Conclusiones:

Durante el trabajo final hemos probado e implementado una gran variedad de estrategias y modelos para cada algoritmo en torno a la predicción de los valores de lluvia y a la clasificación relacionada a si llueve o no.

De todos los modelos probados nos quedamos con uno por método:

- Regresión lineal múltiple con regularización L1:

```
r2_ajustado_entrenamiento    0.22
r2_ajustado_prueba           0.23
mse_train                    0.78
mse_test                     0.51
mae_train                    0.42
mae_test                     0.39
Name: Lasso-alpha=0.00108, dtype: float64
```

- Regresión logística con pesos en función de costo (sin balanceo de clases, sin optimización de hp):

```
Metricas test:
              precision    recall  f1-score   support

           0       0.91      0.78      0.84      2435
           1       0.52      0.74      0.61       763

    accuracy                           0.77      3198
   macro avg       0.71      0.76      0.73      3198
weighted avg       0.81      0.77      0.79      3198
```

- Red neuronal de regresión con optimización de hiperparametros:

```
----------------------------
MSE TEST: 0.42
MSE TRAIN: 0.56
----------------------------
RMSE TEST: 0.65
RMSE TRAIN: 0.75
----------------------------
MAE TEST: 0.29
MAE TRAIN: 0.31
----------------------------
R2 AJUSTADO TEST: 0.36
R2 AJUSTADO TRAIN: 0.44
```

- Red neuronal de clasificación sin optimización de hiperparametros:

```
Reporte de Clasificación:
              precision    recall  f1-score   support

           0       0.91      0.80      0.85      2435
           1       0.54      0.74      0.62       763

    accuracy                           0.79      3198
   macro avg       0.72      0.77      0.74      3198
weighted avg       0.82      0.79      0.80      3198
```

Cada uno de estos fue elegido a partir de la comparación entre todos los modelos del mismo tipo (tanto sin optimización de hiperparametros como con).

A mitad de la materia tomamos ciertos modelos base tanto para regresión como para clasificación que sean simples pero que tengan alguna lógica razonable como para poder comparar con el mejor de nuestros modelos de cada algoritmo dado.

- Para clasificar si llueve o no llueve usamos una regresión logística, basada solamente en una variable llamada ‘Humidity3pm’ que es la más relevante, obteniendo las siguientes métricas:

```
Reporte de Clasificacion:
              precision    recall  f1-score   support

           0       0.88      0.69      0.78      2435
           1       0.41      0.69      0.52       763

    accuracy                           0.69      3198
   macro avg       0.65      0.69      0.65      3198
weighted avg       0.77      0.69      0.71      3198

Area bajo la curva ROC:
0.6919199313204928
```

- Para regresión entre varios intentos nos quedamos con un modelo base simple que verificaba en la variable con más correlación lineal (‘Humidity3pm’) con la variable dependiente si era mayor a determinado umbral asignando un valor de manera binaria en torno al resultado. Esto fue elegido observando gráficos y se obtuvieron las siguientes métricas:

```
MSE: 0.57
MAE: 0.33
R2: 0.15
```

Finalmente comparamos nuestros mejores modelos con su modelo base:

	base	regresion_lineal_lasso	redes_neuronales_regresion_opt
MSE	0.57	0.51	0.42
MAE	0.33	0.39	0.29
R2	0.15	0.23	0.36

	base	regresion_logistica_pesos	redes_neuronales_clasificacion_sin_opt
f1_score_0	0.78	0.84	0.85
f1_score_1	0.52	0.61	0.62
f1_score_macro	0.65	0.73	0.74

Podemos ver que los distintos métodos de machine learning le ganaron al modelo base

De esta comparación resulta que el modelo de redes neuronales tiene mejores métricas que el de regresión, sobre todo en r2 ajustado. Mientras que en el problema de clasificación la diferencia no es tan notoria.

La conclusión del anterior párrafo puede deberse a que el tiempo de experimentación con los algoritmos de redes neuronales de regresión fue mucho mayor al de los algoritmos de clasificación. Se llegó a las últimas consignas del trabajo practico con los últimos minutos de tiempo para entregarlo a plazo y la experimentación tanto con optimización de hiperparametros como de arquitecturas (sin optimizar) en redes neuronales es algo que toma demasiado tiempo. Tal vez con una mayor disponibilidad de tiempo para experimentar por parte de los integrantes del grupo o con un mayor plazo de entrega del trabajo hubiéramos obtenido mejores resultados aún.