# GDS (Neo4j Graph Data Science)

GDS Algorithms

## variants

- Named graph variant
  - The graph to operate over will be read from the graph catalog.
- Anonymous graph variant
  - The graph to operate over will be created and deleted as part of the algorithm execution

Each syntax variant additionally provides different execution modes. These are the supported execution modes:

- stream
  - Returns the results of the algorithm as a stream of records
- stats
  - Returns a single record of summary statistics, but does not write to the Neo4j database
- mutate
  - Writes the results of the algorithm to the in-memory graph and returns a single record of summary statistics. This mode is designed for the named graph variant, as its effects will be invisible on an anonymous graph.
- write
  - Writes the results of the algorithm to the Neo4j database and returns a single record of summary statistics.

## Centrality algorithms

- Production-quality
  - Page Rank
  - Article Rank
  - Eigenvector Centrality
  - Betweenness Centrality
  - Degree Centrality
- Alpha
  - Closeness Centrality
  - Harmonic Centrality
  - HITS
  - Influence Maximization

## Community detection algorithms

Community detection algorithms are used to evaluate how groups of nodes are clustered or partitioned, as well as their tendency to strengthen or break apart.

- Production-quality
  - Louvain
  - Label Propagation
  - Weakly Connected Components
  - Triangle Count
  - Local Clustering Coefficient

## Similarity algorithms

Similarity algorithms compute the similarity of pairs of nodes using different vector-based metrics.

- Production-quality
  - Node Similarity
- Beta
  - K-Nearest Neighbors
- Alpha
  - Approximate Nearest Neighbors
  - Cosine Similarity
  - Euclidean Similarity
  - Jaccard Similarity
  - Overlap Similarity
  - Pearson Similarity

## Path finding algorithms

- Production-quality
  - Dijkstra Source-Target
  - Dijkstra Single-Source
  - A*
  - Yen's algorithm

## Link Prediction algorithms

- Alpha
  - Adamic Adar
  - Common Neighbors
  - Preferential Attachment
  - Resource Allocation
  - Same Community
  - Total Neighbors

## Node embeddings

Node embedding algorithms compute low-dimensional vector representations of nodes in a graph. These vector, also called embeddings, can be used for machine learning.

- Production-quality
    - FastRP
- Beta
    - GraphSAGE
    - Node2Vec

## Machine Learning Models

The machine learning procedures in Neo4j GDS allow you to train supervised machine learning models. Models can then be accessed via the Model Catalog and used to make predictions about your graph.

To help with working with ML models, these are additional guides for pre-processing and hyperparameter tuning available in:

- Pre-processing
- Tuning parameters for training

---

- Alpha
    - Node Classification
    - Link Prediction

## Auxiliary procedures

Auxiliary procedures are extra tools that can be useful in your workflow. The Neo4j GDS library includes the following auxiliary procedures, grouped by quality tier:

- Beta
    - Graph Generation
- Alpha
    - Collapse Path
    - Scale Properties
    - One Hot Encoding
    - Split Relationships

---

- Pregel API

---

## 관련 문서

**Plugin Backlinks: 아무 것도 없습니다.**

From:
https://jace.link/ - **Various Ways**

Permanent link:
**https://jace.link/open/gds**

Last update: **2021/09/01 05:50**