## Código de Fuente Utilizado en Preparación de los Datos

Todos los códigos que se han utilizado para estructuración de Sendinblue son construidos con lenguaje de programación “Python” mediante Google Colaboratory, Visual Studio Code y Anaconda, dando el uso de librerías como Pandas, Numpy y Datetime. Para importar estas librerías, bastaba incluir las líneas “*import pandas as pd*”, “*import datetime as dt*” y “*import numpy as np*” en el código. Se ha realizado capturas de pantallas a los códigos de fuente debido que, en el archivo notebook de Python que se trabajó, cuenta con datos sensibles.

A partir de las siguientes ilustraciones, se visualiza la parte inicial de código, donde consiste en leer, unir los archivos de historiales de envíos, eliminar registros duplicados respectos a cuatro campos (E-mail, Id de mensaje, fecha y estado), normalizar los datos de correos (transformar en minúscula y eliminar espacios vacíos en los extremos), combinar con los datos del universo de cliente, priorizando aquel archivo que tenga mayor volumen de registros.

También se eliminan las columnas que no son necesarias, proveniente de la base de datos del cliente, cuyos nombres se encuentran censurados, y por otro lado los estados que no se consideraron para la construcción del dataset. Para el caso de los meses desde junio hasta diciembre, se renombra los nombres de columnas para así concatenar con todos los meses sin necesidad de generar campos nuevos.

Ilustración 1: Extracción, unión y combinación de los datos, mes de enero

*Imagen que contiene Texto

Descripción generada automáticamente*

*Fuente: Elaboración propia*

Ilustración 2: Extracción, unión y combinación de los datos, mes de febrero

*Texto

Descripción generada automáticamente con confianza baja*

*Fuente: Elaboración propia*

Ilustración 3: Extracción, unión y combinación de los datos, mes de marzo

*Imagen que contiene Texto

Descripción generada automáticamente*

*Fuente: Elaboración propia*

Ilustración 4: Extracción, unión y combinación de los datos, mes de abril

*Imagen que contiene Escala de tiempo

Descripción generada automáticamente*

*Fuente: Elaboración propia*

Ilustración 5: Extracción, unión y combinación de los datos, mes de mayo

*Imagen que contiene Escala de tiempo

Descripción generada automáticamente*

*Fuente: Elaboración propia*

Ilustración 6: Extracción, unión y combinación de los datos, mes de junio

*Imagen que contiene Texto

Descripción generada automáticamente*

*Fuente: Elaboración propia*

Ilustración 7: Extracción, unión y combinación de los datos, mes de julio

*Imagen que contiene Texto

Descripción generada automáticamente*

*Fuente: Elaboración propia*

Ilustración 8: Extracción, unión y combinación de los datos, mes de agosto

*Texto

Descripción generada automáticamente con confianza media*

*Fuente: Elaboración propia*

Ilustración 9: Extracción, unión y combinación de los datos, mes de septiembre

*Texto

Descripción generada automáticamente con confianza baja*

*Fuente: Elaboración propia*

Ilustración 10: Extracción, unión y combinación de los datos, mes de octubre

*Imagen que contiene Texto

Descripción generada automáticamente*

*Fuente: Elaboración propia*

Ilustración 11: Extracción, unión y combinación de los datos, mes de noviembre

*Imagen que contiene Texto

Descripción generada automáticamente*

*Fuente: Elaboración propia*

Ilustración 12: Extracción, unión y combinación de los datos, mes de diciembre

*Imagen que contiene Texto

Descripción generada automáticamente*

*Fuente: Elaboración propia*

Una vez hecho la ejecución de todos los códigos anteriores, como se muestra en la ilustración [13], se procede a unir todos los conjuntos de datos de todos los meses, utilizando la función de concatenación de pandas “*concat()*”, renombrar los campos, especialmente en id del mensaje (por los espacios que contiene). Además, en su posterior, se elimina los duplicados restantes y se reinicia el índice (index) del Dataframe, para este último caso es importante ejecutar cada vez que ocurra algún cambio en los registros (al añadir, eliminar o mover filas).

Ilustración 13: Unión de datos de todos los meses y eliminación de duplicados

Texto

Descripción generada automáticamente

*Fuente: Elaboración propia*

Por otro lado, para mantener la integridad de algunos datos sensibles de datos se utilizó función de “*replace()*”, que permite sustituir un valor “Origen” por otro “Reemplazo”. Cabe destacar que, para los datos categórico, como son “Segmento”, “Subsegmento” y “Agrupación de segmento”, son reemplazados por letras alfabetos y números. A continuación, en la ilustración [14], se visualiza la nomenclatura del código de fuente que se utilizó para su aplicación.

Ilustración 14: Nomenclatura del código de función de Pandas, replace

Imagen que contiene Gráfico

Descripción generada automáticamente

*Fuente: Elaboración propia*

Después, como se visualiza en la ilustración [15], se ejecutó el código que calcule la cantidad de estados que contiene cada “*Envio\_id*”, utilizando la función crosstab(). Luego, para los nuevos campos, se incorpora al dataset en construcción, utilizando la función merge(), y se renombra las columnas.

Ilustración 15: Cálculo de cantidad de estados, entregado, primera apertura y abierto

Texto

Descripción generada automáticamente

*Fuente: Elaboración propia*

Continuando con la ejecución del código de fuente, a partir de ilustración [16], se procede a aplicar los criterios mencionado en capítulo 5 (preparación de los datos), en cambiar los valores de primera apertura y sumar con la cantidad de abiertos por cada correo.

Ilustración 16: Aplicación de criterios, y suma entre primera apertura y abierto

Imagen que contiene Diagrama

Descripción generada automáticamente

*Fuente: Elaboración propia*

Una vez realizado lo anterior, se procesa a identificar y generar campo referente a las fechas de envío por cada registro del dataset, en base a “*Envio\_id*”. Luego, se eliminan los campos que son “Estados”, “Fecha” y “Cantidad entregado”, y también se suprime todas las filas que se repiten de acuerdo dicho campo id de envío, sin considerar a otras columnas. Como resultado, se ha reducido mucho el volumen de los datos, manteniendo toda la información expuesta en el dataset. Todo lo anterior se visualiza en ilustración [17].

Ilustración 17: Identificación de fecha de envíos y disminución de volumen de datos

Texto

Descripción generada automáticamente

*Fuente: Elaboración propia*

Después, se genera nuevos campos extrayendo datos de fecha, cuyos valores corresponden a el día, mes y horario. Asimismo, se procede con la transformación de formato del horario desde 12 a 24 horas, para así extraer la hora que se envió y recibió una encuesta. Todo lo anterior se puede visualizar en la ilustración [18].

Ilustración 18: Extracción de día, mes y horas de envíos

Texto

Descripción generada automáticamente

*Fuente: Elaboración propia*

Una vez obtenido la hora de envío, a partir de ilustración [19], se procede a aplicar los criterios mencionado en capítulo 5 (preparación de los datos), en cambiar los valores de horas de envío, que se encuentren fuera del rango del horario entre 8:00 a 21:00, y cambiar el formato de fecha por “AÑO-MES-DÍA” para un correcto procesamiento utilizando la librería Datetime, eliminando otro formato de la dicha fecha.

Ilustración 19: Aplicación de criterios a fecha y hora de envío

Texto

Descripción generada automáticamente

*Fuente: Elaboración propia*

Continuando con otra ejecución de código de fuente elaborado, tras cambiar el formato dicho anteriormente, se cambia el tipo de dato al campo “Fecha” (de envío) por Datetime, y luego se obtiene el día de la semana (Lunes a Domingo) lo cual se ha efectuado el envío por correo electrónico. Todo lo anterior se visualiza en la ilustración [20].

Ilustración 20: Obtención de día de la semana del envío

Texto

Descripción generada automáticamente con confianza media

*Fuente: Elaboración propia*

Al comprobar que todos los datos estén correctos, verificando a través de calendario, se confecciona y ejecuta el código de fuente donde se determina la fecha de limite “*Fecha\_Termino*”, donde este se encuentra en la carpeta de códigos de fuente, dentro de un archivo denominado como “*fechatermino.py*”. Una vez obtenido la columna de fecha de término, se elabora y ejecuta el código para estimar la contactabilidad de cada registro, donde esto puede encontrarse en la misma carpeta, cuyo archivo se llama “*Contactabilidad.ipynb*”.

Por último, al terminar todos los procesos de construcción del dataset, se procede a anonimizar los datos de E-mail y transformar como campo entero. También, para el caso de “*Envio\_id*”, se realizó cambios convirtiendo en números enteros auto incrementables, para este caso existe dos opciones, utilizar la función de pandas “*Factorize*” al campo existente, o generar columna nueva utilizando el valor de index de Dataframe. Para este último caso se utilizó la segunda opción, ya que el campo “*Envio\_id*” se ha eliminado antes de entrar la ejecución para obtener fechas de límites, con el fin de reducir volumen de datos y agilizar el proceso.

Ilustración 21: Anonimización de datos “E-mail” y modificación en "Envío\_Id "

Imagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamente

*Fuente: Elaboración propia*