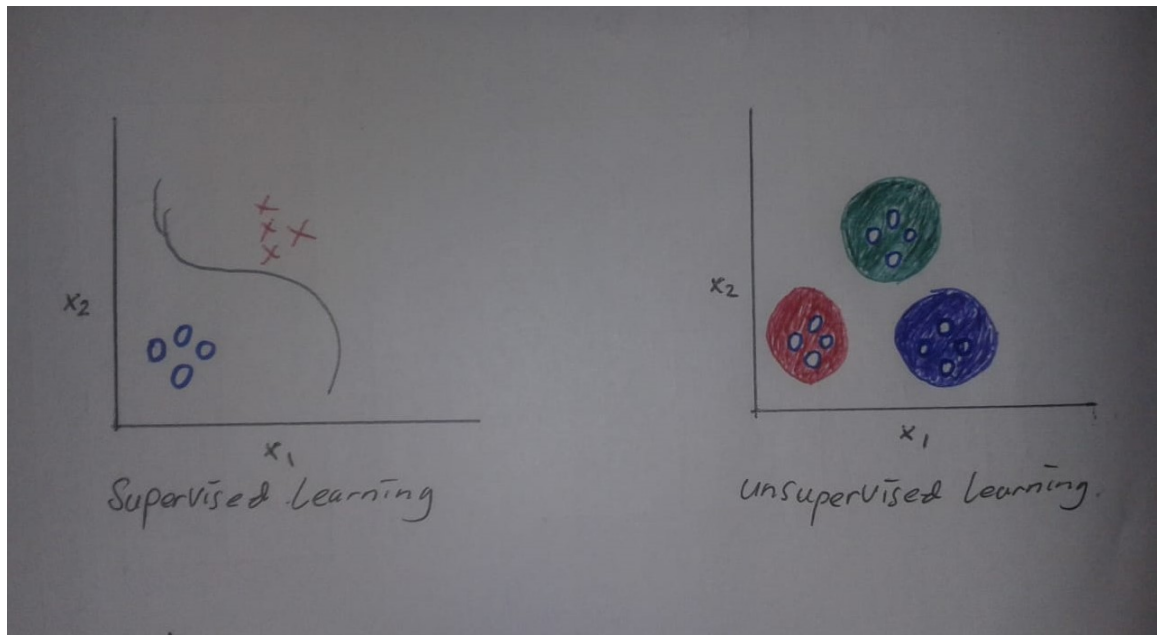


Figure 2.3: Unsupervised Learning.

- Buat Data Testing
- Mencari nilai variabelnya misal a,b,c, dan d
- Mencari nilai recall, precision, accuracy, dan error rate

Di bawah ini adalah contoh dari confusion matrix

$$\text{Recall} = 3/(1+3) = 0,75$$

$$\text{Precision} = 3/(1+3) = 0,75$$

$$\text{Accuracy} = (5+3)/(5+1+1+3) = 0,8$$

$$\text{Error Rate} = (1+1)/(5+1+1+3) = 0,2$$

2.1.6 Cara Kerja K-Fold Cross Validation

1. Untuk cara kerja K-Fold Cross Validation adalah sebagai berikut

- Total instance dibagi menjadi N bagian.
- Fold yang pertama adalah bagian pertama menjadi testing data dan sisanya menjadi training data.
- Hitung akurasi berdasarkan porsi data tersebut dengan menggunakan persamaan.

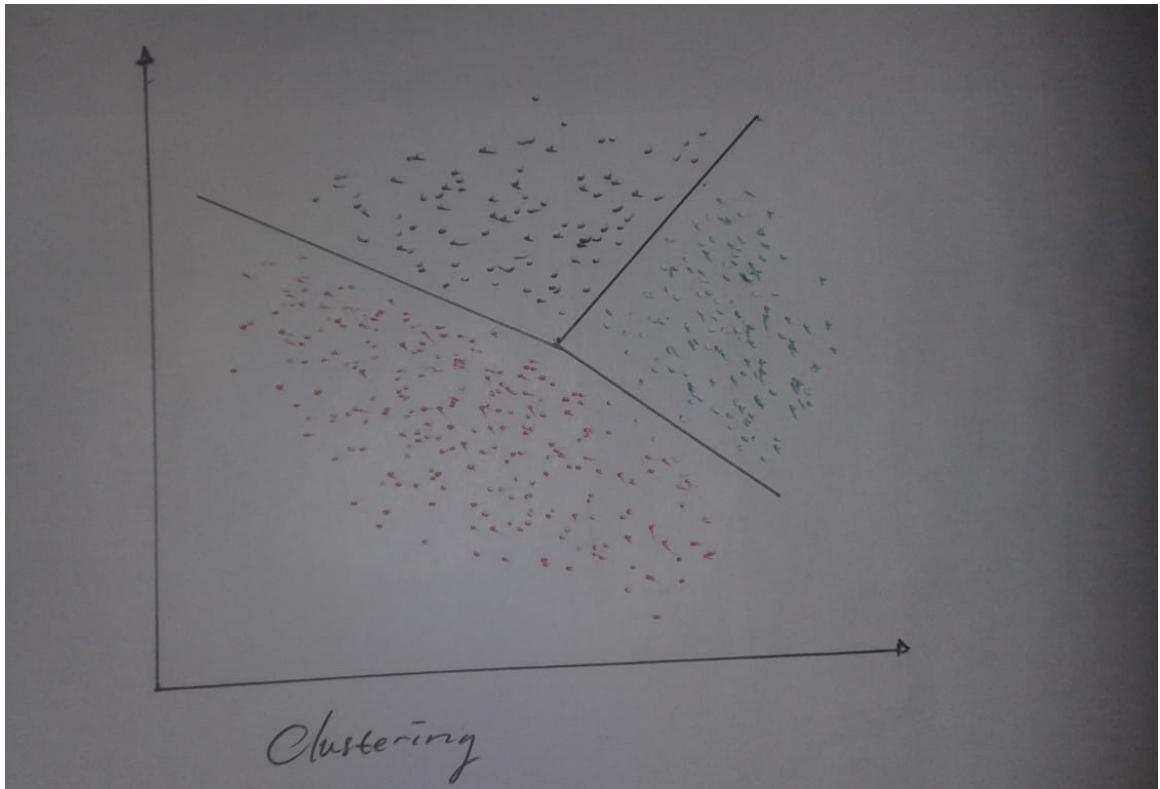


Figure 2.4: Clusterring.

- Fold yang ke dua adalah bagian ke dua menjadi testing data dan sisanya training data.
- Hitung akurasi berdasarkan porsi data tersebut.
- Lakukan step secara berulang hingga habis mencapai fold ke-K.
- Terakhir hitung rata-rata akurasi K buah.

Untuk ilustrasi K-Fold Cross Validation data di lihat pada gambar ??

2.1.7 Decision Tree

1. Decision Tree adalah sebuah metode pembelajaran yang digunakan untuk melakukan klarifikasi dan regresi. Decision Tree digunakan untuk membuat sebuah model yang dapat memprediksi sebuah nilai variabel target dengan cara mempelajari aturan keputusan dari fitur data. Contoh Decision Tree adalah untuk melakukan prediksi apakah Kuda termasuk hewan mamalia atau bukan, lihat pada gambar ??.

	<u>predicted cow</u>	<u>predicted buffalo</u>
<u>True cow</u>	15	5
<u>True buffalo</u>	7	13

Figure 2.5: Evaluasi Dan Akurasi.

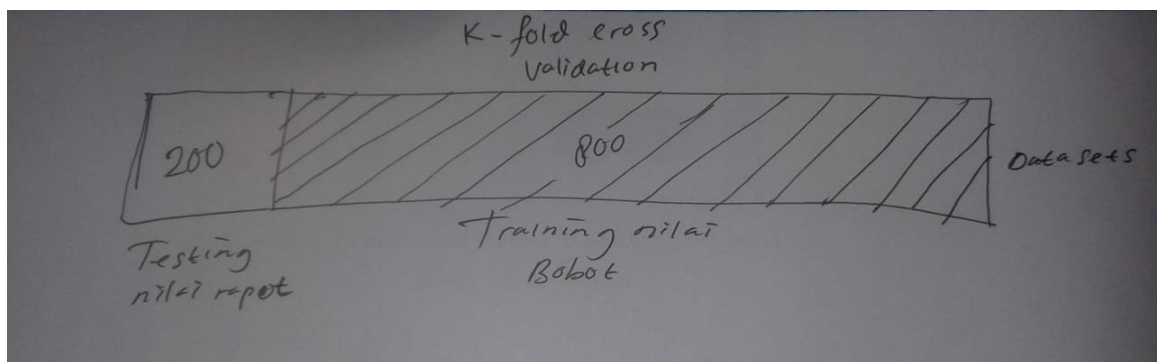


Figure 2.6: K-Fold Cross Validation.

2.1.8 Gain Dan Entropi

1. Gain adalah pengurangan yang diharapkan dalam entrophy. Dalam mechine learning, gain dapat digunakan untuk menentukan sebuah urutan atribut atau memperkecil atribut yang telah dipilih. Urutan ini akan membentuk decision tree. atribut gain dipilih yang paling besar.
2. Entropi adalah ukuran ketidakpastian sebuah variabel acak sehingga dapat di artikan entropi adalah ukuran ketidakpastian dari sebuah atribut.

Ilustrasi dari gain dan entropi adalah bagaimana kita memprediksi jenis kelamin berdasarkan atributnya, perhatikan pada gambar ??

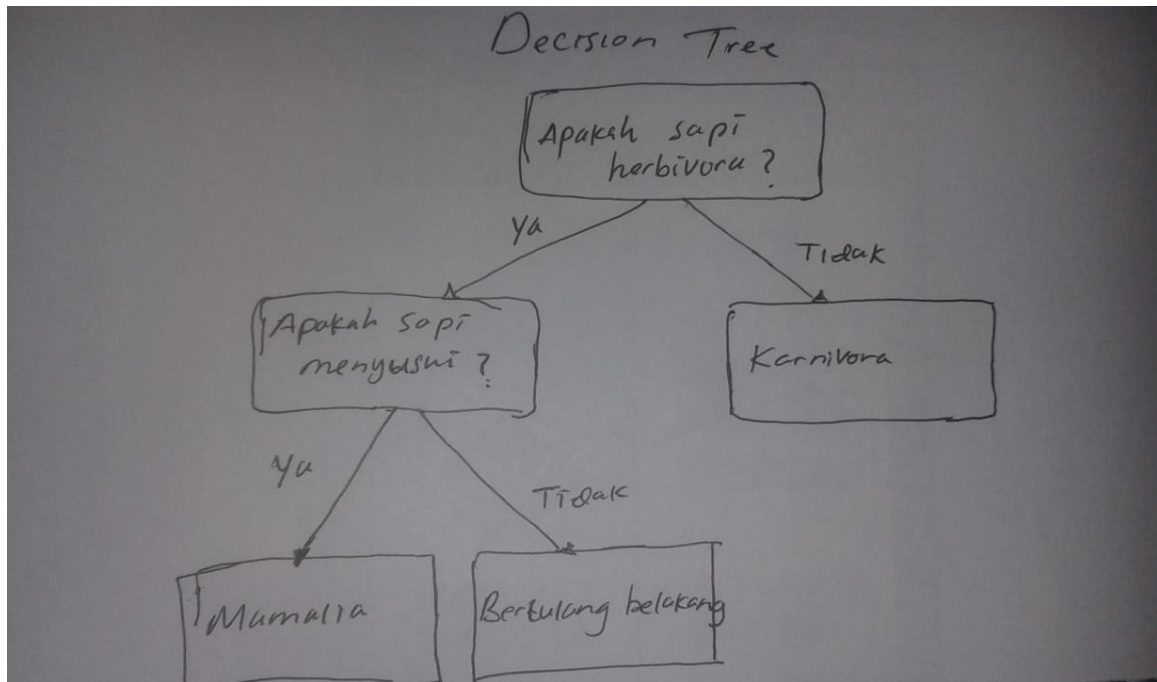


Figure 2.7: Decision Tree.

2.2 Same Topics

Cite every latest journal with same topic

2.2.1 Topic 1

cite for first topic

2.2.2 Topic 2

if you have two topics you can include here to

2.3 Same Method

write and cite latest journal with same method

2.3.1 Method 1

cite and paraphrase method 1

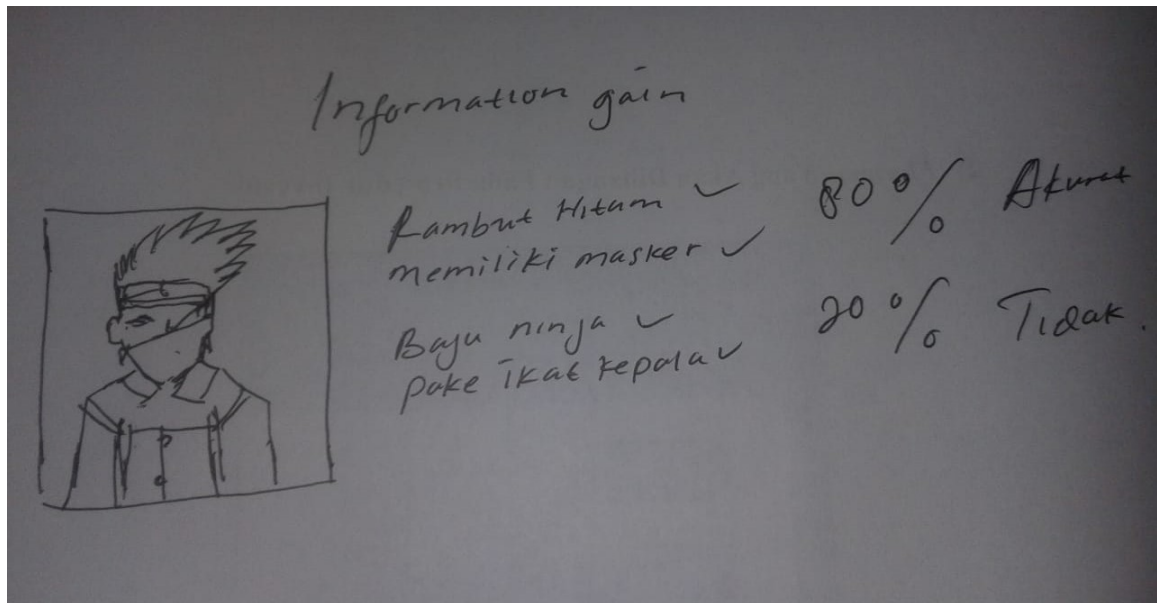


Figure 2.8: Gain Dan Entropi.

2.3.2 Method 2

cite and paraphrase method 2 if you have more method please add new subsection.

Chapter 3

Methods

3.1 The data

Please tell where is the data come from, a little brief of company can be put here.

3.2 Method 1

Definition, steps, algorithm or equation of method 1 and how to apply into your data

3.3 Method 2

Definition, steps, algorithm or equation of method 2 and how to apply into your data

Chapter 4

Experiment and Result

brief of experiment and result.

4.1 Experiment

Please tell how the experiment conducted from method.

4.2 Result

Please provide the result of experiment

Chapter 5

Conclusion

brief of conclusion

5.1 Conclusion of Problems

Tell about solving the problem

5.2 Conclusion of Method

Tell about solving using method

5.3 Conclusion of Experiment

Tell about solving in the experiment

5.4 Conclusion of Result

tell about result for purpose of this research.

Chapter 6

Discussion

Please tell more about conclusion and how to the next work of this study.

Chapter 7

Discussion

Please tell more about conclusion and how to the next work of this study.

Chapter 8

Discussion

Please tell more about conclusion and how to the next work of this study.

Chapter 9

Discussion

Please tell more about conclusion and how to the next work of this study.

Chapter 10

Discussion

Please tell more about conclusion and how to the next work of this study.

Chapter 11

Discussion

Please tell more about conclusion and how to the next work of this study.

Chapter 12

Discussion

Please tell more about conclusion and how to the next work of this study.

Chapter 13

Discussion

Please tell more about conclusion and how to the next work of this study.

Chapter 14

Discussion

Please tell more about conclusion and how to the next work of this study.

Appendix A

Form Penilaian Jurnal

gambar A.1 dan A.2 merupakan contoh bagaimana reviewer menilai jurnal kita.

NO	UNSUR	KETERANGAN	MAKS	KETERANGAN
1	Keefektifan Judul Artikel	Maksimal 12 (dua belas) kata dalam Bahasa Indonesia atau 10 (sepuluh) kata dalam Bahasa Inggris	2	a. Tidak lugas dan tidak ringkas (0) b. Kurang lugas dan kurang ringkas (1) c. Ringkas dan lugas (2)
2	Pencantuman Nama Penulis dan Lembaga Penulis		1	a. Tidak lengkap dan tidak konsisten (0) b. Lengkap tetapi tidak konsisten (0,5) c. Lengkap dan konsisten (1)
3	Abstrak	Dalam Bahasa Indonesia dan Bahasa Inggris yang baik, jumlah 150-200 kata. Isi terdiri dari latar belakang, metode, hasil, dan kesimpulan. Isi tertuang dengan kalimat yang jelas.	2	a. Tidak dalam Bahasa Indonesia dan Bahasa Inggris (0) b. Abstrak kurang jelas dan ringkas, atau hanya dalam Bahasa Inggris, atau dalam Bahasa Indonesia saja (1) c. Abstrak yang jelas dan ringkas dalam Bahasa Indonesia dan Bahasa Inggris (2)
4	Kata Kunci	Maksimal 5 kata kunci terpenting dalam paper	1	a. Tidak ada (0) b. Ada tetapi kurang mencerminkan konsep penting dalam artikel (0,5) c. Ada dan mencerminkan konsep penting dalam artikel (1)
5	Sistematika Pembahasan	Terdiri dari pendahuluan, tinjauan pustaka, metode penelitian, hasil dan pembahasan, kesimpulan dan saran, daftar pustaka	1	a. Tidak lengkap (0) b. Lengkap tetapi tidak sesuai sistematika (0,5) c. Lengkap dan bersistem (1)
6	Pemanfaatan Instrumen Pendukung	Pemanfaatan Instrumen Pendukung seperti gambar dan tabel	1	a. Tidak dimanfaatkan (0) b. Kurang informatif atau komplementer (0,5) c. Informatif dan komplementer (1)
7	Cara Pengacuan dan Pengutipan		1	a. Tidak baku (0) b. Kurang baku (0,5) c. Baku (1)
8	Penyusunan Daftar Pustaka	Penyusunan Daftar Pustaka	1	a. Tidak baku (0) b. Kurang baku (0,5) c. Baku (1)
9	Peristilahan dan Kebahasaan		2	a. Buruk (0) b. Baik (1) c. Cukup (2)
10	Makna Sumbangan bagi Kemajuan		4	a. Tidak ada (0) b. Kurang (1) c. Sedang (2) d. Cukup (3) e. Tinggi (4)

Figure A.1: Form nilai bagian 1.

11	Dampak Ilmiah		7	a. Tidak ada (0) b. Kurang (1) c. Sedang (3) d. Cukup (5) e. Besar (7)
12	Nisbah Sumber Acuan Primer berbanding Sumber lainnya	Sumber acuan yang langsung merujuk pada bidang ilmiah tertentu, sesuai topik penelitian dan sudah teruji.	3	a. < 40% (1) b. 40-80% (2) c. > 80% (3)
13	Derajat Kemutakhiran Pustaka Acuan	Derajat Kemutakhiran Pustaka Acuan	3	a. < 40% (1) b. 40-80% (2) c. > 80% (3)
14	Analisis dan Sintesis	Analisis dan Sintesis	4	a. Sedang (2) b. Cukup (3) c. Baik (4)
15	Penyimpulan	Sangat jelas relevasinya dengan latar belakang dan pembahasan, dirumuskan dengan singkat	3	a. Kurang (1) b. Cukup (2) c. Baik (3)
16	Unsur Plagiat		0	a. Tidak mengandung plagiat (0) b. Terdapat bagian-bagian yang merupakan plagiat (-5) c. Keseluruhannya merupakan plagiat (-20)
TOTAL			36	
Catatan : Nilai minimal untuk diterima 25				

Figure A.2: form nilai bagian 2.

Appendix B

FAQ

M : Kalo Intership II atau TA harus buat aplikasi ? D : Ga harus buat aplikasi tapi harus ngoding

M : Pa saya bingung mau ngapain, saya juga bingung mau presentasi apa? D : Makanya baca de, buka jurnal topik ‘ganteng’ nah kamu baca dulu sehari 5 kali ya, 4 hari udah 20 tuh. Bingung itu tanda kurang wawasan alias kurang baca.

M : Pa saya sudah cari jurnal terindeks scopus tapi ga nemu. D : Kamu punya mata de? coba dicolok dulu. Kamu udah lakuin apa aja? tolong di list laporkan ke grup Tingkat Akhir. Tinggal buka google scholar klik dari tahun 2014, cek nama jurnalnya di scimagojr.com beres.

M : Pa saya belum dapat tempat intership, jadi ga tau mau presentasi apa? D : kamu kok ga nyambung, yang dipresentasikan itu yang kamu baca bukan yang akan kamu lakukan.

M : Pa ini jurnal harus yang terindex scopus ga bisa yang lain ? D : Index scopus menandakan artikel tersebut dalam standar semantik yang mudah dipahami dan dibaca serta bukan artikel asal jadi. Jika diluar scopus biasanya lebih sukar untuk dibaca dan dipahami karena tidak adanya proses review yang baik dan benar terhadap artikel.

M : Pa saya tidak mengerti D : Coba lihat standar alasan

M : Pa saya bingung D : Coba lihat standar alasan

M : Pa saya sibuk D : Mbahmu....

M : Pa saya ganteng D : Ndasmu....

M : Pa saya kece D : wes karepmu lah....

Biasanya anda memiliki alasan tertentu jika menghadapi kendala saat proses bimbingan, disini saya akan melakukan standar alasan agar persepsi yang diterima sama dan tidak salah kaprah. Penggunaan kata alasan tersebut antara lain :

1. Tidak Mengerti : anda boleh menggunakan alasan ini jika anda sudah melakukan tahapan membaca dan meresumekan 15 jurnal. Sudah mencoba dan mempraktekkan teorinya dengan mencari di youtube dan google minimal 6 jam sehari selama 3 hari berturut-turut.

2. Bingung : anda boleh mengatakan alasan bingung setelah maksimal dalam berusaha menyelesaikan tugas bimbingan dari dosen(sudah dilakukan semua). Anda belum bisa mengatakan alasan bingung jika anda masih belum menyelesaikan tugas bimbingan dan poin nomor 1 diatas. Setelah anda menyelesaikan tugas bimbingan secara maksimal dan tahap 1 poin diatas, tapi anda masih tetap bingung maka anda boleh memakai alasan ini.

Bibliography

- [1] Cahyo Darujati and Agustinus Bimo Gumelar. Pemanfaatan teknik supervised untuk klasifikasi teks bahasa indonesia. *Jurnal Bandung Text Mining*, 16(1):5–1, 2012.
- [2] Joshua Eckroth. *Python Artificial Intelligence Projects for Beginners: Get up and running with Artificial Intelligence using 8 smart and exciting AI applications*. Packt Publishing Ltd, 2018.
- [3] Helfi Nasution. Implementasi logika fuzzy pada sistem kecerdasan buatan. *ELKHA*, 4(2), 2012.
- [4] Stuart J Russell and Peter Norvig. *Artificial intelligence: a modern approach*. Malaysia; Pearson Education Limited,, 2016.