Desafío - Predicción de cancelación de reserva

En este desafío validaremos nuestros conocimientos de la sesión. Para lograrlo, necesitarás desarrollar un modelo de red neuronal multicapa, al que deberás ajustar su cantidad de neuronas, cantidad capas ocultas, funciones de activación y aplicar técnicas de regularización. Este modelo será aplicado a un conjunto de datos de clientes que han realizado reservas en un hotel urbano y en un hotel turístico. El objetivo es predecir si un cliente cancelará o no la reserva realizada.

Las características disponibles son:

- **hotel:** Hotel (H1 = Resort Hotel o H2 = City Hotel)
- **is_canceled**: Valor que indica si la reserva se canceló (1) o no (0).
- **lead_time**: Número de días transcurridos entre la fecha de entrada de la reserva en el PMS y la fecha de llegada
- **arrival_date_year**: Año de la fecha de llegada
- **arrival_date_month**: Mes de la fecha de llegada
- **arrival_date_week_number**: Número de semana del año de la fecha de llegada
- **arrival_date_day_of_month**: Día de la fecha de llegada
- **stays_in_weekend_nights**: Número de noches de fin de semana (sábado o domingo) que el huésped se alojó o reservó para alojarse en el hotel.
- **stays_in_week_nights**: Número de noches entre semana (de lunes a viernes) que el cliente se alojó o reservó en el hotel.
- **adults**: Número de adultos
- **children**: Número de niños
- **babies**: Número de bebés
- **meal**: Tipo de comida reservada. Las categorías se presentan en paquetes de comidas de hospitalidad estándar: Undefined/SC - sin paquete de comidas; BB - Alojamiento y desayuno; HB - Media pensión (desayuno y otra comida - normalmente cena); FB - Pensión completa (desayuno, almuerzo y cena)
- **country**: País de origen. Las categorías se representan en el formato ISO 3155-3:2013
- **market_segment**: Designación del segmento de mercado. En las categorías, el término TA significa agencias de viaje y TO operadores turísticos.
- **distribution_channel**: Canal de distribución de reservas. En las categorías, el término TA significa agencias de viaje y TO operadores turísticos.
- **is_repeated_guest**: Valor que indica si el nombre de la reserva era de un huésped repetido (1) o no (0)
- **previous_cancellations**: Número de reservas anteriores canceladas por el cliente antes de la reserva actual
- **previous_bookings_not_canceled**: Número de reservas anteriores no canceladas por el cliente antes de la reserva actual

- **reserved_room_type**: Código del tipo de habitación reservada. El código se presenta en lugar de la designación por razones de anonimato.
- **assigned_room_type**: Código del tipo de habitación asignado a la reserva
- **booking_changes**: Número de cambios/enmiendas realizados en la reserva desde el momento en que se introdujo en el PMS
- **deposit_type**: Indicación de si el cliente hizo un depósito para garantizar la reserva. Esta variable puede asumir tres categorías: No Deposit: no se realizó ningún depósito; No Refund: se realizó un depósito por valor del coste total de la estancia; Refundable: se realizó un depósito por valor inferior al coste total de la estancia.
- **agent**: ID de la agencia de viajes que realizó la reserva
- **company**: ID de la empresa/entidad que realizó la reserva o responsable del pago de la reserva. La identificación se presenta en lugar de la designación por razones de anonimato.
- **days_in_waiting_list**: Número de días que la reserva estuvo en lista de espera antes de ser confirmada al cliente
- **customer_type**: Tipo de reserva, asumiendo una de las cuatro categorías: Contract - cuando la reserva tiene asociada una adjudicación u otro tipo de contrato; Group - cuando la reserva está asociada a un grupo; Transient - cuando la reserva no forma parte de un grupo o contrato, y no está asociada a otra reserva transitoria; Transient-party - cuando la reserva es transitoria, pero está asociada al menos a otra reserva transitoria.
- **adr**: Tarifa media diaria definida dividiendo la suma de todas las transacciones de alojamiento por el número total de noches de estancia.
- **required_car_parking_spaces**: Número de plazas de aparcamiento que necesita el cliente
- **total_of_special_requests**: Número de peticiones especiales realizadas por el cliente (por ejemplo, cama doble o piso alto)
- **reservation_status**: Último estado de la reserva, asumiendo una de las tres categorías: Canceled: el cliente ha cancelado la reserva; Check-Out: el cliente se ha registrado pero ya se ha marchado; No-Show: el cliente no se ha registrado y ha informado al hotel del motivo.
- **reservation_status_date**: Fecha en la que se estableció el último estado. Esta variable se puede utilizar junto con ReservationStatus para saber cuándo se cancela la reserva o cuándo abandona el cliente el hotel.

Descripción

1. Importa las librerías necesarias para aplicar preprocesamiento de datos, visualización y creación de un modelo de red neuronal feedforward con Keras y Tensorflow, además de las librerías para realizar regularización y búsqueda de grilla.

   Luego, debes descartar las columnas **index, arrival_date_year, agent, country, company, reservation_status** y **reservation_status_date.** Revisa después si la base

de datos presenta valores ausentes; en el caso que existan deberá eliminar estos registros siempre y cuando la cantidad sea menor al 4% del total. En caso que sea superior debes decidir si quitar la característica o imputar los datos faltantes aplicando alguna estrategia, que debes describir.

2. Realiza un análisis descriptivo y prepara los datos. Para esto:
    a. Selecciona 5 variables que consideres relevantes (que no sean la variable adr) con respecto a cancelar una reserva.
    b. Construye un histograma para la variable adr con reservas canceladas y no canceladas, y comenta el resultado.
    c. Revisa la presencia de outlier para la variable adr. En caso que presente valores extremos indica la cantidad de outliers por arriba, y la cantidad de outlier por abajo. Eliminar sólo los tres valores más extremos en ambos casos, para los demás cambia sus valores por el valor promedio de la variable.
    d. Transforma las variables categóricas en variables dummies, estandariza las variables independientes con media igual a cero y desviación estándar 1 y realiza una división de los datos para entrenamiento y test, este último con un 33% de registros.

3. Implementa dos modelos de red neuronal multicapa, considerando las siguientes características.
    a. Cada modelo debe contar con tres capas ocultas.
    b. El primer modelo debe tener funciones de activación tanh, relu y tanh en las capas ocultas, mientras que el segundo sólo funciones tanh en sus capas ocultas.
    c. Cada modelo debe tener un mínimo de 20 neuronas para cada capa oculta. Use un optimizador **SGD**, con 10 épocas de entrenamiento; cada capa deberá tener una neurona de sesgo. Debes decidir cómo se inicializan los pesos y el sesgo en cada capa oculta y de salida.

    Muestra los resultados en un gráfico que tendrá una curva de accuracy para cada modelo en cada época de entrenamiento. Cada modelo debe lograr un accuracy superior al 75%. Concluye de acuerdo a los resultados.

4. Construye una red neuronal con regularización Dropout aplicando búsqueda de grilla para tres fold. El modelo debe conseguir un accuracy superior al 80%. La búsqueda debe sintonizar los siguientes hiper parámetros:

    a. Cantidad de capas ocultas: de 3 o 4
    b. Método de optimización: Adam o SGD
    c. learning_rate: 0.03 y 0.06
    d. Cantidad de neuronas en las capas ocultas: 20
    e. Funciones de activación en las capas ocultas: tanh

f. Rate para Dropout: 0.001
g. Inicialización de los pesos: [glorot_normal, glorot_uniform]

Debes mostrar los hiper parámetros óptimos encontrados por la búsqueda de grilla. Calcula además las métricas asociadas a la matriz de confusión al aplicar el modelo con los hiper parámetros óptimos al conjunto de test. Muestra la curva ROC para el mejor modelo incluyendo el AUC.

5. Del conjunto original (sin los valores excluidos) escoge al azar cinco observaciones en que "deposit_type_Non Refund" sea cero, y con esto realiza una predicción usando el mejor modelo encontrado por la búsqueda de grilla anterior. Comenta los resultados.

Requerimientos

1. Carga las bibliotecas y datos, y los analiza prepara y selecciona adecuadamente **(2 puntos)**
2. Implementa modelos de red neuronal multicapa con requisitos dados, y analiza sus resultados. **(4 puntos)**
3. Implementa redes neuronales con regularización Dropout, y analiza sus resultados. **(4 puntos)**

 ¡Mucho éxito!

Consideraciones y recomendaciones
Debes entregar tu trabajo en un archivo Jupyter, en el que incluyas el código necesario así como la explicación de tu procedimiento.