


## Actividad 2 Módulo 6

### Código a revisar


```
[30] from sklearn.model_selection import train_test_split, KFold, cross_val_score, LeaveOneOut
      from sklearn.linear_model import LinearRegression
      from sklearn.datasets import load_diabetes

      X, y = load_diabetes(return_X_y=True)
      X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.4, random_state=0)
      modelo = LinearRegression()
```

```
[31] X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
      modelo.fit(X_train, y_train)
      score_holdout = modelo.score(X_test, y_test)
      print(f"R2 Holdout: {score_holdout:.2f}")
```

 R2 Holdout: 0.45

```
[32] kf=KFold(n_splits=5, shuffle=True, random_state=1)
      scores_kfold = cross_val_score(modelo, X, y, cv=kf, scoring="r2")
      print(f"Resultados K-Fold: {scores_kfold}")
      print(f"Promedio R2 (K-Fold): {scores_kfold.mean():.2f}")
```

 Resultados K-Fold: [0.43843162 0.38982359 0.52792629 0.47359827 0.5744937 ]  
Promedio R2 (K-Fold): 0.48

```
[33] from sklearn.datasets import make_regression
      from sklearn.linear_model import LinearRegression
      from sklearn.model_selection import LeaveOneOut, cross_val_score


      # Generar un data set pequeño con una sola característica
      X, y = make_regression(n_samples=20, n_features=1, noise=10, random_state=0)

      # Crear modelo
      modelo= LinearRegression()

      # Crear esquema L0OCV
      loocv = LeaveOneOut()

      # Evaluar con MSE (negativo)
      scores = cross_val_score(modelo, X, y, cv=loocv, scoring="neg_mean_squared_error")

      # Mostrar promedio del error ( recordando que sklearn devuelve negativo)
      print(f"Promedio del error (L0OCV - MSE): {-scores.mean():.2f}")
```

 Promedio del error (L0OCV - MSE): 85.87

## Respuestas a la actividad

### 1. ¿Cuál técnica entregó el valor de $R^2$ más alto? ¿Y cuál el más bajo?

- **Más alta:** K-Fold con un  $R^2 = 0.481$
- **Más Baja:** Holdout con un  $R^2 = 0.453$

### 2. ¿Cuál técnica crees que refleja mejor la capacidad real de generalización del modelo?

K-Fold refleja mejor la capacidad de generalización. La razón es que utiliza todos los datos tanto para entrenar como para probar el modelo a lo largo de sus múltiples iteraciones.

Esto reduce el sesgo de una única división de datos y ofrece una estimación más estable y confiable de cómo se comportará el modelo con datos nuevos.

### 3. ¿Por qué crees que LOOCV es más costosa en tiempo de cómputo?

LOOCV es computacionalmente costosa porque entrena un modelo diferente para cada muestra en el dataset. Si el dataset tiene  $N$  muestras, LOOCV entrena el modelo  $N$  veces. En cambio, K-Fold solo lo entrena  $k$  veces en este caso, 5 veces y Holdout lo entrena una sola vez.

Para este caso el dataset diabetes con 442 muestras, LOOCV realiza 442 entrenamientos, mientras que K-Fold solo 5.

### 4. ¿En qué caso elegirías Holdout sobre K-Fold, o viceversa?

Holdout cuando tienes un dataset muy grande, al tener suficientes datos, una sola división aleatoria Holdout puede ser suficientemente buena y representativa.

K-Fold podría ser demasiado lento y computacionalmente caro. Por tanto, para datasets pequeños o medianos es utilizable. K-Fold es el estándar de oro para la mayoría de los proyectos.

En estos casos, es crucial aprovechar al máximo los datos disponibles para obtener una evaluación robusta del modelo.

### 5. ¿Qué aprendiste sobre la importancia de evaluar modelos con distintos enfoques?

La lección principal es que no existe una única "verdad" sobre el rendimiento de un modelo. La forma en que evaluamos un modelo influye directamente en los resultados que se obtienen.

Usar un solo método (como Holdout) puede darnos una visión incompleta o engañosa.

Técnicas como K-Fold y LOOCV ofrecen una evaluación más robusta a costa de un mayor costo computacional.

Comprender las ventajas y desventajas de cada técnica de validación es fundamental para elegir la correcta según el tamaño del dataset y los recursos disponibles, asegurando que podamos confiar en que nuestro modelo generaliza bien a datos futuros.