PROYECTO FINAL: BIG DATA

Industria 4.0 2020

Instituto Tecnológico de Nuevo Laredo

Michelle Fernando Ramos Martínez
José Abraham González Andrade
Julieta Hernández Arias
José Emanuel Olvera Campean

Docente:
José Antonio Espino López

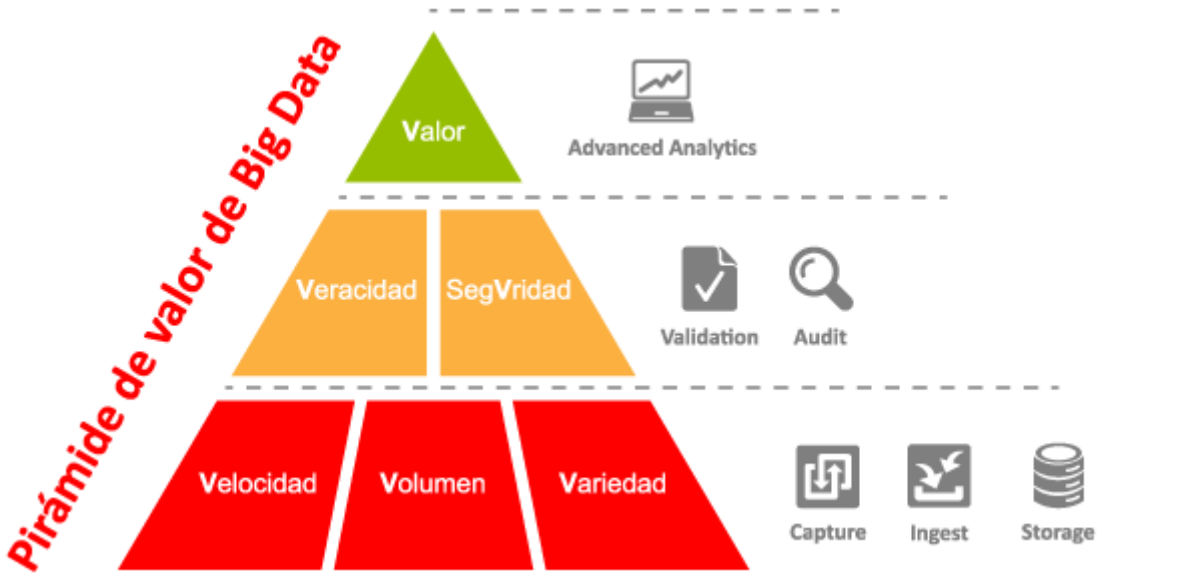
Limpieza de Datos (Data cleansing)

La limpieza de datos, data cleansing o scrubbing es un proceso necesario para asegurar la calidad de los datos que se emplearán para analytics. Este paso es fundamental para minimizar el riesgo que supondría el basar la toma de decisiones en información poco precisa, errónea o incompleta.

El data cleansing se ocupa de solucionar problemas de calidad de datos a dos niveles:

- Problemas relacionados datos procedentes de una única fuente: a este nivel se encuentran las cuestiones relacionadas con la falta de integridad de las restricciones o la precariedad del diseño del esquema; que afectarán a su vez a la unicidad del dato y a su integridad referencial, principalmente. Aunque, en un sentido más práctico en este apartado también podrían englobarse las cuestiones relacionadas con la entrada de datos, en cuanto a redundancias o valores contradictorios, entre otros.

- Problemas relacionados con datos provenientes de diversas fuentes de origen: por norma general surgen como resultado de la heterogeneidad de los modelos de datos y esquemas, que pueden causar conflictos estructurales; aunque, a nivel de instancia, se relacionan con las duplicidades, contradicciones e inconsistencias de los datos.



LAS FASES DEL DATA CLEANSING

El objetivo final de cualquier acción de data cleansing es mejorar la confianza de la organización
en sus datos. Para llevar a cabo una acción de limpieza de datos exhaustiva es necesario seguir las siguientes fases:

1. Análisis de datos: su misión es determinar qué tipo de errores e inconsistencias deben ser eliminados. además de una inspección manual de las muestras de datos, es necesaria la automatización, en otras palabras, la incorporación de programas que actúen sobre los metadatos para detectar problemas de calidad de datos que afecten a sus propiedades.

2. Definición del flujo de transformación y reglas de mapeo: dependiendo del número de fuentes de origen de datos, su heterogeneidad y la previsión de problemas de calidad de los datos, será necesario ejecutar más o menos pasos en la etapa de transformación y adecuación. Lo más adecuado es plantear una acción a dos niveles, una en un estadio temprano, que corrija los problemas relacionados con datos procedentes de una única fuente y los prepare para una buena integración; y otra, que intervenga de forma posterior, tratando los problemas de datos procedentes de una diversidad de fuentes. Para mejorar el control sobre estos procedimientos conviene definir los procesos ETL encuadrándolos en el marco de trabajo concreto.

3. Verificación: el nivel de adecuación y la efectividad de una acción de transformación debe siempre ser testado y evaluado; uno de los principios del data cleansing. Por norma general, esta validación se aplica a través de múltiples iteraciones de los pasos

de análisis, diseño y verificación; ya que algunos errores sólo se ponen de evidencia tras aplicarse a los datos un número determinado de transformaciones.

4. Transformación: consiste en proceder a ejecutar el flujo ETL para cargar y refrescar el data warehouse, o durante la respuesta a consultas, en los casos de multiplicidad de fuentes de origen.

5. Reflujo de datos limpios: una vez se han eliminado los errores de calidad, los datos "limpios" deben reemplazar a los que no lo están en las fuentes originales, para que las aplicaciones de legado puedan beneficiarse también de ellos, evitando necesitar de la aplicación de acciones de data cleansing en el futuro.

Despues del proceso anterior los datos tendrán una mejor calidad lo que implica que el valor de los datos será realmente de utilidad

Data Grindding (Operaciones con los Datos)

Esta parte hace referencia a la herramientas que nos permitirán ejecutar instrucciones que tienen como fin ayudar a tratar los datos para que nos generen información que se tendrá que analizar

PYTHON, HERRAMIENTA EXTENDIDA PARA EL ANÁLISIS DE DATOS

Asentada la base del Análisis de los Datos, ahora tocaba ver cómo se hacía. Y en esta ocasión se aborda usando el lenguaje de programación Python, lenguaje interpretado creado en 1989 por Guido Van Rossum

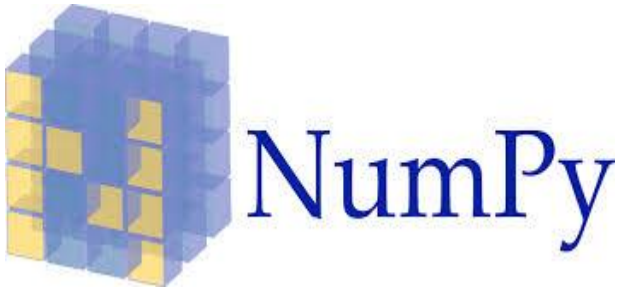
Lo primero que se habló es el porqué de su popularidad, destacándose que tenía una comunidad que lo soportaba muy grande, que está esponsorizado por Google y Facebook entre otros. Y, además, que tenía Big Data, 140000 librerías y que además era eficiente, fiable y abierto

LAS LIBRERÍAS IMPRESCINDIBLES DE PYTHON

Y sin más, se pasó a describir las librerías más importantes para el análisis de datos: numpy, pandas y matplotlib.

Numpy es la base, la librería para los cálculos rápidos, usando la vectorización, o lo que es lo mismo, operaciones entre vectores evitando usar bucles “for”. Y ahí se aprovechó para recordarnos el evitar usar bucles “for” cuando se manejan grandes volúmenes de datos.

El hecho de que la librería numpy esté pensada para manejar grandes volúmenes de datos, esté realizada en C, y tenga un esquema optimizado de memoria, independiente del resto de objetos de Python, es la que la hacen tan importante.



A continuación, se pasó a hablar de la librería más importante del proceso de Ciencia de Datos, pandas, que usando como base la anterior librería con potencial de cálculos muy rápidos, une la posibilidad de manipular datos estructurados (Dataframes y Series) con funciones muy parecidas a las que se usan en las bases de datos (SQL).

LA VISUALIZACIÓN

Pero parece que no es suficiente con estas dos librerías, y se nos insistía bastante, la visualización es muy importante en el proceso que se explicaba al principio de la presentación. En ese descubrimiento de lo que no creemos que está en los datos, como decía Tukey, aflora cuando hacemos una buena visualización.

Matplotlib es la librería por excelencia de visualización de gráficas en 2D, con la que se puede visualizar prácticamente todo lo que se nos ocurra. Para ayudarnos a elegir cómo visualizar, una recomendación era que se acudiera a la galería de ejemplos que contiene tantos, que seguro nos encauza para crear nuestra gráfica.

Finalmente, estas son las plataformas que emplearemos para usar Python: Anaconda como “distribución del software”, Jupyter como aplicación web para poder compartir código y contar historias sobre los datos, sin olvidar un buen editor Notepad++

Análisis

El análisis de 'grandes datos' es el proceso de examinar grandes cantidades de datos de una variedad de tipos (big data) para descubrir patrones ocultos, correlaciones desconocidas y otra información útil.

El objetivo principal del análisis de datos grandes es ayudar a las empresas a tomar mejores decisiones de negocios al permitir a los científicos y otros usuarios de datos analizar grandes volúmenes de datos transaccionales, así como otras fuentes de datos que puedan haber quedado sin explotar por la inteligencia de negocio convencional (BI) programas.

Estas fuentes de datos pueden incluir registros del servidor web y datos de seguimiento de clics en internet, informes de actividades sociales, medios de comunicación, teléfonos móviles registros detallados de llamadas y la información captada por los sensores.

Algunas personas asocian exclusivamente grandes datos y análisis de grandes volúmenes de datos con datos no estructurados de ese tipo, pero consultoras como Gartner y Forrester Research Inc. también consideran las transacciones y otros datos estructurados como formas válidas de datos grandes

Introducción

En el proyecto utilizamos 1 escenario para aplicar los apartados anteriores. El escenario es el de un Taller de una línea Transportista, este tenemos 3 fuentes de datos los cuales serán con los que trabajaremos y aplicaremos una serie de instrucciones para la manipulación y representación de los datos

Taller de Línea Transportista

En un taller de una línea Transportista se generan una gran cantidad de datos debido a los mantenimientos que se hacen, dependiendo el tamaño de la línea transportista son los mantenimientos que se hacen.
Los datos que genera la unidad como kilometraje niveles de aceite, etc. Son también una fuente de datos a considerar para la obtención de datos

En este proyecto tenemos 3 bases de datos que son Despacho de Aceite que es uno de los insumos que más se aplican a las unidades y uno de los que más se quiere controlar su aplicación, Mantenimiento a Unidades que registra lo que se le hace a cada unidad cuando se le realiza un mantenimiento y el de Despacho se registra el Diésel que se le proporciona a cada unidad este siendo uno de los gastos más importantes y por esta razón se requiere minimizar la aplicación a un valor suficiente para los movimientos de la unidad.

Base de Datos de Mantenimiento (Muestra de campos y registros).

	Id	unidad	comentarios	fecha	fechareg	userreg	fechamod	usermod	IdModulo
0	1	1031	RELLENO DE ACEITE	05/05/2017	10:02.2	f490eddd-061c-4b64-bfd3-26ef469b9263	23:42.9	f490eddd-061c-4b64-bfd3-26ef469b9263	CA-09:49:14-1031
1	2	1061	CAMBIO DE ACEITE Y FILTROS	05/05/2017	27:33.1	f490eddd-061c-4b64-bfd3-26ef469b9263	23:50.0	f490eddd-061c-4b64-bfd3-26ef469b9263	CA-09:34:03-1061
2	3	1033	SE PUSIERON 2 SOQUETERAS	08/05/2017	02:28.7	ef281d7a-57db-428e-a186-f1c54d48042c	26:20.1	f490eddd-061c-4b64-bfd3-26ef469b9263	00-00:00:00-000
3	4	1009	SE HIZO UNA TA LACHA POSICIàN 7	08/05/2017	12:03.6	ef281d7a-57db-428e-a186-f1c54d48042c	23:58.4	f490eddd-061c-4b64-bfd3-26ef469b9263	00-00:00:00-000
4	5	1060	SE LE CAMBIO UN PLAFON DE STOP	08/05/2017	13:16.5	ef281d7a-57db-428e-a186-f1c54d48042c	24:01.8	f490eddd-061c-4b64-bfd3-26ef469b9263	00-00:00:00-000

Base de Datos de Despacho de Aceite (Muestra de campos y registros).

	unidad	linicialaceite	lfinalaceite	comentarios	cantidadlitros	fecha	preciol	Costo	Id
0	1032	2830	2834	CAMBIO DE ACEITE	4	18/02/2017	43.3	173.2	1
1	1078	2834	2858	CAMBIO DE ACEITE	24	18/02/2017	43.3	1039.2	2
2	136	2858	2862	CAMBIO DE ACEITE	4	18/02/2017	43.3	173.2	3
3	138	2862	2866	CAMBIO DE ACEITE	4	20/02/2017	43.3	173.2	4
4	1089	2866	2875	CAMBIO DE ACEITE Y FILTROS	9	20/02/2017	43.3	389.7	7

Base de Datos de Despacho de Diesel (Muestra de campos y registros).

	id	noeconomico	KMActual	DespachoLitros	FechaDespacho	fechareg	userreg	fechamod	usermod	KMAnterior	Operador	Rendimiento	Precio
0	1	1007	674258	300.0	12/04/2017	00:00.0	f490eddd-061c-4b64-bfd3-26ef469b9263	00:00.0	f490eddd-061c-4b64-bfd3-26ef469b9263	661020	NaN	0.0	NaN
1	2	1008	1380120	300.0	16/03/2017	00:00.0	f490eddd-061c-4b64-bfd3-26ef469b9263	00:00.0	f490eddd-061c-4b64-bfd3-26ef469b9263	1379910	NaN	0.0	NaN
2	3	1009	460325	159.0	24/03/2017	00:00.0	f490eddd-061c-4b64-bfd3-26ef469b9263	00:00.0	f490eddd-061c-4b64-bfd3-26ef469b9263	456858	NaN	0.0	NaN
3	4	1010	920375	236.0	10/04/2017	00:00.0	f490eddd-061c-4b64-bfd3-26ef469b9263	00:00.0	f490eddd-061c-4b64-bfd3-26ef469b9263	920328	NaN	0.0	NaN
4	5	1011	218724	215.0	05/04/2017	00:00.0	f490eddd-061c-4b64-bfd3-26ef469b9263	00:00.0	f490eddd-061c-4b64-bfd3-26ef469b9263	215419	NaN	0.0	NaN

Estas 3 bases de datos son las que utilizamos para trabajar, siendo las acciones realizadas las que mencionaremos a continuación:

- **Limpieza de Datos**

Durante la limpieza de datos, nos concentramos en tareas que nos permitan corregir detalles existentes en las bases de datos como datos inapropiados o nulos (Ejemplos de estos se pueden ver en los primeros registros de la base de datos de despacho de diesel, donde en los primeros registros podemos observar que cuenta con datos nulos en el campo operador). Para ello, definimos las siguientes funciones para utilizarlas posteriormente en cada una de las bases de datos:

Metodos para la limpieza o llenado

In [91]:
```
#Eliminar filas que contenga datos inapropiados en lugar respectivos
def EliminarCampos(df,var,dato):
    df = df.drop(df[var == dato].index)
    return df
```

In [92]:
```
#Rellenar valores en las filas que sean NaN
def RellenarCamposNaN(df,var,cambio):
    df[var] = df[var].fillna(cambio)
    return df
```

En la siguiente imagen podemos observar el resultado de la aplicación de la función “RellenarCamposNaN” donde sustituimos los datos nulos por “Desconocido” en el caso de “operador” y “usermod” y con ceros en los campos “fecha” y “precio”.

```
In [103]: RellenarCamposNaN(RegDespacho,"fechamod","00:00.0")

          RellenarCamposNaN(RegDespacho,"usermod","Desconocido")

          RellenarCamposNaN(RegDespacho,"Operador","Desconocido")

          RellenarCamposNaN(RegDespacho,"Precio",0.0)
```

Out[103]:

	id	noeconomico	KMActual	DespachoLitros	FechaDespacho	fechareg	userreg	fechamod	usermod	KMAnterior	Operador	Rendimi
0	1	1007	674258	300.000	12/04/2017	00:00.0	f490eddd-061c-4b64-bfd3-26ef469b9263	00:00.0	f490eddd-061c-4b64-bfd3-26ef469b9263	661020	Desconocido	
1	2	1008	1380120	300.000	16/03/2017	00:00.0	f490eddd-061c-4b64-bfd3-26ef469b9263	00:00.0	f490eddd-061c-4b64-bfd3-26ef469b9263	1379910	Desconocido	
2	3	1009	460325	159.000	24/03/2017	00:00.0	f490eddd-061c-4b64-bfd3-26ef469b9263	00:00.0	f490eddd-061c-4b64-bfd3-26ef469b9263	456858	Desconocido	
3	4	1010	920375	236.000	10/04/2017	00:00.0	f490eddd-061c-4b64-bfd3-	00:00.0	f490eddd-061c-4b64-bfd3-	920328	Desconocido	

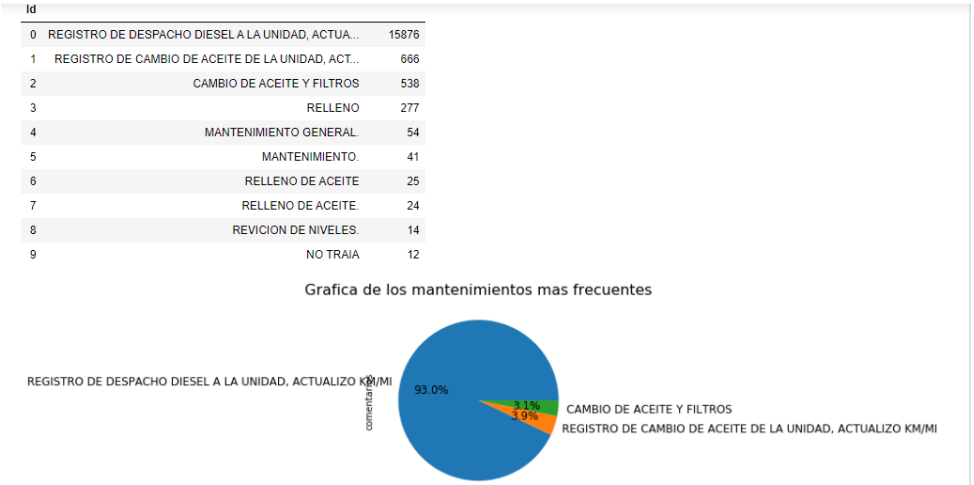
- **Data Grinding**

Una vez los datos estuvieron listos para trabajar con ellos, procedimos a extraer de cada una de las bases de datos la información más relevante para la optimización de los mantenimientos definiendo las funciones necesarias para dichas tareas como listados y agrupaciones o funciones para graficar y permitir una presentación de la información más amigable.

Cantidad de Mantenimientos.

Lo primero que decidimos obtener fue la cantidad de mantenimientos que se realizan actualmente. Conocer esto nos permite tener un panorama sobre la situación actual de la línea transportista y a partir de ahí tomar las decisiones necesarias para la reducción y poder contemplar los resultados posteriores a estas.

	id	
0	REGISTRO DE DESPACHO DIESEL A LA UNIDAD, ACTUA...	15876
1	REGISTRO DE CAMBIO DE ACEITE DE LA UNIDAD, ACT...	666
2	CAMBIO DE ACEITE Y FILTROS	538
3	RELLENO	277
4	MANTENIMIENTO GENERAL.	54
5	MANTENIMIENTO.	41
6	RELLENO DE ACEITE	25
7	RELLENO DE ACEITE.	24
8	REVICION DE NIVELES.	14
9	NO TRAIA	12

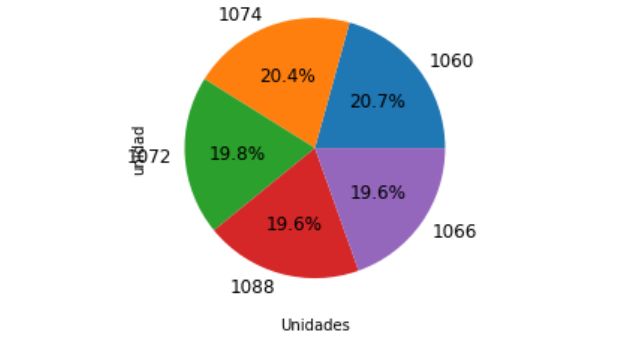


Cantidad de mantenimientos por unidad.

Posteriormente, decidimos que un enfoque hacia los mantenimientos por unidad era nuestro siguiente paso. Extrayendo aquellas unidades que han recibido una mayor cantidad de mantenimientos nos permite buscar una relación entre ellas con el fin de encontrar el motivo de los mismos siendo posible encontrar relaciones basadas en posibles fallas, modelos o relacionadas en base a los viajes realizados (distancia, operador, etc).

Out[113]:

Id	Unidad	Cantidad de Mantenimiento
0	1060	269
1	1074	265
2	1072	257
3	1088	254
4	1066	254

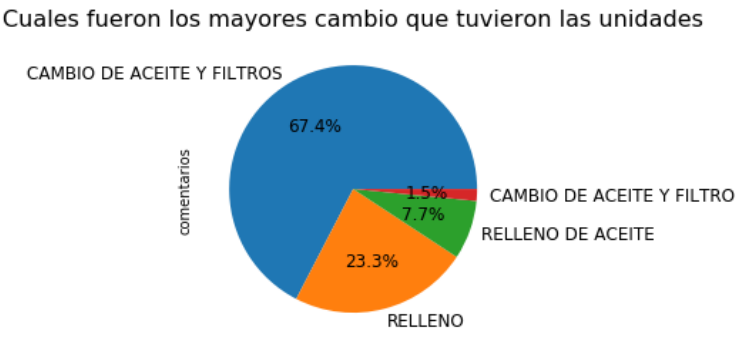


Mayores cambios de aceite.

Estos datos nos permiten tomar uno de los motivos más comunes de mantenimiento de las unidades, analizando cada acción realizada y detectando si existe una cantidad anormal dentro de los parámetros de la empresa. Un pico anormal dentro de estos datos podría significar un fallo con algunas unidades o refacciones de estas.

Out[117]:

Id	Cambios	Cantidad
0	CAMBIO DE ACEITE Y FILTROS	612
1	RELLENO	212
2	RELLENO DE ACEITE	70
3	CAMBIO DE ACEITE Y FILTRO	14



Unidades con mayor gasto (compra de aceite).
Dentro de las decisiones administrativas de cualquier empresa, el factor económico es vital en la toma de decisiones, por lo que decidimos que el conocer que unidades están causando una mayor inversión para la empresa es un dato valioso. De esta manera, es posible encontrar posibles fallas y analizar si es conveniente invertir en las unidades existentes o si estas están causando pérdidas a largo plazo y es más factible invertir en nuevos vehículos.

Out[124]:

Id	Unidad	Litros Totales
18	1031	552
114	L008	251
67	1085	216
117	L011	215
68	1086	194
19	1032	184
113	L0009	181
33	1049	176
17	1030	172
10	1018	162

Estadísticas.

Por último, se procedió a obtener los datos estadísticos de cada conjunto de datos con el fin de conocer los parámetros con los que trabaja cada base de datos. Conocer los promedios, máximas y mínimas actuales nos facilita una comprensión de avances y comparativas futuras entre la situación actual y posterior a las decisiones administrativas que la empresa tome.

Mantenimiento de Unidades

```
In [125]: MttoUnidades.head()
```

Out[125]:

	Id	unidad	comentarios	fecha	fechareg	userreg	fechamod	usermod	IdModulo
0	1	1031	RELLENO DE ACEITE	05/05/2017	10:02.2	f490eddd-061c-4b64-bfd3-26ef469b9263	23:42.9	f490eddd-061c-4b64-bfd3-26ef469b9263	CA-09:49:14-1031
1	2	1061	CAMBIO DE ACEITE Y FILTROS	05/05/2017	27:33.1	f490eddd-061c-4b64-bfd3-26ef469b9263	23:50.0	f490eddd-061c-4b64-bfd3-26ef469b9263	CA-09:34:03-1061
2	3	1033	SE PUSIERON 2 SOQUETERAS	08/05/2017	02:28.7	ef281d7a-57db-428e-a186-f1c54d48042c	26:20.1	f490eddd-061c-4b64-bfd3-26ef469b9263	00-00:00:00-000
3	4	1009	SE HIZO UNA TA LACHA POSICIàN 7	08/05/2017	12:03.6	ef281d7a-57db-428e-a186-f1c54d48042c	23:58.4	f490eddd-061c-4b64-bfd3-26ef469b9263	00-00:00:00-000
4	5	1060	SE LE CAMBIO UN PLAFON DE STOP	08/05/2017	13:16.5	ef281d7a-57db-428e-a186-f1c54d48042c	24:01.8	f490eddd-061c-4b64-bfd3-26ef469b9263	00-00:00:00-000

```
In [126]: MttoUnidades["unidad"].describe()
```

Out[126]:

```
count     19351
unique      148
top        1060
freq        269
Name: unidad, dtype: object
```

DespachoAceite ¶

```
In [127]: DespachoAceite.head()
```

Out[127]:

	unidad	linicialaceite	lfinalaceite	comentarios	cantidadlitros	fecha	preciol	Costo	Id
0	1032	2830	2834	CAMBIO DE ACEITE	4	18/02/2017	43.3	173.2	1
1	1078	2834	2858	CAMBIO DE ACEITE	24	18/02/2017	43.3	1039.2	2
2	136	2858	2862	CAMBIO DE ACEITE	4	18/02/2017	43.3	173.2	3
3	138	2862	2866	CAMBIO DE ACEITE	4	20/02/2017	43.3	173.2	4
4	1089	2866	2875	CAMBIO DE ACEITE Y FILTROS	9	20/02/2017	43.3	389.7	7

```
In [128]: DespachoAceite.describe()
```

Out[128]:

	linicialaceite	lfinalaceite	cantidadlitros	preciol	Costo	Id
count	943.000000	943.000000	943.000000	943.000000	943.000000	943.000000
mean	9098.009544	9089.847296	10.044539	43.253446	434.449629	503.430541
std	4797.872515	4815.513522	9.264922	1.410089	401.374160	282.511066
min	2830.000000	0.000000	0.000000	0.000000	0.000000	1.000000
25%	5575.500000	5568.500000	5.000000	43.300000	216.500000	273.500000
50%	7550.000000	7550.000000	7.000000	43.300000	303.100000	509.000000
75%	15093.500000	15093.500000	10.000000	43.300000	433.000000	746.500000
max	17648.000000	17691.000000	43.000000	43.300000	1861.900000	982.000000

Registro de Despachos

```
In [129]: RegDespacho.head()
```

Out[129]:

	id	noeconomico	KMActual	DespachoLitros	FechaDespacho	fechareg	userreg	fechamod	usermod	KMAnterior	Operador	Rendimiento	Pre
0	1	1007	674258	300.0	12/04/2017	00:00.0	f490eddd-061c-4b64-bfd3-26ef469b9263	00:00.0	f490eddd-061c-4b64-bfd3-26ef469b9263	661020	Desconocido	0.0	
1	2	1008	1380120	300.0	16/03/2017	00:00.0	f490eddd-061c-4b64-bfd3-26ef469b9263	00:00.0	f490eddd-061c-4b64-bfd3-26ef469b9263	1379910	Desconocido	0.0	
2	3	1009	460325	159.0	24/03/2017	00:00.0	f490eddd-061c-4b64-bfd3-26ef469b9263	00:00.0	f490eddd-061c-4b64-bfd3-26ef469b9263	456858	Desconocido	0.0	
3	4	1010	920375	236.0	10/04/2017	00:00.0	f490eddd-061c-4b64-bfd3-26ef469b9263	00:00.0	f490eddd-061c-4b64-bfd3-26ef469b9263	920328	Desconocido	0.0	
4	5	1011	218724	215.0	05/04/2017	00:00.0	f490eddd-061c-4b64-bfd3-26ef469b9263	00:00.0	f490eddd-061c-4b64-bfd3-26ef469b9263	215419	Desconocido	0.0	

```
In [130]: RegDespacho.describe()
```

Out[130]:

	id	KMActual	DespachoLitros	KMAnterior	Rendimiento	Precio
count	15896.000000	1.589600e+04	15896.000000	1.589600e+04	15896.000000	15896.000000
mean	8714.841092	2.400681e+05	210.861765	2.386799e+05	21.156500	9.228315
std	4929.740810	2.384656e+05	2987.067636	2.384937e+05	806.472248	7.026694
min	1.000000	2.930000e+02	3.854000	0.000000e+00	-11759.810000	0.000000
25%	4974.750000	1.012840e+05	70.507000	1.001628e+05	1.180000	0.000000
50%	8966.500000	1.660470e+05	100.000000	1.652820e+05	2.950000	13.850000
75%	12946.250000	2.848858e+05	248.669000	2.834090e+05	7.210000	14.860000
max	16923.000000	3.985590e+06	259844.000000	3.978750e+06	92645.590000	18.900000