Pivotal

Greenplum人工智能工具集—MADLib与图数据分析

Pivotal资深产品经理 吴疆(jwu@pivotal.io)

© Copyright 2019 Pivotal Software, Inc. All rights Reserved.

Agenda

1. 为什么要分析图数据?

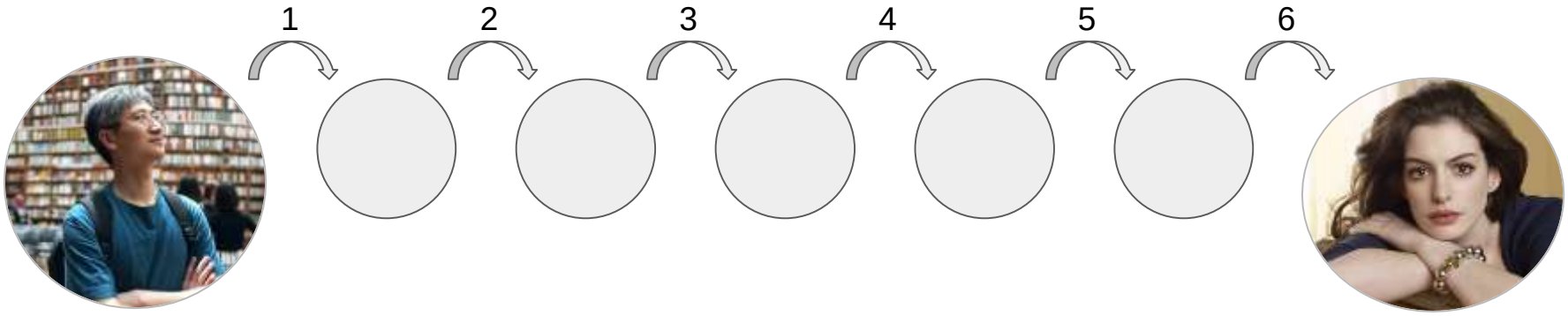
2. 图数据分析的内容

3. 使用Greenplum+MADLib分析图数据

世界是连接在一起的!

“6 Degrees of Separation”

1973, Stanley Milgram, Small-world experiment

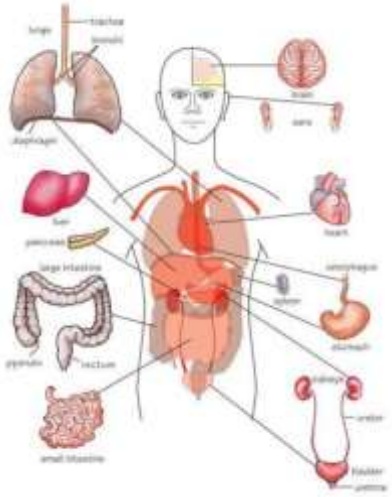


整体论和还原论

还原论 vs. **整体论**

“**把复**杂**的系**统分解为简单的系统进行分析”

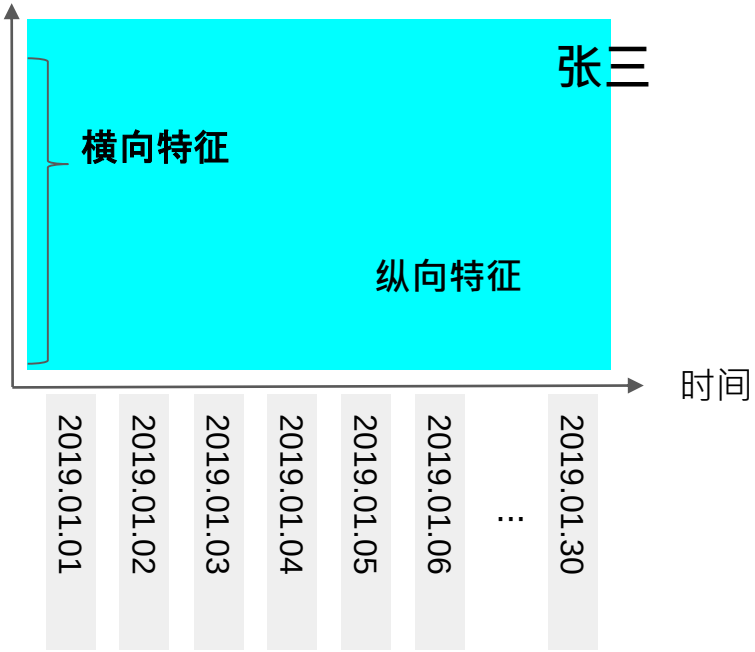
“**复**杂**的系**统需要从整体进行理解”



社会学：个体和关系

特征

- 外貌
- 行为特点
- 性格特点
- 经济条件
- ⋮
- 教育背景



社会学：个体和关系

特征

外貌

行为特点

性格特点

经济条件

⋮

教育背景

家庭

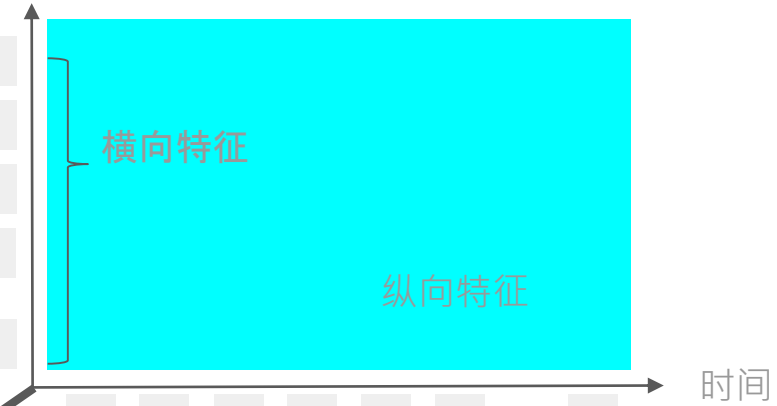
朋友

同事

⋮

社区

社会关系



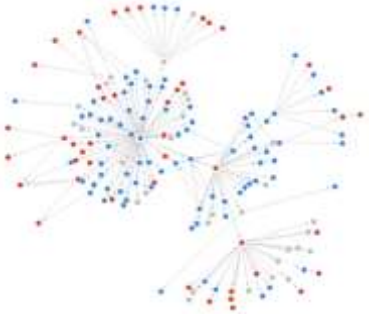
“**Tell me who your friends are** and **I’ll tell you who you are**”
- 墨西哥谚语 -

世界是普遍联系的

社交网络

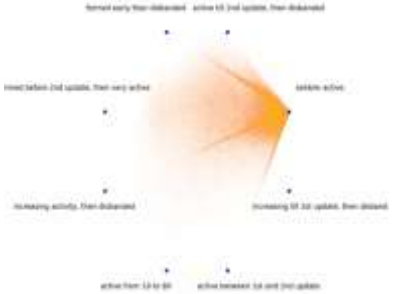
* Grandjean, M. (2016)

传染病学



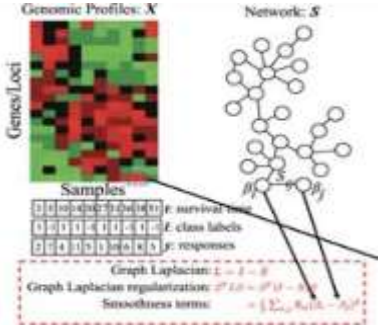
* http://www.netminer.com/community

网络游戏



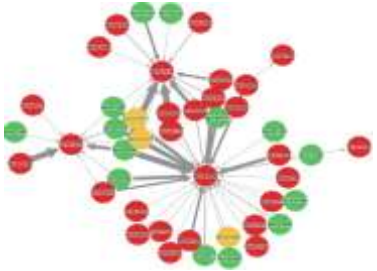
* www.researchgate.net

化学



* https://www.nature.com/articles/

银行风控



* https://cambridge-intelligence.com

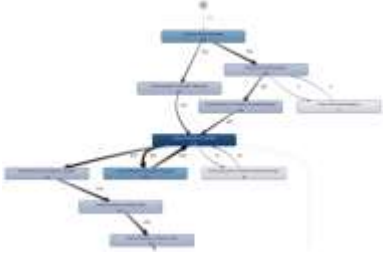
反欺诈

* www.infoglide.com

基因工程

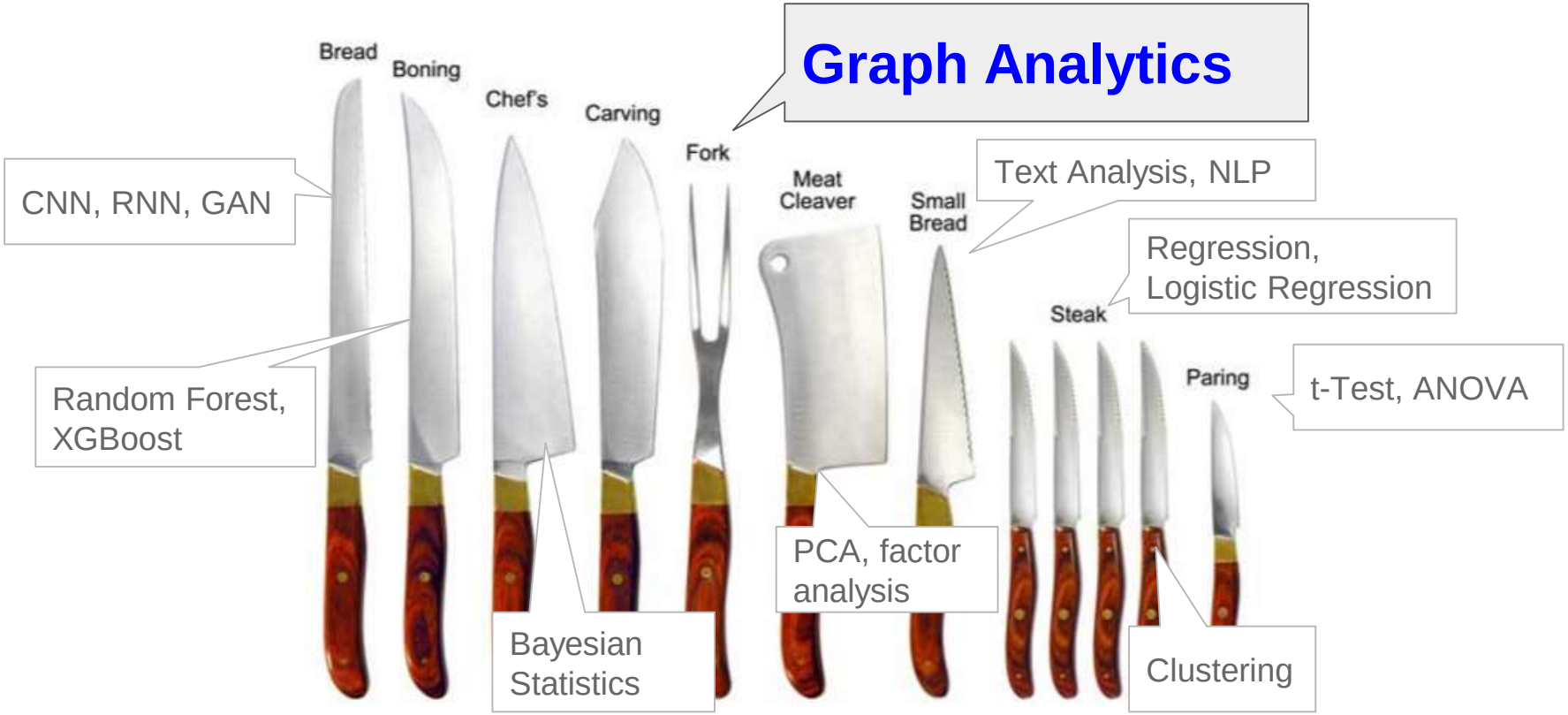
* www.researchgate.net

制造业



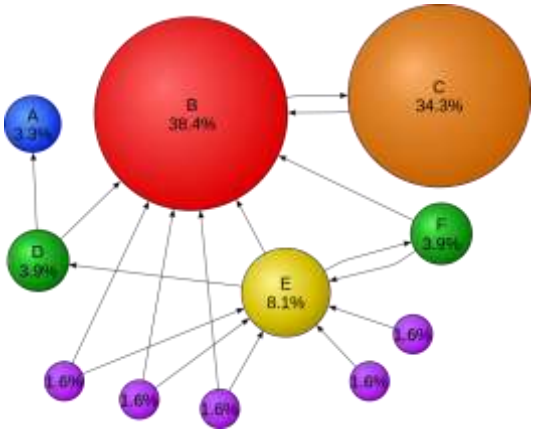
* https://blog.trifinance.com

图数据分析已经成为数据科学家的重要技能



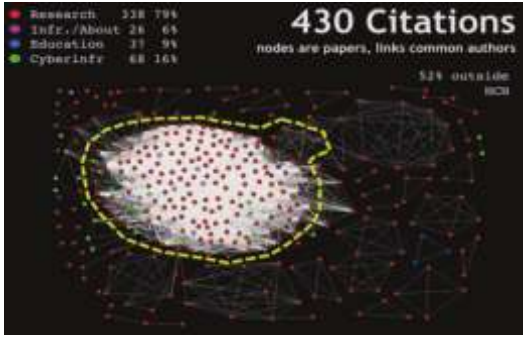
图分析应用 - PageRank

- **根据**图的顶点相关联边的数量和权重来计算顶点的重要程度



* Image from https://en.wikipedia.org/wiki/PageRank

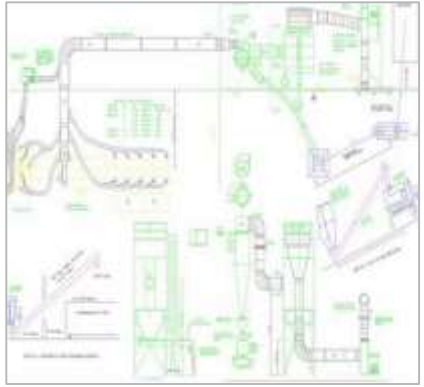
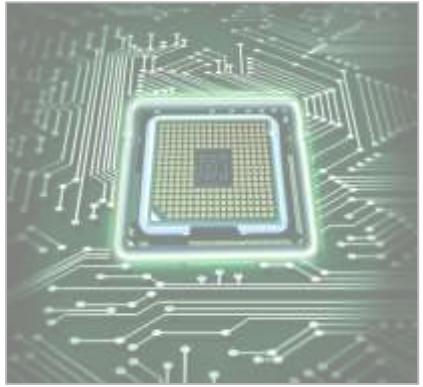
- ❑ **搜索引擎**
- ❑ **科学文件引用因子**
- ❑ 脑科学
- ❑ **城市**规划



图分析应用 – 单源最短路径

- 计算图中某个顶点到其他顶点的最短路径

  - ❑ 导航
  - ❑ **社交**网络
  - ❑ **通**讯网络的延迟
  - ❑ **城市**规划
  - ❑ **超大**规模集成电路设计

Agenda

1. 为什么要分析图数据？
2. **图数据分析的内容？**
3. 使用Greenplum+MADLib分析图数据

图论的起源

- 哥尼斯堡的七桥问题
- 大数学家欧拉的解决了七桥问题并开创了图论

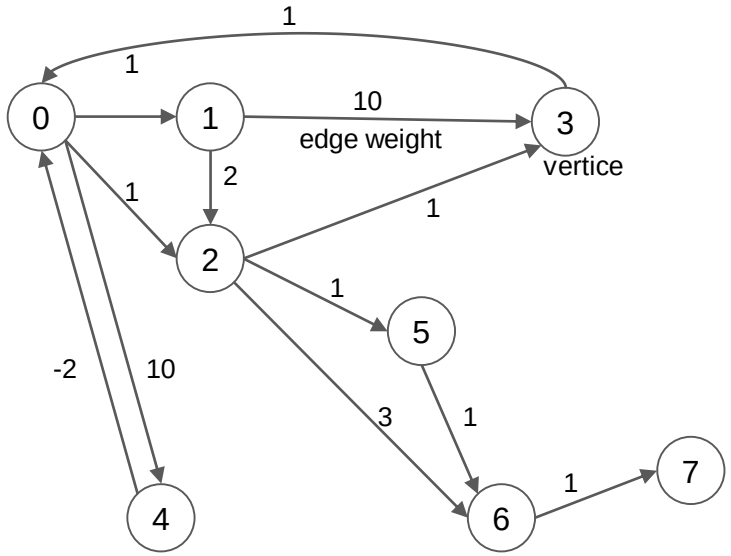
“能否一次走遍这7座桥，并且每座桥只通过1次？”

Euler, 1753

什么是图论?

- 图论**是用来研究事物及其**联系的数学分支.

图的示例



图的基本概念

常用的图算法

图的基本特征	分类	问题	算法
	1 分组问题	*“如何划分子图？”*	弱连通分量
	2 结构问题	*“图的连接结构有哪些特征？”*	密度, 直径, 平均路径长度
	3 路径问题	*“最短路径问题”*	单源最短路径, 全源最短路径, 广度优先搜索
	4 中心性	*“如何定义图的顶点的重要性？”*	顶点的度, 接近中心性, PageRank, HITS

图的常用算法- (1) 分组问题

分组

结构

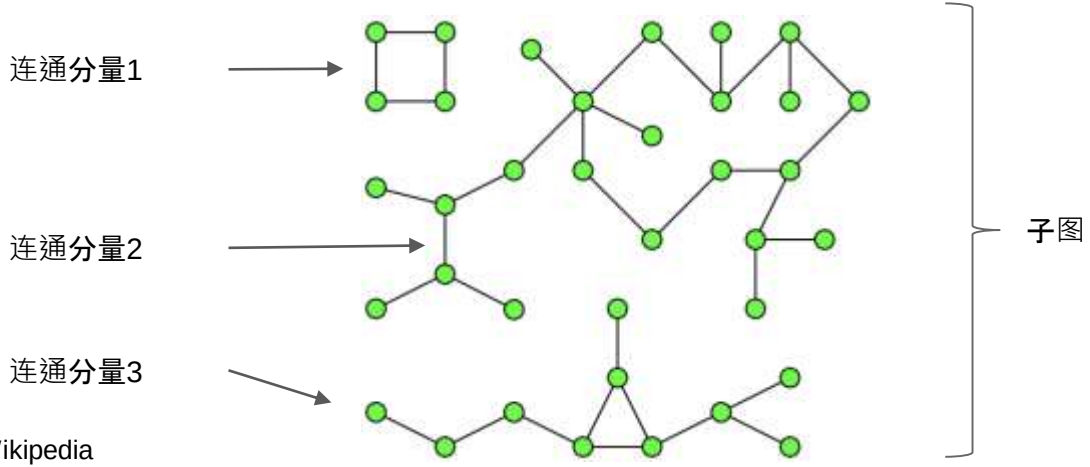
路径

中心性

弱连通分量

- 弱连通分量（Weakly Connected Component,WCC)**是指把一个有向**图弱化为无**向**图之后图的极大联通子图。对于无向图来说，弱连通分量和强连通分量等同。

[示例]



* source: Wikipedia

图的常用算法 - (2) 结构

分组

结构

路径

中心性

密度

- 图的密度是指给定图的边数与当此图成为完全图时所包含边数的比值

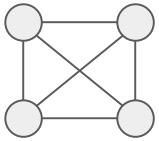
❑ 无向图

$$D = \frac{|E|}{|V|\,(|V| - 1) / 2}$$

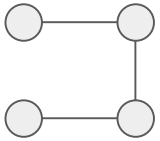
❑ 有向图

$$D = \frac{|E|}{|V|\,(|V| - 1)}$$

[示例]



$$D = \frac{|6|}{|4|(|4|-1)/2} = 1$$



$$D = \frac{|3|}{|4|(|4|-1)/2} = 0.5$$

* source: Wikipedia

图的常用算法 - (3) 路径算法

分组

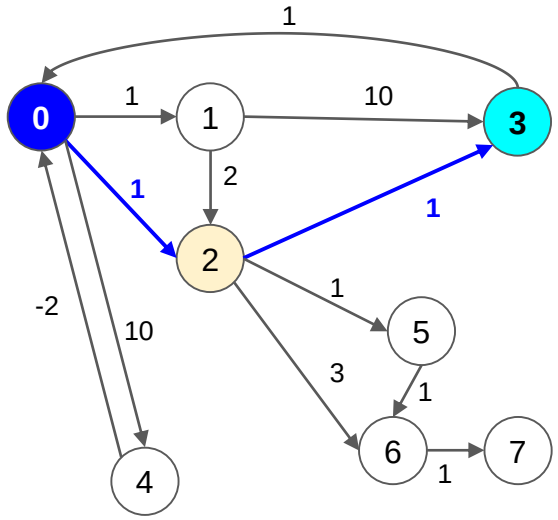
结构

路径

中心性

单源最短路径Single Source Shortest Path (SSSP)

- 从给定顶点到其他顶点的路径的最小值



[顶点0到其他顶点的最短路径]

ID	weight	parent
0	0	0
1	1	0
2	1	0
3 (3)	2 (= 1+1)	2 (2)
4	10	0
5	2	2
6	3	5
7	4	6

* weight : The total weight of the shortest path from the source vertex to this particular vertex.
* parent : The parent of this vertex in the shortest path from source.

图的常用算法- (4) 中心性

分组

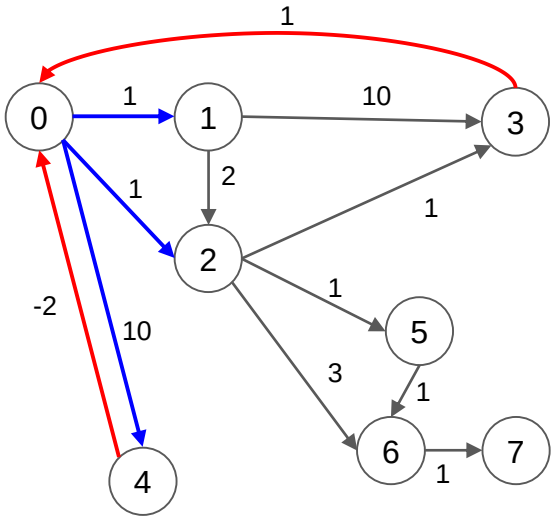
结构

路径

中心性

顶点的In-Degree, Out-Degree

- 顶点的**in-degree 是指以此**顶点为终点的边的条数
- 顶点**的out-degree是指以此**顶点为起点的边的条数



ID	In-degree	Out-degree
0	**2**	**3**
1	1	2
2	2	3
3	2	1
4	1	1
5	1	1
6	2	1
7	1	0

图的常用算法 - (4) 中心性

分组

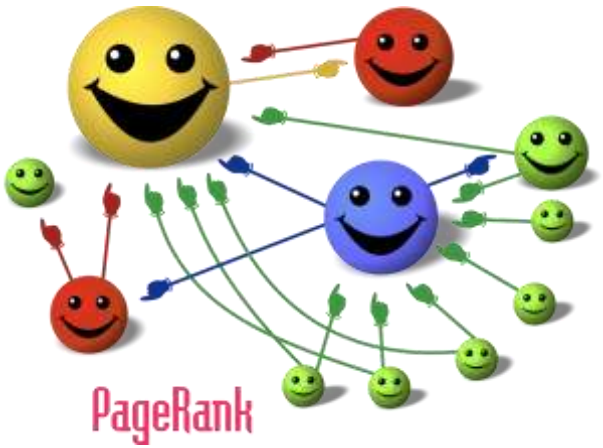
结构

路径

中心性

PageRank (1 / 2)

- PageRank**又称网**页排名算法，因发明者Larry Page**而得名**。PageRank**通**过计算连接到某个网页的连接的数量和质量来评估网页的重要程度。



The size of each face is ***proportional to the total size of the other faces which are pointing to it.***

$$PR(A) = \frac{1-d}{N} + d\left(\frac{PR(B)}{L(B)} + \frac{PR(C)}{L(C)} + \frac{PR(D)}{L(D)} + \cdots\right)$$

- PR(A): PageRank of node A
- N: the total number of Nodes
- L(B): the number of Links from node B
- d: damping factor (probability, at any step, that a surfer will continue randomly clicking on links)

* source: Wikipedia

图的常用算法- (4) 中心性

分组

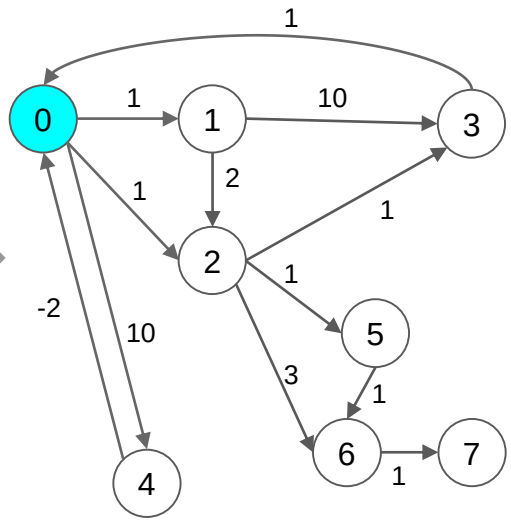
结构

路径

中心性

接近程度（Closeness）

- 顶点的接近中心性，即图的某个顶点与图的中心的接近程度，一般通过计算此顶**点到其他的**顶点的最短路径的和的倒数来衡量。



[全源最短路径]

source	destination	weight
0	0	**0**
0	1	**1**
0	2	**1**
0	3	**2**
0	4	**10**
0	5	**2**
0	6	**3**
0	7	**4**
1	0	4
1	1	0
1	2	2
1	3	3
1	4	14
1	5	3
1	6	4
1	7	5

⋮

[接近中心性]

src_id	closeness
0	0.043
1	0.028
2	0.041
3	[illegible]

$$C(x) = \frac{1}{\sum_y d(y,x)}$$

$$C(0) = \frac{1}{(1+1+2+10+2+3+4)} = 0.043$$

* N: The number of nodes, * d(y, x) : The distance between y and x node

图算法的算法复杂度

Type	Algorithms/ Measures	Time Complexity
Group	Weakly-Connected Component	**O(|V| + |E|)**
Structure	Density	**O(|V|+|E|log|E|)**
	Diameter	**O(|V|^3)**
Path	All Pairs Shortest Path	**O(|V|^3)**
	Single Source Shortest Path	**O(|V|^2)**
	Breadth-First Search	**O(E + V)**
Centrality	In-Degree, Out-Degree	**O(|V| + |E|)**
	Closeness Centrality	**O(|V|^3)**
	PageRank	**O(log*(network size)*/(1-damping factor))**
	Betweenness Centrality	**O(|V|^{2}log|V|+|V||E|)**

* |V|: 顶点个数
* |E|: 边的条数

- 图算法一般都是计算密集型
  - 计算量随着边数和顶点个数的增加增长很快

Graph Analysis
at Scale,
parallel processing
with MADlib
on Greenplum

Agenda

1. 为什么要分析图数据?

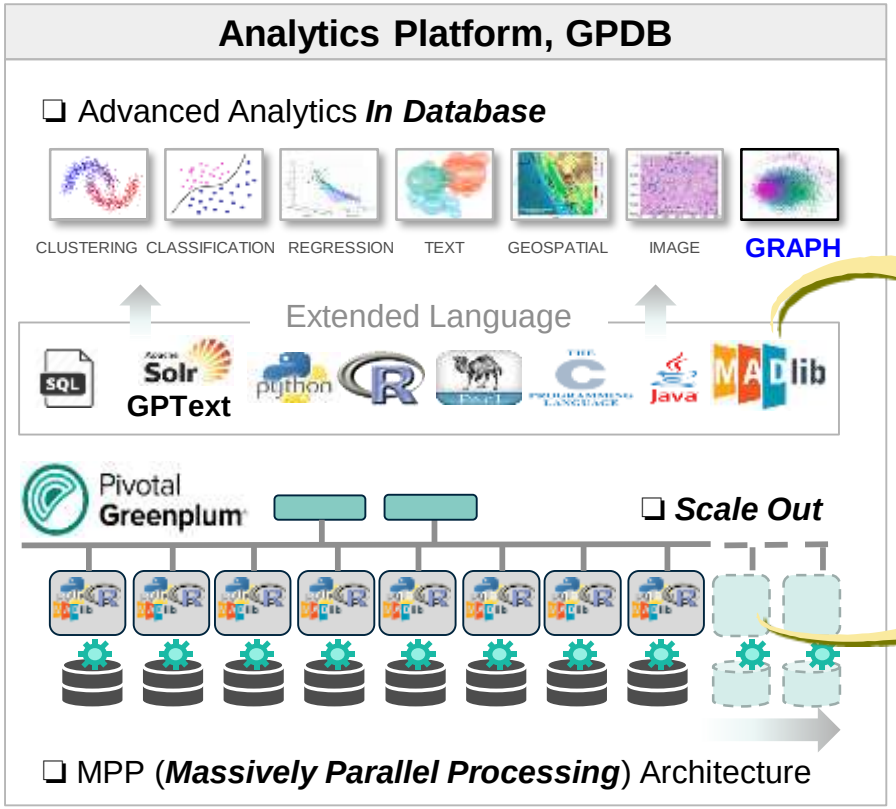
2. 图数据分析的内容?

3. **使用Greenplum + MADLib分析图数据**

图数据分析工具

	小数据量	大数据量
开源	sna, igraph, ergm, network NetworkX, graph-tool, SNAP, pygraphviz Gephi graphviz	MADlib Pivotal Greenplum
商业软件	NetMiner UCINET ⋮ Interactive visualization focused	neo4j DATASTAX OrientDB Graph DB

可扩展的图数据分析平台



Greenplum + MADLib图数据分析平台

- **超大**规模处理能力（十亿级别顶点数或/**和**边数）
- **不需要移动数据，in-database**
- **SQL 接口**

MADlib：可扩展的, In-Database 机器学习工具集

Apache MADlib: Big Data Machine Learning in SQL

Open source, **top level Apache project**

For PostgreSQL and Greenplum Database

Powerful machine learning, graph, statistics and analytics for data scientists

- Open source https://github.com/apache/madlib
- Downloads and docs http://madlib.apache.org/
- Wiki https://cwiki.apache.org/confluence/display/MADLIB/

MADlib：图分析相关算法

Supervised Learning

Neural Networks
Support Vector Machines (SVM)
Conditional Random Field (CRF)
Regression Models
- Clustered Variance
- Cox-Proportional Hazards Regression
- Elastic Net Regularization
- Generalized Linear Models
- Linear Regression
- Logistic Regression
- Marginal Effects
- Multinomial Regression
- Naïve Bayes
- Ordinal Regression
- Robust Variance

Tree Methods
- Decision Tree
- Random Forest

Unsupervised Learning

Association Rules (Apriori)
Clustering (k-Means)
Principal Component Analysis (PCA)
Topic Modelling (Latent Dirichlet Allocation)

Nearest Neighbors

- k-Nearest Neighbors

Graph

All Pairs Shortest Path (APSP)
Breadth-First Search
Hyperlink-Induced Topic Search (HITS)
Average Path Length
Closeness Centrality
Graph Diameter
In-Out Degree
PageRank and Personalized PageRank
Single Source Shortest Path (SSSP)
Weakly Connected Components

Utility Functions

Columns to Vector
Conjugate Gradient
Linear Solvers
- Dense Linear Systems
- Sparse Linear Systems

Mini-Batching
PMML Export
Term Frequency for Text
Vector to Columns

Sampling

Balanced/ Random/ Stratified Sampling

Time Series Analysis

- ARIMA

Data Types and Transformations

Array and Matrix Operations
Matrix Factorization
- Low Rank
- Singular Value Decomposition (SVD)

Norms and Distance Functions
Sparse Vectors
Encoding Categorical Variables
Path Functions
Pivot
Sessionize
Stemming

Statistics

Descriptive Statistics
- Cardinality Estimators
- Correlation and Covariance
- Summary

Inferential Statistics
- Hypothesis Tests

Probability Functions

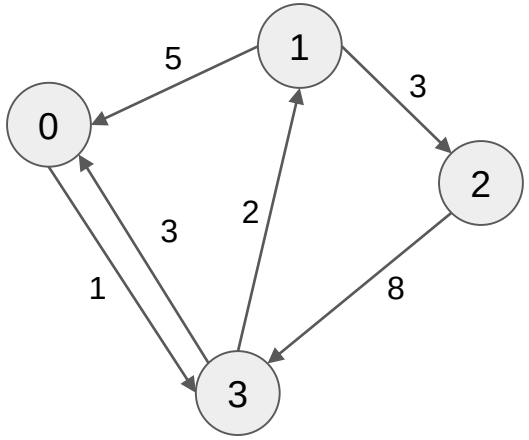
Model Selection

Cross Validation
Prediction Metrics
Train-Test Split

MADlib :图的表示：顶点表+边表

Pivotal Greenplum

[示例图]



顶点表 V

Vertex	Vertex Params
0	...
1	...
2	...
3	...

⋮

边表 →

Source Vertex	Dest Vertex	Edge Weight	Edge Params
0	3	1.0	...
1	0	5.0	...
1	2	3.0	...
2	3	8.0	...
3	0	3.0	...
3	1	2.0	...

⋮

示例：PageRank in MADlib

- 创建**相关的**顶点表和边表

```
DROP TABLE IF EXISTS vertex;
CREATE TABLE vertex(
    id INTEGER
    );

INSERT INTO vertex VALUES
(0),
(1),
(2),
(3),
(4),
(5),
(6);
```

```
DROP TABLE IF EXISTS edge;
CREATE TABLE edge(
    src INTEGER,
    dest INTEGER,
    user_id INTEGER
    )
DISTRIBUTED BY (user_id);
```

```
INSERT INTO edge VALUES
(0, 1, 1), (0, 2, 1), -- user id 1
(0, 4, 1), (1, 2, 1),
(1, 3, 1), (2, 3, 1),
(2, 5, 1), (2, 6, 1),
(3, 0, 1), (4, 0, 1),
(5, 6, 1), (6, 3, 1),
(0, 1, 2), (0, 2, 2), -- user id 2
(0, 4, 2), (1, 2, 2),
(1, 3, 2), (2, 3, 2),
(3, 0, 2), (4, 0, 2),
(5, 6, 2), (6, 3, 2);
```

* http://madlib.apache.org/docs/latest/group__grp__pagerank.html

示例：PageRank in MADlib

- 使用MADLib在一个图上运行PageRank算法

```
DROP TABLE IF EXISTS pagerank_out, pagerank_out_summary;
SELECT madlib.pagerank(
                    'vertex'                    -- Vertex table
                    , 'id'                                          -- Vertex id
column
                    , 'edge'                    -- Edge table
                    , 'src=src, dest=dest'      -- Comma delimited string of
edge arguments
                    , 'pagerank_out'            -- Output table of RageRank
                    , NULL);                    -- Damping factor (default 0.85)
```

```
SELECT * FROM pagerank_out
ORDER BY pagerank DESC;
```

	id integer	pagerank double precision
1	0	0.287518161212111
2	3	0.210171199451415
3	2	0.146637377532288
4	4	0.102910437211324
5	1	0.102910437211324
6	6	0.0972746644343418
7	5	0.0525777229481976

MADlib example：PageRank in MADlib

- 使用MADLib在一个图的子图上运行PageRank算法

```
DROP TABLE IF EXISTS pagerank_gr_out, pagerank_gr_out_summary;
SELECT madlib.pagerank(
        'vertex'                    -- Vertex table
        , 'id'                                  -- Vertex id column
        , 'edge'                    -- Edge table
        , 'src=src, dest=dest'      -- Comma delimited string of edge
arguments
        , 'pagerank_gr_out'         -- Output table of PageRank
        , NULL                      -- Default damping factor (0.85)
        , NULL                      -- Default max iterations (100)
        , 0.00000001                -- Threshold
        , 'user_id'                 -- Grouping column name
        );
```

```
SELECT * FROM pagerank_gr_out
ORDER BY user_id, pagerank DESC;
```

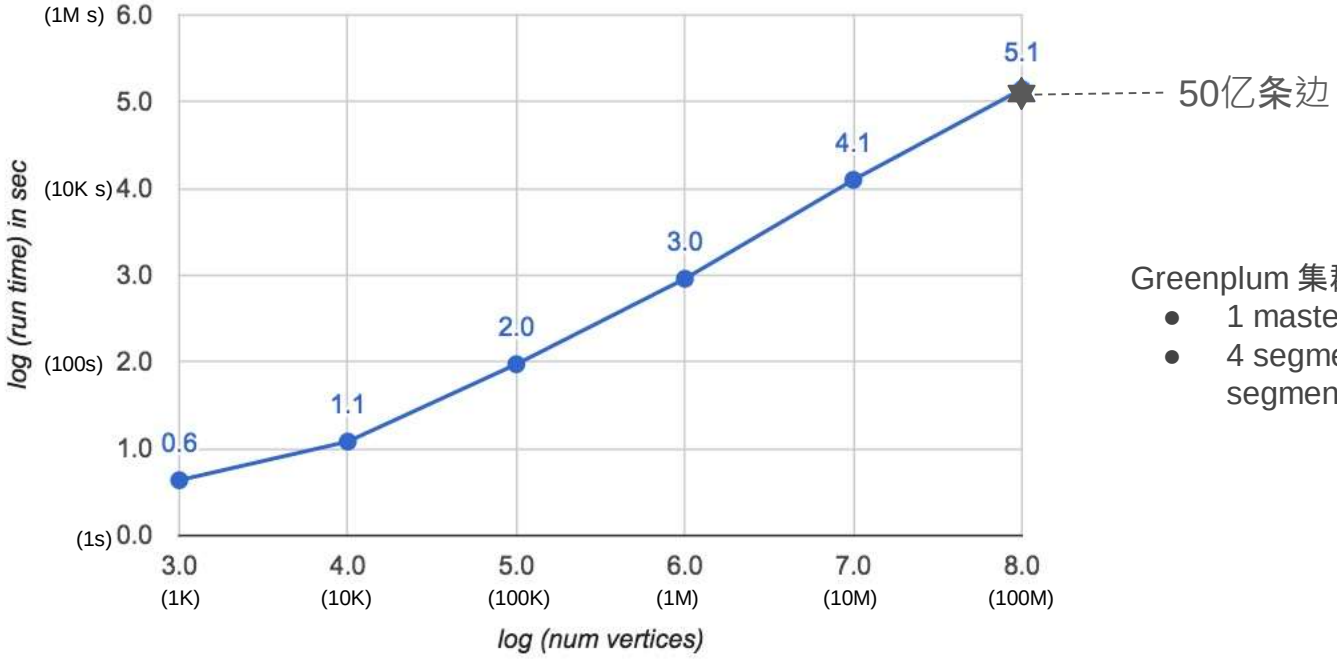
	user_id integer	id integer	pagerank double precision
1	1	0	0.278254883885528
2	1	3	0.201881146670752
3	1	2	0.142881123460599
4	1	6	0.114536378321472
5	1	1	0.10026745615438
6	1	4	0.10026745615438
7	1	5	0.0619115553528898
8	2	0	0.318546250041731
9	2	3	0.237866867733431
10	2	2	0.159148764893974
11	2	1	0.111683344379718
12	2	4	0.111683344379718
13	2	6	0.0396428571428571
14	2	5	0.0214285714285714

PageRank of user id 1

PageRank of user id 2

示例：PageRank in MADlib

PageRank 性能测试

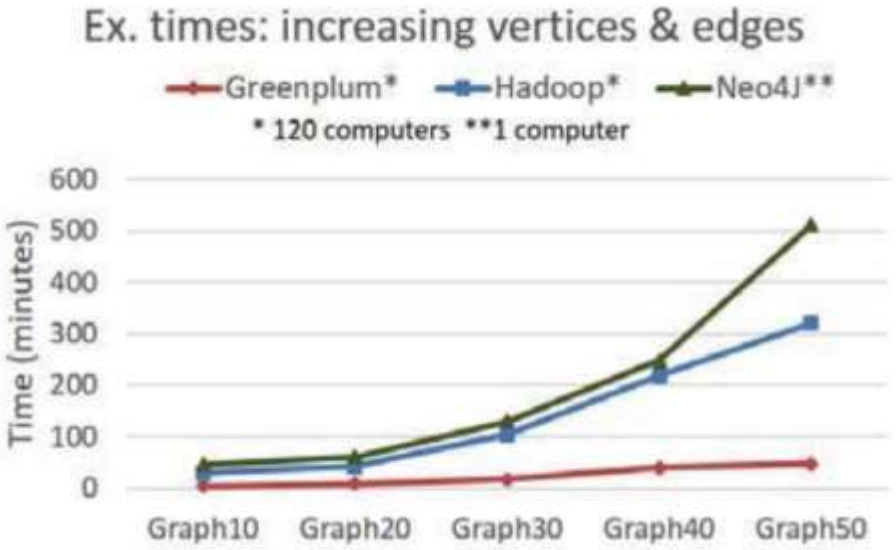


Greenplum 集群规模：

- 1 master
- 4 segment hosts with 6 segments per host

* Note: log-log scale

性能评测



* source: F.d.S.Fernandoes, E.J.H.Yero, ‘Parallel relational databases for diameter calculation of large graphs’, 2016, http://worldcomp-proceedings.com/proc/p2016/PDP3821.pdf

结论

- **图数据分析的应用越来越广泛**

  → 社交网络

  → 电路设计

  → 城市规划

- **图数据分析的新需求**

  → 新架构

  → 新技术

  → 新算法

- **Greenplum + MADLib 图数据分析平台**

  → 充分利用MPP架构的水平扩展能力

  → In-database Analysis避免数据移动

  →使用SQL为接口

GREENPLUM DATABASE®

扫描加入 Greenplum 技术讨论群

微信群组

QQ 群组

欢迎订阅 Greenplum中文社区：greenplum.cn

Pivotal®

Transforming How The World Builds Software

© Copyright 2019 Pivotal Software, Inc. All rights Reserved.