# Multi Regional Input Output model

#### Note this is only a proof of concept data is totally made up

import os  
import pandas as pd  
import numpy as np  
import copy  
#from openpyxl import load\_workbook  
   
cwd = os.getcwd()  
cwd = os.chdir(r'C:\Users\jojoseph\Desktop\Python Models\Python\IO')  
  
print (os.getcwd())

C:\Users\jojoseph\Desktop\Python Models\Python\IO

### Load the input data

##### Input data located in ‘reg\_soc\_acc\_mat’ It will need to be copied to your working directory

#opening the social account matrix excel files from IMPLAN -(89 sectors summarized from study area data/study area explorer )  
new\_MRIO= 'new\_MRIO.xlsx'# open file with social account matrices  
reg\_soc\_acc\_mat=pd.ExcelFile(new\_MRIO)  
print (reg\_soc\_acc\_mat.sheet\_names)  
  
#region3=reg\_soc\_acc\_mat.parse('region2')

['region0']

### Load aggregated sector names

#sector\_names = np.loadtxt('aggregated\_sector\_names.txt', usecols=1, delimiter="\'", dtype=str)[...,None]  
sector\_names = np.loadtxt('aggregated\_sector\_names.txt', usecols=1, delimiter="\'", dtype=str)  
  
sector\_names

array(['Agriculture', 'Mining', 'Construction', 'Durable Goods MFG',  
 'Nondurable Goods MFG', 'Trade, Transportaiton & Utilities',  
 'Information', 'Finance, Insurance & Real Estate',  
 'Professional & Business Services', 'Education & health Services',  
 'Leisure & Hospitality', 'Other Services', 'Government',  
 'Other Nonproducing Sectors'],  
 dtype='<U33')

## Read in reference matrices excel sheet

reference\_matrices = 'reference\_matrices.xlsx'  
ref\_mat=pd.ExcelFile(reference\_matrices)  
print (ref\_mat.sheet\_names)

['Sheet1', 'IndAggMat', 'ConvFact', 'MRSAM\_mult']

### Multi Region SAM multiplier

mrsam\_mult\_df = pd.read\_excel(ref\_mat,'MRSAM\_mult', skiprows=2, usecols = "C:UH")  
mrsam\_mult\_df

1

2

3

4

5

6

7

8

9

10

…

PI3

PI4

PI5

PI6

HH1

HH2

HH3

HH4

HH5

HH6

0

1.115473

0.007422

0.002036

0.003018

0.233903

0.001576

0.026853

0.000707

0

0.001407

…

0.000582

0.000807

0.000491

2.962485e-04

0.004782

0.000882

0.000604

0.000838

0.000510

0.000308

1

0.082254

1.050348

0.000741

0.023355

0.017605

0.057254

0.014521

0.015370

0

0.000346

…

0.000121

0.000189

0.000120

6.409873e-05

0.001122

0.000252

0.000126

0.000197

0.000124

0.000067

2

0.023870

0.016998

1.002141

0.001582

0.005645

0.011959

0.075809

0.005898

0

0.001944

…

0.000137

0.000222

0.000122

6.535571e-05

0.001905

0.000346

0.000142

0.000231

0.000126

0.000068

3

0.271561

0.002096

0.000820

1.009114

0.057339

0.000515

0.006717

0.000285

0

0.000604

…

0.000191

0.000270

0.000138

8.221320e-05

0.001935

0.001100

0.000198

0.000280

0.000143

0.000085

4

0.006459

0.003171

0.004560

0.005610

1.083815

0.001907

0.013748

0.001371

0

0.003210

…

0.000875

0.001302

0.000600

3.550982e-04

0.011055

0.001244

0.000909

0.001351

0.000623

0.000369

5

0.017214

0.001380

0.001754

0.006918

0.004537

1.009796

0.014923

0.003683

0

0.001286

…

0.000267

0.000572

0.000347

1.362379e-04

0.004232

0.000649

0.000277

0.000594

0.000360

0.000141

6

0.026051

0.005356

0.007505

0.011918

0.011387

0.012447

1.091851

0.002326

0

0.005321

…

0.000434

0.001048

0.000455

2.062243e-04

0.018267

0.001918

0.000451

0.001088

0.000473

0.000214

7

0.001722

0.001313

0.001927

0.000758

0.001175

0.000616

0.000802

1.066433

0

0.001343

…

0.000016

0.000038

0.000025

1.569124e-05

0.004713

0.000044

0.000016

0.000040

0.000026

0.000016

8

0.000901

0.001139

0.001703

0.000271

0.000877

0.000535

0.000683

0.000488

1

0.001172

…

0.000010

0.000024

0.000020

6.026703e-06

0.004166

0.000138

0.000011

0.000025

0.000021

0.000006

9

0.000786

0.003396

0.000171

0.000119

0.000290

0.000269

0.000199

0.000546

0

1.056485

…

0.000005

0.000013

0.000007

2.903939e-06

0.000363

0.000014

0.000006

0.000013

0.000007

0.000003

10

0.000061

0.000119

0.000038

0.000018

0.000054

0.000033

0.000081

0.000542

0

0.001380

…

0.000004

0.000005

0.000005

2.901632e-06

0.000089

0.000010

0.000004

0.000005

0.000006

0.000003

11

0.000110

0.001020

0.000017

0.000033

0.000034

0.000065

0.000033

0.000101

0

0.000530

…

0.000001

0.000002

0.000002

9.956247e-07

0.000020

0.000003

0.000001

0.000002

0.000002

0.000001

12

0.000084

0.000196

0.000070

0.000047

0.000076

0.000052

0.000105

0.000070

0

0.000124

…

0.000015

0.000018

0.000021

1.266344e-05

0.000159

0.000030

0.000016

0.000019

0.000022

0.000013

13

0.003649

0.000920

0.000449

0.000251

0.001604

0.000548

0.000804

0.000472

0

0.002653

…

0.000015

0.000033

0.000022

1.207173e-05

0.001072

0.000038

0.000015

0.000034

0.000023

0.000013

14

0.021319

0.032809

0.004830

0.002345

0.009574

0.004901

0.005007

0.003542

0

0.010209

…

0.000171

0.000302

0.000219

1.195468e-04

0.010802

0.000771

0.000178

0.000313

0.000227

0.000124

15

0.066084

0.104604

0.014226

0.006566

0.026143

0.013457

0.013130

0.009866

0

0.023118

…

0.000399

0.000685

0.000473

2.707908e-04

0.031663

0.001748

0.000414

0.000711

0.000491

0.000281

16

0.000994

0.001331

0.000475

0.000221

0.000727

0.000648

0.001119

0.001145

0

0.001662

…

0.000024

0.000033

0.000028

2.336864e-05

0.001122

0.000062

0.000025

0.000035

0.000029

0.000024

17

0.002877

0.010425

0.001137

0.001491

0.002210

0.001511

0.002165

0.003287

0

0.002824

…

0.000132

0.000148

0.000143

1.065146e-04

0.001922

0.000223

0.000137

0.000153

0.000148

0.000111

18

0.004218

0.001406

0.002020

0.000921

0.001731

0.000654

0.000905

0.000595

0

0.001424

…

0.000034

0.000070

0.000076

2.257031e-05

0.004933

0.000172

0.000035

0.000072

0.000079

0.000023

19

0.005011

0.060440

0.000493

0.001468

0.001144

0.003428

0.000961

0.000940

0

0.000096

…

0.000017

0.000022

0.000018

1.117224e-05

0.000219

0.000028

0.000018

0.000023

0.000018

0.000012

20

0.011293

0.050822

0.002511

0.002373

0.008065

0.005273

0.007144

0.003959

0

0.004188

…

0.000262

0.000294

0.000352

2.142539e-04

0.004914

0.000477

0.000272

0.000305

0.000366

0.000223

21

0.003485

0.005876

0.001938

0.001979

0.018338

0.006683

0.012544

0.007879

0

0.010264

…

0.000135

0.000224

0.000141

8.641271e-05

0.004573

0.000336

0.000140

0.000232

0.000147

0.000090

22

0.000540

0.000567

0.000241

0.000181

0.000496

0.000439

0.000713

0.002049

0

0.000392

…

0.000041

0.000063

0.000043

2.844977e-05

0.000563

0.000076

0.000043

0.000066

0.000045

0.000030

23

0.000712

0.000974

0.000477

0.000184

0.000572

0.000322

0.000431

0.012376

0

0.000752

…

0.000023

0.000048

0.000032

1.525851e-05

0.001130

0.000059

0.000024

0.000050

0.000033

0.000016

24

0.000960

0.000812

0.000685

0.000940

0.005338

0.002220

0.004053

0.002612

0

0.001279

…

0.000140

0.000178

0.000114

8.575875e-05

0.001640

0.000330

0.000146

0.000184

0.000118

0.000089

25

0.002523

0.001584

0.001081

0.002875

0.019120

0.006195

0.015281

0.009841

0

0.001564

…

0.000141

0.000178

0.000123

7.032236e-05

0.002593

0.000418

0.000146

0.000185

0.000128

0.000073

26

0.001244

0.001350

0.000894

0.000333

0.001548

0.000710

0.001240

0.000810

0

0.001941

…

0.000028

0.000054

0.000040

1.588696e-05

0.002156

0.000121

0.000029

0.000056

0.000042

0.000017

27

0.000576

0.001092

0.000589

0.000331

0.000680

0.000841

0.000513

0.000382

0

0.001483

…

0.000043

0.000084

0.000067

3.769068e-05

0.001412

0.000056

0.000045

0.000087

0.000069

0.000039

28

0.000997

0.001607

0.000758

0.000260

0.000881

0.000527

0.002096

0.019761

0

0.001517

…

0.000029

0.000051

0.000041

2.039202e-05

0.001808

0.000098

0.000030

0.000053

0.000043

0.000021

29

0.001636

0.002029

0.000590

0.001029

0.001423

0.000779

0.006463

0.002142

0

0.006223

…

0.000141

0.000181

0.000162

1.004711e-04

0.001347

0.000373

0.000146

0.000188

0.000168

0.000104

…

…

…

…

…

…

…

…

…

…

…

…

…

…

…

…

…

…

…

…

…

…

522

0.003792

0.002830

0.002936

0.004255

0.005317

0.006892

0.006459

0.002196

0

0.002445

…

0.004292

0.005073

0.008899

2.346608e-02

0.005082

0.004695

0.004458

0.005264

0.009242

0.024372

523

0.004890

0.004284

0.004598

0.004872

0.006872

0.007225

0.006766

0.002796

0

0.004206

…

0.008313

0.007043

0.006683

1.383806e-02

0.009385

0.007999

0.008634

0.007309

0.006940

0.014372

524

0.000765

0.000699

0.000878

0.000679

0.000944

0.001031

0.001017

0.000423

0

0.000668

…

0.001229

0.001052

0.000968

2.908885e-03

0.001861

0.001347

0.001277

0.001092

0.001006

0.003021

525

0.004617

0.004027

0.004790

0.004280

0.006067

0.006892

0.006670

0.002618

0

0.003751

…

0.005812

0.004905

0.008106

1.983040e-02

0.009803

0.006089

0.006036

0.005090

0.008418

0.020596

526

0.007478

0.006816

0.007899

0.007468

0.009825

0.010063

0.009627

0.004166

0

0.006449

…

0.015440

0.015317

0.015453

1.860017e-02

0.016362

0.015437

0.016035

0.015894

0.016048

0.019318

527

0.017266

0.013224

0.013768

0.018916

0.025338

0.030958

0.028219

0.009685

0

0.011878

…

0.018762

0.018067

0.015978

8.820766e-02

0.025210

0.015946

0.019486

0.018747

0.016594

0.091612

528

0.006069

0.004536

0.004287

0.007306

0.009508

0.012076

0.010602

0.003582

0

0.003925

…

0.006484

0.006590

0.005583

2.093204e-02

0.007631

0.005720

0.006734

0.006838

0.005798

0.021740

529

0.005260

0.004478

0.005358

0.005055

0.006754

0.007982

0.007546

0.002910

0

0.004230

…

0.007365

0.006421

0.004086

2.371863e-02

0.010927

0.006395

0.007649

0.006663

0.004243

0.024634

530

0.008304

0.006863

0.007755

0.008718

0.011379

0.013128

0.012342

0.004567

0

0.006097

…

0.013242

0.013648

0.012276

3.577460e-02

0.014328

0.011847

0.013753

0.014162

0.012749

0.037155

531

0.000348

0.000252

0.000271

0.000402

0.000475

0.000621

0.000572

0.000191

0

0.000221

…

0.000660

0.000572

0.000307

2.139143e-03

0.000478

0.000322

0.000685

0.000594

0.000319

0.002222

532

0.004976

0.003576

0.003355

0.005790

0.007391

0.010624

0.008112

0.002595

0

0.003013

…

0.004542

0.004430

0.004047

1.268942e-02

0.005337

0.004072

0.004717

0.004597

0.004203

0.013179

533

0.003733

0.002523

0.002069

0.004909

0.003853

0.005964

0.003961

0.002237

0

0.002325

…

0.003497

0.003356

0.002897

5.630594e-03

0.003951

0.003253

0.003632

0.003482

0.003008

0.005848

534

0.211105

0.208128

0.191150

0.085573

0.269038

0.163948

0.193643

0.154669

0

0.383858

…

0.002373

0.005684

0.003820

1.374736e-03

0.295193

0.008851

0.002465

0.005898

0.003967

0.001428

535

0.011677

0.012597

0.010758

0.008988

0.028580

0.015028

0.014126

0.008351

0

0.032360

…

0.012752

0.006085

0.005481

1.262157e-03

0.016914

0.344659

0.013244

0.006314

0.005692

0.001311

536

0.011838

0.004383

0.003639

0.030775

0.007856

0.006554

0.007217

0.002922

0

0.003343

…

0.317130

0.039845

0.002454

1.803673e-03

0.004913

0.013991

0.329365

0.041345

0.002549

0.001873

537

0.065828

0.032158

0.029004

0.157128

0.044234

0.039952

0.034783

0.022565

0

0.023088

…

0.083380

0.375494

0.003766

4.127731e-03

0.035545

0.023047

0.086597

0.389628

0.003911

0.004287

538

0.038281

0.053775

0.038690

0.027266

0.058400

0.043947

0.065661

0.043195

0

0.038277

…

0.035561

0.025397

0.465560

1.600859e-02

0.075721

0.096947

0.036933

0.026353

0.483502

0.016626

539

0.228363

0.165972

0.169113

0.258242

0.328479

0.419359

0.378294

0.132603

0

0.147138

…

0.249567

0.254007

0.225730

6.955669e-01

0.300664

0.222912

0.259195

0.263568

0.234429

0.722412

540

0.100878

0.180909

0.378850

0.009940

0.040547

0.024720

0.045924

0.018371

0

0.032603

…

0.000471

0.001125

0.000649

2.626350e-04

0.046405

0.001568

0.000489

0.001168

0.000674

0.000273

541

0.003775

0.004751

0.007274

0.002479

0.007231

0.005639

0.005169

0.001742

0

0.006694

…

0.003134

0.001520

0.001256

2.996093e-04

0.003580

0.070142

0.003255

0.001578

0.001304

0.000311

542

0.003258

0.000945

0.000893

0.008916

0.002076

0.001793

0.002246

0.000594

0

0.000612

…

1.056336

0.007008

0.000518

3.631151e-04

0.000924

0.002611

0.058510

0.007272

0.000538

0.000377

543

0.018416

0.008788

0.010877

0.044880

0.010196

0.011482

0.010704

0.003983

0

0.003805

…

0.014494

1.061179

0.000691

7.600705e-04

0.006202

0.003567

0.015054

0.063482

0.000717

0.000789

544

0.009879

0.013433

0.011772

0.007372

0.014777

0.013533

0.025508

0.010255

0

0.008249

…

0.009673

0.006821

1.105398

3.761450e-03

0.016770

0.022484

0.010046

0.007078

0.109460

0.003907

545

0.051082

0.032565

0.038714

0.058126

0.068261

0.109981

0.105942

0.024251

0

0.026011

…

0.045840

0.045807

0.038454

1.117040e+00

0.053694

0.040015

0.047609

0.047531

0.039936

0.121557

546

0.285034

0.360750

0.539536

0.085242

0.277095

0.168869

0.215830

0.154465

0

0.370638

…

0.002555

0.006117

0.004006

1.470187e-03

1.320183

0.009345

0.002654

0.006348

0.004160

0.001527

547

0.013828

0.015580

0.016450

0.010226

0.031881

0.018559

0.017317

0.008953

0

0.034637

…

0.014134

0.006769

0.005986

1.388626e-03

0.018184

1.381674

0.014680

0.007024

0.006216

0.001442

548

0.013721

0.004824

0.004111

0.036106

0.009021

0.007585

0.008619

0.003181

0

0.003573

…

1.315340

0.042311

0.002691

1.960161e-03

0.005275

0.015003

1.366086

0.043904

0.002795

0.002036

549

0.076045

0.036945

0.036205

0.182416

0.048971

0.046451

0.041132

0.023791

0

0.024078

…

0.087693

1.367859

0.003996

4.382780e-03

0.037406

0.023810

0.091076

1.419345

0.004150

0.004552

550

0.044408

0.061948

0.046626

0.031957

0.067460

0.053120

0.084525

0.049238

0

0.042813

…

0.041737

0.029722

1.501278

1.821000e-02

0.085138

0.109991

0.043347

0.030841

1.559135

0.018913

551

0.254977

0.180854

0.189688

0.288692

0.361659

0.484055

0.443238

0.142765

0

0.157535

…

0.268886

0.272840

0.240261

1.717049e+00

0.322440

0.239259

0.279260

0.283110

0.249521

1.783319

552 rows × 552 columns

## Get the Change in Industry Demand

### extract the first data column (sales) from the users data (Industry spending)

#IndustrySpending  
IndustrySpending = pd.read\_excel(new\_MRIO,'region0', usecols = "C")

### multiply it against the MRSAM\_mult multipliers

IndustryOutput = np.matmul(mrsam\_mult\_df,IndustrySpending)  
IndustryOutput

array([[ 2.41337005e+09], [ 1.37289862e+09], [ 6.57417105e+09], [ 1.36317602e+09], [ 6.17539706e+09], [ 1.01885896e+10], [ 2.66891287e+10], [ 1.38736657e+10], [ 8.56466157e+09], [ 7.76391829e+07], [ 3.42519227e+07], [ 1.10941651e+07], [ 7.77848617e+07], [ 1.86482219e+08], [ 2.66919132e+10], [ 1.14166827e+11], [ 2.70608550e+09], [ 1.28692542e+09], [ 2.73139174e+08], [ 1.18600826e+08], [ 2.17183711e+09], [ 2.36061849e+09], [ 5.44715068e+08], [ 4.14482559e+08], [ 6.06394840e+08], [ 1.15442020e+09], [ 4.47845473e+08], [ 2.39489560e+08], [ 7.69438257e+08], [ 1.18691658e+09], [ 2.61618637e+09], [ 8.65154042e+08], [ 3.92816970e+09], [ 3.08850768e+09], [ 4.02687316e+07], [ 1.37128741e+08], [ 4.35026005e+08], [ 2.26458357e+10], [ 4.67268006e+07], [ 1.03877554e+08], [ 5.15228922e+07], [ 5.14989269e+09], [ 1.71697847e+09], [ 7.17309756e+06], [ 6.08376145e+09], [ 6.26686497e+09], [ 3.53773072e+08], [ 6.59393539e+08], [ 1.79206394e+08], [ 2.08747804e+09], [ 1.25108124e+08], [ 1.18458506e+09], [ 3.73436401e+08], [ 6.47833308e+08], [ 3.97225107e+08], [ 3.60001718e+08], [ 4.68417752e+09], [ 2.69297336e+08], [ 4.29979157e+08], [ 2.65038194e+09], [ 3.45350703e+08], [ 1.58437559e+08], [ 4.29240864e+09], [ 1.77355507e+09], [ 3.07460481e+09], [ 4.21201134e+08], [ 1.27615135e+10], [ 8.53997298e+08], [ 7.88976405e+08], [ 7.60321639e+09], [ 3.70659484e+09], [ 3.34765847e+09], [ 5.91134051e+08], [ 1.55054929e+09], [ 4.33571505e+09], [ 4.17783130e+09], [ 1.12637571e+09], [ 8.81559350e+08], [ 9.48664943e+08], [ 5.60051724e+07], [ 8.69777007e+09], [ 9.91581666e+09], [ 5.40982448e+10], [ 6.60649168e+09], [ 7.22923613e+08], [ 1.09368959e+09], [ 5.76229106e+07], [ 8.28089873e+08], [ 2.27507697e+08], [ 7.54723628e+08], [ 2.72353460e+08], [ 3.68072771e+08], [ 1.00598658e+08], [ 4.28113823e+08], [ 2.33094332e+08], [ 7.12566257e+08], [ 1.28670170e+08], [ 4.88841797e+07], [ 3.21897523e+07], [ 3.07182856e+07], [ 1.08620280e+07], [ 4.93490259e+07], [ 1.09240321e+08], [ 4.61189265e+09], [ 3.16911216e+09], [ 2.01127445e+08], [ 3.93355986e+08], [ 2.67893005e+08], [ 7.86910673e+07], [ 8.52339124e+08], [ 4.38198170e+08], [ 8.12893848e+07], [ 7.40547122e+07], [ 3.14317116e+08], [ 2.02745311e+08], [ 9.60096695e+07], [ 7.12220304e+07], [ 1.83748526e+08], [ 6.15019258e+08], [ 7.06186382e+08], [ 2.50648743e+08], [ 4.37041529e+08], [ 3.28882275e+08], [ 1.56429460e+07], [ 5.31835583e+07], [ 8.19403689e+07], [ 5.40678840e+07], [ 3.44784511e+07], [ 3.37199942e+07], [ 1.86184346e+08], [ 1.53318913e+09], [ 1.06931174e+09], [ 2.31806286e+06], [ 1.40445618e+09], [ 1.82956752e+09], [ 3.37033326e+08], [ 2.10969516e+08], [ 2.42378638e+07], [ 5.09312246e+08], [ 5.13711593e+07], [ 3.05846751e+08], [ 9.98797569e+07], [ 1.06200363e+08], [ 1.34779945e+08], [ 1.09710570e+08], [ 1.82378044e+08], [ 7.68258839e+07], [ 1.71808747e+08], [ 1.05658406e+09], [ 8.22623945e+07], [ 4.79920511e+07], [ 1.58155605e+09], [ 7.08768974e+08], [ 8.70059551e+08], [ 8.21002879e+07], [ 3.05367555e+09], [ 3.78540845e+08], [ 6.68883275e+08], [ 2.45311645e+09], [ 1.04819593e+09], [ 1.02994346e+09], [ 1.53859016e+08], [ 3.16342836e+08], [ 1.21730145e+09], [ 1.01149130e+09], [ 2.38813195e+08], [ 1.72228420e+08], [ 1.25845917e+08], [ 1.57180581e+07], [ 1.50174742e+08], [ 1.37353154e+08], [ 9.98717010e+08], [ 3.38627450e+08], [ 2.02523414e+08], [ 3.38952131e+08], [ 2.15107513e+07], [ 2.37210488e+08], [ 1.23101065e+08], [ 4.15494374e+08], [ 2.04939977e+08], [ 3.27102230e+08], [ 1.24845151e+08], [ 1.19884595e+08], [ 1.86788954e+08], [ 5.52760914e+08], [ 4.13418189e+07], [ 4.04191736e+07], [ 1.52954892e+07], [ 1.87527538e+07], [ 4.78533905e+06], [ 0.00000000e+00], [ 4.25790951e+07], [ 5.20957493e+08], [ 1.03572863e+09], [ 6.27754028e+07], [ 3.22475389e+08], [ 9.24338627e+07], [ 5.48291722e+07], [ 4.62871593e+08], [ 2.28377666e+08], [ 2.21459463e+07], [ 3.62448152e+07], [ 1.29519965e+08], [ 1.19591165e+08], [ 8.77121542e+07], [ 2.10254591e+08], [ 7.82589784e+07], [ 1.66032760e+08], [ 2.98491998e+08], [ 1.01915934e+08], [ 3.25120926e+08], [ 2.44032251e+08], [ 7.94557242e+07], [ 5.42327716e+07], [ 4.74425593e+07], [ 2.43654120e+07], [ 3.45623772e+07], [ 4.03943582e+07], [ 8.97080019e+07], [ 6.07255833e+08], [ 5.17713973e+08], [ 2.73184458e+06], [ 7.10931677e+08], [ 7.81031583e+08], [ 3.84895003e+07], [ 1.04846177e+08], [ 1.70072059e+07], [ 2.44279065e+08], [ 2.46203715e+07], [ 6.59679904e+07], [ 9.85176197e+06], [ 4.78662255e+07], [ 7.11547762e+07], [ 5.89011241e+07], [ 1.16131223e+08], [ 3.94530625e+07], [ 6.37302951e+07], [ 6.09902504e+08], [ 4.94847444e+07], [ 2.46664540e+07], [ 7.58152515e+08], [ 4.10910982e+08], [ 5.62007615e+08], [ 6.56146601e+07], [ 1.45351803e+09], [ 1.34226766e+08], [ 3.28529729e+08], [ 1.21892239e+09], [ 4.71247514e+08], [ 5.12083420e+08], [ 7.38249018e+07], [ 1.79869675e+08], [ 5.74449676e+08], [ 5.06584490e+08], [ 1.66350221e+08], [ 1.10815733e+08], [ 5.57836051e+07], [ 6.29645095e+06], [ 6.58494579e+07], [ 7.57364591e+07], [ 4.16844764e+08], [ 1.52269480e+08], [ 9.17876684e+07], [ 1.55518563e+08], [ 6.23181296e+06], [ 1.10824911e+08], [ 6.26795628e+07], [ 1.19393314e+09], [ 1.28412842e+09], [ 1.27872584e+09], [ 6.53945014e+08], [ 6.25334007e+08], [ 8.18466171e+08], [ 2.29873143e+09], [ 4.77181910e+08], [ 8.58805871e+07], [ 9.63936695e+07], [ 2.70698834e+07], [ 5.01780223e+06], [ 1.30097059e+08], [ 1.63358262e+08], [ 1.01848329e+10], [ 5.04944159e+09], [ 6.65314253e+08], [ 2.17535462e+09], [ 4.79749664e+08], [ 2.11830222e+08], [ 1.68501940e+09], [ 9.62161942e+08], [ 1.38429652e+08], [ 2.03357239e+08], [ 4.78319090e+08], [ 6.01806985e+08], [ 3.26096939e+08], [ 1.24360207e+08], [ 4.43768643e+08], [ 8.28599453e+08], [ 1.64482479e+09], [ 4.30665713e+08], [ 9.55874444e+08], [ 7.52525049e+08], [ 7.14131991e+07], [ 1.10856252e+08], [ 1.91490233e+08], [ 2.10382225e+08], [ 2.82527705e+08], [ 1.50987270e+08], [ 5.71351253e+08], [ 2.91238253e+09], [ 3.41625019e+09], [ 1.73700025e+07], [ 3.38945389e+09], [ 3.88303674e+09], [ 2.52321049e+08], [ 5.07758538e+08], [ 1.45712222e+08], [ 1.19059220e+09], [ 1.90882476e+08], [ 5.47004691e+08], [ 1.77344947e+08], [ 2.59774190e+08], [ 3.35548709e+08], [ 2.57722919e+08], [ 6.19890934e+08], [ 2.28557571e+08], [ 6.23913013e+08], [ 2.57963088e+09], [ 8.33363944e+08], [ 1.06288932e+08], [ 3.89262154e+09], [ 2.21231575e+09], [ 3.38884632e+09], [ 4.72911795e+08], [ 7.62565379e+09], [ 6.94713348e+08], [ 2.46184325e+09], [ 6.57331484e+09], [ 2.66020094e+09], [ 2.63430775e+09], [ 3.53471728e+08], [ 9.96612418e+08], [ 2.47112519e+09], [ 2.75341666e+09], [ 6.81210269e+08], [ 4.23675059e+08], [ 4.59655698e+08], [ 4.69402674e+07], [ 5.75361046e+08], [ 3.45014342e+08], [ 2.15472576e+09], [ 7.62806864e+08], [ 4.82774756e+08], [ 8.76505084e+08], [ 3.08365871e+07], [ 5.59199760e+08], [ 3.19038316e+08], [ 2.47752045e+09], [ 7.95172902e+08], [ 1.89917788e+09], [ 2.73976246e+08], [ 1.63702035e+09], [ 1.33678996e+09], [ 4.62413283e+09], [ 9.29552295e+08], [ 3.00574306e+08], [ 1.83556250e+08], [ 8.14744868e+07], [ 3.92305091e+07], [ 3.21157987e+08], [ 4.29297901e+08], [ 7.27721140e+09], [ 1.28729351e+10], [ 6.97007475e+08], [ 1.19130944e+10], [ 1.12998217e+09], [ 4.30979937e+08], [ 1.75077598e+09], [ 2.54644994e+09], [ 3.50952008e+08], [ 5.76187629e+08], [ 8.40805310e+08], [ 1.29289469e+09], [ 5.90750127e+08], [ 4.03208226e+08], [ 1.19970556e+09], [ 3.39123816e+09], [ 4.95783519e+09], [ 1.47808477e+09], [ 3.32212055e+09], [ 4.29439214e+09], [ 1.23307388e+08], [ 3.10332611e+08], [ 6.55355952e+08], [ 6.27833659e+08], [ 3.10790118e+08], [ 1.60944832e+08], [ 1.74096012e+09], [ 7.45184389e+09], [ 8.72051452e+09], [ 2.83097238e+07], [ 9.40135113e+09], [ 1.06422466e+10], [ 1.72111684e+09], [ 9.35802171e+08], [ 2.69493077e+08], [ 2.72312182e+09], [ 3.70670632e+08], [ 7.84706118e+08], [ 4.62184611e+08], [ 6.30467922e+08], [ 8.79924244e+08], [ 6.65471727e+08], [ 1.01604532e+09], [ 5.23732549e+08], [ 7.59878594e+08], [ 5.82023021e+09], [ 1.02378808e+09], [ 3.00804531e+08], [ 1.04930183e+10], [ 7.66574102e+09], [ 6.83882449e+09], [ 2.50481914e+09], [ 2.32102829e+10], [ 1.86983354e+09], [ 6.40585428e+09], [ 1.73910351e+10], [ 6.54855506e+09], [ 6.72631298e+09], [ 1.01544586e+09], [ 2.20676532e+09], [ 7.67056253e+09], [ 5.78311416e+09], [ 1.23724509e+09], [ 9.06358896e+08], [ 8.86314300e+08], [ 1.23122623e+08], [ 7.87255011e+08], [ 9.77135827e+08], [ 6.16163699e+09], [ 2.10687223e+09], [ 1.44135623e+09], [ 2.05025807e+09], [ 1.51500041e+08], [ 1.22928982e+09], [ 9.07180686e+08], [ 1.41136221e+10], [ 8.51549925e+09], [ 1.04322081e+10], [ 4.47523663e+09], [ 8.28428726e+09], [ 9.78164475e+09], [ 2.60579601e+10], [ 2.96616137e+09], [ 1.55008679e+09], [ 5.11967476e+08], [ 2.44401368e+08], [ 2.44812589e+08], [ 1.98076420e+09], [ 1.52182305e+09], [ 1.98929671e+10], [ 3.93278912e+10], [ 2.30551095e+09], [ 1.63101572e+10], [ 1.13620610e+10], [ 1.65836499e+09], [ 2.01010277e+10], [ 1.05618202e+10], [ 2.26014192e+09], [ 1.87803174e+09], [ 6.75829730e+09], [ 7.06069553e+09], [ 4.61928888e+09], [ 3.49385937e+09], [ 3.52677884e+09], [ 1.58538302e+10], [ 1.55180319e+10], [ 4.16434944e+09], [ 1.82893885e+10], [ 1.04857622e+10], [ 7.25188378e+08], [ 2.38962821e+09], [ 2.92832847e+09], [ 4.00403309e+09], [ 1.29468592e+09], [ 1.10060285e+09], [ 1.50564650e+09], [ 2.71447024e+10], [ 1.49794105e+10], [ 2.91479591e+08], [ 3.45667736e+10], [ 4.11215324e+10], [ 4.30680214e+09], [ 4.41527979e+09], [ 1.57645915e+09], [ 1.16227682e+10], [ 1.92888791e+09], [ 1.90989425e+09], [ 2.73888554e+09], [ 2.85082263e+09], [ 3.91962697e+09], [ 3.00914683e+09], [ 1.13201054e+10], [ 4.75044431e+09], [ 8.93370598e+09], [ 2.28708809e+10], [ 3.83957879e+09], [ 2.97602174e+09], [ 3.79303412e+10], [ 2.83516088e+10], [ 3.00477170e+10], [ 5.54915751e+09], [ 8.83860254e+10], [ 5.75303919e+09], [ 7.72991779e+09], [ 5.90961576e+10], [ 2.32637255e+10], [ 3.03782170e+10], [ 4.73012041e+09], [ 1.36570842e+10], [ 3.05242810e+10], [ 2.81074125e+10], [ 7.00321121e+09], [ 5.19193075e+09], [ 6.26943308e+09], [ 7.78299132e+08], [ 5.24334700e+09], [ 7.84988587e+09], [ 2.34216927e+10], [ 7.94603748e+09], [ 6.14394894e+09], [ 9.95643125e+09], [ 4.51539357e+08], [ 5.82222393e+09], [ 3.57746209e+09], [ 6.36035979e+10], [ 1.00887203e+10], [ 4.92049009e+09], [ 2.27565735e+10], [ 6.07695613e+10], [ 2.61349269e+11], [ 1.12017886e+10], [ 2.44203808e+09], [ 1.04182106e+09], [ 4.73297685e+09], [ 1.48115807e+10], [ 5.06838774e+10], [ 6.70904568e+10], [ 1.11457453e+10], [ 5.39750013e+09], [ 2.46928107e+10], [ 6.96461230e+10], [ 2.84201535e+11]])

### Load the aggregation matrix

#aggregation\_matrix = np.loadtxt('aggregation\_matrix.txt')  
aggregation\_matrix = pd.read\_excel(ref\_mat,'IndAggMat', skiprows=2, usecols = "C:CM")  
aggregation\_matrix

1

2

3

4

5

6

7

8

9

10

…

80

81

82

83

84

85

86

87

88

89

0

1

1

1

0

0

0

0

0

0

0

…

0

0

0

0

0

0

0

0

0

0

1

0

0

0

0

0

0

0

0

0

1

…

0

0

0

0

0

0

0

0

0

0

2

0

0

0

0

0

0

0

0

0

0

…

0

0

0

0

0

0

0

0

0

0

3

0

0

0

0

0

0

0

0

0

0

…

0

0

0

0

0

0

0

0

0

0

4

0

0

0

1

1

1

1

1

1

0

…

0

0

0

0

0

0

0

0

0

0

5

0

0

0

0

0

0

0

0

0

0

…

0

0

0

0

0

0

0

0

0

0

6

0

0

0

0

0

0

0

0

0

0

…

0

0

0

0

0

0

0

0

0

0

7

0

0

0

0

0

0

0

0

0

0

…

0

0

0

0

0

0

0

0

0

0

8

0

0

0

0

0

0

0

0

0

0

…

0

0

0

0

0

0

0

0

0

0

9

0

0

0

0

0

0

0

0

0

0

…

0

0

0

0

0

0

0

0

0

0

10

0

0

0

0

0

0

0

0

0

0

…

1

1

1

1

0

0

0

0

0

0

11

0

0

0

0

0

0

0

0

0

0

…

0

0

0

0

1

1

1

1

0

0

12

0

0

0

0

0

0

0

0

0

0

…

0

0

0

0

0

0

0

0

1

0

13

0

0

0

0

0

0

0

0

0

0

…

0

0

0

0

0

0

0

0

0

1

14 rows × 89 columns

### Load the Impact Conversion Factors

impact\_conversion\_matrix = pd.read\_excel(ref\_mat,'ConvFact', skiprows=1, usecols = "O:S")

### Calculate the other values based on the impact conversion factors

sales = []  
Employment=[]  
Employee\_Compensation=[]  
Proprietors\_Income=[]  
Other\_Property = []  
Indirect\_Business\_Tax = []  
  
sal = IndustryOutput  
for k in range(0,89):  
 sales.append(sal[k][0])  
 Employment.append(float((sal[k][0] \* impact\_conversion\_matrix['Employment'][k])/1000000))  
 Employee\_Compensation.append(float((sal[k][0] \* impact\_conversion\_matrix['Employee Compensation'][k])))  
 Proprietors\_Income.append(float((sal[k][0] \* impact\_conversion\_matrix['Proprietors\' Income'][k])))  
 Other\_Property.append(float(sal[k][0] \* impact\_conversion\_matrix['Other Property Type Income'][k]))  
 Indirect\_Business\_Tax.append(float(sal[k][0] \* impact\_conversion\_matrix['Indirect Business Tax'][k]))  
  
#sales

### now multiply by the aggregation matrix.

aggregated\_dict={}  
aggregated\_dict['Industry']=sector\_names  
aggregated\_dict['Sales'] = np.matmul(aggregation\_matrix, sales)  
aggregated\_dict['Employment'] = np.matmul(aggregation\_matrix, Employment)  
aggregated\_dict['Employee\_Compensation'] = np.matmul(aggregation\_matrix, Employee\_Compensation)  
aggregated\_dict['Proprietors\_Income'] = np.matmul(aggregation\_matrix, Proprietors\_Income)  
aggregated\_dict['Other\_Property'] = np.matmul(aggregation\_matrix, Other\_Property)  
aggregated\_dict['Indirect\_Business\_Tax'] = np.matmul(aggregation\_matrix, Indirect\_Business\_Tax)  
#aggregated\_dict

### Calc Value Added and Labor Income and add to output

Value\_Added=[]  
Labor\_Income=[]  
  
for i in range(0,14):  
 Value\_Added.append(aggregated\_dict['Employee\_Compensation'][i] + aggregated\_dict['Proprietors\_Income'][i] + aggregated\_dict['Other\_Property'][i] + aggregated\_dict['Indirect\_Business\_Tax'][i] )  
 Labor\_Income.append(aggregated\_dict['Employee\_Compensation'][i] + aggregated\_dict['Proprietors\_Income'][i] )  
   
aggregated\_dict['Value\_Added'] = Value\_Added  
aggregated\_dict['Labor\_Income'] = Labor\_Income

### Build the output.

#### This will write out a spreadsheet to your working directory

!jupyter-nbconvert regMatrix.ipynb --to markdown  
!pandoc regMatrix.ipynb -t docx -o regMatrix.docx

This application is used to convert notebook files (\*.ipynb) to various other formats.

WARNING: THE COMMANDLINE INTERFACE MAY CHANGE IN FUTURE RELEASES.

Options

Arguments that take values are actually convenience aliases to full Configurables, whose aliases are listed on the help line. For more information on full configurables, see ‘–help-all’.

–debug

set log level to logging.DEBUG (maximize logging output)

–generate-config

generate default config file

-y

Answer yes to any questions instead of prompting.

–execute

Execute the notebook prior to export.

–allow-errors

Continue notebook execution even if one of the cells throws an error and include the error message in the cell output (the default behaviour is to abort conversion). This flag is only relevant if ‘–execute’ was specified, too.

–stdin

read a single notebook file from stdin. Write the resulting notebook with default basename ’notebook.\*’

–stdout

Write notebook output to stdout instead of files.

–inplace

Run nbconvert in place, overwriting the existing notebook (only relevant when converting to notebook format)

–clear-output

Clear output of current file and save in place, overwriting the existing notebook.

–no-prompt

Exclude input and output prompts from converted document. –log-level= (Application.log\_level)

Default: 30

Choices: (0, 10, 20, 30, 40, 50, ‘DEBUG’, ‘INFO’, ‘WARN’, ‘ERROR’, ‘CRITICAL’)

Set the log level by value or name.

–config= (JupyterApp.config\_file)

Default: ’’

Full path of a config file.

–to= (NbConvertApp.export\_format)

Default: ‘html’

The export format to be used, either one of the built-in formats, or a

dotted object name that represents the import path for an Exporter class

–template= (TemplateExporter.template\_file)

Default: ’’

Name of the template file to use

–writer= (NbConvertApp.writer\_class)

Default: ‘FilesWriter’

Writer class used to write the results of the conversion

–post= (NbConvertApp.postprocessor\_class)

Default: ’’

PostProcessor class used to write the results of the conversion

–output= (NbConvertApp.output\_base)

Default: ’’

overwrite base name use for output files. can only be used when converting

one notebook at a time.

–output-dir= (FilesWriter.build\_directory)

Default: ’’

Directory to write output(s) to. Defaults to output to the directory of each

notebook. To recover previous default behaviour (outputting to the current

working directory) use . as the flag value.

–reveal-prefix= (SlidesExporter.reveal\_url\_prefix)

Default: ’’

The URL prefix for reveal.js. This can be a a relative URL for a local copy

of reveal.js, or point to a CDN.

For speaker notes to work, a local reveal.js prefix must be used.

–nbformat= (NotebookExporter.nbformat\_version)

Default: 4

Choices: [1, 2, 3, 4]

The nbformat version to write. Use this to downgrade notebooks.

To see all available configurables, use --help-all

Examples ——–

The simplest way to use nbconvert is

jupyter nbconvert mynotebook.ipynb

which will convert mynotebook.ipynb to the default format (probably HTML).

You can specify the export format with --to. Options include [‘asciidoc’, ‘custom’, ‘html’, ‘html\_ch’, ‘html\_embed’, ‘html\_toc’, ‘html\_with\_lenvs’, ‘html\_with\_toclenvs’, ‘latex’, ‘latex\_with\_lenvs’, ‘markdown’, ‘notebook’, ‘pdf’, ‘python’, ‘rst’, ‘script’, ‘selectLanguage’, ‘slides’, ‘slides\_with\_lenvs’]

jupyter nbconvert –to latex mynotebook.ipynb

Both HTML and LaTeX support multiple output templates. LaTeX includes ‘base’, ‘article’ and ‘report’. HTML includes ‘basic’ and ‘full’. You can specify the flavor of the format used.

jupyter nbconvert –to html –template basic mynotebook.ipynb

You can also pipe the output to stdout, rather than a file

jupyter nbconvert mynotebook.ipynb –stdout

PDF is generated via latex

jupyter nbconvert mynotebook.ipynb –to pdf

You can get (and serve) a Reveal.js-powered slideshow

jupyter nbconvert myslides.ipynb –to slides –post serve

Multiple notebooks can be given at the command line in a couple of different ways:

jupyter nbconvert notebook\*.ipynb

jupyter nbconvert notebook1.ipynb notebook2.ipynb

or you can specify the notebooks list in a config file, containing::

c.NbConvertApp.notebooks = ["my\_notebook.ipynb"]  
   
 > jupyter nbconvert --config mycfg.py  
  
  
  
[NbConvertApp] WARNING | pattern 'regMatrix.ipynb' matched no files  
pandoc: regMatrix.ipynb: openBinaryFile: does not exist (No such file or directory)