

## Informe Final: Tratamiento de Datos y Análisis Exploratorio

Profesional: Carlos Merino Espinoza

Tratamiento de datos: Phytion Google Colaboratory

Dataset: Telecom X (Clientes)

### 1.- Extracción de datos.

Ruta del archivo: /content/telecomx\_data.json

Filas y columnas iniciales: 7267, 6 columnas

Filas y columnas finales: 7032, 23 columnas

Los datos del tipo json, se importaron desde la API al GoogleColab, y se trabajaron como data frame de nombre:

```
datos_telecom=pd.read_json('/content/telecomx_data.json')
```

Se comenzó con DF de 6 columnas que contenía columnas anidadas, las cuales fueron normalizadas, generando 23 columnas de datos.

Se finalizó la extracción con las siguientes características de las columnas del Data frame:

```
datos_telecom.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7267 entries, 0 to 7266
Data columns (total 21 columns):
#   Column                Non-Null Count  Dtype
---  -
0   customerID            7267 non-null   object
1   Churn                  7267 non-null   object
2   gender                 7267 non-null   object
3   SeniorCitizen          7267 non-null   int64
4   Partner                7267 non-null   object
5   Dependents             7267 non-null   object
6   tenure                 7267 non-null   int64
7   PhoneService           7267 non-null   object
8   MultipleLines           7267 non-null   object
9   InternetService        7267 non-null   object
10  OnlineSecurity          7267 non-null   object
11  OnlineBackup            7267 non-null   object
12  DeviceProtection        7267 non-null   object
13  TechSupport             7267 non-null   object
14  StreamingTV             7267 non-null   object
15  StreamingMovies         7267 non-null   object
16  Contract                7267 non-null   object
17  PaperlessBilling        7267 non-null   object
18  PaymentMethod           7267 non-null   object
19  Charges.Monthly         7267 non-null   float64
20  Charges.Total           7267 non-null   object
dtypes: float64(1), int64(2), object(18)
memory usage: 1.2+ MB
```

## 2.- Tratamiento de Datos

### 2.1.- Análisis preliminar.

Para acelerar y ordenar el análisis de la estructura de los datos, se usó un código para revisar todas las columnas y encontrar características, como: caracteres especiales, filas vacías, comas, espacios, Nan o None.

Listado de las columnas para análisis de los datos disponibles

#### - Datos Básicos del Cliente

Columna (Original)	Traducción/Significado en Español	Valores Posibles/Descripción
customerID	<b>ID del cliente</b>	Identificador único (ej: "0002-ORFBO")
Churn	<b>Baja (Rotación)</b>	Yes (Sí), No (No) – Indica si el cliente dejó el servicio
gender	<b>Género</b>	Male (Hombre), Female (Mujer)
SeniorCitizen	<b>Adulto Mayor</b>	0 (No), 1 (Sí) - Si el cliente es mayor de 65 años
Partner	<b>Pareja</b>	Yes (Sí), No (No) - Si tiene pareja (conyuge, pareja estable)
Dependents	<b>Dependientes</b>	Yes (Sí), No (No) - Si tiene personas dependientes (hijos, familiares a cargo)
tenure	<b>Antigüedad (meses)</b>	Número de meses que el cliente ha estado en la compañía

#### - Servicios de Teléfono

Columna (Original)	Traducción/Significado en Español	Valores Posibles/Descripción
PhoneService	<b>Servicio de Teléfono</b>	Yes (Sí), No (No)
MultipleLines	<b>Líneas Múltiples</b>	Yes (Sí), No (No), No phone service (Sin servicio de teléfono)

#### - Servicios de Internet

Columna (Original)	Traducción/Significado en Español	Valores Posibles/Descripción
InternetService	<b>Tipo de Internet</b>	DSL, Fiber optic (Fibra óptica), No (Sin internet)
OnlineSecurity	<b>Seguridad en Línea</b>	Yes (Sí), No (No), No internet service (Sin internet)
OnlineBackup	<b>Copia de Seguridad en Línea</b>	Yes (Sí), No (No), No internet service
DeviceProtection	<b>Protección de Dispositivos</b>	Yes (Sí), No (No), No internet service
TechSupport	<b>Soporte Técnico</b>	Yes (Sí), No (No), No internet service
StreamingTV	<b>Transmisión de TV</b>	Yes (Sí), No (No), No internet service
StreamingMovies	<b>Transmisión de Películas</b>	Yes (Sí), No (No), No internet service

#### - Datos de Contrato y Facturación

Columna (Original)	Traducción/Significado en Español	Valores Posibles/Descripción
Contract	<b>Tipo de Contrato</b>	Month-to-month (Mensual), One year (1 año), Two year (2 años)
PaperlessBilling	<b>Facturación sin Papel</b>	Yes (Sí), No (No)
PaymentMethod	<b>Método de Pago</b>	Electronic check (Cheque electrónico), Mailed check (Cheque por correo), Bank transfer (Transferencia bancaria), Credit card (Tarjeta de crédito)

#### - Datos Financieros

Columna (Original)	Traducción/Significado en Español	Valores Posibles/Descripción
MonthlyCharges	<b>Cargos Mensuales</b>	Monto en dólares (ej: 65.60) - Factura mensual promedio
TotalCharges	<b>Cargos Totales</b>	Monto en dólares (ej: 593.3) - Suma total histórica pag

#### Notas para el análisis:

1. Churn es la variable objetivo: Ideal para predecir abandono de clientes.
2. Columnas binarias: Yes/No pueden convertirse a 1/0 para los análisis posteriores.
3. SeniorCitizen: Es numérica (0/1), pero representa una categoría.
4. Valores especiales:
  - No internet service y No phone service indican que el cliente no tiene ese servicio contratado. Se cambiarán por 'No'.

## 2.2. Limpieza y Normalización

- **Valores nulos:**
  - **Se eliminaron filas con valores vacíos en Churn (224) y Charges.Total (11).**

Para tomar esta decisión se realizó un test estadístico, que sirva de respaldo, para eliminar el 3.2% de los datos, dando positivo para la eliminación.

- La columna seniorcitizen se dejó como boolean, pero posteriormente hubo que cambiarla al tipo Int64, no se realizaron más cambios en esta columna.
- **Normalización de columnas booleanas:**
  - Columnas como partner, dependents, phoneservice se convirtieron a Int64 (1/0) desde texto (yes/no).
- **Limpieza de texto:**
  - Columnas categóricas (paymentmethod, contract) se estandarizaron a minúsculas y sin caracteres especiales. Se eliminaron incluso los paréntesis redondos (). Todos los datos de texto se convirtieron a minúsculas.
  - Se decidió no eliminar los guiones de la columna customerID, ya que se consideró un código alfanumerico válido para identificar clientes.

Se trabajaron todos los datos del data frame para estandarizar y normalizar, cambiando todos los del tipo object:

```
<class 'pandas.core.frame.DataFrame'>
Index: 7032 entries, 0 to 7266
Data columns (total 21 columns):
#   Column                Non-Null Count  Dtype
---  -
0   customerid            7032 non-null   string
1   churn                 7032 non-null   bool
2   gender               7032 non-null   string
3   seniorcitizen         7032 non-null   bool
4   partner              7032 non-null   bool
5   dependents           7032 non-null   bool
6   tenure               7032 non-null   int64
7   phoneservice         7032 non-null   bool
8   multiplelines        7032 non-null   bool
9   internetervice       7032 non-null   string
10  onlinesecurity       7032 non-null   bool
11  onlinebackup         7032 non-null   bool
12  deviceprotection     7032 non-null   bool
13  techsupport          7032 non-null   bool
14  streamingtv          7032 non-null   bool
15  streamingmovies      7032 non-null   bool
16  contract             7032 non-null   string
17  paperlessbilling     7032 non-null   bool
18  paymentmethod        7032 non-null   string
19  charges.monthly      7032 non-null   float64
20  charges.total        7032 non-null   float64
dtypes: bool(13), float64(2), int64(1), string(5)
memory usage: 583.7 KB
```

### 3. Análisis Exploratorio (Gráficos Clave)

#### 3.1.- Últimas depuraciones de datos.

Primero se completó la tarea de cambiar todos los datos tipo True, False a 1,0:

Tipos de datos y valores únicos:

```
churn          int64
partner        int64
dependents     int64
phoneservice   int64
paperlessbilling int64
onlinesecurity int64
onlinebackup   int64
deviceprotection int64
techsupport    int64
streamingtv    int64
streamingmovies int64
multiplelines  int64
dtype: object
```

Y se dejó para el final la columna seniorcitizen:

```
<class 'pandas.core.frame.DataFrame'>
Index: 7032 entries, 0 to 7266
Data columns (total 1 columns):
#   Column          Non-Null Count  Dtype
---  ---
0   seniorcitizen    7032 non-null   Int64
dtypes: Int64(1)
memory usage: 116.7 KB
None
```

También se creó la columna cuentas diarias:

```
datos_telecom['cuentas_diarias'] = datos_telecom['charges.monthly'] / 30
```

Se usaron 30 días, pero se puede ajustar si se requiere más precisión

#### 3.1.- Graficar churn y relaciones con otras variables

Solo para comenzar a visualizar la naturaleza del problema:

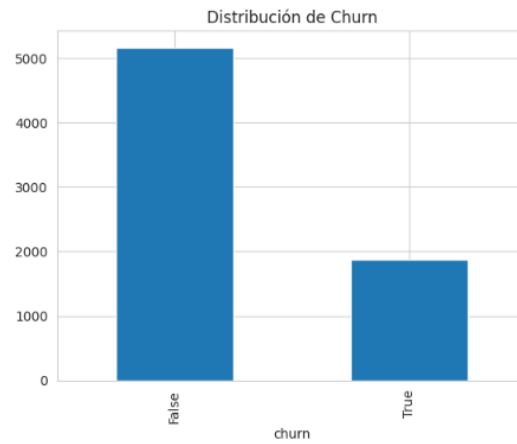


Gráfico de distribución de cuentas diarias

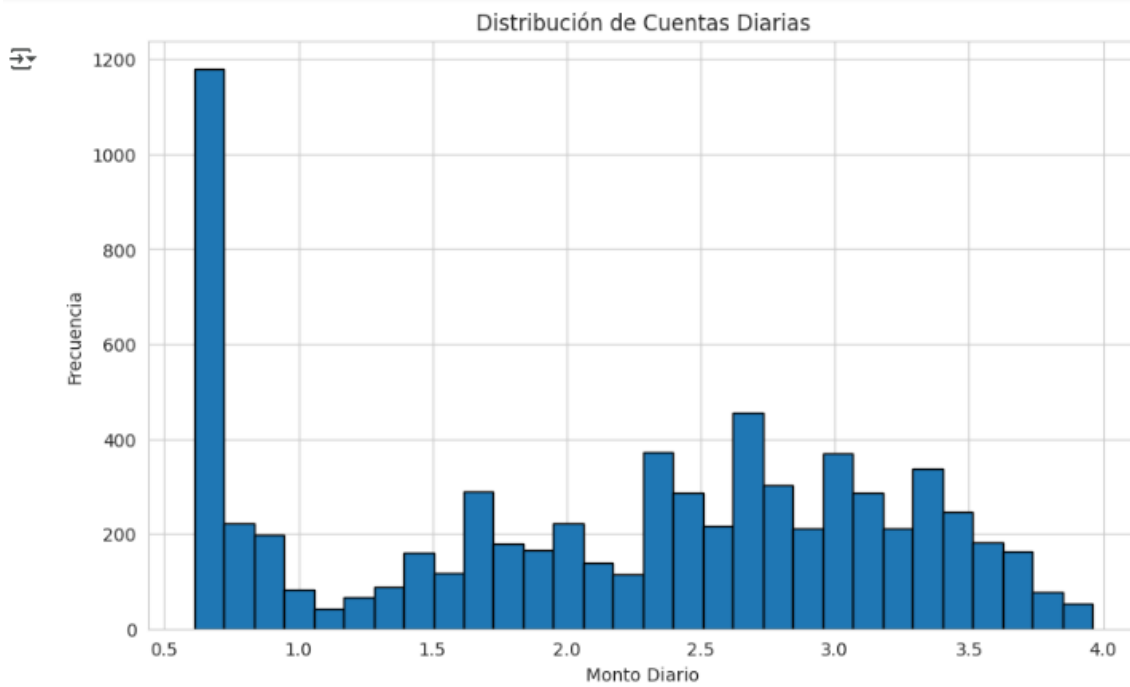
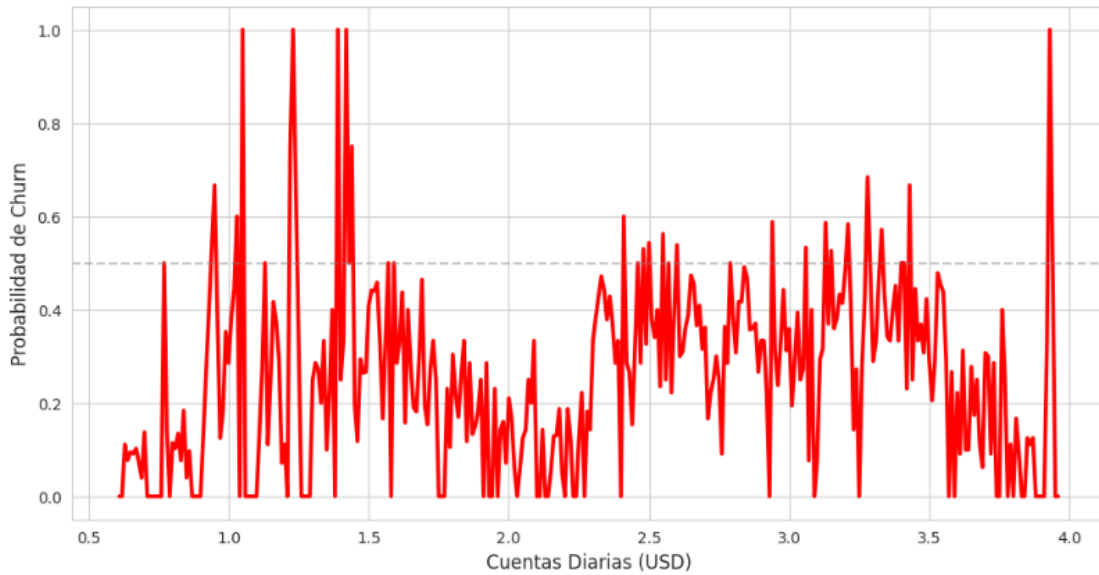


Gráfico Cuentas diarias vs Tasa de Churn vs.

17

Relación entre Cuentas Diarias y Tasa de Churn



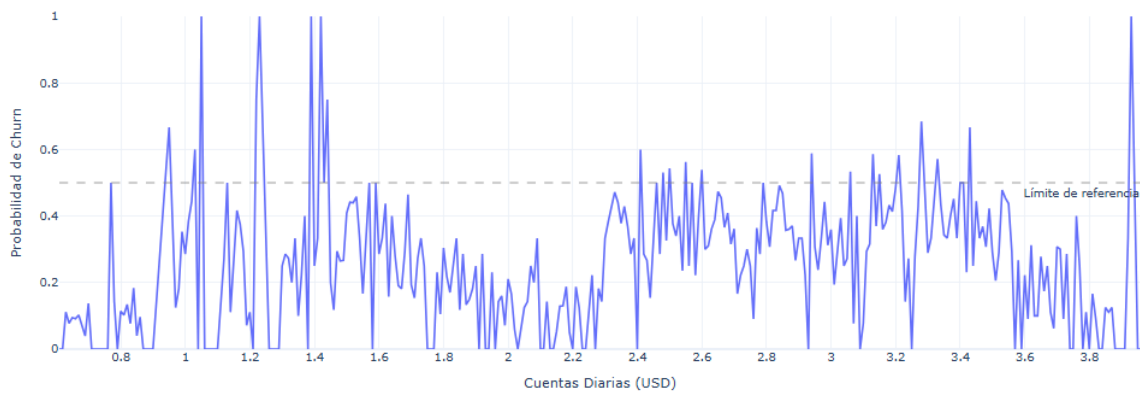
- **Conjeturas preliminares:**

- Clientes con gastos diarios bajos, mayores a 0.9 y menores a 1.46 USD tienen mayor probabilidad de abandono.
- Clientes con gastos diarios muy altos, mayores a 3.91 USD tienen mayor probabilidad de abandono.
- Los clientes de los rangos intermedios a los valores muy altos y muy bajos, se encuentran bajo la probabilidad de 0.5 de tasa de abandono.
- Posible causa: Los que pagan precios gastos diarios muy bajos, son más sensibles a las ofertas por precio que entrega el mercado, siendo menos fieles a la marca, y se propensos a cambiarse ante cualquier oferta menor en el mercado.
- Posible causa: Los que pagan precios muy altos, tienen más recursos y pueden elegir opciones que consideran de más alta calidad, siendo poco propensos a fidelizar por otros factores de la marca.

Para trabajar mejor, se mejoró el gráfico, otorgándole carácter interactivo:



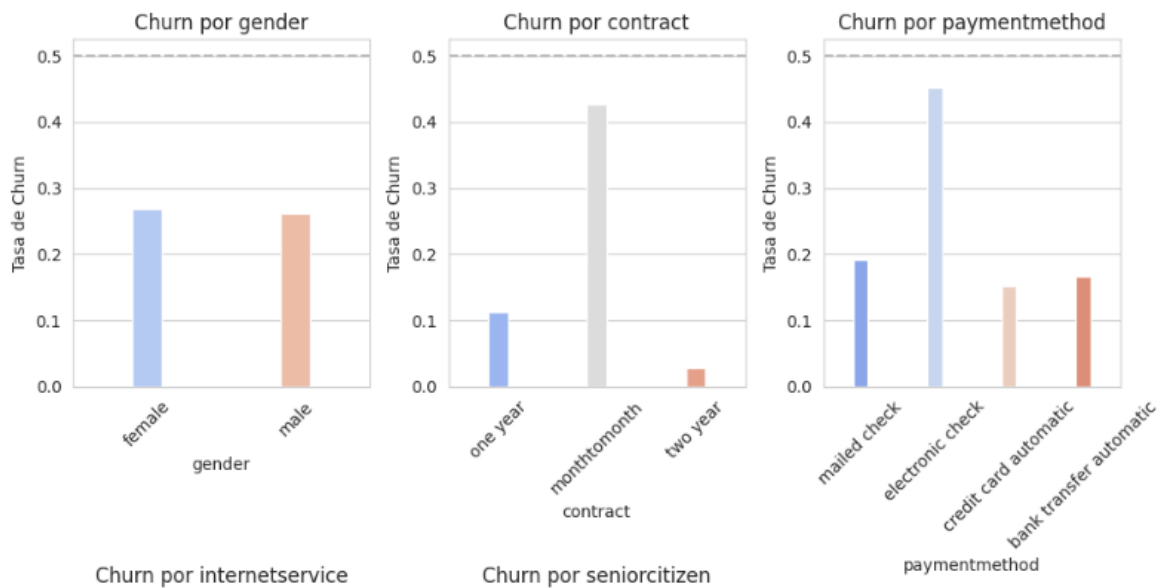
Relación entre Cuentas Diarias y Tasa de Churn (Interactivo)

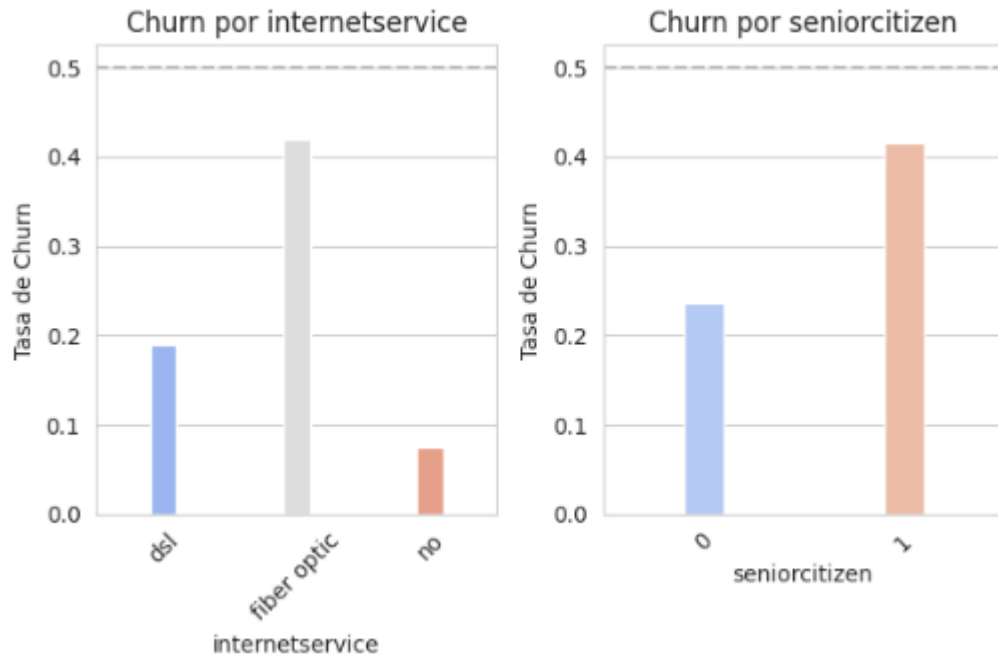


Finalmente se utilizaron más variables, para mejorar evaluar mejor las conjeturas:

Gráficos Lista de columnas categóricas a analizar

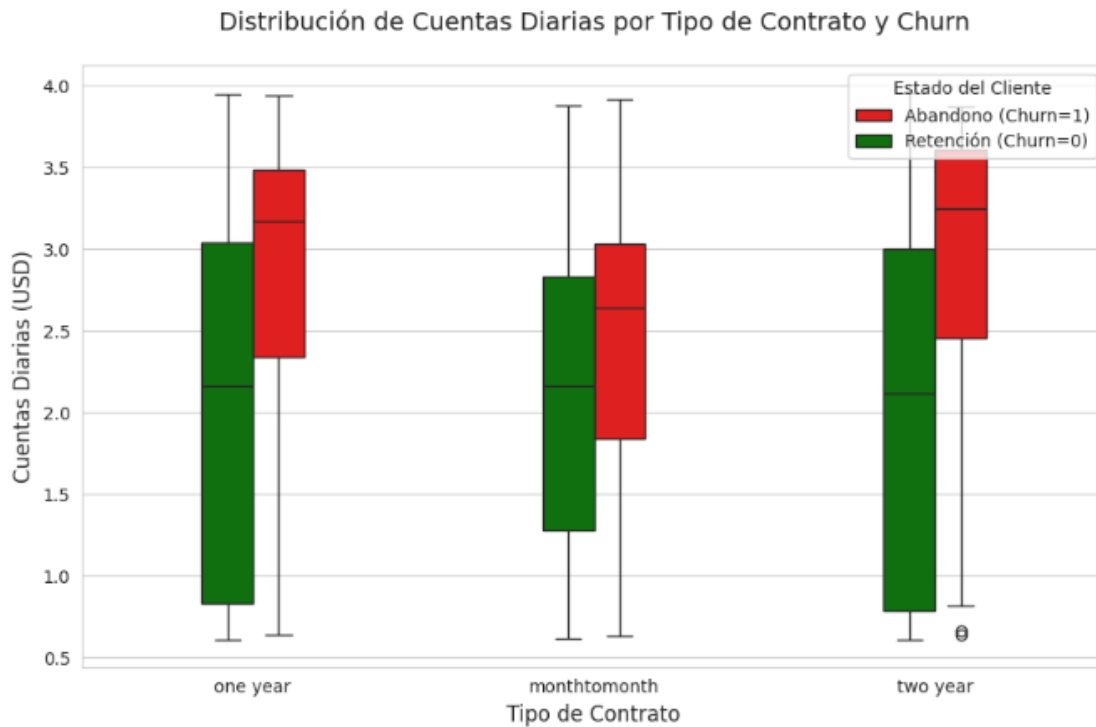
```
columns_categoricas = ['gender', 'contract', 'paymentmethod', 'internetservice', 'seniorcitizen']
```





- Hallazgos:
  - Clientes con contratos month-to-month tienen una tasa de churn mayor al resto, con aproximadamente el 65% (vs. ~15% en contratos anuales). Los contratos a dos años tiene las tasas de abandono más bajas (menor al 5%)
  - No existe una tendencia diferenciadora del comportamiento de abandono, en cuanto al género.
  - El método de pago pago con la tasa d abandono más elevada es electronic check.
  - Los senior citizen presentan una tendencia más marcada al abandono del servicio.
  - Finalmente, los clientes con fibra óptica son los que tiene mayores tasas de abandono.

Gráfico Distribución Boxplot de Cuentas Diarias por Tipo de Contrato y Churn



El gráfico permite visualizar que los contratos tipo mes a mes, generan la menor retención de clientes.

#### 4. Observaciones

- Variables críticas para retención:

- Contrato mensual, fibra óptica y métodos de pago tipo electronic check son los mayores predictores de churn.
- Ideal cambiar contratos a un año o dos años de contrato.
- Paquetes personalizados para adultos mayores (seniorcitizen=1), que muestran un 40% de churn.

## 5. Recomendaciones preliminares para ajustar los análisis.

- Campaña de retención con premios, para clientes con:
  - Contratos tipo mes a mes, que cambien el tipo de contrato a anual o bianual
  - Realizar un análisis detallado de interacción con otras variables, en los siguientes rangos de gastos diarios, para evaluar posibles vías de retención y evitar futuros clientes de estas características:

$$0.9 \leq \text{Gasto diario} \leq 1.46 \text{ USD}$$

$$3.9 < \text{Gasto diario} < 3.94 \text{ USD}$$

- Evaluar el servicio de fibra óptica, directamente con los clientes.
- Evaluar a los senior citizen, para entender, mejor su comportamiento.

### Herramientas Utilizadas

- Python (matplotlib, seaborn, numpy, plotly.express, pandas)

Nota: Todos los análisis mantuvieron la estructura de DataFrame, evitando conversiones a Series. Inicialmente, este aspecto generó serios retrasos en la entrega del challenge, ya que en las clases no se aclaró que ciertos comandos cambian los datos del data frame a series, generando que en algún momento el código se haga inviable, por mezcla de instrucciones en uno u otro sentido. Se generaron 4 informes hasta llegar al informe final, ya que además las soluciones a los errores eran difíciles de encontrar. **Retroalimentar a los alumnos respecto a esto.**