Can you have too much data?

Or: Even a blind squirrel may occasionally find a nut

Or: How do you know that needle you found in the haystack is the one you were looking for?

WAKE FOREST UNIVERSITY

WAKE FOREST UNIVERSITY

Potential Problems on the Analysis Side

Bonferroni's Principle

In a nutshell: *if you look harder than the data supports, you will find a pattern that "fits" the pattern you are looking for*.

If you look for events of a certain type in a data set you can expect events of that type to occur -- **even if the data is completely random** -- and the number of occurrences will grow as the size of the data grows.

WAKE FOREST UNIVERSITY

Example

- **Consider a country with a population of 300,000,000 people. Any individual could be a "bad-guy".**
- **On average each person stays in a hotel once every 100 days, or equivalently, 1% of the population stays in a hotel on any given day.**
- **Hotels hold 100 people on average. There are just enough hotels for everyone**

  => (300,000,000/100)/100 = 30,000 hotels
- **An unnamed government agency collects data on hotel stays for 1000 days, looking for <u>pairs of people</u> who stayed at <u>the same hotel</u> on <u>two different days</u>. Hypothesis: "bad-guys" like to meet at hotels to plan evil-doing.**

WAKE FOREST UNIVERSITY

What does Bonferroni tell us?

- **What if there are actually no "bad-guys" and everyone just behaves normally (randomly).**
- **What if everyone stays in a hotel 1% of the time and chooses one of the 30,000 hotels randomly? What will the data indicate?**
- **P(Alice and Bob both choose the same day to stay in a hotel) = .01 * .01 = .0001.**
- **P(Alice and Bob both stay in a given hotel on the same day) = .0001 / 30000 = $3.3 * 10^{-9}$**
- **P(Alice and Bob stay in same hotel on 2 days) = $3.3 \times 10^{-9} * 3.3 \times 10^{-9} = 10^{-17}$**

WAKE FOREST UNIVERSITY

How many events will look "suspicious"?

- **The number of pairs of individuals is (300,000,000 choose 2) = 4.5×10^{16}**
- **The number of pairs of days is (1000 choose 2) = 5×10^{5}**
- **The number of events that look "suspicious" will be $4.5 \times 10^{16} * 5 \times 10^{5} * 10^{-17} = 225{,}000$**
- **That is, the data will identify 225,000 suspicious events, even if there are no "bad guys".**
- **Hard Question: If there are 100 "bad guys" in the population, is it okay to investigate 225,000 others to find them?**

WAKE FOREST UNIVERSITY

Potential Problems on the Analysis Side

Bonferroni's Principle

In a nutshell: *if you look harder than the data supports, you will find a pattern that "fits" the pattern you are looking for*.

If you look for events of a certain type in a data set you can expect events of that type to occur -- **even if the data is completely random** -- and the number of occurrences will grow as the size of the data grows.

Why do we need to know about relational databases in 2014?

Relational Databases

- **Continue to be the most common way to organize large datasets, although not *the very largest* data sets**
- **Work well for structured data**
- **Provide layers of separation between physical storage and a user's perspective**
- **Work well for very high speed storage and retrieval of moderately large size datasets (up to a few TB)**
- **Handle concurrent access very well**
- **Ensure that data is consistent and valid:**
  - ACID properties

Relational Databases

- **Come with well-developed query languages such as SQL, QBE, etc.**
- **Associated with a simple, well-defined, and well understood *data model* including constraints and operations**
  - Data model – relations over domains; tables with rows and columns
  - Constraints – each value in a column is of a specific data type; all rows have same number of columns, etc
  - Operations – union, intersection, difference, Cartesian product, etc.

Relational Databases

- **The set of operations we perform on relations is referred to as the *relational algebra***
- **The relational database model allows developers to separate structure from content in the same way that *classes* are used in O-O design.**
  - Database schema – the design, identifies the domains (data types), the number of columns, the relationships between tables, constraints, etc.
  - Database instance – like objects in O-O, refers to a snapshot of the data in a DB at any point in time; might be none!

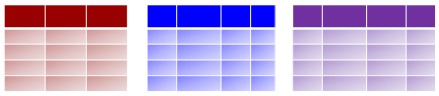
Relational Databases

Relational Algebra
Select, project, join

Jennifer Widom

Relational Algebra

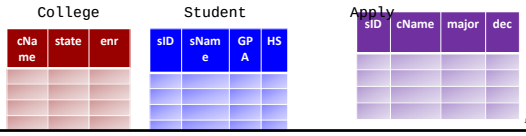
Query (expression) on set of relations produces relation as a result



Relational Algebra

Examples: simple college admissions database

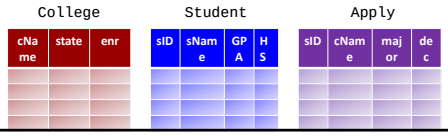
```
College(cName,state,enrollment)
Student(sID,sName,GPA,sizeHS)
Apply(sID,cName,major,decision)
```



Relational Algebra

Simplest query: relation name

Use operators to filter, slice, combine



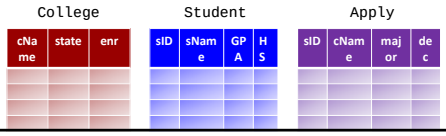
Relational Algebra

Select operator: picks certain rows

Students with GPA>3.7

Students with GPA>3.7 and HS<1000

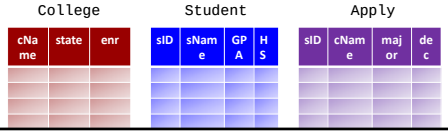
Applications to Stanford CS major



Relational Algebra

Project operator: picks certain columns

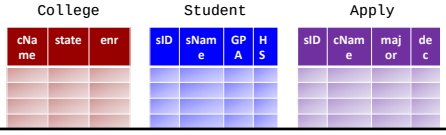
ID and decision of all applications



Relational Algebra

To pick both rows and columns…

ID and name of students with GPA>3.7



Relational Algebra

Duplicates

List of application majors and decisions

College

cName	state	enr

Student

sID	sName	GPA	HS

Apply

sID	cName	major	dec

Jennifer Widom

Relational Algebra

Cross-product: combine two relations (a.k.a. Cartesian product)

College

cName	state	enr

Student

sID	sName	GPA	HS

Apply

sID	cName	major	dec

Jennifer Widom

Relational Algebra

Cross-product: combine two relations (a.k.a. Cartesian product)

Names and GPAs of students with HS>1000 who applied to CS and were rejected

College

cName	state	enr

Student

sID	sName	GPA	HS

Apply

sID	cName	major	dec

Jennifer Widom

Relational Algebra

Natural Join

- Enforce equality on all attributes with same name
- Eliminate one copy of duplicate attributes

College

cName	state	enr

Student

sID	sName	GPA	HS

Apply

sID	cName	major	dec

Jennifer Widom

Relational Algebra

Natural Join

Names and GPAs of students with HS>1000 who applied to CS and were rejected

Names and GPAs of students with HS>1000 who applied to CS at college with enr>20,000 and were rejected

College

cName	state	enr

Student

sID	sName	GPA	HS

Apply

sID	cName	major	dec

Jennifer Widom

Relational Algebra

Natural Join

College

cName	state	enr

Student

sID	sName	GPA	HS

Apply

sID	cName	major	dec

Jennifer Widom

Relational Algebra

Theta Join

- Basic operation implemented in DBMS
- Term "join" often means theta join

College

cName	state	enr

Student

sID	sName	GPA	HS

Apply

sID	cName	major	dec

Relational Algebra

Query (expression) on set of relations produces relation as a result

- Simplest query: relation name
- Use operators to filter, slice, combine
- Operators so far: select, project, cross-product, natural join, theta join

Relational Databases

Relational Algebra

Set operators, renaming, notation

Relational Algebra

Relational algebra query (expression) on set of relations produces relation as a result

College(cName, state, enrollment)

Student(sID, sName, GPA, sizeHS)

Apply(sID, cName, major, decision)

College

cName	state	enr

Student

sID	sName	GPA	HS

Apply

sID	cName	major	dec

Relational Algebra

Union operator

List of college and student names

College

cName	state	enr

Student

sID	sName	GPA	HS

Apply

sID	cName	major	dec

Relational Algebra

Difference operator

IDs of students who didn't apply anywhere

IDs and names of students who didn't apply anywhere

College

cName	state	enr

Student

sID	sName	GPA	HS

Apply

sID	cName	major	dec

Relational Algebra

Intersection operator

Names that are both a college name and a student name

College

cName	state	enr

Student

sID	sName	GPA	HS

Apply

sID	cName	major	dec

Jennifer Widom

Relational Algebra

Intersection doesn't add expressive power (1)

College

cName	state	enr

Student

sID	sName	GPA	HS

Apply

sID	cName	major	dec

Jennifer Widom

Relational Algebra

Intersection doesn't add expressive power (2)

College

cName	state	enr

Student

sID	sName	GPA	HS

Apply

sID	cName	major	dec

Jennifer Widom

Relational Algebra

Rename operator

1.
2.
3.

College

cName	state	enr

Student

sID	sName	GPA	HS

Apply

sID	cName	major	dec

Jennifer Widom

Relational Algebra

Rename operator

To unify schemas for set operators

List of college and student names

College

cName	state	enr

Student

sID	sName	GPA	HS

Apply

sID	cName	major	dec

Jennifer Widom

Relational Algebra

Rename operator

For disambiguation in "self-joins"

Pairs of colleges in same state

College

cName	state	enr

Student

sID	sName	GPA	HS

Apply

sID	cName	major	dec

Jennifer Widom

Relational Algebra summary