KendraGraph Learning Guide – Part 2

AI, GNN & Infrastructure for Orbital Risk Intelligence

Part 1: Graph Neural Networks for Orbital Risk

- 1. Graph Theory Refresher (nodes, edges, adjacency)
- 2. Turning Satellite Pairs into a Graph
- 3. Node & Edge Features (orbital + kinematic features)
- 4. Message Passing and Graph Convolutions
- 5. GCN vs GAT vs GraphSAGE (which fits your use-case)
- 6. Building a Small Orbital GNN in PyTorch Geometric
- 7. Loss Functions for Risk Prediction (BCE / MSE)
- 8. Evaluation Metrics (AUC, Precision@K)
- 9. Inference Pipeline (GNN → Risk → Insight)

Part 2: AI & Insight Layer Design

- 10. Event Summarization (who, when, where, how close)
- 11. Root-Cause Explanation (shared orbital plane, relative velocity)
- 12. Forecasting & Advisory Actions
- 13. Why-Not Reasoning (low-risk justification)
- 14. Counterfactual Generation (what-if changes)
- 15. Confidence, Data Quality & Explainability
- 16. Building a Lightweight Reasoning Engine in Python
- 17. Integrating AI Outputs into the Dashboard

Part 3: Infrastructure & Deployment

- 18. FastAPI Microservices & API Contracts
- 19. Streamlit Frontend Integration
- 20. Logging → JSONL → Parquet → Dashboard
- 21. Docker / Docker-Compose Setup
- 22. Async & Batch Processing
- 23. Local vs Cloud Deployment
- 24. Data Versioning & Model Management
- 25. CI/CD for Kendragraph (GitHub Actions Example)
- 26. Monitoring & Alerting (health checks, Slack webhooks)

Part 4: Future Expansion

- 27. Memory Layer & Historical Similarity
- 28. Operator Policy Profiles & Governance
- 29. Real-Time Conjunction Streaming
- 30. Agentic AI Co-Pilot for Analysts

End of Part 2 — KendraGraph Learning Guide