
Aluno:	Matrícula:	Valor: 25,0	Nota:
--------	------------	-------------	-------

Trabalho Prático II

1. Objetivos.

- Aplicar os conhecimentos em grafos para modelar problemas reais.
- Desenvolver competências em análise e visualização de dados.
- Ampliar o pensamento crítico e tomada de decisão baseada em dados.
- Aprimorar habilidades como engenheiro de software.

2. Descrição.

Esse trabalho é uma continuação do Trabalho Prático I da disciplina. Enquanto na primeira etapa o objetivo era ler os dados e criar um grafo que representa as relações de voto em comum entre deputados, nessa segunda etapa, o objetivo é analisar essas ligações e obter diversos *insights* com relação às conexões entre deputados e partidos políticos. Dentre as relações que poderemos analisar com base nesses dados, temos:

- Quão unido é um partido com relação a suas votações na câmara?
- Quanto um partido diverge de outro(s) com relação às votações?
- Há algum deputado cujo comportamento é muito distinto do resto de seu partido?
- Quais deputados pode ser elos de obtenção de apoio político entre diferentes partidos?
- Um partido se tornou mais (ou menos) unido em suas votações ao longo do tempo?
- Partidos se aproximaram ou distanciaram ao longo do tempo?

Para obter essas conclusões as seguintes visualizações devem ser aplicadas ao grafo de votações:

Centralidade Métrica que determina o quão central é um nó na rede. Será adotada no trabalho a métrica *betwenness* de centralidade. Nessa técnica, primeiramente, é calculado o caminho mínimo de todos os nós para todos os nós. A centralidade de um nó é dada pelo número de vezes que o nó aparece nesses caminhos mínimos dividido pelo número total de caminhos mínimos. Assim, um nó de centralidade alta é aquele que é usado com frequência para alcançar outros nós com custo mínimo na rede.

Heatmap Apresentação da correlação entre dados como um mapa de calor. Cada par de dados é uma célula, sendo sua cor mais clara (quente) quando a correlação é alta e escura (fria) quando a correlação é baixa. Essa visualização de dados ajuda a identificar variáveis (no nosso contexto deputados) altamente correlacionados ou de baixa correlação.

Plotagem do grafo Através da visualização das ligações do grafo de votações, podemos obter importantes *insights* sobre as conexões, como a coesão dos partidos políticos, nós mais alinhados a pautas de outro(s) partidos que não o seu, pote.

3. Criação e processamento do grafo

O seu programa deve considerar como entrada os seguintes arquivos para cada ano de análise:

politiciansYYYY.csv Que apresenta o nome, partido e número de votações que o político participou no ano YYYY;

graphYYYY.csv Que apresenta o número de vezes que o par de deputados votou igual em votações ocorridas no ano YYYY. Pares de deputados que não votaram igual em nenhuma votação são omitidos.

esse par de arquivos será disponibilizado para cada ano, desde 2001 até 2023. É opcional adaptar a implementação do Trabalho 1 para esse formato ou ler os arquivos que disponibilizei. Note que, apesar de ter consituído um exercício interessante o uso da API para acessar os dados, é muito lento para os processamentos demandados nessa etapa.

O foco do presente trabalho não é a implementação dos algoritmos de centralidade, mas as transformações necessárias à rede para obter um bom resultado. Além disso, a utilização de bibliotecas *open source* em projetos de software trás diversos benefícios, como redução de bugs e ganho de velocidade de desenvolvimento. Dessa forma, recomenda-se o uso da biblioteca NetworkX¹ para executar esses algoritmos. Seu programa deve contemplar as seguintes transformações no grafo original de votações:

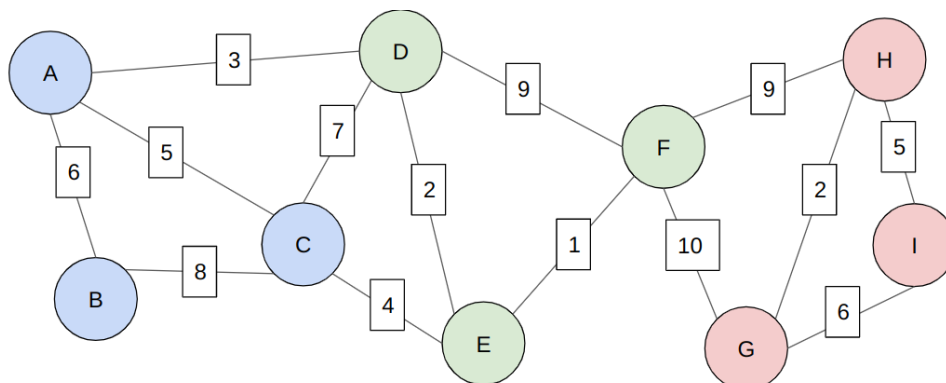
(a) **Filtros:**

Seu programa deve permitir que o usuário aplique dois filtros nos dados a serem analisados:

Por ano De modo a analisar apenas os dados de determinado ano informado pelo usuário, por exemplo "2022".

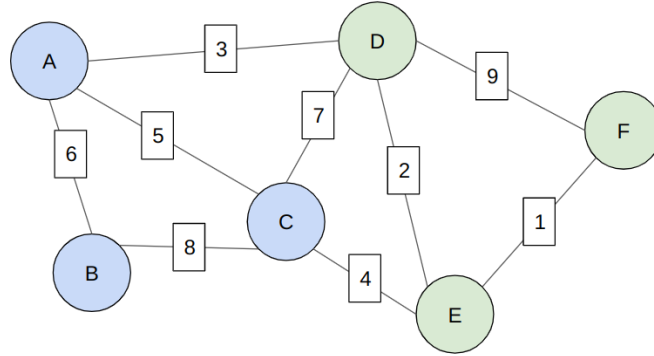
Por partidos Para analisar apenas os dados de um conjunto de partidos a ser informado pelo usuário, por exemplo "PT", "PL" e "MDB". Caso o usuário não queira filtrar por partido, todos partidos devem ser considerados.

Reproduzo aqui um grafo ilustrativo de votações conforme metodologia adotada no Trabalho 1 para exemplificar as transformações necessárias à rede. Suponha que os nós verdes representem deputados do MDB, nós azuis, deputados de PL e nós vermelhos, o PT:



¹<https://networkx.org/documentation/stable/index.html>

Aplicando o filtro dos partidos MDB e PL, os nós representando deputados G, H e I e suas respectivas ligações são removidos:

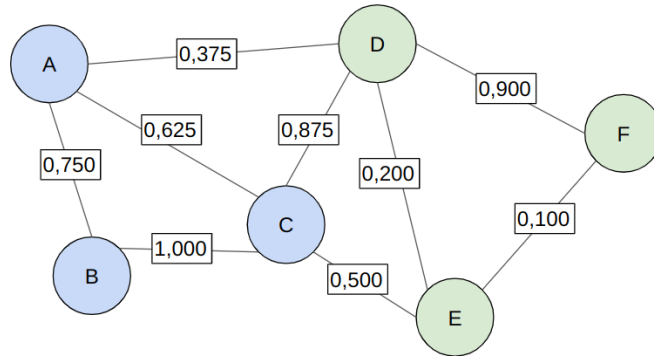


(b) **Normalização**

Após os filtros, é requerida a normalização do peso das arestas no intervalo $[0, 1]$, sendo que 1 indica que o deputado A concordou com o deputado B em 100% das votações em que participou e 0 indica que o deputado A não concordou com o deputado B nenhuma das votações (nesse caso específico a aresta nem existirá). A normalização segue a seguinte equação:

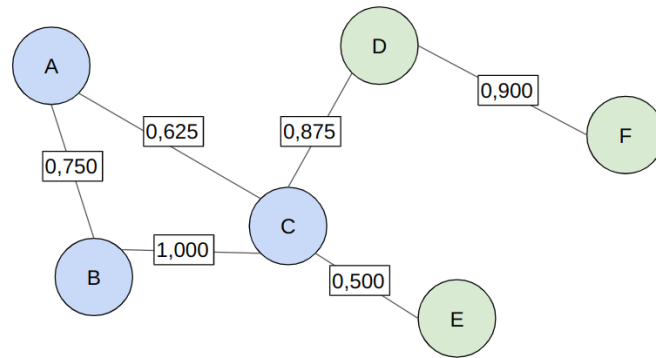
$$w(u, v) = \frac{w(u, v)}{\min(\text{votes}(u), \text{votes}(v))} \quad \forall u \in E, \forall v \in E \quad (1)$$

onde $\text{votes}(u)$ representa o número de votações que o deputado u participou. Consideremos que cada nós azul participou de 8 votações e os verdes de 10 votações:



(c) **Threshold**

Ligações pouco significativas podem dificultar a obtenção de informação útil nesse grafo. Por exemplo, considere que houveram 100 votações no intervalo de tempo em análise. É muito provável que qualquer par de deputados concorde em ao menos uma votação dessas 100, implicando num peso 0,01. Entretanto, essa ligação é fraca demais para ser considerada relevante. Nesse sentido, num segundo momento, em sua rede, as arestas cujo peso é menor que um *threshold* informado como parâmetro devem ser removidas. Considerando o grafo anterior com um *threshold* mínimo de 0,5 (50% de concordância), temos:

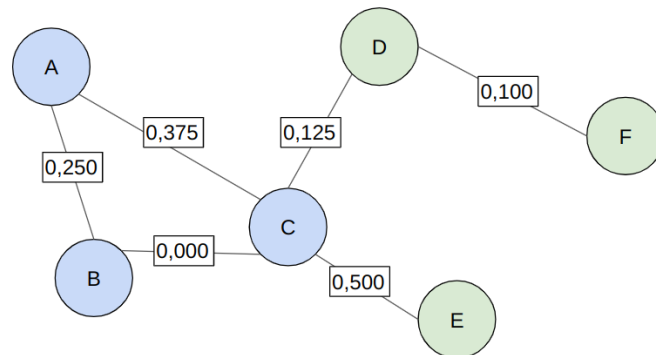


(d) **Inversão de pesos**

Com relação ao cálculo de centralidade, uma transformação adicional é necessária. Como arestas de menor peso são preferíveis em um caminho mínimo, deve-se ainda inverter os pesos das arestas, conforme equação

$$w(u, v) = 1 - w(u, v) \quad \forall u \in E, \forall v \in E \quad (2)$$

Por exemplo, caso a aresta tenha peso 0,900, indicando concordância em 90% dos votos, o peso invertido será de 0,100. Veja a rede de exemplo atualizada:



4. Análise e visualização dos dados

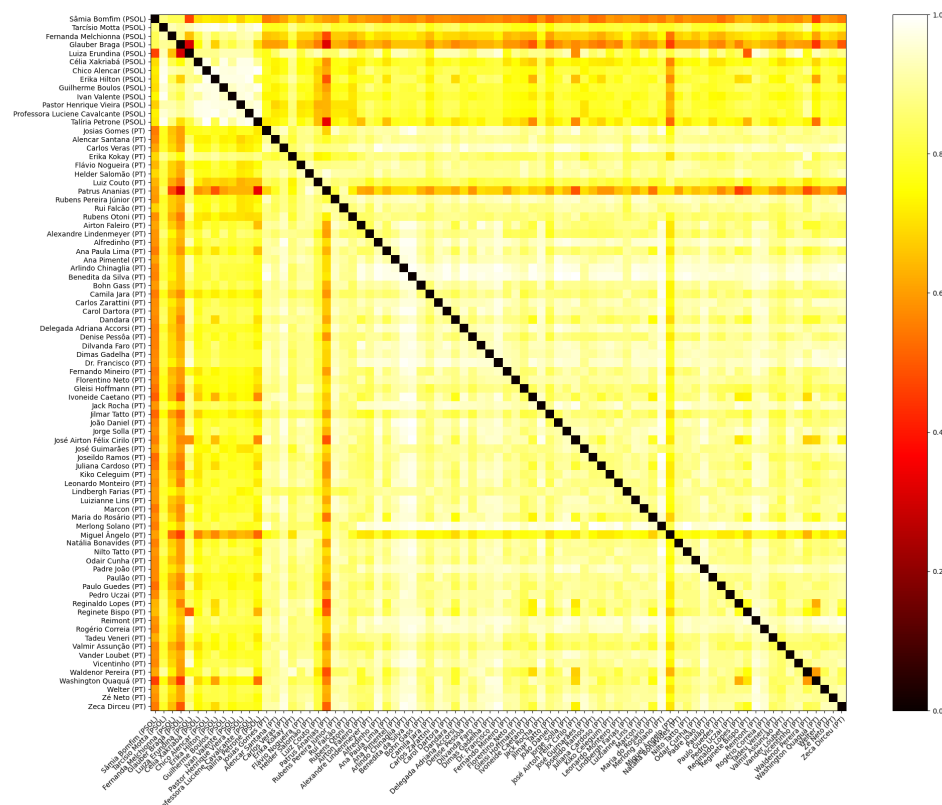
Com o grafo apropriadamente ajustado, aplique a métrica de *betweenness*² da biblioteca **NetworkX**. Essa função retorna um dicionário (*hash map*) que associa cada nó a sua métrica de centralidade. Faz parte do trabalho o estudo e entendimento da biblioteca **NetworkX**, bem como entradas e saídas esperadas.

(a) **Centralidade**

Com auxílio da biblioteca **matplotlib**³ obtenha e salve o gráfico contendo os deputados no eixo X e a medida de centralidade no eixo Y, como no exemplo a seguir, que considera os partidos PT e PSOL no ano de 2023 com *threshold* de 90%..

²<https://networkx.org/documentation/stable/reference/algorithms/centrality.html#shortest-path-betweenness>

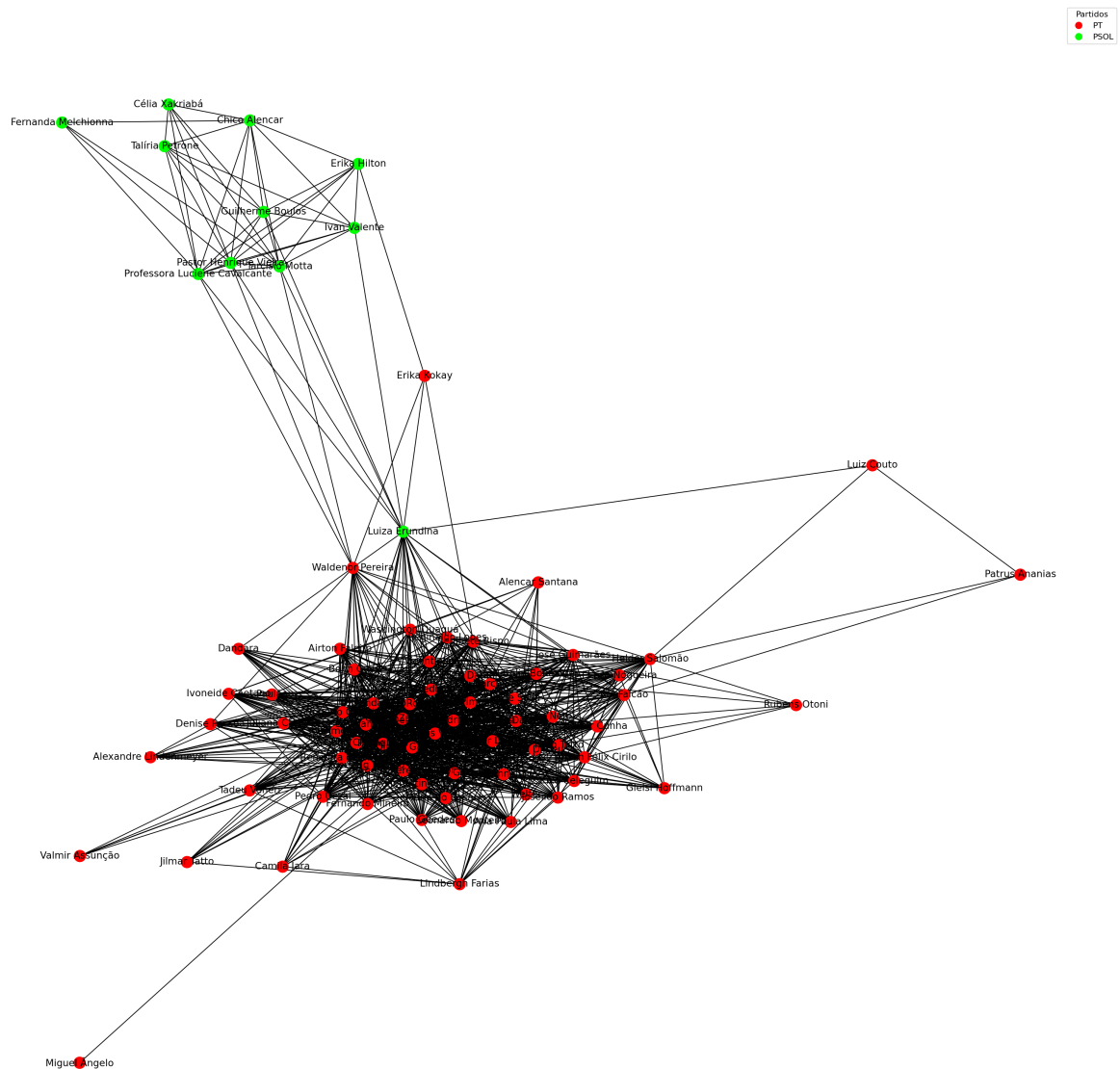
³<https://matplotlib.org/stable/gallery/index.html>



Como os partidos em questão têm pensamentos ideológicos próximos, o *heatmap* apresenta grande predominância de valores claros (próximos a 1). Algumas poucas exceções podem ser observadas, como Sãmia Bonfim (PSOL) com muitos valores próximos a 0,5 e, em menor escala, Luiza Erudina (PSOL) e Patrus Ananias (PT).

(c) Grafo

Por fim, requer-se o plot do grafo para visualização das relações de votos entre os deputados, onde podemos idenficar blocos altamente coesos (usualmente no mesmo partido), políticos deslocados da pauta de seu partido (de acordo com a metodologia conduzida), políticos isolados, potenciais elos de ligação entre grupos ideológicos, etc. Com auxílio do método `draw_spring` (e alguns ajustes), considerando os partidos PT e PSOL em 2023 com *threshold* de 90% é possível obter visualizações como:



Nesse grafo podemos identificar, em vermelho os deputados do PT e em verde os deputados do PSOL. Observe que Luiza Erudina é um ponto central nessa rede, compartilhando muitas ligações tanto com parlamentares do PT quando do PSOL. Nesse plot podemos ver alguns deputados isolados (i.e. que compartilham mais de 90% de concordância com poucos colegas), como Miguel Ângelo, Luiz Otávio e Patrus Ananias.

Note que as análises dadas como exemplo envolvem partidos que defendem pautas e pensamentos ideológicos próximos. Como último requisito do trabalho, pede-se que vocês conduzam suas próprias análises e escrevam um breve relatório, reportando os gráficos e as conclusões que podem ser obtidas. Sugestões de análises que podem ser interessantes: (i) comparar partidos de extrema direita, direita, centro, esquerda e extrema esquerda num mesmo plot; (ii) comparar a proximidade de dois partidos ao longo do tempo (ex. PT e MDB em 2015, 2019 e 2023); (iii) comparar a coesão em um mesmo partido (ex. comparar *heatmap* do PL com *heatmap* do PSOL), (iv) comparar todos os partidos em

2015, 2019 e 2023 e identificar aproximações ou distanciamentos, etc.

5. Interação com o usuário

A interação com o usuário deve ocorrer no arquivo `main` do seu programa (ou através de interface gráfica caso almeje os pontos extra da seção 7). O mesmo deve solicitar ao usuário os anos de votações a analisar, depois os partidos considerados e, após a execução, informar o nome dos arquivos de saída escritos contendo os gráficos. Segue um exemplo de interação com o programa:

```
Informe o ano a considerar (de 2001 a 2023):
<2023>

Informe o percentual mínimo de concordância (threshold) (ex. 0.9):
<0.9>

Informe os partidos a analisar, separados por espaço (ex. PT MDB PL):
<PL PSDB PV>

Processando...

Os plots foram salvos nos arquivos:
- betwenness 2023 PL PSDB PV.png
- heatmap 2023 PL PSDB PV.png
- graph 2023 PL PSDB PV.png
```

6. Avaliação.

O trabalho será avaliado com relação à corretude no processamento dos dados e filtros; à qualidade dos gráficos e visualizações entregues; e à qualidade do relatório gerado. O trabalho deverá ser feito individualmente ou em dupla e enviado via Moodle até as 23:59h do dia 25/08/23. Caso se tenha alguma dúvida com relação à autoria do trabalho o professor poderá solicitar uma apresentação presencial ao aluno (ou dupla).

7. Pontos extra.

A interação com o usuário através do prompt de comando é pouco amigável a usuários com conhecimento básico e intermediário em informática. Ademais, a criação de interfaces gráficas funcionais, atrativas e intuitivas é uma importante habilidade altamente demandada no mercado de trabalho. Serão concedidos até 5 pontos extra para os trabalhos que criarem uma interface gráfica para entrada dos filtros e visualização dos resultados (centralidade, *heatmap* e plot do grafo).

O uso de qualquer linguagem de programação e biblioteca pode ser adotado para criação das interfaces. A integração com o código responsável pela geração dos gráficos faz parte da tarefa extra.

Bom trabalho!