A1.4 Selección de Características

```
import pandas as pd
import numpy as np
from sklearn.linear_model import LinearRegression
from mlxtend.feature_selection import SequentialFeatureSelector as SFS
from sklearn.model_selection import train_test_split
```

1. Impresión de las primeras cinco filas de la base de datos "A1.4 Vino Tinto", así como la demostración de sus dimensiones.

```
data = pd.read_csv("A1.4 Vino Tinto.csv")
print("Primeras cinco filas: \n",data.head(5))
print("\nDimensiones de la base de datos:")
print("Número de filas: ", data.shape[0])
print("Numero de columnas: ", data.shape[1])
```

```
Primeras cinco filas:
    acidezFija  acidezVolatil  acidoCitrico  azucarResidual  cloruros  \
0         7.4           0.70          0.00             1.9     0.076
1         7.8           0.88          0.00             2.6     0.098
2         7.8           0.76          0.04             2.3     0.092
3        11.2           0.28          0.56             1.9     0.075
4         7.4           0.70          0.00             1.9     0.076

   dioxidoAzufreLibre  dioxidoAzufreTotal  densidad    pH  sulfatos  alcohol  \
0                11.0                34.0    0.9978  3.51      0.56      9.4
1                25.0                67.0    0.9968  3.20      0.68      9.8
2                15.0                54.0    0.9970  3.26      0.65      9.8
3                17.0                60.0    0.9980  3.16      0.58      9.8
4                11.0                34.0    0.9978  3.51      0.56      9.4

   calidad
0        5
1        5
2        5
3        6
4        5

Dimensiones de la base de datos:
Número de filas:  1599
Numero de columnas:  12
```

2. Separación de datos de entrenamiento y datos de prueba usando la función proveída por la librería sklearn.modelselection. Los datos de entranamiento son el 80% de los originales y los de prueba son el 20%, son seleccionados de manera aleatoria. Después, se imprimen las dimensiones en pantalla.

```
train, test = train_test_split(data, test_size=0.2)
print("Dimensiones de los datos de entrenamiento (filas, columnas):", train.shape)
print("Dimensiones de los datos de prueba (filas, columnas): ", test.shape)
```

```
Dimensiones de los datos de entrenamiento (filas, columnas): (1279, 12)
Dimensiones de los datos de prueba (filas, columnas):  (320, 12)
```

3. Las caraterísticas seleccionadas son las siguientes se hace mediante la técnica de selección hacia adelante. Se utiliza la librería mxltend para usar la clase SFS, la cual permitirá automáticamente conocer el mejor modelo iterando 10 veces usando las condiciones iniciales.

```
model = LinearRegression()

x_test = test[["acidezFija","acidezVolatil","acidoCitrico","azucarResidual","cloruros","dioxidoAzufreLibre","dioxidoAzufreTotal","densidad",
y_test = test["calidad"]

x_train = train[["acidezFija","acidezVolatil","acidoCitrico","azucarResidual","cloruros","dioxidoAzufreLibre","dioxidoAzufreTotal","densidad
y_train = train["calidad"]

sfs = SFS(
    model,
    k_features=(2,8), # Rango de caracteristicas a aplicar.
    forward=True, # Seleccion hacia adelante.
    scoring="r2", # Metrica de evaluacion.
    cv=10 # Validacion cruzada con 10 iteraciones.
```

```
)

sfs.fit(x_train,y_train)
selected_features = list(sfs.k_feature_names_)
print("Las caracteristicas seleccionadas son: ", selected_features)
```

```
Las caracteristicas seleccionadas son:  ['acidezVolatil', 'cloruros', 'dioxidoAzufreLibre', 'dioxidoAzufreTotal', 'pH', 'sulfatos', 'alc
```

4. Se obtiene R^2 para demostrar la capacidad de predicción del model de selección hacia adelenta.

```
from sklearn.metrics import r2_score

x_train_selected = x_train[selected_features]
x_test_selected = x_test[selected_features]

model.fit(x_train_selected, y_train)

y_pred = model.predict(x_test_selected)

r2 = r2_score(y_test, y_pred)
print("R^2 del modelo con las variables seleccionadas:", r2)
```

```
R^2 del modelo con las variables seleccionadas: 0.39098327331397675
```

5. Se utiliza ahora el modelo de selección hacia atrás como punto de partida de comparación más adelante.

```
model_back = LinearRegression()

sfs_back = SFS(
    model_back,
    k_features=(2,5), # Rango de caracteristicas a aplicar.
    forward=False, # Seleccion hacia atrás.
    scoring="r2", # Metrica de evaluacion.
    cv=10 # Validacion cruzada con 10 iteraciones.
)

sfs_back.fit(x_train,y_train)
selected_features_back = list(sfs_back.k_feature_names_)
print("Las caracteristicas seleccionadas del model hacia atrás son: ", selected_features_back)
```

```
Las caracteristicas seleccionadas del model hacia atrás son:  ['acidezVolatil', 'cloruros', 'dioxidoAzufreTotal', 'sulfatos', 'alcohol']
```

6. Demostración del valor R^2 para el modelo de seleccion hacia atras.

```
x_train_selected_back = x_train[selected_features_back]
x_test_selected_back = x_test[selected_features_back]

model_back.fit(x_train_selected_back, y_train)
y_pred_back = model_back.predict(x_test_selected_back)

r2_back = r2_score(y_test, y_pred_back)
print("R^2 del modelo con las variables seleccionadas:", r2)

if r2_back > r2:
    print("\nEl modelo con selección hacia atrás tiene un mayor R^2, por lo que se ajusta mejor a los datos.")
else:
    print("\nEl modelo con selección hacia adelante tiene un mayor R^2, lo que sugiere que incluir más variables fue beneficioso.")
```

```
R^2 del modelo con las variables seleccionadas: 0.3799267805020232

El modelo con selección hacia adelante tiene un mayor R^2, lo que sugiere que incluir más variables fue beneficioso.
```

El modelo de selección hacia adelante tuvo un mayor valor en su índice, esto sugiere que usar más variables para la predicción de valores captura mejor la variabilidad, por lo que es importante la selección k_features ya que puede influir en R^2.

Código de honor: Doy mi palabra de que he realizado esta actividad con integridad académica.