

**Proyecto Integrado (EA3). Presentación del proyecto y dashboard  
descriptivo**

**Análisis de Transacciones Con Tarjetas de Crédito**

**EULICER ZAPATA ORREGO**

**DAWIN DE JESUS SALAZAR OVIEDO**

**DOCENTE:**

**ANDRES FELIPE CALLEJAS**

**INSTITUCIÓN UNIVERSITARIA DIGITAL DE ANTIOQUIA  
FACULTAD DE INGENIERÍA Y CIENCIAS AGROPECUARIAS  
INGENIERÍA DE SOFTWARE Y DATOS.**

**Medellín, Colombia**

**Diciembre 6 2025**

## Contenido

Resumen (Abstract).....	3
Objetivo General.....	4
Objetivos Específicos.....	4
Links.....	4
Definición del problema / Caso de Uso.....	5
Metodología empleada.....	6
1. Descarga del dataset desde Kaggle.....	6
3. Carga del dataset a un DataFrame en pandas.....	7
4. Creación de la base de datos SQLite.....	8
5. Inserción del dataset en SQLite.....	8
6. Exportación de datos desde SQLite a CSV.....	8
7. Limpieza de los datos.....	9
8. Visualización descriptiva de las variables.....	16
8. Tablero de Análisis de Transacciones por Género:.....	23
Resultados.....	24
Presentación de Resultados y Visualizaciones Clave:.....	24
Bibliografía.....	31

## Resumen (Abstract)

El presente proyecto surge de la necesidad de comprender cómo se comportan los usuarios de tarjetas de crédito a partir de la información disponible en registros

reales de transacciones. Este tipo de análisis es útil para comercios y analistas que requieren identificar zonas de mayor actividad, identificación de género, y características del uso cotidiano de los servicios financieros. Para este estudio se utilizó el *Credit Card Transactions Dataset*, obtenido de la plataforma pública Kaggle y descargado en noviembre de 2025. El conjunto de datos incluye información sobre el monto de cada transacción, la fecha en que fue realizada, la categoría del comercio, el nombre del establecimiento, la ubicación geográfica y género, asociada a cada operación, datos suficientes para realizar un análisis descriptivo del comportamiento transaccional.

El propósito del proyecto es analizar estas transacciones, identificación de los lugares donde compran los usuarios, los tipos de comercios que frecuentan, los montos que suelen gastar y género que mayor gasta. El estudio del dataset se realizará mediante un análisis exploratorio, empleando técnicas descriptivas y visualizaciones que permiten identificar los datos y posibles anomalías presentes en el conjunto de datos.

Los resultados permitirán una mejor comprensión del uso de tarjetas de crédito en diferentes contextos de consumo, ofreciendo información relevante para la toma de decisiones y el entendimiento del comportamiento financiero de los usuarios.

**Palabras clave:** transacciones, Kaggle, tarjetas de crédito, análisis exploratorio, comercios.

## Objetivo General

Analizar las transacciones bancarias en función del género y la ubicación geográfica de los titulares.

## Objetivos Específicos.

- Seleccionar y comprender el dataset utilizado, incluyendo la fuente de datos de Kaggle.
- Identificar y describir las variables relevantes del conjunto de datos.
- Diseñar y construir una base de datos en SQLite que permita almacenar y consultar las transacciones.
- Implementar un proceso de limpieza y transformación de datos para estandarizar variables críticas y generar nuevas características temporales y geográficas, asegurando la integridad y calidad del dataset para su posterior análisis.
- Construir visualizaciones que muestran patrones de gasto, permitiendo visualizar tendencias transaccionales diferenciadas por género y ubicación geográfica.
- Documentar el proceso y elaborar el documento en formato APA.

## Links

**Link git:** [https://github.com/eulicerzapata/Proyecto\\_Integrador5.git](https://github.com/eulicerzapata/Proyecto_Integrador5.git)

**Diagrama de Gantt:**

<https://docs.google.com/spreadsheets/d/1I1Phu9ODemJZHmGOwFeAwHI1w4TL-Qs5J2ONluQouhg/edit?usp=sharing>

**DashBoard:** <https://proyectointegrador5git-p93mwqqeqjdqwwvjfgev3.streamlit.app/>

## Definición del problema / Caso de Uso

El uso de tarjetas de crédito genera diariamente un gran volumen de transacciones que contienen información clave sobre el comportamiento de compra de los usuarios. Sin embargo, muchas instituciones, comercios y analistas carecen de una comprensión clara sobre cómo, dónde y en qué categorías de comercio se realizan estas transacciones, lo que dificulta realizar un análisis de

género y ubicación geográfica, zonas de mayor actividad comercial. Esta falta de conocimiento limita la capacidad de tomar decisiones informadas relacionadas con estrategias comerciales, segmentación de clientes, tendencias relevantes en el consumo.

El presente proyecto aborda esta necesidad mediante el análisis del *Credit Card Transactions Dataset*, un conjunto de datos público obtenido de la plataforma Kaggle y descargado en noviembre de 2025. Este dataset contiene información sobre montos transaccionados, fechas de las operaciones, categorías de comercio, nombres de establecimientos y datos de ubicación geográfica, género, asociados a cada registro. Estos atributos permiten realizar un análisis descriptivo del comportamiento transaccional sin necesidad de técnicas predictivas o modelos avanzados.

El caso de uso se centra en examinar cómo se distribuyen las transacciones según ubicación y tipo de comercio, qué patrones se observan en los montos de gasto y qué comportamientos podrían considerarse inusuales dentro del conjunto de datos. Con ello se busca generar una comprensión clara y fundamentada del consumo mediante tarjetas de crédito, útil para diferentes actores interesados en el análisis de datos financieros.

## Metodología empleada

- Selección y comprensión del dataset
- Identificación y análisis preliminar de variables relevantes
- Limpieza de los datos del dataset.
- Enriquecimiento de los datos.

- Análisis descriptivo de los datos.
- Construcción de la base de datos en SQLite
- Proceso de carga: dataset → SQLite
- Exportación del archivo CSV desde SQLite
- Documentación del proceso según normas APA
- Elaboración del Diagrama de Gantt y WBS del proyecto

La metodología de este proyecto se basó en un proceso de análisis exploratorio de datos y en la implementación de un flujo que permitió la ingestión, transformación y almacenamiento del dataset *Credit Card Transactions Dataset* dentro de una base de datos SQLite.

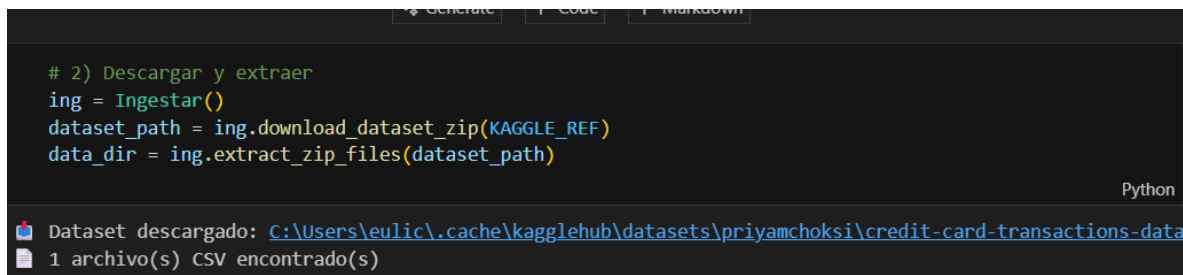
A continuación, se describen detalladamente las etapas ejecutadas, basadas en los scripts desarrollados en el repositorio del proyecto.

## 1. Descarga del dataset desde Kaggle

Para obtener el dataset se utilizó la clase *Ingestar*, ubicada en `src/proyecto_integrador/ingestar.py`.

El método `download_dataset_zip()` emplea la librería **kagglehub**, que permite descargar automáticamente el dataset usando la referencia oficial:

`priyamchoksi/credit-card-transactions-dataset`



```
# 2) Descargar y extraer
ing = Ingestar()
dataset_path = ing.download_dataset_zip(KAGGLE_REF)
data_dir = ing.extract_zip_files(dataset_path)
```

Dataset descargado: `C:\Users\eulic\.cache\kagglehub\datasets\priyamchoksi\credit-card-transactions-data`  
 1 archivo(s) CSV encontrado(s)

## 2. Identificación y extracción del dataset

Posteriormente, el método `extract_zip_files()` revisa el contenido descargado y:

- Si encuentra un archivo `.zip`, lo extrae en un subdirectorío.
- Si encuentra uno o más `.csv`, retorna directamente esa carpeta.
- Si encuentra `.xlsx`, los procesa de igual forma.

De este modo, el sistema se adapta a distintos formatos sin intervención manual.

## 3. Carga del dataset a un DataFrame en pandas

Para unificar los archivos descargados, se utilizó:

load\_dataset\_as\_dataframe()

Este método:

- Lee todos los archivos .csv o .xlsx dentro del directorio.
- Concatena su contenido en un único **DataFrame**
- Detecta problemas de lectura (codificación, separadores)
- Garantiza que el dataset esté completamente cargado antes de insertarse en SQLite

```
✓ Ejecutar ingesta de datos - Carga a SQLite

1 ▶ Run python src/proyecto_integrador/load_to_sqlite.py
11 =====
12 CARGA DE DATOS A SQLITE
13 =====
14
15 Descargando dataset desde Kaggle...
16 Downloading from https://www.kaggle.com/api/v1/datasets/download/priyamchoksi/credit-card-transactions-dataset?dataset_version_number=1..
17
18 0%|          | 0.00/145M [00:00<?, ?B/s]
19 6%|█         | 9.00M/145M [00:00<00:01, 86.7MB/s]
20 12%|█        | 18.0M/145M [00:00<00:01, 73.1MB/s]
21 18%|█▌       | 26.0M/145M [00:00<00:02, 55.1MB/s]
22 24%|█▌      | 35.0M/145M [00:00<00:01, 59.9MB/s]
23 30%|█▌      | 44.0M/145M [00:00<00:01, 68.3MB/s]
24 35%|█▌      | 51.0M/145M [00:00<00:01, 65.9MB/s]
25 43%|█▌      | 62.0M/145M [00:00<00:01, 77.5MB/s]
26 48%|█▌      | 70.0M/145M [00:01<00:01, 77.9MB/s]
27 55%|█▌      | 80.0M/145M [00:01<00:01, 61.8MB/s]
28 60%|█▌      | 87.0M/145M [00:01<00:00, 62.0MB/s]
29 65%|█▌      | 94.0M/145M [00:01<00:00, 64.1MB/s]
30 71%|█▌      | 103M/145M [00:01<00:00, 71.2MB/s]
31 76%|█▌      | 111M/145M [00:01<00:00, 71.3MB/s]
32 82%|█▌      | 119M/145M [00:01<00:00, 74.2MB/s]
33 89%|█▌      | 129M/145M [00:01<00:00, 81.2MB/s]
34 96%|█▌      | 139M/145M [00:02<00:00, 87.0MB/s]
35 100%|███████| 145M/145M [00:02<00:00, 73.0MB/s]
36 Extracting files...
37 Dataset descargado: /home/runner/.cache/kagglehub/datasets/priyamchoksi/credit-card-transactions-dataset/versions/1
38
39 Procesando archivos descargados...
40 1 archivo(s) CSV encontrado(s)
```

Este paso es fundamental para validar la integridad del dataset.

## 4. Creación de la base de datos SQLite

El script de carga (load\_to\_sqlite.py) ejecuta el proceso de creación de la base de datos:

- Se define la ruta a db/proyecto.db.
- Se crea el directorio db/ si no existe.
- Se establece conexión usando sqlite3.connect().

Aunque no se ejecutó una sentencia manual CREATE TABLE, el almacenamiento se realiza mediante `pandas.to_sql`, que crea automáticamente la tabla **transacciones** con las columnas del DataFrame.

## 5. Inserción del dataset en SQLite

El método encargado de este proceso es `insertar_datos(conn, df)`:

```
df.to_sql('transacciones', conn, if_exists='replace', index=False)
```

Este comando:

- Crea la tabla transacciones si no existe
- Reemplaza el contenido en cada ejecución
- Inserta todas las filas del DataFrame en una sola operación

Se imprime además el total de registros cargados, lo que permite validar el proceso de ingestión.

```

38
39  Procesando archivos descargados...
40  1 archivo(s) CSV encontrado(s)
41
42  Cargando datos a /home/runner/work/Proyecto_Integrador5/Proyecto_Integrador5/db/proyecto.db...
43  Leyendo /home/runner/.cache/kagglehub/datasets/priyamchoksi/credit-card-transactions-dataset/versions/1/credit_card_transactions.csv...
44  1,296,675 registros insertados en la base de datos
45
46  =====
47  PROCESO COMPLETADO EXITOSAMENTE
48  =====
49  Base de datos: /home/runner/work/Proyecto_Integrador5/Proyecto_Integrador5/db/proyecto.db
50  Total de registros: 1,296,675
```

## 6. Exportación de datos desde SQLite a CSV

Para cumplir con el flujo solicitado por la universidad:

**dataset** → **SQLite** → **CSV**,

se desarrolló el script `export_to_csv.py`.

Este script:

1. Conecta a `proyecto.db`.
2. Ejecuta la consulta SQL: `SELECT * FROM transacciones`
3. Carga los datos en un DataFrame con `pandas.read_sql_query()`.
4. Exporta el resultado a: `db/export.csv`
5. Muestra el número total de registros exportados.

```
1  ▶ Run python src/proyecto_integrador/export_to_csv.py
11 =====
12 EXPORTACIÓN DE SQLITE A CSV
13 =====
14 1,296,675 registros exportados a /home/runner/work/Proyecto_Integrador5/Proyecto_Integrador5/db/export.csv
15
16 =====
17 PROCESO COMPLETADO EXITOSAMENTE
18 =====
19
20 Flujo completado:
21 Kaggle Dataset
22 SQLite: /home/runner/work/Proyecto_Integrador5/Proyecto_Integrador5/db/proyecto.db
23 CSV: /home/runner/work/Proyecto_Integrador5/Proyecto_Integrador5/db/export.csv
```

De esta manera se evidencia el flujo completo exigido por el Proyecto Integrador V

## 7. Limpieza de los datos

### 1. Selección de variables para carga, limpieza y transformación.

```
36 [PASO 1] Seleccionando columnas necesarias para el analisis
37 - Filas totales: 1,296,675
38 - Columnas seleccionadas: 13 de 24 originales
39 - Filas aceptadas: 1,296,675
```

```
177     - trans_num
178     - trans_date_trans_time
179     - gender
180     - city
181     - state
182     - lat
183     - long
184     - city_pop
185     - merchant
186     - category
187     - amt
188     - merch_lat
189     - merch_long
190     - state_name
191     - anio
192     - mes
193     - dia
194     - hora
```

## 2. Eliminar transacciones duplicadas.

```
41 [PASO 2] Eliminando transacciones duplicadas
42     - Criterio: Numero de transaccion (trans_num)
43     - Filas duplicadas encontradas: 0
44     - Filas eliminadas: 0
45     - Filas aceptadas: 1,296,675
```

Se utiliza el método `drop_duplicates` enfocado en la columna `trans_num` (ID de transacción). La técnica consiste en conservar la primera ocurrencia (`keep='first'`) y descartar cualquier fila posterior que repita ese mismo ID.

### 3. Limpiar la columna de género del titular.

```
47 [PASO 3] Limpiando columna 'gender' (genero del titular)
48 - Transformacion: Convertir a mayusculas y eliminar espacios
49 - Valores nulos encontrados: 0
50 - Valores transformados a mayusculas: 0
51 - Valores validos: ['M', 'F']
52 - Valores invalidos encontrados: 0
53 - Filas eliminadas: 0
54 - Filas aceptadas: 1,296,675
55
```

Estandariza el texto eliminando espacios en blanco (strip) y convirtiendo todo a mayúsculas (upper) para asegurar consistencia (ej: 'm ' se vuelve 'M').

### 4. Limpiar las columnas de ubicación

#### 4.1 Columna ciudad

```
56 [PASO 4] Limpiando columnas de ubicacion
57
58 [4.1] Procesando columna 'city' (ciudad del titular)
59 - Transformacion: Capitalizar primera letra de cada palabra (Title Case)
60 - Valores nulos encontrados: 0
61 - Valores transformados (capitalizados): 0
62 - Filas eliminadas: 0
63 - Filas aceptadas: 1,296,675
64 - Ciudades unicas: 894
--
```

Aplica "Title Case" (str.title()), lo que convierte la primera letra de cada palabra en mayúscula (ej: "new york" -> "New York").

#### 4.2 columna estado

```
66 [4.2] Procesando columna 'state' (estado del titular - abreviatura)
67 - Transformacion: Convertir a MAYUSCULAS (para mapeo posterior)
68 - Valores nulos encontrados: 0
69 - Valores transformados a MAYUSCULAS: 0
70 - Filas eliminadas: 0
71 - Filas aceptadas: 1,296,675
72 - Estados unicos: 51
```

Fuerza mayúsculas sostenidas (str.upper()) para estandarizar abreviaturas.

### 4.3 Columna latitud.

```
74      [4.3] Procesando columna 'lat' (latitud del titular)
75      - Transformacion: Convertir a numerico
76      - Valores nulos encontrados: 0
77      - Filas eliminadas: 0
78      - Filas aceptadas: 1,296,675
```

Usa Coerción de Tipos (pd.to\_numeric con errors='coerce'). Esto intenta convertir valores a números y, si falla (por ejemplo, si hay texto "abc"), lo convierte en NaN (nulo) para luego eliminarlo.

### 4.4 Columna longitud

```
80      [4.4] Procesando columna 'long' (longitud del titular)
81      - Transformacion: Convertir a numerico
82      - Valores nulos encontrados: 0
83      - Filas eliminadas: 0
84      - Filas aceptadas: 1,296,675
```

Usa Coerción de Tipos (pd.to\_numeric con errors='coerce'). Esto intenta convertir valores a números y, si falla (por ejemplo, si hay texto "abc"), lo convierte en NaN (nulo) para luego eliminarlo.

### 4.5 Columna población de la ciudad

```
86      [4.5] Procesando columna 'city_pop' (poblacion de la ciudad)
87      - Transformacion: Convertir a numerico y eliminar valores <= 0
88      - Valores nulos encontrados: 0
89      - Filas eliminadas: 0
90      - Filas aceptadas: 1,296,675
```

Además de convertir a número, aplica una Regla de Negocio (población > 0) para eliminar datos ilógicos.

## 5. Enriquecimiento de la columna estado.

```
92 [PASO 5] Enriqueciendo con nombres completos de estados
93   - Nueva columna: 'state_name'
94   - Estados unicos (abreviaturas): 51
95   - Mapeo: Abreviatura (ej: NY) -> Nombre completo (ej: New York)
96   - Filas aceptadas: 1,296,675
97
98   Ejemplos de mapeo:
99       NC -> North Carolina
100      WA -> Washington
101      ID -> Idaho
102      MT -> Montana
103      VA -> Virginia
```

Utiliza un diccionario (STATE\_NAMES) como tabla de búsqueda. La función map() busca cada abreviatura (clave) y la reemplaza por su valor asociado (nombre completo).

## 6. Limpiando columnas de comercio.

```
105 [PASO 6] Limpiando columnas de comercio
106
107 [6.1] Procesando columna 'merchant' (nombre del comercio)
108 - Transformacion: Eliminar espacios en blanco al inicio/final
109 - Valores nulos encontrados: 0
110 - Valores con espacios en blanco eliminados: 0
111 - Filas eliminadas: 0
112 - Filas aceptadas: 1,296,675
113 - Comercios unicos: 693
114
115 [6.2] Procesando columna 'category' (categoria del comercio)
116 - Transformacion: Convertir a minusculas y eliminar espacios
117 - Valores nulos encontrados: 0
118 - Valores transformados a minusculas: 0
119 - Filas eliminadas: 0
120 - Filas aceptadas: 1,296,675
121 - Categorias unicas: 14
122
123 [6.3] Procesando columna 'merch_lat' (latitud del comercio)
124 - Transformacion: Convertir a numerico
125 - Valores nulos encontrados: 0
126 - Filas eliminadas: 0
127 - Filas aceptadas: 1,296,675
128
129 [6.4] Procesando columna 'merch_long' (longitud del comercio)
130 - Transformacion: Convertir a numerico
131 - Valores nulos encontrados: 0
132 - Filas eliminadas: 0
133 - Filas aceptadas: 1,296,675
```

Merchant: Solo elimina espacios vacíos al inicio y final (strip).

Category: Convierte a minúsculas (lower) para agrupar categorías que podrían estar escritas diferente (ej: "Food" y "food" se vuelven lo mismo).

## 7. Limpiando columna monto de la transacción

```
135 [PASO 7] Limpiando columna 'amt' (monto de la transaccion)
136     - Transformacion: Convertir a numerico y eliminar valores <= 0
137     - Valores nulos encontrados: 0
138     - Filas eliminadas: 0
139     - Filas aceptadas: 1,296,675
140     - Estadisticas de monto:
141         Minimo: $1.00
142         Maximo: $28948.90
143         Promedio: $70.35
144         Mediana: $47.52
145
```

Convierte la columna en numérica y aplica un filtro lógico para asegurar que el monto sea estrictamente positivo ( $> 0$ ).

## 8. Enrichiendo columnas temporales

```
146 [PASO 8] Enrichiendo con columnas temporales
147     - Nuevas columnas: anio, mes, dia, hora
148     - Transformacion: Extraer componentes de fecha/hora
149     - Filas eliminadas: 0
150     - Filas aceptadas: 1,296,675
151     - Rango de fechas:
152         Desde: 2019-01-01 00:00:18
153         Hasta: 2020-06-21 12:13:37
154     - Anos unicos en el dataset: 2
155
```

Parsing: Convierte la columna de texto `trans_date_trans_time` a objetos `datetime` reales de Python/Pandas (`pd.to_datetime`).

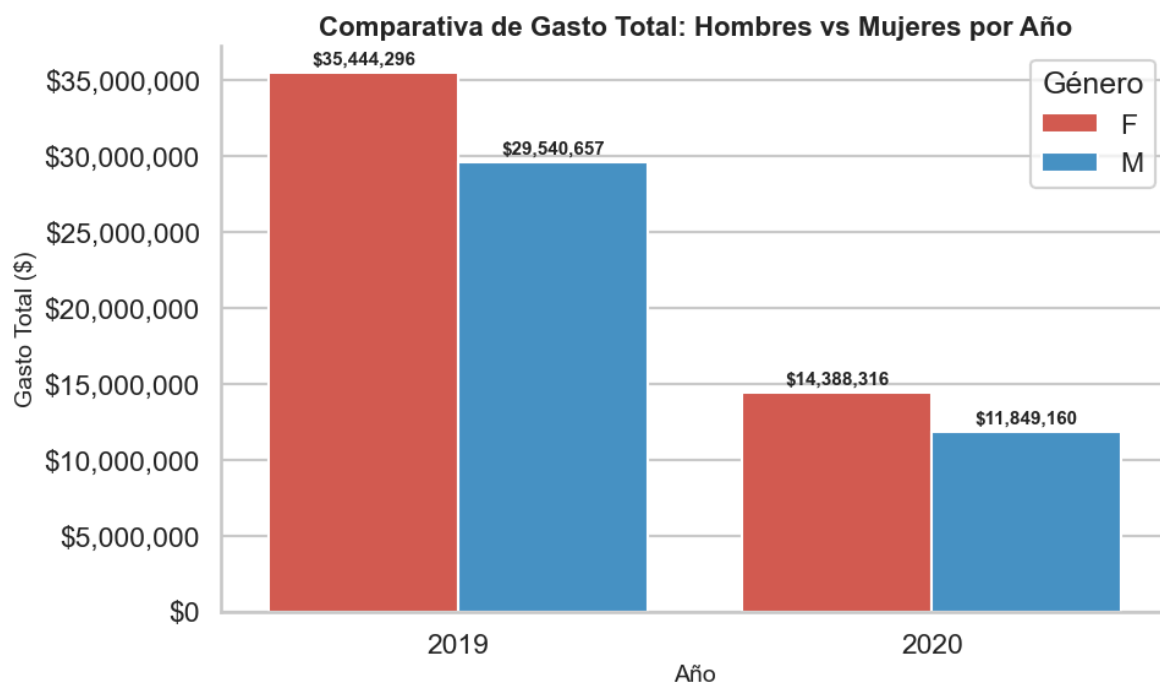
Extracción: Descompone la fecha en sus partes atómicas (year, month, day) creando nuevas columnas numéricas.

## 9. Resultados de la limpieza y enriquecimiento.

```
160  DIMENSIONES:
161    - Filas originales:    1,296,675
162    - Filas finales:      1,296,675
163    - Filas eliminadas:    0
164    - Columnas originales: 24
165    - Columnas finales:    18
166
167  VALORES NULOS/INVALIDOS PROCESADOS:
168    - Total de filas eliminadas por nulos/invalidos: 0
169
170  COLUMNAS DERIVADAS AGREGADAS:
171    - anio
172    - mes
173    - dia
174    - hora
```

## 8. Visualización descriptiva de las variables.

### 1. Gastos de género discriminados por año.

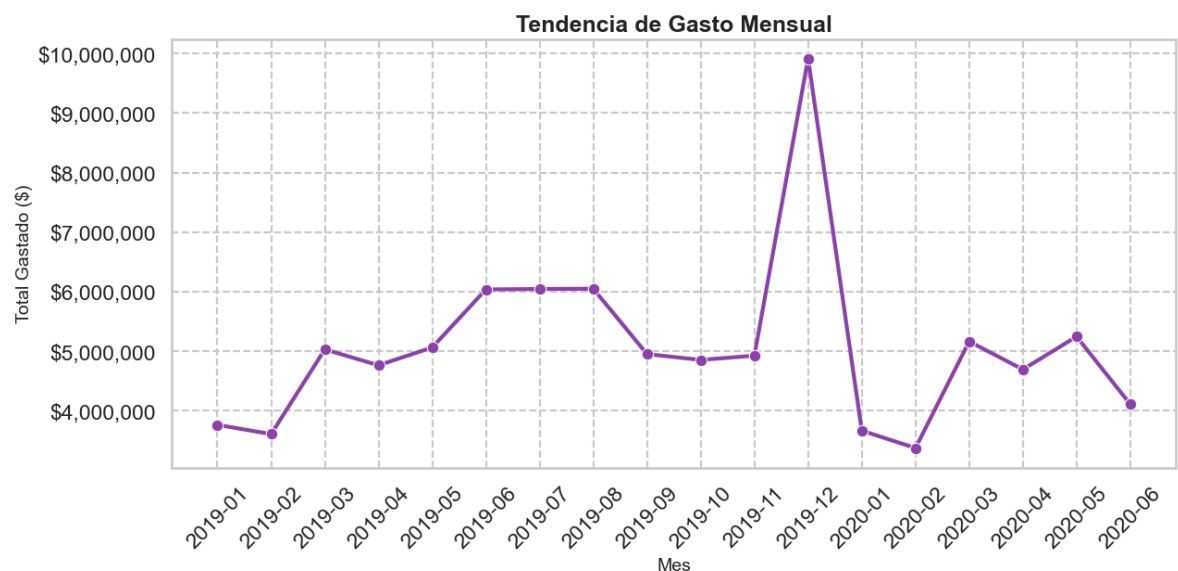


### Gráfico de Barras Agrupadas: Gasto Total por Género y Año

Este gráfico compara el volumen total de transacciones desglosado por Año y Género.

Se observa que el segmento femenino (F) supera consistentemente al masculino (M) en gasto total durante ambos años. En 2019, las mujeres gastaron \$35.4M frente a \$29.5M de los hombres, manteniendo esta superioridad en 2020 (\$14.3M vs \$11.8M). Esto indica que las mujeres son el principal motor económico en este conjunto de datos.

## 2. Transacciones mensuales por categoría.



### Gráfico de Línea: Tendencia de Gasto Mensual

Este gráfico muestra la evolución del monto total de transacciones (eje Y) a lo largo de los meses (eje X) entre 2019 y 2020.

El comportamiento del gasto es relativamente estable durante gran parte de 2019, oscilando entre \$4 y \$6 millones. El dato más relevante es el pico drástico en diciembre de 2019, donde el consumo se dispara hasta casi \$10 millones, marcando una clara estacionalidad de fin de año. Posteriormente, el inicio de 2020 registra una caída abrupta a los niveles mínimos de la serie (cerca de \$3.5M), seguida de una recuperación moderada sin alcanzar los niveles previos.

### 3. gasto mensual por categoría (top 5)

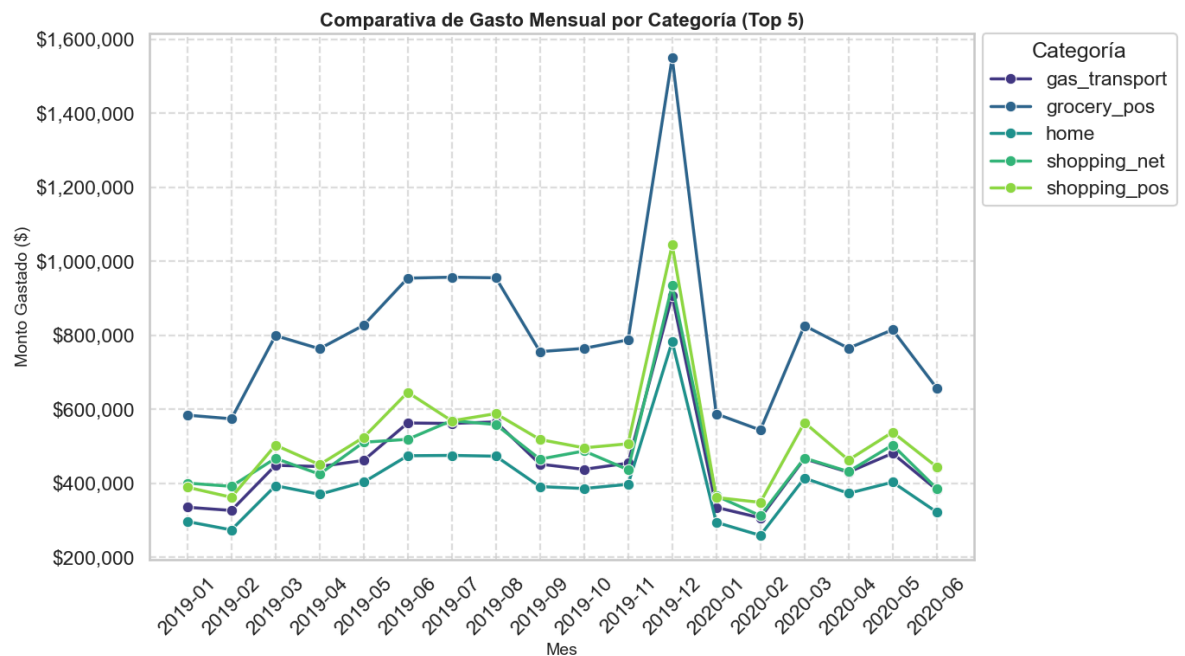


Gráfico de Líneas Múltiples: Comparativa de Gasto Mensual por Categoría (Top 5)

Este gráfico desglosa la evolución mensual del gasto en las 5 categorías principales: grocery\_pos (Supermercado), shopping\_pos (Compras físicas), home (Hogar), shopping\_net (Compras online) y gas\_transport (Gasolina/Transporte).

La categoría grocery\_pos domina consistentemente el gasto durante todo el periodo, manteniéndose siempre por encima de las demás. Se observa un patrón estacional muy marcado en diciembre de 2019, donde todas las categorías experimentan un pico simultáneo, siendo grocery\_pos la que alcanza el máximo histórico (~\$1.5M). Las demás categorías (shopping, home, gas) siguen tendencias muy similares entre sí, moviéndose en bloque con valores entre \$400k y \$600k, lo que sugiere una correlación en el comportamiento de consumo general más allá del rubro específico.

4. Distribución geográfica de los gastos (top 10).

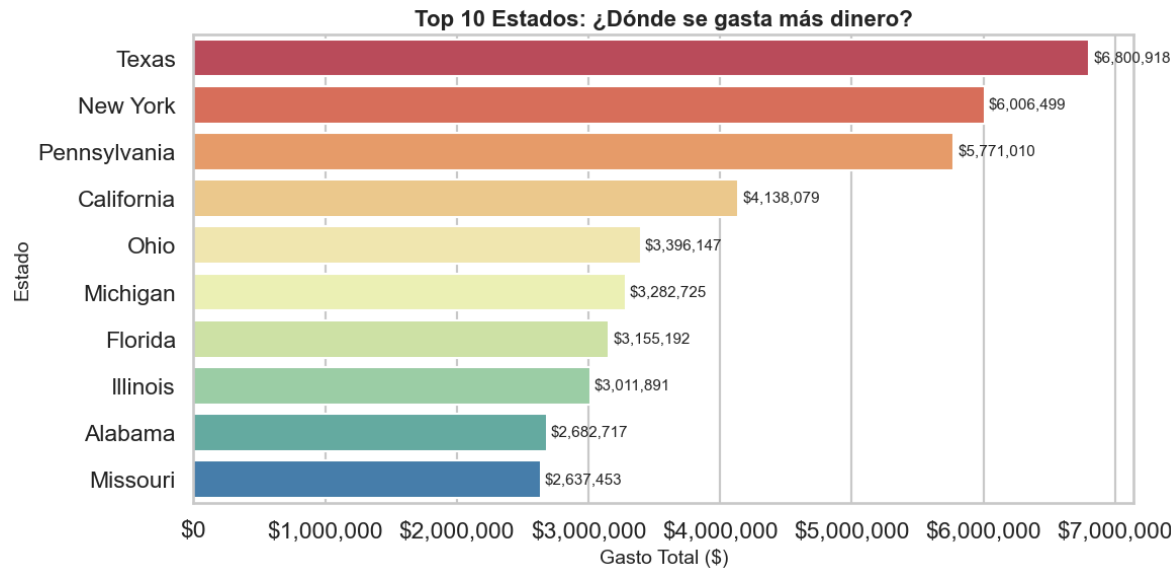


Gráfico de Barras Horizontales: Top 10 Estados por Gasto Total

Este gráfico clasifica los 10 estados con mayor volumen acumulado de transacciones, permitiendo identificar los mercados geográficos más importantes.

Texas lidera el ranking con un gasto total de \$6.8 millones, seguido de cerca por New York (\$6.0M) y Pennsylvania (\$5.7M). Estos tres estados conforman el podio económico, concentrando una parte muy significativa del volumen total. Existe un salto notable entre el tercer lugar (Pennsylvania) y el cuarto (California, \$4.1M), lo que indica que el consumo en este dataset está particularmente polarizado en los tres primeros estados. El resto del Top 10 (Ohio, Michigan, Florida, etc.) muestra un descenso gradual, manteniéndose en el rango de los \$2.6M a \$3.4M.

## 5. Promedio mensual de transacciones discriminadas por género.

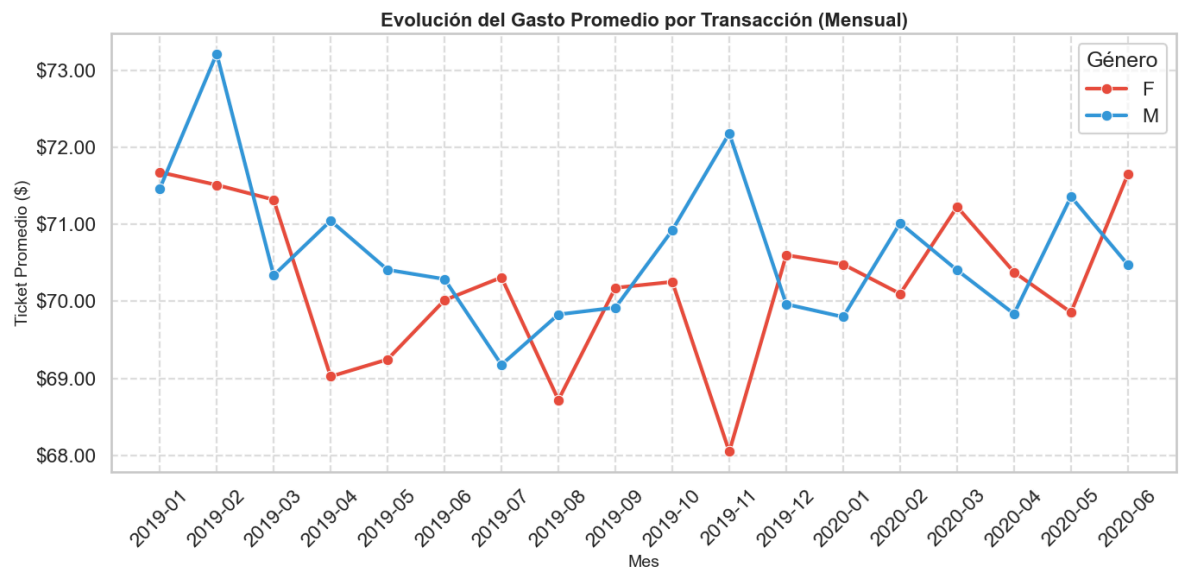
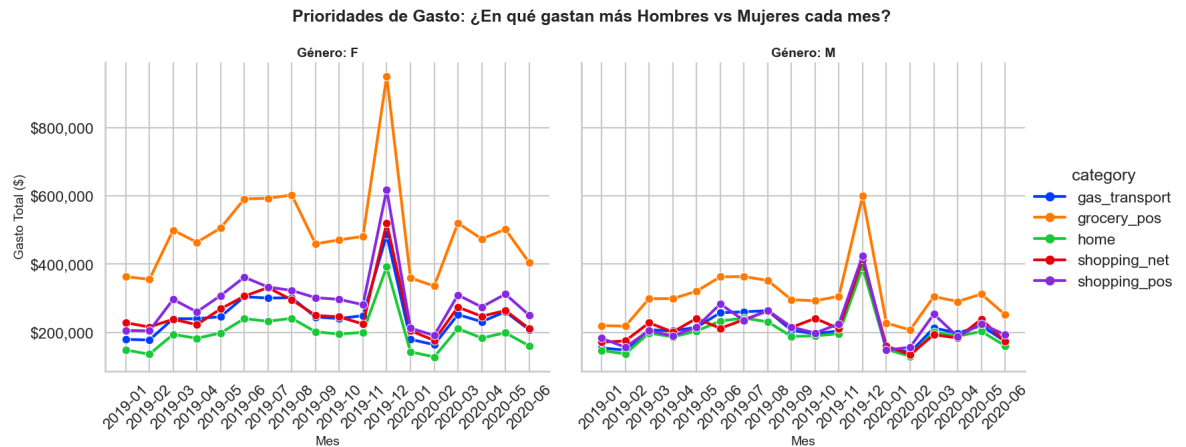


Gráfico de Línea: Evolución del Ticket Promedio Mensual por Género

Este gráfico compara el valor promedio de cada transacción (Ticket Promedio) entre hombres (azul) y mujeres (rojo) a lo largo del tiempo.

Se observa una volatilidad constante en el ticket promedio de ambos géneros, oscilando generalmente entre \$69 y \$72. Sin embargo, destaca una caída abrupta en el ticket promedio de las mujeres en Noviembre de 2019, alcanzando su punto mínimo anual (~\$68).

## 6. Top 5: categorías más compradas según el género.



### Gráfico de Líneas Facetado (Comparativo)

Variables: Mes (Tiempo), Gasto Total (amt), Categoría (Top 5) y Género.

Interpretación: Ambos géneros siguen patrones de consumo casi idénticos, con la categoría Supermercado (grocery\_pos) liderando siempre el gasto, seguida de compras físicas y transporte. La principal diferencia es el volumen: las mujeres gastan significativamente más en todas las categorías, especialmente durante el pico estacional de Diciembre, donde el consumo se dispara universalmente.

7. proporciones de gastos entre hombres y mujeres por top 10 estados:

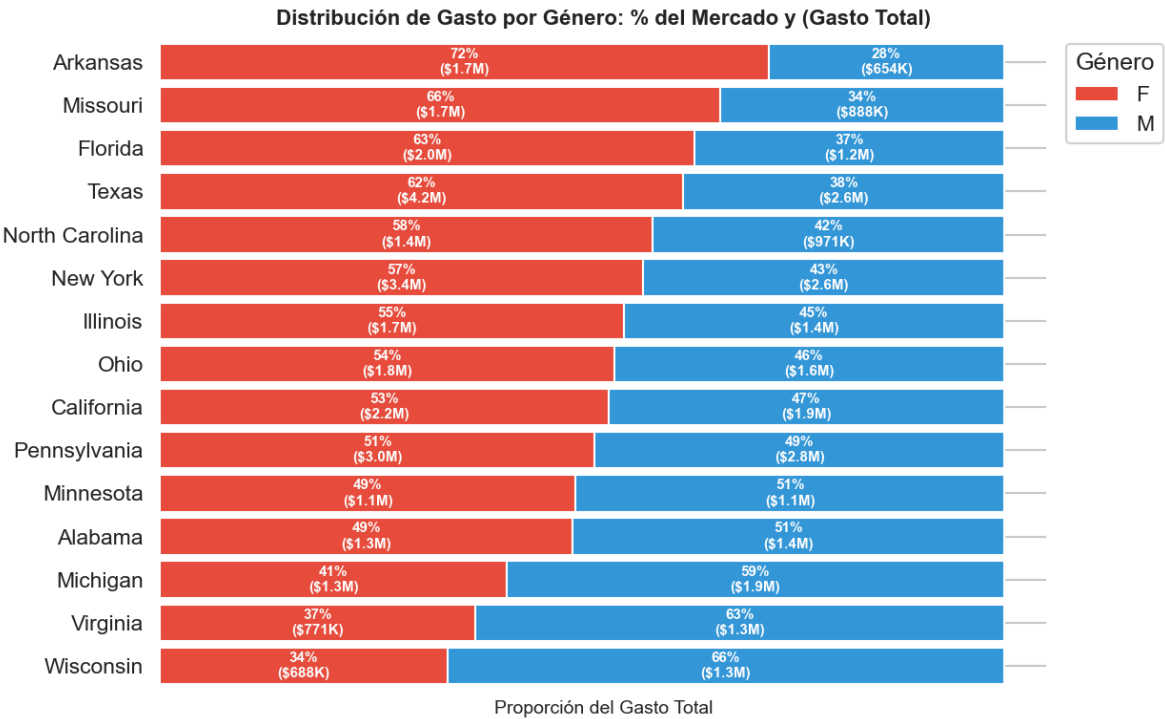
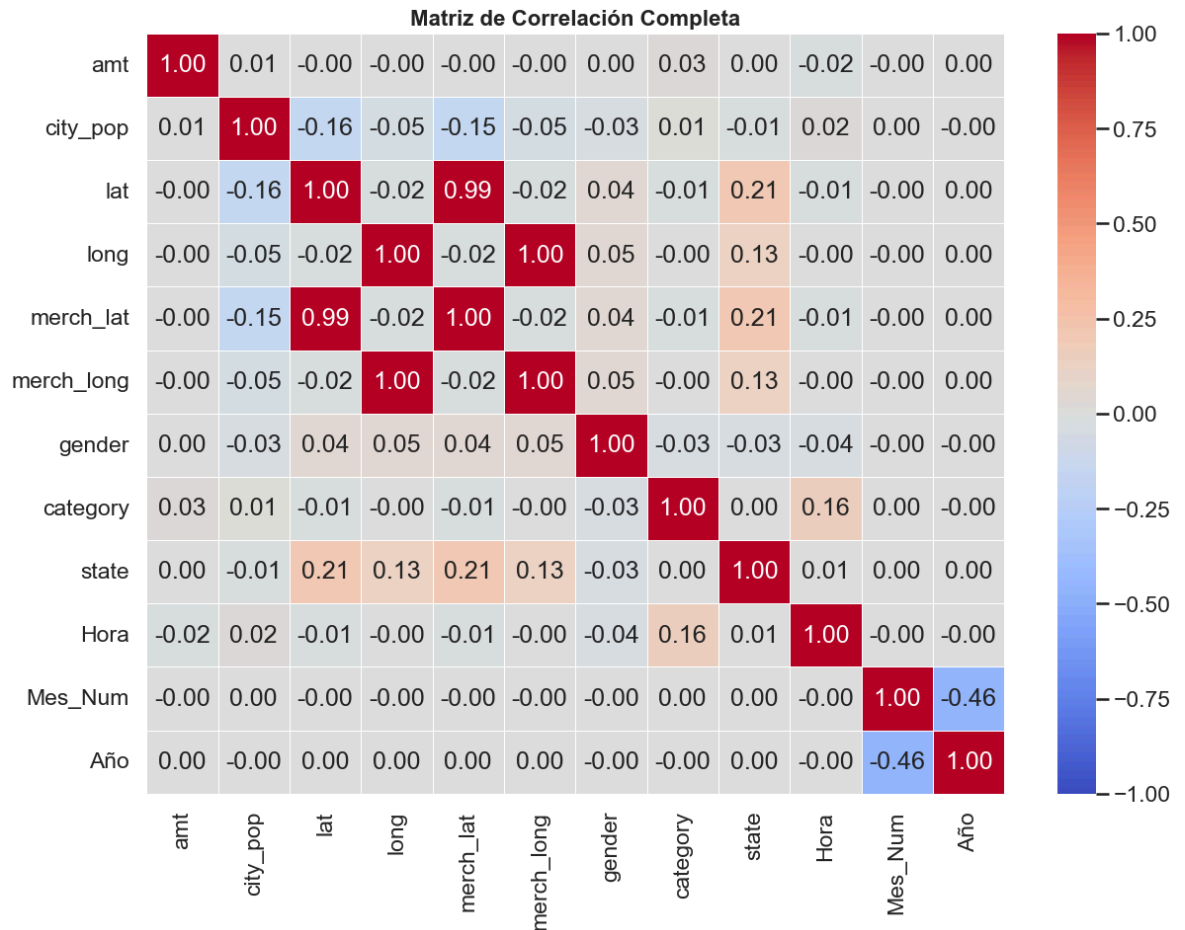


Gráfico de Barras Apiladas (100%): Distribución de Gasto por Género

Variables: Estado, Proporción de Gasto (%), Gasto Total (\$) y Género.

Existe una clara disparidad regional en el comportamiento de gasto. Estados como Arkansas y Missouri están fuertemente dominados por el consumo femenino (72% y 66% respectivamente). En contraste, estados como Wisconsin, Virginia y Michigan muestran la tendencia opuesta, con una mayoría de gasto masculino (llegando al 66% en Wisconsin). Los grandes mercados como New York y California presentan un perfil más equilibrado, aunque con una ligera inclinación hacia el segmento femenino.

## 8. Matriz de correlación entre variables.



### Mapa de Calor: Matriz de Correlación

Variables: Todas las variables numéricas y categóricas del dataset.

Interpretación: El análisis de correlación revela que no existen relaciones lineales fuertes entre la mayoría de las variables del negocio.

El monto de la transacción (amt) es prácticamente independiente de factores como la población de la ciudad (city\_pop), la ubicación (lat, long) o el género (gender), con coeficientes cercanos a 0.00.

Las únicas correlaciones fuertes (cercanas a 1.00) son triviales y esperadas: la latitud del usuario (lat) con la del comercio (merch\_lat), lo cual confirma que las personas compran cerca de donde viven.

Existe una leve correlación negativa (-0.46) entre Mes\_Num y Año, lo cual es un artefacto de los datos (probablemente porque tenemos datos de finales de 2019 y principios de 2020).

## 8. Tablero de Análisis de Transacciones por Género:

## 9. Documentación y control de versiones

El archivo README.md documenta:

- la descripción del proyecto
- los scripts del proceso
- instrucciones de ejecución
- dependencias instaladas
- especificación del flujo de ingestión



The screenshot shows a README file with a section titled "6. Estructura del proyecto". It displays a tree-like structure of the project files and folders, with comments explaining the purpose of each item.

```
piv_2025_2_2/
├── README.md          # Este archivo
├── setup.py           # Configuración del paquete Python
├── src/
│   └── proyecto_integrador/
│       ├── __init__.py    # Exportaciones del módulo
│       ├── ingestar.py    # Clase para descarga y procesamiento desde Kaggle
│       ├── limpiar_datos.py # Limpieza y enriquecimiento de datos
│       ├── load_to_sqlite.py # Script de carga a base de datos SQLite
│       └── export_to_csv.py # Script de exportación desde SQLite a CSV
├── notebooks/
│   └── proyecto_integrador.ipynb # Notebook con análisis exploratorio
├── docs/
│   └── imagenes/          # Imágenes de gráficos de los análisis exploratorios
├── db/
│   └── proyecto.db        # Base de datos SQLite (generada)
├── data/
│   └── dataset_enriquecido.csv # Dataset limpio y enriquecido (generado)
└── csv/
    └── export.csv         # Archivo CSV exportado (generado)
```

El archivo .gitignore omite las carpetas db y csv por tamaño.

## 10. Planificación del proyecto

El archivo diagrama de gantt.xlsx contiene:

- actividades del proyecto
- responsable (Dawin – Eulicer)
- sprints
- fechas
- porcentaje de avance
- gráficos y dashboard

Este archivo cumple la exigencia de planificación con metodología ágil.

## Resultados

El análisis se estructuró en tres dimensiones principales para responder a las preguntas de negocio:

**Análisis Temporal:** Evaluación de tendencias de gasto a lo largo de los años, estacionalidad mensual y patrones horarios de compra.

**Análisis Demográfico (Género):** Comparación directa entre comportamientos de compra de hombres y mujeres, tanto en volumen de transacciones como en montos y categorías preferidas.

**Análisis Geográfico:** Identificación de los estados con mayor actividad económica y visualización de la densidad de transacciones a nivel de ciudad.

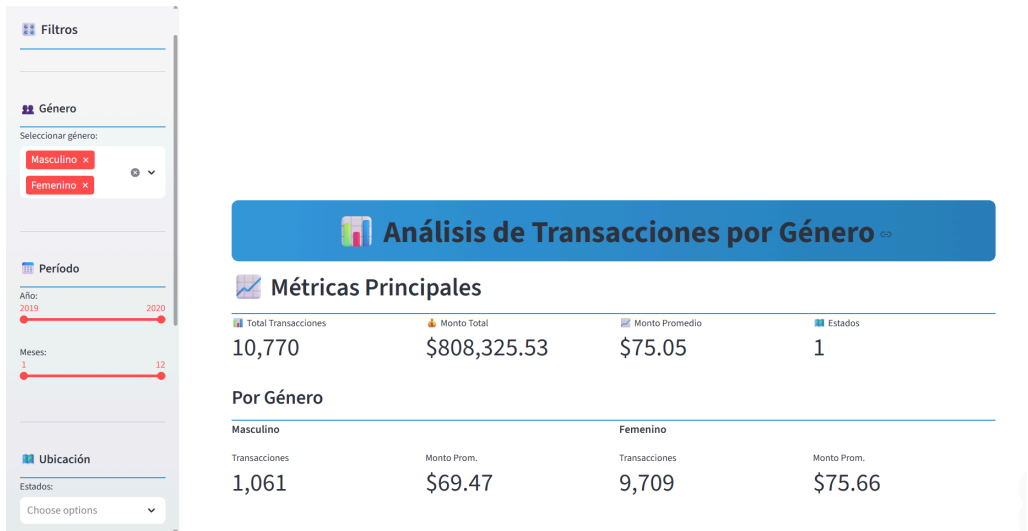
El dashboard permite a los usuarios interactuar con estas dimensiones mediante filtros dinámicos de año, mes, estado, categoría y género, recalculando todas las métricas en tiempo real.

## Presentación de Resultados y Visualizaciones Clave:

A continuación, se detallan las visualizaciones más importantes que deben incluirse en la presentación.

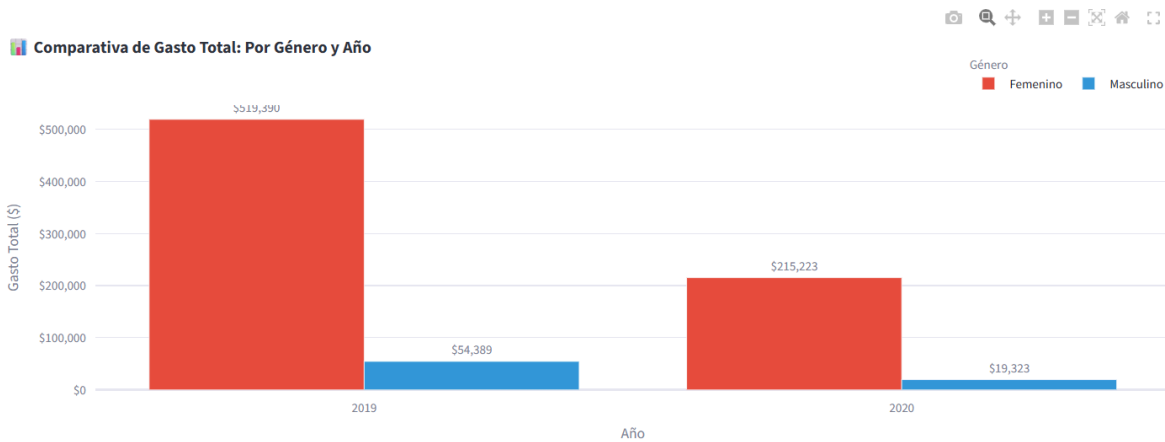
### A. Vista General y KPIs

**Descripción:** Muestra las métricas de alto nivel que resumen el estado total de las transacciones filtradas.



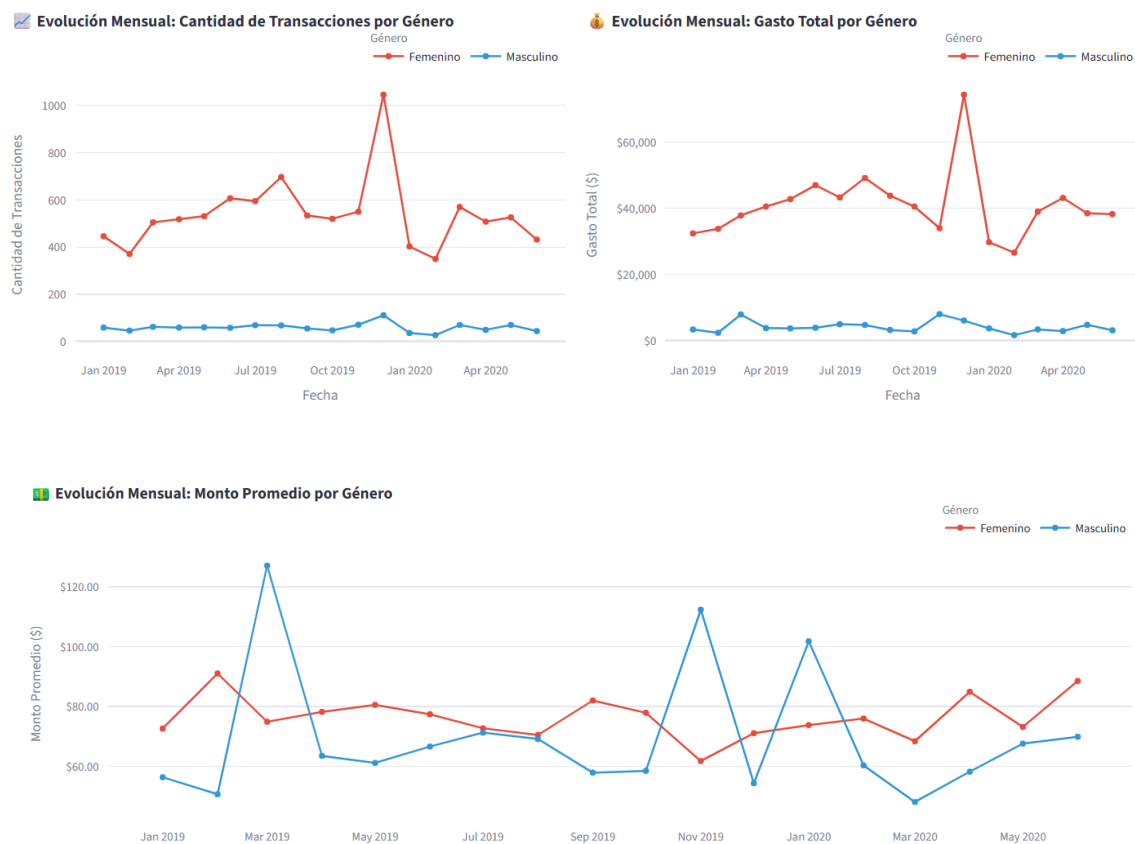
### B. Evolución Temporal del Gasto

**Descripción:** Gráfico de líneas que compara el gasto total por género a través de los años.



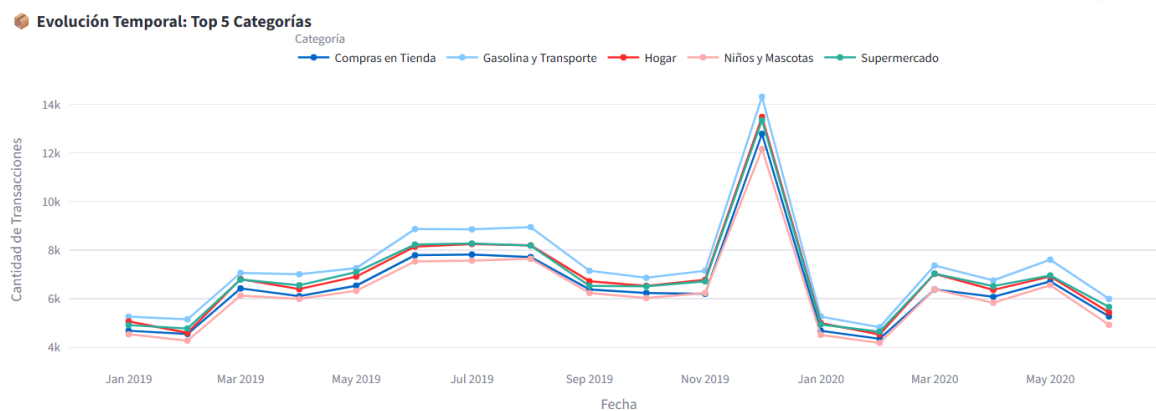
## C. Patrones de Gasto Mensual y Horario

**Descripción:** Muestra la estacionalidad (meses con más gasto) y las horas pico de transacciones.

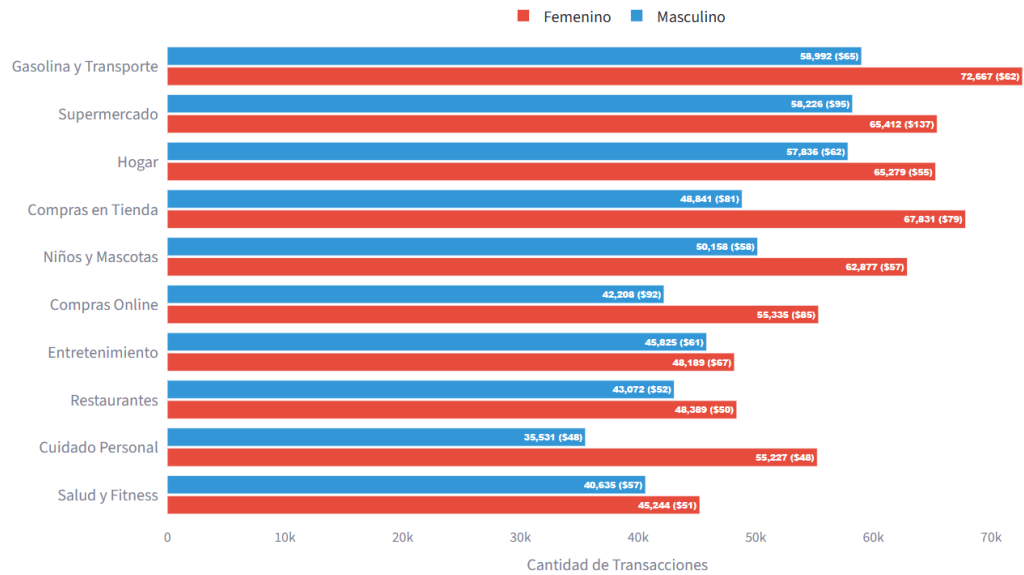


## D. Preferencias por Categoría

**Descripción:** Desglose de en qué gastan más dinero los diferentes géneros.



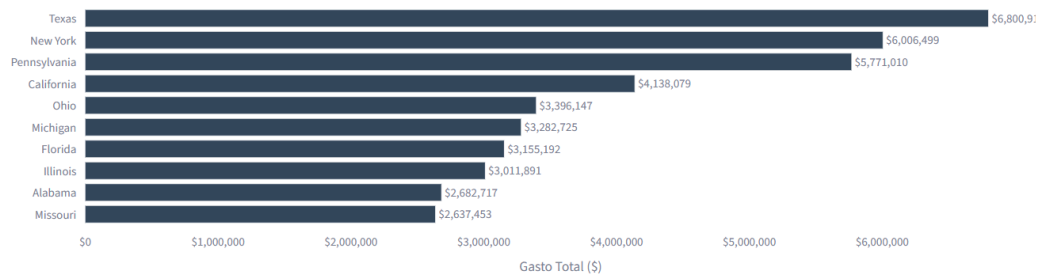
### Top 10 Categorías: Transacciones (Monto Promedio)



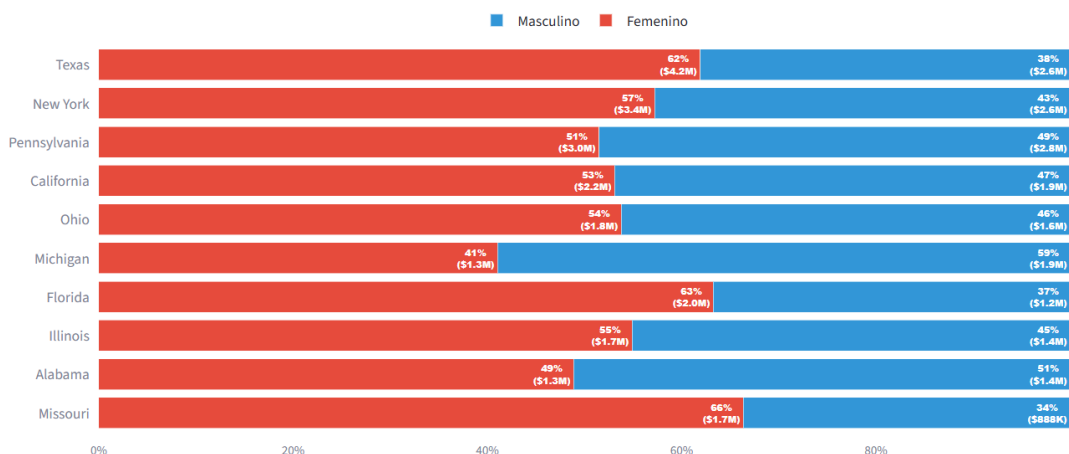
## E. Análisis Geográfico: Top Estados

**Descripción:** Ranking de los estados con mayor volumen de gasto, permitiendo ver tanto el total absoluto como la proporción por género.

### Top 10 Estados: Gasto Total (Mayor Gasto)

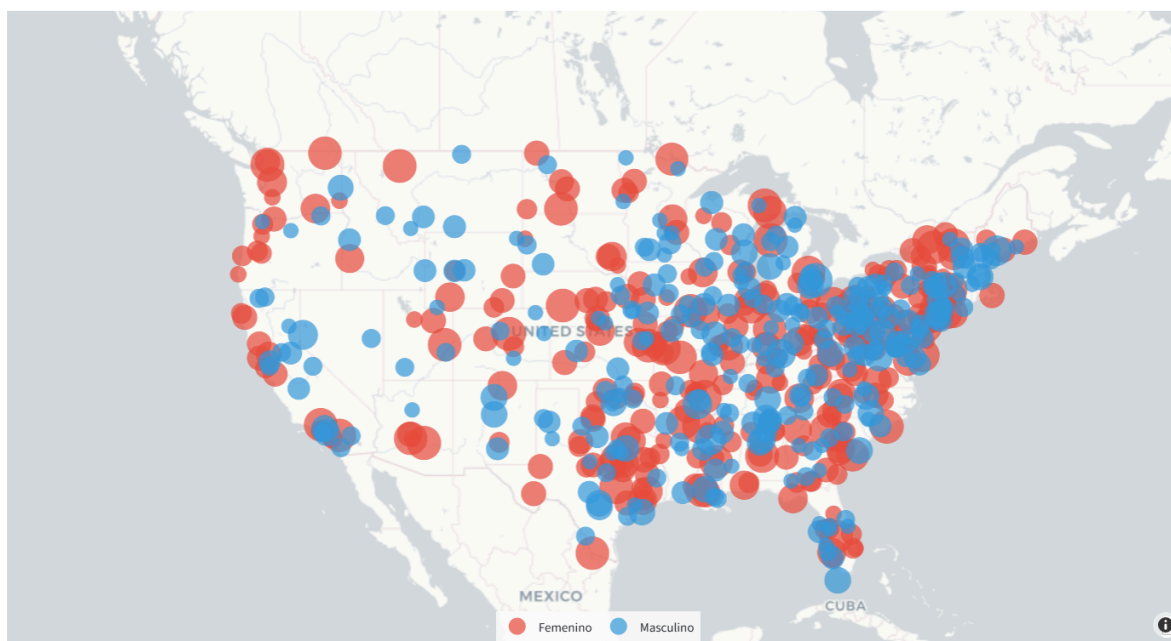


Top 10 Estados: Proporción de Gasto por Género (Orden: Total)



## F. Mapa de Concentración

**Descripción:** Visualización geoespacial de la densidad de transacciones.



### Hallazgos Principales:

Tras el análisis detallado del tablero de visualización de datos, se identificaron los siguientes hallazgos clave relacionados con el comportamiento transaccional, patrones temporales y distribución por categorías y regiones:

#### 1. KPIs Generales

El conjunto de datos analizado refleja un volumen significativo de actividad financiera:

Total de transacciones: 1,296,675

Monto total transado: \$91.2 millones

Ticket promedio por transacción: \$70.35

Estos indicadores permiten evidenciar una alta frecuencia de transacciones junto con un valor promedio relativamente estable.

#### 2. Análisis por Género

El comportamiento por género muestra diferencias notables tanto en volumen como en valor transaccional:

Mujeres (F):

Representan el mayor aporte económico con \$49.8 millones (54.6%), además del mayor número de transacciones.

Esto sugiere una mayor actividad y participación en las actividades comerciales reflejadas en el dataset.

Hombres (M):

Componen el 45.4% del gasto total, equivalente a \$41.4 millones.

Aunque representan un volumen menor que las mujeres, mantienen una participación considerable dentro del conjunto analizado.

#### 3. Análisis Temporal

El análisis por períodos muestra que el monto anual registrado para 2020 es significativamente menor (\$26.2 millones) en comparación con 2019 (\$65 millones).

Esta diferencia no representa necesariamente una caída real en el gasto, sino que se debe a que el dataset solo contiene información hasta junio de 2020, lo cual deja incompleto el registro anual.

#### **4. Análisis por Categorías**

Las categorías de consumo evidencian patrones concentrados en ciertos tipos de gasto:

grocery\_pos (Supermercados/Puntos de venta): Se posiciona como la categoría líder con \$14.4 millones.

shopping\_pos: Registra \$9.3 millones, siendo la segunda categoría más relevante.

shopping\_net: Aporta \$8.6 millones, mostrando una actividad importante en compras digitales o en línea.

Estas tres categorías concentran una parte significativa del gasto total, indicando mayor recurrencia en compras esenciales y comercio minorista.

#### **5. Análisis por Ubicación (Estados)**

A nivel territorial, el gasto se distribuye principalmente en tres estados:

Texas: Lidera con \$6.8 millones.

New York: Le sigue con \$6.0 millones.

Pennsylvania: Ocupa la tercera posición con \$5.8 millones.

Esta concentración sugiere que las regiones con mayor actividad económica o densidad poblacional tienden a representar mayor volumen transaccional.

# Bibliografía

Choksi, P. (2025). *Credit Card Transactions Dataset* [Conjunto de datos]. Kaggle.  
<https://www.kaggle.com/datasets/priyamchoksi/credit-card-transactions-dataset>

SQLite Consortium. (2024). *SQLite Documentation*. <https://sqlite.org/docs.html>

The pandas development team. (2024). *pandas documentation*.  
<https://pandas.pydata.org/docs/>

Python Software Foundation. (2024). *Python documentation*. <https://docs.python.org/3/>

KaggleHub. (2024). *KaggleHub Python Library Documentation*.  
<https://github.com/Kaggle/kagglehub>

Zapata Orrego, E., & Salazar Oviedo, D. (2025). *Proyecto Integrador 5 – Análisis de transacciones con tarjetas de crédito* [Repositorio GitHub].  
[https://github.com/eulicerzapata/Proyecto\\_Integrador5](https://github.com/eulicerzapata/Proyecto_Integrador5)

Schwaber, K., & Sutherland, J. (2020). *The Scrum Guide*. Scrum.org.  
<https://scrumguides.org/>