THE UNIVERSITY OF MELBOURNE

Workshop 6

COMP20008

Elements of Data Processing

Zijie Xu

Agenda

- Data linkage
  - Evaluating performance of matches
  - Blocking
- Theory questions

Data Linkage

- Task: Given two lists of entities/records A & B, we want to match those from A to their equivalent in B
- Problem: Same entity has different representation in different datasets
- Steps
  - Pre-process features
  - Obtain an index tuple (idA, idB) for comparison
  - Score each index tuple with similarity measures
  - Determine matches from the similarity scores

Evaluating performance

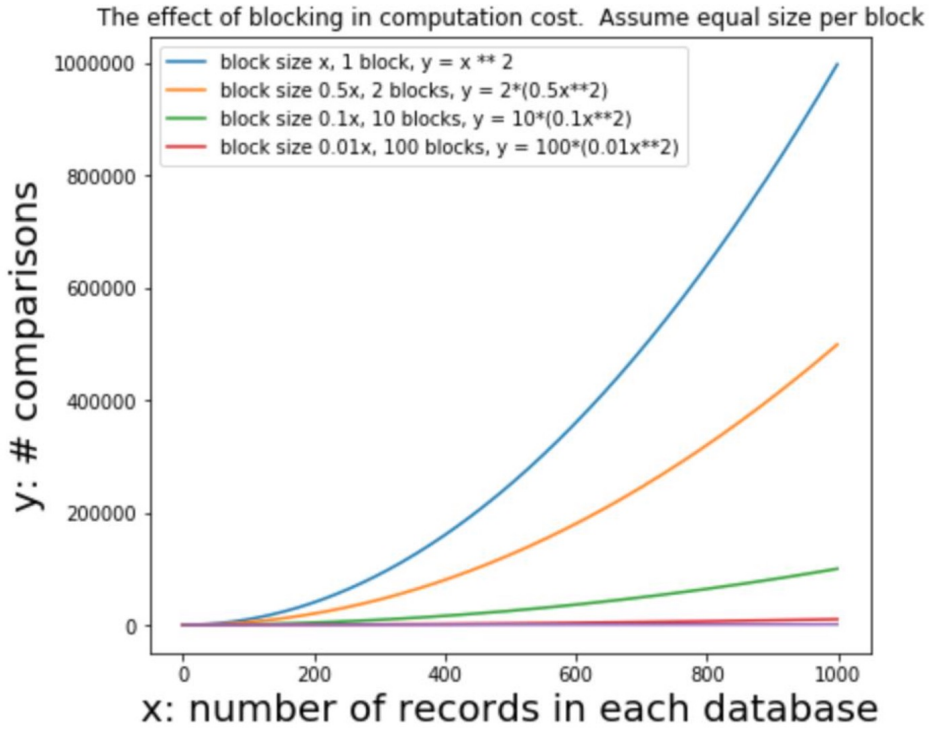
- Confusion matrix
- Some metrics:
  - Accuracy $\frac{TP+TN}{P+N}$
  - Precision $\frac{TP}{TP+FP}$
  - Recall $\frac{TP}{TP+FN}$
  - Many other metrics available

Total = P+N

		Predicted ($\hat{Y}$)	
		$\hat{Y} = +1$	$\hat{Y} = -1$
Actual (Y)	Y = P	TP	FN
	Y = N	FP	TN

Blocking

- Problem: Exhaustive comparison for matches is not be feasible for large datasets (m records in A, n records in B -> m*n comparisons)
- Task: Develop a strategy to group the records, only records from the same group are compared for matches
- While blocking improves computation efficiency, it may have performance tradeoff



Blocking

- Some performacne metrics for blocking:
  - Pair-completeness (PC)
  - PC = Recall = $\frac{TP}{TP+FN}$
  - Reduction Ratio (RR)
  - RR = $1 - \frac{TP+FP}{P+N}$

    $= 1 - \frac{\text{\# pairs compared}}{\text{Total \# pairs}}$

Total = P+N

Actual (Y)	Predicted ($\hat{Y}$)	
	$\hat{Y} = +1$	$\hat{Y} = -1$
Y = P	TP	FN
Y = N	FP	TN

Theoretical questions

- Suppose you are conducting data linkage between two databases, one with m records and the other with n records (assume $m \leq n$). Under a basic (exhaustive) approach, $m \times n$ record comparisons will be needed.
- Assuming that there are no duplicates, what is the maximum number of record matches?
  - m

Theoretical questions

- Suppose you are conducting data linkage between two databases, one with m records and the other with n records (assume $m \leq n$). Under a basic (exhaustive) approach, $m \times n$ record comparisons will be needed.
- Assuming that there are no duplicates, what is the corresponding number of non-matching comparisons required in this circumstance?
  - $n \times m - m = m(n - 1)$

Theoretical questions

- Now suppose a blocking method is employed, where each record is assigned to exactly one block. Assume this method results in *b* number of blocks.
- What is the smallest possible number of comparisons? What is the value of *b* in this scenario?
  - 2 blocks, each dataset maps to a different block
  - No comparison

Theoretical questions

- Now suppose a blocking method is employed, where each record is assigned to exactly one block. Assume this method results in b number of blocks.
- What is the largest possible number of comparisons? What is the value of b in this scenario?
  - 1 single block with all entries
  - $m \times n$ comparison

Theoretical questions

- Now suppose a blocking method is employed, where each record is assigned to exactly one block. Assume this method results in *b* number of blocks.
- What is the advantage of large b? What is the advantage of small b?
  - Trade-off between performance and efficiency
  - Large *b* -> faster comparison but maybe inaccurate
  - Small *b* -> more accurate result but slower

Theoretical questions

- In practice, a record is assigned to more than one block and records are not evenly allocated to blocks
- For example, `alissa kilmartin` is assigned to blocks with `surname_prefix==kilm` and `given_name_prefix==alis`
- How would this affect your answer to previous question?
  - The dominant block size imposes bottleneck for efficiency

Theoretical questions

- In practice, a record is assigned to more than one block and records are not evenly allocated to blocks
- For example, `alissa kilmartin` is assigned to blocks with `surname_prefix==kilm` and `given_name_prefix==alis`
- What is the advantage of records being assigned to multiple blocks?
  - To reduce the chance of matched pairs not allocated to a common block.

Theoretical questions

- We've seen how accuracy does not truly reflect the performance of a blocking algorithm. Instead, the confusion matrix can be used to evaluate the performance based on TP, TN, FP, FN.
- It is desirable to minimise both FP and FN, but it may be difficult for a blocking algorithm to minimise both simultaneously.
- Give an example application where minimising FP is more important than minimising FN.
  - Fraud detection
- Give an example application where minimising FN is more important than minimising FP.
  - Medical diagnosis

THE UNIVERSITY OF MELBOURNE

Thank you

More Resources: Canvas