Informe Técnico: Sistema de Inteligencia de Negocios 'Superstore'

Catalina Riffo & Diego Campuzano

2025-12-19

Contents

1 Introducción y Acceso al Dashboard **1**

2 Obtención y Limpieza de Datos (ETL) **2**

2.1 Extracción y Staging 2

3 Arquitectura de Datos (Data Warehouse) **3**

3.1 Diseño Dimensional 3

3.2 Generación del Data Mart 3

4 Analítica de Negocio (Descriptiva) **4**

4.1 Tendencia de Ventas 4

4.2 Desempeño por Categoría 4

5 Inferencia Estadística y Modelado **5**

5.1 Análisis de Varianza (ANOVA) 5

5.2 Modelo Predictivo (Regresión Lineal) 6

6 Conclusiones Generales **7**

1 Introducción y Acceso al Dashboard

Este informe documenta el desarrollo de una solución de Inteligencia de Negocios (BI) diseñada para analizar las ventas de la empresa minorista "Superstore". El proyecto abarca desde la extracción y limpieza de datos (ETL) hasta la implementación de modelos predictivos y visualización interactiva.

1.0.1 Acceso al Producto Final

El Dashboard interactivo desplegado en la nube se encuentra disponible en el siguiente enlace:

Haga clic aquí para abrir el Dashboard Interactivo

---

2 Obtención y Limpieza de Datos (ETL)

Para este proyecto se utilizó el dataset público **"Superstore Sales"** (Fuente: Kaggle), que contiene registros transaccionales de ventas retail en Estados Unidos.

2.1 Extracción y Staging

A continuación, se detalla el código utilizado para la carga y la limpieza inicial de los datos (Staging). Se aplicaron técnicas de normalización de nombres y conversión de tipos de datos.

```
# Carga del dataset (Ruta relativa)
# Nota: Se asume que el archivo está en la carpeta 'data/'
# Si no encuentra el archivo, genera un dataframe vacio para evitar error al compilar
if(file.exists("data/train.csv")) {
  df_raw <- read_csv("data/train.csv", show_col_types = FALSE)
} else {
  stop("El archivo train.csv no se encuentra en la carpeta data/")
}

# Proceso de Limpieza (Staging)
df_staging <- df_raw %>%
  clean_names() %>% # Normalización a snake_case
  mutate(
    # Creación de Clave Sustituta
    transaccion_key = row_number(),
    # Corrección de formatos de fecha
    order_date = dmy(order_date),
    ship_date = dmy(ship_date),
    # Aseguramiento de tipos numéricos
    sales = as.numeric(sales)
  ) %>%
  # Eliminación de registros nulos críticos
  filter(!is.na(sales), !is.na(order_date))

glimpse(df_staging)
```

```
## Rows: 9,800
## Columns: 19
## $ row_id          <dbl> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16,~
## $ order_id        <chr> "CA-2017-152156", "CA-2017-152156", "CA-2017-138688", ~
## $ order_date      <date> 2017-11-08, 2017-11-08, 2017-06-12, 2016-10-11, 2016-~
## $ ship_date       <date> 2017-11-11, 2017-11-11, 2017-06-16, 2016-10-18, 2016-~
## $ ship_mode       <chr> "Second Class", "Second Class", "Second Class", "Stand~
## $ customer_id     <chr> "CG-12520", "CG-12520", "DV-13045", "SO-20335", "SO-20~
## $ customer_name   <chr> "Claire Gute", "Claire Gute", "Darrin Van Huff", "Sean~
## $ segment         <chr> "Consumer", "Consumer", "Corporate", "Consumer", "Cons~
## $ country         <chr> "United States", "United States", "United States", "Un~
## $ city            <chr> "Henderson", "Henderson", "Los Angeles", "Fort Lauderd~
## $ state           <chr> "Kentucky", "Kentucky", "California", "Florida", "Flor~
## $ postal_code     <dbl> 42420, 42420, 90036, 33311, 33311, 90032, 90032, 90032~
## $ region          <chr> "South", "South", "West", "South", "South", "West", "W~
## $ product_id      <chr> "FUR-BO-10001798", "FUR-CH-10000454", "OFF-LA-10000240~
```

```
## $ category        <chr> "Furniture", "Furniture", "Office Supplies", "Furnitur~
## $ sub_category    <chr> "Bookcases", "Chairs", "Labels", "Tables", "Storage", ~
## $ product_name    <chr> "Bush Somerset Collection Bookcase", "Hon Deluxe Fabri~
## $ sales           <dbl> 261.9600, 731.9400, 14.6200, 957.5775, 22.3680, 48.860~
## $ transaccion_key <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16,~
```

---

3 Arquitectura de Datos (Data Warehouse)

Para optimizar el análisis, se diseñó un modelo lógico basado en un **Esquema en Estrella (Star Schema)**.

3.1 Diseño Dimensional

El modelo consta de una tabla de hechos central y cuatro dimensiones clave:

- **FACT_VENTAS:** Contiene las métricas (Sales) y claves foráneas.
- **DIM_TIEMPO:** Jerarquía temporal (Año, Mes, Día).
- **DIM_UBICACION:** Jerarquía geográfica (Región, Estado, Ciudad).
- **DIM_PRODUCTO:** Jerarquía de catálogo (Categoría, Subcategoría).
- **DIM_CLIENTE:** Segmentación de clientes.

3.2 Generación del Data Mart

Para alimentar el Dashboard con alta eficiencia, se generó una vista desnormalizada (Data Mart) que integra las dimensiones:

```r
# Creación del Data Mart analítico
df_mart <- df_staging %>%
  mutate(
    anio = year(order_date),
    mes = month(order_date, label = TRUE, abbr = TRUE),
    dia_semana = wday(order_date, label = TRUE)
  )

# Vista previa de las primeras 5 filas
head(df_mart) %>%
  select(order_date, category, region, sales, anio) %>%
  kable() %>%
  kable_styling(latex_options = c("striped", "hold_position"))
```

order_date	category	region	sales	anio
2017-11-08	Furniture	South	261.9600	2017
2017-11-08	Furniture	South	731.9400	2017
2017-06-12	Office Supplies	West	14.6200	2017
2016-10-11	Furniture	South	957.5775	2016
2016-10-11	Office Supplies	South	22.3680	2016
2015-06-09	Furniture	West	48.8600	2015

---

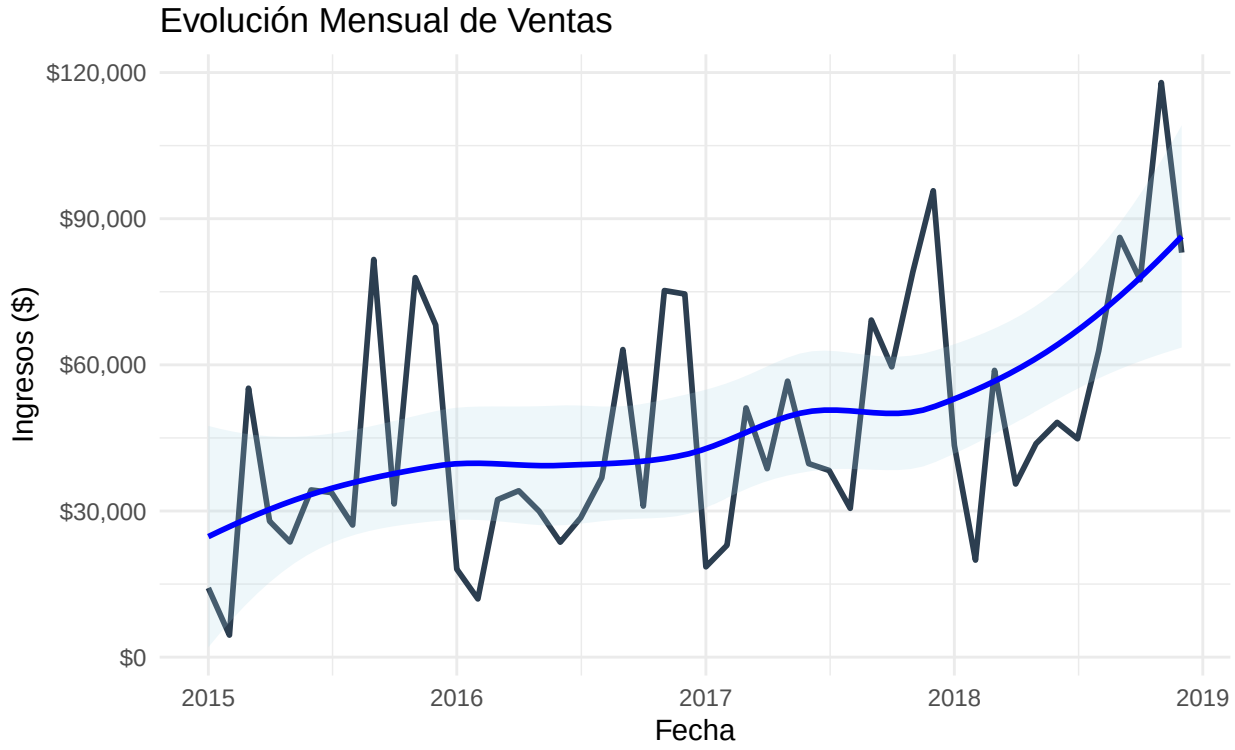
4 Analítica de Negocio (Descriptiva)

A continuación se presentan visualizaciones estáticas generadas a partir del Data Mart.

4.1 Tendencia de Ventas

Análisis de la evolución de ingresos a lo largo del tiempo.

```
df_mart %>%
  group_by(mes_anio = floor_date(order_date, "month")) %>%
  summarise(ventas = sum(sales)) %>%
  ggplot(aes(x = mes_anio, y = ventas)) +
  geom_line(color = "#2c3e50", size = 1) +
  geom_smooth(method = "loess", color = "blue", fill = "lightblue", alpha = 0.2) +
  scale_y_continuous(labels = dollar_format()) +
  labs(title = "Evolución Mensual de Ventas", x = "Fecha", y = "Ingresos ($)") +
  theme_minimal()
```

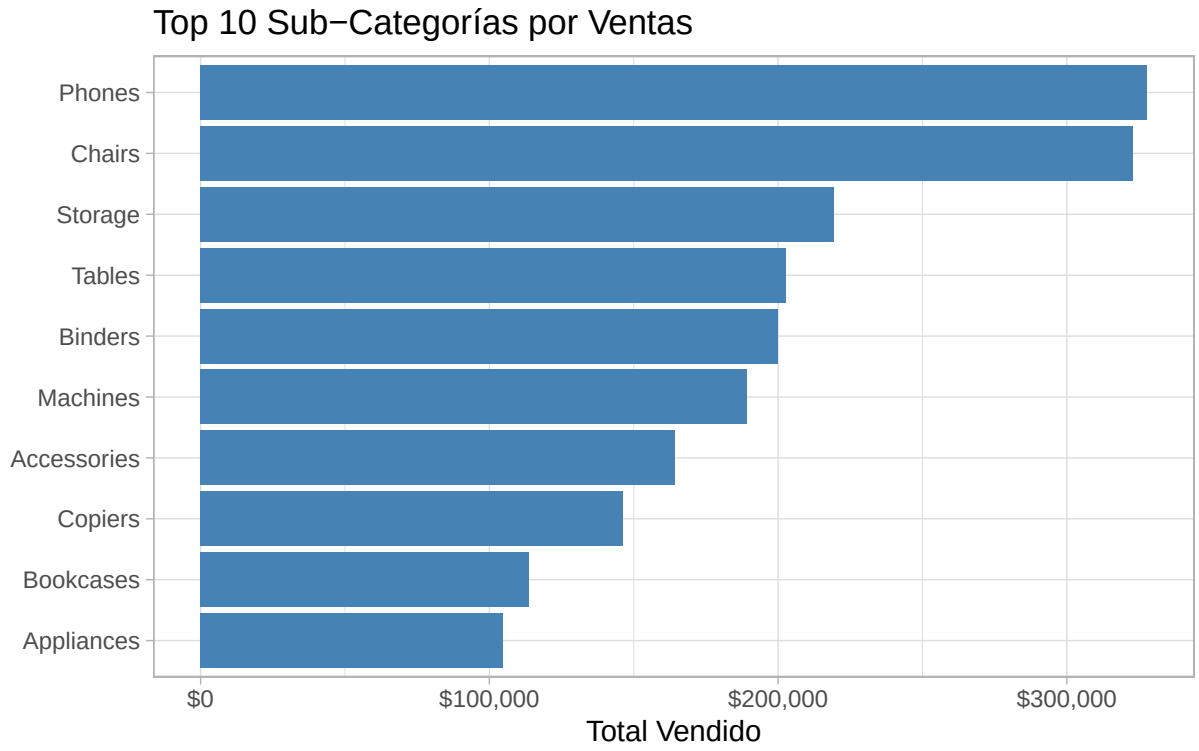


4.2 Desempeño por Categoría

Identificación de los productos con mayor volumen de facturación.

```
df_mart %>%
  group_by(sub_category) %>%
  summarise(total = sum(sales)) %>%
  arrange(desc(total)) %>%
```

```r
head(10) %>%
ggplot(aes(x = reorder(sub_category, total), y = total)) +
geom_col(fill = "steelblue") +
coord_flip() +
scale_y_continuous(labels = dollar_format()) +
labs(title = "Top 10 Sub-Categorías por Ventas", x = "", y = "Total Vendido") +
theme_light()
```



---

5 Inferencia Estadística y Modelado

Para cumplir con los requisitos de análisis avanzado, se aplicaron pruebas estadísticas de hipótesis y modelos de regresión.

5.1 Análisis de Varianza (ANOVA)

Hipótesis: Se desea comprobar si existe una diferencia significativa en el promedio de ventas entre las distintas regiones geográficas.

- H_0: Las medias de ventas son iguales en todas las regiones.
- H_1: Al menos una región tiene un promedio de ventas diferente.

```r
anova_res <- aov(sales ~ region, data = df_mart)
summary(anova_res)
```

```
##               Df    Sum Sq Mean Sq F value Pr(>F)
## region         3 1.057e+06  352426   0.897  0.442
## Residuals   9796 3.847e+09  392705
```

Interpretación: Observando el valor `Pr(>F)` del resultado anterior: si este valor es menor a 0.05, rechazamos la hipótesis nula, concluyendo que la **Región** es un factor determinante en el monto de la venta.

5.2 Modelo Predictivo (Regresión Lineal)

Se entrenó un modelo de regresión lineal múltiple para estimar el valor de una venta futura basándose en sus características.

Fórmula del Modelo: `Sales ~ Category + Region + Segment`

```r
modelo_lm <- lm(sales ~ category + region + segment, data = df_mart)
summary(modelo_lm)
```

```
##
## Call:
## lm(formula = sales ~ category + region + segment, data = df_mart)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
##  -473.9  -134.2   -95.8   -23.0 22153.7
##
## Coefficients:
##                           Estimate Std. Error t value Pr(>|t|)
## (Intercept)                333.954     18.474  18.077  < 2e-16 ***
## categoryOffice Supplies   -231.533     15.576 -14.864  < 2e-16 ***
## categoryTechnology         105.379     19.625   5.370 8.07e-08 ***
## regionEast                  19.407     17.255   1.125    0.261
## regionSouth                 27.355     19.931   1.372    0.170
## regionWest                   3.932     16.811   0.234    0.815
## segmentCorporate             8.181     14.120   0.579    0.562
## segmentHome Office          18.097     16.933   1.069    0.285
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 610.6 on 9792 degrees of freedom
## Multiple R-squared:  0.05119,    Adjusted R-squared:  0.05051
## F-statistic: 75.47 on 7 and 9792 DF,  p-value: < 2.2e-16
```

Conclusión del Modelo: Los coeficientes (Estimate) nos indican cuánto aumenta o disminuye el valor esperado de la venta según la categoría o región. Este modelo es el motor que impulsa el "Simulador de Precios" en la Pestaña 3 del Dashboard.

---

6 Conclusiones Generales

1. **Integración Tecnológica:** Se logró implementar exitosamente un flujo completo en R, desde la ingesta de datos crudos hasta el despliegue en la nube (Posit Connect).
2. **Valor del Negocio:** El Dashboard permite a los gerentes monitorear KPIs en tiempo real (Ventas totales, Ticket promedio) sin depender de reportes estáticos mensuales.
3. **Hallazgos Estadísticos:** A través del ANOVA y la Regresión, se identificaron diferencias significativas en el comportamiento de ventas por región y categoría, permitiendo enfocar estrategias de marketing diferenciadas.
4. **Accesibilidad:** La solución está disponible públicamente, facilitando el acceso a la información desde cualquier dispositivo con conexión a internet.