



# **Sistema de Recomendación de Productos para Comercios Locales**

*Desafío técnico para optar al cargo Data Scientist*

Rindolfo Barra

30 de Septiembre, 2024



# Contexto

Mayorista11 busca desarrollar un sistema de recomendación de productos para comercios locales, con objetivo de optimizar sus ventas con productos que no han sido comprados, pero son populares en la misma zona geográfica.

## Objetivo

Mejorar la experiencia del cliente con información clave: acceso a productos similares, mejor control de inventario y reducción de costes.





# Hipótesis Iniciales

1. **Patrones de compra locales:** Priorizar productos que son populares entre comercios similares en la misma comuna.
2. **Historial de compras del comercio:** Tener en cuenta los productos que el comercio ya ha comprado para evitar recomendaciones redundantes.





# Descripcion de los datos

## ❖ Comercios:

- **id\_commerce**: ID único del comercio.
- **district**: Comuna donde opera el comercio.

## ❖ Productos:

- **id\_product**: ID único del producto.
- **name**: Nombre del producto.
- **category**: Categoría (Ropa, Hogar, etc).
- **price**: Precio unitario.

## ❖ Transacciones:

- **id\_commerce**: Comercio que realizó la compra.
- **id\_product**: Producto adquirido.
- **quantity**: Cantidad comprada.
- **price**: Precio total de la compra.





# Modelo Relacional



Se construyó un modelo tipo estrella con la tabla **transactions** como tabla de hechos y tanto **commerce** como **products** corresponden a las tablas de dimensiones.



# Hallazgos y validaciones

- Identifiqué valores duplicados en la tabla **transacciones**. Asumí que correspondían a datos históricos, por lo que se agruparon por **comercio** y **producto**, sumando la columna **cantidad**.
- Se observó que **todos los comercios habían adquirido al menos uno de todos los productos únicos**, lo que invalidaba la segunda hipótesis: evitar recomendar productos que ya habían sido comprados. Esto llevó a rechazar tal hipótesis, pues no aplicaba en este contexto.
- Se optó por **penalizar** aquellos productos, es decir, los productos que un comercio ya ha adquirido reciben un menor peso en su "**score**" de recomendación, sin ser eliminados por completo del sistema.



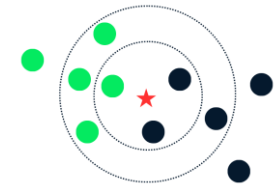
# Modelos Elegidos

Exploramos múltiples modelos para comprender cuál proporcionaba las recomendaciones más precisas y útiles:

- **Jaccard:** seleccionado por su simplicidad en términos de conjuntos, considerando productos adquiridos en común entre comercios.
- **Coseno:** elegido para aprovechar la granularidad de los datos y medir la proximidad entre comercios basados en patrones de compra completos.
- **KNN:** se utilizó para crear recomendaciones basadas en vecindarios de comercios similares, simulando un enfoque de recomendaciones colaborativas.

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}$$

$$\text{similarity} = \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \|\mathbf{B}\|} = \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2} \sqrt{\sum_{i=1}^n B_i^2}}$$





# Procedimiento

- Crear `matriz cliente-producto` con la `cantidad` como indicador de `popularidad`.
- Crear funciones para obtener los productos para cada modelo, excluyendo el comercio objetivo.
- Crear una función para recomendar: en lugar de excluir, penalizamos aquellos productos ya comprados con un menor score.
- Normalizar `Score` para cada modelo de 0 a 1.
- Devolver el `top 10` de recomendaciones.





# Desarrollo Web



## Sistema de Recomendación de Productos para Comercios Locales

Selecciona la comuna

Providencia



Selecciona el comercio

1



Selecciona el modelo de recomendación

Jaccard



Mostrar recomendaciones



# Resultados por Modelos

Top 10 recomendaciones (adjusted\_score)

Comuna: Macul

Comercio: 10

## Jaccard

	id_product	adjusted_score	category	price
0	2	1	Electrónica	76
1	10	0.9056	Ropa	56
2	30	0.867	Juguetes	60
3	17	0.8369	Alimentos	54
4	45	0.8155	Hogar	33
5	14	0.7811	Juguetes	95
6	29	0.7682	Hogar	64
7	34	0.7597	Juguetes	23
8	44	0.7382	Ropa	15
9	39	0.6996	Electrónica	41

## Coseno

	id_product	adjusted_score	category	price
0	2	1	Electrónica	76
1	30	0.9064	Juguetes	60
2	17	0.8369	Alimentos	54
3	29	0.8085	Hogar	64
4	10	0.7916	Ropa	56
5	49	0.7788	Ropa	19
6	44	0.7536	Ropa	15
7	14	0.7242	Juguetes	95
8	39	0.6876	Electrónica	41
9	45	0.6428	Hogar	33

## KNN

	id_product	adjusted_score	category	price
0	2	1	Electrónica	76
1	49	0.9343	Ropa	19
2	29	0.9336	Hogar	64
3	17	0.8201	Alimentos	54
4	12	0.7157	Ropa	48
5	27	0.7029	Alimentos	68
6	25	0.6821	Electrónica	96
7	45	0.6773	Hogar	33
8	11	0.6747	Hogar	44
9	28	0.668	Electrónica	65



# Evaluación de modelos

Modelo	Precisión	Recall	F1-Score
Jaccard	0.25	0,29	0.26
Coseno	0.20	0.23	0.21
KNN	0.29	0.33	0.31

**Jaccard**: Rendimiento balanceado, con espacio para mejorar en la relevancia de las recomendaciones.

**Coseno**: El menos efectivo en precisión y relevancia.

**KNN**: Es el modelo más sólido. Genera mejores recomendaciones, con un buen equilibrio entre precisión y recall.



# Conclusiones

- El sistema de recomendación de productos fue diseñado para comercios locales, priorizando la **personalización de productos** basados en los patrones de compra de **comercios similares** y la **popularidad** en la comuna.
- Se utilizaron tres modelos diferentes para generar las recomendaciones: **Jaccard**, **Coseno** y **KNN**.
- El modelo **K-Nearest Neighbors (KNN)** fue seleccionado por sus mejores métricas estimadas. Por otro lado, mencionar que Jaccard no considera la cantidad y similitud Coseno solo considera un vecino más cercano.
- La **penalización** de productos ya comprados por un comercio fue clave para evitar recomendaciones redundantes. Sin dejar de considerar productos comprados que pudieran ser útiles para reposición o futuras compras.



# Recomendaciones

**Datos temporales:** Incorporar fechas y número de transacciones permitiría captar tendencias temporales, mejorando la calidad de las recomendaciones.

**Evaluación:** Evaluar los modelos con el comportamiento real de los clientes para obtener métricas menos sesgadas.

**Segmentación:** Incorporar la categoría en el cálculo de similitud o usarla como filtro para productos recomendados basados en la categoría de interés.

**Mejora continua:** Probar con redes neuronales. Implementar un sistema **MLOps** en línea permitiría ajustar las recomendaciones en tiempo real con datos actualizados.

