

WeirdNodes: Supplementary Material

A Description of Experimental Validation on Synthetic Data

A.1 Details on Experiments with WeirdNodes and synthetic data

We recap here the experimental setup we described in the main paper (Sec. 4.4). Using the code that is available at <https://github.com/giaruffo/weirdnodes>, we take a graph G_0 as input and generate a perturbed version G_1 of the graph itself. This allows the simulation of a network’s evolution from a time interval T_0 to T_1 , so that we can compare the two snapshots looking for the anomalies that were injected (and consequently labeled) on purpose.

Lets recall that the original network is perturbed by injecting one or more of the following anomalies:

- **Black Holes:** nodes that in T_1 show a significant increase in incoming links, and a significant reduction of outgoing links w.r.t. T_0 .
- **Volcanoes:** nodes that in T_1 show a significant decrease in the number of incoming links, and a significant increase of outgoing links w.r.t. T_0 .
- **Mushrooms:** endpoints of an edge whose weight from T_0 to T_1 is increased of a random factor between 10 and 100.
- **Ghosts:** endpoints of an edge whose weight from T_0 to T_1 is reduced of a random fraction 0.5 and 1.
- **Indirect Exchangers:** endpoints of an edge that has been removed from T_0 to T_1 , and that have established new connections to a set of intermediary nodes.

We ran four different kinds of experiments. In all of them, the first graph has been randomly generated using an Erdős–Rényi model, with 212 nodes and 2885 to allow a direct comparison with graph $G_{\text{Feb2022}}^{\text{Country}}$ graph described in Sec.5. Also, the graph is directed and weighted, and weights are generated by a Gaussian model with the same mean and standard deviation we observed in $G_{\text{Feb2022}}^{\text{Country}}$.

For each experimental settings, we ran the WeirdNodes procedure, initializing the list **CentralityMeasures** to the centralities shown in Table 1. For each group of experiments, we executed WeirdNodes 5 times, so that we can estimate performances’ fluctuations.

Measure	Intuitive interpretation	Description
Degree $d(i)$	Node i ’s connectivity	The overall number of i ’s connections.
Indegree $d^-(i)$	Node i ’s popularity	The number of i ’s incoming connections.
Outdegree $d^+(i)$	Node i ’s branching factor	The number of i ’s outgoing connections.
Strength $s(i)$	Node i ’s weighted connectivity	The overall weight of i ’s connections.
Instrength $s^-(i)$	Node i ’s weighted popularity	The total weight of i ’s incoming connections.
Outstrength $s^+(i)$	Node i ’s weighted branching factor	The total weight of i ’s outgoing connections.
Betweenness $b(i)$	Node i ’s importance as a bridge	The higher the number of shortest paths passing through i , the greater its value.
Closeness $c(i)$	Node i ’s network topological centrality	The smaller the distances to other nodes, the greater its value.
Eigenvector $ev(i)$	Node i ’s network relevance	The probability that a random walker will eventually reach i in an undirected graph.
PageRank $pr(i)$	Node i ’s network relevance	The probability that a random walker will eventually reach i in a directed graph.
Hub Score $h(i)$	Node i ’s network relevance as a sender	The importance of i as a sender, based on the importance of its successors.
Authority Score $a(i)$	Node i ’s network relevance as a receiver	The importance of i as a receiver, based on the importance of its predecessors.

Table 1: A list of centrality measures used to generate node rankings.

“Black Holes and Volcanoes” experiment: We report here the results’ details of one the five experiments we executed under the “Black Holes and Volcanoes” setting. Then, we will report some statistics of all the 5 runs.

Lets recall the setting of the experiment: the original graph G_0 is perturbed in G_1 by picking up 10 random nodes, and - with a probability of 0.5 for each selected node - mutating them into black holes or volcanoes. Nodes are labeled as “black_hole” or “ghost” respectively. When the selected target or source node is detached, the stub edge is rewired to another randomly assigned node. All the strategies are valid (stability checks returned strong or very strong rankings concordance signals).

Evaluation in terms of precision-at-k ($p@k$), average-precision-at-k (avg $p@k$), and recall-at-30 ($r@k$) is shown in Table 2. Node rankings evolution charts for each given centrality-based strategy are in Fig. 1.

	p@1	p@2	p@5	avg p@5	avg p@10	avg p@20	avg p@30	r@30
degree	1.00	1.00	0.40	0.71	0.74	0.70	0.61	0.80
indegree	1.00	1.00	1.00	1.00	0.97	0.75	0.61	0.80
outdegree	1.00	1.00	1.00	1.00	1.00	0.83	0.69	1.00
strength	1.00	1.00	0.80	0.96	0.94	0.75	0.65	0.90
instrength	1.00	1.00	1.00	1.00	0.97	0.75	0.61	0.80
outstrength	1.00	1.00	1.00	1.00	1.00	0.83	0.69	1.00
betweenness	1.00	1.00	1.00	1.00	0.89	0.64	0.51	0.60
closeness	1.00	1.00	1.00	1.00	0.97	0.75	0.61	0.80
eigenvector	1.00	1.00	1.00	1.00	0.97	0.75	0.61	0.80
hits_hubs	1.00	1.00	1.00	1.00	0.94	0.75	0.61	0.80
hits_authorities	1.00	1.00	1.00	1.00	0.94	0.75	0.61	0.80
pagerank	1.00	1.00	1.00	1.00	0.82	0.58	0.45	0.50
summation_strategy	1.00	1.00	1.00	1.00	0.99	0.80	0.69	1.00

Table 2: “Black Holes and Volcanoes” experiment: performances of the different top-30 lists returned by each centrality based strategy and the summation strategy.

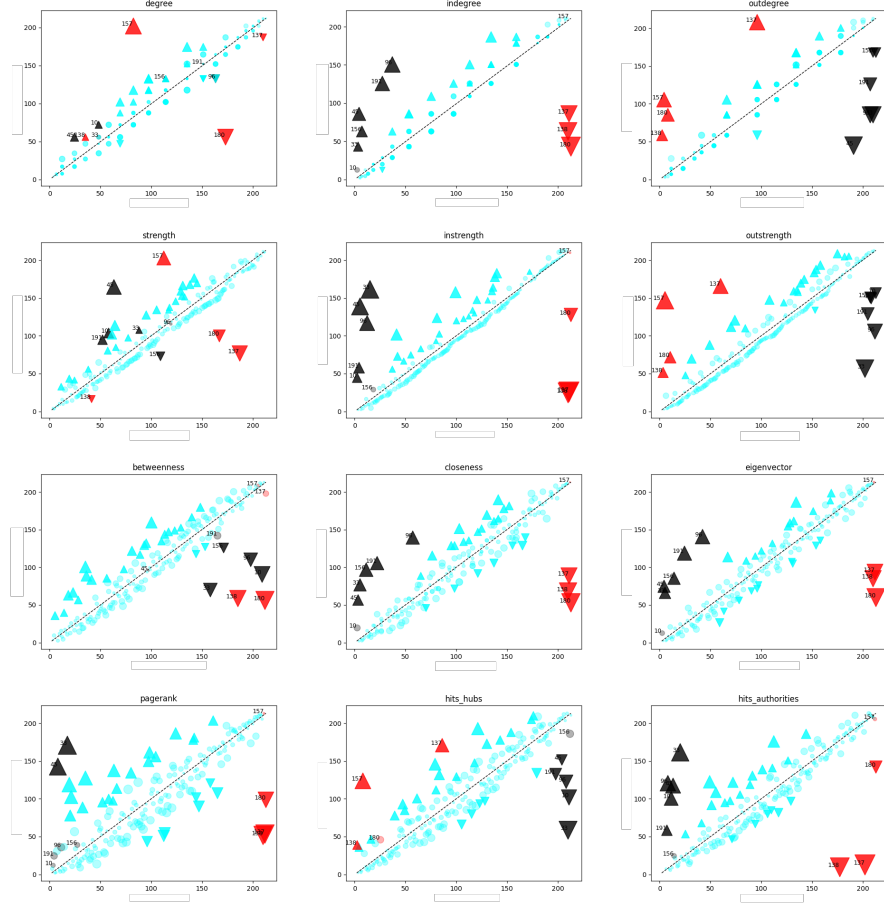


Figure 1: “Black Holes and Volcanoes” experiment: Ranking evolution charts between graph G_0 and its perturbation G_1 . Plots’ horizontal axes describes ranks by the given measure (at top of each plot) at time T_1 , while vertical axes describes ranks at time T_0 . Top-30 nodes that are gaining positions (triangles), and top-30 nodes that are losing positions (inverted triangles) in the rankings are colored according their type: normal (cyan), black holes (black), volcanoes (red).

In Fig. 1, we can see that, after the “Black Holes and Volcanoes” perturbation, there is a clear trend of stable nodes very close to the identity line for each of the centrality-based strategies, as predicted by the strong and very strong signals returned by the stability check. Furthermore, we can spot that among the outliers showing higher ranks residuals, every centrality-based strategy is able to identify many nodes labeled as black holes or volcanoes. It is remarkable that the first 5 positions (see $p@5$ and avg $p@5$) in almost every top-k

lists are correctly labeled as anomalies. Also, we have that all the strategies’ performances are excellent ($p@10$ is always ≥ 0.74 , $r@30$ is often close to 1.0), and that strategies based on outdegree and outstrength over perform ($p@10 = 1.0$).

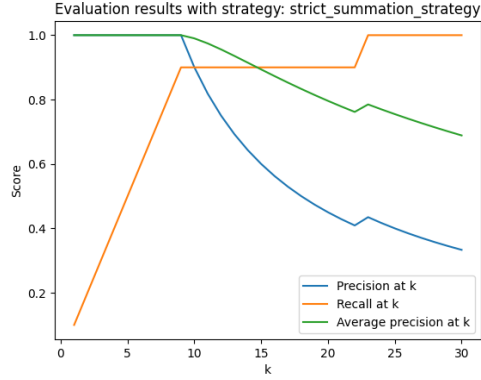


Figure 2: “Black Holes and Volcanoes” experiment: Evaluation of the summation strategy in terms of *precision-at-k*, *precision-at-k*, and *recall-at-k*.

The performance of the “Summation strategy” is shown in the last row of Table 2, and in Fig. 2. Table 3 shows the MAP (Mean Average Precisions) and standard deviations of each strategy for the 5 different runs of this experimental setting. All measures are confirmed exceptionally good in identifying perturbed nodes, with high averages and low standard deviations.

	map@5	map@10	map@20	map@30	r@30
degree	0.87 ± 0.10	0.80 ± 0.08	0.65 ± 0.10	0.54 ± 0.10	0.68 ± 0.19
indegree	1.00 ± 0.00	0.97 ± 0.02	0.77 ± 0.03	0.64 ± 0.03	0.88 ± 0.08
outdegree	1.00 ± 0.00	0.98 ± 0.01	0.81 ± 0.04	0.68 ± 0.02	0.98 ± 0.04
strength	0.93 ± 0.06	0.81 ± 0.13	0.65 ± 0.07	0.54 ± 0.07	0.66 ± 0.15
instrength	1.00 ± 0.00	0.96 ± 0.01	0.77 ± 0.03	0.64 ± 0.02	0.88 ± 0.04
outstrength	0.99 ± 0.02	0.97 ± 0.02	0.80 ± 0.03	0.68 ± 0.02	0.98 ± 0.04
betweenness	0.97 ± 0.02	0.92 ± 0.03	0.70 ± 0.04	0.59 ± 0.05	0.76 ± 0.11
closeness	0.99 ± 0.02	0.93 ± 0.04	0.77 ± 0.05	0.62 ± 0.05	0.84 ± 0.11
eigenvector	1.00 ± 0.00	0.94 ± 0.05	0.74 ± 0.04	0.63 ± 0.04	0.86 ± 0.09
hits_hubs	0.97 ± 0.06	0.92 ± 0.03	0.74 ± 0.04	0.63 ± 0.05	0.86 ± 0.11
hits_authorities	0.88 ± 0.16	0.85 ± 0.14	0.71 ± 0.08	0.60 ± 0.07	0.80 ± 0.16
pagerank	0.90 ± 0.12	0.88 ± 0.06	0.72 ± 0.08	0.60 ± 0.09	0.80 ± 0.17
summation_strategy	0.83 ± 0.22	0.82 ± 0.10	0.72 ± 0.08	0.65 ± 0.06	0.90 ± 0.14

Table 3: “Black Holes and Volcanoes” experiment: performance metrics (MAP@k $\pm$ standard deviation) for all the 5 runs.

“Mushrooms and Ghosts” experiment: We report here the results’ details of one the five experiments we executed under the “Mushrooms and Ghosts”

setting. Then, we will report some statistics of all the 5 runs.

Lets recall the setting of the experiment: the original graph G_0 is perturbed in G_1 by picking up 5 random edges, and - with a probability of 0.5 for each selected edge - significantly increasing or decreasing their weight. Endpoints are marked as “mushroom” or “ghost” respectively.

Some strategies (based on hits hub, and hits authority scores) returned weak or moderate concordance signal, while other measures (degree, indegree, outdegree, closeness, betweenness) were skipped because the “Mushrooms and Ghosts” perturbation does not change values of those centralities. Hence, only strength, instrength, outstrength, pagerank based strategies were used. Evaluation in terms of precision-at-k ($p@k$), average-precision-at-k (avg $p@k$), and recall-at-30 ($r@30$) is shown in Table 4. Node rankings evolution charts for each given centrality-based strategy are in Fig. 3.

	p@1	p@2	p@5	avg p@5	avg p@10	avg p@20	avg p@30	r@30
strength	1.00	1.00	1.00	1.00	0.99	0.80	0.65	0.90
instrength	1.00	1.00	1.00	1.00	0.82	0.64	0.51	0.60
outstrength	1.00	1.00	0.80	0.96	0.74	0.50	0.39	0.40
pagerank	1.00	1.00	0.60	0.87	0.63	0.50	0.39	0.40
summation_strategy	1.00	1.00	1.00	1.00	0.97	0.75	0.61	0.80

Table 4: “Mushrooms and Ghosts” experiment: performances of the different top-30 lists returned by each valid centrality based strategy and the summation strategy.

In this case, strength-based strategy over performs the others, while the summation strategy (shown in the last row of Table 4, and in Fig. 4) is a little bit behind. Table 5 shows the MAP@k and standard deviations of our valid strategies for the 5 different runs of this experimental setting. Again, strength-based and summation strategies perform very well in terms of precision and recall, high averages, and low deviations. Pagerank, in/outstrength based strategies, although performing very well with $p@k$ with low k, often fail to detect all the perturbed nodes in their top-30 lists.

	map@5	map@10	map@20	map@30	r@30
strength	1.00 ± 0.00	0.99 ± 0.01	0.82 ± 0.02	0.67 ± 0.02	0.96 ± 0.05
instrength	1.00 ± 0.00	0.85 ± 0.04	0.63 ± 0.03	0.50 ± 0.03	0.58 ± 0.04
outstrength	0.99 ± 0.02	0.80 ± 0.04	0.58 ± 0.05	0.45 ± 0.04	0.50 ± 0.07
pagerank	0.81 ± 0.09	0.71 ± 0.08	0.51 ± 0.07	0.41 ± 0.06	0.44 ± 0.09
summation_strategy	0.99 ± 0.02	0.96 ± 0.01	0.80 ± 0.03	0.66 ± 0.03	0.92 ± 0.08

Table 5: “Mushrooms and Ghosts” experiment: performance metrics (MAP@k $\pm$ standard deviation) for all the 5 runs.

“Indirect Exchangers and Intermediaries” experiment: We report here the results’ details of one the five experiments we executed under the “Indirect

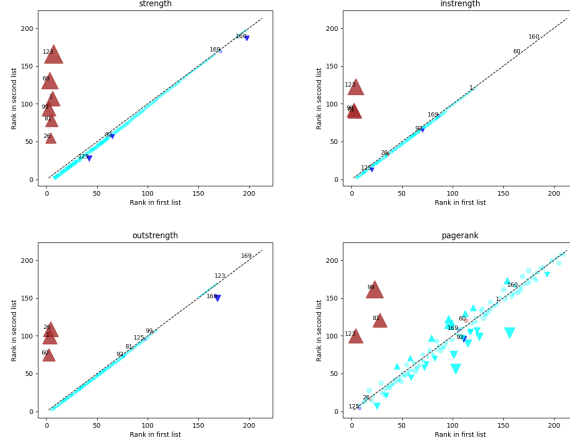


Figure 3: “Mushrooms and Ghosts” experiment: Ranking evolution charts between graph G_0 and its perturbation G_1 . Plots’ horizontal axes describes ranks by the given measure (at top of each plot) at time T_1 , while vertical axes describes ranks at time T_0 . Top-30 nodes that are gaining positions (triangles), and top-30 nodes that are losing positions (inverted triangles) in the rankings are colored according their type: normal (cyan), mushrooms (brown), ghosts (blue).

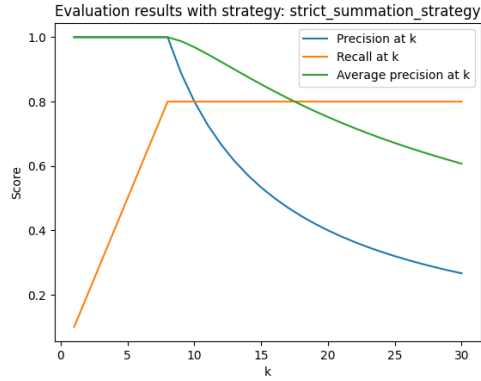


Figure 4: “Mushrooms and Ghosts” experiment: Evaluation of the summation strategy in terms of *precision-at-k*, *precision-at-k*, and *recall-at-k*.

Exchangers and Intermediaries” setting. Then, we will report some statistics of all the 5 runs.

Lets recall the setting of the experiment: the original graph G_0 is perturbed in G_1 by removing up 5 random edges, randomly selecting other 5 intermediary nodes for each removed edge, and wiring source and target nodes by means

of these intermediaries. Each involved node is marked as “indirect_source”, “indirect_target”, or “intermediary”.

All strategies are valid (stability checks returned strong or very strong rankings concordance signals). However, this is the only case out of five runs: rankings based hits hubs and authority scores returned a weak/moderate signal, and they were skipped. The evaluation in terms of precision-at-k ($p@k$), average-precision-at-k (avg $p@k$), and recall-at-30 ($r@k$) is shown in Table 6. Node rankings evolution charts for each given centrality-based strategy are in Fig. 5, showing once again that “normal” nodes, that are the majority, are very close to the identity lines.

	p@1	p@2	p@5	avg p@5	avg p@10	avg p@20	avg p@30	r@30
degree	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.85
indegree	1.00	1.00	1.00	1.00	1.00	1.00	0.98	0.76
outdegree	1.00	1.00	1.00	1.00	1.00	1.00	0.96	0.67
strength	1.00	1.00	1.00	1.00	1.00	1.00	0.99	0.82
instrength	1.00	1.00	1.00	1.00	1.00	1.00	0.98	0.73
outstrength	1.00	1.00	1.00	1.00	1.00	1.00	0.98	0.73
betweenness	1.00	1.00	1.00	1.00	1.00	0.99	0.97	0.70
closeness	1.00	1.00	1.00	1.00	1.00	0.98	0.93	0.61
eigenvector	1.00	1.00	1.00	1.00	1.00	0.99	0.93	0.61
hits_hubs	1.00	1.00	1.00	1.00	0.97	0.96	0.84	0.45
hits_authorities	1.00	1.00	0.80	0.96	0.97	0.92	0.88	0.52
pagerank	1.00	1.00	1.00	1.00	0.99	0.92	0.84	0.45
summation_strategy	1.00	1.00	1.00	1.00	1.00	0.99	0.93	0.61

Table 6: “Indirect Exchangers and Intermediaries” experiment: performances of the different top-30 lists returned by each centrality based strategy and the summation strategy.

The performance of the “summation strategy” is shown in the last row of Table 6, and in Fig. 6. Table 7 shows the MAP@k and standard deviations of all the valid strategies for the 5 different runs of this experimental setting. All strategies detect our inoculated irregularities with high precision: MAP@5 is equal to 1 (or very close to 1) in many cases, meaning that average precision at k remains very high even when $k = 30$. Apparently recall does not perform as well, although $r@30$ ’s values are high. It is worth noting that this experimental settings produce more not normal nodes than the other experiments: in fact, we have at least 10 indirect exchangers, plus up to 25 intermediaries. This means that top-30 lists are definitely too short to capture them all.

“Mixed Anomalies” experiment: We report here the results’ details of one the five experiments we executed under the “Mixed Anomalies” setting. Then, we will report some statistics of all the 5 runs.

Lets recall the setting of the experiment: the original graph G_0 is perturbed in G_1 by picking up 4 random nodes (that are modified as “black_hole” or “ghost”), 2 random edges (each used to change its endpoints to “mushroom” or “ghost”), and 2 other random edges (each used to change its endpoints to “indirect_source” and “indirect_target”, and to make other 5 nodes “intermediary”

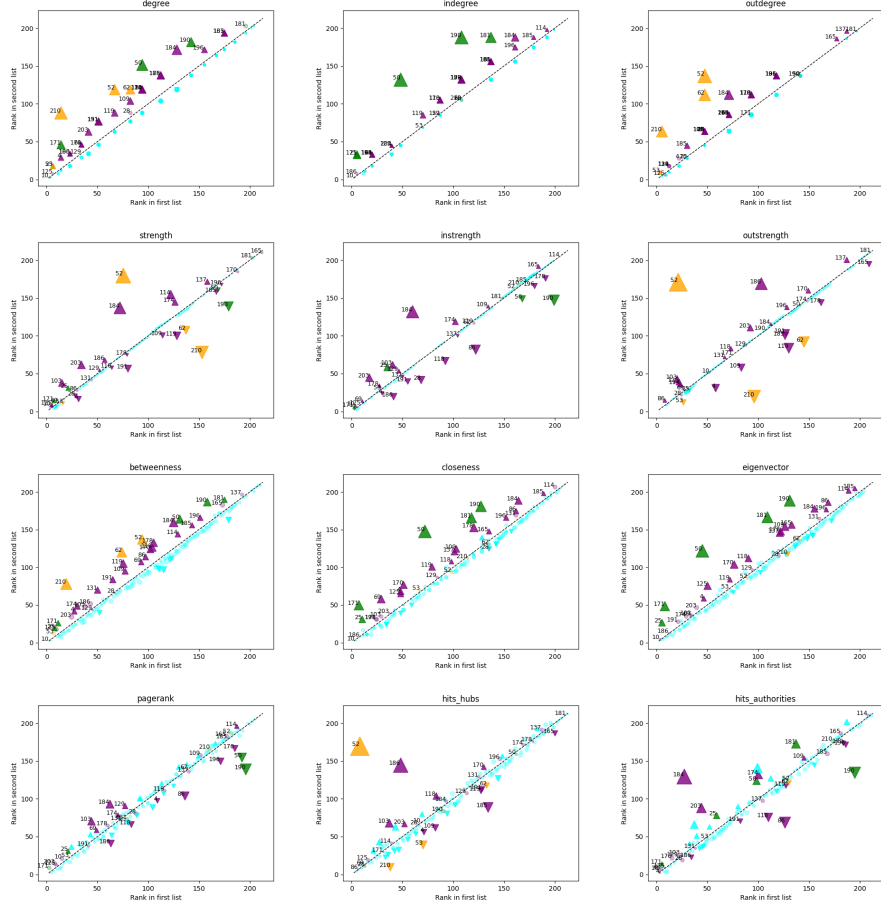


Figure 5: “Indirect Exchangers and Intermediaries” experiment: Ranking evolution charts between graph G_0 and its perturbation G_1 . Plots’ horizontal axes describes ranks by the given measure (at top of each plot) at time T_1 , while vertical axes describes ranks at time T_0 . Top-30 nodes that are gaining positions (triangles), and top-30 nodes that are losing positions (inverted triangles) in the rankings are colored according their type: normal (cyan), indirect sources (orange), indirect targets (green), and intermediaries (purple).

between them).

The evaluation in terms of precision-at-k ($p@k$), average-precision-at-k ($avg\ p@k$), and recall-at-30 ($r@k$) is shown in Table 8. Node rankings evolution charts for each given centrality-based strategy are in Fig. 7.

The performance of the “Summation strategy” is shown in the last row of Table 8, and in Fig. 8. Table 9 shows the MAP@k and deviations of all strategies for the 5 different runs of this experimental setting. Even if we injected a more

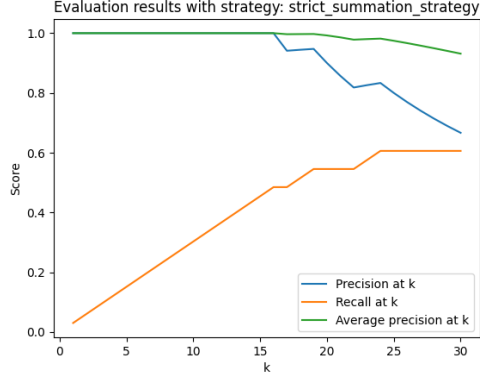


Figure 6: “Indirect Exchangers and Intermediaries” experiment: Evaluation of the summation strategy in terms of *precision-at-k*, *precision-at-k*, and *recall-at-k*.

	map@5	map@10	map@20	map@30	r@30
degree	1.00 ± 0.00	1.00 ± 0.00	0.99 ± 0.01	0.96 ± 0.04	0.80 ± 0.11
indegree	1.00 ± 0.00	1.00 ± 0.00	0.99 ± 0.01	0.93 ± 0.03	0.71 ± 0.06
outdegree	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.01	0.94 ± 0.03	0.72 ± 0.06
strength	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	0.96 ± 0.02	0.79 ± 0.05
instrength	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.01	0.94 ± 0.04	0.74 ± 0.08
outstrength	1.00 ± 0.00	1.00 ± 0.00	0.99 ± 0.01	0.93 ± 0.03	0.70 ± 0.02
betweenness	1.00 ± 0.00	1.00 ± 0.00	0.98 ± 0.01	0.93 ± 0.05	0.70 ± 0.10
closeness	1.00 ± 0.00	1.00 ± 0.00	0.97 ± 0.02	0.89 ± 0.04	0.62 ± 0.07
eigenvector	1.00 ± 0.00	1.00 ± 0.00	0.96 ± 0.03	0.87 ± 0.06	0.58 ± 0.08
hits_hubs	0.97 ± 0.06	0.95 ± 0.07	0.90 ± 0.07	0.83 ± 0.04	0.51 ± 0.07
hits_authorities	0.99 ± 0.02	0.99 ± 0.01	0.95 ± 0.02	0.87 ± 0.02	0.58 ± 0.04
pagerank	0.89 ± 0.24	0.94 ± 0.11	0.88 ± 0.10	0.81 ± 0.07	0.49 ± 0.09
summation_strategy	0.99 ± 0.02	0.98 ± 0.03	0.97 ± 0.02	0.93 ± 0.02	0.70 ± 0.07

Table 7: “Indirect Exchangers and Intermediaries” experiment: performance metrics (MAP@k ± standard deviation) for all the 5 runs.

complex combination of anomalies in the perturbed graph, average precision @ k values are still very high. Also in the “Mixed Anomalies” setting, as we already observed for the previous one, recall can be probably improved if we return longer top-k lists: in fact, we are perturbing up to 20 nodes.

	p@1	p@2	p@5	avg p@5	avg p@10	avg p@20	avg p@30	r@30
degree	1.00	1.00	1.00	1.00	0.99	0.92	0.84	0.68
indegree	1.00	1.00	1.00	1.00	0.89	0.80	0.72	0.50
outdegree	1.00	1.00	1.00	1.00	0.97	0.87	0.79	0.59
strength	1.00	1.00	1.00	1.00	0.99	0.92	0.86	0.73
instrength	1.00	1.00	1.00	1.00	0.97	0.94	0.84	0.68
outstrength	1.00	1.00	1.00	1.00	0.99	0.90	0.81	0.64
betweenness	1.00	1.00	1.00	1.00	0.99	0.83	0.79	0.59
closeness	1.00	1.00	1.00	1.00	0.82	0.64	0.69	0.45
eigenvector	1.00	1.00	1.00	1.00	0.82	0.75	0.72	0.50
hits_hubs	1.00	1.00	1.00	1.00	0.97	0.87	0.72	0.50
hits_authorities	1.00	1.00	0.80	0.96	0.94	0.87	0.72	0.50
pagerank	1.00	0.50	0.40	0.71	0.82	0.70	0.56	0.32
summation_strategy	1.00	1.00	0.60	0.87	0.82	0.83	0.76	0.55

Table 8: “Mixed Anomalies” experiment: performances of the different top-30 lists returned by each valid centrality based strategy and the summation strategy.

	map@5	map@10	map@20	map@30	r@30
degree	0.98 ± 0.02	0.95 ± 0.05	0.90 ± 0.05	0.80 ± 0.08	0.66 ± 0.13
indegree	0.99 ± 0.02	0.86 ± 0.07	0.79 ± 0.07	0.72 ± 0.05	0.53 ± 0.08
outdegree	0.99 ± 0.02	0.95 ± 0.01	0.81 ± 0.09	0.72 ± 0.11	0.53 ± 0.15
strength	0.98 ± 0.02	0.93 ± 0.07	0.86 ± 0.06	0.77 ± 0.09	0.61 ± 0.14
instrength	0.99 ± 0.02	0.95 ± 0.07	0.86 ± 0.06	0.79 ± 0.03	0.63 ± 0.04
outstrength	0.97 ± 0.06	0.96 ± 0.02	0.84 ± 0.04	0.79 ± 0.05	0.64 ± 0.11
betweenness	0.91 ± 0.06	0.85 ± 0.11	0.73 ± 0.11	0.69 ± 0.07	0.48 ± 0.08
closeness	0.99 ± 0.02	0.88 ± 0.06	0.77 ± 0.08	0.72 ± 0.03	0.51 ± 0.07
eigenvector	0.99 ± 0.02	0.86 ± 0.07	0.79 ± 0.04	0.74 ± 0.02	0.54 ± 0.05
pagerank	0.86 ± 0.15	0.88 ± 0.06	0.75 ± 0.05	0.68 ± 0.08	0.47 ± 0.11
summation_strategy	0.93 ± 0.06	0.90 ± 0.06	0.82 ± 0.03	0.71 ± 0.06	0.51 ± 0.07

Table 9: “Mixed Anomalies” experiment: performance metrics (MAP@k $\pm$ standard deviation) for all the 5 runs.

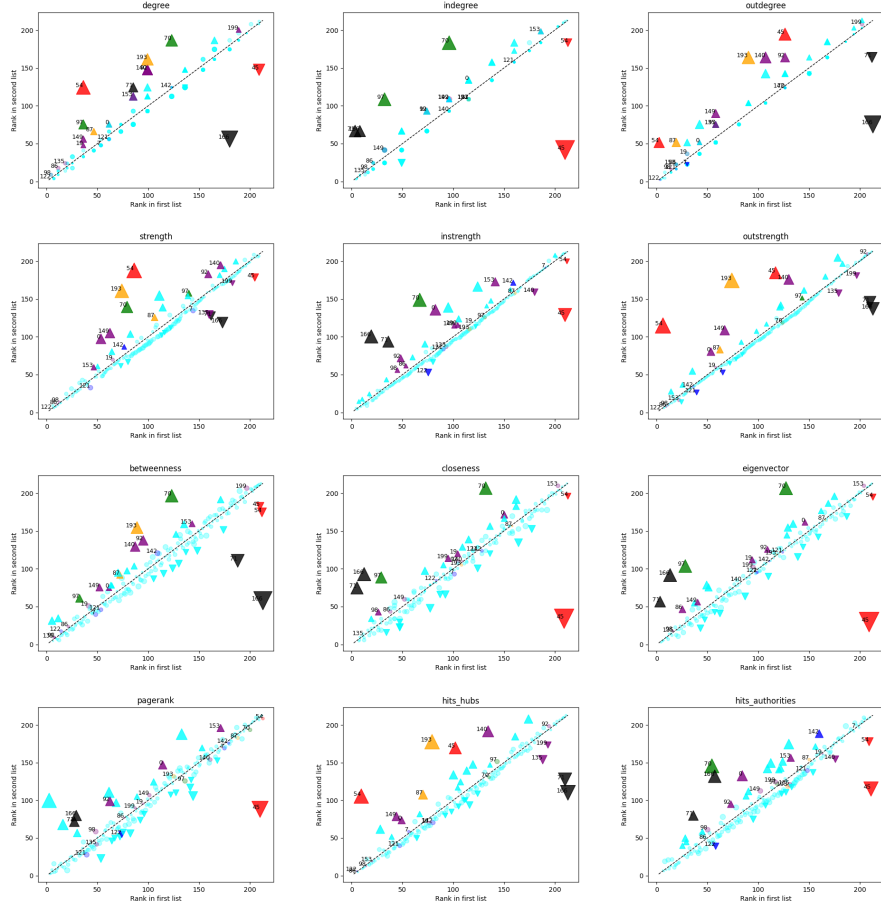


Figure 7: “Mixed Anomalies” experiment: Ranking evolution charts between graph G_0 and its perturbation G_1 . Plots’ horizontal axes describes ranks by the given measure (at top of each plot) at time T_1 , while vertical axes describes ranks at time T_0 . Top-30 nodes that are gaining positions (triangles), and top-30 nodes that are losing positions (inverted triangles) in the rankings are colored according their type, as coded in the previous figures.

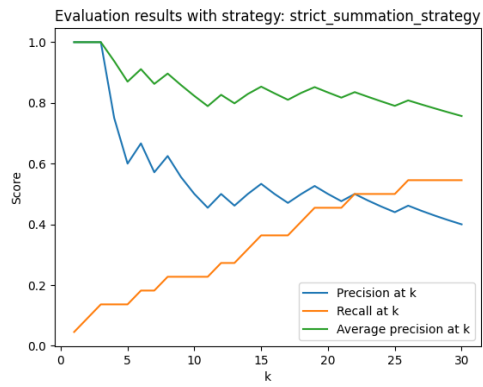


Figure 8: “Mixed Anomalies” experiment: Evaluation of the summation strategy in terms of *precision-at-k*, *precision-at-k*, and *recall-at-k*.

B Dataset Statistics

In this section, we provide quantitative information about the dataset used for our showcase experiment. It contains fully anonymized information about financial transactions. Specifically, each record contains a timestamp, information about the amount, currency and data stream of the transactions. By aggregating the transactions by month, the *amount* variable is distributed according to Fig. 9. While the vast majority of transactions is below 1,000 euros, a considerable number of outliers can reach up to 10^9 euros.

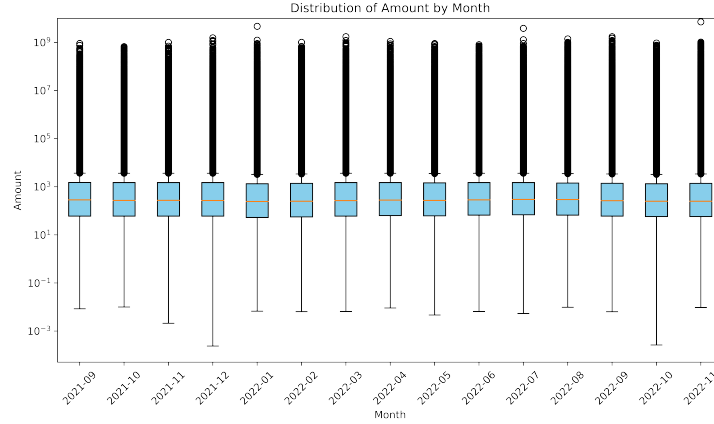


Figure 9: Distribution of the variable *amount* by month.

Furthermore, by aggregating the transactions by BIC, we can build the directed graphs $G_{\text{Sep2021}}^{\text{BIC}}, G_{\text{Oct2021}}^{\text{BIC}}, \dots, G_{\text{Nov2022}}^{\text{BIC}}$. In Fig. 10 we can observe the distributions of the indegree, outdegree, instrength and outstrength across the months.

In Fig. 12 we can observe the relationship between the amount of money moved by each node and its position in the rankings based on the strength. This relationship is fitted to a Zipf's Law with shape parameter $\alpha = 1.5$ in a log-log space.

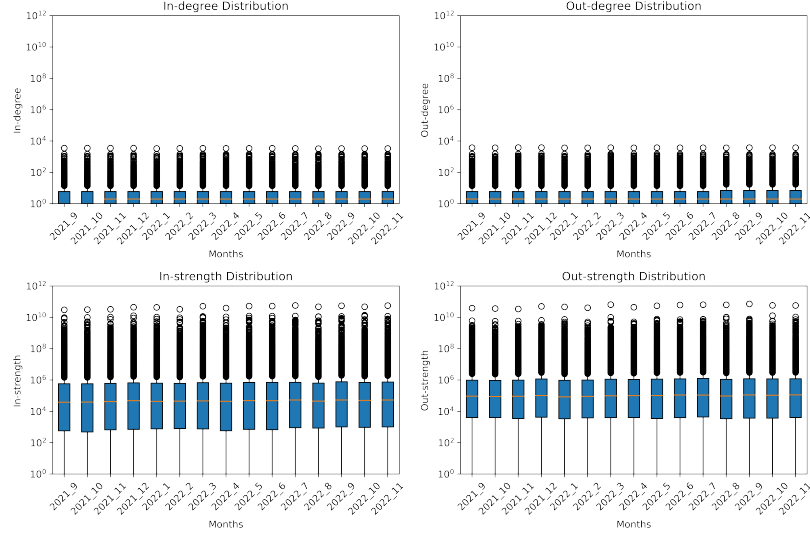


Figure 10: Distribution of indegree and outdegree, both weighted and un-weighted, across the months in G^{BIC} .

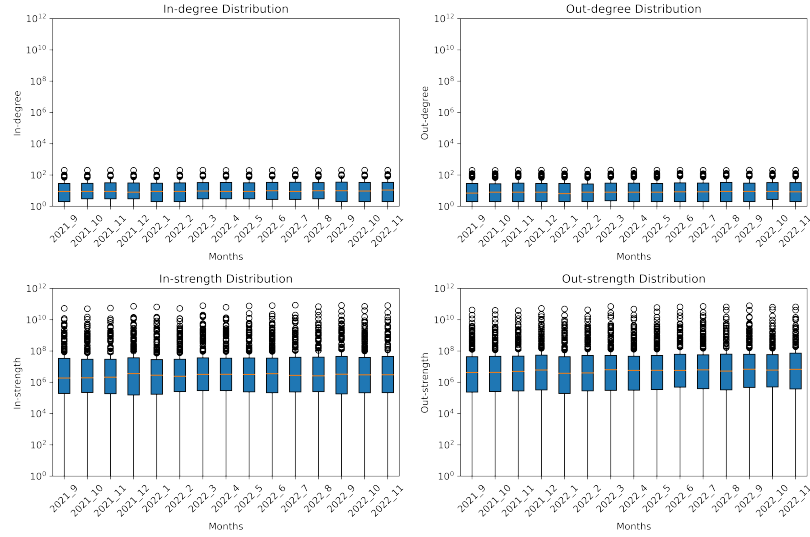


Figure 11: Distribution of indegree and outdegree, both weighted and un-weighted, across the months in G^{Country} .

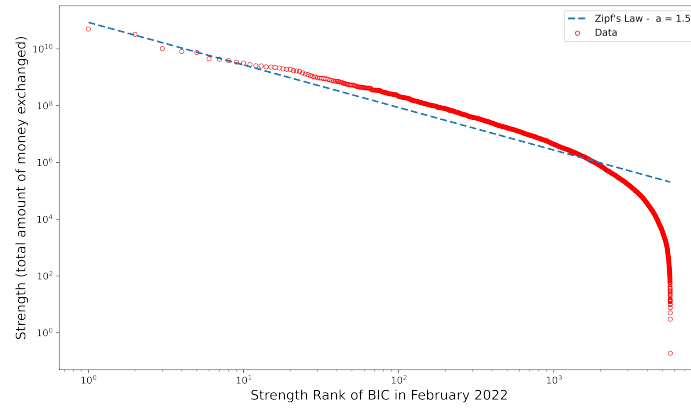


Figure 12: Amount of money moved by a node against its position on the Strength ranking, fitted to a Zipf's law with shape parameter $\alpha = 1.5$.

C Node rankings

Centrality measures can be used to rank nodes: we use rank intervals to identify classes of nodes (or links) with comparable properties, to assume that, under given circumstances, they have similar roles in the network and in its dynamics. Such statistical approach is commonly applied by plotting centrality distributions, that very often show heavy-tailed shapes, corresponding to highly heterogeneous classes of nodes. Focusing on degree centrality, the heterogeneity of the distribution leads to the emergence of nodes with very large (in)degree ¹, making statistically meaningless the adoption of the average $\langle k \rangle$ as a scale of degree centralities. In other words, the average degree should not be used as a good estimate of a randomly selected node’s degree in many real large scale networks. An example of an heterogeneous degree distribution can be see in Fig. 13, where we plot the degree distribution of one of the graphs built upon the dataset used for our case study.

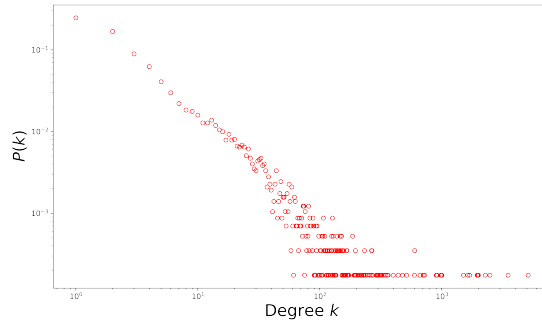


Figure 13: Degree distribution for $G_{\text{Feb2022}}^{\text{BIC}}$.

In our approach, we will compute node rankings based on various measures to track their evolution over time. Consequently, we require methods to compare different rankings.

For this purpose, we will employ the classic Spearman’s rank correlation coefficient, denoted as Spearman’s ρ . This non-parametric measure assesses the rank correlation between two sets of data.

Definition 1. *The Spearman correlation coefficient is defined as the Pearson correlation coefficient between two rank variables r_x and r_y :*

$$\rho(r_x, r_y) = \frac{\text{cov}(r_x, r_y)}{\sigma(r_x)\sigma(r_y)}$$

¹Please, observe that in the literature this is often referred as the “emergence of hubs phenomenon”, but the term “hub” is used there to indicate a high (in)degree node, whereas a node with high hub score, according the HITS algorithm, has a different meaning.

where $cov(r_x, r_y)$ is the covariance of the rank variables, and $\sigma(r_x), \sigma(r_y)$ are the standard deviations of the rank variables.

Another classic measure is the Kendall rank correlation coefficient, commonly denoted as Kendall’s τ coefficient. As Spearman’s ρ , Kendall’s τ is a measure of rank correlation: the similarity of the orderings of the data when ranked by each of the quantities. More precisely, it counts the number of concordant and discordant pairs, and the difference between the two quantities by the total number of possible pairs.

Both correlation coefficients have values in the range $[-1, 1]$, ranging from full disagreement to full agreement between the two series, with value equal or very close to zero when they are independent.

D Additional information on BIC graphs analysis

In Section 5.3 of the main paper we described the application of WeirNodes to the financial dataset provided by Intesa Sanpaolo, with the purpose of identifying anomalous BIC codes.

We executed some extra analysis that has been removed by the recent versions of the paper. They are presented here.

D.1 Ranking evolution charts

In Fig. 14 we show the resulting *ranking evolution charts*. The data points in orange are the top-30 nodes by residual that are gaining positions plus the top-30 that are losing positions in the rankings from Feb 2022 to Mar 2022.

D.2 Applying a threshold to filter out not relevant nodes

Considering the significantly larger quantity of data points compared to the prior study concerning transactions aggregated by countries, we added an optional filtering phase to prevent overloading the AFC analyst nodes with data that fails to satisfy certain baseline requirements. We employed a **threshold based filter** in this process. Adhering to the guidelines set by domain experts, we applied filters to the total strength, which refers to the total sum of money transacted by the node, at the BIC level of aggregation. More precisely, we keep nodes that correspond to:

- BICs belonging to Countries with **low** and **medium** financial risk - according to the ISP internal classification - that move more than $5 \cdot T_{hr}$ Euros in at least one of the two timeframes;
- BICs belonging to Countries with **high** financial risk - according to the ISP internal classification - that move more than T_{hr} Euros in at least one of the two timeframes

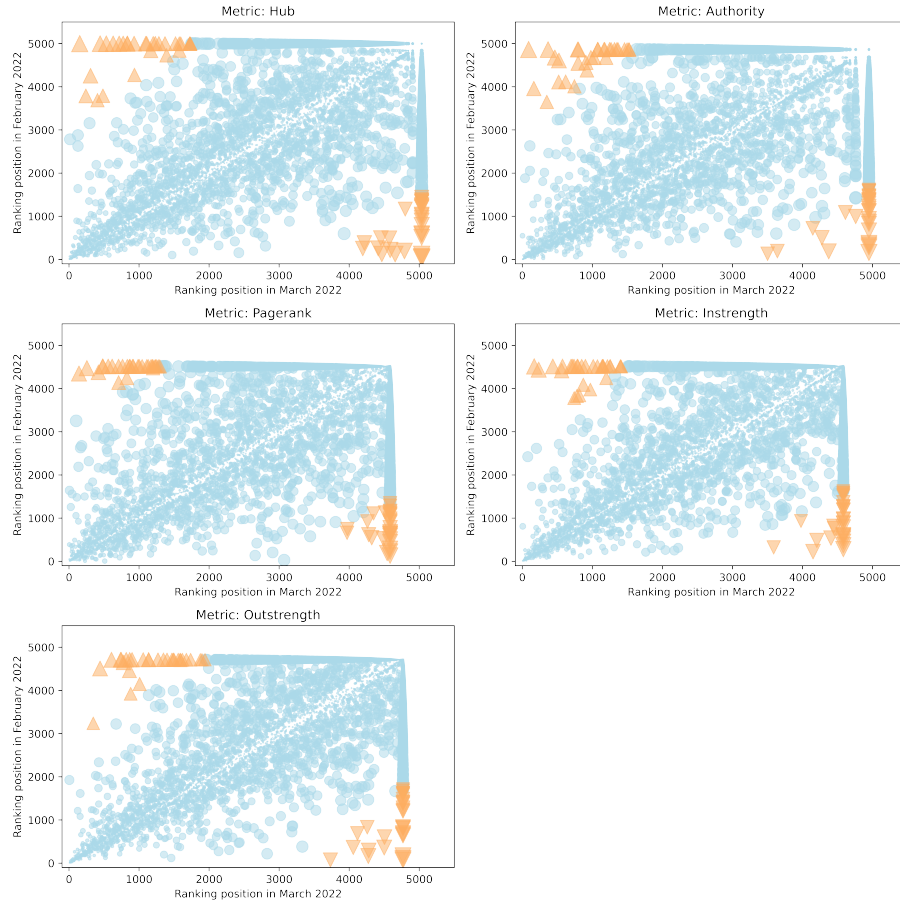


Figure 14: Step (5) of WeirdNodes applied on the BIC networks. The data points in light orange are the top-30 nodes that are gaining positions (triangles) plus the top-30 that are losing positions (inverted triangles) in the rankings from Feb 2022 to Mar 2022. Size is proportional to the absolute value of the residual. While there is a clear trend of stabler nodes close to the identity line, there is a selection of outliers with high residual that undergo significant changes in their ranking positions from February to March 2022.

	p@1	p@2	p@5	avg p@5	avg p@10	avg p@20	avg p@30	avg p@60	r* at 30	r* at 60
Hub	0.0	0.0	0.20	0.20	0.22	0.23	0.22	0.19	0.19	0.29
Authority	1.0	1.0	0.40	1.00	0.83	0.83	0.67	0.56	0.19	0.24
PageRank	0.0	0.0	0.20	0.20	0.22	0.22	0.22	0.19	0.19	0.33
Instrength	1.0	1.0	0.40	1.00	1.00	0.76	0.76	0.39	0.14	0.33
Outstrength	0.0	0.0	0.20	0.20	0.26	0.29	0.29	0.26	0.19	0.24
Average	0.4	0.4	0.28	0.52	0.51	0.47	0.43	0.32	0.18	0.29

Table 10: Evaluation of step (6): The application of the threshold based filter slightly improves the basic rankings’ performances.

where T_{hr} is an internal threshold of attention for high financial risk countries that will remain undisclosed. This procedure removes 29 unique BICs across the different centrality metrics.

The application of this filter, as shown in Table 10, (very) slightly improves the performance of some rankings by moving true positives upwards in the lists, improving precision-at-k for low values of k , w.r.t. performances shown in Sec. 5.3.1 of main paper. This improvement is only marginal; however, no true positives are removed by the filter, indicating the algorithm’s effectiveness even without prior domain knowledge. Other ML-based filtering methods (Isolation Forest and OneClassSVM) did not improve results, as shown in the following Appendix.

D.3 Are all the centrality based strategies necessary?

To evaluate if some centrality based strategy is unnecessary, we calculated correlations also between the different top-k lists. As evidenced in Fig. 15, according to Kendall’s tau, there is a strong concordance signal between the rankings produced by outstrength and hub score, along with a moderate correlation between those produced by instrength and both PageRank and authority scores. Nevertheless, the Spearman correlation among these rankings is very weak or negative, indicating the value in considering all the outputs.

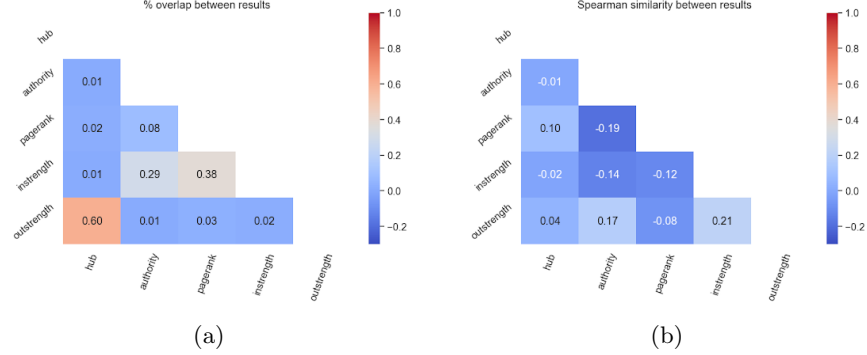


Figure 15: Correlations between the outputs on the different metrics. The similarity of the outputs in the case of BICs is slightly lower compared to the Country level of analysis. Nonetheless, the correlation between the outputs on the different metrics indicates that all centrality measures should be taken into account while forming the final list to avoid losing too much information.

E Comparison with machine learning based methods

The strategy of selecting the top-k nodes according to the different centrality metrics can be compared to other methods of anomaly detection based on machine learning techniques. As an example, we apply two popular techniques, OneClassSVM and IsolationForest. We apply them as an alternative to the hybrid strategy at step (7) of WeirdNodes: once the time-varying comparisons on each metric are performed, we can use ML based techniques on the RECs' output. Unlike what was done while applying the procedure, we do not pick the top-k nodes that lose (*-neg*) and gain (*-pos*) positions in the rankings, but we let the ML-based algorithms choose their own outliers. This is the only difference in an otherwise straightforward process: these outliers will then go through the usual remaining steps of the procedure, obtaining the final, stratified list.

In Tab. 11 we report detailed comparisons between our methodology, SVM and Isolation Forest on the individual network metrics, i.e., after their application on the RECs. The outputs obtained through OneClassSVM and Isolation Forest are no longer ranked, unlike the output of step 6 in our procedure; therefore, we look at the performances of the individual metrics in terms of simple precision and R^* . The values of ν and c are chosen in order to obtain an output whose size is in the same order of magnitude of N . This task itself is not always trivial, as it appears particularly evident from the size of the output of SVM on *PageRank_neg*, which is much larger due to a cluster of data points that overlap in the same position towards the bottom of the rankings both in February and March, and are all identified as outliers by the algorithm. This non-controllability of the size of the output through SVM and IF can indeed happen depending on the inherent distribution and characteristics of the data,

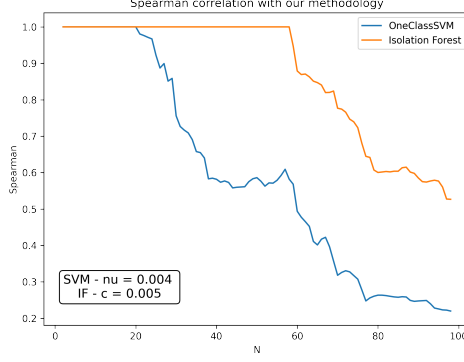


Figure 16: Comparison between the stratified ranking method and other machine learning based anomaly detection techniques.

	avg p@1	r* at 1	avg p@5	r* at 5	avg p@10	r* at 10	avg p@30	r* at 30	avg p@50	r* at 50	avg p@100	r* at 100
WeirdNodes	1.0	0.05	1.0	0.1	0.78	0.14	0.53	0.29	0.47	0.33	0.35	0.52
OneClassSVM	1.0	0.05	1.0	0.1	0.78	0.14	0.54	0.29	0.54	0.29	0.54	0.29
Isolation Forest	1.0	0.05	1.0	0.1	0.78	0.14	0.53	0.29	0.47	0.33	0.33	0.57

Table 11: Performance of the final output of WeirdNodes compared with OneClassSVM and IF. We notice a perfect overlap in the final outputs across the three methods, suggesting that no significant improvement is brought by the application of ML-based methods.

particularly in cases where there are clusters of overlapping points that challenge the algorithms’ ability to effectively differentiate or classify them. In terms of performances, at this stage we gather that the three methodologies achieve very similar results, with Isolation Forest performing slightly better considering both the average precision and the size of the outputs across all metrics.

This similarity is confirmed by Fig. 16, where we focus on the comparison of the final outputs of the pipeline with and without the intermediate application of ML-based filtering methods. As we can see, the outputs have perfect overlap up to $N \simeq 380$, decreasing their correlation after that threshold as N increases. This entails that, since we are interested in the top positions of the final list of potential anomalies, there is no improvement in applying ML-based filters.

Of course, we acknowledge that more sophisticated and tailored ML methods could be much more effective than those tested here. However, we recall that the list of outliers returned to the AFC analysis with WeirdNodes is fully interpretable, while machine learning methods would need an extra explainability layer.

F Analysis of the IBAN subnetworks

F.1 Application to the transactions dataset: going deeper with IBAN codes

We can apply our anomaly detection procedure to a finer-grained level, aiming to identify anomalies among individual bank accounts. The dataset includes anonymized IBAN codes for transactions between January and March 2022.

Using a monthly temporal resolution, we create three temporal layers: $G_{\text{Jan2022}}^{\text{IBAN}}$, $G_{\text{Feb2022}}^{\text{IBAN}}$, and $G_{\text{Mar2022}}^{\text{IBAN}}$.

In each graph, nodes represent bank accounts (identified by their IBAN codes), linked by edges if money was transferred between them within the considered time window. The edges are weighted by the total amount of money transferred. These graphs are extremely sparse ($\sim 3M$ nodes and $\sim 3M$ edges, with a density $\sim 10^{-7}$).

We executed steps (2)–(4) of the procedure on each graph to build centrality-based node rankings and then check for the validity and stability of the temporal property. Unfortunately, unlike the BICs case, the concordance between each ranking returns a low/moderate signal (coefficients < 0.5), independently of the centrality measure adopted, failing to ensure the required stability. Since the stability check is not fully successfully, we should not continue the anomaly detection procedure to identify outliers based on ranking position changes. However, the outputs from the BIC analysis in the previous section can be used as a starting point to explore IBAN transactions through other network analysis techniques.

This outcome is not entirely unexpected. The sparse nature of these graphs presents challenges, as noted in a seminal study on sparse graphs: “graphs with $\Theta(n)$ edges [...] turn out to be much harder to handle. Many new phenomena occur, and there are a host of plausible metrics to consider” [1]. Although there are visible differences between regular BICs and outliers, suggesting detectable activities at the IBAN level, establishing causality is complex. Thus, we should be particularly cautious to directly apply the same anomaly detection procedure used for BIC graphs.

Despite this, strategies can leverage information obtained from the Country and BIC levels. For instance, once anomalous BICs are identified, we can investigate by expanding the IBAN networks underlying these BICs. This allows the AFC analyst to use network-based techniques to gain insights into fine-grained transactions between individual accounts, examining the money exchanges that led to anomalies at the BIC level. An example of this approach is provided below.

More specifically, the IBAN network is built by:

1. selecting a BIC;
2. considering all the IBANs belonging to the selected BIC and the edges among them;

3. adding and following all the edges starting from or arriving to these IBANs to include other IBANs;
4. including all the edges between these new IBANs;
5. adding and following all the edges starting from or arriving to these IBANs to include another set of IBANs.

By doing so, we highlight interesting differences between the networks underlying potential anomalous and regular BICs and, most importantly, between *true positive* and *false positives* BICs.

As an example, we consider an analysis on some *true positive* BICs, compared against *false positives* and *regular* BICs. Considering two different timeframes on which we are looking for anomalies (in this example, February and March 2022), we can build the IBAN networks underlying selected BICs and inspect the changes from one timeframe to the other. We come across two more frequent scenarios: i) the IBAN network changes dramatically in size, gaining or losing several nodes and edges, i.e., changing from a topological perspective or ii) the IBAN network does not change significantly, but inspecting it is still interesting because of the insights gained by looking at suspicious transactions.

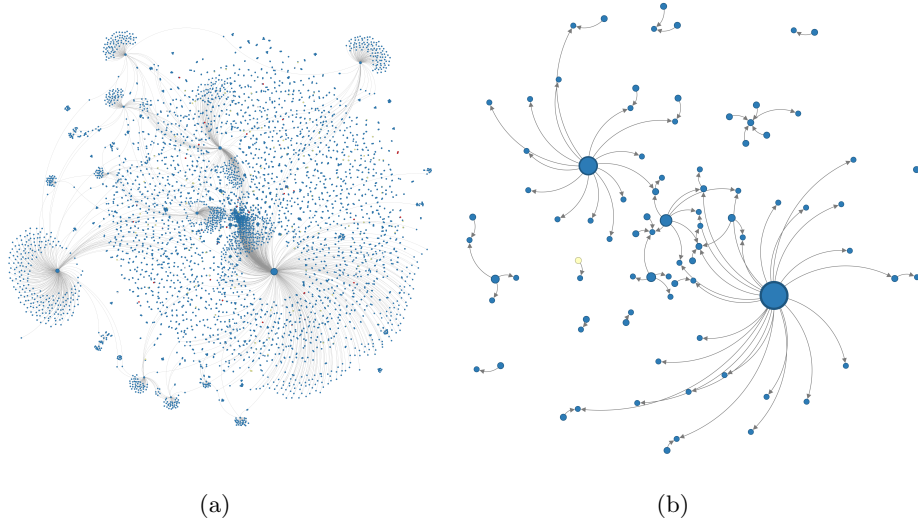


Figure 17: IBAN network underlying the BIC JEQSIT21 in February (a) and March (b). We can notice the drastic drop in nodes and connections. Nodes and edges are colored according to the financial risk of their Country (green for low, yellow for medium and red for high risk).

An example of case i) can be seen in Fig. 17. Here we build and compare the IBAN networks in February and March for BIC JEQSIT21², which was marked

²As in any other phase of the analysis, BICs and IBANs shown in this section are fully

as *true positive* in the manual annotation. In this case the network goes from 6128 nodes and 4472 edges to 86 nodes and 81 edges, suggesting that it could be interesting to manually inspect the few, remaining nodes that are still involved in transactions with the anomalous BIC.

An example of ii) is provided in Fig. 18. Here we build the IBAN networks for the BIC JLPCGISQ, again in February and March 2022. In this case the network does not change dramatically in size, but all the transactions involve countries with medium financial risk, making it worth to be inspected.

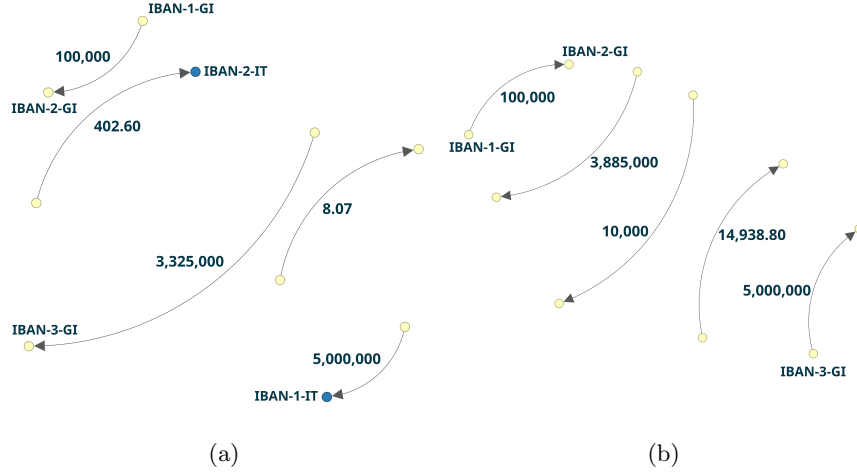


Figure 18: IBAN network underlying the Gibraltar BIC JLPCGISQ in February (a) and March (b). In this case the network does not change dramatically in size, but all the transactions involve countries with medium financial risk.

G Stratified Sorting strategy

The adoption of the stratified strategy (see Algo. 1) at step (6) of WeirdNodes is supported by the observation that novel outliers tend to be positioned more prominently at the top of each basic top-k list. Hence, the evaluation of true positives' placement within each list (see Fig. 19) gave us the idea of prioritizing nodes already ranking high with basic strategies.

Algorithm 1 Stratified Sorting strategy

```
1: procedure STRATIFIEDSORTING(sortedLists,  $K$ )
2:    $\triangleright$  sortedLists: a list of top-k sorted lists of nodes;  $k$ : desired top-k size
3:    $\triangleright$  return topKList: a consolidated top-k list
4:
5:   positionGroups  $\leftarrow \{\}$ 
6:    $\triangleright$  To group nodes sharing identical positions across all lists
7:
8:   for  $i = 1$  to  $k$  do
9:     group  $\leftarrow []$ 
10:    for each sortedList  $\in$  sortedLists do
11:      group.append(sortedList[ $i$ ])
12:    end for
13:    positionGroups[ $i$ ]  $\leftarrow$  group
14:  end for
15:  consolidatedGroups  $\leftarrow []$ 
16:  for each group  $\in$  positionGroups do
17:    Sort group by  $\Delta_{HRA}$  in descending order
18:    consolidatedGroups.append(group)
19:  end for
20:  uniqueNodes  $\leftarrow \{\}$ , topKList  $\leftarrow []$ 
21:  for each group  $\in$  consolidatedGroups do
22:    for each node  $\in$  group do
23:      if node  $\notin$  uniqueNodes then
24:        uniqueNodes.add(node)
25:        topKList.append(node)
26:        if |topKList| =  $k$  then
27:          return topKList
28:        end if
29:      end if
30:    end for
31:  end for
32:  return topKList
33: end procedure
```

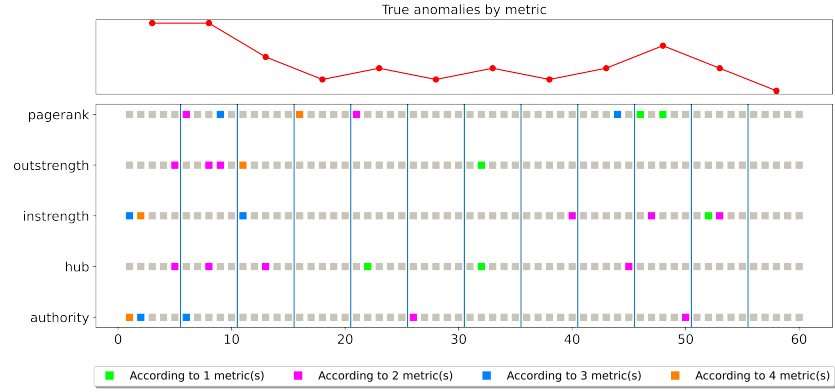


Figure 19: Positions of true positives in the rankings based on the different centrality metrics. The line chart in the top panel represents the ratio true/false positives across all the metrics, showing a decreasing trend as we move away from the top positions of the rankings.

References

- [1] B. Bollobás and O. Riordan. Sparse graphs: metrics and random models.
Random Structures & Algorithms, 39(1):1–38, 2011.