

Análisis de Delitos en Tierra del Fuego: Guía de ETL y Modelado

Guía Completa para la Implementación del Análisis de Delitos en Tierra del Fuego

Objetivo:

Implementar un modelo de análisis de series temporales para detectar patrones estacionales en la ocurrencia de diferentes tipos de delitos en la Provincia de Tierra del Fuego. Se utilizarán técnicas de aprendizaje automático y análisis estadístico para ayudar en la toma de decisiones sobre la asignación de recursos policiales.

1. Transformación del Dataset (ETL):

- Se realiza una limpieza de los datos eliminando columnas sin información.
- Se renombran las columnas para una mayor claridad y comprensión.
- Se eliminan filas con valores irrelevantes o títulos y se rellenan los valores faltantes con ceros.

Código:

```
```python
```

```
import pandas as pd
```

```
from sklearn.preprocessing import MinMaxScaler
```

```
Cargar el archivo Excel
```

```
file_path
```

```
=
```

```
 '/mnt/data/09_03_02_Delitos-con-intervencion-policial-por-mes-segun-tipo-de-delito.-1.xlsx'
```

```
xls = pd.ExcelFile(file_path)
```

```
Leer los datos de la hoja 'Tierra Del Fuego AelAS'
```

```
delitos_tierra = pd.read_excel(xls, sheet_name='Tierra Del Fuego AelAS')
```

```
Limpiar los datos: eliminar columnas vacías
```

```
delitos_tierra_cleaned = delitos_tierra.dropna(axis=1, how='all')
```

```
Renombrar columnas
```

```
new_column_names = ['Tipo_Delito'] + [f"{year}_{month}" for year in range(2014, 2025) for month in
['ene', 'feb', 'mar', 'abr', 'may', 'jun', 'jul', 'ago', 'sept', 'oct', 'nov', 'dic']]
```

```
new_column_names = new_column_names[:len(delitos_tierra_cleaned.columns)]
```

```
delitos_tierra_cleaned.columns = new_column_names
```

```
Rellenar valores faltantes con ceros
```

```
delitos_tierra_cleaned = delitos_tierra_cleaned.fillna(0)
```

```
Eliminar filas que no son parte de los datos reales
```

```
delitos_tierra_cleaned = delitos_tierra_cleaned[delitos_tierra_cleaned['Tipo_Delito'].str.contains('Total Hechos delictuosos',
na=False) | pd.to_numeric(delitos_tierra_cleaned['Tipo_Delito'], errors='coerce').notna()]
```

```
Eliminar filas que contienen solo ceros
```

```
delitos_tierra_cleaned = delitos_tierra_cleaned[(delitos_tierra_cleaned.iloc[:, 1:] != 0).any(axis=1)]
```

```
delitos_tierra_cleaned.reset_index(drop=True, inplace=True)
```

```
...
```

## 2. Preparación para Análisis de Series Temporales:

- Se convierte el dataset en un formato adecuado para el análisis de series temporales.

- Se extraen los datos numéricos y se normalizan utilizando `MinMaxScaler` para asegurar que los datos se encuentren en un rango apropiado para el modelado.

Código:

```
```python
# Extraer datos numéricos para normalización
numeric_data = delitos_tierra_cleaned.iloc[:, 1:].values

# Normalizar datos usando MinMaxScaler
scaler = MinMaxScaler(feature_range=(0, 1))
normalized_data = scaler.fit_transform(numeric_data)

# Convertir datos normalizados de nuevo a DataFrame para un análisis más fácil
delitos_tierra_normalized = pd.DataFrame(
    normalized_data,
    columns=[f'{year}_normalized' for year in range(2014, 2025)],
    index=delitos_tierra_cleaned['Tipo_Delito']
)
```
```

### 3. Implementación del Modelo de Series Temporales (ARIMA):

- Se implementa un modelo ARIMA para analizar patrones en el total de delitos registrados.
- El objetivo es identificar tendencias y patrones estacionales a lo largo del tiempo.

Código:

```
```python
```

```
import statsmodels.api as sm
```

```
# Extraer la serie de tiempo total de delitos
```

```
total_delitos_series = pd.to_numeric(delitos_tierra_cleaned.iloc[0, 1:], errors='coerce').values
```

```
# Ajustar el modelo ARIMA (parámetros p=1, d=1, q=1)
```

```
model = sm.tsa.ARIMA(total_delitos_series, order=(1, 1, 1))
```

```
results = model.fit()
```

```
# Resumen del modelo ARIMA
```

```
arima_summary = results.summary()
```

```
print(arima_summary)
```

```
...
```

Conclusión:

Con este flujo de trabajo, se puede llevar a cabo un análisis profundo de los patrones de delitos en la provincia de Tierra del Fuego. La limpieza y transformación de datos aseguran una mejor calidad para el modelado, mientras que el uso de ARIMA permite identificar tendencias que podrían ayudar a la planificación de recursos policiales.

SARIMAX Results

=====

==

Dep. Variable:	y	No. Observations:	127
----------------	---	-------------------	-----

Model:	ARIMA(1, 1, 1)	Log Likelihood	-739.265
--------	----------------	----------------	----------

Date:	Sat, 26 Oct 2024	AIC	1484.530
-------	------------------	-----	----------

Time: 17:39:21 BIC 1493.038
Sample: 0 HQIC 1487.986
- 127

Covariance Type: opg

=====

==

	coef	std err	z	P> z	[0.025	0.975]

ar.L1	0.2054	0.132	1.557	0.120	-0.053	0.464
ma.L1	-0.7438	0.091	-8.197	0.000	-0.922	-0.566
sigma2	7272.7787	881.121	8.254	0.000	5545.813	8999.744

=====

=====

Ljung-Box (L1) (Q): 0.02 Jarque-Bera (JB): 1.71
Prob(Q): 0.88 Prob(JB): 0.42
Heteroskedasticity (H): 1.80 Skew: -0.18
Prob(H) (two-sided): 0.06 Kurtosis: 3.44

=====

=====

Warnings:

[1] Covariance matrix calculated using the outer product of gradients (complex-step).