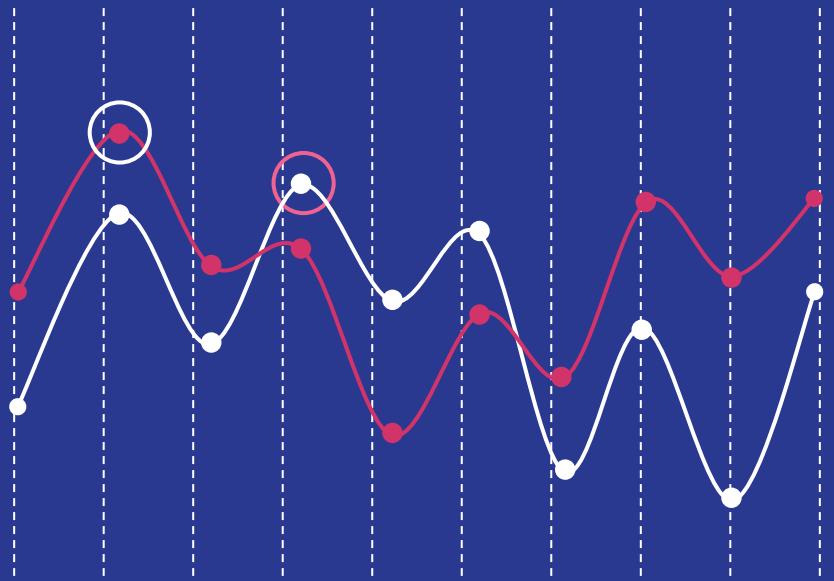
Minería de datos.

Preprocesamiento y Clasificación.

Dataset: MetroPT-3
Grupo: Data Mavericks.

- Brian Sena Simons
- Miguel García López
- Álvaro Santana Sánchez
- Ana Fuentes Rodríguez

Ciencia de Datos
Universidad de Granada



https://github.com/briansenas/MineriaMetroPT-3

Definición del problema

Análisis de Datos

Ventana Deslizante

Conjunto de datos

Vectores Soporte (Ana Fuentes)

Clasificador Bayesiano (Brian Sena)

Árboles (Miguel García)

Regresión Logística (Álvaro Santana)

Bagging (Ana Fuentes)

Stacking (Brian Sena)

Boosting (Miguel García)

AdaBoost (Álvaro Santana)

Detección de anomalías

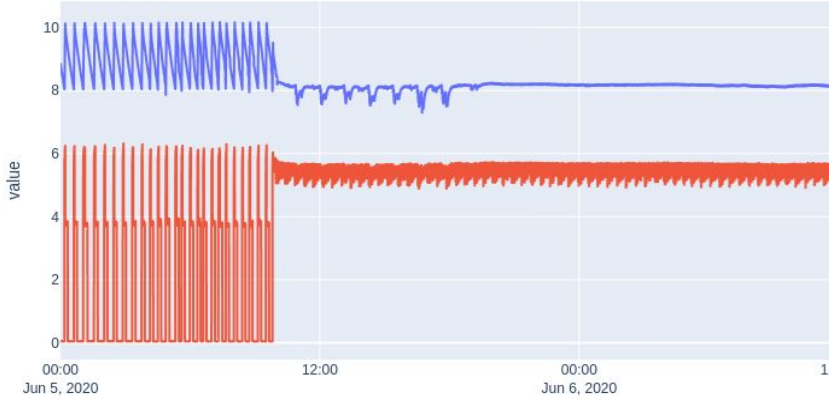


Figura 1: Visualización de datos

Variable	Tipo	Mín.	Q1	Q2	Media	Q3	Máx.
TP2	Numérico	-0.032	-0.014	-0.012	1.368	-0.010	10.676
TP3	Numérico	0.730	8.492	8.960	8.985	9.492	10.302
H1	Numérico	-0.036	8.254	8.784	7.568	9.374	10.288
DV pressure	Numérico	-0.032	-0.022	-0.020	0.05596	-0.018	9.844
Reservoirs	Numérico	0.712	8.494	8.960	8.985	9.492	10.300
Oil temperature	Numérico	15.40	57.77	62.70	62.64	67.25	89.05
Motor current	Numérico	0.020	0.040	0.045	2.050	3.808	9.295
COMP	Numérico	0.000	1.000	1.000	0.837	1.000	1.000
DV eletric	Numérico	0.000	0.000	0.000	0.1606	0.000	1.000
Towers	Numérico	0.000	1.000	1.000	0.9198	1.000	1.000
MPG	Numérico	0.000	1.000	1.000	0.8327	1.000	1.000
LPS	Numérico	0.000	0.000	0.000	0.00342	0.000	1.000
Pressure switch	Numérico	0.000	1.000	1.000	0.9914	1.000	1.000
Oil level	Numérico	0.000	1.000	1.000	0.9042	1.000	1.000
Caudal impulses	Numérico	0.000	1.000	1.000	0.9371	1.000	1.000

Tabla 1: Información básica de los diferentes tipos de datos presentes en MetroPT-3 [1]

- Información de sensores
- Datos pseudo-cíclicos, salvo anomalías

[1] Narjes Davari et al. MetroPT-3 Dataset.

Preprocesamiento

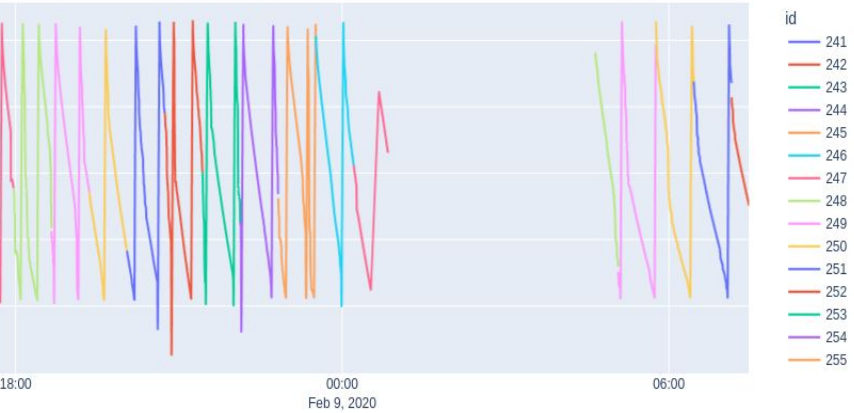


Figura 1: Visualización de la ventana.

Para incorporar temporalidad, se estima una ventana deslizante equivalente a la mediana del tiempo de activación del motor.

Durante el estudio de dicha ventana, se hallan nuevas anomalías además de saltos temporales.

Se estiman características como media, mediana, mínimo, máximo y varianza.

Detalles Experimentación

Es necesario crear las particiones.

Se eliminan discontinuidades temporales.

Se generan pliegues de tamaño equitativo.

Se entremezclan distintos intervalos de tiempo.

Pliegue	Negativo	Positivo	Conjunto
0	635	31	Evaluación
1	635	31	Entrenamiento pliegue 1
2	635	31	Entrenamiento pliegue 2
3	635	31	Entrenamiento pliegue 3
4	635	31	Entrenamiento pliegue 4
5	635	31	Entrenamiento pliegue 3
6	635	31	Entrenamiento pliegue 2
7	638	31	Entrenamiento pliegue 1
8	641	43	Evaluación

Tabla 1: Los 9 pliegues generados.

Ana Fuentes Rodríguez

Máquinas de Vectores de Soporte

Asunciones:

- Separabilidad de las clases.
- Representatividad de los Vectores de Soporte.
- Adecuación del kernel.
- Escalas comparables entre características.

Preprocesamiento:

- Eliminación de saltos temporales.
- No hay valores faltantes.
- Escalado de los datos

Experimentación:

- Parámetros estudiados:
  C: [0.1, 1, 10]
  gamma: [auto, scale, 0.1]
  kernel: [linear, rbf]

Máquinas de Vectores de Soporte

Parámetro	Posibles valores		
C	0.1	1	10
γ	scale	auto	0.1
kernel	linear	rbf	

Tabla 1: Selección de parámetros

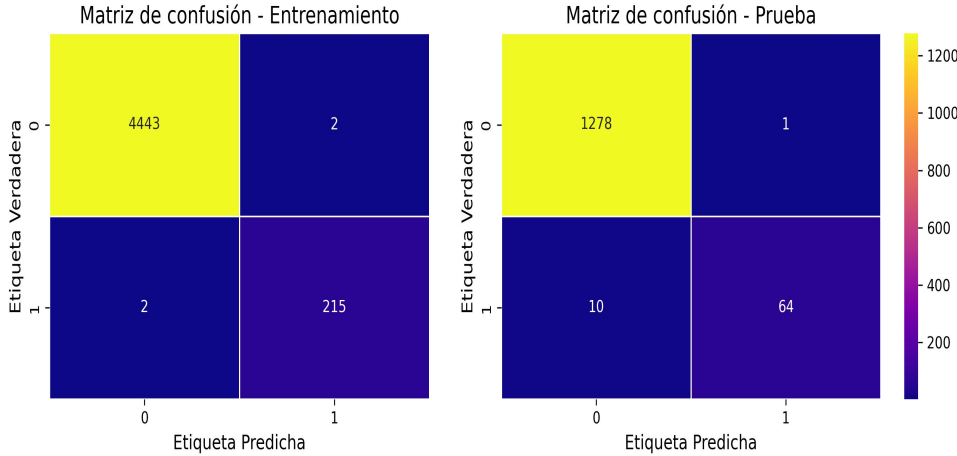


Figura 1: Resultado mejor modelo en entrenamiento y test

Modelo	Parámetros			Precisión	Sensibilidad	F1-score
SVM	C: 10	γ: auto	kernel: rbf	0.985	0.93	0.96

Tabla 2: Resultados mejor modelo.

Bagging (Random Forest, ExtraTrees y Random Subspaces)

Asunciones:
- Alta varianza en los modelos base.
- Diversidad en los modelos base.
- Errores no correlacioados
- Tamaño suficiente del dataset..
- Modelos base independientes.

Preprocesamiento:
- Eliminación de saltos temporales.
- No hay valores faltantes.
- No es necesario escalar los datos

Experimentación:
- Random Forest.
  Parámetros: n_estimators, max_depth, min_samples_split, min_samples_leaf.
- ExtraTrees.
- Parámetros: n_estimators, max_depth, min_samples_split, min_samples_leaf.
- Random Subspaces.
  Parámetros: n_estimators, estimator_max_depth, max_features.

Bagging (Random Forest)

Parámetro	Posibles valores		
n_estimatorss	100	200	300
max_depth	None	10	20
min_samples_split	2	5	10
min_samples_leaf	1	2	4

Tabla 1: Selección de parámetros

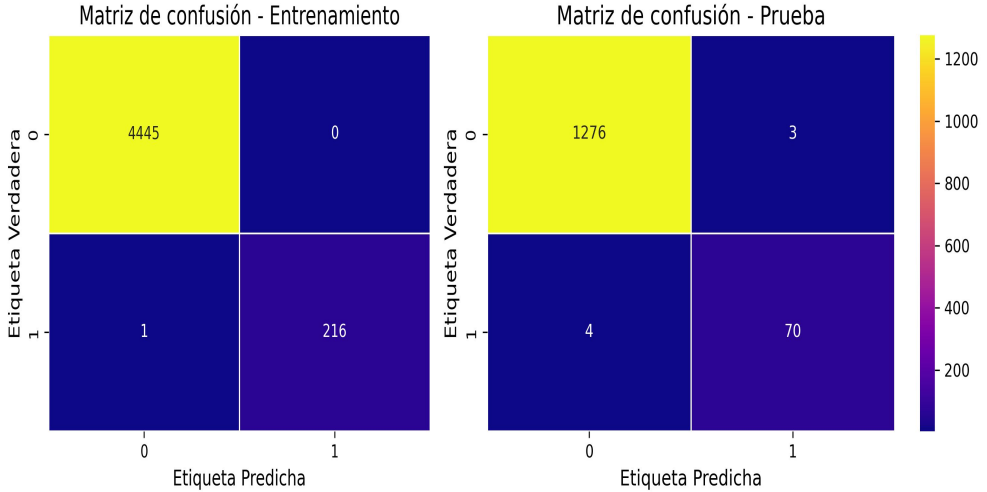


Figura 1: Resultado mejor modelo en entrenamiento y test

Modelo	Parámetros		Precisión	Sensibilidad	F1-score
Random Forest	max_depth: 10 min_samples_leaf: 1	min_samples_split: 5 n_estimators: 100	0.98	0.975	0.975

Tabla 2: Resultados mejor modelo.

Bagging (ExtraTrees)

Parámetro	Posibles valores	
n_estimatorss	100	200
max_depth	None	10
min_samples_split	2	5
min_samples_leaf	1	2

Tabla 1: Selección de parámetros

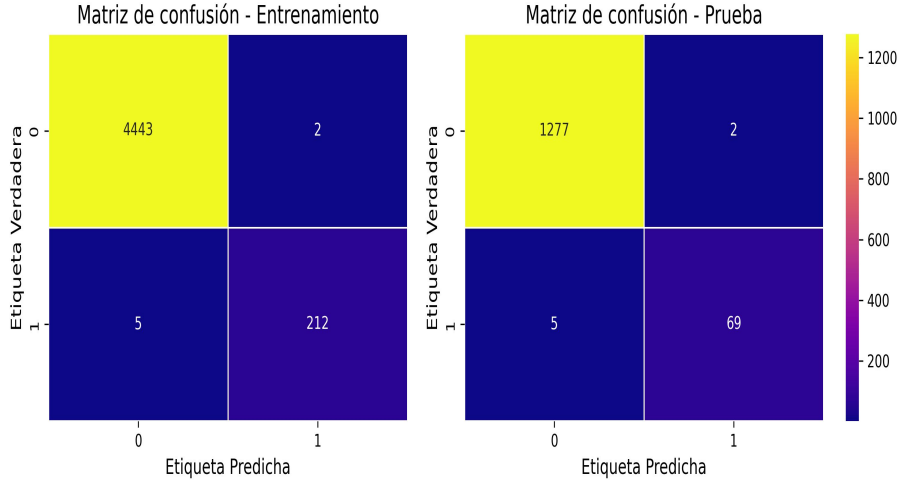


Figura 1: Resultado mejor modelo en entrenamiento y test

Modelo	Parámetros		Precisión	Sensibilidad	F1-score
ExtraTrees	max_depth: 10 min_samples_leaf: 10	min_samples_split: 5 n_estimators: 200	0.985	0.965	0.975

Tabla 2: Resultados mejor modelo.

Bagging (Random Subspaces)

Parámetro	Posibles valores		
n_estimators	50	100	150
max_features	[0.5	0.75	1.0
estimator__max_depth	None	10	20

Tabla 1: Selección de parámetros

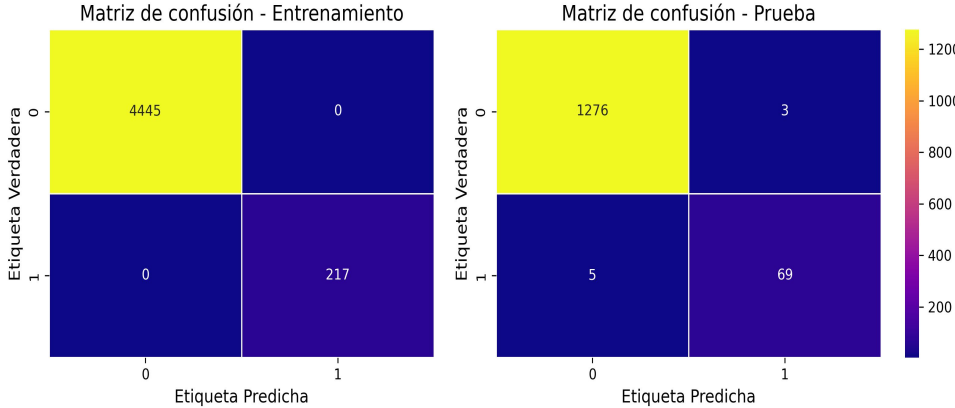


Figura 1: Resultado mejor modelo en entrenamiento y test

Modelo	Parámetros	Precisión	Sensibilidad	F1-score
Random Subspaces	estimator__max_depth: None max_features: 0.75 n_estimators: 50	0.98	0.965	0.975

Tabla 2: Resultados mejor modelo.

Brian Sena Simons 

Clasificador Bayesiano

Asunciones:
- Independencia condicional.
- Distribución normal.
- Clases balanceadas.

Preprocesamiento:
- Eliminación de saltos temporales.
- No hay valores faltantes.
- No es necesario escalar los datos.

Experimentación:
- Parámetro de suavizado.
  - 1e-11 a 1e-9 con 25 valores
- Submuestreo CondensedNearestNeighbour.
  - Los valores de K en [1, 5, 10]
- Sobremuestreo con SMOTE.
- Selección características:
  - MRMR.
  - Chi-cuadrado.

Clasificador Bayesiano

Modelo	Parámetros	F1 Promedio
Bayesiano	$\alpha = 4^{-9}$	0.6802
Bayesiano + MRMR	$\alpha = 1^{-8}$	0.6894
Bayesiano + CHI	$\alpha = 1^{-11}$	0.6847
Bayesiano + CNN	K=10, $\alpha = 4^{-9}$	0.7088
Bayesiano + CNN + MRMR	K=10, $\alpha = 1^{-11}$	0.7320
Bayesiano + CNN + CHI	K=10, $\alpha = 2^{-8}$	**0.7819**
Bayesiano + Smote	$\alpha = 4^{-9}$	0.6811
Bayesiano + Smote + MRMR	$\alpha = 4^{-8}$	0.7047
Bayesiano + Smote + CHI	$\alpha = 5^{-8}$	0.6865

Tabla 1: Búsqueda combinatoria

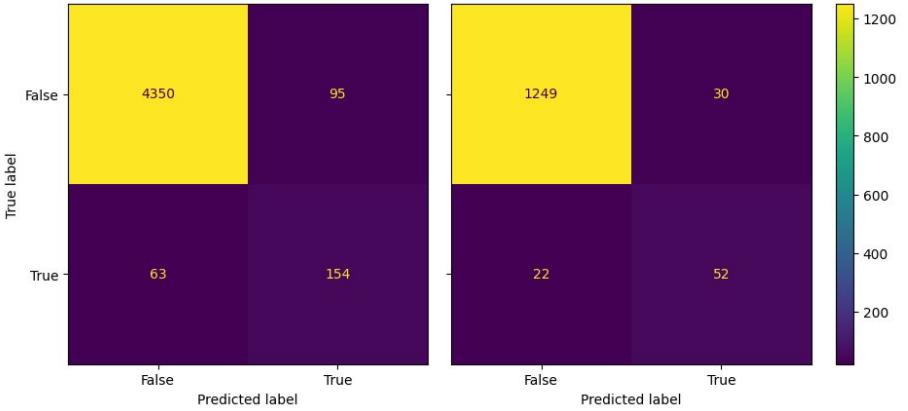


Figura 1: Resultado mejor modelo en entrenamiento y test

Modelo	Parámetros	Precisión	Sensibilidad	F1
Bayesiano + CNN + CHI	$\alpha = 2^{-8}$	0.8084	0.8396	0.8231

Tabla 2: Resultados mejor modelo.

Stacking

Asunciones:
- Diversidad de modelos base.
- Flexibilidad del meta-modelo.
- Distribución equilibrada de clases.
- Independencia de los errores.

Preprocesamiento:
- Eliminación de saltos temporales.
- No hay valores faltantes.
- Adecuado para cada modelo.

Experimentación:
- Modelos elegidos:
  - SVM: Frontera de decisión
  - Árboles: No linealidades fuertes
  - Bayes: Probabilidades
- Parámetros:
  - SVM: Regularización C [0.1, 1, 10]
  - Árboles: Criterio [gini, entropia]
- Técnicas CNN y SMOTE.

Stacking

Modelo	Parámetros	F1 Promedio
Stacking	C=10, $\alpha = 1e^{-9}$, ct="entropía"	0.9376
Stacking + SMOTE	C=10, $\alpha = 1e^{-9}$, ct="gini"	0.9309
Stacking + CNN	C=10, $\alpha = 1e^{-9}$, ct="gini"	0.8726
Stacking + OPT	C=10, $\alpha = 1e^{-9}$, ct="entropía"	**0.9438**

Tabla 1: Búsqueda combinatoria

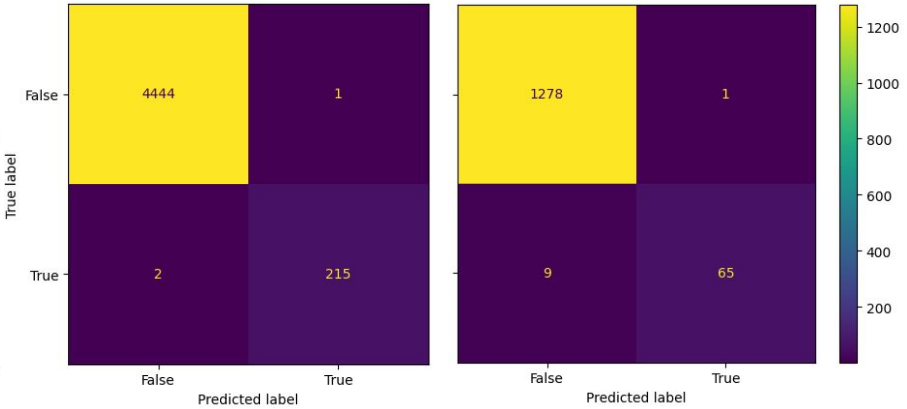


Figura 1: Resultado mejor modelo en entrenamiento y test

Modelo	Parámetros	Precisión	Sensibilidad	F1
Stacking + OPT	C=10, α=$1e^{-9}$, ct="entropía"	0.9889	0.9387	0.9623

Tabla 2: Resultados sobre test del mejor modelo.

Alvaro Santana Sanchez 

Regresión Logística

Asunciones:
- Relación lineal X -> y.
- Variables en X no correlacionadas.
- Independencia de las observaciones.
- Ausencia de valores extremos.

Preprocesamiento:
- Eliminación de ruido.
- PCA.
- Escalado de datos.

Experimentación:
- Regularización.
  - penalización l1 y l2
  - C : np.logspace(-3, 3, 7),
- Pesos en las salidas:
  - 0:1-1:10
  - 0:1-1:20
  - balanceado

Preprocesado

1.Eliminación de Ruido: Ensemble Filter

💡 Eliminación de valores no deseados

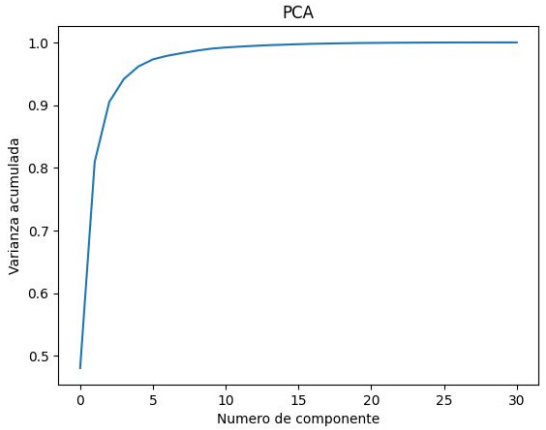
Dataset	Tipo	Count
Original	Anomalía	209
Modificado	Anomalía	217
Original	No Anomalia	4445
Modificado	Anomalía	4443

2: Escalado de datos: StandardScaler (media)

💡 Todas las variables tendrán la misma importancia

3 PCA : de 76 - 30 variables

💡 Evitamos variables muy correlacionadas



Experimentación

Parámetro	Valores
C	10^{-3}, 10^{-2}, 10^{-1}, 10^{0}, 10^{1}, 10^{2}, 10^{3}
penalty	l1, l2
class_weight	balanced, {0: 1, 1: 10}, {0: 1, 1: 20}

Tabla 1: Grid de hiperparametros

C	class_weight	penalty
1000	{0: 1, 1: 10}	l1

Tabla 2: Resultados mejor modelo.

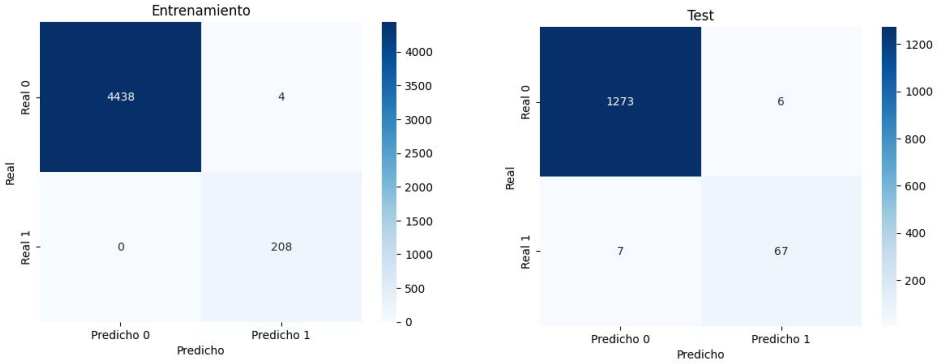


Figura 1: Resultado mejor modelo en entrenamiento y test

AdaBoost

Asunciones:
- Ausencia de ruido excesivo.
- Poco desbalanceo de clases

Preprocesamiento:
- Eliminación de ruido.

Experimentación:
- Optimización: learning rate.
- Pesos en las salidas
- Número estimadores
- Árbol de clasificación:
  - profundidad
  - ejemplos mínimos por nodo y división
  - Criterio
  - Splitter

Preprocesado

1.Eliminación de Ruido: Ensemble Filter

💡 Eliminación de valores no deseados

Dataset	Tipo	Count
Original	Anomalía	209
Modificado	Anomalía	217
Original	No Anomalia	4445
Modificado	Anomalía	4443

Experimentación

Parámetro	Valores
estimator__criterion	gini, entropy
estimator__splitter	best, random
estimator__max_depth	3, 5, 10
estimator__min_samples_split	2, 5, 10
estimator__min_samples_leaf	1, 2, 5
estimator__class_weight	{0: 1, 1: 1}, {0: 1, 1: 10}, {0: 1, 1: 20}, {0: 1, 1: 40}, {0: 1, 1: 100}
n_estimators	1, 2, 10, 50, 100, 200
learning_rate	0.01, 0.1, 0.5, 1

Tabla 1: Grid de hiperparametros

Criterio	max_depth	min_samples_leaf	min_samples_split	splitter	class_weight
gini	3	5	2	best	0:1,1:1

learning_rate	n_estimators
1	200

Tabla 2: Resultados mejor modelo.

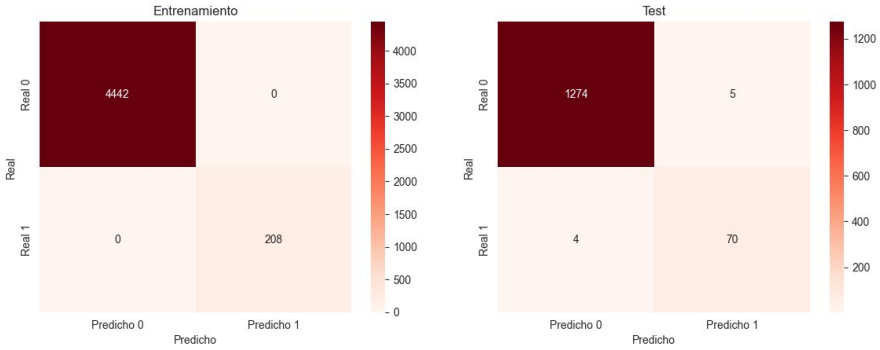
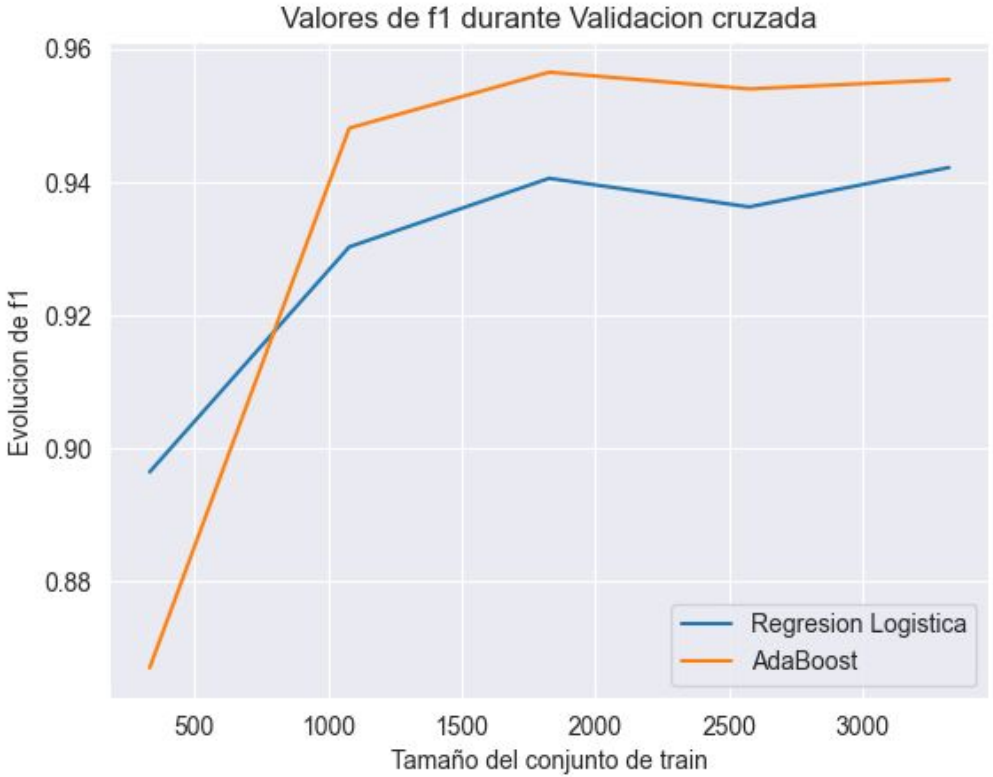


Figura 1: Resultado mejor modelo en entrenamiento y test

Resultados 🥇



F1 test

AdaBoost:0.939
LR:0.917

Precision

AdaBoost:0.933
LR:0.93

Recall

AdaBoost:0.945
LR:0.917

Miguel García López 

Árbol de decisión

Características:

- Pueden trabajar con datos de todo tipo.
- Resilientes a valores extremos (outliers).
- No asume nada sobre los datos.
- Muy variable.

Preprocesamiento:

- No requiere escalado.
- No requiere transformación alguna sobre las variables.

Experimentación:

- Búsqueda de hiperparámetros:
  - max_depth
  - min_samples_split
  - min_samples_leaf
  - criterion

Árbol de decisión

Parámetro	Valor
criterion	entropy
max_depth	15
min_samples_leaf	2
min_samples_split	17

Tabla 1: Mejores parámetros

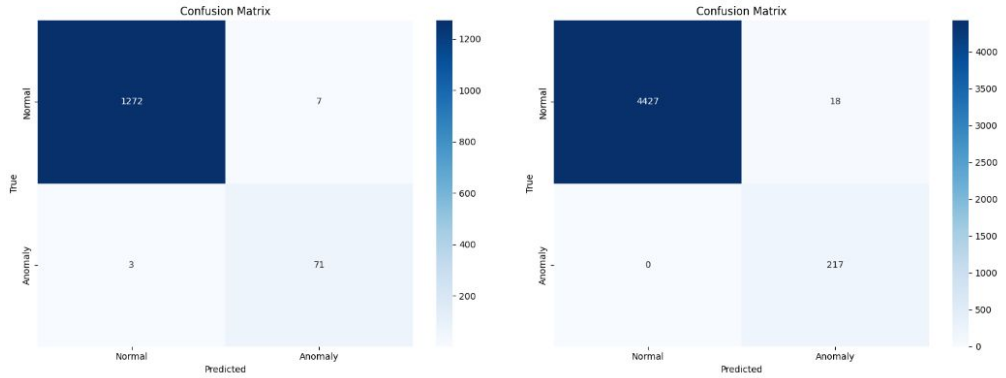


Figura 1: Resultado mejor modelo en entrenamiento y test

	Precision	Recall	F1-Score	Support
False	0.9976	0.9945	0.9961	1279
True	0.9103	0.9595	0.9342	74
Accuracy		0.9926		
Macro Avg	0.9540	0.9770	0.9651	1353
Weighted Avg	0.9929	0.9926	0.9927	1353

Tabla 2: Resultados mejor modelo.

Árbol de decisión

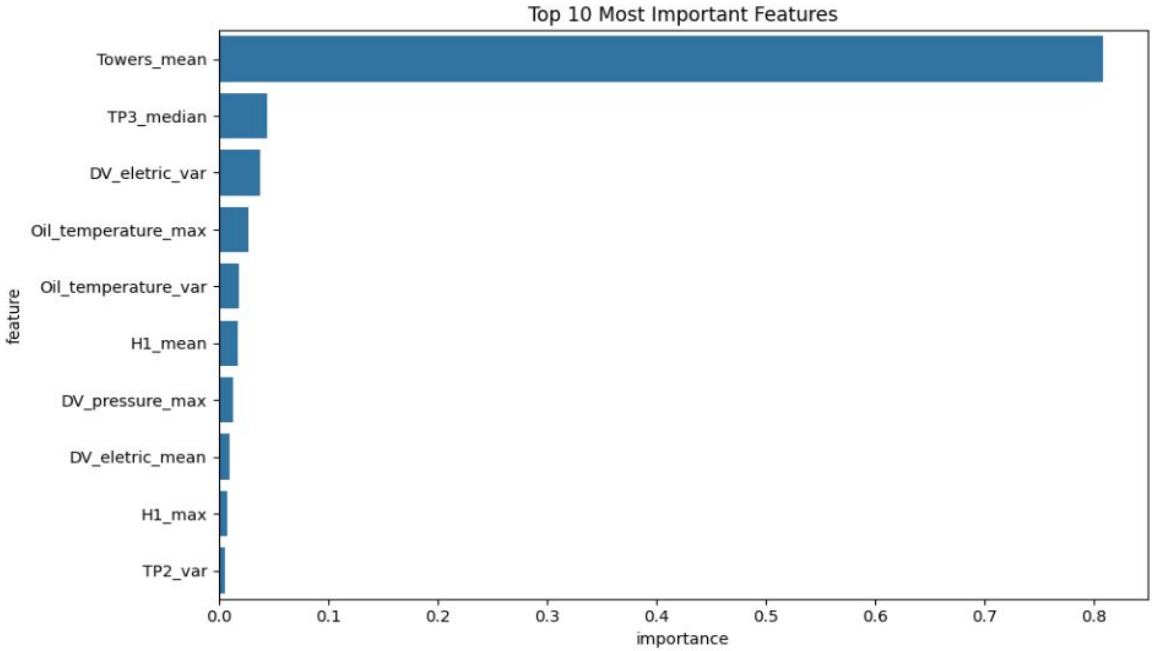


Figura 2: Gráfica de importancia de características.

XGBoost (Gradient Boosting)

Caracteristicas:

- Resilientes a valores extremos (outliers).
- No asume nada sobre los datos.
- Muy robusto y menos variable que un árbol individual.

Preprocesamiento:

- No requiere escalado.
- No requiere transformación alguna sobre las variables.

Experimentación:

- Búsqueda de hiperparámetros:
  - n_estimators
  - colsample_bytree
  - subsample
  - max_depth
  - learning_rate
  - gamma
  - min_child_weight
- Pruebas con búsqueda bayesiana

XGBoost

Parámetro	Valor
colsample_bytree	0.8778
gamma	1.5361
learning_rate	0.2528
max_depth	4
min_child_weight	2
n_estimators	448
subsample	0.5743

Tabla 1: Mejores parámetros

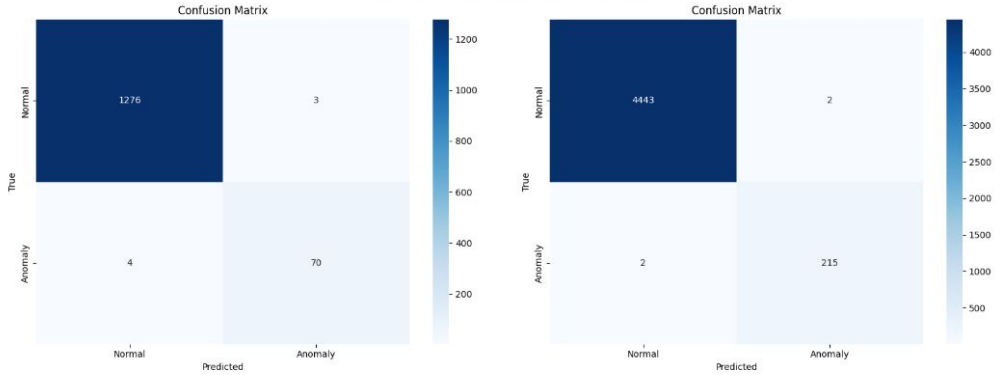


Figura 1: Resultado mejor modelo en entrenamiento y test

	Precision	Recall	F1-Score	Support
False	0.9969	0.9977	0.9973	1279
True	0.9589	0.9459	0.9524	74
Accuracy		0.9948		
Macro Avg	0.9779	0.9718	0.9748	1353
Weighted Avg	0.9948	0.9948	0.9948	1353

Tabla 2: Resultados mejor modelo.

XGBoost

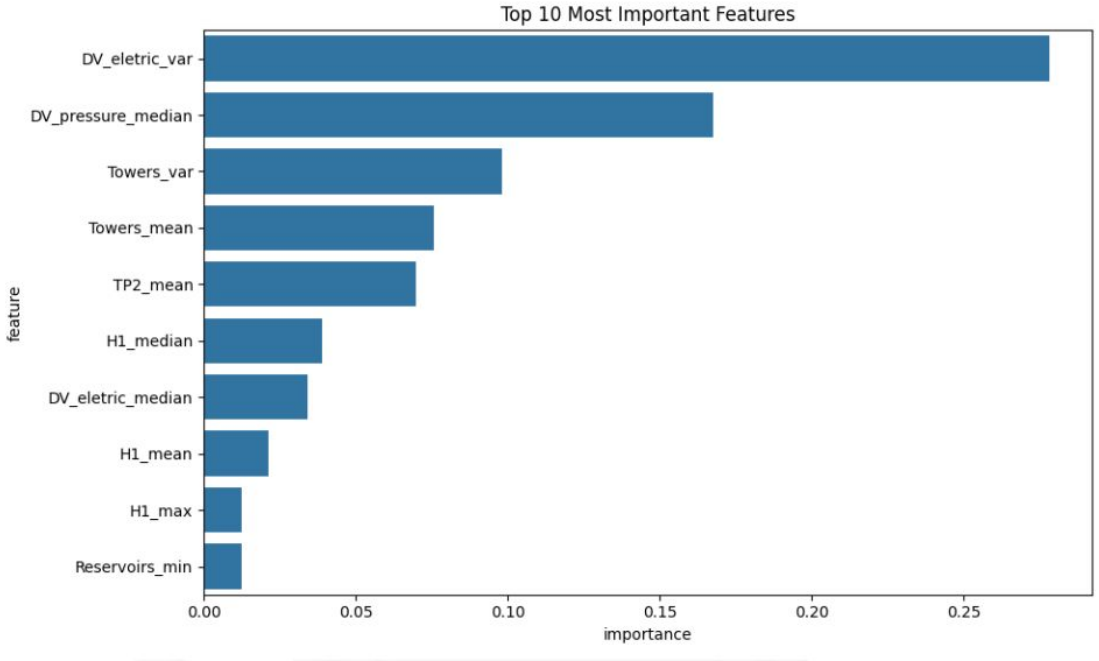
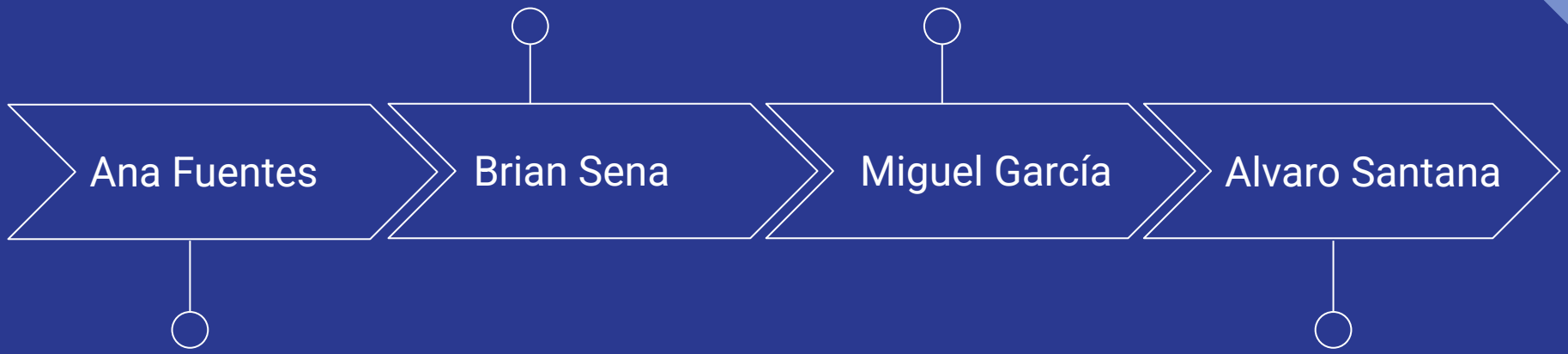


Figura 2: Gráfica de importancia de características.

FIN


Bayesiano + CNN + CHI
Stacking optimizado
Árboles
Gradient Boosting
Ana Fuentes
Brian Sena
Miguel García
Alvaro Santana
SVM
Bagging
Regresión Logística
AdaBoost