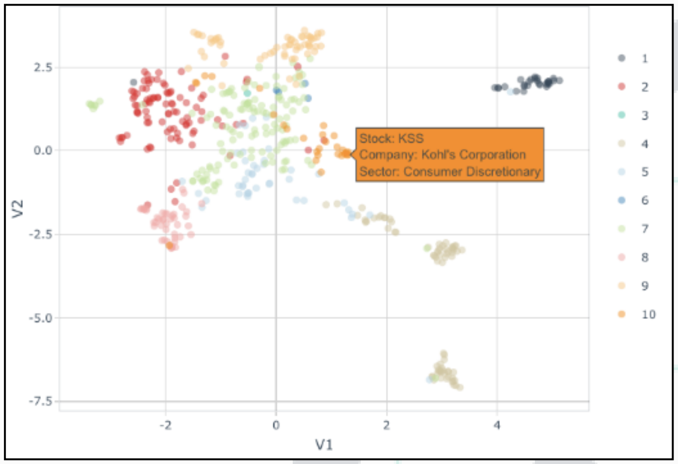
Segmentation & Clustering



Combining K-Means & UMAP to visualize clusters by Stock Price Movements

Summary:

- **Common Applications in Business**: Can be used for finding segments within Customers, Companies, etc.
- **Key Concept:** Transform data into a matrix enabling trends to be compared across units of measure (e.g. user-item matrix)
- **Gotchas:** Data must be normalized or standardized to enable comparison. This often requires calculation proportions of values by customer, company, etc to ensure the larger values do not dominate the trend mining operation.
- **How Many Components/Clusters?** Use a *Scree Plot* to determine the proportion of variance explained or total within sum of squares

Type	Popular Methods	Uses	Data Treatment
Clustering	K-Means Hierarchical Clustering	**Group Detection:** Methods use a measure of similarity (e.g. Euclidean distance) to detect groups within data set	Standardized or normalized
Dimensionality Reduction	PCA UMAP tSNE	**Reduce Width of Data:** Performing Machine Learning on wide data can drastically increase the time for algorithms to converge. Dimensionality reduction can be applied as a preprocessing step to reduce the width (number of columns) of the data but still maintain a high proportion of the overall structure. **Visualization:** Visualizing the first two components as X and Y often can enable cluster visualization. Combining with clustering techniques can provide a useful method of visualization.	Standardized or normalized

Resources

- Business Analysis With R Course (DS4B 101-R) - Modeling - Week 6
- Business Science Problem Framework
- Ultimate R Cheat Sheet | Ultimate Python Cheat Sheet

Data Science Courses for Business

R Cheat Sheet

K-Means

```
set.seed(0)

kmeans_obj <- kmeans(X, centers = 4)
```

UMAP

```
library(umap)

umap_obj <- umap(X)
```

Python Cheat Sheet

K-Means

```
from sklearn.cluster import KMeans

kmeans = KMeans(

    n_clusters=4,

    random_state=0).fit(X)
```

UMAP

```
import umap

reducer = umap.UMAP()

embedding = reducer.fit_transform(X)
```

Business Science University
university.business-science.io

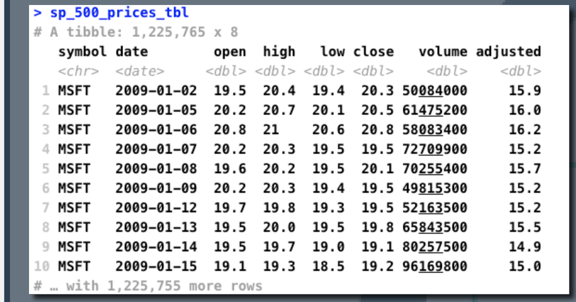
version: 1.0

Segmentation & Clustering

How to apply K-Means & UMAP step-by-step

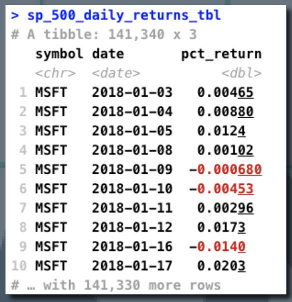
Clustering Workflow

Collect Data



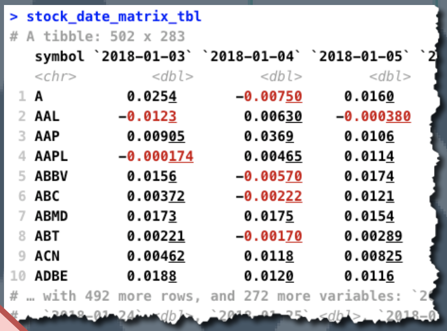
```
> sp_500_prices_tbl
# A tibble: 1,225,765 x 8
   symbol date        open  high   low close   volume adjusted
   <chr>  <date>     <dbl> <dbl> <dbl> <dbl>    <dbl>    <dbl>
 1 MSFT   2009-01-02  19.5  20.4  19.4  20.3 50084000     15.9
 2 MSFT   2009-01-05  20.2  20.7  20.1  20.5 61475200     16.0
 3 MSFT   2009-01-06  20.8  21    20.6  20.8 58083400     16.2
 4 MSFT   2009-01-07  20.2  20.3  19.5  19.5 72709900     15.2
 5 MSFT   2009-01-08  19.6  20.2  19.5  20.1 70255400     15.7
 6 MSFT   2009-01-09  20.2  20.3  19.4  19.5 49815300     15.2
 7 MSFT   2009-01-12  19.7  19.8  19.3  19.5 52163500     15.2
 8 MSFT   2009-01-13  19.5  20.0  19.5  19.8 65843500     15.5
 9 MSFT   2009-01-14  19.5  19.7  19.0  19.1 80257500     14.9
10 MSFT   2009-01-15  19.1  19.3  18.5  19.2 96169800     15.0
# … with 1,225,755 more rows
```

Standardize / Normalize



```
> sp_500_daily_returns_tbl
# A tibble: 141,340 x 3
   symbol date       pct_return
   <chr>  <date>          <dbl>
 1 MSFT   2018-01-03   0.00465
 2 MSFT   2018-01-04   0.00880
 3 MSFT   2018-01-05   0.0124
 4 MSFT   2018-01-08   0.00102
 5 MSFT   2018-01-09  -0.000680
 6 MSFT   2018-01-10  -0.00453
 7 MSFT   2018-01-11   0.00296
 8 MSFT   2018-01-12   0.0173
 9 MSFT   2018-01-16  -0.0140
10 MSFT   2018-01-17   0.0203
# … with 141,330 more rows
```

Spread to User-Item Format



```
> stock_date_matrix_tbl
# A tibble: 502 x 283
   symbol `2018-01-03` `2018-01-04` `2018-01-05` `2
   <chr>         <dbl>        <dbl>        <dbl>
 1 A           0.0254      -0.00750     0.0160
 2 AAL        -0.0123       0.00630    -0.000380
 3 AAP         0.00905      0.0369      0.0106
 4 AAPL       -0.000174     0.00465     0.0114
 5 ABBV        0.0156      -0.00570     0.0174
 6 ABC         0.00372     -0.00222     0.0121
 7 ABMD        0.0173       0.0175      0.0154
 8 ABT         0.00221     -0.00170     0.00289
 9 ACN         0.00462      0.0118      0.00825
10 ADBE        0.0188       0.0120      0.0116
# … with 492 more rows, and 272 more variables: `2
#   `2018-01-24` <dbl>, `2018-01-25` <dbl>, `2018-0
```

K-Means

Obtain cluster assignments

UMAP

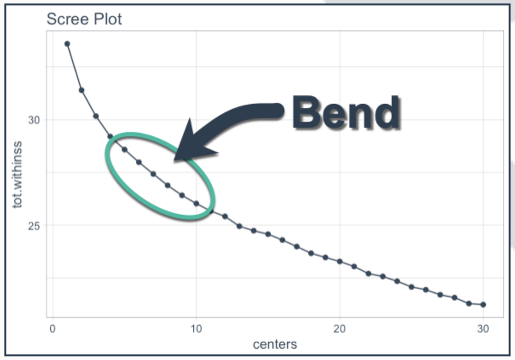
Make 2D Projection

K-Means: Scree Plot

Used to pick a value for K clusters for K-means algorithm.

Iteratively calculate "tot.withinss" for values of K.

Look for a bend.

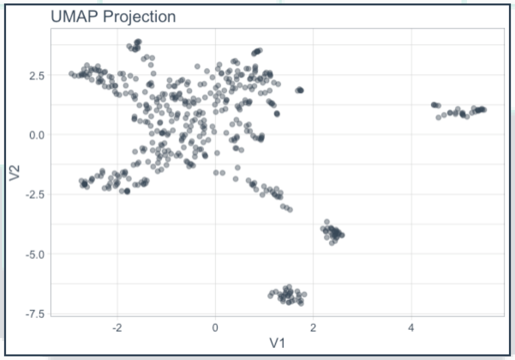


UMAP: 2D Projection

Fast dimensionality reduction algorithm that can be used for visualization.

Better than PCA - PCA is linear, UMAP is nonlinear

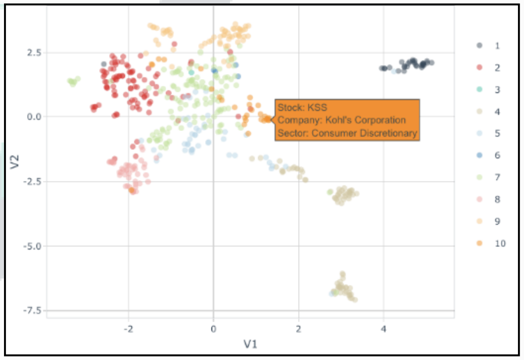
Better than tSNE - tSNE is slow



Combine

Plot the K-Means cluster assignments with the UMAP 2D Projection to obtain a visual.

Add interactivity to enable exploration.



Data Science Courses for Business

Business Science University
university.business-science.io