

ACÀMICA

TEMA DEL DÍA

## **Series de tiempo**

Cuando podemos ver cómo una magnitud cambia en el tiempo, los análisis se hacen más interesantes y, a veces, complejos.

También podemos intentar predecir el futuro, pero te recomendamos no apostar fuerte a esas predicciones.



# Agenda

---

Daily

Explicación: Predicción a Futuro.

**Break**

Hands-on training

Proyecto

Cierre



# Daily



Daily



## Sincronizando...

### Bitácora



¿Cómo te ha ido?  
¿Obstáculos?  
¿Cómo seguimos?

### Challenge



¿Cómo te ha ido?  
¿Obstáculos?  
¿Cómo seguimos?

# Predicción a Futuro



# Train Test Split y Validación Cruzada

## Sliding window vs. forward chaining cross validation

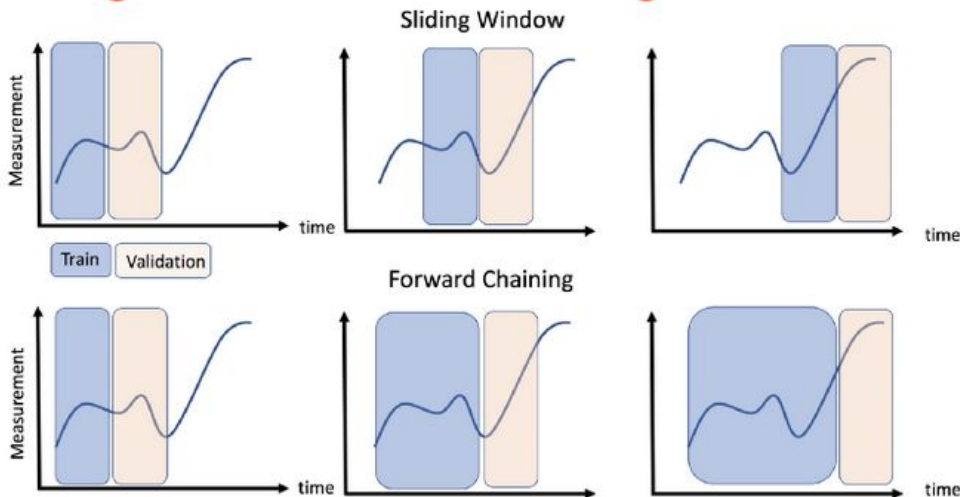


Fig. 5) Basically, there are two kinds of cross-validation for the time series sliding window and forward chaining. In this post, we will consider forward chaining cross-validation method

# Benchmarks

- Predecir el valor anterior.
- Predecir el promedio de  $n$  valores anteriores.
- Regresión lineal o polinómica con  $n$  valores anteriores.



# Estrategia 1 • One-Step prediction

- N: largo de la serie
- n: look back, equivalente a cantidad de features

$y_1$	$y_2$	$y_3$	...	$y_{n-1}$	$y_n$	$y_{n+1}$
-------	-------	-------	-----	-----------	-------	-----------

# Estrategia 1 • One-Step prediction

- N: largo de la serie
- n: look back, equivalente a cantidad de features

$y_1$	$y_2$	$y_3$	...	$y_{n-1}$	$y_n$	$y_{n+1}$
$y_2$	$y_3$	$y_4$	...	$y_n$	$y_{n+1}$	$y_{n+2}$

# Estrategia 1 • One-Step prediction

- N: largo de la serie
- n: look back, equivalente a cantidad de features

$y_1$	$y_2$	$y_3$	...	$y_{n-1}$	$y_n$	$y_{n+1}$
$y_2$	$y_3$	$y_4$	...	$y_n$	$y_{n+1}$	$y_{n+2}$
$y_3$	$y_4$	$y_5$	...	$y_{n+1}$	$y_{n+2}$	$y_{n+3}$

# Estrategia 1 • One-Step prediction

- N: largo de la serie
- n: look back, equivalente a cantidad de features

$y_1$	$y_2$	$y_3$	...	$y_{n-1}$	$y_n$	$y_{n+1}$
$y_2$	$y_3$	$y_4$	...	$y_n$	$y_{n+1}$	$y_{n+2}$
$y_3$	$y_4$	$y_5$	...	$y_{n+1}$	$y_{n+2}$	$y_{n+3}$
...	...	...	...	...	...	...
$y_{N-n-2}$	$y_{N-n-1}$	$y_{N-n}$	...	$y_{N-3}$	$y_{N-2}$	$y_{N-1}$
$y_{N-n-1}$	$y_{N-n}$	$y_{N-n+1}$	...	$y_{N-2}$	$y_{N-1}$	$y_N$

# Estrategia 1 • One-Step prediction

- N: largo de la serie
- n: look back, equivalente a cantidad de features

$y_1$	$y_2$	$y_3$	...	$y_{n-1}$	$y_n$	$y_{n+1}$
$y_2$	$y_3$	$y_4$	...	$y_n$	$y_{n+1}$	$y_{n+2}$
$y_3$	$y_4$	$y_5$	...	$y_{n+1}$	$y_{n+2}$	$y_{n+3}$
...	...	...	...	...	...	...
$y_{N-n-2}$	$y_{N-n-1}$	$y_{N-n}$	...	$y_{N-3}$	$y_{N-2}$	$y_{N-1}$
$y_{N-n-1}$	$y_{N-n}$	$y_{N-n+1}$	...	$y_{N-2}$	$y_{N-1}$	$y_N$

# Estrategia 1 • One-Step prediction

- N: largo de la serie
- n: look back, equivalente a cantidad de features

$y_1$	$y_2$	$y_3$	...	$y_{n-1}$	$y_n$	$y_{n+1}$
$y_2$	$y_3$	$y_4$	...	$y_n$	$y_{n+1}$	$y_{n+2}$
$y_3$	$y_4$	$y_5$	...	$y_{n+1}$	$y_{n+2}$	$y_{n+3}$
...	...	...	...	...	...	...
$y_{N-n-2}$	$y_{N-n-1}$	$y_{N-n}$	...	$y_{N-3}$	$y_{N-2}$	$y_{N-1}$
$y_{N-n-1}$	$y_{N-n}$	$y_{N-n+1}$	...	$y_{N-2}$	$y_{N-1}$	$y_N$

$X$   
Shape  $X = (N - n - 1, n)$

$y$   
Shape  $y = (N - n - 1,)$

# Estrategia 1 • One-Step prediction

- N: largo de la serie
- n: look back, equivalente a cantidad de features

$y_1$	$y_2$	$y_3$	...	$y_{n-1}$	$y_n$	$y_{n+1}$	Train
$y_2$	$y_3$	$y_4$	...	$y_n$	$y_{n+1}$	$y_{n+2}$	
$y_3$	$y_4$	$y_5$	...	$y_{n+1}$	$y_{n+2}$	$y_{n+3}$	
...	...	...	...	...	...	...	Test
$y_{N-n-2}$	$y_{N-n-1}$	$y_{N-n}$	...	$y_{N-3}$	$y_{N-2}$	$y_{N-1}$	
$y_{N-n-1}$	$y_{N-n}$	$y_{N-n+1}$	...	$y_{N-2}$	$y_{N-1}$	$y_N$	
X						y	
Shape X = (N - n - 1, n)						Shape y = (N - n - 1,)	

# Estrategia 1 • One-Step prediction

- N: largo de la serie
- n: look back, equivalente a cantidad de features

Es un problema de regresión

$y_1$	$y_2$	$y_3$	...	$y_{n-1}$	$y_n$	$y_{n+1}$	Train
$y_2$	$y_3$	$y_4$	...	$y_n$	$y_{n+1}$	$y_{n+2}$	
$y_3$	$y_4$	$y_5$	...	$y_{n+1}$	$y_{n+2}$	$y_{n+3}$	
...	...	...	...	...	...	...	
$y_{N-n-2}$	$y_{N-n-1}$	$y_{N-n}$	...	$y_{N-3}$	$y_{N-2}$	$y_{N-1}$	Test
$y_{N-n-1}$	$y_{N-n}$	$y_{N-n+1}$	...	$y_{N-2}$	$y_{N-1}$	$y_N$	
X						y	
Shape X = (N - n - 1, n)						Shape y = (N - n - 1,)	



# Estrategia 2 • Multi-Step prediction

## **Estrategia Recursiva**

Una vez que contamos con un predictor a un paso YA entrenado, vamos agregando a las variables predictoras el resultado de esa predicción.

# Estrategia 2 • Multi-Step prediction

## Estrategia Recursiva

Una vez que contamos con un predictor a un paso YA entrenado, vamos agregando a las variables predictoras el resultado de esa predicción.

$y_1$	$y_2$	$y_3$	...	$y_{n-1}$	$y_n$	$y_{n+1}^{pred}$
$y_2$	$y_3$	$y_4$	...	$y_n$	$y_{n+1}^{pred}$	$y_{n+2}^{pred}$
$y_3$	$y_4$	$y_5$	...	$y_{n+1}^{pred}$	$y_{n+2}^{pred}$	$y_{n+3}^{pred}$
...	...	...	...	...	...	...

# Estrategia 2 • Multi-Step prediction

## Estrategia Recursiva

Una vez que contamos con un predictor a un paso YA entrenado, vamos agregando a las variables predictoras el resultado de esa predicción.

**Ventaja:** solo hay que entrenar un modelo.

**Desventaja:** cuando más nos alejamos de los datos medidos, más probable es se acumulen los errores.

$y_1$	$y_2$	$y_3$	...	$y_{n-1}$	$y_n$	$y_{n+1}^{pred}$
$y_2$	$y_3$	$y_4$	...	$y_n$	$y_{n+1}^{pred}$	$y_{n+2}^{pred}$
$y_3$	$y_4$	$y_5$	...	$y_{n+1}^{pred}$	$y_{n+2}^{pred}$	$y_{n+3}^{pred}$
...	...	...	...	...	...	...

Predicciones  
(NO etiquetas de  
entrenamiento)

# Estrategia 2 • Multi-Step prediction

## Estrategia Recursiva

Una vez que contamos con un predictor a un paso YA entrenado, vamos agregando a las variables predictoras el resultado de esa predicción.

**Ventaja:** solo hay que entrenar un modelo.

**Desventaja:** cuando más nos alejamos de los datos medidos, más probable es se acumulen los errores.

$y_1$	$y_2$	$y_3$	...	$y_{n-1}$	$y_n$	$y_{n+1}^{pred}$	$y_{n+2}^{pred}$	2° predicción
$y_2$	$y_3$	$y_4$	...	$y_n$	$y_{n+1}^{pred}$	$y_{n+2}^{pred}$	$y_{n+3}^{pred}$	3° predicción
$y_3$	$y_4$	$y_5$	...	$y_{n+1}^{pred}$	$y_{n+2}^{pred}$	$y_{n+3}^{pred}$	$y_{n+4}^{pred}$	4° predicción
...	...	...	...	...	...	...		

# Multi-Step prediction

## Estrategia Directa


Creamos un modelo por cada paso que queremos predecir.

Por ejemplo, si queremos predecir tres pasos:

**Ventaja:** no acumula error.

**Desventaja:** hay que crear un modelo por cada paso.

$y_1$	$y_2$	$y_3$	...	$y_{n-1}$	$y_n$	$y_{n+1}$	$y_{n+2}$	$y_{n+3}$
$y_2$	$y_3$	$y_4$	...	$y_n$	$y_{n+1}$	$y_{n+2}$	$y_{n+3}$	$y_{n+4}$
$y_3$	$y_4$	$y_5$	...	$y_{n+1}$	$y_{n+2}$	$y_{n+3}$	$y_{n+4}$	$y_{n+5}$
...	...	...	...	...	...	...	...	...
$y_{N-n-2}$	$y_{N-n-1}$	$y_{N-n}$	...	$y_{N-3}$	$y_{N-2}$	$y_{N-1}$	$y_N$	
$y_{N-n-1}$	$y_{N-n}$	$y_{N-n+1}$	...	$y_{N-2}$	$y_{N-1}$	$y_N$		



$y_1$     $y_2$     $y_3$

# Hands-on training



DS\_Bitácora\_42\_Series\_de\_Tiempo.ipynb

## Sección 2



# Proyecto 03





El Proyecto 3 de tu carrera tiene como objetivo que apliques las herramientas aprendidas dentro de tres dominios específicos sobre los que aprenderás en el Bloque 3:

- Sistemas de Recomendación
- Procesamiento de lenguaje natural
- Series de tiempo

El Proyecto 3 de tu carrera tiene como objetivo que apliques las herramientas aprendidas dentro de tres dominios específicos sobre los que aprenderás en el Bloque 3:

- Sistemas de Recomendación
- Procesamiento de lenguaje natural
- Series de tiempo

Implementa un modelo para predecir el flujo vehicular en una autopista de la Ciudad de Buenos Aires, Argentina.

- **Dataset.** [Aquí](#) puedes descargar el dataset. Deberás descargar –al menos para comenzar– los años 2017, 2018 y 2019. ¡Agradecemos a todos/as los que hacen Datos Abiertos!
- **Notebook.** [Aquí](#) te dejamos un notebook con algunas recomendaciones y líneas de código para que empieces a explorar los datos.
- **[Referencia Extra](#)**

# Para la próxima

---

- Avanza con el notebook de hoy.
- Lee la bitácora 43 y carga las dudas que tengas al Trello.

En el encuentro que viene uno/a de ustedes será seleccionado/a para mostrar cómo resolvió el challenge de la bitácora. De esta manera, ¡aprendemos todos/as de (y con) todas/as, así que vengan preparados/as.

ACÀMICA