

Reporte Técnico Caso “LifeStore”

Instituto Emtech

Curso STFUNED: Fundamentos de Programación con
Python

Grupo 1

Armin Antonio Pimienta Flores

Contenido

Introducción	2
Descripción del Código	2
Solución del problema	8
Conclusión.....	8

Introducción

LifeStore simula una tienda virtual que ofrece productos electrónicos y de computación. La gerencia del área identifico una situación en la cual involucra una acumulación de inventario. De igual manera identifico una reducción de búsqueda de productos determinantes. Por dicha situación se ha solicitado un análisis en el cual dentro del informe deben incluir los siguientes puntos.

1. Productos mas vendidos y productos rezagado a partir de las categorías con menores ventas y categorías con menores búsquedas
2. Productos por reseña en el servicio a partir del análisis de categorías con mayores ventas y categorías con mayores búsquedas.
3. Sugerir una estrategia de productos a retirar del mercado, así como sugerencia de cómo reducir la acumulación de inventario considerando los datos de ingresos y ventas mensuales.

Descripción del Código

En la primera sección se importa las librerías de pandas y numpy y de igual manera el archivo que funciona como base de datos de la tienda “LifeStore”

```
#importar librerias y archivo "lifestore_file"
import pandas as pd
import numpy as np
import lifestore_file
```

En la segunda sección se cargan las diferentes listas de la base de datos como cuadros de datos para poder manipularlos mas adelante.

```
#Definición de dataframes
ventas = pd.DataFrame(lifestore_file.lifestore_sales,
                      columns=["id_sale", "id_product", "score", "date", "refund"])

busquedas=pd.DataFrame(lifestore_file.lifestore_searches,
                       columns=["id_search", "id_product"])

ventas_no_dev= pd.DataFrame(lifestore_file.lifestore_sales,
                           columns=["id_sale", "id_product", "score", "date", "refund"])
productos= pd.DataFrame(lifestore_file.lifestore_products,
                        columns=["id_product", "name", "price", "category", "stock"])
```

En esta sección se hace la limpieza de un cuadro de datos para omitir los datos que contenga una devolución (refund = 1) ya que si una venta contiene un reembolso no se considera venta

```
#extrae de Ventas solo las columnas "id_sale", "id_product", "refund"
ventas_no_dev_1= ventas_no_dev.drop(["score", "date"], axis=1)
#Se omiten las ventas terminadas en devolución
ventas_no_dev_1.drop(ventas_no_dev_1[ventas_no_dev_1.refund == 1].index, inplace=True)
#omite la columna "refund"
ventas_no_dev_2= ventas_no_dev_1.drop(["refund"], axis=1)
```

Esta primera parte donde interacciona el usuario se implemento un código para darle seguridad al programa en donde se ingresa por medio del teclado un usuario y una contraseña.

```
user = "user"
contraseña = "user"
```

```

username = input("Ingrese su nombre de usuario:\n >" )
password = input("Ingrese la contraseña:\n >" )
if username == user:
    if password == contraseña:
        print("Iniciando Informe, espere por favor ")
    else:
        print("Error: contraseña incorrecta")
else:
    print("Error: ususario incorrecto")

```

En la penúltima sección se cargan las funciones para poder sacar los resultados, en otras palabras. En esta serie de operaciones se resuelven los datos solicitados de forma que a la hora de imprimirlos se derive un informe completo con todos los datos.

```

#Productos con mayores ventas
top_5=ventas_no_dev_2[["id_sale","id_product"]]
id_venta = top_5.groupby("id_product")["id_sale"].count()
id_venta.sort_values(ascending=False,inplace=True)
ventas5=id_venta.index.tolist()

##Prodcutos con mayores búsquedas
m_búsquedas = búsquedas.groupby("id_product")["id_search"].count()
m_búsquedas.sort_values(ascending=False,inplace=True)
buscar10=m_búsquedas.index.tolist()

#Menos ventas por categoria: "Procesadores "
categoria_name1="procesadores"
pro_ventas=ventas[["id_sale","id_product"]]
id_ventas1 = pro_ventas.groupby("id_product")["id_sale"].count()
id_ventas1.sort_values(ascending=True,inplace=True)
menos_ventas1=id_ventas1.index.tolist()

#Menos ventas por categoria: "Tarjetas de video "
categoria_name2="tarjetas de video"
tvdo_ventas=ventas[["id_sale","id_product"]]
id_ventas2 = tvdo_ventas.groupby("id_product")["id_sale"].count()
id_ventas2.sort_values(ascending=True,inplace=True)
menos_ventas2=id_ventas2.index.tolist()

#Menos ventas por categoria: "Tarjetas madre "
categoria_name3="tarjetas madre"
tmd_ventas=ventas[["id_sale","id_product"]]
id_ventas3 = tmd_ventas.groupby("id_product")["id_sale"].count()
id_ventas3.sort_values(ascending=True,inplace=True)
menos_ventas3=id_ventas3.index.tolist()

#Menos ventas por categoria: "Discos duros "
categoria_name4="discos duros"
dd_ventas=ventas[["id_sale","id_product"]]
id_ventas4 = dd_ventas.groupby("id_product")["id_sale"].count()
id_ventas4.sort_values(ascending=True,inplace=True)
menos_ventas4=id_ventas4.index.tolist()

#Menos ventas por categoria: "pantallas"
categoria_name5="pantallas"
tv_ventas=ventas[["id_sale","id_product"]]
id_ventas5 = tv_ventas.groupby("id_product")["id_sale"].count()
id_ventas5.sort_values(ascending=True,inplace=True)
menos_ventas5=id_ventas5.index.tolist()

#Menos ventas por categoria: "Bocinas"

```

```

categoria_name6="bocinas"
bo_ventas=ventas[["id_sale","id_product"]]
id_ventas6 = bo_ventas.groupby("id_product")["id_sale"].count()
id_ventas6.sort_values(ascending=True,inplace=True)
menos_ventas6=id_ventas6.index.tolist()

#Menos ventas por categoria: "Audifonos"
categoria_name7="audifonos"
au_ventas=ventas[["id_sale","id_product"]]
id_ventas7 = au_ventas.groupby("id_product")["id_sale"].count()
id_ventas7.sort_values(ascending=True,inplace=True)
menos_ventas7=id_ventas7.index.tolist()

#menos busquedas por categoria: procesadores"
categoria_name1_1="procesadores"
pro_bus_id = busquedas.groupby("id_product")["id_search"].count()
pro_bus_id.sort_values(ascending=True,inplace=True)
menos_busquedas1=pro_bus_id.index.tolist()
#menos busquedas por categoria: tarjetas de videos"
categoria_name1_2 = "tarjetas de video"
tdv_bus_id = busquedas.groupby("id_product")["id_search"].count()
tdv_bus_id.sort_values(ascending=True,inplace=True)
menos_busquedas1_2=tdv_bus_id.index.tolist()
#menos busquedas por categoria: tarjetas madre"
categoria_name1_3="tarjetas madre"
tm_bus_id = busquedas.groupby("id_product")["id_search"].count()
tm_bus_id.sort_values(ascending=True,inplace=True)
menos_busquedas1_3=tm_bus_id.index.tolist()
#menos busquedas por categoria: discos duros"
categoria_name1_4="discos duros"
dd_bus_id = busquedas.groupby("id_product")["id_search"].count()
dd_bus_id.sort_values(ascending=True,inplace=True)
menos_busquedas1_4=dd_bus_id.index.tolist()
#menos busquedas por categoria: pantallas"
categoria_name1_5="pantallas"
tv_bus_id = busquedas.groupby("id_product")["id_search"].count()
tv_bus_id.sort_values(ascending=True,inplace=True)
menos_busquedas1_5=dd_bus_id.index.tolist()
#menos busquedas por categoria: bocinas"
categoria_name1_6="bocinas"
bo_bus_id = busquedas.groupby("id_product")["id_search"].count()
bo_bus_id.sort_values(ascending=True,inplace=True)
menos_busquedas1_6=bo_bus_id.index.tolist()
#menos busquedas por categoria: audifonos"
categoria_name1_7="audifonos"
aud_bus_id = busquedas.groupby("id_product")["id_search"].count()
aud_bus_id.sort_values(ascending=True,inplace=True)
menos_busquedas1_7=aud_bus_id.index.tolist()
#productos reseñas

ventas_no_resena1= ventas_no_dev.drop(["date"], axis=1)
ventas_no_resena1.drop(ventas_no_resena1[ventas_no_resena1.refund == 1].index, inplace=True)
#Mejores reseñas
mejores= ventas_no_resena1.groupby("id_product")["score"].mean()
mejores.sort_values(ascending=False,inplace=True)
top_r_m5=mejores.index.tolist()

#Peores reseñas
peores= ventas_no_resena1.groupby("id_product")["score"].mean()
peores.sort_values(ascending=True,inplace=False)
top_r_p5=peores.index.tolist()

#ventas anuales y promedio por mes
ventas_anuales= round(len(ventas)/12)
ventas_totales_ano = (len(ventas))

```

En la última sección del código se imprimen todos los resultados de forma continua

```
print("Informe Tecnico LifeStore ")
print("Top 5: Productos con mayores ventas\n\n")
j=1
for vttotal5 in ventas5:

    print(str(j)+"--- "+productos.loc[productos.id_product==vttotal5,"name"].item())
    if(j>4):
        break
    j+=1
print("\n\n\n\n")
print("Top 10: Productos con mayores busquedas\n\n")
j2=1
for busqueda_10 in buscar10:
    print(str(j2)+"--- "+productos.loc[productos.id_product==busqueda_10,"name"].item())
    if(j2>4):
        break
    j2+=1

print("\n\n\n\n")
print("Menores ventas por categoria\n\n")
print(f"Menores ventas de {categoria_name1} \n\n")
j3=1
localizacion_categoria1=productos.loc[productos.category==categoria_name1]
producto_x_categoria1=localizacion_categoria1["id_product"]
for ventasm51 in menos_ventas1:
    if(ventasm51 in producto_x_categoria1):
        n1= localizacion_categoria1.loc[localizacion_categoria1["id_product"]==ventasm51,"name"].item()
        print(str(j3)+" --- "+n1)
    if(j3>34):
        break
    j3+=1
print("\n\n\n")
print(f"Menores ventas de {categoria_name2} \n\n")
j4=1
localizacion_categoria2=productos.loc[productos.category==categoria_name2]
producto_x_categoria2=localizacion_categoria2["id_product"]
for ventasm52 in menos_ventas2:
    if(ventasm52 in producto_x_categoria2):
        n2= localizacion_categoria2.loc[localizacion_categoria2["id_product"]==ventasm52,"name"].item()
        print(str(j4)+" --- "+n2)
    if(j4>20):
        break
    j4+=1
print("\n\n\n")
print(f"Menores ventas de {categoria_name3} \n\n")
j5=1
localizacion_categoria3=productos.loc[productos.category==categoria_name3]
producto_x_categoria3=localizacion_categoria3["id_product"]
for ventasm53 in menos_ventas3:
    if(ventasm53 in producto_x_categoria3):
        n3= localizacion_categoria3.loc[localizacion_categoria3["id_product"]==ventasm53,"name"].item()
        print(str(j5)+" --- "+n3)
    if(j5>5):
        break
    j5+=1
print("\n\n\n")
print(f"Menores ventas de {categoria_name4} \n\n")
j6=1
localizacion_categoria4=productos.loc[productos.category==categoria_name4]
producto_x_categoria4=localizacion_categoria4["id_product"]
for ventasm54 in menos_ventas4:
    if(ventasm54 in producto_x_categoria4):
```

```

        n4= localizacion_categoria4.loc[localizacion_categoria4["id_product"]==ventasm54,"name"].item()
        print(str(j6)+" --- "+n4)
    if(j6>5):
        break
    j6+=1
print("\n\n")
print(f"Menores ventas de {categoria_name5} \n\n")
j7=1
localizacion_categoria5=productos.loc[productos.category==categoria_name5]
producto_x_categoria5=localizacion_categoria5["id_product"]
for ventasm55 in menos_ventas5:
    if(ventasm55 in producto_x_categoria5):
        n5= localizacion_categoria5.loc[localizacion_categoria5["id_product"]==ventasm55,"name"].item()
        print(str(j7)+" --- "+n5)
    if(j7>99):
        break
    j7+=1
print("\n\n")
print(f"Menores ventas de {categoria_name6} \n\n")
j8=1
localizacion_categoria6=productos.loc[productos.category==categoria_name6]
producto_x_categoria6=localizacion_categoria6["id_product"]
for ventasm56 in menos_ventas6:
    if(ventasm56 in producto_x_categoria6):
        n6= localizacion_categoria6.loc[localizacion_categoria6["id_product"]==ventasm56,"name"].item()
        print(str(j8)+" --- "+n6)
    if(j8>99):
        break
    j8+=1
print("\n\n")
print(f"Menores ventas de {categoria_name7} \n\n")
j9=1
localizacion_categoria7=productos.loc[productos.category==categoria_name7]
producto_x_categoria7=localizacion_categoria7["id_product"]
for ventasm57 in menos_ventas7:
    if(ventasm57 in producto_x_categoria7):
        n7= localizacion_categoria7.loc[localizacion_categoria7["id_product"]==ventasm57,"name"].item()
        print(str(j9)+" --- "+n7)
    if(j9>99):
        break
    j9+=1
print("\n\n\n")
print("Menores busquedas por categoria\n\n")
print(f"Menores busquedas de {categoria_name1_1} \n\n")
jb1=1
localizacion_categoria1_1=productos.loc[productos.category==categoria_name1_1]
producto_x_categoria1_1=localizacion_categoria1_1["id_product"]
for busm101 in menos_busquedas1:
    if(busm101 in producto_x_categoria1_1):
        n1_1= localizacion_categoria1_1.loc[localizacion_categoria1_1["id_product"]==busm101,"name"].item()
        print(str(jb1)+" --- "+n1_1)
    if(jb1>46):
        break
    jb1+=1
print("\n\n\n")
print(f"Menores busquedas de {categoria_name1_2} \n\n")
jb2=1
localizacion_categoria1_2=productos.loc[productos.category==categoria_name1_2]
producto_x_categoria1_2=localizacion_categoria1_2["id_product"]
for busm102 in menos_busquedas1_2:
    if(busm102 in producto_x_categoria1_2):
        n1_2= localizacion_categoria1_2.loc[localizacion_categoria1_2["id_product"]==busm102,"name"].values
        print(str(jb2)+" --- "+n1_2)
    if(jb2>20):
        break

```

```

    jb2+=1
    print("\n\n\n")
    print(f"Menores busquedas de {categoria_name1_3} \n\n")
    jb3=1
    localizacion_categoria1_3=productos.loc[productos.category==categoria_name1_3]
    producto_x_categoria1_3=localizacion_categoria1_3["id_product"]
    for busm103 in menos_busquedas1_3:
        if(busm103 in producto_x_categoria1_3):
            n1_3= localizacion_categoria1_3.loc[localizacion_categoria1_3["id_product"]==busm103,"name"].values
            print(str(jb3)+" --- "+n1_3)
        if(jb3>33):
            break
        jb3+=1
    print("\n\n\n")
    print(f"Menores busquedas de {categoria_name1_4} \n\n")
    jb4=1
    localizacion_categoria1_4=productos.loc[productos.category==categoria_name1_4]
    producto_x_categoria1_4=localizacion_categoria1_4["id_product"]
    for busm104 in menos_busquedas1_4:
        if(busm104 in producto_x_categoria1_4):
            n1_4= localizacion_categoria1_4.loc[localizacion_categoria1_4["id_product"]==busm104,"name"].values
            print(str(jb4)+" --- "+n1_4)
        if(jb4>36):
            break
        jb4+=1
    print("\n\n\n")
    print(f"Menores busquedas de {categoria_name1_5} \n\n")
    jb5=1
    localizacion_categoria1_5=productos.loc[productos.category==categoria_name1_5]
    producto_x_categoria1_5=localizacion_categoria1_5["id_product"]
    for busm105 in menos_busquedas1_5:
        if(busm105 in producto_x_categoria1_5):
            n1_5= localizacion_categoria1_5.loc[localizacion_categoria1_5["id_product"]==busm105,"name"].values
            print(str(jb5)+" --- "+n1_5)
        if(jb5>99):
            break
        jb5+=1
    print("\n\n\n")
    print(f"Menores busquedas de {categoria_name1_6} \n\n")
    jb6=1
    localizacion_categoria1_6=productos.loc[productos.category==categoria_name1_6]
    producto_x_categoria1_6=localizacion_categoria1_6["id_product"]
    for busm106 in menos_busquedas1_6:
        if(busm106 in producto_x_categoria1_6):
            n1_6= localizacion_categoria1_6.loc[localizacion_categoria1_6["id_product"]==busm106,"name"].values
            print(str(jb6)+" --- "+n1_6)
        if(jb6>99):
            break
        jb6+=1
    print("\n\n\n")
    print(f"Menores busquedas de {categoria_name1_7} \n\n")
    jb7=1
    localizacion_categoria1_7=productos.loc[productos.category==categoria_name1_7]
    producto_x_categoria1_7=localizacion_categoria1_7["id_product"]
    for busm107 in menos_busquedas1_7:
        if(busm107 in producto_x_categoria1_7):
            n1_7= localizacion_categoria1_7.loc[localizacion_categoria1_7["id_product"]==busm107,"name"].values
            print(str(jb7)+" --- "+n1_7)
        if(jb7>27):
            break
        jb7+=1

    print("\n\n\n")
    print ("Top 5 Mejores Reseñas \n")
    m=1

```



```

for mr in top_r_m5 :
    print(str(m)+" --- "+productos.loc[productos.id_product==mr,"name"].item())
    if(m>4):
        break
    m+=1
print("\n\n\n\n")
print ("Top 5 Peores Reseñas \n")
p=1
for pr in top_r_p5 :
    print(str(p)+" --- "+productos.loc[productos.id_product==pr,"name"].item())
    if(p>4):
        break
    p+=1
print("\n\n\n\n")
print(f"Ventas totales anuales: {ventas_totales_ano} \n")
print(f"Ventas promedio por mes: {ventas_anuales} \n")

```

Solución del problema

En el caso de los productos mas vendidos se tiene que son dos SSD, 2 procesadores y una tarjeta madre por lo cual se recomienda mantener un inventario de esos modelos para surtir la demanda. Y en el caso de los productos con menores ventas, la mayoría son procesadores, por lo cual los procesadores son un punto de oportunidad.

En el caso de las búsquedas, se ha notado que hay una preferencia por dos modelos de procesadores de la marca ryzen, pero con lo enunciado en el punto a favor creo que es otro indica que deben revisarse. Los demás productos residen en las SSD y una tarjeta madre. Se recomienda su promoción y si se puede un pequeño descuento para estimular su venta

Dentro del informe se hace énfasis en las menores ventas por categoría, por lo cual se recomienda revisar los modelos de cada una de las categorías para identificar que productos deben ser administrados. Una opción seria darles rebajas y promociones

En el caso de las reseñas se tiene dos tarjetas de video NVIDIA por lo cual son un punto a favor para seguir vendiendo esos modelos, también se tiene un procesador AMD, una tarjeta de video AMD y un SSD. Recomendando seguir ofertando dichos artículos ya que son bien recibido por la clientela

En cuanto a nuestros productos con peores reseñas la mayoría son AMD y un procesador Intel. Se recomienda revisar las reseñas de esos artículos para poder ver si depende de nosotros las reseñas o son en si por el producto y la marca.

En el caso de las ventas totales se deben comparar con el año pasado, las ventas promedio de igual manera.

Conclusión

Esta herramienta es un recurso muy importante ya que al entender los datos(ventas) se pueden hacer cambios en cuanto a la estrategia optada por gerencia con el objetivo de reducir costos, gasto y aumentar las ganancias. Sinceramente, en mi caso tuve limitantes a la hora de realizar el código, por lo cual quedo de manera con puntos que pudieron desarrollarse mejor o culminado. Con esto he

comprendido mis límites y habilidades dentro del manejo del lenguaje de programación Python. Lo que me ha dicho que necesito mas horas de practica y de estudio para poder completar proyectos similares con resultados de calidad.