**PANDAS**

1. Python merge

df.merge()

how=outer : union

how=inner: intersection

how=left: left join

how=right: right join

2. Multi-indexing, multiple columns

3. Pandas idioms

(df.where(df['SUMLEV']==50)

.dropna()

.set\_index(['STNAME','CTYNAME'])

.rename(columns={'ESTIMATESBASE2010': 'Estimates Base 2010'}))

4. drop

df.drop(df[df['Quantity'] == 0].index)

5. apply

import numpy as np

def min\_max(row):

data = row[['POPESTIMATE2010',

'POPESTIMATE2011',

'POPESTIMATE2012',

'POPESTIMATE2013',

'POPESTIMATE2014',

'POPESTIMATE2015']]

# Add two new columns in df called ‘max’ and ‘min’

row['max'] = np.max(data)

row['min'] = np.min(data)

return row

df.apply(min\_max, axis=1)

6. Group by

%%timeit -n 10

for state in df['STNAME'].unique():

avg = np.average(df.where(df['STNAME']==state).dropna()['CENSUS2010POP'])

print('Counties in state ' + state + ' have an average population of ' + str(avg))

**GROUPBY is huge faster than for loop (almost 50 times faster)**

%%timeit -n 10

for group, frame in df.groupby('STNAME'):

avg = np.average(frame['CENSUS2010POP'])

print('Counties in state ' + group + ' have an average population of ' + str(avg))

split-combine pattern

**String**

dicts = {"Republic of Korea": "South Korea",

"United States of America": "United States",

"United Kingdom of Great Britain and Northern Ireland": "United Kingdom",

"China, Hong Kong Special Administrative Region": "Hong Kong"}

dict\_num={"0":"","1":"","2":"","3":"","4":"","5":"","6":"","7":"","8":"","9":""}

# Remove numbers

for key\_num in dict\_num.keys():

energy.loc[index,'Country']=energy.loc[index,'Country'].replace(key\_num,"")

# Remove text in (...)

energy.loc[index,'Country']=re.sub("\(.\*\)", "",energy.loc[index,'Country'])

**CORR**

**Before compute Correlation of columns in DataFrame, convert the value of columns to float**

Top15['Citable docs per Capita']=Top15['Citable docs per Capita'].astype(float)

**TEST (STATISTICAL)**

**p-hacking**

**Format**

'{0:,}'.format(1000000.21312) # Add thousand separate comma