Visualización de evolución en el tiempo

Daniel Giovanni Rodriguez Coral

Guillermo Augusto Duran Boneth

Juan Sebastián Torres Perdomo

Fundación Universitaria Internacional de La Rioja

Especialización en Inteligencia Artificial

Ingeniero Javier Diaz Diaz

Colombia

Diciembre, 2024

Contenido

Introducción

En la actualidad, la visualización de los datos representa un papel fundamental a la hora de querer transmitir información y que ésta genere un valor agregado para el público objetivo. Desde la toma de decisiones a nivel empresarial, como también en diferentes análisis en contextos como el científico, la capacidad de poder transformar y procesar los datos en visualizaciones sumamente claras y que representen adecuadamente el objetivo que se quiere transmitir ha demostrado ser esencial.

Para alcanzar el proceso de visualización de datos, es necesario establecer de antemano una serie de pasos o criterios que faciliten una correcta manipulación de los datos. Entre estos pasos, se destacan la normalización y el procesamiento de los datos.

Otro de los puntos más importantes es definir adecuadamente las herramientas con las que se llevará a cabo el procesamiento de los datos y las visualizaciones, en este sentido, las herramientas utilizadas en el presente trabajo para el procesamiento de los datos son Python y OpenRefine, y para la construcción de las visualizaciones PowerBI.

En este documento se explora de forma detallada e integral el proceso de transformación de datos, desde la limpieza inicial hasta la visualización final, y se destaca como es que estas herramientas se complementan para poder generar valor agregado en los resultados.

Al presentar también un enfoque simple, bajo un ejemplo y resultados, este trabajo pretende evidenciar el impacto que puede llegar a tener el análisis y presentación de los datos en los contextos en que sea pertinentes aplicarlos.

Objetivos

Objetivo General

Desarrollar un dashboard interactivo para la visualización y análisis de datos, aplicando una limpieza, transformación y representación gráfica, con la finalidad de demostrar habilidades en el uso de herramientas como Power BI, Python y OpenRefine.

Objetivos Específicos

Limpiar y transformar los datos de ventas ficticias para garantizar la consistencia y precisión de la información.

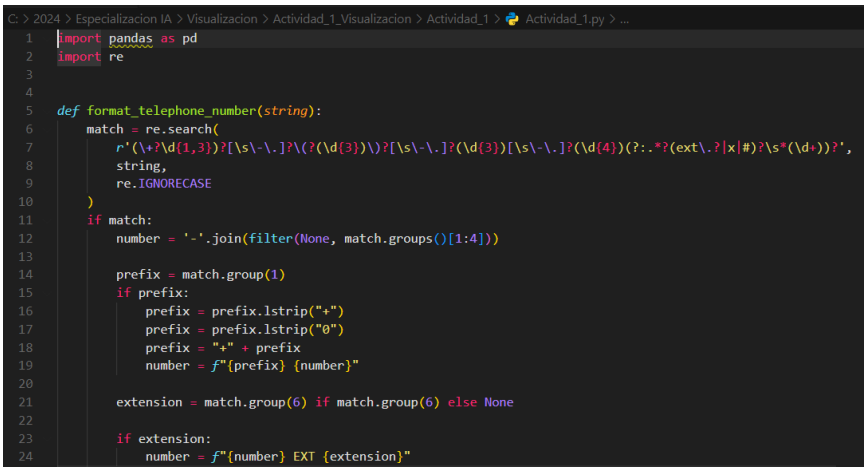
Crear una visualización interactiva que permita identificar algunos hallazgos o patrones importantes dentro del conjunto de datos estudiado.

Refinación de los datos

Refinación en Python

Se realiza una limpieza previa del conjunto de datos suministrado, usando el lenguaje Python, a continuación, algunas evidencias del trabajo realizado:

1. Se crea una función que permite formatear el número de teléfono del cliente en formato Prefijo + Número de Teléfono + Extensión.



```python
import pandas as pd
import re


def format_telephone_number(string):
    match = re.search(
        r'(\+?\d{1,3})?[\s\-\.]?\(?(\d{3})\)?[\s\-\.]?(\d{3})[\s\-\.]?(\d{4})(?:.*?(ext\.?|x|#)?\s*(\d+))?',
        string,
        re.IGNORECASE
    )
    if match:
        number = '-'.join(filter(None, match.groups()[1:4]))

        prefix = match.group(1)
        if prefix:
            prefix = prefix.lstrip("+")
            prefix = prefix.lstrip("0")
            prefix = "+" + prefix
            number = f"{prefix} {number}"

        extension = match.group(6) if match.group(6) else None

        if extension:
            number = f"{number} EXT {extension}"
```

2. Se realiza la limpieza de caracteres extraños en todos los campos haciendo uso de expresiones regulares, se eliminan espacios al comienzo y al final de la palabra. También se realiza el ajuste correspondiente para el campo Fecha_Venta para que el formato sea YYYY-MM-DD

```python
def format_telephone_number(string):
        return number
    return None


ruta_del_archivo = './Initial_load/datos_ventas.csv'

df = pd.read_csv(ruta_del_archivo)

df['ID_Venta'] = df['ID_Venta'].astype(str).str.strip()

df['Producto'] = df['Producto'].astype(str).apply(lambda text: re.sub(r'[^a-zA-Z0-9 ]', '', text)).str.upper()

df['Ciudad'] = df['Ciudad'].astype(str).apply(lambda text: re.sub(r'[^a-zA-Z0-9 ]', '', text)).str.upper()

df['Categoria'] = df['Categoria'].astype(str).apply(lambda text: re.sub(r'[^a-zA-Z0-9 ]', '', text)).str.upper()

df['Precio_Unitario'] = pd.to_numeric(df['Precio_Unitario'], errors='coerce')

df['Cantidad'] = pd.to_numeric(df['Cantidad'], errors='coerce')

df['Fecha_Venta'] = df['Fecha_Venta'].str.replace(r'^[^\w_]+|[^\w_]+$', '', regex=True)
df['Fecha_Venta'] = df['Fecha_Venta'].str.replace(r'^_+|_+$', '', regex=True)
df['Fecha_Venta'] = pd.to_datetime(df['Fecha_Venta'], format='%Y-%m-%d').dt.date
```

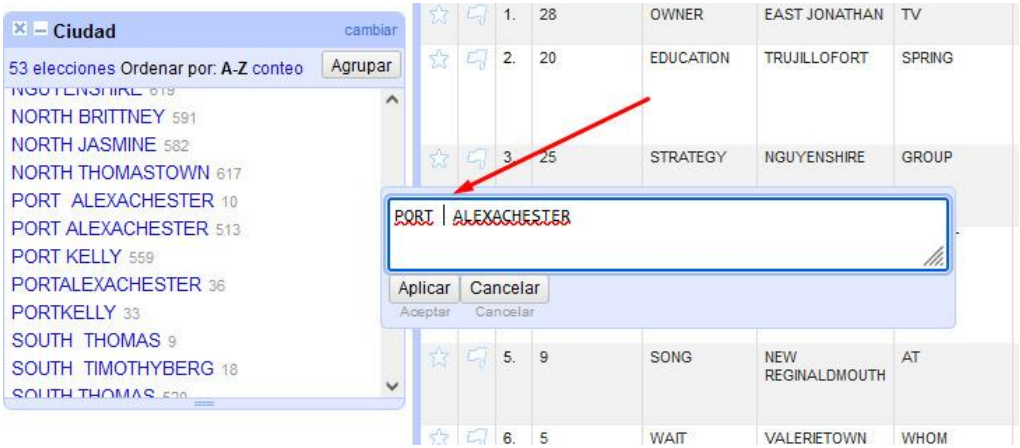
```
df['Email'] = df['Email'].str.replace(r'^[^\w_]+|[^\w_]+$', '', regex=True)
df['Email'] = df['Email'].str.replace(r'^_+|_+$', '', regex=True)

df[['Telefono_Formateado']] = df['Telefono'].apply(lambda x: pd.Series(format_telephone_number(x)))

df['Direccion_Formateada'] = df['Direccion'].str.replace('\n', ' ', regex=False) ##La dirección tenía saldo de línea

df['Metodo_Pago'] = df['Metodo_Pago'].astype(str).apply(lambda text: re.sub(r'[^a-zA-Z0-9 ]', '', text)).str.upper().s

df['Estado'] = df['Estado'].astype(str).apply(lambda text: re.sub(r'[^a-zA-Z0-9 ]', '', text)).str.upper().str.strip()

df['Comentario'] = df['Comentario'].astype(str).apply(lambda text: re.sub(r'[^a-zA-Z0-9 ]', '', text)).str.upper().str

df['Descuento'] = pd.to_numeric(df['Descuento'], errors='coerce')

df['Total'] = df['Precio_Unitario'] * df['Cantidad']

df['total_con_descuento'] = df['Total'] * (1 - df['Descuento'] / 100)

print(df)

df.to_csv("./Result/datos_ventas_clean.csv", index= False)
```

Refinación en OpenRefine

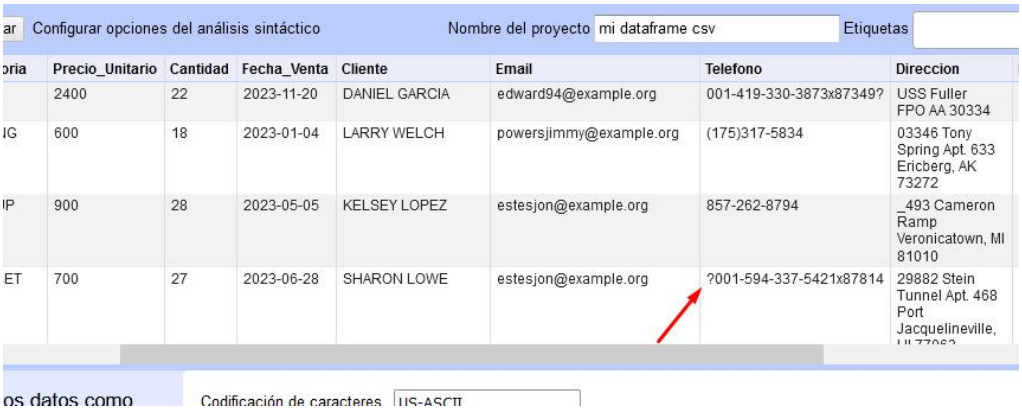
Se continúa con la organización, limpieza y creación de columnas de faltantes del conjunto de datos en OpenRefine, a continuación, algunas evidencias del trabajo realizado:

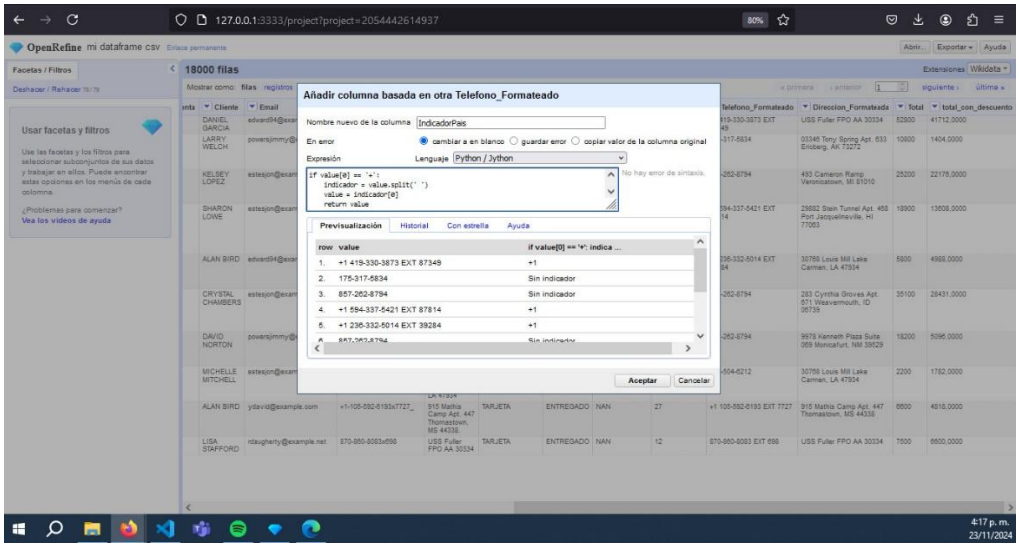
1. Se hace una revisión del formato de los emails, se verifica la falta del símbolo de arroba @, en algunos de los registros, así como espacios en blanco y caracteres que no corresponden.

2. Se estandariza los nombres de las ciudades, tomando un nombre como referencia e igualando los similares.



3. Se estandariza los nombres de teléfono, eliminando caracteres que no pertenecen a dichos datos y separando los indicativos de los número telefónicos, usando código del lenguaje Python.





Añadir columna basada en otra Telefono_Formateado

Nombre nuevo de la columna: Extension

En error: cambiar a en blanco / guardar error / copiar valor de la columna original

Expresión — Lenguaje: Python / Jython

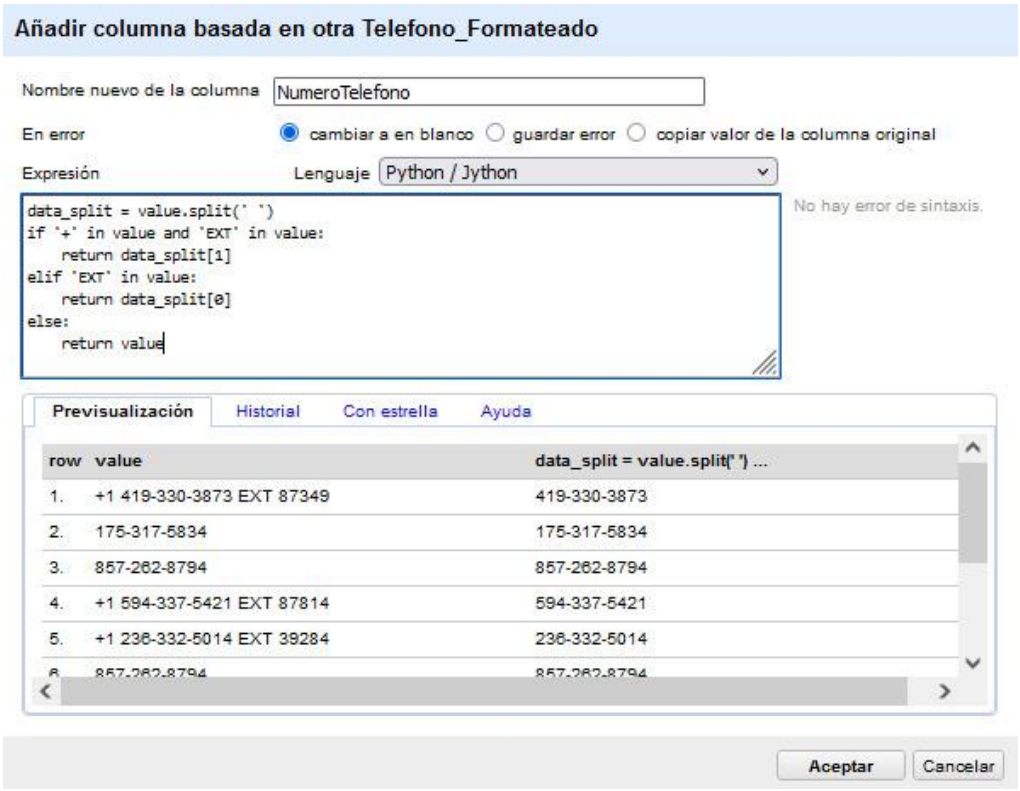
```
    value = data_ext[-1]
    return value
else:
    return "Sin extension"
```

No hay error de sintaxis.

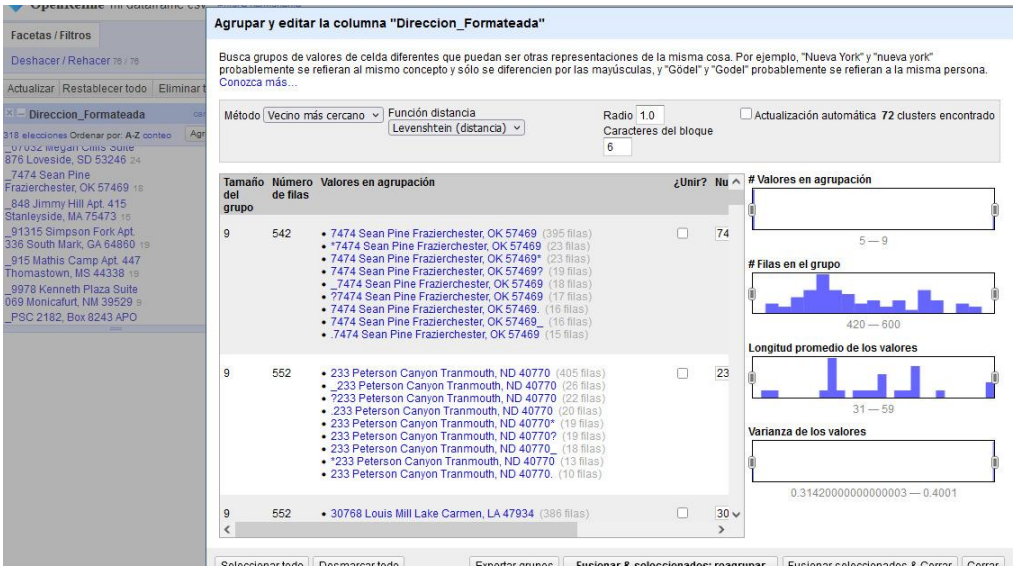
Previsualización | Historial | Con estrella | Ayuda

row	value	if 'EXT' in value: data_ex ...
1.	+1 419-330-3873 EXT 87349	87349
2.	175-317-5834	Sin extension
3.	857-262-8794	Sin extension
4.	+1 594-337-5421 EXT 87814	87814
5.	+1 236-332-5014 EXT 39284	39284
6.	857-262-8794	Sin extension

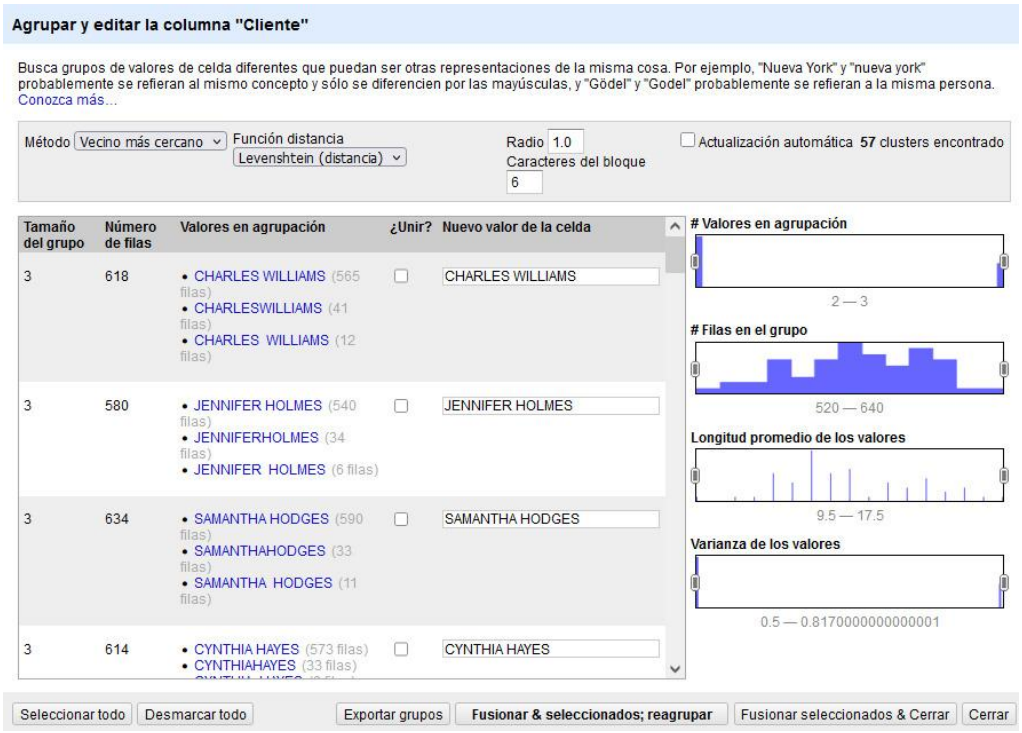
Aceptar | Cancelar



4. Se estandariza los datos de direcciones, eliminando caracteres especiales y borrando espacios.



5. Se estandariza los datos de la columna Cliente, transformando los nombres similares, que solamente se diferencien por cosas mínimas, por ejemplo, espacios o mayúsculas y minúsculas.



Visualización en Power Bi

Junto a este documento se entrega el archivo con el conjunto de datos final y el dashboard realizado con los datos limpios y organizados en PowerBi. A continuación, una captura de pantalla del dashboard y la explicación de sus elementos:

1. Filtros
    - **Fecha Venta:** Permite al usuario seleccionar la fecha que quiere validar, teniendo en cuenta siempre la última fecha de venta registrada en la fuente de datos. Por defecto, el dashboard siempre mostrará la información a la última fecha registrada de acuerdo al filtro aplicado por el usuario. Por ejemplo, si el usuario selecciona en el filtro Fecha Venta la fecha 30/11/2023, si la última fecha de venta registrada fue 20/11/2023, el reporte automáticamente mostrará la información hasta esta fecha.
    - **Categoría:** Permite al usuario seleccionar la categoría de los productos.
2. Un KPI, mostrando la última fecha de compra registrada en la fuente de datos, se incluyó para que el usuario pueda analizar la información de manera más sencilla. Siempre mostrará la última fecha de venta de acuerdo al filtro aplicado por el usuario.

3. Un botón “Reset” que, permite al usuario reiniciar al estado inicial el dashboard.
4. Un Botón “Datos de Clientes “que, al dar clic, el usuario puede visualizar la información de contacto de los clientes.

CLIENTE	CIUDAD	DIRECCION	INDICADOR	TELÉFONO	EXTENSION	EMAIL
ADAM MUELLER	ANDREWSIDE	025 BILLY ROW LISASIDE, AR 96338	+1	557-543-1910	SIN EXTENSION	ydavid@example.com
ADAM MUELLER	ANDREWSIDE	03346 TONY SPRING APT. 633 ERICBERG, AK 73272	SIN INDICADOR	339-100-4282	04920	rbrown@example.org
ADAM MUELLER	ANDREWSIDE	03346 TONY SPRING APT. 633 ERICBERG, AK 73272	SIN INDICADOR	612-821-1834	900	krystalmcdaniel@example.org
ADAM MUELLER	ANDREWSIDE	0827 JILL CAPE SUITE 299 ANGIEPORT, DC 15542	SIN INDICADOR	630-469-6433	SIN EXTENSION	karaware@example.net
ADAM MUELLER	ANDREWSIDE	2644 GONZALEZ MILLS APT. 856 PORT FRANK, IN 25816	+1	096-236-8217	SIN EXTENSION	brianboyd@example.org
ADAM MUELLER	ANDREWSIDE	2644 GONZALEZ MILLS APT. 856 PORT FRANK, IN 25816	SIN INDICADOR	491-474-2586	SIN EXTENSION	estesjon@example.org
ADAM MUELLER	ANDREWSIDE	29340 MURPHY PARKWAY NORTH ROBERT, KS 42070	SIN INDICADOR	602-238-8997	526	hornebeth@example.com
ADAM MUELLER	ANDREWSIDE	29882 STEIN TUNNEL APT. 468 PORT JACQUELINEVILLE, HI 77063	SIN INDICADOR	369-774-6572	373	
ADAM MUELLER	ANDREWSIDE	30768 LOUIS MILL LAKE CARMEN, LA 47934	+1	890-231-3672	1535	cwilliams@example.com
ADAM MUELLER	ANDREWSIDE	493 CAMERON RAMP VERONICATOWN, MI 81010	SIN INDICADOR	101-988-9703	8562	hornebeth@example.com

5. Un KPI que, muestra el valor total de Ventas acumuladas hasta la fecha filtrada o no por el usuario.
6. Un KPI que, muestra el número de ventas acumuladas hasta la fecha filtrada o no por el usuario
7. Un KPI que, muestra el valor total de ventas para la última fecha de ventas registrada de acuerdo al filtro aplicado por el usuario.
8. Un gráfico de barras y líneas que permiten al usuario realizar un seguimiento de las ventas diarias realizadas en los últimos 15 días de ventas registradas, de acuerdo al filtro aplicado por el usuario. Las barras representan la cantidad de ventas realizadas y la línea representa el valor total de Ventas.

   Si el usuario ubica el mouse sobre la gráfica en una fecha, automáticamente aparecerá una ventana emergente con la información correspondiente a esa fecha:

   - En primer lugar, aparecerá la fecha en formato dd/mm/yyyy.
   - En color azul, aparecerá la cantidad de Ventas realizadas en la fecha.
   - En color rojo, aparecerá el Total de Ventas realizadas en la fecha.

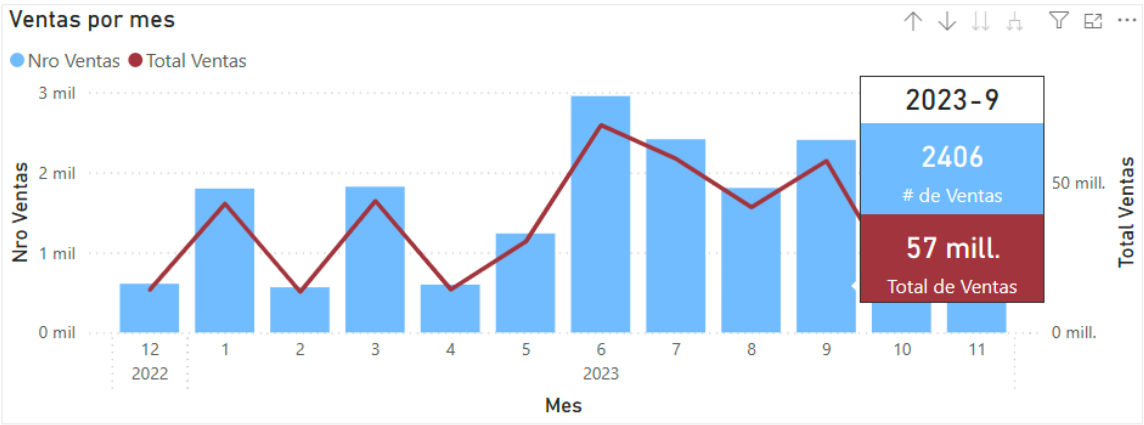
9. Un gráfico de barras que muestran el Total de Ventas realizadas por Producto ordenado descendentemente para la última fecha de venta, de acuerdo al filtro aplicado por el usuario. Si el usuario coloca el mouse sobre las barras aparecerá una ventana emergente con una tabla que contiene el top de los clientes que más han comprado el producto:

10. Un gráfico circular que permite analizar el número de ventas realizadas por método de pago (Tarjeta, Efectivo) para la última fecha de venta de acuerdo al filtro aplicado por el usuario
11. Un gráfico de barras y líneas que permite al usuario realizar un seguimiento de las ventas mensuales de acuerdo al filtro aplicado por el usuario. Las barras

representan la cantidad de ventas mensuales realizadas y la línea representa el valor total de Ventas mensuales.

Si el usuario coloca el mouse sobre la gráfica en una fecha, automáticamente aparecerá una ventana emergente con la información correspondiente a esa fecha:

- En primer lugar, aparecerá la fecha en formato yyyy-mm.
- En color azul, aparecerá la cantidad de Ventas realizadas en el mes.
- En color rojo, aparecerá el Total de Ventas realizadas en el mes.



12. Un gráfico circular que permite analizar el número de ventas realizadas de acuerdo al estado (Entregado o En Camino), para la última fecha de venta, de acuerdo al filtro aplicado por el usuario

Conclusiones

La limpieza y transformación de datos es fundamental para obtener insights precisos y confiables. Al garantizar la consistencia y precisión de la información, hemos podido identificar patrones y tendencias relevantes en los datos de ventas ficticias, lo que a su vez ha permitido tomar decisiones más informadas.

La visualización de datos es una herramienta poderosa para comunicar hallazgos complejos de manera clara y efectiva. El dashboard interactivo desarrollado en Power BI ha permitido explorar los datos de múltiples formas, revelando insights que de otra manera podrían haber pasado desapercibidos. Esto demuestra la importancia de combinar técnicas de limpieza de datos con herramientas de visualización para obtener un análisis completo.

La práctica con datos artificiales es una excelente manera de desarrollar habilidades en limpieza de datos y creación de visualizaciones. Al trabajar con un conjunto de datos generado artificialmente, hemos podido experimentar con diferentes técnicas y herramientas, lo que nos ha permitido fortalecer nuestras competencias en este campo.