PulsoPBA: A Production System for Economic Activity Nowcasting with Neural Networks and Banking Data

Kejsefman Igal[1,2], Caputo Bugallo Agustín[1,4], Valens Upegui Milena[3], Castro Rodrigo[1,4]
[1]Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET)
[2]Instituto de Estudios de América Latina y el Caribe (IEALC-UBA)
[3]Banco de la Provincia de Buenos Aires
[4]Instituto de Ciencias de la Computación (ICC-UBA-CONICET)

Abstract - PulsoPBA is a nowcasting system for weekly estimation of economic activity in the Province of Buenos Aires (Argentina), whose official indicator is published with a lag of up to 10 weeks at monthly granularity. The system processes approximately 1,200 variables derived from banking transactions.

A temporal disaggregation stage constructs the weekly target series from official monthly data. The high dimensionality and heterogeneity of the input series motivated the adoption of neural networks capable of projecting covariates into a lower-dimensional latent space. The system employs TiDE (Time-series Dense Encoder), whose encoder enables intrinsic dimensionality reduction.

The main contribution is a fully automated pipeline deployed in a production environment. The system retrains automatically upon each new official data release, integrating a multi-criteria model selection funnel based on the Weighted Interval Score (WIS) and probabilistic predictions via Monte Carlo Dropout.

Operational results validated against officially published ex post data yield a monthly MAPE between 0.7% and 3.5%, demonstrating the feasibility of a high-frequency nowcasting system built on high-dimensional banking data.

Keywords: Nowcasting, Deep learning, High dimensionality, Banking data, Production system

PulsoPBA: Nowcasting operativo de actividad económica utilizando datos bancarios mediante redes neuronales

Kejsefman Igal[1,2], Caputo Bugallo Agustín[1,4], Valens Upegui Milena[3], Castro Rodrigo[1,4]
[1]Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET)
[2]Instituto de Estudios de América Latina y el Caribe (IEALC-UBA)
[3]Banco de la Provincia de Buenos Aires
[4]Instituto de Ciencias de la Computación (ICC-UBA-CONICET)

Resumen - PulsoPBA es un sistema de *nowcasting* para la estimación de granularidad semanal de la actividad económica de la Provincia de Buenos Aires (Argentina), cuyo

dato oficial se publica con un rezago de hasta 10 semanas y con una granularidad mensual. Para este objetivo, utiliza datos de aproximadamente 1.200 variables derivadas de transacciones bancarias.

El sistema incorpora una etapa de desagregación temporal para construir la serie objetivo semanal partiendo de datos mensuales oficiales. En cuanto al entrenamiento, la alta dimensionalidad y heterogeneidad de estas series motivó la adopción de redes neuronales capaces de proyectar variables a un espacio latente de menor dimensión. Se utiliza la arquitectura TiDE, cuyo encoder viabiliza una reducción de dimensionalidad intrínseca.

El resultado principal es un pipeline automatizado en entorno productivo. El sistema se re-entrena automáticamente al publicarse un nuevo dato oficial, integrando un embudo de selección de modelos multi-criterio basado en el *Weighted Interval Score* (WIS) y predicciones probabilísticas via *Monte Carlo Dropout.*

Los resultados operativos contrastados con datos oficiales publicados ex post muestran un MAPE mensual de entre 0.7% y 3.5%, validando el pipeline y demostrando la factibilidad de un sistema de nowcasting de alta frecuencia a partir de datos bancarios de alta dimensionalidad.

Palabras clave: Nowcasting, Aprendizaje profundo, Alta dimensionalidad, Transacciones bancarias, Sistema en producción

1. Introducción

El *nowcasting* de variables económicas es un problema canónico de predicción de series temporales. A nivel nacional, los sistemas de nowcasting han sido desarrollados y desplegados por Bancos Centrales y agencias públicas de estadísticas en todo el mundo (Aprigliano et al., 2017; Woloszko, 2024; Linzenich y Meunier, 2024). Este trabajo describe PulsoPBA, un sistema end-to-end en entorno productivo de nowcasting de actividad económica de la Provincia de Buenos Aires mediante el procesamiento masivo de registros administrativos de transacciones del banco provincial. Las decisiones de arquitectura se tomaron en función de los requerimientos operativos, no como competencia empírica entre modelos.

El sistema está condicionado por una restricción técnica estructural: el Estimador Mensual de Actividad de la Provincia de Buenos Aires (EMA-PBA), se publica con un rezago de hasta 10 semanas. Esta restricción implica que (1) la variable objetivo de granularidad semanal debe construirse a partir de la serie mensual, mediante una desagregación temporal, (2) el sistema debe generar estimaciones de la semana corriente a partir de covariables disponibles diariamente, y (3) el sistema debe reentrenarse automáticamente cada vez que se publica un nuevo dato oficial, sin intervención humana.

El insumo del sistema son las transacciones diarias del Banco de la Provincia de Buenos Aires, presente en los 135 municipios provinciales. A partir de cuatro fuentes de datos (movimientos de cuenta, préstamos, comercio exterior y consumo) se construyen aproximadamente 1.200 series temporales diarias.

La alta dimensionalidad del espacio de covariables requiere una arquitectura capaz de realizar reducción de dimensionalidad de forma intrínseca. Los métodos econométricos clásicos no escalan a este régimen, y los métodos de ensamble de árboles requieren una ingeniería de atributos, incompatible con un sistema de producción autónomo. Se adoptó TiDE (Das et al., 2023), cuyo encoder denso proyecta el espacio de covariables a una representación latente, viabilizando el entrenamiento en el contexto de restricciones planteadas.
Las contribuciones de este trabajo son: (1) exponer el diseño, implementación y validación de PulsoPBA, un sistema de nowcasting semanal desplegado en producción en el Banco de la Provincia de Buenos Aires. El sistema procesa ~1.200 series bancarias diarias y opera de forma autónoma con reentrenamiento mensual, integrando un embudo de selección multi-criterio basado en el *Weighted Interval Score* (WIS) y predicciones probabilísticas via *Monte Carlo Dropout*; y (2) validar empíricamente las predicciones contra datos oficiales publicados *ex post*.

2. Trabajos Relacionados

Los modelos de nowcasting estiman en tiempo real variables macroeconómicas integrando información a distintas frecuencias (Giannone, Reichlin y Small, 2008). El problema combina dificultades de los datos -alta dimensionalidad, observaciones faltantes, frecuencias heterogéneas- con propiedades del proceso generador: dinámicas no lineales, no estacionariedad y cambios estructurales. Los modelos econométricos clásicos abordan el primer grupo (Bańbura y Modugno, 2014), pero su supuesto de linealidad limita su capacidad ante el segundo (Huber et al., 2021). Los métodos de ensamble de árboles como XGBoost (Chen y Guestrin, 2016) y LightGBM (Ke et al., 2017) ofrecen buen desempeño en alta dimensionalidad, pero requieren diseño manual de atributos predictivos -limitación operativa cuando las covariables superan el millar- y no aprenden representaciones internas que permitan integrar este proceso como parte intrínseca de la arquitectura (Lim y Zohren, 2021).

2.1 Aprendizaje profundo para predicción de series temporales

La literatura sobre aprendizaje profundo para predicción de series temporales ha crecido aceleradamente en la última década. Benidis et al. (2022) presentan las principales arquitecturas -redes recurrentes (RNN, LSTM), redes convolucionales (CNN) y mecanismos de atención (Transformers)- y documentan su capacidad para capturar dependencias temporales complejas, no lineales y no estacionarias. Lim y Zohren (2021) destacan que la principal ventaja de las redes profundas en este contexto es el aprendizaje de representaciones latentes: combinaciones no lineales de observaciones pasadas que condensan información temporal sin requerir una especificación a priori del modelo.
Investigaciones recientes (Zeng et al., 2023) han mostrado que estructuras más simples, basadas en bloques de perceptrones multicapa (MLP) y conexiones residuales, pueden alcanzar -e incluso superar- el desempeño de los modelos más sofisticados, como los *transformers.* Con muchas covariables, la inductive bias de los Transformers (invarianza a la permutación y conectividad global) no necesariamente ayuda, mientras que MLPs tienen mejor trade-off complejidad/generalización y menor costo computacional.

Estos antecedentes motivan el uso de arquitecturas de la familia MLP como candidatas para el sistema presentado en este trabajo. TiDE (Das et al., 2023) y TSMixer (Chen et al., 2023) son arquitecturas MLP diseñadas para forecasting con covariables exógenas.

2.2 Nowcasting con datos bancarios.

El *nowcasting* fue adoptado progresivamente por bancos centrales y organismos estadísticos para anticipar tendencias económicas con baja latencia, sobre todo a partir de la pandemia de COVID-19 (Linzenich y Meunier, 2024; Doerr et al., 2021). En contextos de crisis, alta volatilidad, y no linealidad, Oukhouya et al. (2024), combinaron arquitecturas RNN-LSTM y XGBoost para el nowcasting de índices bursátiles internacionales con resultados superiores a los modelos individuales.

Una línea complementaria demostró el valor predictivo de los datos bancarios y de pagos. Aprigliano et al. (2017) documentaron para Italia que variables del sistema de pagos, con selección vía Lasso, mejoran sustancialmente la precisión del nowcasting del PIB. Barlas et al. (2021) confirmaron con datos del BBVA de Turquía que la información bancaria supera a los modelos basados exclusivamente en series macroeconómicas agregadas. León y Ortega (2018) obtuvieron resultados análogos con datos de alta frecuencia del sistema bancario colombiano.

En Argentina, los antecedentes se concentran en metodologías econométricas a nivel nacional. D'Amato et al. (2016) y Blanco et al. (2022) emplearon ecuaciones puente y modelos de factores dinámicos, superando al modelo autorregresivo base. Llada (2022) incorporó variables de incertidumbre y presión cambiaria. Carrera (2024) combinó técnicas econométricas (ARDL, GETS, GSR) con *machine learning* (Lasso, Ridge, Elastic Net) para anticipar el EMAE con un mes de antelación.

La Sección 4 expone las decisiones de diseño arquitectural de PulsoPBA, analizando por qué las propiedades estructurales de TiDE resultan las más adecuadas para las restricciones operativas del sistema.

3. Fuentes de datos y construcción de la información

3.1 Fuentes de datos

El sistema utiliza cuatro conjuntos de registros administrativos anonimizados del Banco de la Provincia de Buenos Aires, con granularidad diaria. Los datos cubren el período enero de 2013– actualidad, con fechas de inicio variables por fuente. La Tabla 1 resume las características principales.

Tabla 1. Características de los datasets del Banco Provincia

Fuente	Contenido	Variables (post-filtro)	Inicio de la serie
Movimientos de cuenta	Créditos/débitos por actividad (2 dígitos) y región	772	2013
Préstamos	Créditos por tipo y sector de actividad	94	2013
Comercio exterior	Exportaciones e importaciones	327	2019

Fuente	Contenido	Variables (post-filtro)	Inicio de la serie
Consumo	Compras clasificadas 21 rubros comerciales	57	2015

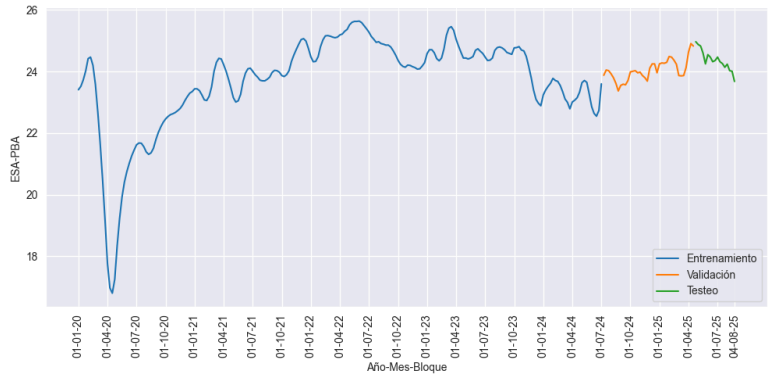
Fuente: Elaboración propia.

A partir de las dimensiones disponibles en cada fuente (región, sector institucional, actividad económica, tipo de transacción, moneda), se construyen series de tiempo de granularidad diaria con claves predefinidas desde enero de 2020 hasta la actualidad. Este procedimiento genera aproximadamente 3.000 variables candidatas. Tras eliminar aquellas cuya mediana es igual a cero -series sin movimiento durante al menos el 50% de los días- se retiene un conjunto final de aproximadamente 1.200 variables. Este volumen de covariables es el que define el problema de alta dimensionalidad que motiva la elección arquitectural presentada en la Sección 4.
El pipeline de preprocesamiento es el punto de partida del sistema. Los montos en moneda extranjera se pesifican al tipo de cambio oficial del día. Se incorporan atributos cíclicos de calendario (mes, semana, trimestre, día, feriados) para que el modelo sea capaz de capturar estacionalidades durante el proceso de entrenamiento. Las series diarias se agregan a frecuencia semanal mediante una estructura de bloques de duración variable que garantiza comparabilidad entre semanas de distintos meses.

3.2 Variable objetivo: desagregación temporal con Chow-Lin

El entrenamiento supervisado requiere que la variable objetivo comparta la frecuencia semanal de las covariables bancarias. El EMA-PBA se publica con frecuencia mensual, lo que exige un procedimiento de desagregación temporal ("semanalización") para construir el Estimador Semanal de Actividad (ESA-PBA).

Gráfico 1: Particiones para entrenamiento y predicción del ESA-PBA:



Fuente: Elaboración Propia

Se evaluaron tres estrategias. La distribución uniforme (naive) reparte el valor mensual en partes iguales, descartando toda información intrasemanal. Los modelos

de espacio de estado (filtros de Kalman) son algoritmos no supervisados que pueden inferir la dinámica semanal a partir de la estructura interna de la serie mensual, pero desaprovechan la información contextual disponible. Se optó por el método supervisado de Chow y Lin (1971), que aprovecha explícitamente las series bancarias como predictores de alta frecuencia mediante una regresión de mínimos cuadrados generalizados entre la serie mensual y una variable auxiliar de alta frecuencia, sujeta a la restricción de que la suma de los valores semanales estimados reproduzca exactamente el agregado mensual observado.

La implementación utilizada (timedisagg[1]; siguiendo a Sax y Steiner, 2013) opera de forma univariada, por lo que el método se aplicó iterativamente con cada una de las ~1.200 series bancarias como variable auxiliar. El promedio simple del conjunto de las desagregaciones conforma el ESA-PBA (Gráfico 1). Dado que cada desagregación satisface por construcción la restricción de agregación temporal y el promedio es una operación lineal, el ESA preserva la coherencia con el EMA-PBA. La estabilidad del indicador se refleja en un coeficiente de variación entre 0,001 y 0,02 entre las desagregaciones individuales.

4. Decisiones de diseño de PulsoPBA

4.1 Arquitecturas evaluadas

El problema de predicción abordado por PulsoPBA impone restricciones arquitecturales mencionadas. El sistema debe operar con aproximadamente 1.200 covariables bancarias heterogéneas, actualizarse automáticamente ante cada nuevo dato oficial, evitar la ingeniería atributos *ex ante* y producir pronósticos probabilísticos utilizables en un entorno de producción.

Tabla 2. Comparación de arquitecturas consideradas para PulsoPBA

			Redes neuronales			
Requisito del sistema	Econo-métricos	Árboles	RNN / LSTM	Trans-formers	TSMixer	**TiDE**
Ingeniería de atributos intrínseca	✗	✗	✓	✓	✓	✓
Separación covariables pasadas / futuras	✗	✗	✗	~	~	✓
Reducción de dimensionalidad nativa	✗	✗	✗	✗	✗	✓
Arquitectura no recursiva	✗	✓	✗	✓	✓	✓
Predicciones probabilísticas	✓	~	✓	✓	✓	✓

✓ = sí ✗ = no ~ = parcial / depende de implementación

[1] https://github.com/jstephenj14/timedisagg

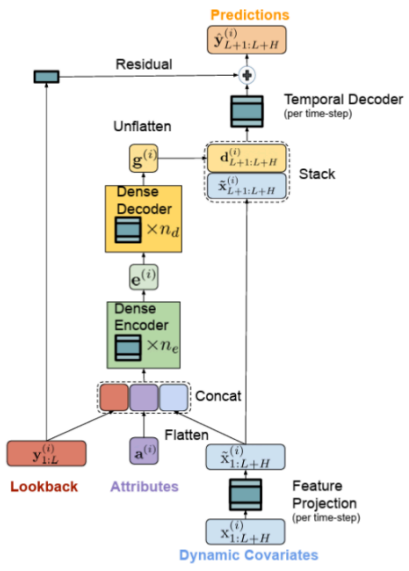
Fuente: Elaboración propia

Se consideraron tres familias de arquitecturas: métodos econométricos clásicos, ensambles de árboles, y redes neuronales. La Tabla 2 muestra que ninguna familia domina en todos los criterios. Los modelos econométricos clásicos ofrecen interpretabilidad y predicción probabilística, pero no resultan adecuados cuando el número de covariables supera largamente el centenar y el sistema debe evitar especificación de rezagos, interacciones y transformaciones. Los ensambles de árboles pueden manejar relaciones no lineales, pero típicamente dependen de una etapa explícita de feature engineering, lo que introduce complejidad de mantenimiento incompatible con un pipeline enteramente automatizado.
Entre las redes neuronales, las redes recurrentes (RNN/LSTM) y los Transformers eliminan parte de esa dependencia de ingeniería de atributos *ex ante,* dado que aprenden representaciones internas de los datos. Sin embargo, las primeras requieren procesamiento secuencial paso a paso, mientras que los segundos trasladan el costo al manejo de representaciones de alta dimensión y mecanismos de interacción más pesados. En ambos casos, la escalabilidad se vuelve un punto crítico cuando el espacio de covariables exógenas crece hasta el orden del millar.

4.2 Arquitecturas TiDE versus TSMixer

TSMixer (Chen et al., 2023) constituye una alternativa más cercana al tipo de solución buscada, ya que pertenece a la familia de modelos MLP modernos para series temporales y evita tanto la recurrencia como la atención completa. No obstante, su mecanismo de mixing opera directamente sobre el espacio de atributos, sin introducir una etapa tan explícita de compresión temprana del bloque de entrada. En un problema como el aquí considerado, esta diferencia deja de ser un detalle de implementación y pasa a ser una propiedad arquitectural relevante.
TiDE —*Time-series Dense Encoder* (Das et al., 2023)— se distingue no por ser la única capaz de incorporar covariables o generar pronósticos probabilísticos, sino por combinar tres propiedades particularmente valiosas para el problema abordado: (i) separación estructurada entre tipos de covariables, (ii) compresión temprana del espacio de entrada de alta dimensión hacia una representación latente reducida, y (iii) procesamiento no secuencial. En un contexto con aproximadamente 1.200 series exógenas heterogéneas, esta combinación resulta especialmente adecuada frente a alternativas que requieren mayor ingeniería de atributos, operan de forma recurrente o mezclan directamente sobre el espacio completo de variables.

Figura 1: Arquitectura Tide



Fuente: Das et al., 2023

Arquitectónicamente, TiDE combina una conexión residual lineal global con un encoder-decoder denso (Figura 1). La conexión lineal mapea directamente la ventana de *lookback* al horizonte de predicción, capturando tendencias de largo plazo y garantizando que el modelo sea al menos tan expresivo como un predictor lineal óptimo. El componente no lineal opera en tres

etapas: primero, una *feature projection* aplica un bloque residual denso por paso temporal, proyectando el espacio de covariables a una representación latente de dimensión reducida —operación crítica cuando el espacio de entrada supera el millar de variables— que además descubre representaciones intermedias informativas durante el entrenamiento. Segundo, el encoder recibe la concatenación de los valores históricos y las covariables proyectadas, produciendo un embedding compacto mediante bloques residuales apilados. Finalmente, el decoder mapea este embedding al horizonte de predicción.

5. Resultados: pipeline TiDE para Nowcasting de Alta Dimensionalidad

El resultado principal de este trabajo es un pipeline automatizado *end-to-end* de nowcasting desplegado en el entorno productivo del Banco de la Provincia de Buenos Aires. El sistema se ejecuta de forma autónoma: al publicarse un nuevo dato del EMA-PBA, se dispara un ciclo completo de reentrenamiento, selección de modelos y generación de predicciones, sin intervención humana. Una vez por semana se realizan las predicciones.

Figura 2. Esquema de nowcasting.

Nota: todos los meses están compuestos por cuatro bloques/semanas.
Fuente: Elaboración propia.

En términos esquemáticos (Figura 2), el modelo puesto en producción utiliza los datos bancarios de las semanas $t-n$ a t y el historial del ESA-PBA de las semanas $t-n$ a $t-1$ para generar el nowcast de la semana t. Este esquema replica las condiciones reales de disponibilidad de información: los datos bancarios de la semana corriente están disponibles al final de cada semana, mientras que el dato oficial del EMA-PBA se publica con un rezago de hasta 10 semanas.

En esta sección se presentan los componentes del pipeline como resultado en sí mismo: el proceso de entrenamiento y optimización con la arquitectura TiDE -seleccionada en la Sección 4 por ser la mejor satisface el requisitos técnicos del proyecto-, el embudo de selección multi-criterio del modelo, el paso a producción y el esquema de predicciones probabilísticas. La Sección 6 presenta los resultados empíricos de la validación operativa del sistema.

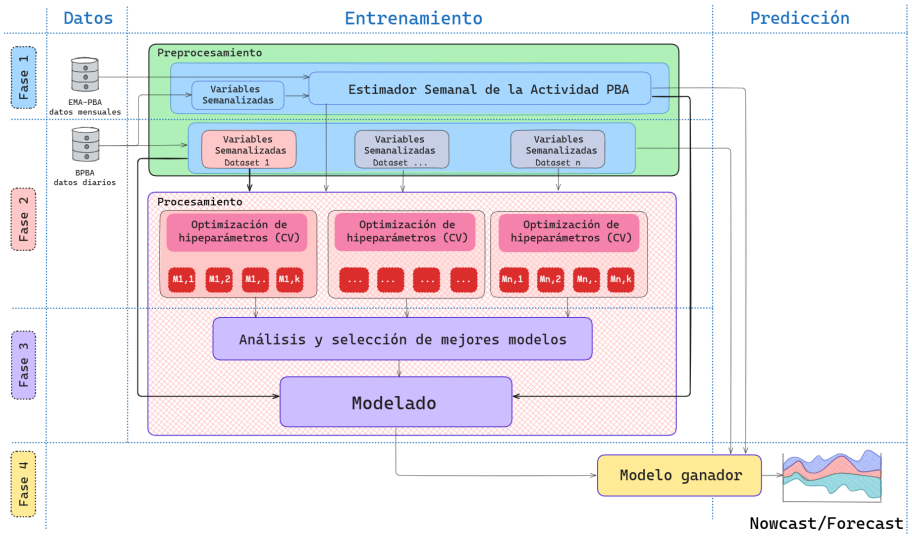
5.1 Pipeline de entrenamiento: preprocesamiento, modelado y validación

El proceso de entrenamiento se organiza en cuatro etapas (Figura 3): partición de los datos y preprocesamiento, optimización de hiperparámetros mediante validación cruzada temporal, evaluación de generalización sobre un conjunto de testeo reservado,

y selección del modelo final mediante un embudo multi-criterio. Todo el proceso se ejecuta automáticamente cada mes, al publicarse un nuevo dato del EMA-PBA. La implementación de TiDE utiliza la librería Darts[2] (Herzen et al., 2022).

1.a) *Partición de datos y prevención de data leakage:* Los datos se dividen en tres subconjuntos cronológicos: entrenamiento, validación y testeo (Gráfico 1). Los subconjuntos de entrenamiento y validación se utilizan dentro de un esquema de validación cruzada (CV) temporal con ventana móvil (*rolling window-CV*). El testeo se mantiene reservado hasta la etapa de evaluación final. Dado que TiDE utiliza rezagos (*lags*) de las variables como insumo, las observaciones necesarias para el *Input Chunk Length* (ICL) se excluyen del comienzo de cada ventana de validación y del conjunto de testeo, garantizando que el modelo no realice predicciones basadas en información utilizada durante el entrenamiento. La Tabla 3 resume los parámetros temporales del modelo.

Figura 3. Pipeline de PulsoPBA.



Fuente: Elaboración propia.

1.b) Preprocesamiento: Antes de la optimización, cada serie se transforma mediante log-diferenciación para estabilizar varianza y eliminar tendencias nominales -crítico en un contexto de alta inflación- y se normaliza con RobustScaler, menos sensible a outliers que los escaladores estándar. Todas las transformaciones se calculan exclusivamente sobre el conjunto de entrenamiento y se aplican con esos parámetros a validación y testeo, evitando data leakage.

[2] https://github.com/unit8co/darts

2) Optimización de hiperparámetros: en la Tabla 3 se resume la estructura temporal de rezagos y horizonte de predicción empleada. Estos parámetros no formaron parte del espacio de búsqueda de hiperparámetros, sino que se establecieron según criterios prácticos y de estabilidad del modelo.

Tabla 3. Parámetros temporales del modelo

Parámetro	Valor	Interpretación
Input Chunk Length (ICL)	8 semanas	Ventana de historia (2 meses)
Output Chunk Length (OCL)	1 semana	Predicción un paso adelante
Forecast Horizon (FH)	12 semanas en validación / 8 semanas en test	Horizonte máximo (3 meses)

Fuente: Elaboración propia

Optuna[3] (Akiba et al., 2019) ejecuta k=100 *trials* para cada ciclo de entrenamiento, explorando el espacio de búsqueda detallado en la Tabla 4. El optimizador ADAM (Kingma y Ba, 2015) se utiliza con búsqueda conjunta de sus hiperparámetros y los parámetros arquitecturales de TiDE.

Tabla 4. Espacio de búsqueda de hiperparámetros

Parámetro	Mín.	Máx.	Tipo
Learning Rate	1e-5	1e-2	Logarítmico
Beta1	0.85	0.95	Uniforme
Num. Encoder Layers	1	7	Entero
Num. Decoder Layers	1	7	Entero
Hidden Size	10	100	Entero
Batch Size	4	32	Log-entero
Num. Epochs	5	100	Entero

Fuente: Elaboración propia

PulsoPBA incorpora la mitigación del sobreajuste del propio pipeline mediante tres mecanismos de regularización: dropout, early stopping y pruning mediante tres mecanismos de regularización: dropout, early stopping y pruning[4].

[3] https://github.com/optuna/optuna

[4] (i) Dropout: desactiva aleatoriamente una fracción de neuronas durante el entrenamiento, reduciendo la co-dependencia entre conexiones; (ii) Early stopping: que interrumpe el entrenamiento cuando la métrica de validación deja de mejorar durante un número predefinido de épocas; y (iii) Pruning: mediante Optuna, que

Para cada configuración, el error se mide mediante el MAPE mediano sobre las ventanas de validación cruzada (Ecuación 1):

$$MAPE_{median} = median\{\frac{1}{S}\sum_{s=1}^{S}|\frac{PulsoPBA_{w,s} - ESA_{w,s}}{ESA_{w,s}}|,\ w = 1,..,W\} \quad (1)$$

donde S es el número de semanas en cada ventana y W el número de ventanas de validación cruzada. Las 10 configuraciones con menor MAPE en CV se seleccionan como candidatas para evaluación en testeo.

3) *Evaluación en testeo:* Cada una de las 10 configuraciones candidatas es reentrenada utilizando sus hiperparámetros óptimos y la totalidad de los datos asignados a entrenamiento y validación, garantizando que el modelo disponga de toda la información relevante antes de ser evaluado sobre datos no observados previamente. El error en testeo se calcula comparando las estimaciones semanales con los valores observados de ESA-PBA (Ecuación 2):

$$MAPE_{test} = \frac{1}{S}\sum_{s=1}^{S}|\frac{\hat{PulsoPBA}_s - ESA_s}{ESA_s}| \quad (2)$$

Las predicciones sobre el conjunto de testeo se generan de forma probabilística mediante Monte Carlo Dropout (Gal y Ghahramani, 2016): al mantener activo el *dropout* durante la inferencia, se obtienen múltiples muestras que aproximan la distribución predictiva, proporcionando tanto una estimación puntual (mediana) como intervalos de confianza basados en cuantiles. Este método cuantifica la incertidumbre sin requerir arquitecturas adicionales ni sobrecarga computacional significativa, lo que facilita su integración en un pipeline con reentrenamiento mensual.

5.2 Selección de modelos: embudo multi-criterio

La cuarta etapa del entrenamiento consiste en la selección del mejor modelo: un sistema de nowcasting en producción requiere criterios de selección que van más allá del error puntual en la validación. Un modelo puede tener bajo MAPE y aun así ser inadecuado si sus intervalos de confianza no cubren los datos observados, o si predice sistemáticamente la dirección incorrecta de los cambios semanales. El embudo de selección formaliza algorítmicamente criterios de utilidad práctica que reflejan las necesidades de las áreas a cargo de la toma de decisiones.

Filtro 1 - Generalización: Se calcula ΔMAPE = MAPE_CV − MAPE_test y se descarta la mitad inferior de los modelos según esta medida. Este diferencial es un *proxy* de sobreajuste: un modelo que funciona considerablemente mejor en validación que en testeo tiene capacidad de generalización limitada.

---

descarta anticipadamente configuraciones con bajo rendimiento inicial en validación, concentrando el cómputo en regiones prometedoras del espacio de hiperparámetros.

Filtro 2 - Cobertura del intervalo de confianza. Se requiere que al menos el 75% de las observaciones de testeo semanales caigan dentro del intervalo [P5, P95] de las predicciones probabilísticas. Los modelos con cobertura inferior son descartados.
Filtro 3 - Coincidencia de signo: Se mide el porcentaje de semanas en que la dirección predicha (crecimiento o decrecimiento respecto de la semana anterior) coincide con la dirección observada. Se descarta la mitad inferior según esta métrica.
En cada instancia se asegura que al menos un modelo sobreviva al filtro, evitando que el ranking final quede vacío.
Ranking final: Los modelos que superan los tres filtros se ordenan por el Weighted Interval Score (WIS; Bracher et al., 2021), métrica diseñada para predicciones probabilísticas que pondera simultáneamente la amplitud del intervalo, su calibración y la cobertura sobre los datos observados (Ecuación 3):

$$WIS = \frac{1}{K+1}\left(|y - \hat{y}| + \sum_{k=1}^{K} \left[\alpha_k (u_k - l_k) + \frac{2}{\alpha_k}(l_k - y) I_{inf} + \frac{2}{\alpha_k}(u_k - y) I_{sup} \right] \right) \quad (3)$$

donde y es el valor observado, $\hat{y}$ la mediana predicha, K el número de intervalos de confianza ponderados, l_k y u_k los límites inferior y superior del k-ésimo intervalo, e I_{inf}, I_{sup} las funciones indicadoras de excedencia. Un WIS menor indica mejor calidad predictiva probabilística. Esta métrica prioriza modelos con intervalos bien calibrados y estables fuera de muestra, por sobre modelos con mínimos locales de error en validación cruzada. El modelo seleccionado por WIS es el que se despliega en producción para el ciclo de predicciones del mes correspondiente.

5.3 Paso a producción y generación de predicciones

Una vez identificado el modelo con menor WIS, se procede al entrenamiento definitivo. Esta fase consiste en reentrenar el modelo seleccionado utilizando la totalidad de los datos disponibles —conjuntos de entrenamiento, validación y testeo— junto con la configuración de hiperparámetros óptima. De este modo, el modelo aprovecha toda la información histórica para ajustar sus parámetros internos (pesos y sesgos) y consolidar los patrones más relevantes en la relación entre las variables bancarias y la actividad económica.
Completado este paso, el modelo queda desplegado en el entorno productivo y genera predicciones bloque-semanales de forma automática. Los nowcasts producidos son probabilísticos: cada predicción semanal incluye una estimación puntual (mediana de las muestras de *Monte Carlo Dropout*) y sus correspondientes intervalos de confianza, lo que permite cuantificar la incertidumbre asociada a cada estimación y comunicar a las áreas de decisión no solo el nivel esperado de actividad sino también el grado de confiabilidad de esa estimación.
Este ciclo completo -desde la incorporación del nuevo dato oficial hasta la generación de predicciones- se repite mensualmente sin intervención humana, constituyendo el núcleo operativo de PulsoPBA. La Sección 6 presenta los resultados numéricos de este sistema en operación.

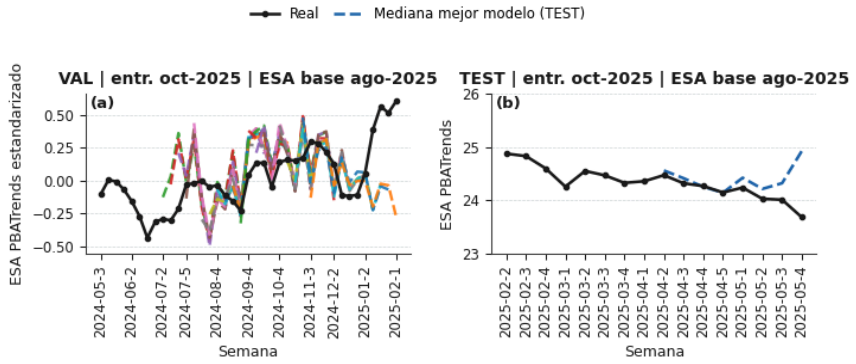
6. Resultados: validación operativa

En esta sección se presentan los resultados numéricos del sistema en operación. Los resultados corresponden a cuatro ciclos consecutivos de entrenamiento ejecutados entre Octubre de 2025 y Enero de 2026, denominados por el mes del último dato oficial incorporado: ESA-Agosto, ESA-Septiembre, ESA-Octubre y ESA-Noviembre.

6.1 Desempeño en entrenamiento: validación cruzada y testeo

Para cada ciclo de entrenamiento, de las 100 configuraciones exploradas por Optuna se seleccionaron las 10 con menor MAPE en validación cruzada. Estas fueron evaluadas sobre el conjunto de testeo reservado (ver Gráfico 2).

Gráfico 2: Comparación de predicciones en los conjuntos de validación y testeo del modelo ganador para el ESA-Agosto



Nota: Las predicciones de distintos colores en el conjunto de validación corresponden a distintas predicciones del rolling window CV.
Fuente: elaboración propia.

Posteriormente los 10 mejores modelos son evaluados en el embudo de selección multi-criterio. Finalmente, en el subconjunto de modelos que atravesaron el embudo, se selecciona para la puesta en producción aquel que cuenta con menor WIS.

La Tabla 5 presenta las métricas del modelo puesto en producción en cada uno de los cuatro ciclos. Es importante notar que cada fila de la Tabla 5 no representa una observación puntual: el MAPE test de cada ciclo es el promedio del error absoluto porcentual sobre las S=8 semanas del conjunto de testeo (Ecuación 2), y las demás métricas (cobertura, coincidencia de signo) se computan igualmente sobre esas 8 observaciones semanales.

Tabla 5. Métricas de los modelos seleccionados

Entrenamiento	Cobertura IC [P5-P95]	% Mismo signo	MAPE test	ΔMAPE	WIS
ESA-Agosto	87%	71%	1.07%	0.14	0.35
ESA-Septiembre	87%	85%	0.49%	0.35	1.10
ESA-Octubre	100%	43%	0.49%	1.15	0.71

ESA-Noviembre	100%	43%	1.24%	0.14	1.20

Fuente: elaboración propia

La Tabla 6 muestra los hiperparámetros óptimos del modelo seleccionado en cada ciclo. La variabilidad entre configuraciones ganadoras, con el número de capas del encoder oscilando entre 2 y 6, y el *hidden size* entre 20 y 51, evidencia que no existe una configuración universalmente óptima: el sistema se beneficia del reentrenamiento y la re-optimización mensual para adaptarse a las condiciones cambiantes de las series.

Tabla 6. Hiper Parámetros de los modelos seleccionados en cada entrenamiento

Entrenamiento	Num Encoder Layers	Num Decoder Layers	Hidden Size	Batch Size	Epochs
ESA-Agosto	6	4	48	32	40
ESA-Septiembre	3	3	20	16	44
ESA-Octubre	4	4	51	32	42
ESA-Noviembre	2	3	28	32	36

Fuente: elaboración propia

6.2 Validación contra datos oficiales publicados

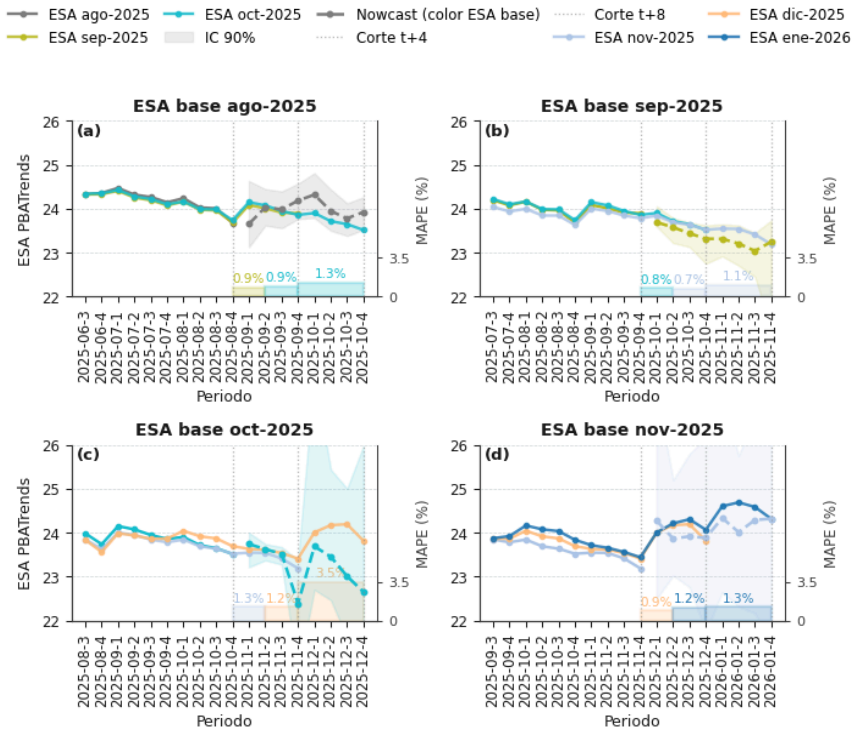
La validación más exigente para un sistema en producción es la comparación con datos oficiales publicados después de que las predicciones fueron generadas. El Gráfico 3 muestra las predicciones de cada entrenamiento junto con las series observadas del ESA-PBA *ex post* correspondientes a los meses cubiertos por los cuatro ciclos.

En la validación operativa, el ciclo ESA-Agosto generó nowcasts para septiembre y octubre de 2025; ESA-Septiembre para octubre y noviembre; ESA-Octubre para noviembre y diciembre; y ESA-Noviembre para diciembre de 2025 y enero de 2026. Al contrastar estas predicciones con los ESA publicados *ex post*, los errores mensuales MAPE se ubicaron entre 0,7% y 3,5%. Los mejores desempeños se observaron en los ciclos ESA-Septiembre y ESA-Agosto, con errores cercanos o inferiores a 1% en el primer mes de predicción. El mayor error corresponde al segundo mes del ciclo de entrenamiento ESA-Octubre (3.5%) -diciembre de 2025- en un contexto de alta volatilidad económica, consistente con el aumento de incertidumbre al extender el horizonte a ocho semanas.

Los intervalos probabilísticos obtenidos mediante *Monte Carlo Dropout* también resultan informativos para capturar la volatilidad. En los ciclos ESA-Octubre y ESA-Noviembre los intervalos se amplían con mayor intensidad hacia el segundo mes de predicción, justamente donde el contraste *ex post* muestra mayor error relativo.

Este patrón refuerza el valor de reportar nowcasts probabilísticos: la amplitud del intervalo aporta una información sobre la confiabilidad esperada de la estimación

Gráfico 3. Nowcasts y series observadas ex post.



Nota: Las líneas sólidas representan datos observados semanalizados; las líneas con marcadores, los nowcasts de cada entrenamiento. Las bandas sombreadas corresponden a los intervalos de confianza al 90%.
Fuente: Elaboración propia

7. Conclusiones

Este trabajo presenta PulsoPBA, un sistema automatizado de nowcasting semanal desplegado en producción en el Banco de la Provincia de Buenos Aires. El sistema procesa ~1.200 series bancarias diarias y genera predicciones probabilísticas de granularidad semanal con MAPE mensual de entre 0.7% y 3.5% sobre datos oficiales publicados *ex post.*

Las contribuciones de este trabajo son dos. Primero, se expone el diseño, implementación y validación de PulsoPBA como sistema *end-to-end* en producción. El pipeline integra desagregación temporal para construir la variable objetivo

semanal, la arquitectura TiDE —cuya compresión temprana del espacio de entrada hacia una representación latente permite el entrenamiento estable con más de mil covariables heterogéneas sin ingeniería de atributos previa al entrenamiento—, optimización automática de hiperparámetros con Optuna, y un embudo de selección multi-criterio basado en el *Weighted Interval Score* (WIS) que formaliza criterios de utilidad práctica —generalización, cobertura de intervalos de confianza, coincidencia de signo y calidad probabilística— superando la selección por error puntual. Las predicciones se generan de forma probabilística mediante *Monte Carlo Dropout*, cuantificando la incertidumbre sin sobrecarga computacional significativa. El ciclo completo de reentrenamiento y selección se ejecuta mensualmente automáticamente.

Segundo, se validan empíricamente las predicciones contra datos oficiales del EMA-PBA publicados ex post, demostrando la factibilidad de un sistema de nowcasting de alta frecuencia construido sobre datos bancarios de alta dimensionalidad.

El sistema presenta limitaciones que deben considerarse al interpretar los resultados. El período de validación operativa abarca cuatro ciclos de entrenamiento (octubre 2025 – enero 2026), un horizonte que no incluye ciclos estacionales ni eventos macroeconómicos de naturaleza diversa; la extensión de esta ventana permitirá evaluar la robustez del sistema. Asimismo, la dependencia exclusiva de datos del Banco Provincia implica que cambios en la representatividad del banco respecto de la actividad provincial podrían afectar la calidad predictiva.

Como trabajo futuro se identifican: (1) técnicas de interpretabilidad para incrementar la trazabilidad del modelo, aspecto relevante tanto para la investigación en aprendizaje automático como para la política económica basada en evidencia; y (2) la extensión a modelos multi-output que estimen componentes desagregados de la actividad provincial por sector o región.

Referencias.

Akiba, T., Sano, S., Yanase, T., Ohta, T., & Koyama, M. (2019). Optuna: A next-generation hyperparameter optimization framework. arXiv:1907.10902.

Aprigliano, V., Ardizzi, G., & Monteforte, L. (2017). Using the payment system data to forecast the Italian GDP (Temi di discussione, No. 1098). Banca d'Italia.

Bańbura, M., & Modugno, M. (2014). Maximum likelihood estimation of factor models on datasets with arbitrary pattern of missing data. Journal of Applied Econometrics, 29(1), 133–160.

Barlas, A. B., Mert, S. G., Isa, B. O., Ortiz, Á., Rodrigo, T., Soybilgen, B., & Yazgan, E. (2021). Big data information and nowcasting: Consumption and investment from bank transactions in Turkey. arXiv:2107.03299.

Benidis, K., Rangapuram, S. S., Flunkert, V., Wang, Y., Maddix, D., Turkmen, C., & Januschowski, T. (2022). Deep learning for time series forecasting: Tutorial and literature survey. ACM Computing Surveys, 55(6), 1–36.

Blanco, E., Dogliolo, F., & Garegnani, L. (2022). Nowcasting durante la Pandemia: un apartado para Argentina (Documentos de Trabajo, No. 99). BCRA.

Bracher J., Ray E., Gneiting T. & Reich, N. (2021) Evaluating epidemic forecasts in an interval format. PLoS Comput Biol 17(2): e1008618.

Carrera, G. (2024). Anticipar el EMAE para decidir mejor: modelos de Nowcasting para el pronostico de la actividad economica mensual argentina. Trabajo Final de Posgrado, Universidad de Buenos Aires, Facultad de Ciencias Economicas.

Chen, S.-A., Li, C.-L., Yoder, N., Arik, S. O., & Pfister, T. (2023). TSMixer: An all-MLP architecture for time series forecasting. arXiv:2303.06053.

Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 785–794). ACM.

Chow, G. C., & Lin, A. (1971). Best linear unbiased interpolation, distribution, and extrapolation of time series by related series. The Review of Economics and Statistics, 53(4), 372–375.

D'Amato, L., Garegnani, L., & Blanco, E. (2016). Nowcasting de PIB: evaluando las condiciones ciclicas de la economia argentina. Ensayos Economicos, 1(74), 7-26. Banco Central de la Republica Argentina.

Das, A., Kong, W., Leach, A., Mathur, S., Sen, R., & Yu, R. (2023). TiDE: Long-term time-series forecasting with time-series dense encoder. arXiv:2304.08424.

Doerr, S., Gambacorta, L., & Serena Garralda, J. M. (2021). Big data and machine learning in central banking (BIS Working Papers No. 930). Bank for International Settlements. https://www.bis.org/publ/work930.htm

Gal, Y., & Ghahramani, Z. (2016). Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. 33rd ICML, 1050–1059.

Giannone, D., Reichlin, L., & Small, D. (2008). Nowcasting: The real-time informational content of macroeconomic data. Journal of Monetary Economics, 55(4), 665–676. https://doi.org/10.1016/j.jmoneco.2008.05.010

Herzen, J., Lässig, F., Piazzetta, S. G., Neuer, T., Tafti, L., Raille, G., Van Pottelbergh, T. (2022). Darts: User-friendly modern machine learning for time series. Journal of Machine Learning Research, 23(124), 1–6.

Huber, F., Koop, G., Onorante, L., Pfarrhofer, M., & Schreiner, J. (2021). Nowcasting in a pandemic using non-parametric mixed frequency VARs (Working Paper Series, No. 2510). European Central Bank. https://doi.org/10.2866/75873

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., & Liu, T.-Y. (2017). LightGBM: A highly efficient gradient boosting decision tree. In Advances in Neural Information Processing Systems 30 (pp. 3146–3154). Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. Proceedings of the 3rd ICLR.

León, C., & Ortega, F. (2018). Nowcasting economic activity with electronic payments data: A predictive modeling approach. Revista de Economía del Rosario, 21(2), 381–407.

Lim, B., & Zohren, S. (2021). Time-series forecasting with deep learning: A survey. Philosophical Transactions of the Royal Society A, 379(2194), 20200209.

Linzenich, J., & Meunier, B. (2024). Nowcasting Made Easier: A Toolbox for Economists (Working Paper Series No. 3004). European Central Bank. https://doi.org/10.2139/ssrn.5060436

Oukhouya, H., Kadiri, H., El Himdi, K., & Guerbaz, R. (2024). Forecasting international stock market trends: XGBoost, LSTM, LSTM-XGBoost, and backtesting XGBoost models. Statistics, Optimization & Information Computing, 12(1), 200-209.

Sax, C., & Steiner, P. (2013). Temporal Disaggregation of Time Series. The R Journal, 5(2), 80–87.

Zeng, A., Chen, M., Zhang, L., & Xu, Q. (2023). Are Transformers Effective for Time Series Forecasting? Proceedings of the AAAI Conference on Artificial Intelligence, 37(9), 11121-11128.