

```
In [ ]: #Sprint 12. Machine learning avançat
```

```
In [ ]: #S12 T01: Pipelines, grid search i text mining
```

```
In [ ]: #Nivell 1
```

```
In [ ]: #Exercici 1  
#Agafa el conjunt de dades que vulguis i realitza un pipeline i un gridsearch aplicant l'algorisme de Random Forest.
```

```
In [803]: #Llibreries import  
import pandas as pd  
import numpy as np  
import sklearn.externals  
import joblib  
import nltk  
#import category_encoders as ce
```

```
In [804]: import matplotlib.pyplot as plt
```

```
In [838]: #Llibreries from
from sklearn import datasets
from sklearn import svm
from sklearn.feature_selection import SelectKBest
from sklearn.feature_selection import chi2
from sklearn.model_selection import train_test_split
from sklearn.compose import ColumnTransformer
from sklearn.impute import SimpleImputer
from sklearn.preprocessing import StandardScaler
from sklearn.ensemble import RandomForestRegressor
from sklearn.pipeline import Pipeline
from unidecode import unidecode
from sklearn.datasets import make_classification
from sklearn.svm import SVC
from sklearn.linear_model import Ridge
from sklearn.model_selection import GridSearchCV
from sklearn.metrics import accuracy_score
from sklearn.decomposition import PCA
from sklearn import preprocessing
from sklearn.pipeline import make_pipeline
from nltk.tokenize import word_tokenize
from nltk.probability import FreqDist
from nltk.corpus import stopwords
from nltk.stem import PorterStemmer
from sklearn.feature_extraction.text import CountVectorizer
from nltk.tokenize import RegexpTokenizer
```

```
In [651]: #Dataset de locals comercials de Barcelona entre els anys 2008 i 2011.
df_dades=pd.read_csv('Locals_comercials.csv', engine="python", error_bad_lines=False, warn_bad_lines=False, sep=',')
```

```
In [652]: df_dades.head(10)
```

Out[652]:

	i»¿DTE	BARRIS	DOSZEROONZE	DOSZERODEU	DOSZEROZERONOU	DOSZEROZEROVUIT
0	1	1. el Raval	2.176	2.445	2.773	3.065
1	1	2. el Barri Gàtic	2.632	3.026	3.750	3.828
2	1	3. la Barceloneta	2.135	3.005	--	--
3	1	4. Sant Pere Santa Caterina i la Ribera	2.482	2.828	3.146	3.450
4	2	5. el Fort Pienc	2.224	2.422	2.736	2.893
5	2	6. la Sagrada Família	2.267	2.490	2.535	2.677
6	2	7. la Dreta de l'Eixample	2.302	2.649	2.606	3.134
7	2	8. l'Antiga Esquerra de l'Eixample	2.471	3.006	3.068	3.583
8	2	9. la Nova Esquerra de l'Eixample	2.314	2.802	2.973	3.337
9	2	10. Sant Antoni	2.051	2.369	2.833	2.953

```
In [653]: df_dades.dtypes
```

Out[653]:

i»¿DTE	int64
BARRIS	object
DOSZEROONZE	object
DOSZERODEU	object
DOSZEROZERONOU	object
DOSZEROZEROVUIT	object
dtype:	object

```
In [654]: #Modifiquem els de columna per facilitar la interpretació  
df_dades.rename(columns={'DOSZEROONZE': 'LOC2011'}, inplace=True)  
df_dades.rename(columns={'DOSZERODEU': 'LOC2010'}, inplace=True)  
df_dades.rename(columns={'DOSZEROZERONOU': 'LOC2009'}, inplace=True)  
df_dades.rename(columns={'DOSZEROZEROVUIT': 'LOC2008'}, inplace=True)
```

```
In [655]: #Moltes instàncies tenen valor '--'.
#Filtrem aquestes instàncies del dataset de Locals Comercials.
df_dades[df_dades['LOC2008'].str[0] == '-']
```

Out[655]:

	Í»¿DTE	BARRIS	LOC2011	LOC2010	LOC2009	LOC2008
2	1	3. la Barceloneta	2.135	3.005	--	--
20	4	21. Pedralbes	--	--	--	--
21	5	22. Vallvidrera el Tibidabo i les Planes	--	--	--	--
35	7	36. la Font d'en Fargues	--	--	--	--
38	7	39. Sant GenÃs dels Agudells	--	--	--	--
39	7	40. Montbau	--	--	--	--
40	7	41. la Vall d'Hebron	--	--	--	--
41	7	42. la Clota	--	--	--	--
45	8	46. el TurÃ³ de la Peira	1.572	--	--	--
47	8	48. la Guineueta	--	--	--	--
48	8	49. Canyelles	--	1.828	--	--
52	8	53. la Trinitat Nova	--	--	--	--
53	8	54. Torre BarÃ³	--	--	--	--
54	8	55. Ciutat Meridiana	1.729	--	--	--
55	8	56. Vallbona	--	--	--	--
57	9	58. BarÃ³ de Viver	--	--	--	--
66	10	67. la Vila OlÃmpica del Poblenou	2.377	2.700	--	--

```
In [656]: #Valors -- de la columna LOC2011.
df_dades['LOC2011'] = df_dades['LOC2011'].str.replace('--', '0')#Marquem els valors -- com a valors 0.
#df_dades.drop(df_dades[df_dades['LOC2011'] == '--'].index, inplace = True)
```

```
In [657]: #Valors -- de la columna LOC2010.
df_dades['LOC2010'] = df_dades['LOC2010'].str.replace('--', '0')#Marquem els valors -- com a valors 0.
```

```
In [658]: #Valors -- de La columna LOC2009.  
df_dades['LOC2009'] = df_dades['LOC2009'].str.replace('--','0')#Marquem els valors -- com a valors 0.
```

```
In [659]: #Valors -- de La columna LOC2008.  
df_dades['LOC2008'] = df_dades['LOC2008'].str.replace('--','0')#Marquem els valors -- com a valors 0.
```

```
In [660]: df_dades[df_dades['LOC2010'].str[0] == '0']
```

Out[660]:

		ï»¿DTE	BARRIS	LOC2011	LOC2010	LOC2009	LOC2008
11	3	12. la Marina del Prat Vermell - Zona Franca		0	0	0	1.895
20	4		21. Pedralbes	0	0	0	0
21	5	22. Vallvidrera el Tibidabo i les Planes		0	0	0	0
35	7		36. la Font d'en Fargues	0	0	0	0
38	7		39. Sant GenÃs dels Agudells	0	0	0	0
39	7		40. Montbau	0	0	0	0
40	7		41. la Vall d'Hebron	0	0	0	0
41	7		42. la Clota	0	0	0	0
45	8		46. el TurÃ³ de la Peira	1.572	0	0	0
46	8		47. Can Peguera	0	0	0	2.936
47	8		48. la Guineueta	0	0	0	0
50	8		51. Verdun	1.471	0	0	1.746
52	8		53. la Trinitat Nova	0	0	0	0
53	8		54. Torre BarÃ³	0	0	0	0
54	8		55. Ciutat Meridiana	1.729	0	0	0
55	8		56. Vallbona	0	0	0	0
57	9		58. BarÃ³ de Viver	0	0	0	0

```
In [661]: #Columnes de Locals Comercials  
df_dades['LOC2011'] = df_dades['LOC2011'].str.replace('.','')#Esborrem el caracter '.'.  
df_dades['LOC2010'] = df_dades['LOC2010'].str.replace('.','')#Esborrem el caracter '.'.  
df_dades['LOC2009'] = df_dades['LOC2009'].str.replace('.','')#Esborrem el caracter '.'.  
df_dades['LOC2008'] = df_dades['LOC2008'].str.replace('.','')#Esborrem el caracter '.'.
```

```
In [662]: df_dades.head()
```

Out[662]:

	i»¿DTE	BARRIS	LOC2011	LOC2010	LOC2009	LOC2008
0	1	1. el Raval	2176	2445	2773	3065
1	1	2. el Barri Gàtic	2632	3026	3750	3828
2	1	3. la Barceloneta	2135	3005	0	0
3	1	4. Sant Pere Santa Caterina i la Ribera	2482	2828	3146	3450
4	2	5. el Fort Pienc	2224	2422	2736	2893

```
In [663]: df_dades.dtypes#Disposem de valors de tipus object.
```

Out[663]:

i»¿DTE	int64
BARRIS	object
LOC2011	object
LOC2010	object
LOC2009	object
LOC2008	object
dtype:	object

```
In [664]: df_dades['LOC2011'] = pd.to_numeric(df_dades['LOC2011'])#Transformem a valors de tipus numeric la columna LOC2011.
```

```
In [665]: df_dades['LOC2010'] = pd.to_numeric(df_dades['LOC2010'])#Transformem a valors de tipus numeric la columna LOC2010.
```

```
In [666]: df_dades['LOC2009'] = pd.to_numeric(df_dades['LOC2009'])#Transformem a valors de tipus numeric la columna LOC2009.
```

```
In [667]: df_dades['LOC2008'] = pd.to_numeric(df_dades['LOC2008'])#Transformem a valors de tipus numeric la columna LOC2008.
```

```
In [669]: df_dades.dtypes
```

```
Out[669]: i»_DTE      int64
BARRIS      object
LOC2011     int64
LOC2010     int64
LOC2009     int64
LOC2008     int64
dtype: object
```

```
In [670]: # Esborrem la columna 'i»_DTE' >> es una dada prescindible.
del df_dades["i»_DTE"]
```

```
In [671]: df_dades#Dins la columna barris hi ha valors concatenats. Els volem posar en columnes separades.
```

```
Out[671]:
```

	BARRIS	LOC2011	LOC2010	LOC2009	LOC2008
0	1. el Raval	2176	2445	2773	3065
1	2. el Barri Gàtic	2632	3026	3750	3828
2	3. la Barceloneta	2135	3005	0	0
3	4. Sant Pere Santa Caterina i la Ribera	2482	2828	3146	3450
4	5. el Fort Pienc	2224	2422	2736	2893
...	...	...	...	...	...
68	69. Diagonal Mar i el Front Marítim del Poblenou	2651	2991	2445	3752
69	70. el Besòs i el Maresme	2121	2751	2842	3669
70	71. Provençals del Poblenou	2255	2516	2660	3857
71	72. Sant Martí de Provençals	1812	2175	2266	1957
72	73. la Verneda i la Pau	1534	1915	1973	2199

73 rows × 5 columns

```
In [681]: #Afegim una columna que contingui la posició de barri a la ciutat de Barcelona.  
df_dades['DISTRICTE'] = df_dades['BARRIS'].str.split('.',expand=True)#Escollem el nom de DISTRICTE per la nova columna.
```

```
In [682]: df_dades
```

```
Out[682]:
```

	BARRIS	LOC2011	LOC2010	LOC2009	LOC2008	DISTRICTE
0	1. el Raval	2176	2445	2773	3065	1
1	2. el Barri Gàtic	2632	3026	3750	3828	2
2	3. la Barceloneta	2135	3005	0	0	3
3	4. Sant Pere Santa Caterina i la Ribera	2482	2828	3146	3450	4
4	5. el Fort Pienc	2224	2422	2736	2893	5
...	...	...	...	...	...	...
68	69. Diagonal Mar i el Front Marítim del Poblenou	2651	2991	2445	3752	69
69	70. el Besòs i el Maresme	2121	2751	2842	3669	70
70	71. Provençals del Poblenou	2255	2516	2660	3857	71
71	72. Sant Martí de Provençals	1812	2175	2266	1957	72
72	73. la Verneda i la Pau	1534	1915	1973	2199	73

73 rows × 6 columns

```
In [683]: df_dades['DISTRICTE'] = pd.to_numeric(df_dades['DISTRICTE'])#Transformem a valors de tipus numeric la columna DISTRICTE.
```

```
In [684]: df_dades['DISTRICTE'].dtypes
```

```
Out[684]: dtype('int64')
```

```
In [685]: #Afegim una columna que contingui la denominació del barri de Barcelona.  
df_dades['BARRI'] = df_dades['BARRIS'].str.replace('.','')#Esborrem el caracter '.'.  
df_dades['BARRI'] = df_dades['BARRI'].str.replace('[0-9]','')#Esborrem els numeros.
```

```
In [686]: df_dades['BARRI']=df_dades['BARRI'].str.upper() #Posem el nom dels Barris en Majuscula.
```

```
In [687]: #Manegem una instrucció per treure els accENTS dels noms dels Barris de Barcelona.  
df_dades['BARRI'] = df_dades['BARRI'].str.normalize('NFKD').str.encode('ascii', errors='ignore').str.decode('utf-8')
```

```
In [688]: df_dades
```

```
Out[688]:
```

	BARRIS	LOC2011	LOC2010	LOC2009	LOC2008	DISTRICTE	BARRI
0	1. el Raval	2176	2445	2773	3065	1	EL RAVAL
1	2. el Barri GÀtic	2632	3026	3750	3828	2	EL BARRI GA2TIC
2	3. la Barceloneta	2135	3005	0	0	3	LA BARCELONETA
3	4. Sant Pere Santa Caterina i la Ribera	2482	2828	3146	3450	4	SANT PERE SANTA CATERINA I LA RIBERA
4	5. el Fort Pienc	2224	2422	2736	2893	5	EL FORT PIENC
...	...	...	...	...	...	...	...
68	69. Diagonal Mar i el Front Marítim del Poblenou	2651	2991	2445	3752	69	DIAGONAL MAR I EL FRONT MARITIM DEL POBLENOU
69	70. el Besòs i el Maresme	2121	2751	2842	3669	70	EL BESÒS I EL MARESME
70	71. Provençals del Poblenou	2255	2516	2660	3857	71	PROVENÇALS DEL POBLENOU
71	72. Sant Martí de Provençals	1812	2175	2266	1957	72	SANT MARTÍ DE PROVENÇALS
72	73. la Verneda i la Pau	1534	1915	1973	2199	73	LA VERNEDA I LA PAU

73 rows × 7 columns

```
In [689]: # Esborrem la columna 'BARRIS' >> hem dividit el contingut de 'BARRIS' en 2 columnes. Ja no la necessitem.  
del df_dades["BARRIS"]
```

```
In [690]: df_dades
```

```
Out[690]:
```

	LOC2011	LOC2010	LOC2009	LOC2008	DISTRICTE	BARRI
0	2176	2445	2773	3065	1	EL RAVAL
1	2632	3026	3750	3828	2	EL BARRI GA2TIC
2	2135	3005	0	0	3	LA BARCELONETA
3	2482	2828	3146	3450	4	SANT PERE SANTA CATERINA I LA RIBERA
4	2224	2422	2736	2893	5	EL FORT PIENC
...	...	...	...	...	...	...
68	2651	2991	2445	3752	69	DIAGONAL MAR I EL FRONT MARATIM DEL POBLENOU
69	2121	2751	2842	3669	70	EL BESA2S I EL MARESME
70	2255	2516	2660	3857	71	PROVENAALS DEL POBLENOU
71	1812	2175	2266	1957	72	SANT MARTA DE PROVENAALS
72	1534	1915	1973	2199	73	LA VERNEDA I LA PAU

73 rows × 6 columns

```
In [691]: #Dividim el dataset en train i test.
```

```
df_loc = df_dades[['LOC2011','LOC2010','LOC2009','LOC2008']]#filtrem només les dades que fan referència als locals Comercials.
```

```
X_train, X_test, y_train, y_test = train_test_split(df_loc,df_dades['DISTRICTE'],random_state=0)
```

```
print(X_train.shape, X_test.shape, y_train.shape, y_test.shape)
```

```
(54, 4) (19, 4) (54,) (19,)
```

```
In [692]: #Creem una pipeline
```

```
pipeline = make_pipeline(preprocessing.StandardScaler(), RandomForestRegressor(n_estimators=100))
```

```
print(type(pipeline))
```

```
<class 'sklearn.pipeline.Pipeline'>
```

```
In [693]: pipeline.fit(X_train, y_train)
```

```
Out[693]: Pipeline(steps=[('standardscaler', StandardScaler()), ('randomforestregressor', RandomForestRegressor())])
```

```
In [694]: predicted = pipeline.predict(X_test)  
predicted
```

```
Out[694]: array([39.12      , 39.88515798, 22.18      , 49.12      , 36.95      ,  
   39.51      , 43.11      , 28.97544733, 50.57      , 21.34      ,  
   57.2       , 39.88515798, 39.88515798, 50.3       , 42.39      ,  
   31.38      , 37.65      , 34.64      , 39.46      ])
```

```
In [695]: pipeline.score(X_test, y_test).round(3)
```

```
Out[695]: -0.076
```

```
In [902]: pipeline.get_params()#Visualització dels parametres del Pipeline.
```

```
Out[902]: {'memory': None,
 'steps': [('standardscaler', StandardScaler()),
 ('randomforestregressor', RandomForestRegressor())],
 'verbose': False,
 'standardscaler': StandardScaler(),
 'randomforestregressor': RandomForestRegressor(),
 'standardscaler_copy': True,
 'standardscaler_with_mean': True,
 'standardscaler_with_std': True,
 'randomforestregressor_bootstrap': True,
 'randomforestregressor_ccp_alpha': 0.0,
 'randomforestregressor_criterion': 'squared_error',
 'randomforestregressor_max_depth': None,
 'randomforestregressor_max_features': 'auto',
 'randomforestregressor_max_leaf_nodes': None,
 'randomforestregressor_max_samples': None,
 'randomforestregressor_min_impurity_decrease': 0.0,
 'randomforestregressor_min_samples_leaf': 1,
 'randomforestregressor_min_samples_split': 2,
 'randomforestregressor_min_weight_fraction_leaf': 0.0,
 'randomforestregressor_n_estimators': 100,
 'randomforestregressor_n_jobs': None,
 'randomforestregressor_oob_score': False,
 'randomforestregressor_random_state': None,
 'randomforestregressor_verbose': 0,
 'randomforestregressor_warm_start': False}
```

```
In [ ]:
```

```
In [698]: #Gridsearch
```

```
In [1023]: from sklearn.ensemble import RandomForestClassifier
from sklearn.datasets import make_classification
from sklearn.compose import ColumnTransformer
from sklearn.preprocessing import StandardScaler
```

```
In [1083]: X = df_loc  
y = df_dades[['DISTRICTE']]  
  
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=1)
```

```
In [1084]: parametes_grid = {  
    'randomforestregressor__max_depth': [10, 20, 30, 40],  
    'randomforestregressor__max_features': [1, 2, 3],  
    'randomforestregressor__max_leaf_nodes': [2],  
    'randomforestregressor__max_samples': [1, 20, 40],  
    'randomforestregressor__min_samples_leaf': [1,2,3],  
    'randomforestregressor__min_samples_split': [2,3,4],  
    'randomforestregressor__n_estimators': [50, 100, 150],  
    'randomforestregressor__n_jobs': [1,2,3],  
    'randomforestregressor__random_state': [1,2,3]  
} #Disposem uns valors per cadascu dels atributs referits.
```

```
In [1085]: model = Ridge()#Optem pel model Ridge.  
grid = GridSearchCV(pipeline, param_grid=parametes_grid, n_jobs=4, cv=5)
```

```
In [1086]: grid
```

```
Out[1086]: GridSearchCV(cv=5,  
                        estimator=Pipeline(steps=[('standardscaler', StandardScaler()),  
                                              ('randomforestregressor',  
                                               RandomForestRegressor())]),  
                        n_jobs=4,  
                        param_grid={'randomforestregressor__max_depth': [10, 20, 30, 40],  
                                    'randomforestregressor__max_features': [1, 2, 3],  
                                    'randomforestregressor__max_leaf_nodes': [2],  
                                    'randomforestregressor__max_samples': [1, 20, 40],  
                                    'randomforestregressor__min_samples_leaf': [1, 2, 3],  
                                    'randomforestregressor__min_samples_split': [2, 3, 4],  
                                    'randomforestregressor__n_estimators': [50, 100, 150],  
                                    'randomforestregressor__n_jobs': [1, 2, 3],  
                                    'randomforestregressor__random_state': [1, 2, 3]})
```

```
In [1088]: grid.fit(X_train, y_train)
```

```
C:\Users\Xavier\anaconda3\lib\site-packages\sklearn\pipeline.py:394: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples,), for example using ravel().
  self._final_estimator.fit(Xt, y, **fit_params_last_step)
```

```
Out[1088]: GridSearchCV(cv=5,
                        estimator=Pipeline(steps=[('standardscaler', StandardScaler()),
                                                  ('randomforestregressor',
                                                   RandomForestRegressor())]),
                        n_jobs=4,
                        param_grid={'randomforestregressor__max_depth': [10, 20, 30, 40],
                                    'randomforestregressor__max_features': [1, 2, 3],
                                    'randomforestregressor__max_leaf_nodes': [2],
                                    'randomforestregressor__max_samples': [1, 20, 40],
                                    'randomforestregressor__min_samples_leaf': [1, 2, 3],
                                    'randomforestregressor__min_samples_split': [2, 3, 4],
                                    'randomforestregressor__n_estimators': [50, 100, 150],
                                    'randomforestregressor__n_jobs': [1, 2, 3],
                                    'randomforestregressor__random_state': [1, 2, 3]})
```

```
In [1077]: print("Score = %3.2f" %(grid.score(X_test,y_test)))
```

```
Score = -0.24
```

```
In [1076]: print("Best params:\n", grid.best_params_)
```

```
Best params:
{'randomforestregressor__max_depth': 30, 'randomforestregressor__max_features': 3, 'randomforestregressor__max_leaf_nodes': 2, 'randomforestregressor__max_samples': 20, 'randomforestregressor__min_samples_leaf': 1, 'randomforestregressor__min_samples_split': 3, 'randomforestregressor__n_estimators': 50, 'randomforestregressor__n_jobs': 2, 'randomforestregressor__random_state': 1}
```

```
In [1074]: # summarize the results of the grid search
print("Best Score:", grid.best_score_)
```

```
Best Score: 0.07567872576178991
```

```
In [1070]: gs_predicted = grid.predict(X_test)
```

```
In [1071]: gs_predicted
```

```
Out[1071]: array([17.99111255, 45.12883849, 42.96919564, 22.99925144, 44.23212132,
 29.15899045, 45.12883849, 45.12883849, 42.5159193 , 43.1148623 ,
 42.6759193 , 45.12883849, 38.11373496, 45.12883849, 42.5159193 ,
 45.12883849, 40.27648496, 43.1919193 , 45.12883849, 20.69252922,
 42.78319564, 41.9999193 ])
```

```
In [1073]: print("Score:", grid.score(X_test, y_test).round(3))
```

```
Score: -0.235
```

```
In [ ]: #Exercici 2
#Agafa un text en anglès que vulguis, i calcula'n la freqüència de les paraules.
```

```
In [ ]: import nltk
nltk.download('popular')#Installem la llibreria nltk per utilitzar les funcions de tokenització.
```

In [ ]: #Book: A Christmas Carol BEING A GHOST STORY OF CHRISTMAS STAVE ONE MARLEY'S GHOST by Charles Dickens.

#Marley was dead, to begin with. There is no doubt whatever about that. The register of his burial was signed by the clergyman, the clerk, the undertaker, and the chief mourner. Scrooge signed it. And Scrooge's name was good upon 'Change for anything he chose to put his hand to. Old Marley was as dead as a door-nail.

#Mind! I don't mean to say that I know, of my own knowledge, what there is particularly dead about a door-nail. I might have been inclined, myself, to regard a coffin-nail as the deadeast piece of ironmongery in the trade. But the wisdom of our ancestors is in the simile; and my unhallowed hands shall not disturb it, or the Country's done for. You will, therefore, permit me to repeat, emphatically, that Marley was as dead as a door-nail.

#Scrooge knew he was dead? Of course he did. How could it be otherwise? Scrooge and he were partners for I don't know how many years. Scrooge was his sole executor, his sole[12] administrator, his sole assign, his sole residuary legatee, his sole friend, and sole mourner. And even Scrooge was not so dreadfully cut up by the sad event, but that he was an excellent man of business on the very day of the funeral, and solemnised it with an undoubted bargain.

#The mention of Marley's funeral brings me back to the point I started from. There is no doubt that Marley was dead. This must be distinctly understood, or nothing wonderful can come of the story I am going to relate. If we were not perfectly convinced that Hamlet's Father died before the play began, there would be nothing more remarkable in his taking a stroll at night, in an easterly wind, upon his own ramparts, than there would be in any other middle-aged gentleman rashly turning out after dark in a breezy spot—say St. Paul's Church-yard, for instance—literally to astonish his son's weak mind.

#Scrooge never painted out Old Marley's name. There it stood, years afterwards, above the warehouse door: Scrooge and Marley. The firm was known as Scrooge and Marley. Sometimes people new to the business called Scrooge Scrooge, and sometimes Marley, but he answered to both names. It was all the same to him.

In [806]: text = "Marley was dead, to begin with. There is no doubt whatever about that. The register of his burial was signed by the clergyman, the clerk, the undertaker, and the chief mourner. Scrooge signed it. And Scrooge's name was good upon 'Change for anything he chose to put his hand to. Old Marley was as dead as a door-nail. Mind! I don't mean to say that I know, of my own knowledge, what there is particularly dead about a door-nail. I might have been inclined, myself, to regard a coffin-nail as the deadest piece of ironmongery in the trade. But the wisdom of our ancestors is in the smile; and my unhallowed hands shall not disturb it, or the Country's done for. You will, therefore, permit me to repeat, emphatically, that Marley was as dead as a door-nail. Scrooge knew he was dead? Of course he did. How could it be otherwise? Scrooge and he were partners for I don't know how many years. Scrooge was his sole executor, his sole[12] administrator, his sole assign, his sole residuary legatee, his sole friend, and sole mourner. And even Scrooge was not so dreadfully cut up by the sad event, but that he was an excellent man of business on the very day of the funeral, and solemnised it with an undoubted bargain. The mention of Marley's funeral brings me back to the point I started from. There is no doubt that Marley was dead. This must be distinctly understood, or nothing wonderful can come of the story I am going to relate. If we were not perfectly convinced that Hamlet's Father died before the play began, there would be nothing more remarkable in his taking a stroll at night, in an easterly wind, upon his own ramparts, than there would be in any other middle-aged gentleman rashly turning out after dark in a breezy spot—say St. Paul's Church-yard, for instance—literally to astonish his son's weak mind. Scrooge never painted out Old Marley's name. There it stood, years afterwards, above the warehouse door: Scrooge and Marley. The firm was known as Scrooge and Marley. Sometimes people new to the business called Scrooge Scrooge, and sometimes Marley, but he answered to both names. It was all the same to him."

```
In [807]: text
```

Out[807]: "Marley was dead, to begin with. There is no doubt whatever about that. The register of his burial was signed by the clergyman, the clerk, the undertaker, and the chief mourner. Scrooge signed it. And Scrooge's name was good upon 'Change for anything he chose to put his hand to. Old Marley was as dead as a door-nail. Mind! I don't mean to say that I know, of my own knowledge, what there is particularly dead about a door-nail. I might have been inclined, myself, to regard a coffin-nail as the deadest piece of ironmongery in the trade. But the wisdom of our ancestors is in the simile; and my unhallowed hands shall not disturb it, or the Country's done for. You will, therefore, permit me to repeat, emphatically, that Marley was as dead as a door-nail. Scrooge knew he was dead? Of course he did. How could it be otherwise? Scrooge and he were partners for I don't know how many years. Scrooge was his sole executor, his sole[12] administrator, his sole assign, his sole residuary legatee, his sole friend, and sole mourner. And even Scrooge was not so dreadfully cut up by the sad event, but that he was an excellent man of business on the very day of the funeral, and solemnised it with an undoubted bargain. The mention of Marley's funeral brings me back to the point I started from. There is no doubt that Marley was dead. This must be distinctly understood, or nothing wonderful can come of the story I am going to relate. If we were not perfectly convinced that Hamlet's Father died before the play began, there would be nothing more remarkable in his taking a stroll at night, in an easterly wind, upon his own ramparts, than there would be in any other middle-aged gentleman rashly turning out after dark in a breezy spot—say St. Paul's Church-yard, for instance—literally to astonish his son's weak mind. Scrooge never painted out Old Marley's name. There it stood, years afterwards, above the warehouse door: Scrooge and Marley. The firm was known as Scrooge and Marley. Sometimes people new to the business called Scrooge Scrooge, and sometimes Marley, but he answered to both names. It was all the same to him."

```
In [808]: token = nltk.word_tokenize(text)#Transformem l'estructura del text. Disposem les paraules en unitats.
```

In [809]: token

```
Out[809]: ['Marley',
 'was',
 'dead',
 ',',
 ',',
 'to',
 'begin',
 'with',
 '..',
 'There',
 'is',
 'no',
 'doubt',
 'whatever',
 'about',
 'that',
 '..',
 'The',
 'register',
 'of',
 'his',
 'burial',
 'was',
 'signed',
 'by',
 'the',
 'clergyman',
 ',',
 'the',
 'clerk',
 ',',
 'the',
 'undertaker',
 ',',
 'and',
 'the',
 'chief',
 'mourner',
 '..',
 'Scrooge',
 'signed',
 'it',
```

' . ',  
'And',  
'Scrooge',  
"'s",  
'name',  
'was',  
'good',  
'upon',  
"'Change",  
'for',  
'anything',  
'he',  
'chose',  
'to',  
'put',  
'his',  
'hand',  
'to',  
' . ',  
'Old',  
'Marley',  
'was',  
'as',  
'dead',  
'as',  
'a',  
'door-nail',  
' . ',  
'Mind',  
'!',  
'I',  
'do',  
"'n't",  
'mean',  
'to',  
'say',  
'that',  
'I',  
'know',  
' , ',  
'of',  
'my',

'own',  
'knowledge',  
'',  
'what',  
'there',  
'is',  
'particularly',  
'dead',  
'about',  
'a',  
'door-nail',  
'.',  
'I',  
'might',  
'have',  
'been',  
'inclined',  
'',  
'myself',  
'',  
'to',  
'regard',  
'a',  
'coffin-nail',  
'as',  
'the',  
'deadest',  
'piece',  
'of',  
'ironmongery',  
'in',  
'the',  
'trade',  
'.',  
'But',  
'the',  
'wisdom',  
'of',  
'our',  
'ancestors',  
'is',  
'in',

'the',  
'simile',  
';',  
'and',  
'my',  
'unhallowed',  
'hands',  
'shall',  
'not',  
'disturb',  
'it',  
'',  
'or',  
'the',  
'Country',  
"'s",  
'done',  
'for',  
'..',  
'You',  
'will',  
'',  
'therefore',  
'',  
'permit',  
'me',  
'to',  
'repeat',  
'',  
'emphatically',  
'',  
'that',  
'Marley',  
'was',  
'as',  
'dead',  
'as',  
'a',  
'door-nail',  
'..',  
'Scrooge',  
'knew',

'he',  
'was',  
'dead',  
'? ',  
'Of',  
'course',  
'he',  
'did',  
'. ',  
'How',  
'could',  
'it',  
'be',  
'otherwise',  
'? ',  
'Scrooge',  
'and',  
'he',  
'were',  
'partners',  
'for',  
'I',  
'do',  
"n't",  
'know',  
'how',  
'many',  
'years',  
. ',  
'Scrooge',  
'was',  
'his',  
'sole',  
'executor',  
, ',  
'his',  
'sole',  
'[ ',  
'12',  
'] ',  
'administrator',  
, ',

'his',  
'sole',  
'assign',  
'',  
'his',  
'sole',  
'residuary',  
'legatee',  
'',  
'his',  
'sole',  
'friend',  
'',  
'and',  
'sole',  
'mourner',  
'.',  
'And',  
'even',  
'Scrooge',  
'was',  
'not',  
'so',  
'dreadfully',  
'cut',  
'up',  
'by',  
'the',  
'sad',  
'event',  
'',  
'but',  
'that',  
'he',  
'was',  
'an',  
'excellent',  
'man',  
'of',  
'business',  
'on',  
'the',

'very',  
'day',  
'of',  
'the',  
'funeral',  
'',  
'and',  
'solemnised',  
'it',  
'with',  
'an',  
'undoubted',  
'bargain',  
'.',  
'The',  
'mention',  
'of',  
'Marley',  
"'s",  
'funeral',  
'brings',  
'me',  
'back',  
'to',  
'the',  
'point',  
'I',  
'started',  
'from',  
'..',  
'There',  
'is',  
'no',  
'doubt',  
'that',  
'Marley',  
'was',  
'dead',  
'..',  
'This',  
'must',  
'be',

'distinctly',  
'understood',  
'',  
'or',  
'nothing',  
'wonderful',  
'can',  
'come',  
'of',  
'the',  
'story',  
'I',  
'am',  
'going',  
'to',  
'relate',  
'.',  
'If',  
'we',  
'were',  
'not',  
'perfectly',  
'convinced',  
'that',  
'Hamlet',  
"'s",  
'Father',  
'died',  
'before',  
'the',  
'play',  
'began',  
'',  
'there',  
'would',  
'be',  
'nothing',  
'more',  
'remarkable',  
'in',  
'his',  
'taking',

'a',  
'stroll',  
'at',  
'night',  
'',  
'in',  
'an',  
'easterly',  
'wind',  
'',  
'upon',  
'his',  
'own',  
'ramparts',  
'',  
'than',  
'there',  
'would',  
'be',  
'in',  
'any',  
'other',  
'middle-aged',  
'gentleman',  
'rashly',  
'turning',  
'out',  
'after',  
'dark',  
'in',  
'a',  
'breezy',  
'spot-say',  
'St.',  
'Paul',  
"'s",  
'Church-yard',  
'',  
'for',  
'instance-literally',  
'to',  
'astonish',

'his',  
'son',  
"'s",  
'weak',  
'mind',  
. ,  
'Scrooge',  
'never',  
'painted',  
'out',  
'Old',  
'Marley',  
"'s",  
'name',  
. ,  
'There',  
'it',  
'stood',  
. ,  
'years',  
'afterwards',  
. ,  
'above',  
'the',  
'warehouse',  
'door',  
. : ,  
'Scrooge',  
'and',  
'Marley',  
. . ,  
'The',  
'firm',  
'was',  
'known',  
'as',  
'Scrooge',  
'and',  
'Marley',  
. . ,  
'Sometimes',  
'people',

```
'new',
'to',
'the',
'business',
'called',
'Scrooge',
'Scrooge',
',',
'and',
'sometimes',
'Marley',
',',
'but',
'he',
'answered',
'to',
'both',
'names',
'.',
'It',
'was',
'all',
'the',
'same',
'to',
'him',
'..']
```

In [810]: #Esborrem del text signes de puntuació (que apareixen com a paraules).  
token = [word for word in token if word.isalnum()]

```
In [811]: print('Paraules del text sense signes de puntuació:\n', token)
```

Paraules del text sense signes de puntuació:

```
['Marley', 'was', 'dead', 'to', 'begin', 'with', 'There', 'is', 'no', 'doubt', 'whatever', 'about', 'that', 'The', 'register', 'of', 'his', 'burial', 'was', 'signed', 'by', 'the', 'clergyman', 'the', 'clerk', 'the', 'undertaker', 'and', 'the', 'chief', 'mourner', 'Scrooge', 'signed', 'it', 'And', 'Scrooge', 'name', 'was', 'good', 'upon', 'for', 'anything', 'he', 'chose', 'to', 'put', 'his', 'hand', 'to', 'Old', 'Marley', 'was', 'as', 'dead', 'as', 'a', 'Mind', 'I', 'do', 'mean', 'to', 'say', 'that', 'I', 'know', 'of', 'my', 'own', 'knowledge', 'what', 'there', 'is', 'particularly', 'dead', 'about', 'a', 'I', 'might', 'have', 'been', 'inclined', 'myself', 'to', 'regard', 'a', 'as', 'the', 'deadliest', 'piece', 'of', 'ironmongery', 'in', 'the', 'trade', 'But', 'the', 'wisdom', 'of', 'our', 'ancestors', 'is', 'in', 'the', 'simile', 'and', 'my', 'unhallowed', 'hands', 'shall', 'not', 'disturb', 'it', 'or', 'the', 'Country', 'done', 'for', 'You', 'will', 'therefore', 'permit', 'me', 'to', 'repeat', 'emphatically', 'that', 'Marley', 'was', 'as', 'dead', 'as', 'a', 'Scrooge', 'knew', 'he', 'was', 'dead', 'Of', 'course', 'he', 'did', 'How', 'could', 'it', 'be', 'otherwise', 'Scrooge', 'and', 'he', 'were', 'partners', 'for', 'I', 'do', 'know', 'how', 'many', 'years', 'Scrooge', 'was', 'his', 'sole', 'executor', 'his', 'sole', '12', 'administrator', 'his', 'sole', 'assign', 'his', 'sole', 'residuary', 'legatee', 'his', 'sole', 'friend', 'and', 'sole', 'mourner', 'And', 'even', 'Scrooge', 'was', 'not', 'so', 'dreadfully', 'cut', 'up', 'by', 'the', 'sad', 'event', 'but', 'that', 'he', 'was', 'an', 'excellent', 'man', 'of', 'business', 'on', 'the', 'very', 'day', 'of', 'the', 'funeral', 'and', 'solemnised', 'it', 'with', 'an', 'undoubted', 'bargain', 'The', 'mention', 'of', 'Marley', 'funeral', 'brings', 'me', 'back', 'to', 'the', 'point', 'I', 'started', 'from', 'There', 'is', 'no', 'doubt', 'that', 'Marley', 'was', 'dead', 'This', 'must', 'be', 'distinctly', 'understood', 'or', 'nothing', 'wonderful', 'can', 'come', 'of', 'the', 'story', 'I', 'am', 'going', 'to', 'relate', 'If', 'we', 'were', 'not', 'perfectly', 'convinced', 'that', 'Hamlet', 'Father', 'died', 'before', 'the', 'play', 'began', 'there', 'would', 'be', 'nothing', 'more', 'remarkable', 'in', 'his', 'taking', 'a', 'stroll', 'at', 'night', 'in', 'an', 'easterly', 'wind', 'upon', 'his', 'own', 'ramparts', 'than', 'there', 'would', 'be', 'in', 'any', 'other', 'gentleman', 'rashly', 'turning', 'out', 'after', 'dark', 'in', 'a', 'breezy', 'Paul', 'for', 'to', 'astonish', 'his', 'son', 'weak', 'mind', 'Scrooge', 'never', 'painted', 'out', 'Old', 'Marley', 'name', 'There', 'it', 'stood', 'years', 'afterwards', 'above', 'the', 'warehouse', 'door', 'Scrooge', 'and', 'Marley', 'The', 'firm', 'was', 'know', 'as', 'Scrooge', 'and', 'Marley', 'Sometimes', 'people', 'new', 'to', 'the', 'business', 'called', 'Scrooge', 'Scrooge', 'and', 'sometimes', 'Marley', 'but', 'he', 'answered', 'to', 'both', 'names', 'It', 'was', 'all', 'the', 'same', 'to', 'him']
```

```
In [812]: mot_freq = FreqDist(token) #Ordenació i/o Comptabilització dels mots existents al text.
```

```
print('Les 10 paraules més repetides al text de Charles Dickens:')
```

```
for i in mot_freq.most_common(10):
```

```
    print('La paraula', i[0], 'es repeteix', i[1], 'vegades.');
```

Les 10 paraules més repetides al text de Charles Dickens:

La paraula the es repeteix 18 vegades.  
La paraula was es repeteix 12 vegades.  
La paraula to es repeteix 12 vegades.  
La paraula Scrooge es repeteix 11 vegades.  
La paraula his es repeteix 10 vegades.  
La paraula Marley es repeteix 9 vegades.  
La paraula of es repeteix 8 vegades.  
La paraula and es repeteix 8 vegades.  
La paraula dead es repeteix 6 vegades.  
La paraula that es repeteix 6 vegades.

```
In [813]: #Nivell 2
```

```
In [814]: #Exercici 1
```

```
#Treu les stopwords i realitza stemming al teu conjunt de dades.
```

In [815]: token

```
Out[815]: ['Marley',
 'was',
 'dead',
 'to',
 'begin',
 'with',
 'There',
 'is',
 'no',
 'doubt',
 'whatever',
 'about',
 'that',
 'The',
 'register',
 'of',
 'his',
 'burial',
 'was',
 'signed',
 'by',
 'the',
 'clergyman',
 'the',
 'clerk',
 'the',
 'undertaker',
 'and',
 'the',
 'chief',
 'mourner',
 'Scrooge',
 'signed',
 'it',
 'And',
 'Scrooge',
 'name',
 'was',
 'good',
 'upon',
 'for',
```

'anything',  
'he',  
'chose',  
'to',  
'put',  
'his',  
'hand',  
'to',  
'Old',  
'Marley',  
'was',  
'as',  
'dead',  
'as',  
'a',  
'Mind',  
'I',  
'do',  
'mean',  
'to',  
'say',  
'that',  
'I',  
'know',  
'of',  
'my',  
'own',  
'knowledge',  
'what',  
'there',  
'is',  
'particularly',  
'dead',  
'about',  
'a',  
'I',  
'might',  
'have',  
'been',  
'inclined',  
'myself',  
'to',

'regard',  
'a',  
'as',  
'the',  
'deadest',  
'piece',  
'of',  
'ironmongery',  
'in',  
'the',  
'trade',  
'But',  
'the',  
'wisdom',  
'of',  
'our',  
'ancestors',  
'is',  
'in',  
'the',  
'simile',  
'and',  
'my',  
'unhallowed',  
'hands',  
'shall',  
'not',  
'disturb',  
'it',  
'or',  
'the',  
'Country',  
'done',  
'for',  
'You',  
'will',  
'therefore',  
'permit',  
'me',  
'to',  
'repeat',  
'emphatically',

'that',  
'Marley',  
'was',  
'as',  
'dead',  
'as',  
'a',  
'Scrooge',  
'knew',  
'he',  
'was',  
'dead',  
'Of',  
'course',  
'he',  
'did',  
'How',  
'could',  
'it',  
'be',  
'otherwise',  
'Scrooge',  
'and',  
'he',  
'were',  
'partners',  
'for',  
'I',  
'do',  
'know',  
'how',  
'many',  
'years',  
'Scrooge',  
'was',  
'his',  
'sole',  
'executor',  
'his',  
'sole',  
'12',  
'administrator',

'his',  
'sole',  
'assign',  
'his',  
'sole',  
'residuary',  
'legatee',  
'his',  
'sole',  
'friend',  
'and',  
'sole',  
'mourner',  
'And',  
'even',  
'Scrooge',  
'was',  
'not',  
'so',  
'dreadfully',  
'cut',  
'up',  
'by',  
'the',  
'sad',  
'event',  
'but',  
'that',  
'he',  
'was',  
'an',  
'excellent',  
'man',  
'of',  
'business',  
'on',  
'the',  
'very',  
'day',  
'of',  
'the',  
'funeral',

'and',  
'solemnised',  
'it',  
'with',  
'an',  
'undoubted',  
'bargain',  
'The',  
'mention',  
'of',  
'Marley',  
'funeral',  
'brings',  
'me',  
'back',  
'to',  
'the',  
'point',  
'I',  
'started',  
'from',  
'There',  
'is',  
'no',  
'doubt',  
'that',  
'Marley',  
'was',  
'dead',  
'This',  
'must',  
'be',  
'distinctly',  
'understood',  
'or',  
'nothing',  
'wonderful',  
'can',  
'come',  
'of',  
'the',  
'story',

'I',  
'am',  
'going',  
'to',  
'relate',  
'If',  
'we',  
'were',  
'not',  
'perfectly',  
'convinced',  
'that',  
'Hamlet',  
'Father',  
'died',  
'before',  
'the',  
'play',  
'began',  
'there',  
'would',  
'be',  
'nothing',  
'more',  
'remarkable',  
'in',  
'his',  
'taking',  
'a',  
'stroll',  
'at',  
'night',  
'in',  
'an',  
'easterly',  
'wind',  
'upon',  
'his',  
'own',  
'ramparts',  
'than',  
'there',

'would',  
'be',  
'in',  
'any',  
'other',  
'gentleman',  
'rashly',  
'turning',  
'out',  
'after',  
'dark',  
'in',  
'a',  
'breezy',  
'Paul',  
'for',  
'to',  
'astonish',  
'his',  
'son',  
'weak',  
'mind',  
'Scrooge',  
'never',  
'painted',  
'out',  
'Old',  
'Marley',  
'name',  
'There',  
'it',  
'stood',  
'years',  
'afterwards',  
'above',  
'the',  
'warehouse',  
'door',  
'Scrooge',  
'and',  
'Marley',  
'The',

```
'firm',
'was',
'known',
'as',
'Scrooge',
'and',
'Marley',
'Sometimes',
'people',
'new',
'to',
'the',
'business',
'called',
'Scrooge',
'Scrooge',
'and',
'sometimes',
'Marley',
'but',
'he',
'answered',
'to',
'both',
'names',
'It',
'was',
'all',
'the',
'same',
'to',
'him']
```

```
In [816]: stop_words = set(stopwords.words('english'))
```

```
In [817]: print(stop_words)#Visualitzem el catalèg de "stop words".
```

```
{'at', "didn't", "isn't", 'been', 'myself', "don't", 'in', 'than', "should've", 'why', 'once', 'as', 'yours', 'those', 'both', 'can', 'not', 'such', 'yourselves', 'itself', 'over', "aren't", 'down', "wouldn't", 'too', 'against', "you've", 'here', 'her', 're', "mightn't", 'they', 'more', 'we', 'he', 'should', "you'll", 'to', 'couldn', 'where', 'while', 'few', 'm', 'with', "weren't", 'below', 'for', "shouldn't", 'was', 'its', 'what', 'you', 'hasn', 'my', 'do', 'so', 'don', 'or', 'o', 'your', 'between', 's', 'haven', 'd', 'themselves', 'our', 'wouldn', 'doesn', "shan't", 'didn', 'very', 'before', 'again', 'yourself', 'them', 'any', 'who', 'has', 'being', 'off', 'during', 'shan', 'i', 'their', 've', 'from', 'his', "won't", "hasn't", 'all', "mustn't", 'needn', 'and', 'further', 'the', 'how', 'no', 'theirs', 'by', 'were', 'under', 'after', "doesn't", 'me', 'if', 't', 'own', 'be', 'mightn', "couldn't", 'other', 'whom', 'this', 'until', 'most', "you'd", "you're", 'have', 'shouldn', 'which', 'having', 'ours', 'ain', 'out', 'into', 'it', 'is', 'mustn', 'of', 'will', "wasn't", 'she', 'are', 'because', 'doing', 'ma', 'on', "hadn't", "haven't", 'there', 'himself', 'above', 'hers', 'a', 'does', 'each', 'some', 'isn', 'am', "needn't", "it's", 'll', 'herself', 'when', 'y', 'only', 'then', 'about', "she's", 'same', 'him', 'wasn', 'through', 'nor', 'now', 'did', 'had', 'an', 'aren', 'ourselves', 'hadn', 'won', 'weren', 'that', "that'll", 'but', 'just', 'up', 'these'}
```

```
In [834]: #Extracció de les paraules "stopword" del text.
```

```
paraules_no_stopwords = []
for paraula in token:
    paraula_tokenized = word_tokenize(paraula)
    stopwords_supr = [mot for mot in paraula_tokenized if mot not in stop_words]
    paraules_no_stopwords.append(stopwords_supr)
```

```
In [827]: print('Paraules del text no stopwords:\n', paraules_no_stopwords)
```

Paraules del text no stopwords:

```
[['Marley'], [], ['dead'], [], ['begin'], [], ['There'], [], [], ['doubt'], ['whatever'], [], [], ['The'], ['register'], [], [], ['burial'], [], ['signed'], [], [], ['clergyman'], [], ['clerk'], [], ['undertaker'], [], [], ['chief'], ['mourner'], ['Scrooge'], ['signed'], [], ['And'], ['Scrooge'], ['name'], [], ['good'], ['upon'], [], ['anything'], [], ['chose'], [], ['put'], [], ['hand'], [], ['Old'], ['Marley'], [], [], ['dead'], [], [], ['Mind'], ['I'], [], ['mean'], [], ['say'], [], ['I'], ['know'], [], [], ['knowledge'], [], [], [], ['particularly'], ['dead'], [], [], ['I'], ['might'], [], [], ['inclined'], [], [], ['regard'], [], [], [], ['deadest'], ['piece'], [], ['ironmongery'], [], [], ['trade'], ['But'], [], ['wisdom'], [], [], ['ancestors'], [], [], [], ['simile'], [], [], ['unhallowed'], ['hands'], ['shall'], [], ['disturb'], [], [], ['Country'], ['done'], [], ['You'], [], [], ['therefore'], ['permit'], [], [], ['repeat'], ['emphatically'], [], ['Marley'], [], [], ['dead'], [], [], ['Scrooge'], ['knew'], [], [], ['dead'], ['Of'], ['course'], [], [], ['How'], ['could'], [], [], ['otherwise'], ['Scrooge'], [], [], [], ['partners'], [], ['I'], [], ['know'], [], ['many'], ['years'], ['Scrooge'], [], [], ['sole'], ['executor'], [], [], ['sole'], ['12'], ['administrator'], [], ['sole'], ['assign'], [], ['sole'], ['residuary'], ['legatee'], [], ['sole'], ['friend'], [], ['sole'], ['mourner'], ['And'], ['even'], ['Scrooge'], [], [], [], ['dreadfully'], ['cut'], [], [], [], ['sad'], ['event'], [], [], [], ['excellent'], ['man'], [], ['business'], [], [], [], ['day'], [], [], ['funeral'], [], ['solemnised'], [], [], [], ['undoubted'], ['bargain'], ['The'], ['mention'], [], ['Marley'], ['funeral'], ['bring'], [], ['back'], [], [], ['point'], ['I'], ['started'], [], ['There'], [], [], ['doubt'], [], [], ['Marley'], [], ['dead'], ['This'], ['must'], [], ['distinctly'], ['understood'], [], ['nothing'], ['wonderful'], [], ['come'], [], [], ['story'], ['I'], [], ['going'], [], ['relate'], ['If'], [], [], [], ['perfectly'], ['convinced'], [], ['Hamlet'], ['Father'], ['died'], [], [], ['play'], ['began'], [], ['would'], [], ['nothing'], [], ['remarkable'], [], [], ['taking'], [], ['stroll'], [], ['night'], [], [], ['easterly'], ['wind'], ['upon'], [], [], ['ramparts'], [], [], ['would'], [], [], [], ['gentleman'], ['rashly'], ['turning'], [], [], ['dark'], [], [], ['breezy'], ['Paul'], [], [], ['astonish'], [], ['son'], ['weak'], ['mind'], ['Scrooge'], ['never'], ['painted'], [], ['Old'], ['Marley'], ['name'], ['There'], [], ['stood'], ['years'], ['afterwards'], [], [], ['warehouse'], ['door'], ['Scrooge'], [], ['Marley'], ['The'], ['firm'], [], ['known'], [], ['Scrooge'], [], ['Marley'], ['Sometimes'], ['people'], ['new'], [], [], ['business'], ['called'], ['Scrooge'], ['Scrooge'], [], ['sometimes'], ['Marley'], [], [], ['answered'], [], [], ['names'], ['It'], [], [], [], [], []]
```

```
In [821]: #Stemming al text de Charles Dickens.
```

```
ps = PorterStemmer()
paraules_stem = []
for word in token:
    rootWord = ps.stem(word)
    paraules_stem.append(rootWord)
```

```
In [822]: print('Paraules del text després del stemming:\n', paraules_stem)
```

Paraules del text després del stemming:

```
['marley', 'wa', 'dead', 'to', 'begin', 'with', 'there', 'is', 'no', 'doubt', 'whatev', 'about', 'that', 'the', 'regist', 'of', 'hi', 'burial', 'wa', 'sign', 'by', 'the', 'clergyman', 'the', 'clerk', 'the', 'undertak', 'and', 'the', 'chief', 'mourner', 'scroog', 'sign', 'it', 'and', 'scroog', 'name', 'wa', 'good', 'upon', 'for', 'anyth', 'he', 'choose', 'to', 'put', 'hi', 'hand', 'to', 'old', 'marley', 'wa', 'as', 'dead', 'as', 'a', 'mind', 'I', 'do', 'mean', 't o', 'say', 'that', 'I', 'know', 'of', 'my', 'own', 'knowledg', 'what', 'there', 'is', 'particularli', 'dead', 'abou t', 'a', 'I', 'might', 'have', 'been', 'inclin', 'myself', 'to', 'regard', 'a', 'as', 'the', 'deadest', 'piec', 'of', 'ironmongeri', 'in', 'the', 'trade', 'but', 'the', 'wisdom', 'of', 'our', 'ancestor', 'is', 'in', 'the', 'simil', 'an d', 'my', 'unhallow', 'hand', 'shall', 'not', 'disturb', 'it', 'or', 'the', 'countri', 'done', 'for', 'you', 'will', 'therrefor', 'permit', 'me', 'to', 'repeat', 'emphat', 'that', 'marley', 'wa', 'as', 'dead', 'as', 'a', 'scroog', 'knew', 'he', 'wa', 'dead', 'Of', 'cours', 'he', 'did', 'how', 'could', 'it', 'be', 'otherwis', 'scroog', 'and', 'he', 'w ere', 'partner', 'for', 'I', 'do', 'know', 'how', 'mani', 'year', 'scroog', 'wa', 'hi', 'sole', 'executor', 'hi', 'sole', '12', 'administr', 'hi', 'sole', 'assign', 'hi', 'sole', 'residuari', 'legate', 'hi', 'sole', 'friend', 'and', 'sole', 'mourner', 'and', 'even', 'scroog', 'wa', 'not', 'so', 'dread', 'cut', 'up', 'by', 'the', 'sad', 'event', 'bu t', 'that', 'he', 'wa', 'an', 'excel', 'man', 'of', 'busi', 'on', 'the', 'veri', 'day', 'of', 'the', 'funer', 'and', 'solemnis', 'it', 'with', 'an', 'undoubt', 'bargain', 'the', 'mention', 'of', 'marley', 'funer', 'bring', 'me', 'bac k', 'to', 'the', 'point', 'I', 'start', 'from', 'there', 'is', 'no', 'doubt', 'that', 'marley', 'wa', 'dead', 'thi', 'must', 'be', 'distinctli', 'understood', 'or', 'noth', 'wonder', 'can', 'come', 'of', 'the', 'stori', 'I', 'am', 'go', 'to', 'relat', 'If', 'we', 'were', 'not', 'perfectli', 'convinc', 'that', 'hamlet', 'father', 'die', 'befor', 'the', 'play', 'began', 'there', 'would', 'be', 'noth', 'more', 'remark', 'in', 'hi', 'take', 'a', 'stroll', 'at', 'nigh t', 'in', 'an', 'easterli', 'wind', 'upon', 'hi', 'own', 'rampart', 'than', 'there', 'would', 'be', 'in', 'ani', 'oth er', 'gentleman', 'rashli', 'turn', 'out', 'after', 'dark', 'in', 'a', 'breezi', 'paul', 'for', 'to', 'astonish', 'hi', 'son', 'weak', 'mind', 'scroog', 'never', 'paint', 'out', 'old', 'marley', 'name', 'there', 'it', 'stood', 'yea r', 'afterward', 'abov', 'the', 'warehous', 'door', 'scroog', 'and', 'marley', 'the', 'firm', 'wa', 'known', 'as', 'scroog', 'and', 'marley', 'sometim', 'peopl', 'new', 'to', 'the', 'busi', 'call', 'scroog', 'scroog', 'and', 'someti m', 'marley', 'but', 'he', 'answer', 'to', 'both', 'name', 'It', 'wa', 'all', 'the', 'same', 'to', 'him']
```

```
In [823]: #Nivell 3
```

```
In [ ]: #Exercici 1  
#Realitza sentiment analysis al teu conjunt de dades.(trobar la polaritat i subjectivitat d'un missatge)
```

```
In [1110]: #Llibreries import  
import pandas as pd  
import nltk
```

```
In [1111]: #Libraries from
from nltk.tokenize import word_tokenize
from nltk import sent_tokenize
from tokenizers import Tokenizer
from sklearn.linear_model import Ridge
from textblob import TextBlob
```

```
In [1112]: text = "Marley was dead, to begin with. There is no doubt whatever about that. The register of his burial was signed by the clergyman, the clerk, the undertaker, and the chief mourner. Scrooge signed it. And Scrooge's name was good upon 'Change for anything he chose to put his hand to. Old Marley was as dead as a door-nail. Mind! I don't mean to say that I know, of my own knowledge, what there is particularly dead about a door-nail. I might have been inclined, myself, to regard a coffin-nail as the deadest piece of ironmongery in the trade. But the wisdom of our ancestors is in the smile; and my unhallowed hands shall not disturb it, or the Country's done for. You will, therefore, permit me to repeat, emphatically, that Marley was as dead as a door-nail. Scrooge knew he was dead? Of course he did. How could it be otherwise? Scrooge and he were partners for I don't know how many years. Scrooge was his sole executor, his sole[12] administrator, his sole assign, his sole residuary legatee, his sole friend, and sole mourner. And even Scrooge was not so dreadfully cut up by the sad event, but that he was an excellent man of business on the very day of the funeral, and solemnised it with an undoubted bargain. The mention of Marley's funeral brings me back to the point I started from. There is no doubt that Marley was dead. This must be distinctly understood, or nothing wonderful can come of the story I am going to relate. If we were not perfectly convinced that Hamlet's Father died before the play began, there would be nothing more remarkable in his taking a stroll at night, in an easterly wind, upon his own ramparts, than there would be in any other middle-aged gentleman rashly turning out after dark in a breezy spot—say St. Paul's Church-yard, for instance—literally to astonish his son's weak mind. Scrooge never painted out Old Marley's name. There it stood, years afterwards, above the warehouse door: Scrooge and Marley. The firm was known as Scrooge and Marley. Sometimes people new to the business called Scrooge Scrooge, and sometimes Marley, but he answered to both names. It was all the same to him."
```

```
In [1113]: tokenized_text=sent_tokenize(text)
```

```
In [1114]: print(tokenized_text)
```

['Marley was dead, to begin with.', 'There is no doubt whatever about that.', 'The register of his burial was signed by the clergyman, the clerk, the undertaker, and the chief mourner.', 'Scrooge signed it.', "And Scrooge's name was good upon 'Change for anything he chose to put his hand to.", 'Old Marley was as dead as a door-nail.', 'Mind!', "I don't mean to say that I know, of my own knowledge, what there is particularly dead about a door-nail.", 'I might have been inclined, myself, to regard a coffin-nail as the deadeast piece of ironmongery in the trade.', "But the wisdom of our ancestors is in the simile; and my unhallowed hands shall not disturb it, or the Country's done for.", 'You will, therefore, permit me to repeat, emphatically, that Marley was as dead as a door-nail.', 'Scrooge knew he was dead?', 'Of course he did.', 'How could it be otherwise?', "Scrooge and he were partners for I don't know how many years.", 'Scrooge was his sole executor, his sole[12] administrator, his sole assign, his sole residuary legatee, his sole friend, and sole mourner.', 'And even Scrooge was not so dreadfully cut up by the sad event, but that he was an excellent man of business on the very day of the funeral, and solemnised it with an undoubted bargain.', "The mention of Marley's funeral brings me back to the point I started from.", 'There is no doubt that Marley was dead.', 'This must be distinctly understood, or nothing wonderful can come of the story I am going to relate.', "If we were not perfectly convinced that Hamlet's Father died before the play began, there would be nothing more remarkable in his taking a stroll at night, in an easterly wind, upon his own ramparts, than there would be in any other middle-aged gentleman rashly turning out after dark in a breezy spot—say St. Paul's Church-yard, for instance—literally to astonish his son's weak mind.", "Scrooge never painted out Old Marley's name.", 'There it stood, years afterwards, above the warehouse door: Scrooge and Marley.', 'The firm was known as Scrooge and Marley.', 'Sometimes people new to the business called Scrooge Scrooge, and sometimes Marley, but he answered to both names.', 'It was all the same to him.]

```
In [1115]: token = nltk.word_tokenize(text)
```

```
In [1116]: print(token)
```

```
['Marley', 'was', 'dead', ',', 'to', 'begin', 'with', '.', 'There', 'is', 'no', 'doubt', 'whatever', 'about', 'that', '.', 'The', 'register', 'of', 'his', 'burial', 'was', 'signed', 'by', 'the', 'clergyman', ',', 'the', 'clerk', ',', 'the', 'undertaker', ',', 'and', 'the', 'chief', 'mourner', '.', 'Scrooge', 'signed', 'it', '.', 'And', 'Scrooge', "'s", 'name', 'was', 'good', 'upon', "'Change", 'for', 'anything', 'he', 'chose', 'to', 'put', 'his', 'hand', 'to', '.', 'Old', 'Marley', 'was', 'as', 'dead', 'as', 'a', 'door-nail', '.', 'Mind', '!!', 'I', 'do', "n't", 'mean', 'to', 'say', 'that', 'I', 'know', ',', 'of', 'my', 'own', 'knowledge', ',', 'what', 'there', 'is', 'particularly', 'dead', 'about', 'a', 'door-nail', '.', 'I', 'might', 'have', 'been', 'inclined', ',', 'myself', ',', 'to', 'regard', 'a', 'c offin-nail', 'as', 'the', 'deadest', 'piece', 'of', 'ironmongery', 'in', 'the', 'trade', '.', 'But', 'the', 'wisdom', 'of', 'our', 'ancestors', 'is', 'in', 'the', 'simile', ';', 'and', 'my', 'unhallowed', 'hands', 'shall', 'not', 'dist urb', 'it', ',', 'or', 'the', 'Country', "'s", 'done', 'for', '.', 'You', 'will', ',', 'therefore', ',', 'permit', 'm e', 'to', 'repeat', ',', 'emphatically', ',', 'that', 'Marley', 'was', 'as', 'dead', 'as', 'a', 'door-nail', '.', 'Sc rooge', 'knew', 'he', 'was', 'dead', '?', 'Of', 'course', 'he', 'did', '.', 'How', 'could', 'it', 'be', 'otherwise', '?', 'Scrooge', 'and', 'he', 'were', 'partners', 'for', 'I', 'do', "n't", 'know', 'how', 'many', 'years', '.', 'Scrooge', 'was', 'his', 'sole', 'executor', ',', 'his', 'sole', '[', '12', ']', 'administrator', ',', 'his', 'sole', 'assi gn', ',', 'his', 'sole', 'residuary', 'legatee', ',', 'his', 'sole', 'friend', ',', 'and', 'sole', 'mourner', '.', 'A nd', 'even', 'Scrooge', 'was', 'not', 'so', 'dreadfully', 'cut', 'up', 'by', 'the', 'sad', 'event', ',', 'but', 'tha t', 'he', 'was', 'an', 'excellent', 'man', 'of', 'business', 'on', 'the', 'very', 'day', 'of', 'the', 'funeral', ',', 'and', 'solemnised', 'it', 'with', 'an', 'undoubted', 'bargain', '.', 'The', 'mention', 'of', 'Marley', "'s", 'funera l', 'brings', 'me', 'back', 'to', 'the', 'point', 'I', 'started', 'from', '.', 'There', 'is', 'no', 'doubt', 'that', 'Marley', 'was', 'dead', '.', 'This', 'must', 'be', 'distinctly', 'understood', ',', 'or', 'nothing', 'wonderful', 'c an', 'come', 'of', 'the', 'story', 'I', 'am', 'going', 'to', 'relate', '.', 'If', 'we', 'were', 'not', 'perfectly', 'convinced', 'that', 'Hamlet', "'s", 'Father', 'died', 'before', 'the', 'play', 'began', ',', 'there', 'would', 'be', 'nothing', 'more', 'remarkable', 'in', 'his', 'taking', 'a', 'stroll', 'at', 'night', ',', 'in', 'an', 'easterly', 'w ind', ',', 'upon', 'his', 'own', 'ramparts', ',', 'than', 'there', 'would', 'be', 'in', 'any', 'other', 'middle-age d', 'gentleman', 'rashly', 'turning', 'out', 'after', 'dark', 'in', 'a', 'breezy', 'spot-say', 'St.', 'Paul', "'s", 'Church-yard', ',', 'for', 'instance-literally', 'to', 'astonish', 'his', 'son', "'s", 'weak', 'mind', '.', 'Scroog e', 'never', 'painted', 'out', 'Old', 'Marley', "'s", 'name', '.', 'There', 'it', 'stood', ',', 'years', 'afterward s', ',', 'above', 'the', 'warehouse', 'door', ':', 'Scrooge', 'and', 'Marley', '.', 'The', 'firm', 'was', 'known', 'a s', 'Scrooge', 'and', 'Marley', '.', 'Sometimes', 'people', 'new', 'to', 'the', 'business', 'called', 'Scrooge', 'Sc rooge', ',', 'and', 'sometimes', 'Marley', ',', 'but', 'he', 'answered', 'to', 'both', 'names', '.', 'It', 'was', 'al l', 'the', 'same', 'to', 'him', '.']
```

```
In [1120]: textblob=TextBlob(text)
```

```
In [1127]: print("Polaritat: ", textblob.sentiment.polarity)#Polaritat del text.  
# -1 indica sentiment negatiu i +1 indica sentiment positiu.
```

Polaritat: 0.05925324675324676

```
In [1128]: print("Subjectivitat: ", textblob.sentiment.subjectivity)#Subjectivitat del text - valor [0,1].
```

Subjectivitat: 0.490487012987013

```
In [ ]:
```

```
In [ ]:
```