Universidad de Buenos Aires

Facultad de Ingeniería

2do Cuatrimestre 2021

[75.06/95.58]

Organización de Datos

Trabajo Práctico - Parte 2

Algoritmos de Machine Learning

Integrantes:	**Mail:**	**Padrón:**
Flouret, Agustín Miguel	aflouret@fi.uba.ar	102298
Milhas, Facundo	fmilhas@fi.uba.ar	102727

Índice

1. Introducción

En este trabajo práctico se utilizaron distintos algoritmos de machine learning para predecir la lluvia de hamburguesas al día siguiente, utilizando diversas técnicas de preprocesamiento y comparando sus métricas.

2. Preprocesamientos utilizados

Nombre del preprocesamiento	Explicación simple	Nombre de la función de Python
Básico	Es el preprocesamiento básico utilizado en el dataset original. Esta función se encarga de que haya coherencia en las unidades de los datos, elimina filas en las que la variable target se encuentra vacía y codifica la variable target de forma binaria.	`basic_preprocessing`
Split	Separa el dataset en train y holdout. Utilizado una sola vez en todo el TP, antes de entrenar los modelos.	`split`
FillNumericalMissings	Completa los valores faltantes en los features numéricos, utilizando la media o la mediana.	`fill_numerical_missings`
KNNStandard	Elimina features categoricos y aplica StandardScaler	`preprocessing_knn_standard`
KNNMinMax	Elimina features categoricos y aplica MinMaxScaler	`preprocessing_knn_min_max`
KNNNormalizer	Elimina features categoricos y aplica Normalizer	`preprocessing_knn_normalizer`
Arboles1	Aplica Dummy Encoding a todas las features categoricas	`preprocessing_arboles_1`
Arboles2	Elimina las features categoricas	`preprocessing_arboles_2`
Arboles3	Aplica Dummy Encoding y elimina columnas con alta cantidad de nulos	`preprocessing_arboles_3`
Arboles4	Elimina features irrelevantes	`preprocessing_arboles_4`
Redes1	Aplica One Hot Encoding en variables categoricas y Standard Scaler en variables numericas	`preprocessing_redes_1`
Redes2	Elimina features irrelevantes y aplica Standard Scaler en variables numericas	`preprocessing_redes_2`

3. Modelos

Nombre Modelo	Nombre del preprocesamiento	AUC-ROC	Accuracy	Precision	Recall	F1 score
Arboles de Decisión	Arboles1	0.86	0.84	0.70	0.48	0.57
Naive Bayes	Arboles2	0.84	0.83	0.63	0.55	0.58
Random Forest	Arboles1	0.88	0.86	0.75	0.51	0.61
Boosting	Arboles1	0.88	0.85	0.74	0.53	0.62
KNN	KNNStandard	0.88	0.85	0.76	0.45	0.57
Redes Neuronales	Redes1	0.90	0.87	0.75	0.61	0.67

3.1. Métricas de holdout

Nombre Modelo	Nombre del preprocesamiento	AUC-ROC	Accuracy	Precision	Recall	F1 score
Redes Neuronales	Redes1	0.90	0.86	0.73	0.59	0.65

4. Conclusión

Analizando los resultados de los modelos, podemos concluir que el mejor modelo obtenido en este trabajo es la red neuronal, ya que obtuvo el mayor score AUC-ROC, con un valor de 0.90. Por otro lado, el peor modelo es Naive Bayes, ya que obtuvo el menor puntaje en casi todas las métricas.

Sin embargo, si se quisiera tener la menor cantidad de falsos positivos, KNN es el más indicado, ya que posee el mayor score de Precision. Por otro lado, si se necesitase obtener la mayor cantidad de dias que llueven hamburguesas sin preocuparse por los falsos positivos, la métrica que hay que maximizar es el Recall. En ese caso, la red neuronal es superior a todos los otros modelos.