

# Pembelajaran Mesin dan Kecerdasan Buatan

## Tugas Minggu Ke-11

Diki Fernandi - 20124023

May 2025

## 1 Mind Map dan Concept Map

### 1.1 Mind Map

Mind map digunakan dalam penelitian ini sebagai alat visualisasi hierarkis untuk memetakan komponen utama penelitian secara terstruktur. Diagram ini membantu dalam:

- Memusatkan topik utama (*Data DBD Jakarta 2008–2024*) sebagai akar permasalahan.
- Membagi elemen penelitian menjadi cabang-cabang seperti *Pengumpulan Data Sekunder*, *Analisis Korelasi*, dan *Pemodelan Matematis*.
- Menghubungkan sub-kategori (contoh: *Data Klim BMKG* dan *Korelasi Spearman*) secara intuitif.

Keunggulan mind map terletak pada kemampuannya menyajikan informasi kompleks secara sederhana melalui struktur radial yang mudah dipahami.

### 1.2 Concept Map

Berbeda dengan mind map, concept map dalam penelitian ini dirancang untuk:

- Menunjukkan hubungan kausal antar konsep (misal: pengaruh *Data Iklim BMKG* terhadap *Formulasi Model Populasi Nyamuk*).
- Memberi label pada setiap hubungan (contoh: "*Sumber Data*", "*Metode*") untuk mempertegas dependensi.
- Mengorganisir proses analisis secara non-linear berbasis jaringan (*network*), terutama untuk memvisualisasikan alur pemodelan matematis dan analisis risiko.

Concept map dipilih untuk menangkap kompleksitas interaksi antar variabel penelitian yang tidak dapat diwakili oleh struktur hierarkis tradisional.

Berikut Mind Map dan Concept Map dari penelitian yang akan dilakukan.

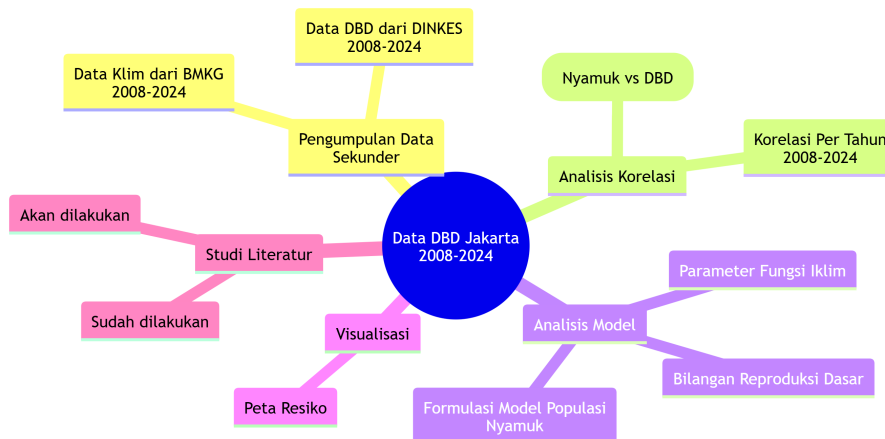


Figure 1: Mind Map

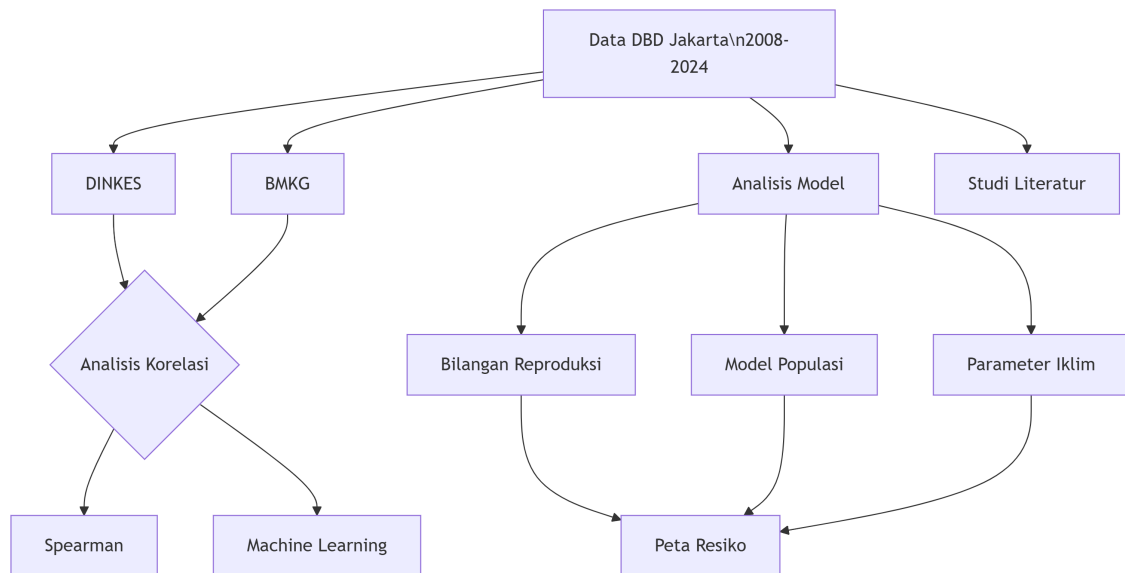


Figure 2: Concept Map

## 2 Data Science and Python Venv

### 2.1 Penjelasan secara singkat tentang data science

Data science (ilmu data) adalah bidang interdisipliner yang menggunakan metode ilmiah, algoritma, dan sistem untuk mengekstraksi pengetahuan atau wawasan dari data, baik yang terstruktur maupun tidak terstruktur. Ilmu ini menggabungkan statistik, ilmu komputer, dan pemahaman tentang domain (bidang aplikasi) untuk menyelesaikan masalah nyata melalui analisis dan pemodelan data.

### 2.2 Perbedaan antara Data, Data Science, dan Data Scientist

- **Data:** Data adalah kumpulan fakta mentah yang belum memiliki makna. Contohnya berupa angka, teks, gambar, atau sinyal yang belum diolah.
- **Data Science:** Ilmu yang mempelajari cara mengumpulkan, mengolah, menganalisis, dan menginterpretasi data untuk menghasilkan informasi atau wawasan yang berguna. Data science menggabungkan statistika, ilmu komputer, dan pemahaman domain.
- **Data Scientist:** Profesional yang bekerja di bidang data science. Mereka menggunakan algoritma, statistik, dan pemrograman untuk menganalisis data dan memberikan solusi berbasis data.

### 2.3 Empat Aspek Dasar Data Science

Berikut ini adalah empat fondasi utama dalam data science:

1. **Pengetahuan Domain (Domain Expertise):** Pemahaman mendalam tentang bidang aplikasi tempat data digunakan, seperti kesehatan, keuangan, atau teknologi.
2. **Matematika dan Statistika:** Digunakan untuk membuat model prediktif dan melakukan inferensi dari data, termasuk konsep seperti regresi, probabilitas, dan uji hipotesis.
3. **Ilmu Komputer (Computer Science):** Kemampuan dalam bahasa pemrograman seperti Python atau R, serta penggunaan pustaka (library) dan framework untuk analisis data dan machine learning.
4. **Komunikasi dan Visualisasi:** Kemampuan menyampaikan hasil analisis dalam bentuk grafik, tabel, atau narasi yang mudah dipahami oleh pihak non-teknis.

## 2.4 Create Virtual Environment

Akan dilakukan pembuatan Venv lalu melakukan instalasi beberapa package selanjutnya menyimpan dalam requirements.txt, berikut tampilan untuk memperjelas.

```
PS D:\SEM 2\Machine learning\my_project> python -m venv env1
PS D:\SEM 2\Machine learning\my_project> Set-ExecutionPolicy Bypass -Scope Process
PS D:\SEM 2\Machine learning\my_project> .\env1\Scripts\activate
(env1) PS D:\SEM 2\Machine learning\my_project> pip install numpy pandas
>> pip freeze > requirements.txt
Collecting numpy
  Using cached numpy-2.0.2-cp39-cp39-win_amd64.whl (15.9 MB)
Collecting pandas
  Using cached pandas-2.2.3-cp39-cp39-win_amd64.whl (11.6 MB)
Collecting python-dateutil>=2.8.2
  Using cached python_dateutil-2.9.0.post0-py3-none-any.whl (229 kB)
Collecting pytz>=2020.1
  Using cached pytz-2025.2-py2.py3-none-any.whl (509 kB)
Collecting tzdata>=2022.7
  Using cached tzdata-2025.2-py2.py3-none-any.whl (347 kB)
Collecting six>=1.5
  Using cached six-1.17.0-py2.py3-none-any.whl (11 kB)
Installing collected packages: pytz, tzdata, six, numpy, python-dateutil, pandas
Successfully installed numpy-2.0.2 pandas-2.2.3 python-dateutil-2.9.0.post0 pytz-2025.2 six-1.17.0 tzdata-2025.2
WARNING: You are using pip version 22.0.4; however, version 25.1.1 is available.
You should consider upgrading via the 'D:\SEM 2\Machine learning\my_project\env1\Scripts\python.exe -m pip install --upgrade pip' command.
(env1) PS D:\SEM 2\Machine learning\my_project> deactivate
```

Figure 3: Venv Pertama

Selanjutnya akan dibuat Venv kedua menggunakan requirements.txt, dengan tampilan sebagai berikut.

```
PS D:\SEM 2\Machine learning\my_project> python -m venv env2
PS D:\SEM 2\Machine learning\my_project> .\env2\Scripts\activate
(env2) PS D:\SEM 2\Machine learning\my_project> pip install -r requirements.txt
Collecting numpy==2.0.2
  Using cached numpy-2.0.2-cp39-cp39-win_amd64.whl (15.9 MB)
Collecting pandas==2.2.3
  Using cached pandas-2.2.3-cp39-cp39-win_amd64.whl (11.6 MB)
Collecting python-dateutil==2.9.0.post0
  Using cached python_dateutil-2.9.0.post0-py2.py3-none-any.whl (229 kB)
Collecting pytz==2025.2
  Using cached pytz-2025.2-py2.py3-none-any.whl (509 kB)
Collecting six==1.17.0
  Using cached six-1.17.0-py2.py3-none-any.whl (11 kB)
Collecting tzdata==2025.2
  Using cached tzdata-2025.2-py2.py3-none-any.whl (347 kB)
Installing collected packages: pytz, tzdata, six, numpy, python-dateutil, pandas
Successfully installed numpy-2.0.2 pandas-2.2.3 python-dateutil-2.9.0.post0 pytz-2025.2 six-1.17.0 tzdata-2025.2
WARNING: You are using pip version 22.0.4; however, version 25.1.1 is available.
You should consider upgrading via the 'D:\SEM 2\Machine learning\my_project\env2\Scripts\python.exe -m pip install --upgrade pip' command.
(env2) PS D:\SEM 2\Machine learning\my_project> █
```

Figure 4: Venv Kedua menggunakan requirements.txt

## 3 Practicing Python for ML

Kode berikut mengimplementasikan alur kerja dasar *Machine Learning* menggunakan Python dan `scikit-learn`. Pertama, dataset Iris dimuat melalui fungsi `load_iris()`, di mana variabel `X` merepresentasikan fitur-fitur numerik (panjang/lebar sepal dan petal), dan `y` merupakan label kelas bunga (Setosa, Versicolor, Virginica). Data kemudian dibagi menjadi *training set* (70%) dan *test set* (30%) menggunakan fungsi `train_test_split` dengan parameter `random_state=42` untuk memastikan hasil yang dapat direproduksi. Enam model klasifikasi berbeda diinisialisasi untuk perbandingan performa, mencakup pendekatan:

- linear: `LogisticRegression`,
- diskriminan: `LinearDiscriminantAnalysis`,
- instance-based: `KNeighborsClassifier`,
- tree-based: `DecisionTreeClassifier`,
- probabilistik: `GaussianNB`,
- kernel-based: `SVC`.

Setiap model dilatih pada data training dan dievaluasi pada data test menggunakan metrik akurasi serta *classification report* yang mencakup nilai *precision*, *recall*, dan *F1-score* per kelas. Hasil evaluasi ditampilkan dalam bentuk tabel **pandas** yang membandingkan performa seluruh model.

Hasil pengujian menunjukkan bahwa sebagian besar model mencapai akurasi sempurna (1.0 atau 100%) pada dataset Iris, dengan rincian sebagai berikut:

Table 1: Perbandingan Performa Model Klasifikasi pada Dataset Iris

Model	Accuracy	Setosa			Versicolor			Virginica		
		Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1
Logistic Regression	1.00	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
LDA	1.00	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
KNN	1.00	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
Decision Tree	1.00	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
Naive Bayes	0.98	1.0	1.0	1.0	1.0	1.0	1.0	0.93	1.0	0.96
SVM	1.00	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0

- **Kesempurnaan Klasifikasi:** Lima dari enam model (Logistic Regression, LDA, KNN, Decision Tree/CART, dan SVM) mencapai akurasi 100% dalam memprediksi semua kelas. Hal ini menunjukkan bahwa dataset Iris merupakan kasus yang relatif mudah dipelajari oleh berbagai algoritma klasifikasi.
- **Performa Naive Bayes:** Model Naive Bayes menunjukkan sedikit penurunan performa pada kelas *virginica* (precision = 0.93, F1 = 0.96), meskipun tetap mencapai akurasi keseluruhan sebesar 98%. Hal ini disebabkan oleh asumsi independensi fitur yang mungkin tidak sepenuhnya terpenuhi.
- **Konsistensi Metrik:** Nilai precision, recall, dan F1-score yang seragam (1.0) untuk semua kelas pada model-model terbaik mengindikasikan:
  1. Tidak ada *false positive* (precision = 1)
  2. Tidak ada *false negative* (recall = 1)
  3. Keseimbangan optimal antara keduanya (F1 = 1)
- **Implikasi Praktis:** Hasil ini memperkuat bahwa dataset Iris bersifat *linearly separable*, sehingga model linear seperti Logistic Regression dan LDA mampu bekerja sebaik model non-linear (SVM dengan kernel default RBF).

## Lampiran 1

Klik di sini untuk mengakses Link Collabs Python lengkap untuk bagian Practicing Python for ML