## Guilt by Association

The "Guilt by Association" model relies on the premise that fraudulent activity in financial and social networks is rarely isolated but tends to cluster among connected entities. This principle suggests that if a user interacts frequently with known fraudsters, they are statistically likely to be high-risk themselves, even if their individual behavior does not yet appear suspicious. To quantify this risk, the system represents entities as nodes and transactions as edges within a graph, allowing us to measure the distance and connection strength between users. A "suspicion score" is then propagated from a small set of confirmed fraudsters, known as the seed set, outward to the rest of the network. Consequently, this method identifies potential threats by evaluating their proximity to known bad actors in the transaction web, rather than relying solely on isolated attributes.

## The Mathematics of Personalized PageRank

The Rank Vector: Represents the probability distribution (or suspicion scores) of all nodes at iteration t.

The Transition Matrix: A row-normalized matrix representing the graph structure, where entry M_{ij} denotes the probability of moving from node i to node j via a transaction.

α (The Teleportation Probability): A parameter (typically set to 0.15) that determines the likelihood of the surfer jumping back to the seed set rather than following a link. This parameter controls how far the "suspicion" spreads from the seeds; a higher α keeps the score closer to the source.

(1-α): Represents the damping factor, or the probability that the surfer continues traversing the graph's edges.

## The "Teleportation" vector

In Standard PageRank: The vector p is a uniform distribution where every node has a value of 1/N (where N is the total number of nodes). This implies the surfer is equally likely to restart at any node.

In Personalized PageRank (Fraud Detection): The vector p is non-uniform. It is defined such that it has non-zero values only for the known fraudsters (the seed set) and zero for all other nodes.

By concentrating the probability mass of p solely on the confirmed fraudulent nodes, the algorithm forces the random surfer to frequently return to these specific nodes. Consequently, the resulting rank r measures the proximity of other nodes to the seed set. Nodes that are closely connected to the seeds (receivers of direct or short-path transactions) will accumulate higher "suspicion scores" because the random surfer visits them more often than distant nodes.

## Data Representation

To handle the sparsity of real-world transaction graphs efficiently, we utilized the Compressed Sparse Row (CSR) format via the scipy.sparse library.

Choice: We chose CSR over a dense matrix or standard Adjacency Lists for the core Power Iteration engine.

Complexity: This reduces the space complexity from $O(V^2)$ (which is infeasible for large N) to O(V + E), storing only non-zero transactions.

Why it matters: The primary operation in our algorithm is the vector-matrix multiplication (rM). CSR is specifically optimized for fast row slicing and matrix-vector products, making it significantly faster than Coordinate (COO) or List of Lists (LIL) formats for arithmetic operations.

Optimization: For the incremental update feature, we temporarily convert the matrix to LIL (List of Lists) format to efficiently insert new edges in O(1) time, then convert it back to CSR for the recalculation phase.

## Pseudocode handling the "Dangling Node"

```
DanglingNodes = {i | out_degree(i) == 0}


dangling_sum = 0
```

```
:FOR EACH node i IN DanglingNodes DO
dangling_sum = dangling_sum + r_old[i]

dangling_dist = (1 - alpha) * dangling_sum

r_new = ((1 - alpha) * r_prop) + (alpha * p) + (dangling_dist * p)
```

## Cold Start

The Cold Start problem refers to the system's inability to accurately score new nodes that have recently joined the network but have few or no transactions.

The Issue: PPR relies entirely on graph topology and link structure to propagate suspicion. A new user (node) appears as an isolated entity or a disconnected component with no incoming or outgoing edges.

Consequence: Without connections to the existing seed set or the main component, the algorithm assigns a default or zero score to these nodes. Consequently, a new fraudster can initially operate undetected until they establish enough transactions to "bridge" themselves to the known network.

Mitigation: To address this, a hybrid approach would be required, combining graph-based scores with content-based analysis (e.g., checking IP addresses or registration metadata) until sufficient graph history is built.

## Dynamic Graph Updates

The Issue: The standard Power Iteration method has a time complexity of O(V+E). Re-running the entire algorithm from scratch for every single new transaction is computationally prohibitively expensive and causes high latency in a real-time detection system.

Approximation vs. Accuracy: While we implemented an "Incremental Update" (Warm Start) strategy to handle new edges efficiently, this is effectively an approximation. Over time, as thousands of incremental updates accumulate, the rank vector may drift from the true mathematical solution.

Mitigation: A practical system requires a periodic "full re-computation" (e.g., nightly) to reset the error drift, while relying on incremental updates for near real-time scoring during the day.