# COMP7105 Advanced topics in data science

Introduction

Nikos Mamoulis
nikos@cs.hku.hk


THE UNIVERSITY OF HONG KONG
DEPARTMENT OF
COMPUTER SCIENCE

# Who am I?

- Visiting professor at HKU@CS
- Professor at University of Ioannina, CSE department

# Course Information

- Course website
  - At HKU Moodle
- Instructor
  - Prof. Nikos Mamoulis (nikos@cs.hku.hk)
  - Room: TBA
- Teaching assistant(s)
  - Yuan Mingruo (u3008435@connect.hku.hk)
- Lectures
  - Every Friday 7-10pm at room KK-201 (first four lectures via zoom)

# Course Material

- Slides
- Written notes by the instructor
- Chapters from textbooks
  - Database System Concepts, https://www.db-book.com
  - Introduction to Data Mining, https://www-users.cse.umn.edu/~kumar001/dmbook/index.php
  - Mining of Massive Datasets, http://www.mmds.org
  - Spatial Data Management, https://doi.org/10.2200/S00394ED1V01Y201111DTM021
- Scientific papers (how to read a paper)

# Course Assessment

- 4 programming assignments
  - Roughly one assignment every 3-4 weeks
  - Querying and analytics on real data collections
  - In your preferred programming language
- Final examination

# About the Course Content

- Advanced computational methods applicable to data analysis problems
  - Managing, searching, analyzing multi-dimensional data
  - Recommender systems
  - Temporal and time-series analytics
  - Streaming data analytics
  - Adaptive and learned indexing
  - Provenance and explainability of outputs

# Course Learning Outcomes

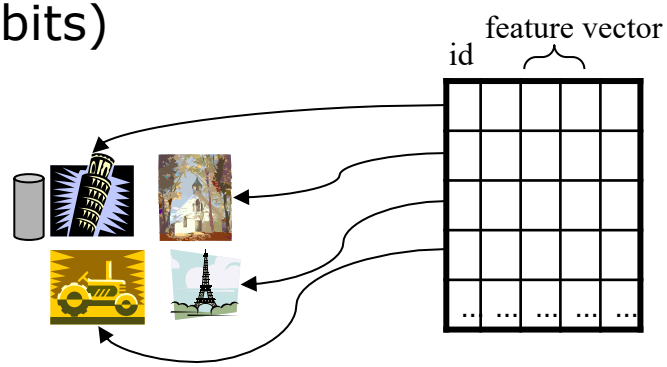
- Understand the nature of multidimensional data and temporal historical data, and the fundamental management and search methods for such data
- Learn fundamental on-line data analytics approaches for data streams as well as learned indexes that adapt to the data distribution and query workload
- Understand the value of explaining query results and learn methods for data provenance

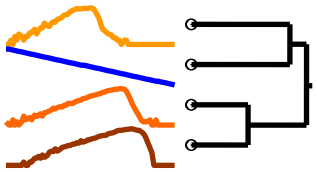
# Necessary Background

- Database management systems
  - The relational model, relational queries, data storage in memory and on disk
- Linear Algebra
  - Basic concepts and operations
- Programming
  - Excellent knowledge of at least one programming language (e.g., C/C++, Java, Python)

# Multidimensional data analysis

- Simple data types (numbers, characters, bits)
- Multidimensional objects and indexing
  - spatial objects
  - feature vectors
  - sequences (strings, bitstrings, time-series)
  - sparse vectors
  - facts in historical transactional data
  - temporal data
- Queries and analysis tasks in multidimensional spaces
  - multidimensional range selection queries
  - distance-based search and similarity search
  - recommendations
  - cluster analysis
  - top-k and skyline search
  - on-line analytical processing
  - time-travel search



Time-series management and analytics

# Data Types and Similarity

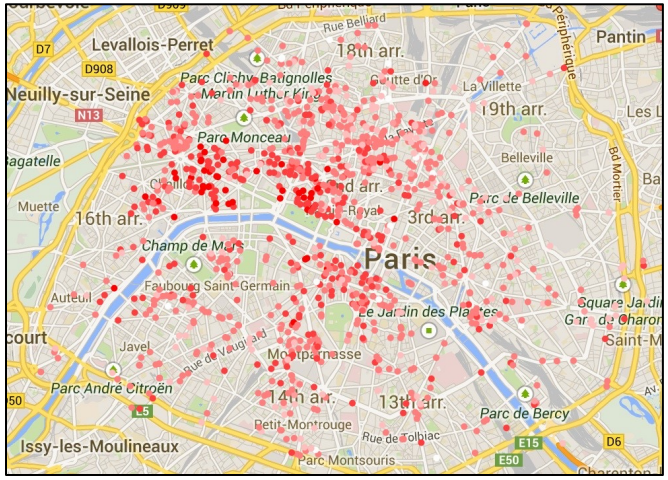
| ssn | name | lot |
|---|---|---|
| 13-324 | Jones | 22 |
| 13-322 | Smith | 45 |
| 12-824 | Parker | 125 |
| 21-397 | Smith | 12 |

- Objects characterized by multiple features
  - Employee characterized by name, gender, age, salary, etc.
- Problem: how do we define the degree of difference between different values
  - Is 1 very different to 5 and why?
- Problem: how do we define the similarity between objects
  - Is (John, M, 45, 20K) similar to (Mary, F, 25, 15K)?
- Why is measuring similarity important?

# Spatial Data

- Numerous applications
  - Mobile services, geo-sciences, CAD, astronomy, military, routing

Lines like roads




Points like hotel locations

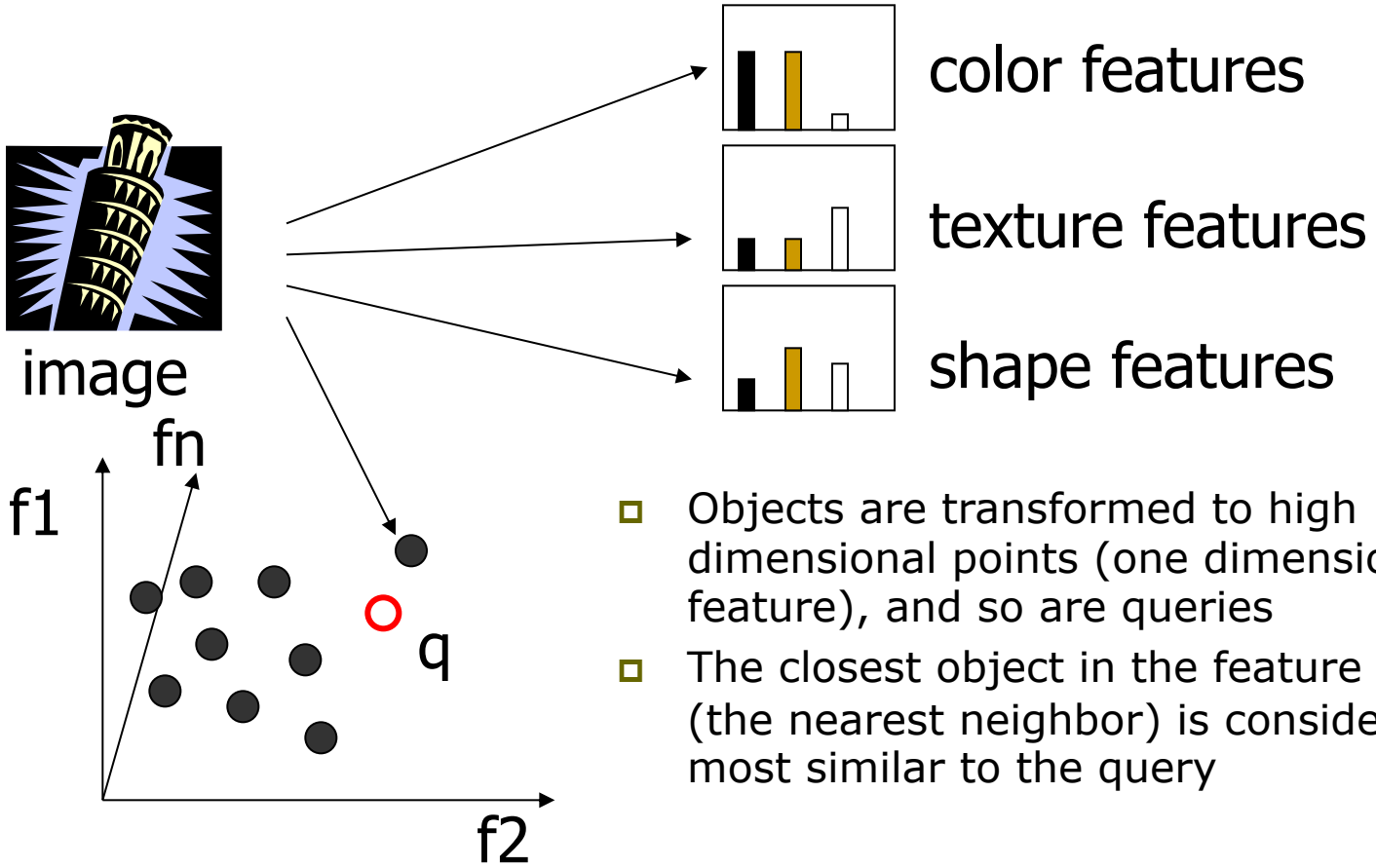
Polygons like lakes

# Spatial Data

- Just two (or maybe three) dimensions
- Point data
  - One value per dimension
- Non-point data
  - More complex geometric representation
- Spatial Queries
  - Range selection: find all mobile users in HKU campus
  - Nearest neighbor: find the nearest ATM to my location
  - Spatial join: find pairs of hotels and restaurants near each other
- We will learn: models, indexing, query evaluation
- Concepts & index/search methods for spatial points generalize for multidimensional objects

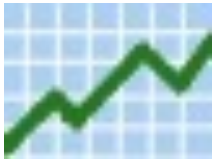
# Dense Multidimensional Data

- For each object, all features (dimensions) have a value and all are equally important




- Objects are transformed to high dimensional points (one dimension per feature), and so are queries
- The closest object in the feature space (the nearest neighbor) is considered the most similar to the query

# Dense Multidimensional Data



- Important queries
  - Range similarity search: find image feature vectors with distance a most ε to a query vector q
  - NN similarity search: find the k image feature vectors with the smallest distance to a query vector q
- Problem: curse of dimensionality
- We will learn: indexing methods for multidimensional points

# Sparse Multidimensional Data

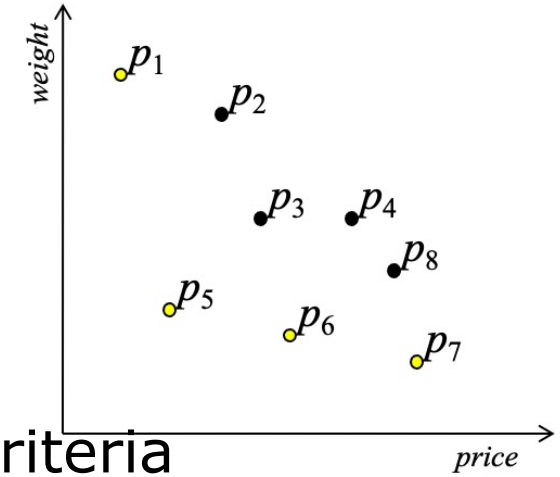
| | HP1 | HP2 | HP3 | TW | SW1 | SW2 | SW3 |
|---|---|---|---|---|---|---|---|
| *A* | 4 | | | 5 | 1 | | |
| *B* | 5 | 5 | 4 | | | | |
| *C* | | | | 2 | 4 | 5 | |
| *D* | | 3 | | | | | 3 |

- For each object, few features (different for different objects) are important
  - supermarket transactions, text documents, movie ratings by users, etc.
- We will learn:
  - similarity (ranking) measures and indexes
  - recommendation techniques

| | team | coach | play | ball | score | game | win | lost | timeout | season |
|---|---|---|---|---|---|---|---|---|---|---|
| Document 1 | 3 | 0 | 5 | 0 | 2 | 6 | 0 | 2 | 0 | 2 |
| Document 2 | 0 | 7 | 0 | 2 | 1 | 0 | 0 | 3 | 0 | 0 |
| Document 3 | 0 | 1 | 0 | 0 | 1 | 2 | 2 | 0 | 3 | 0 |

# Advanced Multidimensional Tasks

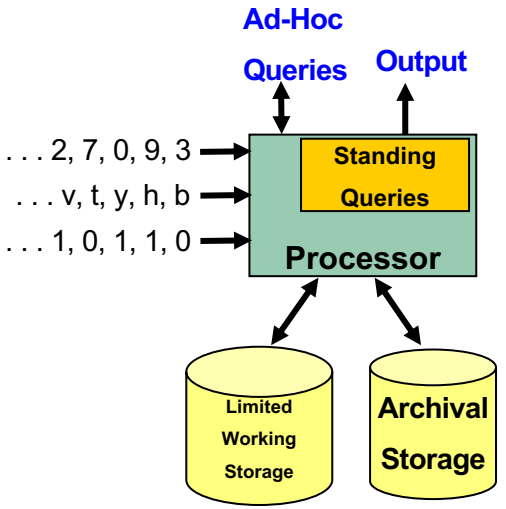
- Top-k queries
  - Rank laptops based on price, size, and battery life
- Skyline queries
  - Find the the set of laptops that are not dominated by others in all criteria




- Online analytical processing
  - Compute total sales for each (region,item) pair
- Time travel search
  - Find all clerks employed from 05/94 to 06/96
- Cluster analysis
  - Automatically divide images into groups such that images in the same group are similar to each other

# Streaming data analytics

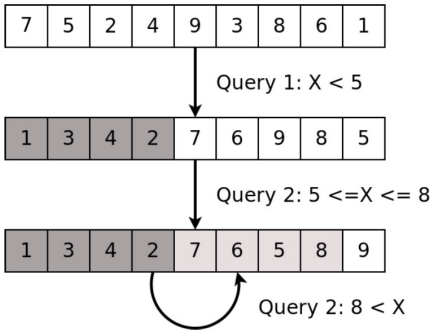
- Data arrive fast in one or more streams
  - If not processed immediately (or stored), data are lost forever
- We will learn:
  - Maintain a stream sample
  - Lookup set-membership
    - Bloom filter
  - Count distinct elements seen so far
    - FM-sketch
  - Count moments
    - AMS-sketch
  - Sliding window queries
    - Answer queries on the last k elements of the stream



# Adaptive Indexes

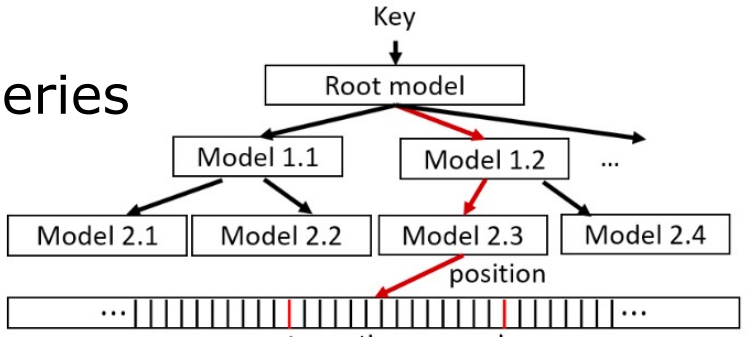
- Building an entire index on a dataset before querying it may not pay off
  - Index construction may take time/resources
  - We may not query all data
  - Data may be ephemeral
- Adaptive indexing: index construction adapts to query workload during query evaluation
- We will learn:
  - Database cracking
  - Adaptive merging
  - Multidimensional cracking

Adaptive index: index is built progressively at query time



# Learned Indexes

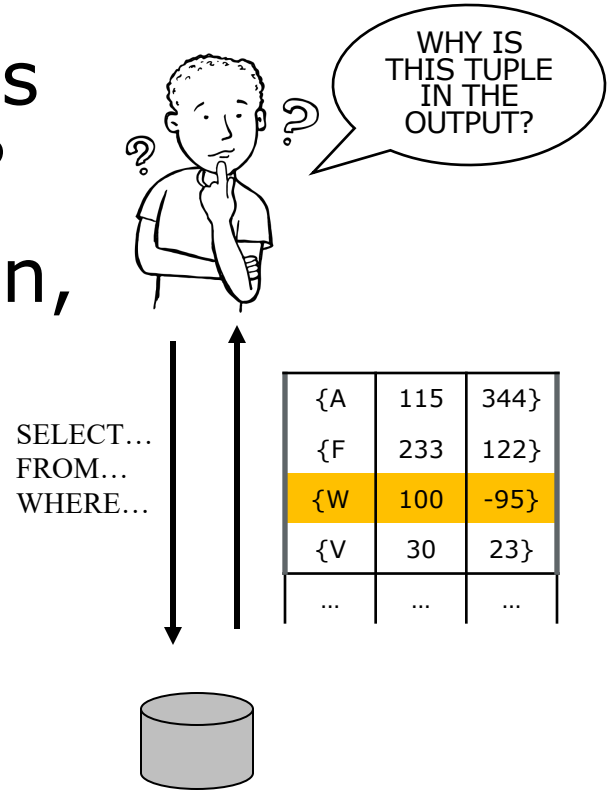
- Idea: replace parts of a traditional index by prediction models
- Potential benefits:
  - Models occupy less space than index structures
  - Models can be faster to use than index search algorithms
- Challenges:
  - Find models that accurately capture the data distribution
  - Maintain index during updates
  - Handle different data types and queries
- We will learn:
  - Recursive model indexing for static data
  - Learned indexing for dynamic data
  - Multidimensional learned indexes




Learned index: index nodes are replaced by ML models

# Data Provenance

- How do input data in a process contribute to specific outputs?
- Applications: trust, explanation, debugging, reproducibility
- We will learn:
  - Data provenance concepts
  - Data provenance techniques
    - Metadata propagation
    - Operator inversion
    - Backward tracing



# Tentative Schedule

- Week 1: Introduction, data types
- Week 2: Spatial data and spatial queries
- Week 3: Dense multidimensional data
- Week 4: Sparse multidimensional data
- Week 5: Multidimensional queries (part 1)
- Week 6: Multidimensional queries (part 2)
- Week 7: Data streams
- Week 8: Adaptive and learned indexes
- Week 9: Data Provenance (part 1)
- Week 10: Data Provenance (part 2)

# Tentative Assignment Deadlines

- Assignment 1, due Feb 25
- Assignment 2, due March 11
- Assignment 3, due March 25
- Assignment 4, due April 15

Implement programs that solve practical problems based on course material

# Remember

- The main objective (for me and you) is to learn from this course
- Ask questions
- Speak up if you don't understand

Let's start!