Athlete Load Management & Performance Optimization Platform

This project develops an integrated data management system combining sports performance metrics with medical recovery data to predict and prevent athlete injuries. The system processes training load data, recovery indicators, and injury history through a unified SQL database, implementing machine learning models for risk prediction and providing analytical insights through comprehensive SQL reporting.

Objectives:

1. Design and implement integrated SQL schemas linking performance and medical domains
2. Develop injury risk prediction models achieving >80% accuracy metrics
3. Create 8 analytical SQL reports demonstrating complex query capabilities
4. Implement feature engineering for key metrics (ACWR, training monotony, recovery rates)
5. Evaluate models using Precision, Recall, F1-score, and AUC-ROC metrics
6. Document data generation/sourcing processes for reproducibility

Dataset Information:

Primary Source: Synthetic database creation using the Python library “Random” for the documented parameters.

Data Scope: 50 Athletes, 6 months historical data, daily measurements, 5000 values

Ethical Compliance: All data anonymized, no personal identifiers

Technical Implementation:
Team Members

Samuel Greeman and Harsha Prakash

Harsha Prakash (Health Data Science Focus):

- Designed medical/recovery database schema (3 tables, 15+ attributes)
- Implemented injury risk prediction model (Random Forest, XGBoost)
- Created 4 medical-focused SQL reports
- Conducted model evaluation and validation

Samuel Greeman (Sports Data Science Focus):

- Designed performance/training load schema (4 tables, 20+ attributes)
- Implemented load calculation algorithms (ACWR, monotony, strain)
- Created 4 performance-focused SQL reports
- Developed data integration pipelines
- Built feature engineering framework

Tools and Technologies:

Database & Query:

- PostgreSQL 14.0 - Primary relational database
- pgAdmin 4 - Database management interface
- SQL - Complex queries, stored procedures, views

Programming Languages & Frameworks:

- Python 3.9 - Primary development language
- Pandas 1.5.0 - Data manipulation and processing
- NumPy 1.23 - Numerical computations
- Scikit-learn 1.2 - Machine learning models
- Matplotlib/Seaborn - Data visualization

Development Environment:

- Git/GitHub - Version control and collaboration
- Jupyter Notebooks - Exploratory analysis
- VS Code - Integrated development environment
- Overleaf - Documentation and reporting

Implemented SQL Reports:

1. Injury distribution by muscle group and training intensity
2. Recovery progress tracking with timeline analysis
3. Workload spike correlation with injury probability
4. Training load averages by position and time period
5. Predicted vs actual injury outcome comparison
6. Risk factor analysis by athlete demographics
7. Weekly performance load summaries per athlete
8. Model-predicted risk levels by team/time window