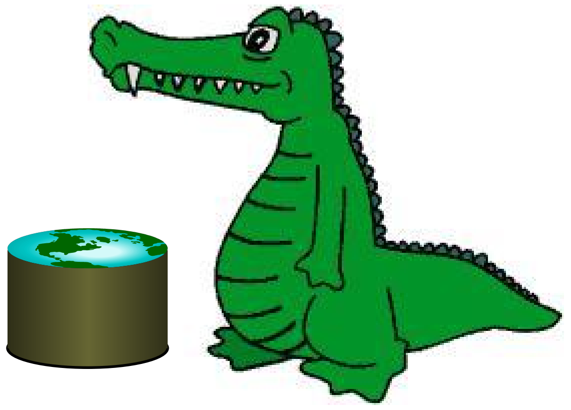
CAP4770/5771
Lab 1
Joining DataFrames in Pandas

University of Florida, CISE Department
TA: Xiaofeng Zhou



Quick Review: Pandas(Lab0-2)

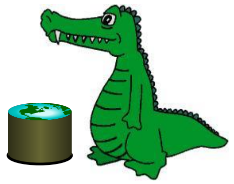
DataFrame: two dimensional data structures(like SQL table or spreadsheet)

SQL-like operators

- Row filters (selection)
- Column filters (project)
- Form groups (group by)

And more from Python:

1. lambda functions
2. Visualization: matplotlib



Goal

Join

- Join operation in Pandas
- Fuzzy join using string edit distance

Data analysis and visualization

- precision
- Recall

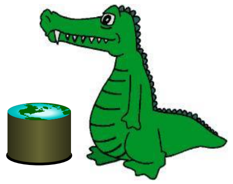
Join operation in Pandas

SQL:

Join combines records from two or more tables in a relational database

Dataframe in Pandas (equijoin):

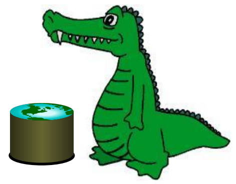
```
import pandas as pd
Students = pd.DataFrame({'student_id': [1, 2], 'name':
['Alice', 'Bob']})
Grades = pd.DataFrame({'student_id': [1, 1, 2, 2],
'class_id': [1, 2, 1, 3], 'grade': ['A', 'C', 'B', 'B']})
pd.merge(Students, Grades, on='student_id')
```



Fuzzy join using string edit distance

1. Levenshtein distance
   a. metric to measure difference between two string sequences
   b. e.g. kitten → sitting
      i. kitten → sitten (substitution of "s" for "k")
      ii. sitten → sittin (substitution of "i" for "e")
      iii. sittin → sitting (insertion of "g" at the end)

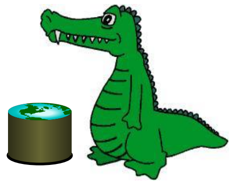
      The Levenshtein distance is 3



Fuzzy join using string edit distance - cont

2. Restaurant data set contains 4 fields:
    a. id: unique for each row
    b. cluster: indicating if two rows are duplicate, i.e. about the same restaurant
    c. name: name of the restaurant(not 100% accurate).
    d. city: location of the restaurant

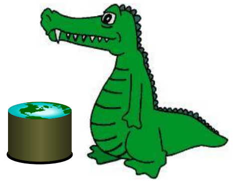
To find duplicate records: join the restaurant table with itself (self-join) on column ‘cluster’ (exact search, 100% correct)



Fuzzy join using string edit distance - cont

3. Fuzzy join and finding duplicate pair
    a. Cartesian product
        add a **dummy column** to enable product
    b. Add join criterion
        add a column to product table to store the Levenshtein distance of the names
    c. Filter the Cartesian product based on join criterion

This way we can also find duplicate records in the data set based on edit distance of names (not 100% correct)



Data analysis and visualization

To evaluate the fuzzy join accuracy in terms of finding duplicate pairs of records, we use

- Precision
  = (# correct duplicate pairs found) / (# all similar pairs found)
- Recall
  = (# correct duplicate pairs found) / (# all duplicate pairs)

(All duplicates pairs are stored in DataFrame ‘clusters’)

(All pairs with different L-distance are stored in ‘prod’)

Data analysis and visualization - cont

For different Levenshtein-distance thresholds as join filter criterion, visualize a “precision, recall-threshold” graph

1. Another way to visualize to show precision/recall trade-off: precision-recall graph. (DIY4)
2. There is another metric - Levenshtein. ratio that can also be used. Try to compare the two metrics used in the restaurant dataset. (DIY5)