

# **CERDAS MENGUASAI GIT**



---

# CERDAS MENGUASAI GIT

## Dalam 24 Jam

---

**Rolly M. Awangga**  
Informatics Research Center



Kreatif Industri Nusantara

**Penulis:**

Rolly Maulana Awangga

ISBN : 978-602-53897-0-2

**Editor:**

M. Yusril Helmi Setyawan

**Penyunting:**

Syafrial Fachrie Pane

Khaera Tunnisa

Diana Asri Wijayanti

**Desain sampul dan Tata letak:**

Deza Martha Akbar

**Penerbit:**

Kreatif Industri Nusantara

**Redaksi:**

Jl. Ligar Nyawang No. 2

Bandung 40191

Tel. 022 2045-8529

Email : awangga@kreatif.co.id

**Distributor:**

Informatics Research Center

Jl. Sariasih No. 54

Bandung 40151

Email : irc@poltekpos.ac.id

Cetakan Pertama, 2019

Hak cipta dilindungi undang-undang

Dilarang memperbanyak karya tulis ini dalam bentuk dan dengan cara  
apapun tanpa ijin tertulis dari penerbit

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

# CONTRIBUTORS

---

ROLLY MAULANA AWANGGA, Informatics Research Center., Politeknik Pos  
Indonesia, Bandung, Indonesia



# CONTENTS IN BRIEF

---





# DAFTAR ISI

---



# DAFTAR GAMBAR

---



# DAFTAR TABEL

---



## Listings

---





# FOREWORD

---

Sepatah kata dari Kaprodi, Kabag Kemahasiswaan dan Mahasiswa



# KATA PENGANTAR

---

Buku ini diciptakan bagi yang awam dengan git sekalipun.

R. M. AWANGGA

*Bandung, Jawa Barat*  
*Februari, 2019*



# ACKNOWLEDGMENTS

---

Terima kasih atas semua masukan dari para mahasiswa agar bisa membuat buku ini lebih baik dan lebih mudah dimengerti.

Terima kasih ini juga ditujukan khusus untuk team IRC yang telah fokus untuk belajar dan memahami bagaimana buku ini mendampingi proses Inter-ship.

R. M. A.



# ACRONYMS

---

ACGIH	American Conference of Governmental Industrial Hygienists
AEC	Atomic Energy Commission
OSHA	Occupational Health and Safety Commission
SAMA	Scientific Apparatus Makers Association





# GLOSSARY

---

git	Merupakan manajemen sumber kode yang dibuat oleh linus torvald.
bash	Merupakan bahasa sistem operasi berbasiskan *NIX.
linux	Sistem operasi berbasis sumber kode terbuka yang dibuat oleh Linus Torvald



# SYMBOLS

---

- $A$  Amplitude
- $\&$  Propositional logic symbol
- $a$  Filter Coefficient
  
- $\mathcal{B}$  Number of Beats



# INTRODUCTION

---

ROLLY MAULANA AWANGGA, S.T., M.T.

Informatics Research Center  
Bandung, Jawa Barat, Indonesia

Pada era disruptif saat ini. git merupakan sebuah kebutuhan dalam sebuah organisasi pengembangan perangkat lunak. Buku ini diharapkan bisa menjadi penghantar para programmer, analis, IT Operation dan Project Manajer. Dalam melakukan implementasi git pada diri dan organisasinya.

Rumusnya cuman sebagai contoh aja biar keren[?].

$$ABC\mathcal{DEF}\alpha\beta\Gamma\Delta\sum_{def}^{abc} \tag{I.1}$$



# BAB 1

---

## CHAPTER 1

---

### 1.1 1174006 - Kadek Diva Krishna Murti

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

```
1 @inproceedings{awangga2017colenak ,
2   title={Colenak: GPS tracking model for post-stroke
3     rehabilitation program using AES-CBC URL encryption and QR-
4     Code},
5   author={Awangga, Rolly Maulana and Fathonah, Nuraini Siti and
6     Hasanudin, Trisna Irmayadi},
7   booktitle={Information Technology, Information Systems and
8     Electrical Engineering (ICITISEE), 2017 2nd International
9     conferences on},
10  pages={255--260},
11  year={2017},
12  organization={IEEE}
13 }
```





**Gambar 1.1** Kecerdasan Buatan.

1. Lorem ipsum dolor sit amet, consectetur adipiscing elit.
2. Lorem ipsum dolor sit amet, consectetur adipiscing elit.
3. Lorem ipsum dolor sit amet, consectetur adipiscing elit.

### **1.1.1 Teori**

### **1.1.2 Praktek**

### **1.1.3 Penanganan Error**

### **1.1.4 Bukti Tidak Plagiat**



**Gambar 1.2** Kecerdasan Buatan.

## **1.2 1174035 - Luthfi Muhammad Nabil**

Kecerdasan buatan merupakan kecerdasan yang dimasukkan ke sistem yang dapat diatur untuk kepentingan ilmiah. Kecerdasan buatan biasa disebut AI (Artificial Intelligence) yang didefinisikan sebagai kecerdasan ilmiah. AI memiliki kemampuan untuk menerjemahkan data dari luar, dan mempelajari data tersebut untuk dipelajari demi mencapai tujuan dan melakukan tugas tertentu sesuai hasil adaptasi berdasarkan data yang didapat.

### **1.2.1 Sejarah dan perkembangan Kecerdasan Buatan**

AI mulai berkembang sesuai dengan konsep yang dikemukakan pada awal abad 17, Rene Descartes menyebutkan bahwa tubuh hewan bukanlah apa-apa melainkan mesin-mesin yang rumit. Lalu Blaise Pascal menciptakan mesin perhitungan digital mekanis pertama pada 1642. Selanjutnya pada abad ke

19, Charles Babbage dan Ada Lovelace menciptakan sebuah mesin penghitung mekanis yang dapat diprogram.

Pada tahun 1950-an, Program AI pertama yang sudah dapat difungsikan telah ditulis pada 1951 untuk menjalankan mesin Ferranti Mark I di University of Manchester yang merupakan sebuah program permainan naskah yang ditulis oleh Christopher Strachey. John McCarthy menyebutkan istilah "kecerdasan buatan" pada konferensi pertama yang disediakan untuk persoalan ini. Dilanjut pada tahun 1956, Beliau menemukan bahasa pemrograman yang bernama Lisp.

Di tahun 1960-1970 an, Joel Moses menerangkan kekuatan pertimbangan simbolis untuk mengimplementasikan masalah di dalam program Macsyma, program berbasis pengetahuan yang sudah sukses dalam bidang matematika. Lalu di tahun 1980-an, jaringan saraf mulai digunakan secara luas menggunakan algoritma perambatan balik, pertama kali dijelaskan oleh Paul John Werbos pada tahun 1974. Selanjutnya pada tahun 1982, para ahli fisika menggunakan teknik - teknik statistika untuk menganalisis sifat dari penyimpanan dan optimasi pada jaringan saraf. Lalu dilanjutkan pada tahun 1985 sedikitnya empat kelompok riset menemukan algoritma pembelajaran propagansi balik. Algoritma ini berhasil diimplementasikan ke ilmu komputer dan psikologi. Dan pada tahun 1990, ditandai perolehan besar dalam berbagai bidang AI dan demonstrasi dari berbagai aplikasi yang sudah mengimplementasi. Seperti Deep Blue, sebuah komputer dari permainan catur yang dapat mengalahkan Garry Kasparov dalam sebuah pertandingan 6 game yang terkenal pada 1997.

### 1.2.2 Supervised Learning

Supervised learning adalah kondisi yang menggunakan variabel input dan output untuk dapat dilakukan pemetaan input output yang sudah didapat. Disebut Supervised Learning karena proses dari pembelajaran algoritma dari pembelajaran yang disumbangkan dengan dataset dapat dipikirkan seperti seorang guru yang mengawasi proses pembelajaran. Proses pembelajaran dari algoritma akan berhenti saat algoritma sudah mendapatkan level dari performansi yang dapat diterima.

Masalah dari Supervised learning dapat dikelompokkan menjadi masalah dengan regresi dan klasifikasi

- Klasifikasi : Masalah dalam klasifikasi yang dimana output dari variable itu adalah kategori, seperti "Laki - laki" atau "Perempuan, dan "Muda" dan "Tua"
- Regresi : Masalah dalam regresi adalah jika pengeluaran dari variabel adalah sebuah nilai asli, seperti "suhu", dan "tinggi"

### 1.2.3 Unsupervised Learning

Unsupervised learning adalah kondisi dimana kamu hanya memiliki input data tanpa memiliki variabel output yang sesuai. Tujuan dari unsupervised learning adalah untuk memodelkan distribusi pada data untuk mengetahui lebih lanjut mengenai data. Disebut unsupervised learning karena pada metode ini, tidak ada jawaban yang tepat dan tidak ada pengarah. Sehingga algoritma akan ditinggalkan sesuai rancangan demi menemukan dan dapat mengolah data yang menarik pada saat yang akan datang.

### 1.2.4 Jenis - Jenis Dataset

Dataset merupakan objek yang merepresentasikan data dan relasinya di memori. Strukturnya dapat mirip sesuai dengan struktur yang ada pada database namun bisa diubah sesuai dengan kebutuhan. Dataset juga berisi koleksi dari tabel data dan relasi data.

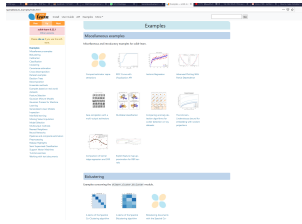
- Training set : merupakan sebuah dataset yang digunakan untuk kepentingan pembelajaran. Kepentingan tersebut akan disesuaikan dengan parameter yang ada.
- Test dataset : adalah sebuah dataset yang bersifat independen dibandingkan dengan training dataset, namun mengikuti probabilitas distribusi yang sama dengan training dataset. Jika model sudah sesuai dengan training dataset maka dataset sudah dapat disesuaikan dengan test dataset. Penyesuaian dari training dataset .

### 1.2.5 Instalasi dan Percobaan Kompilasi dari Library Scikit-learn

1. Buka anaconda prompt
2. Ketik di anaconda prompt yaitu : "pip install -U scikit-learn" untuk instalasi

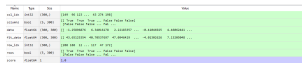
**Gambar 1.3** Instalasi Scikit Learn

3. Setelah selesai instalasi, pilih salah satu example dari website Scikit (Contoh : )



**Gambar 1.4** Daftar Example

4. Lalu coba jalankan aplikasi tersebut, bisa dicek hasil dari Variable explorernya



**Gambar 1.5** Variable Explorer

## 5. Sample kode

```

1 print(__doc__)
2
3 # Author: Kemal Eren <kemal@kemaleren.com>
4 # License: BSD 3 clause
5
6 import numpy as np
7 from matplotlib import pyplot as plt
8
9 from sklearn.datasets import make_biclusters
10 from sklearn.cluster import SpectralCoclustering
11 from sklearn.metrics import consensus_score
12
13 data, rows, columns = make_biclusters(
14     shape=(300, 300), n_clusters=5, noise=5,
15     shuffle=False, random_state=0)
16
17 plt.matshow(data, cmap=plt.cm.Blues)
18 plt.title("Original dataset")
19
20 # shuffle clusters
21 rng = np.random.RandomState(0)
22 row_idx = rng.permutation(data.shape[0])
23 col_idx = rng.permutation(data.shape[1])
24 data = data[row_idx][:, col_idx]
25
26 plt.matshow(data, cmap=plt.cm.Blues)
27 plt.title("Shuffled dataset")
28
29 model = SpectralCoclustering(n_clusters=5, random_state=0)
30 model.fit(data)

```

```

31 score = consensus_score(model.biclusters_,
32                          (rows[:, row_idx], columns[:, col_idx]
33                          )))
34 print("consensus score: {:.3f}".format(score))
35
36 fit_data = data[np.argsort(model.row_labels_)]
37 fit_data = fit_data[:, np.argsort(model.column_labels_)]
38
39 plt.matshow(fit_data, cmap=plt.cm.Blues)
40 plt.title("After biclustering; rearranged to show biclusters"
41          )
42 plt.show()

```

## 1.2.6 Mencoba Loading and example dataset

- Percobaan 1 (Memuat data iris dan digits dari datasets)

```

1 from sklearn import datasets #Untuk import class/fungsi
   dataset dari scikit-learn library
2 iris = datasets.load_iris() #Untuk memuat dan memasukkan
   dataset iris ke variabel bernama iris
3 digits = datasets.load_digits() #Untuk memuat dan memasukkan
   dataset digits ke variabel digits

```

```

In [18]: """
...: Created on Wed Feb 26 15:55:58 2020
...:
...: @author: Luthfi Muhammad Nabil
...: """
...:
...: from sklearn import datasets
...: iris = datasets.load_iris()
...: digits = datasets.load_digits()

```

Gambar 1.6 Hasil Percobaan 1

- Percobaan 2 (Menampilkan data dari digits)

```

1 print(digits.data) #Menampilkan object berformat Dictionary-
   like yang nanti akan ditampilkan pada console

```

```

In [19]: 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.]]

```

Gambar 1.7 Hasil Percobaan 2

- Percobaan 3 (Menampilkan digits.target)

```
1 digits.target #Menunjukkan data angka yang berhubungan dengan
                setiap digit gambar yang sedang dipelajari
```

```
In [20]: digits.target
Out[20]: array([0, 1, 2, ..., 8, 9, 8])
```

**Gambar 1.8** Hasil Percobaan 3

- Percobaan 4 (Menampilkan data 2 dimensi)

```
1 digits.images[0] #Akan mengambil data dengan berformat array
                   2 Dimensi, dengan bentuk parameter (n_samples, n_features)
```

```
In [22]: digits.images[0]
Out[22]:
array([[ 0.,  0.,  5., 13.,  9.,  1.,  0.,  0.],
       [ 0.,  0., 13., 15., 10., 15.,  5.,  0.],
       [ 0.,  3., 15.,  2.,  0., 11.,  8.,  0.],
       [ 0.,  4., 12.,  0.,  0.,  8.,  8.,  0.],
       [ 0.,  5.,  8.,  0.,  0.,  9.,  8.,  0.],
       [ 0.,  4., 11.,  0.,  1., 12.,  7.,  0.],
       [ 0.,  2., 14.,  5., 10., 12.,  0.,  0.],
       [ 0.,  0.,  6., 13., 10.,  0.,  0.,  0.]])
```

**Gambar 1.9** Hasil Percobaan 4

- Full sample

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Wed Feb 26 15:55:58 2020
4
5 @author: Luthfi Muhammad Nabil
6 """
7
8 from sklearn import datasets #Untuk import class/fungsi
9                               dataset dari scikit-learn library
10 iris = datasets.load_iris() #Untuk memuat dan memasukkan
11                               dataset iris ke variabel bernama iris
12 digits = datasets.load_digits() #Untuk memuat dan memasukkan
13                               dataset digits ke variabel digits
14
15 #%%
16 print(digits.data) #Menampilkan object berformat Dictionary-
17                     like yang nanti akan ditampilkan pada console
18 #%%
19 digits.target #Menunjukkan data angka yang berhubungan dengan
20               setiap digit gambar yang sedang dipelajari
21 #%%
22 digits.images[0] #Akan mengambil data dengan berformat array
23                 2 Dimensi, dengan bentuk parameter (n_samples, n_features)
```



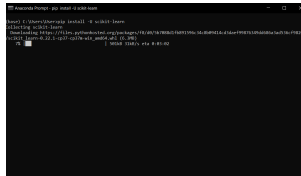
6. ecerdasan Buatan menjadi sebuah industri pada tahun 1980-1988. Kembalinya Jaringan Syaraf tiruan pada tahun 1986-sekarang.

### 1.3.2 Instalasi

Membuka <https://scikit-learn.org/stable/tutorial/basic/tutorial.html> lalu mencobanya.

#### 1.3.2.1 Instalasi library scikit, mencoba kompilasi dan ujicoba contoh kode

1. Buka anaconda prompt lalu ketikkan "pip install -U scikit-learn" untuk menginstall library scikit



**Gambar 1.11** Install Library Scikit

2. Pilih salah satu example dari website tersebut lalu jalankan

```

1  # -*- coding: utf-8 -*-
2  "" ""
3  Created on Wed Feb 26 18:16:44 2020
4
5  @author: User
6  "" ""
7
8  import matplotlib.pyplot as plt
9  from sklearn.svm import SVC
10 from sklearn.ensemble import RandomForestClassifier
11 from sklearn.metrics import plot_roc_curve
12 from sklearn.datasets import load_wine
13 from sklearn.model_selection import train_test_split
14
15 X, y = load_wine(return_X_y=True)
16 y = y == 2
17
18 X_train, X_test, y_train, y_test = train_test_split(X, y,
19                                                    random_state=42)
20 svc = SVC(random_state=42)
21 svc.fit(X_train, y_train)

```

3. buka variable explolernya



Name	Type	Size	Value
x	float64	(150, 4)	[[1.622001 1.710000 2.430000 ... 3.080000 3.080000 1.800000]]
x_test	float64	(45, 4)	[[1.360000 1.100000 2.560000 ... 3.480000 1.300000 8.450000]]
x_train	float64	(135, 4)	[[1.110000 2.300000 2.670000 ... 1.430000 1.170000 1.150000]]
y	bool	(150, 1)	[False False False ... True True True]
y_test	bool	(45, 1)	[False False True ... False False True]
y_train	bool	(135, 1)	[False False False ... False False False]

Gambar 1.12 Variable Explorer

### 1.3.2.2 Mencoba loading an example dataset

1. mengambil data iris dan digit dari dataset

```
1 from sklearn import datasets #untuk mengimport dataset dari
  library learn-scikit
2 iris = datasets.load_iris() #membuat sebuah variable iris
  yang mempunyai isi yaitu dataset iris
3 digits = datasets.load_digits() #membuat sebuah variable
  digits yang mempunyai isi yaitu dataset digits
```

2. Menampilkan data digits

```
1 print(digits.data) #memberikan akses ke fitur yang dapat
  digunakan untuk mengklasifikasikan sampel digit dan
  menampilkannya di console
```

```
In [15]: runfile('D:/Semester4/Chapter1/nol.py', wdir='D:/Semester4/Chapter1')
[[ 0.  0. ...  0.  0.]
 [ 0.  0. ... 10.  0.]
 [ 0.  0. ... 16.  0.]
 ...
 [ 0.  0. ...  6.  0.]
 [ 0.  0. ... 12.  0.]
 [ 0.  0. ... 13.  0.]]
```

Gambar 1.13 Data Digits

3. menampilkan digits.target

```
1 digits.target #memberikan informasi tentang data yang
  berhubungan atau juga dapat dijadikan sebagai label
```

```
In [16]: digits.target
Out[16]: array([0, 1, 2, ..., 8, 9, 8])
```

Gambar 1.14 Digits Target

4. menampilkan data bentuk 2D.

```
1 digits.images[0] #Data selalu berupa array 2D, shape (
  n_samples, n_features), meskipun data aslinya mungkin
  memiliki bentuk yang berbeda.
```

```
In [17]: digits.images[0]
Out[17]:
array([[ 0.,  0.,  5., 13.,  9.,  1.,  0.,  0.],
       [ 0.,  0., 13., 15., 10., 15.,  5.,  0.],
       [ 0.,  3., 15.,  2.,  0., 11.,  8.,  0.],
       [ 0.,  4., 12.,  0.,  0.,  8.,  8.,  0.],
       [ 0.,  5.,  8.,  0.,  0.,  9.,  8.,  0.],
       [ 0.,  4., 11.,  0.,  1., 12.,  7.,  0.],
       [ 0.,  2., 14.,  5., 10., 12.,  0.,  0.],
       [ 0.,  0.,  6., 13., 10.,  0.,  0.,  0.]])
```

**Gambar 1.15** Data 2D

## 1.4 1174042 Faisal Najib Abdullah

### 1.4.1 Teori

#### 1. Definisi, sejarah, dan perkembangan kecerdasan buatan.

Definisi kecerdasan buatan adalah suatu pengetahuan yang dapat membuat komputer untuk meniru kecerdasan manusia yang berhubungan dengan penangkapan, pemodelan, dan penyimpanan kecerdasan manusia dalam 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[?].

#### 2. 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 mengelompokan suatu data ke data yang sudah ada. Sedangkan unsupervised learning tidak memiliki data latih, sehingga dari data yang ada, kita mengelompokan data tersebut menjadi 2 bagian atau 3 bagian dan seterusnya.

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.

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.

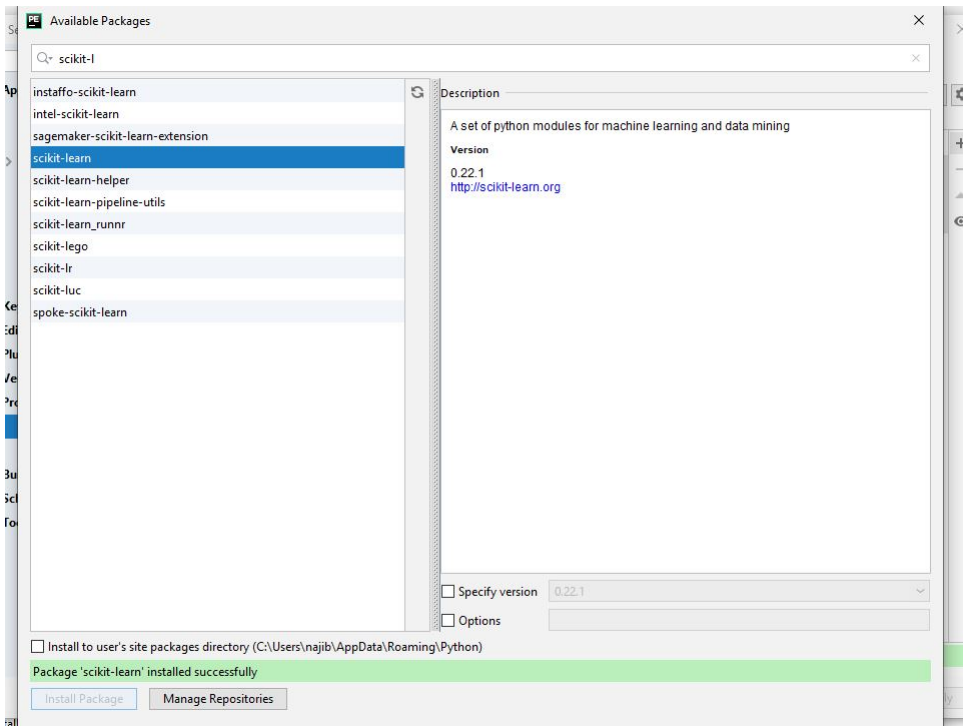
Data set adalah cabang aplikasi dari Artificial Intelligence/Kecerdasan Buatan yang fokus pada pengembangan sebuah sistem yang mampu belajar sendiri tanpa harus berulang kali di program oleh manusia.

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[?].

## 1.4.2 Instalasi

**1.4.2.1 Instalasi Library Scikit dari Pycharm** Masuk pada menu settings terus pilih Project Interpreter kemudian tambah library lalu cari dan install scikit



**Gambar 1.16** Installasi

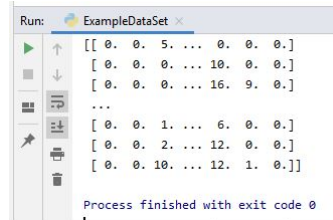
Mencoba Library

```
1 from sklearn import datasets
```

```

2 #pada baris ini merupakan sebuah perintah untuk mengimport class
  datasets dari packaged sklearn
3 iris = datasets.load_iris()
4 #pada baris kedua ini dimana iris merupakan suatu estimator/
  parameter yang berfungsi untuk mengambil data pada item
  datasets.load iris
5 digits = datasets.load_digits()
6 #pada baris ketiga ini dimana digits merupakan suatu estimator/
  parameter yang berfungsi untuk mengambil data pada item
  datasets.load digits

```



**Gambar 1.17** Mencoba Loading an example Dataset

```

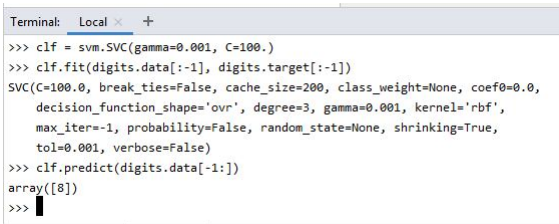
1 from sklearn import svm
2 #pada baris ini merupakan sebuah perintah untuk mengimport class
  svm dari packaged sklearn
3 clf = svm.SVC(gamma=0.001, C=100.)
4 #pada baris kedua ini clf sebagai estimator/parameter, svm.SVC
  sebagai class, gamma sebagai parameter untuk menetapkan nilai
  secara manual
5 clf.fit(digits.data[:-1], digits.target[:-1])
6 #pada baris ketiga ini clf sebagai estimator/parameter, fit
  sebagai metode, digits.data sebagai item,[:-1] sebagai
  syntax pythonnya dan menampilkan outputannya
7 clf.predict(digits.data[:-1:])
8 #pada baris terakhir ini clf sebagai estimator/parameter, predict
  sebagai metode lainnya, digits.data sebagai item dan
  menampilkan outputannya

```

```

1 from sklearn import svm
2 #pada baris ini merupakan sebuah perintah untuk mengimport class
  svm dari packaged sklearn
3 from sklearn import datasets
4 #pada baris ini merupakan sebuah perintah untuk mengimport class
  datasets dari packaged sklearn
5 clf = svm.SVC(gamma='scale')
6 #pada baris ketiga ini clf sebagai estimator/parameter, svm.SVC
  sebagai class, gamma sebagai parameter untuk menetapkan nilai
  secara manual dengan nilai scale

```



```

Terminal: Local x +
>>> clf = svm.SVC(gamma=0.001, C=100.)
>>> clf.fit(digits.data[:-1], digits.target[:-1])
SVC(C=100.0, break_ties=False, cache_size=200, class_weight=None, coef0=0.0,
    decision_function_shape='ovr', degree=3, gamma=0.001, kernel='rbf',
    max_iter=-1, probability=False, random_state=None, shrinking=True,
    tol=0.001, verbose=False)
>>> clf.predict(digits.data[-1:])
array([8])
>>>

```

Gambar 1.18 Learning and Predicting

```

7 iris = datasets.load_iris()
8 #pada baris keempat ini iris sebagai estimator/parameter,
  datasets.load_iris() sebagai item dari suatu nilai
9 X, y = iris.data, iris.target
10 #pada baris kelima ini X, y sebagai estimator/parameter, iris.
   data, iris.target sebagai item dari 2 nilai yang ada
11 clf.fit(X, y)
12 #pada baris keenam ini clf sebagai estimator/parameter dengan
   menggunakan metode fit untuk memanggil estimator X, y dengan
   outputannya
13
14
15 import pickle
16 #pickle merupakan sebuah class yang di import
17 s = pickle.dumps(clf)
18 #pada baris ini s sebagai estimator/parameter dengan pickle.dumps
   merupakan suatu nilai/item dari estimator/parameter clf
19 clf2 = pickle.loads(s)
20 #pada baris ini clf2 sebagai estimator/parameter, pickle.loads
   sebagai suatu item, dan s sebagai estimator/parameter yang
   dipanggil
21 clf2.predict(X[0:1])
22 #pada baris ini clf2.predict sebagai suatu item dengan
   menggunakan metode predict untuk menentukan suatu nilai dari
   (X[0:1])
23 y[0]
24 #pada estimator/parameter y berapapun angka yang diganti nilainya
   akan selalu konstan yaitu 0
25
26
27 from joblib import dump, load
28 #pada baris berikut ini merupakan sebuah perintah untuk
   mengimport class dump, load dari packaged joblib
29 dump(clf, 'filename.joblib')
30 #pada baris berikutnya dump di sini sebagai class yang didalamnya
   terdapat nilai dari suatu item clf dan data joblib
31 clf = load('filename.joblib')
32 #pada baris terakhir clf sebagai estimato/parameter dengan suatu
   nilai load berfungsi untuk mengulang data sebelumnya

```

```

1 #Type Casting

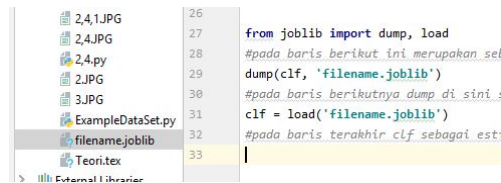
```

```
... clf.fit(X, y)
SVC(C=100.0, break_ties=False, cache_size=200, class_weight=None, coef0=0.0,
    decision_function_shape='ovr', degree=3, gamma=0.001, kernel='rbf',
    max_iter=-1, probability=False, random_state=None, shrinking=True,
    tol=0.001, verbose=False)
```

Gambar 1.19 Model Presistence

```
>>> clf2.predict(X[0:1])
array([0])
>>> #pada baris ini clf2.predict
... y[0]
0
```

Gambar 1.20 Model Presistence



```
26
27 from joblib import dump, load
28 #pada baris berikut ini merupakan sel
29 dump(clf, 'filename.joblib')
30 #pada baris berikutnya dump di sini :
31 clf = load('filename.joblib')
32 #pada baris terakhir clf sebagai est
33
```

Gambar 1.21 Model Presistence

```
2 from sklearn import svm
3 #pada baris ini merupakan sebuah perintah untuk mengimport class
  svm dari packaged sklearn
4 from sklearn import random_projection
5 #pada baris ini merupakan sebuah perintah untuk mengimport class
  random projection dari packaged sklearn
6 rng = np.random.RandomState(0)
7 #rng sebagai estimator/parameter dengan nilai suatu itemnya yaitu
  np.random.RandomState(0)
8 X = rng.rand(10, 2000)
9 #X sebagai estimator/parameter dengan nilai item rng.rand
10 X = np.array(X, dtype='float32')
11 #X sebagai estimator/parameter dengan nilai item np.array
12 X.dtype
13 #X.dtype sebagai item pemanggil
14 transformer = random_projection.GaussianRandomProjection()
15 #transformer sebagai estimator/parameter dengan memanggil class
  random projection
16 X_new = transformer.fit_transform(X)
17 #X new di sini sebagai estomator/parameter dan menggunakan metode
  fit
18 X_new.dtype
19 #X new.dtype sebagai item
20
21
22 from sklearn import datasets
23 #pada baris ini merupakan sebuah perintah untuk mengimport class
  datasets dari packaged sklearn
24 from sklearn.svm import SVC
```

```
25 #pada baris ini merupakan sebuah perintah untuk mengimport class
    SVC dari packaged sklearn.svm
26 iris = datasets.load_iris()
27 #iris sebagai estimator/parameter dengan item datasets.load iris
    ()
28 clf = SVC(gamma='scale')
29 #clf sebagai estimator/parameter dengan nilai class SVC pada
    parameter gamma sebagai set penilaian
30 clf.fit(iris.data, iris.target)
31 #estimator/parameter clf menggunakan metode fit dengan itemnya
32 list(clf.predict(iris.data[:3]))
33 #menambahkan item list dengan metode predict
34 clf.fit(iris.data, iris.target_names[iris.target])
35 #estimator/parameter clf menggunakan metode fit dengan itemnya
36 list(clf.predict(iris.data[:3]))
37 #menambahkan item list dengan metode predict
38
39
40
41 #Refitting and Updating Parameters
42 import numpy as np
43 #pada baris ini merupakan sebuah perintah untuk mengimport class
    svm dari np
44 from sklearn.svm import SVC
45 #pada baris ini merupakan sebuah perintah untuk mengimport class
    SVC dari packaged sklearn.svm
46 rng = np.random.RandomState(0)
47 #rng sebagai estimator/parameter dengan nilai suatu itemnya yaitu
    np.random.RandomState(0)
48 X = rng.rand(100, 10)
49 #X sebagai estimator/parameter dengan nilai item rng.rand
50 y = rng.binomial(1, 0.5, 100)
51 #y sebagai estimator/parameter dengan nilai item rng.binomial
52 X_test = rng.rand(5, 10)
53 #X test sebagai estimator/parameter dengan nilai item rng.rand
54 clf = SVC()
55 #clf sebagai estimator/parameter dan class SVC
56 clf.set_params(kernel='linear').fit(X, y)
57 #set params sebagai item
58 clf.predict(X_test)
59 #menggunakan metode predict
60 clf.set_params(kernel='rbf', gamma='scale').fit(X, y)
61 clf.predict(X_test)
62
63
64 #Multiclass vs. Multilabel Fitting
65 from sklearn.svm import SVC
66 #pada baris ini merupakan sebuah perintah untuk mengimport class
    SVC dari packaged sklearn.svm
67 from sklearn.multiclass import OneVsRestClassifier
68 #pada baris ini merupakan sebuah perintah untuk mengimport class
    OneVsRestClassifier dari packaged sklearn.multiclass
69 from sklearn.preprocessing import LabelBinarizer
70 #pada baris ini merupakan sebuah perintah untuk mengimport class
    LabelBinarizer dari packaged sklearn.preprocessing
71 X = [[1, 2], [2, 4], [4, 5], [3, 2], [3, 1]]
```

```

72 y = [0, 0, 1, 1, 2]
73 classif = OneVsRestClassifier(estimator=SVC(gamma='scale',
       random_state=0))
74 classif.fit(X, y).predict(X)
75 y = LabelBinarizer().fit_transform(y)
76 classif.fit(X, y).predict(X)
77
78
79 from sklearn.preprocessing import MultiLabelBinarizer
80 y = [[0, 1], [0, 2], [1, 3], [0, 2, 3], [2, 4]]
81 y = MultiLabelBinarizer().fit_transform(y)
82 classif.fit(X, y).predict(X)

```

### 1.4.3 Penanganan error

```

>>> from joblib import dump, load
>>> #pada baris berikut ini merupakan sebuah perintah untuk mengimport class dump, load dari packaged joblib
... dump(clf, 'filename.joblib')
Traceback (most recent call last):
  File "<stdin>", line 2, in <module>
NameError: name 'clf' is not defined
>>> #pada baris berikutnya dump di sini sebagai class yang didalamnya terdapat nilai dari suatu item clf dan data joblib
... clf = load('filename.joblib')

```

Gambar 1.22 Error

#### 1.4.3.1 ScreenShoot Error

#### 1.4.3.2 Tuliskan Kode Error dan Jenis Erornya

```

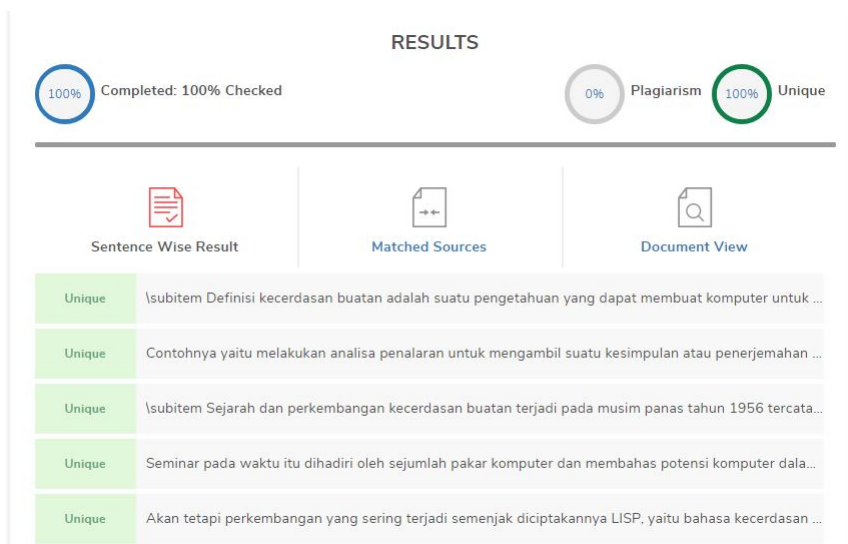
File "<stdin>", line 2, in <module>
NameError: name 'clf' is not defined

```

**1.4.3.3 Solusi Pemecahan Masalah Error** Ini karna kode di jalankan perbaris perbaris, jika kode dijanlankan bersamaan makan kode berjan sesuai prosedur.

### 1.4.4 Plagiat





**Gambar 1.23    Error**

## BAB 2

---

## CHAPTER 2

---

### 2.1 1174006 - Kadek Diva Krishna Murti

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

```
1 @inproceedings{awangga2017colenak ,
2   title={Colenak: GPS tracking model for post-stroke
3     rehabilitation program using AES-CBC URL encryption and QR-
4     Code},
5   author={Awangga, Rolly Maulana and Fathonah, Nuraini Siti and
6     Hasanudin, Trisna Irmayadi},
7   booktitle={Information Technology, Information Systems and
8     Electrical Engineering (ICITISEE), 2017 2nd International
9     conferences on},
10  pages={255--260},
11  year={2017},
12  organization={IEEE}
13 }
```



**Gambar 2.1** Kecerdasan Buatan.

1. Lorem ipsum dolor sit amet, consectetur adipiscing elit.
2. Lorem ipsum dolor sit amet, consectetur adipiscing elit.
3. Lorem ipsum dolor sit amet, consectetur adipiscing elit.

#### **2.1.1 Teori**

#### **2.1.2 Praktek**

#### **2.1.3 Penanganan Error**

#### **2.1.4 Bukti Tidak Plagiat**



**Gambar 2.2** Kecerdasan Buatan.

## BAB 3

---

## CHAPTER 3

---

### 3.1 1174006 - Kadek Diva Krishna Murti

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

```
1 @inproceedings{awangga2017colenak ,
2   title={Colenak: GPS tracking model for post-stroke
3     rehabilitation program using AES-CBC URL encryption and QR-
4     Code},
5   author={Awangga, Rolly Maulana and Fathonah, Nuraini Siti and
6     Hasanudin, Trisna Irmayadi},
7   booktitle={Information Technology, Information Systems and
8     Electrical Engineering (ICITISEE), 2017 2nd International
9     conferences on},
10  pages={255--260},
11  year={2017},
12  organization={IEEE}
13 }
```



**Gambar 3.1** Kecerdasan Buatan.

1. Lorem ipsum dolor sit amet, consectetur adipiscing elit.
2. Lorem ipsum dolor sit amet, consectetur adipiscing elit.
3. Lorem ipsum dolor sit amet, consectetur adipiscing elit.

#### **3.1.1 Teori**

#### **3.1.2 Praktek**

#### **3.1.3 Penanganan Error**

#### **3.1.4 Bukti Tidak Plagiat**



**Gambar 3.2** Kecerdasan Buatan.

## BAB 4

---

## CHAPTER 4

---

### 4.1 1174006 - Kadek Diva Krishna Murti

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

```
1 @inproceedings{awangga2017colenak ,
2   title={Colenak: GPS tracking model for post-stroke
3     rehabilitation program using AES-CBC URL encryption and QR-
4     Code},
5   author={Awangga, Rolly Maulana and Fathonah, Nuraini Siti and
6     Hasanudin, Trisna Irmayadi},
7   booktitle={Information Technology, Information Systems and
8     Electrical Engineering (ICITISEE), 2017 2nd International
9     conferences on},
10  pages={255--260},
11  year={2017},
12  organization={IEEE}
13 }
```



**Gambar 4.1** Kecerdasan Buatan.

1. Lorem ipsum dolor sit amet, consectetur adipiscing elit.
2. Lorem ipsum dolor sit amet, consectetur adipiscing elit.
3. Lorem ipsum dolor sit amet, consectetur adipiscing elit.

#### **4.1.1 Teori**

#### **4.1.2 Praktek**

#### **4.1.3 Penanganan Error**

#### **4.1.4 Bukti Tidak Plagiat**



**Gambar 4.2** Kecerdasan Buatan.

## BAB 5

---

## CHAPTER 5

---

### 5.1 1174006 - Kadek Diva Krishna Murti

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

```
1 @inproceedings{awangga2017colenak ,
2   title={Colenak: GPS tracking model for post-stroke
3     rehabilitation program using AES-CBC URL encryption and QR-
4     Code},
5   author={Awangga, Rolly Maulana and Fathonah, Nuraini Siti and
6     Hasanudin, Trisna Irmayadi},
7   booktitle={Information Technology, Information Systems and
8     Electrical Engineering (ICITISEE), 2017 2nd International
9     conferences on},
10  pages={255--260},
11  year={2017},
12  organization={IEEE}
13 }
```





**Gambar 5.1** Kecerdasan Buatan.

1. Lorem ipsum dolor sit amet, consectetur adipiscing elit.
2. Lorem ipsum dolor sit amet, consectetur adipiscing elit.
3. Lorem ipsum dolor sit amet, consectetur adipiscing elit.

#### **5.1.1 Teori**

#### **5.1.2 Praktek**

#### **5.1.3 Penanganan Error**

#### **5.1.4 Bukti Tidak Plagiat**



**Gambar 5.2** Kecerdasan Buatan.

## BAB 6

---

## CHAPTER 6

---

### 6.1 1174006 - Kadek Diva Krishna Murti

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

```
1 @inproceedings{awangga2017colenak ,
2   title={Colenak: GPS tracking model for post-stroke
3     rehabilitation program using AES-CBC URL encryption and QR-
4     Code},
5   author={Awangga, Rolly Maulana and Fathonah, Nuraini Siti and
6     Hasanudin, Trisna Irmayadi},
7   booktitle={Information Technology, Information Systems and
8     Electrical Engineering (ICITISEE), 2017 2nd International
9     conferences on},
10  pages={255--260},
11  year={2017},
12  organization={IEEE}
13 }
```



**Gambar 6.1** Kecerdasan Buatan.

1. Lorem ipsum dolor sit amet, consectetur adipiscing elit.
2. Lorem ipsum dolor sit amet, consectetur adipiscing elit.
3. Lorem ipsum dolor sit amet, consectetur adipiscing elit.

#### **6.1.1 Teori**

#### **6.1.2 Praktek**

#### **6.1.3 Penanganan Error**

#### **6.1.4 Bukti Tidak Plagiat**



**Gambar 6.2** Kecerdasan Buatan.

# BAB 7

---

## CHAPTER 7

---

### 7.1 1174006 - Kadek Diva Krishna Murti

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

```
1 @inproceedings{awangga2017colenak ,
2   title={Colenak: GPS tracking model for post-stroke
3     rehabilitation program using AES-CBC URL encryption and QR-
4     Code},
5   author={Awangga, Rolly Maulana and Fathonah, Nuraini Siti and
6     Hasanudin, Trisna Irmayadi},
7   booktitle={Information Technology, Information Systems and
8     Electrical Engineering (ICITISEE), 2017 2nd International
9     conferences on},
10  pages={255--260},
11  year={2017},
12  organization={IEEE}
13 }
```



**Gambar 7.1** Kecerdasan Buatan.

1. Lorem ipsum dolor sit amet, consectetur adipiscing elit.
2. Lorem ipsum dolor sit amet, consectetur adipiscing elit.
3. Lorem ipsum dolor sit amet, consectetur adipiscing elit.

#### **7.1.1 Teori**

#### **7.1.2 Praktek**

#### **7.1.3 Penanganan Error**

#### **7.1.4 Bukti Tidak Plagiat**



**Gambar 7.2** Kecerdasan Buatan.