

# FINAL REPORT

## ISYE 6740, FALL 2022

Out-of-stock substitution recommendation system

Ryan Keeney  
rkeeney6@gatech.edu

### **Abstract**

This paper explores the unsupervised clustering of similar images, with the goal of making visually similar and categorically dissimilar recommendations. To demonstrate this, different methods of measuring image similarity were reviewed, and 5 Pokémon “substitutes” were recommended given an input image. A convoluted neural net was found to have the best performance in clustering images with visually similar features vs. PCA and ISOMAP. However, it also occasionally returned images with similar context (stated anti-objective).

## Introduction (Problem Statement)

Identifying substitutable substitutions for out-of-stock products can improve the customer experience and the company's profit [6]. The use of product graphs, labeled product features, and customer purchase habits are often used to build these recommendation systems. Companies such as Doordash utilize these systems regularly, as demonstrated in figure 1.

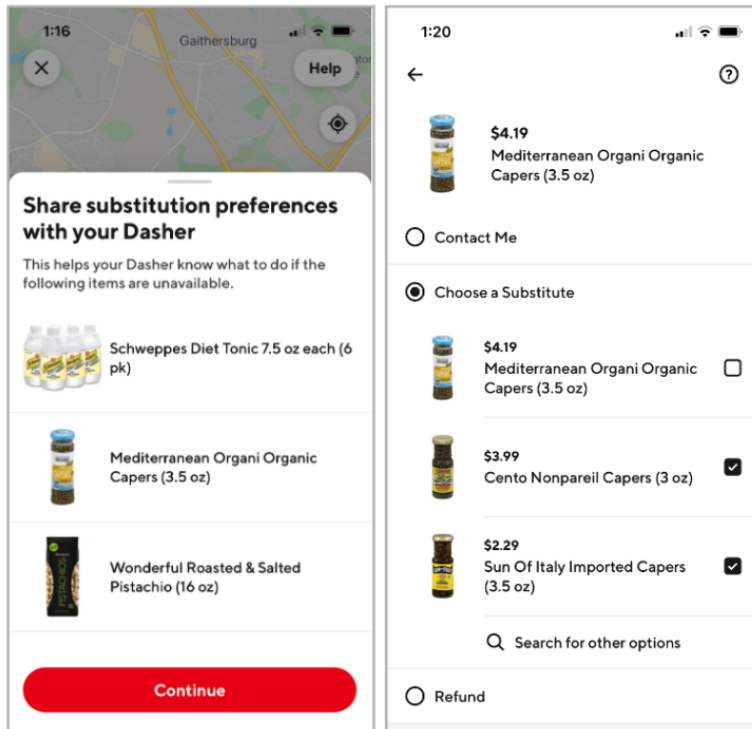


Figure 1: Good recommendation

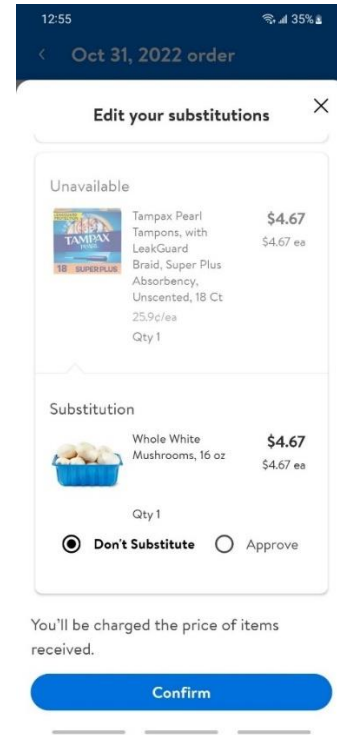


Figure 2: Poor recommendation

A good substitution recommender should provide products that are (1) available, (2) similar, and (3) within an acceptable price range of the desired out-of-stock item. However, some product substitution recommenders provide results that do not match the customer's intent. In figure 2, Twitter user @ask\_aubry was recommended whole white mushrooms as a replacement for tampons – a poor recommendation that seems to be based on (1) availability, (2) *visual similarity*, and (3) price similarity.

Motivated by this unique example, this paper explores the unsupervised clustering of similar images, intending to make recommendations that are *visually* similar and *categorically* dissimilar. To demonstrate this, different methods of measuring image similarity were reviewed, and given an input image, 5 Pokémon “substitutes” are recommended.

As an additional motivation, this presents the opportunity to utilize a convolution neural net that was trained on Imagenet [7]. These models were discussed extensively in ISyE 6740, and practical experience with these models was desired by the writer.

## Background (Data Sources)

The goal of making recommendations that are *visually* similar and *categorically* dissimilar can be broken down into three key steps: (1) Image pre-processing, (2a) feature extraction and (2b) dimensionality reduction, and (3) cluster analysis. A list of common dimensional reduction techniques reviewed is provided in table 1.

### COMMON DIMENSIONAL REDUCTION TECHNIQUES

---

Principal component analysis, PCA<sup>1</sup>  
Manifold learning, ISOMAP<sup>1</sup>  
Deep learning via convolution neural net, CNN<sup>1</sup>  
Independent component analysis, ICA  
Multi-dimensional scaling, MDS  
LLE  
t-SNE  
Autoencoders

<sup>1</sup>*Selected methods*

*Table 1: Common dimensional reduction techniques*

## Pokémon Images Database

By using Pokémon as the recommended substitutes, the model will inherently recommend a categorically dissimilar product if a non-Pokémon image is an input. This dataset also presents some unique challenges and opportunities for evaluation. The effectiveness of the algorithm can be measured through the evaluation of the output clusters or recommendations when inputting Pokémon images. For example, recommended substitution images should contain Pokémon that are related through evolution (e.g., Pikachu to Raichu), type (e.g., Electric), features (e.g., Zig-zag tail), or color composition (e.g., Yellow). The database has 809 images of Pokémon [2].



*Figure 3: Poliwhirl #61*

## PCA

PCA is a form of linear dimensionality reduction [1]. PCA is a projection-based method that transforms the data by projecting it onto a set of orthogonal axes. Non-linear kernel extensions of PCA exist, as well as singular value decomposition, SVD, but were not evaluated. The steps for PCA are listed in table 2.

STEP	DESCRIPTION
1	<p>Given <math>m</math> data points, <math>\{x^1, \dots, x^m\} \in \mathbb{R}^d</math></p> <p>Estimate the mean and covariance matrix</p> $\mu = \frac{1}{m} \sum_{i=1}^m x^i, \quad C = \frac{1}{m} \sum_{i=1}^m (x^i - \mu)(x^i - \mu)^T$ <p>Where <i>Captures</i> the variability of the data points along different directions, it also captures correlations, covariance between different coordinates (features in the <math>x</math> vector).</p>
2	<p>Take the eigenvectors, <math>w^1, w^2 \dots</math> of <math>C</math> corresponding to the largest eigenvalue <math>\lambda_1</math>, second largest eigenvalue <math>\lambda_2</math>, etc.</p> $C = U\Lambda U^T$ <p><math>C</math> has <math>d \times d</math> dimensions</p> <p><math>U = \text{eigenvectors}</math></p> $\Lambda = \begin{matrix} \lambda_1 & & \\ & \dots & \\ & & \lambda_d \end{matrix}$ <p>diagonal matrix of the eigenvalues, arranged from largest to smallest.</p>
3	<p>Compute the reduced representation</p> $z^i = \begin{pmatrix} \frac{w^{1T}(x^i - \mu)}{\sqrt{\lambda_1}} \\ \dots \end{pmatrix}$ <p><u>With</u></p> <p><math>w^{1T}</math>: transposed eigenvector</p> <p><math>(x^i - \mu)</math>: mean</p> <p><math>w^{1T}(x^i - \mu)</math>: gives a number</p> <p><math>\frac{w^{1T}(x^i - \mu)}{\sqrt{\lambda_1}}</math>: normalizes it dividing by the squareroot of the corresponding eigenvalue</p> <p><u>Resulting in</u></p> <p><math>z_i</math>, principal components, each entry is a principal component, e.g., <math>\frac{w^{1T}(x^i - \mu)}{\sqrt{\lambda_1}}</math> is the 1st principal component.</p> <p><math>z_i</math>: <math>k</math>-dimensional vector (reduced dimension) with projection along features</p>

Table 2: PCA algorithm

## ISOMAP

Isomap is a non-linear dimensionality reduction method. It attempts to preserve geodesic (local) distances [4,5]. The goal of ISOMAP is to produce a low dimensional representation of the data that preserves the ‘walking distance’ of the cloud data (manifold). It is calculated in the following steps [4,5].

1. **Create adjacency matrix of local Euclidean distances:** Find neighbors  $N(i)$  of each data point,  $x^i$ , within distance  $\epsilon$  and let the adjacency matrix,  $A$ , recording neighbor Euclidean distance.
2. **Create shortest path pairwise matrix:** Matrix  $D$  between each pair of points,  $x^i, x^j$  based on  $A$ .
3. **Reduce dimensional representation:** Find a low dimensional representation that preserves the distances in  $D$

## Convolution neural net (VGG16)

VGG16 is a convolution neural net (CNN ) architecture that was used to win ILSVR (Imagenet) competition in 2014 [3,7]. It is available through the Keras [<https://keras.io/api/applications/>] library. Utilizing a pre-trained model to extract features significantly reduces the computational time (eliminates the need to re-train model), and the performance of the model is well established [3,7].

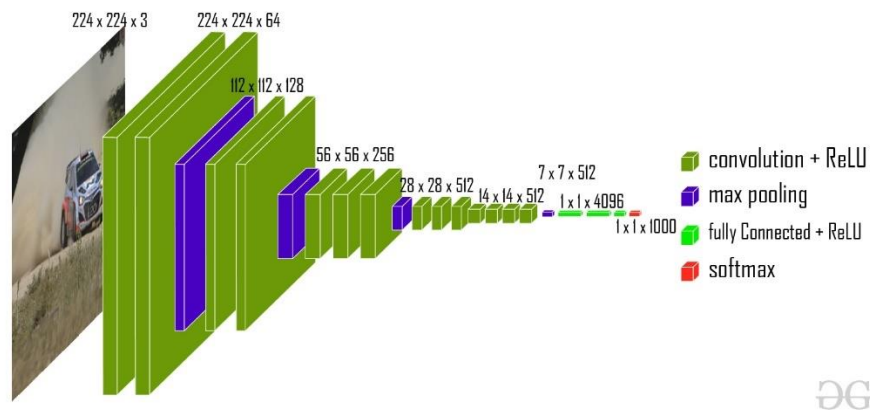


Figure 4: VGG-16 architecture [3]

## Methodology

The process of identifying visually similar images was broken down into four steps, and three different methodologies were evaluated.

1. Image preprocessing
2. Featured extraction
3. Dimensionality reduction
4. Recommendation clustering

With a secondary objective of obtaining practical experience with a deep learning model, the following three methods were evaluated.

Method 1	Method 2	Method 3
No feature extraction or dimensionality reduction. RGB file is used directly for clustering.	Feature extraction and dimensionality reduction through ISOMAP or PCA were completed before clustering.	Complete feature extraction with VGG16 then dimensionality reduction through ISOMAP or PCA before clustering

## Image pre-processing

809 images [3] were loaded and converted to RBG files (120, 120, 3) for methods 1 and 2 (default image size), and (224, 224, 3) for method 3 due to the requirements for VGG16.

For methods 1 and 2, RGBA and Greyscale conversions of images were evaluated but discarded for RGB formats to match the parameters required for VGG16.

## Feature extraction

Feature extraction is skipped in method 1, performed with PCA or ISOMAP in method 2, and performed with VGG16 in method 3. Deconstruction (blurring via gaussian and box) techniques for method 2 were also evaluated but did not appear to improve performance.

## Dimensionality reduction

In methods 2 and 3, feature extraction was performed through PCA or ISOMAP. Various numbers of features were evaluated,  $n = 2, \dots, 100$ , and for the final analysis  $n = 100$  features were selected based on literature review [1,6,7].

## Recommendation Clustering

In all methods, the clusters were evaluated via K-Means and the top-n recommendations were gathered using a K-nearest neighbor algorithm. The K-Means algorithm was evaluated at multiple clusters, ranging from 18 (the number of types) to 100.

## Evaluation

The recommended substitution images were evaluated visually through dimensional representation (PCA, ISOMAP) and through comparison of image recommendations. For example, an image should be related to the input image through features such as evolution (e.g., Pikachu to Raichu), type (e.g., Electric), features (e.g., Zig-zag tail), or color composition (e.g.,

Yellow). Formal evaluation of the recommendations, such as a supervised classification model based on Pokémon type are beyond the scope of this project, although

## Evaluation and Final Results

The findings are summarized in table 3. Overall, visual inspection did not detect a significant difference between ISOMAP and PCA for dimensionality reduction for methods 1 and 2. For method 3, PCA outperformed ISOMAP for dimensionality reduction. Method 3 provided the most intuitive groupings and recommendations, while method 1 and 2 were sensitive to image scale as demonstrated in table 5. All methods did not provide convincing recommendations when the input image differed significantly from the Pokémon in the data (e.g., box of tampons).

Method 1	Method 2	Method 3
No feature extraction or dimensionality reduction. RGB file used directly for clustering.	Feature extraction and dimensionality reduction through ISOMAP or PCA completed before clustering.	Complete feature extraction with VGG16 then dimensionality reduction through ISOMAP or PCA before clustering
<b>Findings, Advantages</b>		
Best color-matching across clusters	Allowed for flexible examination of feature dimensions	Best performance when inputting similar Pokémon images
Second best performance when inputting similar Pokémon images		Clusters of images had similar features
<b>Findings, Disadvantages</b>		
	Sensitive to image scale	
Sensitive to image scale	Worst performance when inputting similar Pokémon images	Can return images with similar context (stated anti-objective)
Poor recommendation when input was visually complex and dissimilar from Pokémon.	Poor recommendation when input was visually complex and dissimilar from Pokémon.	Poor recommendation when input was visually complex and dissimilar from Pokémon.

*Table 3 Summary of findings*

Method 3 provided the best output clusters, often matching features such as wings or “plant-like” over color.



Figure 5 Example cluster, Method 3



Figure 6: Example cluster, Method 3

Method 1 and 2 were particularly sensitive to image scale and background color.









INPUT	METHOD 1	METHOD 2	METHOD 3
			
			

Table 4 Comparing input image scale



Overall, the outputs of method 3 were the best, followed by method 1, and then method 2. A sample recommendation for a Pokémon and non-Pokémon are recorded in table 5.

































INPUT	RECOMMENDED SUBSTITUTIONS, METHOD 1
	    
	    
INPUT	RECOMMENDED SUBSTITUTIONS, METHOD 2
	    
	    
INPUT	RECOMMENDED SUBSTITUTIONS, METHOD 3
	    
	    

Table 5 Comparing similar images for Pokémon and non-Pokémon

Method 3 would often recommend products that were contextually similar (e.g., look like a fruit or vegetable) but were not visually similar. However, overall, this method provided the best results. Since this is a stated anti-objective, this was not preferred and recorded as a disadvantage of this method. For making recommendations that are *visual* similar and *categorically* dissimilar, additional context through labeled data could be established as a min-max problem: maximize visual similarity while minimizing contextual similarity.


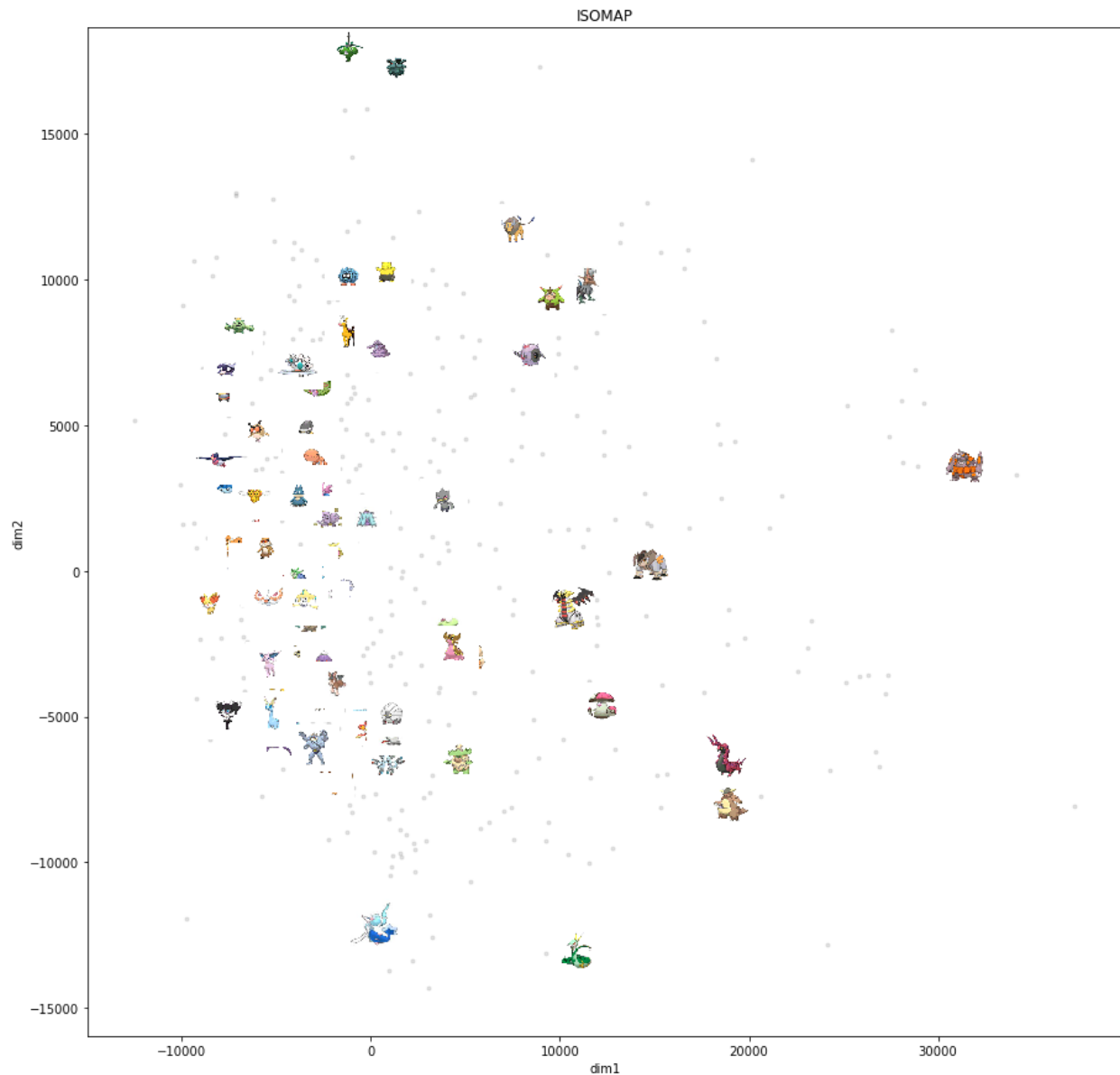
INPUT	TOP-5 RECOMMENDED SUBSTITUTIONS, DESC., METHOD 3					
						
						
						
						
						
						

Table 6 Method 3 recommendations for non-Pokémon

One advantage of the method 2 was the ability to vary the selected  $n$  features and plot the dimensions to identify which features were being selected. Unfortunately, this method did was not particularly useful on this diverse dataset, although it did provide more insight when greyscale images were used, often appearing to map features along dimensions such as scale, orientation, brightness, and texture.



*Figure 7 ISOMAP 1st and 2nd dimensions*

When clustering was performed, the number of clusters was varied, and the sum of squared distance was evaluated. Inertia measures how well a dataset was clustered by K-Means. It is

calculated by measuring the distance between each data point and its centroid, squaring this distance, and summing these squares across one cluster.

The Pokémon data is presenting a challenge in that there is not an elbow point. The sum of squared losses is still reducing even at 50 clusters, indicating the extreme diversity within the dataset.

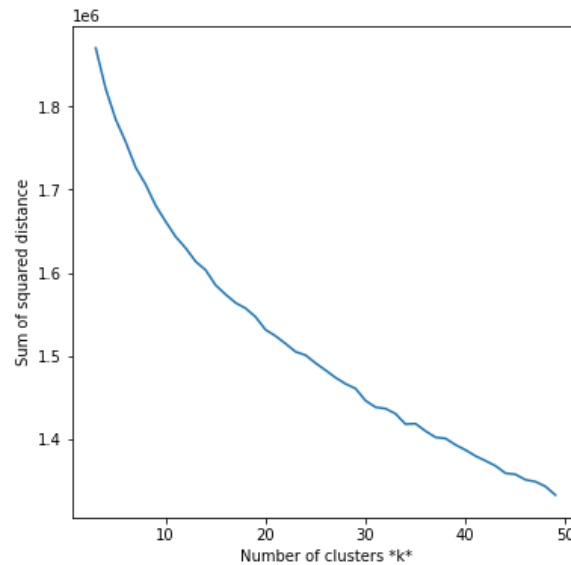


Figure 8 Sum of squared distance, Method 3

## Future Extensions Recommendations

### Supervised classification for feature identification

Formal evaluation of the recommendations, such as a supervised classification model based on Pokémon type are beyond the scope of this project, or misclassification rate based on clusters may help enable the algorithm to identify key Pokémon attributes such as color, feature, or type.

### RGB density clustering

Additionally, another clustering method, using RGB density curves to match color compositions may improve the ability to provide visually similar recommendations, but may not account for features such as orientation, scale, and textures.

### Min-max optimization

For making recommendations that are visual similar and categorically dissimilar, additional context through labeled data could be established as a min-max problem: maximize visual similarity while minimizing contextual similarity.

## Appendix

### References

- [1] Hastie, T., Tibshirani, R., Friedman, J. (2001). *The Elements of Statistical Learning*. New York, NY, USA: Springer New York Inc..\
- [2] KVPRATAMA. (2020, August 10)]. *Pokémon Images Dataset, Version 2*. Retrieved 10/29/2022 from <https://www.kaggle.com/datasets/kvpratama/pokemon-images-dataset>
- [3] GeeksforGeeks. (2022, August 24). VGG-16 | CNN model. <https://www.geeksforgeeks.org/vgg-16-cnn-model/>
- [4] M. Balasubramanian, E. L. Schwartz, *The Isomap Algorithm and Topological Stability*. Science 4 January 2002: Vol. 295 no. 5552 p. 7
- [5] Tenenbaum, J. B., de Silva, V. & Langford, J. C. (2000). *A Global Geometric Framework for Nonlinear Dimensionality Reduction*. Science, 290, 2319.
- [6] Pande, A., Gupta, A.D., Ni, K., Biswas, R., & Majumdar, S. (2020). *Substitution Techniques for Grocery Fulfillment and Assortment Optimization Using Product Graphs*.
- [7] Simonyan, K. and Zisserman, A., *Very Deep Convolutional Networks for Large-Scale Image Recognition*, 2014.