Evidencia 1. Redes bayesianas: caso discreto

AUTHORS
Mayra Sarahí de Luna Castillo A01635774
Juan Manuel Hernández Solano A00572208
Alejandra Velasco Zárate A01635453
José Antonio Juárez Pacheco A0057218
José Carlos Yamuni Contreras A01740285

PUBLISHED
August 16, 2023

Abstract

El sistema de transporte en México desempeña un papel crucial en la vida cotidiana y el desarrollo económico del país. Por ello es importante conocer la opinión de los usuarios para saber y trabajar en las áreas de mejora. En el proyecto "Redes Bayesianas: Caso discreto" se utilizó una base de datos que recopila información sobre los datos personales de los usuarios, los tiempos de viaje, el nivel de satisfacción y otros factores clave relacionados con la movilidad. Para este proyecto, primero se hizo un preprocesamiento de los datos para limpiar la base, se crearon dos Grafos Acíclicos Dirigidos (DAG) con las variables más importantes, se evaluó el rendimiento de cada una para seleccionar la que se ajustaba mejor a los datos y se entrenaron los datos con la DAG seleccionada. Gracias a lo anterior se pudo conocer las probabilidades de las "queries" (hipótesis) asignadas, lo que nos da un panorama amplio del uso y la eficiencia del transporte en México.

Introducción

El proyecto que se muestra a continuación tiene por objetivo conocer los medios de transporte más utilizados por la población mexicana y su nivel de satisfacción en temas de seguridad y eficiencia haciendo uso de la información obtenida por la Encuesta Nacional de Movilidad y Transporte por el Instituto de Investigaciones Jurídicas de la UNAM. Lo anterior se logrará a través de la implementación de redes bayesianas, las cuales proporcionan una representación visual para un conjunto de variables aleatorias y para las relaciones que hay entre ellas. La estructura de estas

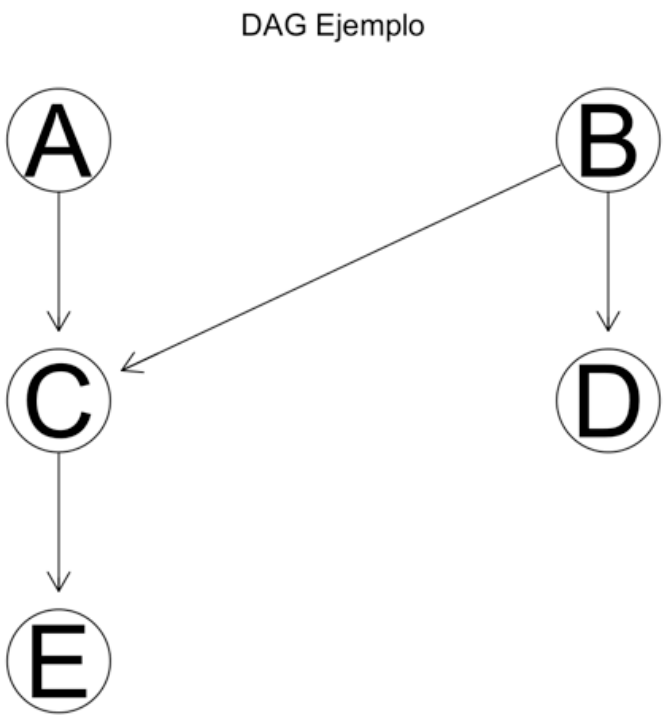
redes permiten especificar la función de probabilidad conjunta de las variables como el producto de funciones de probabilidad condicionada. La principal diferencia de estos modelos se encuentra en sus arcos ya que son dirigidos y representan dependencia condicional entre variables. El objetivo de este trabajo es utilizar las redes bayesianas para hacer inferencias sobre algunas hipótesis y obtener la probabilidad de que sean ciertas o no, con el fin de conocer las relaciones y dependencias que hay entre variables y/o eventos.

Marco Teórico

Las redes bayesianas, también conocidas como redes causales probabilísticas, son herramientas estadísticas que representan un conjunto de incertidumbres asociadas sobre la base de las relaciones de independencia condicional que se establecen entre ellas. Son grafos acíclicos dirigidos en el que cada nodo representa una variable aleatoria que tiene asociada una función de probabilidad condicional (Santiesteban, 2012).

El modelo probabilístico es descrito por un grafo acíclico dirigida (DAG), donde los vértices de la gráfica que representan las variables se denominan nodos. Estos nodos se representan como círculos que dentro contienen el nombre de la variable y las conexiones entre nodos se denominan arcos. Estos arcos tienen terminación de flecha, lo que indica la dependencia entre variables. El nodo donde se origina el arco se llama padre, mientras que el nodo donde termina el arco se llama hijos. Los nodos a los que se puede llegar desde otros nodos se llaman descendientes. Los nodos que conducen una ruta a un nodo específico se llaman ancestros. El punto principal de las Redes Bayesianas es permitir que se realice una inferencia probabilística.

```
Loading required namespace: Rgraphviz
```



En esta DAG los nodos padre son A y B, el nodo hijo es el E. C y E son descendientes de A y A y C son ancestros de E. En una red bayesiana no hay bucles ni ciclos, ya que ningún nodo puede ser su propio antepasado o descendiente.

Distribuciones de probabilidad conjunta

La probabilidad conjunta es la probabilidad de que una serie de eventos sucedan simultáneamente. La probabilidad conjunta de varias variables se puede calcular a partir del producto de probabilidades indivudales de los nodos.

$$P(X_1, \ldots, X_n) = \Pi_{i=1}^{n} P(X_i \mid parents(X_i))$$

En el ejemplo propuesto , la distribución conjunta de probabilidad es:

$$P(A, B, C, D, E) = P(A)P(B)P(C \mid A, B)P(D \mid B)P(E \mid C)$$

Si un nodo no tiene un padre, como el nodo A, su distribución de probabilidad se describe como incondicional. De lo contrario, la distribución de probabilidad local del nodo está condicionada a otros nodos (Wolf et al., 2019).

Teorema de Bayes

El Teorema de Bayes parte de una situación en la que es posible conocer las probabilidades de que ocurran una serie de sucesos A_i . Se tiene un evento B cuya ocurrencia proporciona información, ya que las probabilidades de que ocurra B son distintas si el suceso A_i sucede.

$$P(A \mid B) = \frac{P(B \mid A)P(A)}{P(B)}$$

Donde $P(A)$ es la probabilidad a priori, $P(B \mid A)$ es la probabilidad condicional, $P(B)$ es la probabilidad total y el resultado $P(A \mid B)$ la probabilidad a posteriori.

Esta es la teoría detrás de las redes bayesianas.

Inferencia

A partir de una red ya construida, y dados los valores concretos de algunas variables de una instancia, podrían tratar de estimarse los valores de otras variables de la misma instancia aplicando razonamiento probabilístico. El razonamiento probabilístico sobre las redes bayesianas consiste en propagar los efectos de las evidencias (variables conocidas) a través de la red para conocer las probabilidades a posteriori de las variables desconocidas. De esta manera se puede determinar un valor estimado para dichas variables en función de los valores de probabilidad obtenidos (Santiesteban, 2012).

Con la metodología se puede ver la creación y aplicación de la DAG con ciertos datos para responder unas preguntas.

Metodología

0. Preprocesamiento de datos

Importar Bibliotecas

In [1]:
```
# Ignorar warnings ----------------------------------------------------------

import warnings
warnings.filterwarnings("ignore")

# Lectura y manipulación de datos -------------------------------------------

import pandas as pd
import numpy as np

# Visualización de datos ----------------------------------------------------

import matplotlib.pyplot as plt
```

In [2]:
```
# Leer la base de datos, se agrega el encoding por el texto
raw_data = pd.read_csv('enmt_unam.csv', encoding='latin-1')
raw_data.head()
```

Out[2]:

	con1	D_R	edo	muni	loca	folio	ageb	hr_ini1	min_ini1	hr_ter1	...	Pondi2	Pondi_v	P
0	1		28	41	1	5	098-6	9	34	10	...	11032	5469	
1	2		26	18	1	1	203-A	13	40	14	...	63083	11709	
2	3		15	37	18	4	044-A	12	8	13	...	31357	13213	
3	4		15	109	3	5	078-A	11	33	12	...	61769	9125	
4	5		12	67	16	4	999-9	10	40	11	...	99437	65655	

5 rows × 698 columns

Seleccionamos las columnas de la variables que nos interesan para responder las preguntas seleccionadas

In [3]:
```
data = raw_data[['Tam_loc','h10_2','h11_1','h14_2n','ing_fam','p16','p17_1','
data.head()
```

Out[3]:

	Tam_loc	h10_2	h11_1	h14_2n	ing_fam	p16	p17_1	p17_4	h21_1
0	1	2	55	3	3	98	1	1	9
1	1	2	49	7	8	4	1	2	12
2	3	2	44	7	3	4	1	1	1
3	1	1	60	3	0	2	1	2	7
4	4	2	60	3	3	6	1	2	7

Renombramos el nombre de las columnas para entender que variables son.

In [4]:
```
data = data.rename(columns={'Tam_loc':'Residencia','h10_2':'Sexo', 'h11_1': '
                            'p16':'Transporte','p17_1':'Eficiencia','p17_4':'

data.head()
```

Out[4]:

	Residencia	Sexo	Edad	Educacion	ing_fam	Transporte	Eficiencia	Seguridad	Ocupación
0	1	2	55	3	3	98	1	1	9
1	1	2	49	7	8	4	1	2	12
2	3	2	44	7	3	4	1	1	1
3	1	1	60	3	0	2	1	2	7
4	4	2	60	3	3	6	1	2	7

In [5]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1191 entries, 0 to 1190
Data columns (total 9 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   Residencia   1191 non-null   int64
 1   Sexo         1191 non-null   int64
 2   Edad         1191 non-null   int64
 3   Educacion    1191 non-null   int64
 4   ing_fam      1191 non-null   int64
 5   Transporte   1191 non-null   int64
 6   Eficiencia   1191 non-null   int64
 7   Seguridad    1191 non-null   int64
 8   Ocupación    1191 non-null   int64
dtypes: int64(9)
memory usage: 83.9 KB
```

Columna 1: Tam_loc

In [6]:
```
data['Residencia'].unique()
```

Out[6]:
```
array([1, 3, 4, 2])
```

In [8]:
```
info_Tam_loc = {1: 'grande', 2: 'grande', 3: 'pequeño', 4: 'pequeño'}

info_Tam_loc.get(1)

Tam_loc_renamed = [info_Tam_loc.get(x) for x in data['Residencia'].values]

data['Residencia'] = Tam_loc_renamed

data['Residencia']
```

Out[8]:
```
0         grande
1         grande
2       pequeño
3         grande
4       pequeño
         ...
1186      grande
1187      grande
1188      grande
1189    pequeño
1190      grande
Name: Residencia, Length: 1191, dtype: object
```

Columna 2: Sexo

In [9]:
```
info_Sexo = {1: 'hombre', 2: 'mujer'}

info_Sexo_renamed = [info_Sexo.get(x) for x in data['Sexo'].values]

data['Sexo'] = info_Sexo_renamed

data['Sexo']
```

Out[9]:
```
0         mujer
1         mujer
2         mujer
3       hombre
4         mujer
         ...
1186      mujer
1187      mujer
1188      mujer
1189      mujer
1190      mujer
Name: Sexo, Length: 1191, dtype: object
```

Columna 3: Edad

In [10]:
```
data['Edad'].unique()
```

Out[10]:
```
array([55, 49, 44, 60, 29, 65, 56, 64, 38, 70, 51, 61, 27, 39, 40, 37, 30,
       31, 45, 62, 50, 36, 25, 42, 48, 32, 21, 53, 57, 80, 22, 46, 23, 43,
       26, 28, 63, 41, 67, 24, 35, 68, 54, 77, 58, 52, 78, 75, 34, 47, 59,
       33, 74, 66, 72, 73, 69, 18, 71, 76, 84, 20, 90, 79, 83, 87, 89, 82,
       81,  0, 19,  3, 97, 85])
```

In [11]:
```
#info_Edad = {1: 'hombre', 2: 'mujer'}

info_Edad_renamed = []

for x in data['Edad'].values:
    if x < 18:
        info_Edad_renamed.append('joven')

    elif x > 18 and x < 35:
        info_Edad_renamed.append('adulto_joven')

    else:
        info_Edad_renamed.append('adulto_mayor')

data['Edad'] = info_Edad_renamed

data['Edad']
```

Out[11]:
```
0        adulto_mayor
1        adulto_mayor
2        adulto_mayor
3        adulto_mayor
4        adulto_mayor
            ...
1186     adulto_mayor
1187     adulto_joven
1188     adulto_mayor
1189     adulto_mayor
1190     adulto_mayor
Name: Edad, Length: 1191, dtype: object
```

In [12]:
```
data
```

Out[12]:

	Residencia	Sexo	Edad	Educacion	ing_fam	Transporte	Eficiencia	Seguridad
0	grande	mujer	adulto_mayor	3	3	98	1	1
1	grande	mujer	adulto_mayor	7	8	4	1	2
2	pequeño	mujer	adulto_mayor	7	3	4	1	1
3	grande	hombre	adulto_mayor	3	0	2	1	2
4	pequeño	mujer	adulto_mayor	3	3	6	1	2
...	...	...	...	...	...	...	...	...
1186	grande	mujer	adulto_mayor	4	7	98	2	2
1187	grande	mujer	adulto_joven	7	8	9	2	2
1188	grande	mujer	adulto_mayor	4	8	4	2	2
1189	pequeño	mujer	adulto_mayor	3	5	9	2	1
1190	grande	mujer	adulto_mayor	4	8	2	2	1

1191 rows × 9 columns

Columna 4: Educación

In [13]:
```
#Se observan los valores únicos de educación
data.Educacion.unique()
```

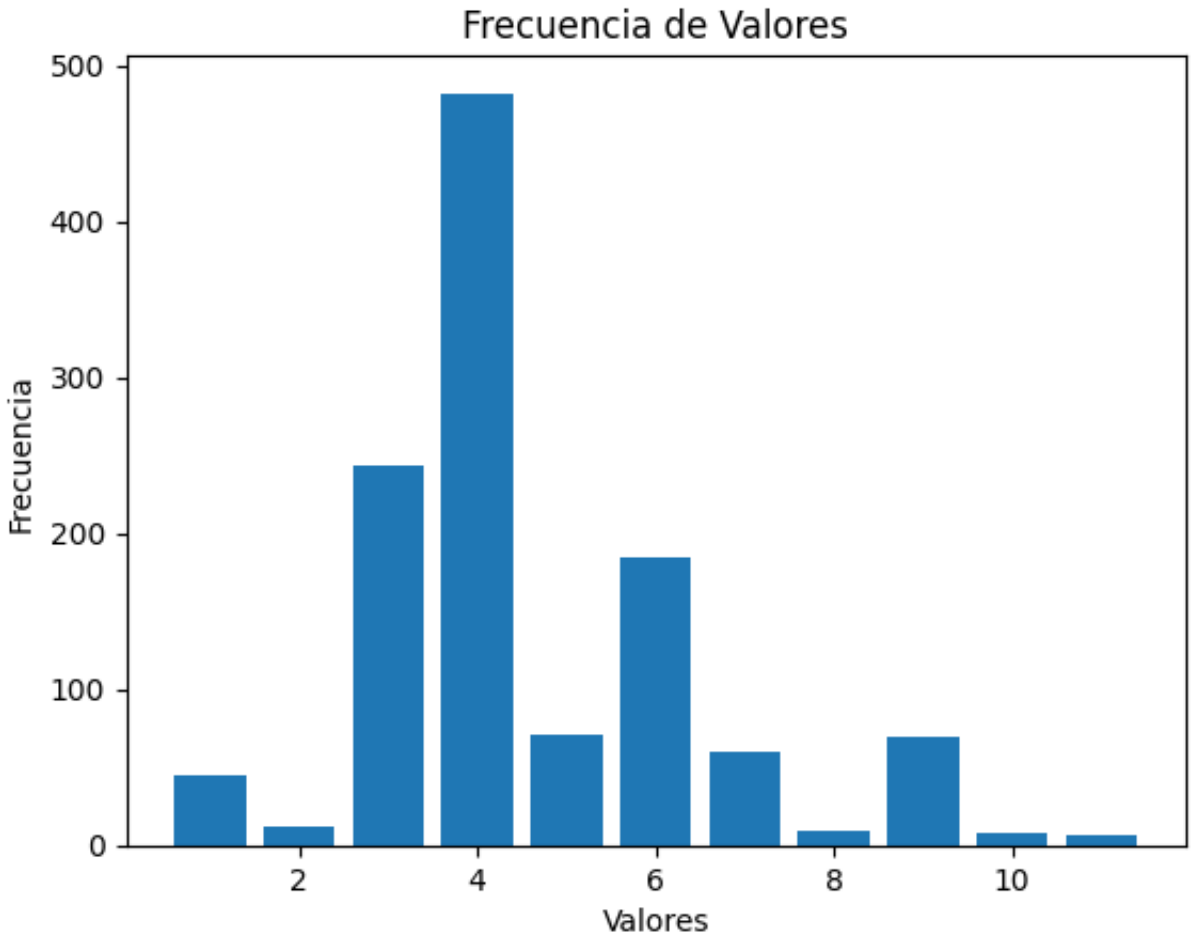
Out[13]:
```
array([ 3,  7,  6, -1,  4,  8,  9,  1, 99,  5,  2, 10, 11, 98])
```

In [14]:
```
#Se reemplazan los valores -1, 98, 99 por la moda de la columna, ya que estos
moda = data['Educacion'].mode()
valores_inadecuados = [-1, 98, 99]

for i, value in enumerate(data.Educacion):
    if value in valores_inadecuados:
        data.Educacion[i] = moda
```

In [15]:
```
#Se obtienen las frecuencias de los valores únicos para así tener una idea de
frecuencias = data['Educacion'].value_counts()

print(frecuencias)
```

```
4     482
3     243
6     185
5      71
9      70
7      60
1      45
2      12
8       9
10      8
11      6
Name: Educacion, dtype: int64
```

In [16]:
```
# Se puede visualizar de forma más clara con un gráfico de barras
plt.bar(frecuencias.index, frecuencias.values)
plt.xlabel('Valores')
plt.ylabel('Frecuencia')
plt.title('Frecuencia de Valores')
plt.show()
```



In [17]:
```
# Se agrupan los valores que representen un nivel de estudios de secundaria t
# y normal con preparatoria, y maestría o doctorado y carrera secretarial con
valores_raros1 = [10, 11]
valores_raros2 = [7, 8]

for i, value in enumerate(data.Educacion):
    if value in valores_raros1:
        data.Educacion[i] = 9
    elif value in valores_raros2:
        data.Educacion[i] = 6
    elif value == 5:
        data.Educacion[i] = 4
```

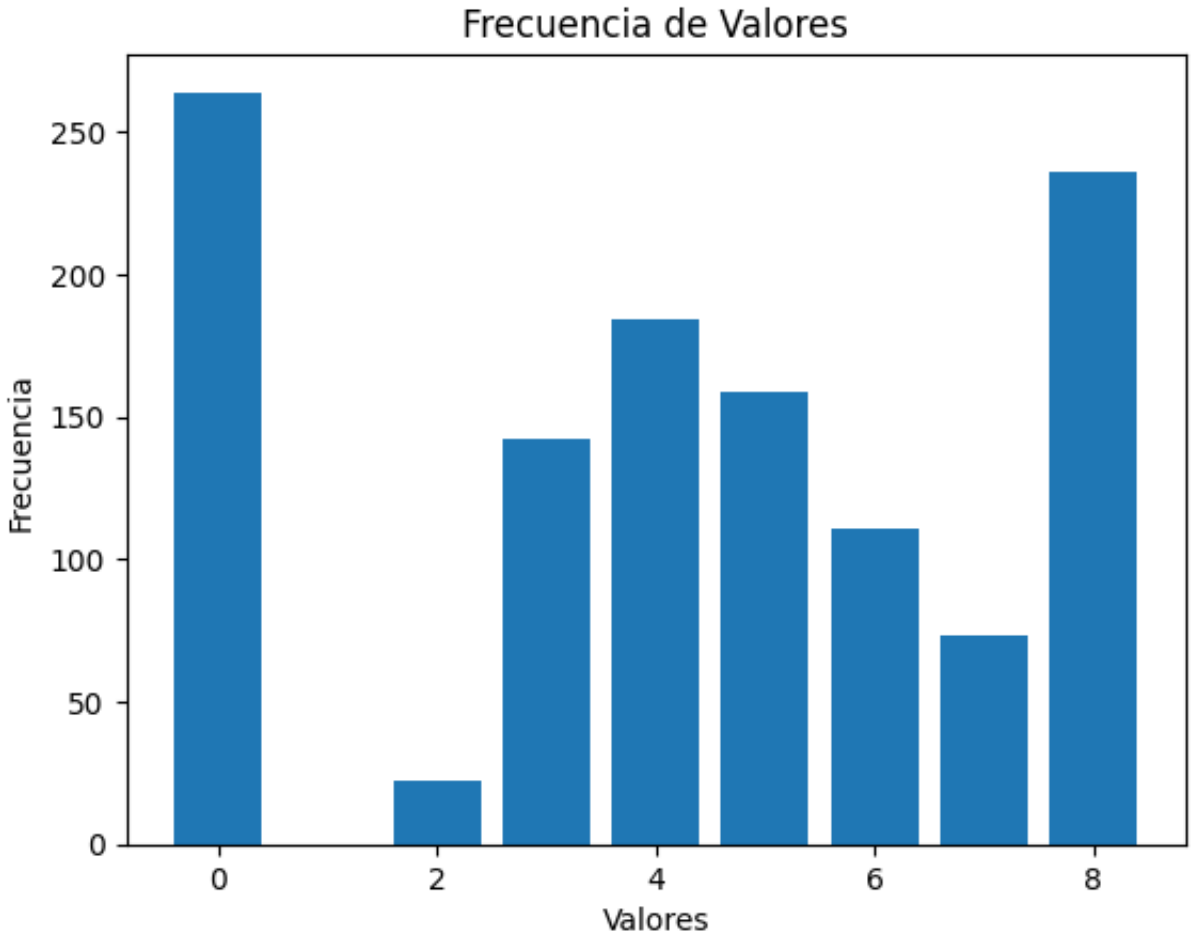
In [18]:
```
#Se renombram los seis niveles de educación
info_Educacion = {1:'ninguno', 2:'preescolar', 3:'primaria', 4:'secundaria',
n_Educacion = [info_Educacion.get(x) for x in data['Educacion'].values]
data['Educacion'] = n_Educacion
```

Columna 5: Ingreso Familiar

In [19]:
```
#Se revisan valores únicos de la columna de ingreso familiar
data.ing_fam.unique()
```

Out[19]:
```
array([3, 8, 0, 6, 5, 4, 7, 2])
```

In [20]:
```
#Se obtienen las frecuencias y realizamos el gráfico de barras
#Debido a la distribución simétrica de los datos, el ingreso se divide en alt
frecuencias = data['ing_fam'].value_counts()

plt.bar(frecuencias.index, frecuencias.values)
plt.xlabel('Valores')
plt.ylabel('Frecuencia')
plt.title('Frecuencia de Valores')
plt.show()
```



In [21]:
```
#Se renoombran los valores, siendo el 6142 pesos el valor que divide los bajo
info_ing_fam = {0:'bajo ingreso', 2:'bajo ingreso', 3:'bajo ingreso', 4:'bajo
                7:'alto ingreso', 8:'alto ingreso'}
n_ing_fam = [info_ing_fam.get(x) for x in data['ing_fam'].values]
data['ing_fam'] = n_ing_fam
data.head()
```

Out[21]:

	Residencia	Sexo	Edad	Educacion	ing_fam	Transporte	Eficiencia	Seguridad
0	grande	mujer	adulto_mayor	primaria	bajo ingreso	98	1	1
1	grande	mujer	adulto_mayor	preparatoria	alto ingreso	4	1	2
2	pequeño	mujer	adulto_mayor	preparatoria	bajo ingreso	4	1	1
3	grande	hombre	adulto_mayor	primaria	bajo ingreso	2	1	2
4	pequeño	mujer	adulto_mayor	primaria	bajo ingreso	6	1	2

Columna 6: Transporte

In [22]:
```
#Utilizamos unique() para observar que transporte cuenta con 18 variables úni
data.Transporte.unique()
```

Out[22]:
```
array([98,  4,  2,  6,  9,  5, 13,  7, 99,  8, 12, 14, 18,  3, 23, 19,  1,
       15])
```

In [23]:
```
#Verificamos que la columna no tenga valores nulls
pd.isna(data['Transporte']).sum()
```

Out[23]:
```
0
```

In [24]:
```
#Se reemplazan los valores 20, 23, 98, 99 por la moda de la columna, ya que e
moda3 = data['Transporte'].mode()
valores_inadecuados3 = [98, 99, 20, 23]

for i, value in enumerate(data.Transporte):
    if value in valores_inadecuados3:
        data.at[i, 'Transporte'] = moda3
```

In [25]:
```
#Verificamos que los valores se hayan reemplazados
data.Transporte.unique()
```

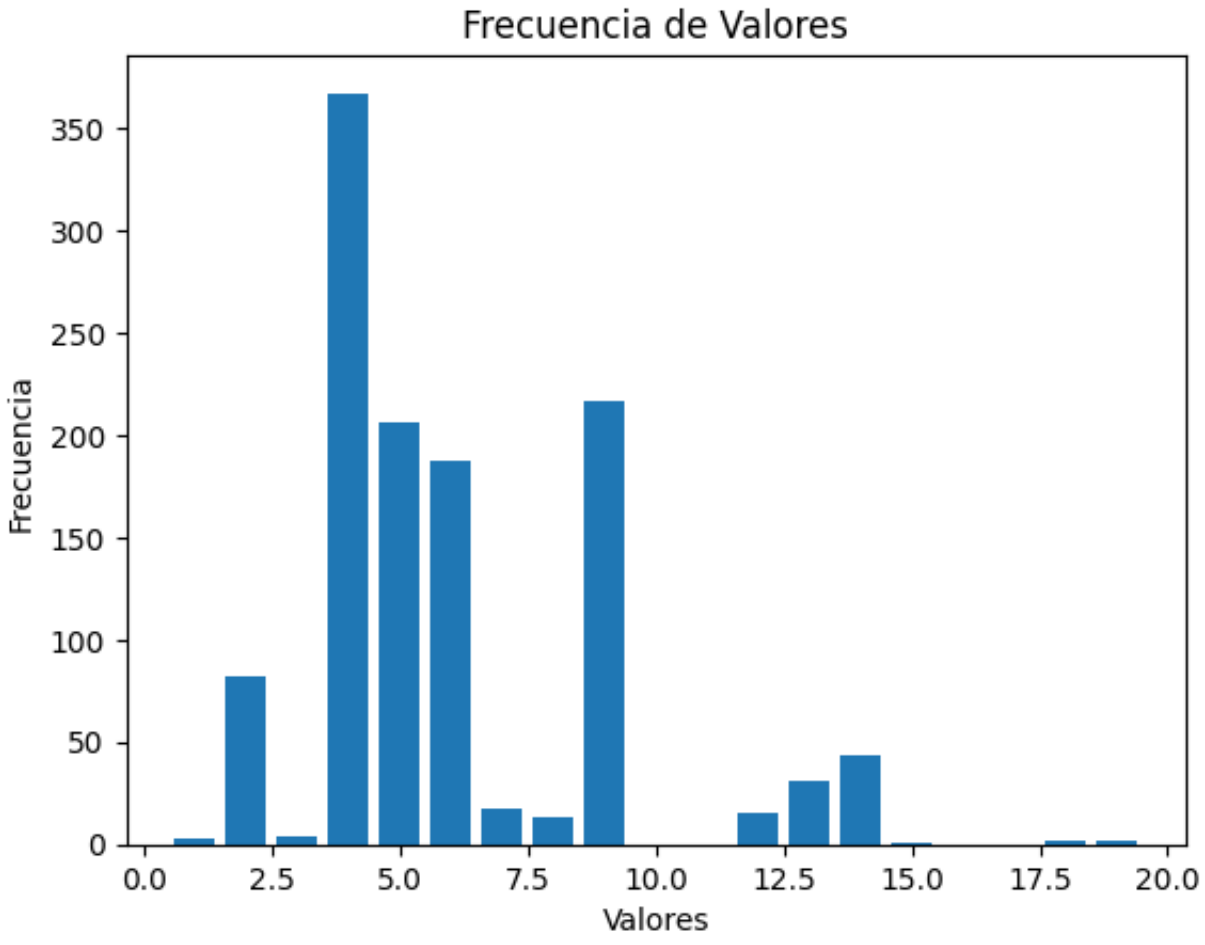
Out[25]:
```
array([ 4,  2,  6,  9,  5, 13,  7,  8, 12, 14, 18,  3, 19,  1, 15])
```

In [26]:
```
#Utilizamos la función de value_counts() para contar cuantos valores tenemos
frecuence = data['Transporte'].value_counts()

print(frecuence)
```

```
4      367
9      217
5      206
6      187
2       82
14      44
13      31
7       17
12      15
8       13
3        4
1        3
18       2
19       2
15       1
Name: Transporte, dtype: int64
```

In [27]:

```
#Generamos un histograma para verlo de forma más visual
plt.bar(frecuence.index, frecuence.values)
plt.xlabel('Valores')
plt.ylabel('Frecuencia')
plt.title('Frecuencia de Valores')
plt.show()
```



In [28]:
```
#Se genera un diccionario y se renombran los 18 diferentes medios de transpor
info_Transporte = {1:'Tren', 2:'Tren urbano (Metro)', 3:'Transporte eléctrico
                   7:'Autobús foráneo', 8:'BRT', 9:'Taxi', 12:'Motocicleta', 1
                   15:'Patineta', 18:'Animal', 19:'Avión'}
n_Transporte = [info_Transporte.get(x) for x in data['Transporte'].values]
data['Transporte'] = n_Transporte
data.head()
```

Out[28]:

	Residencia	Sexo	Edad	Educacion	ing_fam	Transporte	Eficiencia	Seguridad
0	grande	mujer	adulto_mayor	primaria	bajo ingreso	Automóvil	1	1
1	grande	mujer	adulto_mayor	preparatoria	alto ingreso	Automóvil	1	2
2	pequeño	mujer	adulto_mayor	preparatoria	bajo ingreso	Automóvil	1	1
3	grande	hombre	adulto_mayor	primaria	bajo ingreso	Tren urbano (Metro)	1	2
4	pequeño	mujer	adulto_mayor	primaria	bajo ingreso	Colectivo (Combi)	1	2

Columna 7: Eficiencia

In [29]:
```
#Se observa que en eficiencia solo se tienen dos valores únicos, siendo 1 igu
data.Eficiencia.unique()
```

Out[29]:
```
array([1, 2])
```

In [30]:
```
#Se renombran los valores
info_Eficiencia = {1:'eficiente', 2:'ineficiente'}
n_Eficiencia = [info_Eficiencia.get(x) for x in data['Eficiencia'].values]
data['Eficiencia'] = n_Eficiencia
data.head()
```

Out[30]:

	Residencia	Sexo	Edad	Educacion	ing_fam	Transporte	Eficiencia	Seguridad
0	grande	mujer	adulto_mayor	primaria	bajo ingreso	Automóvil	eficiente	1
1	grande	mujer	adulto_mayor	preparatoria	alto ingreso	Automóvil	eficiente	2
2	pequeño	mujer	adulto_mayor	preparatoria	bajo ingreso	Automóvil	eficiente	1
3	grande	hombre	adulto_mayor	primaria	bajo ingreso	Tren urbano (Metro)	eficiente	2
4	pequeño	mujer	adulto_mayor	primaria	bajo ingreso	Colectivo (Combi)	eficiente	2

Columna 8: Seguridad

In [31]:
```
#Se observa que en seguridad solo se tienen dos valores únicos, siendo 1 igua
data.Seguridad.unique()
```

Out[31]:
```
array([1, 2])
```

In [32]:
```
#Se renombran los valores
info_Seguridad = {1:'seguro', 2:'inseguro'}
n_Seguridad = [info_Seguridad.get(x) for x in data['Seguridad'].values]
data['Seguridad'] = n_Seguridad
data.head()
```

Out[32]:

	Residencia	Sexo	Edad	Educacion	ing_fam	Transporte	Eficiencia	Seguridad
0	grande	mujer	adulto_mayor	primaria	bajo ingreso	Automóvil	eficiente	seguro
1	grande	mujer	adulto_mayor	preparatoria	alto ingreso	Automóvil	eficiente	inseguro
2	pequeño	mujer	adulto_mayor	preparatoria	bajo ingreso	Automóvil	eficiente	seguro
3	grande	hombre	adulto_mayor	primaria	bajo ingreso	Tren urbano (Metro)	eficiente	inseguro
4	pequeño	mujer	adulto_mayor	primaria	bajo ingreso	Colectivo (Combi)	eficiente	inseguro

Columna 9: Ocupación

In [33]:
```
#Se observan los valores únicos de ocupación
data.Ocupación.unique()
```

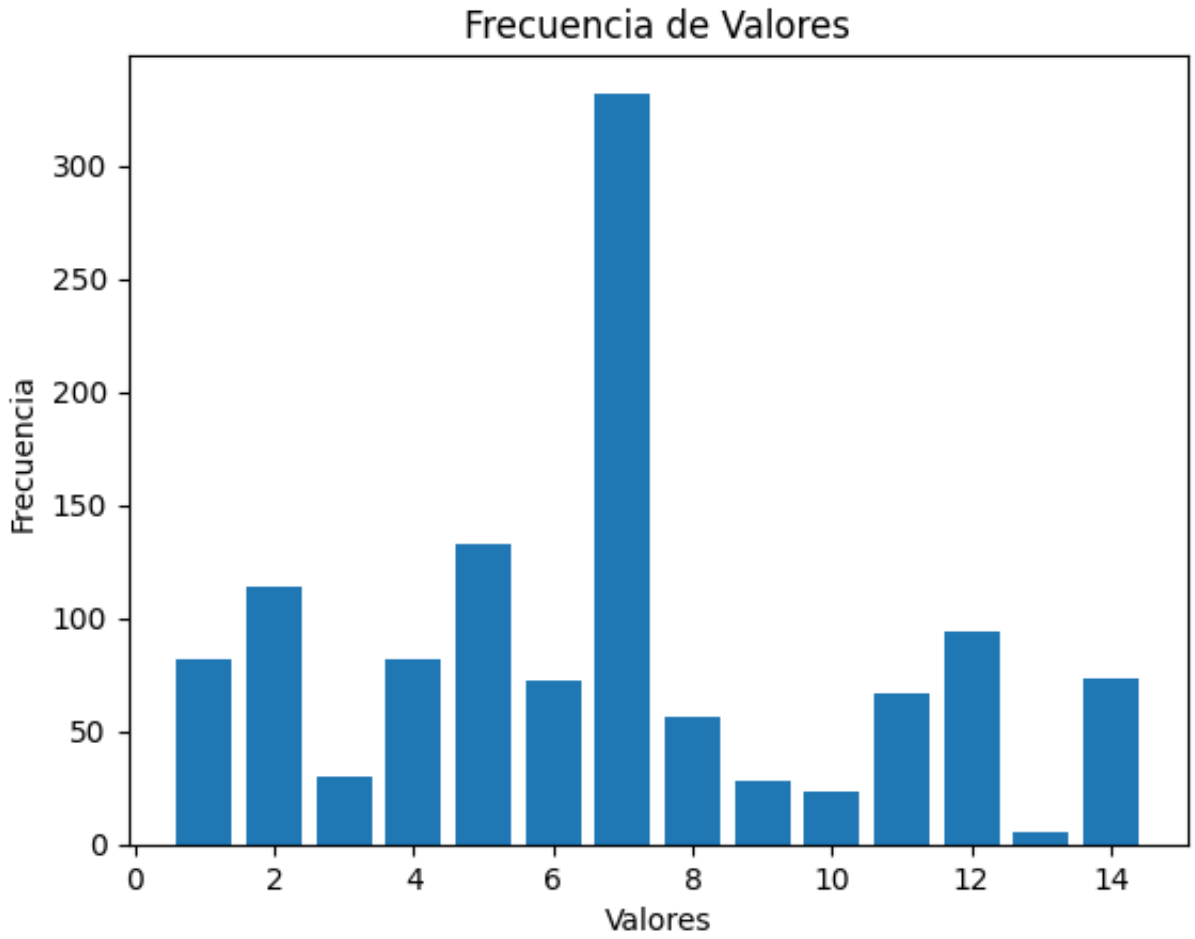
Out[33]:
```
array([ 9, 12,  1,  7, 13,  2, -1,  4,  5, 14, 11, 10,  3, 99,  6,  8, 97,
       98])
```

In [34]:
```
#Se reemplazan los valores -1, 97, 98, 99 por la moda de la columna, ya que e
moda2 = data['Ocupación'].mode()
valores_inadecuados2 = [-1, 97, 98, 99]

for i, value in enumerate(data.Ocupación):
    if value in valores_inadecuados2:
        data.at[i, 'Ocupación'] = moda2
```

In [35]:
```
#Se obtienen las frecuencias de los valores únicos para así tener una idea de
frecuencias2 = data['Ocupación'].value_counts()

print(frecuencias2)
```

```
7     332
5     133
2     114
12     94
1      82
4      82
14     73
6      72
11     67
8      56
3      30
9      28
10     23
13      5
Name: Ocupación, dtype: int64
```

In [36]:
```
# Se puede visualizar de forma más clara con un gráfico de barras
plt.bar(frecuencias2.index, frecuencias2.values)
plt.xlabel('Valores')
plt.ylabel('Frecuencia')
plt.title('Frecuencia de Valores')
plt.show()
```



Se hace la agrupación y el renombramiento. Al final se tienen únicamente cuatro grupos

- Empleados: Profesionists, Educadores, Agricultor, Empleado y Trabajador industrial
- Jefes: Administrador, Trabajador por cuenta propia y Patrón
- Servicios: Técnico, Reparador, Servicios domésticos, Servicios
- Comerciantes: este apartado no se une a ningún subconjunto ya que es con gran diferencia la moda
- Vendedor ambulante: este apartado se deja por separado ya que una de las queries asignadas al equipo envuelve directamente a este apartado

In [37]:
```
empleados = [1, 3, 4, 8, 14]
jefes = [6, 12, 13]
servicios = [2, 5, 10, 11]


for i, value in enumerate(data.Ocupación):
    if value in empleados:
        data.Ocupación[i] = 'empleado'
    elif value in jefes:
        data.Ocupación[i] = 'jefe'
    elif value in servicios:
        data.Ocupación[i] = 'servidor'
    elif value == 7:
        data.Ocupación[i] = 'comerciante'
    elif value == 9:
        data.Ocupación[i] = 'vendedor_ambulante'
data.head()
```

Out[37]:

	Residencia	Sexo	Edad	Educacion	ing_fam	Transporte	Eficiencia	Seguridad
0	grande	mujer	adulto_mayor	primaria	bajo ingreso	Automóvil	eficiente	seguro
1	grande	mujer	adulto_mayor	preparatoria	alto ingreso	Automóvil	eficiente	inseguro
2	pequeño	mujer	adulto_mayor	preparatoria	bajo ingreso	Automóvil	eficiente	seguro
3	grande	hombre	adulto_mayor	primaria	bajo ingreso	Tren urbano (Metro)	eficiente	inseguro
4	pequeño	mujer	adulto_mayor	primaria	bajo ingreso	Colectivo (Combi)	eficiente	inseguro

In [39]:
```
# Exportar la nueva base
data.to_csv('clean_data_evidencia1.csv', index=False)
```

In [ ]:

1. Lectura y análisis de los datos

Importación de librerías necesarias para redes bayesianas.

```
library(bnlearn)
```

```
#if (!requireNamespace("BiocManager", quietly = TRUE))
#  install.packages("BiocManager")
#BiocManager::install()
#BiocManager::install(c("graph", "Rgraphviz"))
```

Lectura de la base de datos final, con las variables necesarias para responder las queries establecidas.

```
data <- read.csv("/Users/tonitojuarez/Documents/RStudio/clean_c
head(data)
```

```
  Residencia   Sexo         Edad    Educacion      ing_fam
Transporte
1     grande  mujer adulto_mayor     primaria bajo ingreso
Automóvil
2     grande  mujer adulto_mayor preparatoria alto ingreso
Automóvil
3    pequeño  mujer adulto_mayor preparatoria bajo ingreso
Automóvil
4     grande hombre adulto_mayor     primaria bajo ingreso
Tren urbano (Metro)
5    pequeño  mujer adulto_mayor     primaria bajo ingreso
Colectivo (Combi)
6     grande  mujer adulto_joven preparatoria alto ingreso
Taxi
  Eficiencia Seguridad          Ocupación
1  eficiente    seguro vendedor_ambulante
2  eficiente  inseguro               jefe
3  eficiente    seguro           empleado
4  eficiente  inseguro        comerciante
5  eficiente  inseguro        comerciante
6  eficiente  inseguro               jefe
```

Dimensión de la base de datos.

```
dim(data)
```

```
[1] 1191    9
```

Verificar datos faltantes

```
sum(is.na(data))
```

```
[1] 0
```

Conversión de variables a factor para el método MLE

```
data$Residencia<-as.factor(data$Residencia)
data$Sexo<-as.factor(data$Sexo)
data$Edad<-as.factor(data$Edad)
data$Educacion<-as.factor(data$Educacion)
data$ing_fam<-as.factor(data$ing_fam)
data$Transporte<-as.factor(data$Transporte)
data$Eficiencia<-as.factor(data$Eficiencia)
data$Seguridad<-as.factor(data$Seguridad)
data$Ocupación<-as.factor(data$Ocupación)
```

2. Creación de las DAGs

DAG 1

```
DAG<-empty.graph(nodes = c("Edad", "Sexo", "ing_fam", "Educacio
```

Creación de relación y nodo entre variables

```
arc.set<-matrix(c("Edad", "Educacion",
                  "Sexo", "Educacion",
                  "ing_fam", "Educacion",
                  "Educacion", "Ocupación",
                  "Educacion", "Residencia",
                  "Ocupación", "Transporte",
                  "Residencia", "Transporte",
                  "Transporte", "Eficiencia",
                  "Transporte", "Seguridad"), byrow = TRUE, nco
                dimnames = list(NULL, c("from", "to")))
arc.set
```

```
     from          to
[1,] "Edad"        "Educacion"
[2,] "Sexo"        "Educacion"
[3,] "ing_fam"     "Educacion"
[4,] "Educacion"   "Ocupación"
[5,] "Educacion"   "Residencia"
[6,] "Ocupación"   "Transporte"
```

```
 [7,] "Residencia" "Transporte"
 [8,] "Transporte" "Eficiencia"
 [9,] "Transporte" "Seguridad"
```

Implementación de los nodos a la DAG 1

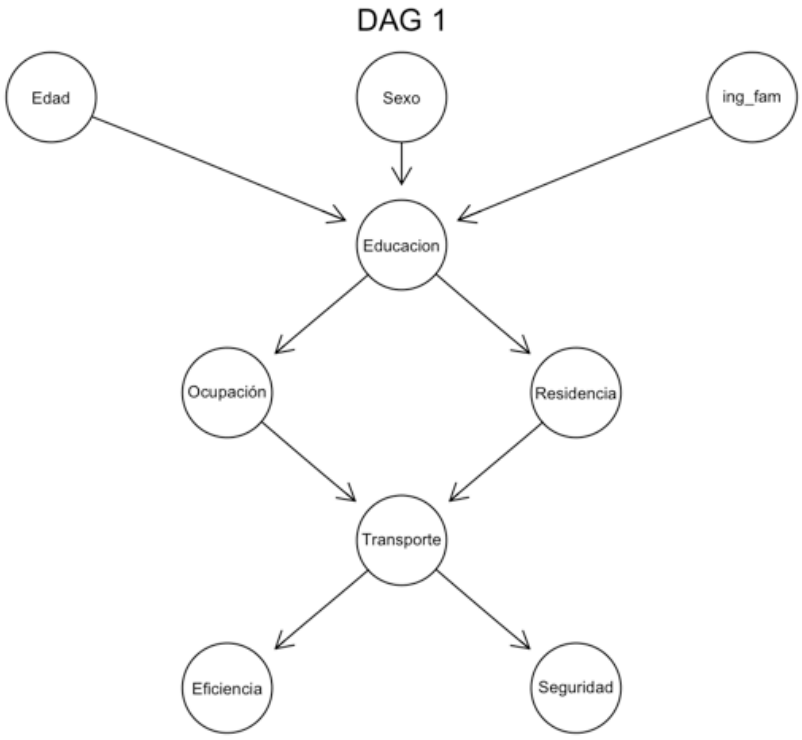
```
arcs(DAG)<-arc.set
DAG
```

```
  Random/Generated Bayesian network

  model:
   [Edad][Sexo][ing_fam][Educacion|Edad:Sexo:ing_fam]
[Ocupación|Educacion]
   [Residencia|Educacion][Transporte|Ocupación:Residencia]
   [Eficiencia|Transporte][Seguridad|Transporte]
  nodes:                                9
  arcs:                                 9
    undirected arcs:                    0
    directed arcs:                      9
  average markov blanket size:          2.89
  average neighbourhood size:           2.00
  average branching factor:             1.00

  generation algorithm:                 Empty
```

Visualización de la DAG 1

```
graphviz.plot(DAG, main = "DAG 1")
```

```
Loading required namespace: Rgraphviz
```



La primera DAG propuesta consta de 3 nodos padres: ‘Edad’, ‘Sexo’ e ‘Ingreso familiar’, la razón de esto es que edad y sexo son características íntrinsicas del ser humano, es decir, hacen referencia a la naturaleza del ser humano, por lo que no dependen de ningún factor externo fuera de los atributos humanos. El ingreso familiar se consideró como nodo padre porque por razones estructurales de la sociedad y la economía, la educación depende del ingreso familiar. Las familias con ingresos más altos generalmente tienen más recursos disponibles para invertir en la educación de calidad de sus hijos. Por otro lado, las familias de bajos ingresos pueden tener dificultades para costear estos recursos, esto se debe a las desigualdades socioeconómicas y las limitaciones de acceso a empleos bien remunerados y esto puede perpetuar un ciclo intergeneracional de desventaja (Torres, 2020). La educación juega un papel crucial en la determinación de las oportunidades laborales y el éxito profesional de una persona, es decir, la ocupación laboral del individuo. Para ascender en la jerarquía laboral y acceder a roles de mayor responsabilidad y remuneración, a menudo se requiere una educación continua y el desarrollo de habilidades adicionales. Las personas con educación superior pueden tener más oportunidades de avanzar en sus carreras que los que no cuentan con educación (‘La educación en México y su influencia en la ocupación’, s.f.). Por otro lado, tanto la residencia como la ocupación que se tiene pueden influir en el tipo de transporte que se utiliza diariamente, ya se por distintos factores como: la distancia al trabajo, los costos y valores personales pueden influir al momento de

optar por vehículos privados, transporte público, bicicletas u otras alternativas. Por último, la eficiencia y seguridad del transporte dependen del transporte más utilizado y preferido, ya que estos atributos están directamente relacionados a el medio de transporte. Bajo estos argumentos se obtuvo la primera propuesta para la DAG.

Estimación de parámetros para la DAG 1

```
bn.mle<-bn.fit(DAG, data = data, method = "mle")
```

Comprobación del método de máxima verosimilitud (MLE) con probabilidad condicional

```
bn.mle$Educacion
```

```
  Parameters of node Educacion (multinomial distribution)

Conditional probability table:

, , Sexo = , ing_fam = alto ingreso

              Edad
Educacion      adulto_joven adulto_mayor joven
  ninguno       0.000000000  0.000000000
  preescolar    0.000000000  0.000000000
  preparatoria  0.000000000  0.000000000
  primaria      0.000000000  0.000000000
  profesional   0.000000000  0.000000000
  secundaria    1.000000000  1.000000000

, , Sexo = hombre, ing_fam = alto ingreso

              Edad
Educacion      adulto_joven adulto_mayor joven
  ninguno       0.083333333  0.062500000
  preescolar    0.000000000  0.000000000
  preparatoria  0.333333333  0.291666667
  primaria      0.000000000  0.208333333
  profesional   0.083333333  0.104166667
  secundaria    0.500000000  0.333333333

, , Sexo = mujer, ing_fam = alto ingreso

              Edad
Educacion      adulto_joven adulto_mayor joven
```

```
  ninguno       0.000000000  0.023622047
  preescolar    0.016393443  0.007874016
  preparatoria  0.368852459  0.236220472
  primaria      0.073770492  0.230971129
  profesional   0.106557377  0.083989501
  secundaria    0.434426230  0.417322835

, , Sexo = , ing_fam = bajo ingreso

              Edad
Educacion      adulto_joven adulto_mayor       joven
  ninguno       0.000000000  0.000000000 0.000000000
  preescolar    0.000000000  0.000000000 0.000000000
  preparatoria  0.000000000  0.000000000 0.000000000
  primaria      0.000000000  0.000000000 0.000000000
  profesional   0.000000000  0.000000000 0.000000000
  secundaria    1.000000000  1.000000000 1.000000000

, , Sexo = hombre, ing_fam = bajo ingreso

              Edad
Educacion      adulto_joven adulto_mayor joven
  ninguno       0.000000000  0.051282051
  preescolar    0.000000000  0.000000000
  preparatoria  0.076923077  0.333333333
  primaria      0.000000000  0.051282051
  profesional   0.000000000  0.025641026
  secundaria    0.923076923  0.538461538

, , Sexo = mujer, ing_fam = bajo ingreso

              Edad
Educacion      adulto_joven adulto_mayor       joven
  ninguno       0.016949153  0.073490814 0.000000000
  preescolar    0.000000000  0.018372703 0.000000000
  preparatoria  0.254237288  0.149606299 0.000000000
  primaria      0.093220339  0.322834646 0.000000000
  profesional   0.067796610  0.062992126 0.000000000
  secundaria    0.567796610  0.372703412 1.000000000
```

Estructura de la DAG 1

```
arc.strength(DAG, data = data, criterion = "x2")
```

```
        from         to     strength
1       Edad  Educacion 1.433130e-01
2       Sexo  Educacion 1.738102e-07
```

```
3    ing_fam  Educacion 2.661524e-01
4  Educacion  Ocupación 2.652852e-16
5  Educacion Residencia 5.871046e-12
6  Ocupación Transporte 7.572555e-01
7 Residencia Transporte 1.009214e-02
8 Transporte Eficiencia 2.282981e-20
9 Transporte  Seguridad 3.256080e-06
```

DAG 2

```
DAG2<-empty.graph(nodes = c("Edad", "Sexo", "Educacion", "Ocupa
```

Creación de relación y nodo entre variables de la DAG 2

```
arc.set2<-matrix(c("Edad", "Educacion",
                   "Sexo", "Educacion",
                   "Educacion", "Ocupación",
                   "Educacion", "Residencia",
                   "Ocupación", "ing_fam",
                   "Residencia", "ing_fam",
                   "Residencia", "Eficiencia",
                   "Eficiencia", "Seguridad",
                   "Eficiencia", "Transporte"), byrow = TRUE, nc
                 dimnames = list(NULL, c("from", "to")))
arc.set2
```

```
     from         to
[1,] "Edad"       "Educacion"
[2,] "Sexo"       "Educacion"
[3,] "Educacion"  "Ocupación"
[4,] "Educacion"  "Residencia"
[5,] "Ocupación"  "ing_fam"
[6,] "Residencia" "ing_fam"
[7,] "Residencia" "Eficiencia"
[8,] "Eficiencia" "Seguridad"
[9,] "Eficiencia" "Transporte"
```

Implementación de los nodos a la DAG 2

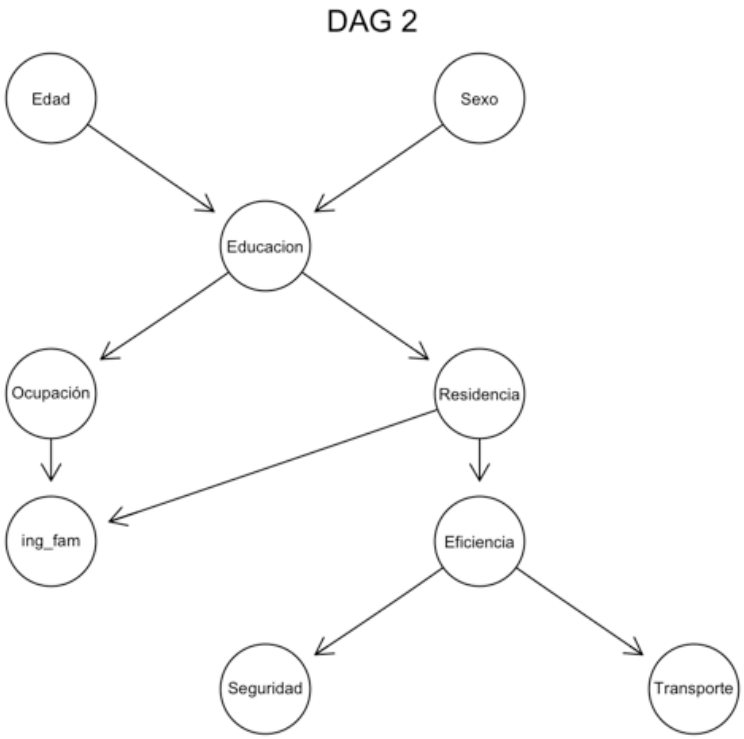
```
arcs(DAG2)<-arc.set2
DAG2
```

```
  Random/Generated Bayesian network

  model:
```

```
   [Edad][Sexo][Educacion|Edad:Sexo][Ocupación|Educacion]
[Residencia|Educacion]
   [ing_fam|Ocupación:Residencia][Eficiencia|Residencia]
[Seguridad|Eficiencia]
   [Transporte|Eficiencia]
  nodes:                                  9
  arcs:                                   9
    undirected arcs:                      0
    directed arcs:                        9
  average markov blanket size:            2.44
  average neighbourhood size:             2.00
  average branching factor:               1.00

  generation algorithm:                   Empty
```

Visualización de la DAG 2

```
graphviz.plot(DAG2, main = "DAG 2")
```



Los cambios realizados en la segunda propuesta de la DAG fue que el nodo de 'Ingreso familiar' ya no es nodo padre, sino nodo descendiente de 'Ocupación' y 'Residencia'. Este cambio fue porque el ingreso familiar tiende a depender de la ocupación y la residencia debido a las interacciones complejas entre factores económicos, sociales y geográficos, en primer lugar porque la ocupación suele ser la primera fuente de ingreso debido a las renumeraciones en el mercado laboral. En segundo, la residencia puede influir en el ingreso debido a factores como

el costo de vida, las oportunidades laborales disponibles en un área geográfica y la presencia de industrias específicas (Agualongo y Garcés, 2020). Otro cambio fue que la eficiencia del transporte público y privado depende de la residencia, ya que la movilización o el trasporte público es una de los sectores gubernamentales que más financiamiento y planeación requieren. Depende de la zona geográfica y las características de la residencia, es el presupuesto que se tendrá para la movilización y planeación de calles, de la vía pública y de los medios de transporte. Todo esto recae directamente en le eficiencia del transporte público y privado (Calvillo y Moncada, 2008). El último cambio fue que la seguridad y el transporte más utilizado/preferido depende de la eficiencia, la razón de este cambio va mucho de la mano con la razón de que la eficiencia depende de la residencia. La eficiencia del transporte está relacionada con la rapidez y la comodidad con la que las personas pueden desplazarse de un lugar a otro. Si un medio de transporte es eficiente, es más probable que las personas lo prefieran, ya que les permite ahorrar tiempo y viajar de manera más cómoda. Así mismo, un sistema de transporte eficiente suele estar respaldado por una planificación cuidadosa de rutas y horarios. Esto puede conducir a rutas más seguras que evitan áreas peligrosas o congestionadas, reduciendo así el riesgo de accidentes y situaciones peligrosas. Es por estas razones, que la eficiencia esta directamente ligada a la elección de transporte y la seguridad de la misma.

Estimación de parámetros para la DAG 2

```
bn.mle2<-bn.fit(DAG2, data = data, method = "mle")
```

Comprobación del método de máxima verosimilitud (MLE) con probabilidad condicional

```
bn.mle2$Educacion
```

```
  Parameters of node Educacion (multinomial distribution)

Conditional probability table:

, , Sexo =

             Edad
Educacion      adulto_joven adulto_mayor       joven
  ninguno       0.000000000  0.000000000 0.000000000
```

```
  preescolar    0.000000000  0.000000000 0.000000000
  preparatoria  0.000000000  0.000000000 0.000000000
  primaria      0.000000000  0.000000000 0.000000000
  profesional   0.000000000  0.000000000 0.000000000
  secundaria    1.000000000  1.000000000 1.000000000

, , Sexo = hombre

              Edad
Educacion      adulto_joven adulto_mayor joven
  ninguno       0.040000000  0.057471264
  preescolar    0.000000000  0.000000000
  preparatoria  0.200000000  0.310344828
  primaria      0.000000000  0.137931034
  profesional   0.040000000  0.068965517
  secundaria    0.720000000  0.425287356

, , Sexo = mujer

              Edad
Educacion      adulto_joven adulto_mayor       joven
  ninguno       0.008333333  0.048556430 0.000000000
  preescolar    0.008333333  0.013123360 0.000000000
  preparatoria  0.312500000  0.192913386 0.000000000
  primaria      0.083333333  0.276902887 0.000000000
  profesional   0.087500000  0.073490814 0.000000000
  secundaria    0.500000000  0.395013123 1.000000000
```

Estructura de la DAG 2

```
arc.strength(DAG2, data = data, criterion = "x2")
```

```
        from         to     strength
1       Edad  Educacion 2.824048e-04
2       Sexo  Educacion 2.362316e-11
3  Educacion  Ocupación 2.652852e-16
4  Educacion Residencia 5.871046e-12
5  Ocupación    ing_fam 2.014204e-06
6 Residencia    ing_fam 3.271055e-08
7 Residencia Eficiencia 2.652738e-10
8 Eficiencia  Seguridad 1.119869e-56
9 Eficiencia Transporte 2.282981e-20
```

3. Evaluación del rendimiento de las DAGs

Criterios basado en la verosimilitud para probar que tan bueno son los DAGs

Bayesian Information Criterion (BIC)

DAG 1

```
score(DAG, data = data, type = "bic")
```

```
[1] -10714.4
```

DAG 2

```
score(DAG2, data = data, type = "bic")
```

```
[1] -10042.96
```

Mientras más grande sea el BIC, mejor será el modelo. DAGs con scores más altos ajustan mejor a los datos.

Akaike Information Criterion (AIC)

DAG 1

```
score(DAG, data = data, type = "aic")
```

```
[1] -9964.728
```

DAG 2

```
score(DAG2, data = data, type = "aic")
```

```
[1] -9735.463
```

Después de analizar los resultados de los métodos de rendimiento BIC y AIC para las 2 redes bayesianas propuestas anteriormente, se puede observar que ambos valores de las métricas son mayores en la segunda DAG. Esto significa que la DAG 2 ajusta de una mejor manera los datos del trabajo y permiten tener una mejor aproximación a las probabilidades e hipótesis planteadas. Esta DAG número 2 se va a comparar con la DAG propuesta por Hill-Climbing para ver cual es la mejor y así poder hacer los queries.

4. Optimización de la DAG seleccionada con Hill-Climbing (HC)

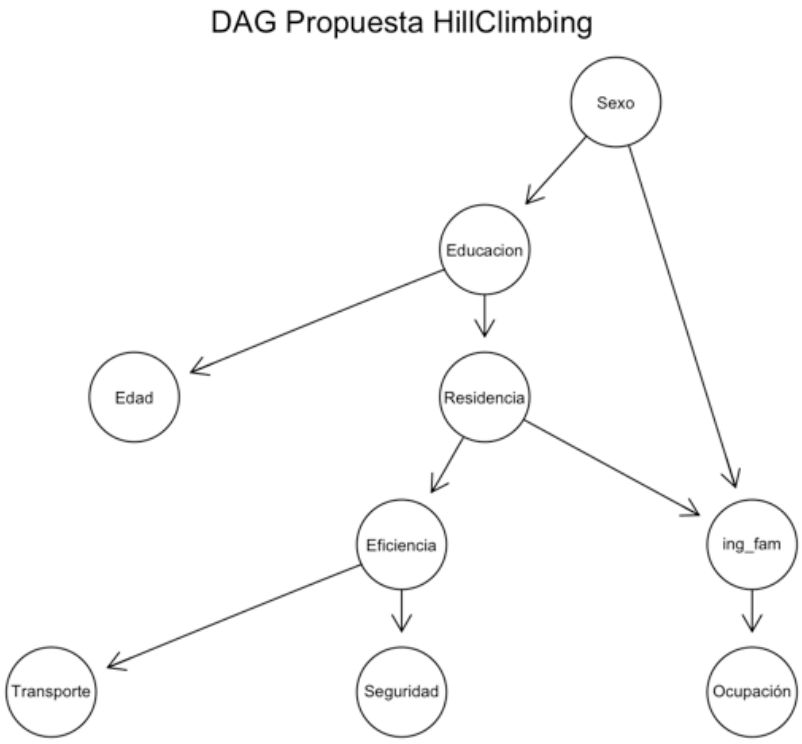
```
best_DAG<-hc(data)
```

```
modelstring(best_DAG)
```

```
[1] "[Sexo][Educacion|Sexo][Residencia|Educacion]
[Edad|Educacion][ing_fam|Residencia:Sexo]
[Eficiencia|Residencia][Transporte|Eficiencia]
[Seguridad|Eficiencia][Ocupación|ing_fam]"
```

Visualización de la nueva DAG

```
graphviz.plot(best_DAG, main = "DAG Propuesta HillClimbing")
```



Esta DAG propuesta por la función Hill-Climbing no tiene mucho sentido ya que se pueden observar nodos en los que la relación entre variables no llegan a ser coherentes. Por un lado, decir que la edad depende de la educación sería una afirmación ilógica porque la edad es una característica intrínseca y natural del tiempo transcurrido desde el nacimiento de una persona, mientras que la educación es un proceso que implica la adquisición de conocimientos, habilidades y experiencias a lo largo de la vida. Estas 2 nociones son conceptos diferentes y no están vinculadas en términos de causalidad directa. Por otro lado, establecer que la ocupación depende del ingreso familiar podría ser incoherente porque son 2 conceptos diferentes que generalmente no están directamente relacionados en términos de causa y efecto. La ocupación se refiere al trabajo, profesión o actividad que una persona realiza para ganarse la vida, mientras que el ingreso familiar se refiere a la cantidad de

dinero que una familia gana de diversas fuentes. Si bien el ingreso familiar puede influir en las decisiones de carrera de un individuo, no determina completamente la ocupación que elijan. Es por estas 2 razones, que se asenta la conclusión que la DAG propuesta de Hill-Climbing no tiene fundamentos lógicos y racionales (Benno, 1985).

5. Evaluación del rendimiento de la óptima DAG

Bayesian Information Criterion (BIC)

```
score(best_DAG, data = data, type = "bic")
```

```
[1] -9943.853
```

Akaike Information Criterion (AIC)

```
score(best_DAG, data = data, type = "aic")
```

```
[1] -9738.01
```

Se puede observar que en la métrica AIC el resultado de la DAG número 2 es ligeramente mejor que el de la DAG propuesta por la función hill-climbing. Aún así, la función hill-climbing tiene un resultado mejor en la métrica BIC con respecto a la DAG número 2 propuesta al inicio. Ambos DAG son buenos, sin embargo, la métrica BIC suele tener más peso que la métrica AIC. Es por eso, que la DAG propuesta por la función hill-climbing es una mejor estructura.

Por la misma razón de que la DAG propuesta por la función de Hill-Climbing carece de razonamiento lógico y porque las métricas BIC y AIC de ambas DAGs tienen valores cercanos, se utilizará la DAG 2 para realizar las preguntas de hipótesis.

Aplicación

1. Impresión de diccionario

Se utiliza el diccionario para tener las variables como referencia para resolver las queries.

```
unique(data$Residencia)
```

```
[1] grande  pequeño
Levels: grande pequeño
```

```
unique(data$Sexo)
```

```
[1] mujer  hombre
Levels:  hombre mujer
```

```
unique(data$Edad)
```

```
[1] adulto_mayor adulto_joven joven
Levels: adulto_joven adulto_mayor joven
```

```
unique(data$Educacion)
```

```
[1] primaria     preparatoria secundaria   profesional
ninguno
[6] preescolar
Levels: ninguno preescolar preparatoria primaria profesional
secundaria
```

```
unique(data$ing_fam)
```

```
[1] bajo ingreso alto ingreso
Levels: alto ingreso bajo ingreso
```

```
unique(data$Transporte)
```

```
 [1] Automóvil            Tren urbano (Metro)  Colectivo
(Combi)
 [4] Taxi                 Camión               Mototaxi
 [7] Autobús foráneo      BRT                  Motocicleta
[10] Bicicleta            Animal               Transporte
eléctrico
[13] Avión                Tren                 Patineta
15 Levels: Animal Autobús foráneo Automóvil Avión Bicicleta
BRT ... Tren urbano (Metro)
```

```
unique(data$Eficiencia)
```

```
[1] eficiente   ineficiente
Levels: eficiente ineficiente
```

```
unique(data$Seguridad)
```

```
[1] seguro   inseguro
Levels: inseguro seguro
```

```
unique(data$Ocupación)
```

```
[1] vendedor_ambulante jefe                empleado
comerciante
[5] servidor
Levels: comerciante empleado jefe servidor vendedor_ambulante
```

Entrenar la DAG con los datos para responder las queries.

```
bn<-bn.fit(DAG2, data = data)
```

1.- Queremos saber si el transporte público en ciudades grandes es más eficiente que en ciudades pequeñas.

Probabilidad de eficiencia transporte público para ciudades grandes:

```
cpquery(bn, event = (Eficiencia == "eficiente") , evidence = (F
```

```
[1] 0.5681742
```

Probabilidad de eficiencia transporte público para ciudades pequeñas:

```
cpquery(bn, event = (Eficiencia == "eficiente") , evidence = (F
```

```
[1] 0.7631411
```

Se puede ver que hay más posibilidades de que el transporte público sea más eficiente en localidades pequeñas comparada a localidades grandes.

2.- ¿Qué probabilidad hay de que una persona viaje en tren, dado que sea vendedor ambulante?

En este ejemplo se utiliza la variable de tren urbano porque este es más concurrido que el tren ferrocarril:

```
cpquery(bn, event = (Transporte == "Tren urbano (Metro)") , evi
```

```
[1] 0.06859692
```

La probabilidad de que una persona viaje en tren dado que es vendedor ambulante es de 7%.

3.- ¿Quiénes son más probables a sentirse seguros en el transporte público, los hombres con estudios universitarios o las mujeres con estudios universitarios?

Probabilidad de que hombres con estudios universitarios se sientan seguros en transporte:

```
cpquery(bn, event = (Seguridad == "seguro") , evidence = ((Sexo
```

```
[1] 0.4232786
```

Probabilidad de que mujeres con estudios universitarios se sientan seguros en transporte:

```
cpquery(bn, event = (Seguridad == "seguro") , evidence = ((Sexo
```

```
[1] 0.4216091
```

Se puede decir que es más probable que un hombre con estudios universitarios se sienta más seguro en el transporte público que una mujer con los mismos estudios.

4.- ¿Cómo influye el sexo de la persona en la elección del medio de transporte más utilizado, tomando en cuenta el nivel de ingreso familiar y la eficiencia del transporte público?

Primero, se debe encontrar el medio de transporte más utilizado.

```
freq_table <- table(data$Transporte)
most_common_name <- names(freq_table)[which.max(freq_table)]

print(paste("El medio de transporte más utilizado es:", most_co
```

```
[1] "El medio de transporte más utilizado es: Automóvil"
```

Se sabe que para la pregunta 4, el automóvil es el medio de transporte más utilizado.

Generación de probabilidades para los diferentes casos que se puede presentar para hombre y mujer respectivamente

Hombre:

Probabilidad de que el ser hombre influya en la elección de automóvil

como medio de transporte cuando el nivel de ingreso familiar es bajo y el transporte público es eficiente:

```
probh1 <- cpquery(bn, event = ((Transporte == "Automóvil") & (i
probh1
```

```
[1] 0.06890484
```

Probabilidad de que el ser hombre influya en la elección de automóvil como medio de transporte cuando el nivel de ingreso familiar es bajo y el transporte público es ineficiente:

```
probh2 <- cpquery(bn, event = ((Transporte == "Automóvil") & (i
probh2
```

```
[1] 0.08459939
```

Probabilidad de que el ser hombre influya en la elección de automóvil como medio de transporte cuando el nivel de ingreso familiar es alto y el transporte público es eficiente:

```
probh3 <- cpquery(bn, event = ((Transporte == "Automóvil") & (i
probh3
```

```
[1] 0.06393292
```

Probabilidad de que el ser hombre influya en la elección de automóvil como medio de transporte cuando el nivel de ingreso familiar es alto y el transporte público es ineficiente:

```
probh4 <- cpquery(bn, event = ((Transporte == "Automóvil") & (i
probh4
```

```
[1] 0.08988381
```

Mujer:

Probabilidad de que el ser mujer influya en la elección de automóvil como medio de transporte cuando el nivel de ingreso familiar es bajo y el transporte público es eficiente:

```
probm1 <- cpquery(bn, event = ((Transporte == "Automóvil") & (i
probm1
```

```
[1] 0.07069373
```

Probabilidad de que el ser mujer influya en la elección de automóvil como medio de transporte cuando el nivel de ingreso familiar es bajo y el transporte público es ineficiente:

```
probm2 <- cpquery(bn, event = ((Transporte == "Automóvil") & (i
probm2
```

```
[1] 0.08586365
```

Probabilidad de que el ser mujer influya en la elección de automóvil como medio de transporte cuando el nivel de ingreso familiar es alto y el transporte público es eficiente:

```
probm3 <- cpquery(bn, event = ((Transporte == "Automóvil") & (i
probm3
```

```
[1] 0.06337616
```

Probabilidad de que el ser mujer influya en la elección de automóvil como medio de transporte cuando el nivel de ingreso familiar es alto y el transporte público es ineficiente:

```
probm4 <- cpquery(bn, event = ((Transporte == "Automóvil") & (i
probm4
```

```
[1] 0.0884571
```

Con todas las probabilidades y sus combinaciones, se hace una suma de probabilidades y así responder la pregunta

Suma probabilidad Hombre:

```
probh <- probh1 + probh2 + probh3 + probh4
probh
```

```
[1] 0.307321
```

Suma probabilidad mujer:

```
probm <- probm1 +probm2 + probm3 + probm4
probm
```

```
[1] 0.3083906
```

Con estos resultados se puede decir que la mujer es más probable a elegir el automóvil como medio de transporte más utilizado, tomando en cuenta su nivel de ingreso familiar y la eficiencia de este transporte público. Pero como la diferencia entre probabilidades es muy pequeña, se puede inferir que el sexo no influye en la elección del transporte público dado el ingreso familiar y eficiencia.

Conclusión

Este trabajo demuestra la efectividad y utilidad de las Redes Bayesianas como herramienta analítica para responder preguntas e investigar hipótesis en diversos contextos. Además, las redes bayesianas proporcionan una representación visual intuitiva de las relaciones causales entre variables, permitiendo modelar y cuantificar la incertidumbre de manera coherente. La aplicación de las Redes Bayesianas en la resolución de queries e hipótesis ha permitido un enfoque estructurado y sistemático para analizar datos complejos. Al capturar las dependencias probabilísticas entre las variables, estas redes proporcionan una forma rigurosa de evaluar el impacto de cambios en una variable sobre otras, así como de estimar la probabilidad de eventos futuros dadas las observaciones actuales. En conclusión, las redes bayesianas permiten hacer inferencias respecto al transporte público y privado en México, así como obtener un resultado de gran utilidad en la toma de decisiones.

Referencias

Agualongo, D. y Garcés, A. (2020). El nivel socioeconómico como factor de influencia

en temas de salud y educación. Universidad de las Fuerzas Armadas Espe. [PDF]

Benno, S. (1985). Educación y dependencia: el papel de la educación comparada. UNESCO. [PDF]

Calvillo, A. y Moncada, G. (2008). Eficiencia del transporte público y privado. El consumidor. [PDF]

La educación en México y su influencia en la ocupación. (s. f.) Centro de

Estudios Espinosa Yglesias. Recuperado de https://ceey.org.mx/la-educacion-en-mexico-y-su-influencia-en-la-ocupacion/

Santiesteban, J. C., Pérez, d. y Hernández, C. (2012). Definici?n de Redes Bayesianas y sus aplicaciones. Revista Vinculando. https://vinculando.org/articulos/redes-bayesianas.html

Torres, G. y Ayala, E. (noviembre 2020). El ingreso familiar como determinante de la asistencia escolar de los jóvenes en México. Problemas del desarrollo, 201. Recuperado de https://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S0301-70362020000200085

Wolf et. al. (11 de marzo de 2019). Dinámica y controles de procesos-químicos. [PDF]