Ciencia de datos aplicada al estudio de la Obesidad y otras enfermedades crónicas en Córdoba

Práctico N°3

Introducción al aprendizaje automático

Grupo 2

➢ Fernando Apaza
➢ Edgardo Garrigo
➢ Nicolas E Ponce

Objetivo y alcance

Para esta materia el objetivo es poder hacer un primer acercamiento a un proceso de aprendizaje automático,Nos enfocaremos en el en el proceso de: selección de un modelo, ajuste de hiperparámetros y evaluación,regulador, métricas, similar a lo que hicieron en el segundo laboratorio de esta materia. En este laboratorio no se espera que se encuentre el mejor modelo con sus mejores parámetros, sino que se logre la buena práctica de realizar los pasos necesarios en un proceso de aprendizaje automático, desde la división del dataset hasta la evaluación del modelo. Para realizar el práctico vamos a utilizar el dataset generado en la materia anterior.

La necesidad es la siguiente:

1. Poder predecir de forma automática la presencia de obesidad (todos los grados) en toda la población.
2. Poder predecir de forma automática la presencia de obesidad en mujeres (grupo 1) y todos los hombres (grupo 2).
3. Poder determinar cuales son las variables que son consideradas factores de riesgo para presentar obesidad, tanto para toda la población como discriminando por sexos.

Para ello se debe:

- Crear una variable que responda a la demanda (obesidad = IMC > 29.9)
- Cargar los datos, separando del dataset la etiqueta a predecir.
- Dividir el dataset en el conjunto de entrenamiento y conjunto de test
- Analizar y justificar qué features se utilizarán para lograr la mejor predicción.
- Elegir dos modelos de clasificación (uno por cada requerimiento). Los que Uds. se sientan más cómodos, pero también justificando conceptualmente la elección de la función de regularización.
- Entrenar y evaluar los modelos, fijando la semilla aleatoria para hacer repetible el experimento.
- En cuanto a los hiper-parámetros:
    a. Probar primero con los default y elegir alguna/s métrica/s para reportar los resultados.
    b. Luego usar grid-search y 5-fold cross-validation para explorar muchas combinaciones posibles de valores, reportando accuracy promedio y varianza para todas las configuraciones.
- Para la mejor configuración encontrada, evaluar sobre el conjunto de entrenamiento y sobre el conjunto de evaluación, reportando:
    * Accuracy
    * Precision
    * Recall
    * F1
    * Matriz de confusión

Se evaluarán los siguientes aspectos:

1- Que se apliquen los conceptos vistos con los profes en el teórico y en el práctico.

2- Que el entregable no sea solo la notebook. El informe debe tener un mensaje claro y debe presentarse en un formato legible para cualquier tipo de stakeholder.

3- Capacidad de análisis.

4- Criterio para elegir que solución aplicar en cada caso y con qué método implementarla.

Deadline pautado para la entrega: Lunes 16/08/2020

Estructura del informe

El informe debe estar en un formato que no sea Jupyter Notebook, por ejemplo .html, .pdf, .md. El objetivo es poder redactar y justificar las conclusiones obtenidas a partir de las preguntas disparadoras, utilizando material gráfico y/o interactivo como soporte para complementar las ideas. Además, se debe presentar o enviar la notebook en donde se trabajó (jupyter notebook, colab notebook, etc).

Se evaluarán los siguientes aspectos:

Que se apliquen los conceptos vistos en el dictado de la materia.

El informe debe tener un mensaje claro y debe presentarse en un formato legible para cualquier tipo de stakeholder.

Que los cálculos estadísticos sean utilizados solo como herramientas para responder a las consignas.

Indicar el criterio aplicado al momento de elegir las variables a analizar.

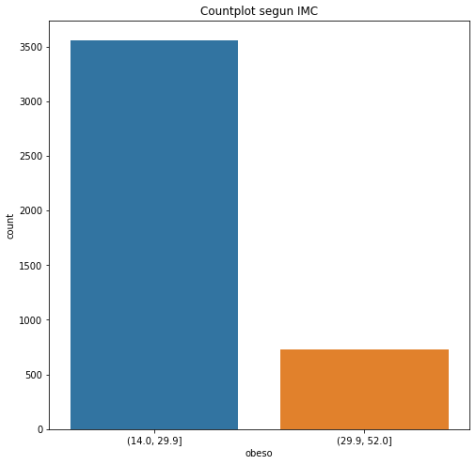
Informe:

Metodología:

El dataset original, generado en el práctico anterior, posee 4288 registros, de los cuales 1814 corresponden a la población masculina y los 2474 restantes a la femenina. Se trabajó con tres dataset, el original para estudiar la población total y los dataset correspondientes a hombres y a mujeres, para estudiar separadamente por sexo

Se generó un nuevo feature denominado "obeso" el cual tiene valor 1 si la persona presenta una IMC > 29.9 y por lo tanto es obesa y valor de 0 si la persona presenta una IMC < 29.9, considerandola como no obesa. Este feature será nuestro “Target” o etiqueta

Analisis previo del dataset: Como se puede ver el grafico countplot el atributo obeso se encuentra desbalanceados, ya que existen mas datos de encuesados "normales" = IMC(14.0, 29.9), que para aquellos que calsificados como "obesos" = IMC (29.9,52.0).



83% "normales" vs 17% obesos

Tenemos 239 features disponibles para hacer la predicción. Sin embargo, los features que a nuestro criterio ayudarán significativamente a la predicción son los siguientes:

1. dlp: Presencia de dislipemia. Es característico de la obesidad
2. dbt_tipo2: Presencia de diabetes tipo 2: los diabéticos T2 tienen mayor probabilidad de ser obesos".
3. cancer: La obesidad se asocia con al menos 13 tipos distintos de cáncer.
4. actfis: El sedentarismo está relacionado a la obesidad
5. stress: La ansiedad está relacionada a la mala alimentación
6. imagen: Imagen corporal evaluada por el entrevistador
7. nes: El nivel socioeconomico influye en el tipo de alimentación
8. peso: Es el numerador del cálculo de IMC
9. cc: Circunferencia de cintura

10. fgl1: Ingesta de glúcidos según frecuencia
11. fgr1: Ingesta de grasas según frecuencia
12. hta: La obesidad contribuye a la hipertensión por varios mecanismos
13. eent: El nivel educativo influye en el tipo de alimentación
14. falco: Ingesta de alcohol según frecuencia
15. fgas: Ingesta de gaseosas según frecuencia
16. fcrap: Ingesta de comida chatarra según frecuencia

El dataset resultante se dividió para entrenamiento (60%) y para test (40%)

De acuerdo a la bibliografía Random Forest es el algoritmo de clasificación y regresión más popular utilizado ampliamente en clasificación de enfermedades. Utilizaremos clasificadores más sencillos como **Naive Bayes (Gausiano) (NBG) y Árboles de decisión (DT)** ya que creemos más ordenado ir desde lo mas sencillo a lo mas complejo, además consideramos que, si con algoritmos sencillos se consiguen buenos resultados no es necesario incurrir en más gasto de puesta a punto y computacional.

En el algoritmo de **Árboles de decisión** implementado por la librería sklearn, el hiperparámetro referido al término de regularización se denomina **ccp_alpha**, por defecto su valor es 0. Este algoritmo denominado de poda de mínimo costo y complejidad se utiliza para podar un árbol evitando un sobre ajuste excesivo ("overfitting"). El parámetro de complejidad alfa (que puede tomar valores mayor o igual a 0), es el que define la medida de costo y complejidad, $R_\alpha(T)$ de un árbol dado T mediante la ecuación:

$$R_\alpha(T) = R(T) + \alpha|T|$$

Donde $|T|$ es el número de nodos terminales en el árbol T, y R(T) se define tradicionalmente como la tasa total de clasificación errónea de los nodos terminales.

Previamente al entrenamiento de los clasificadores, se escalaron los datos del dataset mediante StandarScaler de sklearn.

Se comenzó entrenando ambos clasificadores para los tres conjuntos de datos con sus parámetros por defectos y configurando la semilla en un valor de 42 para posibilitar la repetibilidad del experimento.

Se repitió el experimento pero esta vez mediante validación cruzada con 5-Kfold y se reportó las métricas de la media de Accuracy (Acc) y la desviación estándar (Acc Std).

Se exploraron diferentes combinaciones de hiperparametros para el clasificador DT mediante **GridSearchCV** . Los hiperparametros seleccionados fueron:

```
'max_depth': [3, 5, 10],
```

```
'criterion': ['gini', 'entropy'],
'max_features': ['auto', 'sqrt'],
'ccp_alpha': [0, 1, 3, 5]
```

Finalmente, una vez hallados los hiperparametros ideales se realizó un nuevo entrenamiento con los mismos. Al final del trabajo se reporta las métricas de Accuracy, Precision, Recall, F1, y Matriz de confusión para los clasificadores NB, DT por defecto, y DT con hiperparámetros óptimos.

Resultados:

Los valores arrojados, ver tabla 1, muestran un valor alto y muy semejante de la métrica Accuracy (Acc) , rango (0.8759 - 0.9091) para ambos clasificadores y todas los conjuntos de datos. Se observa mejor entrenamiento para la población femenina Acc 0.9010, 0.9091 seguido por población masculina Acc 0.9008 , 0.8898 y finalmente por la población total con Acc 0.8759, 0.8881 para NBG y DT respectivamente:

	Accuracy	
	NB Gaussiano	Decision Tree
Población TOTAL	0,8759	0,8881
Población Femenina	0,9010	0,9091
Población Masculina	0,9008	0,8898

Tabla 1: Resultado inicial de clasificadores con parámetros por defecto

La validación cruzada mediante 5-K fold arrojó valores similares, los mismo se pueden observar en la tabla 2. El rango de los valores medios de Acc fue de (0.885 - 0.92) y el rango de Acc Std fue de (0.009, 0.030) para ambos clasificadores y todas los conjuntos de datos. Se observa para la población femenina Acc 0.90 / Std 0.02, y Acc 0.90 / Std 0.01 seguido por población masculina Acc 0.89 / Std 0.02 , y Acc 0.92 / Std 0.03 y finalmente por la población total con Acc 0.89 / Std 0.01, Acc 0.885 / Std 0.009 para NBG y DT respectivamente

	Accuracy			
	NB Gaussiano		Decision Tree	
	Acc	STD	Acc	STD
Población TOTAL	0,89	0,01	0,885	0,009
Población Femenina	0,90	0,02	0,90	0,01
Población Masculina	0,89	0,02	0,92	0,03

Tabla 2: Resultado de clasificadores con parámetros por defecto mediante entrenamiento con 5-kfold

Mediante **GridSearchCV** se evaluaron el juego de hiperparametros encontrando los siguientes como óptimos, para toda la población *{'ccp_alpha': 0, 'criterion': 'gini', 'max_depth': 10, 'max_features': 'auto'}*, para las poblaciones femenina *{'ccp_alpha': 0, 'criterion': 'entropy', 'max_depth': 10, 'max_features': 'auto'}* y masculina el mismo juego *{'ccp_alpha': 0, 'criterion': 'gini', 'max_depth': 5, 'max_features': 'auto'}*.

En la **tabla 3** se observa el resume los valores de las diferentes métricas resultantes para los distintos dataset y distintos clasificadores

Se observa que en los DT optimizados hay una mejoría de la métrica F1, la cual refleja una mejora en *Recall* y *Precision*. Al tratarse de un dataset desbalanceado, F1 es una mejor métrica que *Accuracy* para evaluar la performance del modelo.

		NBG		DT default		DT optimo	
Población TOTAL	Accuracy	0,8759		0,8881		0,8846	
	Precision	0,6329		0,6891		0,6958	
	Recall	0,7452		0,6935		0,6419	
	F1	0,6844		0,6913		0,6678	
	Matriz de Confusion	1272	134	1309	97	1319	87
		79	231	95	215	111	199
Población Femenina	Accuracy	0,9010		0,9091		0,9162	
	Precision	0,6734		0,7225		0,7530	
	Recall	0,8024		0,7485		0,7485	
	F1	0,7322		0,7353		0,7508	
	Matriz de Confusion	758	65	775	48	782	41
		33	134	42	125	42	125
Población Masculina	Accuracy	0,9008		0,8898		0,8994	
	Precision	0,6803		0,6667		0,7167	
	Recall	0,8000		0,7200		0,6880	
	F1	0,7353		0,6923		0,7020	
	Matriz de Confusion	554	47	556	45	567	34
		25	100	35	90	39	86

Tabla 3: Métricas para los tres conjuntos de datos y los clasificadores seleccionados

Con el objetivo de averiguar qué es lo que sucedió con el modelo optimizado para el dataset completo, realizamos una cuadro comparativo de las métricas más detallado:

default model PARA TODOS

	precision	recall	f1-score	support
0	0.93	0.93	0.93	1406
1	0.69	0.69	0.69	310
accuracy			**0.89**	**1716**
macro avg	0.81	0.81	0.81	1716
weighted avg	0.89	0.89	0.89	1716

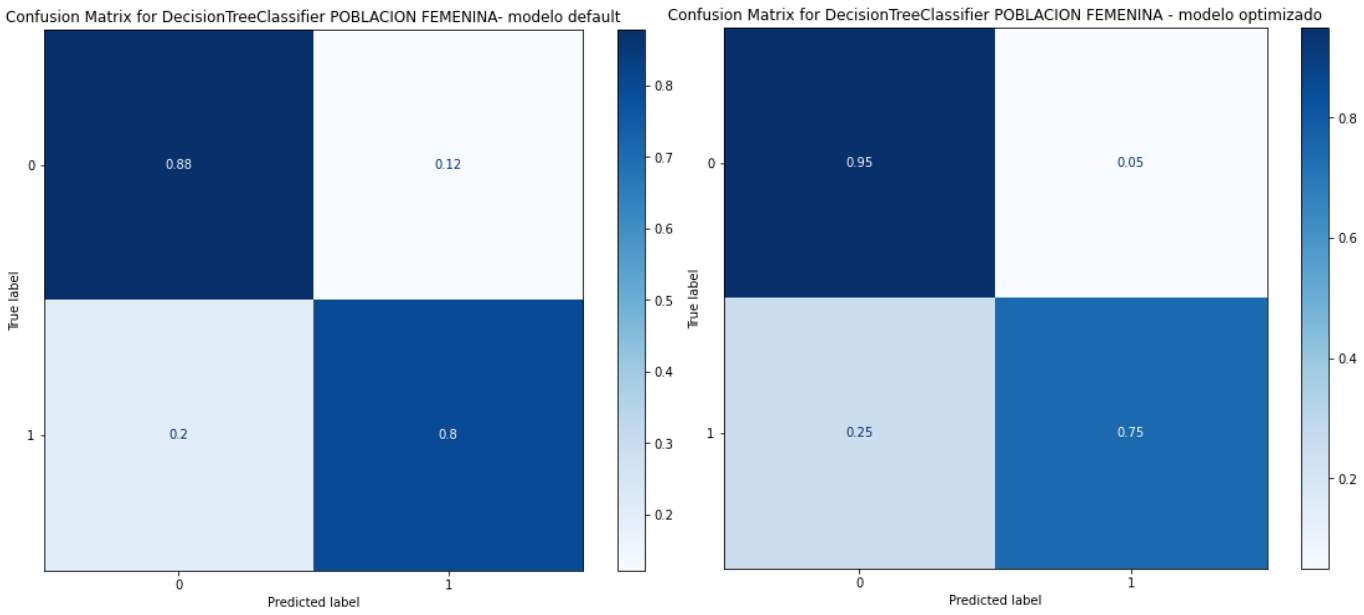
Modelo Optimizado PARA TODOS

	precision	recall	f1-score	support
0	0.92	0.94	0.93	1406
1	0.70	0.64	0.67	310
accuracy			**0.88**	**1716**
macro avg	0.81	0.79	0.80	1716
weighted avg	0.88	0.88	0.88	1716

Claramente se observa que la optimización disminuye el F1 del grupo 1 “obesos”, aún cuando aumenta la precisión. Esta nueva configuracion no seria la ideal para detectar con mayor seguridad a los obesos en el futuro.

Por otro lado, a modo de ejemplo y para hacer más evidente la mejoras que si se obtuvieron en el modelo DT para la población femenina, se gráfica la matriz de confusión con los datos normalizados. El modelo con hiperparámetros por default (A) y optimizado (B).

A) **B)**



Siendo un modelo optimizado, mejoró en la detección de mí población “0” = “normales”, pero disminuyó la capacidad de detectar “1”=obesos. Hay que considerar que si lo que queremos es detectar los obesos con mayor seguridad debemos aumentar la performance de los True positive para “1”.

Conclusion:

Se logró aplicar, a la base de datos depurada en el práctico anterior, los conceptos adquiridos durante la cursada de la materia Introducción al Aprendizaje Automático.

Los 16 features seleccionados fueron correctos ya que con los clasificadores planteados se obtuvieron muy buenos resultados y semejantes entre ellos.

Al realizar la validación cruzada de los clasificadores, se observó una desviación estándar del 3 y 0.9%. Estos valores son extremadamente bajos, lo que significa que los modelos propuestos tienen una varianza muy baja, lo que en realidad es muy bueno, ya que eso significa que la predicción que obtuvimos en un conjunto de prueba no es por casualidad. Por el contrario, el modelo tendrá un rendimiento más o menos similar en todos los conjuntos de datos de prueba.

Los hiperparametros encontrados como óptimos nos dicen que no fue necesario el uso de regularización, lo que se entiende que las variables pudieron ser segmentadas con

relativa facilidad sin incurrir en un sobre ajustado u “overfitting”. Los parámetros hallados como óptimos son diferentes segun el dataset empleado. La profundidad del DT varía entre 10 y 5 lo que indica que se convergio a la solución, relativamente rápido, repercutiendo en ahorro de cálculo.

Los modelos DT entrenados con la base de datos discriminada por sexo, luego de la optimizacion mejoran su performance, tanto en accuracy como en F1. Al observar todas las métricas uno podría seleccionar al clasificador NBG como el elegido para implementar debido a los buenos resultados obtenidos y la economía computacional que brinda. Sin embargo, el aumento de la complejidad del clasificador DT no es muy grande y se obtienen mejores valores en la métrica matriz de confusión (menores evaluaciones erróneas y más evaluaciones correctas).

Sin embargo el modelo de la base de datos completa, luego de la optimización presenta menor rendimiento en métricas de accuracy y F1. Es necesario explorar y configurar más combinaciones de hiperparámetros del modelo clasificador DT mediante **GridSearchCV** para llegar a una optimización real. En base a estos resultados, es preferible trabajar con el modelo por default.

Quedaría como trabajo futuro evaluar Random Forest para observar si hay un incremento significativo en la performance, y eventualmente otros clasificadores que podamos aprender en las próximas materias de la Diplomatura.