CGI/PNC Capstone Project

Adam Duvall, Charles Tyler, Jacob Humble, Jiayi Zhang

Purpose Revisited

Do linkages exist between companies in the Health Care and Real Estate sectors?

1. Where do we get our data from?
2. What data is relevant?
3. What format should our data be in?
4. How do we predict these linkages? Supervised or unsupervised models?
5. What should our output from these models be?
6. What is the best way to visualize this information?

Database Creation

- Initial Ideas: Web Scraping for general information, SEC 10k filings, etc.

- Settled on Dun & Bradstreet/Mergent for its exhaustive information on private companies

- Text Scraping from D&B company synopses for several key features

Database Creation - Text Scraping Process

1. Export lists of companies 100 at a time (Restriction set by Mergent)

1. Convert lists into text files in Word

1. Use scraping code to pull desired features from text files

1. Features from code are outputted into excel files, which from there are imported into Python

Database

- All healthcare companies from Pennsylvania, New York, West Virginia (7640)
- All real estate companies from Pennsylvania, New York, West Virginia (2465)
- 9 main features:
  - Company name
  - Executive name
  - Employment number
  - Year founded
  - Sales Volume
  - Primary SIC code
  - Phone number
  - Corporate family

Next Step: Predicting Linkages

- Now that we have assembled and cleaned all of our data, the next step is to use these characteristics to predict potential linkages between companies

- Which features will help establish linkages without overfitting the model?

- After trial and error, used four features: Company Name, Phone Number, Head Executive, and Corporate Family

- Predict model fit by running classifier and identifying accuracy. 456 vs 496 predicted, accuracy rate of ~91%

Next Step: Predicting Linkages

```
c.string('Executives', 'Executives', method='jarowinkler', threshold=0.95,label='Executives')

c.string('Company', 'Company', method='jarowinkler', threshold=0.85,label='Company')

c.string('Corporate Family', 'Corporate Family', method='jarowinkler', threshold=0.85, label='Corporate Family')

c.exact('Phone Number', 'Phone Number',label='Phone Number')
# %%

features = c.compute(candidate_links, df_a, df_b)
features['totallinks']= features['Executives'] + features['Company'] + features['Corporate Family'] +features['Phone Number']


print(features)
```

Next Step: Predicting Linkages

```
#%%
featuresab = features[features['totallinks']==2]
featuresab = featuresab[featuresab['Executives'] == 1]
featuresab = featuresab[featuresab['Company Name'] == 1]


featuresac = features[features['totallinks']==2]
featuresac = featuresac[featuresac['Executives'] == 1]
featuresac = featuresac[featuresac['Corporate Family'] == 1]

featuresad = features[features['totallinks']==2]
featuresad = featuresad[featuresad['Executives'] == 1]
featuresad = featuresad[featuresad['Phone Number'] == 1]

featuresbc = features[features['totallinks']==2]
featuresbc = featuresbc[featuresbc['Company Name'] == 1]
featuresbc = featuresbc[featuresbc['Corporate Family'] == 1]

featuresbd = features[features['totallinks']==2]
featuresbd = featuresbd[featuresbd['Company Name'] == 1]
featuresbd = featuresbd[featuresbd['Phone Number'] == 1]

featurescd = features[features['totallinks']==2]
featurescd = featurescd[featurescd['Corporate Family'] == 1]
featurescd = featurescd[featurescd['Phone Number'] == 1]
```

```
#%%
featuresabc3 = features[features['totallinks']==3]
featuresabc3 = featuresabc3[featuresabc3['Executives'] == 1]
featuresabc3 = featuresabc3[featuresabc3['Company Name'] == 1]
featuresabc3 = featuresabc3[featuresabc3['Corporate Family'] == 1]

featuresabd3 = features[features['totallinks']==3]
featuresabd3 = featuresabd3[featuresabd3['Executives'] == 1]
featuresabd3 = featuresabd3[featuresabd3['Company Name'] == 1]
featuresabd3 = featuresabd3[featuresabd3['Phone Number'] == 1]

featuresbcd3 = features[features['totallinks']==3]
featuresbcd3 = featuresbcd3[featuresbcd3['Company Name'] == 1]
featuresbcd3 = featuresbcd3[featuresbcd3['Corporate Family'] == 1]
featuresbcd3 = featuresbcd3[featuresbcd3['Phone Number'] == 1]

featuresabcd = features[features['totallinks']==4]
```

Network Graph

See html

Potential improvement

1. Initial request was for the percent likelihood that these companies were linked, however we could only define strength by the amount and type of combinations that companies had in common and the accuracy of our model

1. Gathering even more data across multiple states, especially Ohio, or gather data from other sources and compare to Mergent database

1. Create an interactive dashboard which would allow someone to search a company name, etc and automatically pull up the node and linkages associated with it