Project design

The goal of this project is to analyze the body of a group of **e-mails** from three different newsgroups to **automatically extract the topics** and to provide a **categorization** of them.
The final results are the automatic extraction of topics from the e-mail set and the labeling of each e-mail inside one of those topics.
The analysis is the following:

1. **LSA** is performed on the **tfidf matrix** representing the e-mails.
2. The **singular values** resulting from the LSA are explored to guess the number of topics by looking at the **maximum variance** explained.
3. **K-means clustering** is performed on the LSA output, by selecting k as the number of singular values explaining the maximum variance.
4. **LDA** is executed by selecting a number of topics for the analysis sligthly greater than k of step 3.
5. Results between 3. and 4. are compared.

The following functions have been implemented:

- **ETL** ← a python class to extract transform and load the newsgroup e-mail, the models and the result of the analysis.
- **Clustering** ← a python class for the word to vector transformation of the documents, the execution of the LSA and K-means Clustering.
- **Plots** ← a Jupyter Notebook with the plots of the clustering results.
- **LDA** ← a Jupyter Notebook with the LDA and the plot of the results.

Tools

The project has been coded in Python 3.6, the data for the analysis, as well as the results have been stored in pandas DataFrame objects.
NLTK and Gensim have been used for text processing.
The algorithm and metrics used for modeling and evaluation belong to the sklearn library.
The ouput has been loaded into csv files and files representing the serialization of models.
The visualizations have been realized using matplotlib, seaborn and pyLDAvis.

Data

The data are e-mails who belong to the 20Newsgroup dataset (http://qwone.com/~jason/20Newsgroups/20news-19997.tar.gz).
Three different Newsgroup have been chosen:

- comp.graphics
- rec.autos
- talk.misc.religion

Variable Name	**Type**	**Description**
Path	String	File System Path
Category	String	Newsgroup Category
Filename	String	Post number
Raw Text	String	Raw text of the message
Modeling Text	List of Strings	Uni-gram tokens of the e-mail

Algorithm(s)

- **Extract** : ← the data have been downloaded and loaded into a pandas Dataframe
- **Transform** : ← the data have been custom cleaned. Function for header, footer and quote cleanup have been implemented through the use of regular expressions. NLTK and Gensim have been used to eliminate english stopwords, plurals.
- **Model** :
  1. Modeling List of mail-unigram is transformed in a 1000- dimensional space through count matrix and Tf Idf transformation.
  2. LSA through Truncated SVD → reduction to a 50 dimensional space.
  3. K-means: 3 – means clustering on the reduced space.
  4. LDA: Looking for 4 – topic clustering of space.
- **Evaluation:**
  - The number of feature and the number of components have been chosen by maximizing the variance explained first, the V-measure second.
- **Load** : ← The pipeline produce the following output
  - ‘text.csv’: dataframe with the information described in the data section.

- ‘clustering.csv’ : dataframe with three principal component output of the LSA step, clustering labels output of the k-means algorithm, original newsgroup category of the e-mails.
- sv.pkl : ← singular values output of the LSA step.

What you'd do differently next time

I am pretty satisfied with the results of this project, both the analysis and the artifact produced.
The main improvement of this project have been the adoption of an iterative process.
More precisely, I ran small iterations of the whole pipeline since the start of the project, and I think this has been the main motivation behind the good quality of results, and the less time needed to have them.
For the next project I plan to emphasize this process even more by further minimizing the changes of each step and running multiple steps.