UDACITY

PROJECT

Creating Customer Segments

A part of the Machine Learning Engineer Nanodegree Program

PROJECT REVIEW

NOTES

SHARE YOUR ACCOMPLISHMENT!

Meets Specifications

Awesome adjustment here in your submission, you clearly have a solid grasp on these unsupervised learning techniques. Wish you the best of luck in your future!

If you would like to dive in deeper into Machine Learning material, here might be some cool books to check out

- An Introduction to Statistical Learning Code is in R, but great for understanding
- elements of statistical learning More mathy
- Python Machine Learning I have this one, great intuitive ideas and goes through everything in code.

Data Exploration

Three separate samples of the data are chosen and their establishment representations are proposed based on the statistical description of the dataset.

Nice job referencing the descriptive stats of the dataset for justification for your sample establishment. Just note that it may be more appropriate to use the median / quartiles for justification rather than the mean, since the median and quartiles are more robust to outliers, which we have here.

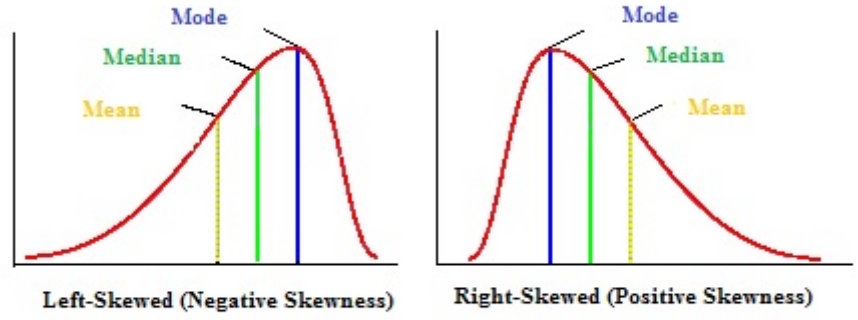
A prediction score for the removed feature is accurately reported. Justification is made for whether the removed feature is relevant.

Good! We can also predict all features with the use of a for loop

```
for feature in data.columns:
    new_data = data.drop(feature, axis=1)

    # TODO: Split the data into training and testing sets using the given feature as the target
    from sklearn import cross_validation
    label_data=data[feature]
    X_train, X_test, y_train, y_test = cross_validation.train_test_split(new_data, label_data, test_size=0.25, random
_state=0)

    # TODO: Create a decision tree regressor and fit it to the training set
    from sklearn import tree
    regressor = tree.DecisionTreeRegressor(random_state=0)
    regressor.fit(X_train, y_train)

    # TODO: Report the score of the prediction using the testing set
    score = regressor.score(X_test,y_test)
    print feature, score
```

Student identifies features that are correlated and compares these features to the predicted feature. Student further discusses the data distribution for those features.

Great job with the correlation between the features, as Grocery/Detergents_Paper are correlated and Milk/Detergents_Paper and Milk/Grocery are also correlated but not to the same degree. To expand on your comment of "*The data for these features are not normally distributed. Most of the data points lie in the range of 0-10,000('Milk','Frozen', 'Detergents_Paper', and 'Delicatessen') or 0-20,000 ('Grocery' and 'Fresh') and only a few data points has extremely high value.*" As we can also see that these resemble a skewed right distribution(most points on the left side), as we can actually get an idea of this from the basic stats of the dataset, since the mean is above the median.



Data Preprocessing

Feature scaling for both the data and the sample data has been properly implemented in code.

Student identifies extreme outliers and discusses whether the outliers should be removed. Justification is made for any data points removed.

Nice job discovering the indices of the five data points which are outliers for multiple features of `[65, 66, 75, 128, 154]`

Interesting idea to not remove these data points with your comment of "*since there is no enough evidence of these outliers significantly impact the performance of the model we are building or even lead to false conclusions of the problem we are trying to solve.*" A good learning experience here would be to run clustering with and without these data points and see how the silhouette scores change. Do these 5 data points influence our clustering

algorithm?

Feature Transformation

The total variance explained for two and four dimensions of the data from PCA is accurately reported. The first four dimensions are interpreted as a representation of customer spending with justification.

You really should also mention what the relationship between the positive and negative features represent in the third and forth component for a more complete answer. As a principal component with feature weights that have opposite directions can reveal how customers buy more in one category while they buy less in the other category.

But impressive with your analysis of the first two components. Since PCA deals with the variance of the data and the correlation between features, the first component would represent that we have some customers who purchase a lot of Milk, Grocery and Detergents_Paper products while other customers purchase very few amounts of Milk, Grocery and Detergents_Paper, hence spread in the data. So good with your comments such as "*a high value in first dimension indicates a spending behavior tendency like a grocery store*"

Pro Tip: You can also visualize the percent of variance explained to get a very clear understanding of the drop off between dimension. Here is a some starter code, as np.cumsum acts like `+=` in python.

```
import matplotlib.pyplot as plt
x = np.arange(1, 7)
plt.plot(x, np.cumsum(pca.explained_variance_ratio_), '-o')
```

PCA has been properly implemented and applied to both the scaled data and scaled sample data for the two-dimensional case in code.

Clustering

The Gaussian Mixture Model and K-Means algorithms have been compared in detail. Student's choice of algorithm is justified based on the characteristics of the algorithm and data.

Just note that speed really wouldn't be a factor with this small dataset. As the main two differences in these two algorithms are the speed and structural information of each:

Speed:

- K-Mean much faster and much more scalable
- GMM slower since it has to incorporate information about the distributions of the data, thus it has to deal with the co-variance, mean, variance, and prior probabilities of the data, and also has to assign probabilities to belonging to each clusters.

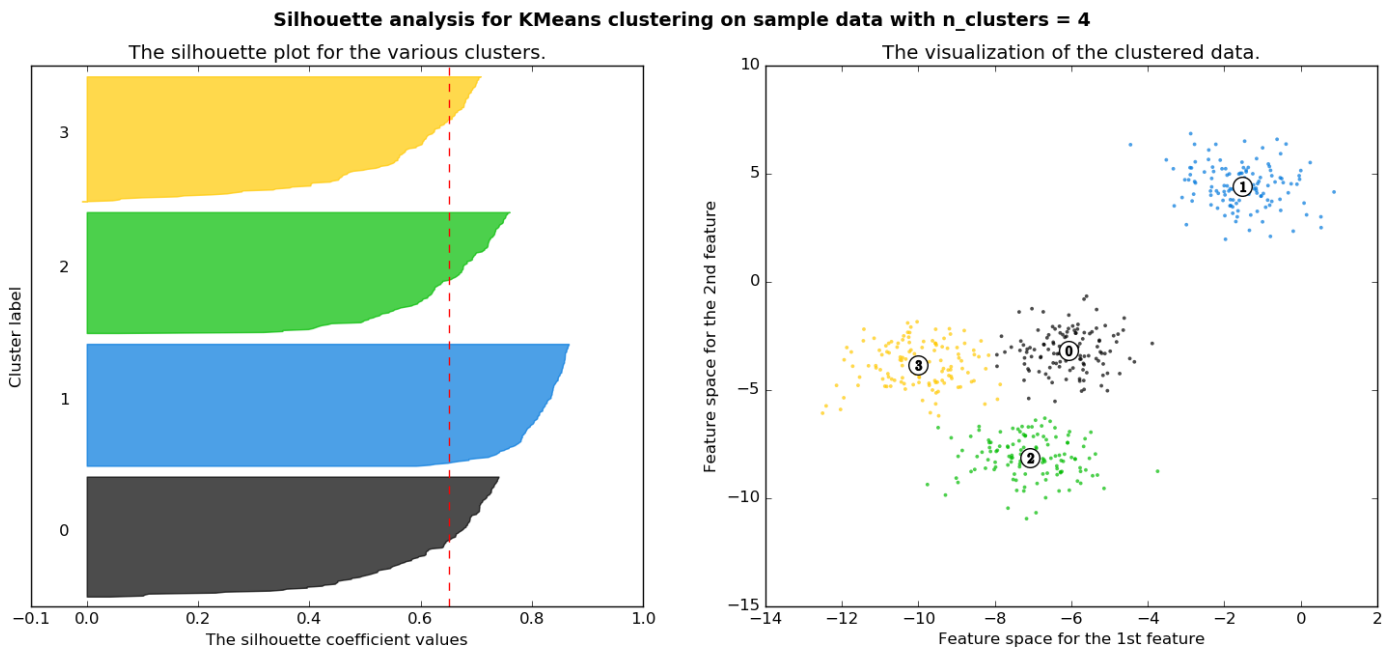
Structure:

- K-Means straight boundaries (hard clustering)
- GMM you get much more structural information, thus you can measure how wide each cluster is, since it works on probabilities (soft clustering)

Several silhouette scores are accurately reported, and the optimal number of clusters is chosen based on the best reported score. The cluster visualization provided produces the optimal number of clusters based on the clustering algorithm chosen.

Great job with your for loop! Another really cool interpretation technique for Silhouette score is like this

(http://scikit-learn.org/stable/auto_examples/cluster/plot_kmeans_silhouette_analysis.html)



The establishments represented by each customer segment are proposed based on the statistical description of the dataset. The inverse transformation and inverse scaling has been properly implemented and applied to the cluster centers in code.

Nice job here comparing the purchasing behaviors of the cluster centroids to the descriptive stats of the dataset, i.e. mean. I also like how you have referenced the reduce PCA plot here as well, as these clusters do differ in terms of Dimension 1?

Pro Tip: We can also add the median values from the data and very easily visualize the cluster centroids with a pandas bar plot

```
true_centers = true_centers.append(data.describe().ix['50%'])
true_centers.plot(kind = 'bar', figsize = (16, 4))
```

Sample points are correctly identified by customer segment, and the predicted cluster for each sample point is discussed.

Great justification for your predictions by comparing the purchasing behavior of the sample to the purchasing behavior of the cluster centroid! If you were to use GMM, we could also check out the probabilities for belonging to each cluster with the use of `clusterer.predict_proba()`

Conclusion

Student correctly identifies how an A/B test can be performed on customers after a change in the wholesale distributor's service.

"*The wholesale distributor should take samples from cluster 1 spliting into control group and variation group and do the same to cluster 0. Conduct the A/B tests seperately for each group to see if the hypothesis that most of customers in cluster 1 would reach positively to the change in delivery service*"

Excellent! We should run separate A/B tests for each cluster independently. As if we were to use all of our customers we would essentially have multiple variables(different delivery methods and different purchasing behaviors).

The two clusters that we have in our model reveal two different consumer profiles that can be tested via A/B test. To better assess the impact of the changes on the delivery service, we would have to split the segment 0 and segment 1 into subgroups measuring its consequences within a delta time. Hypothetically we can raise a scenario where the segment 0 is A/B tested. For this we divide the segment 0 (can also be implemented in segment 1) into two sub-groups of establishments where only one of them would suffer the implementation of the new delivery period of three days a week, and the another would remain as a control with five days a week as usual. After a certain period of time, we could, through the consumption levels of the establishments, come to some conclusions, such as: whether the new frequency of deliveries is sufficient or not for a buyer. Where a sensible increase in overall consumption of all products may indicate the need for the estabelishment to mantain a storage because of the decreasing delivery frequency; or if it negatively affects the consumption profile of certain products, like groups of costumers who have greater buying fresh produce that can be negatively impacted, precisely because of the demand for fresh products with a higher delivery frequency. We can not say that the change in frequency will affect equally all customers because of the different consumption profiles that are part of the two segments. There will therefore consumers that will be affected, and possibly groups of buyers who will not undergo any change.

Student discusses with justification how the clustering data can be used in a supervised learner for new predictions.

"*The wholesale distributor should use the customer segment as the target variable and the customer spending data as training data to train a classification algorithm. And then use the new customer estimated product spending as the testing data for the classifier to predict which customer segment each new customer belongs to.*"

Nice idea to use the cluster assignment as new labels. Another cool idea would be to use a subset of the newly engineered features as new features(great

for curing the curse of dimensionality). PCA is really cool and seem almost like magic at time. Just wait till you work with hundreds of features and you can reduce them down into just a handful. This technique becomes very handy especially with images. There is actually a handwritten digits dataset, using the "famous MNIST data" where you do just this and can get around a 98% classification accuracy after doing so. This is a kaggle competition and if you want to learn more check it out here KAGGLE

Comparison is made between customer segments and customer 'Channel' data. Discussion of customer segments being identified by 'Channel' data is provided, including whether this representation is consistent with previous results.

Real world data is really never perfectly linearly separable but it seems as our K-Means algorithm did a decent job. Interesting sample here!

DOWNLOAD PROJECT

Have a question about your review? Email us at review-support@udacity.com and include the link to this review.

RETURN TO PATH

Rate this review

Student FAQ