



Analisis Pola Pengeluaran Mahasiswa

Aplikasi Data Mining untuk menganalisis pola pengeluaran mahasiswa dan memberikan rekomendasi pengelolaan keuangan menggunakan algoritma K-Means Clustering.



Fitur Utama

- **Prediksi Kategori Pengeluaran:** Mengklasifikasikan pengeluaran menjadi "Hemat", "Sedang", atau "Boros"
- **Visualisasi Interaktif:** Grafik dan chart untuk memahami pola pengeluaran
- **Sistem Rekomendasi:** Tips personal berdasarkan kategori pengeluaran
- **Perbandingan:** Membandingkan dengan rata-rata mahasiswa lain
- **Budget Planning:** Rencana budget ideal sesuai kategori



Teknologi yang Digunakan

- **Python 3.7+**
- **Streamlit** - Web framework
- **Scikit-learn** - Machine Learning
- **Pandas & NumPy** - Data manipulation
- **Plotly** - Interactive visualization
- **K-Means Clustering** - Algorithm utama



Prerequisites

Pastikan Python 3.7+ sudah terinstall di sistem Anda.



Quick Start

1. Clone/Download Project

```
bash
```

```
# Download semua file ke dalam satu folder
```

2. Install Dependencies

```
bash
```

```
pip install streamlit pandas numpy scikit-learn matplotlib seaborn plotly joblib
```

3. Jalankan Semua Proses

```
bash  
  
python run_all.py
```

4. Jalankan Aplikasi Web

```
bash  
  
streamlit run app.py
```

Struktur Project

```
project/  
├─ generate_data.py          # Generate dataset simulasi  
├─ explore_data.py          # Eksplorasi dan visualisasi data  
├─ kmeans_analysis.py       # Analisis K-Means clustering  
├─ recommendations.py       # Sistem rekomendasi  
├─ app.py                   # Aplikasi web Streamlit  
├─ run_all.py               # Script untuk menjalankan semua proses  
├─ README.md                # Dokumentasi ini  
├─ student_spending_data.csv # Dataset yang digenerate  
├─ student_spending_clustered.csv # Dataset dengan hasil clustering  
├─ kmeans_model.pkl         # Model K-Means yang sudah dilatih  
├─ scaler.pkl               # Scaler untuk preprocessing  
└─ cluster_labels.pkl       # Label cluster
```

Dataset

Dataset berisi 500 data simulasi mahasiswa dengan fitur:

- **uang_saku:** Uang saku bulanan (Rp 400k - 2M)
- **pengeluaran_makanan:** Pengeluaran untuk makanan
- **pengeluaran_transport:** Pengeluaran untuk transportasi
- **pengeluaran_hiburan:** Pengeluaran untuk hiburan
- **semester:** Semester mahasiswa (1-8)

Algoritma

K-Means Clustering

- **Input Features:** Uang saku, pengeluaran (makanan, transport, hiburan), rasio pengeluaran, semester
- **Output:** 3 Cluster (Hemat, Sedang, Boros)
- **Preprocessing:** StandardScaler untuk normalisasi
- **Evaluation:** Silhouette Score dan Elbow Method

Sistem Rekomendasi

- **Rule-based system** berdasarkan kategori cluster
- **Personal tips** berdasarkan pola pengeluaran individual
- **Budget planning** dengan alokasi ideal per kategori

Cara Menggunakan Aplikasi

1. **Buka aplikasi** di browser (biasanya <http://localhost:8501>)
2. **Input data pengeluaran** di sidebar:
 - Uang saku bulanan
 - Pengeluaran makanan
 - Pengeluaran transport
 - Pengeluaran hiburan
 - Semester saat ini
3. **Klik "Analisis Pengeluaran"**
4. **Lihat hasil:**
 - Kategori pengeluaran Anda
 - Visualisasi breakdown pengeluaran
 - Tips dan rekomendasi personal
 - Rencana budget ideal
 - Perbandingan dengan mahasiswa lain

Screenshot

Input Data

- Form input di sidebar dengan validasi
- Real-time calculation dan feedback

Hasil Analisis

- Metric cards dengan informasi keuangan
- Kategori pengeluaran dengan color coding
- Pie chart breakdown pengeluaran
- Bar chart perbandingan

Rekomendasi

- Tips personal berdasarkan kategori
- Warning untuk pengeluaran berlebihan
- Budget planning dengan alokasi ideal



Hasil yang Diharapkan

Kategori "Hemat" (Rasio < 70%)

- Tips untuk investasi dan pengembangan diri
- Motivasi untuk mempertahankan kebiasaan baik
- Saran untuk tidak terlalu pelit

Kategori "Sedang" (Rasio 70-90%)

- Tips untuk optimasi pengeluaran
- Saran budgeting dan tracking
- Rekomendasi aktivitas hemat

Kategori "Boros" (Rasio > 90%)

- Warning dan tips urgent
- Saran drastis untuk mengurangi pengeluaran
- Bantuan mengontrol pengeluaran impulsif



Customization

Menambah Fitur Input

Edit file `app.py` bagian sidebar input untuk menambah parameter baru.

Mengubah Algoritma

- Ganti K-Means dengan algoritma lain di `kmeans_analysis.py`
- Update prediction function di `app.py`

Menambah Rekomendasi

Edit `recommendations.py` untuk menambah tips atau kategori baru.

Styling

Ubah CSS di `app.py` untuk menyesuaikan tampilan.

Troubleshooting

Error "Model files tidak ditemukan"

```
bash  
python kmeans_analysis.py
```

Error "Data tidak ditemukan"

```
bash  
python generate_data.py
```

Streamlit tidak bisa diakses

- Pastikan port 8501 tidak digunakan aplikasi lain
- Coba gunakan port lain: `streamlit run app.py --server.port 8502`

Error import library

```
bash  
pip install --upgrade [nama_library]
```

Development Notes

Dataset Simulation

- Data dibuat realistis berdasarkan pola pengeluaran mahasiswa Indonesia
- Rasio pengeluaran disesuaikan dengan kondisi ekonomi mahasiswa
- Variasi berdasarkan semester untuk mencerminkan perbedaan kebutuhan

Model Training

- Menggunakan StandardScaler untuk normalisasi

- K-Means dengan random_state untuk reproducibility
- Evaluasi menggunakan Silhouette Score

Web Application

- Responsive design dengan Plotly charts
- Real-time prediction tanpa reload
- Input validation dan error handling



Future Improvements

Fitur Tambahan

- ☐ Export hasil analisis ke PDF
- ☐ Histori pengeluaran dan tracking progress
- ☐ Integrasi dengan e-wallet untuk auto-input
- ☐ Notifikasi reminder budget
- ☐ Social comparison dengan teman

Technical Improvements

- ☐ Database integration untuk menyimpan data user
- ☐ User authentication dan profil
- ☐ API endpoint untuk mobile app
- ☐ Real-time data streaming
- ☐ A/B testing untuk rekomendasi

Model Improvements

- ☐ Ensemble methods (Random Forest + K-Means)
- ☐ Deep Learning untuk pattern recognition
- ☐ Time series forecasting untuk prediksi pengeluaran
- ☐ Reinforcement Learning untuk personalized recommendations



Contributing

1. Fork project ini
2. Buat branch untuk fitur baru: `git checkout -b feature/AmazingFeature`
3. Commit perubahan: `git commit -m 'Add some AmazingFeature'`
4. Push ke branch: `git push origin feature/AmazingFeature`
5. Buka Pull Request

