UNIVERSIDAD
DE GRANADA

Práctica 3

Competición en Zindi

Mauricio Luque Jiménez

78004003D

maulujim@correo.ugr.es

Subgrupo de prácticas A1

7 de enero de 2026

Inteligencia de Negocio

RANK	USER	PUBLIC SCORE	LAST SUBMISSION	# SUBM
330	 LuqueJimenezMauricio_UGR_IN Go to placement	37.53324436	~2 hours ago	3

	ID	SUBMITTED	SUBMITTER	PUBLIC SCORE	COMMENT	FILE
☑	YVvWrNxW	~2 hours ago	LuqueJimenezMauricio_UGR_IN	37.53324436	De nuevo CatBoost, esta vez con 2000 iteraciones en vez de 1000	
☐	8qyhvmX5	~2 hours ago	LuqueJimenezMauricio_UGR_IN	43.25740334	Uso de LightGBM en vez de CatBoost	
☑	Le5krAtB	~2 hours ago	LuqueJimenezMauricio_UGR_IN	38.31910013	–	

Sumario

Introducción

Esta práctica consiste en participar en una competición de regresión en Zindi orientada a predecir la calidad del aire. En concreto, el objetivo es predecir la concentración diaria de PM2.5 para cada ciudad, una medida habitual de contaminación atmosférica asociada a riesgos relevantes para la salud. Los datos abarcan aproximadamente los últimos tres meses y contienen información meteorológica y variables derivadas de observaciones de satélites para múltiples ciudades. El conjunto de entrenamiento tiene 30.557 instancias con 82 atributos y la variable objetivo se llama target. Un detalle importante del dataset es que, además de *target*, en entrenamiento aparecen columnas relacionadas con el objetivo como *target_min, target_max, target_variance* y *target_count*. Estas columnas no están en test, así que no se pueden usar como predictores en el modelo final. En este caso, para evitar cualquier duda y mantener el pipeline limpio, vamos a eliminar del conjunto de características todas las columnas que no existen en test y dejar únicamente las variables comunes. En cuanto a tipos de datos, hay pocas variables categóricas claras y mucha variable numérica (algo lógico tratándose de un problema de regresión). Destacan:

- *Date*: se puede tratar como fecha para extraer atributos derivados (mes, día, día de la semana).
- *Place_ID*: identificador de ciudad o estación, útil como variable categórica.
- *Place_ID X Date*: identificador compuesto para la entrega, que no debe entrar como predictor directo.

Análisis Exploratorio

La variable objetivo presenta asimetría positiva: hay muchos valores medios y una cola con valores altos. Esto se puede explicar a partir de lo que suele pasar en contaminación: picos puntuales elevan mucho el máximo. De esta manera, nos quedan dos posibles caminos: entrenar directamente para RMSE en la escala original, o aplicar una transformación tipo logarítmica y luego deshacerla al predecir. En este caso, vamos a empezar con la escala original para no trastocar mucho los datos.

Hay columnas con proporciones de valores perdidos muy elevadas (algunas variables satelitales pueden venir muy incompletas según ciudad y día). Aquí hay dos filosofías: eliminar columnas con demasiados NaN o mantenerlas y usar modelos que manejen NaN de forma nativa. Como en esta práctica interesa iterar rápido y LightGBM y CatBoost manejan bien los valores ausentes, opté por mantenerlas en la mayoría de pruebas iniciales. Si se dispusiera de más tiempo, una criba por porcentaje de NaN y un análisis de importancia de variables podría mejorar estabilidad y generalización.

Preprocesado

Para el preprocesado, se han seguido una serie de pasos que se van a explicar a continuación. En primer lugar, para limpiar las columnas, se ha separado *target* como variable objetivo, se ha eliminado de las características cualquier columna que no esté en test, para no depender de información que luego no existe, y se ha dejado *Place_ID* X *Date* fuera del entrenamiento, conservándolo sólo para construir el CSV final. Para tratar la fecha, se ha convertido la variable *Date* a formato fecha, generando variables sencillas como mes, día del mes, día de la semana, etc. Por último, los valores ausentes no se han modificado, ya que los algoritmos que vamos a usar son capaces de manejarlos bien.

Modelos de Regresión Utilizados

En esta práctica me centré en dos familias de modelos que suelen rendir muy bien en tabular con muchos NaN y relaciones no lineales: gradient boosting sobre árboles.

LightGBM es un framework de gradient boosting muy eficiente en tiempo y memoria, especialmente útil cuando hay muchas variables numéricas y patrones no lineales. Suele rendir bien con poco preprocesado y permite ajustar hiperparámetros de forma incremental sin volverse loco Se ha utilizado utilizó como baseline competitivo: rápido de entrenar, razonablemente robusto con NaN y con capacidad de capturar interacciones complejas entre variables meteorológicas y observaciones satelitales.

Por otra parte, CatBoost es también gradient boosting sobre árboles con un punto fuerte muy relevante aquí: el tratamiento de variables categóricas como Place_ID sin tener que aplicar one-hot. Además, suele comportarse muy bien con datasets “sucios” con muchos valores ausentes y con mezclas de escalas. Con CatBoost se han probado dos configuraciones basadas en el número de iteraciones, porque es un ajuste sencillo que permite comprobar si el modelo se beneficia de mayor capacidad sin complicar el resto del pipeline.

Por último, para validar ambos algoritmos, se ha realizado una validación cruzada de cinco particiones.

Resultados y Subidas a Zindi

Exp.	Fecha	Score local (RMSE)	Score Zindi (RMSE)	Preprocesado	Algoritmo	Fichero CSV
1	07/07/2026	23.189219	38.31910013	NaN de forma nativa	CatBoostRegressor	submission_02_catboost_1000.csv
2	07/07/2026	20,579488	43.25740334	Eliminación de columnas no presentes en test (target y derivados), extracción de variables de fecha, Place_ID como categórica, NaN sin imputación explícita	LightGBM Regressor	submission_01_lightgbm.csv
3	07/07/2026	22,208919	37.53324436	Igual que Exp. 1	CatBoostRegressor	submission_03_catboost_2000.csv