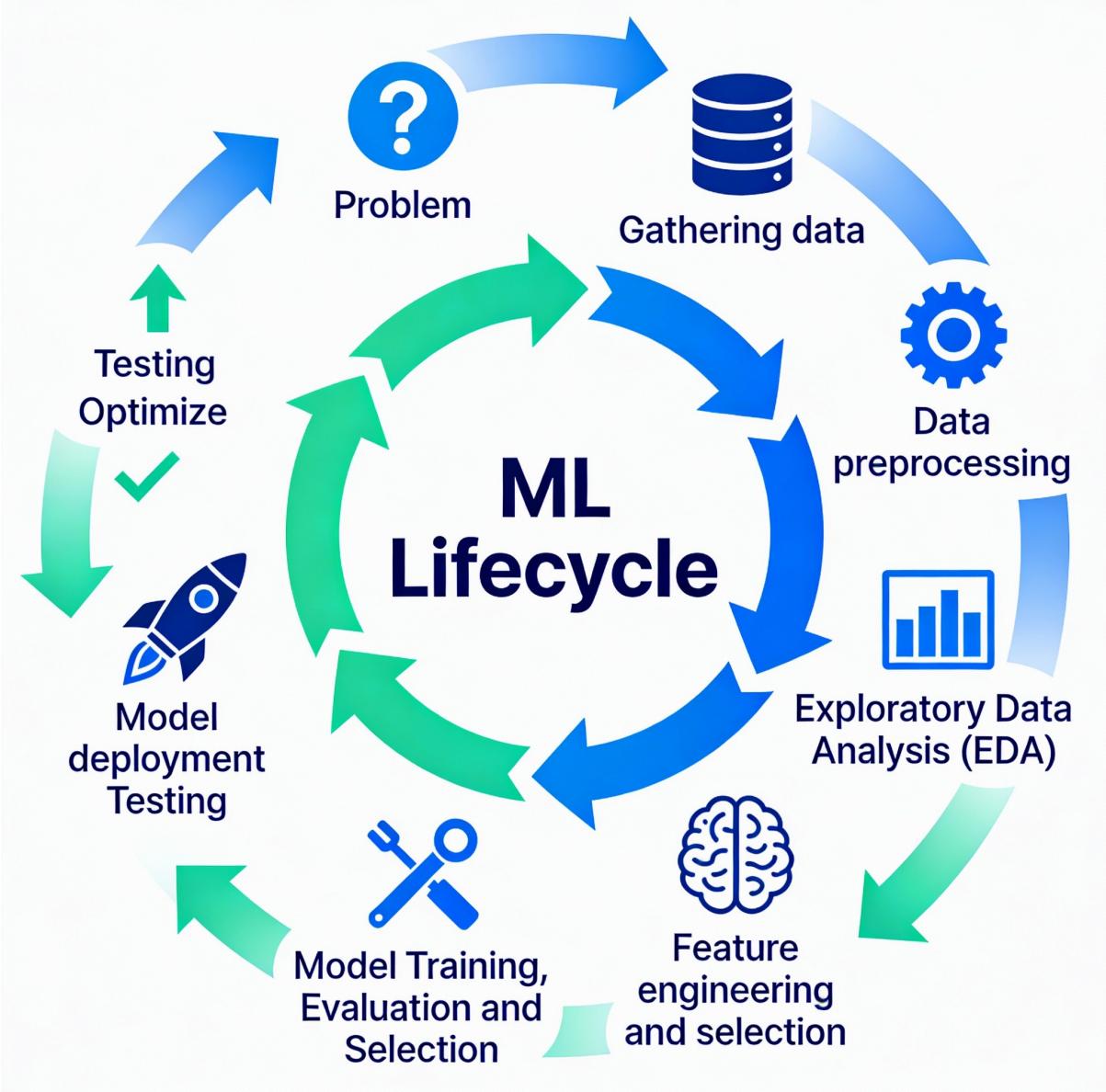


★ MLDLC (MACHINE LEARNING DEVELOPMENT LIFE CYCLE):

1. Problem
2. Gathering data
3. Data preprocessing
4. Exploratory Data Analysis (EDA)
5. Feature engineering and selection
6. Model Training, Evaluation and Selection
7. Model deployment
8. Testing
9. Optimize



1. Problem Definition:

- Clearly define **what problem you are solving**.
- Decide whether it is **classification, regression, clustering, or prediction**.
- Identify **business objective** and **success metrics** (accuracy, cost reduction, time saving).

2. Data Gathering:

- Collect data from sources such as:
 - Databases
 - CSV/Excel files
 - APIs
 - Sensors / logs
- Ensure data relevance to the problem statement.

3. Data Preprocessing:

- Clean the raw data:
 - Handle **missing values**
 - Remove **duplicates**
 - Fix **incorrect data**
- Convert data into usable format (encoding, scaling, normalization).

4. Exploratory Data Analysis (EDA):

Understand data patterns using: & Identify:

- | | |
|--|---|
| <ul style="list-style-type: none">• Graphs• Charts• Summary statistics | <ul style="list-style-type: none">- Trends- Outliers- Relationships betⁿ variables |
|--|---|

5. Feature Engineering & Selection:

- **Create new meaningful features** from existing data.
- Remove irrelevant or redundant features.
- Select features that **improve model performance**.

6. Model Training:

- Choose appropriate ML algorithm:
 - Linear Regression
 - Decision Tree
 - Random Forest
 - SVM
 - Neural Networks
- Train the model using training data.

7. Model Evaluation & Selection:

- Test model performance using:
 - Accuracy
 - Precision
 - Recall
 - F1-score
 - RMSE
- Compare multiple models and **select the best performing** one.

8. Model Deployment:

- Deploy the final model into:
 - Web application
 - Mobile app
 - Cloud service
- Make it **available** for **real-world usage**.

9. Testing & Monitoring:

- Test model with real-time or unseen data.
- Monitor performance:
 - Data drift
 - Accuracy drop
 - Unexpected behavior

10. Optimization & Improvement:

- Retrain model with new data.
 - Tune hyperparameters.
 - Improve features and algorithms.
- Repeat the cycle for better results.

Key Point:

- This is a continuous cycle, not a one-time process.
- Feedback from deployment leads back to data collection and improvement.

In Short:

Definition / One liner:

→ “Machine learning lifecycle starts from problem definition and data collection, followed by preprocessing, model training, deployment, monitoring, and continuous optimization using real-time traffic data.”

Real-Life ML Lifecycle Example:

❖ Traffic Congestion Prediction

1. Problem Definition:

- **Goal:** Predict traffic congestion in a city to reduce travel time and delays.

2. Data Gathering:

- Traffic cameras
- GPS data from vehicles
- Google Maps / navigation apps
- Weather reports
- Historical traffic records

3. Data Preprocessing:

-  **Clean the data:**
 - Remove missing GPS signals
 - Handle incorrect speed values
 - Convert timestamps into usable format
 - Normalize speed and distance data

4. Exploratory Data Analysis (EDA):

-  **Analyze patterns:**
 - Peak traffic hours (morning/evening)
 - Roads with frequent congestion
 - Impact of rain or accidents on traffic

5. Feature Engineering & Selection:

-  Create useful features:
 - Time of day
 - Day of week
 - Road Type
- Remove unnecessary features.

6. Model Training:

-  Train ML models such as:
 - Linear Regression
 - Decision Tree
 - Random Forest
 - Neural Networks
- The model learns traffic behavior patterns.

7. Model Evaluation:

-  Evaluate using:
 - Accuracy
 - RMSE
 - Prediction error
 - Comparison with real traffic data
- Select the best model.

8. Model Deployment:

-  Deploy the model in:
 - Traffic control systems
 - Navigation apps
- The model now predicts real-time congestion.

9. Testing & Monitoring:

-  **Monitor:**

- Prediction accuracy
- Changes due to new roads
- Festivals, accidents, road work

10. Optimization & Improvement:

-  **Improve by:**

- Retraining with new traffic data
- Updating model parameters
- Adding new features (events, road closures)

➡ The cycle repeats continuously.