

Análisis e informe 2020

Lifestore

Arturo Jair Mazón Ocampo

Introducción	2
Definición de código	3
El menú.	3
El algoritmo principal.	5
Función de conteo	5
Función de orden	6
El algoritmo principal:variación general .	7
Solución al problema.	8
1.- La filosofía y enfoque de los productos.	8
2.- La capacidad de dar un seguimiento mensual con el volumen y valor de las ventas	9
3.-La distribución de las ventas por categoría	9
4.- El valor de las reseñas	9
Conclusión	10

Introducción

El presente trabajo tiene como objetivo explicar el contenido del programa al cual se le ha llamado “Sistema de análisis Lifestore”, y ser una guía que acompañe a aclarar cualquier duda en conjunción con los comentarios contenidos en el programa.

La explicación del código será realizada en tres partes

1. El menú
2. El algoritmo principal
3. El algoritmo principal en su variación general.

Siendo la primera parte el contenido del menú y los detalles de su funcionamiento, tal como las variables y cada parte de este, la segunda tratará sobre la parte abstracta del algoritmo de ordenamiento y selección en el cual está fundamentado todo el programa, y la tercera parte, se mostrará como usando el algoritmo anterior con una variación puede utilizarse para ordenar y clasificar varias listas, en forma de categorías o criterios.

Una vez expuesto el código, se darán métodos para optimizar el rendimiento de las ventas y ganancias de Lifestore, utilizando la información que el programa es capaz de proveer.

Concluyendo con comentarios por parte del autor en términos de lo que representa el haber contratado este servicio, y lo que podría deparar el futuro de LifeStore con análisis de datos de manera continua.

Definición de código

Antes de empezar, se recomienda tener a la mano el código que se puede descargar o consultar en el siguiente enlace:

https://github.com/ArtMazon/Reporte_venta

El menú.

La estructura del menú de manera abstracta puede verse de la siguiente manera

while salida:

 Bloque de información de cuenta del usuario

for variable in conjunto de usuarios:

 if condición de usuario y contraseña:

 Bloque de asignación de acceso.

 if acceso:

 if privilegio:

 Bloque de aviso de privilegio no administrativo.

 else:

 while salida:

 Bloque de presentación del menú de opciones.

 if menu:

 Bloque de los 50 productos más vendidos

 elif menu:

 Bloque de los 100 productos más buscados

 elif menu:

for lista_categorias:

Menú de las categorías de productos

for lista_categorias_cincuenta_con_orden_inverso:

Bloque de los 50 productos con menores venta por categoría

elif menu:

for lista_categorias:

Menú de las categorías de productos

for lista_categorias_cincuenta_con_orden_inverso:

Bloque de los 100 productos con menores búsquedas por categoría

elif menu:

Bloque de los 20 productos con mejores reseñas

elif menu:

Bloque de los 20 productos con peores reseñas

elif menu:

for meses:

Menú de los meses a analizar.

for lista_categorias_cincuenta_con_orden_inverso:

Bloque de total de ventas e ingresos mensuales

elif menu:

Bloque de total de ingresos y ventas anuales.

else:

Bloque de advertencia de una opción no válida.

Bloque de opción para volver al menú principal.

else:

Bloque de advertencia de información de usuario incorrecta.

El algoritmo principal.

El algoritmo principal en el cual se basa este proyecto está compuesto de dos partes esenciales, la primera permite contabilizar cierto criterio, como las ventas o las búsquedas, y una función de orden.

Función de conteo

```
lista_pivote_uno = lista_con_subconjunto_de_criterio[:]
lista_sin_orden=[]
while lista in lista_pivote_uno:
    criterio=lista_pivote_uno[inciso]
    contador=0
    for variable in lista_con_subconjunto_de_criterio:
        if criterio,variable:
            contador+=1
            lista_pivote_uno.remove(variable)
    lista_sin_orden.append([criterio,i])
```

La función de orden del algoritmo principal esta conformado por los siguientes elementos:

1. Una lista pivote: está lista será una copia de una lista con las ventas que cumplen cierto criterio (ordenadas por categoría, mes, etc)

2. Una lista sin orden: esta lista tendrá la información de la lista pivote contabilizada con respecto a cierto criterio y cada elemento es una lista con el formato [criterio,i,alguna otra información].
Ejemplo : [id de producto, número de ventas] . {mes, ventas por mes}, [id de producto, número de búsquedas]
3. while de la lista pivote: permite hacer el conteo de cada elemento en la lista pivote, aunado con un .remove sobre la lista pivote garantiza que cada vez que un elemento sea contado, será eliminado de la lista, el proceso termina cuando la lista pivote se vuelve vacía.
4. Variable criterio: esta variable será el criterio que deseamos contar dentro de nuestra lista subconjunto . Ejemplo: ventas, búsquedas, calificación.
5. variable contador: esta será la variable auxiliar para el proceso de comparación y conteo.
6. for con la lista subconjunto de criterio: todo este bloque realiza la siguiente operación, primero toma el criterio y compara este valor con la posición correspondiente de todos los valores que vamos considerar para el conteo, si fuera en el caso de ventas, se compararía el id de producto con todos los id de producto de la lista lifestore_sales, si el criterio se encuentra en alguno de los elementos, el contador aumenta en una unidad, y remueve ese elemento de la lista pivote, esta operación se realiza con todos los elementos posibles de la lista subconjunto.
7. lista sin orden.append([criterio,i]): una vez terminado el proceso anterior, se añade el criterio y cuantas veces se encuentre este en los elementos a comparar, el proceso se reinicia ya con un nuevo valor para el criterio ya que se removieron todos los elementos similares en el proceso anterior, dejando la lista pivote solo con valores de criterio nuevo.

Función de orden

```
lista_pivote_dos = lista_sin_orden[:]
```

```
lista_con_orden=[]
```

```
while lista in lista_pivote_dos:
```

```
    maximo=lista_pivote_dos[inciso]
```

```
    for variable in lista_pivote_dos:
```

```
        if maximo < variable[inciso]:
```

```
            maximo=variable[inciso]
```

```
    for variable in lista_pivote_dos:
```

```
if maximo == variable[inciso]:  
  
    lista_con_orden.append(variable)  
  
    lista_pivote_dos.remove(variable)
```

La estructura del algoritmo de orden es similar , la diferencia principal está en los dos bloques de color azul celeste y naranja.

Primer bloque: En este bloque se toma la variable maximo, la cual se le asigna el valor a ordenar del primer elemento la lista que deseamos ordenar, y se compara con todos los elementos de la lista a ordenar, si es el mayor de todos, el proceso continúa, si no, se intercambia el valor de este por el del nuevo elemento y se repite el proceso hasta que encontramos un máximo de esta lista.

Segundo bloque: Una vez encontrado el máximo de los datos, se verifica que si hay elementos con este mismo valor, de ser así estos elementos se agregan a la lista con orden , y se eliminan de la lista pivote.

Una vez que se termina este proceso una vez, la lista pivote dos elimina el primer elemento máximo y sus copias, dejando una sub lista a la cual se repite el proceso hasta que se vuelve una lista vacía.

El algoritmo principal:variación general .

El algoritmo principal en la forma anterior funciona para contar y ordenar una sola lista, la variación general permite repetir el proceso sobre una lista de listas, donde cada elemento será contado y ordenado bajo los mismos criterios.

```
listas_sin_orden=[]
```

```
for lista in informacion_en_listas_diferentes:
```


```
    lista_especifica=[]
```

```
    lista_pivote_uno=lista[:]
```

Funcion de conteo añadiendo listas a listas_sin orden.

```
for lista in listas_sin_orden.
```

Función de orden sobre la variable “lista”



El esquema es parecido a lo visto anteriormente, la diferencia principal es que ahora una nueva variable iterada llamada lista, influye sobre la variable lista_pivote_uno, lo cual hace que el algoritmo principal se realice sobre todas las diferentes listas que se encuentran la información separada por listas, encontrada en la variable informacion_en_listas_diferentes.

Solución al problema.

La presente parte del informe tiene como objetivo ofrecer una serie de recomendaciones e indicaciones sobre posibles acciones que en conjunto con la administración de la compañía LifeStore orientándose en la información que es posible recopilar y analizar usando el programa, si bien cabe aclarar que estas recomendaciones deben pensarse como sugerencias basadas en la información que se pudo analizar, y no representan soluciones finales a los problemas de rendimiento de la compañía, promoviendo la constante inversión en esta tecnología para volverlo un sistema más robusto y con la capacidad de realizar tareas más complejas.

1.-La filosofía y enfoque de los productos.

Con el análisis encontrado en los 50 productos más vendidos se encontró en la información provista que las ventas con precios de gama media a alta, tales como procesadores y discos de estado sólido , LifeStore presenta una mayor facilidad para promover u ofrecer de una manera más atractiva a los clientes productos de este tipo de gama, pero esto solo se sabe con respecto a la información provista, en caso de otro tipo de artículos como audífonos, o monitores es posible que exista una deficiencia en el momento de promocionarlos u otro imprevisto, considerando que productos más costos son vendidos con mayor facilidad podemos suponer que no es un problema relacionado al presupuesto del cliente promedio.

Dependiendo de la visión de LifeStore, y su plan de trabajo, será necesario considerar que productos que podrían ser considerados como artículos de menor costo requerirán un mayor volumen para lograr que el valor promedio por venta en el reporte anual corresponde a un valor satisfactorio dentro de los márgenes previstos por la compañía.

También se puede optar por concentrarse en hardware exclusivo de computadoras armables de gama media alta, y mover el enfoque de la marca LifeStore como una tienda confiable y de facilidad para trabajar con ese tipo de productos.

2.- La capacidad de dar un seguimiento mensual con el volumen y valor de las ventas

Mientras se mantenga el formato de las bases de datos en forma de listas, el programa puede seguir utilizándose para crear reportes mensuales, los cuales puede permitir hacer un análisis mensual con respecto al volumen y el valor promedio por venta , esto permitirá ver el impacto de cualquier estrategia que se proponga la empresa con respecto a meses antes de la implementación de estas.

3.-La distribución de las ventas por categoría

Dependiendo del tipo de filosofía y enfoque que tome LifeStore, siempre podrán estar pendientes de aun dentro de los tipos de productos cual está vendiendo más saber exactamente cuales de estos productos son los que están moviendo la mayor cantidad de valor en cada categoría, creando así interes con las marcas que están mostrando un mejor desempeño , además de proveer una forma de pronosticar la cantidad de stock que se requieran , basándose en promedios mensuales.

4.- El valor de las reseñas

Crear una calificación promedio para los productos que ofrece la compañía es una herramienta que permite realizar dos actividades con eficiencia, la primera es encontrar productos de marcas no tan conocidas que resultan un buen producto para los compradores, si estos resultan un buen rendimiento para la compañía puede invertir en estos de una manera más segura con un menor riesgo.

También permite crear un control de daños en caso que al lanzar un producto que tendrá ventas por ser una marca reconocida o de prestigio debido a problemas de fábrica u otro imperfecto comience a tener un pésimo rendimiento en las reseñas de los usuarios, ciertamente el precio de un mal producto no solamente se paga en un valor monetario, sino también en la confianza y prestigio de la clientela tanto nueva como frecuente de LifeStore.



Conclusión

LifeStore representa una compañía que ha logrado mostrar una organización eficiente y pragmática con la decisión de incluir el análisis de datos entre sus herramientas para lograr un mayor desarrollo, esto se refleja también en la organización de sus bases de datos y sus formatos, estas características son clave para la implementación e inversión de análisis de datos más robustos y de mayor complejidad, pienso que la compañía se encuentra en un estado ideal para comenzar a preparar el camino para un enfoque de análisis de este tipo.

Si bien el análisis de datos no corresponde a una solución final, es una de las herramientas más importantes que pueden conseguir con los recursos que existen a la mano, además de representar plusvalía con cada decisión que se tome en la compañía basándose en la información que solamente el análisis de datos puede aportar.