

Tipos De Grafos

Héctor Hugo García López

Introducción

Un grafo esta formado por un conjunto de nodos que llamaremos N , y un conjunto de la forma (u, v) llamado aristas al que nos referiremos por A . Dada su composición es conveniente ver un grafo como un dibujo. Los grafos se pueden usar para ilustrar procesos, rutas, cambios de estados en ciertos procesos y hasta un árbol genealógico. En este documento se mostrará los tipos grafos, cuales son las condiciones que tienen que cumplir y un ejemplo de su aplicación.

Para hacer una mejor referencia a los grafos, los vamos a dibujar en python y para eso necesitamos hacer uso de dos librerias, como se presenta a continuación.

```
1 import networkx as nx
2 import matplotlib.pyplot as plt
```

PrimerProyecto.py

1. Grafo simple no dirigido acíclico

Primeramente se definirá ciertos conceptos. Un camino es un subconjunto de A (que puede ser A) y recorre un subconjunto de N (que puede incluir todo N).

Un ciclo es un camino que une por lo menos 3 nodos distintos, y el nodo de inicio es el mismo que el nodo final.

En la figura 1 se representa un isómero de carbono, en el cual los nodos son los diferentes elementos del compuesto, y las aristas son los enlaces que tienen un elemento con otro.

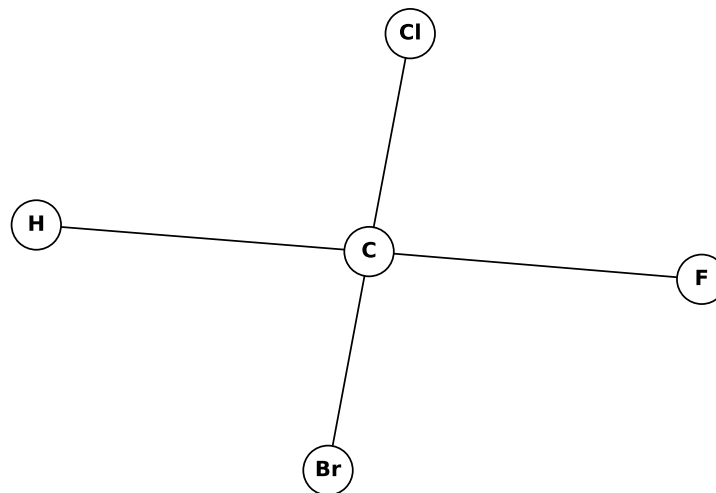


Figura 1: Grafo no dirigido acíclico

El código que da paso a la figura antes mencionada es el siguiente.

```
1 #Primero simple no dirigido aciclico
2 H=nx.Graph()
3
4 H.add_edges_from([( 'C', 'Cl' ), ( 'C', 'H' ), ( 'C', 'F' ), ( 'C', 'Br' )])
5
6 nx.draw(H, node_color="white", node_size=800, with_labels=True,
7         font_weight="bold", edgecolors="black")
8
9 plt.savefig('Primero.eps', format='eps', dpi=1000)
```

PrimerProyecto.py

La última línea del código se usa para guardar una imagen tipo extensión eps.

2. Grafo simple no dirigido cíclico

En la sección anterior se vió que es un ciclo en un grafo, en la figura 2 se observa un grafo que representa la caja de cambios de un carro, donde el nodo 0 es estar en neutral.

Es no dirigido porque es lo mismo si pasa de la primera marcha a la segunda, análogamente con las demás marchas; es cíclica ya que eso se repite muchas veces durante un mismo trayecto.

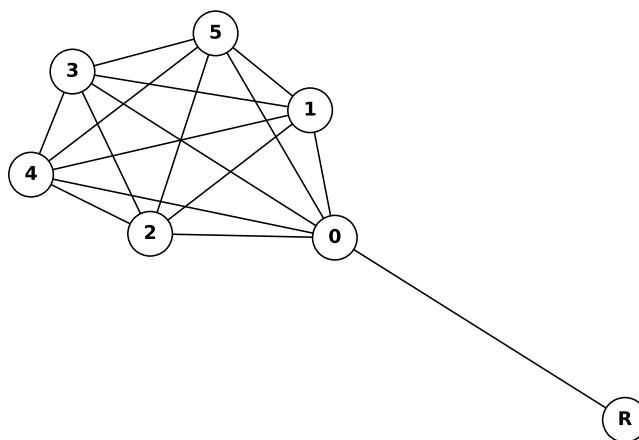


Figura 2: Grafo no dirigido cíclico

3. Grafo simple no dirigido reflexivo

Un nodo reflexivo es el que tiene un arista que va hacia sí mismo, es decir, para cualquier $u \in V$ existe $(u, u) \in A$. Entonces se puede decir que un grafo reflexivo es el que tiene nodos reflexivos. En este documento los nodos reflexivos son de color rojo.

En el siguiente ejemplo 3 se tiene el horario de cierto profesor, los nodos muestran

las materias y las aristas las horas que da de clase en una semana específica. Se puede apreciar que las materias de Cálculo 1, Álgebra y Conjuntos hay repeticiones (porque están de color rojo) lo que quiere decir que después de ir, por ejemplo, a Álgebra, la siguiente clase podría ser Álgebra, o salir de esa clase e ir a otra distinta.

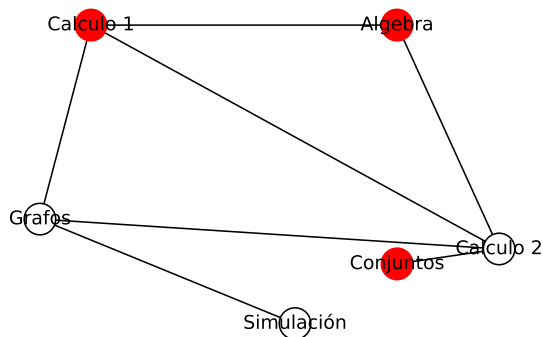


Figura 3: Grafo no dirigido reflexivo

Para llevar a cabo este grafo se hizo dos conjuntos llamados `nodos1` y `nodos2`, en las siguientes líneas (15) y (17) al dibujar `nodos1` y `nodos2` se le asignan los colores y las posiciones ya establecidas en el diccionario `pos`

```

1 #Tercero simple no dirigido reflexivo
2 G=nx.Graph()
3 G.add_edges_from([( 'Calculo 1', 'Algebra'), ('Calculo 1', 'Calculo 2')
4                   ,
5                   ( 'Calculo 1', 'Grafos'), ('Algebra', 'Calculo 2'),
6                   ( 'Calculo 2', 'Grafos'), ('Simulación', 'Grafos'),
7                   ( 'Calculo 2', 'Conjuntos')])
8
9 nodos1 = { 'Calculo 1', 'Algebra', 'Conjuntos' }
10 nodos2 = { 'Calculo 2', 'Simulación', 'Grafos' }
11
12 pos = { 'Calculo 1':(20, 35), 'Algebra':(50,35),
13         'Calculo 2':(60, 20), 'Simulación':(40,15),
14         'Grafos':(15,22), 'Conjuntos':(50,19)}
15
16 nx.draw_networkx_nodes(G, pos, nodelist=nodos1, node_size=400,
17                        node_color='r', node_shape='o')
18 nx.draw_networkx_nodes(G, pos, nodelist=nodos2, node_size=400,
19                        node_color='w', node_shape='o',
20                        edgecolors="k")

```

```

21 nx.draw_networkx_edges(G, pos)
22 nx.draw_networkx_labels(G, pos)
23 plt.axis('off')
24
25 plt.savefig('Tercero.eps', format='eps', dpi=1000)

```

PrimerProyecto.py

4. Grafo simple dirigido acíclico

Un grafo dirigido es aquel que se especifica la dirección de la arista, es decir, si $(u, v) \in A$ entonces $(v, u) \notin A$.

Un ejemplo para la figura 4 puede ser un centro de distribución de materiales, que entrega a varios proveedores que simbolizan los nodos y cada arista es el flujo de material.

Para que este ejemplo sea considerado como un grafo simple solo se puede enviar material.

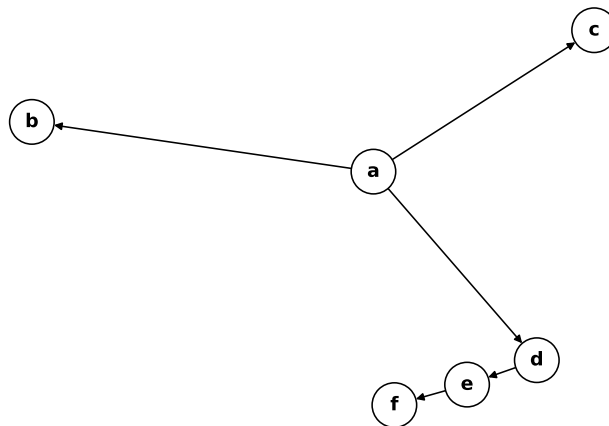


Figura 4: Grafo dirigido acíclico

Para realizar el grafo se hizo uso de la función `DiGraph()` y con ella las aristas se vuelven dirigidas

```
1 #Cuarto simple dirigido aciclico
2 H=nx.DiGraph()
3 H.add_edges_from([( 'a', 'b'), ( 'a', 'c'), ( 'a', 'd'), ( 'd', 'e'),
4                   ( 'e', 'f'), ( 'c', 'a')])
5 nx.draw(H, node_color="white", node_size=800, with_labels=True,
6         font_weight="bold", edgecolors="black")
7 plt.savefig('Cuarto.eps', format='eps', dpi=1000)
```

PrimerProyecto.py

5. Grafo simple dirigido cíclico

Al igual que en el grafo no dirigido, este grafo tiene la característica de que contiene un ciclo. Si queremos representar como diagrama de flujo, un videojuego que se elige entrar solo (solitario) o en grupo (cooperativo) y después de esa opción se va a jugar, y cuando termina el juego, se devuelve al menú principal.

En la figura 5 se aprecia el ciclo y las flechas que nos dice que es un grafo dirigido

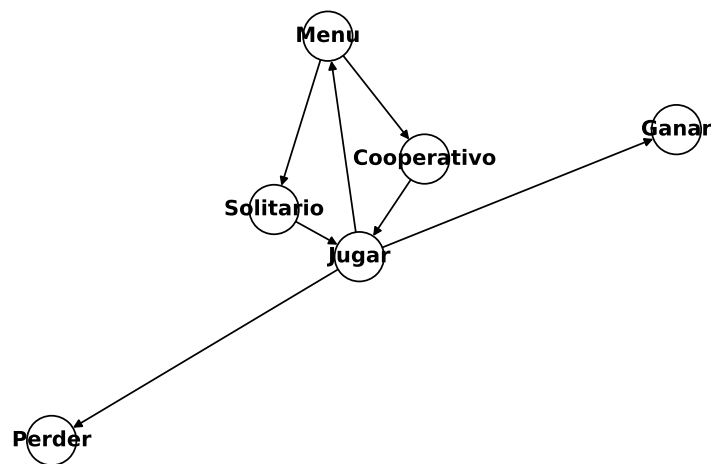


Figura 5: Grafo dirigido cíclico

6. Grafo simple dirigido reflexivo

Supongamos que se tiene el diagrama de flujo de los procesos para la elaboración de productos en una empresa. El primer paso es la recepción donde se verifica la calidad de los productos que entran y se cuentan los mismos con el fin de ver si lo que llegó es lo que se pidió. En caso de no cumplir con el conteo, se vuelve a verificar. Si está en mal estado se puede llevar a calidad e intentar recuperar el producto, pero si el producto no queda del todo bien, sigue entrando a calidad.

Este caso puede ser representado mediante un grafo, como lo muestra la figura 6 donde los nodos representan los procesos y las aristas es el flujo de los productos. Los nodos reflexivos son como el nodo 1 o calidad, que pueden entrar varias veces al mismo proceso.

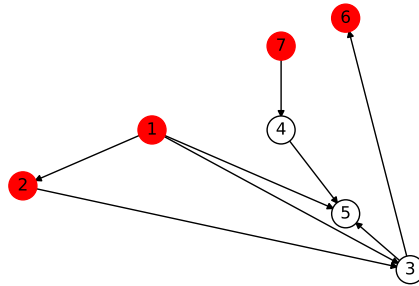


Figura 6: Grafo dirigido reflexivo

7. Multigrafo no dirigido acíclico

Un multigrafo es un grafo en el cual puede haber mas de un arista entre cada par de nodos, esto es, tiene multiplicidades en sus nodos. Gráficamente se representa con dos líneas, pero en los siguientes gráficos solo se muestra una línea.

Este tipo de grafos se pueden usar para representar los vuelos de cierto tipo de aerolínea entre algunos países, como lo representa en la figura 7 donde los nodos son los países y las aristas son la cantidad de vuelos.

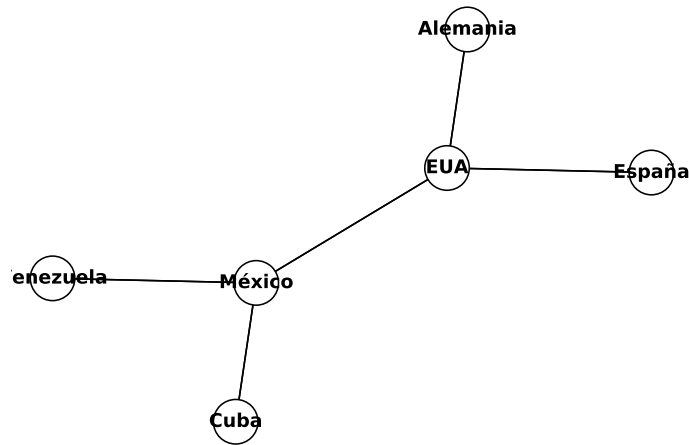


Figura 7: Multigrafo no dirigido acíclico

Para dibujar estos grafos se usa la función de networkx MultiGraph() que te permite agregar aristas que, aunque vistos como conjuntos son iguales, importan porque representan flujos (cantidades de vuelos en nuestro ejemplo) diferentes

```

1 #Septimo multigrafo no dirigido aciclico
2 H=nx.MultiGraph()
3 H.add_edges_from([( 'Mexico', 'EUA'), ( 'EUA', 'Mexico'),
4                   ( 'EUA', 'Espa a'), ( 'Espa a', 'EUA'),
5                   ( 'Mexico', 'Venezuela'), ( 'Venezuela', 'Mexico'),
6                   ( 'Cuba', 'Mexico'), ( 'Mexico', 'Cuba'),
7                   ( 'EUA', 'Alemania'), ( 'Alemania', 'EUA')])
8 nx.draw(H, node_color="white", node_size=800, with_labels=True,
9         font_weight="bold", edgecolors="black")
10 plt.savefig('Septimo.eps', format='eps', dpi=1000)

```

PrimerProyecto.py

8. Multigrafo no dirigido cíclico

Al igual que en el grafo simple, este grafo contiene por lo menos un ciclo. Para visualizarlo mejor se hace referencia al ejemplo anterior, en la figura 8 vemos que ahora, Venezuela y Cuba tienen vuelos entre sí, lo mismo que Venezuela y España, formándose dos ciclos

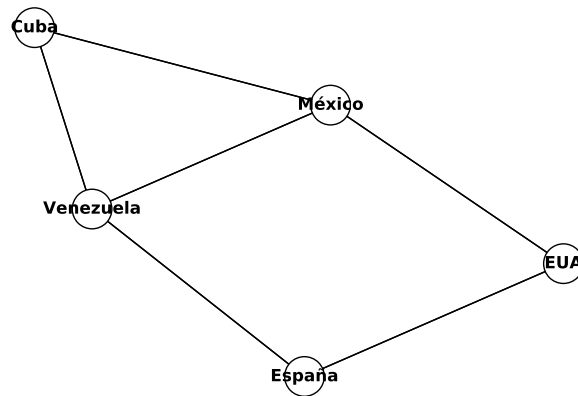


Figura 8: Multigrafo no dirigido cíclico

Lo diferente de este gráfico con el anterior consiste en que se agregan las aristas para conectar Cuba-Venezuela y España-Venezuela

9. Multigrafo no dirigido reflexivo

Como ya se ha comentado, los nodos reflexivos se pintan de rojo. El ejemplo que se tomó para este tipo de grafos es donde cada nodo representa una ciudad y un arco representa la capacidad del drenaje pluvial, como la capacidad es la misma en ambas direcciones es no dirigido, ya que hay drenaje que conecta con otros ductos dentro de la misma ciudad entonces el grafo es reflexivo.

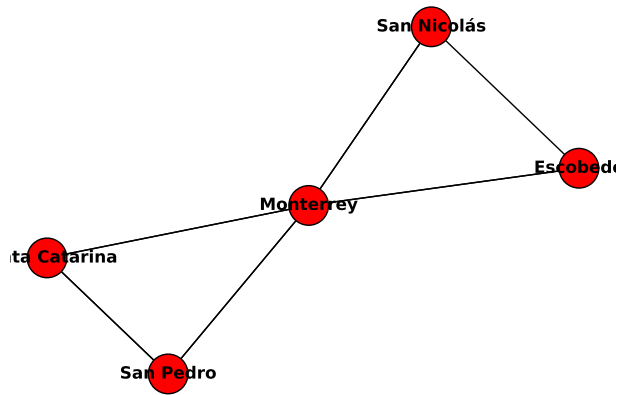


Figura 9: Multigrafo no dirigido reflexivo

10. Multigrafo dirigido acíclico

En la figura 10 se representa a 5 estaciones de trabajo (work station) las cuales están conectadas a un servidor. Los aristas representan el flujo de información que existe entre las estaciones de trabajo y el servidor, cada computadora es independiente, salvo por el servidor, por lo que toda conexión entre computadoras es a través del mismo.

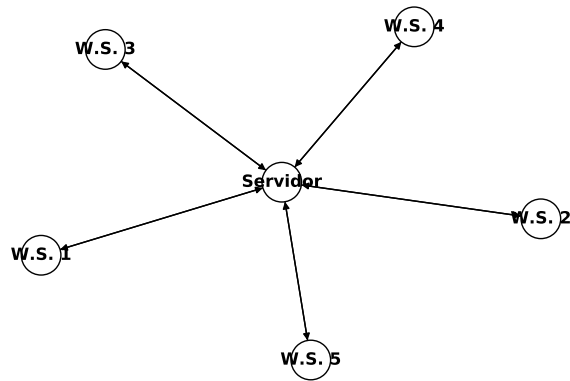


Figura 10: Multigrafo dirigido acíclico

El código necesario para la gráfica es el siguiente.

```

1 #Decimo multigrafo dirigido aciclico
2 H=nx.MultiDiGraph()
3 H.add_edges_from([( 'Jose ', 'Claudia '), ( 'Claudia ', 'Jose '),
4                   ( 'Claudia ', 'Agustin '), ( 'Agustin ', 'Claudia '),
5                   ( 'Agustin ', 'Priscila '), ( 'Agustin ', 'Felipe '),
6                   ( 'Agustin ', 'Cristina '), ( 'Cristina ', 'Carlos '),
7                   ( 'Carlos ', 'Pedro '), ( 'Pedro ', 'Carlos '),
8                   ( 'Priscila ', 'Carlos '), ( 'Carlos ', 'Priscila ')]])
9 nx.draw(H, node_size=800, with_labels=True,
10         font_weight="bold", edgecolors="black", node_color='white')
11 plt.savefig('Decimo.eps', format='eps', dpi=1000)

```

PrimerProyecto.py

Donde vemos que se usa la función MultiDiGraph(), al igual que en DiGraph() un par ordenado de vertices se convierte en una flecha.

11. Multigrafo dirigido cíclico

Para este ejemplo se tiene la transmisión de enfermedades en una oficina, donde los nodos son las personas y los aristas representan la probabilidad de transmitir la enfermedad, que puede ser distinta si es de Felipe a Agustín o de Agustín a Felipe, lo mismo ocurre con los demás nodos

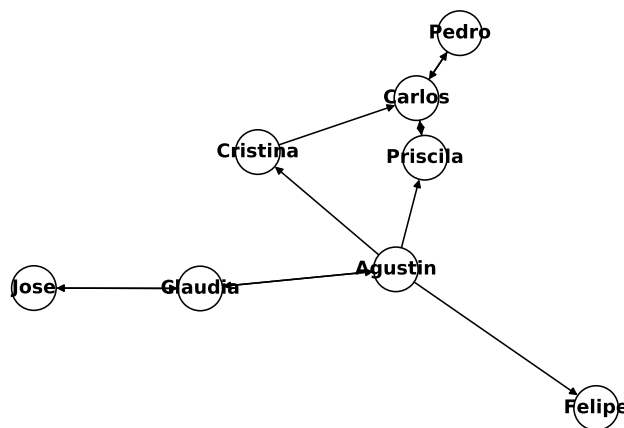


Figura 11: Multigrafo dirigido cíclico

12. Multigrafo dirigido reflexivo

Supóngase que se tiene un evento probabilístico, donde cada evento tiene una cierta probabilidad dado el evento anterior, y un evento se puede repetir. En la figura 12, se puede ver que los nodos son los eventos, ya que se puede repetir el evento entonces el grafo es reflexivo, y las aristas son las probabilidades de ocurrencia del evento j dado que ocurrió el evento i

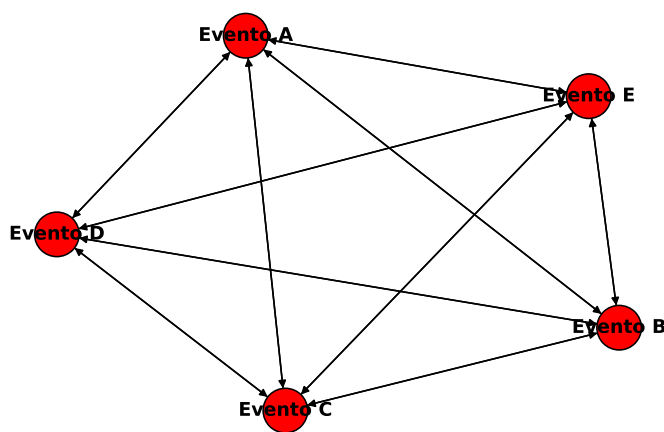


Figura 12: Multigrafo dirigido reflexivo

Referencias

- [1] A. C. Barrero, G. W. de García, and R. M. M. Parra, *Introducción a la Teoría de Grafos*. ELIZCOM SAS, 2010.