schulzdLab6

February 2, 2021

1 Lab 6: EDA with Clustering

David Schulz

1.1 Introduction

High dimensional data sets have too many variables to be able to analyze each variable individually. We need to turn to more sophisticated techniques such as dimensionality reduction and clustering. In this lab, we are going to analyze 63,542 emails. The raw text will be converted into a feature matrix using a "bag of words" model. Each column of the feature matrix corresponds to one word, each row corresponds to one email, and the entry stores the number of times that word was found in that email. We will perform dimensionality reduction using the Truncated SVD method, cluster the emails, and compare the "inherent" structure to the given class labels.

1.2 Part I: Load and Transform the Data

[1]:
```
import pandas as pd
from glob import glob
import json
from sklearn.feature_extraction.text import CountVectorizer

data = []

files = glob('../Lab 5/email_json/*.json', recursive=True)

for single_file in files:
    with open(single_file, 'r') as f:
        json_file = json.load(f)
        data.append({
            'category': json_file['category'],
            'to_address': json_file['to_address'],
            'from_address': json_file['from_address'],
            'subject': json_file['subject'],
            'body': json_file['body']
        })
```

```
data = pd.DataFrame.from_dict(data)
vect = CountVectorizer(binary=True, min_df=10)
X = vect.fit_transform(data['body'])
```

1.3 Part II: Cluster the Emails

[2]:
```
import numpy as np
from sklearn.cluster import DBSCAN
from sklearn.decomposition import TruncatedSVD

svd = TruncatedSVD(n_components=10)

svd_fit = svd.fit_transform(X)
c0 = svd_fit[:, 0]
c1 = svd_fit[:, 1]
x = np.hstack([c0.reshape((-1,1)), c1.reshape((-1,1))])

clustering = DBSCAN().fit(x)
```

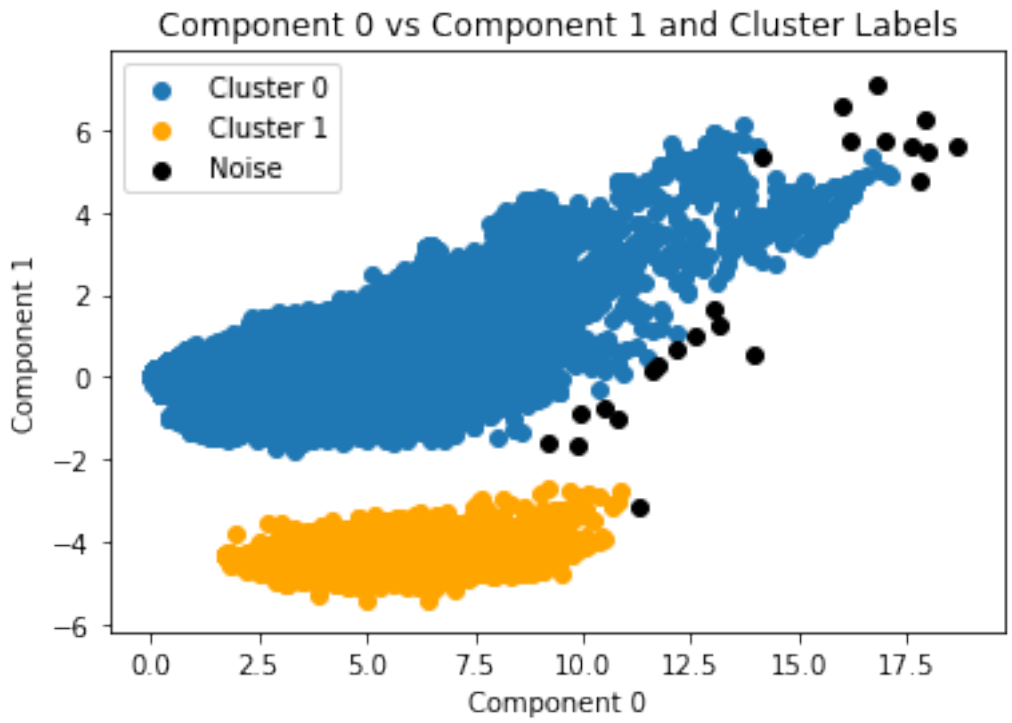
[3]:
```
import matplotlib.pyplot as plt

labels = clustering.labels_

x0 = c0[np.where(labels == 0)]
y0 = c1[np.where(labels == 0)]
x1 = c0[np.where(labels == 1)]
y1 = c1[np.where(labels == 1)]
xnoise = c0[np.where(labels == -1)]
ynoise = c1[np.where(labels == -1)]

plt.title('Component 0 vs Component 1 and Cluster Labels')
plt.xlabel('Component 0')
plt.ylabel('Component 1')
plt.scatter(x0, y0, label='Cluster 0')
plt.scatter(x1, y1, color='orange', label='Cluster 1')
plt.scatter(xnoise, ynoise, color='black', label='Noise')
plt.legend()
```

[3]:
```
<matplotlib.legend.Legend at 0x7f1eea77be90>
```



[4]:
```
ham_ind = data.index[data['category'] == 'ham'].tolist()
spam_ind = data.index[data['category'] == 'spam'].tolist()

c0_ind = np.where(labels == 0)[0]
c1_ind = np.where(labels == 1)[0]

h_c0 = np.intersect1d(ham_ind, c0_ind)
h_c1 = np.intersect1d(ham_ind, c1_ind)
s_c0 = np.intersect1d(spam_ind, c0_ind)
s_c1 = np.intersect1d(spam_ind, c1_ind)
print("Ham and Cluster 0:", h_c0.size)
print("Ham and Cluster 1:", h_c1.size)
print("Spam and Cluster 0:", s_c0.size)
print("Spam and Cluster 1:", s_c1.size)
```

```
Ham and Cluster 0: 17280
Ham and Cluster 1: 5780
Spam and Cluster 0: 40447
Spam and Cluster 1: 0
```

	Ham	Spam
Cluster 0	17280	40447
Cluster 1	5780	0

1.4 Part III: Calculating Document Frequencies of Words

[5]:
```
from scipy.sparse import csc_matrix

c0_data = X[c0_ind]
c1_data = X[c1_ind]

c0_data = csc_matrix(c0_data)
c1_data = csc_matrix(c1_data)

c0_freq = c0_data.sum(axis=0)
c1_freq = c1_data.sum(axis=0)

cols = vect.get_feature_names()
love = cols.index("love")
works = cols.index("works")
different = cols.index("different")

print("Cluster 0:")
print("Love -", c0_freq[0, love])
print("Works -", c0_freq[0, works])
print("Different -", c0_freq[0, different])
print("")
print("Cluster 1:")
print("Love -", c1_freq[0, love])
print("Works -", c1_freq[0, works])
print("Different -", c1_freq[0, different])
```

```
Cluster 0:
Love - 2013
Works - 2367
Different - 2086

Cluster 1:
Love - 23
Works - 629
Different - 779
```

1.5 Part IV: Find Enriched Words with Statistical Testing

[6]:
```
from scipy.stats import binom_test

cluster_1_expected_prob = c1_freq[0, works] / c1_data.shape[0]
pvalue = binom_test(c0_freq[0, works], c0_data.shape[0],␣
 ↪cluster_1_expected_prob, alternative="greater")
print("Works p-value:", pvalue)
```

```
cluster_1_expected_prob = c1_freq[0, love] / c1_data.shape[0]
pvalue = binom_test(c0_freq[0, love], c0_data.shape[0],␣
 ↪cluster_1_expected_prob, alternative="greater")
print("Love p-value:", pvalue)
```

```
Works p-value: 0.9999999999999999
Love p-value: 0.0
```

[10]:
```
enriched = []
for word in vect.vocabulary_:
    word_ind = cols.index(word)
    c0_word_freq = c0_freq[0, word_ind]
    cluster_1_expected_prob = c1_freq[0, word_ind] / c1_data.shape[0]
    pvalue = binom_test(c0_word_freq, c0_data.shape[0],␣
 ↪cluster_1_expected_prob, alternative="greater")
    if pvalue < 0.05:
        enriched.append((pvalue, word, c0_word_freq))

enriched = list(filter(lambda x: x[1].isalpha(), enriched))
enriched.sort(key = lambda x: x[0])

for i in range(200):
    print(enriched[i])
```

```
(0.0, 'love', 2013)
(0.0, 'enlightens', 25)
(0.0, 'blinds', 25)
(0.0, 'low', 4411)
(0.0, 'loss', 1789)
(0.0, 'democratic', 254)
(0.0, 'offer', 3397)
(0.0, 'tv', 1118)
(0.0, 'we', 17269)
(0.0, 'shipped', 159)
(0.0, 'cd', 1192)
(0.0, 'here', 13609)
(0.0, 'our', 13374)
(0.0, 'site', 4714)
(0.0, 'office', 2582)
(0.0, 'professional', 1948)
(0.0, 'adobe', 1119)
(0.0, 'acrobat', 1059)
(0.0, 'pro', 1275)
(0.0, 'microsoft', 1537)
(0.0, 'yourself', 2927)
(0.0, 'explode', 441)
```

```
(0.0, 'special', 4067)
(0.0, 'alert', 3476)
(0.0, 'tmxo', 364)
(0.0, 'trimax', 364)
(0.0, 'providers', 458)
(0.0, 'broadband', 534)
(0.0, 'over', 6921)
(0.0, 'bpl', 366)
(0.0, 'technologies', 842)
(0.0, 'otc', 513)
(0.0, 'deliver', 1303)
(0.0, 'encrypted', 433)
(0.0, 'high', 5270)
(0.0, 'video', 1922)
(0.0, 'herein', 525)
(0.0, 'prepared', 880)
(0.0, 'us', 9452)
(0.0, 'upon', 1638)
(0.0, 'guaranteed', 939)
(0.0, 'inclusive', 393)
(0.0, 'risks', 828)
(0.0, 'lose', 1515)
(0.0, 'your', 25832)
(0.0, 'money', 5384)
(0.0, 'licensed', 1285)
(0.0, 'broker', 1766)
(0.0, 'dealer', 497)
(0.0, 'market', 2008)
(0.0, 'investment', 1369)
(0.0, 'banker', 385)
(0.0, 'advisor', 531)
(0.0, 'underwriter', 369)
(0.0, 'purchasing', 665)
(0.0, 'selling', 820)
(0.0, 'negotiating', 498)
(0.0, 'cash', 1124)
(0.0, 'price', 7072)
(0.0, 'advertisement', 1484)
(0.0, 'near', 1291)
(0.0, 'future', 2537)
(0.0, 'parties', 673)
(0.0, 'officers', 542)
(0.0, 'directors', 472)
(0.0, 'employees', 942)
(0.0, 'buy', 5846)
(0.0, 'shares', 736)
(0.0, 'sell', 1798)
(0.0, 'profit', 1601)
```

```
(0.0, 'rise', 1274)
(0.0, 'bullish', 796)
(0.0, 'hottest', 1124)
(0.0, 'news', 5999)
(0.0, 'released', 1396)
(0.0, 'gnitpick', 308)
(0.0, 'additional', 3222)
(0.0, 'threatened', 111)
(0.0, 'every', 4406)
(0.0, 'walls', 114)
(0.0, 'pay', 2212)
(0.0, 'went', 1402)
(0.0, 'him', 3053)
(0.0, 'he', 6804)
(0.0, 'needle', 109)
(0.0, 'away', 2260)
(0.0, 'doctor', 1066)
(0.0, 'stabbed', 14)
(0.0, 'himself', 933)
(0.0, 'death', 881)
(0.0, 'full', 3557)
(0.0, 'his', 5552)
(0.0, 'hands', 1218)
(0.0, 'stolen', 147)
(0.0, 'bitter', 143)
(0.0, 'spring', 694)
(0.0, 'mist', 152)
(0.0, 'she', 3675)
(0.0, 'watch', 2580)
(0.0, 'her', 4009)
(0.0, 'eye', 587)
(0.0, 'toward', 586)
(0.0, 'studio', 1175)
(0.0, 'inc', 3469)
(0.0, 'learn', 3377)
(0.0, 'property', 859)
(0.0, 'pcap', 20)
(0.0, 'insertion', 20)
(0.0, 'registered', 729)
(0.0, 'attack', 540)
(0.0, 'mfc', 14)
(0.0, 'their', 7425)
(0.0, 'authenticity', 21)
(0.0, 'technological', 128)
(0.0, 'conversely', 22)
(0.0, 'occupations', 24)
(0.0, 'reminded', 52)
(0.0, 'thrice', 49)
```

```
(0.0, 'designer', 137)
(0.0, 'network', 2173)
(0.0, 'doctors', 448)
(0.0, 'increased', 924)
(0.0, 'artist', 214)
(0.0, 'digital', 600)
(0.0, 'creature', 114)
(0.0, 'up', 10852)
(0.0, 'growth', 1836)
(0.0, 'techniques', 1059)
(0.0, 'margaret', 132)
(0.0, 'thatcher', 33)
(0.0, 'mocking', 15)
(0.0, 'condescension', 10)
(0.0, 'defiance', 27)
(0.0, 'artwork', 136)
(0.0, 'signify', 21)
(0.0, 'internet', 2367)
(0.0, 'companies', 1380)
(0.0, 'speech', 1643)
(0.0, 'synthesizers', 21)
(0.0, 'learns', 33)
(0.0, 'city', 1382)
(0.0, 'hackers', 74)
(0.0, 'stakes', 46)
(0.0, 'dictators', 18)
(0.0, 'lives', 1397)
(0.0, 'who', 6860)
(0.0, 'sculpture', 47)
(0.0, 'life', 4836)
(0.0, 'intimidation', 15)
(0.0, 'eleanor', 23)
(0.0, 'relaying', 69)
(0.0, 'children', 1365)
(0.0, 'cialis', 3181)
(0.0, 'boost', 305)
(0.0, 'erection', 1506)
(0.0, 'benefits', 1605)
(0.0, 'hours', 2604)
(0.0, 'fast', 3307)
(0.0, 'ready', 3325)
(0.0, 'meals', 311)
(0.0, 'millions', 2040)
(0.0, 'men', 4129)
(0.0, 'online', 6787)
(0.0, 'revno', 672)
(0.0, 'revision', 3263)
(0.0, 'tridge', 803)
```

```
(0.0, 'samba', 3343)
(0.0, 'committer', 666)
(0.0, 'tridgell', 534)
(0.0, 'branch', 1026)
(0.0, 'nick', 832)
(0.0, 'timestamp', 801)
(0.0, 'ctdb', 547)
(0.0, 'modified', 2718)
(0.0, 'timed', 62)
(0.0, 'recruiting', 52)
(0.0, 'marketing', 687)
(0.0, 'consultants', 79)
(0.0, 'upwards', 60)
(0.0, 'bull', 379)
(0.0, 'gimmicks', 25)
(0.0, 'send', 2952)
(0.0, 'request', 2527)
(0.0, 'bremover', 25)
(0.0, 'llc', 1029)
(0.0, 'placeville', 57)
(0.0, 'ca', 5231)
(0.0, 'php', 1831)
(0.0, 'refinance', 623)
(0.0, 'loan', 1085)
(0.0, 'free', 5575)
(0.0, 'debt', 344)
(0.0, 'refinancing', 109)
(0.0, 'mortgage', 354)
(0.0, 'equity', 590)
(0.0, 'credit', 2057)
(0.0, 'purchase', 1636)
(0.0, 'visit', 5328)
(0.0, 'pgp', 827)
(0.0, 'signed', 1155)
```

[11]:
```
enriched = []
for word in vect.vocabulary_:
    word_ind = cols.index(word)
    c1_word_freq = c1_freq[0, word_ind]
    cluster_0_expected_prob = c0_freq[0, word_ind] / c0_data.shape[0]
    pvalue = binom_test(c1_word_freq, c1_data.shape[0],␣
 ↪cluster_0_expected_prob, alternative="greater")
    if pvalue < 0.05:
        enriched.append((pvalue, word, c1_word_freq))

enriched = list(filter(lambda x: x[1].isalpha(), enriched))
enriched.sort(key = lambda x: x[0])
```

```
for i in range(200):
    print(enriched[i])
```

```
(0.0, 'in', 4840)
(0.0, 'the', 5780)
(0.0, 'and', 5768)
(0.0, 'of', 4742)
(0.0, 'to', 5385)
(0.0, 'can', 3010)
(0.0, 'is', 4730)
(0.0, 'this', 4090)
(0.0, 'that', 3719)
(0.0, 'use', 2190)
(0.0, 'data', 2455)
(0.0, 'but', 3399)
(0.0, 'please', 5709)
(0.0, 'any', 2086)
(0.0, 'there', 2495)
(0.0, 'how', 2165)
(0.0, 'using', 2181)
(0.0, 'which', 2062)
(0.0, 'version', 2036)
(0.0, 'linear', 275)
(0.0, 'library', 843)
(0.0, 'provide', 5780)
(0.0, 'would', 2138)
(0.0, 'read', 5780)
(0.0, 'org', 5780)
(0.0, 'message', 1636)
(0.0, 'http', 5780)
(0.0, 'hi', 1903)
(0.0, 'my', 2179)
(0.0, 'code', 5780)
(0.0, 'row', 400)
(0.0, 'problem', 1512)
(0.0, 'do', 5780)
(0.0, 'wrote', 3005)
(0.0, 'mailing', 5780)
(0.0, 'list', 5723)
(0.0, 'posting', 5780)
(0.0, 'example', 1323)
(0.0, 'ecrc', 33)
(0.0, 'simpsonatnospamucl', 33)
(0.0, 'ac', 558)
(0.0, 'gower', 35)
(0.0, 'www', 5780)
```

```
(0.0, 'ucfagls', 33)
(0.0, 'freshwaters', 31)
(0.0, 'help', 5780)
(0.0, 'stat', 5780)
(0.0, 'math', 5780)
(0.0, 'ethz', 5780)
(0.0, 'ch', 5780)
(0.0, 'https', 5695)
(0.0, 'mailman', 5780)
(0.0, 'listinfo', 5780)
(0.0, 'guide', 5780)
(0.0, 'project', 5780)
(0.0, 'html', 5713)
(0.0, 'commented', 5780)
(0.0, 'minimal', 5780)
(0.0, 'self', 5780)
(0.0, 'contained', 5780)
(0.0, 'reproducible', 5780)
(0.0, 'error', 1286)
(0.0, 'want', 1702)
(0.0, 'thanks', 2786)
(0.0, 'following', 1277)
(0.0, 'nabble', 398)
(0.0, 'archive', 481)
(0.0, 'anyone', 755)
(0.0, 'am', 2212)
(0.0, 'model', 645)
(0.0, 'varadhan', 61)
(0.0, 'ph', 409)
(0.0, 'professor', 595)
(0.0, 'gerontology', 68)
(0.0, 'university', 1175)
(0.0, 'fax', 902)
(0.0, 'rvaradhan', 52)
(0.0, 'jhmi', 53)
(0.0, 'edu', 555)
(0.0, 'jhsph', 57)
(0.0, 'agingandhealth', 50)
(0.0, 'bounces', 534)
(0.0, 'mailto', 590)
(0.0, 'question', 799)
(0.0, 'values', 861)
(0.0, 'package', 1367)
(0.0, 'trying', 1118)
(0.0, 'stats', 527)
(0.0, 'alternative', 1502)
(0.0, 'deleted', 1426)
(0.0, 'regression', 289)
```

```
(0.0, 'longitudinal', 27)
(0.0, 'tried', 838)
(0.0, 'statistics', 679)
(0.0, 'true', 986)
(0.0, 'plot', 806)
(0.0, 'frame', 827)
(0.0, 'rserve', 15)
(0.0, 'matrix', 862)
(0.0, 'persp', 14)
(0.0, 'sas', 213)
(0.0, 'variance', 176)
(0.0, 'cbind', 222)
(0.0, 'lme', 95)
(0.0, 'col', 275)
(0.0, 'reml', 29)
(0.0, 'gdata', 18)
(0.0, 'variables', 499)
(0.0, 'function', 2058)
(0.0, 'variable', 540)
(0.0, 'brian', 438)
(0.0, 'rnews', 12)
(0.0, 'packages', 592)
(0.0, 'subset', 282)
(0.0, 'lapply', 165)
(0.0, 'dimitris', 53)
(0.0, 'rizopoulos', 47)
(0.0, 'biostatistical', 44)
(0.0, 'leuven', 44)
(0.0, 'kapucijnenvoer', 44)
(0.0, 'tel', 619)
(0.0, 'kuleuven', 50)
(0.0, 'biostat', 145)
(0.0, 'jiho', 24)
(0.0, 'dataframes', 22)
(0.0, 'column', 448)
(0.0, 'envir', 50)
(0.0, 'enclos', 36)
(0.0, 'advance', 618)
(0.0, 'irisson', 17)
(0.0, 'cwis', 50)
(0.0, 'vectors', 164)
(0.0, 'factor', 439)
(0.0, 'sep', 272)
(0.0, 'grdevices', 114)
(0.0, 'aov', 46)
(0.0, 'venables', 44)
(0.0, 'ripley', 426)
(0.0, 'coefficients', 143)
```

```
(0.0, 'iasonas', 17)
(0.0, 'lamprianou', 17)
(0.0, 'oxford', 351)
(0.0, 'cran', 295)
(0.0, 'vector', 567)
(0.0, 'rows', 369)
(0.0, 'columns', 426)
(0.0, 'randomforest', 24)
(0.0, 'liaw', 41)
(0.0, 'nans', 12)
(0.0, 'ggplot', 41)
(0.0, 'ox', 327)
(0.0, 'parks', 315)
(0.0, 'bioconductor', 64)
(0.0, 'efg', 21)
(0.0, 'glynn', 19)
(0.0, 'stowers', 17)
(0.0, 'deepankar', 34)
(0.0, 'tseries', 12)
(0.0, 'pchisq', 11)
(0.0, 'df', 274)
(0.0, 'sqrt', 93)
(0.0, 'pnorm', 32)
(0.0, 'statistical', 295)
(0.0, 'normality', 35)
(0.0, 'rsitesearch', 33)
(0.0, 'cberry', 41)
(0.0, 'tajo', 41)
(0.0, 'jolla', 41)
(0.0, 'logistic', 88)
(0.0, 'piecewise', 10)
(0.0, 'teachingdemos', 30)
(0.0, 'xaxs', 10)
(0.0, 'axis', 240)
(0.0, 'intermountainmail', 67)
(0.0, 'tktoplevel', 14)
(0.0, 'rcmdr', 26)
(0.0, 'socserv', 50)
(0.0, 'jfox', 51)
(0.0, 'tcltk', 55)
(0.0, 'jgr', 22)
(0.0, 'rjava', 35)
(0.0, 'dtaa', 14)
(0.0, 'dataframe', 149)
(0.0, 'nrow', 197)
(0.0, 'anova', 131)
(0.0, 'bolker', 26)
(0.0, 'covariance', 79)
```

```
(0.0, 'mahalanobis', 10)
(0.0, 'ncol', 174)
(0.0, 'byrow', 44)
(0.0, 'odfweave', 16)
(0.0, 'goslee', 26)
(0.0, 'messag', 19)
(0.0, 'matrices', 130)
(0.0, 'xyplots', 14)
(0.0, 'sweave', 61)
(0.0, 'tinn', 35)
(0.0, 'miktex', 23)
(0.0, 'leisch', 15)
(0.0, 'numeric', 343)
```

1.6 Reflection Questions

1. Make a guess as to why the emails might form two distinct clusters.
    - In lab 5, I guessed that the two clusters represented the spam emails and the normal emails, but that was shown to be incorrect. I then said that the bottom cluster might represent the emails where the model was much more certain they were ham, likely because of specific words that were almost never seen in spam emails, while the rest are much less certain and sometimes undeterministic.
2. Compare the ham/spam labels to the cluster labels using the confusion matrix you generated. Are spam messages in both clusters or a single cluster? Are all of the messages in the clusters with spam labeled as spam?
    - Spam messages are only in Cluster 0, but not all messages in Cluster 0 are spam.
3. Skim through the top 200 words for each cluster. Can you identify any patterns for either of the clusters?
    - The top 200 words in cluster 1 are typically more technical terms, such as regression, vectors, or covariance. A lot of them don't even seem to be words, but possibly other terms the R language works. I also noticed a lot of the words in cluster 0 are in the theme of economics, such as professional, profit, companies, or purchase.
4. a. Select the rows in the DataFrames for the emails in cluster 0. Print the top 25. Do the same for cluster 1.

[12]:
```
print(data.iloc[c0_ind[:25]])
```

```
                                                 body category  \
0   \n\n\n\n\n\n\n\n\n\n\n\n\nLove works a differe…     spam
1   This one will explode\nSpecial Situation Alert…     spam
2   \n\n\n\n\n\nSee " String Instructions" on page…     spam
3   \n\n\n\n\n\n\n\neasy to think of all the possi…     spam
4   \n\n\n\n\n\nCialis will boost up your erection…     spam
5   ----------------------------------------------…      ham
```

```
6   \nRecruiting new marketing consultants for a l…     spam
7   I gained 4 inches\n\nhttp://uwipntyi.tnstp.com…     spam
8   \n\n\n\n\n\n\nHello, Your refinance applicatio…     spam
9   -----BEGIN PGP SIGNED MESSAGE-----\nHash: SHA1…      ham
10  ==========================================…      ham
11  \n\n\n\n\n\n\nVIAGRA\nIf you have a problem ge…     spam
13  On Sunday 22 April 2007 17:38, Matt Diephouse …      ham
14  Jude DaShiell wrote:\n> Those aliases need to …      ham
15     \n    SEC Filing Alert Netflix, Inc. has fi…      ham
17  \n\n\n\n\n\n\nLotteryagent (TM) is the only on…     spam
18  \n\n\n\n\n\n\n\n\n \n"But, hospital on the oth…     spam
19  Hello,\n\nLife Should be Full of Luxuries, yet…     spam
20  \n\n\nYou have received this announcement beca…     spam
21  Reverted in r18519. This feature is still bein…      ham
22  * Steve Langasek:\n\n>>   All other non-permis…      ham
23  \nHi Robert,\n\nI use GD::Graph and have found…      ham
24  \nReuters Financial Information\n\n\n\n\n\n\nB…     spam
25  use our site to incurLargeSaving\nhttp://icdnl…     spam
27  \n\n\n\n\n\n\nVIAGRA\nIf you have a problem ge…     spam

                                        from_address  \
0          Pablo Timmons <SophiRosemary4926@yahoo.com>
1       "Inez Tanner" <ocuseethed@executiveemail.com>
2                  "Joar Moree" <Joar@arslanzade.com>
3                 bass Elkins <Gailk@waydelivery.com>
4                   Works Fast <kmamie@netcityhk.com>
5                                  tridge@samba.org
6   "North, Reggie" <Hurley2Blount@jennieagbayanih…
7   "Estrada, Stanley" <StanleywEstrada@common-wea…
8                 "Dewitt" <ikqdl@flax9.uwaterloo.ca>
9      "Stefan (metze) Metzmacher" <metze@samba.org>
10                              slashdot@slashdot.org
11  "Wankeeta Hogland" <yosemitelifeguards@hsj.com…
13                      chromatic <chromatic@wgz.org>
14                      Gaijin <gaijin@clearwire.net>
15                  <alert@broadcast.shareholder.com>
17  "Tyler Patterson" <eightstar.com@lottozubotto…
18                "Bobby" <apancieraausoc@inf.uth.gr>
19  "Kenton Gilliam" <Rupertaccomplicecelebrate@rr…
20  "Crohn's Disease Newsletter" <qj_ci9zmi@riskbe…
21  "Allison Randal via RT" <parrotbug-followup@pa…
22                  Florian Weimer <fw@deneb.enyo.de>
23                Nigel Peck <nigel@miswebdesign.com>
24               "Chester Elder" <agsfcb@reuters.com>
25                 "Nona Reeves" <ztguc@incamail.com>
27  "Willim GREENBERG" <dissidentromance@encodeinc…

                                              subject  \
```

0	Software Compatibility…ain't it great?
1	At which bookshelves
2	It's just like Raistlin described to me once.
3	Have in mixture
4	Benefits of Cialis
5	Rev 328: show ctdb control timeout in http://s…
6	Come Join Us
7	A Larger Male Organ
8	brandy than transliterate
9	Re: [Samba4][PATCH] Updating the winbind proto…
10	[Slashdot] Headlines for 2007-07-06
11	Re:
13	Re: [PATCH] Re-work Parrot_process_args
14	Re: slackware aliases anyone?
15	New SEC Document(s) for Netflix, Inc.
17	Official Lottery tickets from around the world
18	It's almost there
19	Re:
20	Crohn's & Me: A New Crohn's Resource
21	[perl #42898] [PATCH] src/library.c , honor PA…
22	Re: Final text of GPL v3
23	Re: Charting Module
24	=?KOI8-R?Q?Reuters Financial Information?=
25	YouGottaSeeThis
27	Re:

	to_address
0	smile@speedy.uwaterloo.ca
1	"gnitpick" <gnitpick@speedy.uwaterloo.ca>
2	catchall@flax9.uwaterloo.ca
3	the00@plg2.math.uwaterloo.ca
4	manager@speedy.uwaterloo.ca
5	samba-cvs@samba.org
6	gnitpick@flax9.uwaterloo.ca
7	smiles@flax9.uwaterloo.ca
8	"Berta" <gnitpick@flax9.uwaterloo.ca>
9	Kai Blin <kai@samba.org>
10	avcooper@flax9.uwaterloo.ca
11	<soundtrackdeficient@flax9.uwaterloo.ca>
13	matt@diephouse.com, parrot-porters@perl.org
14	"Speakup is a screen review system for Linux."…
15	"Andrew Coopers" <avcoopers@flax9.uwaterloo.ca>
17	<gnitpick@speedy.uwaterloo.ca>
18	"Ramona Rivera" <henna@canola1.uwaterloo.ca>
19	theorize@plg.uwaterloo.ca
20	"Subscriber" <cruiseca@flax9.uwaterloo.ca>
21	"OtherRecipients of perl Ticket #42898": ;
22	debian-legal@lists.debian.org

```
23        "Brown, Rodrick" <rodrick.brown@lehman.com>
24                          theorize@plg.uwaterloo.ca
25                        gnitpick@flax9.uwaterloo.ca
27                         <smile@flax9.uwaterloo.ca>
```

[13]:
```
print(data.iloc[c1_ind[:25]])
```

```
                                                  body category  \
12   On Mon, 2007-06-04 at 18:25 -0500, Robert Wilk…      ham
16   \nHello everybody, i wish to input data from t…      ham
26   In my previous email, I meant to say:\n\nP1 <-…      ham
30   Chandra,\n\nyou might want to have a look at p…      ham
32   Hello R-Users:\n \nI am want to use tobit regr…      ham
47   Hello: \n\nI would like to make h-scatter plot…      ham
48   On Fri, 2007-04-27 at 20:29 +0300, Ralf Finne …      ham
49   Hi,\n\nThe silverman's paper introduction offe…      ham
66   I would like to convert the following SAS code…      ham
99   \nHi,\n\n  library(car)\n  ?levene.test\n\n\n\…      ham
114  \nWe were promised this package last spring bu…      ham
119  subset() was not defined inside myfun(); try t…      ham
156  \n--- croero@hotmail.com wrote:\n\n> \n> Hello…      ham
170  You can now use contourLines in the grDevices …      ham
184  Hi\n\nr-help-bounces@stat.math.ethz.ch napsal …      ham
185  Hi Neil,\n\nngottlieb@marinercapital.com wrote…      ham
186  My guess is that you have Gnome >=2.0, while t…      ham
199  G'day all,\n\nOn Tue, 8 May 2007 12:10:25 +080…      ham
201  Harold,\n\nActually there is a maximum size, e…      ham
219  Hi Andy,\n\nIt worked for classification, but …      ham
222  On Tue, 3 Jul 2007, hadley wickham wrote:\n\n>…      ham
225  "Li, Hua "  writes:\n\n> Dear list members: On…      ham
261  Hi,\n\n   I haven't been able to figure out ho…      ham
264  I'm using the latest R on Windows XP:\n\n> R.v…      ham
301  \nDeepankar,\n\nOn 19 April 2007 at 21:32, DEE…      ham

                                          from_address  \
12             Gavin Simpson <gavin.simpson@ucl.ac.uk>
16                     Miguel Caro <mcaro72@gmail.com>
26                "Ravi Varadhan" <rvaradhan@jhmi.edu>
30               Bettina Gruen <gruen@ci.tuwien.ac.at>
32                   Abdus Sattar <upsattar@yahoo.com>
47                       "Hong Su An" <anhong@msu.edu>
48                   Rajarshi Guha <rguha@indiana.edu>
49                 "Patrick Wang" <pwang@berkeley.edu>
66               Lucia Costanzo <lcostanz@uoguelph.ca>
99             Tomas Goicoa <tomas.goicoa@unavarra.es>
114                  francogrex <francogrex@mail.com>
119  "Dimitris Rizopoulos" <dimitris.rizopoulos@med…
156                    John Kane <jrkrideau@yahoo.ca>
```

```
170             "hadley wickham" <h.wickham@gmail.com>
184                Petr PIKAL <petr.pikal@precheza.cz>
185              Roland Rau <roland.rproject@gmail.com>
186            "Michael Lawrence" <lawremi@iastate.edu>
199          Berwin A Turlach <berwin@maths.uwa.edu.au>
201           Marc Schwartz <marc_schwartz@comcast.net>
219                       clayton.springer@novartis.com
222           Prof Brian Ripley <ripley@stats.ox.ac.uk>
225                     Seth Falcon <sfalcon@fhcrc.org>
261                   Judith Flores <juryef@yahoo.com>
264         "Earl F. Glynn" <efg@stowers-institute.org>
301                  Dirk Eddelbuettel <edd@debian.org>

                                                 subject  \
12   Re: [R] Why is the R mailing list so hard to f…
16              [R] how to input data from the keyboard
26   Re: [R] Time series\optimization  question not…
30                    Re: [R] distance method in kmeans
32           [R] Library & Package for Tobit regression
47                  [R] H-scatter plot in geostatistics
48   Re: [R] Integrating R-programs into larger sys…
49      [R] How to get the number of modes using kde2d
66   [R] converting proc mixed to lme for a random …
99                           Re: [R] Levene Test with R
114                        [R] Where is package "Umacs"?
119  Re: [R] lapply not reading arguments from the …
156                             Re: [R] importing data
170                 Re: [R] [R-sig-Geo] Clines library
184                                     [R] Odp:  Anova
185                       Re: [R] R Book Advice Needed
186  Re: [R] Problem installing gnomeGUI in Ubuntu:…
199                    Re: [R] Bad optimization solution
201       Re: [R] What is the maximum size of a matrix?
219                     Re: [R] NA and NaN randomForest
222           Re: [R] possible bug in ggplot2 v0.5.2???
225                Re: [R] questions on package of KEGG
261         [R] Removing vertical line in Tinn R editor
264         [R] Need 64-bit integers on 32-bit platform
301                  Re: [R] Problem installing packages

                                              to_address
12               Robert Wilkins <irishhacker@gmail.com>
16                              r-help@stat.math.ethz.ch
26   "'Ravi Varadhan'" <rvaradhan@jhmi.edu>,\n   "'…
30      Ranga Chandra Gudivada <chandra_bio@yahoo.com>
32                              R-help@stat.math.ethz.ch
47                              r-help@stat.math.ethz.ch
48                     Ralf Finne <Ralf.Finne@syh.fi>
```

```
49                           r-help@stat.math.ethz.ch
66                           r-help@stat.math.ethz.ch
99   "along zeng" <xh.along@gmail.com>, r-help <r-h…
114                          r-help@stat.math.ethz.ch
119                       "jiho" <jo.irisson@gmail.com>
156       croero@hotmail.com, r-help@stat.math.ethz.ch
170       "Andrew Niccolai" <andrew.niccolai@yale.edu>
184          Iasonas Lamprianou <lamprianou@yahoo.com>
185                        ngottlieb@marinercapital.com
186                       fsando <fsando@fs-analyse.dk>
199                          r-help@stat.math.ethz.ch
201                   "Doran, Harold" <HDoran@air.org>
219                          r-help@stat.math.ethz.ch
222                hadley wickham <h.wickham@gmail.com>
225                     "Li, Hua " <Hua.Li@uth.tmc.edu>
261                    RHelp <r-help@stat.math.ethz.ch>
264                          r-help@stat.math.ethz.ch
301                    DEEPANKAR BASU <basu.15@osu.edu>
```

4. b. Do you think the to and from addresses and subject lines provide additional help in identifying patterns?

   - Yes. A lot of the subjects in cluster 1 seem to be replies, and all of them contain the string "[R]". Also, many of the to addresses in cluster 1 are r-help@stat.math.ethz.ch.

5. The clusters represent emails from two separate mailing lists. One mailing list is for the R programming language, while the other mailing list is for a university. Which mailing list contained all of the spam?

   - The university mailing list contained all of the spam.