Entrega Final - Introducción a la Inteligencia Artificial: ¿Se cancelará la reserva del hotel?

Integrantes.

Paola Andrea Posada Restrepo <paola.posada1@udea.edu.co>

Stiven Guerra Chaverra<stiven.guerra@udea.edu.co>

Sebastián Gómez Ramírez <sebastian.gomez35@udea.edu.co>

1. Descripción del problema predictivo a resolver

Tras buscar entre múltiples datasets que se adecuarán al presente proyecto se decidió abordar un problema de aprendizaje supervisado de clasificación, el cual busca determinar si la reserva realizada en un hotel va a ser cancelada o no; esta clasificación se realiza utilizando información relacionada a una reserva o a la evolución de la misma. El problema abordado puede contribuir a diferentes hoteles para tener en cuenta una posible cancelación y así considerar el uso del espacio reservado.

2. Descripción del dataset a utilizar

Para el desarrollo del ejercicio mencionado se hará uso del dataset *Hotel Booking* tomado de Kaggle, el cual cuenta con 119.391 datos reales recolectados de un Hotel de ciudad y un Hotel Resort entre las fechas 01 de Julio de 2015 y 31 de Agosto del 2017. Para el caso de uso se realizó una selección solo para el año 2015, trabajando con un total de 21996 para facilitar el procesamiento y trabajo con el conjunto de datos. Este dataset contiene múltiples datos respecto a cada reserva realizada, las cuales poseen más de 30 características entre las que destacan la fecha de la reserva, cantidad de niños, cantidad de bebés, país de proveniencia, cantidad de cambios en la reserva, si ha tenido una reserva cancelada previamente, entre otros. También vale la pena destacar varios datos categóricos como el tipo de cliente, tipo de habitación, el tipo del depósito, etc. Finalmente, se cuenta también con el dato de si la reserva fue tomada o cancelada.

3. Iteraciones de desarrollo.

a. Preprocesado de datos

Se realizó una búsqueda de los datos faltantes y afortunadamente para nuestro modelo y se encontró que el dataset presenta datos nulos en 3 columnas: país, compañía y agente; con 133, 20691 y 3099 datos nulos respectivamente.

La primera característica electa a eliminar, derivada de la decisión inicial de tomar datos que solo comprenden fechas dentro del año 2015, es en efecto la columna *arrival_year*, ya que todos los registros contarán con el mismo dato en esta columna.

Como se mencionó previamente, al empezar con el análisis de los datos de tipo objeto, se deciden eliminar un total de 5 características debido a su variedad de datos, impidiendo así la categorización de las mismas. Las características mencionadas son: *name*, *email*, *card_number*, *phone_number* y *reservation_status_date*. Además, debido a la alta correlación de la columna reservation_status con nuestra columna is_cancel, se elimina esta primera puesto que significan casi lo mismo.

El siguiente paso dado correspondía a identificar datos nulos en el dataset y durante el proceso decidir si hay alguna columna que deba ser eliminada debido a una alta cantidad de datos faltantes. Para nuestra fortuna no fue necesario eliminar columnas por esta causa. Sin embargo, se identificaron datos nulos en las columnas *country, children, agent, company.* Cabe destacar que más del 90% de datos de la columna *company* correspondían a datos vacíos, no obstante, debido a la definición de esta columna, se decidió conservarla ya que el dato vacío se puede interpretar como que no se realizó reserva por medio de una compañía.

Volviendo al procedimiento realizado al resto de columnas con datos faltantes, se les realizó un tratamiento en el cual se reemplazaron los datos faltantes por 0, indicando que no tenían niños en la columna *children*, no tenía compañía o agencia en las columnas *company, agent,* respectivamente o NC, que significa *No Country* para columna *country*.

Finalmente, las columnas candidatas para categorizar fueron las siguientes:

- hotel
- arrival_month_date
- meal
- company
- agent
- country
- market_segment
- distribution_channel
- is_repeated_guest
- reserved_room_type
- assigned_room_type
- deposit_type
- customer_type

Para todas las columnas categóricas que seguían siendo de tipo objeto, se les realizó un encoding para permitir un escalado posterior.

Se finaliza el proceso de preprocesamiento con un total de 28 características, donde 13 de ellas son categóricas.

b. Modelos Supervisados

La primera técnica de la cual se realizó implementación fue el ***Random Forest***, principalmente debido a la familiaridad que se ha tenido con esta a lo largo de los laboratorios. Sin embargo, se tuvo en mente que el problema principal que se presenta con este algoritmo es el sobreajuste, pero al investigar al respecto, se descubrió que este problema se puede solucionar haciendo *escalado* de los datos, lo cual, ya se estaba realizando.

Posteriormente se investiga y se presenta la opción de trabajar con ***Support Virtual Machines***, específicamente con el ***Support Vector Classifier***. La excusa para este caso es realizar un contraste con el Random Forest, mientras que este último presenta problemas de overfitting, el ***SVM*** es bastante resiliente al overfitting.

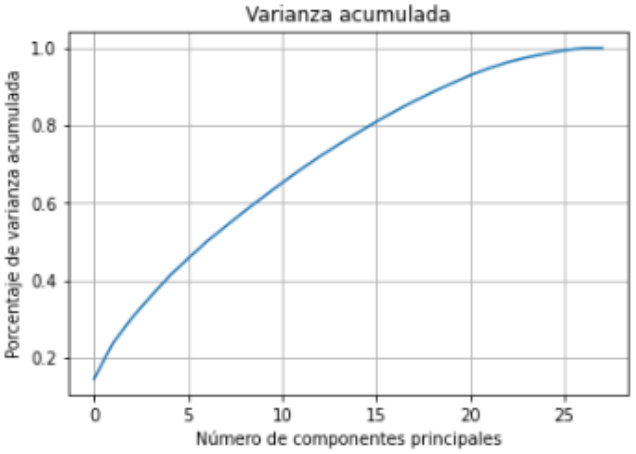
Para encontrar las mejores combinaciones de hiper parámetros para cada algoritmo se utilizó el GridSearch. A continuación se presentan las mejores combinaciones de hiperparametros encontradas para RF y SVM respectivamente.

```
Fitting 3 folds for each of 56 candidates, totalling 168 fits
RandomForestClassifier(class_weight='balanced', criterion='entropy',
                       max_features=8, n_estimators=200)
0.9846535828015716
```

```
Fitting 3 folds for each of 40 candidates, totalling 120 fits
SVC(C=10, class_weight='balanced', gamma=0.01)
0.9670223128925878
```

c. Modelos No Supervisados

Para la implementación de un modelo no supervisado nos aferramos al *Principal Component Analysis* (PCA), justificado con la idea de que el dataset demuestra alta correlación entre unos cuantos componentes, los cuales se podrían reducir.
Después de analizar la gráfica de Varianza acumulada Vs Número de componentes principales, se puede concluir que un número óptimo de componentes principales podría estar entre 22 y 25, para evitar una penalización muy alta sobre las métricas de desempeño.



Mediante el uso de gridsearch se encontró que el mejor hiper parámetro para trabajar con la combinación de PCA y Random Forest eran 27 componentes; por otra parte para la combinación de PCA y Support Virtual Machine el número de componentes debe ser 28.

```
('pca', PCA(n_components=27)),
```

```
('pca', PCA(n_components=28))
```

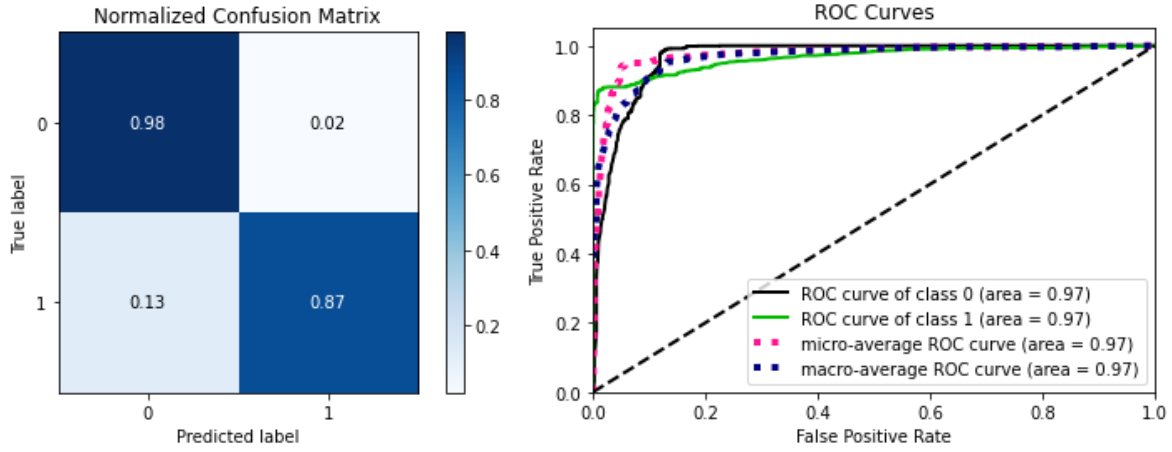
d. Resultados, métricas y curvas de aprendizaje

Para el caso se hizo uso de dos métricas muy directas como lo son el *balanced accuracy score* y el *roc auc score*. Estas métricas proveen un dato entre 0 y 1 y una curva de precisión, respectivamente. A continuación presentamos los resultados obtenidos por estas métricas para cada modelo utilizado:

- **Random Forest**

	precision	recall	f1-score	support
0	0.93	0.98	0.96	1385
1	0.97	0.87	0.92	814
accuracy			0.94	2199
macro avg	0.95	0.93	0.94	2199
weighted avg	0.94	0.94	0.94	2199

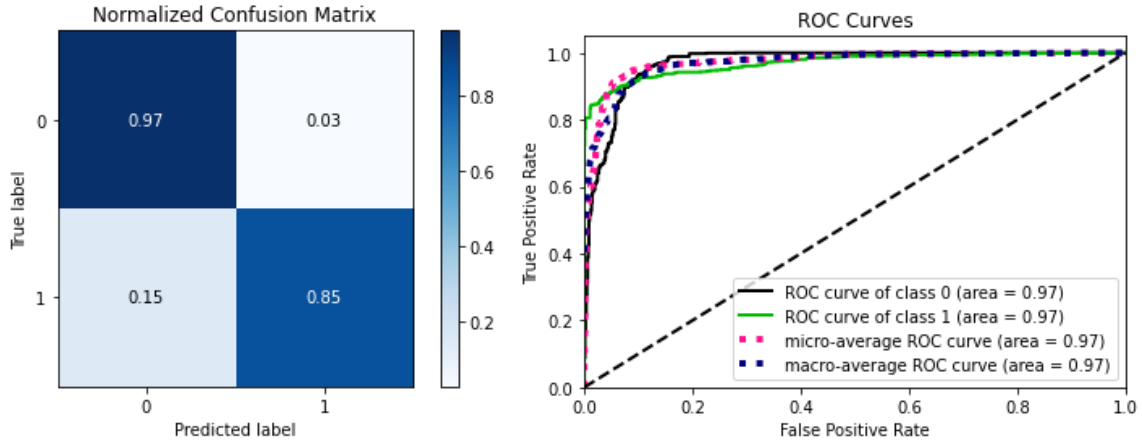
	AUC medio	AUC intervalo de confianza	Accuracy medio	Accuracy intervalo de confianza
0	0.870113	0.127721	0.796034	0.135428



- **Support Virtual Machines - Support Vector Classifier**

	precision	recall	f1-score	support
0	0.92	0.97	0.94	1385
1	0.95	0.85	0.90	814
accuracy			0.93	2199
macro avg	0.93	0.91	0.92	2199
weighted avg	0.93	0.93	0.93	2199

	AUC medio	AUC intervalo de confianza	Accuracy medio	Accuracy intervalo de confianza
0	0.910188	0.075145	0.783461	0.136358



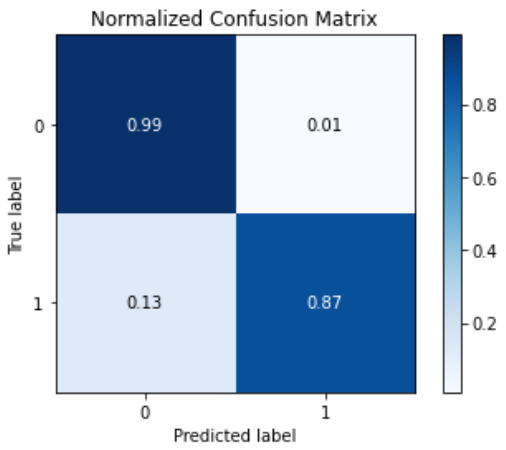
- **PCA y Random Forest**

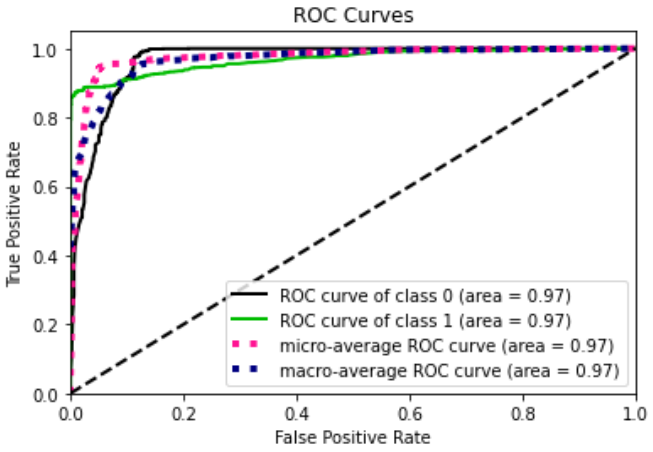
Mejor combinación de hiperparametros:

```
Fitting 3 folds for each of 336 candidates, totalling 1008 fits
Pipeline(steps=[('pca', PCA(n_components=27)),
                ('forest',
                 RandomForestClassifier(class_weight='balanced',
                                        criterion='entropy',
                                        n_estimators=250))])
0.9749983687814359
```

	precision	recall	f1-score	support
0	0.93	0.99	0.96	1385
1	0.98	0.87	0.92	814
accuracy			0.95	2199
macro avg	0.95	0.93	0.94	2199
weighted avg	0.95	0.95	0.95	2199

	AUC medio	AUC intervalo de confianza	Accuracy medio	Accuracy intervalo de confianza
0	0.868249	0.134069	0.794589	0.136229





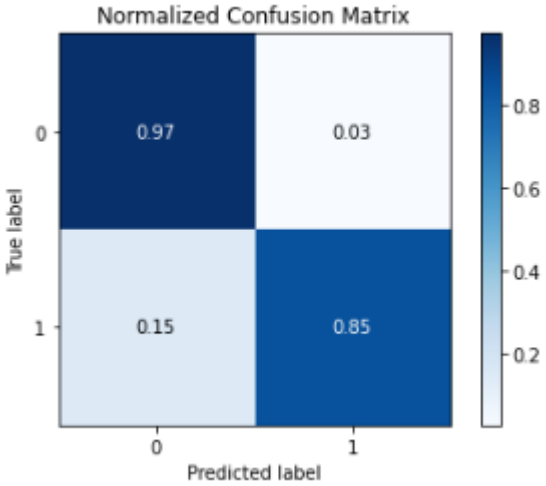
- **PCA y Support Virtual Machine**

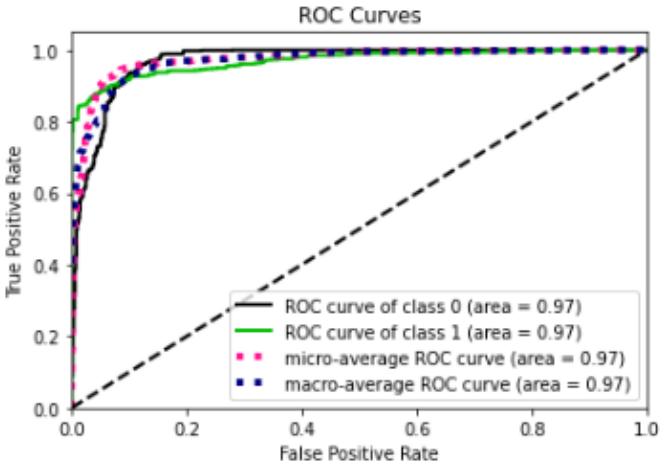
Mejor combinación de hiperparametros:

```
Fitting 3 folds for each of 240 candidates, totalling 720 fits
Pipeline(steps=[('pca', PCA(n_components=28)),
                ('svm', SVC(C=10, class_weight='balanced', gamma=0.01))])
0.96747074263061
```

	precision	recall	f1-score	support
0	0.92	0.97	0.94	1385
1	0.95	0.85	0.90	814
accuracy			0.93	2199
macro avg	0.93	0.91	0.92	2199
weighted avg	0.93	0.93	0.93	2199

	AUC medio	AUC intervalo de confianza	Accuracy medio	Accuracy intervalo de confianza
0	0.910102	0.075257	0.783353	0.13634





4. Retos y consideraciones de despliegue

A este punto no se logran identificar retos significativos más que la utilización de métricas posiblemente más apropiadas que requieren más investigación y estudio de implementación. Entre retos menos complejos se puede pensar en la implementación de otros algoritmos y el uso de diferentes hiper parámetros fuera de los proporcionados por el GridSearch.

5. Conclusiones

Hablando de las conclusiones en orden de procedimiento, se empieza por darle un reconocimiento a la importancia del reconocimiento descriptivos de las características, puesto que esto supondrá la existencia de datos que generan ruido en el modelo. Por lo que es importante entender los datos con los que se trabajan para poder remover datos de tipo único como identificaciones, correos, etc; transformar fechas si son requeridas y finalmente, poder entender por qué existen altas correlaciones.

Siguiendo con el tratamiento de datos hay que destacar el efecto que puede tener un escalado de datos sobre el resultado final, la influencia de este proceso dependerá en gran medida de primero, si el algoritmo a usar se ve afectado por datos con alta varianza y segundo, si los datos realmente cuentan con alta varianza.

En conclusiones más específicas al dataset manipulado, primero cabe señalar que el uso de PCA no logra generar una diferencia significativa o cambio notable, esto se debe a que la reducción realizada por el algoritmo es de 28 a 27 características, lo cual no genera ningún cambio apreciable en el procedimiento.

Entrando en discusión entre los algoritmos supervisados, se puede notar que nuevamente, no existe mucha diferencia entre los resultados obtenidos, todos son bastante coincidentes y no presentan ningún comportamiento que haga valera la pena de presentar las diferencias.

Finalmente, solo queda mencionar que el procedimiento de análisis de datos se llevó a cabo de manera satisfactoria, el modelo parece reaccionar de manera correcta a los datos y predecir resultados acertadamente, las métricas presentan un buen índice de aceptación sobre las predicciones, por lo que se concluye que se realizó un buen trabajo tanto en la recolección de la data (por parte del creador) como en el manejo de los datos (por parte de los desarrolladores).