HENRY

Data Science

M4L9 | Análisis de series temporales

→ soyhenry.com

Objetivos

- Analizar los componentes esenciales de las series de tiempo, diferenciando tendencia, estacionalidad y ruido para interpretar patrones temporales en los datos.
- Determinar la estacionariedad de una serie mediante pruebas estadísticas y evaluar la autocorrelación con funciones ACF y PACF para seleccionar modelos adecuados.
- Implementar modelos ARIMA, SARIMA y Prophet, además de enfoques basados en machine learning como Random Forest y XGBoost, optimizando la predicción de series temporales.

#TEMAS

Agenda

COMENCEMOS →

.01 Definición y componentes

.02 Modelos ARIMA y SARIMA

.03 Modelos aditivos: Prophet

.04 Random Forest u XGBoost para forecasting

.05 Avance de PI

<-->

¿Qué vimos en la lecture?

<01>

Definición y componentes

Qué es una serie temporal

- Observaciones ordenadas en el tiempo (diarias, semanales, mensuales).
- Pueden ser estacionarias (media/varianza constantes) o no estacionarias.
- En no estacionarias hay tendencia, estacionalidad o cambios estructurales.
- Reconocer la estructura define el modelado y las transformaciones previas.

Estacionariedad vs No-estacionariedad

- Estacionaria: media, varianza y autocorrelación estables en el tiempo.
- No estacionaria: tendencia/estacionalidad que cambian la media periódicamente.
- ARIMA asume estacionariedad → requerirá diferenciar si no la hay.

Componentes y descomposición

- Tendencia: dirección de largo plazo (CityScoot crece con usuarios).
- Estacionalidad: repeticiones regulares (picos fin de semana, baja en invierno).
- Ciclo: fluctuaciones largas ligadas a economía u otros factores.
- Ruido: variación aleatoria; descomponer ayuda a interpretar y modelar.

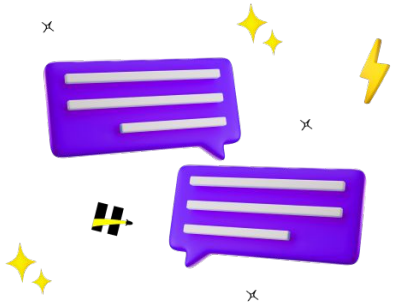
ACF y PACF

- ACF: similitud serie–pasado en múltiples rezagos; revela periodicidades.
- PACF: relación con rezagos “netos”, quitando efectos intermedios.
- Guían órdenes p (AR) y q (MA) en ARIMA/SARIMA.
- CityScoot: picos en lags de 7 días ⇒ estacionalidad semanal.

<02>

Modelos ARIMA y SARIMA

AR, MA y ARMA



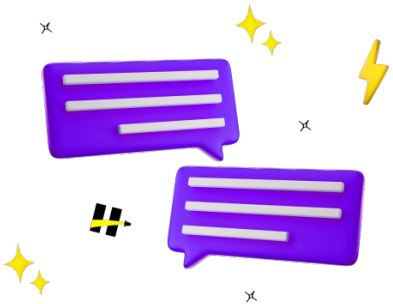
- AR(p): el valor actual depende linealmente de p valores pasados.
- MA(q): depende de q errores pasados (shocks transitorios).
- ARMA(p,q): combina memoria en valores y en errores; exige estacionariedad.
- Útiles sin tendencia/estacionalidad marcada.

ARIMA: integrar con diferenciación

- ARIMA(p,d,q): “d” diferencias para estabilizar tendencia.
- En CityScoot, diferencia de orden 1 elimina crecimiento sostenido.
- Luego modelar dinámica de corto plazo con AR/MA.
- Proceso: identificar (ACF/PACF) → estimar → diagnosticar residuos.

SARIMA: estacionalidad explícita

- SARIMA(p,d,q)(P,D,Q)s añade términos estacionales y periodo s.
- Captura picos/ciclos recurrentes (ej. s=7 para patrón semanal).
- Evita confundir estacionalidad con tendencia.
- Adecuado cuando coexisten tendencia y periodicidad.



Elección y buenas prácticas

- AR/MA: series estacionarias simples; ARIMA: tendencia; SARIMA: ciclos.
- Parsimonia: preferir el modelo más simple con buen desempeño (AIC/BIC).
- Revisar estacionariedad tras diferenciar y autocorrelación en residuos.
- Comparar MAE/RMSE entre configuraciones antes de decidir.

<03>

Modelos aditivos: Prophet

Prophet: modelo aditivo

- Descompone como y(t)= tendencia g(t) + estacionalidad s(t) + eventos h(t) + ϵ.
- Tendencia piecewise (changepoints), estacionalidades con series de Fourier.
- Soporta feriados/eventos explícitos de negocio.
- Automatiza pasos que en SARIMA requieren diagnóstico manual.

Prophet vs SARIMA

- SARIMA: enfoque estadístico en rezagos; requiere (estacional) estacionarizar.
- Prophet: enfoque estructural e interpretable (tendencia/estacionalidad/eventos).
- SARIMA da control fino; Prophet favorece rapidez y mantenimiento.
- Usarlos en conjunto permite contrastar y validar.

<04>

Random Forest u XGBoost para forecasting

Forecasting supervisado: enfoque

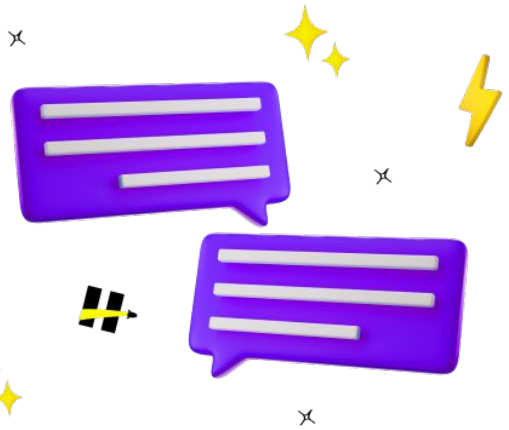
- Reformular como regresión: filas=fechas, y=futuro, X=features.
- Incorporar contexto externo (clima, marketing, feriados).
- Random Forest/XGBoost aprenden relaciones no lineales/multivariadas.
- Más sensibles a cambios operativos que los modelos puramente autorregresivos.

Ingeniería de features

- Lags: t−1, t−7, t−14 capturan inercia y patrón semanal.
- Ventanas móviles: medias/desvíos (rolling) para tendencia local.
- Calendario: día de semana, mes, fin de semana/feriado.
- Regresores externos: temperatura, lluvia, eventos, gasto en marketing.

Horizonte de predicción

- Directa: un modelo por horizonte (t+1, t+7) → evita acumulación de error.
- Recursiva: un modelo 1-paso y se encadena → eficiente pero propaga error.
- CityScoot: recursiva para 1–3 días; directa para 2 semanas/1 mes.
- Elegir según costo computacional y estabilidad del patrón.



Validación temporal y leakage

- Nunca mezclar pasado y futuro al partir datos (no aleatorio).
- Expanding/sliding window preservan la dirección del tiempo.
- Cuidar ventanas móviles: calcular solo con datos previos al corte.
- Métricas (MAE/RMSE) sobre bloques futuros reflejan desempeño real.

Comparación con enfoques clásicos

- Clásicos: descomposición clara (tendencia/estacionalidad) y control paramétrico.
- ML: integra múltiples fuentes y capta no linealidades/interacciones.
- Interpretación vía importancia de variables/SHAP.
- Complementarios, no excluyentes, según datos y objetivo.

<DATA SCIENCE/>

Vayamos a la práctica

Avance de PI

Consigna

FinanceGuard, el banco digital en crecimiento acelerado, continúa enfrentando el desafío de reducir su alta tasa de abandono de clientes. Tras los avances previos centrados en modelos supervisados (Regresión Logística y Gradient Boosting) y análisis no supervisado (clustering y reducción de dimensionalidad), este cuarto avance marca la etapa de cierre del proyecto. En esta fase, el estudiante debe integrar los hallazgos de todo el proceso para construir una visión global que combine desempeño técnico y valor estratégico para el negocio.

En este avance asumes el **rol de integrador analítico.** Tu **responsabilidad** es consolidar los resultados de los modelos desarrollados en los avances anteriores, comparar su rendimiento y sintetizar los aprendizajes obtenidos. Además, deberás plasmar los insights más relevantes en un reporte técnico que sirva como documento final de comunicación con el equipo de retención del banco.

Tareas a realizar

En Reporte_Modelos.pdf, tendrás que **consolidar y comparar los resultados obtenidos en los avances anteriores, agregar las visualizaciones necesarias, insights de negocio y recomendaciones estratégicas**

1. **Síntesis de resultados por avance:**

- **Avance 1 - Regresión Logística:**
  - Performance del modelo baseline
  - Interpretabilidad y coeficientes más importantes
  - Fortalezas y limitaciones identificadas
- **Avance 2 - Gradient Boosting:**
  - Mejor modelo de boosting identificado
  - Feature importance del mejor modelo
  - Ganancia en performance vs modelo baseline
- **Avance 3 - Aprendizaje No Supervisado:**
  - Segmentos de clientes identificados
  - Insights de negocio por cluster
  - Features derivadas del clustering

2. **Lecciones aprendidas:**

- ¿Cuándo usar modelos supervisados vs no supervisados?
- Consideraciones para futuros proyectos de churn

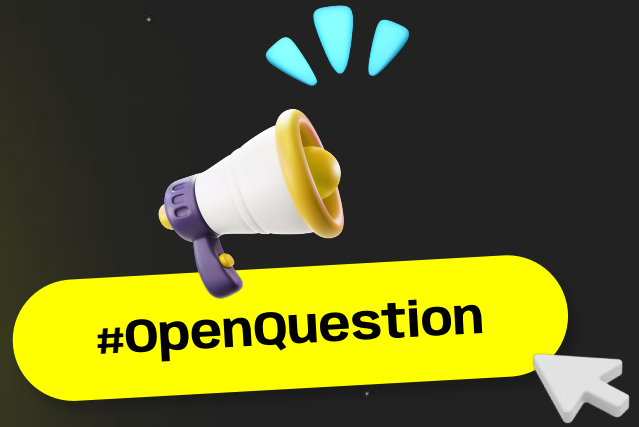
Extra credit

Como parte de las clases de aprendizaje supervisado, implementar los siguientes análisis en **4_Extra_credit.ipynb**, para el mejor modelo supervisado, es decir, el de mejor resultado según métrica de evaluación:

Optimización de threshold personalizada:

- Optimización del punto de corte según métricas de negocio
- Matriz de confusión con costos personalizados

HENRY



¿Preguntas?

HENRY

¡Muchas gracias!

→ soyhenry.com