Import required packages

24-nov

```
- add dbscan to basic clustering ...
```

TOC

In [34]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
import warnings
warnings.simplefilter('ignore',DeprecationWarning)
import seaborn as sns
import time
import copy

from pylab import rcParams
#import hdbscan

from sklearn.model_selection import ShuffleSplit
from sklearn.preprocessing import StandardScaler

#from sklearn.datasets import make_blobs

from sklearn.ensemble import RandomForestClassifier
from sklearn.calibration import CalibratedClassifierCV
from sklearn.svm import SVC
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import StratifiedKFold, cross_val_score

from sklearn import metrics
from sklearn import metrics as mt
from sklearn.metrics import log_loss
from sklearn.metrics import accuracy_score as acc
from sklearn.metrics import confusion_matrix as conf
from sklearn.metrics import f1_score, precision_score, recall_score, classi
fication_report
from sklearn.metrics import accuracy_score
from sklearn.metrics import precision_recall_fscore_support as score

from sklearn.cluster import KMeans

from tabulate import tabulate

from IPython.core.interactiveshell import InteractiveShell
InteractiveShell.ast_node_interactivity = "all"

from __future__ import print_function
```

Read in cleaned dataset from .csv file

In [35]:
```
data_dir = '../data/'
data_file = 'mashable_clean_dataset_for_lab_02_task_02.csv'

file_2_read = data_dir + data_file
df = pd.read_csv(file_2_read)

df_cluster = copy.deepcopy(df)

del df_cluster['data_channel']
```

In [36]:
```
for column in ['LDA_00', 'LDA_01', 'LDA_02', 'LDA_03', 'LDA_04']:
    new_col_name = 'ln_' + column
    print (new_col_name)
    df_cluster[new_col_name] = np.log(df_cluster[column]+1)
```

```
ln_LDA_00
ln_LDA_01
ln_LDA_02
ln_LDA_03
ln_LDA_04
```

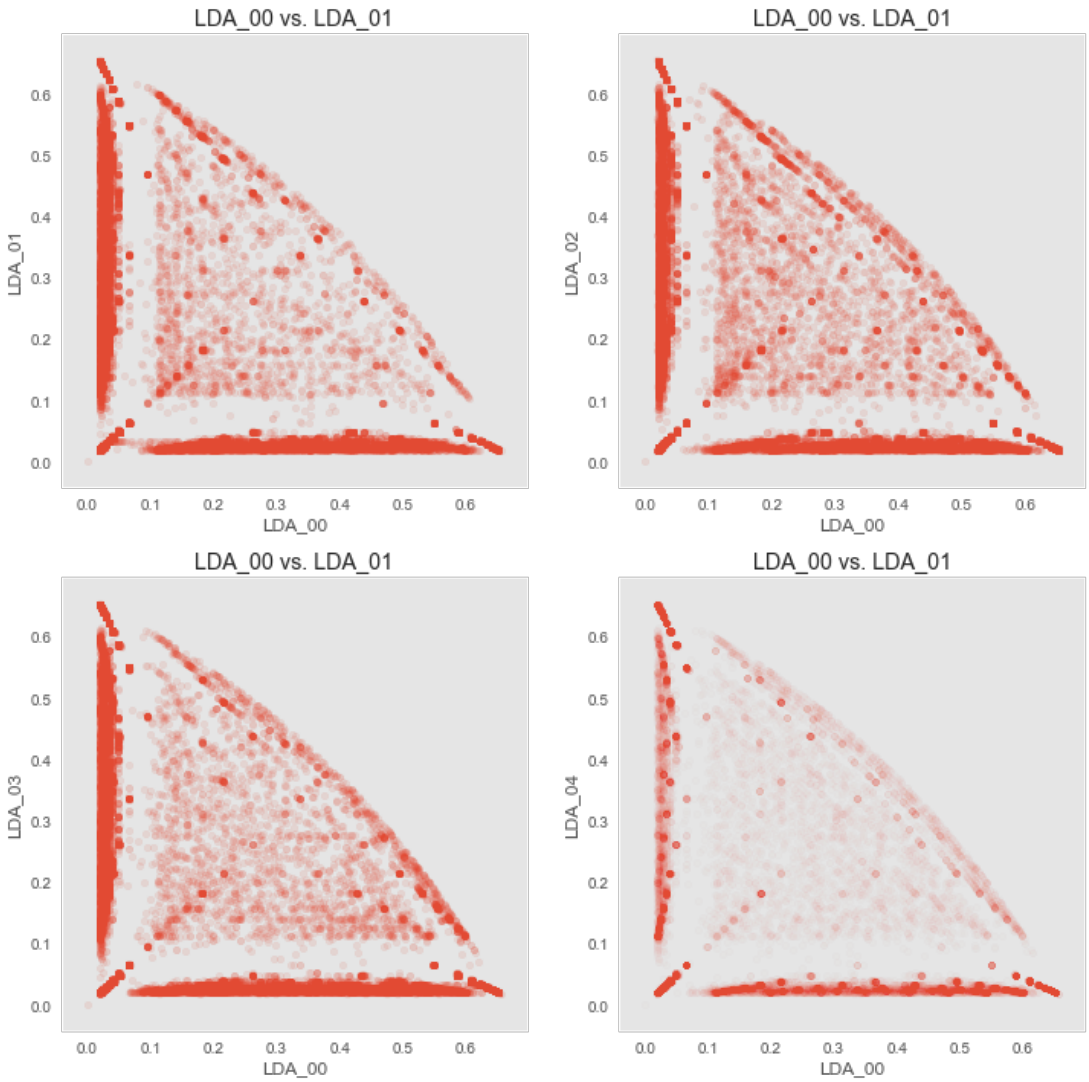
In [37]:
```
col_names = df_cluster.columns.values.tolist()
col_names
df_cluster.describe().T
```

```
Out[37]: ['n_tokens_title',
          'num_keywords',
          'kw_avg_max',
          'weekday_is_monday',
          'weekday_is_tuesday',
          'weekday_is_wednesday',
          'weekday_is_thursday',
          'weekday_is_friday',
          'is_weekend',
          'LDA_00',
          'LDA_01',
          'LDA_02',
          'LDA_03',
          'LDA_04',
          'global_subjectivity',
          'global_rate_positive_words',
          'rate_positive_words',
          'max_positive_polarity',
          'min_negative_polarity',
          'max_negative_polarity',
          'title_sentiment_polarity',
          'abs_title_subjectivity',
          'data_channel_n',
          'ln_n_tokens_content',
          'ln_num_hrefs',
          'ln_num_imgs',
          'ln_num_videos',
          'ln_kw_min_min',
          'ln_kw_avg_min',
          'ln_kw_min_max',
          'ln_kw_avg_avg',
          'ln_self_reference_avg_sharess',
          'ln_global_rate_negative_words',
          'ln_min_positive_polarity',
          'ln_abs_title_sentiment_polarity',
          'ln_LDA_00',
          'ln_LDA_01',
          'ln_LDA_02',
          'ln_LDA_03',
          'ln_LDA_04']
```

Out[37]:

	count	mean	std	min	25%	50%	75%
n_tokens_title	39644.0	10.398749	2.114037	2.0	9.000000	10.000000	12.000000
num_keywords	39644.0	7.223767	1.909130	1.0	6.000000	7.000000	9.000000
kw_avg_max	39644.0	1.913205	1.000000	0.0	1.271003	1.800325	2.442234
weekday_is_monday	39644.0	0.168020	0.373889	0.0	0.000000	0.000000	0.000000
weekday_is_tuesday	39644.0	0.186409	0.389441	0.0	0.000000	0.000000	0.000000
weekday_is_wednesday	39644.0	0.187544	0.390353	0.0	0.000000	0.000000	0.000000
weekday_is_thursday	39644.0	0.183306	0.386922	0.0	0.000000	0.000000	0.000000
weekday_is_friday	39644.0	0.143805	0.350896	0.0	0.000000	0.000000	0.000000
is_weekend	39644.0	0.130915	0.337312	0.0	0.000000	0.000000	0.000000
LDA_00	39644.0	0.184599	0.262975	0.0	0.025051	0.033387	0.240958
LDA_01	39644.0	0.141256	0.219707	0.0	0.025012	0.033345	0.150831
LDA_02	39644.0	0.216321	0.282145	0.0	0.028571	0.040004	0.334218
LDA_03	39644.0	0.223770	0.295191	0.0	0.028571	0.040001	0.375763
LDA_04	39644.0	0.234029	0.289183	0.0	0.028574	0.040727	0.399986
global_subjectivity	39644.0	0.443370	0.116685	0.0	0.396167	0.453457	0.508333
global_rate_positive_words	39644.0	0.039625	0.017429	0.0	0.028384	0.039023	0.050279
rate_positive_words	39644.0	0.682150	0.190206	0.0	0.600000	0.710526	0.800000
max_positive_polarity	39644.0	0.756728	0.247786	0.0	0.600000	0.800000	1.000000
min_negative_polarity	39644.0	0.478056	0.290290	0.0	0.300000	0.500000	0.700000
max_negative_polarity	39644.0	0.892500	0.095373	0.0	0.875000	0.900000	0.950000
title_sentiment_polarity	39644.0	1.071425	0.265450	0.0	1.000000	1.000000	1.150000
abs_title_subjectivity	39644.0	0.341843	0.188791	0.0	0.166667	0.500000	0.500000
data_channel_n	39644.0	4.184366	2.205607	1.0	2.000000	4.000000	6.000000
ln_n_tokens_content	39644.0	5.889971	1.255442	0.0	5.509388	6.016157	6.575076
ln_num_hrefs	39644.0	2.156564	0.809445	0.0	1.609438	2.197225	2.708050
ln_num_imgs	39644.0	1.116427	0.973755	0.0	0.693147	0.693147	1.609438
ln_num_videos	39644.0	0.400420	0.680486	0.0	0.000000	0.000000	0.693147
ln_kw_min_min	39644.0	1.174410	1.733030	0.0	0.000000	0.000000	1.791759
ln_kw_avg_min	39644.0	5.302209	1.132463	0.0	4.968076	5.470168	5.883322
ln_kw_min_max	39644.0	5.045209	4.521016	0.0	0.000000	7.244942	8.974745
ln_kw_avg_avg	39644.0	7.976327	0.489467	0.0	7.776304	7.962442	8.189031
ln_self_reference_avg_sharess	39644.0	6.667697	3.280186	0.0	6.889782	7.696667	8.556606
ln_global_rate_negative_words	39644.0	0.016419	0.010571	0.0	0.009569	0.015221	0.021506
ln_min_positive_polarity	39644.0	0.089255	0.060260	0.0	0.048790	0.095310	0.095310
ln_abs_title_sentiment_polarity	39644.0	0.128709	0.173844	0.0	0.000000	0.000000	0.223144

In [38]:
```
from matplotlib import pyplot as plt
plt.style.use("ggplot")

%matplotlib inline


X1 = df_cluster[['ln_LDA_00','ln_LDA_01', 'ln_LDA_02', 'ln_LDA_03', 'ln_LDA
_04']].values


plt.figure(figsize = (12,12))
plt.subplot(221)

plt.scatter(X1[:, 1], X1[:, 0],
            s = 20,
           alpha = 0.10)
plt.xlabel('LDA_00'), plt.ylabel('LDA_01')
plt.grid()
plt.title('LDA_00 vs. LDA_01')


plt.subplot(222)
plt.scatter(X1[:, 2], X1[:, 0],
            s = 20,
           alpha = 0.10)
plt.xlabel('LDA_00'), plt.ylabel('LDA_02')
plt.grid()
plt.title('LDA_00 vs. LDA_01')


plt.subplot(223)
plt.scatter(X1[:, 3], X1[:, 0],
            s = 20,
            alpha = 0.10)
plt.xlabel('LDA_00'), plt.ylabel('LDA_03')
plt.grid()
plt.title('LDA_00 vs. LDA_01')


plt.subplot(224)
plt.scatter(X1[:, 4], X1[:, 0],
            s = 20,
           alpha = 0.01)
plt.xlabel('LDA_00'), plt.ylabel('LDA_04')
plt.grid()
plt.title('LDA_00 vs. LDA_01')

plt.show();
```



In [39]:
```
# set required variables for model comparison

comparison_tbl = pd.DataFrame(columns = [
    'model_name',
    'n_clusters',
    'inertia',
    'silhouette',
    'process_time'])

i_index = []
i_index = 0

# preparation for cross validation and model comparison, each classifier is
appended once model is fit

models = []
```

Table of Contents

KMeans - LDA

K-Means - LDA scores

In [40]:
```
for n_lda in range(2, 12):

    tic = time.clock()

    print ("n_lda = ", n_lda)

    X1 = df_cluster[['ln_LDA_00','ln_LDA_01', 'ln_LDA_02', 'ln_LDA_03', 'ln_LDA_04']]

    cls_lda = KMeans(n_clusters = n_lda,
                     init = 'k-means++',
                     random_state = 1);

    cls_lda.fit(X1)

    kmeans_labels = cls_lda.labels_ # the labels from kmeans clustering
    kmeans_centers = cls_lda.cluster_centers_

    kmeans_inertia = cls_lda.inertia_
    print ("inertia = ", kmeans_inertia)

    kmeans_silhouette = metrics.silhouette_score(X1,
                                                 kmeans_labels,
                                                 metric = 'euclidean',
                                                 sample_size = 10000)
    print ("silhouette = ", kmeans_silhouette)

    toc =  time.clock()
# ... -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
# ... - save statistics for model comparison
# ... -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-

    exe_time = '{0:.4f}'.format(toc-tic)

    raw_data = {
    'model_name' : 'KMeans - LDA features',
    'n_clusters' : n_lda,
    'inertia': kmeans_inertia,
    'silhouette': kmeans_silhouette,
    'process_time' : exe_time
    }

    df_tbl = pd.DataFrame(raw_data,
    columns = ['model_name', 'n_clusters', 'inertia', 'silhouette', 'process_time'],
    index = [i_index + 1])

    comparison_tbl = comparison_tbl.append(df_tbl)

# ... -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
# ... - make some plots of clusters
# ... -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-

    plt.figure(figsize=(12, 12));
    plt.subplot(221);
    X1 = X1.values;
    plt.scatter(X1[:, 0], X1[:, 1],
```

```
n_lda =  2
```

Out[40]:
```
KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
    n_clusters=2, n_init=10, n_jobs=1, precompute_distances='auto',
    random_state=1, tol=0.0001, verbose=0)
```

```
inertia =  6016.62121709
silhouette =  0.2666600254
```

Out[40]:
```
<matplotlib.figure.Figure at 0x7f4b777868d0>
```

Out[40]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4b80027f28>
```

Out[40]:
```
<matplotlib.collections.PathCollection at 0x7f4b77121a90>
```

Out[40]:
```
<matplotlib.collections.PathCollection at 0x7f4b775cba90>
```

Out[40]:
```
<matplotlib.text.Text at 0x7f4b90101080>
```

Out[40]:
```
(<matplotlib.text.Text at 0x7f4b9054b7b8>,
 <matplotlib.text.Text at 0x7f4b774e6278>)
```

Out[40]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4b776d3e10>
```

Out[40]:
```
<matplotlib.collections.PathCollection at 0x7f4b7776c3c8>
```

Out[40]:
```
<matplotlib.collections.PathCollection at 0x7f4b7777d438>
```

Out[40]:
```
(<matplotlib.text.Text at 0x7f4b775cb9e8>,
 <matplotlib.text.Text at 0x7f4b775e5828>)
```

Out[40]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4b775cbd30>
```

Out[40]:
```
<matplotlib.collections.PathCollection at 0x7f4b7773b940>
```

Out[40]:
```
<matplotlib.collections.PathCollection at 0x7f4b77654240>
```

Out[40]:
```
(<matplotlib.text.Text at 0x7f4b7708ec50>,
 <matplotlib.text.Text at 0x7f4b802ec5c0>)
```

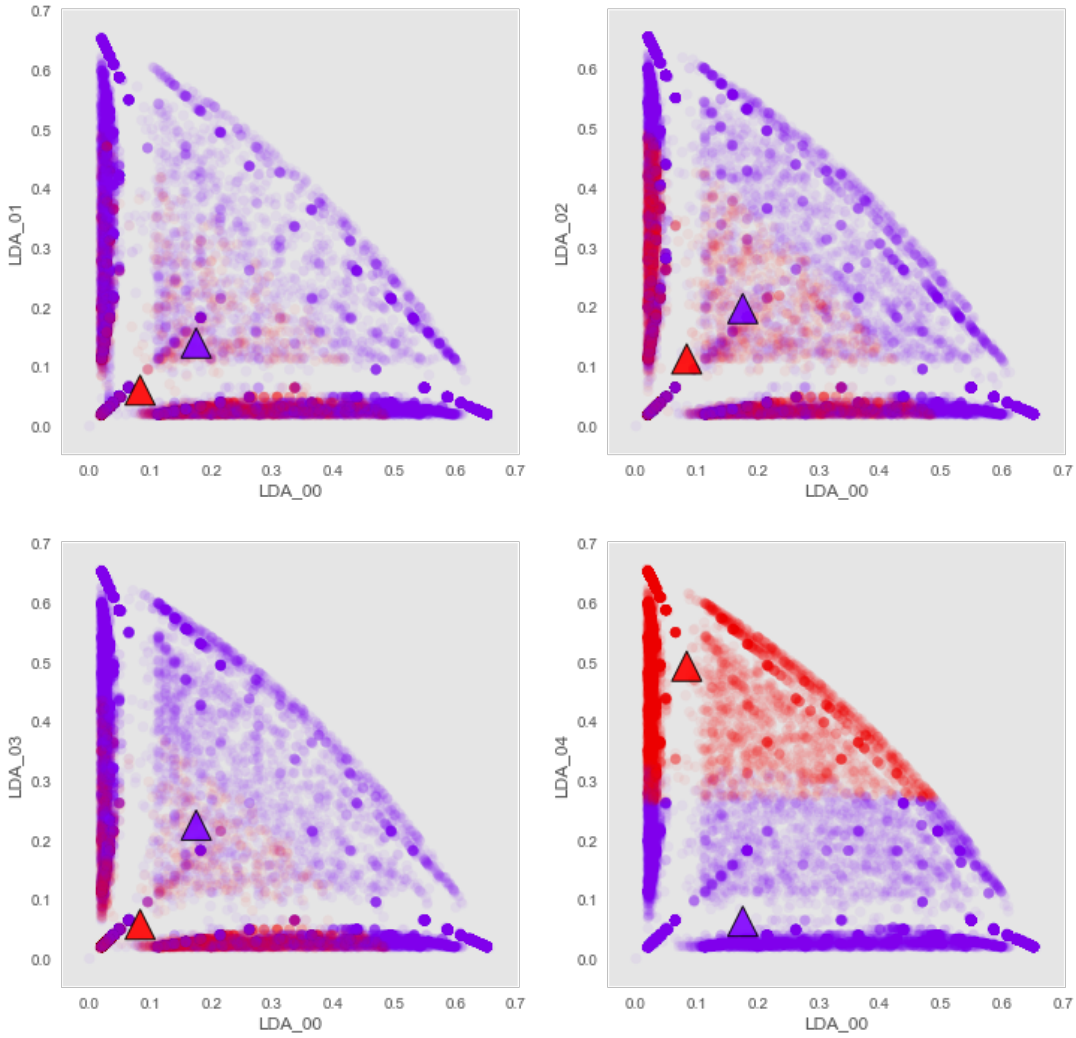
Out[40]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4b7773b908>
```

Out[40]:
```
<matplotlib.collections.PathCollection at 0x7f4b77691ef0>
```

Out[40]:
```
<matplotlib.collections.PathCollection at 0x7f4b77597780>
```

Out[40]:
```
(<matplotlib.text.Text at 0x7f4b80037710>,
 <matplotlib.text.Text at 0x7f4b7765a908>)
```

6016.62121709




```
n_lda =  3
```

Out[40]:
```
KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
    n_clusters=3, n_init=10, n_jobs=1, precompute_distances='auto',
    random_state=1, tol=0.0001, verbose=0)
```

```
inertia =  4120.02063847
silhouette =  0.390843057284
```

Out[40]:
```
<matplotlib.figure.Figure at 0x7f4b77654a58>
```

Out[40]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4b7718a080>
```

Out[40]:
```
<matplotlib.collections.PathCollection at 0x7f4b80431780>
```

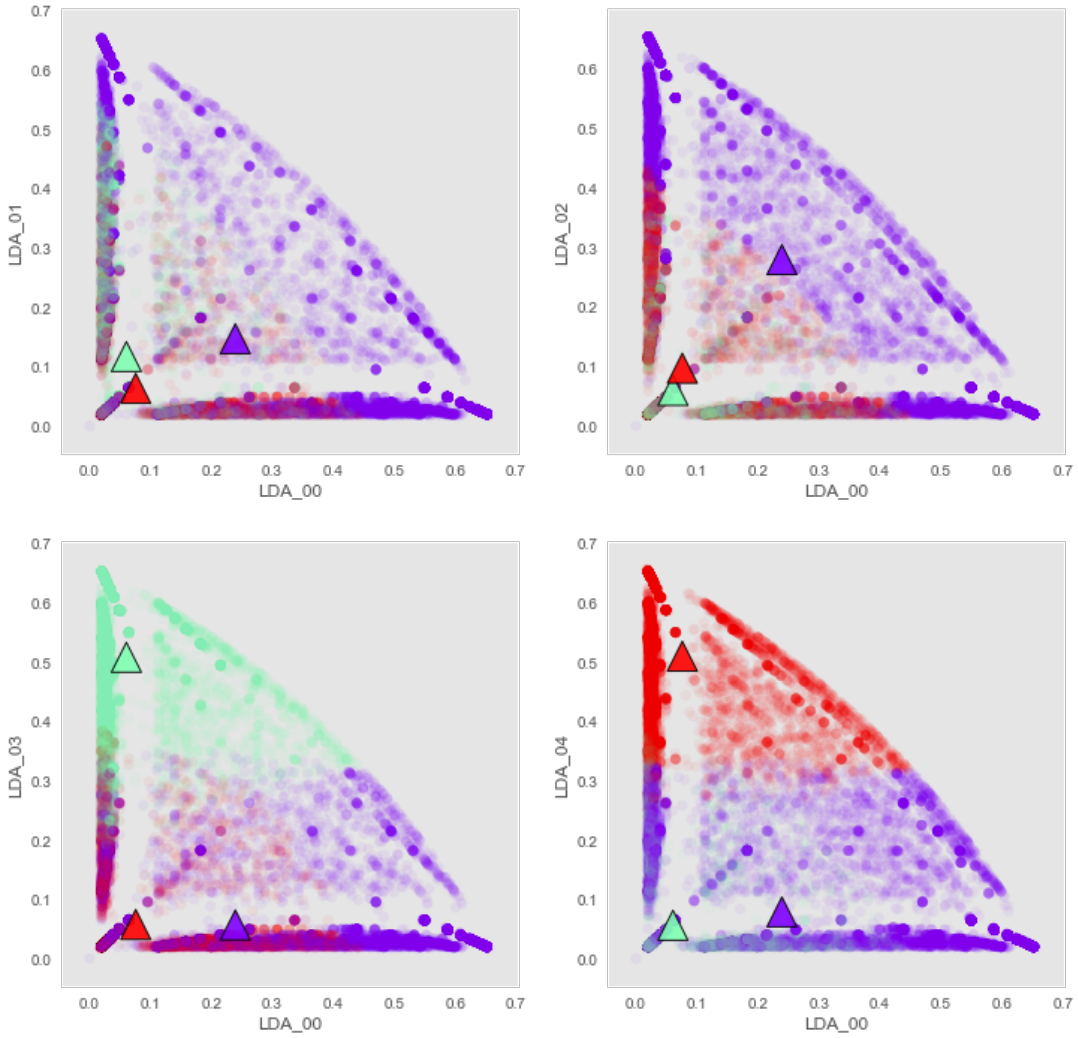
Out[40]:
```
<matplotlib.collections.PathCollection at 0x7f4b9054ab70>
```

Out[40]:
```
<matplotlib.text.Text at 0x7f4b9037aa90>
```

Out[40]:
```
(<matplotlib.text.Text at 0x7f4b77182a58>,
 <matplotlib.text.Text at 0x7f4b802ff3c8>)
```

```
Out[40]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b9037a7f0>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b7748a978>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b802d4b70>

Out[40]: (<matplotlib.text.Text at 0x7f4b774deba8>,
          <matplotlib.text.Text at 0x7f4b7777d128>)

Out[40]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b90316b70>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b77707550>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b8041d0b8>

Out[40]: (<matplotlib.text.Text at 0x7f4b905ba940>,
          <matplotlib.text.Text at 0x7f4b771d8c88>)

Out[40]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b9019a358>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b90521f28>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b8002e588>

Out[40]: (<matplotlib.text.Text at 0x7f4b77768518>,
          <matplotlib.text.Text at 0x7f4b80048c50>)
```

4120.02063847




```
n_lda =  4
```

Out[40]:
```
KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
    n_clusters=4, n_init=10, n_jobs=1, precompute_distances='auto',
    random_state=1, tol=0.0001, verbose=0)
```

```
inertia =  2523.83403252
silhouette =  0.497298881063
```

Out[40]:
```
<matplotlib.figure.Figure at 0x7f4b77423eb8>
```

Out[40]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4b90227e10>
```

Out[40]:
```
<matplotlib.collections.PathCollection at 0x7f4b8017a860>
```

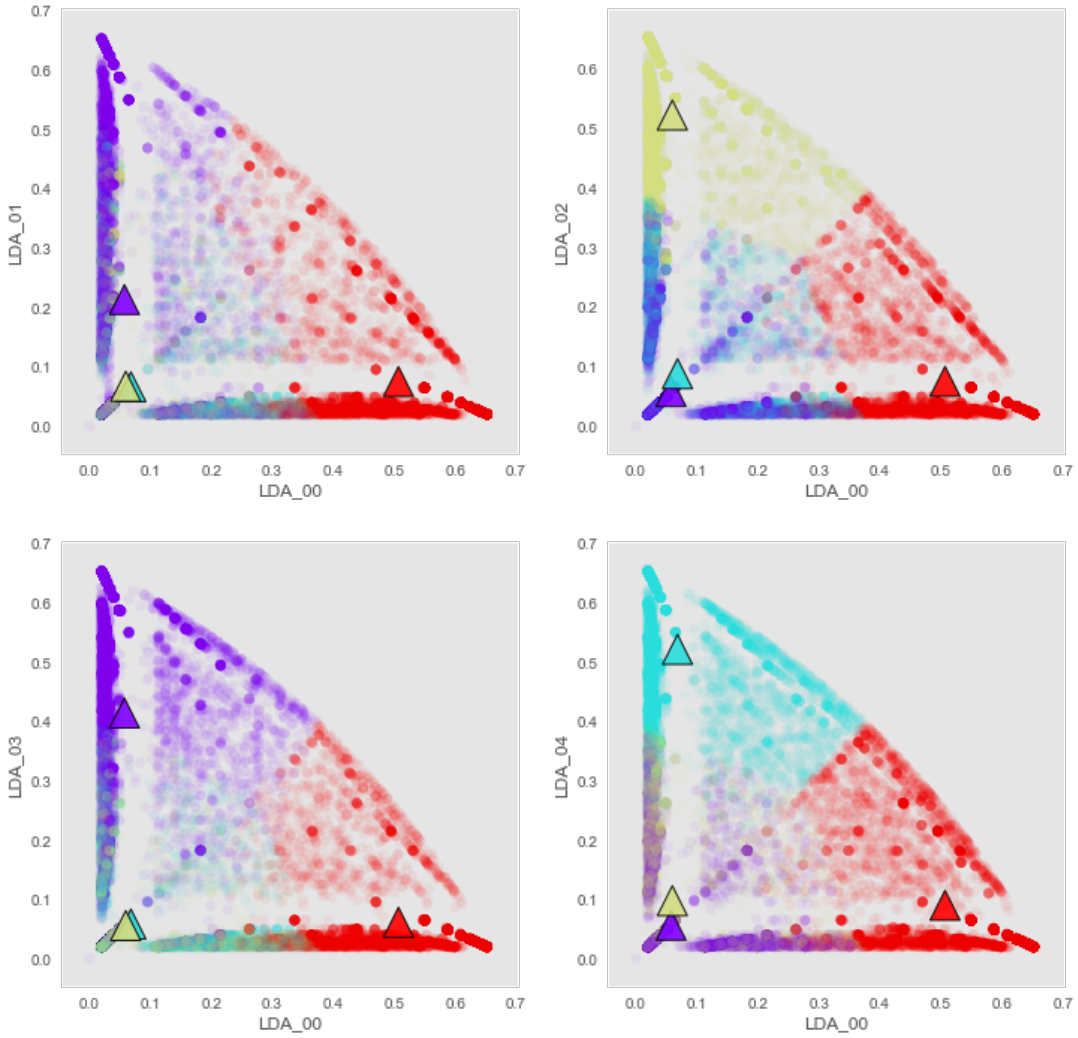
Out[40]:
```
<matplotlib.collections.PathCollection at 0x7f4b803912b0>
```

Out[40]:
```
<matplotlib.text.Text at 0x7f4b7746eda0>
```

Out[40]:
```
(<matplotlib.text.Text at 0x7f4b776d8668>,
 <matplotlib.text.Text at 0x7f4b8044c198>)
```

```
Out[40]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b90429be0>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b9050fdd8>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b9040ac88>

Out[40]: (<matplotlib.text.Text at 0x7f4b773e1b70>,
          <matplotlib.text.Text at 0x7f4b8058a390>)

Out[40]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b9050f908>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b80237a58>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b777363c8>

Out[40]: (<matplotlib.text.Text at 0x7f4b773e86d8>,
          <matplotlib.text.Text at 0x7f4b77477ac8>)

Out[40]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b802378d0>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b777484a8>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b77748cf8>

Out[40]: (<matplotlib.text.Text at 0x7f4b7746dc88>,
          <matplotlib.text.Text at 0x7f4b77472eb8>)
```

2523.83403252




```
n_lda =  5
```

Out[40]:
```
KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
    n_clusters=5, n_init=10, n_jobs=1, precompute_distances='auto',
    random_state=1, tol=0.0001, verbose=0)

inertia =  1418.81957692
silhouette =  0.564738207976
```

Out[40]:
```
<matplotlib.figure.Figure at 0x7f4b77654ba8>
```

Out[40]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4b776dd748>
```

Out[40]:
```
<matplotlib.collections.PathCollection at 0x7f4b7716fb38>
```

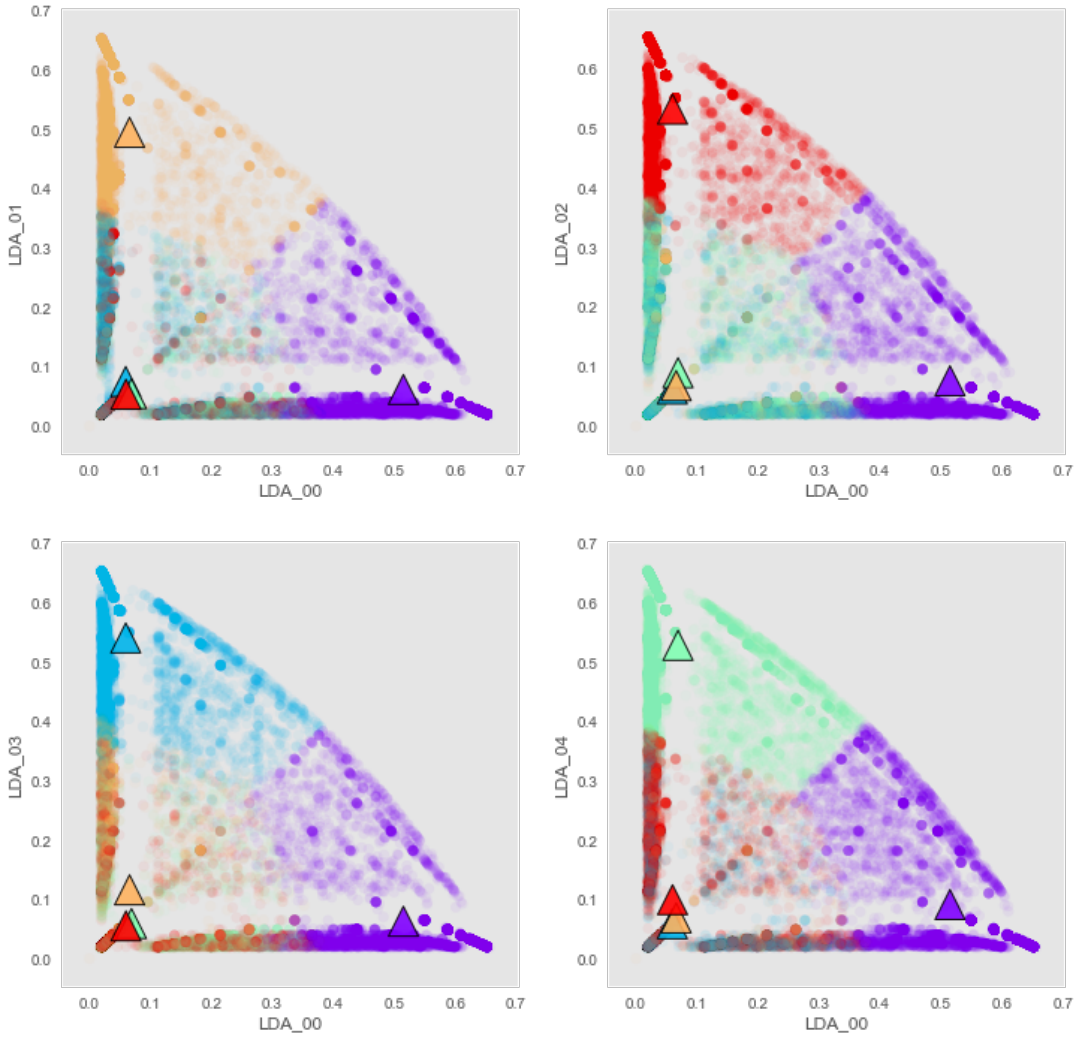
Out[40]:
```
<matplotlib.collections.PathCollection at 0x7f4b7713f3c8>
```

Out[40]:
```
<matplotlib.text.Text at 0x7f4b7716fb70>
```

Out[40]:
```
(<matplotlib.text.Text at 0x7f4b770c07b8>,
 <matplotlib.text.Text at 0x7f4b774c22e8>)
```

```
Out[40]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b7713ffd0>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b92cc3f60>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b92ccc7f0>

Out[40]: (<matplotlib.text.Text at 0x7f4b771581d0>,
          <matplotlib.text.Text at 0x7f4b770940b8>)

Out[40]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b92cd4198>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b92c5b240>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b92c5ba90>

Out[40]: (<matplotlib.text.Text at 0x7f4b92cdb1d0>,
          <matplotlib.text.Text at 0x7f4b770b8eb8>)

Out[40]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b92c60550>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b92c1ec50>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b92ba54e0>

Out[40]: (<matplotlib.text.Text at 0x7f4b92c47470>,
          <matplotlib.text.Text at 0x7f4b92c536a0>)
```

1418.81957692




```
n_lda =  6
```

Out[40]:
```
KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
    n_clusters=6, n_init=10, n_jobs=1, precompute_distances='auto',
    random_state=1, tol=0.0001, verbose=0)
```

```
inertia =  1257.64273723
silhouette =  0.521731190677
```

Out[40]:
```
<matplotlib.figure.Figure at 0x7f4b92b877f0>
```

Out[40]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4b92c7c898>
```

Out[40]:
```
<matplotlib.collections.PathCollection at 0x7f4b8006eef0>
```

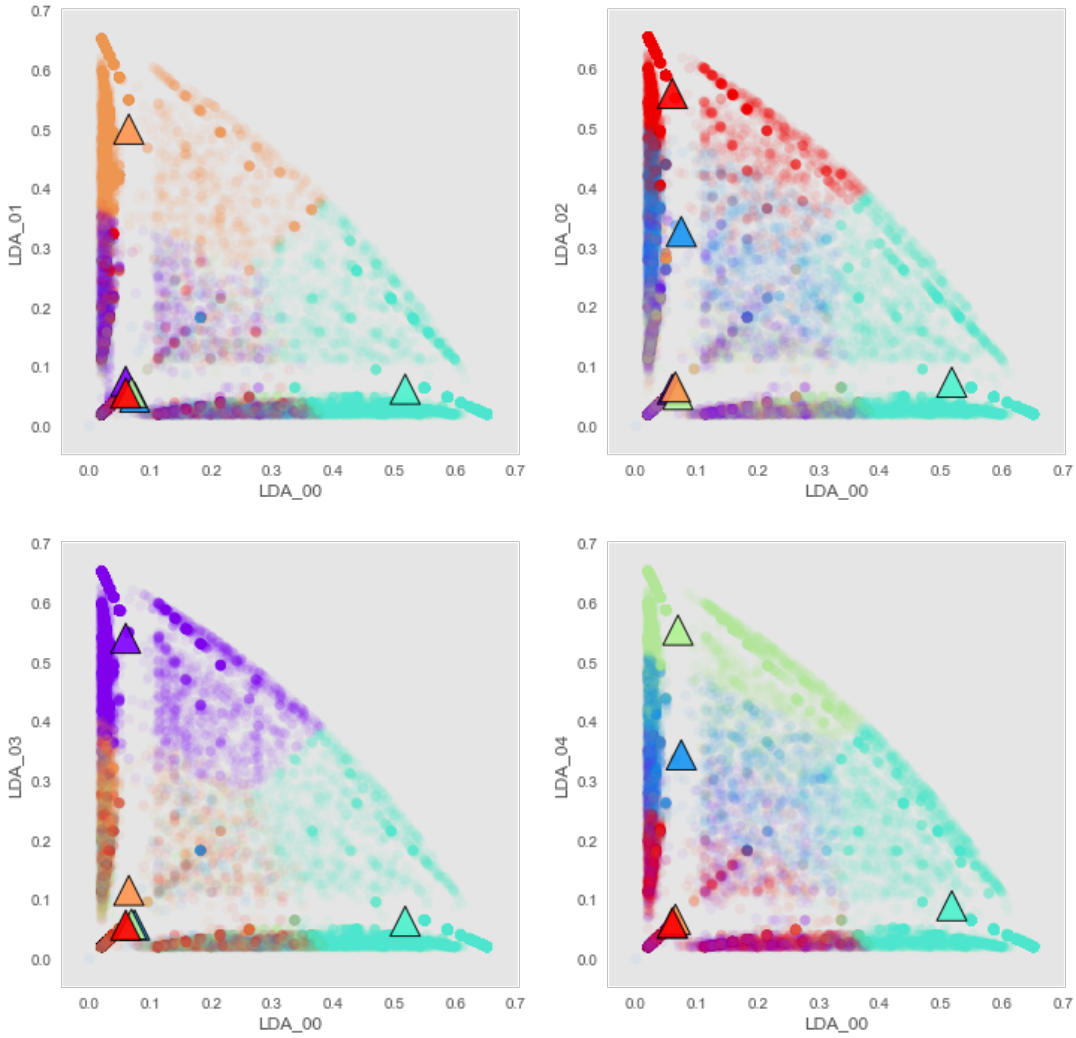
Out[40]:
```
<matplotlib.collections.PathCollection at 0x7f4b9040a470>
```

Out[40]:
```
<matplotlib.text.Text at 0x7f4b9050f518>
```

Out[40]:
```
(<matplotlib.text.Text at 0x7f4b92b6dc88>,
 <matplotlib.text.Text at 0x7f4b80070198>)
```

```
Out[40]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b9050fb38>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b771ed518>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b7713d7b8>

Out[40]: (<matplotlib.text.Text at 0x7f4b800707b8>,
          <matplotlib.text.Text at 0x7f4b9040a080>)

Out[40]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b773ba6a0>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b802f5668>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b806d6198>

Out[40]: (<matplotlib.text.Text at 0x7f4b771230f0>,
          <matplotlib.text.Text at 0x7f4b7773d748>)

Out[40]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b9250a7b8>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b90568828>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b900a50f0>

Out[40]: (<matplotlib.text.Text at 0x7f4b773b73c8>,
          <matplotlib.text.Text at 0x7f4b777487b8>)
```

1257.64273723




```
n_lda =  7
```

Out[40]:
```
KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
    n_clusters=7, n_init=10, n_jobs=1, precompute_distances='auto',
    random_state=1, tol=0.0001, verbose=0)
```

```
inertia =  1135.46696089
silhouette =  0.502022137869
```

Out[40]:
```
<matplotlib.figure.Figure at 0x7f4b92b87ac8>
```

Out[40]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4b92c154a8>
```

Out[40]:
```
<matplotlib.collections.PathCollection at 0x7f4b804047f0>
```

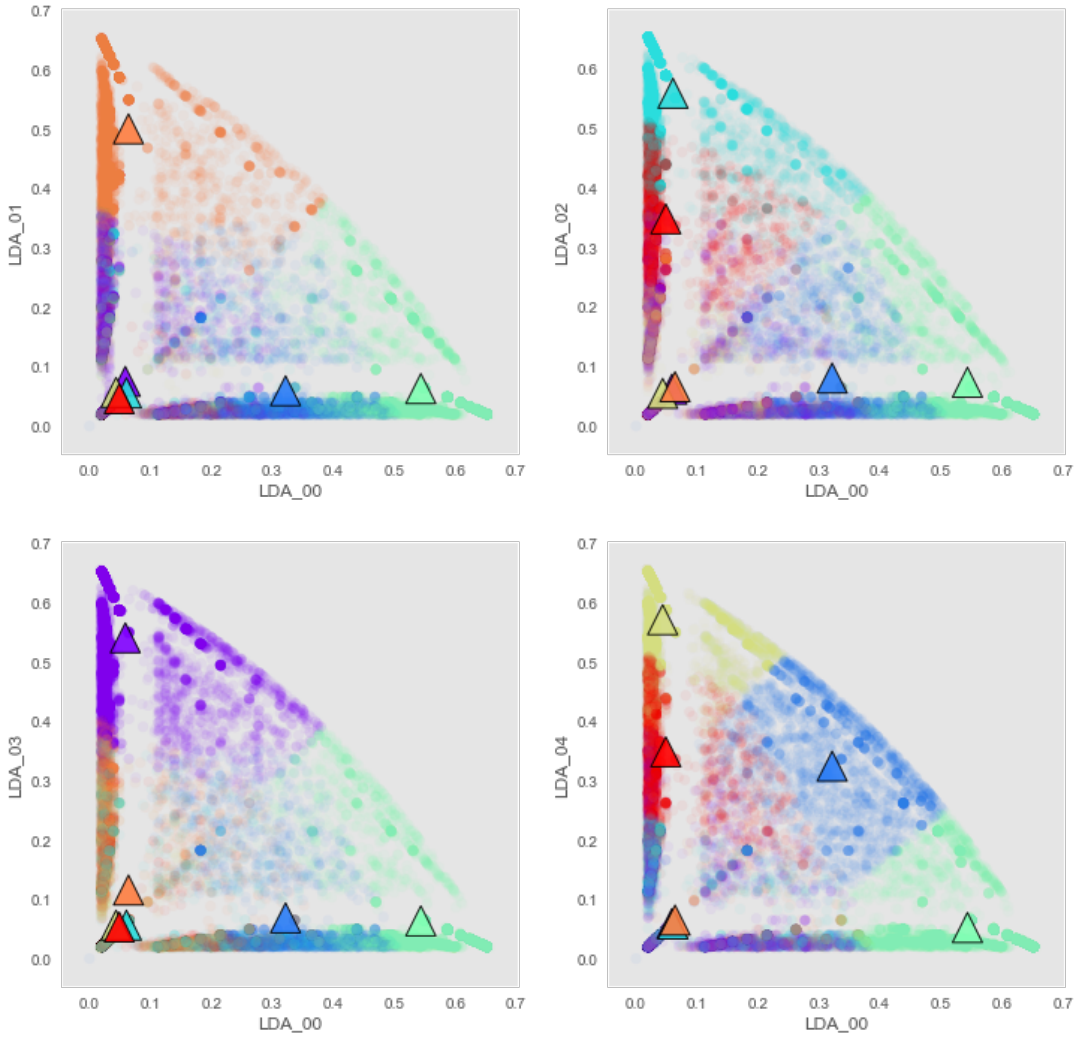
Out[40]:
```
<matplotlib.collections.PathCollection at 0x7f4b776dd1d0>
```

Out[40]:
```
<matplotlib.text.Text at 0x7f4b8015bc50>
```

Out[40]:
```
(<matplotlib.text.Text at 0x7f4b80065e10>,
 <matplotlib.text.Text at 0x7f4b773bb860>)
```

```
Out[40]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b804040b8>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b80242048>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b903a76a0>

Out[40]: (<matplotlib.text.Text at 0x7f4b776dd2e8>,
          <matplotlib.text.Text at 0x7f4b774252e8>)

Out[40]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b7775d438>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b776da128>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b77768160>

Out[40]: (<matplotlib.text.Text at 0x7f4b77786080>,
          <matplotlib.text.Text at 0x7f4b77454978>)

Out[40]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b776da588>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b804d1f60>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b90521240>

Out[40]: (<matplotlib.text.Text at 0x7f4b90119400>,
          <matplotlib.text.Text at 0x7f4b771a1c18>)
```

1135.46696089




```
n_lda =  8
```

Out[40]:
```
KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
    n_clusters=8, n_init=10, n_jobs=1, precompute_distances='auto',
    random_state=1, tol=0.0001, verbose=0)
```

```
inertia =  1023.64621652
silhouette =  0.476994076662
```

Out[40]:
```
<matplotlib.figure.Figure at 0x7f4b77059dd8>
```

Out[40]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4b92bcebe0>
```

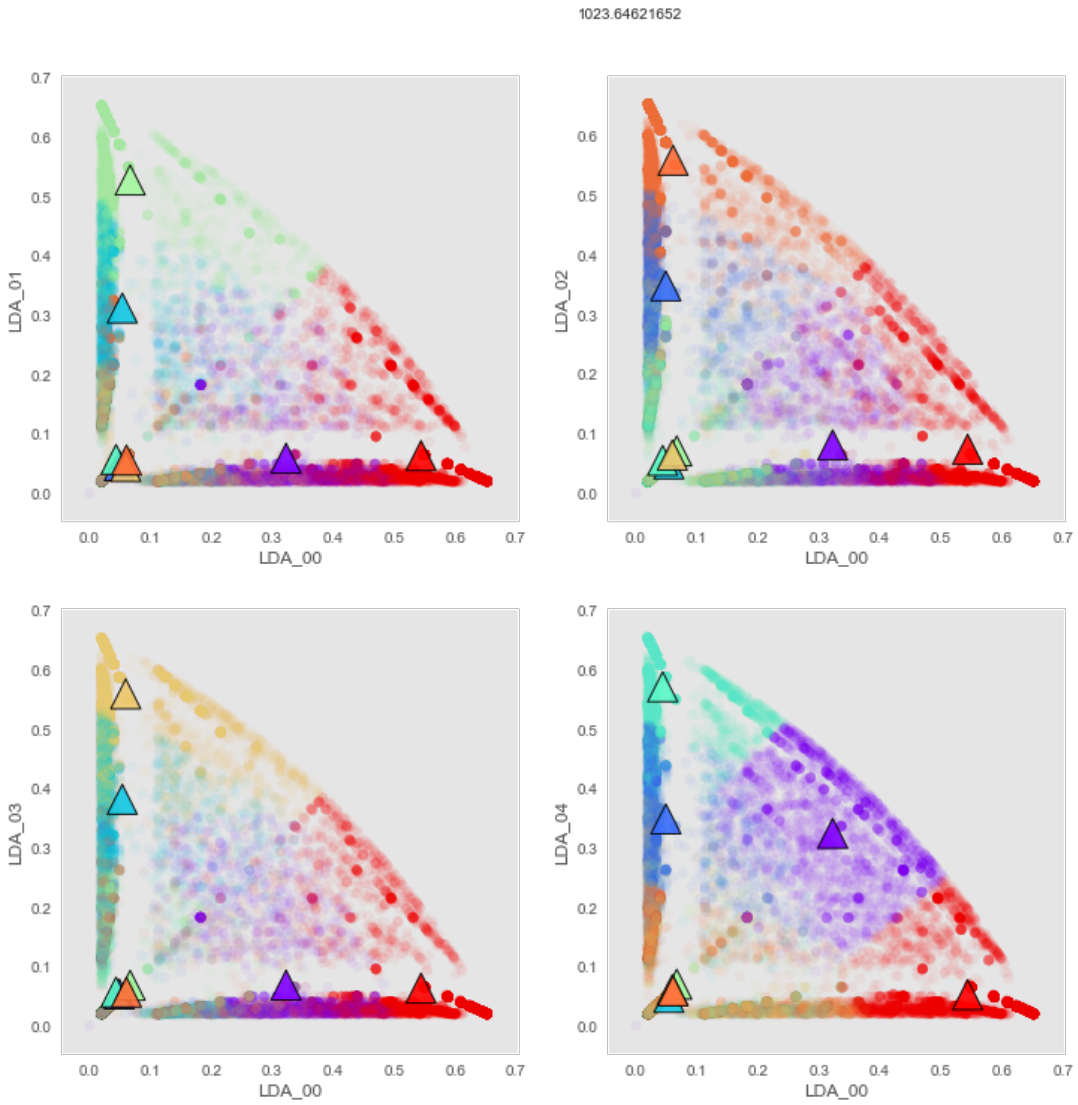
Out[40]:
```
<matplotlib.collections.PathCollection at 0x7f4b80040c50>
```

Out[40]:
```
<matplotlib.collections.PathCollection at 0x7f4b800054e0>
```

Out[40]:
```
<matplotlib.text.Text at 0x7f4b80040c88>
```

Out[40]:
```
(<matplotlib.text.Text at 0x7f4b777541d0>,
 <matplotlib.text.Text at 0x7f4b8059be80>)
```

```
Out[40]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b8021b2b0>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b77591588>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b775aa080>

Out[40]: (<matplotlib.text.Text at 0x7f4b775cd1d0>,
          <matplotlib.text.Text at 0x7f4b777c1128>)

Out[40]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b77591e10>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b77368978>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b7735f278>

Out[40]: (<matplotlib.text.Text at 0x7f4b77648908>,
          <matplotlib.text.Text at 0x7f4b777eb630>)

Out[40]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b773687f0>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b90239dd8>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b92668b38>

Out[40]: (<matplotlib.text.Text at 0x7f4b77186ba8>,
          <matplotlib.text.Text at 0x7f4b771acdd8>)
```




```
n_lda =  9
```

Out[40]:
```
KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
    n_clusters=9, n_init=10, n_jobs=1, precompute_distances='auto',
    random_state=1, tol=0.0001, verbose=0)
```

```
inertia =  937.873675896
silhouette =  0.477307505862
```

Out[40]:
```
<matplotlib.figure.Figure at 0x7f4b90377e80>
```

Out[40]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4b801ea860>
```

Out[40]:
```
<matplotlib.collections.PathCollection at 0x7f4b7708e828>
```

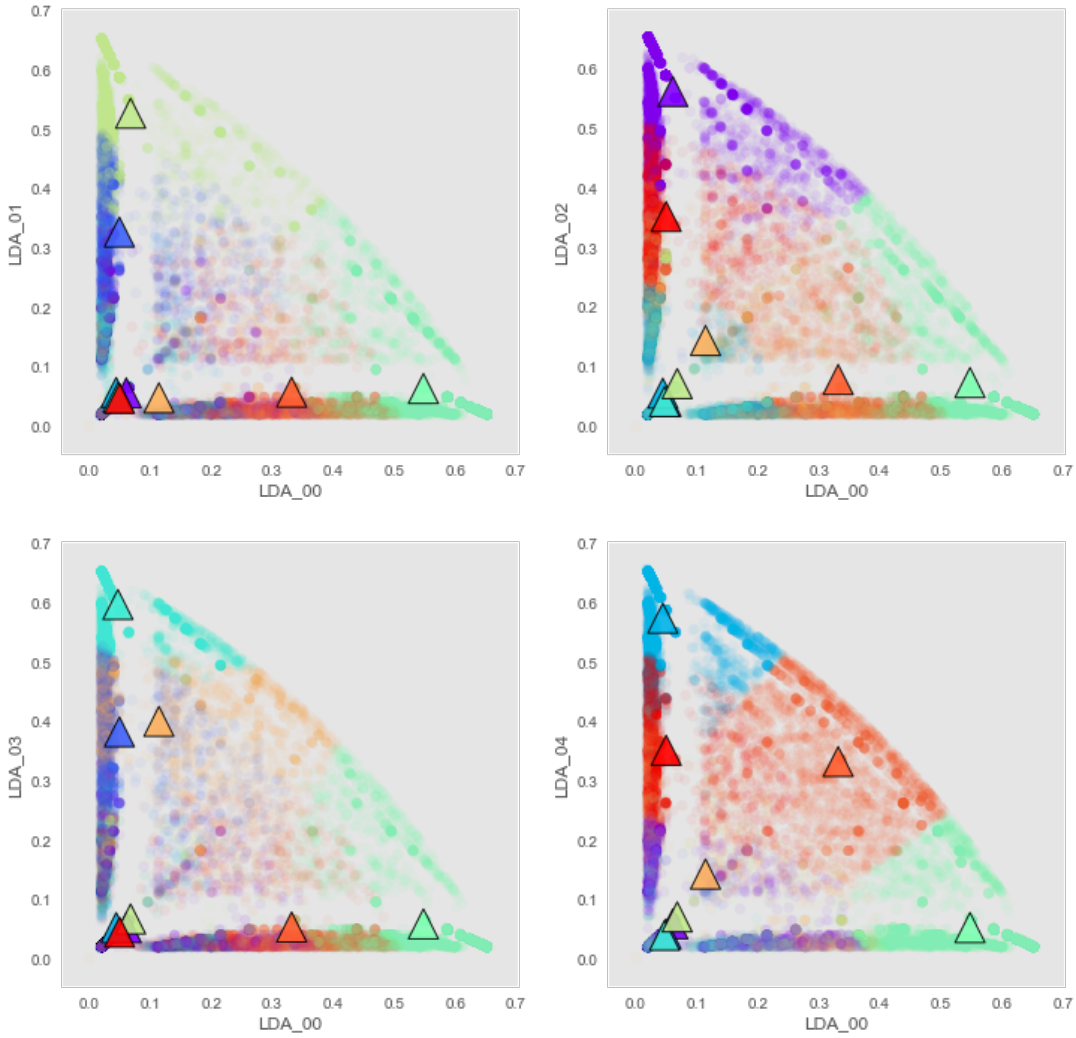
Out[40]:
```
<matplotlib.collections.PathCollection at 0x7f4b800274a8>
```

Out[40]:
```
<matplotlib.text.Text at 0x7f4b90355d30>
```

Out[40]:
```
(<matplotlib.text.Text at 0x7f4b775cd780>,
 <matplotlib.text.Text at 0x7f4b7731d5c0>)
```

```
Out[40]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b904e46d8>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b777f48d0>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b9852c0f0>

Out[40]: (<matplotlib.text.Text at 0x7f4b7755b2b0>,
          <matplotlib.text.Text at 0x7f4b77472ac8>)

Out[40]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b807357f0>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b9256b908>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b7774f668>

Out[40]: (<matplotlib.text.Text at 0x7f4b80526198>,
          <matplotlib.text.Text at 0x7f4b776e4d68>)

Out[40]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b9256b9e8>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b802d30b8>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b77466550>

Out[40]: (<matplotlib.text.Text at 0x7f4b774e2f60>,
          <matplotlib.text.Text at 0x7f4b805489b0>)
```

937.873675896




```
n_lda =  10
```

Out[40]:
```
KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
    n_clusters=10, n_init=10, n_jobs=1, precompute_distances='auto',
    random_state=1, tol=0.0001, verbose=0)
```

```
inertia =  859.164016267
silhouette =  0.483459889792
```

Out[40]:
```
<matplotlib.figure.Figure at 0x7f4b7778cb38>
```

Out[40]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4b7757ff60>
```

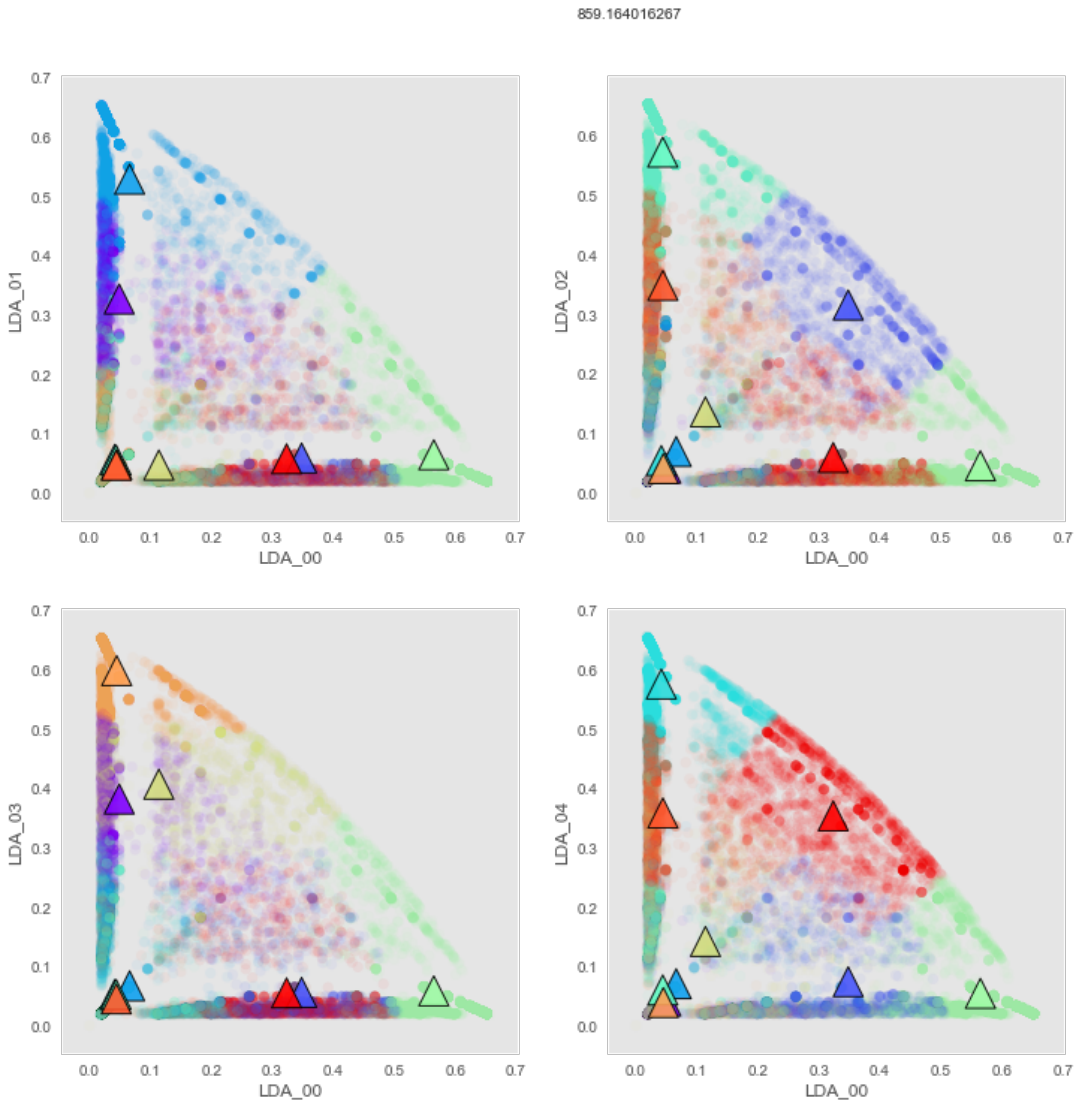
Out[40]:
```
<matplotlib.collections.PathCollection at 0x7f4b7775d4a8>
```

Out[40]:
```
<matplotlib.collections.PathCollection at 0x7f4b770f10f0>
```

Out[40]:
```
<matplotlib.text.Text at 0x7f4b770e45c0>
```

Out[40]:
```
(<matplotlib.text.Text at 0x7f4b8048b400>,
 <matplotlib.text.Text at 0x7f4b900a5f60>)
```

```
Out[40]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b8048bf28>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b770984a8>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b776d15c0>

Out[40]: (<matplotlib.text.Text at 0x7f4b776dd828>,
          <matplotlib.text.Text at 0x7f4b802380f0>)

Out[40]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b77098828>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b773e1c88>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b77737ef0>

Out[40]: (<matplotlib.text.Text at 0x7f4b7747a128>,
          <matplotlib.text.Text at 0x7f4b7718aa20>)

Out[40]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b773a8710>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b777b0fd0>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b777b0e48>

Out[40]: (<matplotlib.text.Text at 0x7f4b777427f0>,
          <matplotlib.text.Text at 0x7f4b77748b38>)
```




```
n_lda =  11
```

Out[40]:
```
KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
    n_clusters=11, n_init=10, n_jobs=1, precompute_distances='auto',
    random_state=1, tol=0.0001, verbose=0)
```

```
inertia =  794.23042098
silhouette =  0.48590635357
```

Out[40]:
```
<matplotlib.figure.Figure at 0x7f4b92559dd8>
```

Out[40]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4b7779c780>
```

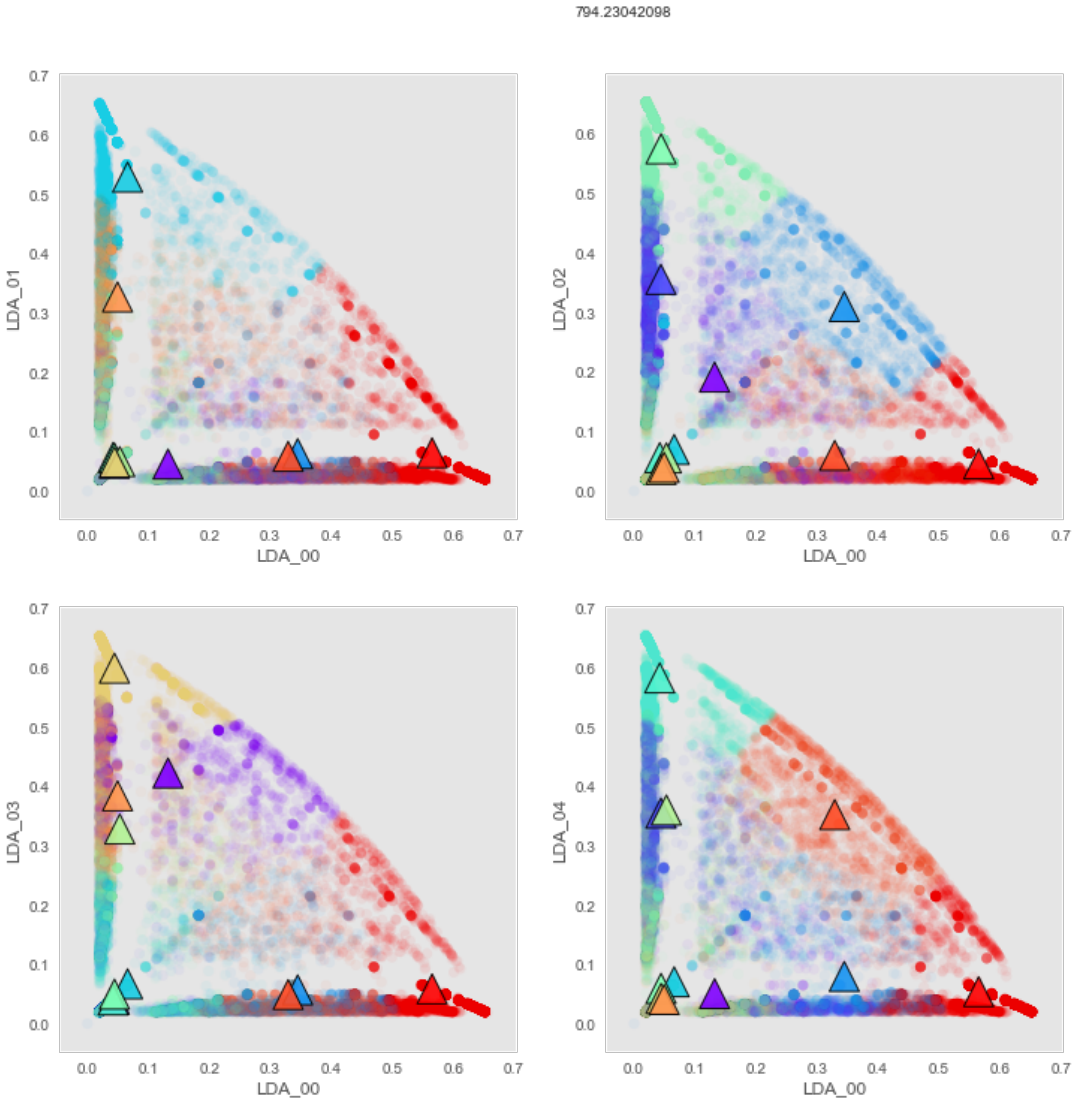
Out[40]:
```
<matplotlib.collections.PathCollection at 0x7f4b776116a0>
```

Out[40]:
```
<matplotlib.collections.PathCollection at 0x7f4b7763c160>
```

Out[40]:
```
<matplotlib.text.Text at 0x7f4b77611c18>
```

Out[40]:
```
(<matplotlib.text.Text at 0x7f4b775efcc0>,
 <matplotlib.text.Text at 0x7f4b92b7bd30>)
```

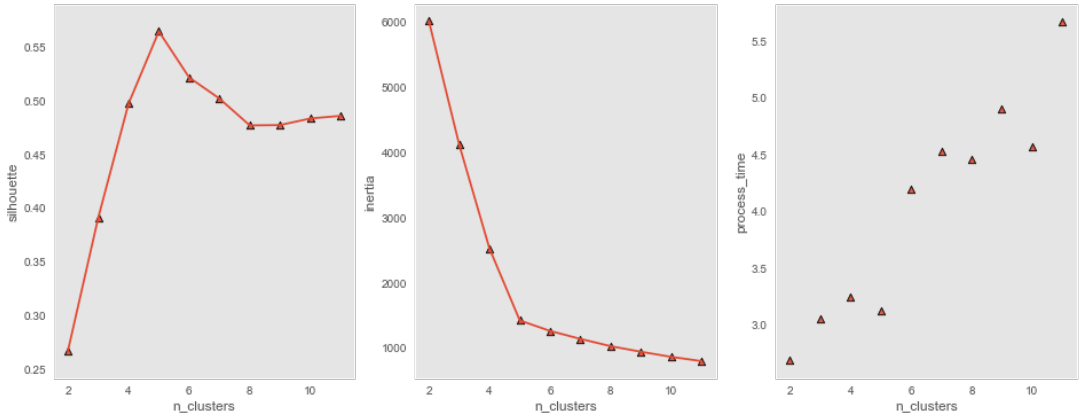
```
Out[40]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b7763cc18>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b775ca240>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b775ef160>

Out[40]: (<matplotlib.text.Text at 0x7f4b77685080>,
          <matplotlib.text.Text at 0x7f4b77748cf8>)

Out[40]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b775c6978>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b77788470>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b773592e8>

Out[40]: (<matplotlib.text.Text at 0x7f4b77134748>,
          <matplotlib.text.Text at 0x7f4b803a8b70>)

Out[40]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b92b7e208>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b92b64828>

Out[40]: <matplotlib.collections.PathCollection at 0x7f4b92bc25c0>

Out[40]: (<matplotlib.text.Text at 0x7f4b770e6048>,
          <matplotlib.text.Text at 0x7f4b770e3ac8>)
```



Out[40]:

	model_name	n_clusters	inertia	silhouette	process_time
1	KMeans - LDA features	2	6016.621217	0.266660	2.6897
1	KMeans - LDA features	3	4120.020638	0.390843	3.0500
1	KMeans - LDA features	4	2523.834033	0.497299	3.2474
1	KMeans - LDA features	5	1418.819577	0.564738	3.1258
1	KMeans - LDA features	6	1257.642737	0.521731	4.1921
1	KMeans - LDA features	7	1135.466961	0.502022	4.5223
1	KMeans - LDA features	8	1023.646217	0.476994	4.4569
1	KMeans - LDA features	9	937.873676	0.477308	4.9037
1	KMeans - LDA features	10	859.164016	0.483460	4.5667
1	KMeans - LDA features	11	794.230421	0.485906	5.6639

In [41]:
```
# ... -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
# ... - plot metrics across models for comparison
# ... -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-

plt.figure(figsize=(16, 6));

# ... silhouette values

plt.subplot(131);
plt.scatter(comparison_tbl['n_clusters'],
            comparison_tbl['silhouette'],
            s = 40,
            linewidths = 1.0,
            marker = '^',
            edgecolors = 'black',
            alpha = 0.90);

plt.plot(comparison_tbl['n_clusters'],
         comparison_tbl['silhouette'])

plt.xlabel('n_clusters'), plt.ylabel('silhouette');
plt.grid();

# ... inertia values

plt.subplot(132);
plt.scatter(comparison_tbl['n_clusters'],
            comparison_tbl['inertia'],
            s = 40,
            linewidths = 1.0,
            marker = '^',
            edgecolors = 'black',
            alpha = 0.90);

plt.plot(comparison_tbl['n_clusters'],
         comparison_tbl['inertia'])

plt.xlabel('n_clusters'), plt.ylabel('inertia');
plt.grid();

# ... process time

plt.subplot(133);
plt.scatter(comparison_tbl['n_clusters'],
            comparison_tbl['process_time'],
            s = 40,
            linewidths = 1.0,
            marker = '^',
            edgecolors = 'black',
            alpha = 0.90);

#plt.plot(comparison_tbl['n_clusters'],
#         comparison_tbl['process_time'])

plt.xlabel('n_clusters'), plt.ylabel('process_time');
```



K-Means - images & videos

In [42]:
```
for n_lda in range(2, 10):

    tic = time.clock()

    X1 = df_cluster[['ln_num_imgs','ln_num_videos', 'ln_num_hrefs']]

    cls_lda = KMeans(n_clusters = n_lda,
                     init = 'k-means++',
                     random_state = 1)

    cls_lda.fit(X1)

    kmeans_labels = cls_lda.labels_ # the labels from kmeans clustering
    kmeans_centers = cls_lda.cluster_centers_
    kmeans_inertia = cls_lda.inertia_

    print ("n_lda = ", n_lda)
    print ("inertia = ", kmeans_inertia)

    kmeans_silhouette = metrics.silhouette_score(X1,
                                                 kmeans_labels,
                                                 metric = 'euclidean',
                                                 sample_size = 10000)
    print ("silhouette = ", kmeans_silhouette)

    toc =  time.clock()

# ... -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
# ... - save statistics for model comparison
# ... -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-

    exe_time = '{0:.4f}'.format(toc-tic)

    raw_data = {
    'model_name' : 'KMeans - images_videos_hrefs features',
    'n_clusters' : n_lda,
    'inertia': kmeans_inertia,
    'silhouette': kmeans_silhouette,
    'process_time' : exe_time
    }

    df_tbl = pd.DataFrame(raw_data,
    columns = ['model_name', 'n_clusters', 'inertia', 'silhouette', 'process_time'],
    index = [i_index + 1])

    comparison_tbl = comparison_tbl.append(df_tbl)

# ... -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
# ... - make some plots of clusters
# ... -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-

    plt.figure(figsize = (16, 6));

    plt.subplot(131);
    X1 = X1.values;
```

Out[42]:
```
KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
    n_clusters=2, n_init=10, n_jobs=1, precompute_distances='auto',
    random_state=1, tol=0.0001, verbose=0)

n_lda =  2
inertia =  49187.3739481
silhouette =  0.458518171368
```

Out[42]:
```
<matplotlib.figure.Figure at 0x7f4b7766bc18>
```

Out[42]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4b92729be0>
```

Out[42]:
```
<matplotlib.collections.PathCollection at 0x7f4b7718ab70>
```

Out[42]:
```
<matplotlib.collections.PathCollection at 0x7f4b80028c88>
```

Out[42]:
```
<matplotlib.text.Text at 0x7f4b9032bfd0>
```

Out[42]:
```
(<matplotlib.text.Text at 0x7f4b777fd940>,
 <matplotlib.text.Text at 0x7f4b80132e80>)
```

Out[42]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4b9032bb00>
```

Out[42]:
```
<matplotlib.collections.PathCollection at 0x7f4b776daa20>
```

Out[42]:
```
<matplotlib.collections.PathCollection at 0x7f4b77768390>
```

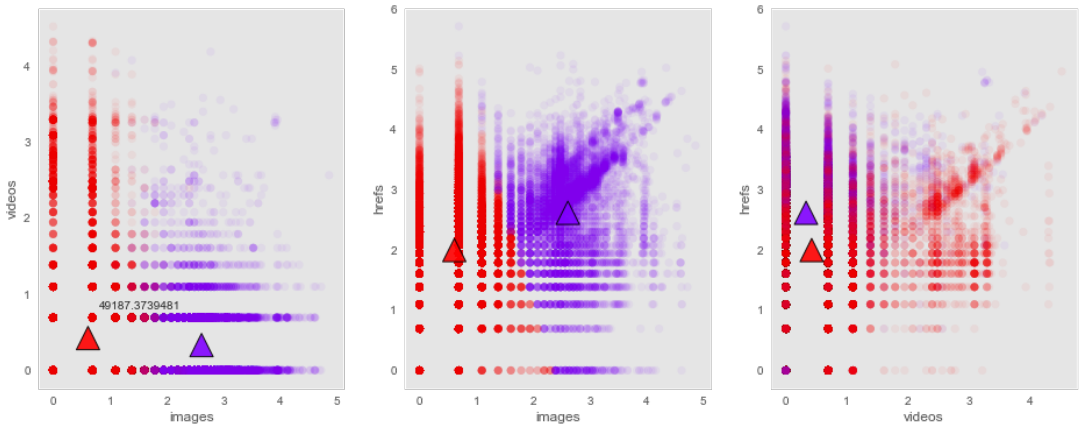
Out[42]:
```
(<matplotlib.text.Text at 0x7f4b8059b278>,
 <matplotlib.text.Text at 0x7f4b774a8eb8>)
```

Out[42]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4b801cd240>
```

Out[42]:
```
<matplotlib.collections.PathCollection at 0x7f4b80404eb8>
```

Out[42]:
```
<matplotlib.collections.PathCollection at 0x7f4b771c4588>
```

Out[42]:
```
(<matplotlib.text.Text at 0x7f4b775745f8>,
 <matplotlib.text.Text at 0x7f4b7757f860>)
```



Out[42]:
```
KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
    n_clusters=3, n_init=10, n_jobs=1, precompute_distances='auto',
    random_state=1, tol=0.0001, verbose=0)

n_lda =  3
inertia =  37625.0279826
silhouette =  0.47886917581
```

Out[42]: <matplotlib.figure.Figure at 0x7f4b902fa208>

Out[42]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b80041ac8>

Out[42]: <matplotlib.collections.PathCollection at 0x7f4b80020c18>

Out[42]: <matplotlib.collections.PathCollection at 0x7f4b800444a8>

Out[42]: <matplotlib.text.Text at 0x7f4b80020c50>

Out[42]: (<matplotlib.text.Text at 0x7f4b776d1828>,
 <matplotlib.text.Text at 0x7f4b776e4f60>)

Out[42]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b802f45c0>

Out[42]: <matplotlib.collections.PathCollection at 0x7f4b7774a4e0>

Out[42]: <matplotlib.collections.PathCollection at 0x7f4b7774ad30>

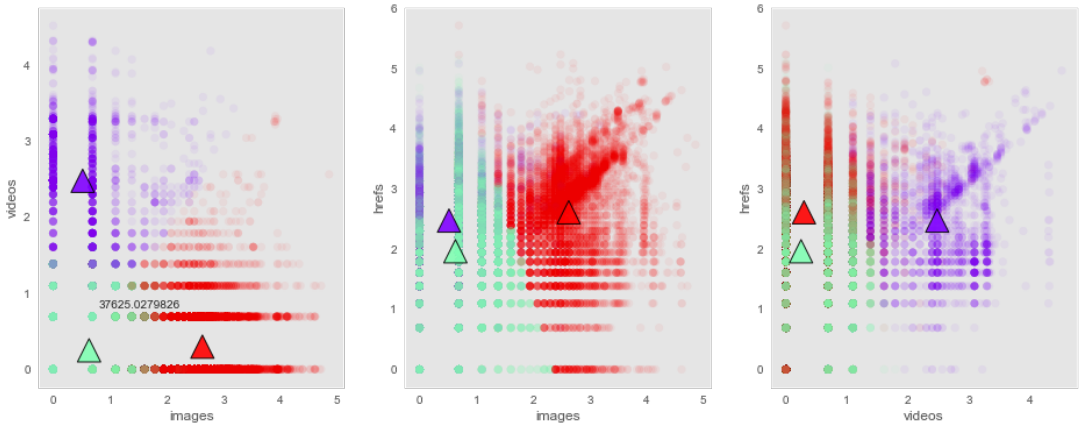
Out[42]: (<matplotlib.text.Text at 0x7f4b776e40b8>,
 <matplotlib.text.Text at 0x7f4b80225ef0>)

Out[42]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b775a56d8>

Out[42]: <matplotlib.collections.PathCollection at 0x7f4b775cd828>

Out[42]: <matplotlib.collections.PathCollection at 0x7f4b77382710>

Out[42]: (<matplotlib.text.Text at 0x7f4b77599710>,
 <matplotlib.text.Text at 0x7f4b8022de10>)



Out[42]: KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
    n_clusters=4, n_init=10, n_jobs=1, precompute_distances='auto',
    random_state=1, tol=0.0001, verbose=0)

n_lda = 4
inertia = 28917.9668688
silhouette = 0.34285776496

Out[42]: <matplotlib.figure.Figure at 0x7f4b776da278>

Out[42]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b774445f8>

Out[42]: <matplotlib.collections.PathCollection at 0x7f4b8059b0b8>

Out[42]: <matplotlib.collections.PathCollection at 0x7f4b7718aba8>

Out[42]: <matplotlib.text.Text at 0x7f4b9032b128>

Out[42]: (<matplotlib.text.Text at 0x7f4b9003f518>,
<matplotlib.text.Text at 0x7f4b776bf2b0>)

Out[42]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b77239630>

Out[42]: <matplotlib.collections.PathCollection at 0x7f4b80027160>

Out[42]: <matplotlib.collections.PathCollection at 0x7f4b9039e2b0>

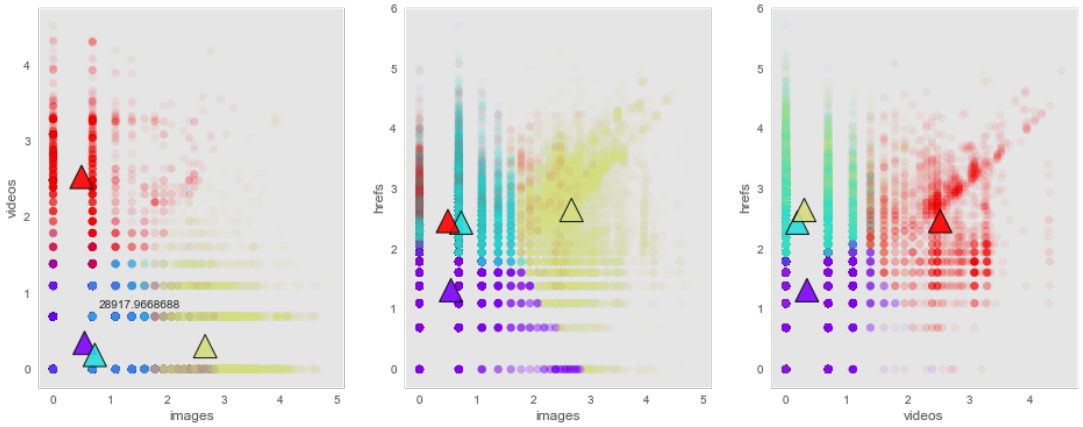
Out[42]: (<matplotlib.text.Text at 0x7f4b7743fac8>,
<matplotlib.text.Text at 0x7f4b80132390>)

Out[42]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b904fcdd8>

Out[42]: <matplotlib.collections.PathCollection at 0x7f4b771be780>

Out[42]: <matplotlib.collections.PathCollection at 0x7f4b905ba2b0>

Out[42]: (<matplotlib.text.Text at 0x7f4b775e5470>,
<matplotlib.text.Text at 0x7f4b777fdc18>)



Out[42]: KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
n_clusters=5, n_init=10, n_jobs=1, precompute_distances='auto',
random_state=1, tol=0.0001, verbose=0)

n_lda = 5
inertia = 24309.3293114
silhouette = 0.343457963649

Out[42]: <matplotlib.figure.Figure at 0x7f4b776dacc0>

Out[42]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b92737710>

Out[42]: <matplotlib.collections.PathCollection at 0x7f4b770e52b0>

Out[42]: <matplotlib.collections.PathCollection at 0x7f4b90600ac8>

Out[42]: <matplotlib.text.Text at 0x7f4b770e5ac8>

Out[42]: (<matplotlib.text.Text at 0x7f4b804f3cf8>,
<matplotlib.text.Text at 0x7f4b8048bef0>)

Out[42]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b773a89b0>

Out[42]: <matplotlib.collections.PathCollection at 0x7f4b77311898>

Out[42]: <matplotlib.collections.PathCollection at 0x7f4b90236748>

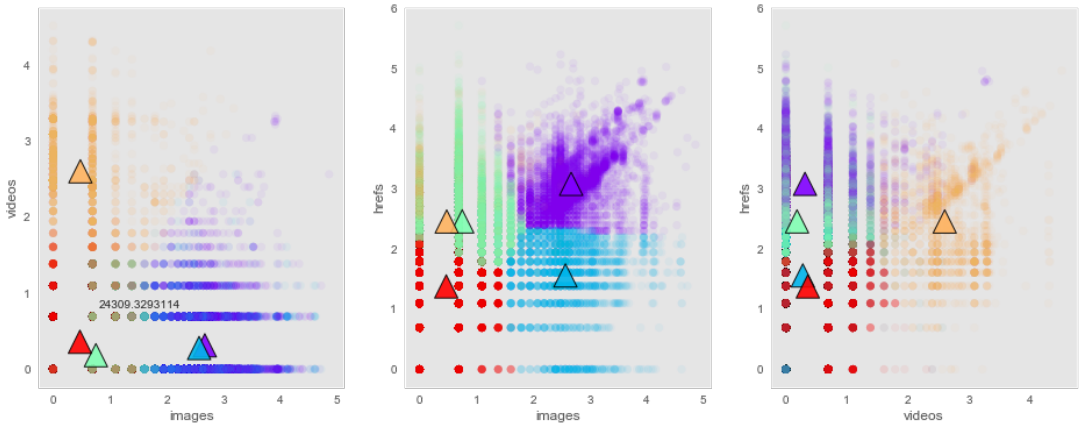
Out[42]: (<matplotlib.text.Text at 0x7f4b8058af98>,
 <matplotlib.text.Text at 0x7f4b90433550>)

Out[42]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b776e70b8>

Out[42]: <matplotlib.collections.PathCollection at 0x7f4b77425e48>

Out[42]: <matplotlib.collections.PathCollection at 0x7f4b776b1390>

Out[42]: (<matplotlib.text.Text at 0x7f4b777079b0>,
 <matplotlib.text.Text at 0x7f4b774e28d0>)



Out[42]: KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
    n_clusters=6, n_init=10, n_jobs=1, precompute_distances='auto',
    random_state=1, tol=0.0001, verbose=0)

```
n_lda =  6
inertia =  21231.9882546
silhouette =  0.322695512927
```

Out[42]: <matplotlib.figure.Figure at 0x7f4b771e0da0>

Out[42]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b905b56d8>

Out[42]: <matplotlib.collections.PathCollection at 0x7f4b774545f8>

Out[42]: <matplotlib.collections.PathCollection at 0x7f4b77466390>

Out[42]: <matplotlib.text.Text at 0x7f4b77454278>

Out[42]: (<matplotlib.text.Text at 0x7f4b7772b0b8>,
 <matplotlib.text.Text at 0x7f4b92bc20b8>)

Out[42]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b774546a0>

Out[42]: <matplotlib.collections.PathCollection at 0x7f4b775ffb00>

Out[42]: <matplotlib.collections.PathCollection at 0x7f4b77623390>

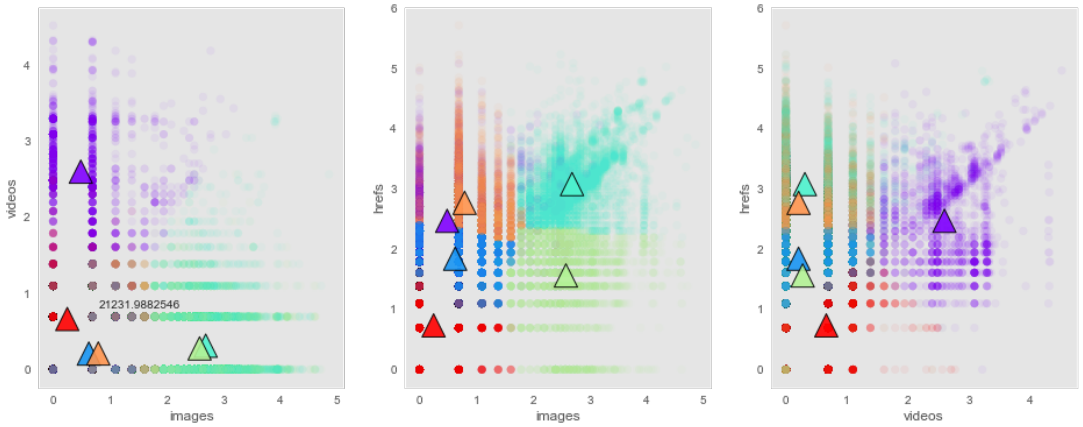
Out[42]: (<matplotlib.text.Text at 0x7f4b777e4160>,
 <matplotlib.text.Text at 0x7f4b7709e6d8>)

Out[42]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b775ffb38>

Out[42]: <matplotlib.collections.PathCollection at 0x7f4b770fbc50>

Out[42]: <matplotlib.collections.PathCollection at 0x7f4b770ea630>

Out[42]: (<matplotlib.text.Text at 0x7f4b77620d30>,
 <matplotlib.text.Text at 0x7f4b925975f8>)



Out[42]: KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
    n_clusters=7, n_init=10, n_jobs=1, precompute_distances='auto',
    random_state=1, tol=0.0001, verbose=0)

n_lda = 7
inertia = 18723.7392632
silhouette = 0.346752276222

Out[42]: <matplotlib.figure.Figure at 0x7f4b776c8c88>

Out[42]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b90072470>

Out[42]: <matplotlib.collections.PathCollection at 0x7f4b77139128>

Out[42]: <matplotlib.collections.PathCollection at 0x7f4b7712b198>

Out[42]: <matplotlib.text.Text at 0x7f4b77139400>

Out[42]: (<matplotlib.text.Text at 0x7f4b77647f28>,
 <matplotlib.text.Text at 0x7f4b7764a470>)

Out[42]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b777ea0f0>

Out[42]: <matplotlib.collections.PathCollection at 0x7f4b770d5dd8>

Out[42]: <matplotlib.collections.PathCollection at 0x7f4b7764add8>

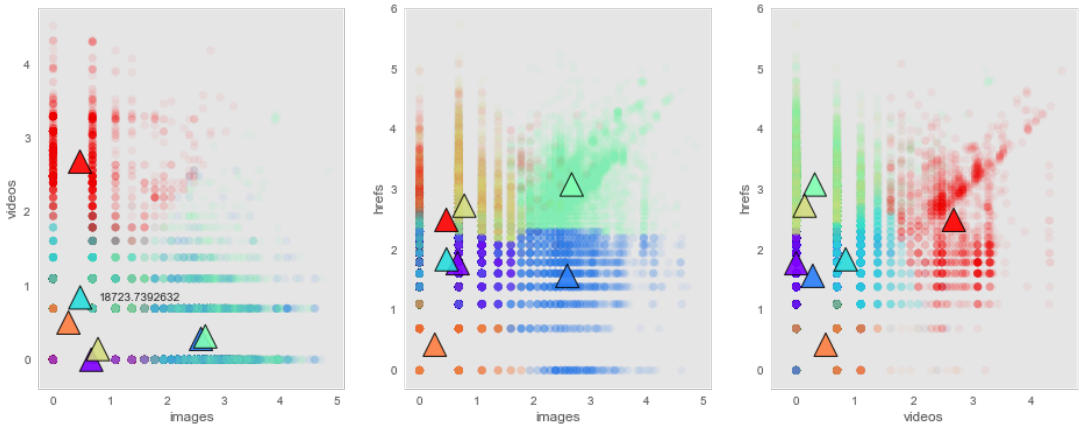
Out[42]: (<matplotlib.text.Text at 0x7f4b92b77518>,
 <matplotlib.text.Text at 0x7f4b776a6ef0>)

Out[42]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b770d5198>

Out[42]: <matplotlib.collections.PathCollection at 0x7f4b776b4b70>

Out[42]: <matplotlib.collections.PathCollection at 0x7f4b77466748>

Out[42]: (<matplotlib.text.Text at 0x7f4b7764a0f0>,
 <matplotlib.text.Text at 0x7f4b773e0438>)



Out[42]:
```
KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
    n_clusters=8, n_init=10, n_jobs=1, precompute_distances='auto',
    random_state=1, tol=0.0001, verbose=0)

n_lda =  8
inertia =  17210.2410622
silhouette =  0.328824160966
```

Out[42]:
```
<matplotlib.figure.Figure at 0x7f4b773f5a58>
```

Out[42]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4b90159748>
```

Out[42]:
```
<matplotlib.collections.PathCollection at 0x7f4b773d60f0>
```

Out[42]:
```
<matplotlib.collections.PathCollection at 0x7f4b773e1400>
```

Out[42]:
```
<matplotlib.text.Text at 0x7f4b773d6898>
```

Out[42]:
```
(<matplotlib.text.Text at 0x7f4b92bdd710>,
 <matplotlib.text.Text at 0x7f4b92b7c198>)
```

Out[42]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4b773d6908>
```

Out[42]:
```
<matplotlib.collections.PathCollection at 0x7f4b92b8e908>
```

Out[42]:
```
<matplotlib.collections.PathCollection at 0x7f4b92bc7e80>
```

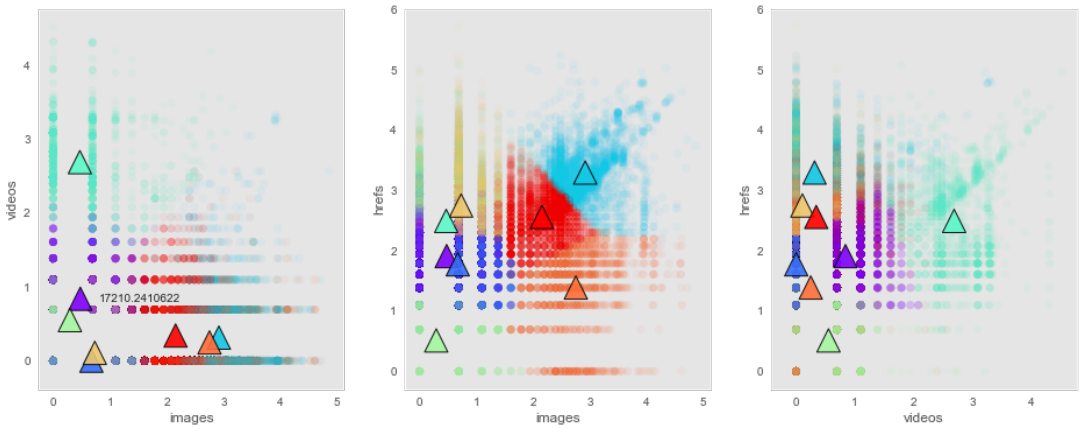
Out[42]:
```
(<matplotlib.text.Text at 0x7f4b773e1d68>,
 <matplotlib.text.Text at 0x7f4b777079e8>)
```

Out[42]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4b804d13c8>
```

Out[42]:
```
<matplotlib.collections.PathCollection at 0x7f4b7770d780>
```

Out[42]:
```
<matplotlib.collections.PathCollection at 0x7f4b80126b38>
```

Out[42]:
```
(<matplotlib.text.Text at 0x7f4b8041d668>,
 <matplotlib.text.Text at 0x7f4b90463ac8>)
```



Out[42]:
```
KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
    n_clusters=9, n_init=10, n_jobs=1, precompute_distances='auto',
    random_state=1, tol=0.0001, verbose=0)

n_lda =  9
inertia =  15960.0254427
silhouette =  0.314761926227
```

Out[42]:
```
<matplotlib.figure.Figure at 0x7f4b92702fd0>
```

Out[42]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4b7775d198>
```

Out[42]:
```
<matplotlib.collections.PathCollection at 0x7f4b9267a518>
```

Out[42]:
```
<matplotlib.collections.PathCollection at 0x7f4b9054a9e8>
```

Out[42]:
```
<matplotlib.text.Text at 0x7f4b776f8c50>
```

Out[42]:
```
(<matplotlib.text.Text at 0x7f4b770e5ba8>,
 <matplotlib.text.Text at 0x7f4b771d89b0>)
```

Out[42]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4b804fcf28>
```

Out[42]:
```
<matplotlib.collections.PathCollection at 0x7f4b92bc2128>
```

Out[42]:
```
<matplotlib.collections.PathCollection at 0x7f4b775a4c18>
```

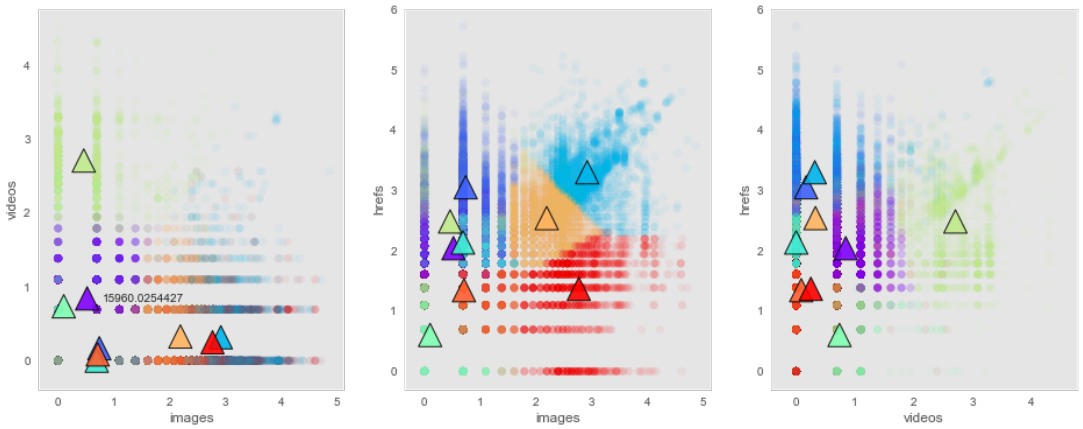
Out[42]:
```
(<matplotlib.text.Text at 0x7f4b773b75f8>,
 <matplotlib.text.Text at 0x7f4b77442518>)
```

Out[42]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4b805b4908>
```

Out[42]:
```
<matplotlib.collections.PathCollection at 0x7f4b775879e8>
```

Out[42]:
```
<matplotlib.collections.PathCollection at 0x7f4b771c4198>
```

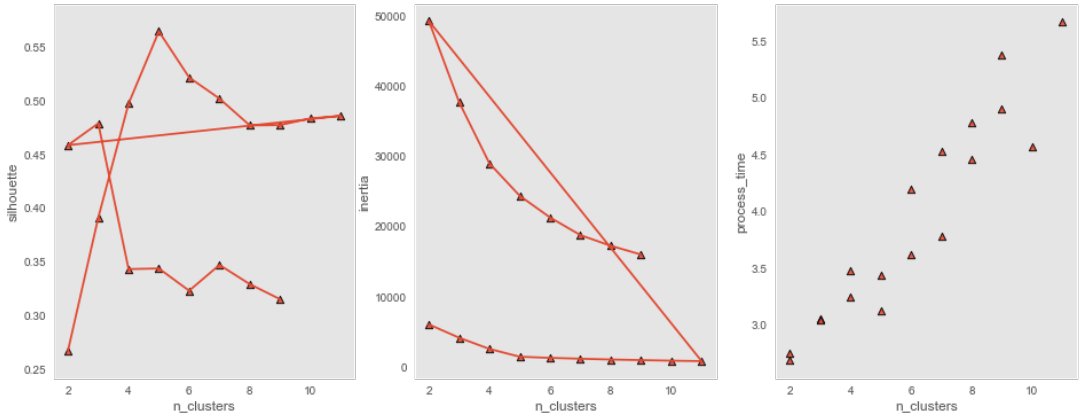
Out[42]:
```
(<matplotlib.text.Text at 0x7f4b775a4940>,
 <matplotlib.text.Text at 0x7f4b90020748>)
```



Out[42]:

	model_name	n_clusters	inertia	silhouette	process_time
1	KMeans - LDA features	2	6016.621217	0.266660	2.6897
1	KMeans - LDA features	3	4120.020638	0.390843	3.0500
1	KMeans - LDA features	4	2523.834033	0.497299	3.2474
1	KMeans - LDA features	5	1418.819577	0.564738	3.1258
1	KMeans - LDA features	6	1257.642737	0.521731	4.1921
1	KMeans - LDA features	7	1135.466961	0.502022	4.5223
1	KMeans - LDA features	8	1023.646217	0.476994	4.4569
1	KMeans - LDA features	9	937.873676	0.477308	4.9037
1	KMeans - LDA features	10	859.164016	0.483460	4.5667
1	KMeans - LDA features	11	794.230421	0.485906	5.6639
1	KMeans - images_videos_hrefs features	2	49187.373948	0.458518	2.7502
1	KMeans - images_videos_hrefs features	3	37625.027983	0.478869	3.0448
1	KMeans - images_videos_hrefs features	4	28917.966869	0.342858	3.4816
1	KMeans - images_videos_hrefs features	5	24309.329311	0.343458	3.4371
1	KMeans - images_videos_hrefs features	6	21231.988255	0.322696	3.6142
1	KMeans - images_videos_hrefs features	7	18723.739263	0.346752	3.7820
1	KMeans - images_videos_hrefs features	8	17210.241062	0.328824	4.7750
1	KMeans - images_videos_hrefs features	9	15960.025443	0.314762	5.3742

In [43]:
```
# ... -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
# ... - plot metrics across models for comparison
# ... -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-

plt.figure(figsize=(16, 6));

# ... silhouette values

plt.subplot(131);
plt.scatter(comparison_tbl['n_clusters'],
            comparison_tbl['silhouette'],
            s = 40,
            linewidths = 1.0,
            marker = '^',
            edgecolors = 'black',
            alpha = 0.90);

plt.plot(comparison_tbl['n_clusters'],
         comparison_tbl['silhouette'])

plt.xlabel('n_clusters'), plt.ylabel('silhouette');
plt.grid();

# ... inertia values

plt.subplot(132);
plt.scatter(comparison_tbl['n_clusters'],
            comparison_tbl['inertia'],
            s = 40,
            linewidths = 1.0,
            marker = '^',
            edgecolors = 'black',
            alpha = 0.90);

plt.plot(comparison_tbl['n_clusters'],
         comparison_tbl['inertia'])

plt.xlabel('n_clusters'), plt.ylabel('inertia');
plt.grid();

# ... process time

plt.subplot(133);
plt.scatter(comparison_tbl['n_clusters'],
            comparison_tbl['process_time'],
            s = 40,
            linewidths = 1.0,
            marker = '^',
            edgecolors = 'black',
            alpha = 0.90);

#plt.plot(comparison_tbl['n_clusters'],
#         comparison_tbl['process_time'])

plt.xlabel('n_clusters'), plt.ylabel('process_time');
```



K-Means - All in

In [44]:
```
X1 = df_cluster

for n_lda in range(2, 12):

    tic = time.clock()

    cls_lda = KMeans(n_clusters = n_lda,
                     init = 'k-means++',
                     random_state = 1);
    cls_lda.fit(X1);

    kmeans_labels = cls_lda.labels_ # the labels from kmeans clustering
    kmeans_centers = cls_lda.cluster_centers_
    kmeans_inertia = cls_lda.inertia_

    print ("n_lda, inertia ", n_lda, kmeans_inertia)

    kmeans_silhouette = metrics.silhouette_score(X1,
                                                 kmeans_labels,
                                                 metric = 'euclidean',
                                                 sample_size = 10000)
    print ("silhouette = ", kmeans_silhouette)

    toc =  time.clock()
# ... -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
# ... - save statistics for model comparison
# ... -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-

    exe_time = '{0:.4f}'.format(toc-tic)

    raw_data = {
    'model_name' : 'KMeans - all_in',
    'n_clusters' : n_lda,
    'inertia': kmeans_inertia,
    'silhouette': kmeans_silhouette,
    'process_time' : exe_time
    }

    df_tbl = pd.DataFrame(raw_data,
    columns = ['model_name', 'n_clusters', 'inertia', 'silhouette', 'process_time'],
    index = [i_index + 1])

    comparison_tbl = comparison_tbl.append(df_tbl)

# ... -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
# ... - make some plots of clusters
# ... -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
```

```
Out[44]: KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
             n_clusters=2, n_init=10, n_jobs=1, precompute_distances='auto',
             random_state=1, tol=0.0001, verbose=0)

         n_lda, inertia  2 1400523.17552
         silhouette =  0.342422985149

Out[44]: KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
             n_clusters=3, n_init=10, n_jobs=1, precompute_distances='auto',
             random_state=1, tol=0.0001, verbose=0)

         n_lda, inertia  3 1144630.43362
         silhouette =  0.351734514309

Out[44]: KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
             n_clusters=4, n_init=10, n_jobs=1, precompute_distances='auto',
             random_state=1, tol=0.0001, verbose=0)

         n_lda, inertia  4 999391.40361
         silhouette =  0.358355487029

Out[44]: KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
             n_clusters=5, n_init=10, n_jobs=1, precompute_distances='auto',
             random_state=1, tol=0.0001, verbose=0)

         n_lda, inertia  5 920707.821029
         silhouette =  0.263202464355

Out[44]: KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
             n_clusters=6, n_init=10, n_jobs=1, precompute_distances='auto',
             random_state=1, tol=0.0001, verbose=0)

         n_lda, inertia  6 865749.225149
         silhouette =  0.1922855623

Out[44]: KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
             n_clusters=7, n_init=10, n_jobs=1, precompute_distances='auto',
             random_state=1, tol=0.0001, verbose=0)

         n_lda, inertia  7 826467.6524
         silhouette =  0.17609069104

Out[44]: KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
             n_clusters=8, n_init=10, n_jobs=1, precompute_distances='auto',
             random_state=1, tol=0.0001, verbose=0)

         n_lda, inertia  8 790950.316967
         silhouette =  0.18491368321

Out[44]: KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
             n_clusters=9, n_init=10, n_jobs=1, precompute_distances='auto',
             random_state=1, tol=0.0001, verbose=0)

         n_lda, inertia  9 760728.643599
         silhouette =  0.187606204853

Out[44]: KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
             n_clusters=10, n_init=10, n_jobs=1, precompute_distances='auto',
             random_state=1, tol=0.0001, verbose=0)
```

```
n_lda, inertia  10 737976.545004
silhouette =  0.185523611394
```

Out[44]:
```
KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
    n_clusters=11, n_init=10, n_jobs=1, precompute_distances='auto',
    random_state=1, tol=0.0001, verbose=0)
```

```
n_lda, inertia  11 707754.988617
silhouette =  0.186351549287
```

In [45]:
```
# ... -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
# ... - plot metrics across models for comparison
# ... -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-

plt.figure(figsize=(16, 6));

# ... silhouette values

plt.subplot(131);
plt.scatter(comparison_tbl['n_clusters'],
            comparison_tbl['silhouette'],
            s = 40,
            linewidths = 1.0,
            marker = '^',
            edgecolors = 'black',
            alpha = 0.90);

plt.plot(comparison_tbl['n_clusters'],
         comparison_tbl['silhouette'])

plt.xlabel('n_clusters'), plt.ylabel('silhouette');
plt.grid();

# ... inertia values

plt.subplot(132);
plt.scatter(comparison_tbl['n_clusters'],
            comparison_tbl['inertia'],
            s = 40,
            linewidths = 1.0,
            marker = '^',
            edgecolors = 'black',
            alpha = 0.90);

plt.plot(comparison_tbl['n_clusters'],
         comparison_tbl['inertia'])

plt.xlabel('n_clusters'), plt.ylabel('inertia');
plt.grid();

# ... process time

plt.subplot(133);
plt.scatter(comparison_tbl['n_clusters'],
            comparison_tbl['process_time'],
            s = 40,
            linewidths = 1.0,
            marker = '^',
            edgecolors = 'black',
            alpha = 0.90);

#plt.plot(comparison_tbl['n_clusters'],
#         comparison_tbl['process_time'])

plt.xlabel('n_clusters'), plt.ylabel('process_time');
```

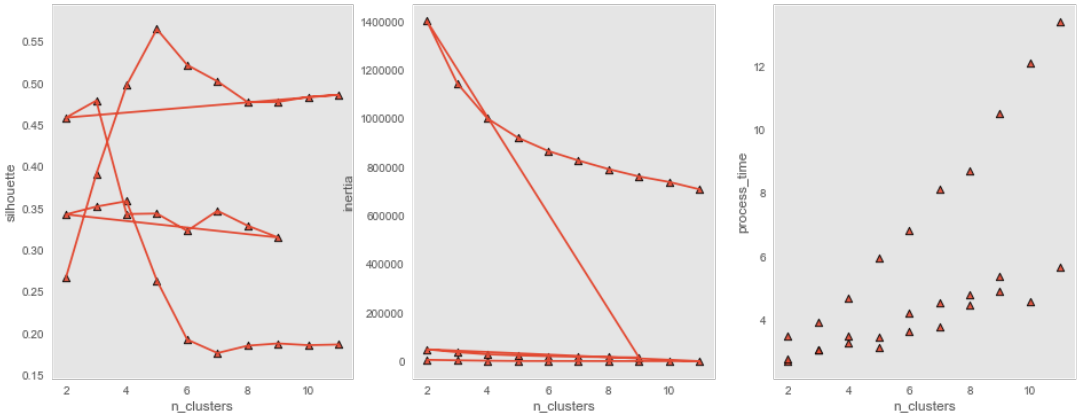


Table of Contents

DBScan

DBSCAN

http://hdbscan.readthedocs.io/en/latest/comparing_clustering_algorithms.html (http://hdbscan.readthedocs.io/en/latest/comparing_clustering_algorithms.html)

DBSCAN is a density based algorithm – it assumes clusters for dense regions. It is also the first actual clustering algorithm we've looked at: it doesn't require that every point be assigned to a cluster and hence doesn't partition the data, but instead extracts the 'dense' clusters and leaves sparse background classified as 'noise'.
In practice DBSCAN is related to agglomerative clustering.
As a first step DBSCAN transforms the space according to the density of the data: points in dense regions are left alone, while points in sparse regions are moved further away. Applying single linkage clustering to the transformed space results in a dendrogram, which we cut according to a distance parameter (called epsilon or eps in many implementations) to get clusters. Importantly any singleton clusters at that cut level are deemed to be 'noise' and left unclustered. This provides several advantages: we get the manifold following behaviour of agglomerative clustering, and we get actual clustering as opposed to partitioning. Better yet, since we can frame the algorithm in terms of local region queries we can use various tricks such as kdtrees to get exceptionally good performance and scale to dataset sizes that are otherwise unapproachable with algorithms other than K-Means.
There are some catches however. Obviously epsilon can be hard to pick; you can do some data analysis and get a good guess, but the algorithm can be quite sensitive to the choice of the parameter. The density based transformation depends on another parameter (min_samples in sklearn).
Finally the combination of min_samples and eps amounts to a choice of density and the clustering only finds clusters at or above that density; if your data has variable density clusters then DBSCAN is either going to miss them, split them up, or lump some of them together depending on your parameter choices.

So, in summary:

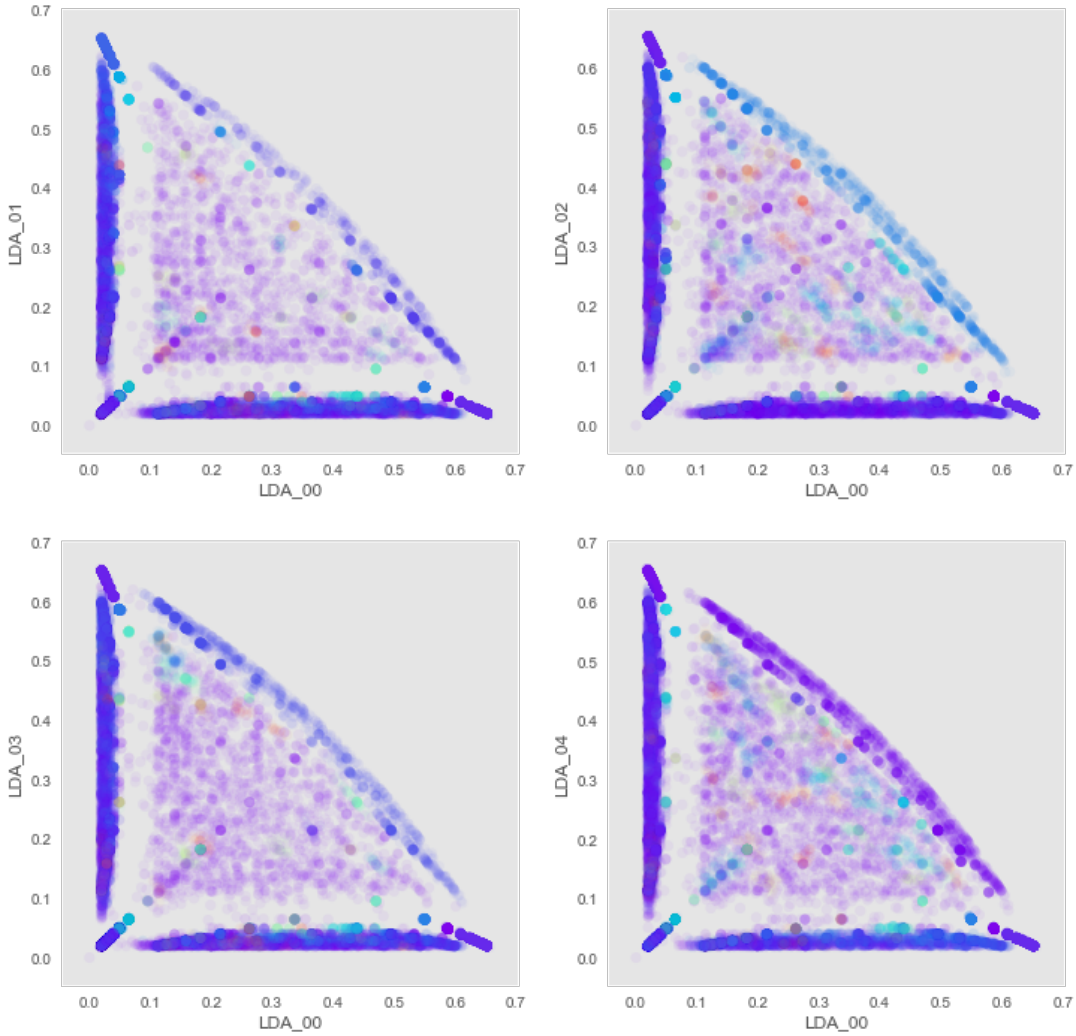
- **Don't be wrong!**: Clusters don't need to be globular, and won't have noise lumped in; varying density clusters may cause problems, but that is more in the form of insufficient detail rather than explicitly wrong. DBSCAN is the first clustering algorithm we've looked at that actually meets the 'Don't be wrong!' requirement.
  - **Intuitive parameters**: Epsilon is a distance value, so you can survey the distribution of distances in your dataset to attempt to get an idea of where it should lie. In practice, however, this isn't an especially intuitive parameter, nor is it easy to get right.
  - **Stability**: DBSCAN is stable across runs (and to some extent subsampling if you re-parameterize well); stability over varying epsilon and min samples is not so good.
  - **Performance**: This is DBSCAN's other great strength; few clustering algorithms can tackle datasets as large as DBSCAN can.

So how does it cluster our test dataset? I played with a few epsilon values until I got somethign reasonable, but there was little science to this – getting the parameters right can be hard.

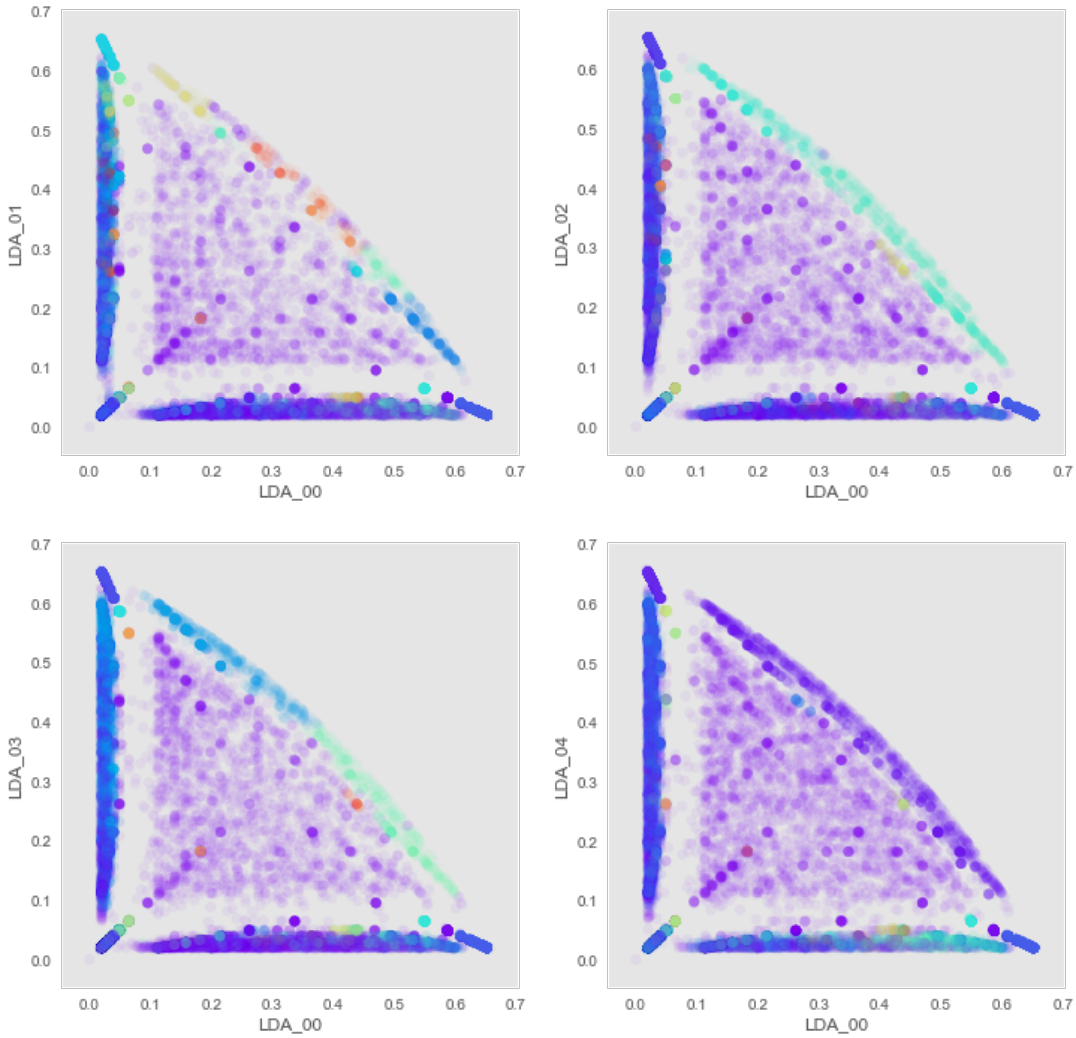
In [61]:
```
# set required variables for model comparison

dbscan_tbl = pd.DataFrame(columns = [
    'model_name',
    'n_clusters',
    'epsilon',
    'min_points',
    'inertia',
    'silhouette',
    'process_time'])

i_index = []
i_index = 0

# preparation for cross validation and model comparison, each classifier is appended once mode
l is fit

models = []
```

In [62]:
```
%%time

from sklearn.cluster import DBSCAN

params = []
for epsilon in [0.020, 0.03, 0.05, 0.06, 0.07]:
    for min_pts in range (10, 200, 20):

        tic = time.clock()

        X1 = df_cluster[['ln_LDA_00','ln_LDA_01', 'ln_LDA_02', 'ln_LDA_03', 'ln_LDA_04']]

        # append on the clustering

        cls_fam = DBSCAN(eps = epsilon,
                         min_samples = min_pts,
                         n_jobs = -1)
        cls_fam.fit(X1)

        dbscan_labels = cls_fam.labels_ # the labels from kmeans clustering

        dbscan_nclusters = len(set(dbscan_labels))

        print ("eps, min_pts, nclusters = ", epsilon, min_pts, dbscan_nclusters)

        dbscan_silhouette = metrics.silhouette_score(X1,
                                                     dbscan_labels,
                                                     metric = 'euclidean',
                                                     sample_size = 10000)
        print ("silhouette = ", dbscan_silhouette)

        toc =  time.clock()

# ... -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
# ... - save statistics for model comparison
# ... -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-

        exe_time = '{0:.4f}'.format(toc-tic)

        raw_data = {
        'model_name' : 'DBScan - LDA features',
        'n_clusters' : dbscan_nclusters,
        'epsilon' : epsilon,
        'min_points' : min_pts,
        'inertia': 0,
        'silhouette': dbscan_silhouette,
        'process_time' : exe_time
        }

        df_tbl = pd.DataFrame(raw_data,
        columns = ['model_name', 'n_clusters', 'epsilon', 'min_points', 'inertia', 'silhouette
', 'process_time'],
        index = [i_index + 1])

        dbscan_tbl = dbscan_tbl.append(df_tbl)
```

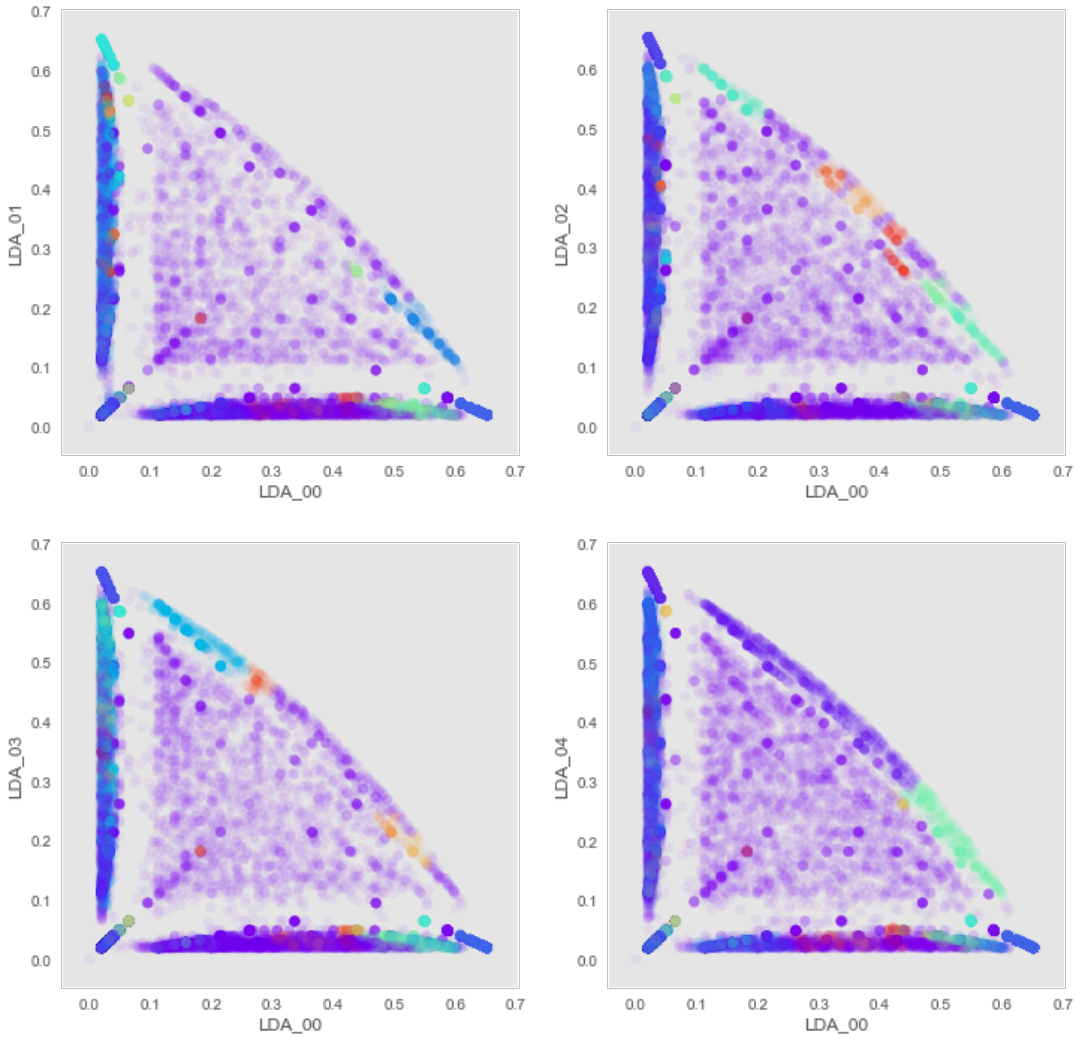
```
eps, min_pts, nclusters =  0.02 10 153
silhouette =  -0.0855960794612
```



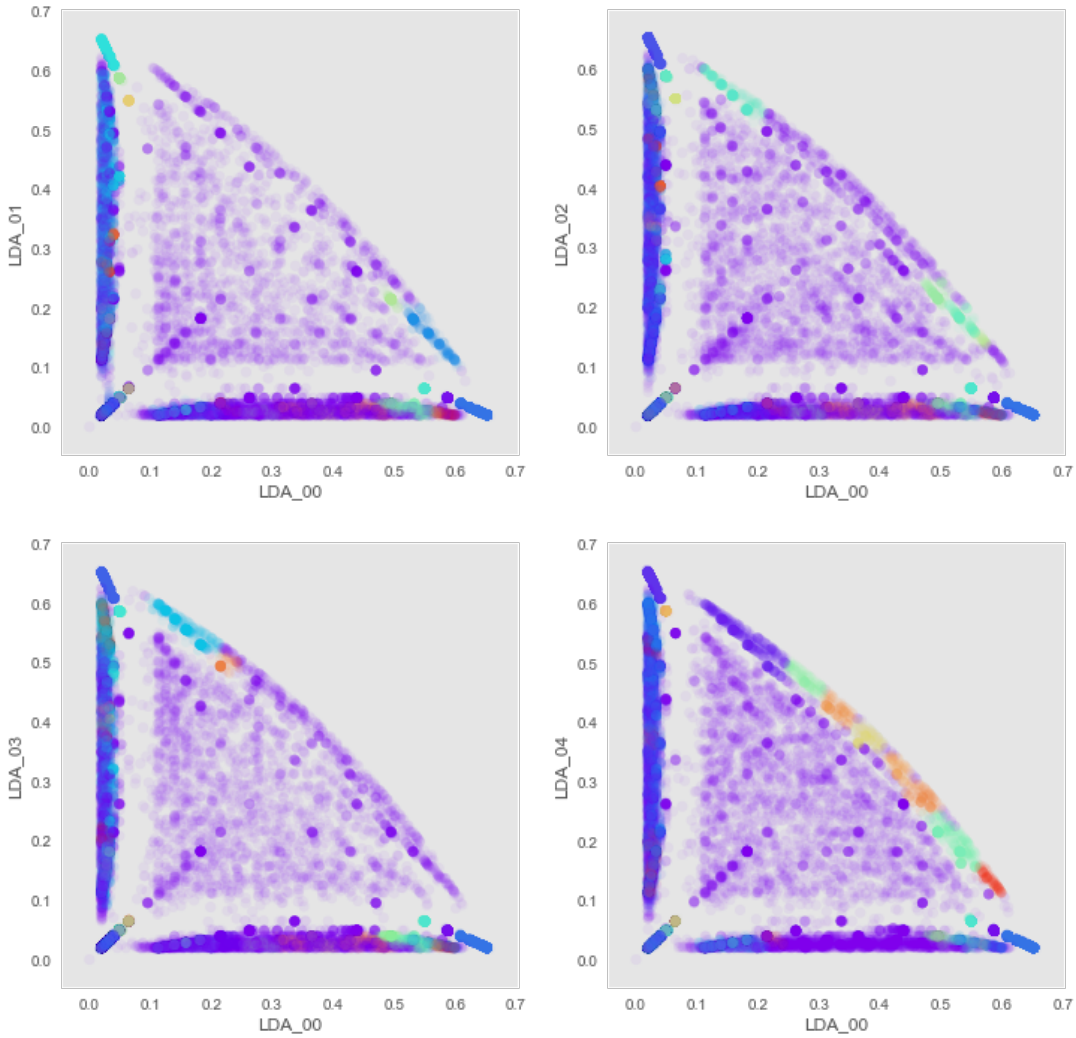

```
eps, min_pts, nclusters =  0.02 30 60
silhouette =  0.031905900258
```



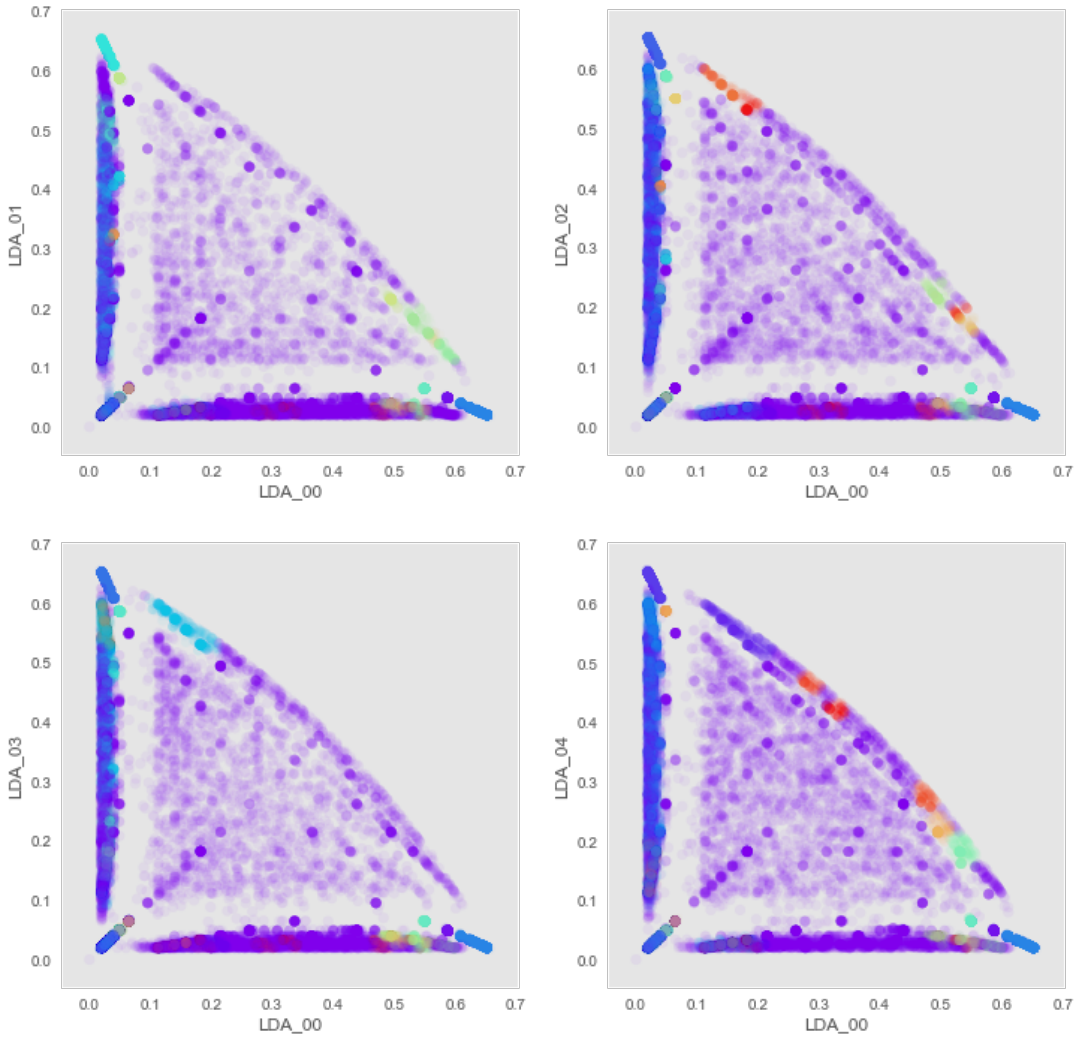

```
eps, min_pts, nclusters =  0.02 50 56
silhouette =  0.101426571856
```



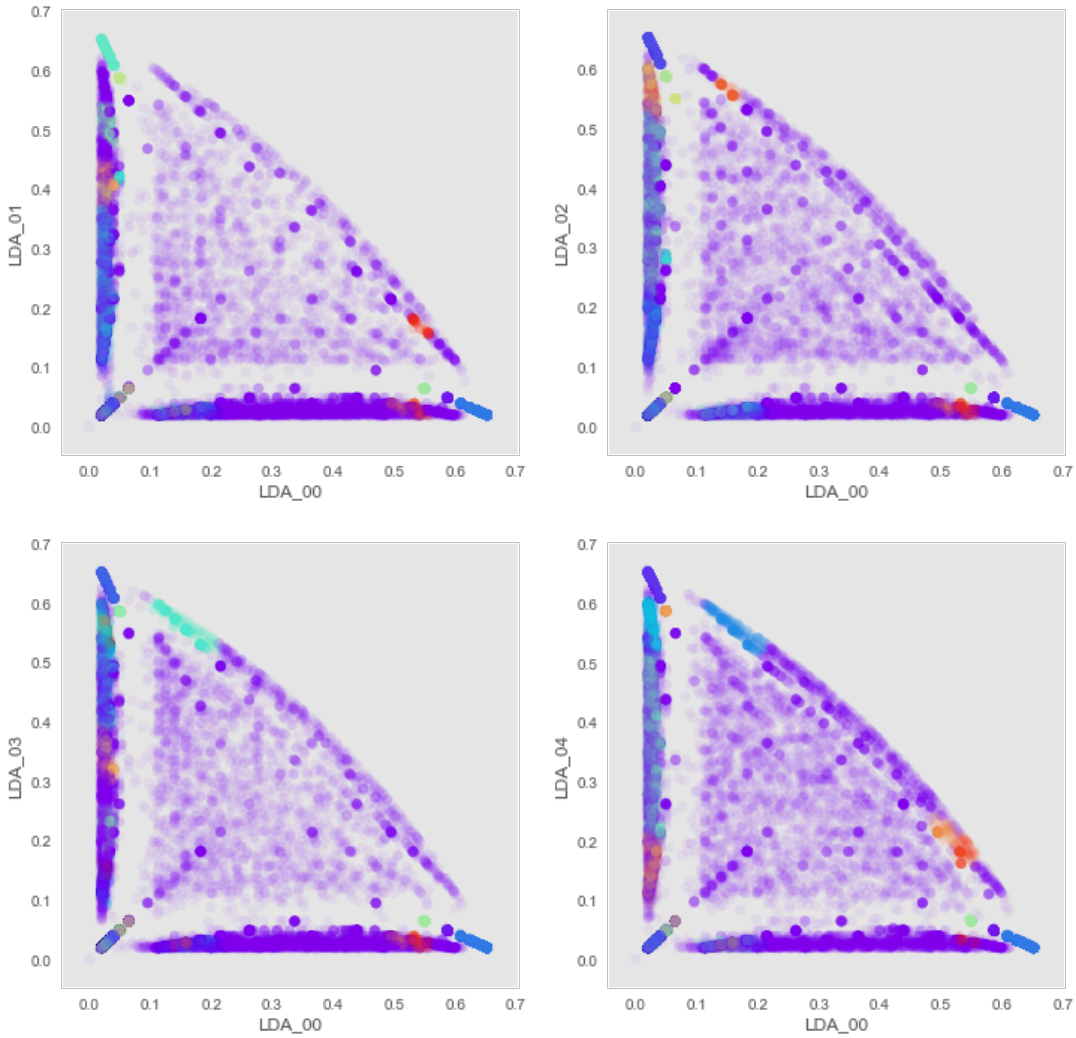

```
eps, min_pts, nclusters =  0.02 70 48
silhouette =  0.0771194097828
```



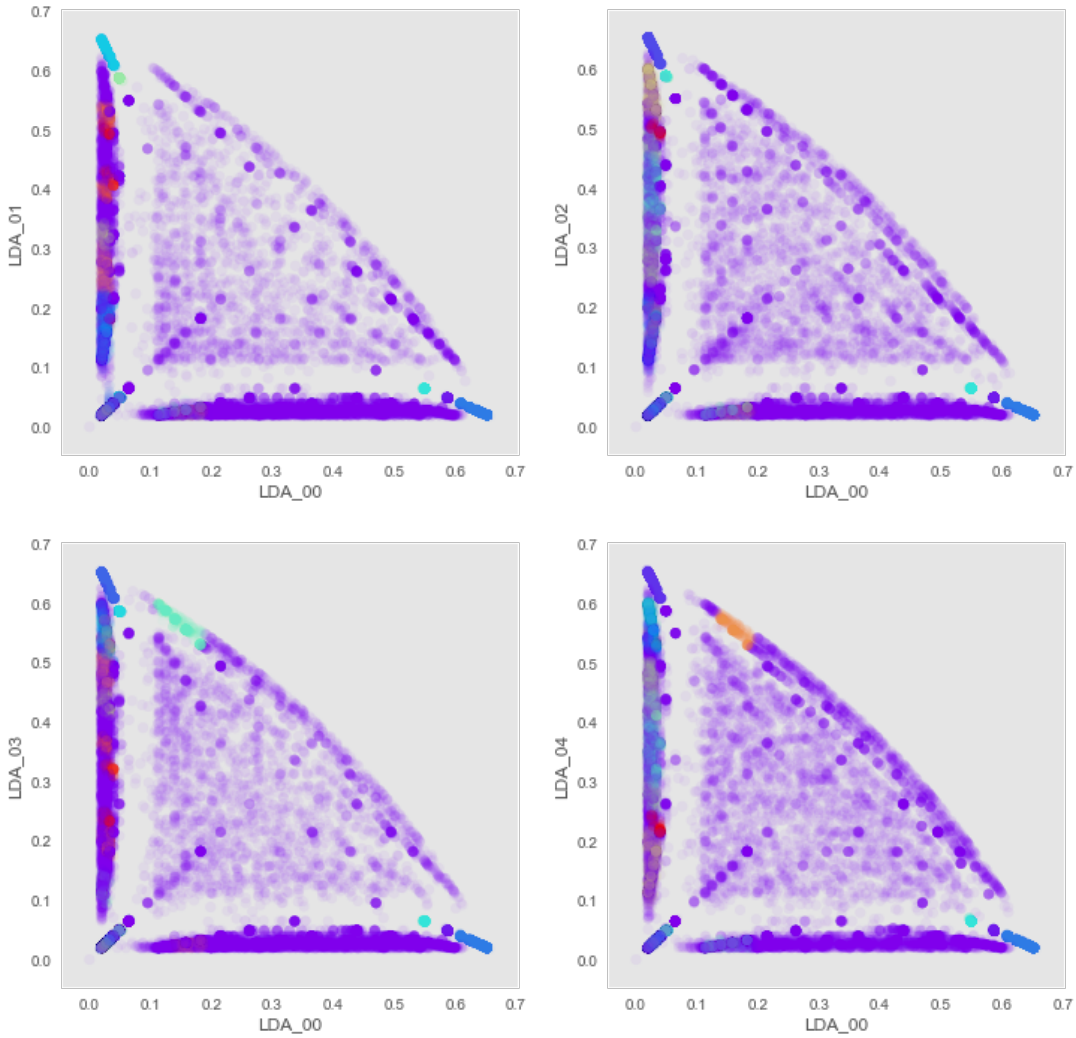

```
eps, min_pts, nclusters =  0.02 90 41
silhouette =  0.036829602996
```



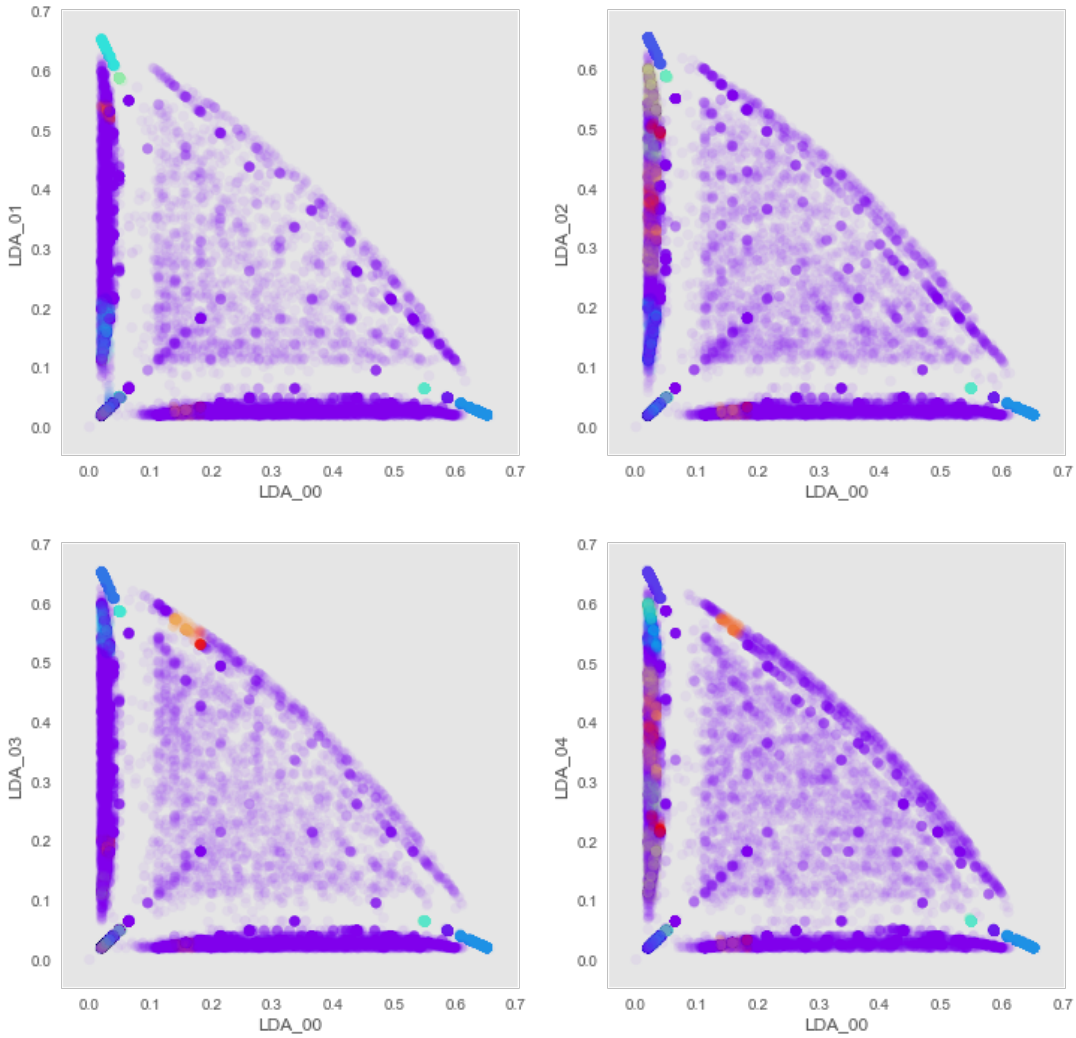

```
eps, min_pts, nclusters =  0.02 110 33
silhouette =  0.0431833243125
```



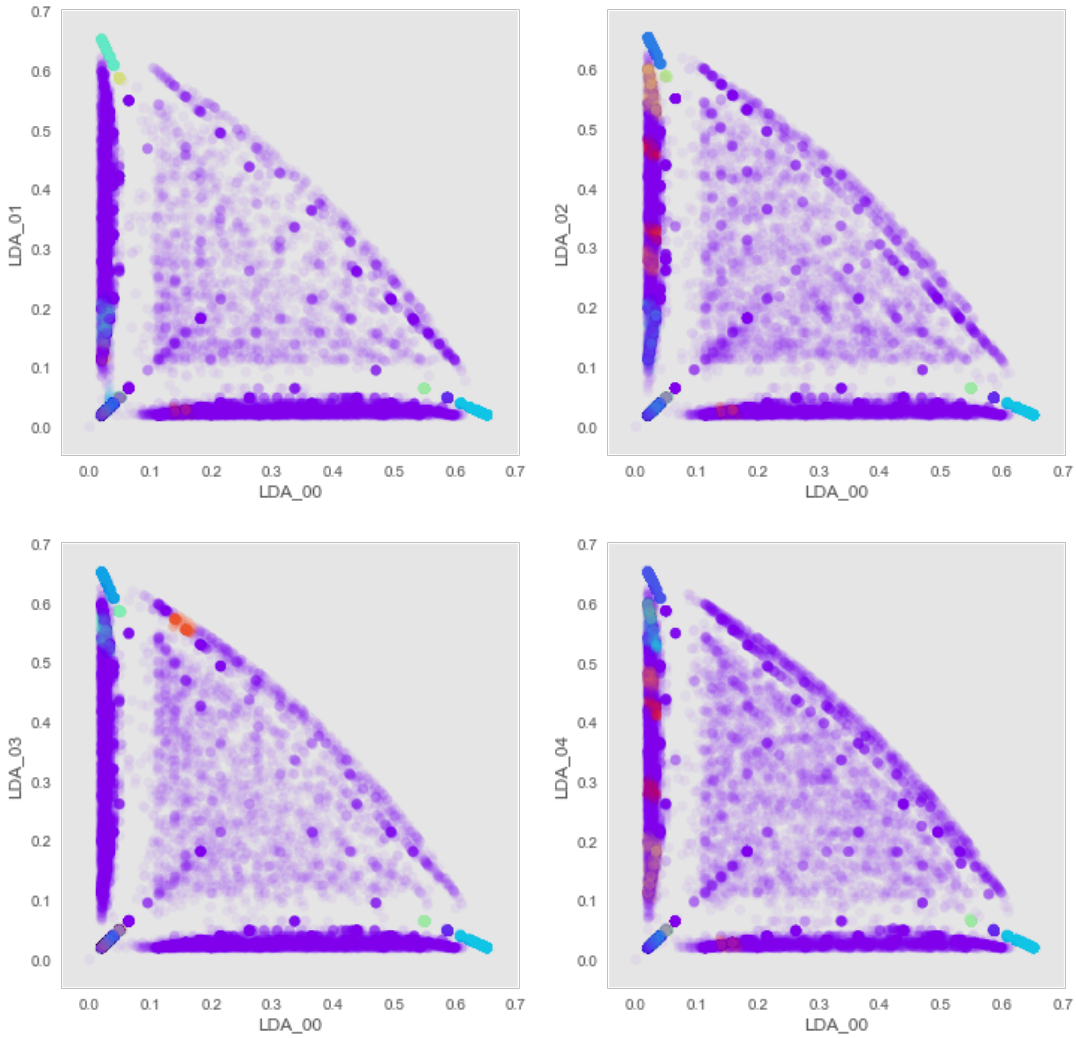

```
eps, min_pts, nclusters =  0.02 130 32
silhouette =  0.0250743431644
```



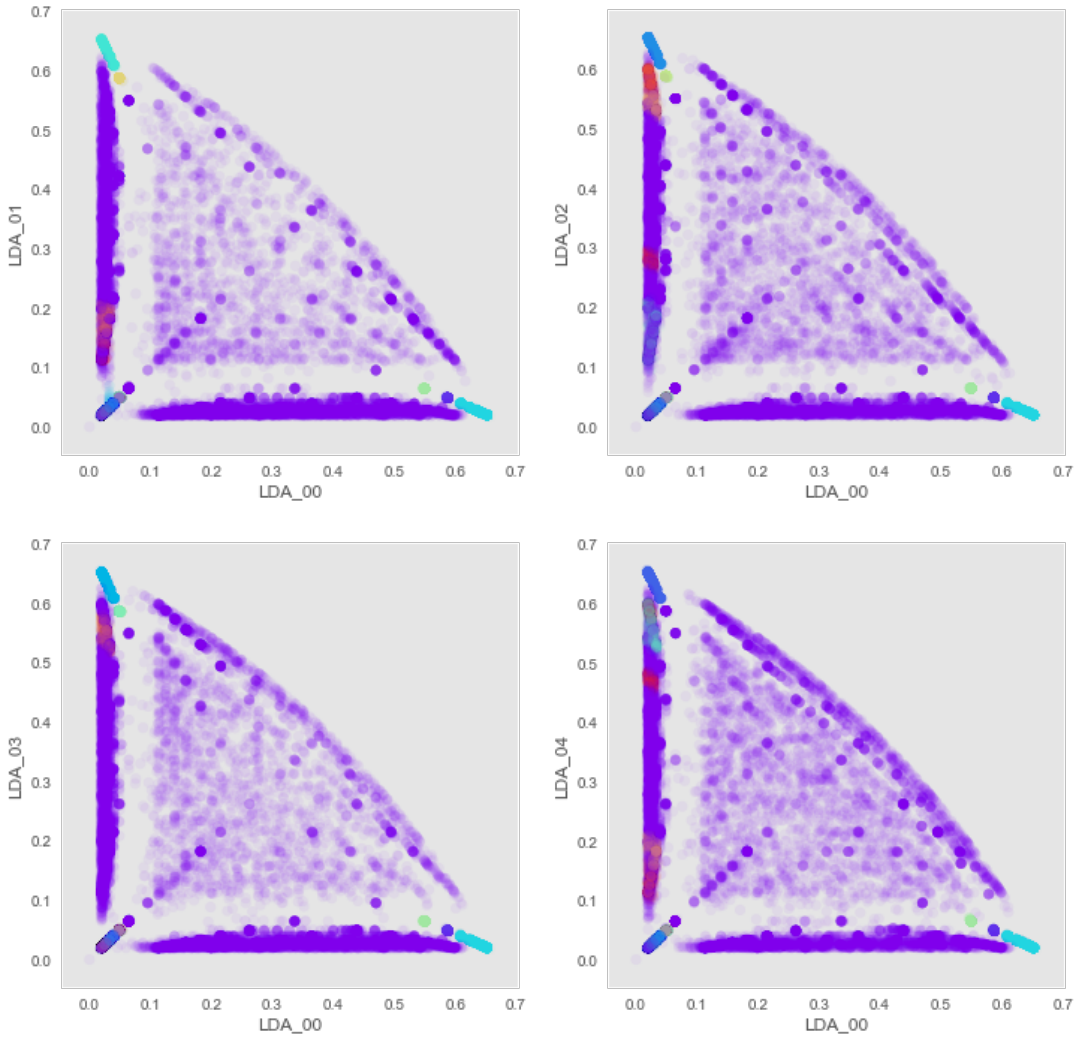

```
eps, min_pts, nclusters =  0.02 150 27
silhouette =  -0.012064911841
```



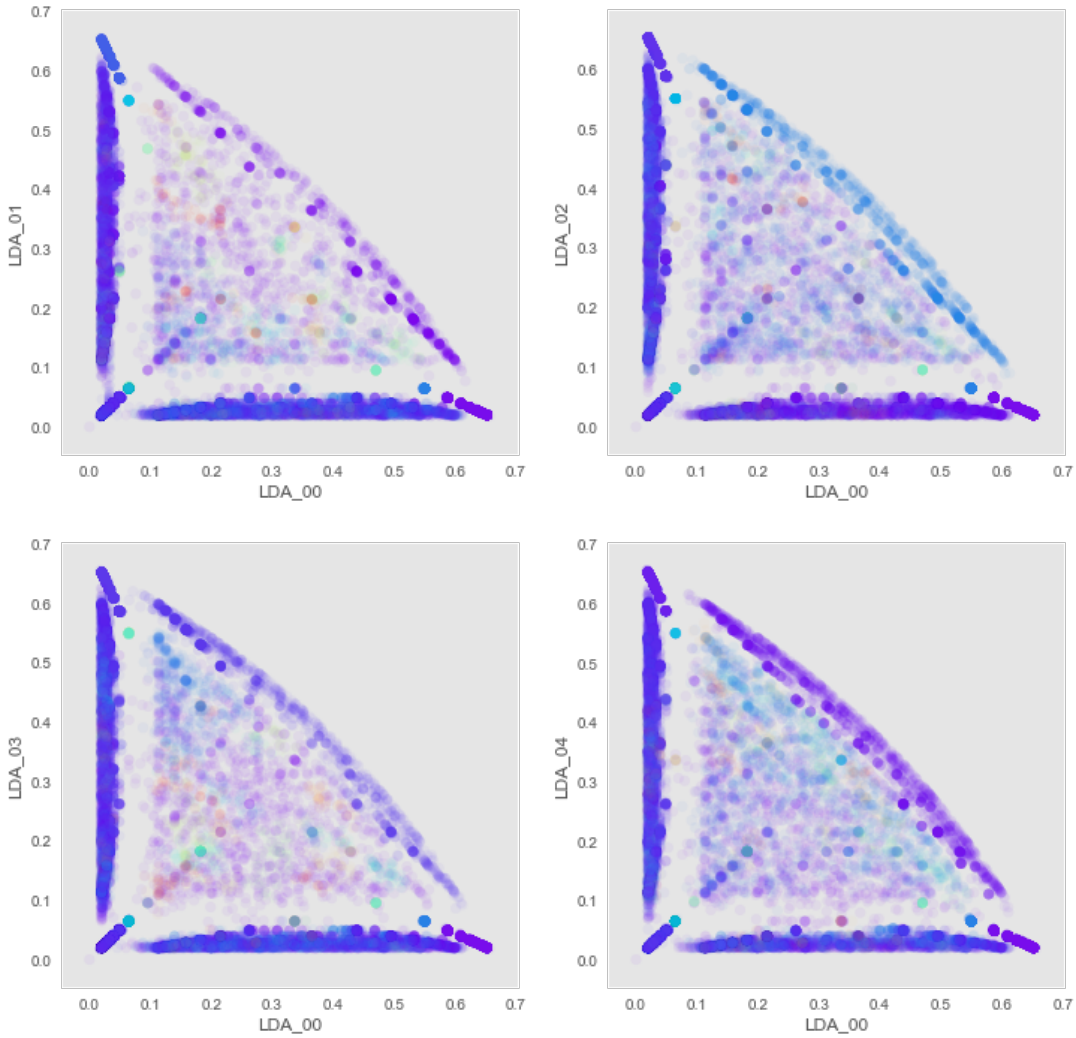

```
eps, min_pts, nclusters =  0.02 170 19
silhouette =  -0.0598985059903
```



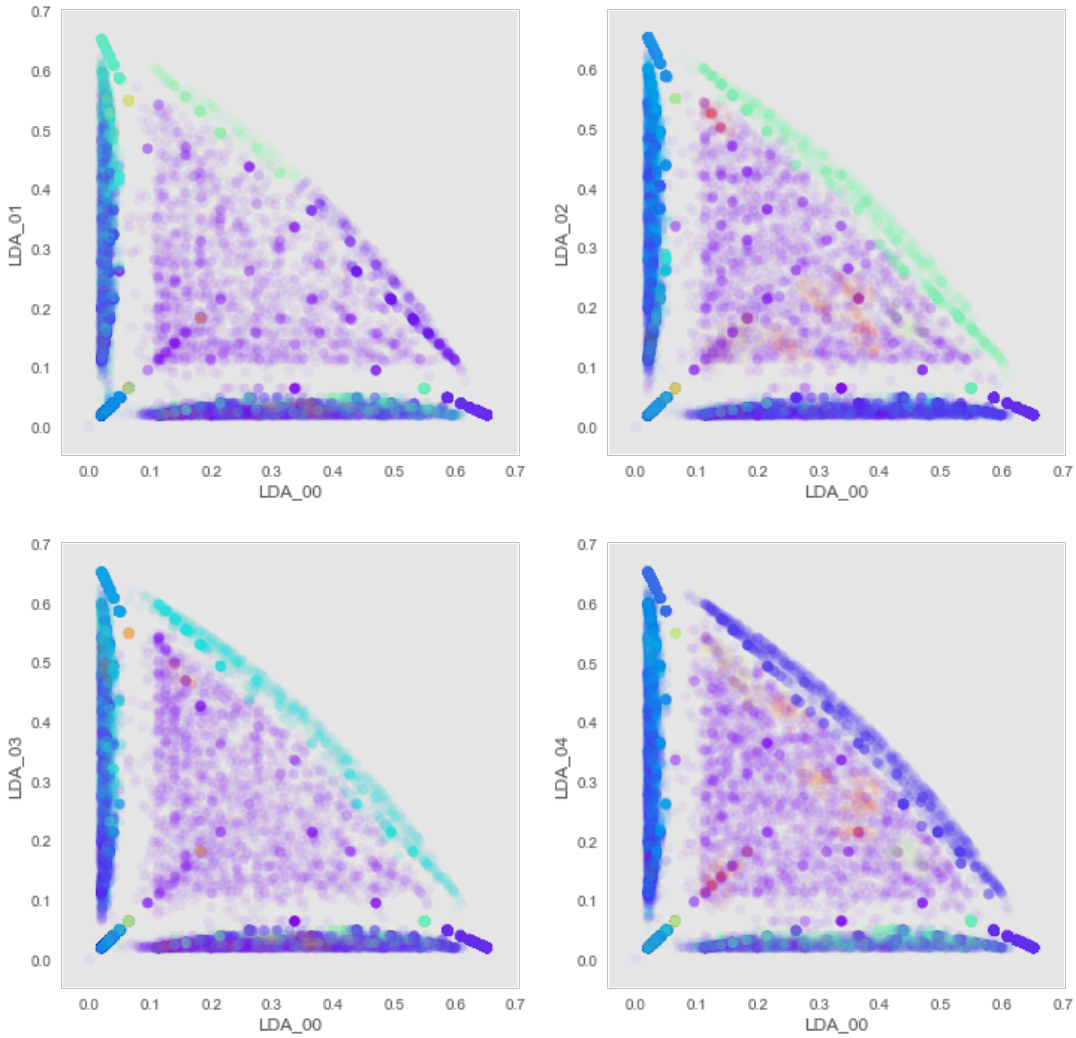

```
eps, min_pts, nclusters =  0.02 190 17
silhouette =  -0.0711635056725
```



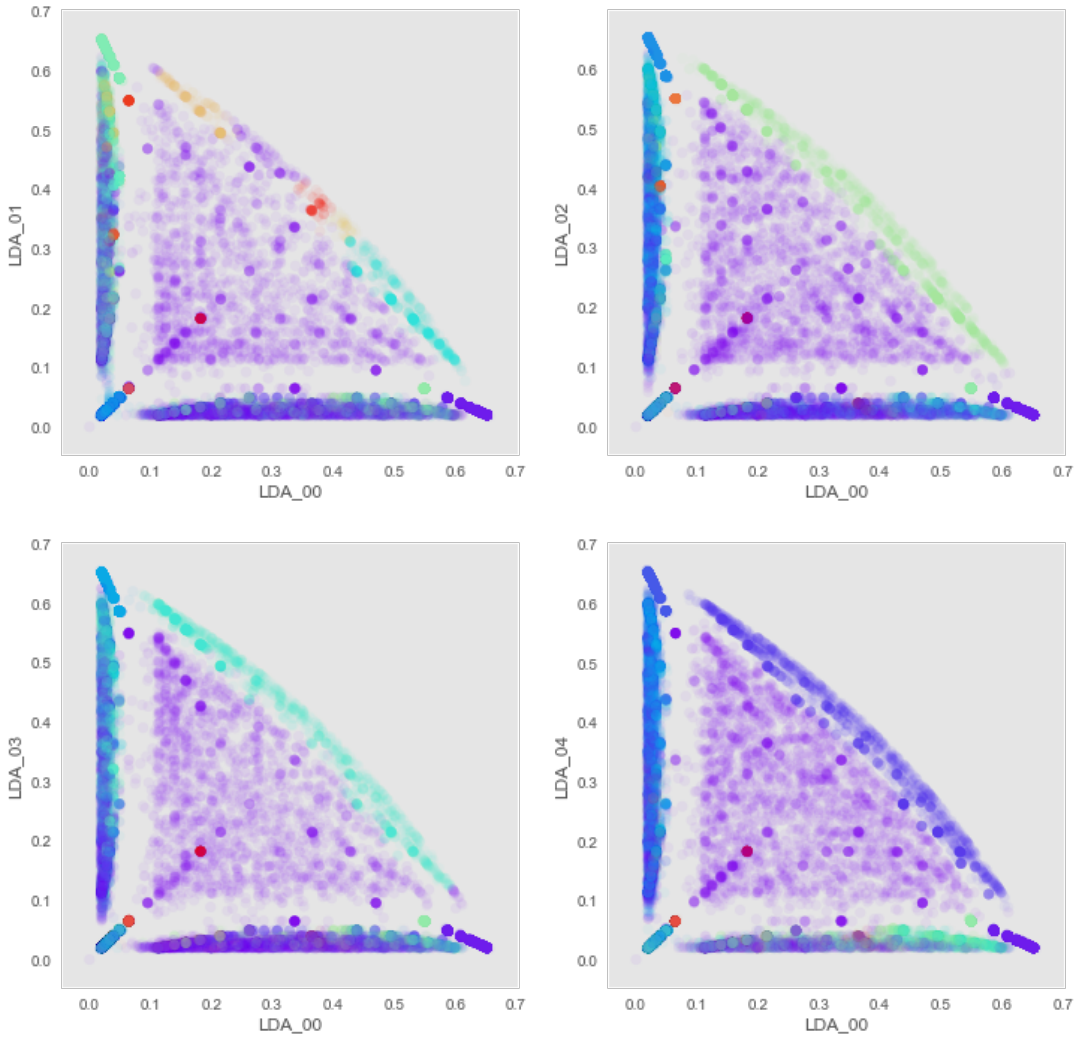

```
eps, min_pts, nclusters =  0.03 10 124
silhouette =  -0.210357563554
```



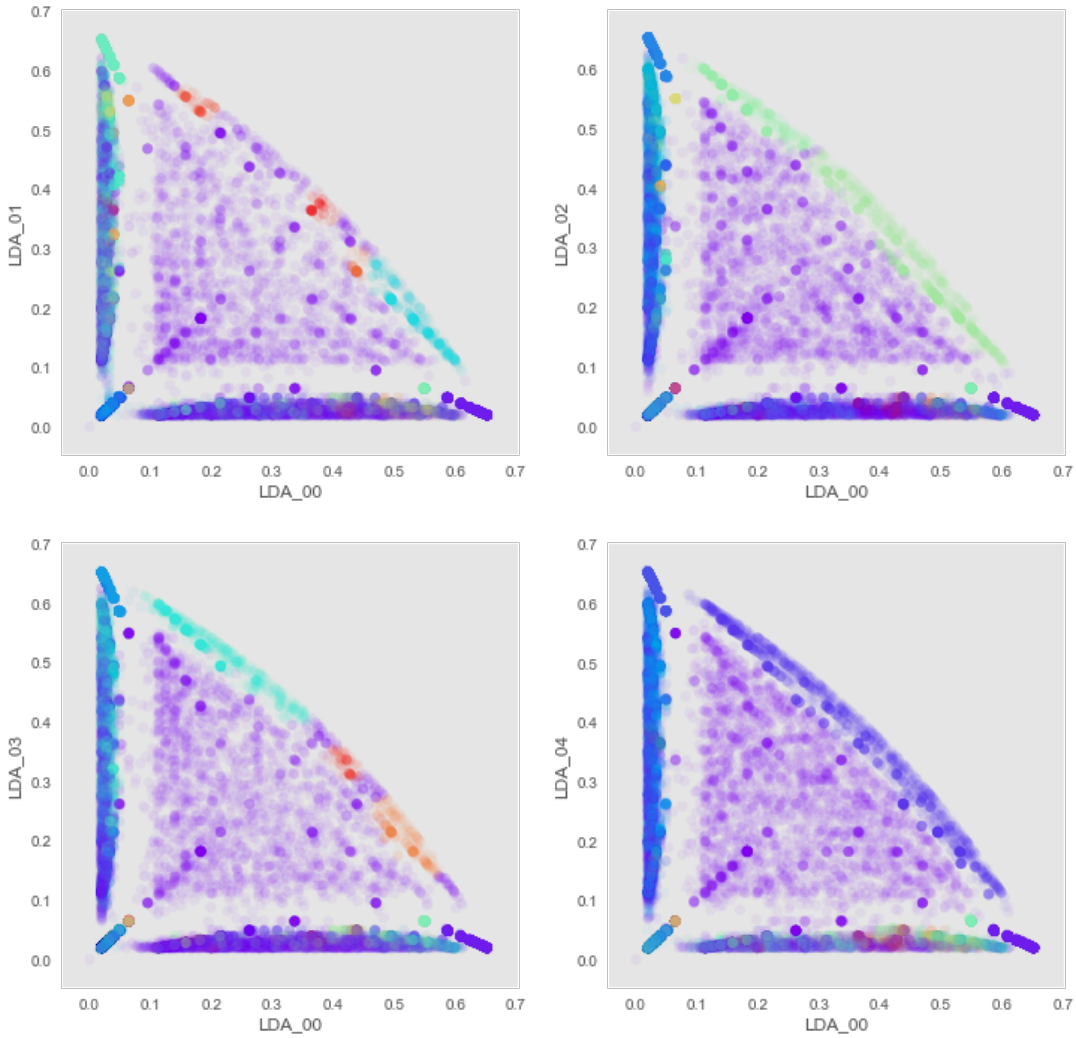

```
eps, min_pts, nclusters =  0.03 30 37
silhouette =  -0.0213734049662
```



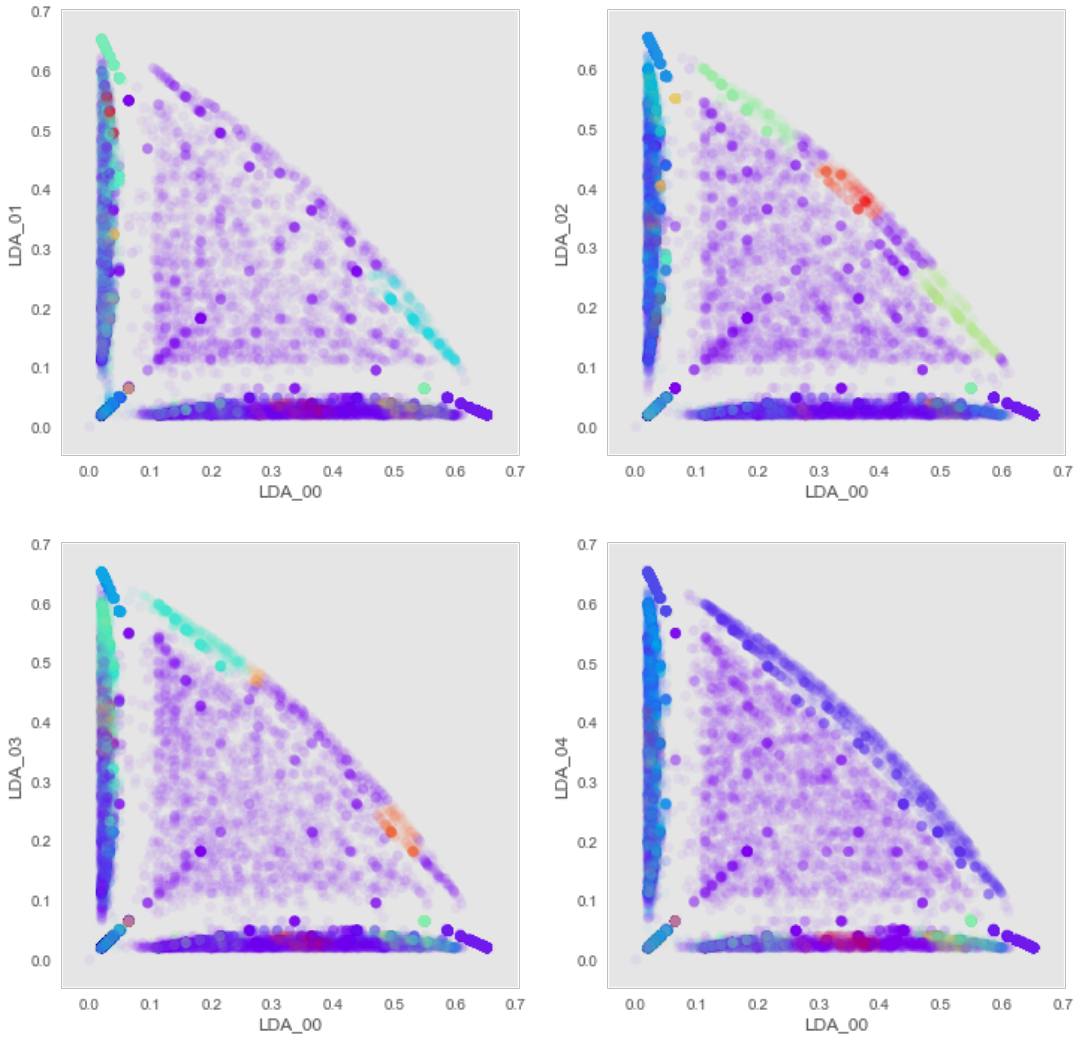

```
eps, min_pts, nclusters =  0.03 50 27
silhouette =  0.145458718953
```



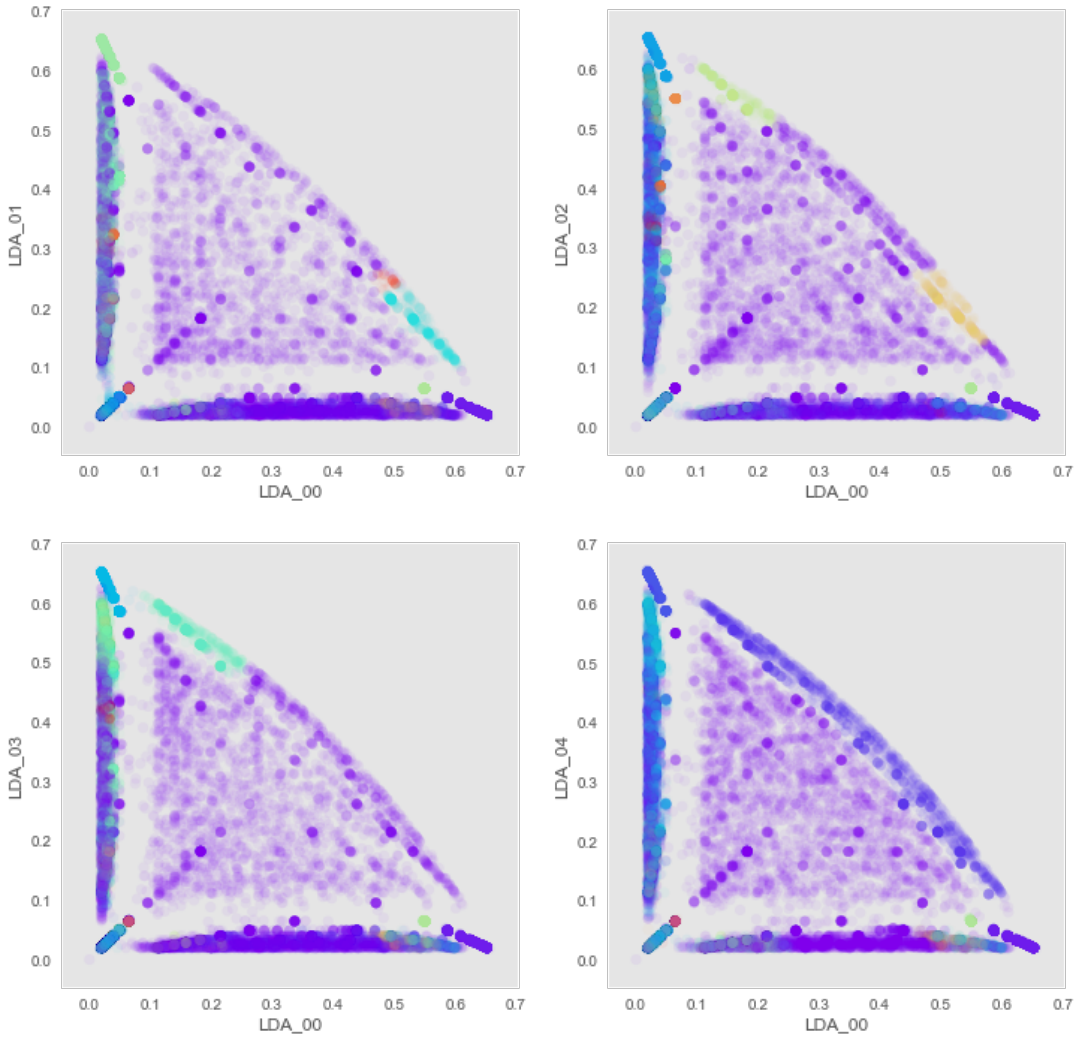

```
eps, min_pts, nclusters =  0.03 70 29
silhouette =  0.138460818714
```



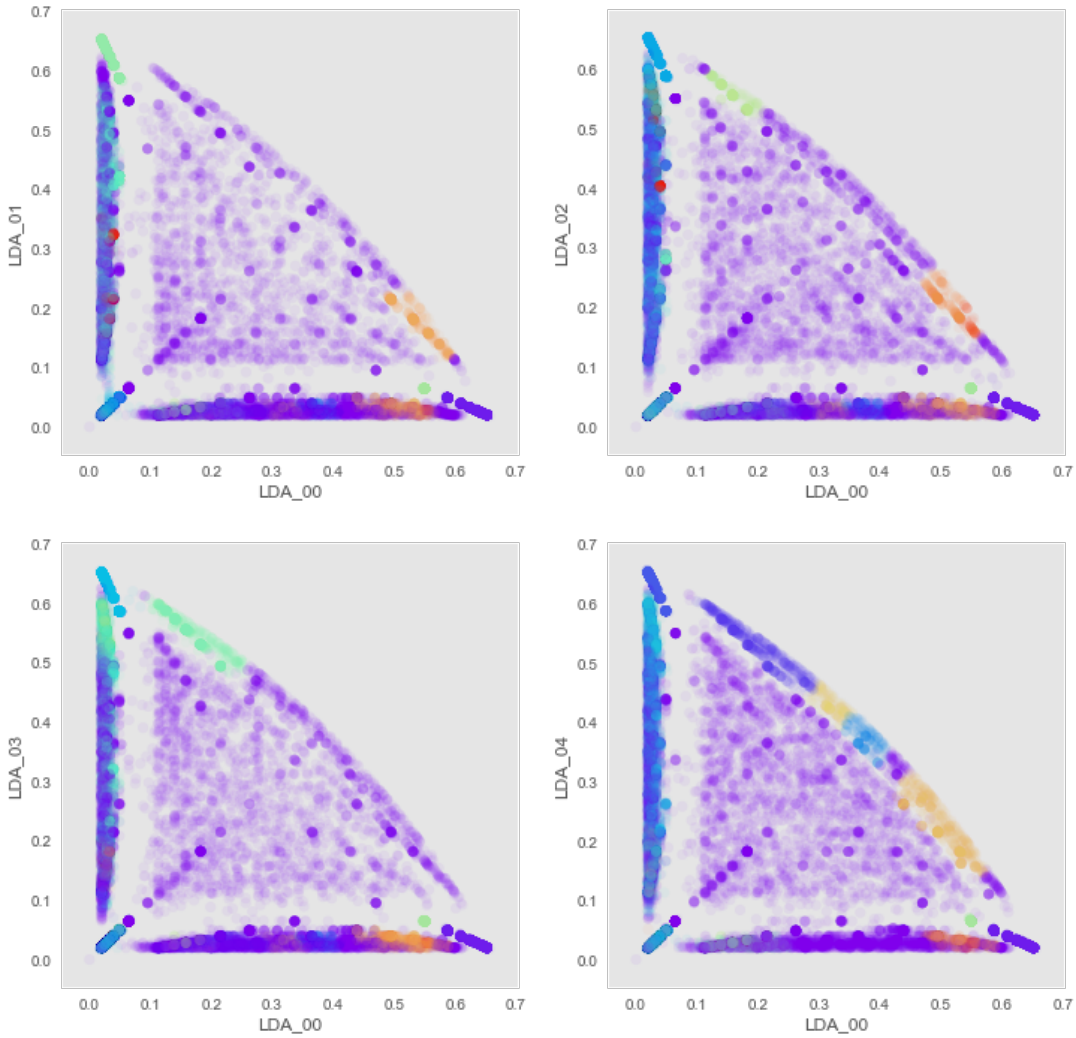

```
eps, min_pts, nclusters =  0.03 90 32
silhouette =  0.162128722293
```



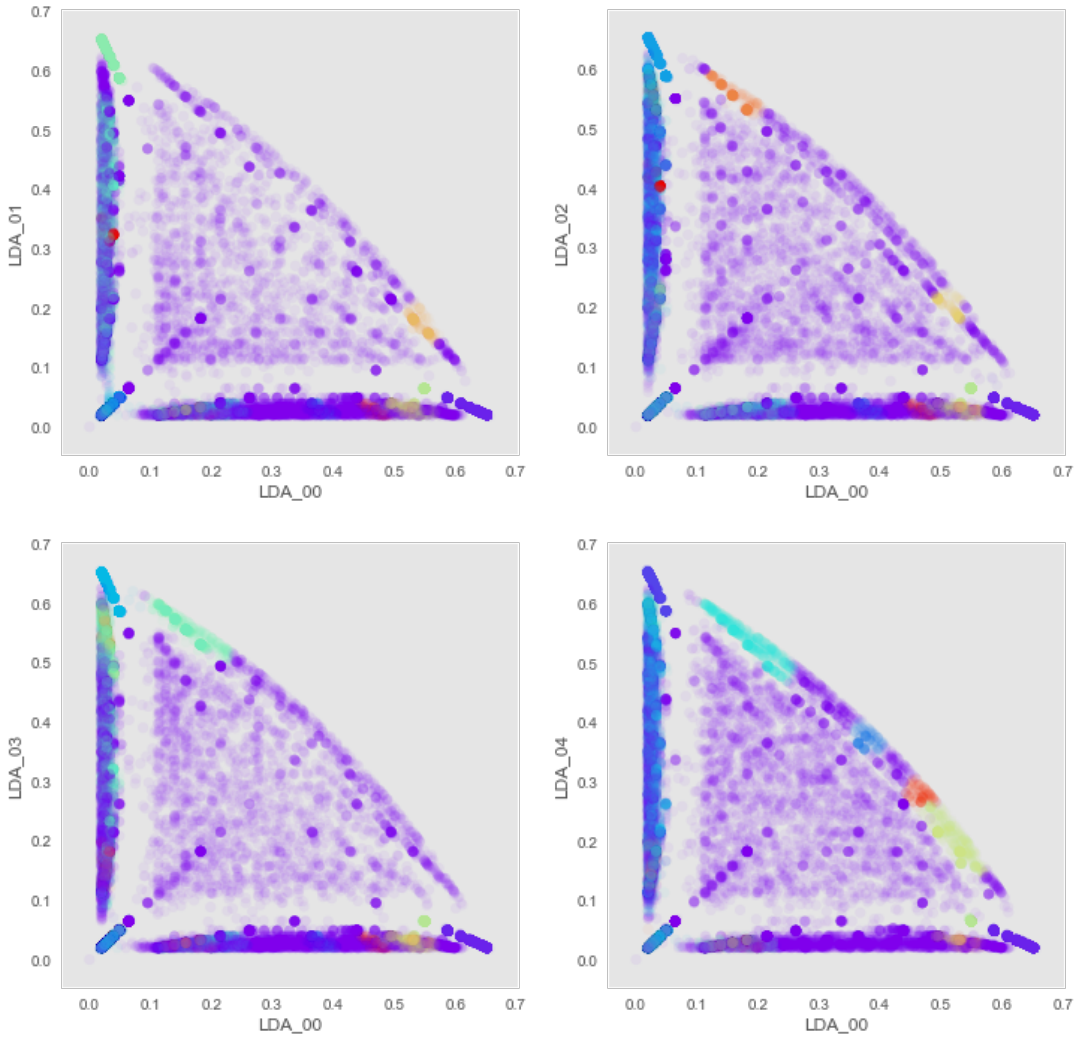

```
eps, min_pts, nclusters =  0.03 110 28
silhouette =  0.146650573747
```



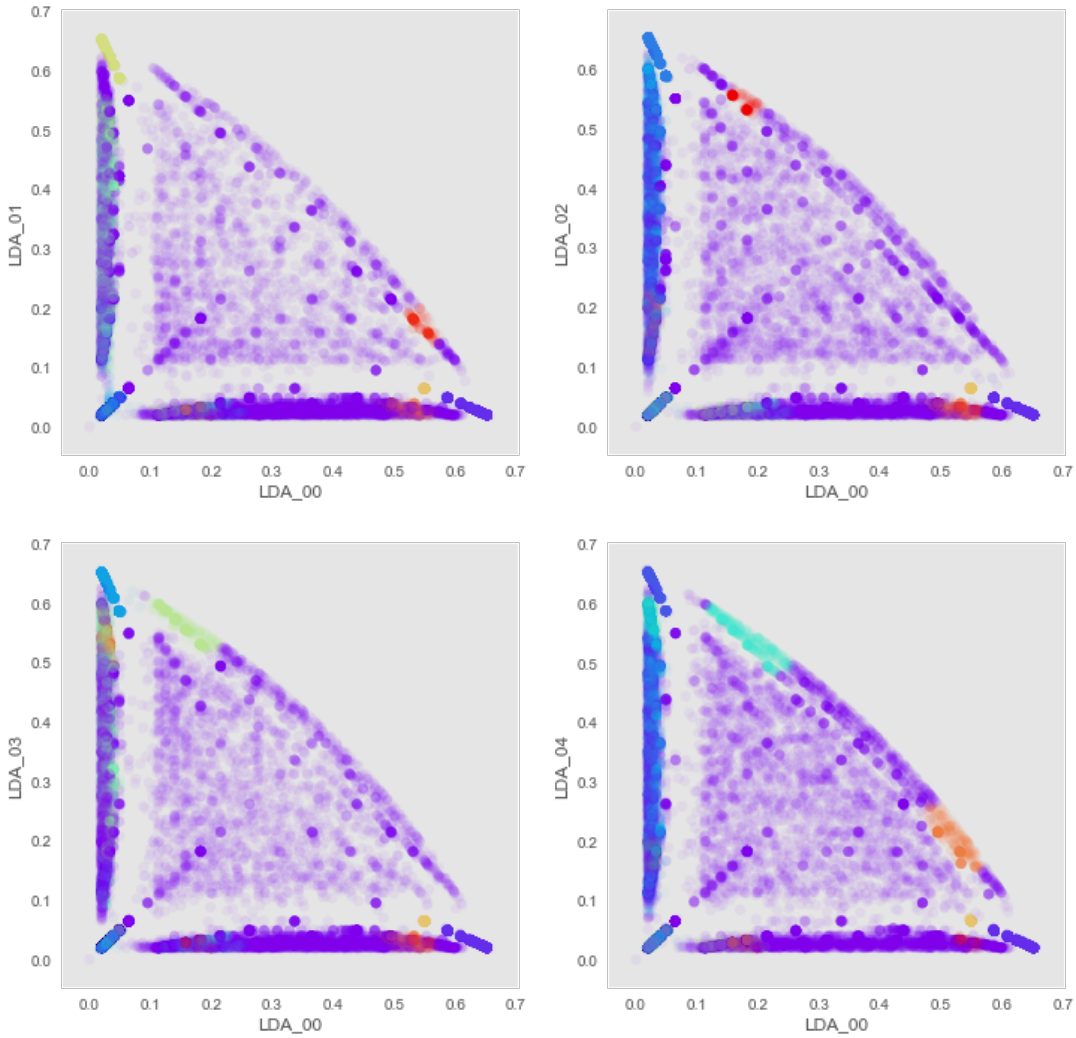

```
eps, min_pts, nclusters =  0.03 130 27
silhouette =  0.170528060286
```



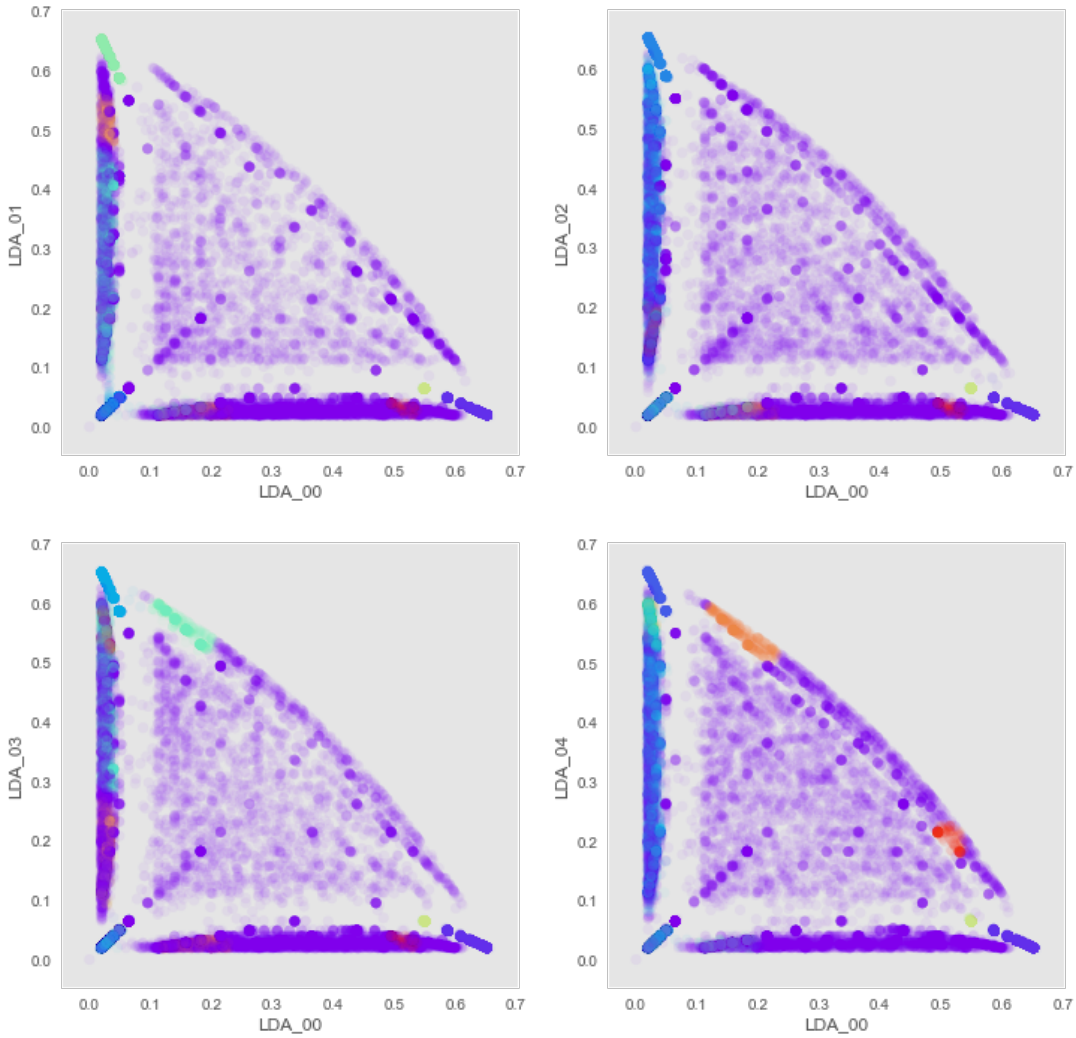

```
eps, min_pts, nclusters =  0.03 150 24
silhouette =  0.118947353013
```



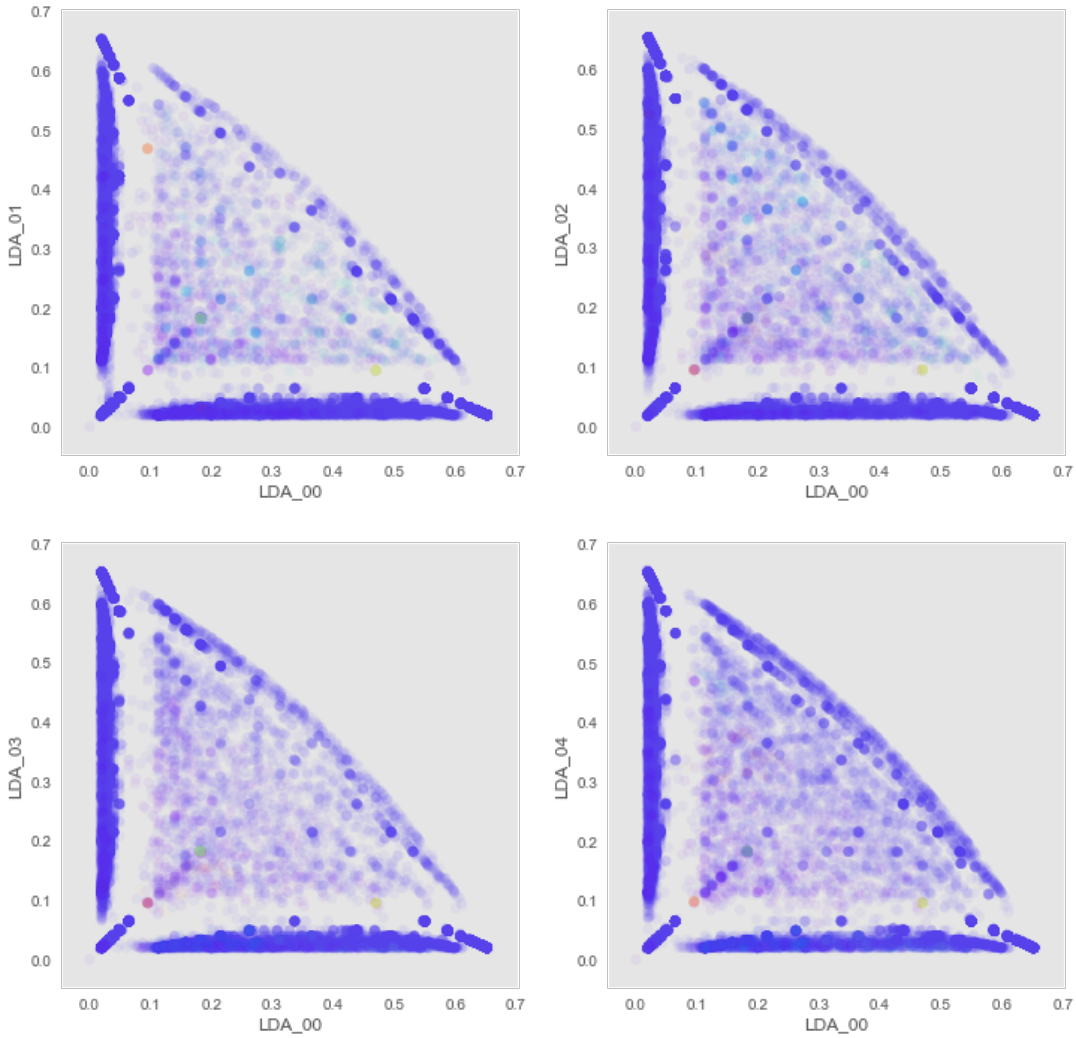

```
eps, min_pts, nclusters =  0.03 170 19
silhouette =  0.109054800548
```



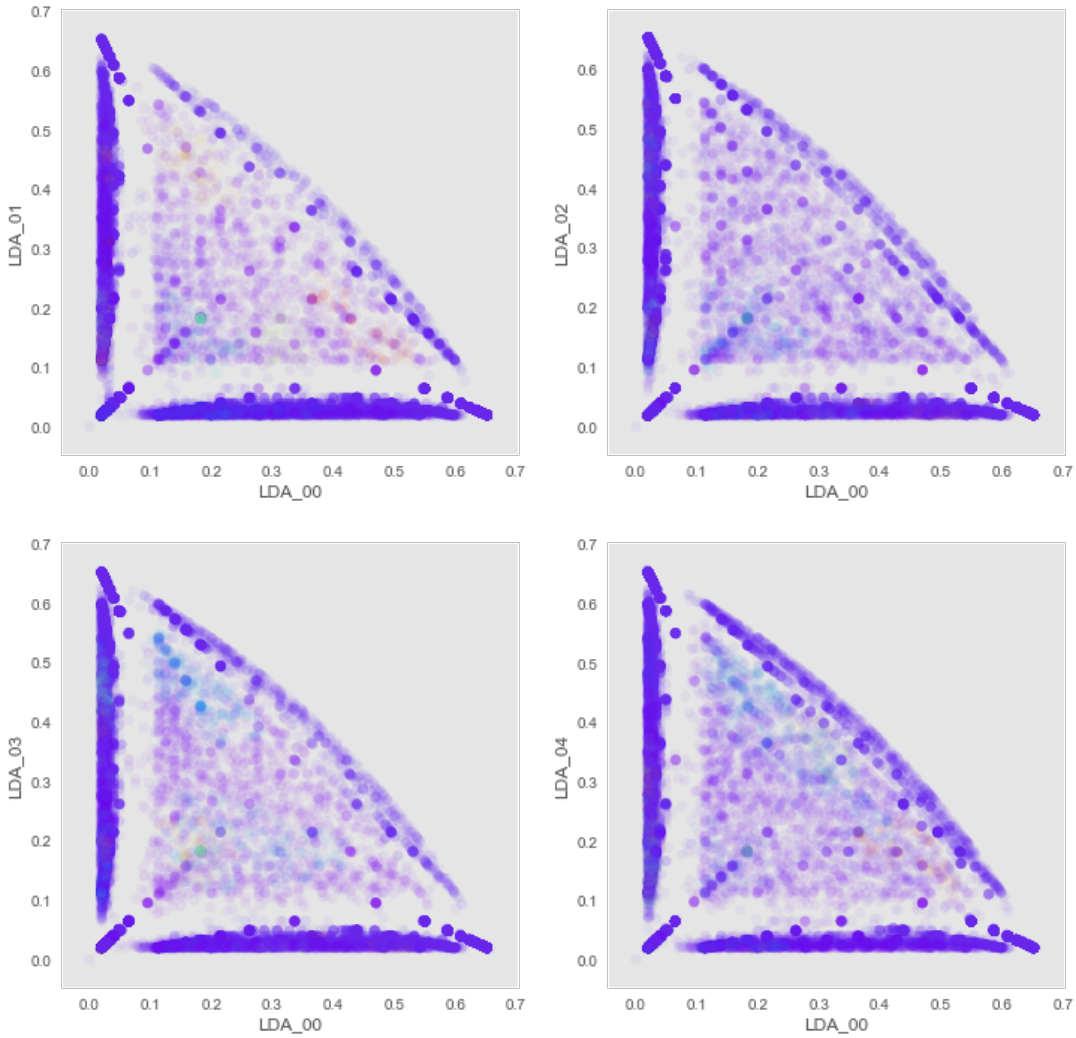

```
eps, min_pts, nclusters =  0.03 190 18
silhouette =  0.079039109307
```



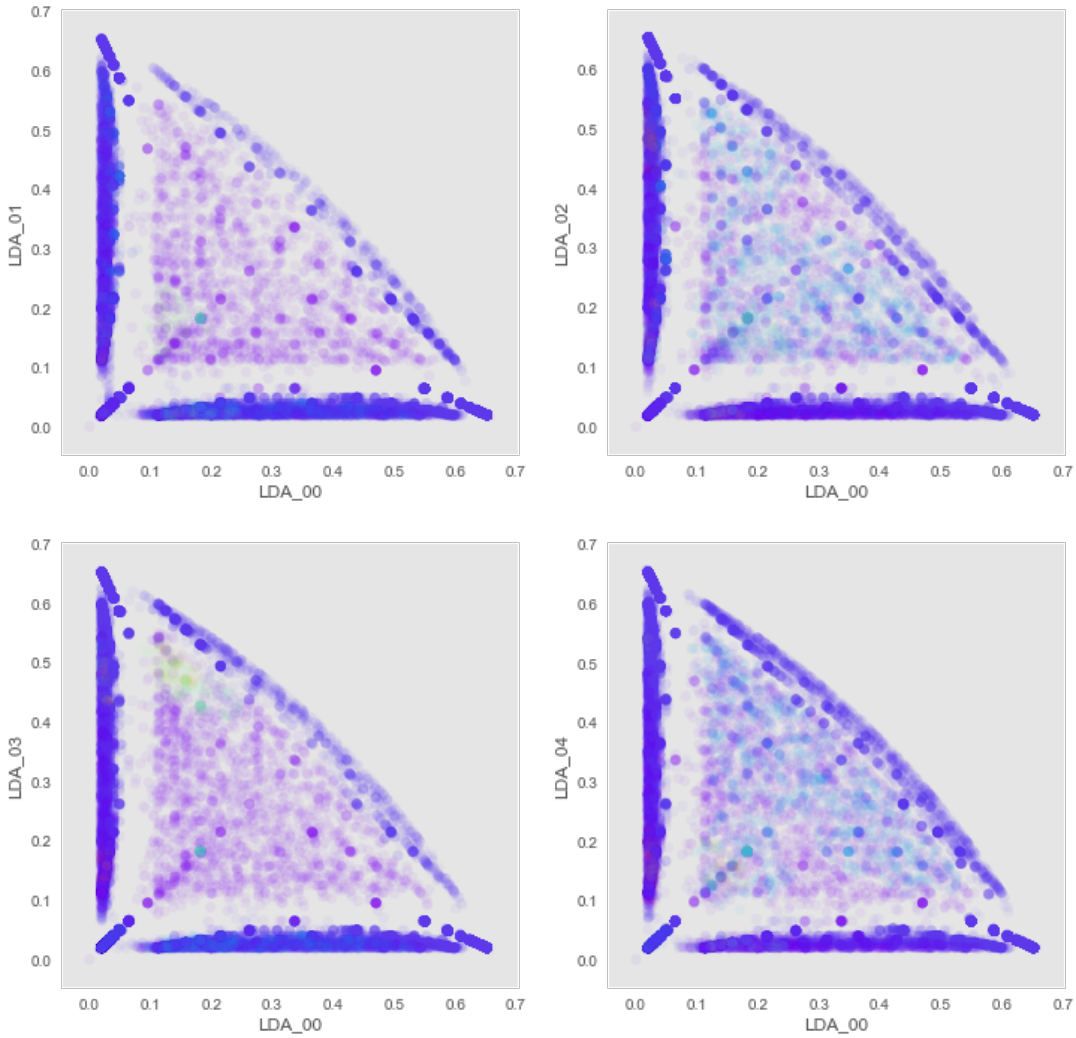

```
eps, min_pts, nclusters =  0.05 10 13
silhouette =  -0.554975391505
```



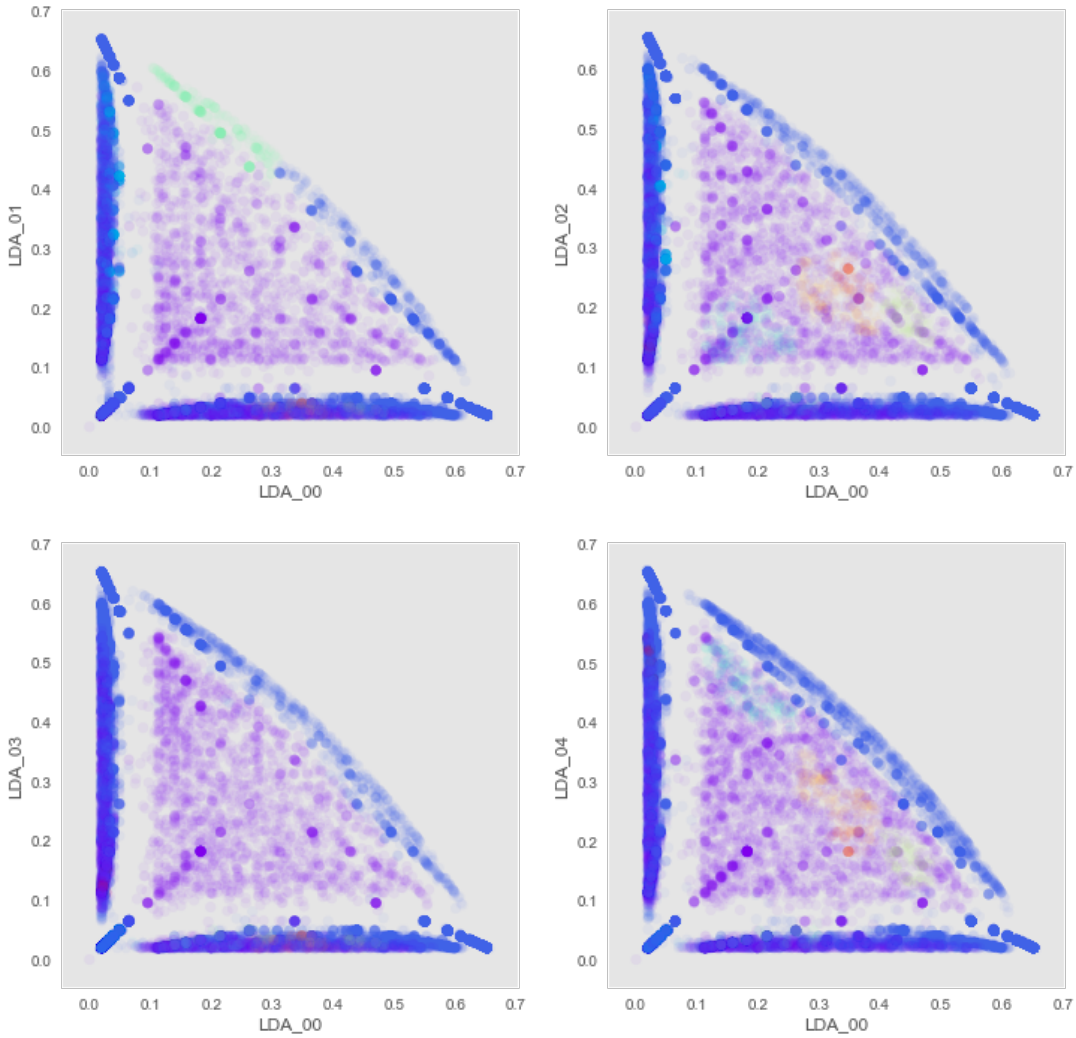

```
eps, min_pts, nclusters =  0.05 30 22
silhouette =  -0.583245695382
```



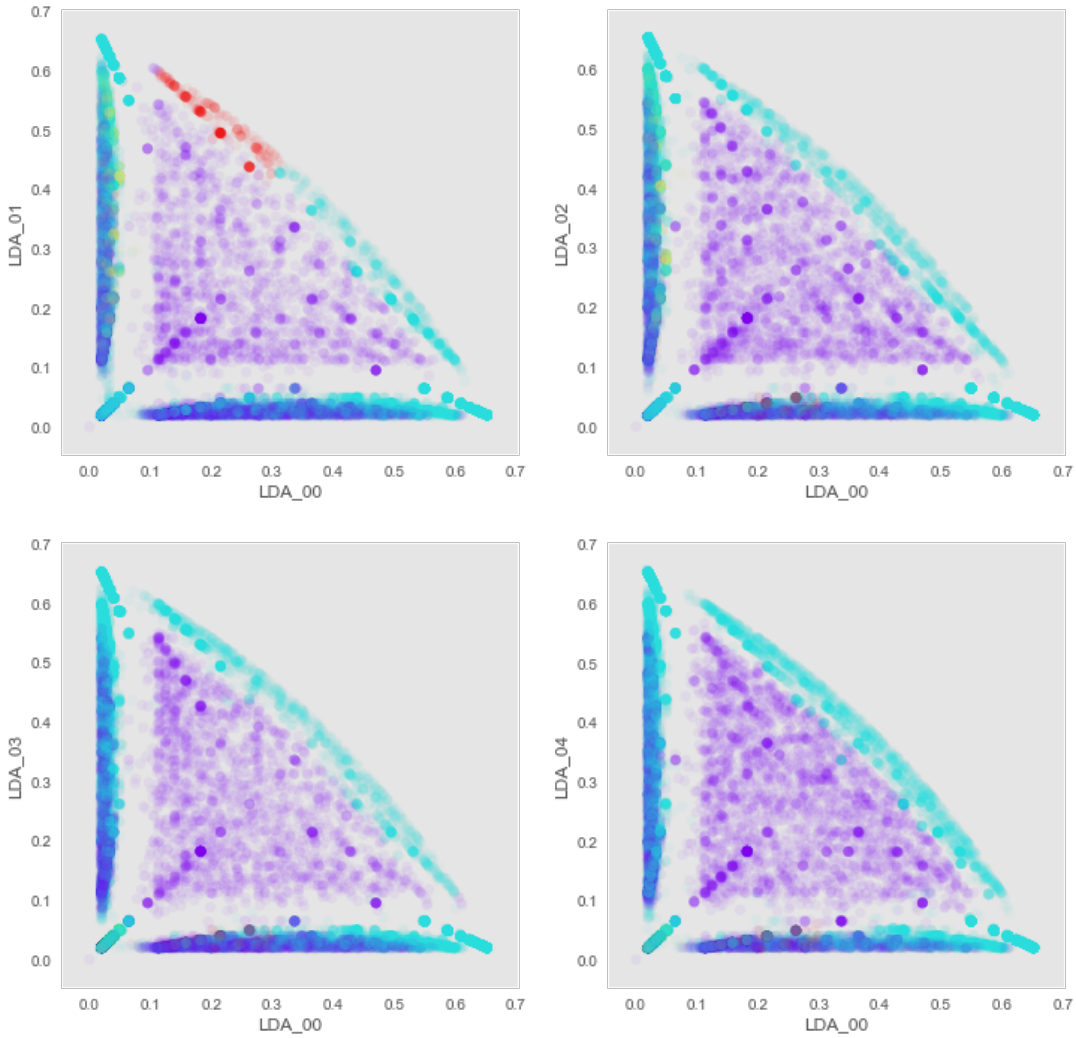

```
eps, min_pts, nclusters =  0.05 50 15
silhouette =  -0.508581617653
```



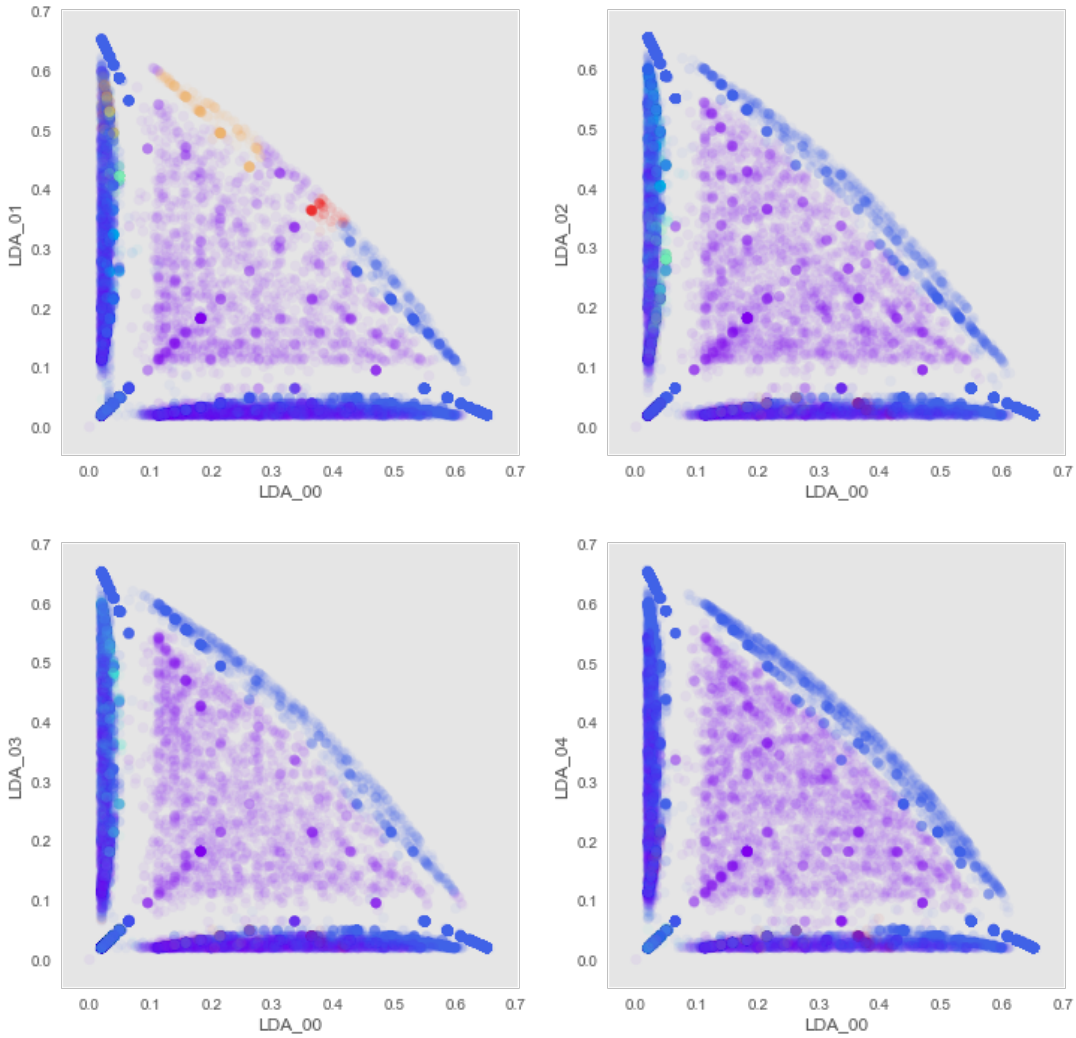

```
eps, min_pts, nclusters =  0.05 70 9
silhouette =  -0.384019284454
```



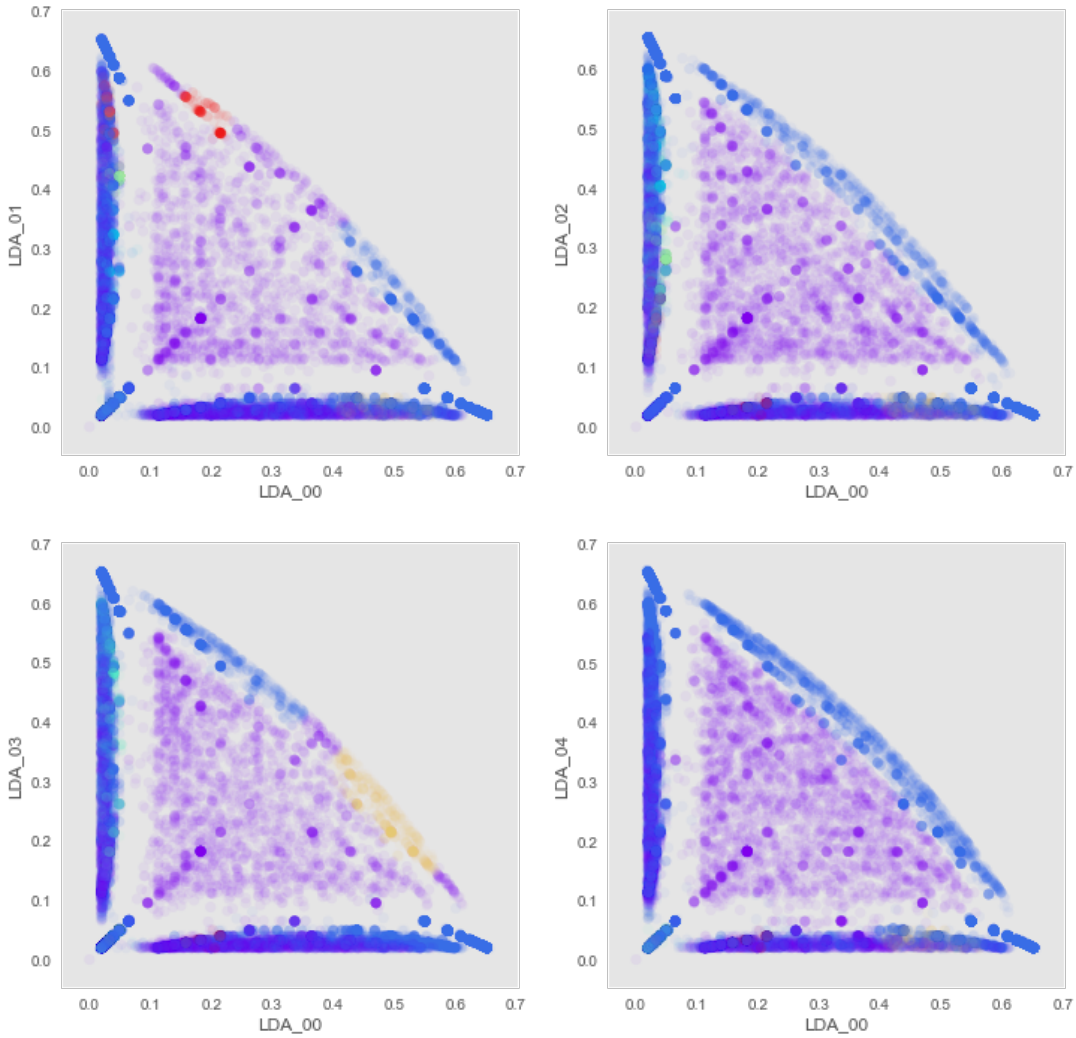

```
eps, min_pts, nclusters =  0.05 90 4
silhouette =  -0.143251757675
```



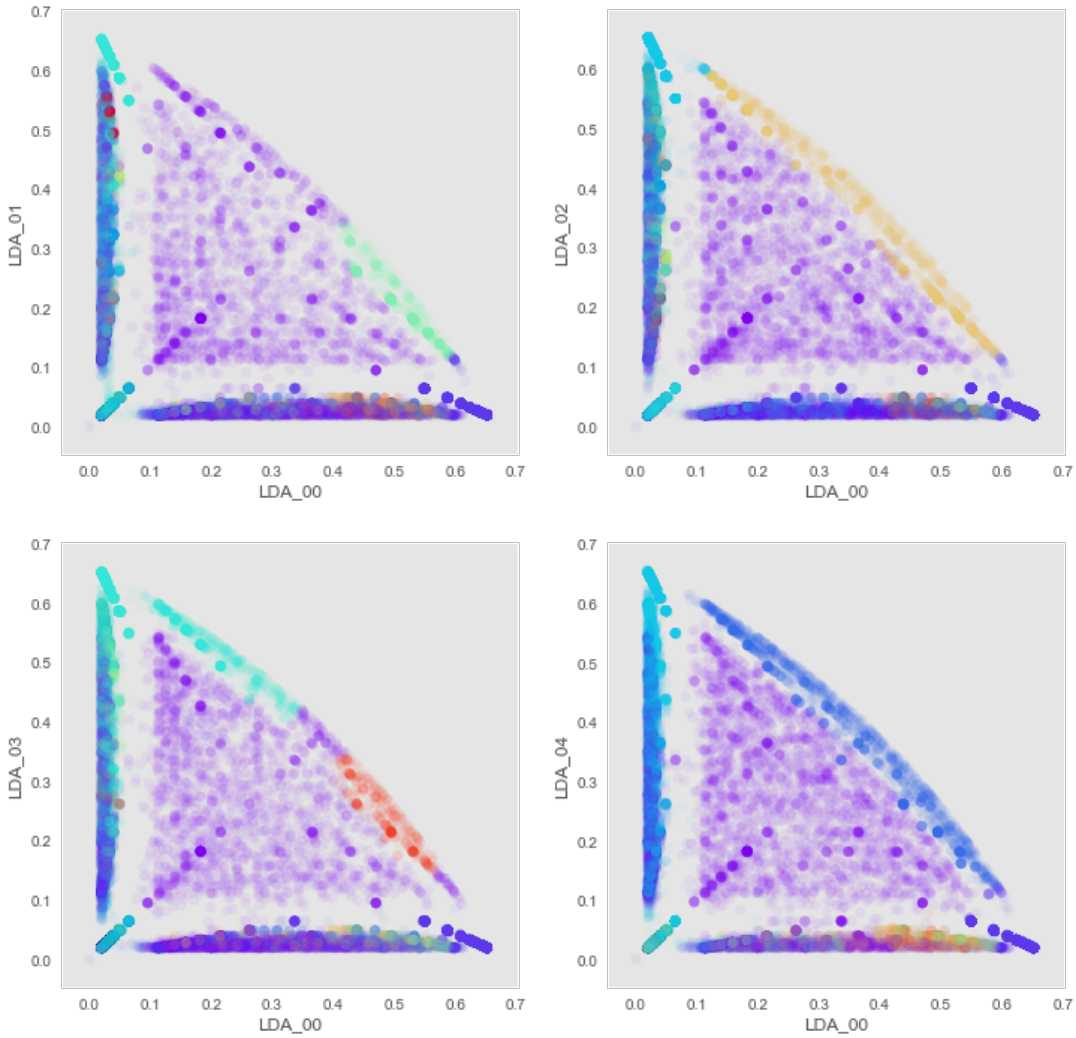

```
eps, min_pts, nclusters =  0.05 110 9
silhouette =  -0.266437896558
```



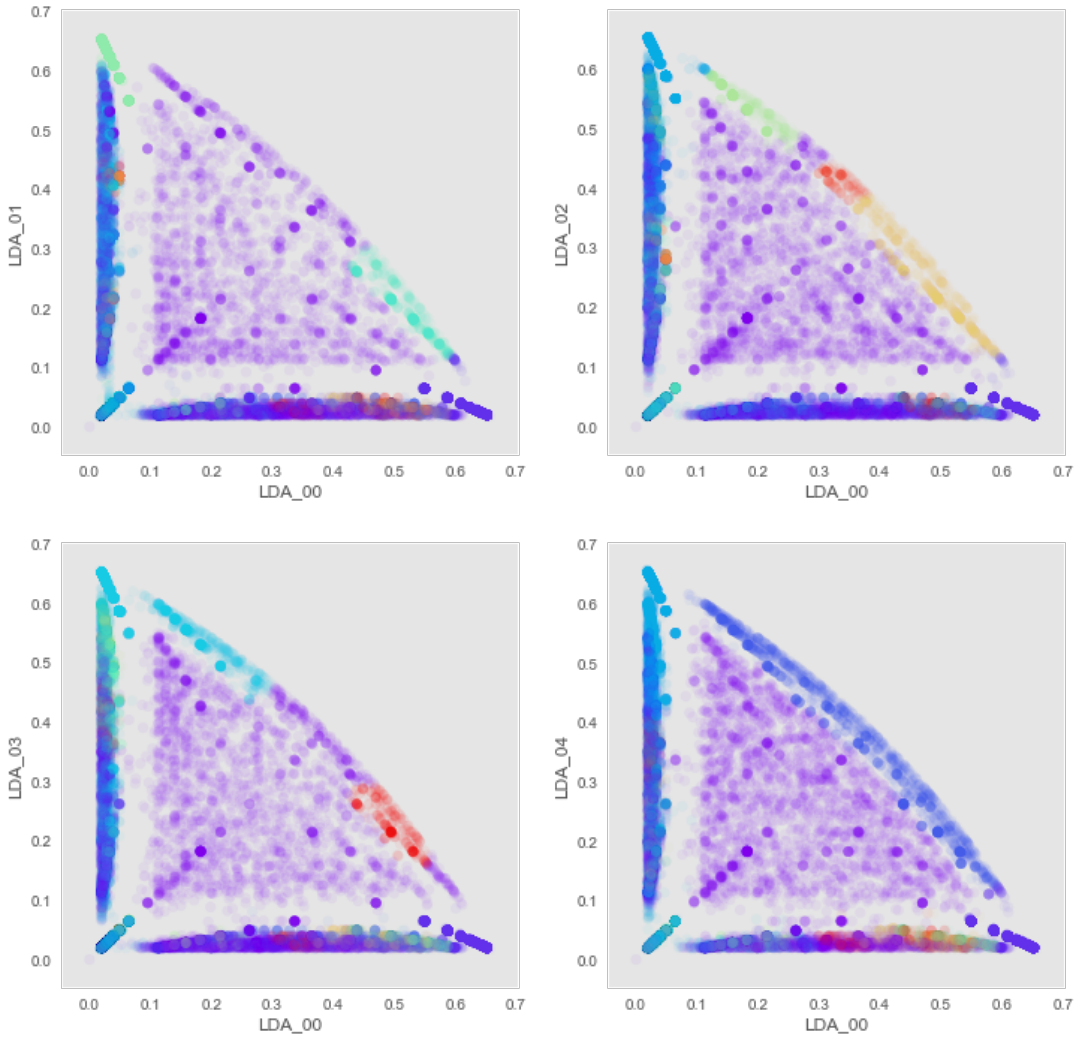

```
eps, min_pts, nclusters =  0.05 130 8
silhouette =  -0.291059815599
```



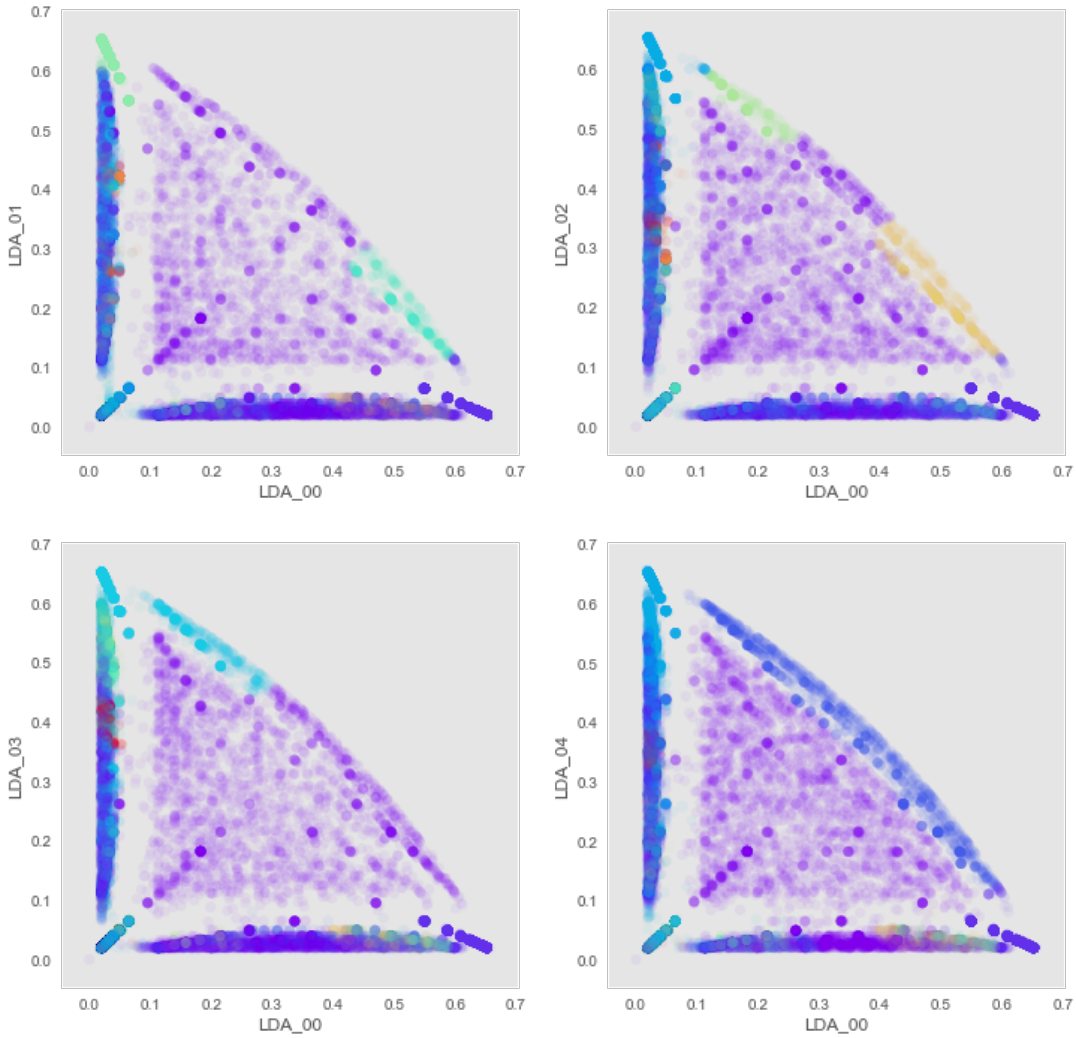

```
eps, min_pts, nclusters =  0.05 150 15
silhouette =  -0.0876455920594
```



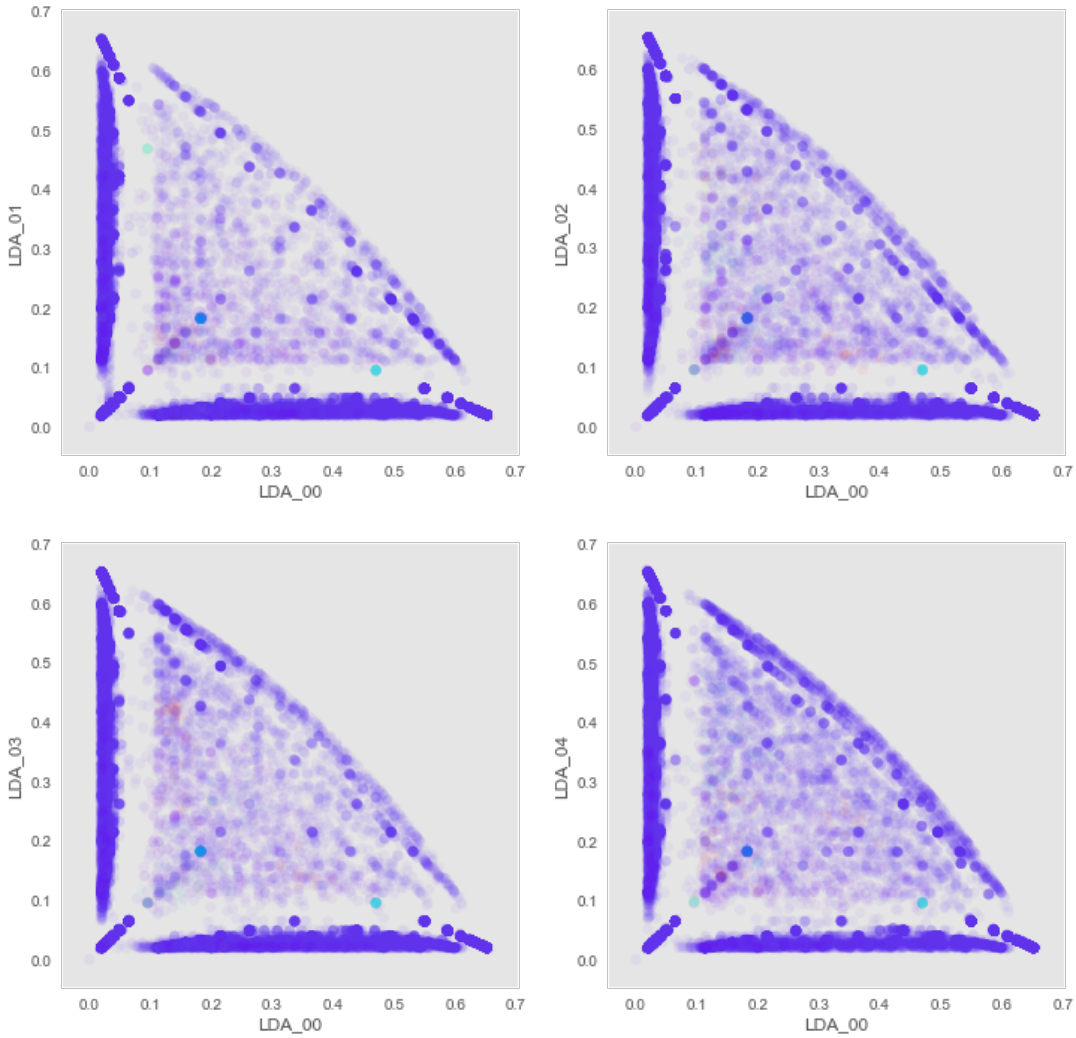

```
eps, min_pts, nclusters =  0.05 170 18
silhouette =  -0.00363225510685
```



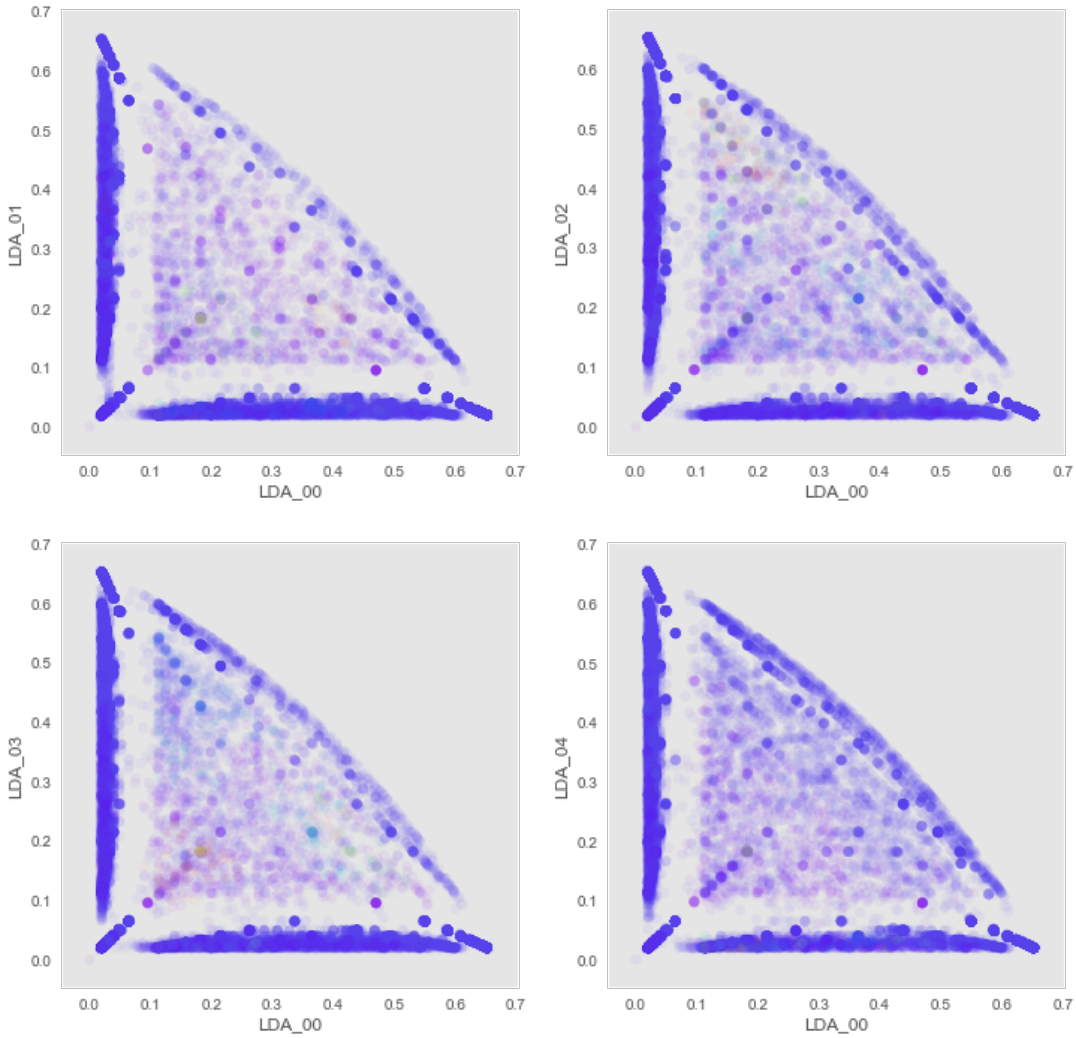

```
eps, min_pts, nclusters =  0.05 190 18
silhouette =  -0.0219842244597
```



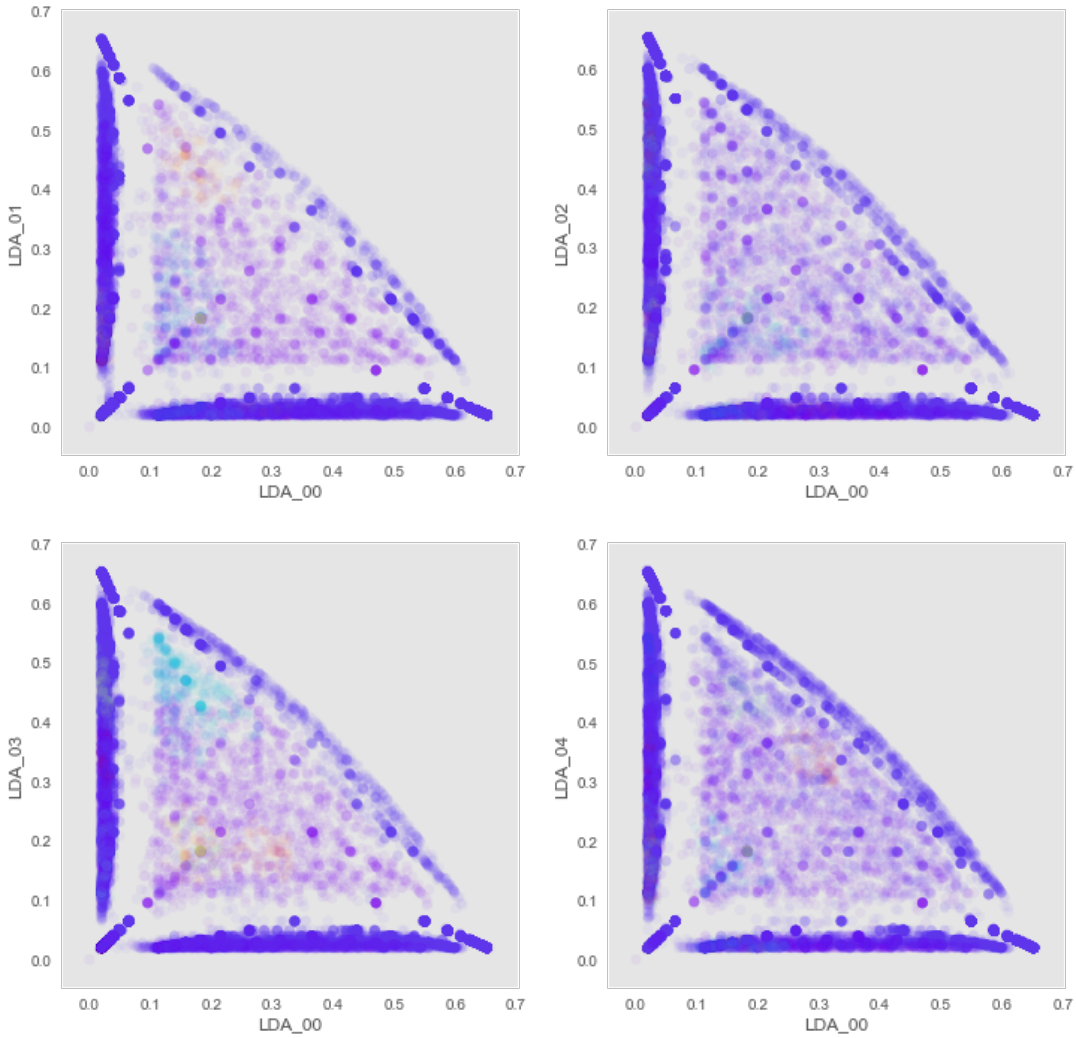

```
eps, min_pts, nclusters =  0.06 10 17
silhouette =  -0.625087159318
```



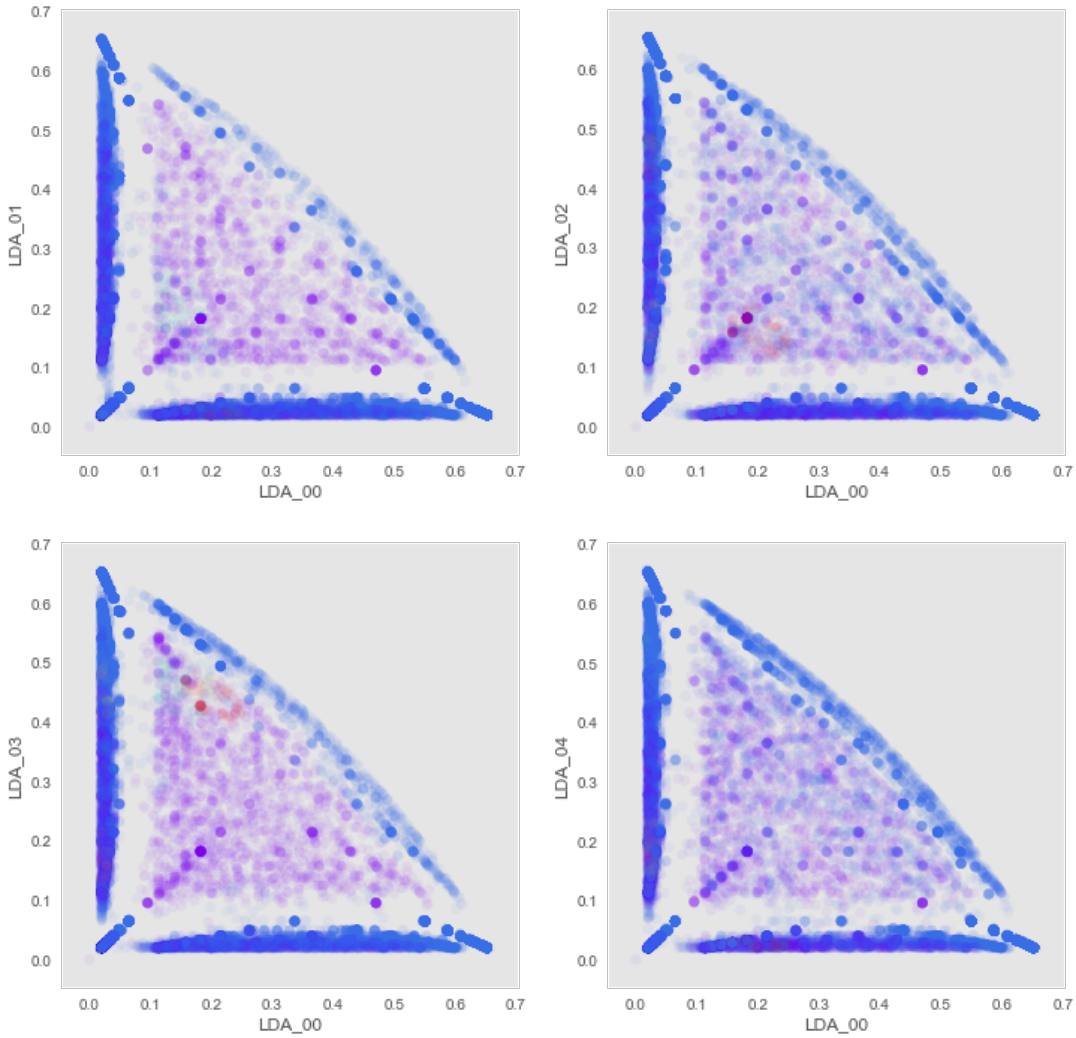

```
eps, min_pts, nclusters =  0.06 30 13
silhouette =  -0.509683854142
```



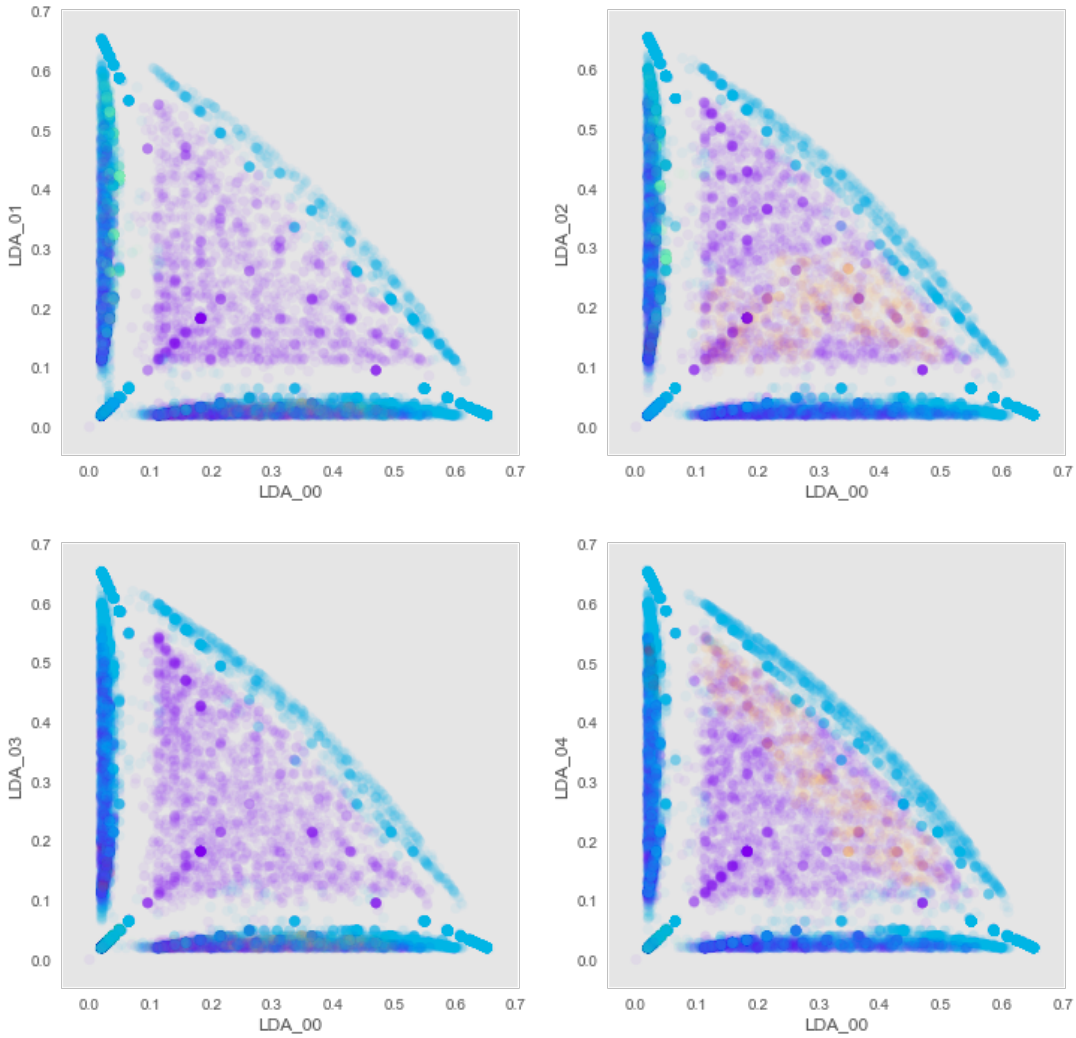

```
eps, min_pts, nclusters =  0.06 50 16
silhouette =  -0.547325246322
```



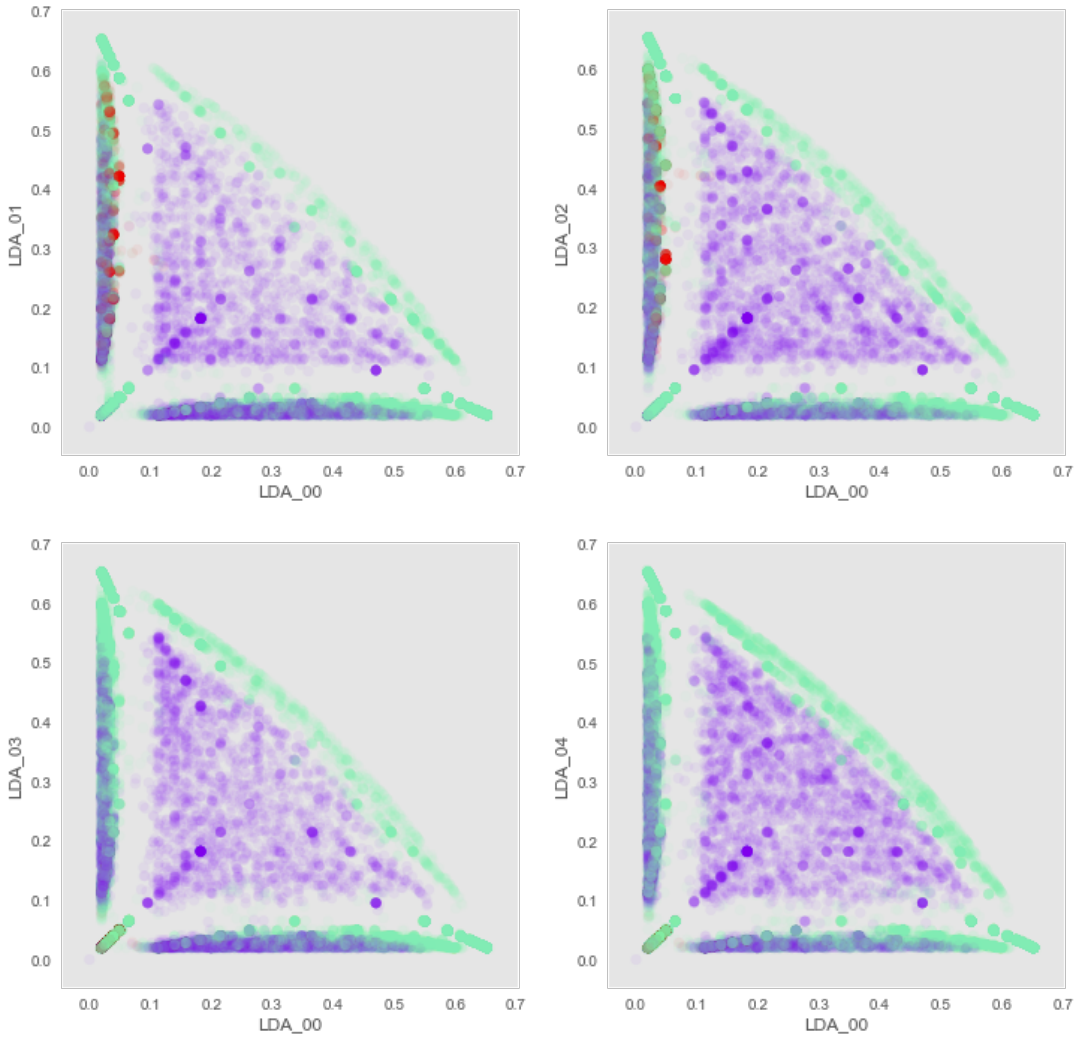

```
eps, min_pts, nclusters =  0.06 70 8
silhouette =  -0.458514547266
```



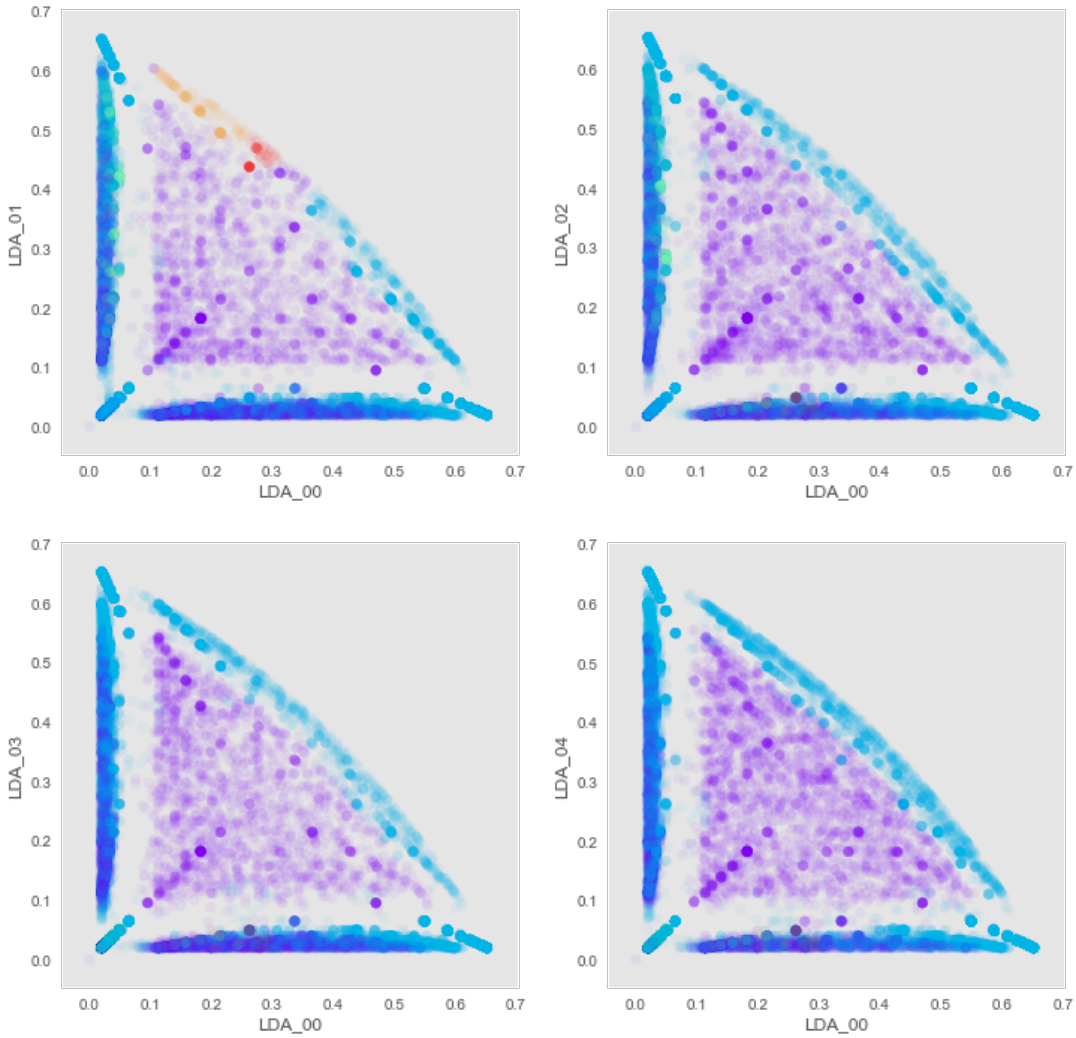

```
eps, min_pts, nclusters =  0.06 90 5
silhouette =  -0.294859343785
```



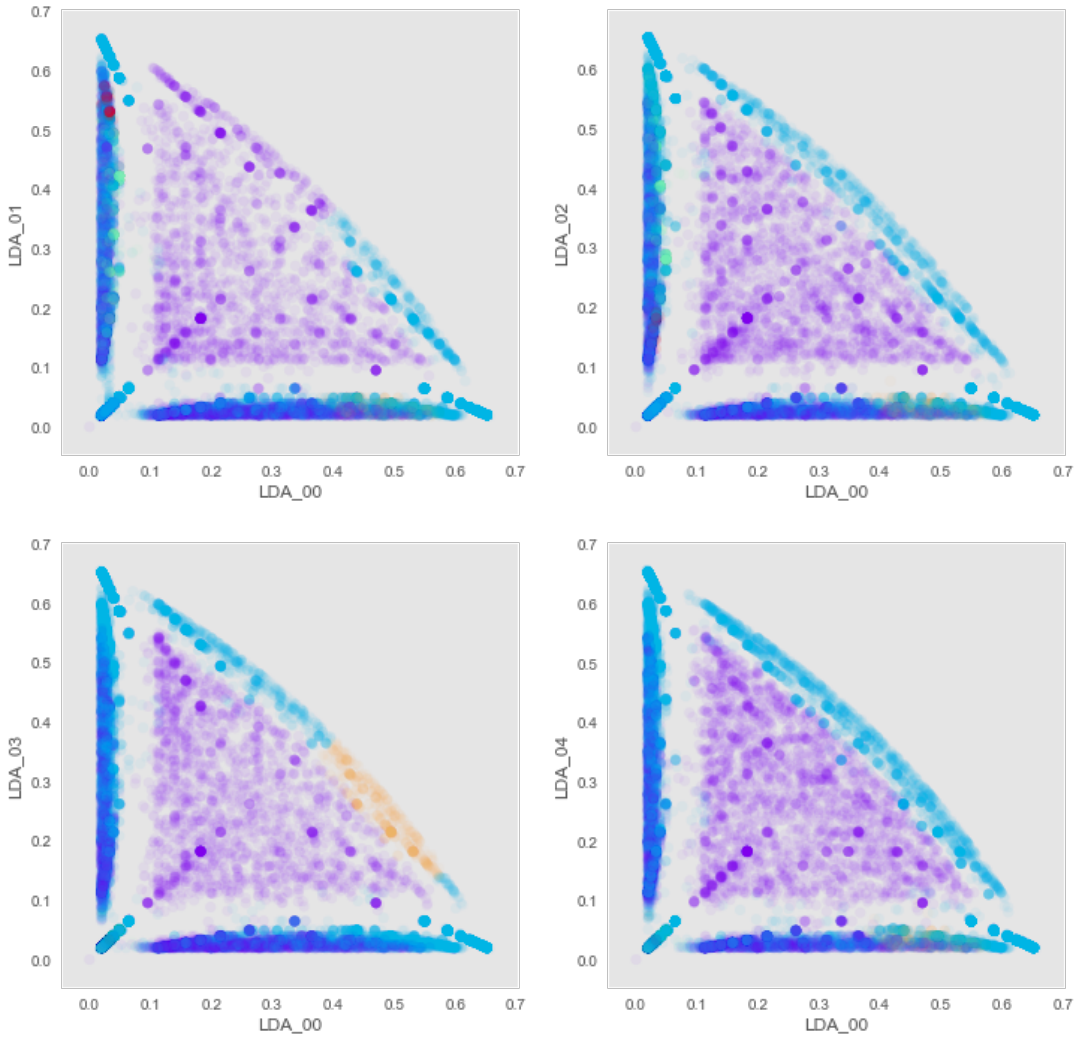

```
eps, min_pts, nclusters =  0.06 110 3
silhouette =  -0.115204301547
```



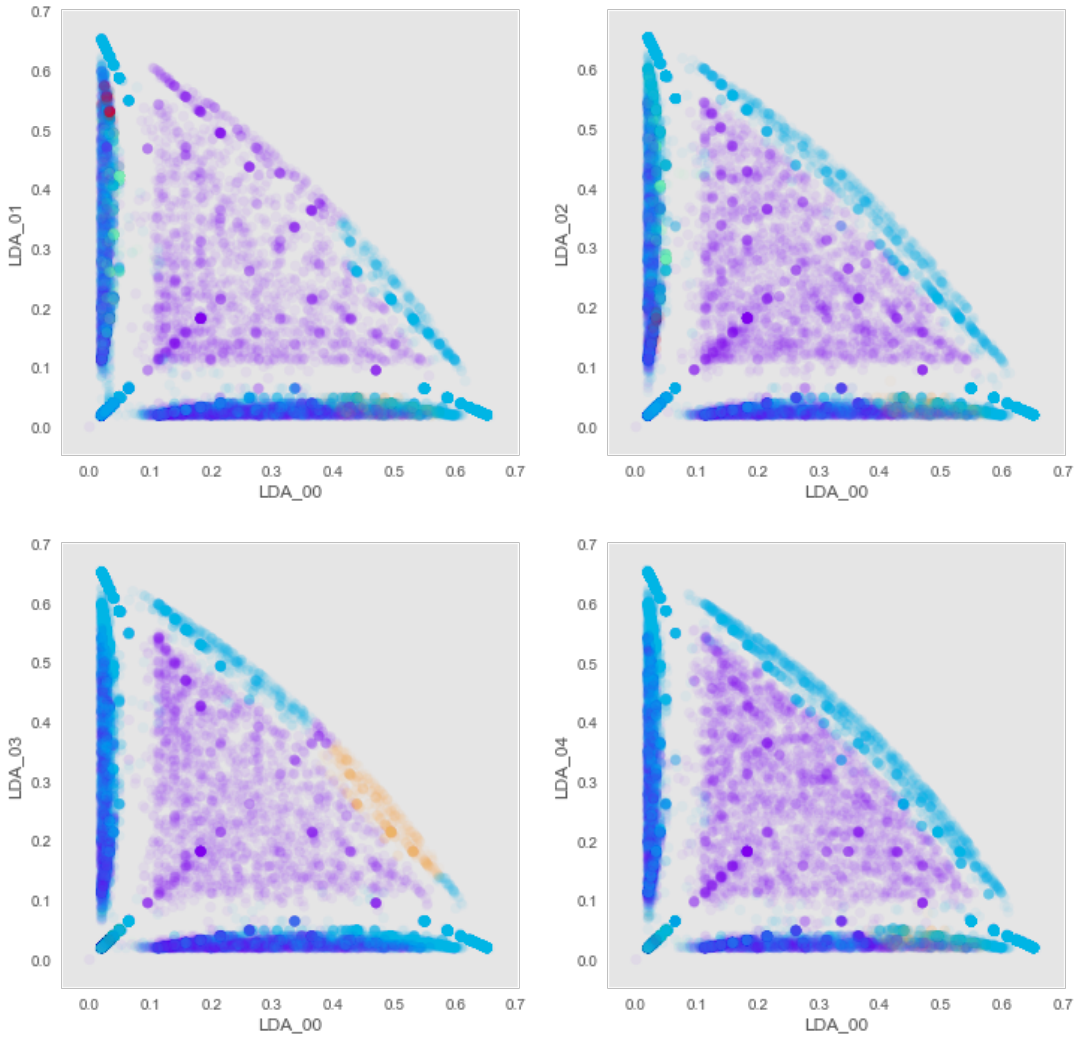

```
eps, min_pts, nclusters =  0.06 130 5
silhouette =  -0.162619911913
```



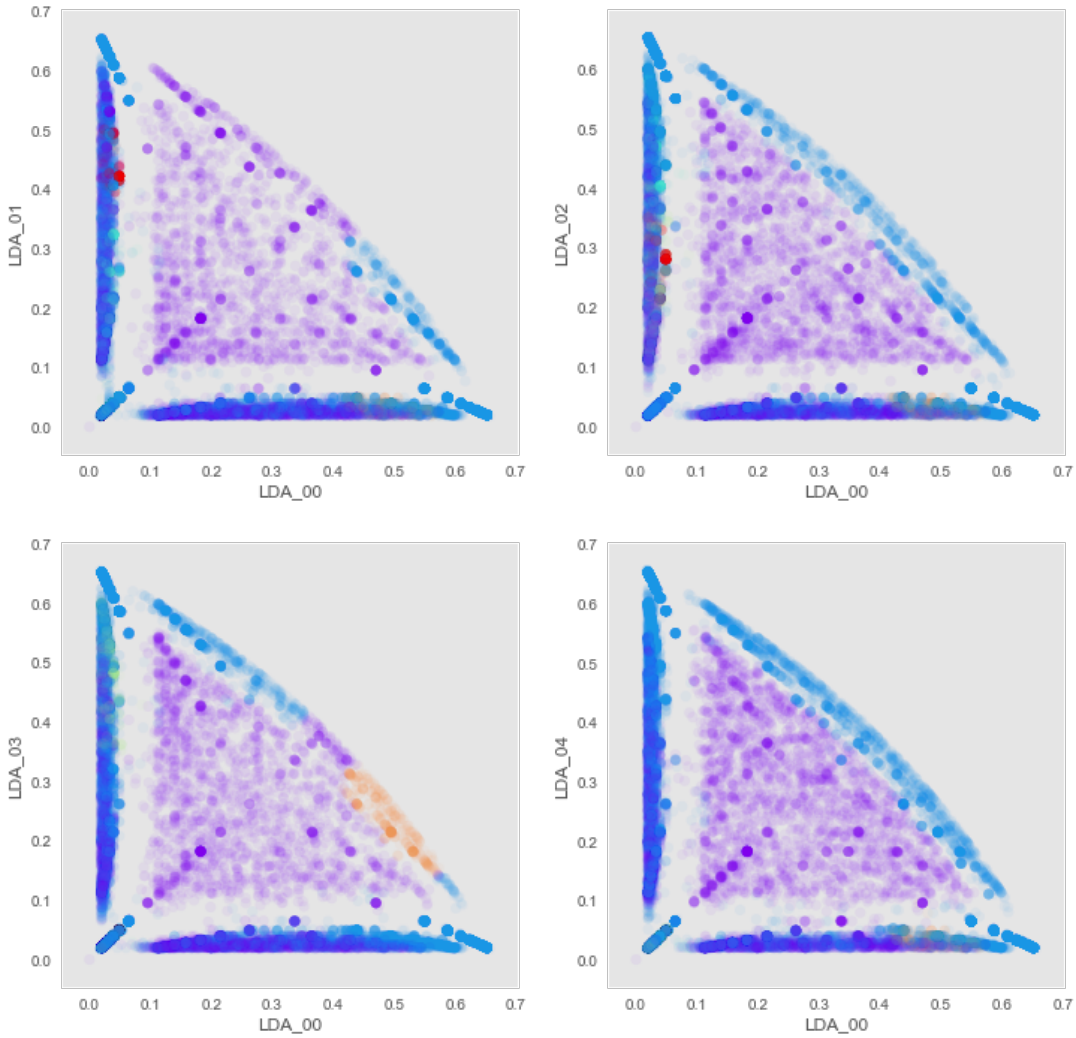

```
eps, min_pts, nclusters =  0.06 150 5
silhouette =  -0.236628413768
```



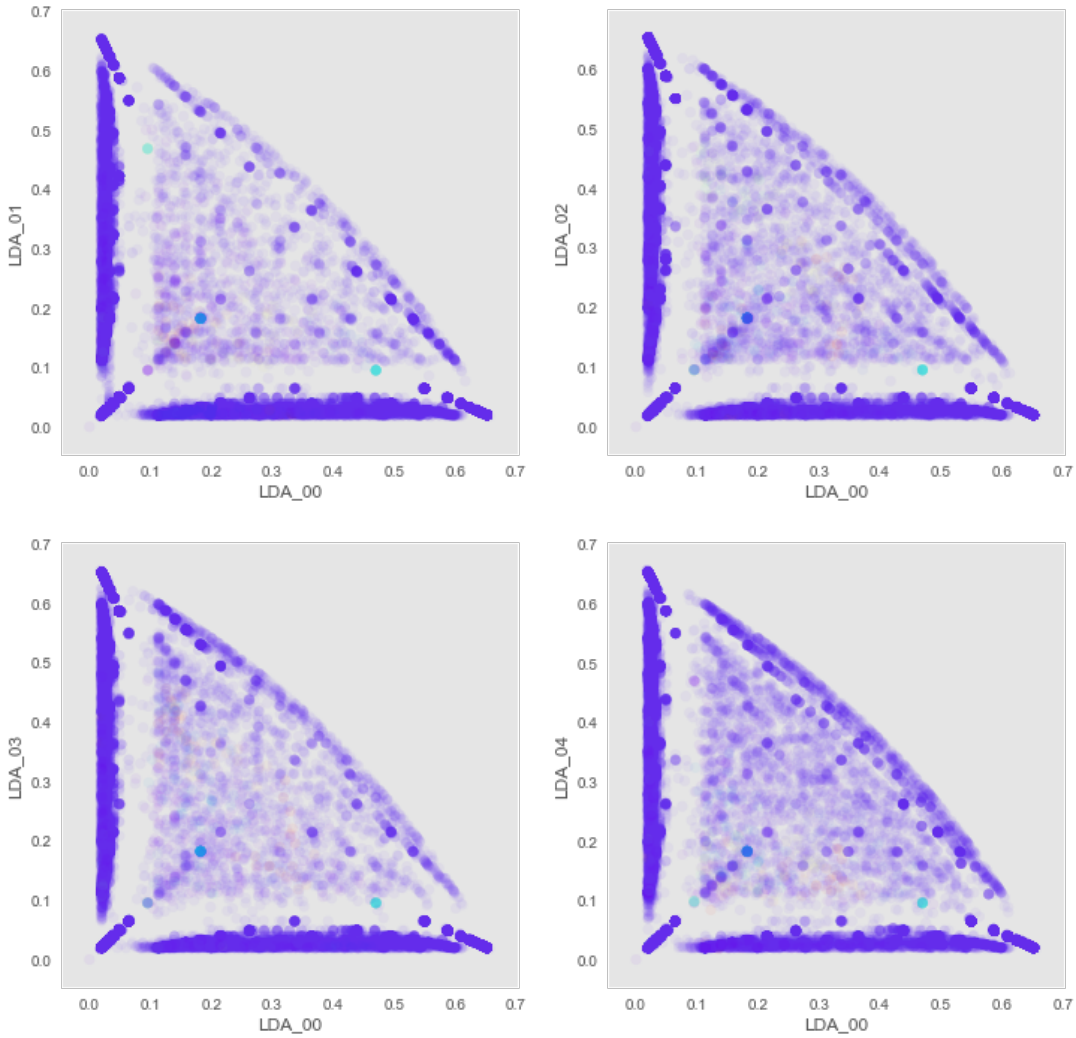

```
eps, min_pts, nclusters =  0.06 170 5
silhouette =  -0.240219582278
```



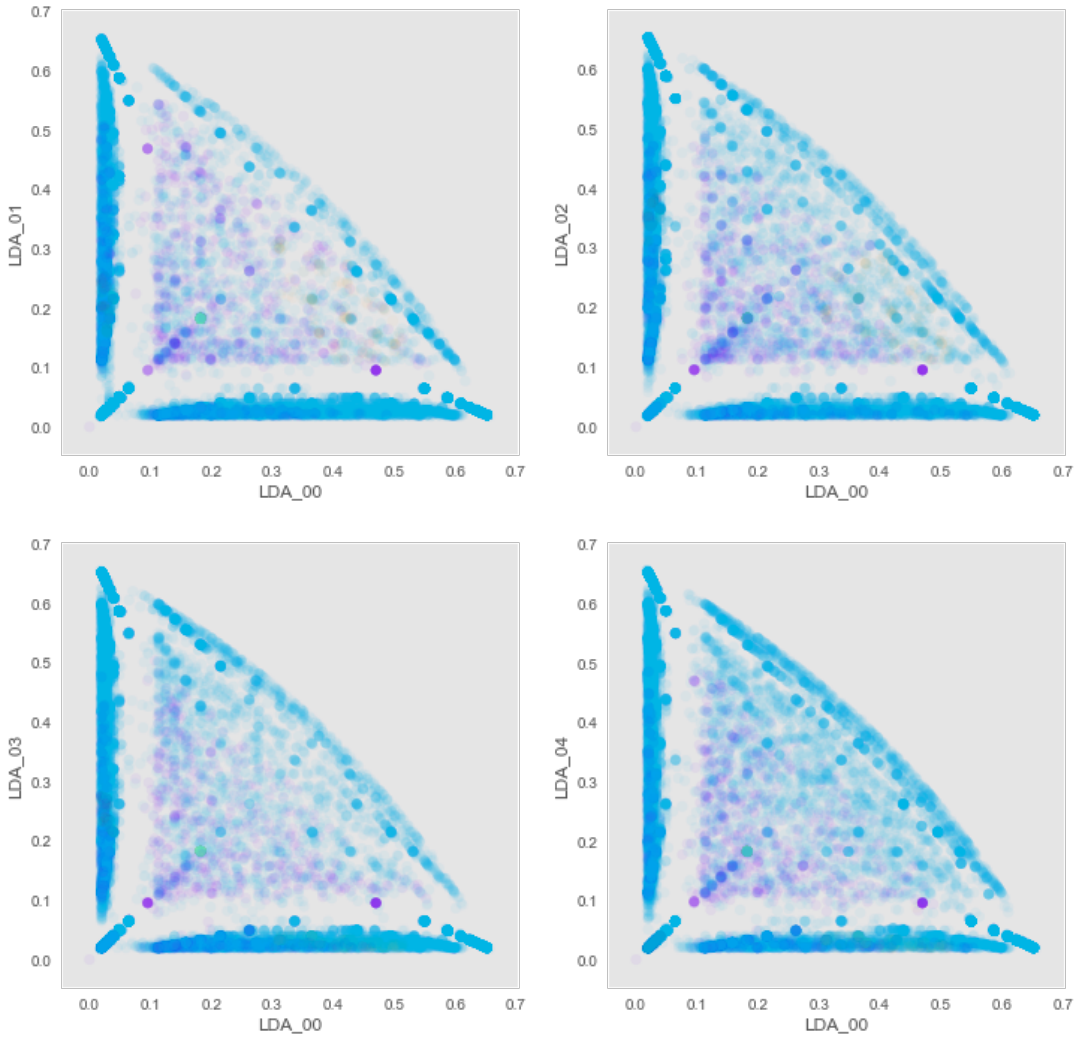

```
eps, min_pts, nclusters =  0.06 190 6
silhouette =  -0.328024893617
```



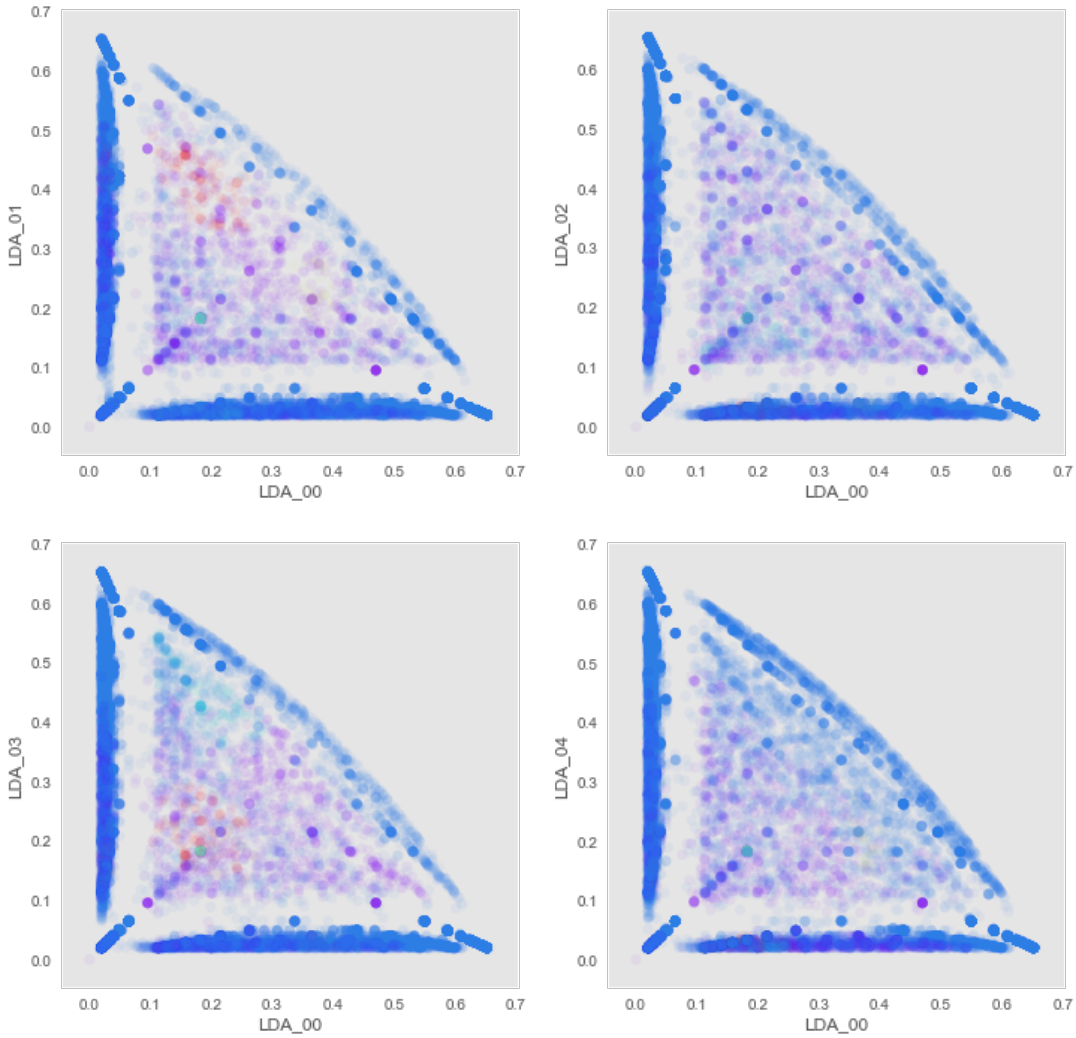

```
eps, min_pts, nclusters =  0.07 10 19
silhouette =  -0.593974977881
```



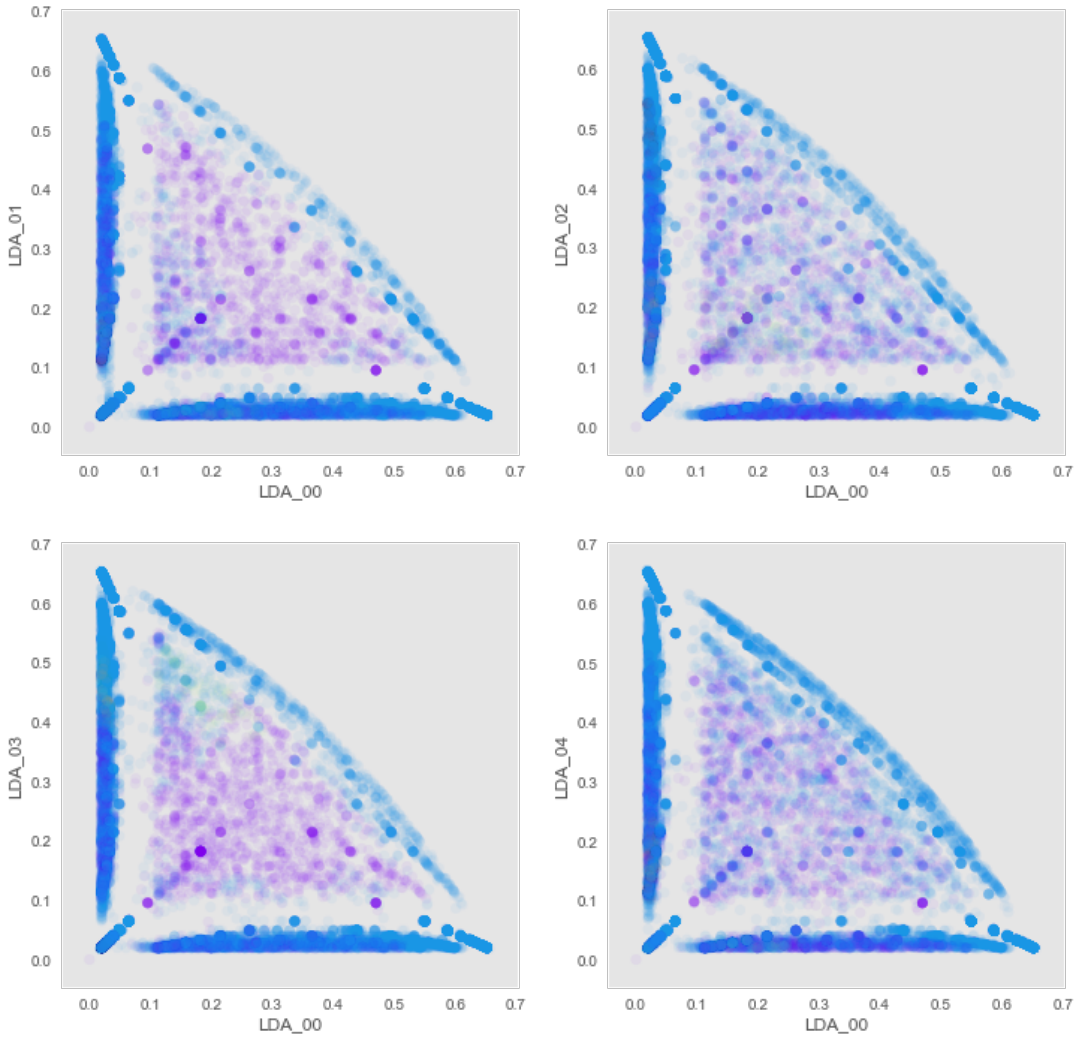

```
eps, min_pts, nclusters =  0.07 30 5
silhouette =  -0.331453669464
```



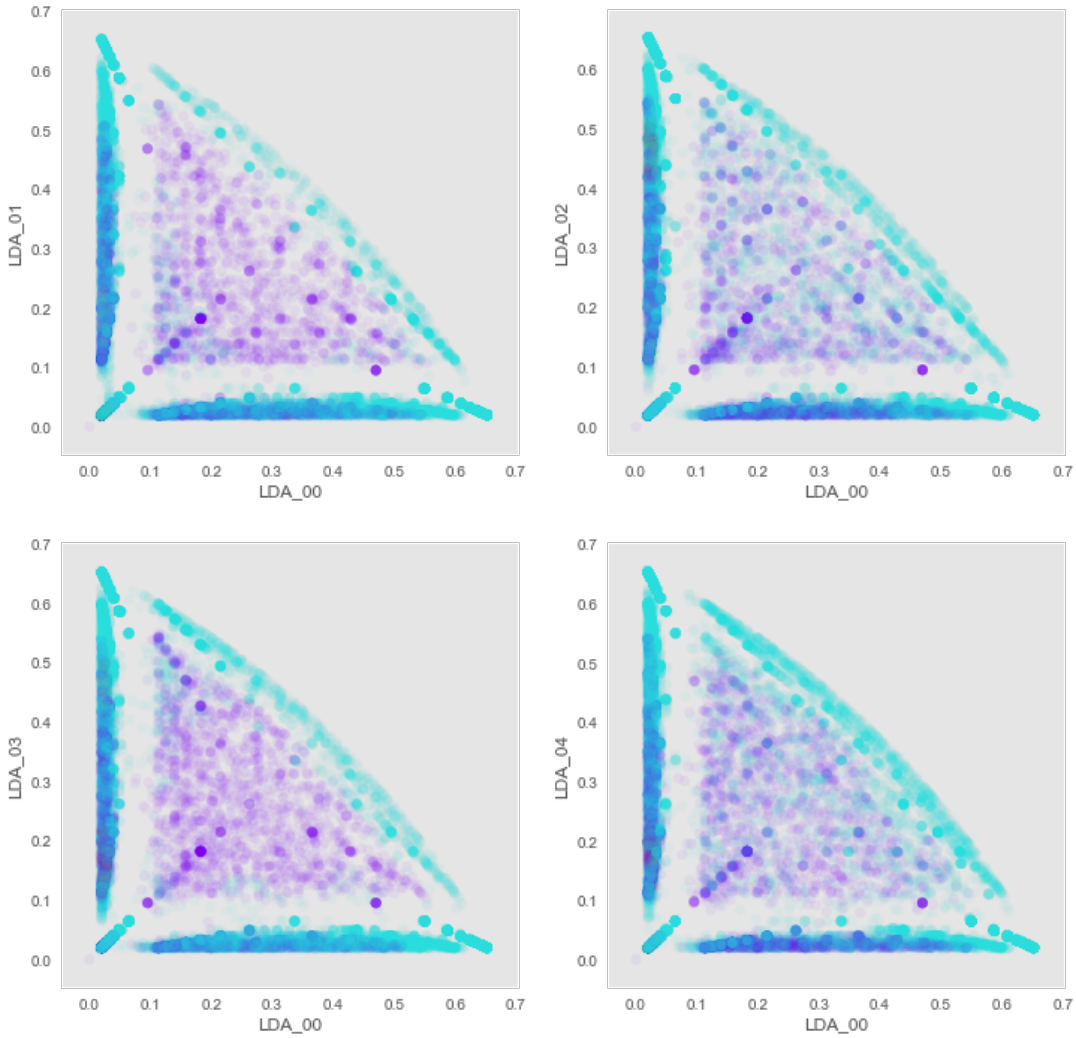

```
eps, min_pts, nclusters =  0.07 50 7
silhouette =  -0.391438906883
```



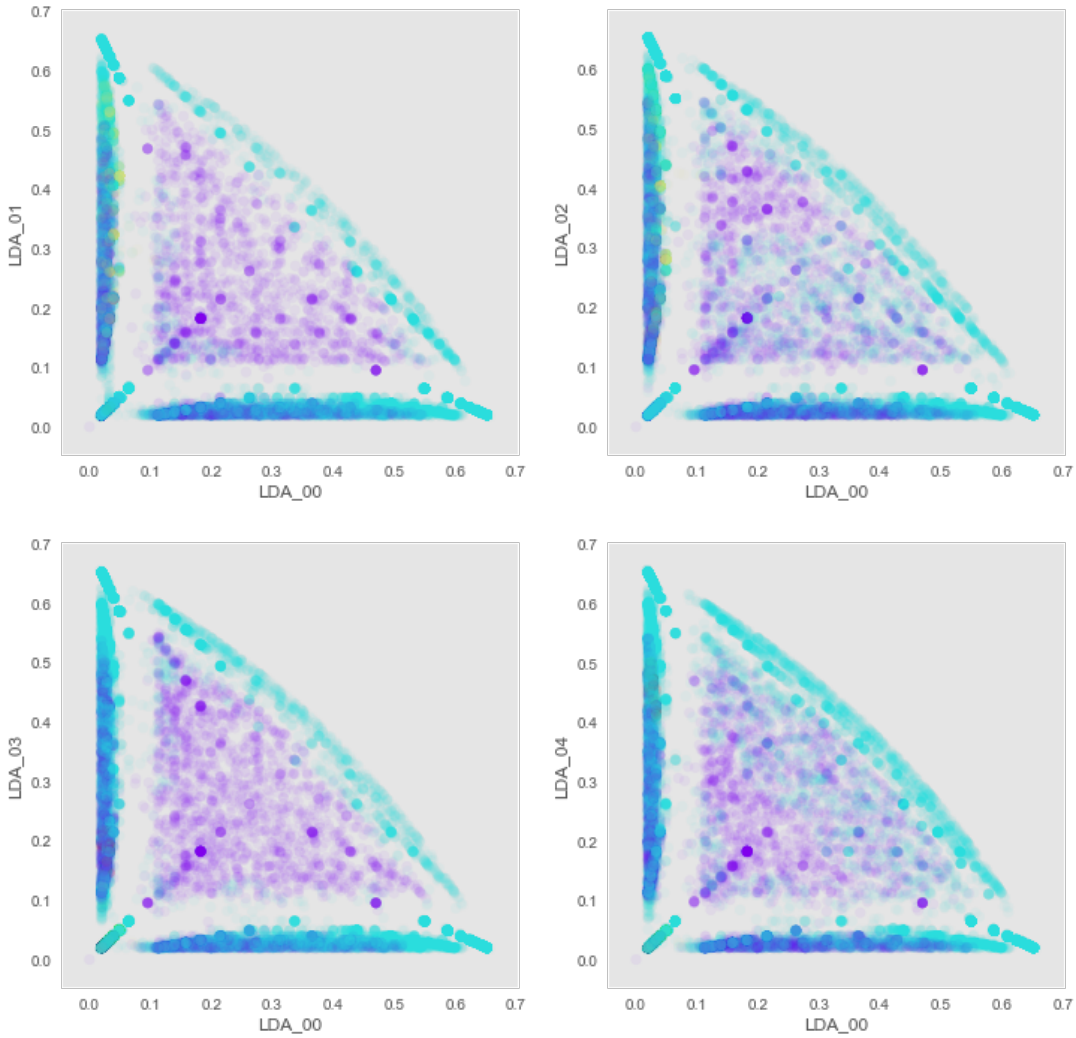

```
eps, min_pts, nclusters =  0.07 70 6
silhouette =  -0.357396694589
```



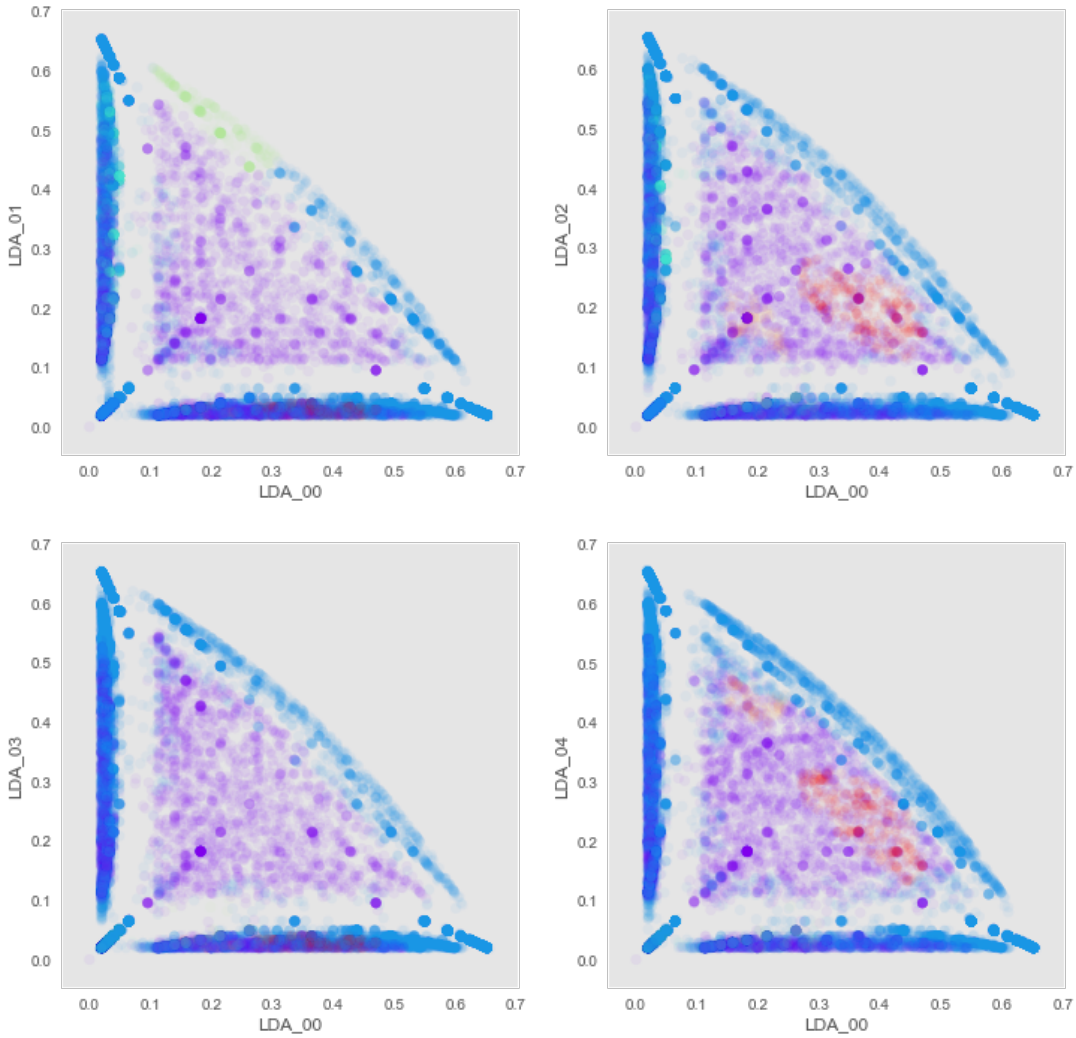

```
eps, min_pts, nclusters =  0.07 90 4
silhouette =  -0.327372947206
```



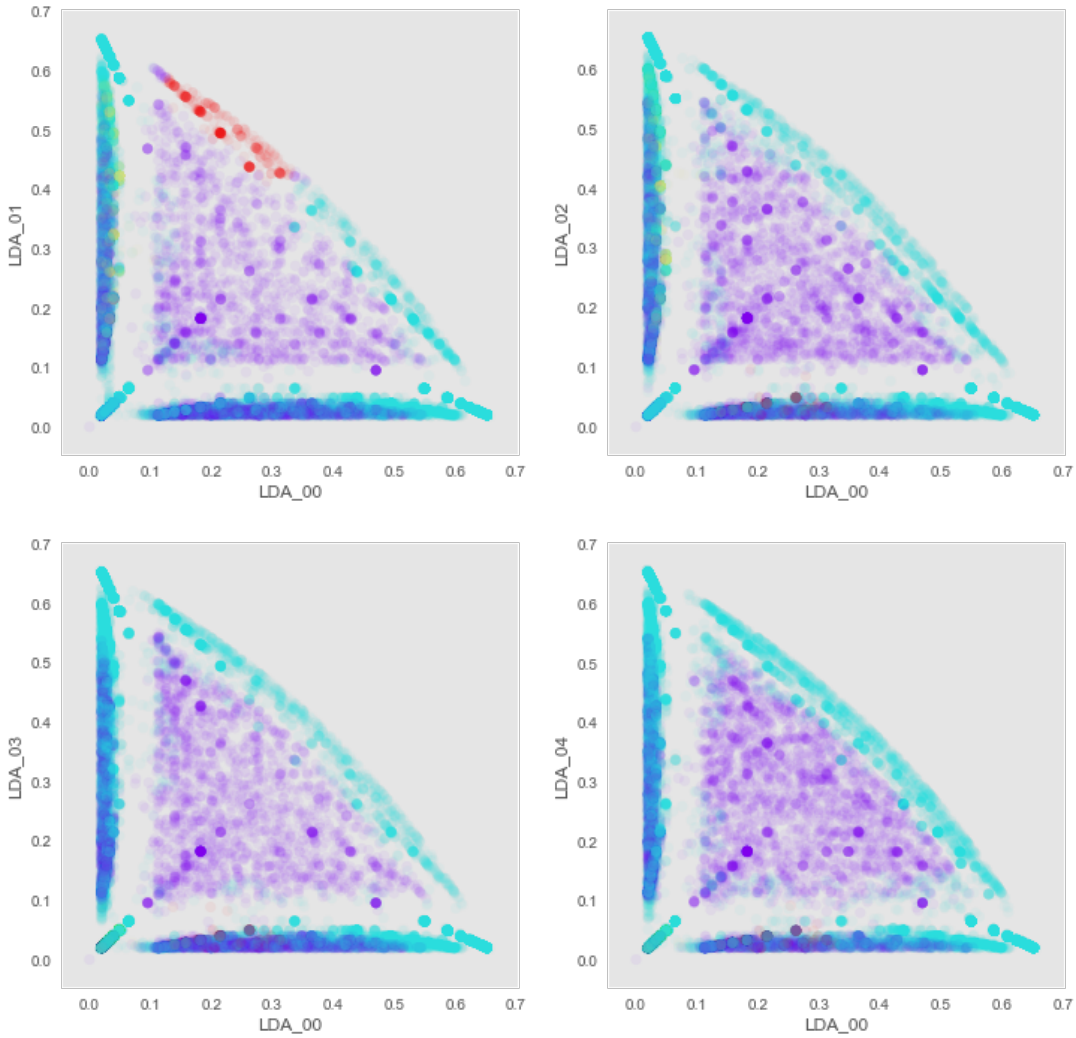

```
eps, min_pts, nclusters =  0.07 110 4
silhouette =  -0.249839903911
```



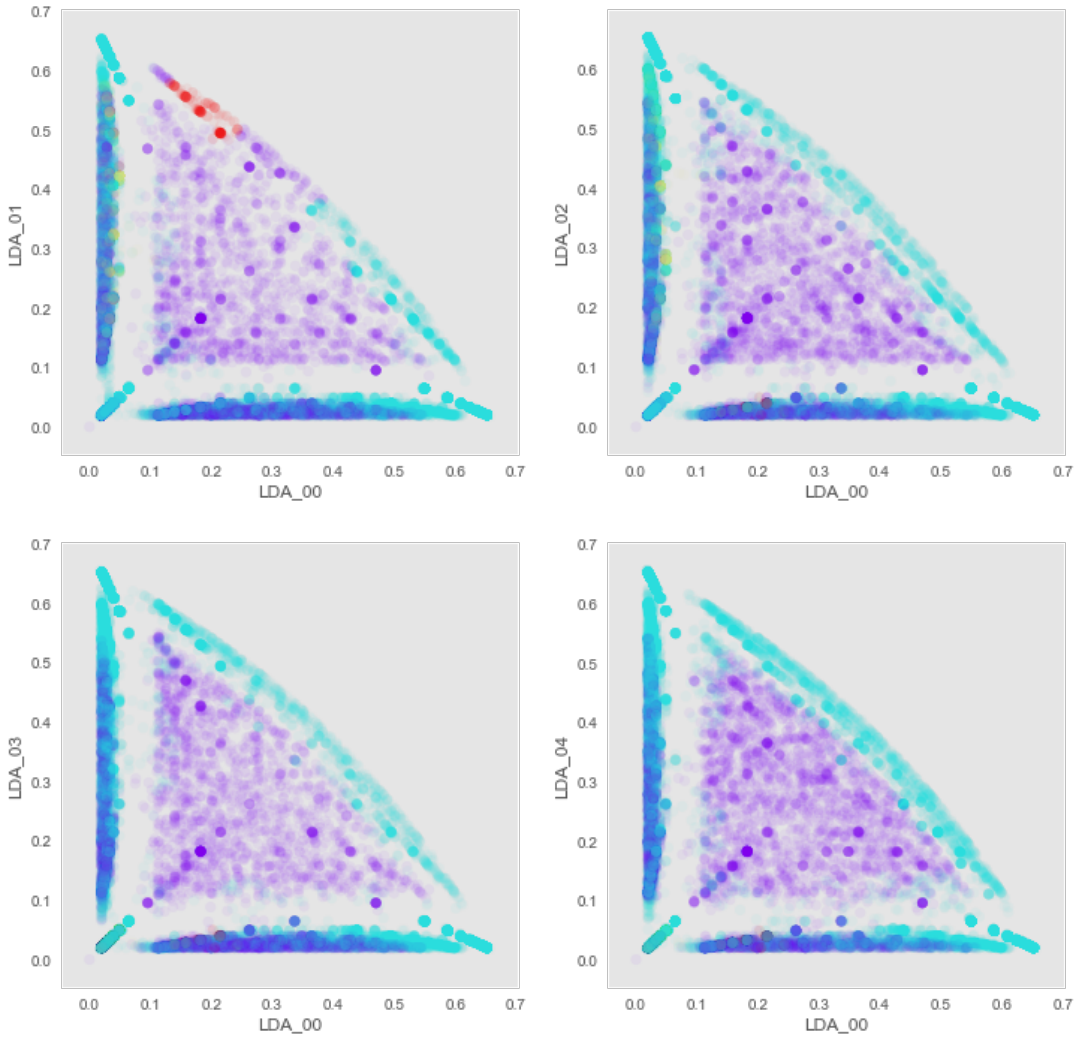

```
eps, min_pts, nclusters =  0.07 130 6
silhouette =  -0.340759262329
```



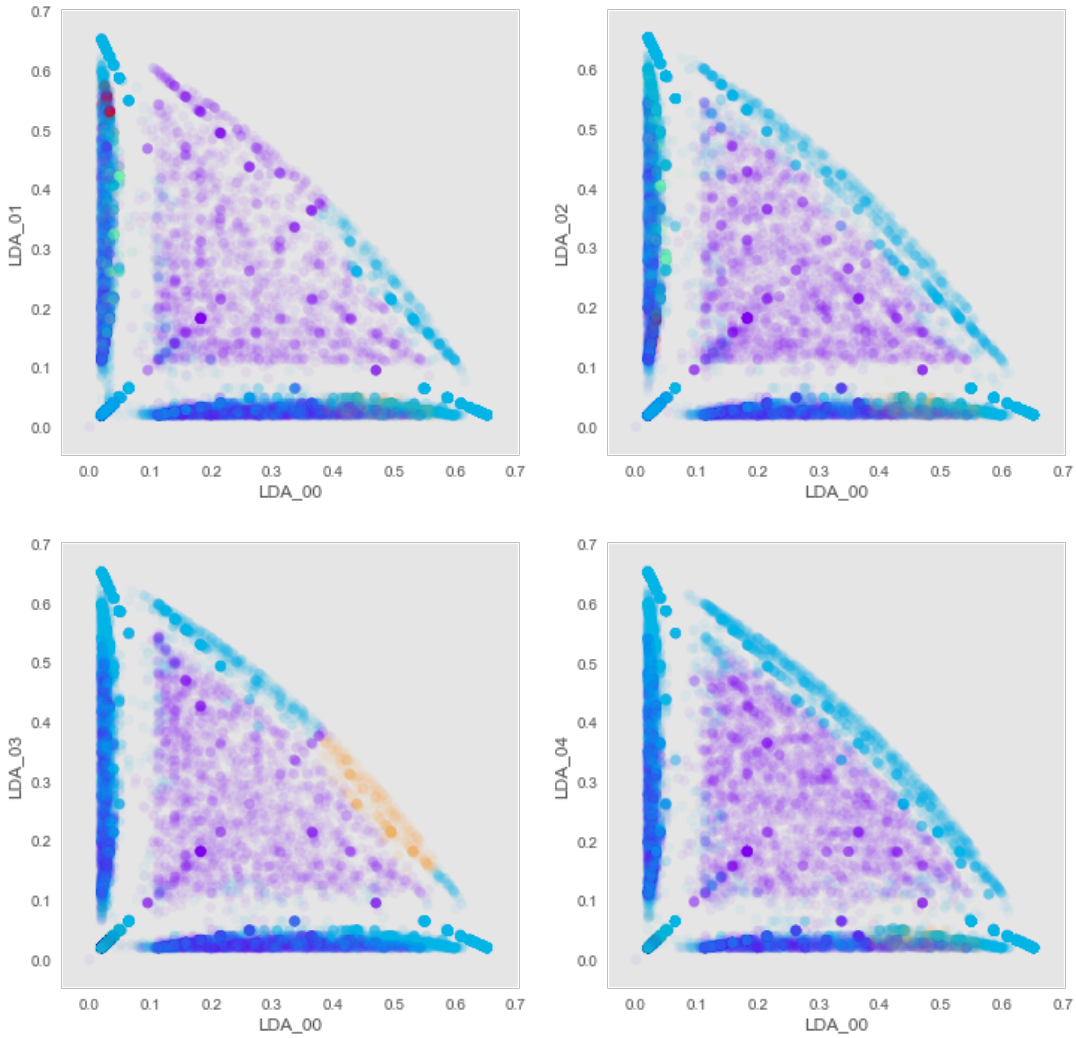

```
eps, min_pts, nclusters =  0.07 150 4
silhouette =  -0.148342782135
```




```
eps, min_pts, nclusters =  0.07 170 4
silhouette =  -0.152319241175
```




```
eps, min_pts, nclusters =  0.07 190 5
silhouette =  -0.242241225481
```




```
CPU times: user 6min 49s, sys: 10.8 s, total: 7min
Wall time: 6min 45s
Parser   : 124 ms
```

In [63]:
```
dbscan_tbl
```

Out[63]:

	model_name	n_clusters	epsilon	min_points	inertia	silhouette	process_time
1	DBScan - LDA features	153	0.02	10	0	-0.085596	5.4238
1	DBScan - LDA features	60	0.02	30	0	0.031906	4.1262
1	DBScan - LDA features	56	0.02	50	0	0.101427	3.9388
1	DBScan - LDA features	48	0.02	70	0	0.077119	3.9319
1	DBScan - LDA features	41	0.02	90	0	0.036830	3.7415
1	DBScan - LDA features	33	0.02	110	0	0.043183	3.6792
1	DBScan - LDA features	32	0.02	130	0	0.025074	3.6319
1	DBScan - LDA features	27	0.02	150	0	-0.012065	3.5670
1	DBScan - LDA features	19	0.02	170	0	-0.059899	3.4972
1	DBScan - LDA features	17	0.02	190	0	-0.071164	3.4778
1	DBScan - LDA features	124	0.03	10	0	-0.210358	5.2265
1	DBScan - LDA features	37	0.03	30	0	-0.021373	4.1868
1	DBScan - LDA features	27	0.03	50	0	0.145459	3.9858
1	DBScan - LDA features	29	0.03	70	0	0.138461	4.0710
1	DBScan - LDA features	32	0.03	90	0	0.162129	3.9581
1	DBScan - LDA features	28	0.03	110	0	0.146651	3.9046
1	DBScan - LDA features	27	0.03	130	0	0.170528	3.9858
1	DBScan - LDA features	24	0.03	150	0	0.118947	3.8420
1	DBScan - LDA features	19	0.03	170	0	0.109055	4.0999
1	DBScan - LDA features	18	0.03	190	0	0.079039	3.8014
1	DBScan - LDA features	13	0.05	10	0	-0.554975	4.0695

In [60]:
```
# ... -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
# ... - plot metrics across models for comparison
# ... -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-

plt.figure(figsize=(16, 6));

# ... silhouette values

plt.subplot(131);
plt.scatter(dbscan_tbl['min_points'],
            dbscan_tbl['silhouette'],
            s = 40,
            linewidths = 1.0,
            marker = '^',
            edgecolors = 'black',
            alpha = 0.90);

plt.plot(dbscan_tbl['min_points'],
         dbscan_tbl['silhouette'])

plt.xlabel('min_points'), plt.ylabel('silhouette');
plt.grid();

# ... inertia values

plt.subplot(132);
plt.scatter(dbscan_tbl['min_points'],
            dbscan_tbl['n_clusters'],
            s = 40,
            linewidths = 1.0,
            marker = '^',
            edgecolors = 'black',
            alpha = 0.90);

plt.plot(dbscan_tbl['min_points'],
         dbscan_tbl['n_clusters'])

plt.xlabel('min_points'), plt.ylabel('n_clusters');
plt.grid();

# ... process time

plt.subplot(133);
plt.scatter(dbscan_tbl['min_points'],
            dbscan_tbl['process_time'],
            s = 40,
            linewidths = 1.0,
            marker = '^',
            edgecolors = 'black',
            alpha = 0.90);

#plt.plot(dbscan_tbl['n_clusters'],
#         dbscan_tbl['process_time'])

plt.xlabel('min_points'), plt.ylabel('process_time');
```

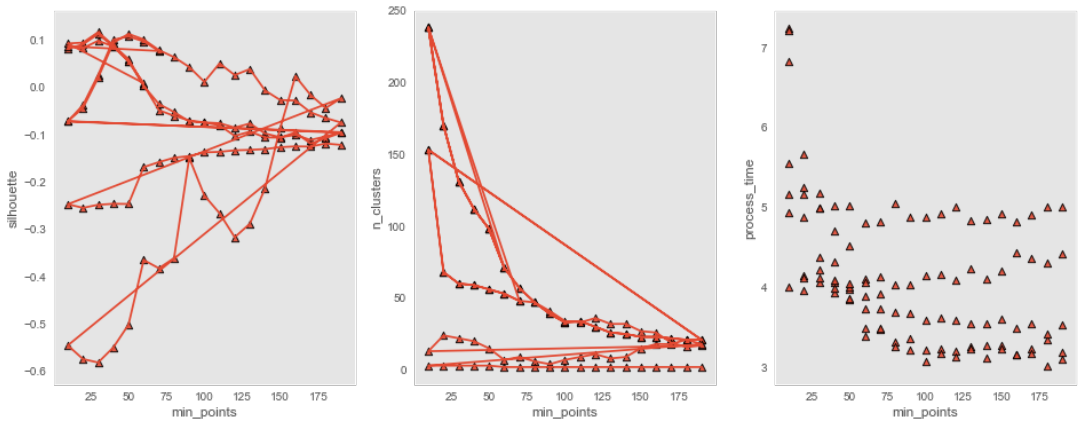


Table of Contents

Spectral Clustering

Spectral Clustering - 5 selected features

In [80]:
```
# set required variables for model comparison

spc_tbl = pd.DataFrame(columns = [
    'model_name',
    'n_clusters',
    'inertia',
    'silhouette',
    'process_time'])

i_index = []
i_index = 0

# preparation for cross validation and model comparison, each classifier is appended once mode
l is fit

models = []
```

In [81]:
```
from sklearn.cluster import SpectralClustering

# If a string, this may be one of
#  ‘nearest_neighbors’, ‘precomputed’, ‘rbf’
#  or one of the kernels supported by sklearn.metrics.pairwise_kernels

for n_clstr in range(2, 12):

    tic = time.clock()

    print ("n_clusters = ", n_clstr)

    X1 = df_cluster[['data_channel_n',
                     'ln_n_tokens_content',
                     'ln_num_hrefs',
                     'ln_num_imgs',
                     'ln_num_videos',]]

    X1 = X1.sample(frac = 0.1)

    spc = SpectralClustering(n_clusters = n_clstr,
                             affinity = 'nearest_neighbors')

    spc_labels = spc.fit_predict(X1)

    spc_silhouette = metrics.silhouette_score(X1,
                                              spc_labels,
                                              metric = 'euclidean',
                                              sample_size = 10000)
    print ("silhouette = ", spc_silhouette)

    toc =  time.clock()
# ... -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
# ... - save statistics for model comparison
# ... -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-

    exe_time = '{0:.4f}'.format(toc-tic)

    raw_data = {
    'model_name' : 'spc - LDA features',
    'n_clusters' : n_clstr,
    'inertia': 0,
    'silhouette': spc_silhouette,
    'process_time' : exe_time
    }

    df_tbl = pd.DataFrame(raw_data,
    columns = ['model_name', 'n_clusters', 'inertia', 'silhouette', 'process_time'],
    index = [i_index + 1])

    spc_tbl = spc_tbl.append(df_tbl)

# ... -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
# ... - make some plots of clusters
# ... -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
```

```
n_clusters =  2

/home/mcdevitt/anaconda3/lib/python3.6/site-packages/sklearn/manifold/spectral_embedding_.p
y:234: UserWarning: Graph is not fully connected, spectral embedding may not work as expect
ed.
  warnings.warn("Graph is not fully connected, spectral embedding"

silhouette =  0.513948316463
```

Out[81]: <matplotlib.figure.Figure at 0x7f4b925cd240>

Out[81]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b77124780>

Out[81]: <matplotlib.collections.PathCollection at 0x7f4b803ecd30>

Out[81]: (<matplotlib.text.Text at 0x7f4b77307dd8>,
 <matplotlib.text.Text at 0x7f4b91e4fac8>)

Out[81]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b9014a9e8>

Out[81]: <matplotlib.collections.PathCollection at 0x7f4b777dc940>

Out[81]: (<matplotlib.text.Text at 0x7f4b91e93cf8>,
 <matplotlib.text.Text at 0x7f4b805a39b0>)

Out[81]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b776da550>

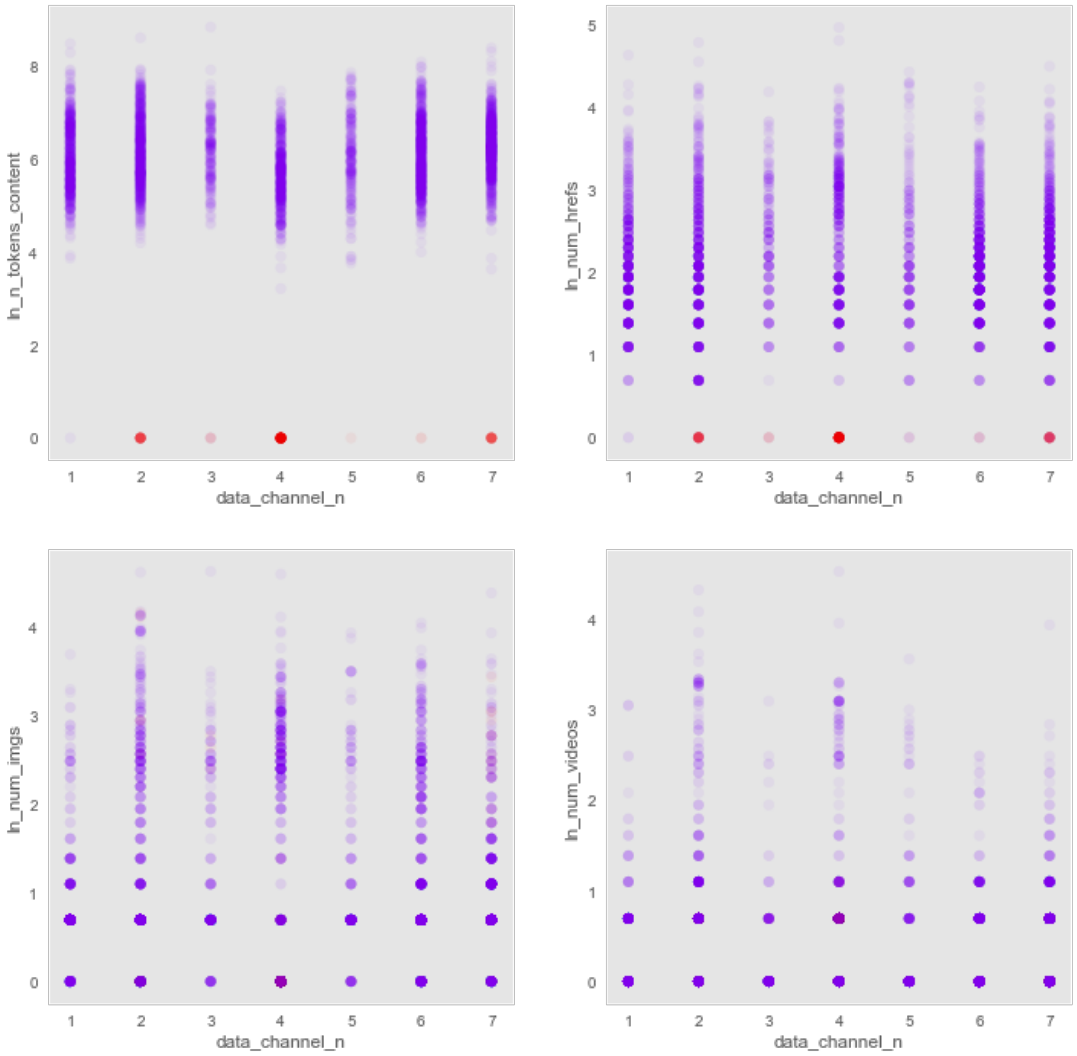
Out[81]: <matplotlib.collections.PathCollection at 0x7f4b80166828>

Out[81]: (<matplotlib.text.Text at 0x7f4b77762dd8>,
 <matplotlib.text.Text at 0x7f4b91e2db38>)

Out[81]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b80493c50>

Out[81]: <matplotlib.collections.PathCollection at 0x7f4b773e3470>

Out[81]: (<matplotlib.text.Text at 0x7f4b8029f828>,
 <matplotlib.text.Text at 0x7f4b9002df60>)




```
n_clusters =  3
silhouette =  0.469157506079
```

Out[81]: `<matplotlib.figure.Figure at 0x7f4b925cd128>`

Out[81]: `<matplotlib.axes._subplots.AxesSubplot at 0x7f4b77574ac8>`

Out[81]: `<matplotlib.collections.PathCollection at 0x7f4b91d32cc0>`

Out[81]:
```
(<matplotlib.text.Text at 0x7f4b92bbc518>,
 <matplotlib.text.Text at 0x7f4b9058f4a8>)
```

Out[81]: `<matplotlib.axes._subplots.AxesSubplot at 0x7f4b776125c0>`

Out[81]: `<matplotlib.collections.PathCollection at 0x7f4b8016d1d0>`

Out[81]:
```
(<matplotlib.text.Text at 0x7f4b7743ff28>,
 <matplotlib.text.Text at 0x7f4b805575f8>)
```

Out[81]: `<matplotlib.axes._subplots.AxesSubplot at 0x7f4b92a1c320>`

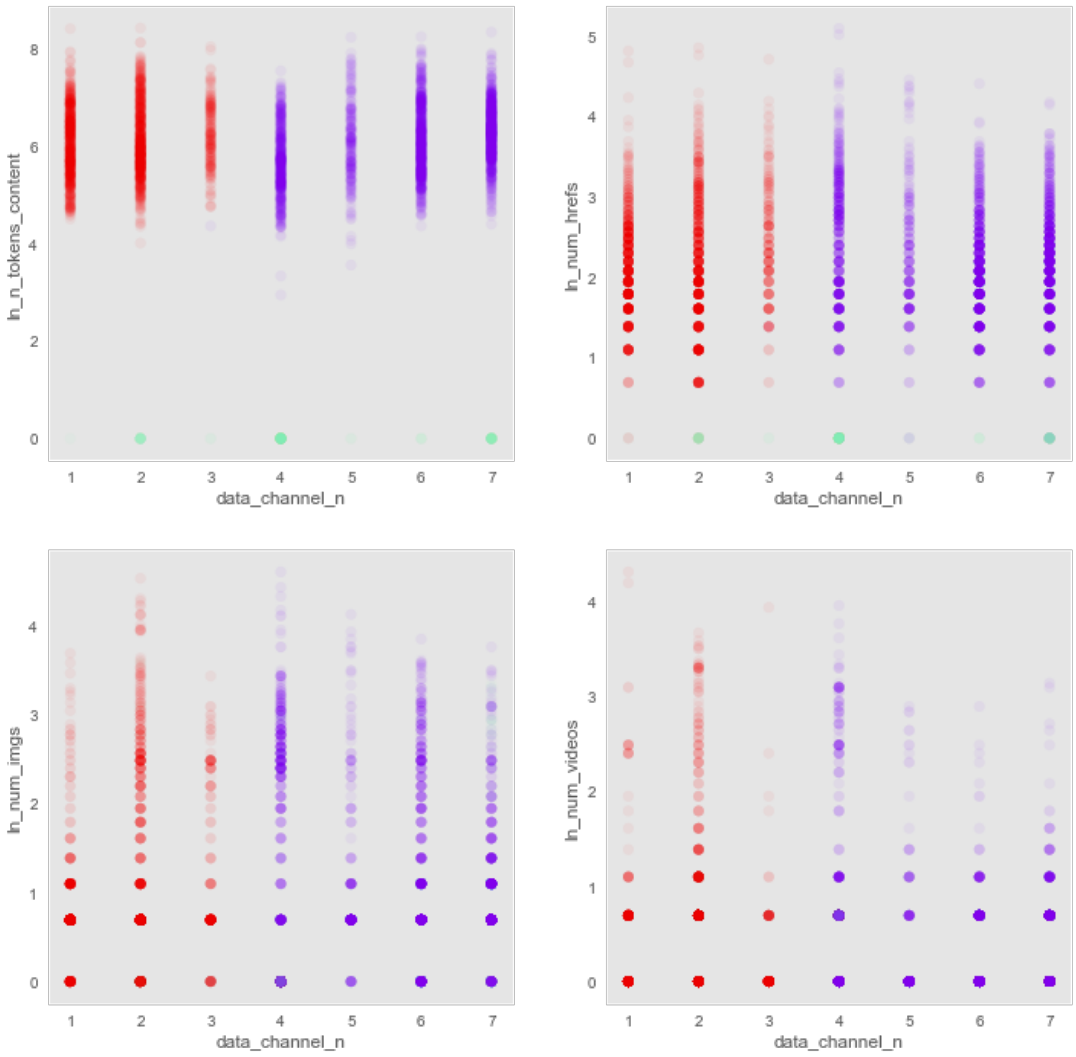
Out[81]: `<matplotlib.collections.PathCollection at 0x7f4b7772b358>`

Out[81]: (<matplotlib.text.Text at 0x7f4b774dec50>,
<matplotlib.text.Text at 0x7f4b900d4f98>)

Out[81]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b903016d8>

Out[81]: <matplotlib.collections.PathCollection at 0x7f4b91bf29e8>

Out[81]: (<matplotlib.text.Text at 0x7f4b91e67048>,
<matplotlib.text.Text at 0x7f4b91e67668>)




```
n_clusters =  4
silhouette =  0.277642447465
```

Out[81]: <matplotlib.figure.Figure at 0x7f4b905339e8>

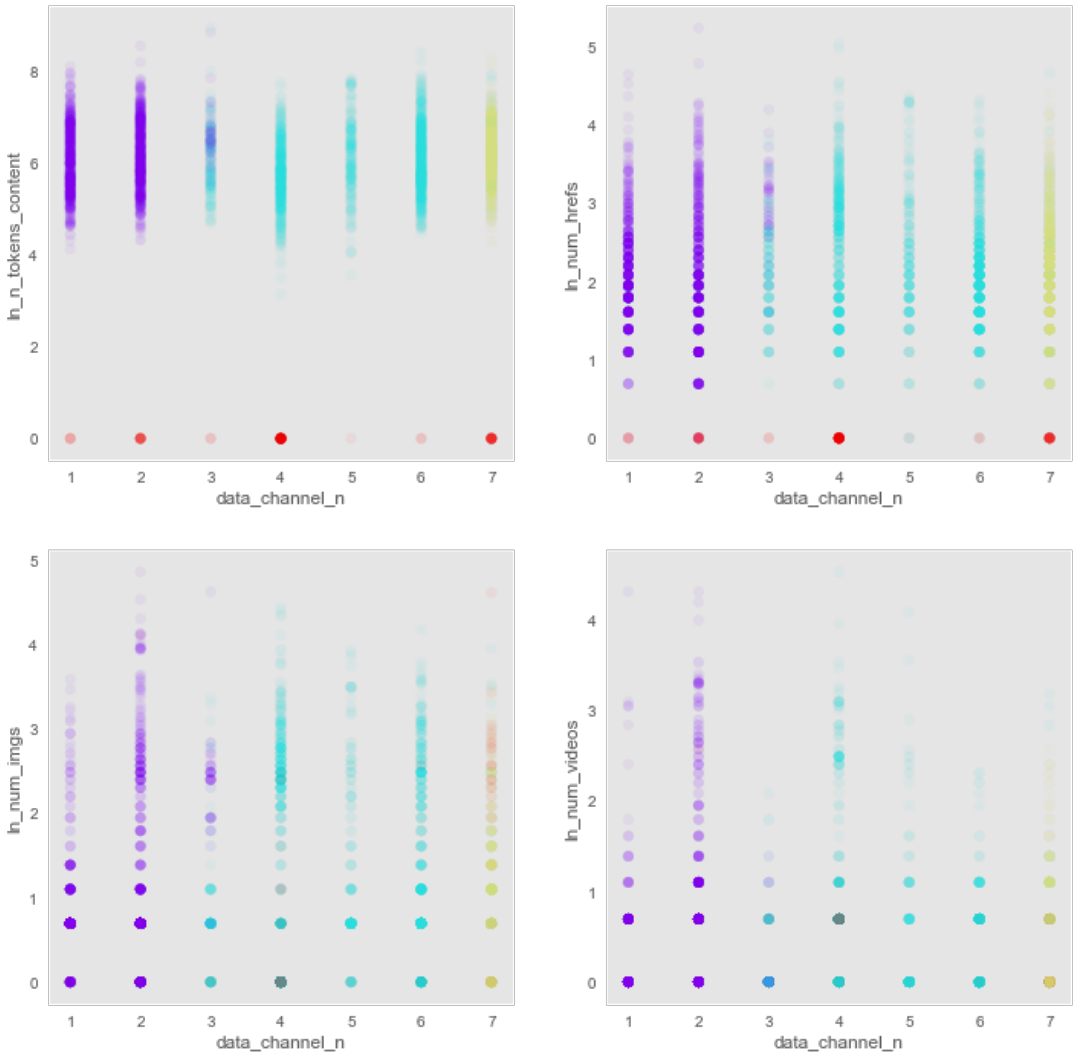
Out[81]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b7748a780>

Out[81]: <matplotlib.collections.PathCollection at 0x7f4b77771cc0>

Out[81]: (<matplotlib.text.Text at 0x7f4b777e4048>,
<matplotlib.text.Text at 0x7f4b91bd9400>)

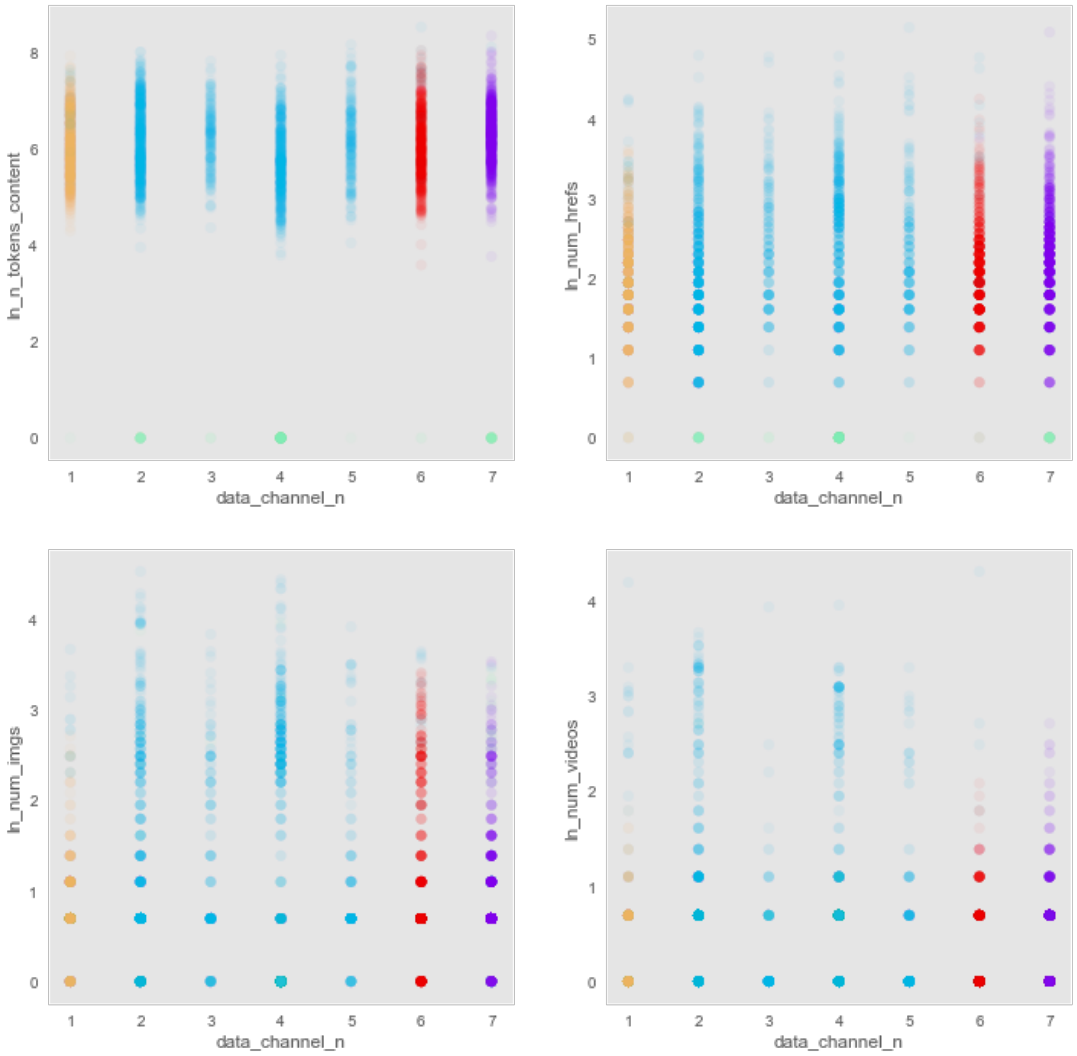
Out[81]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b775d6518>

```
Out[81]: <matplotlib.collections.PathCollection at 0x7f4b92c60588>

Out[81]: (<matplotlib.text.Text at 0x7f4b775d88d0>,
          <matplotlib.text.Text at 0x7f4b773c2b00>)

Out[81]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b92c60ba8>

Out[81]: <matplotlib.collections.PathCollection at 0x7f4b92cc5a90>

Out[81]: (<matplotlib.text.Text at 0x7f4b91f520b8>,
          <matplotlib.text.Text at 0x7f4b91f55438>)

Out[81]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b91f557f0>

Out[81]: <matplotlib.collections.PathCollection at 0x7f4b91b7b208>

Out[81]: (<matplotlib.text.Text at 0x7f4b92cc5eb8>,
          <matplotlib.text.Text at 0x7f4b92ce0748>)
```




```
n_clusters =  5
silhouette =  0.137926081974

Out[81]: <matplotlib.figure.Figure at 0x7f4b77325a58>
```

```
Out[81]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b804f32b0>

Out[81]: <matplotlib.collections.PathCollection at 0x7f4b91bfc2e8>

Out[81]: (<matplotlib.text.Text at 0x7f4b91ea0e10>,
          <matplotlib.text.Text at 0x7f4b91dc70f0>)

Out[81]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b91cb5fd0>

Out[81]: <matplotlib.collections.PathCollection at 0x7f4b91dffb00>

Out[81]: (<matplotlib.text.Text at 0x7f4b91ccae48>,
          <matplotlib.text.Text at 0x7f4b91cb31d0>)

Out[81]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b91add278>

Out[81]: <matplotlib.collections.PathCollection at 0x7f4b91ac6fd0>

Out[81]: (<matplotlib.text.Text at 0x7f4b91adf630>,
          <matplotlib.text.Text at 0x7f4b91aeb860>)

Out[81]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b91afdb00>

Out[81]: <matplotlib.collections.PathCollection at 0x7f4b92cccda0>

Out[81]: (<matplotlib.text.Text at 0x7f4b91c98588>,
          <matplotlib.text.Text at 0x7f4b91dc37f0>)
```




```
n_clusters =  6
silhouette =  0.166255692316
```

Out[81]: `<matplotlib.figure.Figure at 0x7f4b7718a4a8>`

Out[81]: `<matplotlib.axes._subplots.AxesSubplot at 0x7f4b91dc7048>`

Out[81]: `<matplotlib.collections.PathCollection at 0x7f4b7760e128>`

Out[81]:
```
(<matplotlib.text.Text at 0x7f4b91ba94a8>,
 <matplotlib.text.Text at 0x7f4b91bc7cf8>)
```

Out[81]: `<matplotlib.axes._subplots.AxesSubplot at 0x7f4b804172e8>`

Out[81]: `<matplotlib.collections.PathCollection at 0x7f4b900d4048>`

Out[81]:
```
(<matplotlib.text.Text at 0x7f4b775a4550>,
 <matplotlib.text.Text at 0x7f4b92cc6940>)
```

Out[81]: `<matplotlib.axes._subplots.AxesSubplot at 0x7f4b92c58d68>`

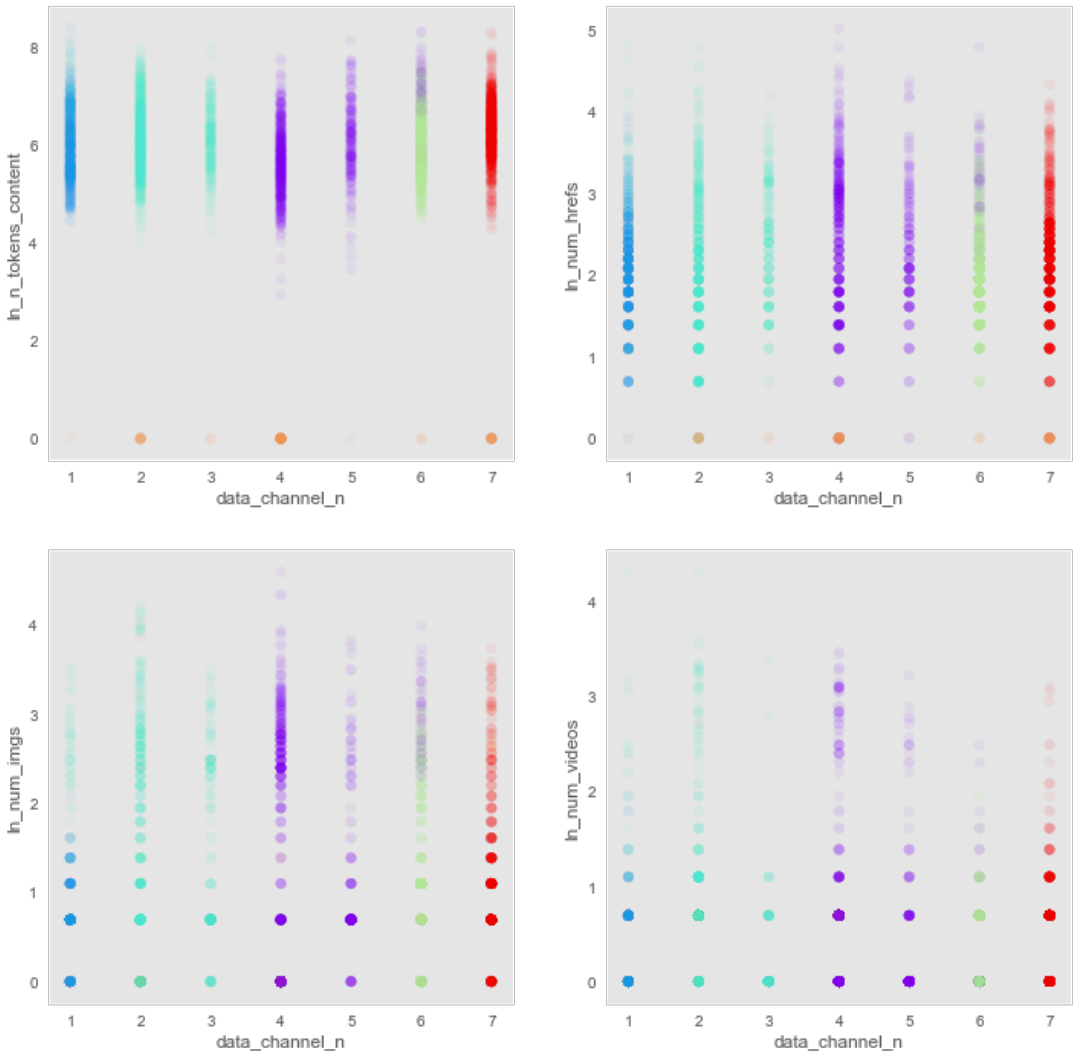
Out[81]: `<matplotlib.collections.PathCollection at 0x7f4b805a0278>`

Out[81]: (<matplotlib.text.Text at 0x7f4b9037a630>,
<matplotlib.text.Text at 0x7f4b91f5d198>)

Out[81]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b7731df98>

Out[81]: <matplotlib.collections.PathCollection at 0x7f4b92c47668>

Out[81]: (<matplotlib.text.Text at 0x7f4b8002ee80>,
<matplotlib.text.Text at 0x7f4b775b9470>)



n_clusters = 7
silhouette = 0.12099864669

Out[81]: <matplotlib.figure.Figure at 0x7f4b91ec15c0>

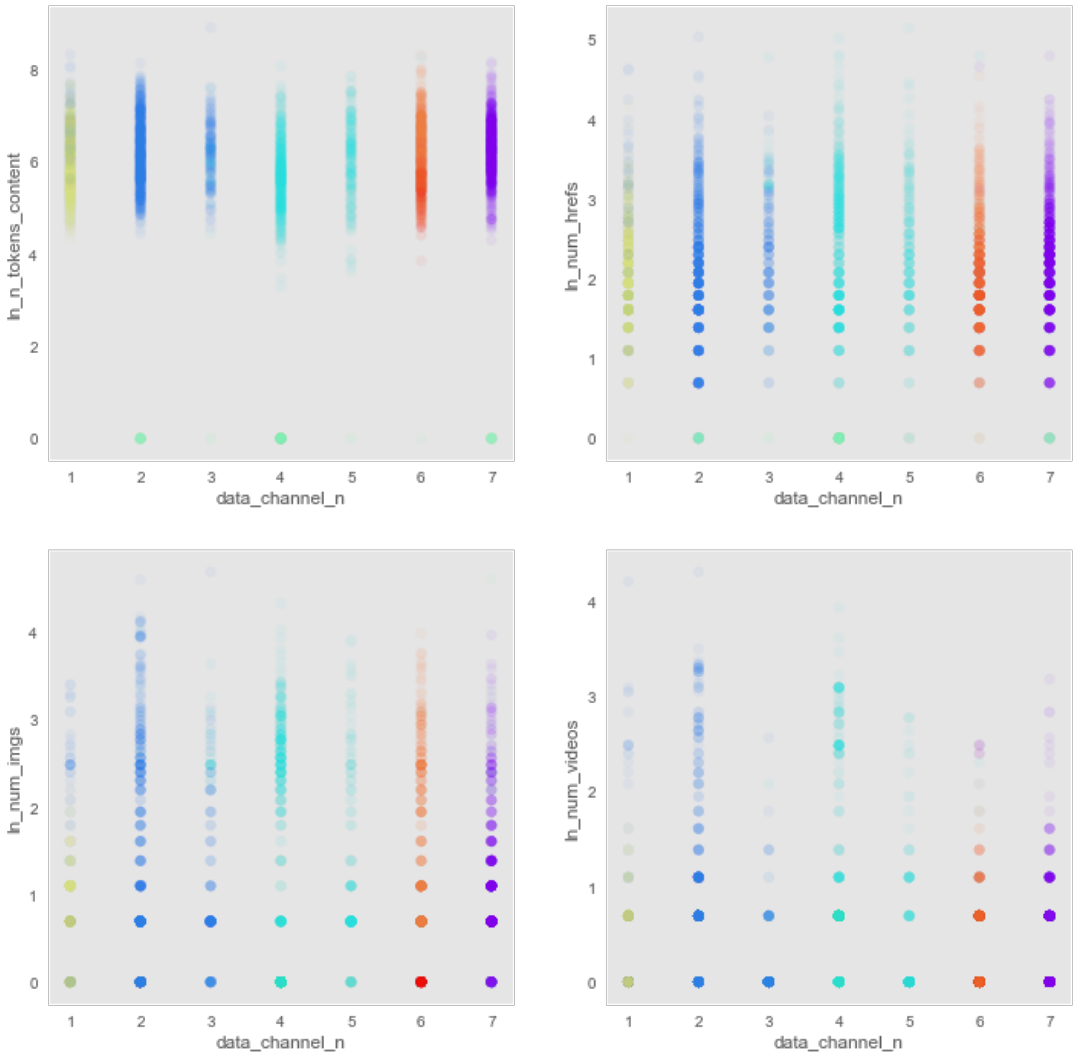
Out[81]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b91b94be0>

Out[81]: <matplotlib.collections.PathCollection at 0x7f4b7739b470>

Out[81]: (<matplotlib.text.Text at 0x7f4b91d71c88>,
<matplotlib.text.Text at 0x7f4b926079b0>)

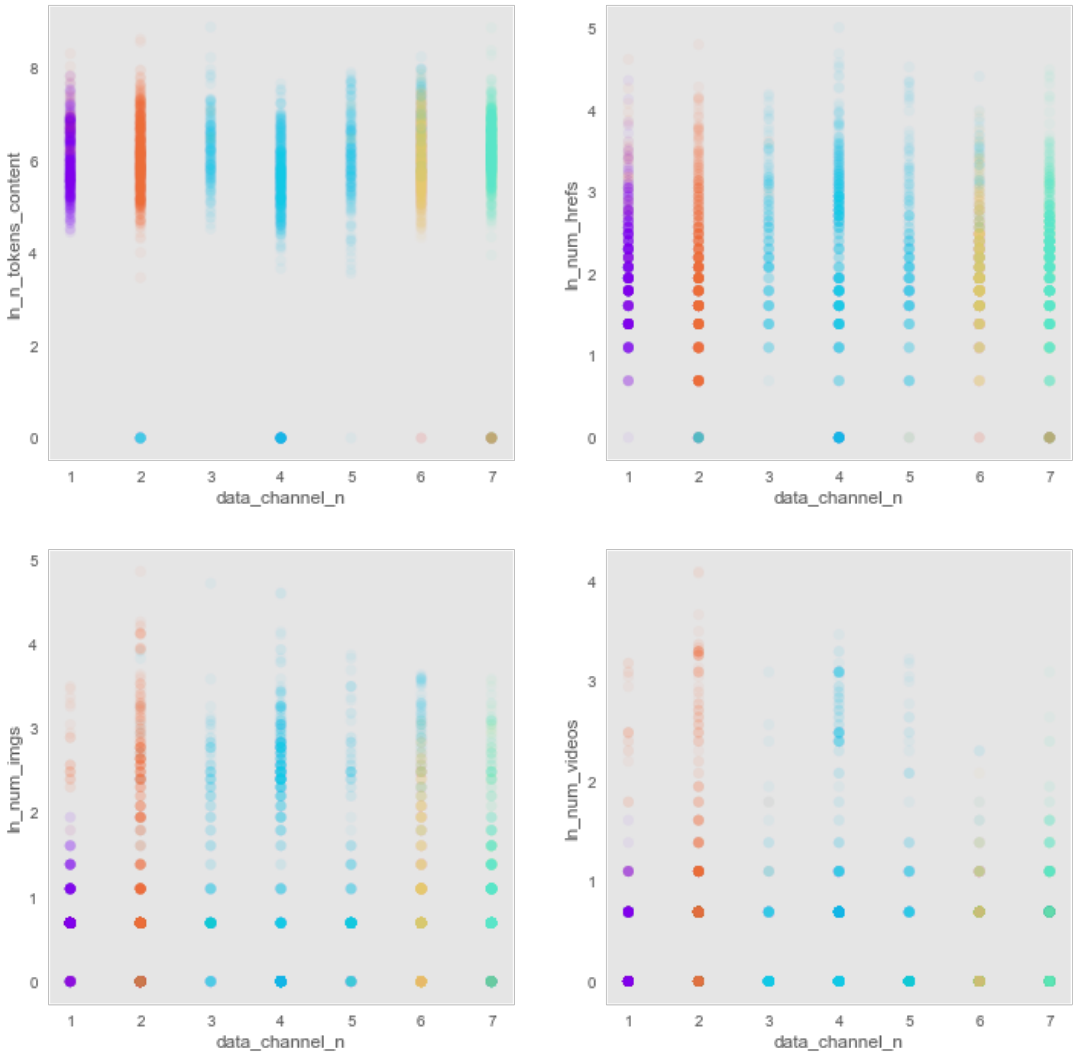
Out[81]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b802d3550>

```
Out[81]: <matplotlib.collections.PathCollection at 0x7f4b80238748>

Out[81]: (<matplotlib.text.Text at 0x7f4b777625c0>,
          <matplotlib.text.Text at 0x7f4b776f27f0>)

Out[81]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b8045e4e0>

Out[81]: <matplotlib.collections.PathCollection at 0x7f4b92c46dd8>

Out[81]: (<matplotlib.text.Text at 0x7f4b770f64e0>,
          <matplotlib.text.Text at 0x7f4b91e4f240>)

Out[81]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b7753d358>

Out[81]: <matplotlib.collections.PathCollection at 0x7f4b7742f748>

Out[81]: (<matplotlib.text.Text at 0x7f4b91e4f860>,
          <matplotlib.text.Text at 0x7f4b91b8cc88>)
```




```
n_clusters =  8
silhouette =  0.104695361366

Out[81]: <matplotlib.figure.Figure at 0x7f4b904df048>
```

```
Out[81]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b91ba9d68>

Out[81]: <matplotlib.collections.PathCollection at 0x7f4b91b76f98>

Out[81]: (<matplotlib.text.Text at 0x7f4b926aad68>,
          <matplotlib.text.Text at 0x7f4b90046fd0>)

Out[81]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b80027f28>

Out[81]: <matplotlib.collections.PathCollection at 0x7f4b8029db70>

Out[81]: (<matplotlib.text.Text at 0x7f4b80190b38>,
          <matplotlib.text.Text at 0x7f4b804f3630>)

Out[81]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b91e24320>

Out[81]: <matplotlib.collections.PathCollection at 0x7f4b804d8e80>

Out[81]: (<matplotlib.text.Text at 0x7f4b8040eeb8>,
          <matplotlib.text.Text at 0x7f4b91f25d30>)

Out[81]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b774b6080>

Out[81]: <matplotlib.collections.PathCollection at 0x7f4b92cc0ba8>

Out[81]: (<matplotlib.text.Text at 0x7f4b91f251d0>,
          <matplotlib.text.Text at 0x7f4b804d1908>)
```




```
n_clusters =  9
silhouette =  0.143887177055
```

Out[81]: `<matplotlib.figure.Figure at 0x7f4b772e5358>`

Out[81]: `<matplotlib.axes._subplots.AxesSubplot at 0x7f4b770e4278>`

Out[81]: `<matplotlib.collections.PathCollection at 0x7f4b77414d68>`

Out[81]:
```
(<matplotlib.text.Text at 0x7f4b804256d8>,
 <matplotlib.text.Text at 0x7f4b92c32a58>)
```

Out[81]: `<matplotlib.axes._subplots.AxesSubplot at 0x7f4b9032b7b8>`

Out[81]: `<matplotlib.collections.PathCollection at 0x7f4b7743fb00>`

Out[81]:
```
(<matplotlib.text.Text at 0x7f4b77385cf8>,
 <matplotlib.text.Text at 0x7f4b771562b0>)
```

Out[81]: `<matplotlib.axes._subplots.AxesSubplot at 0x7f4b80238c18>`

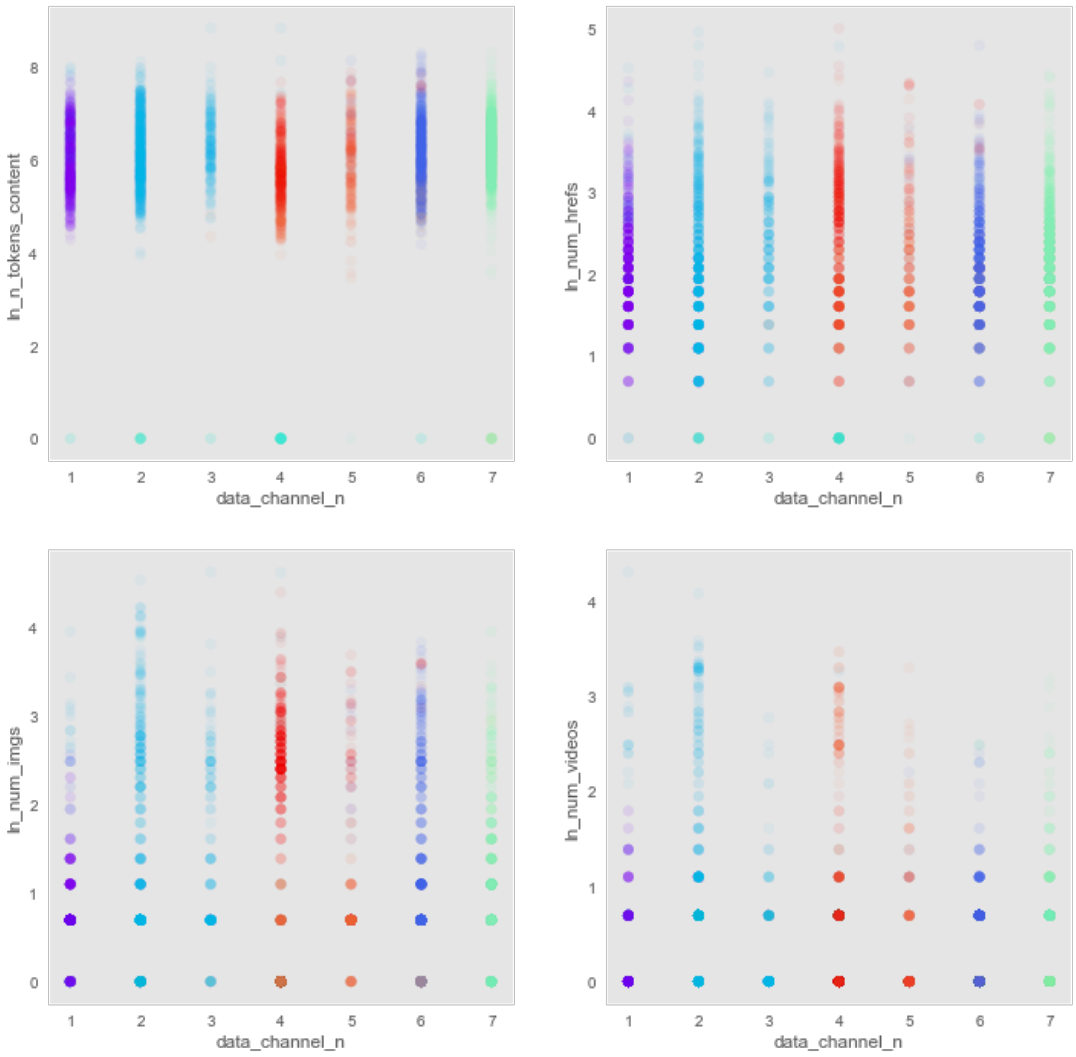
Out[81]: `<matplotlib.collections.PathCollection at 0x7f4b8056dd68>`

Out[81]: (<matplotlib.text.Text at 0x7f4b773f4278>,
<matplotlib.text.Text at 0x7f4b92bb82b0>)

Out[81]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b805a32b0>

Out[81]: <matplotlib.collections.PathCollection at 0x7f4b77311668>

Out[81]: (<matplotlib.text.Text at 0x7f4b92c50080>,
<matplotlib.text.Text at 0x7f4b92c507f0>)



n_clusters = 10
silhouette = 0.00688832942488

Out[81]: <matplotlib.figure.Figure at 0x7f4b775360b8>

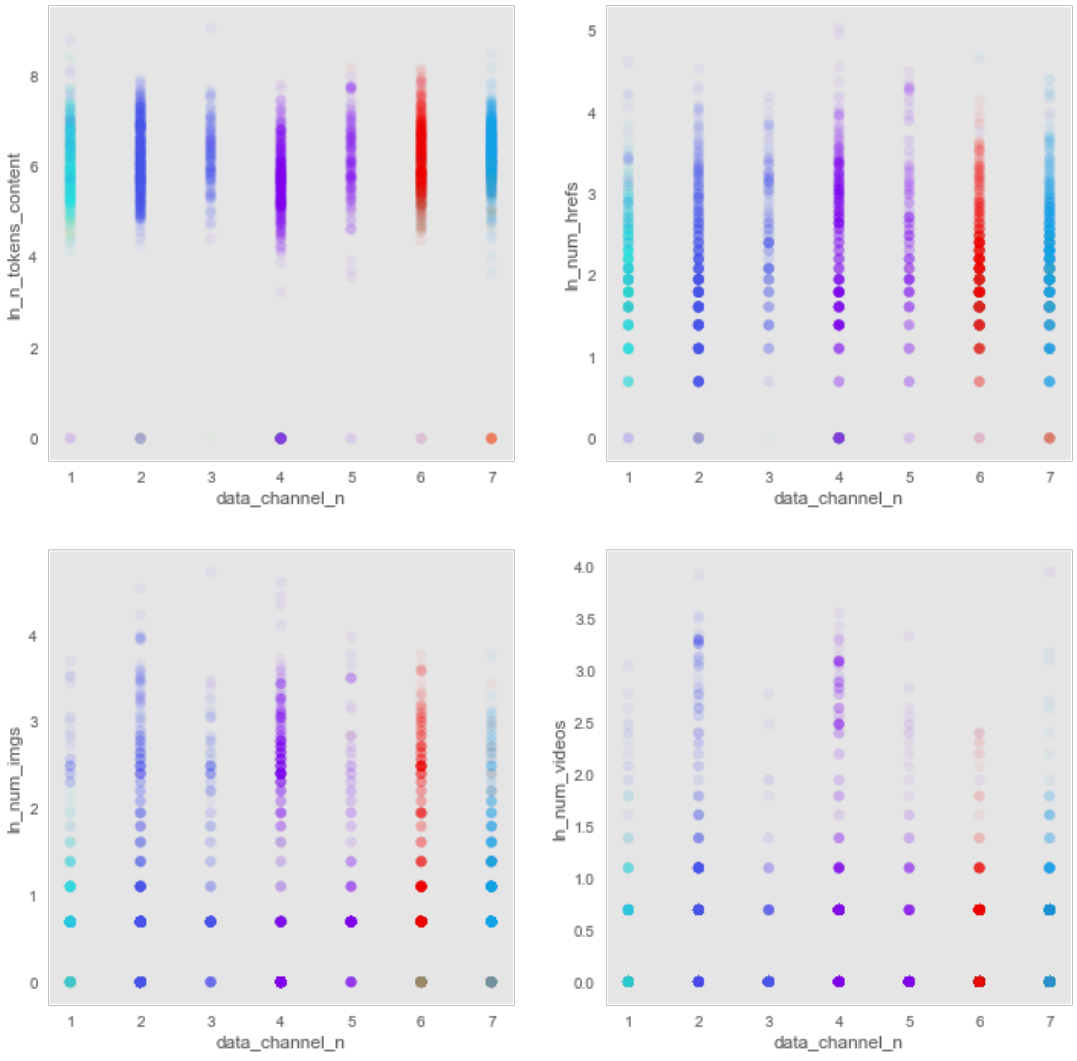
Out[81]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b905b5cc0>

Out[81]: <matplotlib.collections.PathCollection at 0x7f4b803a80b8>

Out[81]: (<matplotlib.text.Text at 0x7f4b802d4588>,
<matplotlib.text.Text at 0x7f4b904299e8>)

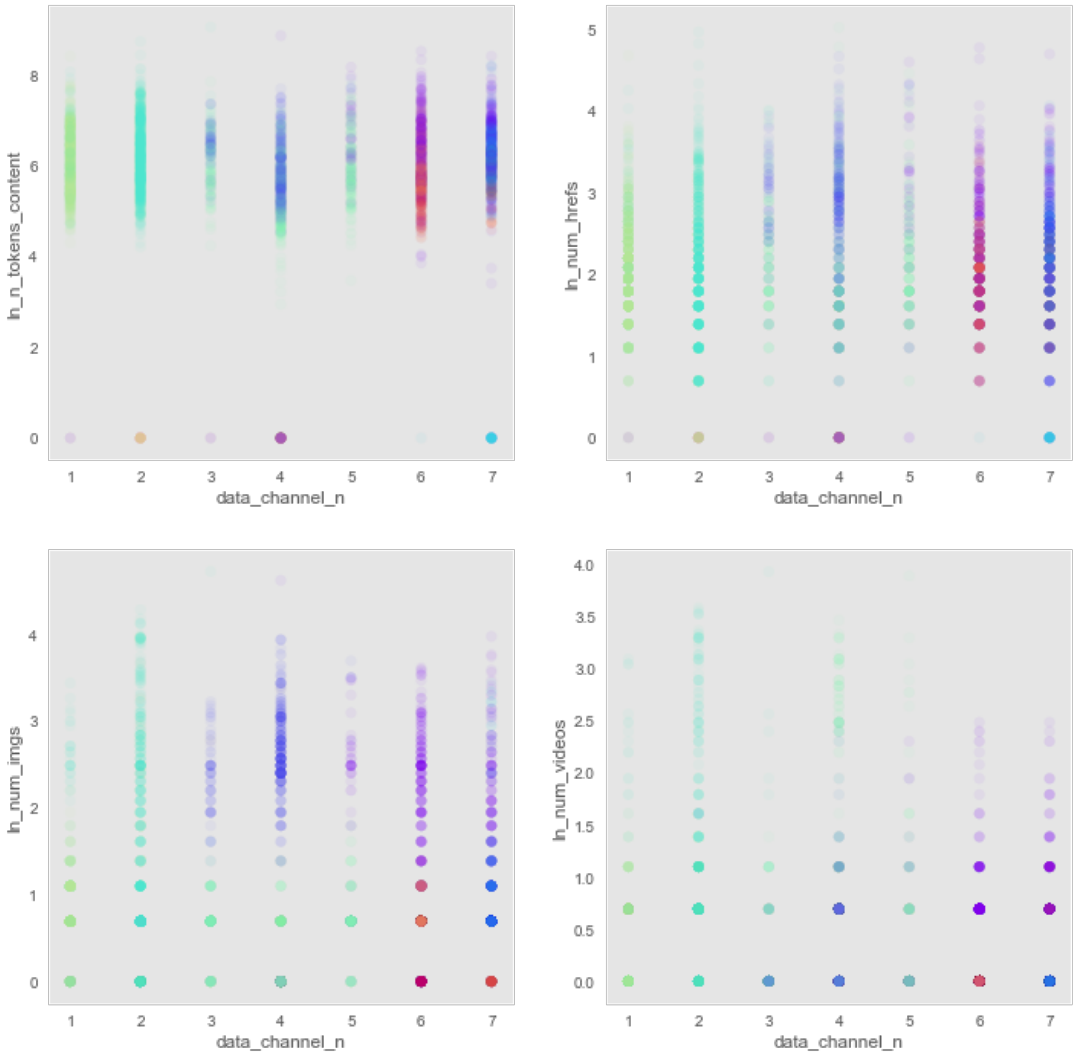
Out[81]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4bc945dac8>

```
Out[81]: <matplotlib.collections.PathCollection at 0x7f4b924ff9e8>

Out[81]: (<matplotlib.text.Text at 0x7f4b775b9ba8>,
          <matplotlib.text.Text at 0x7f4b7772b780>)

Out[81]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b92535c50>

Out[81]: <matplotlib.collections.PathCollection at 0x7f4b900a5710>

Out[81]: (<matplotlib.text.Text at 0x7f4b900a4940>,
          <matplotlib.text.Text at 0x7f4b805da748>)

Out[81]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b90046978>

Out[81]: <matplotlib.collections.PathCollection at 0x7f4b90432b70>

Out[81]: (<matplotlib.text.Text at 0x7f4b8029fcc0>,
          <matplotlib.text.Text at 0x7f4b91b9fcf8>)
```




```
n_clusters =  11
silhouette =  0.107343619097

Out[81]: <matplotlib.figure.Figure at 0x7f4b770badd8>
```

```
Out[81]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b770f6a58>

Out[81]: <matplotlib.collections.PathCollection at 0x7f4b92bbd1d0>

Out[81]: (<matplotlib.text.Text at 0x7f4b92c47cf8>,
          <matplotlib.text.Text at 0x7f4b92645080>)

Out[81]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b92bbda58>

Out[81]: <matplotlib.collections.PathCollection at 0x7f4b92cadac8>

Out[81]: (<matplotlib.text.Text at 0x7f4b92bbae10>,
          <matplotlib.text.Text at 0x7f4b91f7f080>)

Out[81]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b92ccf4a8>

Out[81]: <matplotlib.collections.PathCollection at 0x7f4b91eb9f60>

Out[81]: (<matplotlib.text.Text at 0x7f4b92cc35f8>,
          <matplotlib.text.Text at 0x7f4b91ee1828>)

Out[81]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4b91efee10>

Out[81]: <matplotlib.collections.PathCollection at 0x7f4b91bbbdd8>

Out[81]: (<matplotlib.text.Text at 0x7f4b91ed5470>,
          <matplotlib.text.Text at 0x7f4b91ed5cf8>)
```

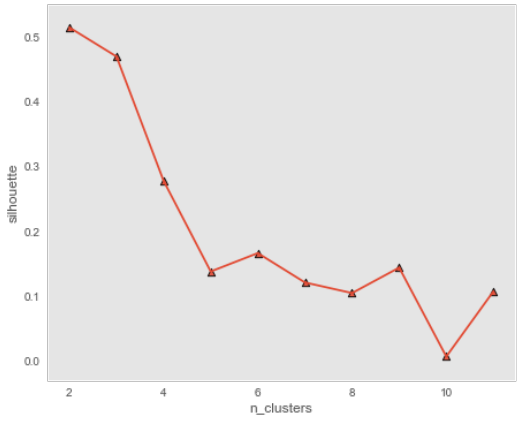


Out[81]:

	model_name	n_clusters	inertia	silhouette	process_time
1	spc - LDA features	2	0	0.513948	3.0362
1	spc - LDA features	3	0	0.469158	2.2964
1	spc - LDA features	4	0	0.277642	2.5837
1	spc - LDA features	5	0	0.137926	2.4575
1	spc - LDA features	6	0	0.166256	2.3021
1	spc - LDA features	7	0	0.120999	2.6126
1	spc - LDA features	8	0	0.104695	2.4773
1	spc - LDA features	9	0	0.143887	2.5353
1	spc - LDA features	10	0	0.006888	2.4749
1	spc - LDA features	11	0	0.107344	2.7446

In [82]:

```
# ... -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
# ... - plot metrics across models for comparison
# ... -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-

plt.figure(figsize=(16, 6));

# ... silhouette values

plt.subplot(121);
plt.scatter(spc_tbl['n_clusters'],
            spc_tbl['silhouette'],
            s = 40,
            linewidths = 1.0,
            marker = '^',
            edgecolors = 'black',
            alpha = 0.90);

plt.plot(spc_tbl['n_clusters'],
         spc_tbl['silhouette'])

plt.xlabel('n_clusters'), plt.ylabel('silhouette');
plt.grid();


# ... process time

plt.subplot(133);
plt.scatter(spc_tbl['n_clusters'],
            spc_tbl['process_time'],
            s = 40,
            linewidths = 1.0,
            marker = '^',
            edgecolors = 'black',
            alpha = 0.90);

#plt.plot(spc_tbl['n_clusters'],
#         spc_tbl['process_time'])

plt.xlabel('n_clusters'), plt.ylabel('process_time');
plt.grid();


plt.show();
```



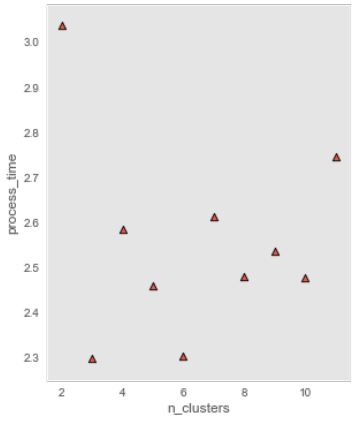


Table of Contents

end of file

In [ ]: