



# Reporte de Modelos

Nawel Carimán Fuenzalida

17 de Septiembre del 2020

## 1. Modelos AR

### 1.1. Análisis previo

#### 1.1.1. Estructura general AR

Una modelo  $AR(n_a)$  se define como

$$y(t) + a_1 y(t-1) + \dots + a_{n_a} y(t-n_a) = e(t) \quad (1)$$

Introduciendo el polinomio  $A(q) = 1 + a_1 q^{-1} + \dots + a_{n_a} q^{-n_a}$  (en función del operador *delay*  $q$ ), la ecuación anterior se puede reescribir como:

$$A(q)y(t) = e(t) \quad (2)$$

Luego, definiendo el vector de parámetros  $\theta$  como

$$\theta = [a_1 \ a_2 \ \dots \ a_{n_a}] \quad (3)$$

y el vector de regresión

$$\varphi(t) = [-y(t-1) \ -y(t-2) \ \dots \ -y(t-n_a)] \quad (4)$$

con lo que se obtiene que la predicción es

$$\hat{y}(t|\theta) = \theta^T \varphi(t) \quad (5)$$

#### 1.1.2. Preprocesamiento de datos

Los datos que se utilizarán en los modelos  $AR$  serán los datos de glicemia crudos y unos pasados por dos tipos de filtros, el primero un filtro FIR y un promedio móvil.

- El filtro FIR que se utiliza está diseñado para tener una frecuencia de pasabanda en  $0,25mHz$  y frecuencia de rechazabanda de  $0,5mHz$ . Además, cuenta con corrección de fase, generando que este filtro sea no causal.

- El filtro de ventana móvil utiliza una venta de 5 muestras con una ventana rectangular. Este no cuenta con corrección de fase, por lo que puede clasificarse como un filtro causal.

Adicionalmente se trabajan obtuvo las diferencias de la señal, con el fin de poder trabajar con un modelo integral (modelo *ARI*). Muchas señales suelen no ser estacionarias pero su diferencia si, por lo que probar trabajar con esta transformación puede tener un mejor resultado.

En la figura 1 y 2 se puede ver un día completo de las señales en el tiempo. Aquí podemos ver que visualmente el filtro FIR sigue aceptablemente bien a la señal original, mientras que el filtro de ventana móvil sigue la forma con el desfase esperado. Por otro lado, en las señales diferenciales podemos ver más claramente el efecto de ambos filtros en relación a las variaciones de la señal original.

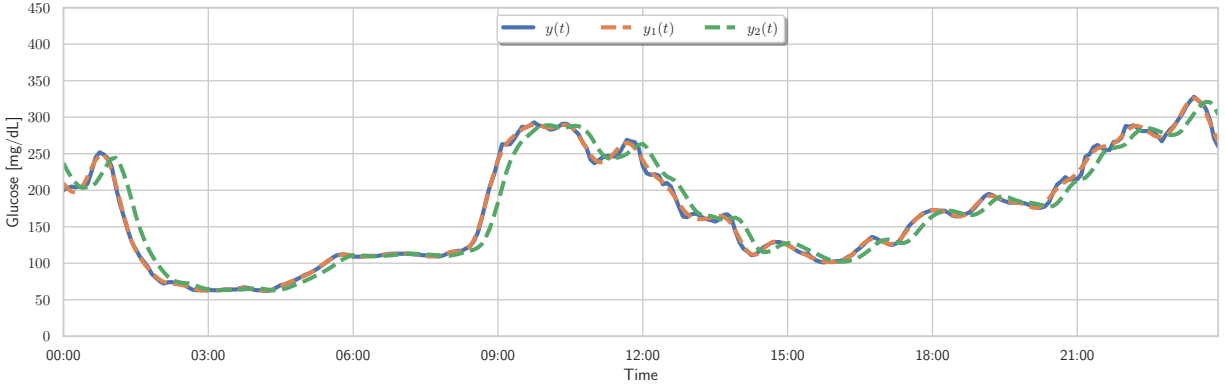


Figura 1: Gráfico de señales en el tiempo:  $y(t)$  señal original;  $y_1(t)$  señal filtrada con FIR;  $y_2(t)$  señal filtrada con ventana móvil

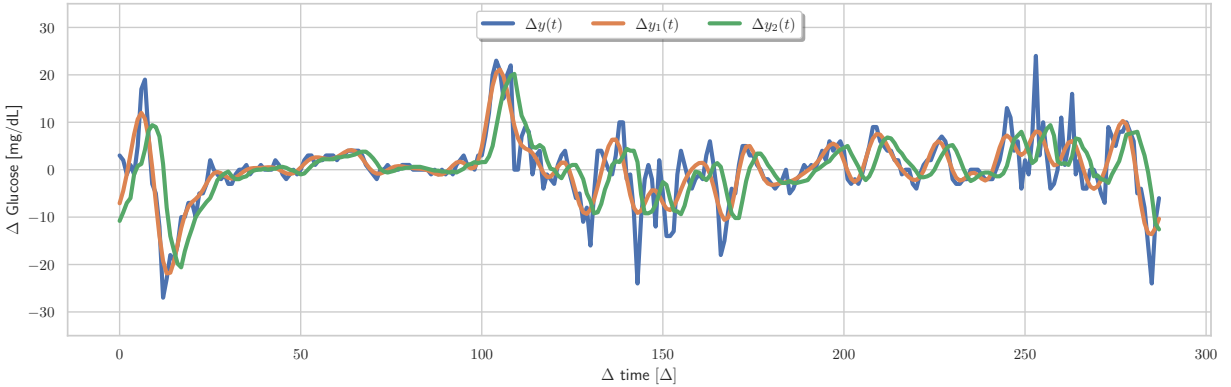


Figura 2: Gráfico de señales diferencial en el tiempo:  $y(t)$  señal original;  $y_1(t)$  señal filtrada con FIR;  $y_2(t)$  señal filtrada con ventana móvil

Luego, en las figuras 3 y 4 podemos ver los periodogramas de las señales y de sus diferencias. Aquí se puede ver el efecto del filtro FIR, que genera una fuerte caída en la rechazabanda, a diferencia de la ventana móvil que tiene un efecto menos fuerte sobre el periodograma.

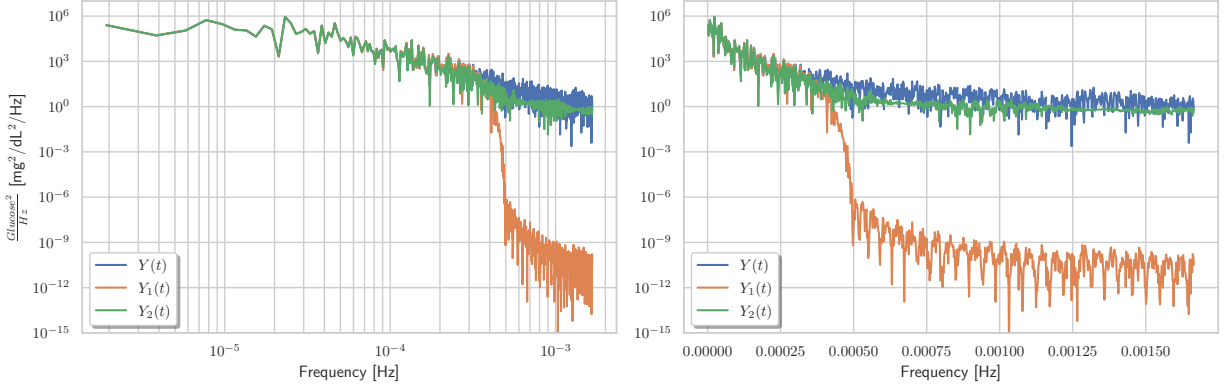


Figura 3: Gráfico del periodograma de las señales:  $y(t)$  señal original;  $y_1(t)$  señal filtrada con FIR;  $y_2(t)$  señal filtrada con ventana móvil

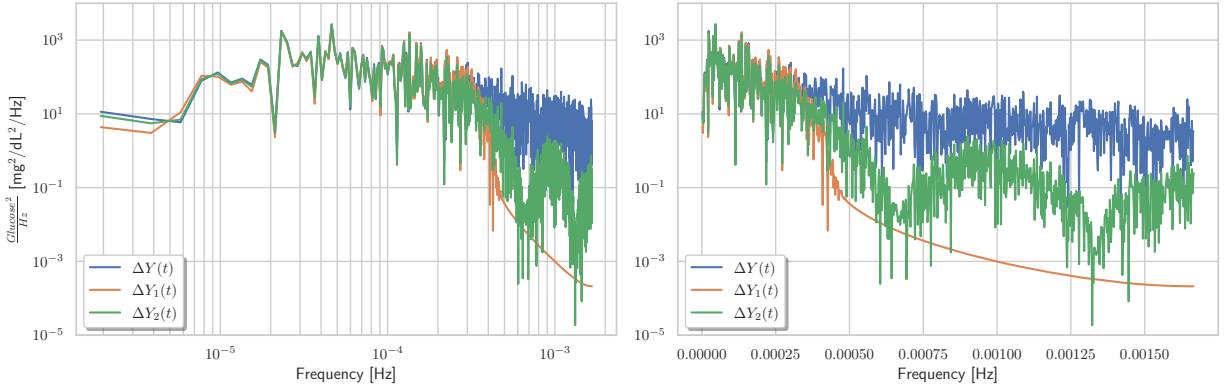


Figura 4: Gráfico del periodograma de las señales diferenciales:  $y(t)$  señal original;  $y_1(t)$  señal filtrada con FIR;  $y_2(t)$  señal filtrada con ventana móvil

Finalmente, en la figura 5 se muestran los resultados de la función de autocorrelación (ACF) y autocorrelación parcial (PACF) para cada grupo de señales. Estas fueron obtenidas con la librería *statmodels* de Python. Aquí notamos que la autocorrelación parcial para las señales originales (no diferenciadas) están fuera de rango, lo que indica que esta señal no cumple con ser lo suficiente estacionaria, mientras que las señales diferenciales si muestran el comportamiento esperado.

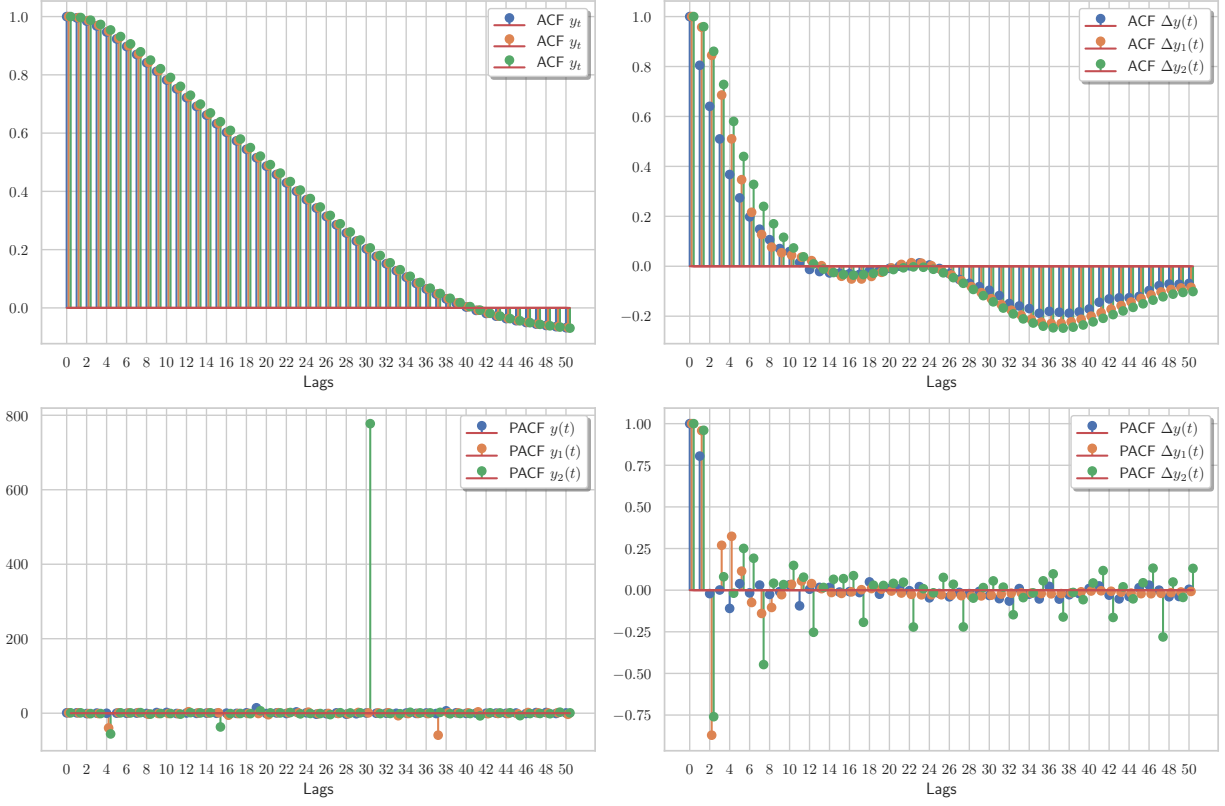


Figura 5: Gráfico del periodograma de las señales diferenciales:  $y(t)$  señal original;  $y_1(t)$  señal filtrada con FIR;  $y_2(t)$  señal filtrada con ventana móvil

## 1.2. Resultados

### 1.2.1. Modelo AR con datos crudos

Dado el poder de computo y simplicidad del modelo, se iteró para  $n_a$  entre 1 y 100. Se evaluó el desempeño utilizando RMSE, y su resultado se puede ver en la figura 6

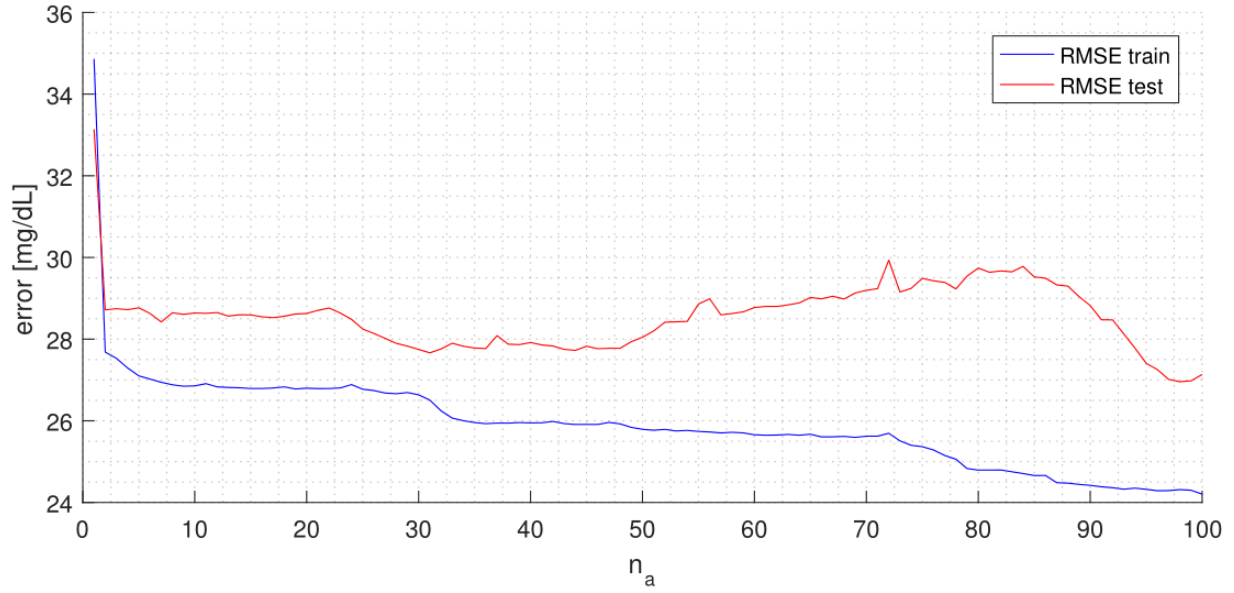


Figura 6: Gráfico del RMSE para distintos ordenes del modelo

Aquí notamos que existe un cambio bastante brusco entre  $n_a$  de 1 a 2, luego se mantiene relativamente constante hasta 30, entre 40 y 80 pareciera haber sobre entrenamiento y al en el tramo final, entre 90 y 100 caen nuevamente las métricas. Esto podría indicar que existe algún tipo de *seasonary* entrando a esta zona, o que los transientes de alimentación son despreciados en esta zona.

Luego, en la figura 7 se puede ver los valores de los parámetros de  $\theta$  para distintos ordenes. Aquí es importante notar que la mayoría de los ordenes utilizan bastante los primeros dos parámetros, mientras que los de mayor orden tienden a irse a cero. Por lo mismo, se puede concluir que un modelo autoregresivo de orden 2,  $AR(2)$  es el modelo más simple que cumple con lo deseado.

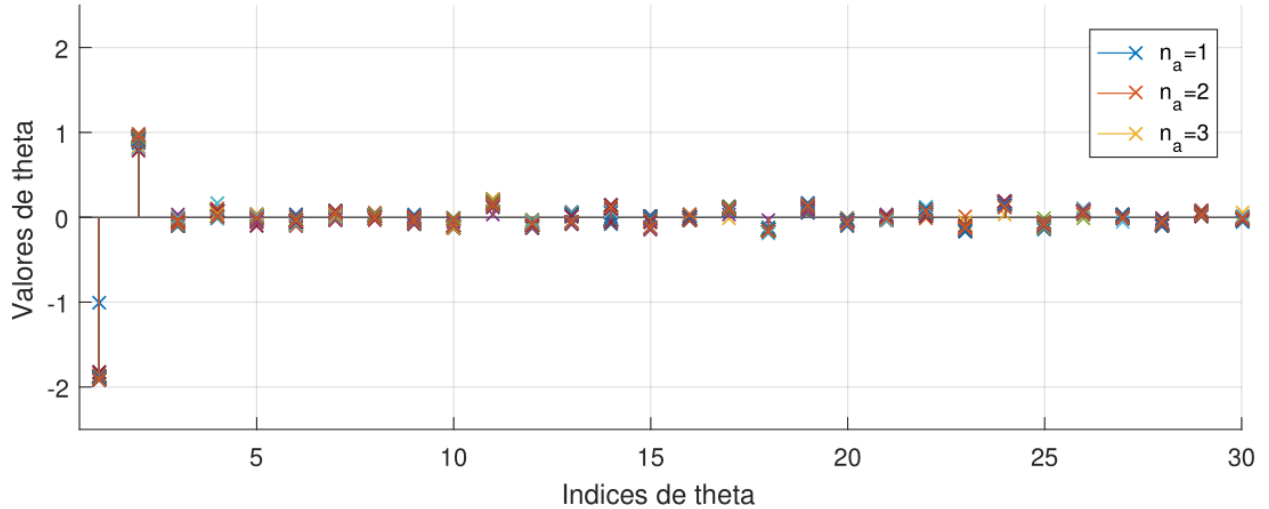


Figura 7: Gráfico parámetros de  $\theta$  para distintos ordenes del modelo

En la figura 8 y 9 se puede ver el gráfico en el tiempo y los gráficos del error respectivamente.

En la tabla 1 se pueden el resumen de las métricas.

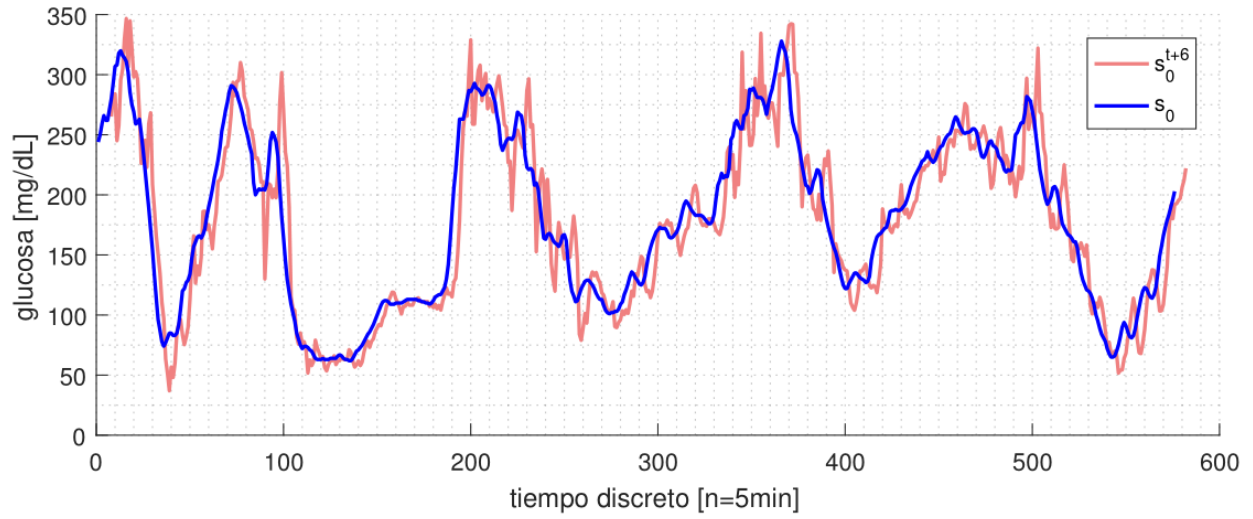


Figura 8: Gráfico de serie de tiempo real vs comparada para AR(2)

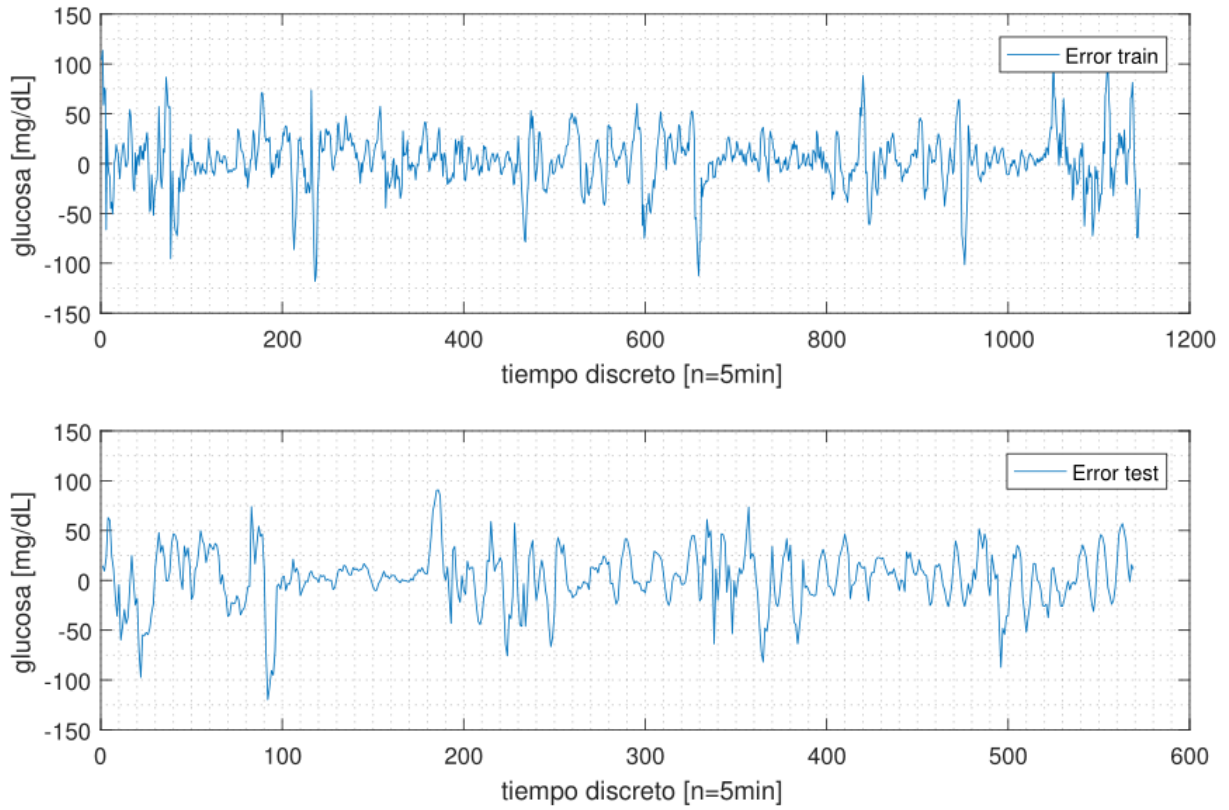


Figura 9: Gráfico de error para el training y testingo de modelo AR(2)

	Entrenamiento	Prueba
<b>RMSE</b>	27.68	28.72
<b>Media</b>	3.06	2.54
<b>Desviación estándar</b>	27.53	28.63

Cuadro 1: Resumen de métricas para datos crudos

### 1.2.2. Modelo AR con filtro FIR



Nuevamente se iteró utilizando el poder computacional, obteniendo el gráfico de la figura 10.

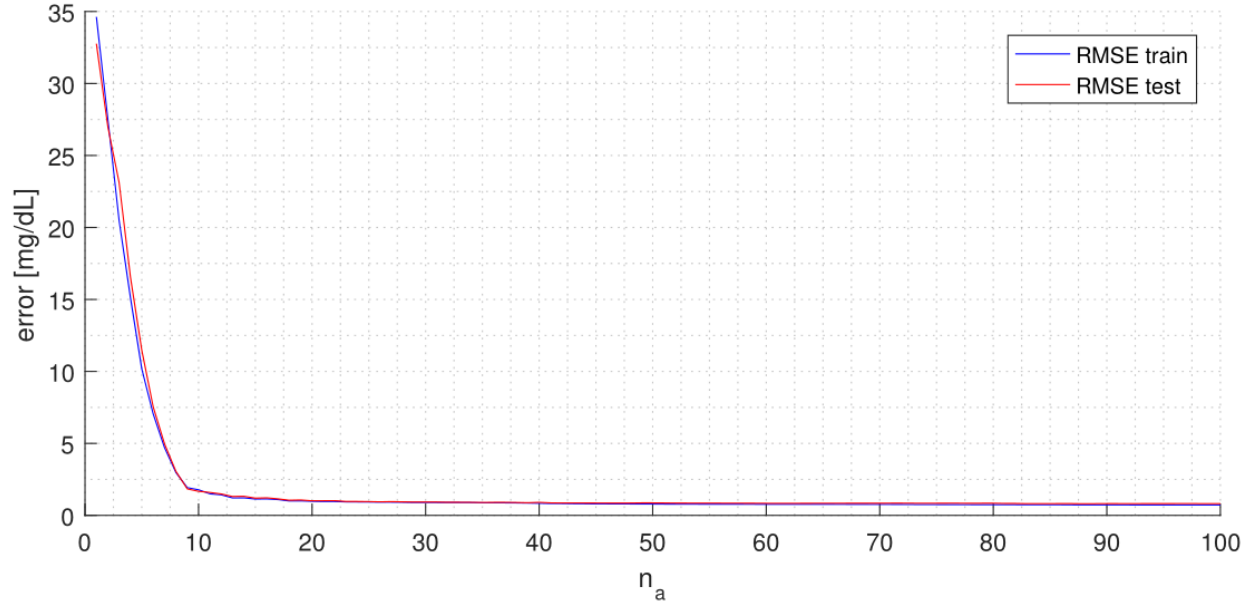


Figura 10: Gráfico del RMSE para distintos ordenes del modelo

De este gráfico podemos notar que si se ve que al aumentar el orden del modelo se disminuye drásticamente el RMSE, donde se ve que se alcanza un valor aceptable cercano a  $n_a = 20$ .

Respecto al comportamiento del vector de parámetros  $\theta$ , podemos apreciar que tiene un comportamiento menos marcado en la figura 11, donde dentro de los primeros 20 valores se obtienen parámetros bastante altos, volviendo este modelo inestable y muy sensible a ruidos en las mediciones.

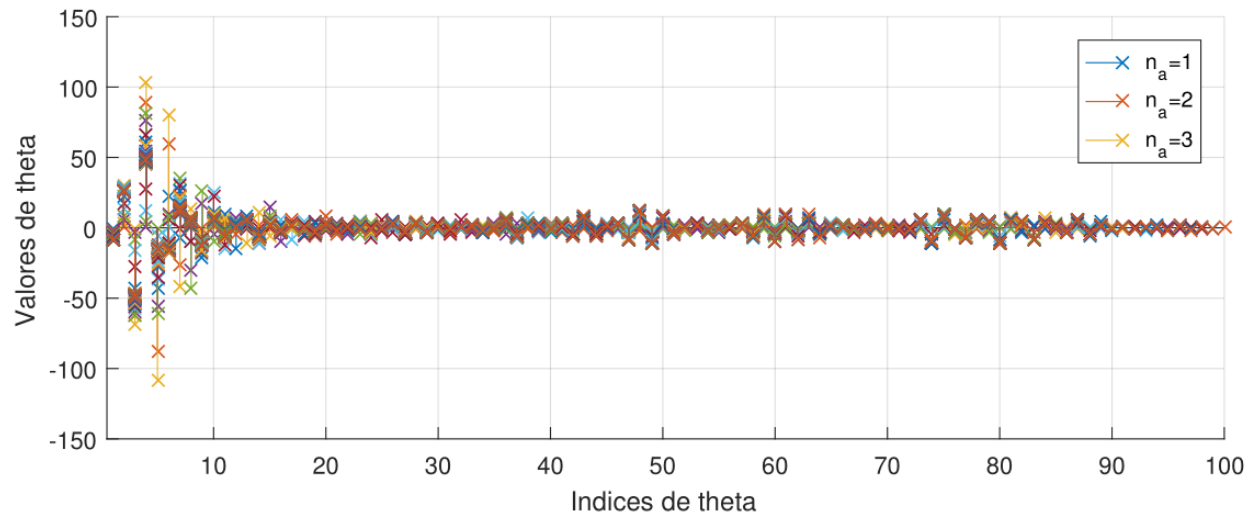


Figura 11: Gráfico parámetros de  $\theta$  para distintos ordenes del modelo

En la figura 12 y 13 se puede ver el gráfico en el tiempo y los gráficos del error respectivamente. En la tabla 2 se pueden el resumen de las métricas.

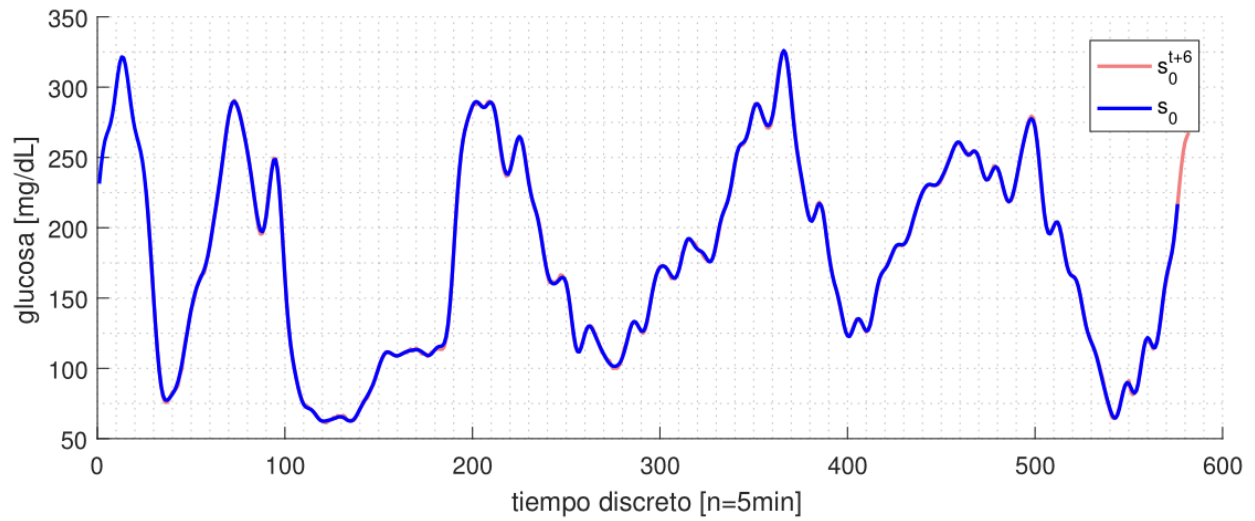


Figura 12: Gráfico de serie de tiempo real vs comparada para AR(20)



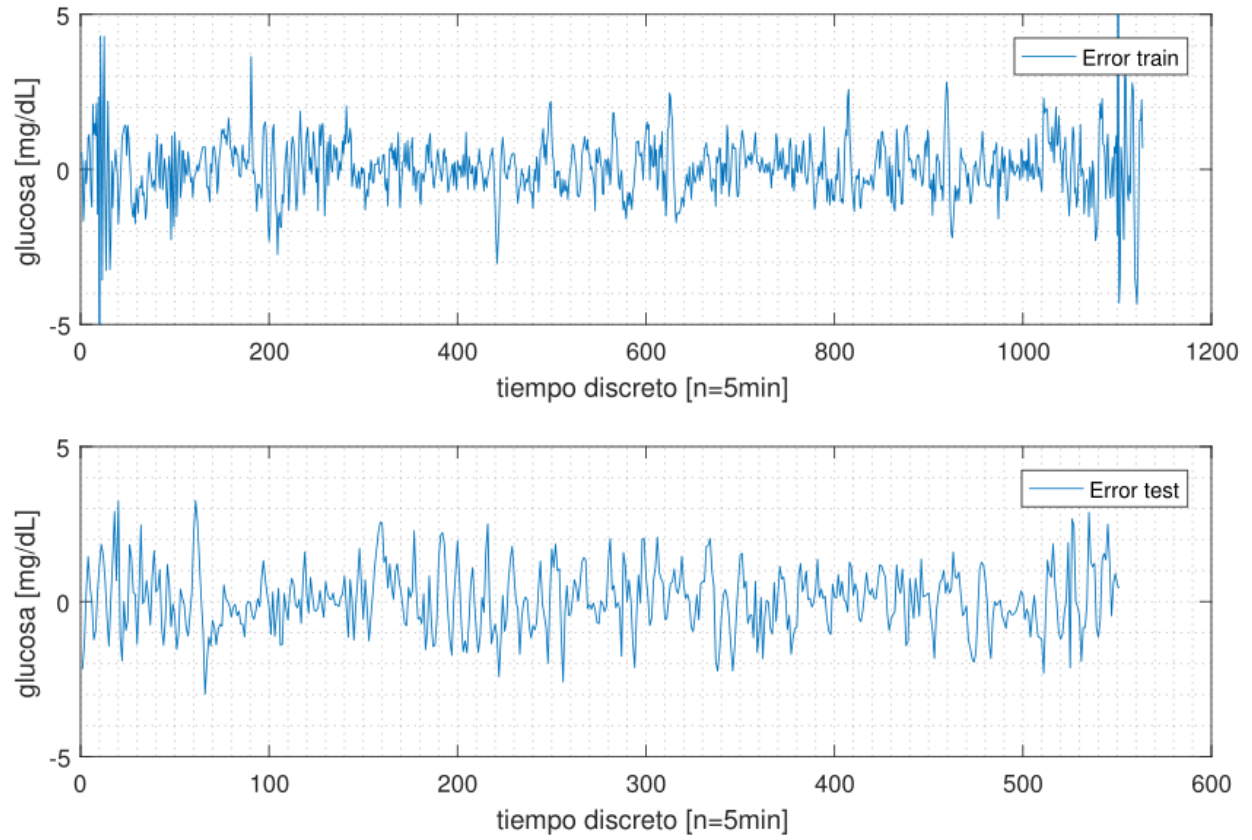


Figura 13: Gráfico de error para el training y testingo de modelo AR(20)

	Entrenamiento	Prueba
<b>RMSE</b>	0.92	1.01
<b>Media</b>	0.036	0.057
<b>Desviación estándar</b>	0.98	1.02

Cuadro 2: Resumen de métricas para datos con filtro FIR

### 1.2.3. Modelo AR con filtro ventana móvil

Finalmente, para el ultimo conjunto de datos se realizó el mismo procedimiento que antes, pero esta vez hasta  $n_a = 150$ . Los resultados se ven en la figura 14.

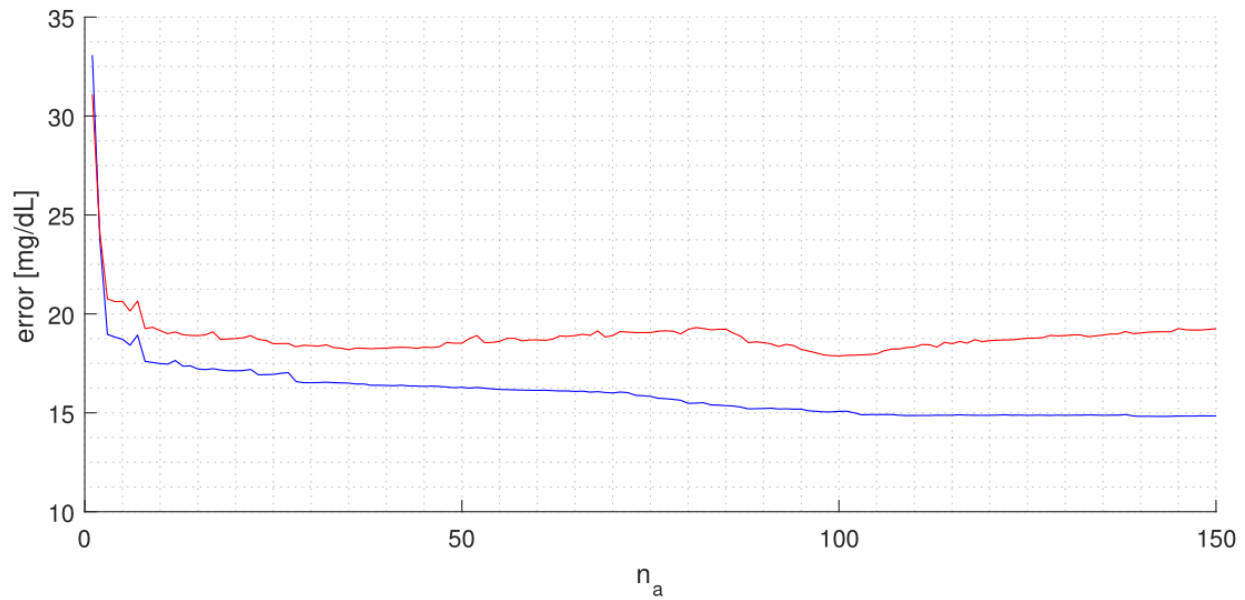


Figura 14: Gráfico del RMSE para distintos ordenes del modelo

Notamos que este gráfico está pareciendo estabilizarse para  $n_a = 20$ , obteniendo un buen desempeño cercano a 80.

Nuevamente notamos que ahora el modelo necesita muchos valores del vector de parámetros, y que estos son oscilantes, al igual que en el modelo FIR, aunque de una amplitud menor. Además, notamos que cerca de 80 pareciera que estos se estabilizan un poco, por lo que este es el valor escogido.

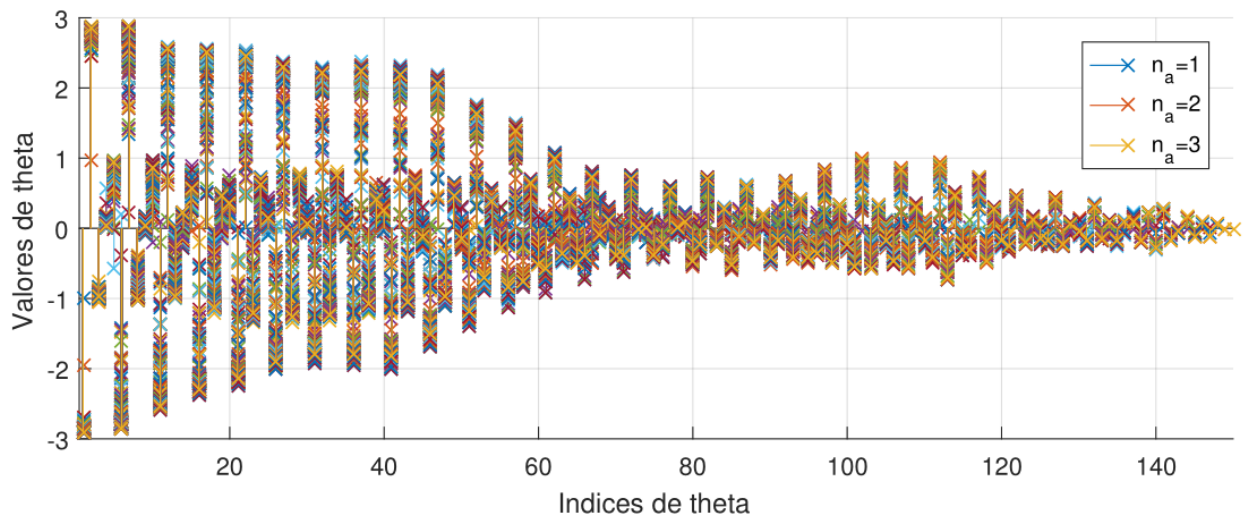


Figura 15: Gráfico parámetros de  $\theta$  para distintos ordenes del modelo

En la figura 16 y 17 se puede ver el gráfico en el tiempo y los gráficos del error respectivamente. En la tabla 3 se pueden el resumen de las métricas.

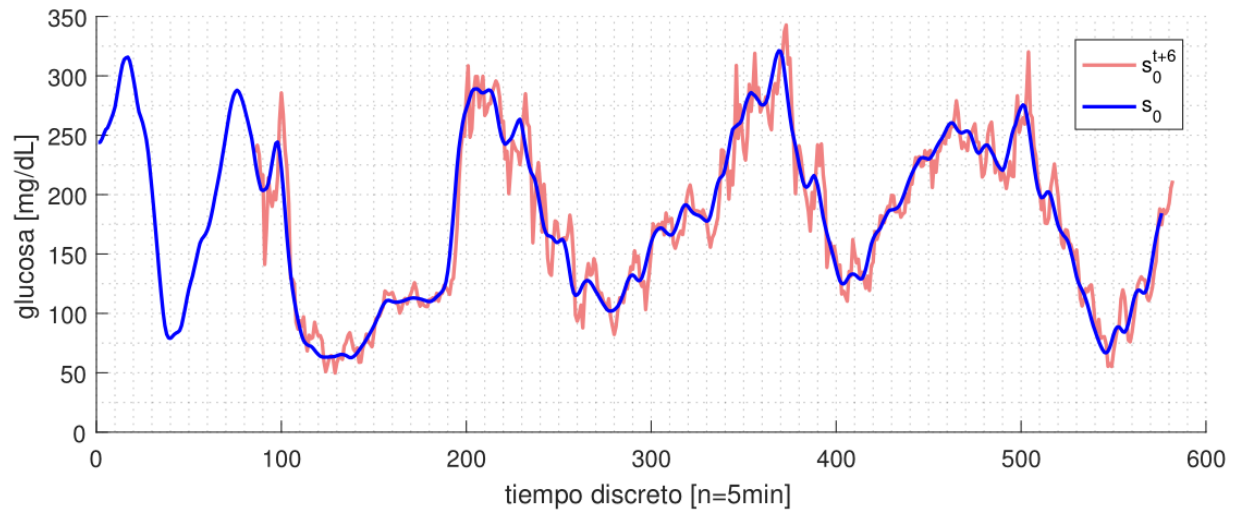


Figura 16: Gráfico de serie de tiempo real vs comparada para AR(20)

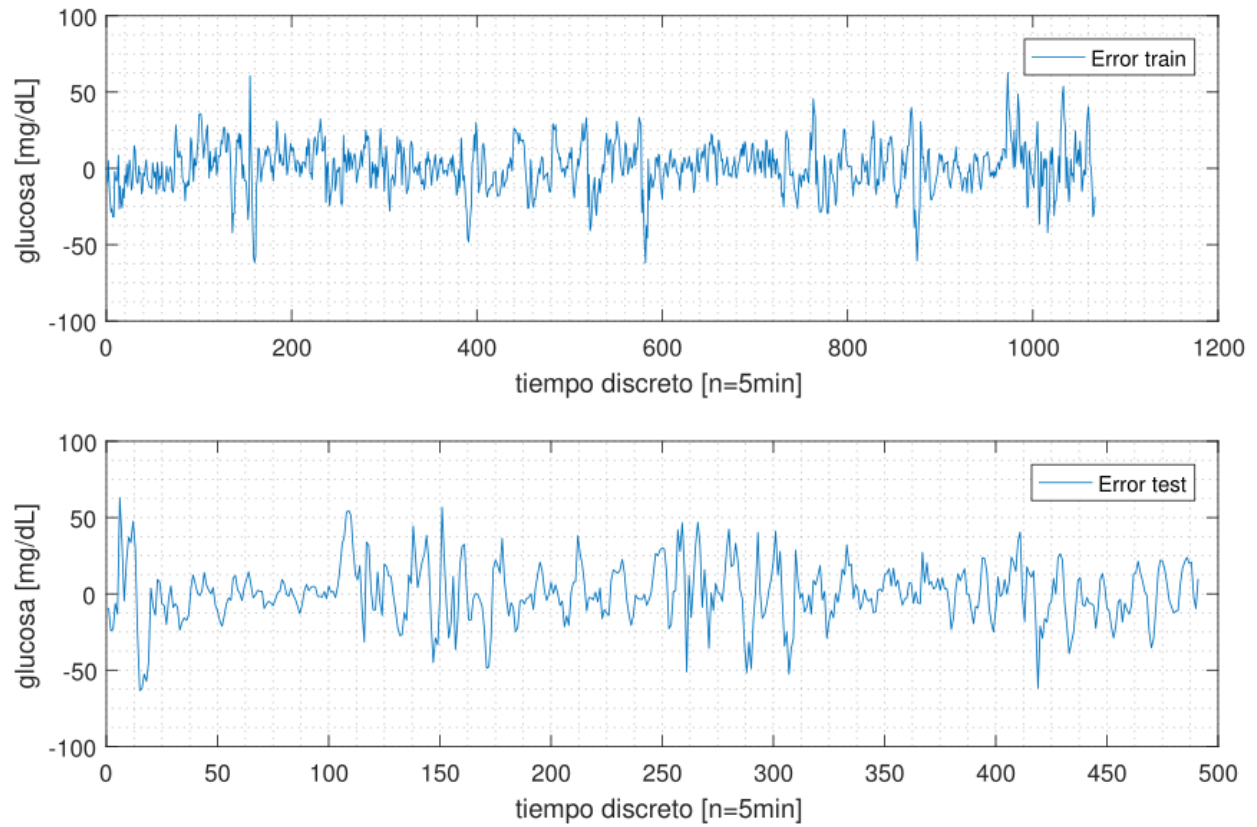


Figura 17: Gráfico de error para el training y testingo de modelo AR(80)

	Entrenamiento	Prueba
<b>RMSE</b>	15.48	19.21
<b>Media</b>	0.72	0.62
<b>Desviación estándar</b>	15.47	19.22

Cuadro 3: Resumen de métricas para datos con filtro de ventana móvil