

UJIAN TENGAH SEMESTER (UTS)
MACHINE LEARNING



NAMA : ARI ROCADI

NIM : 20220801339

SESI : EU001/ONLINE

Link Github :

https://github.com/ariocadi/utsml_ariocadi_20220801339

Jawaban Essay

1. ***Machine Learning*** (Pembelajaran Mesin) adalah cabang dari kecerdasan buatan (AI), disiplin ilmu yang mencakup perancangan dan pengembangan algoritma yang memungkinkan komputer untuk mengembangkan perilaku berdasarkan data empiris, seperti dari sensor data basis data. Dalam kata lain, machine learning memungkinkan sistem untuk meningkatkan kinerjanya secara otomatis melalui pengalaman tanpa perlu diprogram secara eksplisit.
2. Berikut adalah beberapa contoh penerapan *Machine Learning* dalam kehidupan sehari-hari, serta alasan dan manfaat penggunaannya:
 - **Pengenalan Wajah untuk Keamanan**, sistem keamanan di smartphone (seperti *Face ID* di *android/iPhone*) atau pengawasan publik menggunakan teknologi pengenalan wajah. karena memudahkan akses pengguna ke perangkat dan aplikasi tanpa perlu mengingat kode akses.
Manfaat dari segi keamanan, memberikan metode autentikasi yang lebih aman dan cepat daripada kata sandi atau PIN. Pengurangan Penipuan, mencegah akses tidak sah ke perangkat atau area tertentu.
 - **Prediksi Penyakit dalam Kesehatan**, aplikasi kesehatan dan rumah sakit menggunakan ML untuk menganalisis data medis dan memprediksi kemungkinan penyakit pada pasien.
Manfaat Deteksi Dini, membantu dalam deteksi dini penyakit yang dapat meningkatkan peluang pengobatan yang berhasil. Personalisasi Perawatan, memberikan rekomendasi perawatan yang dipersonalisasi berdasarkan data pasien.
 - **Kendaraan Otonom**, mobil otonom seperti *Tesla* menggunakan ML untuk mengemudi sendiri dengan memproses data dari sensor dan kamera.
Manfaat keselamatan, mengurangi kesalahan manusia dalam mengemudi yang dapat menyebabkan kecelakaan. Efisiensi, mengoptimalkan rute perjalanan untuk menghemat waktu dan bahan bakar.
 - **Filter Spam di Email**, layanan email seperti Gmail menggunakan ML untuk mendeteksi dan memfilter email spam.
Manfaat keamanan dan Privasi, melindungi pengguna dari email yang berpotensi berbahaya atau penipuan. Efisiensi, mengurangi waktu yang dihabiskan pengguna untuk menghapus email yang tidak diinginkan, sehingga meningkatkan produktivitas.

- **Rekomendasi Produk di *E-commerce***, sebagai contoh *Amazon*, *eBay*, dan platform *e-commerce* lainnya menggunakan algoritma rekomendasi untuk menyarankan produk kepada pengguna berdasarkan riwayat penelusuran dan pembelian mereka. Manfaat personalisasi pengalaman pengguna, membantu pengguna menemukan produk yang mereka sukai tanpa harus mencarinya secara manual. Meningkatkan penjualan, rekomendasi yang relevan dapat meningkatkan peluang pembelian dan meningkatkan pendapatan penjualan.

3. Berikut macam-macam taxonomi dalam penerapan Machine Learning:

1. Berdasarkan Jenis Pembelajaran

Pembelajaran Terawasi (Supervised Learning), Algoritma dilatih menggunakan data yang sudah diberi label, yaitu input-output pairs. Contoh Klasifikasi (Classification) dan Regresi (Regression). Algoritma Linear Regression, Logistic Regression, Support Vector Machines (SVM), K-Nearest Neighbors (KNN), Neural Networks. Aplikasi prediksi harga rumah, klasifikasi email spam, deteksi penyakit.

Pembelajaran Tidak Terawasi (Unsupervised Learning), Algoritma mencoba menemukan pola atau struktur dalam data yang tidak berlabel. Contoh: Clustering dan Asosiasi. Algoritma K-Means, Hierarchical Clustering, Apriori, Principal Component Analysis (PCA). Aplikasi Segmentasi pelanggan, deteksi anomali, pengurangan dimensi.

Pembelajaran Semi-Terawasi (Semi-Supervised Learning), menggabungkan sejumlah kecil data berlabel dengan sejumlah besar data tidak berlabel selama pelatihan. Algoritma Semi-Supervised SVM, Graph-Based Methods. Aplikasi pengakuan gambar dan suara, analisis teks.

Pembelajaran Penguatan (Reinforcement Learning), sistem belajar melalui interaksi dengan lingkungan dan menerima umpan balik dalam bentuk hadiah atau hukuman. Algoritma Q-Learning, Deep Q-Networks (DQN), Policy Gradient Methods. Aplikasi permainan komputer, robotik, kendaraan otonom.

2. Berdasarkan Tipe Data yang Digunakan

Pembelajaran dengan Data Berdimensi Tinggi (High-Dimensional Data), algoritma yang dirancang untuk menangani data dengan jumlah fitur yang sangat banyak. Contoh Neural Networks, Support Vector Machines (SVM).

Pembelajaran dengan Data Berdimensi Rendah (Low-Dimensional Data), algoritma yang cocok untuk data dengan jumlah fitur yang lebih sedikit. Contoh Linear Regression, Logistic Regression.

3. Berdasarkan Metode atau Teknik yang Digunakan

Metode Statistik (Statistical Methods), pendekatan berbasis statistik untuk membuat prediksi atau keputusan. Contoh Bayesian Networks, Linear Regression.

Metode Pembelajaran Dalam (Deep Learning), menggunakan jaringan saraf tiruan dengan banyak lapisan (deep neural networks). Contoh Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN). Aplikasi pengenalan gambar, pemrosesan bahasa alami, prediksi urutan waktu.

Metode Berbasis Pohon Keputusan (Tree-Based Methods), Menggunakan struktur pohon untuk membuat keputusan berdasarkan fitur data. Contoh Decision Trees, Random Forest, Gradient Boosting Trees. Aplikasi klasifikasi dan regresi, analisis risiko.

4. Berdasarkan Tujuan Pembelajaran

Klasifikasi (Classification), mengategorikan data ke dalam kelas atau label yang berbeda. Contoh Logistic Regression, Decision Trees, SVM. Aplikasi deteksi email spam, diagnosis penyakit, pengenalan wajah.

Regresi (Regression), memprediksi nilai kontinu. Contoh Linear Regression, Polynomial Regression, Ridge Regression. Aplikasi prediksi harga saham, estimasi biaya proyek, prediksi suhu.

Clustering, mengelompokkan data yang mirip ke dalam cluster yang sama. Contoh K-Means, Hierarchical Clustering, DBSCAN. Aplikasi Segmentasi pelanggan, analisis pasar, deteksi anomali.

Reduksi Dimensi (Dimensionality Reduction), mengurangi jumlah fitur dalam data untuk mengurangi kompleksitas. Contoh PCA, t-SNE, LDA. Aplikasi visualisasi data, pra-pemrosesan data, kompresi data.

Pengenalan Anomali (Anomaly Detection), mendeteksi data yang tidak biasa atau mencurigakan. Contoh Isolation Forest, One-Class SVM, Autoencoders. Aplikasi deteksi penipuan, pemantauan kesehatan mesin, deteksi intrusi.

5. Berdasarkan Pendekatan Pemodelan

Model Generatif (Generative Models), memodelkan distribusi data dan dapat digunakan untuk menghasilkan data baru. Contoh Gaussian Mixture Models, Hidden Markov Models, Generative Adversarial Networks (GANs). Aplikasi pembuatan gambar, pemodelan bahasa, sintesis suara.

Model Diskriminatif (Discriminative Models), memodelkan batas keputusan antara kelas data. Contoh Logistic Regression, SVM, Conditional Random Fields (CRFs). Aplikasi Klasifikasi, pengenalan pola, segmentasi gambar.

6. Berdasarkan Metode Optimasi

Optimasi Berbasis Gradien (Gradient-Based Optimization), menggunakan teknik turun gradien untuk meminimalkan fungsi biaya. Contoh Stochastic Gradient Descent (SGD), Adam, RMSprop. Aplikasi pelatihan jaringan saraf, regresi logistik, model deep learning.

Optimasi Berbasis Populasi (Population-Based Optimization), menggunakan populasi solusi potensial dan mengevolusikannya seiring waktu. Contoh Genetic Algorithms, Particle Swarm Optimization. Aplikasi Optimasi hiperparameter, pemecahan masalah kompleks.

Jawaban Studi Kasus

1. Sempurnakan Data Set Universitas Esa Unggul

```

Welcome  utsno3.py  X
D: > Pembelajaran > machine learning > UTS > utsno3.py > ...
1  # Data Set
2  data = [
3      {"hari": "senin", "datang": 2, "biaya": 30000 * 2, "mahasiswa": "ani"},
4      {"hari": "selasa", "datang": 3, "biaya": 35000 * 3, "mahasiswa": "budi"},
5      {"hari": "rabu", "datang": 4, "biaya": 25000 * 4, "mahasiswa": "jono"},
6      {"hari": "kamis", "datang": 1, "biaya": 15000 * 1, "mahasiswa": "lono"},
7      {"hari": "jumat", "datang": 2, "biaya": 20000 * 2, "mahasiswa": "joni"},
8      {"hari": "sabtu", "datang": 5, "biaya": 30000 * 5, "mahasiswa": "ani"},
9      {"hari": "minggu", "datang": 2, "biaya": 35000 * 2, "mahasiswa": "budi"}
10 ]
11
12 # a) Berapa rata-rata mahasiswa datang pada minggu ini?
13 total_datang = sum(entry["datang"] for entry in data)
14 rata_rata_datang = total_datang / len(data)
15
16 # b) Kapan biaya tertinggi terjadi?
17 biaya_tertinggi = max(data, key=lambda x: x["biaya"])
18
19 # c) Hari apa biaya lebih dari 110000?
20 biaya_lebih_110k = [entry["hari"] for entry in data if entry["biaya"] > 110000]
21
22 # d) Siapa yang paling banyak datang ke kampus?
23 datang_per_mahasiswa = {}
24 for entry in data:
25     mahasiswa = entry["mahasiswa"]
26     datang_per_mahasiswa[mahasiswa] = datang_per_mahasiswa.get(mahasiswa, 0) + entry["datang"]
27 paling_banyak_datang = max(datang_per_mahasiswa, key=datang_per_mahasiswa.get)
28
29 # e) Siapa yang datang pada hari minggu?
30 datang_minggu = [entry["mahasiswa"] for entry in data if entry["hari"] == "minggu"]
31
32 # f) Berapa biaya tertinggi dan terendah?
33 biaya_tertinggi_nilai = max(entry["biaya"] for entry in data)
34 biaya_terendah_nilai = min(entry["biaya"] for entry in data)
35
36 # g) Berapa frekuensi datang tertinggi dan terendah?
37 frekuensi_datang_tertinggi = max(entry["datang"] for entry in data)
```

```

35
36 # g) Berapa frekuensi datang tertinggi dan terendah?
37 frekuensi_datang_tertinggi = max(entry["datang"] for entry in data)
38 frekuensi_datang_terendah = min(entry["datang"] for entry in data)
39
40 # Print the results
41 print(f"a) Rata-rata mahasiswa datang pada minggu ini: {rata_rata_datang}")
42 print(f"b) Biaya tertinggi terjadi pada hari {biaya_tertinggi['hari']} dengan biaya {biaya_tertinggi['biaya']}")
43 print(f"c) Hari dengan biaya lebih dari 110000: {' '.join(biaya_lebih_110k)}")
44 print(f"d) Mahasiswa yang paling banyak datang ke kampus: {paling_banyak_datang}")
45 print(f"e) Mahasiswa yang datang pada hari minggu: {' '.join(datang_minggu)}")
46 print(f"f) Biaya tertinggi: {biaya_tertinggi_nilai}, Biaya terendah: {biaya_terendah_nilai}")
47 print(f"g) Frekuensi datang tertinggi: {frekuensi_datang_tertinggi}, Frekuensi datang terendah: {frekuensi_datang_terendah}")
48

```

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Python + - [ ] ... X

PS C:\Users\elroc> & C:/Users/elroc/AppData/Local/Microsoft/WindowsApps/python3.11.exe "d:/Pembelajaran/machine_learning/UTS/utsno3.py"
a) Rata-rata mahasiswa datang pada minggu ini: 2.7142857142857144
b) Biaya tertinggi terjadi pada hari sabtu dengan biaya 150000
c) Hari dengan biaya lebih dari 110000: sabtu
d) Mahasiswa yang paling banyak datang ke kampus: ani
e) Mahasiswa yang datang pada hari minggu: budi
f) Biaya tertinggi: 150000, Biaya terendah: 15000
g) Frekuensi datang tertinggi: 5, Frekuensi datang terendah: 1
PS C:\Users\elroc>
Ln 48, Col 1 (2429 selected) Spaces: 4 UTF-8 CRLF Python 3.11.9 64-bit (Microsoft Store)

```

2. Codingan

```

Welcome utsno3.py utsno4.py 2 X utsno4.r
D:\Pembelajaran> machine_learning> UTS> utsno4.py> ...
1 import pandas as pd
2 import matplotlib.pyplot as plt
3
4 # Membuat data frame info_mahasiswa
5 fakultas = ["Bisnis", "D3 Perhotelan", "ICT", "Ilmu Komunikasi", "Seni dan Desain"]
6 jumlah_mahasiswa = [260, 28, 284, 465, 735]
7 akreditasi = ["A", "A", "B", "A", "A"]
8 info_mahasiswa = pd.DataFrame({"fakultas": fakultas, "Jumlah Mahasiswa": jumlah_mahasiswa, "Akreditasi": akreditasi})
9
10 # Membuat diagram batang dengan matplotlib
11 plt.figure(figsize=(10, 6))
12 plt.bar(info_mahasiswa["Fakultas"], info_mahasiswa["Jumlah Mahasiswa"], color=['#FF7F50', '#B8860B', '#3CB371', '#00BFFF', '#FF00FF'])
13 plt.xlabel('Fakultas')
14 plt.ylabel('Jumlah Mahasiswa')
15 plt.title('Jumlah Mahasiswa per Fakultas')
16 plt.xticks(rotation=0)
17 plt.show()
18

```

Output

