Data Cleaning

A modo de sugerencia general, para interpretar qué ocurre con cada notebook se recomienda correr las mismas ejecutando celda por celda.

Datos de Contactabilidad

Data Cleaning - Mails

Dataset Recibido

1. Lectura del dataset recibido.
2. Se dropean filas y columnas que no tienen ningún valor.
3. Se obtienen datos de mails de las columnas correspondientes.
4. Se agrupan todas las columnas que tienen datos de mails en una sola (melt).
5. Se asignan propietarios de los datos en función de cada campo (propio, empleador, familiar).
6. Se agregan las columnas PROVEEDOR_DATO y TIPO.
7. Generación de ciertas reglas para identificar si el SUBTIPO es Laboral o Personal.
8. Se construye un score base (ya que luego es actualizado por una vista).
9. Se agregan campos adicionales (Fecha de ingreso, verificado, operador verificación).

Dataset Enviado

10. Lectura del .csv enviado. Siendo más estrictos, no sería el .csv “enviado” sino aquel que se extrae y contiene IDMOROSO, NOMBRE, ESTADO, FILTROA e IDCLIENTE (el .csv enviado sólo tiene IDMOROSO o Documento y NOMBRE). Refiere al .csv de casos que extraemos de Active para enviar a enriquecer.
11. Se crea una columna NRO_DOC, que contendrá los DNIs limpios para hacer la unión con el dataframe recibido.
    a. En caso que se hayan enviado a enriquecer casos de AMEX, se deberán descomentar las líneas correspondientes que indican que si el IDCliente empieza con “AMEX”, el número de documento está ubicado en el campo DATOADIC en lugar de IDMOROSO.
12. DNI Cleaning: se generan procesos de limpieza de los IDMorosos/DATOADIC para extraer el número de DNI y el tipo de Documento (este punto podría llegar a optimizarse redefiniendo ciertas funciones).
13. Se unen ambos dataframes ya limpios (dataset recibido + dataset enviado).
14. Se renombran y preparan las columnas para adecuarse a la tabla donde será insertada la data.
15. Se cambian ciertos tipos de datos (por ejemplo, CUIT pasa de ser un int a un str por cuestiones de notación; al ser números demasiado grandes, terminan representados en notación científica, y al insertar quedan de esta forma).
16. Se revisan los duplicados que puedan existir.

    a. Por ejemplo, el proveedor nos puede indicar que el mail test@test.com es el mail del moroso, pero también nos indica que es el mail de algún familiar; como no sabemos a quién atribuirle, se generan ciertas reglas.
    b. Tenemos dos opciones:
        i. Los casos son pocos y los podemos dropear manualmente. En este caso, la regla general es que si el dato es otorgado como Propio y como Ajeno, queda como Propio y se dropean las filas que indican que es Ajeno.
        ii. Los casos son muchos y debemos pasar al notebook de Duplicados. Las reglas están definidas dentro de este notebook.
17. Se exporta el archivo, cambiando simplemente la fecha que está al final del nombre.

Data Cleaning - Teléfonos

Dataset Recibido

1. Lectura del dataset recibido.
2. Se dropean filas y columnas que no tienen ningún valor.
3. Se reduce el dataframe a aquellas columnas con datos de teléfonos.
4. Se agrupan todas las columnas que tienen datos de teléfonos en una sola (melt).
5. Se asignan propietarios de los datos en función de cada campo (propio, empleador, familiar, ajeno).
6. Se agregan las columnas PROVEEDOR_DATO y TIPO, en función de los campos correspondientes (principalmente, dividimos entre TEL.CELULAR y TEL.FIJO).
7. Se identifica el SUBTIPO (si es Laboral o Personal).
8. Se construye un score base (ya que luego es actualizado por una vista).
9. Se agregan campos adicionales (Fecha de ingreso, verificado, operador verificación).

Dataset Enviado

10. Lectura del .csv enviado. Siendo más estrictos, no sería el .csv “enviado” sino aquel que se extrae y contiene IDMOROSO, NOMBRE, ESTADO, FILTROA e IDCLIENTE (el .csv enviado sólo tiene IDMOROSO o Documento y NOMBRE). Refiere al .csv de casos que extraemos de Active para enviar a enriquecer.
11. Se crea una columna NRO_DOC, que contendrá los DNIs limpios para hacer la unión con el dataframe recibido.
    a. En caso que se hayan enviado a enriquecer casos de AMEX, se deberán descomentar las líneas correspondientes que indican que si el IDCliente empieza con “AMEX”, el número de documento está ubicado en el campo DATOADIC en lugar de IDMOROSO.
12. DNI Cleaning: se generan procesos de limpieza de los IDMorosos/DATOADIC para extraer el número de DNI y el tipo de Documento (este punto podría llegar a optimizarse redefiniendo ciertas funciones).
13. Se unen ambos dataframes ya limpios (dataset recibido + dataset enviado).

14. Se renombran y preparan las columnas para adecuarse a la tabla donde será insertada la data.
15. Se cambian ciertos tipos de datos (por ejemplo, CUIT pasa de ser un int a un str por cuestiones de notación; al ser números demasiado grandes, terminan representados en notación científica, y al insertar quedan de esta forma).
16. Se revisan los duplicados que puedan existir.
    a. Por ejemplo, el proveedor nos puede indicar que el mail test@test.com es el mail del moroso, pero también nos indica que es el mail de algún familiar; como no sabemos a quién atribuirle, se generan ciertas reglas.
    b. Tenemos dos opciones:
        i. Los casos son pocos y los podemos dropear manualmente. En este caso, la regla general es que si el dato es otorgado como Propio y como Ajeno, queda como Propio y se dropean las filas que indican que es Ajeno.
        ii. Los casos son muchos y debemos pasar al notebook de Duplicados. Las reglas están definidas dentro de este notebook. En general, para el dataset de teléfonos, esta es la opción que ocurre con mayor frecuencia. (*)
17. Se exporta el archivo, cambiando simplemente la fecha que está al final del nombre.

(*) Duplicados

1. Lee el dataset completo que exporta Data Cleaning - Teléfonos o Data Cleaning - Mails.
2. Analiza duplicados en el subset IDMOROSO, DATO, PROVEEDOR_DATO.
3. Se asignan puntajes a cada posible dato en las categorías Tipo, Subtipo, Propietario_dato y Verificado.
4. Se suman los puntajes para cada dato.
5. Se agrupan por IDMOROSO, DATO, PROVEEDOR_DATO. Para elegir con cuál de los duplicados nos quedamos, se selecciona aquel que obtiene el máximo puntaje.
6. Se analiza si existen duplicados en dato pero con diferente fecha de ingreso (es menos probable, pero en alguna ocasión sucedió).
7. Se unen los dataframes que se fueron generando.
8. Se hace un pequeño control de calidad, analizando si continuamos teniendo duplicados en el subset IDMOROSO, DATO, PROVEEDOR_DATO.
9. Se exporta el .csv que insertará luego en DAM_CONTACTABILIDAD, cambiando la fecha que está al final del nombre.

Datos Patrimoniales

Data Cleaning - Inmuebles

1. Lectura del dataset recibido.
2. Se dropean filas y columnas que no tienen ningún valor.
3. Se obtienen datos de inmuebles de las columnas correspondientes (Propiedades). En este caso, sólo podemos obtener información de la cantidad de inmuebles registrados.

4. Se renombran las columnas y se dropean aquellas filas que tengan 0 en la columna Propiedades, ya que no nos aporta nada.
5. Si el moroso tiene 2 propiedades, queremos que aparezcan 2 filas (una por propiedad), por lo que se repite el index.
6. Se asignan las columnas propietario, porcentaje y fecha.
7. Se generan las columnas domicilio, localidad, país (no tenemos esos datos, por lo que quedan en “No Disponible”).
8. Lectura del .csv enviado. Siendo más estrictos, no sería el .csv “enviado” sino aquel que se extrae y contiene IDMOROSO, NOMBRE, ESTADO, FILTROA e IDCLIENTE (el .csv enviado sólo tiene IDMOROSO o Documento y NOMBRE). Refiere al .csv de casos que extraemos de Active para enviar a enriquecer.
9. Se crea una columna NRO_DOC, que contendrá los DNIs limpios para hacer la unión con el dataframe recibido.
    a. En caso que se hayan enviado a enriquecer casos de AMEX, se deberán descomentar las líneas correspondientes que indican que si el IDCliente empieza con “AMEX”, el número de documento está ubicado en el campo DATOADIC en lugar de IDMOROSO.
10. DNI Cleaning: se generan procesos de limpieza de los IDMorosos/DATOADIC para extraer el número de DNI y el tipo de Documento (este punto podría llegar a optimizarse redefiniendo ciertas funciones).
11. Se unen ambos dataframes ya limpios (dataset recibido + dataset enviado).
12. Se renombran y preparan las columnas para adecuarse a la tabla donde será insertada la data.
13. Se exporta la data en formato .csv, cambiando la fecha al final del nombre.

Data Cleaning - Vehiculos

1. Lectura del dataset recibido.
2. Se dropean filas y columnas que no tienen ningún valor.
3. Se obtienen datos de vehículos de las columnas correspondientes. En este caso, tenemos la cantidad total de patentados de los últimos 20 años para cada moroso. Pero a nivel de detalle, sólo tenemos marca, modelo, año, etc. del último vehículo patentado. Es por esto que vamos a mirar solamente las columnas Dominio, Fecha Automotor, Fecha de Compra, Marca y Modelo.
4. Se genera una concatenación de Marca y Modelo, ya que en la tabla final tenemos un único campo para insertar estos dos datos. Esto se une y forma la columna Fabricante
5. Se renombran las columnas y se dropean aquellas filas que tengan 0 en la columna Propiedades, ya que no nos aporta nada.
6. Se asignan las columnas propietario, porcentaje y fecha.
7. Volviendo a la columna Fabricante, se observa que cuando no tiene modelo o marca, el proveedor escribe “No Consta”. Además, en algunos casos, al concatenar se repite el nombre de la marca (por ejemplo, dentro del campo Marca podemos tener Ford y en Modelo podemos tener Ford F100 en algunos casos). Para esto, se generan ciertas reglas y se utilizan expresiones regulares.
8. Lectura del .csv enviado. Siendo más estrictos, no sería el .csv “enviado” sino aquel que se extrae y contiene IDMOROSO, NOMBRE, ESTADO, FILTROA e IDCLIENTE

(el .csv enviado sólo tiene IDMOROSO o Documento y NOMBRE). Refiere al .csv de casos que extraemos de Active para enviar a enriquecer.

9. Se crea una columna NRO_DOC, que contendrá los DNIs limpios para hacer la unión con el dataframe recibido.
    a. En caso que se hayan enviado a enriquecer casos de AMEX, se deberán descomentar las líneas correspondientes que indican que si el IDCliente empieza con “AMEX”, el número de documento está ubicado en el campo DATOADIC en lugar de IDMOROSO.
10. DNI Cleaning: se generan procesos de limpieza de los IDMorosos/DATOADIC para extraer el número de DNI y el tipo de Documento (este punto podría llegar a optimizarse redefiniendo ciertas funciones).
11. Se unen ambos dataframes ya limpios (dataset recibido + dataset enviado).
12. Se renombran y preparan las columnas para adecuarse a la tabla donde será insertada la data.
13. Se solucionan algunos problemas referidos a cómo presenta las fechas el proveedor (fecha de compra y año del automotor):
    a. En algunos casos, la fecha tiene el formato correcto.
    b. En otros casos, la fecha tiene formato int (42668 por ejemplo), por lo que hay que generar una conversión al formato correcto. Para esto se definen ciertas funciones y reglas.
14. Se exporta la data en formato .csv, cambiando la fecha al final del nombre.

IMPORTANTE: Para los siguientes pipelines (Deudas - Scoring) se deberá utilizar una copia del dataset recibido. Ver documento Data Cleaning - General. Principalmente, la idea es desmergear ciertas celdas del header y renombrar otras.

Data Cleaning - Deudas (no utilizable para Insiders por el momento)

1. Lectura del dataset recibido.
2. Se dropean filas y columnas que no tienen ningún valor.
3. Se obtienen datos de deuda de las columnas correspondientes. Dentro de la columna Comportamiento en banco Central del Titular:
    a. Cant. Sit 1, 2, 3, 4 y 5.
        i. Es necesario que los valores que levantan en las celdas Cant. Sit 1,2,3,4,5 sean int, ya que luego se utilizan posteriormente como índices para un loop. Por lo que si aparecen como floats, se deberán convertir a int. En el notebook esto está implementado.
    b. Entidad 1, 2, 3, 4.
    c. Deuda 1, 2, 3, 4.
4. Se definen ciertas funciones para iterar por la cantidad de situaciones que tenga cada moroso.
    a. En el proveedor PyP, el máximo es 4, y se asignan de forma ordinal: si el moroso está en situación 1 con tres entidades, y situacion 2 con una entidad, a las Entidades Nº1, 2 y 3 les corresponden situación 1, mientras que a la Entidad Nº4 le corresponde situación 2.

    b. Estas funciones iteran y asignan a cada entidad la situación y deuda correspondiente.
    c. En el caso del proveedor Insiders ocurre un problema con la cantidad de situaciones y no se puede asignar una deuda a una entidad particular (ver Problema al final de esta sección).
5. Se une el dataframe con las deudas procesadas, con las columnas CUIT y DNI.
6. Se asignan las columnas propietario, fuente y fecha.
7. Lectura del .csv enviado. Siendo más estrictos, no sería el .csv “enviado” sino aquel que se extrae y contiene IDMOROSO, NOMBRE, ESTADO, FILTROA e IDCLIENTE (el .csv enviado sólo tiene IDMOROSO o Documento y NOMBRE). Refiere al .csv de casos que extraemos de Active para enviar a enriquecer.
8. Se crea una columna NRO_DOC, que contendrá los DNIs limpios para hacer la unión con el dataframe recibido.
    a. En caso que se hayan enviado a enriquecer casos de AMEX, se deberán descomentar las líneas correspondientes que indican que si el IDCliente empieza con “AMEX”, el número de documento está ubicado en el campo DATOADIC en lugar de IDMOROSO.
9. DNI Cleaning: se generan procesos de limpieza de los IDMorosos/DATOADIC para extraer el número de DNI y el tipo de Documento (este punto podría llegar a optimizarse redefiniendo ciertas funciones).
10. Se unen ambos dataframes ya limpios (dataset recibido + dataset enviado).
11. Se renombran y preparan las columnas para adecuarse a la tabla donde será insertada la data.
12. Se revisan las equivalencias de los índices antes de dropear uno.
13. Se mapean las situaciones a sus valores numéricos correspondientes.
14. Se agrega la columna período, y se multiplica por 1000 el valor del monto de deuda en cada caso (el proveedor nos otorga por ejemplo 34, significando una deuda de 34000).
15. Se exporta la data en formato .csv, cambiando la fecha al final del nombre.

Problema: La cantidad total de situaciones en algunas filas excede el total máximo de 4, por lo que no podemos identificar correctamente a qué corresponde cada una. Por ejemplo, el proveedor comenta que el moroso se encuentra operando con 7 entidades. Pero al momento de detallar las entidades, solamente tenemos el detalle de 4 como máximo (las columnas Entidad 1,2,3,4). Por lo tanto, no podemos asegurar qué situación corresponde a cada entidad. Esto es algo que el proveedor tampoco nos lo puede asegurar.

Data Cleaning - Scoring

1. Lectura del dataset recibido.
2. Se dropean filas y columnas que no tienen ningún valor.
3. Se segmentan los campos que serán tenidos en cuenta al momento de generar un scoring:
    a. Vehículos patentados entre 2015-2020 / Vehículos patentados antes del 2015.
    b. Más de 2 propiedades / 1 Propiedad / Sin propiedades.
    c. Situación frente al BCRA 1 o 2.

    d. Rangos de ingresos (autónomo, dependencia, jubilación).
4. Se generan columnas booleanas para identificar cada caso (si tiene más de una propiedad, si no tiene propiedades, si tiene vehículos patentados entre 2015 y 2020, etc.).
5. Lectura del .csv enviado. Siendo más estrictos, no sería el .csv “enviado” sino aquel que se extrae y contiene IDMOROSO, NOMBRE, ESTADO, FILTROA e IDCLIENTE (el .csv enviado sólo tiene IDMOROSO o Documento y NOMBRE). Refiere al .csv de casos que extraemos de Active para enviar a enriquecer.
6. Se crea una columna NRO_DOC, que contendrá los DNIs limpios para hacer la unión con el dataframe recibido.
    a. En caso que se hayan enviado a enriquecer casos de AMEX, se deberán descomentar las líneas correspondientes que indican que si el IDCliente empieza con “AMEX”, el número de documento está ubicado en el campo DATOADIC en lugar de IDMOROSO.
7. DNI Cleaning: se generan procesos de limpieza de los IDMorosos/DATOADIC para extraer el número de DNI y el tipo de Documento (este punto podría llegar a optimizarse redefiniendo ciertas funciones).
8. Se unen ambos dataframes ya limpios (dataset recibido + dataset enviado).
9. Se seleccionan las columnas del dataframe (aquellas booleanas que generamos y las que queremos puntuar).
10. Se generan los scorings correspondientes para cada categoría.
11. Se suman los scores de cada categoría para generar finalmente un score patrimonial por moroso.
12. Se mapean los scorings a las 4 categorías definidas (AA, A, B, C). De esta forma, podemos segmentar morosos en 4 grupos diferentes en función del score patrimonial que reciben.
13. Se ajustan los datos al formato de tabla deseado.
14. Se generan las columnas adicionales.
15. Se exporta la data en formato .csv, cambiando la fecha al final del nombre.

DAM_LOG

1. Lectura del .csv enviado. Siendo más estrictos, no sería el .csv “enviado” sino aquel que se extrae y contiene IDMOROSO, NOMBRE, ESTADO, FILTROA e IDCLIENTE (el .csv enviado sólo tiene IDMOROSO o Documento y NOMBRE). Refiere al .csv de casos que extraemos de Active para enviar a enriquecer.
    a. En el caso que el envío incluya datos de AMEX, descomentar la cuarta línea que indica que nos quedamos además con el campo DATOADIC.
2. Se agregan las fechas de envío e inserción (fecha de cambio de estados es algo que ya no se utiliza, por lo que siempre se agrega un equivalente a la fecha de inserción).
3. Se engloban todos los estados en las categorías correspondientes (Referencia: ver documento de Glosario de Estados).
4. Lectura del .csv recibido por parte del proveedor. La idea es cruzar contra los IDs y ver cuáles están enriquecidos.
5. Reducimos el .csv recibido a CUIL y DNI, para hacer el join.

    a. En caso que se hayan enviado a enriquecer casos de AMEX, se deberán descomentar las líneas correspondientes que indican que si el IDCliente empieza con “AMEX”, el número de documento está ubicado en el campo DATOADIC en lugar de IDMOROSO.
6. DNI Cleaning: se generan procesos de limpieza de los IDMorosos/DATOADIC para extraer el número de DNI y el tipo de Documento (este punto podría llegar a optimizarse redefiniendo ciertas funciones).
7. Se unen ambos dataframes ya limpios (dataset recibido + dataset enviado).
8. Se seleccionan las columnas del dataframe (aquellas booleanas que generamos y las que queremos puntuar).
9. Se asigna un booleano a los casos que devuelve la unión, ya que son los que se enriquecieron.
10. Se exporta la data en formato .csv, cambiando la fecha al final del nombre.