Occupation Explorer

Tiffany Barkley & Anna Swigart

Introduction

There are hundreds of different occupations that Americans hold. Given the breadth of possibilities, it is no wonder that many people struggle with deciding which career paths to pursue. When trying to understand one's choices for possible occupations for which current skills, abilities, and preferences might transfer to, there are currently few ways to consider multiple variables together. Many resources in this area focus on similar occupations around particular educational backgrounds or industry-specific skillsets, while leaving out other knowledge, work activities, and work context measures that may be important to an individual.

While sources like the US Department of Labor's O*NET Resource Center provide basic tools for exploring employment opportunities, they focus on specific characteristics that aren't very meaningful on their own. The analysis we conducted provides new insight about ways to identify the degree to which and dimensions on which nearly any occupation may transfer to another occupation, based on a diverse set of attributes. The primary measures used here are Euclidian distance and clustering algorithms to identify pairwise and cluster-wise occupations that are highly similar. We also use Euclidean distance to identify the major differences between occupation pairs.

Problem

Since people often consider many factors when they think about choosing or switching occupations, we sought a dataset that included a large array of features for each occupation. In our search, we were pleased to find the Occupational Information Network (O*NET), which is developed under the sponsorship of the US Department of Labor and the Employment and Training Administration. This dataset contained 974 unique occupations, assessed according to the following 10 domains of features (248 total features): Abilities (52), Interests (6), Job Zone (1), Knowledge (33), Skills (35), Work Activities (41), Work Context (56), Work Context Time/Seasonality (2), Work Styles (16), and Work Values (6).

Additional feature domains that were in the dataset but excluded from our analysis include 'Green Occupations', longform occupation description text in 'Occupation Data', and 'Education, Training, and Experience' which we found to be represented well in the 'Job Zones' measure. Several other data files included additional data, descriptions, and ways to categorize domains such as Work Context and Task. We removed this additional data when forming our master dataframes. Some domains also had multiple scales on which the features were evaluated. For instance, Abilities was assessed on an Importance scale of 1-5 as well a level scale of 0-7. We needed one value for each feature, per occupation, and decided to ignore feature ratings on the 'level' scale because it was only used for a subset of occupations and the meaning of this was not entirely clear, despite the existence of thorough documentation. Some occupations had a Y value in a column that was called 'not_relevant'. This field was calculated based on some proportion of experts rating that occupation who felt the particular feature was irrelevant. After some investigation, it seemed that the Importance scales would capture features of low importance and irrelevance roughly equally, so we elected to keep occupation feature values that were tagged as not relevant.

More details about the data exploration we performed in order to familiarize ourselves with and select features can be found in this iPython notebook.

Solution

Once our data was cleaned up and contained only the features we cared about, we created a pandas pivot table for each domain, with unique occupation code as rows and each occupation feature as a column (with hierarchical column structures to preserve Domain identity). We then merged the dataframes based on the unique occuption code and normalized the dataset in order to get every feature onto the same scale. The original scales were either 1-3, 1-5, or 0-7, and were between 0 and 1 after applying a min-max normalization.

At this point we plotted a histogram to visualize each feature and inspect any patterns or problems that could be identified. We compared histograms for normed and raw data, and realized we had a problem. We had initially elected to fill NA values in our dataset with zeros, which introduced skew into our data for variables on an original scale of 1 and higher. From this, we revised the data processing and ultimately elected to remove occupations that had NAs (because those that had any NA values had them for a lot of features).

Our goal was to implement a clustering algorithm in python scikit-learn to cluster groups of similar occupations so that a user of the system could obtain career recommendations given an input occupation. A main challenge with clustering was the high-dimensionality of our data set. We attempted to use Principal Component Analysis to reduce the dimensions so that we could at least visualize the data points, but found that two PCA components only explained slightly over half of the variance in our data set. This made it difficult to gauge the effectiveness of the clustering. We began using DBSCAN, but found that it classified more than half of the occupations as noise and that the resulting clusters were highly overlapping. We then switched to hierarchical clustering using the Ward algorithm in scikit-learn, which yielded better results. Below is an example of one of the clusters that was computed using the Ward algorithm:

```
Occupations in cluster 39 (13 total):
  Environmental_Compliance_Inspectors
  Soil_and_Plant_Scientists
  Zoologists_and_Wildlife_Biologists
  Soil_and_Water_Conservationists
  Range_Managers
  Park_Naturalists
  Foresters
  Environmental_Scientists_and_Specialists__Including_Health
  Environmental_Restoration_Planners
  Geoscientists__Except_Hydrologists_and_Geographers
  Hydrologists
  Archeologists
  Forest_Fire_Inspectors_and_Prevention_Specialists
```

Details

Our DBSCAN attempt required us to tune two parameters: (1) the maximum distance between two samples for them to be considered as in the same neighborhood; and (2) the number of samples in a neighborhood for a point to be considered as a core point. We attempted to tune the parameters by looping through what we determined to be reasonable values for each input, and choosing the parameters with the optimal silhouette coefficient. We had to tune our loop because some of the values we had selected resulted in error messages that were difficult to interpret. In the end, none of the values showed a good silhouette coefficient, and all classified too many of the occupations as noise. We determined that our data set was not well suited for density clustering.

The Ward hierarchical clustering algorithm only required an input of the number of clusters. Through trying different values and evaluating whether the resulting clusters made sense to us given what we know about the occupations, we settled on 50 clusters as a reasonable value. We could not find a way in scikit to more quantitatively confirm the optimal number of clusters.

In terms of engineering challenges, while our data had high-dimensionality, the number of samples was relatively low (less than 1,000). As such, we were able to do all of our mining locally using iPython notebooks.

These analyses we conducted can be viewed and replicated with this iPython notebook.

Related Work and Resources

Hierarchical clustering:
http://scikit-learn.org/stable/auto_examples/cluster/plot_ward_structured_vs_unstructured.html

General scikit clustering guidance:
http://www.astro.washington.edu/users/vanderplas/Astr599/notebooks/17_SklearnIntro

mpld3 tooltip scatterplot example: http://mpld3.github.io/examples/scatter_tooltip.html

Further Work

We would like to expand the work we have done to create a web application and/or visualization in order to put our algorithms to work with interested users.

We found that the hierarchical clustering performed well in terms of generating informative clusters that yielded non-obvious similar occupations. On the output clusters, we are considering applying a supervised learning technique to figure out which features best distinguish between the different careers within a cluster.