**Tarea 2: Consultas SQL con SQLAlchemy**

**Curso:** Data Science and Machine Learning Applied to Financial Markets – Módulo III  
**Nombre:** Andrés Padrón Quintana  
**Fecha:** 13 de octubre de 2025

**Bloque 1 – Consultas fáciles**

1. **Clientes con ingresos > 50,000**. Se usó WHERE ingresos > 50000. Lista de clientes con ingresos altos.
2. **Número total de clientes por país**. GROUP BY pais, COUNT(\*), ordenado desc. Identifica países con más clientes en la base.
3. **Clientes de estrato 2 con al menos una tarjeta** JOIN clientes-tarjetas con filtro estrato=2. Filtra clientes de ese segmento que poseen tarjeta.
4. **Promedio de ingresos por sexo > 40,000** GROUP BY sexo, HAVING AVG(ingresos) > 40000. Se observa que ambos sexos superan este promedio.
5. **Top 5 clientes por ingresos** ORDER BY ingresos DESC LIMIT 5. Identifica los clientes con mayores ingresos.
6. **Nombre, tipo de tarjeta y monto** JOIN entre clientes y tarjetas. Relaciona cada cliente con el tipo de tarjeta y su monto asociado.
7. **Clientes con más de una tarjeta** GROUP BY id\_cliente y HAVING COUNT(\*) > En esta base no aparece ningún cliente con múltiples tarjetas.
8. **Clientes con ingresos superiores al promedio** Subconsulta con AVG(ingresos). Identifica a los clientes con ingresos por arriba de la media global.
9. **Ranking de ingresos por país** Función de ventana RANK() OVER (PARTITION BY pais). Ordena clientes dentro de su país según ingresos.
10. **Clientes con tarjeta de crédito en el top 10% de ingresos** Función PERCENT\_RANK() sobre ingresos y filtro >= 0.9. Obtiene a los clientes con crédito que están en el percentil más alto.

**Bloque 2 – Consultas intermedias**

1. **Top 3 países con mayor ingreso promedio (clientes con crédito)** JOIN, filtro tipo='Credito', GROUP BY pais, ORDER BY promedio DESC LIMIT 3. Se destacan Perú, Madagascar y Korea del Sur.
2. **Clientes con ingresos mayores al promedio de su estrato** Subconsulta correlacionada por estrato. Muestra líderes de ingresos dentro de cada estrato socioeconómico.
3. **Cliente con más tarjetas en cada país** RANK() OVER (PARTITION BY pais ORDER BY COUNT(\*) DESC). En esta base, todos tienen solo una tarjeta, por lo que todos aparecen con valor 1.
4. **Monto total de tarjetas por cliente > promedio global** SUM(monto) por cliente y HAVING > promedio. Selecciona a los clientes con montos por arriba del nivel medio.
5. **5 clientes más jóvenes en el top 10% de ingresos (con crédito)** NTILE(10) sobre ingresos, filtro decil=1, orden por edad asc. Lista los jóvenes dentro de la élite de ingresos.
6. **Clientes con débito y crédito** HAVING COUNT(DISTINCT tipo) = 2. En esta base no hay registros, ya que cada cliente solo tiene un tipo de tarjeta.
7. **Ingreso promedio por estrato y sexo > promedio general** Doble GROUP BY con HAVING. Resalta las combinaciones estrato-sexo con ingresos elevados sobre la media.
8. **Cliente con mayor monto total por país** ROW\_NUMBER() OVER (PARTITION BY pais ORDER BY SUM(monto) DESC. Muestra al cliente más relevante financieramente en cada país.
9. **Clientes con ingresos > promedio de su país y países con ≥ 5 clientes** Subconsulta para promedio por país y HAVING COUNT>=5. Identifica a los “outliers positivos” en países suficientemente representados.
10. **Estrato con mayor promedio de montos de crédito** AVG(monto) por estrato, ordenado desc. El estrato 3 es el que concentra los montos más altos en tarjetas de crédito.

**Conclusión**

Todas las consultas se ejecutaron correctamente en Python con SQLAlchemy y pandas. Se emplearon cláusulas de filtrado (WHERE, HAVING), agregaciones (GROUP BY), subconsultas y funciones de ventana (RANK, NTILE, ROW\_NUMBER). Los resultados reflejan la estructura del dataset, donde cada cliente posee una sola tarjeta y se identifican patrones de ingresos por país, estrato y tipo de tarjeta.