retail_data_analytics

June 17, 2019

1 Consigna práctico Análisis y Curación: Limpiando un Dataset

1. Importando los datos:
    - Verificar si no hay problemas en la importación: importar los datos, visualizarlos, análisis de tipos, corrección en los tipos de los datos de entrada.
    - Asegurar que el archivo sales posee Ids/Claves únicas. Para el resto de los archivos, £tenemos algún atributo que se comporte como clave única?, en caso positivo chequear que no se repite. En caso de no tener una clave única identificatoria, £sería relevante asignar una clave única a cada registro?, chequear que no existen datos duplicados para estos casos.
    - Despersonalizar los datos y guardarlos en un nuevo archivo. Tener en cuenta nunca modificar los datos crudos u originales.
2. Pasos necesarios para limpieza del dataset:
    - Etiquetas de variables/columnas: no usar caracteres especiales. Verificar que no haya problemas de codificación/encoding.
    - Tratar valores faltantes (NaN).
    - Codificar variables: las variables categóricas deberán ser tratadas como variables numéricas.
    - Verificar la consistencia de las variables: constatar que los valores de cada atributo tienen sentido, detectar valores que no son consistentes con el resto.
    - Identificar valores atípicos en nuestro dataset. £Qué es conveniente hacer con ellos? Evaluar cada caso.
    - Juntar las columnas de interés en un mismo DataFrame (Sales con Features).
    - Para simplificar el problema resamplear los datos £Transformar Weekly sales en ventas mensuales?. Graficar la distribución de las ventas mensuales para cada año para 5 tiendas a elección. Comparar sus distribuciones. £Se reconoce alguna distribución conocida?
    - Analizar correlación entre número departamento y ventas semanales/mensuales, £posee alguna incidencia el número de departamento sobre las ventas?, en caso negativo eliminar esta variable de nuestros dataframes.
    - Una vez que tenemos las features de interés de nuestro set de datos, aplicar algún método de normalización sobre los mismos, para evitar tener un sesgo de unas variables sobre otra (se pueden utilizar técnicas como z-score/min-max scaling). Guardar el dataset normalizado con un nombre representativo.

- Finalmente, reducir los features de interés mediante sus combinaciones lineales (aplicando Principal Component Analysis). Guardar el dataset con nombre representativo.
- Enumere formas eficientes de guardado y acceso de grandes volúmenes de datos.
- Guardar todos los archivos depurados con nombres representativos. Bonus: entregar el práctico corriendo en una imágen de Docker.

Material de lectura recomendado:
https://www.machinelearningplus.com/time-series/time-series-analysis-python/
https://towardsdatascience.com/playing-with-time-series-data-in-python-959e2485bff8
https://sebastianraschka.com/Articles/2014_about_feature_scaling.html

1.1 Resolución práctico 2

- Primero creamos una copia de los datasets originales.
- Agregamos columna IsMarkdown para los casos en los que hubo descuentos.
- Asegurar que el archivo sales posee Ids/Claves únicas.

Out[8]:

	Unnamed: 0	Store	Dept	Date	Weekly_Sales	IsHoliday	Sale_Id
425795	425795	45	97	2012-03-08	6779.88	False	421422
421422	421422	45	97	2012-03-08	6779.88	False	421422
421263	421263	45	95	2012-04-27	50693.76	False	421263
425794	425794	45	95	2012-04-27	50693.76	False	421263
421198	421198	45	95	2011-01-28	45751.50	False	421198
425793	425793	45	95	2011-01-28	45751.50	False	421198
421181	421181	45	95	2010-01-10	46860.82	False	421181
425792	425792	45	95	2010-01-10	46860.82	False	421181
425791	425791	45	93	2012-10-19	2270.50	NaN	421011
421011	421011	45	93	2012-10-19	2270.50	NaN	421011
425790	425790	45	93	2011-04-03	1886.44	False	420926
420926	420926	45	93	2011-04-03	1886.44	False	420926
420856	420856	45	92	2012-07-27	44551.11	False	420856
425789	425789	45	92	2012-07-27	44551.11	False	420856
420849	420849	45	92	2012-08-06	50688.59	False	420849
425788	425788	45	92	2012-08-06	50688.59	False	420849
425787	425787	45	92	2011-11-18	54868.94	False	420820
420820	420820	45	92	2011-11-18	54868.94	False	420820
425786	425786	45	92	2010-10-22	46663.68	False	420764
420764	420764	45	92	2010-10-22	46663.68	False	420764
425785	425785	45	90	2012-07-20	23035.84	False	420569
420569	420569	45	90	2012-07-20	23035.84	False	420569
425784	425784	45	90	2010-05-03	22653.30	False	420445
420445	420445	45	90	2010-05-03	22653.30	False	420445
425783	425783	45	87	2012-02-24	7224.30	False	420405
420405	420405	45	87	2012-02-24	7224.30	False	420405
425782	425782	45	87	2011-12-23	10939.66	False	420396
420396	420396	45	87	2011-12-23	10939.66	False	420396
420320	420320	45	87	2010-09-07	6331.09	NaN	420320

```
425781      425781     45    87 2010-09-07       6331.09       NaN   420320
...           ...     ...   ...        ...           ...       ...      ...
1469         1469       1    11 2010-05-11      25063.26     False     1469
421584     421584       1    11 2010-05-11      25063.26     False     1469
421583     421583       1    11 2010-03-26      17592.13     False     1437
1437         1437       1    11 2010-03-26      17592.13     False     1437
421582     421582       1    10 2011-06-24      29831.79     False     1359
1359         1359       1    10 2011-06-24      29831.79     False     1359
421581     421581       1    10 2010-12-11      26288.47     False     1327
1327         1327       1    10 2010-12-11      26288.47     False     1327
421580     421580       1     5 2011-10-28      26391.79     False      662
662           662       1     5 2011-10-28      26391.79     False      662
421579     421579       1     5 2011-07-29      17406.68     False      649
649           649       1     5 2011-07-29      17406.68     False      649
421578     421578       1     4 2012-07-20      38080.05     False      557
557           557       1     4 2012-07-20      38080.05     False      557
421577     421577       1     4 2012-01-13      36582.36     False      530
530           530       1     4 2012-01-13      36582.36     False      530
504           504       1     4 2011-07-15      33930.80     False      504
421576     421576       1     4 2011-07-15      33930.80     False      504
421575     421575       1     2 2010-04-16      45025.02     False      153
153           153       1     2 2010-04-16      45025.02     False      153
421574     421574       1     2 2010-03-19      43615.49     False      149
149           149       1     2 2010-03-19      43615.49     False      149
421573     421573       1     1 2012-10-26      27390.81     False      142
142           142       1     1 2012-10-26      27390.81     False      142
136           136       1     1 2012-09-14      19616.22     False      136
421572     421572       1     1 2012-09-14      19616.22     False      136
421571     421571       1     1 2011-09-09      17746.68      True       83
83             83       1     1 2011-09-09      17746.68      True       83
51             51       1     1 2011-01-28      18461.18     False       51
421570     421570       1     1 2011-01-28      18461.18     False       51

[8452 rows x 7 columns]
```

Tal como se observa en la tabla anterior, el archivo sales tiene 8452 filas duplicadas. - Para el caso de **features** clave única podría ser la unión de los atributos Store+Date ya que dicha combinacion no debiera tener duplicados. - Para el caso de **stores** la columna Store deberia ser una clave única sin duplicados. - En la tabla **users** el atributo que se comporta como clave unica es la columna users

```
Out[9]: Empty DataFrame
        Columns: [Unnamed: 0, Store, Type, Size]
        Index: []
```

```
Out[10]: Empty DataFrame
         Columns: [Unnamed: 0, users, stores]
         Index: []
```

```
Out[11]: Empty DataFrame
         Columns: [Unnamed: 0, users, stores]
         Index: []
```

```
Out[13]: Empty DataFrame
         Columns: [Unnamed: 0, Store, Date, Temperature, Fuel_Price, MarkDown1, MarkDown2, Mar
         Index: []
```

Tal como puede verse ninguna de estas tres tablas de datos tiene valores duplicados en sus columnas de índices

1.1.1 Despersonalizar los datos y guardarlos en un nuevo archivo

```
Out[17]:    Unnamed: 0                                users          stores
         0           0  <md5 HASH object @ 0x7f64104260a8>      16-32-7-44
         1           1  <md5 HASH object @ 0x7f64104261e8>     20-37-10-11
         2           2  <md5 HASH object @ 0x7f6410426198>  34-14-18-16-29
         3           3  <md5 HASH object @ 0x7f6410426120>     22-14-39-25
         4           4  <md5 HASH object @ 0x7f64104260f8>   31-7-27-21-13
```

- Asegurar que las etiquetas de las variables no contengan caracteres especiales.

```
Out[19]: Index(['Unnamed: 0'], dtype='object')
```

```
Out[20]: Index(['Unnamed: 0'], dtype='object')
```

```
Out[21]: Index(['Unnamed: 0'], dtype='object')
```

```
Out[22]: Index(['Unnamed: 0'], dtype='object')
```

El único campo que contiene caracteres especiales es el campo 'Unnamed: 0' que se observa en todos los datasets. Esta columna no aporta nada y puede removerse.

No existen campos con cadenas de texto, salvo el nombre de los usuarios, el cual debe ser despersonalizado, con lo cual no lo analizarémos. Pero si hubiera alguna columna cuyo tipo fuera String, podría analizarse la rareza de los Strings utilizando ftfy.badness.

1.1.2 Tratamiento de valores faltantes

```
Out[24]: Store              0
         Dept               0
         Date               0
         Weekly_Sales       0
         IsHoliday      42118
         Sale_Id            0
         dtype: int64
```

```
Out[25]: 42118
```

Hay 42.118 valores faltantes que pertenecen al feature IsHoliday. Esta medida en sí, no dice nada más que con cuantos valores debemos lidiar.

Primero tenemos que ver si en términos relativos su impacto.

Out[26]: 0.9010840872154741

Out[27]: 0.9010840872154741

Eliminar las filas que tienen NAN en la columna IsHoliday implicaria quedarse con el 90% de los datos del dataset original. Ahora verificamos si las fechas con faltantes en IsHoliday son aleatorias o si pertenecen a periodos continuos de tiempo:

Out[28]:

	Store	Dept	Date	Weekly_Sales	IsHoliday	Sale_Id
162595	17	40	2010-01-10	48169.96	NaN	162595
288112	30	12	2010-02-04	103.32	NaN	288112
176071	18	90	2010-02-07	13766.11	NaN	176071
274909	28	80	2010-02-19	18431.14	NaN	274909
338781	35	83	2010-02-26	5253.18	NaN	338781
111477	12	27	2010-03-09	2164.30	NaN	111477
290612	30	56	2010-03-12	336.00	NaN	290612
304996	32	9	2010-03-19	11239.75	NaN	304996
287270	30	6	2010-03-26	20.86	NaN	287270
207148	22	1	2010-04-06	12956.89	NaN	207148
67800	7	92	2010-04-16	14781.33	NaN	67800
33657	4	31	2010-04-23	1628.00	NaN	33657
131360	14	25	2010-04-30	15710.50	NaN	131360
303644	31	98	2010-05-02	10141.32	NaN	303644
83347	9	35	2010-05-03	1673.00	NaN	83347
111486	12	27	2010-05-11	1958.95	NaN	111486
158050	17	5	2010-05-14	22696.12	NaN	158050
171616	18	32	2010-05-21	10900.45	NaN	171616
377131	40	52	2010-05-28	979.13	NaN	377131
321848	34	10	2010-06-08	18040.17	NaN	321848
34237	4	35	2010-06-18	5075.00	NaN	34237
255023	26	83	2010-06-25	4412.29	NaN	255023
22497	3	16	2010-07-05	12688.20	NaN	22497
138429	15	3	2010-07-16	10890.75	NaN	138429
306276	32	19	2010-07-23	1444.65	NaN	306276
25770	3	41	2010-07-30	971.00	NaN	25770
288425	30	14	2010-08-10	1065.99	NaN	288425
126491	13	85	2010-08-13	5624.13	NaN	126491
313035	32	92	2010-08-20	92808.43	NaN	313035
310736	32	58	2010-08-27	3603.00	NaN	310736
...	...	...	...	...	...	...
310678	32	56	2012-04-13	7207.37	NaN	310678
355398	38	12	2012-04-20	88.10	NaN	355398
345909	36	94	2012-04-27	34293.48	NaN	345909
212482	22	41	2012-05-10	878.00	NaN	212482
282130	29	38	2012-05-18	45789.12	NaN	282130
337331	35	58	2012-05-25	740.00	NaN	337331
45338	5	44	2012-06-01	1064.82	NaN	45338
35047	4	41	2012-06-04	868.00	NaN	35047

```
294116     31     3 2012-06-07       7275.10       NaN   294116
402696     43    79 2012-06-15      13116.14       NaN   402696
337614     35    60 2012-06-22         99.00       NaN   337614
218374     23    11 2012-06-29      31719.37       NaN   218374
405607     44     6 2012-07-09         17.82       NaN   405607
10371       2     1 2012-07-13      23148.57       NaN    10371
124724     13    56 2012-07-20       4863.03       NaN   124724
385009     41    30 2012-07-27       3273.49       NaN   385009
369034     39    81 2012-08-06      21175.10       NaN   369034
7493        1    67 2012-08-17       5640.72       NaN     7493
168748     18    10 2012-08-24      15006.65       NaN   168748
385300     41    32 2012-08-31       6153.99       NaN   385300
255827     26    92 2012-09-03     106049.59       NaN   255827
120576     13    22 2012-09-14      16170.95       NaN   120576
360212     38    92 2012-09-21      46074.34       NaN   360212
189829     20    19 2012-09-28        -19.90       NaN   189829
425434     41    40 2012-10-02      58574.33       NaN   386131
261170     27    32 2012-10-08       8127.42       NaN   261170
51760       6    22 2012-10-19      16188.59       NaN    51760
330578     34    97 2012-10-26      18521.65       NaN   330578
169719     18    18 2012-11-05         -1.53       NaN   169719
200367     21    21 2012-12-10       2833.09       NaN   200367

[143 rows x 6 columns]
```

Tal como se ve, las fechas con faltantes parecen ser aleatorias.

```
Out[29]: Store             0
         Date              0
         Temperature       0
         Fuel_Price        0
         MarkDown1      4158
         MarkDown2      5269
         MarkDown3      4577
         MarkDown4      4726
         MarkDown5      4140
         CPI             585
         Unemployment    585
         IsHoliday         0
         IsMarkdown        0
         Col3              0
         dtype: int64
```

Vamos a analizar en que casos todas las columnas de markdown son NaN, ya que en el análisis nuestro nos interesa solo saber si en una fecha hubo o no descuentos.

```
Out[31]: 0.2526251526251526
```

En el caso de la tabla features si eliminamos todas las filas con valores faltantes nos quedamos solo con el 25% de las filas. Por esto no seria recomendable descartar estas filas porque perderiamos mucha informacion de las columnas que no tienen valores faltantes, como temperatura y precio. Se podria ver la correlacion entre las ventas semanales y la existencia de descuento o no o alternativamente la correlación entre ventas semanales y cada una de las columnas de descuentos para los casos en que ninguna de ellas posee faltantes. Si no hubiera correlacion podriamos descartar estas columnas.

Cabe observar que en esta tabla tambien esta la variable IsHoliday y no tiene faltantes por lo que no seria necesario eliminar las filas con faltantes de la tabla sales.

```
Out[32]: 0.5054945054945055
```

Podemos ver que en la mitad de los registros del dataset, se encuetran todas las columnas MarkDown en nulo.

```
Out[33]: Store    0
         Type     0
         Size     6
         dtype: int64
```

```
Out[34]: 0.8666666666666667
```

La tabla stores contiene 6 faltantes en la columna Size que representan 13,33% del data set.

```
Out[35]: array([ 1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16, 17,
                18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34,
                35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45])
```

```
Out[36]: array(['A', 'B', 'C'], dtype=object)
```

Analizando el dataset de Stores, podemos ver que existen 3 tipos distintos de tiendas (A, B y C). El único valor faltante en este dataset es el Size de 6 tiendas, el cual lo podemos rellenar con el promedio de Size para cada tipo de tienda.

```
Out[40]: Store    0
         Type     0
         Size     0
         dtype: int64
```

Ya completamos la información de las 6 tiendas faltantes con los respectivos promedios para cada tipo de tienda.

```
Out[41]: users     0
         stores    0
         dtype: int64
```

1.1.3 Codificar variables: las variables categóricas deberán ser tratadas como variables numéricas.

Las unicas variables categoricas del dataset son IsHoliday y Type

Vamos a hacer una transformación de las variables categóricas Type de stores y IsHoliday de features y IsHoliday de sales.

Out[45]:

	Store	Type	Size
0	1	0	151315.000000
1	2	0	202307.000000
2	3	1	37392.000000
3	4	0	175194.277778
4	5	1	34875.000000

Out[46]:

	Store	Date	Temperature	Fuel_Price	MarkDown1	MarkDown2	MarkDown3	\
0	1	2010-05-02	42.31	2.572	NaN	NaN	NaN	
1	1	2010-12-02	38.51	2.548	NaN	NaN	NaN	
2	1	2010-02-19	39.93	2.514	NaN	NaN	NaN	
3	1	2010-02-26	46.63	2.561	NaN	NaN	NaN	
4	1	2010-05-03	46.50	2.625	NaN	NaN	NaN	

	MarkDown4	MarkDown5	CPI	Unemployment	IsHoliday	IsMarkdown	\
0	NaN	NaN	211.096358	8.106	0	False	
1	NaN	NaN	211.242170	8.106	1	False	
2	NaN	NaN	211.289143	8.106	0	False	
3	NaN	NaN	211.319643	8.106	0	False	
4	NaN	NaN	211.350143	8.106	0	False	

	Col3	is_na
0	12010-05-02 00:00:00	True
1	12010-12-02 00:00:00	True
2	12010-02-19 00:00:00	True
3	12010-02-26 00:00:00	True
4	12010-05-03 00:00:00	True

Out[47]:

	Store	Dept	Date	Weekly_Sales	IsHoliday	Sale_Id
0	1	1	2010-05-02	24924.50	0	0
1	1	1	2010-12-02	46039.49	1	1
2	1	1	2010-02-19	41595.55	0	2
3	1	1	2010-02-26	19403.54	0	3
4	1	1	2010-05-03	21827.90	0	4

1.1.4 Verificar la consistencia de las variables: constatar que los valores de cada atributo tienen sentido, detectar valores que no son consistentes con el resto.

1.1.5 Identificar valores atípicos en nuestro dataset. £Qué es conveniente hacer con ellos? Evaluar cada caso.

Analizamos si existen ventas semanales negativas

Out[48]:

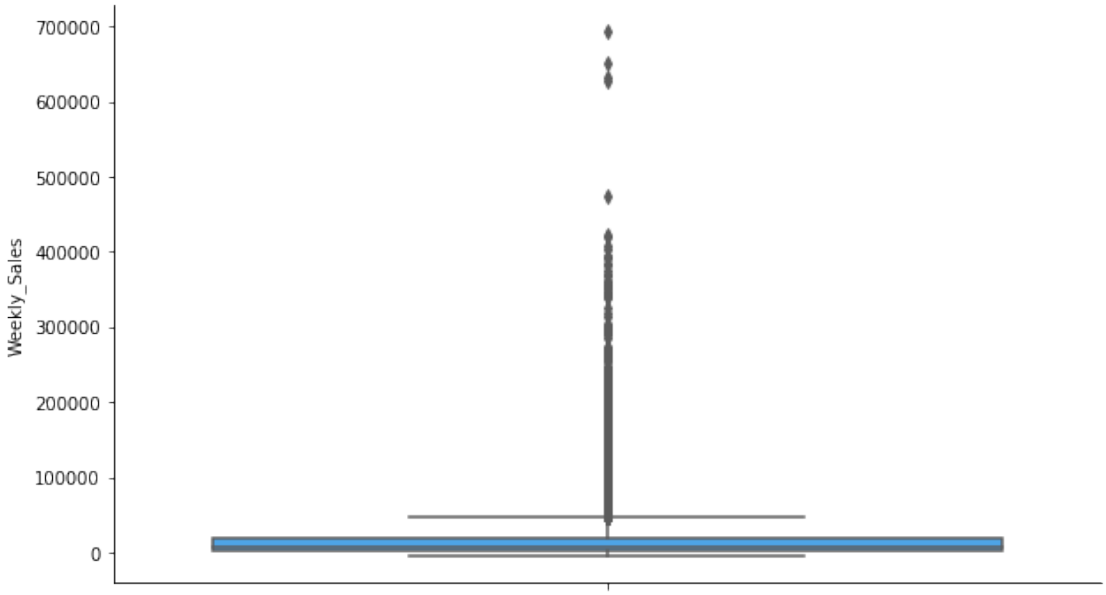
	Store	Dept	Date	Weekly_Sales
count	425796.000000	425796.000000	425796	425796.000000
unique	NaN	NaN	143	NaN
top	NaN	NaN	2011-12-23 00:00:00	NaN
freq	NaN	NaN	3056	NaN
first	NaN	NaN	2010-01-10 00:00:00	NaN
last	NaN	NaN	2012-12-10 00:00:00	NaN
mean	22.200035	44.260944	NaN	15980.254676
std	12.785342	30.494688	NaN	22711.970177
min	1.000000	1.000000	NaN	-4988.940000
25%	11.000000	18.000000	NaN	2080.495000
50%	22.000000	37.000000	NaN	7610.830000
75%	33.000000	74.000000	NaN	20204.122500
max	45.000000	99.000000	NaN	693099.360000

	IsHoliday	Sale_Id
count	425796.000000	425796.000000
unique	NaN	NaN
top	NaN	NaN
freq	NaN	NaN
first	NaN	NaN
last	NaN	NaN
mean	0.261365	210779.929558
std	0.625209	121698.163454
min	0.000000	0.000000
25%	0.000000	105381.750000
50%	0.000000	210804.500000
75%	0.000000	316161.250000
max	2.000000	421569.000000

Observamos que existen ventas negativas, asique analizamos cuantas fueron sobre el total de ventas

```
Ventas semanales negativas: 1299
Proporción sobre el total de ventas: 0.0030507567003917367
```

Existen 1299 ventas semanales con valor negativo. Debería analizarse por que motivo se han cargado ventas con valores negativos. Representan el 0.3% del dataset.



Se observan muchos outliers con ventas semanales mucho más altas que el resto. Habría que analizar si al separar por tipo de tienda sigue apareciendo esta relación, ya que existen 3 tipos distintos de tiendas, y de distintos tamaños, lo que indica que puede ser lógico que una tienda tenga más ventas que otras.

Out[51]:

	Store	Date	Temperature	Fuel_Price
count	8190.000000	8190	8190.000000	8190.000000
unique	NaN	182	NaN	NaN
top	NaN	2011-07-15 00:00:00	NaN	NaN
freq	NaN	45	NaN	NaN
first	NaN	2010-01-10 00:00:00	NaN	NaN
last	NaN	2013-12-07 00:00:00	NaN	NaN
mean	23.000000	NaN	59.356198	3.405992
std	12.987966	NaN	18.678607	0.431337
min	1.000000	NaN	-7.290000	2.472000
25%	12.000000	NaN	45.902500	3.041000
50%	23.000000	NaN	60.710000	3.513000
75%	34.000000	NaN	73.880000	3.743000
max	45.000000	NaN	101.950000	4.468000

	MarkDown1	MarkDown2	MarkDown3	MarkDown4
count	4032.000000	2921.000000	3613.000000	3464.000000
unique	NaN	NaN	NaN	NaN
top	NaN	NaN	NaN	NaN
freq	NaN	NaN	NaN	NaN
first	NaN	NaN	NaN	NaN
last	NaN	NaN	NaN	NaN
mean	7032.371786	3384.176594	1760.100180	3292.935886

```
std       9262.747448    8793.583016   11276.462208   6792.329861
min      -2781.450000    -265.760000    -179.260000      0.220000
25%       1577.532500      68.880000       6.600000    304.687500
50%       4743.580000     364.570000      36.260000   1176.425000
75%       8923.310000    2153.350000     163.150000   3310.007500
max     103184.980000  104519.540000  149483.310000  67474.850000

            MarkDown5          CPI  Unemployment    IsHoliday IsMarkdown  \
count     4050.000000  7605.000000   7605.000000  8190.000000       8190
unique            NaN          NaN           NaN          NaN          2
top               NaN          NaN           NaN          NaN      False
freq              NaN          NaN           NaN          NaN       4140
first             NaN          NaN           NaN          NaN        NaN
last              NaN          NaN           NaN          NaN        NaN
mean      4132.216422   172.460809      7.826821     0.071429        NaN
std      13086.690278    39.738346      1.877259     0.257555        NaN
min       -185.170000   126.064000      3.684000     0.000000        NaN
25%       1440.827500   132.364839      6.634000     0.000000        NaN
50%       2727.135000   182.764003      7.806000     0.000000        NaN
75%       4832.555000   213.932412      8.567000     0.000000        NaN
max     771448.100000   228.976456     14.313000     1.000000        NaN

                          Col3 is_na
count                     8190  8190
unique                    8190     2
top     62010-10-15 00:00:00  True
freq                         1  4140
first                      NaN   NaN
last                       NaN   NaN
mean                       NaN   NaN
std                        NaN   NaN
min                        NaN   NaN
25%                        NaN   NaN
50%                        NaN   NaN
75%                        NaN   NaN
max                        NaN   NaN
```

Podemos observar que en las columnas de MarkDown que sabemos que representan si hubo o no descuentos en esa semana, en algunos casos tienen valores negativos. Habría que analizar que representan realmente estos valores para entender si tiene sentido que esos valores sean negativos.

1.1.6 Juntar las columnas de interés en un mismo DataFrame (Sales con Features).

Out[53]:

	Weekly_Sales	Date	Store	Dept	Temperature	Fuel_Price	CPI	\
0	24924.50	2010-05-02	1	1	42.31	2.572	211.096358	
1	50605.27	2010-05-02	1	2	42.31	2.572	211.096358	
2	13740.12	2010-05-02	1	3	42.31	2.572	211.096358	
3	39954.04	2010-05-02	1	4	42.31	2.572	211.096358	

4	32229.38	2010-05-02	1	5	42.31	2.572	211.096358
5	5749.03	2010-05-02	1	6	42.31	2.572	211.096358
6	21084.08	2010-05-02	1	7	42.31	2.572	211.096358
7	40129.01	2010-05-02	1	8	42.31	2.572	211.096358
8	16930.99	2010-05-02	1	9	42.31	2.572	211.096358
9	30721.50	2010-05-02	1	10	42.31	2.572	211.096358
10	24213.18	2010-05-02	1	11	42.31	2.572	211.096358
11	8449.54	2010-05-02	1	12	42.31	2.572	211.096358
12	41969.29	2010-05-02	1	13	42.31	2.572	211.096358
13	19466.91	2010-05-02	1	14	42.31	2.572	211.096358
14	10217.55	2010-05-02	1	16	42.31	2.572	211.096358
15	13223.76	2010-05-02	1	17	42.31	2.572	211.096358
16	4729.50	2010-05-02	1	18	42.31	2.572	211.096358
17	1947.05	2010-05-02	1	19	42.31	2.572	211.096358
18	5034.10	2010-05-02	1	20	42.31	2.572	211.096358
19	8907.63	2010-05-02	1	21	42.31	2.572	211.096358
20	13623.98	2010-05-02	1	22	42.31	2.572	211.096358
21	24146.49	2010-05-02	1	23	42.31	2.572	211.096358
22	8272.90	2010-05-02	1	24	42.31	2.572	211.096358
23	11609.50	2010-05-02	1	25	42.31	2.572	211.096358
24	11737.12	2010-05-02	1	26	42.31	2.572	211.096358
25	2293.00	2010-05-02	1	27	42.31	2.572	211.096358
26	1085.29	2010-05-02	1	28	42.31	2.572	211.096358
27	7024.95	2010-05-02	1	29	42.31	2.572	211.096358
28	5491.00	2010-05-02	1	30	42.31	2.572	211.096358
29	3455.92	2010-05-02	1	31	42.31	2.572	211.096358

	Unemployment	IsHoliday	IsMarkdown	MarkDown1	MarkDown2	MarkDown3	\
0	8.106	0	False	NaN	NaN	NaN	
1	8.106	0	False	NaN	NaN	NaN	
2	8.106	0	False	NaN	NaN	NaN	
3	8.106	0	False	NaN	NaN	NaN	
4	8.106	0	False	NaN	NaN	NaN	
5	8.106	0	False	NaN	NaN	NaN	
6	8.106	0	False	NaN	NaN	NaN	
7	8.106	0	False	NaN	NaN	NaN	
8	8.106	0	False	NaN	NaN	NaN	
9	8.106	0	False	NaN	NaN	NaN	
10	8.106	0	False	NaN	NaN	NaN	
11	8.106	0	False	NaN	NaN	NaN	
12	8.106	0	False	NaN	NaN	NaN	
13	8.106	0	False	NaN	NaN	NaN	
14	8.106	0	False	NaN	NaN	NaN	
15	8.106	0	False	NaN	NaN	NaN	
16	8.106	0	False	NaN	NaN	NaN	
17	8.106	0	False	NaN	NaN	NaN	
18	8.106	0	False	NaN	NaN	NaN	
19	8.106	0	False	NaN	NaN	NaN	

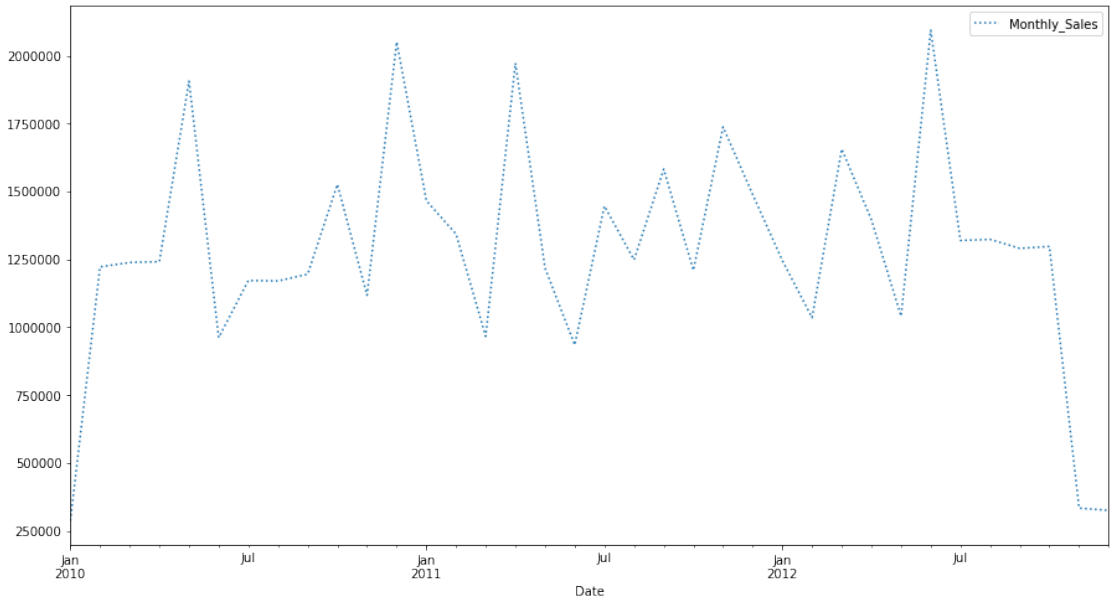
20	8.106	0	False	NaN	NaN	NaN
21	8.106	0	False	NaN	NaN	NaN
22	8.106	0	False	NaN	NaN	NaN
23	8.106	0	False	NaN	NaN	NaN
24	8.106	0	False	NaN	NaN	NaN
25	8.106	0	False	NaN	NaN	NaN
26	8.106	0	False	NaN	NaN	NaN
27	8.106	0	False	NaN	NaN	NaN
28	8.106	0	False	NaN	NaN	NaN
29	8.106	0	False	NaN	NaN	NaN

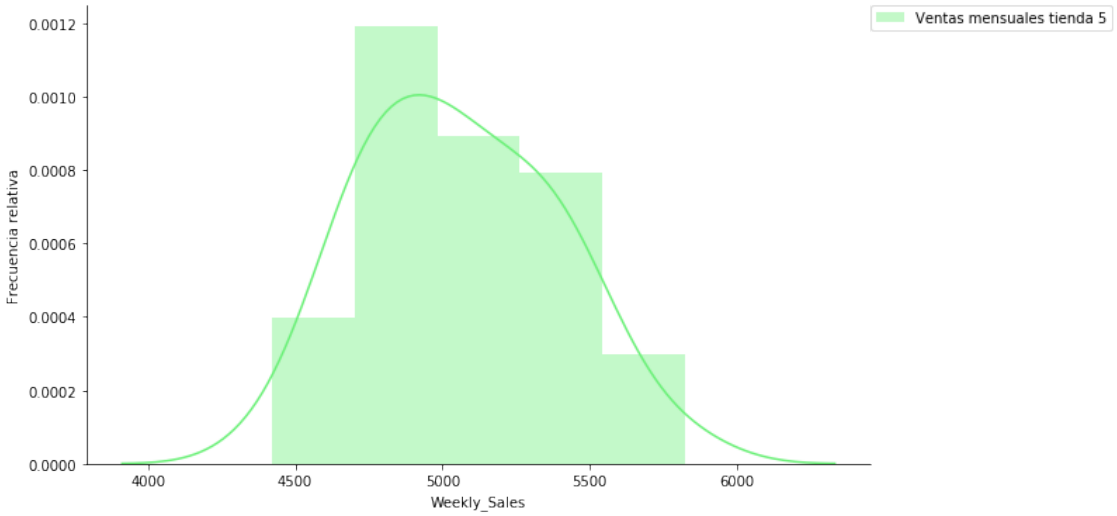
	MarkDown4	MarkDown5
0	NaN	NaN
1	NaN	NaN
2	NaN	NaN
3	NaN	NaN
4	NaN	NaN
5	NaN	NaN
6	NaN	NaN
7	NaN	NaN
8	NaN	NaN
9	NaN	NaN
10	NaN	NaN
11	NaN	NaN
12	NaN	NaN
13	NaN	NaN
14	NaN	NaN
15	NaN	NaN
16	NaN	NaN
17	NaN	NaN
18	NaN	NaN
19	NaN	NaN
20	NaN	NaN
21	NaN	NaN
22	NaN	NaN
23	NaN	NaN
24	NaN	NaN
25	NaN	NaN
26	NaN	NaN
27	NaN	NaN
28	NaN	NaN
29	NaN	NaN

1.1.7 Para simplificar el problema resamplear los datos £Transformar Weekly sales en ventas mensuales?. Graficar la distribución de las ventas mensuales para cada año para 5 tiendas a elección. Comparar sus distribuciones. £Se reconoce alguna distribución conocida?

Ventas mensuales para la tienda 5:

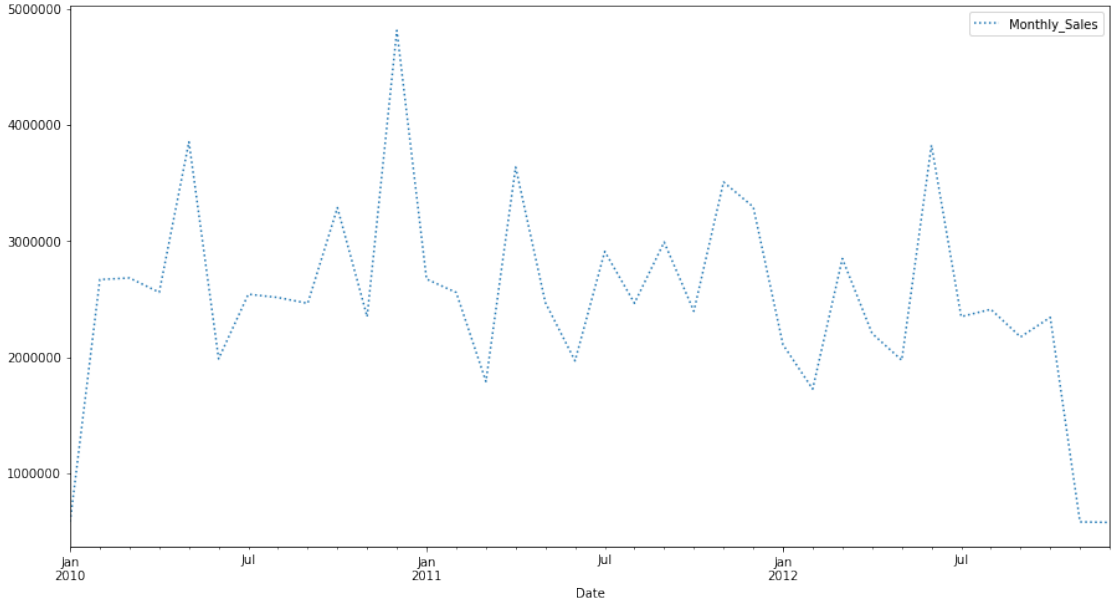
Out[55]: <matplotlib.axes._subplots.AxesSubplot at 0x7f63e04c6320>

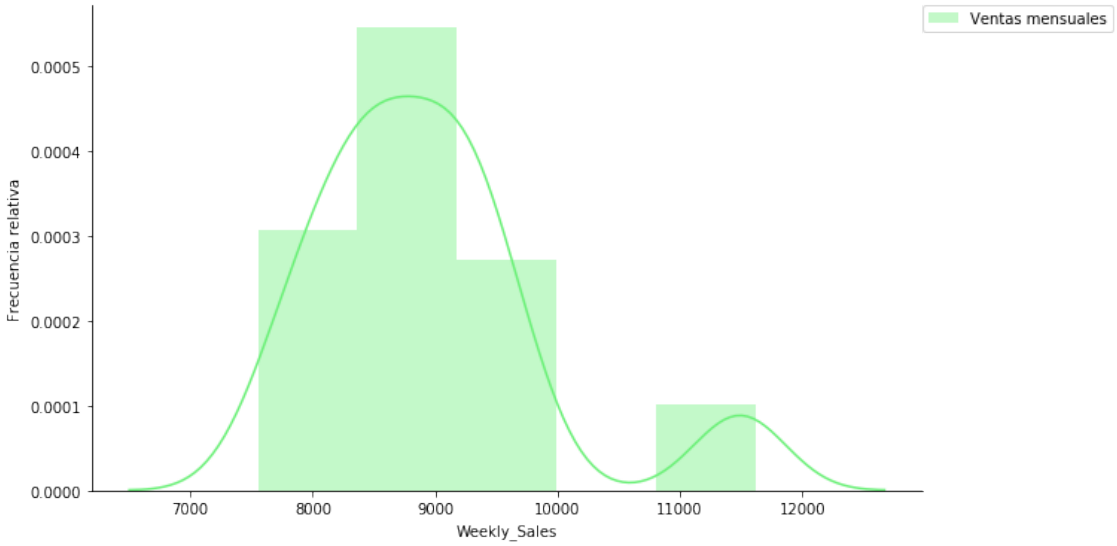




Ventas mensuales para la tienda 15:

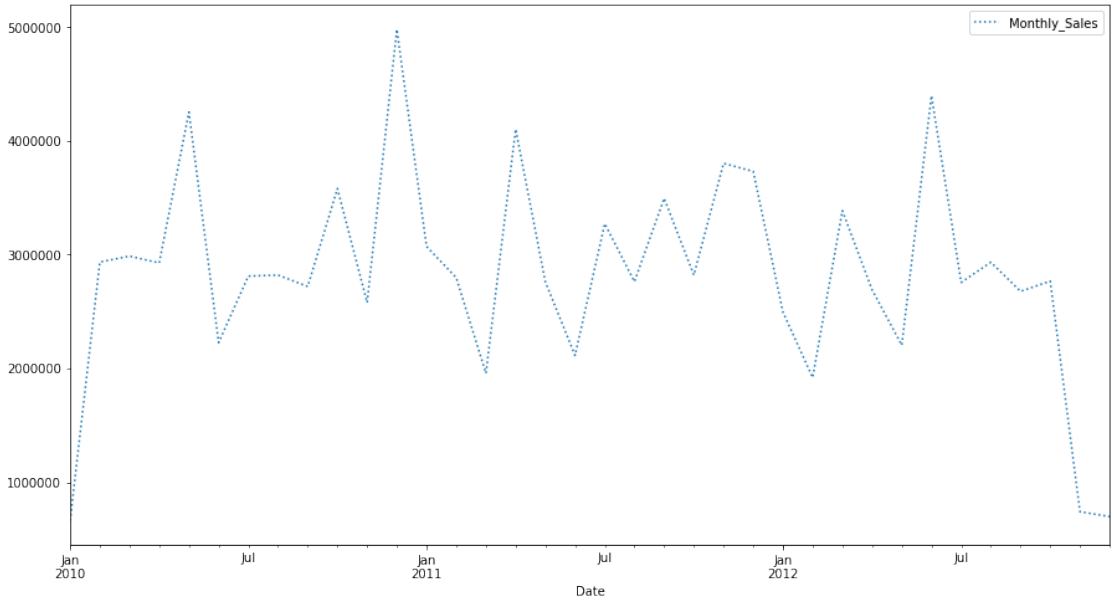
Out[57]: <matplotlib.axes._subplots.AxesSubplot at 0x7f63e03f8470>

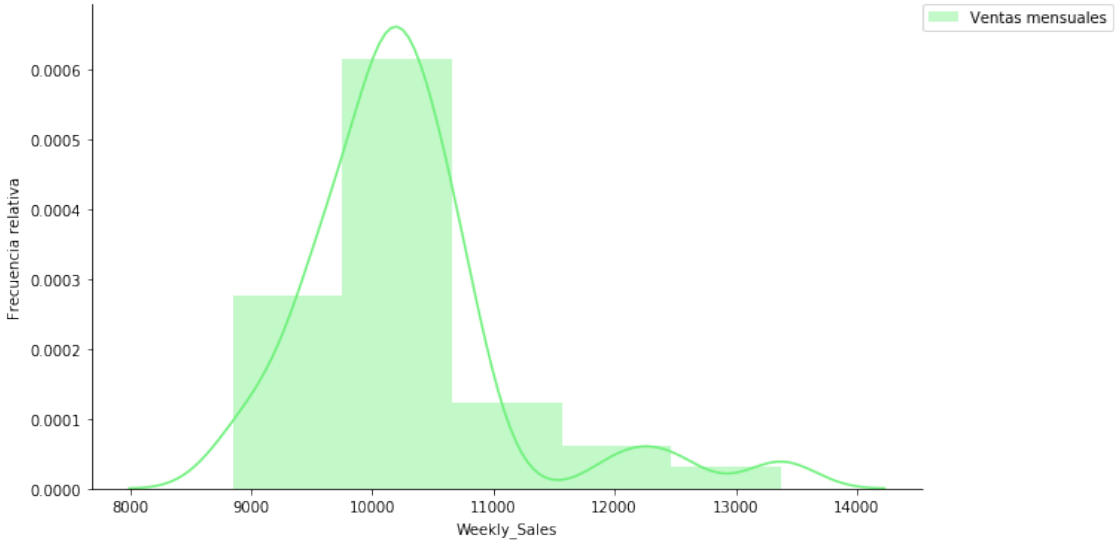




Ventas mensuales para la tienda 25:

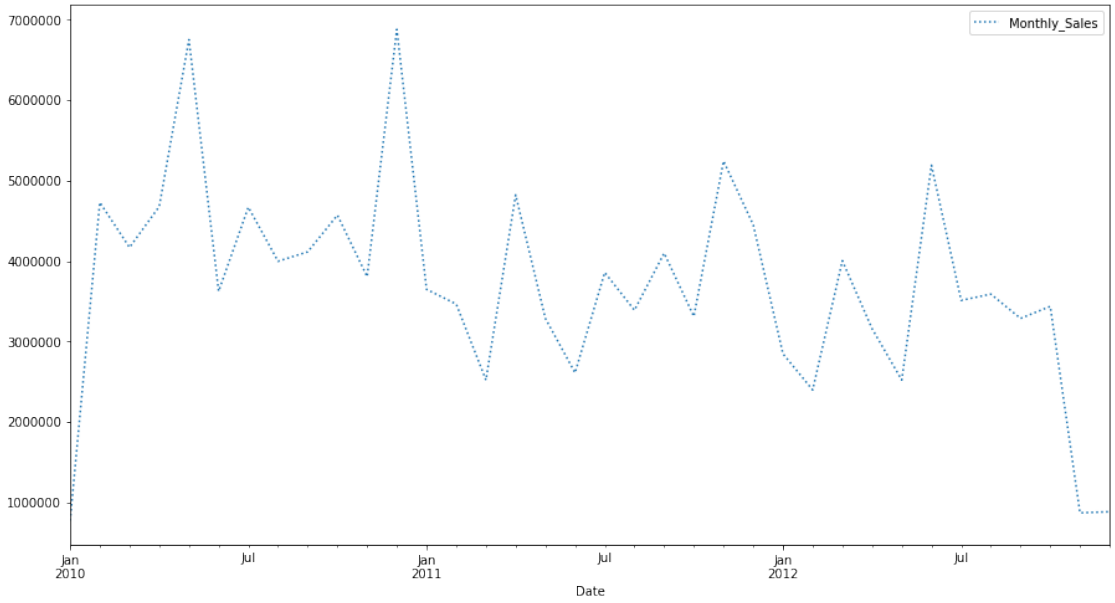
Out[59]: <matplotlib.axes._subplots.AxesSubplot at 0x7f63e04f1518>

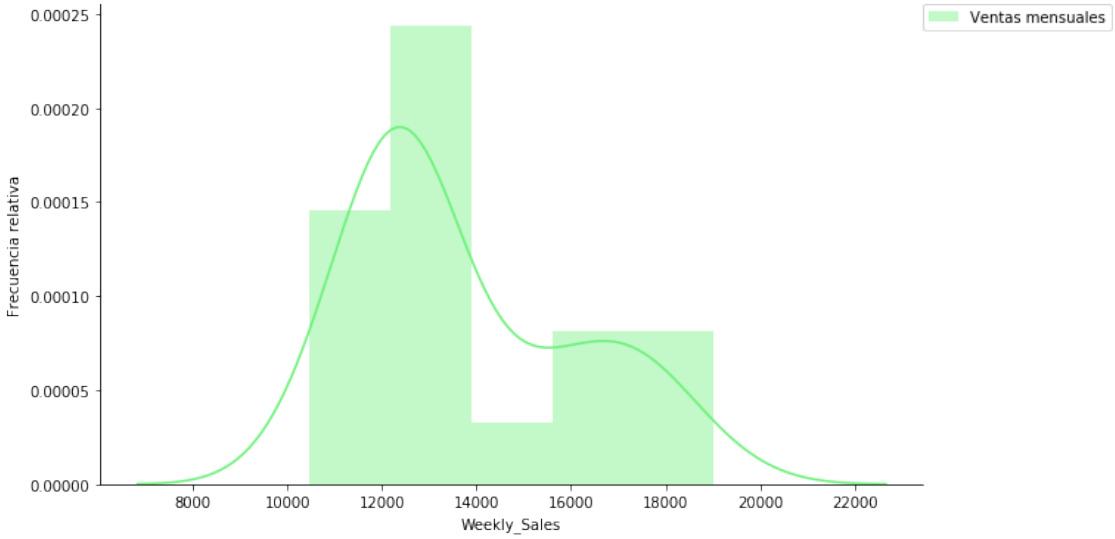




Ventas mensuales para la tienda 35:

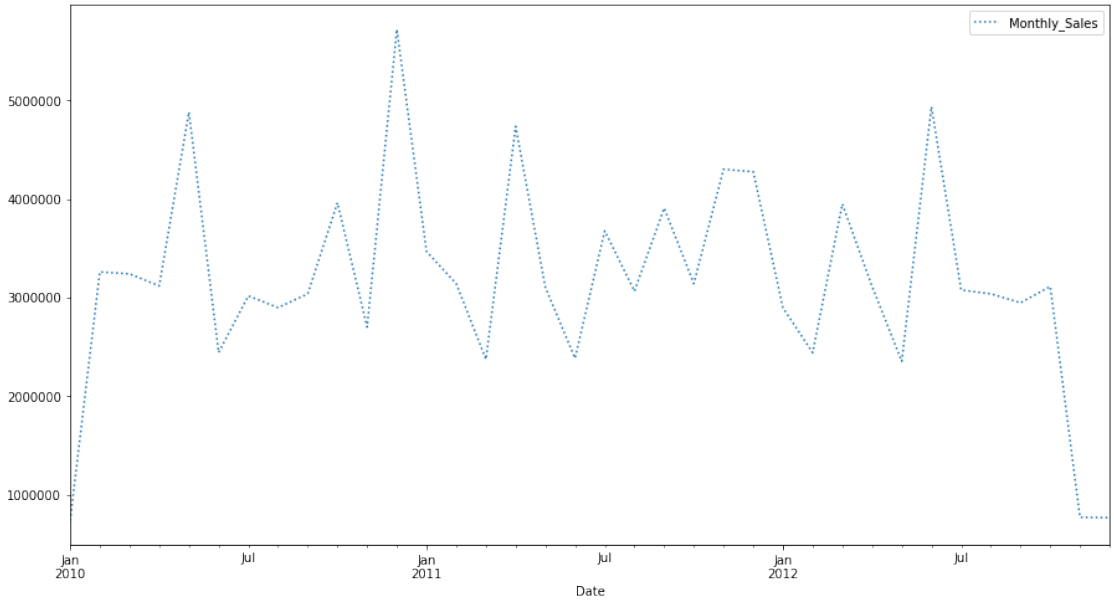
Out[61]: <matplotlib.axes._subplots.AxesSubplot at 0x7f63de4e9828>

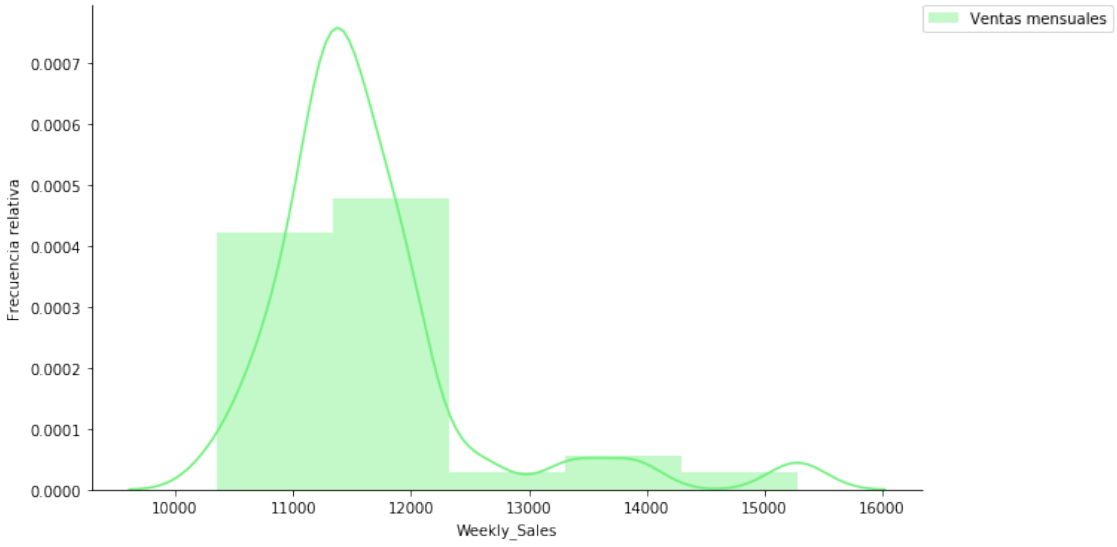




Ventas mensuales para la tienda 45:

Out[63]: <matplotlib.axes._subplots.AxesSubplot at 0x7f63de3a1278>





KS Test tienda 5:

Out[66]: KstestResult(statistic=0.09260209721011892, pvalue=0.9170690324282338)

KS Test tienda 15:

Out[67]: KstestResult(statistic=0.13307991667580743, pvalue=0.512717417916444)

```
KS Test tienda 25:
```

```
Out[68]: KstestResult(statistic=0.18198421050631586, pvalue=0.16260379827610152)
```

```
KS Test tienda 35:
```

```
Out[69]: KstestResult(statistic=0.2368035429641564, pvalue=0.02928658251783439)
```

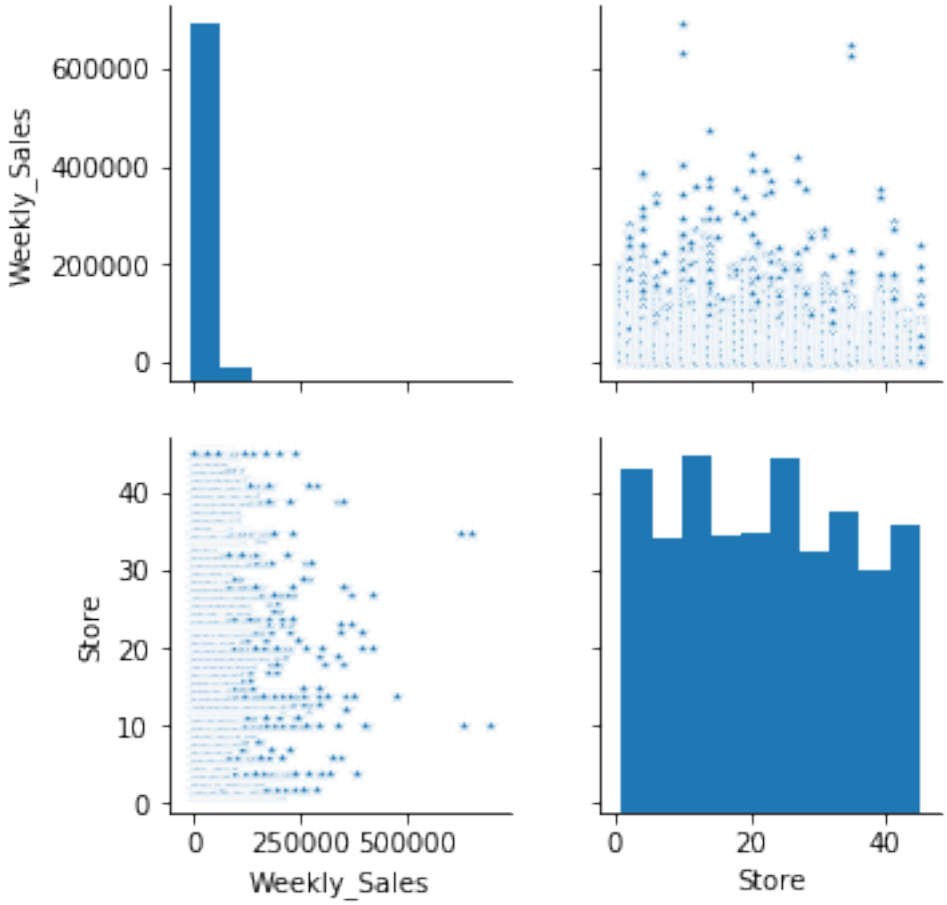
```
KS Test tienda 45:
```

```
Out[70]: KstestResult(statistic=0.2320676970466904, pvalue=0.03458267703760765)
```

Luego de haber aplicado los test de normalidad con los p-valores obtenidos no podemos afirmar que se corresponda a una distribución normal en ningún caso.

1.1.8 Analizar correlación entre número departamento y ventas semanales/mensuales, £posee alguna incidencia el número de departamento sobre las ventas?, en caso negativo eliminar esta variable de nuestros dataframes.

```
Out[71]:               Weekly_Sales     Store
         Weekly_Sales      1.000000 -0.102101
         Store            -0.102101  1.000000
```



De acuerdo al coeficiente de correlacion no puede inferirse que haya relacion lineal entre la variable Weekly Sales y Store. el grafico de dispersión refleja esta falta de relacion. por esta razon podemos eliminar la variable Departamento.

Out[74]:

	Weekly_Sales	Date	Store	Temperature	Fuel_Price	CPI	Unemployment	IsHoliday	IsMarkdown
0	24924.50	2010-05-02	1	42.31	2.572	211.096358	8.106	0	False
1	50605.27	2010-05-02	1	42.31	2.572	211.096358	8.106	0	False
2	13740.12	2010-05-02	1	42.31	2.572	211.096358	8.106	0	False
3	39954.04	2010-05-02	1	42.31	2.572	211.096358	8.106	0	False
4	32229.38	2010-05-02	1	42.31	2.572	211.096358	8.106	0	False

1.1.9 Una vez que tenemos las features de interés de nuestro set de datos, aplicar algún método de normalización sobre los mismos, para evitar tener un sesgo de unas variables sobre otra (se pueden utilizar técnicas como z-score/min-max scaling). Guardar el dataset normalizado con un nombre representativo.

Out[76]:

	Weekly_Sales	Store	Temperature	Fuel_Price	CPI	Unemployment	IsHoliday
0	0.393812	-1.658153	-0.964108	-1.721081	1.018748	0.07831	-0.275085
1	1.524529	-1.658153	-0.964108	-1.721081	1.018748	0.07831	-0.275085
2	-0.098632	-1.658153	-0.964108	-1.721081	1.018748	0.07831	-0.275085
3	1.055559	-1.658153	-0.964108	-1.721081	1.018748	0.07831	-0.275085
4	0.715444	-1.658153	-0.964108	-1.721081	1.018748	0.07831	-0.275085

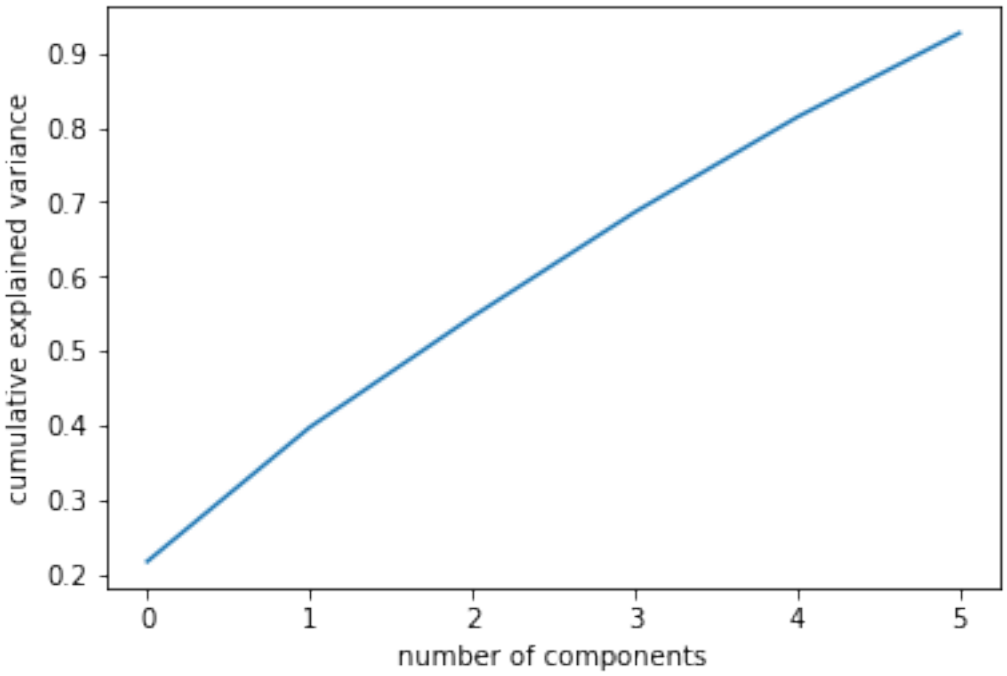
1.1.10 Finalmente, reducir los features de interés mediante sus combinaciones lineales (aplicando Principal Component Analysis).

Out[77]:
```
PCA(copy=True, iterated_power='auto', n_components=6, random_state=None,
  svd_solver='auto', tol=0.0, whiten=False)
```

```
shape of X_pca (425796, 6)
```

```
[0.21697279 0.18012888 0.14834528 0.140539   0.1276846  0.11297919]
```

```
suma: 0.5454469525068202
```



No vale la pena utilizar componentes principales ya que no hay 2 variables que representen toda la variacion sino que la misma esta muy distribuida

1.1.11 Guardar el dataset con nombre representativo.