

customer_segments

December 14, 2015

1 Creating Customer Segments

In this project you, will analyze a dataset containing annual spending amounts for internal structure, to understand the variation in the different types of customers that a wholesale distributor interacts with.

Instructions:

- Run each code block below by pressing **Shift+Enter**, making sure to implement any steps marked with a TODO.
- Answer each question in the space provided by editing the blocks labeled “Answer:”.
- When you are done, submit the completed notebook (.ipynb) with all code blocks executed, as well as a .pdf version (File > Download as).

```
In [1]: # Import libraries: NumPy, pandas, matplotlib
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

# Tell iPython to include plots inline in the notebook
%matplotlib inline

# Read dataset
data = pd.read_csv("wholesale-customers.csv")
print "Dataset has {} rows, {} columns".format(*data.shape)
print data.head() # print the first 5 rows
```

Dataset has 440 rows, 6 columns

	Fresh	Milk	Grocery	Frozen	Detergents_Paper	Delicatessen
0	12669	9656	7561	214	2674	1338
1	7057	9810	9568	1762	3293	1776
2	6353	8808	7684	2405	3516	7844
3	13265	1196	4221	6404	507	1788
4	22615	5410	7198	3915	1777	5185

1.1 Feature Transformation

1) In this section you will be using PCA and ICA to start to understand the structure of the data. Before doing any computations, what do you think will show up in your computations? List one or two ideas for what might show up as the first PCA dimensions, or what type of vectors will show up as ICA dimensions.

Answer: - Because PCA is based on the covariance matrix, maybe only one item will pop out such as Fresh. This has a standard deviation compared to other items so this will probably dominate the first component. - ICA will show independant vectors that explains the original signal. It's a little bit hard to guess at first what will come out.

1.1.1 PCA

```
In [2]: # TODO: Apply PCA with the same number of dimensions as variables in the dataset
from sklearn.decomposition import PCA
pca = PCA(n_components=6)
pca.fit(data)

# Print the components and the amount of variance in the data contained in each dimension
print pd.DataFrame(
    pca.components_,
    columns = ['PC{}'.format(i) for i in range(1,7)],
    index = ['Fresh', 'Milk', 'Grocery', 'Frozen', 'Detergents_Paper', 'Delicatessen']
)
print pd.DataFrame(
    pca.explained_variance_ratio_.reshape((1,6)) ,
    columns = ['PC{}'.format(i) for i in range(1,7)],
    index = ['Variance']
)
```

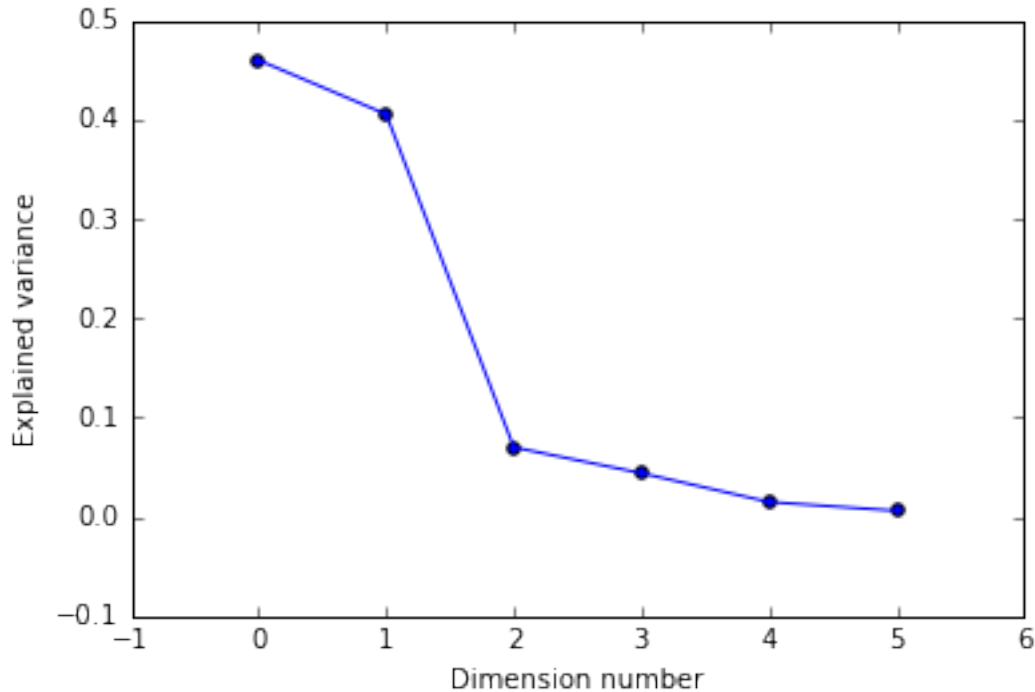
	PC1	PC2	PC3	PC4	PC5	PC6		
Fresh			-0.976537	-0.121184	-0.061540	-0.152365	0.007054	-0.068105
Milk			-0.110614	0.515802	0.764606	-0.018723	0.365351	0.057079
Grocery			-0.178557	0.509887	-0.275781	0.714200	-0.204410	0.283217
Frozen			-0.041876	-0.645640	0.375460	0.646292	0.149380	-0.020396
Detergents_Paper			0.015986	0.203236	-0.160292	0.220186	0.207930	-0.917077
Delicatessen			-0.015763	0.033492	0.410939	-0.013289	-0.871284	-0.265417
	PC1	PC2	PC3	PC4	PC5	PC6		
Variance	0.459614	0.405172	0.07003	0.044023	0.015022	0.006138		

1.2 2) How quickly does the variance drop off by dimension? If you were to use PCA on this dataset, how many dimensions would you choose for your analysis? Why?

Answer: - The variance drops off rapidly after the second dimension. With only two dimensions we have approximately 86% of variance. - To decide how many dimensions to keep we have to ask ourselves why we are doing PCA. Is it to reduce storage space, to reduce dimensionality or some other reason? The storage here is not a problem. We want keep as much information as possible while reducing the risk of falling in the curse of dimensionality. Thus, by reducing by three dimensions, from 6 to 3, we still keep 93% of the variance and we allow any kind of machine learning algorithm to better fit the space with the same sample size.

```
In [4]: plt.plot(pca.explained_variance_ratio_)
plt.scatter(xrange(6), pca.explained_variance_ratio_)
plt.ylabel("Explained variance")
plt.xlabel("Dimension number")
```

```
Out[4]: <matplotlib.text.Text at 0x102035090>
```



3) What do the dimensions seem to represent? How can you use this information?

Answer:

- Every component seems to represent a different kind of buyer.
 - For example PC1 is dominated by the negative value of Fresh. It's negatively correlated to everything else. This category is much more important for this group of customers. Meaning that when it's buying a lot of Fresh it's not buying much of anything else.
 - In PC2 Frozen and especially Fresh is negatively correlated with Milk, Grocery and Detergent.Paper. It could be seen as two buying habits: one with Fresh and Frozen and another with Milk, Grocery and Detergent.Paper.
- Combined together we see 6 different typical customers with their buying habits. Last one for example described mostly by the lack of “detergent paper” and “delicatessen” and the presence of grocery (but also keeping in mind that this last component only represents .6% of the variance)
- So we could combine items together in special offers or layout on the floor to maximise spending.

1.2.1 ICA

```
In [108]: # TODO: Fit an ICA model to the data
# Note: Adjust the data to have center at the origin first!
from sklearn.decomposition import FastICA
ica = FastICA(n_components=6)
S_ = ica.fit_transform(data)
A_ = ica.mixing_
print np.dot(S_, A_.T).shape
# Print the independent components
pd.DataFrame(
    ica.components_,
    index = ['PC{}'.format(i) for i in range(1,7)],
```

```
columns = ['Fresh', 'Milk', 'Grocery', 'Frozen', 'Detergents_Paper', 'Delicatessen']
)
```

(440, 6)

```
Out[108]:
```

	Fresh	Milk	Grocery	Frozen	Detergents_Paper	Delicatessen
PC1	-3.978395e-06	8.810875e-07	7.432629e-07	6.674419e-07	-2.325392e-06	
PC2	8.631979e-07	1.473903e-07	-7.759939e-07	-1.114811e-05	5.405627e-07	
PC3	1.602467e-07	9.796655e-06	-5.981916e-06	-3.324834e-07	3.785341e-06	
PC4	-1.970741e-07	1.754937e-06	-7.266282e-06	-2.995833e-07	2.770327e-06	
PC5	2.750160e-07	-2.592235e-06	-1.147771e-05	1.492804e-06	2.800186e-05	
PC6	3.882090e-07	2.098617e-07	5.984854e-07	5.051525e-07	-5.176794e-07	

	Delicatessen
PC1	9.794714e-07
PC2	5.974442e-06
PC3	-5.993927e-06
PC4	1.856041e-06
PC5	5.727265e-06
PC6	-1.807449e-05

```
In [109]: # Just to put it on a more visible scale I'll normalise the components
print pd.DataFrame(
    ica.components_/np.linalg.norm(ica.components_, axis=1).reshape(6,1),
    index = ['PC{}'.format(i) for i in range(1,7)],
    columns = ['Fresh', 'Milk', 'Grocery', 'Frozen', 'Detergents_Paper', 'Delicatessen']
)
```

	Fresh	Milk	Grocery	Frozen	Detergents_Paper	Delicatessen
PC1	-0.812617	0.179969	0.151817	0.136330	-0.474979	0.200064
PC2	0.067896	0.011593	-0.061037	-0.876865	0.042518	0.469925
PC3	0.011873	0.725877	-0.443227	-0.024635	0.280473	-0.444117
PC4	-0.024054	0.214197	-0.886878	-0.036565	0.338129	0.226537
PC5	0.008887	-0.083766	-0.370892	0.048239	0.904856	0.185072
PC6	0.021443	0.011592	0.033058	0.027902	-0.028594	-0.998357

4) For each vector in the ICA decomposition, write a sentence or two explaining what sort of object or property it corresponds to. What could these components be used for?

Answer: ICA is trying to find independent signals in the original data. Something independent that, when mixed together, produce the data that we have. As suggested by Mitchell in this [post](#) this could be a deli counter in a grocery store or a bar attached to a restaurant. This could be an owner with two types of establishments and orders everything together. So let's look at every components and try to come up with an explanation for them.

Just a note, wholesale is selling in bulk. It could be to either regular households or businesses. I will sometimes provide an explanation for one or the other or both.

PC1 - Fresh and Detergents.Paper are notoriously missing and the rest is more or less equivalent. Career oriented people could fit the profile. They often eat outside home so they don't buy much Fresh and they don't have to clean as much. They probably even use the laundry mat instead of washing their own clothes.

PC2 - Dominated by the lack of Frozen and the presence of Delicatessen. This could be complementary to PC1 where the people tend to eat outside of home, this component could be the deli or sandwich place that this person goes to. It's mostly influenced by the presence of Delicatessen and a place that sells fresh products (Subway for example) would be a bigger consumer of this.

PC3 - Big focus on Milk. Ice cream shop that does their own products comes to mind. They would mostly need Milk and not the rest.

PC4 - Large negative value in Grocery indicate a group of health conscious people that avoids the "middle aisle", staying on the outside preferring fresh and raw ingredients.

PC5 - Detergent_Paper is largely dominant. This would be a group focused on cleaning. Laundry mat, house cleaning, janitors, ...

PC6 - A group of bachelor that doesn't like to cook and buys pre made food such as sandwiches.

1.3 Clustering

In this section you will choose either K Means clustering or Gaussian Mixed Models clustering, which implements expectation-maximization. Then you will sample elements from the clusters to understand their significance.

1.3.1 Choose a Cluster Type

5) What are the advantages of using K Means clustering or Gaussian Mixture Models?

Answer: - The advantage of K Means is - that *it scales well to large number of samples and has been used across a large range of application areas in many different fields* - *K-means will always converge, however this may be to a local minimum* 1 - Because of its simplicity it's easy to understand - Much easier to interpret the results (centroids) than more complicated algorithms

- The advantages of GMM
 - Instead of defining a hard limit to divide the space as with K-Means and Voronoi diagrams it allows every point to have a “membership” to every clusters
 - If needed a “no man’s land” can be defined. Meaning that your membership can be at least as strong as a certain threshold to declare you of a certain cluster
 - Gaussians are often well behaved: algorithmically friendly, differentiable, mean and variance is well understood.

6) Below is some starter code to help you visualize some cluster data. The visualization is based on [this demo](#) from the sklearn documentation.

```
In [7]: # Import clustering modules
        from sklearn.cluster import KMeans
        from sklearn.mixture import GMM
```

```
In [8]: # TODO: First we reduce the data to two dimensions using PCA to capture variation
        pca = PCA(n_components=2)
        reduced_data = pca.fit_transform(data)
        print reduced_data[:10] # print upto 10 elements
```

```
[[ -650.02212207  1585.51909007]
 [ 4426.80497937  4042.45150884]
 [ 4841.9987068   2578.762176  ]
 [ -990.34643689 -6279.80599663]
 [-10657.99873116 -2159.72581518]
 [ 2765.96159271 -959.87072713]
 [ 715.55089221  -2013.00226567]
 [ 4474.58366697  1429.49697204]
 [ 6712.09539718 -2205.90915598]
 [ 4823.63435407  13480.55920489]]
```

```
In [9]: # TODO: Implement your clustering algorithm here, and fit it to the reduced data for visualizat
        # The visualizer below assumes your clustering object is named 'clusters'
        clusters = GMM(n_components=4)
        clusters.fit(reduced_data)
        print clusters
```

```

GMM(covariance_type='diag', init_params='wmc', min_covar=0.001,
    n_components=4, n_init=1, n_iter=100, params='wmc', random_state=None,
    thresh=None, tol=0.001, verbose=0)

In [10]: # Plot the decision boundary by building a mesh grid to populate a graph.
x_min, x_max = reduced_data[:, 0].min() - 1, reduced_data[:, 0].max() + 1
y_min, y_max = reduced_data[:, 1].min() - 1, reduced_data[:, 1].max() + 1
hx = (x_max-x_min)/1000.
hy = (y_max-y_min)/1000.
xx, yy = np.meshgrid(np.arange(x_min, x_max, hx), np.arange(y_min, y_max, hy))

# Obtain labels for each point in mesh. Use last trained model.
Z = clusters.predict(np.c_[xx.ravel(), yy.ravel()])

In [11]: # TODO: Find the centroids for KMeans or the cluster means for GMM
centroids = clusters.means_
print centroids

[[ 2339.15204219 -6708.93065712]
 [ 7174.54719282  5469.02876453]
 [-15372.37194307 -3334.43379857]
 [-9486.9742574  34645.20428228]]

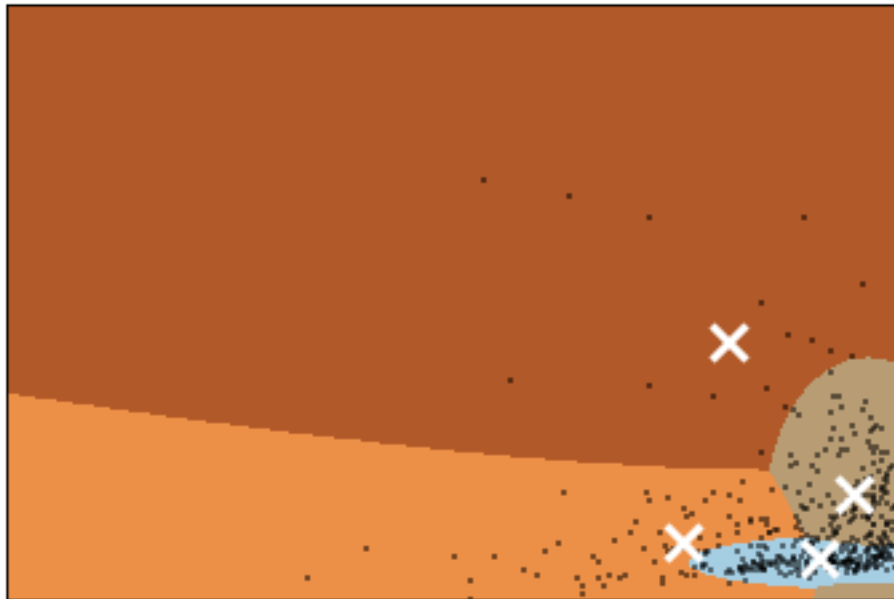
In [12]: # Put the result into a color plot
Z = Z.reshape(xx.shape)
plt.figure(1)
plt.clf()
plt.imshow(Z, interpolation='nearest',
            extent=(xx.min(), xx.max(), yy.min(), yy.max()),
            cmap=plt.cm.Paired,
            aspect='auto', origin='lower')

plt.plot(reduced_data[:, 0], reduced_data[:, 1], 'k.', markersize=2)
plt.scatter(centroids[:, 0], centroids[:, 1],
            marker='x', s=169, linewidths=3,
            color='w', zorder=10)
plt.title('Clustering on the wholesale grocery dataset (PCA-reduced data)\n'
          'Centroids are marked with white cross')
plt.xlim(x_min, x_max)
plt.ylim(y_min, y_max)
plt.xticks(())
plt.yticks(())
plt.show()

```

Clustering on the wholesale grocery dataset (PCA-reduced data)

Centroids are marked with white cross



After trying a couple of different number of clusters I find that 4 seems to best create groups. Two main ones on the bottom right are quite clear to see. The other two (that takes most of the screen) would be described more as not being the other two more than being clusters of their own. Sort of the category “Other”. They are very spread out. The lowest group at the bottom right (spread on the horizontal axis) seems to have its own complementary group, its own “Other” group (the large space at the bottom). Same thing for the top two groups.

7) What are the central objects in each cluster? Describe them as customers.

Answer: - The central objects are the means of every gaussian mixtures. Data points are given a class based on which mixture provides the highest probability but it has a “membership” to all other mixtures as well. It’s a little bit of this and a little bit of that. The central objects are the highest point of every mixtures. - Choosing the top two dimensions of the PCA, we see that the points spread along both axis - If we see the central points, the means of the gaussians, as customers they would be your average customer for that group. If we were to have stereotypes for those groups, the central points would be your champion, the face of every group.

```
In [13]: df = pd.DataFrame(
    np.concatenate(
        (data, clusters.predict(reduced_data).reshape((440,1))),
        axis=1),
    columns=['Fresh', 'Milk', 'Grocery', 'Frozen', 'Detergents_Paper', 'Delicatessen', 'label'])
```

Let’s look at the median of every category for each label

```
In [14]: plt.cm.Paired(0)
```

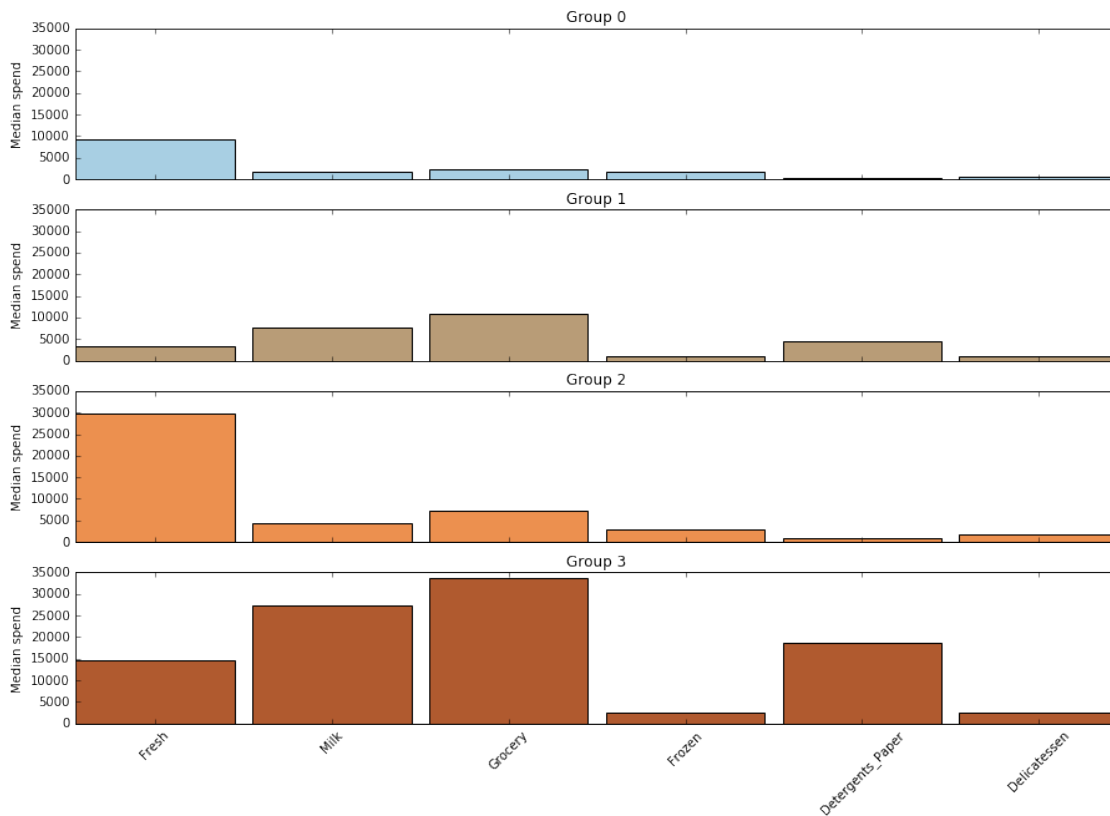
```
Out[14]: (0.65098041296005249, 0.80784314870834351, 0.89019608497619629, 1.0)
```

Let’s show a bar plot of the median of each group for every categories. I chose the same color as the plot as question number 6.

```

In [15]: fig, ax = plt.subplots(4,1, sharey=True, figsize=(15,10), sharex=True)
        ax = ax.ravel()
        ind = np.arange(6)
        width=0.9
        columns = ['Fresh', 'Milk', 'Grocery', 'Frozen', 'Detergents_Paper', 'Delicatessen']
        colors = ["#a8cfe3", "#b89c77", "#ec904f", "#b05a2f"]
        for label in range(4):
            data = df[df.label==label][columns].median()
            ax[label].bar(ind+width, data.values, width, color=colors[label])
            ax[label].set_xticks(ind+1.5*width)
            ax[label].set_xticklabels(columns, rotation=45)
            ax[label].set_ylabel('Median spend')
            ax[label].set_title("Group {}".format(label))

```



As I mentioned at the question number 6 there seemed to be two groups with two sub groups in each for a total of 4. There are two main groups and two others that I considered to be the “other” category of the main two. Here we see that group 0 and 3 are very proportional to each other and same thing for group 1 and 2. So I will analyze them together.

Group 0 and 3 Relative to one another, the categories Frozen and Delicatessen account for a lot less than every thing else in the basket. This sounds like an average family that goes a lot in the middle isle (cans, pasta and such).

Group 1 and 2 Relative to one another, the category Fresh is dominating the basket, the rest playing a much minor role in spending. We can see a link with the first component of PCA where Fresh was the most

influential factor. So this is a group focused on eating Fresh. Perhaps a group of people who are following the Paleo diet or quite simply just a restaurant or a caterer.

Overall we could probably use two groups only. The first two dimensions of PCA already accounted for over 85% of the variance and it shows in this analysis of GMM. Using 4 looked interesting at first only to realize that there were only two on a different scale.

1.3.2 Conclusions

8) Which of these techniques did you feel gave you the most insight into the data?

Answer: - A combination of both is really the answer here especially if we can cover enough variance with 3 dimensions or less. Reducing the number of dimensions enough allow you to visualize your dataset. - With this visualization you can then judge if the KMean or GMM is struggling to find groups or if you can see the pattern that they detected. - Choosing just one or the other I would go for GMM or Kmean alone. PCA and ICA are really hard to interpret. Clustering algorithms on the other hand allow you to separate your dataset into smaller groups so you can try to find what makes that group unique.

9) How would you use that technique to help the company design new experiments?

Answer: - Understanding your customers can give you an advantage on how to increase their spending. As I mentioned above you could have complementary offers that would be attractive to your most important group. Or change the layout of your store to either bring products that sell well together or go in the opposite direction completely and put them far apart so you get more traffic on the floor.

10) How would you use that data to help you predict future customer needs?

Answer:

- You could try to predict the impact of the socio-economical realities on your groups. For example, if you figure out that you have a group of young parents then you know they won't need diapers for the next 5 years. Their needs will evolve. Or if your customers are other businesses and you find out that there is a group that are best represented as Sport Restaurant then you better be ready for the Stanley Cup, the Super Bowl and the World Series with enough chicken wings and pop corn.

In []: