

Segmentaciones del dataset de Ever Peak con Python

January 19, 2026

Segmentación con if, elif y else

Primer segmentación conforme al volumen de compra

```
[1]: import pandas as pd
df = pd.read_csv("/datasets/everpeak_clean.csv")

# Calcular promedio y mediana
cantidad_promedio = df['quantity'].mean()
cantidad_mediana = df['quantity'].median()
print("Promedio:", cantidad_promedio)
print("Mediana:", cantidad_mediana)
print()

# Segmentación con media o promedio
if cantidad_promedio > 22:
    print("En promedio: volumen alto")
else:
    print("En promedio: volumen bajo")

# Segmentación con mediana
if cantidad_mediana > 22:
    print("Según la mediana: volumen alto")
else:
    print("Según la mediana: volumen bajo")
```

Promedio: 32.3598

Mediana: 14.0

En promedio: volumen alto

Según la mediana: volumen bajo

```
[2]: import pandas as pd
df = pd.read_csv("/datasets/everpeak_clean.csv")

# Calcular promedio y mediana
cantidad_promedio = df['quantity'].mean()
cantidad_mediana = df['quantity'].median()
print("Promedio:", cantidad_promedio)
```

```

print("Mediana:", cantidad_mediana)
print()

# Segmentación con media o promedio
if cantidad_promedio > 22:
    print("En promedio: volumen alto")
elif cantidad_promedio >= 10:
    print("En promedio: volumen medio")
else:
    print("En promedio: volumen bajo")

# Segmentación con mediana
if cantidad_mediana > 22:
    print("Según la mediana: volumen alto")
elif cantidad_mediana >= 10:
    print("Según la mediana: volumen medio")
else:
    print("Según la mediana: volumen bajo")

```

Promedio: 32.3598

Mediana: 14.0

En promedio: volumen alto

Según la mediana: volumen medio

Segmentación por edad y poder de compra

```

[3]: import pandas as pd
df = pd.read_csv("/datasets/everpeak_clean.csv")

def classify_volume(row):
    age = row['customer_age'] #extrae la edad
    qty = row['quantity'] #extrae la cantidad

    # 1. Manejo de valores nulos/faltantes
    # pd.isna() verifica de forma robusta si el valor es NaN
    if pd.isna(age) or pd.isna(qty):
        return "Error en Datos"

    # --- 2. Segmentación de Altas Cantidades ---
    if qty > 22:
        if age > 55:
            return "Sr. High Volume"
        else:
            return "Jr. High Volume"

    # --- 3. Segmentación Bajas Cantidades ---

```

```

elif qty <= 22:
    if age > 55:
        return "Sr. Low Volume"
    else:
        return "Jr. Low Volume"

# aplicar función y verificar cambios
df["volume_segment"] = df.apply(classify_volume, axis=1)
print(df['volume_segment'].value_counts())

```

```

Jr. Low Volume      3004
Sr. High Volume     1197
Sr. Low Volume       735
Jr. High Volume      64
Name: volume_segment, dtype: int64

```

Segmentación por metodo de pago y cantidad

```

[4]: import pandas as pd
df = pd.read_csv("/datasets/everpeak_retail.csv")

def classify_payment(row):
    card = row['payment_method']
    qty = row['quantity']

    # 1. Manejo de valores nulos/faltantes
    if pd.isna(card) or pd.isna(qty):
        return "Error en Datos"

    # --- 2. Segmentación de Altas Cantidades ---
    if qty > 22:
        if card == 'credit_card' or card == 'debit_card':
            return "card_high_volume"
        else:
            return "no_card_high_volume"

    # --- 3. Segmentación Bajas Cantidades ---
    elif qty <= 22:
        if card == 'credit_card' or card == 'debit_card':
            return "card_low_volume"
        else:
            return "no_card_low_volume"

# aplicar función y verificar cambios
df['payment_segment'] = df.apply(classify_payment, axis=1)
print(df['payment_segment'].value_counts())

```

```

card_low_volume      2725

```

```
no_card_low_volume      1041
card_high_volume         909
no_card_high_volume      333
Name: payment_segment, dtype: int64
```