UNIVERSIDAD INTERNACIONAL SAN ISIDRO LABRADOR

ESPECIALIZACIÓN DE DATA SCIENCE

PROYECTO NO. 1: DATA MINING

ULISES JOSÉ BUSTAMANTE MORA

DR. SAMUEL SALDAÑA VALENZUELA

MAYO, 2024

Contenido

ANTECEDENTES

Pronosticar cambios en el sector de migración es de suma importancia ya que este fenómeno social conlleva consigo múltiples beneficios, según (ONU, 2016):

> "La migración puede propiciar un aumento de la tasa de crecimiento del PIB en los países de destino, el incremento de los salarios de los migrantes, y la expansión de los beneficios indirectos de las remesas para los países de origen."

Explicando la importancia de la migración, si este tema económico lo combinamos con técnicas analísticas, tendríamos un análisis macroeconómico importante para que en ese se basen leyes y normativas fundamentadas en datos y a la vez, para la creación de planes de desarrollo económico y poblacional.

Lo anterior mencionado se podría lograr con la aplicación de técnicas estadísticas junto con software especializado que nos permite analizar una gran cantidad de datos y que este mismo nos arroje el resultado de dicho análisis estadístico para obtener una visión realista sobre la situación y tomar decisiones basadas en datos y no en sesgos.

Dando más fuerza a la idea previamente planteada, según (Leyva, 2023)

> "La estadística constituye una herramienta de gran importancia para la toma de decisiones de trascendencia en nuestras sociedades, y sobre todo en nuestros entornos educativos"

JUSTIFICACIÓN

Partiendo del punto que está más que demostrado las grandes ventajas del análisis de datos para la resolución de problemas, que según (Amazon, s.f.):

> "El análisis de datos ayuda a las empresas a obtener una mayor visibilidad y un conocimiento más profundo de sus procesos y servicios. Les proporciona información detallada sobre la experiencia del cliente y sus problemas. Al cambiar el paradigma más allá de los datos para conectar los conocimientos con la acción, las empresas pueden crear experiencias personalizadas para los clientes y productos digitales relacionados, optimizar las operaciones y aumentar la productividad de los empleados."

Con la aplicación correcta de la tecnología a problemas sociales, podríamos desarrollar ideas de alto impacto a la población que permitan mejorar su calidad de vida. Con esto se obtiene una sociedad sana y segura donde las generaciones, conforme pasan, van agregando más valor agregado gracias a una cultura de conciencia a la política.

OBJETIVO GENERAL

- Revisar datos históricos relevantes sobre migración, incluyendo tendencias demográficas y flujos migratorios.

OBJETIVOS ESPECÍFICOS

- Medir cuantitativamente mediante métodos estadísticos, características importantes en cambios en la migración.
- Desarrollar conclusiones basadas en hechos obtenidos de un análisis de datos exploratorio del dataset.
- Transformar el conjunto de datos mediante técnicas de Data Mining para su correcta interpretación.

CRISP-DM

Comprensión del negocio

El objetivo principal de cualquier país es resguardar los derechos fundamentales de todos sus habitantes, esto mediante la creación de políticas que favorezcan dicha misión. En la toma correcta de decisiones para aprobar o rechazar leyes se precisa de datos para corroborar si es factible o no. Esto es de suma importancia ya que puede afectar la calidad de vida de las personas.

El tema de la migración es un tema fundamental para cualquier país, este mismo debe de tener ciertas características que, según el estado sociopolítico de la nación, para que de esta manera sacar lo mejor de las ventajas del fenómeno social de la migración. Para lo toma optima de leyes, se debe tener bajo consideración como esta ha afectado a la nación en épocas anteriores.

En este contexto, se precisa de las técnicas de Data Mining para el análisis de datos como correlaciones, visualización de datos y estadísticos a partir del conjunto de datos previamente recolectados para juzgar en qué periodo se obtuvo mejores resultados y ver sus correlaciones para que de esta manera plantear una solución al problema.

Comprensión de los datos

El conjunto de datos bajo estudio será una recopilación de 1979 hasta el 2015, sobre todos los vuelos, tanto de partida como de llegada del país de Nueza Zelanda, con información extra como el destino y nacionalidad de dichas personas. A la vez, contará con 86525 entradas, sin datos faltantes y donde también destaca la cantidad de personas que se involucran en dichas entradas.

Tomando en cuenta el tipo de dato por variable, se distribuyen de la siguiente manera:

- Measure: Cualitativa nominal
- Country: Cualitativa nominal
- Citizenship: Cualitativa nominal
- Year: Cuantitativa discreta
- Value: Cuantitativa continua

Ilustración 1 Número total de entradas y de datos faltantes y de tipos de datos

Número total de entradas y de datos faltantes y de tipos de datos

```
In [9]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 86526 entries, 0 to 86525
Data columns (total 5 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   Measure      86526 non-null  object
 1   Country      86526 non-null  object
 2   Citizenship  86526 non-null  object
 3   Year         86526 non-null  int64
 4   Value        86454 non-null  float64
dtypes: float64(1), int64(1), object(3)
memory usage: 3.3+ MB
```

Ilustración 2 Valores únicos entre columnas Measure, Citizenship, Year y Country

Valores únicos en la columna de Measure

```
In [14]: df["Measure"].unique()
```

```
Out[14]: array(['Arrivals', 'Departures', 'Net'], dtype=object)
```

Valores únicos en la columna de Citizenship

```
In [19]: df["Citizenship"].unique()
```

```
Out[19]: array(['New Zealand Citizen', 'Australian Citizen',
       'Total All Citizenships'], dtype=object)
```

Valores únicos en la columna de Year

```
In [20]: df["Year"].unique()
```

```
Out[20]: array([1979, 1980, 1981, 1982, 1983, 1984, 1985, 1986, 1987, 1988, 1989,
       1990, 1991, 1994, 1992, 1993, 1995, 1996, 1997, 1998, 1999, 2000,
       2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011,
       2012, 2013, 2014, 2016, 2015], dtype=int64)
```

Valores únicos en la columna de Country

```
In [18]: df["Country"].unique()
```

```
Out[18]: array(['Oceania', 'Antarctica', 'American Samoa', 'Australia',
       'Cocos Islands', 'Cook Islands', 'Christmas Island', 'Fiji',
       'Micronesia', 'Guam', 'Kiribati', 'Marshall Islands',
       'Northern Mariana Islands', 'New Caledonia', 'Norfolk Island',
       'Nauru', 'Niue', 'New Zealand', 'French Polynesia',
       'Papua New Guinea', 'Pitcairn Island', 'Palau', 'Solomon Islands',
       'French Southern Territories', 'Tokelau', 'Tonga', 'Tuvalu',
       'Vanuatu', 'Wallis and Futuna', 'Samoa', 'Asia', 'Afghanistan',
       'Armenia', 'Azerbaijan', 'Bangladesh', 'Brunei Darussalam',
```

Preparación de los datos

El avistamiento de valores faltantes es crucial para obtener resultados precisos, esto ya que, al haberlos, el modelo estadístico podría interpretarlo de manera incorrecta. Es importante remplazarlos con técnicas como la asignación del promedio o mediana. Lo cual suele funcionar ya que, matemáticamente hablando, no afecta significativamente al modelo.

Ilustración 3 Cambiar los valores faltantes por la mediana

Notar si hay datos faltantes en el conjunto de datos

```
In [26]: df.isnull().any()
```

```
Out[26]: Measure        False
         Country        False
         Citizenship    False
         Year           False
         Value           True
         dtype: bool
```

Se nota que efectivamente hay valores faltantes

```
In [34]: ISNULLfiltro = df["Value"].isnull()
         df[ISNULLfiltro].head()
```

Out[34]:

	Measure	Country	Citizenship	Year	Value
68535	Arrivals	Czechoslovakia	New Zealand Citizen	2009	NaN
68536	Arrivals	Czechoslovakia	Australian Citizen	2009	NaN
68537	Arrivals	Czechoslovakia	Total All Citizenships	2009	NaN
69294	Departures	Czechoslovakia	New Zealand Citizen	2009	NaN
69295	Departures	Czechoslovakia	Australian Citizen	2009	NaN

Se cambia el valor NaN por la mediana de la columna y se comprueba que no hayan más.

```
In [39]: mediana = df["Value"].median()

         df["Value"].fillna(mediana, inplace=True)
```

```
In [38]: df.isnull().any()
```

```
Out[38]: Measure        False
         Country        False
         Citizenship    False
         Year           False
         Value          False
         dtype: bool
```

Otro punto importante es el control de datos atípico, estos datos tienden a cambiar el resultado final negativamente, esto ya que muestra como hecho un valor totalmente alejado de la normalidad, lo cual, a tomarse en cuenta, cambia totalmente la fórmula en el modelo estadístico.

En este caso, se construye un histograma para lograr ver si hay valores normalizados (no los hay ya que a cinco bins se presenta absolutamente una columna, lo cual indica que hay valores atípicos). A la vez, también se dibuja un boxplot con el mismo propósito (sí se presentan valores atípicos.)

Para la construcción de un histograma se precisa de una variable numérica, en este caso sería la columna “Value”. Dado que nos ayuda para la agrupación de todos sus datos y dependiendo de la cantidad de bins, estos se agrupan y visualmente se pueden ver los rangos donde se caen mas y menos los datos.

En caso del boxplot, se necesita una variable categórica y una numérica, esto para segmentar y ver en cuales dichos segmentos recaen la mayor cantidad de entradas y de esta manera, también ver la cantidad de datos que se salen de lo común por categoría. Se usarán las columnas “Value” (numérica) y “Measure” (categorica).

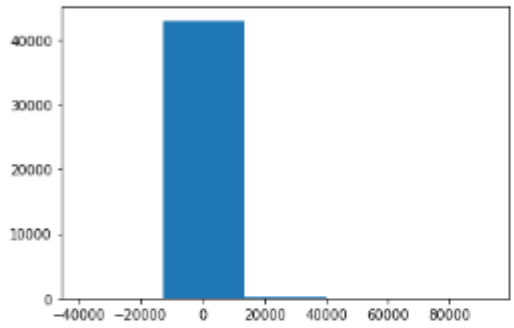
En este caso, no se realizó un remplazo, ya que, dado a las temporadas de viajes a través de año y agentes políticos externos, llega a ser normal que haya una fluctuación de números mayor o menor mediada según la época del año. Se recalca que se realizó un muestreo aleatorio del 50% dado a la longitud del conjunto de datos.

Para esta visualización no se precisó de un tratamiento especial, aunque para mejorar su distribución se pudo realizar una estandarización de los datos, esto cambian su valor bajo una misma desviación y mínimos y máximo, esto ayuda matemáticamente al modelo estadístico ya que los datos estar normalizados.

Ilustración 4 Análisis de datos atípicos

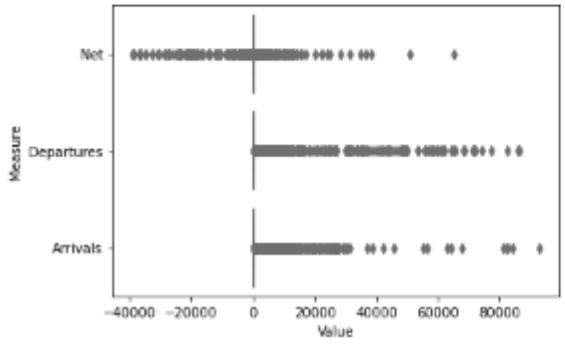
Ver si la columna de Value esta normalizada

```
In [28]: import matplotlib.pyplot as plt

         muestra_aleatoria = df.sample(frac=0.5, random_state=101)

         plt.hist(muestra_aleatoria["Value"], bins=5)
         plt.show()
```



Boxplot para ver si hay datos atipicos

```
In [32]: sns.boxplot(x=muestra_aleatoria["Value"], y = muestra_aleatoria["Measure"], color='skyblue')
Out[32]: <AxesSubplot:xlabel='Value', ylabel='Measure'>
```



Modelado de datos

Saber las medidas de tendencia central es de suma importancia ya que nos brinda de a primera mano información bastante valiosa sobre el dataset, y a partir de estas mismas sacar primeras conclusiones, dentro de ellas se presentan: el promedio y la mediana. A la vez, existen las medidas de dispersión que cumplen también con un papel importante. Estas serían: la varianza y desviación estándar.

Dado a la naturaleza del conjunto de datos, la única variable viable para aplicar las anteriores medidas seria la del total de personas. Se nota otras medidas importantes, como el

conteo, mínimos, máximos y los percentiles. Notamos un promedio bajo a comparación con la desviación estándar, los cual nos indica que los datos no son muy cercanos entre sí.

Para construir dicha tabla, se precisa solamente de una variable numerica.

Ilustración 5 Medidas de tendencia y dispersión

Medidas de tendencia y de disperción

```
In [13]: muestra_aleatoria.describe()["Value"]

Out[13]: count    43263.000000
         mean       241.898320
         std       2979.671533
         min     -39114.000000
         25%          0.000000
         50%          0.000000
         75%          6.000000
         max      92660.000000
         Name: Value, dtype: float64
```

Para calcular la moda en variables, se puede aplicar tanto en variables categóricas como numéricas.

Obteniendo la moda de cada variable

```
In [17]: muestra_aleatoria.mode()
```

Out[17]:

	Measure	Country	Citizenship	Year	Value
0	Net	Ukraine	New Zealand Citizen	2010	0.0

Ilustración 6 Moda de cada variable

Para la mediana, coeficiente de variación, curtosis y tablas cruzadas, solo se aplica a variables numéricas.

Ilustración 7 La mediana de variables numéricas

Cálculo de la mediana

```
In [18]: muestra_aleatoria.median()

Out[18]: Year     1997.0
         Value       0.0
         dtype: float64
```

Ilustración 8 Coeficiente de variación

Cálculo del coeficiente de variación

```
In [21]: muestra_aleatoria["Value"].std() / muestra_aleatoria["Value"].mean()

Out[21]: 12.317867844954
```

Ilustración 9 Curtosis

Curtosis del conjunto de datos

```
In [22]: muestra_aleatoria.kurtosis()

Out[22]: Year      -1.200321
         Value    322.655767
         dtype: float64
```

Ilustración 10 Tabla de agrupación

Agrupacion de la suma de personas, segmentada por la ciudadania y tipo de vuelo

```
In [24]: muestra_aleatoria.pivot_table(values = "Value", index= "Citizenship", columns = "Measure", aggfunc = sum)
```

Out[24]:

Measure	Arrivals	Departures	Net
Citizenship			
Australian Citizen	258471.0	161052.0	63468.0
New Zealand Citizen	1296458.0	2674016.0	-1115723.0
Total All Citizenships	3356208.0	3156397.0	614900.0

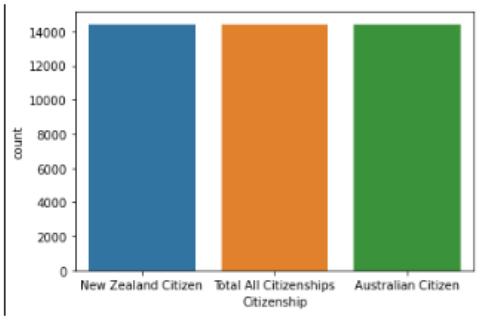
Saber el conteo de las entradas totales por segmentación es una práctica muy importante, esto para notar a primera vista si hay un patrón en su comportamiento, dando como resultado un factor importante a tomar en cuenta. En este caso, cuando segmentamos por nacionalidad y modo de vuelo, se presenta bastante distribuida entre todas las opciones.

Se precisa de una variable categórica o numerica que funcione como categórica para construir este tipo de gráfico, esto ya que hace un conteo por categoría del total de entradas.

Ilustración 11 Conteo por nacionalidad

Conteo de entradas por nacionalidad

In [16]:
```
sns.countplot(x='Citizenship',data=muestra_aleatoria)
plt.show()
```



In [28]:
```
CitiGroupBy = muestra_aleatoria.groupby("Citizenship")["Value"].count()
CitiGroupBy = pd.DataFrame(CitiGroupBy).reset_index()
CitiGroupBy.head()
```

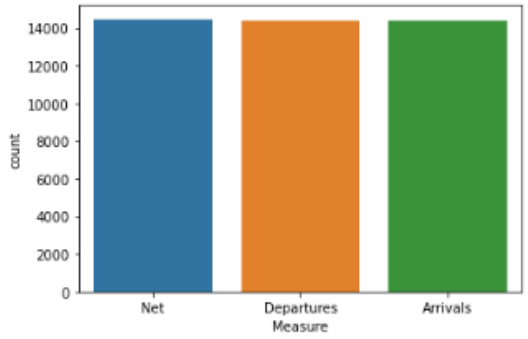
Out[28]:

	Citizenship	Value
0	Australian Citizen	14411
1	New Zealand Citizen	14429
2	Total All Citizenships	14423

Ilustración 12 Conteo por modo de vuelo

Conteo de entradas por modo de vuelo

```
In [29]: sns.countplot(x='Measure',data=muestra_aleatoria)
         plt.show()
```




```
In [27]: MeasureGroupBy = muestra_aleatoria.groupby("Measure")["Value"].count()
         MeasureGroupBy = pd.DataFrame(MeasureGroupBy).reset_index()
         MeasureGroupBy.head()
```

Out[27]:

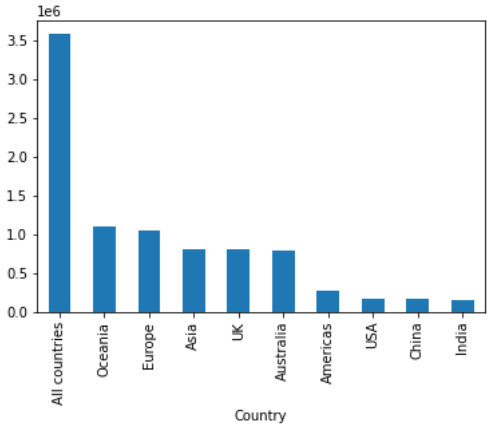
	Measure	Value
0	Arrivals	14373
1	Departures	14411
2	Net	14479

Otro gráfico de barras a considerar es uno en que nos muestre el país o continente donde más movimientos tienen con Nueva Zelanda, esta información nos ayudaría para tener un panorama más claro acerca de la situación actual migratoria, también podría explicar fenómenos internacionales.

Ilustración 13 Países más comunes

Paises mas comunes

```
In [36]: top_10_paises = muestra_aleatoria.groupby("Country")["Value"].sum().sort_values(ascending=False).head(10)
         top_10_paises.plot(kind="bar")
Out[36]: <AxesSubplot:xlabel='Country'>
```



Un tipo de grafico muy importante para agregar, sería el Scatterplot, que nos dice si hay algún tipo de correlación entre dos variables numéricas, saber este dato sería de gran importancia ya que podría analizar a partir de aquí diferentes tipos de mediciones o formulación de hipótesis.

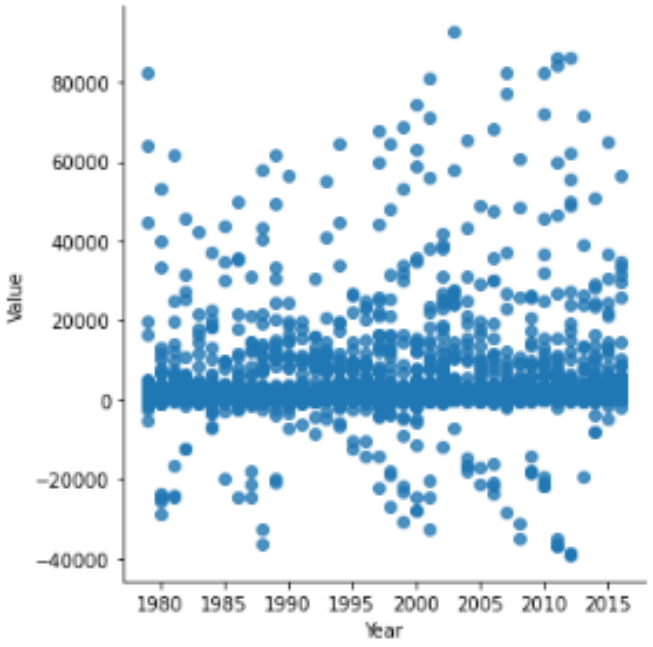
Para nuestro caso, creamos un gráfico con el eje en X el año y en el eje Y el valor. Podríamos notar principalmente una leve correlación ya que la mayoría de los puntos se presentan juntos, es cierto que hay otros que se salen de la regla, pero cabe a rescatar que son una minoría.

Para la construcción del mismo se precisa de dos variables numéricas, esto ya que, al ponerlas en un plano cartesiano, el grafico dibuja un punto donde las dos variables se intersecan, así con todas las entradas. Permitiendo visualmente ver el comportamiento de todos los puntos de conjunto de datos.

Ilustración 14 Scatterplot

Scatterplot para notar algun tipo de correlación

```
In [17]: sns.lmplot(x='Year',y='Value',data=muestra_aleatoria)
Out[17]: <seaborn.axisgrid.FacetGrid at 0x17917d30130>
```

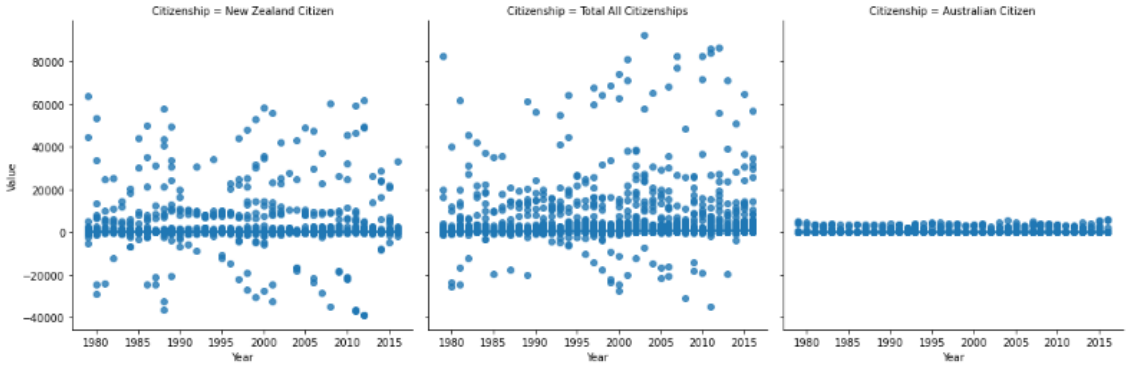


También diseñamos un Scatterplot, pero con la segmentación por ciudadanía para granular más el análisis y notar si alguno de ellos si presentan una mejor visibilidad con respectos a las dos variables a comparar. Notamos que las ciudadanías de Australia y Nueva Zelanda si presentan una mejor correlación.

Para la construcción de este gráfico, se agrega una variable categórica que funciona como filtro para el cálculo de cada uno.

Scatterplot segmentado por ciudadania

```
In [22]: sns.lmplot(x='Year',y='Value',data=muestra_aleatoria,col = "Citizenship")
Out[22]: <seaborn.axisgrid.FacetGrid at 0x17917bc1e80>
```



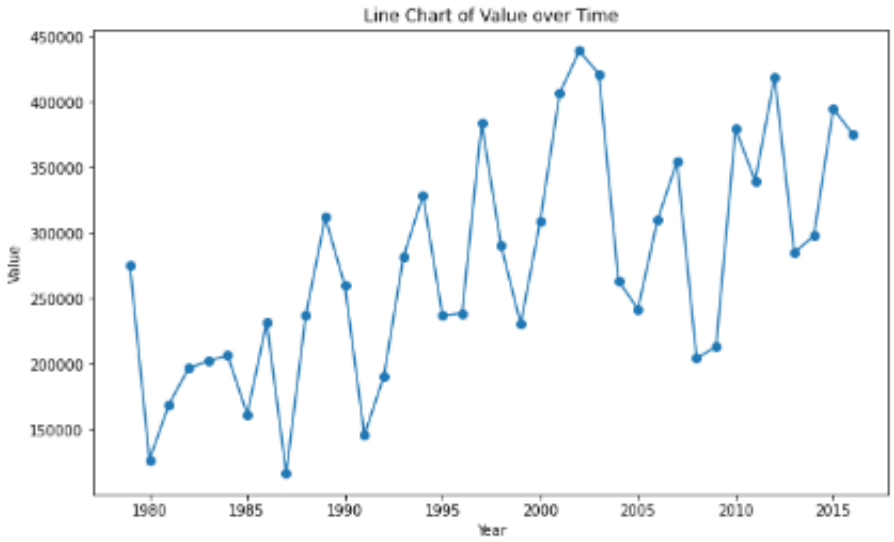
Ya que trabajamos con tiempo, como puede ser la columna de Año, declaramos que sería de gran importancia crear un gráfico de línea para notar visualmente algún tipo de tendencia. Mediante este análisis, notamos una tendencia al alza conforme fueron pasando los años. Este descubrimiento es de alta relevancia ya que nos indicia que en futuros años también podría crecer aún más.

Para construir un lineplot se precisa de una variable categórica o numerica que funcione como categoría, el cual es nuestro caso, pero que hago referencia a una fecha como puede ser días, meses, trimestres, semestres o año. Y una variable numerica que nos ayude a resumir los datos.

Ilustración 15 Lineplot

Lineplot

```
In [19]: plt.figure(figsize=(10, 6))
         plt.plot(YearGroupBy["Year"], YearGroupBy["Value"], marker='o', linestyle='-')
         plt.xlabel('Year')
         plt.ylabel('Value')
         plt.title('Line Chart of Value over Time')

Out[19]: Text(0.5, 1.0, 'Line Chart of Value over Time')
```



Como último paso, se precisaría de un arreglo en el dataset original para que sea totalmente funcional a la hora de aplicar un futuro modelo de Machine Learning, para eso, aplicaremos técnicas de transformación de datos. Entre ellas creación de dummies variables, selección de variables al utilizar y separación del dataset en conjuntos de datos más pequeños.

La creación de dummies se recomienda cuando quisiéramos ingresar una variable categórica como factor a un modelo de ML, el problema sería que dicho modelo es incapaz de interpretar letras, por ende, se debe cambiar dichas características entre números, de esta manera, el modelo podrá leer con satisfacción los datos.

Ilustración 16 Dummies variables

Preparación del dataset para un futuro algoritmo de Machine Learning

Creacion de dummies para las variables categoricas

In [36]:
```
dummies = pd.get_dummies(df['Measure'], drop_first=True)

df_ml = pd.concat([df, dummies], axis=1)
df_ml.head()
```

Out[36]:

	Measure	Country	Citizenship	Year	Value	Departures	Net
0	Arrivals	Oceania	New Zealand Citizen	1979	11817.0	0	0
1	Arrivals	Oceania	Australian Citizen	1979	4436.0	0	0
2	Arrivals	Oceania	Total All Citizenships	1979	19965.0	0	0
3	Arrivals	Antarctica	New Zealand Citizen	1979	10.0	0	0
4	Arrivals	Antarctica	Australian Citizen	1979	0.0	0	0

El siguiente punto es la selección de variables, se recomienda usar las columnas más importantes a la hora de entrenar un modelo de Machine Learning, esto ya que daría mejor resultados, el problema de agregar la máxima cantidad de variables es que podría entorpecer o confundir al modelo. En esta ocasión solo utilizaremos el año y el tipo de vuelo.

Por último, el dataset es dividido entre variables independientes (X) y variables dependientes (y), que esta ultima el dato que nos gustaría predecir en base a las variables independientes. A la vez, estos mismos dos conjuntos de datos se vuelven a dividir, dando un total de cuatro datasets. Estos cuatro serán utilizados a lo largo de modelos de Machine Learning.

Ilustración 17 Selección de variables y separación

Preparación del dataset para entrenar y evaluar el modelo

In [35]:
```
from sklearn.model_selection import train_test_split

X = df_ml[["Year", "Departures", "Net" ]]
y = df_ml["Value"]
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.30, random_state = 101)
```

Datasets listos

In [37]:
```
X_train.head()
```

Out[37]:

	Year	Departures	Net
49743	2000	0	1
19360	1987	1	0
40772	1996	0	1
85812	2016	0	1
50068	2000	0	1

In [38]:
```
y_train.head()
```

Out[38]:
```
49743   -2.0
19360    0.0
40772   -2.0
85812   -1.0
50068    0.0
Name: Value, dtype: float64
```

TIPO DE DATA MINING

Dentro del Data Mining hay diferentes tipos de especializaciones, una de esas técnicas es la predictiva, que se enfoca en predecir resultados futuros basados en patrones y tendencias. Se basa en algoritmos estadísticos y de aprendizaje automático para identificar relaciones y patrones en los datos que pueden utilizarse para hacer predicciones sobre eventos futuros.

Dado a las características del conjunto de datos a trabajar, junto con los posibles problemas que podría solucionar, el data minig predictivo es el que le podemos sacar más provecho por sus atributos para pronosticar eventos futuros con datos del pasado. Con el fin de tomar acciones más precisas y que nos brinden del mayor beneficio posible.

Gracias a las técnicas de la minería de datos predictivas podrías predecir futuros niveles de oferta y demanda basada al aumento o disminución de personas. A la vez, crear presupuestos quinquenales para generar estabilidad económica mediante políticas monetaria y fiscal acorde a las necesidades del momento.

CONCLUSIONES

- Satisfactoriamente se obtuvieron varias medidas estadísticas que lograron brindar más información sobre el conjunto de datos, esto siendo la base de otras futuras etapas para un análisis predictivo más formal.
- Se obtuvo un análisis profundo sobre los patrones y tendencias del tema a tratar mediante la ejecución de gráficos representativos donde intuitivamente se ejemplifican dichas características.
- Se implementaron técnicas de manipulación de datos, incluyendo la sustitución de valores y filtros, con el objetivo de extraer el máximo conocimiento posible del conjunto de datos, estableciendo así las bases sólidas para el análisis subsiguiente.

RECOMENDACIONES

- Alentamos al lector a investigar a profundo la rama del Data Mining y todos sus beneficios, esto ya que es una herramienta sumamente efectiva para la investigación y resolución de problemas.
- Encarecidamente incentivamos a todos los organismos, ya sean públicos o privados, a la inversión para la implementación de Data Mining dentro de ellas, esto ya que está más que demostrado la efectivad para la toma de decisiones de alto impacto.

BIBLIOGRAFÍA

Amazon. (s.f.). *AWS*. Obtenido de AWS: https://aws.amazon.com/es/what-is/data-analytics/#:~:text=El%20an%C3%A1lisis%20de%20datos%20puede,mejorar%20exponencialmente%20el%20rendimiento%20empresarial.

Leyva, J. L. (24 de 05 de 2023). *Universidad Veracruzana*. Obtenido de Universidad Veracruzana: https://www.uv.mx/prensa/general/destacan-papel-de-la-estadistica-en-la-toma-de-decisiones-sociales/

ONU. (21 de 9 de 2016). *ONU*. Obtenido de ONU: https://refugeesmigrants.un.org/es/la-migraci%C3%B3n-es-beneficiosa-para-todos-si-se-gestiona-correctamente#:~:text=La%20migraci%C3%B3n%20puede%20propiciar%20un,para%20los%20pa%C3%ADses%20de%20origen.