Image Compression using SVD and Dimensionality Reduction using PCA

Presented By:
Sainadh Chilukamari & Saransh Bhalla

Overview

- What is SVD
- Image Compression
- Compressed images using SVD
- Quality and Space
- Demo
- Observations

- About PCA and principal Components
- About the Dataset
- Data Preprocessing
- Results of PCA

Singular Value Decomposition

- Given any *mxn* matrix **A**, algorithm to find matrices **U**, **V**, and **W** such that

  $\mathbf{A} = \mathbf{U}\,\mathbf{S}\,\mathbf{V}^T$

  U is *mxm* and orthogonal eigenvectors of AA^T

  S is *mxn* and diagonal (Singular Values)

  V is *nxn* and orthogonal eigenvectors of A^TA

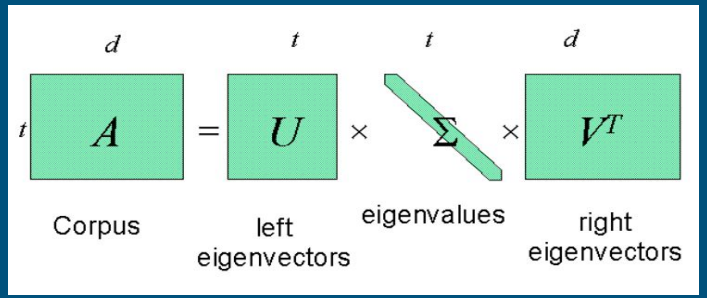


Image Compression

To reduce the cost of storage of digital images.

Storing/sharing high resolution digital images comes at a cost

- More memory is needed to share/accept/store such file(s)
- Web sites on the internet are generally made up of many pictures – systems can become completely clogged with the storage/transmission of images
- You need capable machinery

Image Compression

- Apply SVD to images and then extracting "enough" information from the images to recreate images that are "close" to their originals.
- Based on the level of detail that is needed in an image, we can choose to store more or less bytes in the compressed file.
- Steps:
    - Decompose a matrix **A** into $\mathbf{U}\,\mathbf{S}\,\mathbf{V}^T$
    - Set the Singular values in decreasing order and accordingly adjust the other matrices
    - Select a desired rank - remove redundant information
    - Truncate the matrices based on the k value. This is the compressed file
    - Use the compressed file to get back the original image

Image Compression

Suppose we have a grayscale image

- (126x128 pixels)
- We can use a matrix to represent this image
- Each component is represented by a 0-255
- Write it as a 126X128 matrix A

Next, we can decompose A by SVD which gives us

- $A = USV^T$
- U is 126x126, S is 126X128, and V is 128x128
- columns of U= are eigenvectors of AA^T
- diagonal entries of S= the normalized singular values ($\sqrt{\lambda}$for A^TA)
- columns of V= are eigenvectors of A^TA
- U and V are orthogonal matrices

Image Compression

To compress an image, after applying SVD:

- Retain the first few singular values.
- Discard the lower singular values based on the value of k or $\sigma_n <<< \sigma_1$

Since the singular values σ_i are ordered as

$\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_n \geq 0$, significant compression of the image is possible if the spectrum of singular values has only a few very strong entries.

By eliminating small singular values, we can approximate A

Compression Measures

- The rate of compression achieved using SVD with different ranks of approximation.
  - (Size of original image - Size of approximated image) / size of original image
- Mean Square Error (MSE): is the measure of degradation of compressed image quality as compared to the original image.
  - $MSE = \Sigma\Sigma (A(x,y) - A'(x,y))^2) / (m*n);$

Storage Space Calculations

Data stored in compressed file:

- U matrix (mxk)
- S array (k)
- V matrix (nxk)
- Value m
- Value n
- Value k

Storage strategy: IEEE standard 754-2008 half precision floating point (binary16)

Total space = 2*k(m+n+1)+6

Demo

SVD compressed images for different values of k

Original Image: CAS (500 x 376)

Rank 100, MSE: 11.91%, CR:74.49%

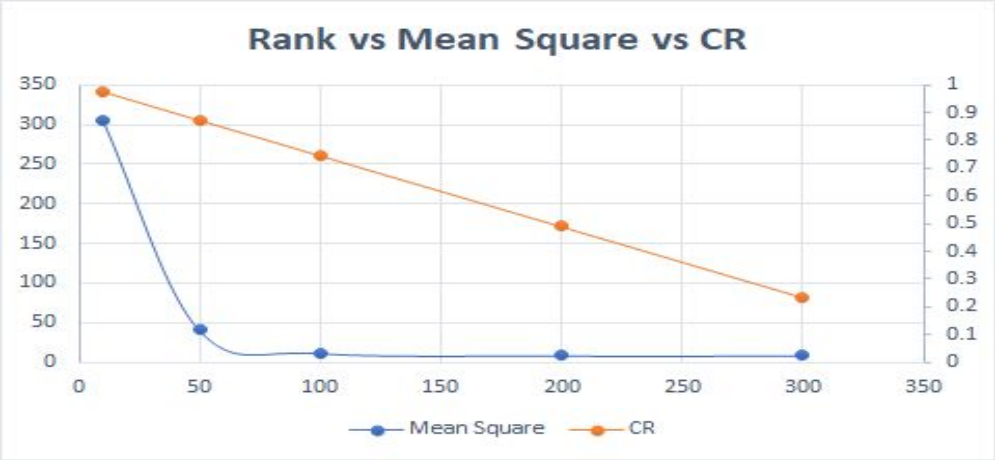
SVD compressed images for different values of k

Rank 50, MSE: 40.84%, CR: 87.24%

Rank 10, MSE: 304.94%, CR: 97.44%

Observations

- k represents the number of singular values used in the reconstruction of the compressed image.
- Smaller the value of k, more is the compression ratio but image quality decreases.
- As the value of k increases, image quality improves (MSE is small) and more storage space is more.



Dimensionality Reduction using PCA

Principal Component Analysis

- Principal Component Analysis (PCA) is a dimensionality reduction technique that can be utilized for extracting information from a high-dimensional space by projecting it into a lower-dimensional subspace.
- It tries to preserve the essential parts that have more variation of the data and remove the non-essential parts with fewer variation.
- One important thing to note is that, it is an Unsupervised dimensionality reduction technique, you can cluster the similar data points based on the feature correlation between them without any supervision.

Principal Components

- When the data is projected into a lower dimension (assume x dimensions) from a higher space, the x dimensions are the called the Principal Components which captures most of the variance (info.) of data.
- Principal components have both direction and magnitude.
    - Direction represents across which principal axes the data is mostly spread out.
    - Magnitude signifies the amount of variance that Principal Component captures.

Dataset - Wholesale Customers

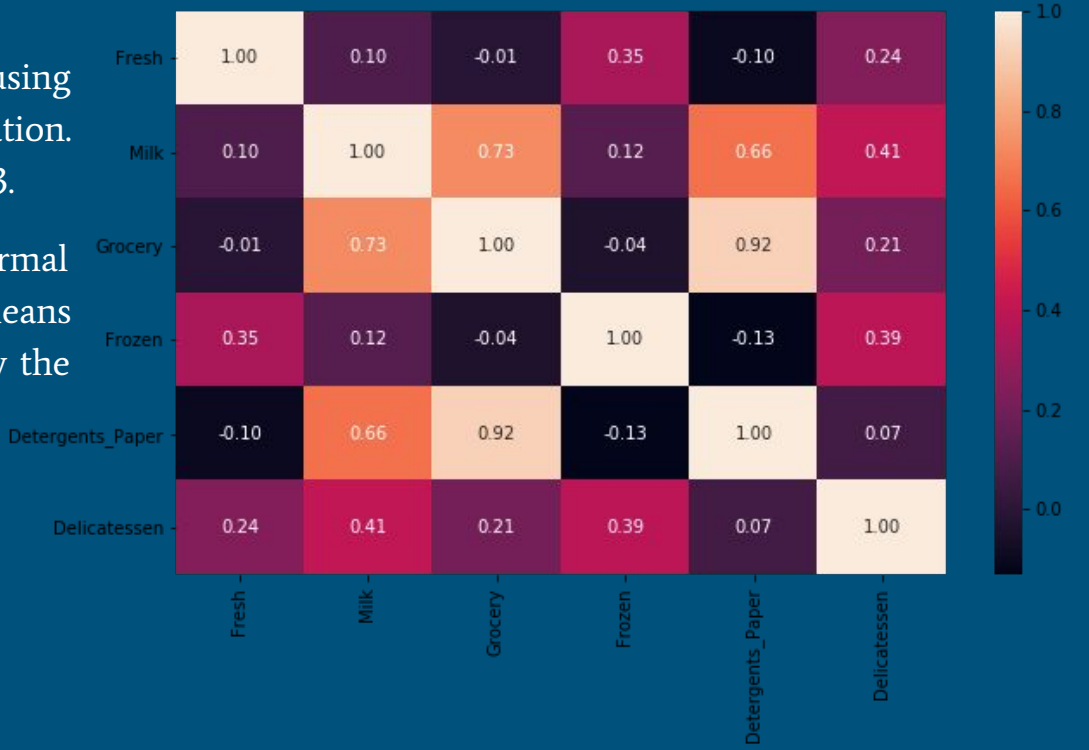
- Data on various customers annual spending amounts (reported in monetary units) of diverse product categories for internal structure.
- Data has 440 samples which composed of six important product categories:
    - ‘Fresh’, ‘Milk’, ‘Grocery’, ‘Frozen’, ‘Detergents_Paper’, and ‘Delicatessen’.
- Assuming sample 0 - restaurant, sample 1 - retailer, sample 2 - small cafe.

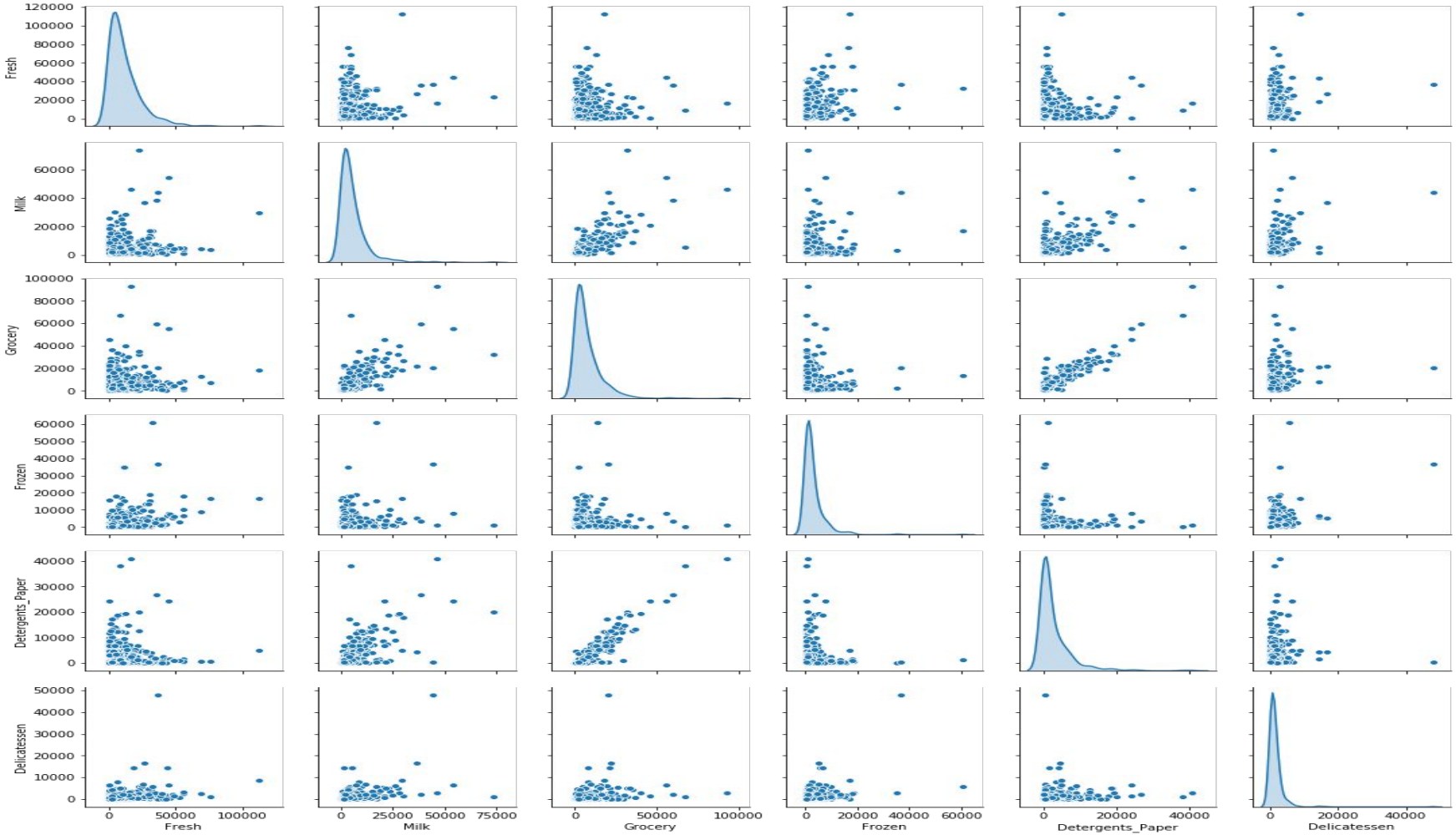
	Fresh	Milk	Grocery	Frozen	Detergents_Paper	Delicatessen
0	9898	961	2861	3151	242	833
1	45640	6958	6536	7368	1532	230
2	518	4180	3600	659	122	654

- Fresh: 12000.2977
- Milk: 5796.2
- Grocery: 7951.277
- Frozen: 3071.9
- Detergents_paper: 2881.4
- Delicatessen: 1524.8

Dataset Preprocessing

- We have tried to predict Fresh category using other variables to check the correlation. Regressor has an prediction score of -0.33.
- Few features seem to show a skewed normal along its own marginal axes which means that a particular feature isn't affected by the other

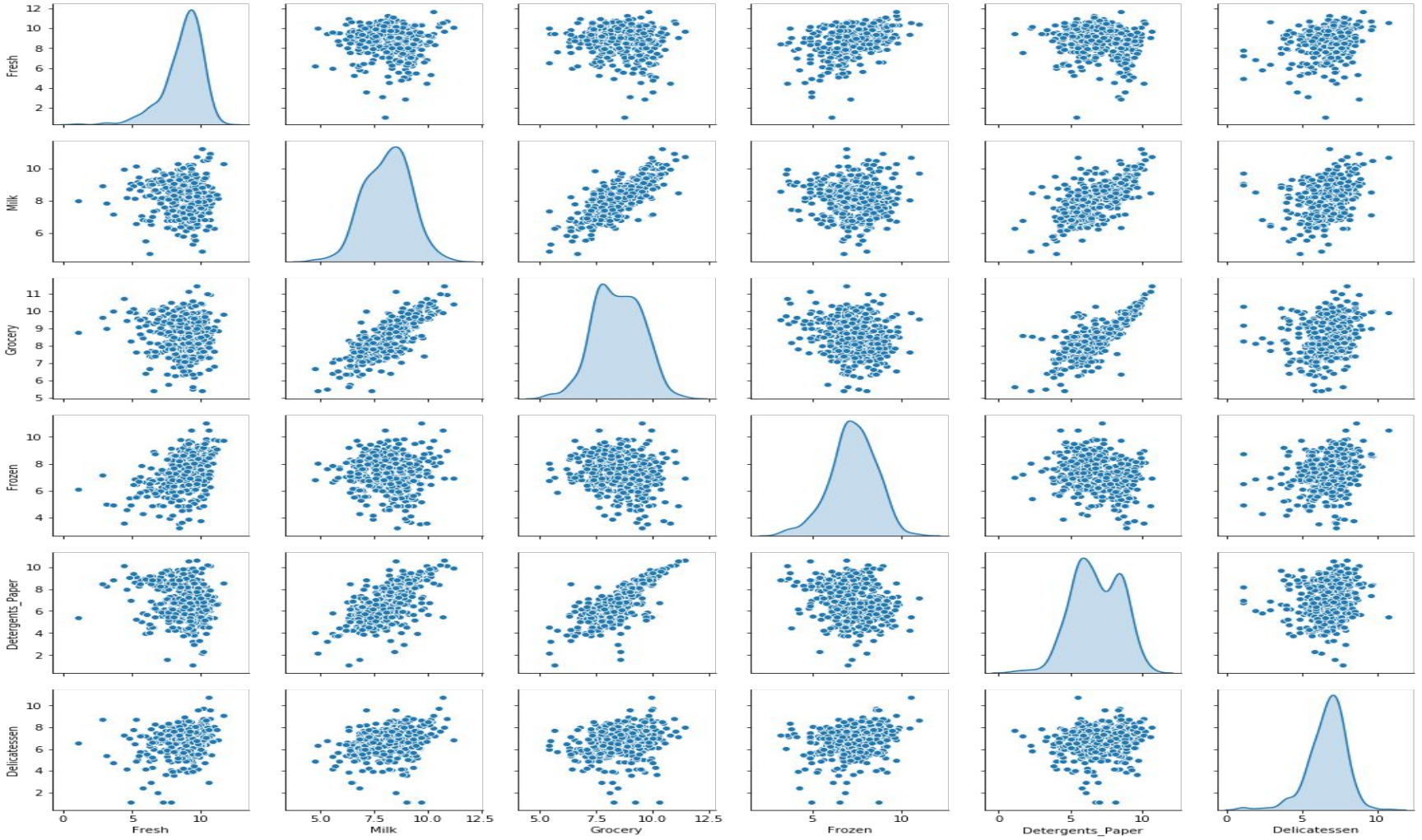



Fresh
Milk
Grocery
Frozen
Detergents_Paper
Delicatessen

Dataset Preprocessing

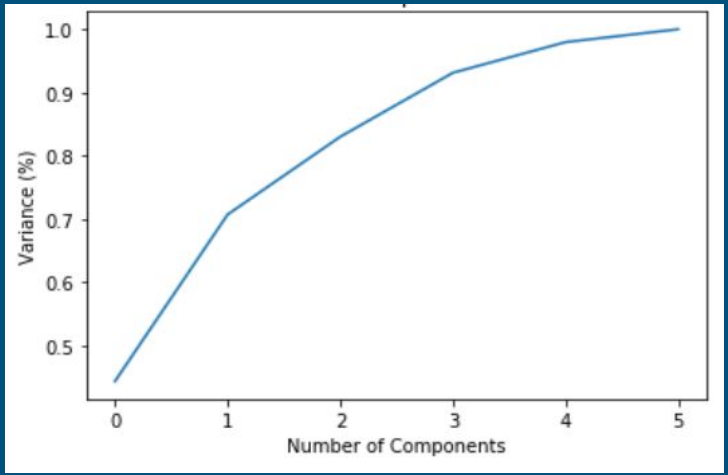
- Preprocessed the data to create a better representation of customers by performing a scaling on the data and detecting (and optionally removing) outliers.
- Applied the natural logarithm - scaling
- These outliers does not add value to algorithm (for eg k-means) and makes the algorithm perform worse in few cases. An outlier step is calculated as 1.5 times the interquartile range (IQR).
- A data point with a feature that is beyond an outlier step outside of the IQR for that feature is considered abnormal.

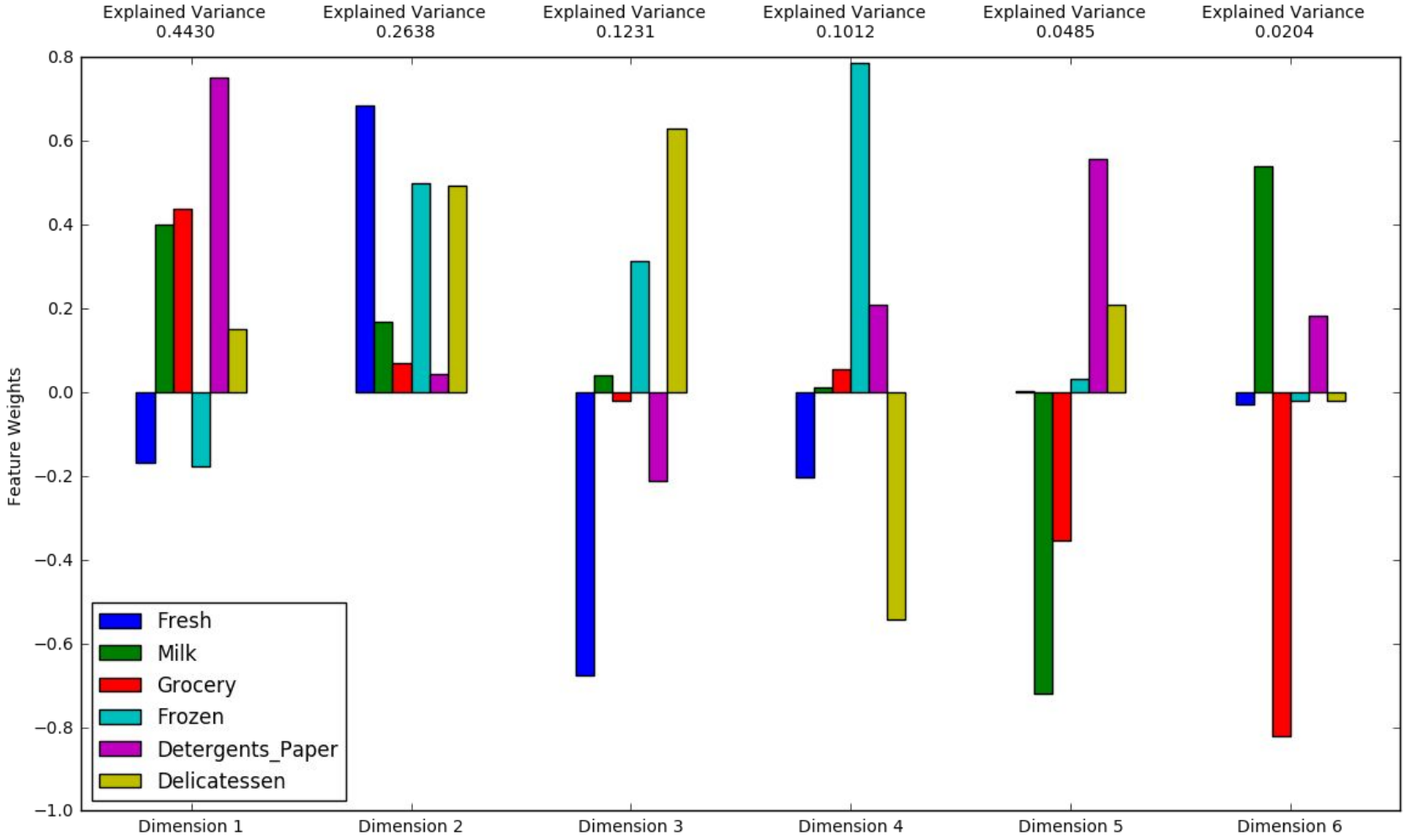
	Fresh	Milk	Grocery	Frozen	Detergents_Paper	Delicatessen
0	10.350606	7.558517	8.404920	9.149316	7.775276	8.374246
1	6.432940	4.007333	4.919981	4.317488	1.945910	2.079442
2	9.345046	6.939254	6.366470	8.592301	4.304065	5.509388

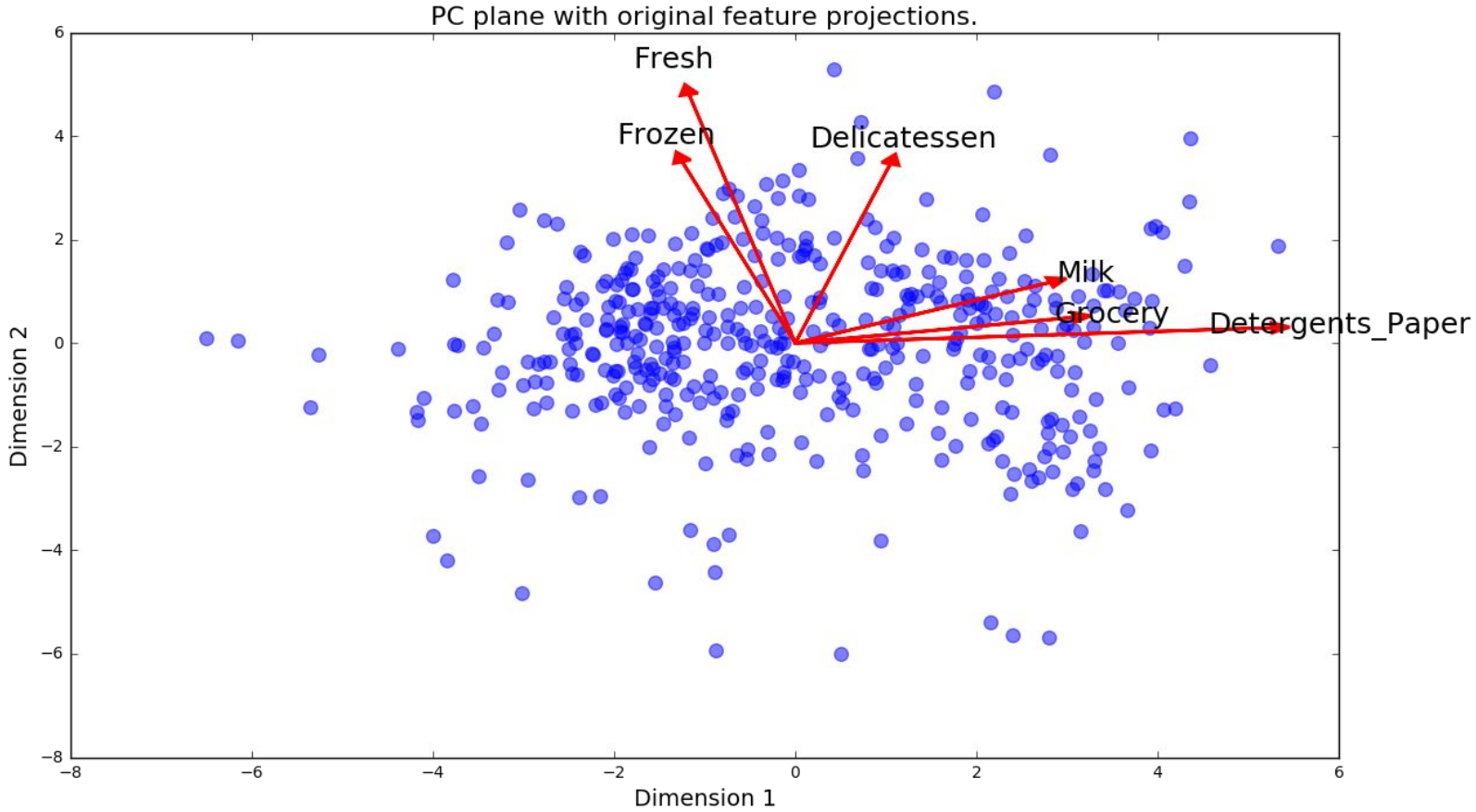

Fresh
Milk
Grocery
Frozen
Detergents_Paper
Delicatessen
Fresh
Milk
Grocery
Frozen
Detergents_Paper
Delicatessen

Feature Transformation

- Now, the data has been scaled to a more normal distribution and has had any necessary outliers removed.
- We can now apply PCA to good data to discover which dimensions about the data best maximize the variance of features involved.




Explained Variance
0.4430
Explained Variance
0.2638
Explained Variance
0.1231
Explained Variance
0.1012
Explained Variance
0.0485
Explained Variance
0.0204
Feature Weights
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6
−0.8
−1.0
Fresh
Milk
Grocery
Frozen
Detergents_Paper
Delicatessen
Dimension 1
Dimension 2
Dimension 3
Dimension 4
Dimension 5
Dimension 6


PC plane with original feature projections.
Fresh
Frozen
Delicatessen
Milk
Grocery
Detergents_Paper
Dimension 1
Dimension 2
6
4
2
0
−2
−4
−6
−8
−8
−6
−4
−2
0
2
4
6

Questions?