**8. Validation and Comparison**

**8.1 Define Key Metrics**

1. **Speed**
   * **Inference Latency**: time for the model (RF/GNN/RL policy) to output an action plan once given a contingency snapshot.
   * **End‑to‑End Reaction Time**: total time from contingency detection → mitigation plan → simulated execution completion.
2. **Accuracy / Reliability**
   * **Violation Clearance Rate**: fraction of scenarios where all violations are fully resolved (post\_severity within limits).
   * **Residual Severity**: average remaining overload (%) or voltage deviation (pu) in “uncleared” cases.
3. **Economic Efficiency**
   * **Total Mitigation Cost**: sum of fuel, VoLL, startup, and equipment wear for each scenario.
   * **Cost per Cleared Scenario**: total cost divided by number of cleared scenarios.
   * **Cost Savings**: difference in cost between ML policy and rule‑based baseline.
4. **Plan Quality**
   * **Plan Length**: number of steps in the mitigation sequence (favoring shorter, simpler plans).
   * **Environmental Impact**: CO₂ emissions or other metrics per plan.

**8.2 Experimental Design**

1. **Dataset Split**
   * **Training Set**: 70% of scenarios (all types) for model training.
   * **Validation Set**: 10% for hyperparameter tuning.
   * **Held‑Out Test Set**: 20% unseen scenarios for final evaluation—balanced across N‑1 and N‑2.
2. **Baseline Generation**
   * **Rule‑Based Engine**: use your existing mitigation rules to generate plans for every test scenario.
   * Record all metrics (speed, clearance, cost, plan length).
3. **ML Policy Inference**
   * **RF / GNN**: feed test scenarios into the trained classifier/regressor.
   * **RL Agent**: run each test scenario as an episode in the Gym environment, retrieving the learned policy’s plan.
4. **Simulation of Plans**
   * For each policy’s recommended plan, **execute** the same plan in your pandapower simulation to measure true post\_severity, economic\_cost, and execution\_time\_s.

**8.3 Comparative Analysis**

| **Metric** | **Rule‑Based Baseline** | **ML Policy** | **Improvement** |
| --- | --- | --- | --- |
| Average Inference Latency | e.g. 50 ms | RF: 5 ms / GNN: 30 ms / RL: 100 ms | –90% / –40% / +100% |
| Violation Clearance Rate | 92% | RF: 93% / GNN: 95% / RL: 97% | +1% / +3% / +5% |
| Mean Total Cost ($) | 10 200 | RF: 9 800 / GNN: 9 500 / RL: 9 000 | –3.9% / –6.9% / –11.8% |
| Average Plan Length (steps) | 3 | RF: 1 (single step) / GNN: 1 / RL: 2 | –67% / –67% / –33% |
| Mean CO₂ (kg) | 500 | RF: 480 / GNN: 450 / RL: 430 | –4% / –10% / –14% |

* **Interpretation:** RL yields the best reliability and cost savings but is slower to infer; RF is blazing fast and nearly matches baseline performance; GNN hits a sweet‑spot of topology‑aware accuracy with moderate latency.

**8.4 Statistical Significance**

* **Paired Tests**
  + Use a **paired t‑test** (for cost, time) and **McNemar’s test** (for clearance rate) to confirm improvements are not due to chance.
  + Null hypothesis: no difference between baseline and ML policy.
* **Confidence Intervals**
  + Report 95% CIs for mean cost savings and clearance rate improvements.

**8.5 Visualization & Reporting**

1. **Boxplots** of total cost across scenarios (baseline vs. each ML policy).
2. **CDF Plots** of clearance time (seconds) to compare reaction speed distributions.
3. **Bar Charts** for clearance rate and error bars (95% CI).
4. **Heatmaps** over grid topology showing where ML policies most improve (e.g. regions with frequent N‑2 issues).

**8.6 Continuous Monitoring**

* **Drift Detection**
  + If your live grid changes (new lines, seasonal load shifts), monitor key metrics for **model performance degradation**.
* **Retraining Schedule**
  + Automate retraining (monthly or when > 5% drop in clearance rate).
* **A/B Testing**
  + In a sandbox or “digital twin” environment, run new ML policies alongside rule‑based engine to validate before full rollout.

By following this structured validation and comparison framework, you’ll not only quantify the benefits of each ML approach but also build confidence for operator adoption—demonstrating faster, more accurate, and more economical contingency mitigation than conventional RTCA alone.