Banking Customer Churn Prediction

* André Esteban Vera #745232

* Nicolás Martínez Gutiérrez #751746

* Gonzalo Cano Padilla #745901

Objetivo General

- Desarrollar un modelo supervisado para predecir el abandono de clientes bancarios (churn) a partir de sus características demográficas y financieras, maximizando el F1 (con foco en la clase minoritaria “Churn”) y garantizando robustez mediante validación estratificada y optimización de hiperparámetros.

Objetivos Específicos

1. Caracterizar el dataset : describir variables, distribución de Exited, y posibles patrones/atípicos que afecten el abandono.
2. Definir el problema y la variable objetivo (Exited ∈ {0,1}) y preparar las features eliminando identificadores para evitar fuga de información.
3. Implementar un preprocesamiento reproducible con ColumnTransformer:
4. Abordar el desbalanceo de clases evaluando ponderación de clases en los modelos y comparando su impacto sobre recall y F1 de la clase Churn.
5. Entrenar y optimizar modelos base y avanzados:
6. - Regresión Logística y SVM (kernel RBF) integrados en pipelines con el mismo preprocesamiento.
7. Optimización Bayesiana para seleccionar hiperparámetros que maximicen F1.
8. Evaluar con validación robusta (StratifiedKFold=5) usando F1 como métrica principal, y reportes complementarios (accuracy, precision, recall, matriz de confusión y classification_report out-of-fold con cross_val_predict).
9. Comparar el desempeño entre modelos y seleccionar el mejor con base en F1 y el equilibrio entre precision/recall de Churn.
10. Comunicar hallazgos y limitaciones: interpretar resultados, discutir variables relevantes/efectos (cuando aplique, p.ej. coeficientes en logística) y recomendaciones para su aplicación operativa (detección temprana y retención de clientes).

Marco Teórico

- **Regresión Logística:** modelo estadístico para *clasificación binaria* que estima la probabilidad de abandono mediante una función sigmoide. Ofrece interpretabilidad sobre el impacto de variables como edad, saldo o historial.
- **Máquinas de Vectores de Soporte (SVM):** buscan el *hiperplano óptimo* que maximiza el margen entre clases.
  - Cuando los datos no son linealmente separables, se usa un kernel, siendo el RBF el más común:
  - Este kernel permite detectar patrones no lineales de abandono en los clientes.
- **Redes Neuronales (MLP):** modelos *feed-forward* con capas ocultas que capturan **r**elaciones complejas y no lineales entre variables financieras y demográficas. Aprenden mediante retropropagación y una función de pérdida como la *entropía cruzada*.
- **Hiperparámetros:** configuran el comportamiento de los modelos antes del entrenamiento.
  - *SVM:* C (margen) y γ (influencia de los puntos).
  - *MLP:* número de capas, neuronas, tasa de aprendizaje, etc.
  - *Reg. Logística:* C (regularización).
- **Optimización Bayesiana:** método para ajustar hiperparámetros usando *procesos gaussianos*, eligiendo combinaciones de forma inteligente y eficiente para mejorar el rendimiento

Métrica de Evaluación F1 - Score

Precision = TP / (TP + FP)

Recall = TP / (TP + FN)

F1 = 2 * (Precision * Recall) / (Precision + Recall)

Análisis de Dataset

- **Origen:**
  Proviene de Kaggle y fue publicado por Saurabh Badole bajo el nombre “Banking Customer Churn Prediction Dataset”. Contiene datos de clientes de un banco europeo para predecir si se irán o permanecerán en la institución.
- **Contenido:**
  El dataset tiene 10,000 registros (uno por cliente) y 14 columnas con información demográfica y financiera, como:
- CreditScore, Age, Balance, EstimatedSalary
- Geography (país), Gender, Tenure
- NumOfProducts, HasCrCard, IsActiveMember
- Exited = variable objetivo (1 = se fue, 0 = se quedó)
- **Propósito del análisis:**
  Predecir la probabilidad de abandono de clientes (churn) para ayudar al banco a identificar y retener clientes en riesgo.
- **Transformaciones necesarias:**
- Estandarización (StandardScaler) → para variables numéricas.
- Codificación categórica (OneHotEncoder) → para variables de texto como Geography y Gender.
- **Resultado esperado:**
  Un modelo capaz de estimar la probabilidad de que un cliente abandone, destacando factores clave como la edad, el puntaje crediticio, el país y la actividad del cliente.

Regresión Logística con regularización

Mejores hiperparámetros:

- Penalty = Elasticnet
- Solver = saga
- C = 0.114
- Class_weight = balanced

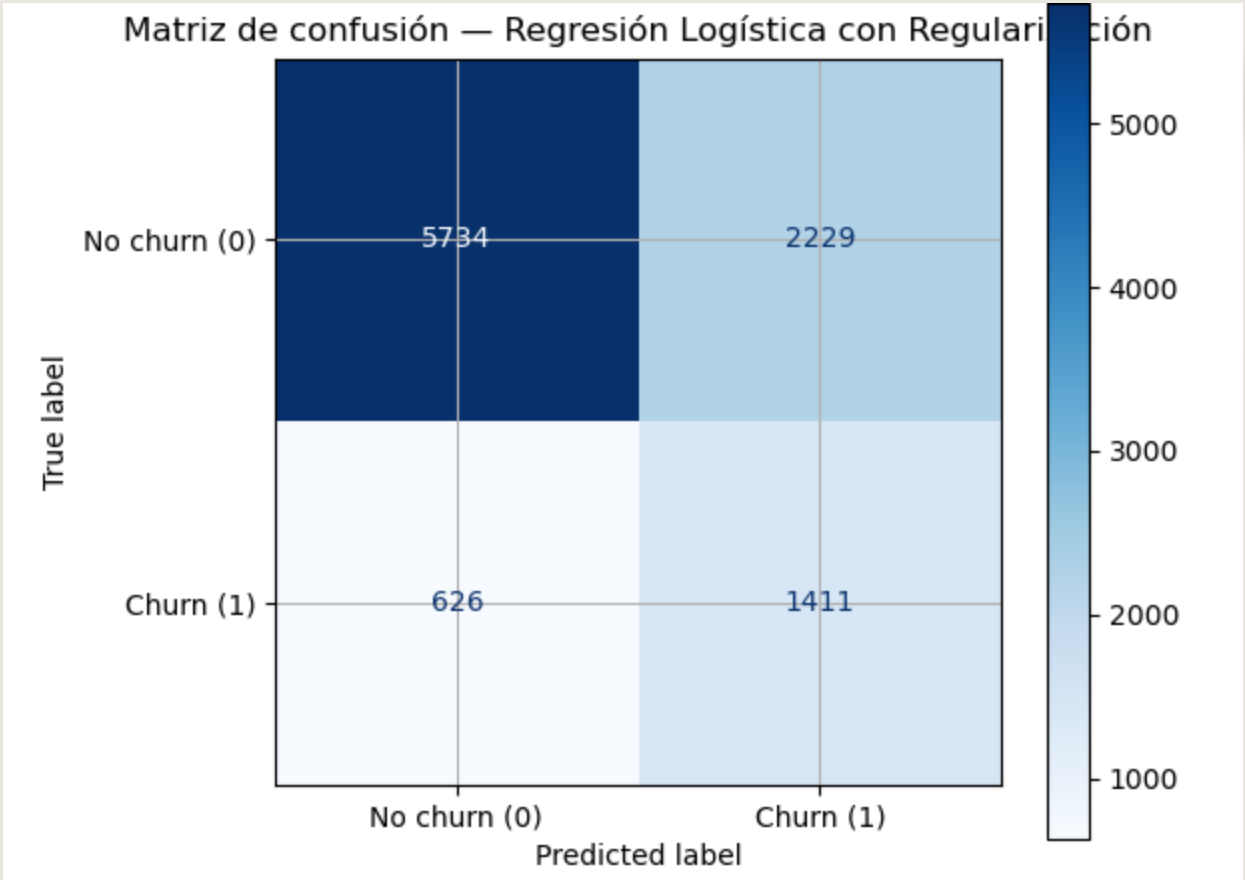
Desempeño Promedio

- F1: 0.4975 ± 0.0167
- Accuracy: 0.7145
- Precision (Churn): 0.39
- Recall (Churn): 0.69

Desempeño global

Métrica	Valor	Interpretación
Accuracy	0.71	El modelo acierta en el **71 %** de las predicciones totales.
Precisión (True / Churn)	0.39	De los clientes que el modelo predijo que abandonarían, el **39 %** realmente se fue.
Recall (True / Churn)	0.69	El modelo identifica correctamente al **69 %** de los clientes que efectivamente abandonan.
F1 (True / Churn)	0.50	Representa un **equilibrio moderado entre precisión y sensibilidad** para la clase de abandono.
F1 macro promedio	0.65	Indica un rendimiento **balanceado entre ambas clases.**
F1 ponderado promedio	0.74	Refleja un **buen rendimiento global**, considerando el desbalance entre clientes que se quedan y los que abandonan.

Matriz de Confusión Regresión Logística



Máquina de vectores de soporte con kernel RBF

Mejores Hiperparámetros
C = 358.7964
Gamma = 0.01047
Class-weight = 'Balanced'

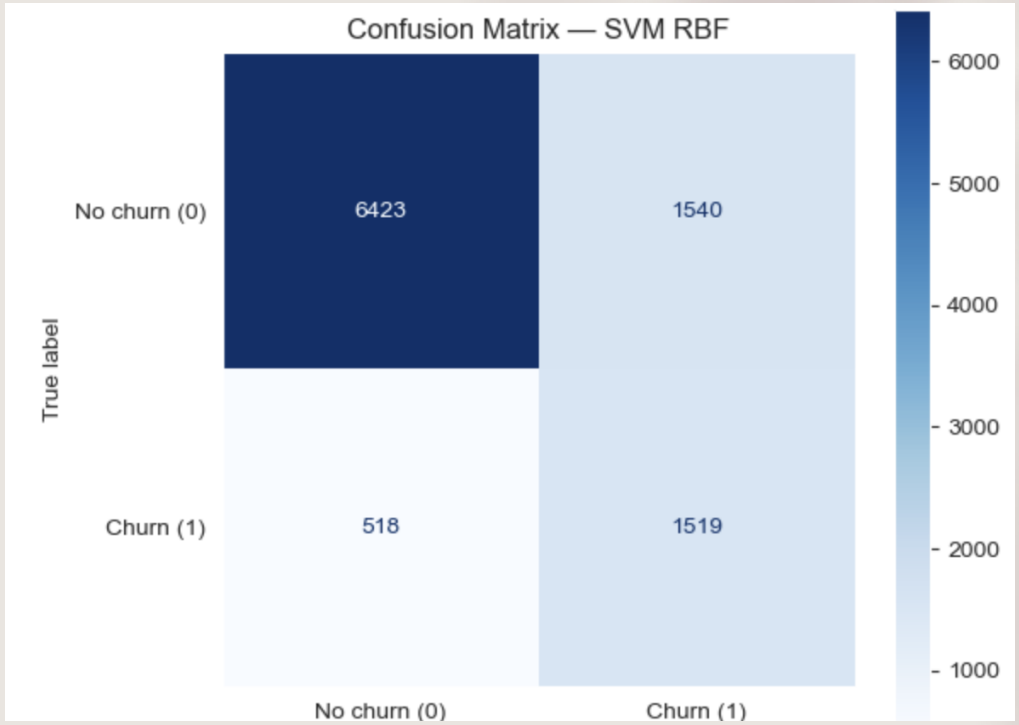
Scores SVM

- Reporte de clasificación (OOF)

Clase	Precision	Recall	F1	Support
No churn (0)	0.9254	0.8066	0.8619	7,963
Churn (1)	0.4966	0.7457	0.5962	2,037
Accuracy	–	–	**0.7942**	**10,000**
Macro avg	0.7110	0.7762	0.7290	10,000
Weighted avg	0.8380	0.7942	0.8078	10,000

Matriz de Confusión SVM

- Verdaderos Negativos (6423): Clientes correctamente identificados como no propensos a abandonar.
- Falsos Positivos (1540): Clientes que el modelo clasificó como “churn” pero en realidad permanecieron.
- Falsos Negativos (518): Clientes que abandonaron, pero el modelo no logró anticipar.
- Verdaderos Positivos (1519): Casos de abandono correctamente detectados.



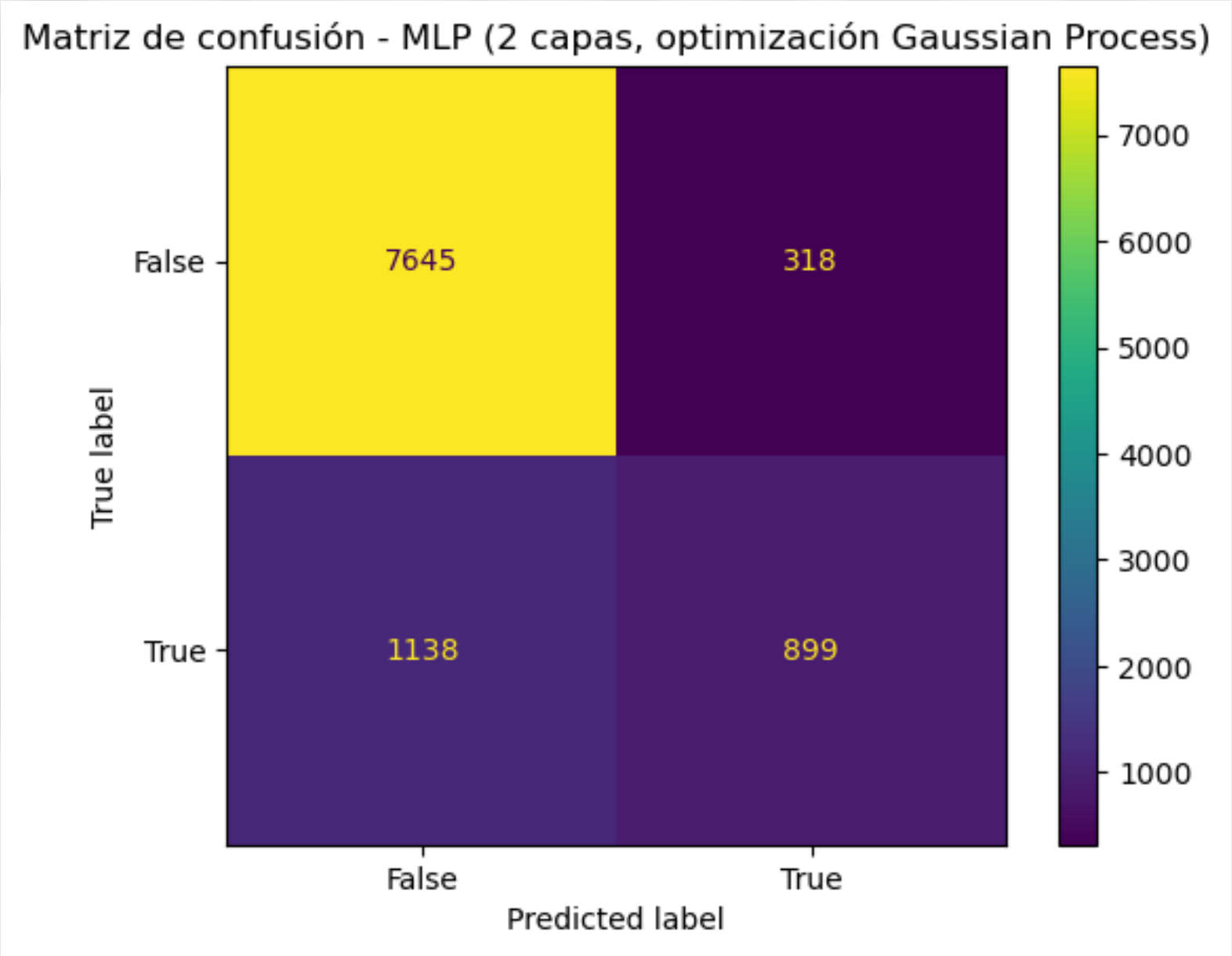
MODELO MULTI-LAYER PERCEPTRON (MLP)

Optimización de hiperparámetros vía procesos gaussianos

Hiperparámetro	Descripción	Valor óptimo
`hidden_layer_sizes`	Estructura de la red: define el número de neuronas en cada capa oculta.	(255, 127)
`alpha`	Parámetro de regularización L2. Controla el tamaño de los pesos del modelo para prevenir sobreajuste.	0.01000
`learning_rate_init`	Tasa de aprendizaje inicial del optimizador. Define qué tan grandes son los pasos durante la actualización de pesos.	0.03728
`solver`	Algoritmo de optimización utilizado para el aprendizaje de pesos.	*Adam*
`activation`	Función de activación aplicada a las neuronas ocultas.	*logistic (sigmoide)*
`max_iter`	Número máximo de iteraciones de entrenamiento.	1000

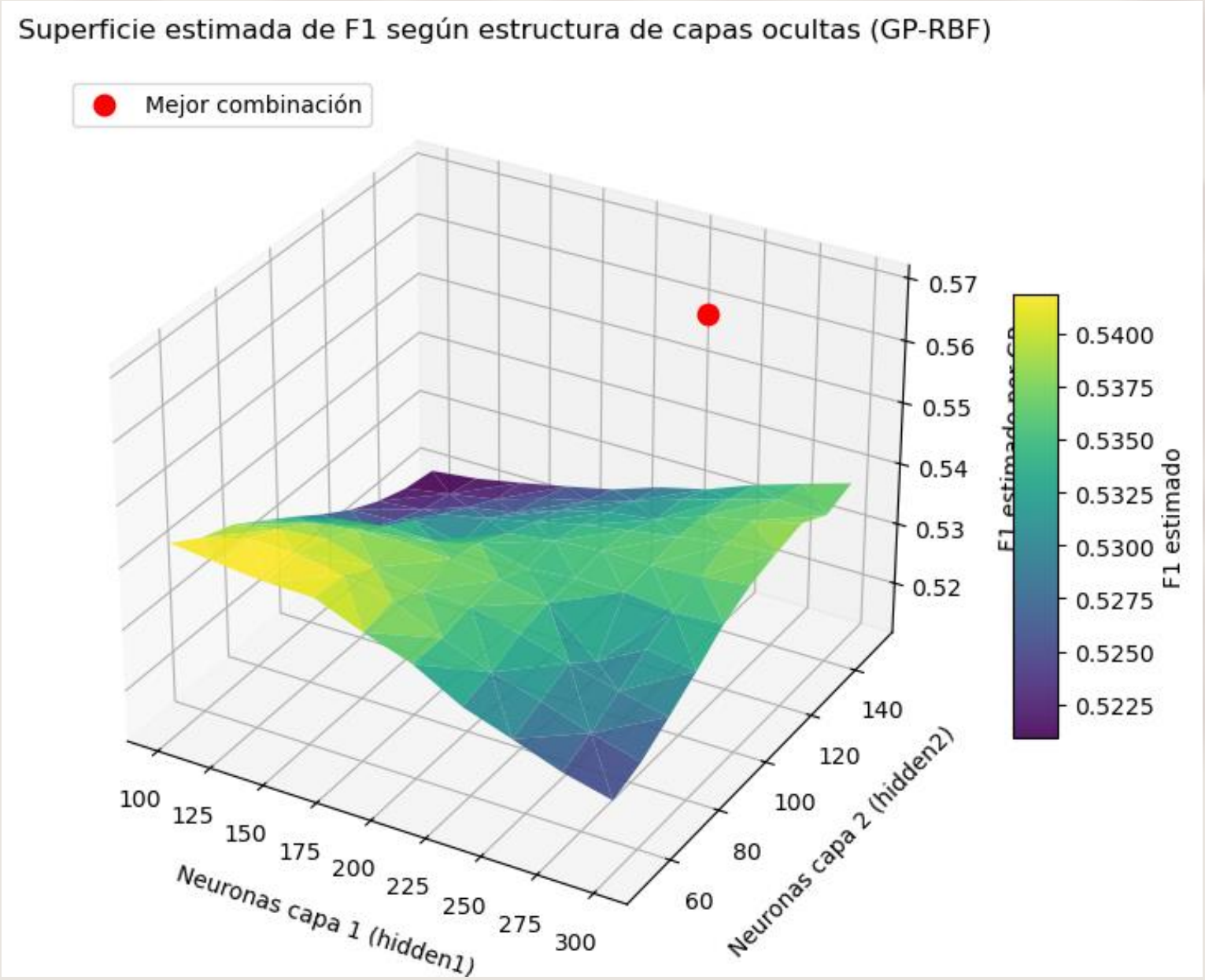
Desempeño Global

Métrica	Valor	Interpretación
Accuracy	**0.85**	El modelo acierta en el 85 % de las predicciones totales.
Precision (True)	**0.74**	De los clientes que el modelo predijo que abandonarían, el 74 % realmente se fue.
Recall (True)	**0.44**	El modelo identifica al 44 % de los clientes que efectivamente abandonan.
F1 (True)	**0.55**	Representa un equilibrio razonable entre precisión y sensibilidad.
F1 macro promedio	**0.73**	Indica un rendimiento balanceado entre ambas clases.
F1 ponderado promedio	**0.84**	Refleja un buen rendimiento global considerando el desbalance de clases.



Matriz de Confusión MLP

Visualización 3D del Espacio de Búsqueda



Comparación de modelos

Desempeño General

Modelo	Accuracy	F1 Promedio (CV)	F1 (Churn)	Precisión (Churn)	Recall (Churn)
Regresión Logística	0.714	0.498 ± 0.017	0.497	0.388	0.692
SVM RBF Optimizado	**0.794**	**0.599 ± 0.020**	**0.596**	0.497	**0.746**
MLP Optimizado	**0.85**	0.5513 ± 0.0228	0.5682	**0.74**	0.44

Modelo	Verdaderos Negativos (TN)	Falsos Positivos (FP)	Falsos Negativos (FN)	Verdaderos Positivos (TP)
Regresión Logística	5784	2229	626	1411
SVM RBF	6423	1540	**518**	**1519**
MLP	**7645**	**318**	1138	899

Matrices de Confusión

Mejores Hiperparámetros Obtenidos por Modelo

Modelo	Hiperparámetros Óptimos
Regresión Logística	`{ 'penalty': 'elasticnet', 'solver': 'saga', 'class_weight': 'balanced', 'C': 0.114 }`
SVM con kernel RBF	`{ 'C': 358.796, 'gamma': 0.0105, 'class_weight': 'balanced', 'shrinking': False }`
MLP Optimizado	`{ 'activation': 'logistic', 'alpha': 0.001, 'hidden_layer_sizes': 255, 127, 'learning_rate_init': 0.3728, 'solver': 'adam' }`

- En conjunto, el SVM RBF optimizado se considera el modelo más balanceado y operativo, con una alta capacidad para detectar clientes que abandonan sin comprometer excesivamente la estabilidad del modelo.

Conclusión General

- El proyecto cumplió con los objetivos planteados, al construir un modelo supervisado capaz de predecir el abandono de clientes bancarios mediante un flujo completo: análisis del dataset, preprocesamiento, optimización y validación.
- Se identificaron las variables más influyentes (edad, balance, productos y membresía activa) y se aplicaron técnicas para corregir el desbalance de clases.
- De los modelos evaluados, el SVM con kernel RBF mostró el mejor equilibrio entre precisión y recall, cumpliendo el objetivo de crear un modelo robusto, confiable y útil para anticipar el churn.