Problem Statement 15

Supervised Learning for City-Level IP Geolocation

Reference: IETF RFCs 8805 (self-published IP geolocation data), **IPPM** 2330/2681/7679/7680 (framework, RTT, one-way delay/loss), **ICMP** 792/4443 (active probes), **BGP** 4271 **+** 6793 (origin/ASNs), **RDAP** 9081/9082/9083 (registry/ASN/prefix data), and **DNS** 1035 **+** 3152/3596 (reverse DNS, ip6.arpa) to ground a supervised city-level IP geolocation system.

Objective

Build a **supervised ML system** that predicts a public IP address's **city-level location** using labeled IP→city datasets, improving accuracy and confidence over baseline heuristics.

Problem

Rule-based and database-only geolocation is often stale or coarse. Your task is to train and serve a **machine-learned city classifier** that generalizes across ASNs, prefixes, and time, while producing **well-calibrated confidence** and **kilometer error bounds**.

Data (examples/allowed sources)

- Labeled IP→city pairs (open datasets or organizer-provided dumps).
- **Aux features** you derive: ASN, prefix length, BGP origin, RTTs from vantage points, traceroute last-hop hints, reverse DNS tokens, time zone offset patterns, content-language cues, known PoP/IXP proximity, historical stability.
- Split strategy to avoid leakage: **by prefix/ASN and by time** (train/val/test).

Core Tasks

1. **Feature Engineering**
   - Aggregate prefix/ASN stats; encode rDNS tokens; summarize multi-vantage RTTs (p10/p50); optional graph features (distance to known PoPs).
2. **Modeling**
   - Start with **gradient-boosted trees** or **regularized logistic/softmax** (city classification).
   - Add **probability calibration** (Platt/Isotonic) and a **geo-centroid regressor** for km-error estimation.
3. **Generalization & Robustness**
   - Handle **class imbalance** and rare cities (e.g., focal loss / reweighting / hierarchical city→region).
   - Detect **anycast/VPN/CGNAT** candidates and return "low confidence/region-only."

4. **Serving & API**

    - Expose /predict?ip= returning {city, probability, lat, lon, confidence_radius_km, top_k}.
    - Log inference telemetry for error analysis.

5. **Validation**

    - Strict eval on **held-out ASNs/prefixes** and **future-dated test set**.

Deliverables

- **Training pipeline** (reproducible code + config).
- **Model artifact** + **inference API** (containerized).
- **Benchmark report**: baseline vs. model (tables/plots).
- **Error analysis**: by ASN, city size, continent; confusion map; SHAP/feature importance.
- **README** with setup, data handling, and ethical considerations.

Evaluation Criteria

- **Accuracy:** Top-1 city accuracy; **Top-k (k=3)**; **median & 90p geo error (km)**.
  Calibration: ECE/Brier score; confidence radius coverage (e.g., 90% of truths inside).
- **Generalization:** Performance on **prefix/ASN-held-out** and **temporal holdout**.
- **Engineering quality:** Clear pipeline, API, docs, and reproducibility.
- **Responsibility:** Privacy safeguards, bias analysis (urban vs rural/region), and clear "low-confidence" handling.

Constraints & Guardrails

- No storage of PII beyond public IP and derived features.
- Respect probe/RTT rate limits; cache and anonymize where appropriate.
- Clearly flag uncertain cases (anycast/VPN/CGNAT) rather than over-assert.