

Quiz 1

Pregunta 1

Contexto del Caso:

Contexto de la Compañía:

AlfaTech es una empresa de desarrollo de software que se especializa en soluciones para la gestión de recursos humanos. La empresa ha estado enfrentando desafíos para mantener la eficiencia operativa mientras escala rápidamente para atender a un mercado más amplio.

Datos Disponibles:

- Ingresos Mensuales
- Horas de Mantenimiento del Sistema
- Tickets de Soporte Mensuales
- Costos de Operaciones
- Satisfacción del Cliente
- Web_Traffic

Problema:

Recientemente, AlfaTech ha experimentado un aumento en los tickets de soporte, lo que ha afectado la satisfacción general del cliente. El equipo directivo desea monitorear un KPI operacional que les permita evaluar y mejorar la eficiencia del soporte técnico.

Pregunta:

¿Cuál de los siguientes KPIs sería el más adecuado para que AlfaTech monitoree la eficiencia del soporte técnico y tome medidas correctivas?

1. Ingresos Mensuales (Ingresos totales generados en un mes)
2. Horas de Mantenimiento del Sistema
(Total de horas dedicadas al mantenimiento de sistemas en un mes)
3. Tickets de Soporte Mensuales
(Número total de tickets de soporte generados por los usuarios en un mes)
4. Costos de Operaciones (Gastos totales relacionados con las operaciones del negocio en un mes)

EL KPI más indicado sería el de **Tickets de Soporte Mensuales**.

Es el más adecuado ya que se relaciona directamente con el problema de la empresa. Si estos tickets aumentan muestra que hay un problema en la eficacia del soporte técnico de la empresa lo cual afecta la satisfacción del cliente quien queda con problemas, también ver una

disminución de estos tickets significa que menos gente pide ayuda lo cual hace que estén más satisfechos, por lo que seguir de cerca este KPI ayudaría a la empresa con su problema. Además un análisis detallado de estos tickets podría identificar problemas comunes para los clientes.

Pregunta 2

Este conjunto de datos representa transacciones de clientes múltiples con diferentes cohortes y varias transacciones. Cada fila es una transacción, donde 'State' puede ser 'Active' (activa) o 'Cancelled' (cancelada). Solo las transacciones activas contribuyen al cálculo del CLTV.

```
In [1]: import pandas as pd
import numpy as np

np.random.seed(42)

dates = pd.date_range(start="2019-01-01", end="2021-12-31", freq='M')
cohorts = ['2019-01', '2019-06', '2020-01', '2020-06', '2021-01']

num_customers = 100
customer_ids = np.arange(1, num_customers + 1)
customer_cohorts = np.random.choice(cohorts, size=num_customers, replace=True)

customers_df = pd.DataFrame({
    'Customer_ID': customer_ids,
    'Cohort': customer_cohorts
})

transactions = []

for index, row in customers_df.iterrows():
    num_transactions = np.random.poisson(5)
    transaction_dates = np.sort(np.random.choice(dates, size=num_transactions, replace=True))

    for date in transaction_dates:
        amount = np.random.normal(100, 20)
        state = 'Active' if np.random.rand() > 0.1 else 'Cancelled'
        transactions.append([row['Customer_ID'], row['Cohort'], date, amount, state])

transactions_df = pd.DataFrame(transactions, columns=['Customer_ID', 'Cohort', 'Transaction_Date', 'Amount', 'State'])
```

Dado el conjunto de datos, calcule el CLTV que llevamos adquirido del cliente con ID 007. Asuma que el margen de beneficio es del 20%.

- **Customer Lifetime Value (CLTV):** Predice el revenue total que un negocio puede esperar de manera razonable de un cliente a través de la relación comercial que tenga con el negocio.
 - Formula:
$$\text{CLTV} = \frac{\text{Average Revenue per User (ARPU)}}{\text{Churn Rate}} \times \text{Gross Margin}$$

```
In [2]: transactions_df
```

Out[2]:

	Customer_ID	Cohort	Transaction_Date	Amount	State
0	1	2020-06	2019-02-28	97.894857	Active
1	1	2020-06	2019-08-31	80.893071	Active
2	1	2020-06	2021-01-31	93.118389	Active
3	1	2020-06	2021-04-30	115.015718	Active
4	1	2020-06	2021-07-31	94.836304	Active
...
519	100	2019-01	2019-06-30	118.678903	Active
520	100	2019-01	2020-02-29	93.426760	Active
521	100	2019-01	2020-03-31	60.479991	Active
522	100	2019-01	2020-08-31	120.557642	Active
523	100	2019-01	2021-06-30	86.949817	Active

524 rows × 5 columns

In [21]: `id_007 = transactions_df.query("Customer_ID == 7 and State == 'Active'")`

In [22]: `id_007`

Out[22]:

	Customer_ID	Cohort	Transaction_Date	Amount	State
31	7	2020-01	2019-06-30	121.935925	Active
32	7	2020-01	2019-07-31	75.235383	Active
33	7	2020-01	2021-01-31	83.506973	Active
35	7	2020-01	2021-07-31	96.322165	Active

In [24]: `arpu = id_007.Amount.sum()`
`arpu`

Out[24]: 377.00044675827917

In [25]: `cltv = arpu * 0.2`
`cltv`

Out[25]: 75.40008935165584