Universidad Autónoma de Baja California

Facultad de Ciencias Químicas e Ingenierías

Asignatura: Inteligencia Artificial
Alumno: Ramsses Palafox Ballardo / 01224684
Reporte de Practica: Introducción a "Machine Learning"

Introducción a "Machine Learning" : Los sistemas de machine Learning fundamentalmente se utilizan para los siguientes procesos;

Recolección de datos, Procesamiento de datos, Identificación del modelo, Entrenamiento, Validación, Identificación de hiperparámetros y Predicción

En todos los casos el aprendizaje se hace de forma iterativa por lo que se saca provecho de los métodos de regresión lineal para optimizar el modelo.

Ejercicio 1-4: Obtención del mejor hiperparámetro ***Alpha.***
Utilizando la biblioteca sklearn se puede hacer uso de métodos de entrenamiento para modelos en este caso se utilizó el modelo LASSO, con ayuda de "Datasets" divididos por el método de KFOLDS obtenemos n conjuntos de datos según se indica en el parámetro "n_splits", con estos conjuntos de datos se entrena y evalúa el modelo obteniendo un coeficiente de correlación que se presenta como un valor " 0 < X < 1 " que se puede interpretar como la exactitud del resultado obtenido por el modelo hipotético sobre los datos reales.

```
k_fold2 = KFold(n_splits=4)
AALPHA = 0
KALPHA = 0
IALPHA = 0
for i in frange(0.02,1,.02):
    AALPHA = 0
    print("alpha = {}".format(i))
    linear_model3 = Lasso(alpha = i)
    for k2, (train, test) in enumerate(k_fold2.split(X, y)):
        linear_model3.fit(X[train], y[train])
        # Se valida con el conjunto de datos de prueba
        print("[fold {}]  coeficiente de correlación: {}".format(k2,
linear_model3.score(X[test], y[test])))
        AALPHA = AALPHA + (linear_model3.score(X[test],y[test]))
    print(AALPHA/4)
    if(KALPHA == 0):
        KALPHA = (AALPHA/4)
    else:
        if(KALPHA < (AALPHA/4)):
            print("Mejor coeficiente previo: {} ".format(KALPHA))
            print("Coeficiente actual: {} ".format(AALPHA/4))
            IALPHA = i
            KALPHA = (AALPHA/4)
print("Mejor Alpha en promedio {} con un coeficiente de correlación de :
{}".format(IALPHA,KALPHA))
```

con la secuencia anterior se determina por medio de evaluaciones y comparaciones el Alpha aplicado a los conjuntos obtenidos por KFOLDS que obtuvo un mayor coeficiente de correlación.

En el caso de el "dataset" Diabetes, el sistema concluyo que un Alp ha de 0.14 fue el más optimo obteniendo un coeficiente de correlac ión de 0.484911167052655565.
Por el contrario con el "dataset" Boston se encontró que el mejor Alpha fue 9.78 con un coeficiente de correlación de 0.283889046 2637975; podemos asumir que los modelos no están correctamen te entrenados ya que ambos coeficientes de correlación fueron de cepcionantes, sin embargo los coeficientes promedio se vieron afe ctados por valores negativos que el sistema arrojaba al usar un n umero de folds superior a 3, ambos modelos fueron entrenados c on n_splits = 4, al rehacer los entrenamientos con n_splits = 3 se obtuvieron alphas y coeficientes muy diferentes en ambos casos.

Ejercicio 2: Programar sse, mse y rmse.
Para este ejercicio se solo se realizo una función "SSEf" que calcu lara de forma iterativa por medio de llamadas cicladas, el valor de las diferencias al cuadrado, haciendo el MSE y RMSE por definición siendo MSE la normalización de SSE y RMSE la raíz de MSE.

```
def SSEf(Yr,Yi):
    return (Yr-Yi)**2
def SSE(Yr,Yi):
    Sum = 0
    for i in range(0,len(Yr)):
        Sum = Sum + SSEf(Yr[i],Yi[i])
def MSE(Yr,Yi):
    n = len(Yr)
    Y = SSE(Yr,Yi)
    return (Y/n)
def RMSE(Yr,Yi):
    return MSE(Yr,Yi)**.5
```

Para los ejercicios anteriores se necesitó utilizar sets de datos, para realizar el entrenamiento y prueba de los modelos propuestos, en especifico se utilizaron el dataset Diabetes que cuenta con 442,10 elementos en su parte "data" y 442 en su parte "target" que se entiende como la salida del fenómeno medido en este caso cada elemento cuenta con 10 componentes los cuales son: Edad, Genero, índice de masa corporal, presión sanguínea y seis muestras de sangre; junto a estos datos se almacena el "target" el cual es una valoración del progreso del paciente después de un año.

En el caso del database "Boston" este representa detalles de precios de bienes raíces utilizando trece atributos para valorar el inmueble, siendo esto ultimo el valor de "target", entre estos atributos se encuentran el índice de crimen per capita del pueblo o asentamiento donde se encuentra el inmueble, la proporción en relación a otros inmuebles en la zona residencial, la proporción de viviendas en la zona, concentración de oxido nitrico en el ambiente PPM, numero promedio de habitaciones por inmueble en la zona, la edad promedio de los ocupantes anteriores, la ubicación en relación a la zona comercial, la cantidad de autopistas que cruzan la zona, los impuestos promedio por cada 10k, índice de escuelas y maestros en la zona, índice de segregación indicando la cantidad de personas de color en la zona, el porcentaje de la población de bajos recursos y la media del valor de los inmuebles ocupados en escalas de 1k.