**FACULTAD DE INGENIERÍA**

**CARRERA DE INGENIERÍA DE SISTEMAS COMPUTACIONALES**



**INFORME**

**TÍTULO:** INFORME DE ANÁLISIS DE DATOS DE VENTAS Y MEJORA DE PROPUESTA

**CURSO:** BASE DE DATOS AVAN. Y BIG DATA

**DOCENTE DEL CURSO:** Ing. ELVIS WILSON ALCANTARA PINEDO

**AUTORES:**

* EMERSON ANDRES ANCHAPURI MANGO

CHORRILLOS – PERÚ

2024-2

INDICE

[Introducción 3](#_Toc182409440)

[Objetivo 3](#_Toc182409441)

[Metodología 3](#_Toc182409442)

[Resultados 3](#_Toc182409443)

[Conclusiones 3](#_Toc182409444)

[IMÁGENES DEL CASO PROPUESTO 4](#_Toc182409445)

[MEJORA DE LA PROPUESTA 6](#_Toc182409446)

[Optimización del Cálculo de Media, Mediana y Moda 6](#_Toc182409447)

[Análisis de Correlación Mejorado 6](#_Toc182409448)

[Filtrado de Datos de Forma Eficiente 6](#_Toc182409449)

[Generación de Nuevas Columnas con Indicadores 7](#_Toc182409450)

[Visualización Mejorada con Gráficos Estadísticos 7](#_Toc182409451)

[IMÁGENES DE LA MEJORA DEL CASO PROPUESTO 7](#_Toc182409452)

[BIBLIOGRAFIA 10](#_Toc182409453)

**INFORME DE ANÁLISIS DE DATOS DE VENTAS**

# Introducción

Este informe analiza los datos de ventas de productos en la tienda para el período de enero a diciembre de 2023. Los datos incluyen el identificador de producto, la cantidad vendida y el inventario disponible. Nuestro objetivo es calcular métricas descriptivas como la media, mediana, moda y la correlación entre la cantidad vendida y el inventario, para entender las tendencias y la relación entre estas variables.

# Objetivo

Calcular la media, mediana y moda de las cantidades vendidas de los productos.

Evaluar la correlación entre el inventario disponible y las ventas, para determinar si hay una relación significativa entre ambas variables.

# Metodología

Usaremos PySpark en Google Colab para procesar y analizar los datos, calculando:

* Media: para obtener el promedio de ventas.
* Mediana: para observar el valor central en la distribución de ventas.
* Moda: para identificar las cantidades de venta más frecuentes.
* Correlación: para analizar si el inventario tiene relación con el número de ventas.

# Resultados

* Media: La cantidad promedio de productos vendidos fue de 150 unidades.
* Mediana: El valor mediano en la distribución de ventas fue de 130 unidades.
* Moda: La cantidad de venta más común fue de 100 unidades.
* Correlación: La correlación entre inventario y ventas fue de -0.3, lo que sugiere una relación negativa débil, indicando que un aumento en inventario tiende a reducir las ventas ligeramente, probablemente debido a la sobreoferta o falta de demanda.

# Conclusiones

La media y la mediana muestran que la mayoría de los productos tienen ventas de alrededor de 130 a 150 unidades. La moda de 100 unidades indica que algunos productos son más vendidos que otros de manera significativa. La correlación negativa leve entre inventario y ventas sugiere que se podría optimizar el inventario para reducir costos y evitar sobreoferta en productos de baja demanda.

# IMÁGENES DEL CASO PROPUESTO

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

# MEJORA DE LA PROPUESTA

# Optimización del Cálculo de Media, Mediana y Moda

* Mejora: Se implementaron funciones de PySpark para el cálculo de las métricas descriptivas (media, mediana y moda) en lugar de operaciones tradicionales en Pandas, aprovechando el procesamiento distribuido de Spark para manejar grandes volúmenes de datos de manera más eficiente.
* Explicación:
  + Media: Utilizamos la función avg() de PySpark para obtener la media de las cantidades vendidas, lo cual es más escalable y rápido en comparación con Pandas cuando se trabaja con datos voluminosos.
  + Mediana: Se empleó una combinación de ordenamiento y selección de registros con PySpark para calcular la mediana, lo que permite un procesamiento más ágil sin necesidad de conversión a Pandas.
  + Moda: Para la moda, se realizó un conteo de frecuencias por cantidad vendida con el método groupBy().count(), identificando el valor de venta más frecuente sin necesidad de transformar los datos.

# Análisis de Correlación Mejorado

* Mejora: Se optimizó el cálculo de la correlación entre inventario y ventas mediante el uso de la función corr() de PySpark.
* Explicación:
  + La función corr() se aplicó directamente sobre las columnas de ventas e inventario en PySpark, proporcionando un cálculo eficiente de la correlación sin necesidad de cambiar de entorno ni herramientas. Esta optimización mejora el rendimiento y evita la carga adicional de transformar los datos a un DataFrame de Pandas.

# Filtrado de Datos de Forma Eficiente

* Mejora: Implementación de filtros en Spark antes de realizar los cálculos, limitando el análisis a los datos relevantes (por ejemplo, excluir productos con inventario excesivo o ventas atípicamente bajas).
* Explicación:
  + Se utilizó filter() en PySpark para seleccionar solo los productos dentro de un rango de inventario o ventas específico, reduciendo la cantidad de datos en cada operación de análisis y optimizando el uso de memoria y tiempo de procesamiento.

# Generación de Nuevas Columnas con Indicadores

* Mejora: Adición de una columna categórica para clasificar los productos como “Alto” o “Bajo” en ventas según el umbral de 150 unidades.
* Explicación:
  + Con la función withColumn() y when(), se creó una columna Nivel\_Venta que clasifica los productos. Este indicador permite un análisis segmentado y facilita visualizar los productos que podrían requerir diferentes estrategias de inventario.

# Visualización Mejorada con Gráficos Estadísticos

* Mejora: Generación de gráficos de distribución y promedios de ventas por categoría para mejorar la interpretación de las métricas calculadas.
* Explicación:
  + Los datos se convierten a un DataFrame de Pandas después de los cálculos principales en PySpark. Luego, se usan gráficos de Matplotlib para mostrar la distribución de las ventas y el promedio de ventas por categoría, lo que permite visualizar patrones y tendencias clave de forma intuitiva.

# IMÁGENES DE LA MEJORA DEL CASO PROPUESTO

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Texto

Descripción generada automáticamente con confianza media

Tabla

Descripción generada automáticamente con confianza media

Gráfico, Gráfico de barras

Descripción generada automáticamente

Gráfico, Histograma

Descripción generada automáticamente

# BIBLIOGRAFIA

* Armbrust, M., Das, T., Xin, R., Zaharia, M., Dave, A., & Sen, P. (2015). Spark SQL: Relational data processing in Spark. Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data, 1383–1394. <https://doi.org/10.1145/2723372.2742797>
* Pérez, F., & Granger, B. E. (2007). IPython: A system for interactive scientific computing. Computing in Science & Engineering, 9(3), 21–29. <https://doi.org/10.1109/MCSE.2007.53>
* Zaharia, M., Chowdhury, M., Das, T., Dave, A., & Ma, J. (2012). Resilient distributed datasets: A fault-tolerant abstraction for in-memory cluster computing. Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation.<https://www.usenix.org/conference/nsdi12/technical-sessions/presentation/zaharia>
* Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. Computing in Science & Engineering, 9(3), 90–95. <https://doi.org/10.1109/MCSE.2007.55>
* Zaharia, M., Chowdhury, M., Franklin, M. J., Shenker, S., & Stoica, I. (2010). Spark: Cluster computing with working sets. Proceedings of the 2nd USENIX Conference on Hot Topics in Cloud Computing, 10(10-10), 95. <https://www.usenix.org/conference/hotcloud-10/spark-cluster-computing-working-sets>
* Meng, X., Bradley, J., Yavuz, B., Sparks, E., Venkataraman, S., & Liu, D. (2016). MLlib: Machine learning in Apache Spark. The Journal of Machine Learning Research, 17(1), 1235–1241. <https://jmlr.org/papers/volume17/15-237/15-237.pdf>
* Gittens, A., Liu, D., Mahoney, M. W., & Drineas, P. (2016). Revisiting the Nystrom method for improved large-scale machine learning. The Journal of Machine Learning Research, 17(1), 3977–4041. <https://jmlr.org/papers/volume17/15-237/15-237.pdf>