MLflow with DAGsHub Integration Project

This project demonstrates the integration of MLflow with DAGsHub for experiment tracking and model management in machine learning workflows.

■ Features

- MLflow experiment tracking
- Remote tracking with DAGsHub
- Model versioning and management
- Metrics and parameter logging
- Artifact storage and management

■ Prerequisites

- Python 3.8+
- Git
- DAGsHub account
- Required Python packages: `bash pip install mlflow scikit-learn numpy`

■ Setup

Clone the repository: `bash git clone https://github.com/vijaytakbhate2002/remote-experiment-tracking-with-mlflow-dagshub.git cd remote-experiment-tracking-with-mlflow-dagshub`

Set up DAGsHub credentials in Git Bash: `bash export MLFLOW_TRACKING_URI=https://dagshub.com/<username>/<repo-name>.mlflow export MLFLOW_TRACKING_USERNAME=<your-dagshub-username> export MLFLOW_TRACKING_PASSWORD=<your-dagshub-token>`

Install dependencies: `bash pip install -r requirements.txt`

■■■■ Running the Project

Run the MLflow project: `bash python 1-mlflow-project.py`

View results in DAGsHub:

3. Go to your DAGsHub repository
4. Navigate to the "Experiments" tab
5. View metrics, parameters, and artifacts

■ Project Structure

```
dagshub-mlflow-dvc-project/
■■■ 1-mlflow-project.py    # Main MLflow implementation
```

```
■■■ requirements.txt       # Project dependencies
■■■ README.md             # Project documentation
■■■ mlruns/               # Local MLflow tracking directory
```

■ What's Included

- Experiment tracking with MLflow
- Model training and evaluation
- Parameter logging
- Metric tracking (accuracy, precision, recall, F1-score)
- Model artifact storage
- Remote tracking with DAGsHub

■ MLflow Features Used

- Experiment Management
- Parameter Tracking
- Metric Logging
- Model Versioning
- Artifact Storage
- Remote Tracking

■ Environment Variables

The following environment variables need to be set:

- `MLFLOW_TRACKING_URI`: Your DAGsHub MLflow tracking URI
- `MLFLOW_TRACKING_USERNAME`: Your DAGsHub username
- `MLFLOW_TRACKING_PASSWORD`: Your DAGsHub token

■ Contributing

1. Fork the repository
2. Create your feature branch (`git checkout -b feature/AmazingFeature`)
3. Commit your changes (`git commit -m 'Add some AmazingFeature'`)
4. Push to the branch (`git push origin feature/AmazingFeature`)
5. Open a Pull Request

■ License

■ Authors

- Your Name - Initial work

■ Acknowledgments

- MLflow team for the amazing experiment tracking tool
- DAGsHub for providing the remote tracking infrastructure