Aprendizaje Automático Taller 2

Andres Racines Bolaños.
Mestria En Ingenieria
Pontifiacia Universidad Javeriana
Cali-Colombia
Email:racines@javerianacali.edu.co

***Resumen*- Este taller se realizara utilizando el mismo conjunto de datos que le fue asignado para el taller 1 y una de las siguientes técnicas de aprendizaje a su elección: Dataset asignado : "allrep" Técnica de aprendizaje : "Maquinas de vectores de soporte"**

.

I. Preparación de Datos

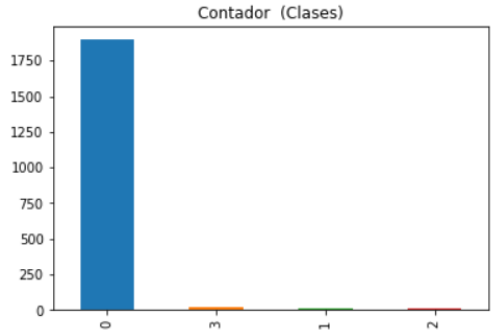
Se realiza el cargue de los datos de train y test del repositorio https://archive.ics.uci.edu/ml/machine-learning-databases/thyroid-disease/allrep.data. Se elimina la columna 27 que no contiene valores. Se asigna el nombre a la columna 28 que contiene la clase con la etiqueta "C", dicha columna contiene la clase acompa de un numero :

	0	1	2	3	4	5	6	7	8	9	...	19	20	21	22	23	24	25	26	28	29
0	35	F	f	f	f	f	f	f	f	f	...	NaN	f	NaN	f	NaN	f	NaN	f	other	negative.|219
1	63	M	f	f	f	f	f	f	f	f	...	2.5	t	108.0	t	0.96	t	113.0	f	SVI	negative.|2059
2	25	F	f	f	f	f	f	f	f	f	...	2.4	t	61.0	t	0.82	t	75.0	f	SVHD	negative.|399

Se procede a eliminar el valor numérico de "C" columna para dejar solo las categorías válidas para la clase y se convierte todos los valores de las columnas categóricos a enteros.

	0	1	2	3	4	5	6	7	8	9	...	19	20	21	22	23	24	25	26	28	C
1	50	1	0	0	0	0	0	0	0	0	...	24	0	78	0	37	0	77	0	3	0
2	12	0	0	0	0	0	0	0	0	0	...	23	0	32	0	23	0	39	0	2	0
3	40	0	0	0	0	0	0	0	0	1	...	20	0	115	0	44	0	105	0	4	0
4	77	0	0	0	0	0	0	0	0	0	...	11	0	90	0	25	0	107	0	3	0

Se realiza un histograma para ver la distribución de las clases y la proporción entre estas Sobre entrenamiento




```
Clase 0: 1895
Clase 1: 17
Clase 2: 12
Clase 3: 23
Proporcion: 111.47 : clase[0] / clase[1]
Proporcion: 157.92 : clase[0] / clase[2]
Proporcion: 82.39 : clase[0] / clase[3]
```

Como se puede observar la clase[0] contiene un mayor número de observaciones que las demás clases.

Se procede a extraer aleatoriamente un 10% de los datos del conjunto de entrenamiento. Este subconjunto constituirá el conjunto de validación, que se usará para estimar los valores de los parámetros de la técnica SVM. Todas las técnicas son entrenadas con el mismo conjunto de datos para que sea posible comparar sus desempeños. En la siguiente celda, se divide el conjunto de entrenamiento como se enunció.

```
Original dataset shape trainval Counter({0: 1707, 3: 23, 1: 13, 2: 9})
Original dataset shape testval Counter({0: 188, 1: 4, 2: 3})
```

II. Primera Iteración "Seleccionar el mejor costo para la SVM"

A continuación se evalúan diferentes funciones de costo para la SVM Lineal para establecer cuál de las opciones obtiene el mejor desempeño, el procedimiento se realiza con el conjunto de validación extraído en el punto anterior.

```
############## Accuracy Linear SVM C=1.0 ##########
0.9794871794871794
[[189   0   0   1]
 [  1   1   1   0]
 [  1   0   0   0]
 [  0   0   0   1]]

############## Accuracy Linear SVM C=5.0 ##########
0.9846153846153847
[[189   0   0   1]
 [  0   2   1   0]
 [  1   0   0   0]
 [  0   0   0   1]]

############## Accuracy Linear SVM C=10.0 ##########
0.9794871794871794
[[188   0   1   1]
 [  0   2   1   0]
 [  1   0   0   0]
 [  0   0   0   1]]

############## Accuracy Linear SVM C=50.0 ##########
0.9794871794871794
[[188   0   1   1]
 [  0   2   1   0]
 [  1   0   0   0]
 [  0   0   0   1]]

############## Accuracy Linear SVM C=500.0 ##########
0.9794871794871794
[[188   0   1   1]
 [  0   2   1   0]
 [  1   0   0   0]
 [  0   0   0   1]]
```

Como es posible observar las diferentes configuraciones del valor "C" no genera un mejor rendimiento entre las diferentes modelos de SVM .

Por tal motivo no se escoje espeficara el valor de “C” en los modelos en adelante.

III. Segunda Iteracion “Seleccionar el mejor Kernel para SVM”

A continuación se explorara dos configuraciones de kernel adicionales para con esto intentar mejorar el rendimiento de las SVM.

```
############## Accuracy SVM RBF ##################
0.9743589743589743
[[190   0   0   0]
 [  3   0   0   0]
 [  1   0   0   0]
 [  1   0   0   0]]

############## Accuracy SVM Sigmoid ##################
0.9743589743589743
[[190   0   0   0]
 [  3   0   0   0]
 [  1   0   0   0]
 [  1   0   0   0]]
```

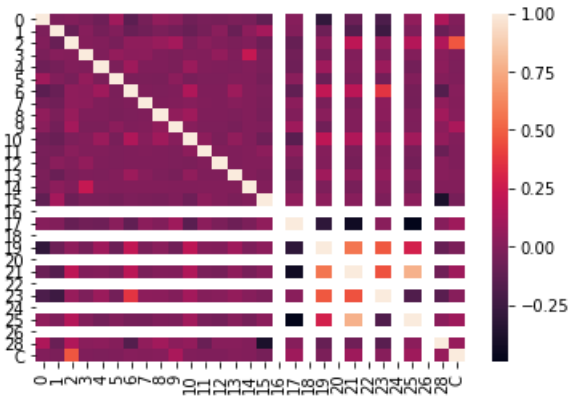
Como se observa en la grafica anterior la medion de Acuracy de los kernel RBF y Sigmoid no mejoran la predicion obtenidad con el kernel Lineal. Por lo tanto se utilizara el kernel lineal para continuar con los experimentos.

Ahora se procede a realizar la predicion con el conjunto de datos de de test. Como se puede notar en la grafica a persar de que el clasificador muestra un acurracy 96% se puede observar que al mirar otras metricas como la precision y el recall , la clase 2 aparece con una precision de 0% lo cual esta demostrando que el modelo es no esta clasificando correctamente.

```
############## Accuracy Linear SVM FINAL ##################
0.9609053497942387
[[930   2   0   3]
 [ 10   1   0   0]
 [  8   1   0   0]
 [ 14   0   0   3]]
              precision    recall  f1-score   support

           0       0.97      0.99      0.98       935
           1       0.25      0.09      0.13        11
           2       0.00      0.00      0.00         9
           3       0.50      0.18      0.26        17

    micro avg       0.96      0.96      0.96       972
    macro avg       0.43      0.32      0.34       972
 weighted avg       0.94      0.96      0.95       972
```

IV. Tercera Iteración “ eliminación de atributos”

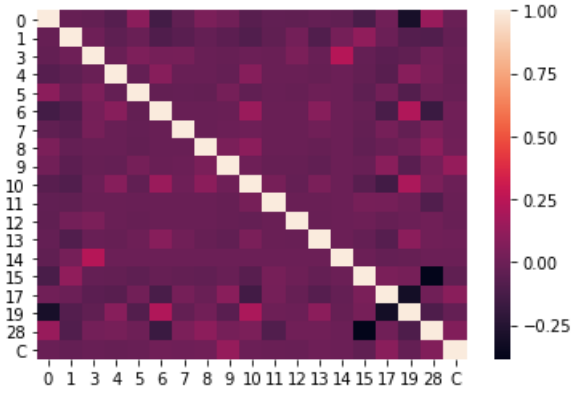
Apesar que se obtuvo una buena predicion del clasificador "SVM LINEAL", se intera mejorar eliminando aquellos atributos que contengan alta correlacion para si se puede mejorar la predicion, a continuacion se grafica la matriz de correlacion para ver cuales atributos estan altamente correlacionados



Se eliminan los datos que tienen alta correlación tanto del conjunto de train como del conjunto de test

```
del ds[21]
del ds[23]
del ds[25]
del test[21]
del test[23]
del test[25]
```

Se procede a graficar nuevamente la matriz de correlacion para verificar que los datos han sido eliminados correctamente.



Nuevamente se corre el modelo con el nuevo conjunto de datos, con las columnas que tenian alta correlacion.

```
############## Accuracy Linear SVM FINAL ##################
0.9609053497942387
[[930   2   0   3]
 [ 10   1   0   0]
 [  8   1   0   0]
 [ 14   0   0   3]]
              precision    recall  f1-score   support

           0       0.97      0.99      0.98       935
           1       0.25      0.09      0.13        11
           2       0.00      0.00      0.00         9
           3       0.50      0.18      0.26        17

    micro avg       0.96      0.96      0.96       972
    macro avg       0.43      0.32      0.34       972
 weighted avg       0.94      0.96      0.95       972
```

Como es posible observar la predicion no mejora con la implementacion de la eliminacion de de caracteristicas con alta correalacion y al igual que la anterior iteracion la clase 2 siguie con 0.0% en la precision.

V. Cuarta Iteración "Balanceo de datos usando Oversampling"

Como se pudo observar en la etapa de preparacion de los datos, donde se evidencio que una de las clases "0" predominaba sobre la otras como se muestra a continuacion:

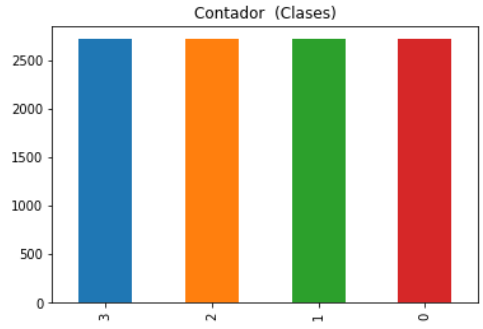
```
0    1895
3      23
1      17
2      12
Name: C, dtype: int64
```

Se realiza el resamplen para igualar las clases minoritarias a la clase mayoritaria el resultado se presenta a continuacion

```
3    1895
2    1895
1    1895
0    1895
Name: C, dtype: int64
```

Se grafica numevente histograma y observar la proporciones de las clases .

```
Clase 0: 2713
Clase 1: 2713
Clase 2: 2713
Clase 3: 2713
Proporcion: 1.0 : clase[0] / clase[1]
Proporcion: 1.0 : clase[0] / clase[2]
Proporcion: 1.0 : clase[0] / clase[3]
```



El resultado del modelo con la implementacion del oversampling en el cual pararentemente disminuye el rendimiento del modelo como se muestra a continuacion:

```
############## Accuracy Linear SVM FINAL ##################
0.8960905349794238
[[839  22  29  45]
 [  0  10   1   0]
 [  1   3   5   0]
 [  0   0   0  17]]
              precision    recall  f1-score   support

           0       1.00      0.90      0.95       935
           1       0.29      0.91      0.43        11
           2       0.14      0.56      0.23         9
           3       0.27      1.00      0.43        17

    micro avg       0.90      0.90      0.90       972
    macro avg       0.43      0.84      0.51       972
 weighted avg       0.97      0.90      0.92       972
```

Lo que se puede destacar con el oversampling es que todas las clases obtuvieron una metrica de precision positiva. Al contrario a lo que sucedia en los experimentos anteriores.

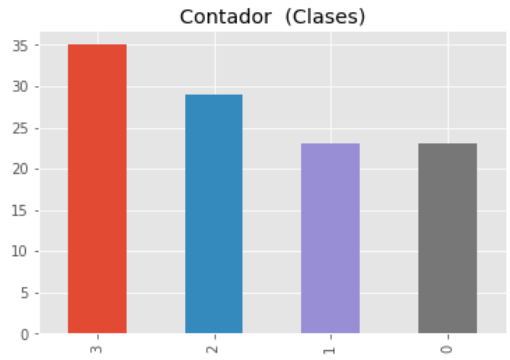
VI. Quinta Iteración "Balanceo de datos usando Undersampling"

La técnica consiste en igualar la clase mayoritaria a la clase que menor muestra posea. Los resultados después del procedimiento se muestran a continuación.

```
3    35
2    29
1    23
0    23
Name: C, dtype: int64
```

Se grafica nuevamente el histograma y se obtiene la proporcion de las clases .

```
Clase 0: 23
Clase 1: 23
Clase 2: 29
Clase 3: 35
Proporcion: 1.0 : clase[0] / clase[1]
Proporcion: 0.79 : clase[0] / clase[2]
Proporcion: 0.66 : clase[0] / clase[3]
```



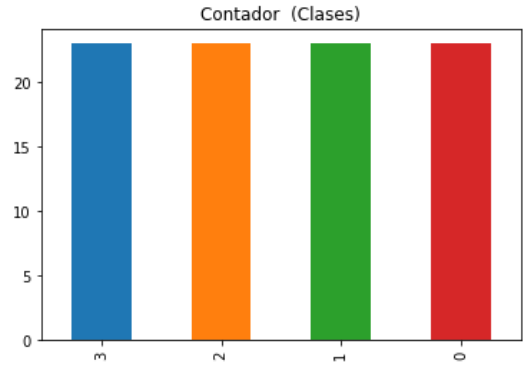
Luego de aplicar la tecnica se puede observar que el acurracy del modelo se mantiene en comparacion con oversampling, pero al explorar las otras metricas es posible observar que el porcentaje de clase 2 disminuye en dos puntos.

```
############## Accuracy Linear SVM FINAL ##################
0.8909465020576132
[[837  20  35  43]
 [  1   8   2   0]
 [  1   3   5   0]
 [  0   0   1  16]]
              precision    recall  f1-score   support

           0       1.00      0.90      0.94       935
           1       0.26      0.73      0.38        11
           2       0.12      0.56      0.19         9
           3       0.27      0.94      0.42        17

    micro avg       0.89      0.89      0.89       972
    macro avg       0.41      0.78      0.48       972
 weighted avg       0.97      0.89      0.92       972
```

VII. Sexta Iteración "Balanceo de datos usando Undersampling y Oversamplin simultáneamente"

Este experimento consiste en reducir la clase mayoritaria Clase 0 a la segunda clase mayoritaria Clase 3 que cuenta con 23 observaciones e igualar las clases minoritarias Clase 1 y Clase 2 a la Clase 3.

```
Clase 0: 23
Clase 1: 23
Clase 2: 23
Clase 3: 23
Proporcion: 1.0 : clase[0] / clase[1]
Proporcion: 1.0 : clase[0] / clase[2]
Proporcion: 1.0 : clase[0] / clase[3]
```



En los resultados de esta tecnica como a pesar de que el acurracy aumenta las metricas de precision y recall son inferiores a los modelos de oversampling y oversampling.

```
############## Accuracy Linear SVM FINAL ################
0.8816872427983539
[[830  31  28  46]
 [  1   8   2   0]
 [  1   5   3   0]
 [  0   0   1  16]]
              precision    recall  f1-score   support

           0       1.00      0.89      0.94       935
           1       0.18      0.73      0.29        11
           2       0.09      0.33      0.14         9
           3       0.26      0.94      0.41        17

   micro avg       0.88      0.88      0.88       972
   macro avg       0.38      0.72      0.44       972
weighted avg       0.97      0.88      0.92       972
```

VIII. Conclusiones

A pesar que la métrica de acurracy para los diferentes modelos en general fue alta, No es posible evaluar un modelo con tan solo esta métrica, es necesario verificar otras métricas como la precisión, recall, f1-core, también se puede concluir que el oversampling sirvió para mejorar la precisión del modelo puesto que se obtuvieron valores de precisión positivos, recall y f1-score para todas las clases.