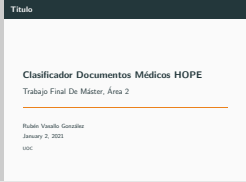
Clasificador Documentos Médicos HOPE

Trabajo Final De Máster, Área 2

Rubén Vasallo González

January 2, 2021

UOC



Índice

Presentar proyecto ayuda a profesionales sanitarios encontrar referencias bibliográficas.

Intro: Explicaremos los objetivos.

Metodología: Analizaremos los datos y comentaremos problemas que nos han surgido durante el proyecto y como los hemos solventado.

Modelos Predictivos: Veremos 3 modelos predictivos analizados.

Conclusiones: Comentaremos las conclusiones finales.

Introducción

Introducción

Introducción

- Ayudar a los profesionales sanitarios a encontrar referencias bibliográficas adaptadas y personalizadas al paciente.
- Ayudar al proyecto HOPE a mejorar su algoritmo de Inteligencia Artificial (NLP).

Introducción
- Ayudar a los profesionales sanitarios a encontrar referencias bibliográficas adaptadas y personalizadas al paciente.
- Ayudar al proyecto HOPE a mejorar su algoritmo de Inteligencia Artificial (NLP).

El master nace con el objetivo de poder ayudar a HOPE a recomendar referencias bibliográficas en base al feedback obtenido de los propios profesionales sanitarios.
HOPE: (Health Operations for Personalized Evidence) basado en (NLP) identifica la información clave de casos clínicos y referencias bibliográficas registradas en la Historia Clínica Electrónica. Pero no siempre acaba obteniendo referencias utiles para los casos personalizados.
Actualmente profesionales sanitarios dan feedback de esas recomendaciones y se quiere aprovechar esa información para mejorar el algoritmo.

Objetivos

- **(OP)** Recomendar al profesional sanitario las referencias bibliográficas actuales útiles y personalizadas pudiendo realizar una clasificación (ranking) de más interés a menos.

Objetivos
- **(OP)** Recomendar al profesional sanitario las referencias bibliográficas actuales útiles y personalizadas pudiendo realizar una clasificación (ranking) de más interés a menos.

El objetivo es poder recomendar al profesional sanitario referencias bibliográficas ordenadas de más interés (utilidad) a menos.

Objetivo principal, entrenar un modelo predictivo basado en series temporales para poder realizar la clasificación por ranking.

Objetivos

- **(OP)** Recomendar al profesional sanitario las referencias bibliográficas actuales útiles y personalizadas pudiendo realizar una clasificación (ranking) de más interés a menos.
- *(OS)* Analizamos el Conjunto de datos y componentes.
- *(OS)* Enriquecemos el conjunto de datos.
- *(OS)* Analizamos 3 posibles modelos predictivos.
- *(OS)* Realizamos la recomendación final

Objetivos
- **(OP)** Recomendar al profesional sanitario las referencias bibliográficas actuales útiles y personalizadas pudiendo realizar una clasificación (ranking) de más interés a menos.
- *(OS)* Analizamos el Conjunto de datos y componentes.
- *(OS)* Enriquecemos el conjunto de datos.
- *(OS)* Analizamos 3 posibles modelos predictivos.
- *(OS)* Realizamos la recomendación final

Para ello analizaremos el conjunto de datos y sus componentes (atributos).

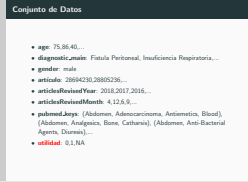
Intentamos enriquecer el conjunto debido a su poco volumen de datos.

Analizamos ...

Metodología

Conjunto de Datos

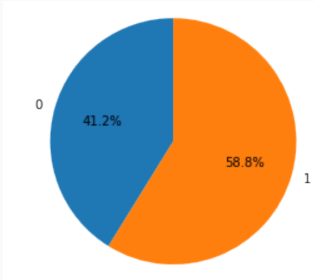
- **age**: 75,86,40,...
- **diagnostic_main**: Fistula Peritoneal, Insuficiencia Respiratoria,...
- **gender**: male
- **artículo**: 28694230,28805236,...
- **articlesRevisedYear**: 2018,2017,2016,...
- **articlesRevisedMonth**: 4,12,6,9,...
- **pubmed_keys**: (Abdomen, Adenocarcinoma, Antiemetics, Blood), (Abdomen, Analgesics, Bone, Catharsis), (Abdomen, Anti-Bacterial Agents, Diuresis),...
- **utilidad**: 0,1,NA



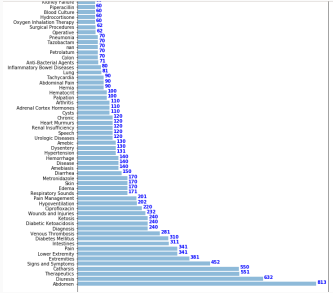
- Conjunto de datos con muchísimos atributos en formato documento y sin que persistan en todas las observaciones.
- Se decide junto al cliente escoger los atributos que mas se repiten en todas las observaciones, que son los que se muestran a continuación.

Limitaciones encontradas

- Poco volumen de información (51 obs. con atrb. *utilidad* informado).
- Atributo a predecir sin la información suficiente para poder hacer un *ranking*.
- Conjunto de datos Sesgado.



(a) Atributo *utilidad*

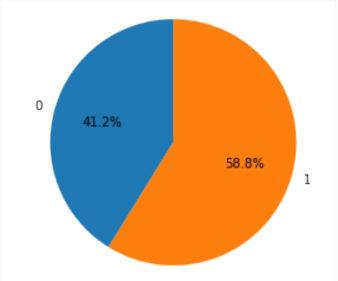


(b) Atributo *pubmed_keys*

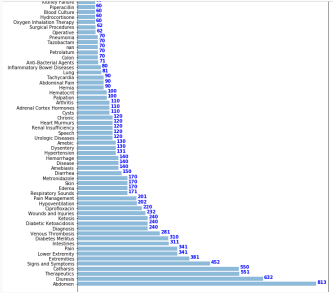
- Conjunto de datos con poco volumen.
- No todas las observaciones tienen indicado el atributo a predecir (utilidad).
- No se puede realizar un *ranking* con la información actual.
- Para intentar ganar algo de volumen se decide expandir el atributo pubmed_keys para tener mas volumen de datos y detectar si algunas keywords tienen más importancia que otros.
- Se decide con el cliente hacer un clasificador binario para indicar si el articulo es útil o no. Si se consigue el detalle de la utilidad durante el transcurso del ejercicio se volvería a objetivo principal.
- Como existen observaciones sin el atributo utilidad, se acuerda intentar enriquecer el conjunto de datos basándonos en los atributos que si tienen informado el atributo a predecir.

Limitaciones encontradas

- Poco volumen de información (51 obs. con atrb. *utilidad* informado).
- Atributo a predecir sin la información suficiente para poder hacer un *ranking*.
- Conjunto de datos Sesgado.



(a) Atributo *utilidad*



(b) Atributo *pubmed_keys*

- También se decide generar un segundo conjunto de datos partiendo del conjunto de datos original pero expandiendo el atributo *pubmed_keys* para ver si la información de ese atributo a nivel individual puede ser un valor determinante.
- Se observa el conjunto de datos sesgado en el atributo a predecir y en el atributo *pubmed_keys*. Esto afectara a que la mayoría de modelos seguramente estarán sobreajustados.

Análisis de componentes principales (PCA)

- **Conjunto 1**: Dataset Completo
- **Conjunto 2**: Dataset Completo pero añadiendo el mes y año del artículo y cogiendo solo las observaciones que se ha informado el atributo utilidad.
- **Conjunto 3**: Dataset conjunto 2 pero eliminando los atributos *gender* y artículo y se expande el atributo *pubmed_keys*

Análisis de componentes principales (PCA)
- **Conjunto 1**: Dataset Completo
- **Conjunto 2**: Dataset Completo pero añadiendo el mes y año del artículo y cogiendo solo las observaciones que se ha informado el atributo utilidad.
- **Conjunto 3**: Dataset conjunto 2 pero eliminando los atributos *gender* y artículo y se expande el atributo *pubmed_keys*

- Conjunto 1 con age, diagnostic_main, gender, artículo, pubmed_keys y utilidad.
- Conjunto 2 añadiendo el mes y año del artículo y cogiendo solo las observaciones que se ha informado el atributo utilidad.
- Conjunto 3

Análisis de componentes principales (PCA)

- Se Transformo todos los atributos Categóricos (texto) a Continuos (números continuos).
- Se estandarizó los valores a un rango de entre 1 y -1.
- Para el entrenamiento de los modelos predictivos, se dividió el conjunto en dos grupos. 1 Grupo de entrenamiento con el 75% de observaciones. 2 Grupo para la validación con el 25% de observaciones.

Análisis de componentes principales (PCA)
- Se Transformo todos los atributos Categóricos (texto) a Continuos (números continuos).
- Se estandarizó los valores a un rango de entre 1 y -1.
- Para el entrenamiento de los modelos predictivos, se dividió el conjunto en dos grupos. 1 Grupo de entrenamiento con el 75% de observaciones. 2 Grupo para la validación con el 25% de observaciones.

Resultados del análisis de componentes principales (PCA)

- Conjunto 1: Con solo **3 atributos**, el modelo es capaz de **explicar (predecir) el 95%** de las observaciones.

```
pd.DataFrame(pca.components_,columns=features,index = ['PC1','PC2','PC3'])
```

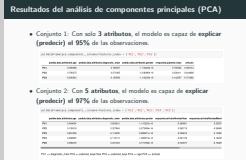
	pedido.data.attributes.age	pedido.data.attributes.diagnostic_main	pedido.data.attributes.gender	respuesta.pubmed_keys	articulo
PC1	-0.050066	0.705007	-1.110223e-16	0.705462	0.052744
PC2	0.760475	0.071805	-1.318390e-16	0.030411	-0.644668
PC3	-0.630542	-0.130930	-1.110223e-16	0.142297	-0.751682

- Conjunto 2: Con **5 atributos**, el modelo es capaz de **explicar (predecir) el 97%** de las observaciones.

```
pd.DataFrame(pca.components_,columns=features,index = ['PC1','PC2','PC3','PC4','PC5'])
```

	pedido.data.attributes.age	pedido.data.attributes.diagnostic_main	pedido.data.attributes.gender	respuesta.articlesRevisedYear	respuesta.articlesRevisedMont
PC1	-0.094054	0.603912	-1.110223e-16	-0.365651	0.33837
PC2	0.135318	0.327845	-5.273559e-16	0.362710	-0.55606
PC3	-0.891259	-0.174252	-5.828671e-16	-0.236812	-0.13360
PC4	0.382620	-0.115174	-1.929013e-15	-0.621075	0.16245
PC5	0.098479	0.051453	3.774758e-15	-0.540677	-0.72925

PC1 => diagnostic_main PC2 => pubmed_keys/Year PC3 => pubmed_keys PC4 => age PC5 => articulo



La diferencia entre el conjunto 1 y 2 seguramente este debido a que el conjunto completo contiene un gran volumen de observaciones con el atributo *utilidad* sin definir, lo que hace que prácticamente cualquier valor que tengan los atributos se acaben asociando a un resultado sin definir.

Resultados del análisis de componentes principales (PCA)

- Conjunto 3: Con solo **4 atributos**, el modelo es capaz de **explicar (predecir) el 90%** de las observaciones.

```
pd.DataFrame(pca.components_,columns=features,index = ['PC1','PC2','PC3','PC4'])
```

	pedido.data.attributes.age	pedido.data.attributes.diagnostic_main	respuesta.articlesRevisedYear	respuesta.articlesRevisedMonth	respuesta.pubmed_keys
PC1	0.202186	0.688327	-0.236152	-0.080358	0.650462
PC2	-0.299028	-0.000864	-0.648047	0.698266	-0.055149
PC3	-0.908866	-0.006936	0.153411	-0.221720	0.318152
PC4	-0.015928	-0.145721	-0.702451	-0.673071	-0.179024

PC1 => diagnostic_main / pubmed_keys PC2 => articlesRevisedMonth PC3 => pubmed_keys PC4 => age

Descartamos usar el conjunto 1 para los modelos debido a su gran volumen de resultados sin definir. Utilizaremos los otros dos conjuntos para analizar los resultados de los posteriores modelos.

Intentando enriquecer los datos (K-Nearest-Neighbor)

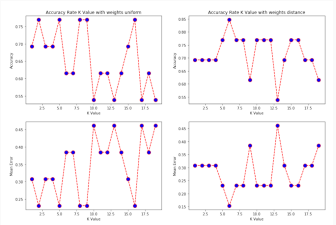
- **Conjunto 1**: Dataset Completo cogiendo solo las observaciones que se ha informado el atributo utilidad.
- **Conjunto 2**: Dataset conjunto 1 pero eliminando los atributos *gender* y artículo y se expande el atributo *pubmed_keys*

Intentando enriquecer los datos (K-Nearest-Neighbor)
- **Conjunto 1**: Dataset Completo cogiendo solo las observaciones que se ha informado el atributo utilidad.
- **Conjunto 2**: Dataset conjunto 1 pero eliminando los atributos *gender* y artículo y se expande el atributo *pubmed_keys*

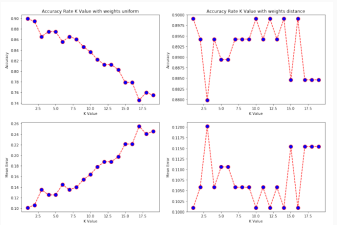
- Conjunto 1 con age, diagnostic_main, gender, artículo, mes y año del articulo, pubmed_keys y cogiendo solo las observaciones que se ha informado el atributo utilidad.
- Conjunto 2
- Se aplican a los dos conjuntos las transformaciones mencionadas anteriormente.

Resultados del enriquecimiento (K-Nearest-Neighbor)

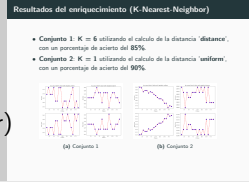
- **Conjunto 1**: **K = 6** utilizando el calculo de la distancia '**distance**', con un porcentaje de acierto del **85%**.
- **Conjunto 2**: **K = 1** utilizando el calculo de la distancia '**uniform**', con un porcentaje de acierto del **90%**.



(a) Conjunto 1



(b) Conjunto 2



- distance: Cuanto mas cerca, más probabilidad de pertenecer al grupo (mas peso).
- uniform: Todos los puntos tienen el mismo peso.

Aconsejamos no utilizar el modelo, ya que los conjuntos se observan **sesgados** y esto puede afectar al entrenamiento de los posteriores modelos.

Análisis Modelos Predictivos

Conjuntos de datos para el estudio de los modelos

- **Conjunto 1**: Dataset Completo cogiendo solo las observaciones que se ha informado el atributo utilidad.
- **Conjunto 2**: Dataset conjunto 1 pero eliminando los atributos *gender* y artículo y se expande el atributo *pubmed_keys*

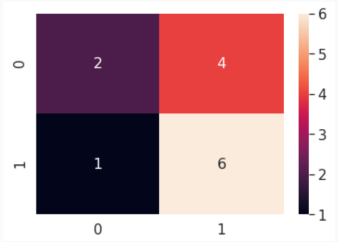
Nota: Se utilizaron los mismos conjuntos para los 3 modelos y se aplican las mismas transformaciones mencionadas anteriormente.

Conjuntos de datos para el estudio de los modelos
• **Conjunto 1**: Dataset Completo cogiendo solo las observaciones que se ha informado el atributo utilidad.
• **Conjunto 2**: Dataset conjunto 1 pero eliminando los atributos *gender* y artículo y se expande el atributo *pubmed_keys*
Nota: Se utilizaron los mismos conjuntos para los 3 modelos y se aplican las mismas transformaciones mencionadas anteriormente.

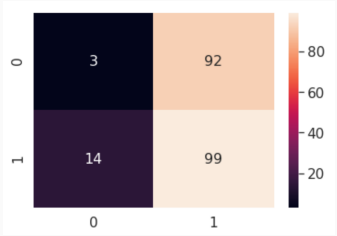
- Conjunto 1 con age, diagnostic_main, gender, artículo, mes y año del articulo, pubmed_keys y cogiendo solo las observaciones que se ha informado el atributo utilidad.
- Conjunto 2
- Se aplican a los dos conjuntos las transformaciones mencionadas anteriormente.

Resultados del Modelo 1 (Regresión logística)

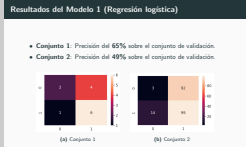
- **Conjunto 1**: Precisión del **65%** sobre el conjunto de validación.
- **Conjunto 2**: Precisión del **49%** sobre el conjunto de validación.



(a) Conjunto 1



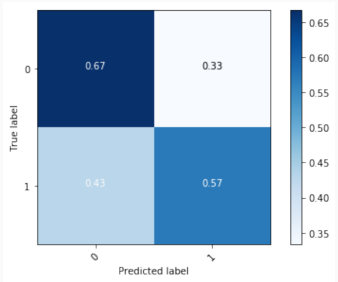
(b) Conjunto 2



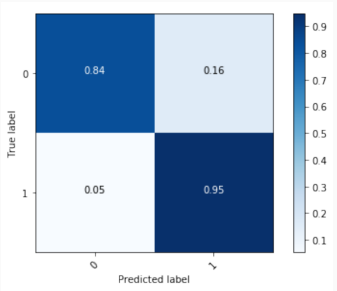
Explicar Resultados Regresión logística

Resultados del Modelo 2 (Random Forests)

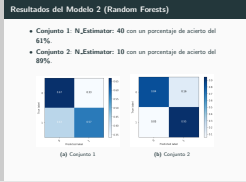
- **Conjunto 1**: **N_Estimator: 40** con un porcentaje de acierto del **61%**.
- **Conjunto 2**: **N_Estimator: 10** con un porcentaje de acierto del **89%**.



(a) Conjunto 1



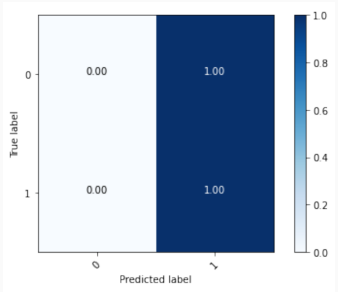
(b) Conjunto 2



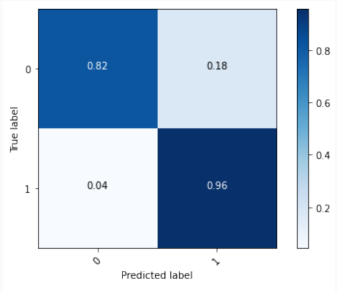
- Se realiza el estudio del número de arboles que ha de tener el modelo para que este de su porcentaje de acierto mas elevado.
- Con el conjunto 2, el modelo se ajusta a unos resultados aceptables ya que esta lo suficientemente ajustado para que de resultados razonables sin estar sobreajustado.

Resultados del Modelo 3 (Support Vector Machines)

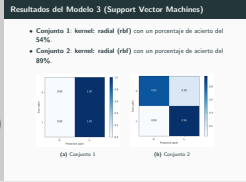
- **Conjunto 1**: **kernel: radial (rbf)** con un porcentaje de acierto del **54%**.
- **Conjunto 2**: **kernel: radial (rbf)** con un porcentaje de acierto del **89%**.



(a) Conjunto 1



(b) Conjunto 2



- Se realizo un estudio de con los kernel 'polynomial', 'radial (rbf)' y 'sigmoid' y calculamos cuales son los mejores hiperparametros para cada uno de estos.
- Se observo que el kernel que mejor resultado era el radial y para el conjunto 2, el modelo se ajusta a unos resultados aceptables igual que ocurre para el caso del modelo de Random Forests.

Conclusiones y Trabajos futuros

Comparando los Modelos

Modelos	Logit	**Bosques Aleatorios**	SVM
Conjunto 1	65.78%	61.53%	53.85%
Conjunto 2	49.03%	**89.9%**	89.42%

Comparando los Modelos

Modelos	Logit	Bosques Aleatorios	SVM
Conjunto 1	65.78%	61.53%	53.85%
Conjunto 2	49.03%	**89.9%**	89.42%

- El modelo que mejor precisión da es el Bosques aleatorios.
- Para poder predecir nuevos resultados, sera necesario aplicar a estos, las transformaciones que se han aplicado al conjunto 2.

Nota: No añadimos a la comparativa el modelo K-NN ya que, aunque se podría utilizar como modelo predictivo igual que en los otros casos, debido a que no se entreno con los mismos atributos que los otros modelos.

Trabajos futuro

- Mejorar el modelo actual.
  - Mejorar el actual modelo de Bosques aleatorios añadiendo más observaciones.
  - Valorar si otros modelos predictivos tienen mejor resultado.
  - Valorar si se altera la importancia de los atributos relevantes.
- Entrenar un modelo capaz de realizar un ranking de utilidad.

Trabajos futuro
- Mejorar el modelo actual.
  - Mejorar el actual modelo de Bosques aleatorios añadiendo más observaciones.
  - Valorar si otros modelos predictivos tienen mejor resultado.
  - Valorar si se altera la importancia de los atributos relevantes.
- Entrenar un modelo capaz de realizar un ranking de utilidad.

- Si se consiguen mas observaciones, se puede reentrenar el modelo de Bosques aleatorios para mejorar la predicción de este.

- Al conseguir mas observaciones, se puede probar con mas modelos para ver si alguno mejora la predicción del de Bosques aleatorios.

- Al añadir mas observaciones, puede ser que se altere la importancia de los atributos relevantes, por lo que conviene realizar el PCA cada cierto tiempo para comprobar si esto sucede.

Gracias por esta oportunidad

¿Preguntas, dudas?

rvasallo@uoc.edu

Gracias
Gracias por esta oportunidad
¿Preguntas, dudas?
rvasallo@uoc.edu