

**Disciplina:** Visualização de Dados

**Professora:** Raquel C. de Melo-Minardi

## Primeiro Trabalho Prático

**Valor:** 20 pontos

### 1 Objetivo

Introduzir o aluno ao desenvolvimento de visualizações usando a biblioteca d3js <sup>1</sup>.

### 2 Motivação

No processo de análise visual de dados, há algumas tarefas analíticas [1] comumente executadas pelo usuário tais como identificar, localizar, distinguir, categorizar, agrupar, distribuir, ordenar, comparar, associar, correlacionar, entre outras. O projeto de visualizações de dados deve, então, levar em conta essas necessidades básicas e facilitar essas tarefas analíticas. Nesse trabalho prático, vocês devem implementar um algoritmo que facilite ao usuário / leitor a identificação e associação de objetos similares visualmente, ou seja, através de atributos visuais pré-atentivos.

### 3 Dados

Neste projeto, desejamos analisar visualmente um conjunto de grafos bipartidos e identificar semelhanças / padrões entre eles. Um grafo bipartido ou bigrafo é um grafo cujos vértices podem ser divididos em dois conjuntos disjuntos  $U$  e  $V$  tais que toda aresta conecta um vértice em  $U$  a um vértice em  $V$ , ou seja,  $U$  e  $V$  são conjuntos independentes.

O objetivo do analista de dados, no contexto deste trabalho prático, é identificar os nós similares em cada grafo distinguindo os nós que possuem um nó semelhante em todos os grafos dos que não possuem.

No arquivo *all.txt* você encontrará, em cada linha, ou um cabeçalho de grafo ou uma aresta do respectivo grafo. No exemplo abaixo, você pode ver a linha "graph 0" indicando o início do grafo de identificador 0 e as linhas seguintes são arestas do grafo 0. Após a primeira coluna com a palavra chave "edge", você encontrará o identificador único do nó do tipo 1 (pessoa). As duas colunas seguintes são rótulos dos nós de tipo 2 e indicam o tipo e o identificador do mesmo. Imagine que o nó de tipo 1 represente uma pessoa e o nó do tipo 2, o lugar visitado por essa pessoa em um determinado período de tempo. Esse lugar tem um tipo e um identificador. Assim, a linha 2 do exemplo abaixo representa que a pessoa 1 visitou um lugar do tipo S e identificador 92.

```
graph 0
edge 1 S 92
```

---

<sup>1</sup><http://www.d3js.org>

edge 1 A 12  
 edge 1 Y 99  
 edge 1 W 95  
 edge 1 D 24  
 edge 1 I 65  
 edge 1 S 92  
 edge 1 Y 96  
 edge 1 Y 97  
 edge 1 Q 90  
 edge 1 F 46  
 edge 1 I 7  
 edge 1 L 33  
 edge 1 Q 90  
 edge 1 A 12  
 edge 1 M 78  
 edge 1 E 45  
 edge 1 M 74  
 edge 1 Y 97  
 edge 1 P 89  
 edge 1 G 52  
 edge 1 P 89  
 edge 1 D 26  
 edge 2 R 80  
 edge 2 P 89  
 edge 2 V 94  
 edge 2 P 89

...

Desejamos tornar mais fácil que o usuário visualize o conjunto de grafos (suponha que cada grafo represente uma comunidade de pessoas) identificando pessoas que são semelhantes de acordo com os tipos de lugares frequentados. Sugerimos que se use como métrica de similaridade a distância euclidiana entre vetores com as frequências de cada tipo de lugar visitado por cada pessoa. Por exemplo, a pessoa 1 teria o seguinte vetor ( $A = 2, B = 0, C = 0, D = 2, E = 1, F = 1, G = 1, H = 0, I = 2, J = 0, K = 0, L = 1, M = 2, N = 0, O = 0, P = 2, Q = 2, R = 0, S = 2, T = 0, U = 0, V = 0, W = 1, X = 0, Y = 4, Z = 0$ ) ou normalizando (dividindo pelo número de visitas que seria 23) ficaria ( $A = 0.08, B = 0, C = 0, D = 0.08, E = 0.04, F = 0.04, G = 0.04, H = 0, I = 0.08, J = 0, K = 0, L = 0.04, M = 0.08, N = 0, O = 0, P = 0.08, Q = 0.08, R = 0, S = 0.08, T = 0, U = 0, V = 0, W = 0.04, X = 0, Y = 0.17, Z = 0$ ).

Dessa forma, você deve desenvolver um algoritmo que (1) compare todo o conjunto de grafos (2) identifique os nós (pessoas) que são conservados / similares em todos os grafos (3) compute uma localização para cada objeto representativo de uma pessoa na tela. Uma sugestão é implementar cada pessoa como um gráfico de rosca no qual as proporções de visitas a cada tipo de lugar (coluna 3) sejam as partes do todo. Cada gráfico (representativo de uma pessoa) deve ficar posicionado de acordo com a conservação ou não no conjunto de grafos. Um grafo não conservado deve ser desenhado nas margens da tela em tamanho menor enquanto os conservados devem ocupar posições mais centrais (ver exemplo.pdf).

Caso duas pessoas visitem o mesmo lugar (olhar o id do lugar na coluna 4), uma aresta rotulada com o tipo e identificador do nó deve ser traçada entre os respectivos objetos visuais.

A visualização deve ser interativa de forma que dados sobre o número de visitas e o percentual devem ser exibidos quando o mouse for posicionado sobre os objetos visuais (*detalhes sob*

*demanda*).

## 4 Tarefas

1. Implementar a visualização dos gráficos sugeridos em d3js.
2. Implementar a técnica de interação *detalhes sob demanda*.
3. Analisar as limitações desta representação. Caso alguma dessas limitações dificulte ou impossibilite a análise do conjunto de dados, tomar as decisões necessárias e documentá-las.
4. Analisar os dados e escrever um pequeno texto explicando as descobertas (padrões, tendências, exceções, etc) feitas com o uso da visualização.
5. Documentar o projeto em um texto de, no máximo, 3 páginas.

## 5 Entrega

1. Códigos fonte e *link* para página *web* contendo o trabalho.
2. Documentação

Unicamente pelo *Moodle* até o dia 12 de abril de 2015.

## 6 Pontos extras

Valem pontos extras:

1. O desenvolvimento de técnicas adicionais de interação como por exemplo filtragem, seleção, destaque, *zoom*, etc. (até 20% do valor do trabalho).
2. O acréscimo de algoritmos diferenciados para identificação de similaridades entre os objetos. (até 10% do valor do trabalho)
3. A proposta e implementação de representações visuais alternativas para os dados. (até 30% do valor do trabalho)

## Referências

- [1] Emile Morse, Michael Lewis, and Kai A Olsen. Evaluating visualizations: using a taxonomic guide. *International Journal of Human-Computer Studies*, 53(5):637–662, 2000.