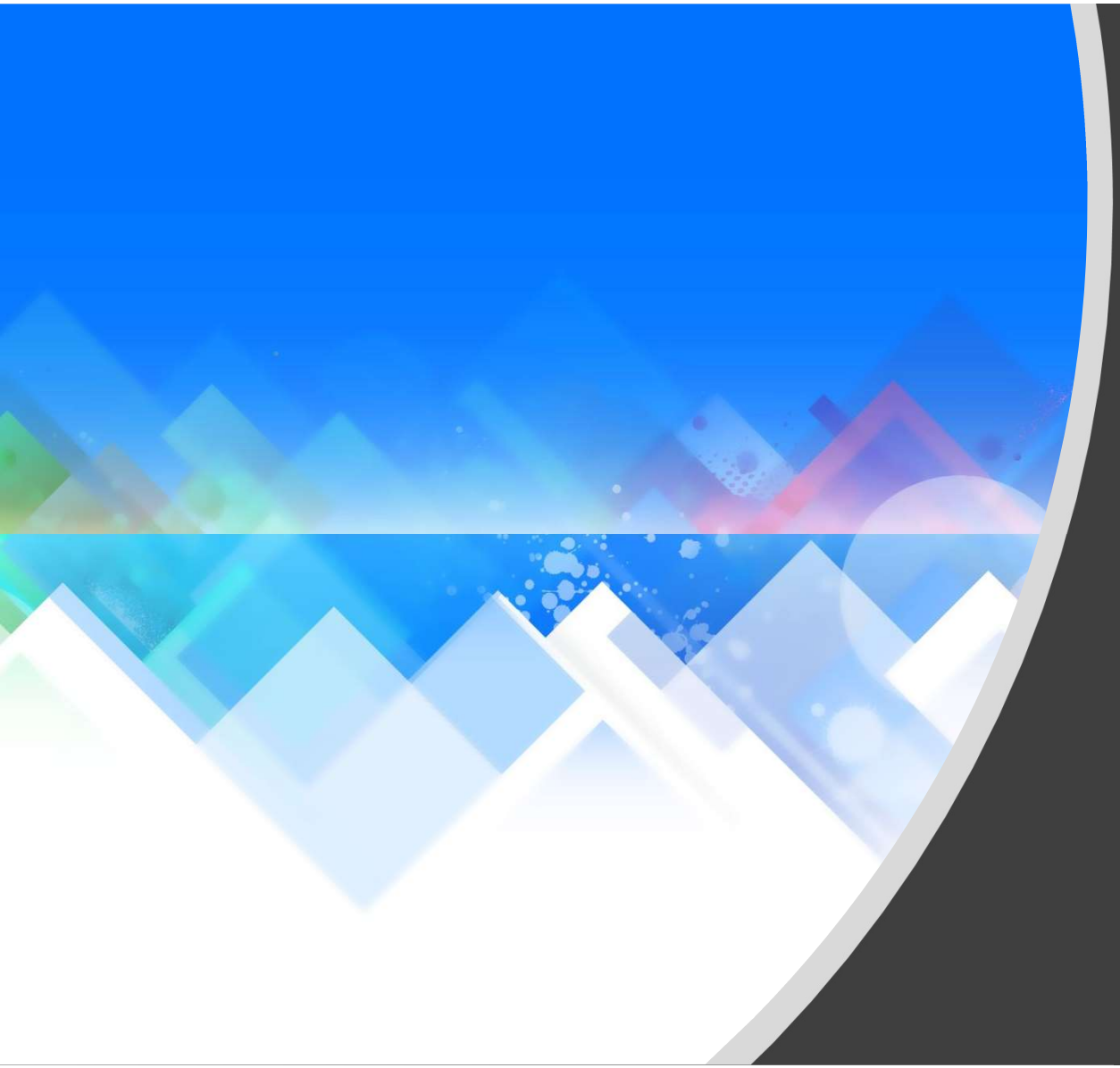


Optera Practicum

Matthew Peetz

Regis University

Outline

- Optera Data Project and Presentation File
- Objective
  - The goal of this project is to impute missing category 3 emissions based on reported com pany values
- EDA
  - The initial file is huge, 450 mb, it was cut up into individual sheets to start
- Working Data
  - Working File Created to perform modeling on
- Modeling

Data Set

- 8381 companies with Scope Three Emissions
- 2534 companies with Revenue data
- Reporting Data
- Scope 1 and 2 Data

Revenue Matching

- Initial Revenue / Company name matching very poor
  - 88 matches
- Matching using Optera name code list
  - 273 matches
- Difflib Library was tuned to match names to revenue data
  - **dl.get_close_matches(name, company_name_list, n=1, cutoff=0.4**
  - **810 matches 2020, 534 matches 2021**

```
The company name was: Alfa Financial Software Holdings
The suggested name list includes:
['Alfa Financial Software Hold', 'Fairfax Financial Holdings Ltd']


The company name was: Alfa Laval Corporate AB
The suggested name list includes:
['Alfa-Laval AB', 'CAB Cakaran Corporation Bhd']


The company name was: Alfen NV
The suggested name list includes:
['Adyen N.V.', 'Ageas NV']


The company name was: Algonquin Power & Utilities Corporation
The suggested name list includes:
['Algonquin Power & Utilities Corp', 'Lion Industries Corporation']


The company name was: Aliansce Sonae Shopping Centers SA
The suggested name list includes:
['American Shipping Co ASA', 'Avianca Holdings SA']


The company name was: ALIMENTOS CENTRALIZADOS DE MEXICO S DE RL DE CV  (axionlog cuautitlan)
The suggested name list includes:
[]


The company name was: All Access Apparel, Inc.
The suggested name list includes:
['Delta Apparel Inc.', 'Weis Markets, Inc.']


The company name was: Allegion Plc
The suggested name list includes:
['Allegion Plc', 'Adient Plc']
```

Revenue Matching

```
# Sorting the frame and creating a new revenue column


#0-30 Rows
# 30 is revenue_final
# 29 is Revenue (in $USD)
# 26 is Revenue


i = 0

while i < 8915:
  j = 30
  value = results_df.iat[i, j]

  if value == 0:
    j = 29                              # j is 29
    value = results_df.iat[i, j]
    if value == 0:
      j = 26                            # j is 26
      value = results_df.iat[i, j]
      if value == 0:
        results_df.iat[i, 30] = 'no revenue value'
        i = i + 1

      else:
        print(j)
        results_df.iat[i, 30] = value
        i = i +1
    else:
      print(j)
      results_df.iat[i, 30] = value
      i = i +1

  else:
    print('this is where you are stuck')
    results_df.iat[i, 30] = value
    i = i + 1
```

:ry	cat_1	...	cat_13	cat_14	cat_15	cat_16	cat_17	revenue	Company Name	Data Year	Revenue (in $USD)	revenue_final
:es	109.12	...	0.00	0.0	0.0	0.0	0.0	0.000000e+00	0	0.0	0.000000e+00	no revenue value
are ma	5281773.00	...	0.00	0.0	0.0	0.0	0.0	0.000000e+00	0	0.0	0.000000e+00	no revenue value
:es	229.10	...	0.00	0.0	0.0	0.0	0.0	0.000000e+00	0	0.0	0.000000e+00	no revenue value
ing	360246.54	...	13856.59	0.0	0.0	0.0	0.0	0.000000e+00	0	0.0	0.000000e+00	no revenue value
ırel	0.00	...	0.00	0.0	0.0	0.0	0.0	0.000000e+00	0	0.0	0.000000e+00	no revenue value
...	...	...	...	...	...	...	...	...	...	...	...	...
ing	671611.80	...	0.00	0.0	0.0	0.0	0.0	2.845088e+09	Advantest Corp	2021.0	2.845088e+09	2845088230.0
tail	0.00	...	0.00	0.0	0.0	0.0	0.0	0.000000e+00	Advance Auto Parts Inc	2021.0	1.010632e+10	10106321000.0

The data is not normally distributed

```
# Shapiro test for distribution
from scipy.stats import shapiro

# normality test
stat, p = shapiro(complete_df.cat_1_adj)
print('Statistics=%.3f, p=%.3f' % (stat, p))

# interpret
alpha = 0.05
if p > alpha:
    print('Sample looks normally distributed (fail to reject H0)')
else:
    print('Sample does not look normally distributed (reject H0)')
```

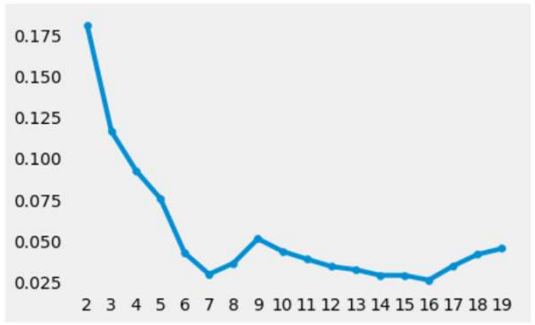
```
Statistics=0.462, p=0.000
Sample does not look normally distributed (reject H0)
```

```
from scipy.stats import normaltest

# normality test
stat, p = normaltest(complete_df.cat_1_adj)
print('Statistics=%.3f, p=%.3f' % (stat, p))

# interpret
alpha = 0.05
if p > alpha:
    print('Sample looks normally distributed (fail to reject H
else:
    print('Sample does not look normally distributed (reje
```

```
Statistics=511.901, p=0.000
Sample does not look normally distributed (reject H0)
```

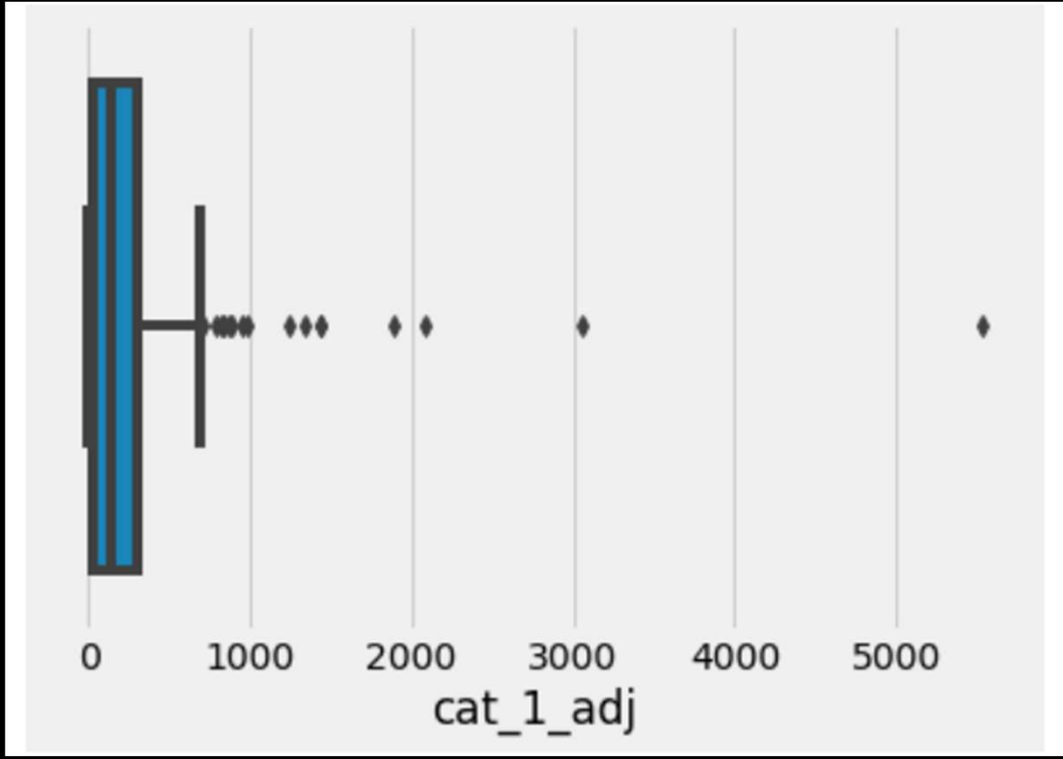
Modeling

Modeling



Based on the above chart the best model would use and N of 7

```
[ ] from sklearn.metrics import r2_score
    print(r2_score(y_test,preds))
    0.18106223746879946

[ ] from sklearn.metrics import explained_variance_score
    print(explained_variance_score(y_test,preds))
    0.20651954813138163
```



Modeling

```
Model Summary:
    number_of_trees    number_of_internal_trees
--  -----------------  --------------------------
    50                 50

ModelMetricsRegression: gbm
** Reported on train data. **

MSE: 34680.37773528339
RMSE: 186.22668373593348
MAE: 89.84306297704802
RMSLE: NaN
Mean Residual Deviance: 34680.37773528339
```