WISCONSIN
UNIVERSITY OF WISCONSIN–MADISON

CS639:
Data Management for Data Science

Lecture 23: Data Cleaning

[based on slides by John Canny]

Theodoros Rekatsinas

Dirty Data

- The Statistics View:
  - There is a process that produces data
  - We want to model ideal samples of that process, but in practice we have non-ideal samples:
    - **Distortion** – some samples are corrupted by a process
    - **Selection Bias** - likelihood of a sample depends on its value
    - **Left and right censorship** - users come and go from our scrutiny
    - **Dependence** – samples are supposed to be independent, but are not (e.g. social networks)
  - You can add new models for each type of imperfection, but you can't model everything.
  - What's the best trade-off between accuracy and simplicity?

Dirty Data

- The Database View:
    - I got my hands on this data set
    - Some of the values are missing, corrupted, wrong, duplicated
    - Results are absolute (relational model)
    - You get a better answer by improving the quality of the values in your dataset

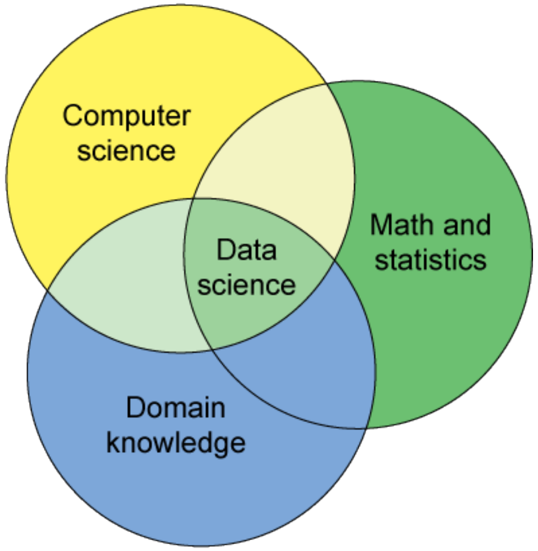
Dirty Data

- The Domain Expert's View:
  - This Data Doesn't look right
  - This Answer Doesn't look right
  - What happened?

- Domain experts have an implicit model of the data that they can test against…

Dirty Data

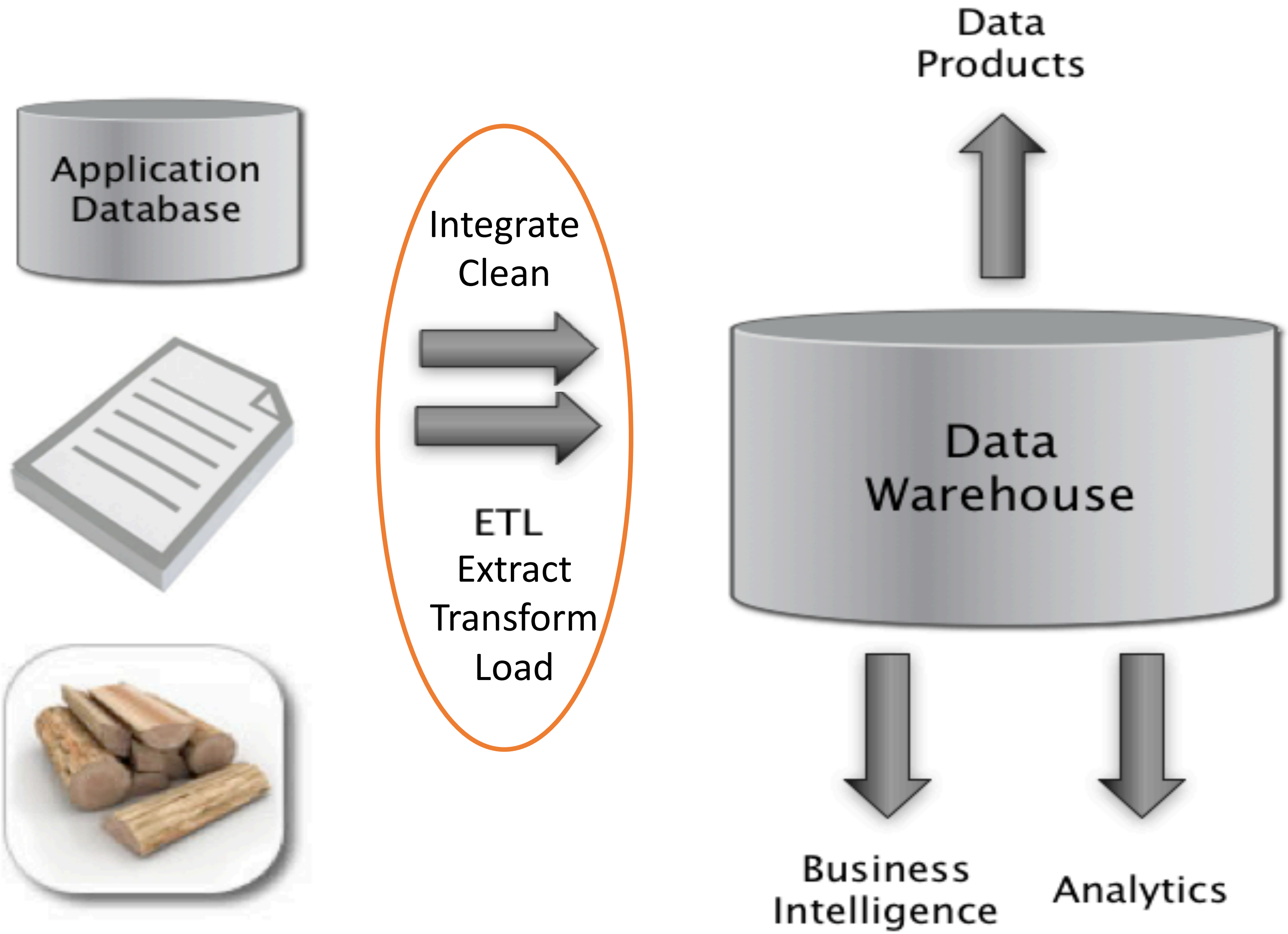
- The Data Scientist's View:
    - Some Combination of all of the above



Data Quality Problems

- (Source) Data is dirty on its own.
- Transformations corrupt the data (complexity of software pipelines).
- Data sets are clean but integration (i.e., combining them) screws them up.
- “Rare” errors can become frequent after transformation or integration.
- Data sets are clean but suffer “bit rot”
  - Old data loses its value/accuracy over time
- Any combination of the above

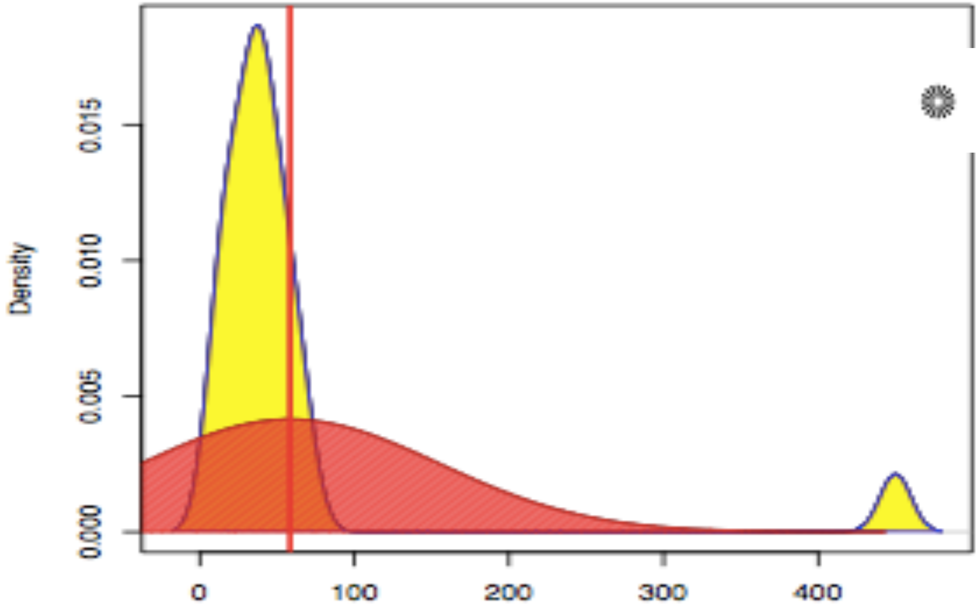
Big Picture: Where can Dirty Data Arise?



Numeric Outliers

12	13	14	21	22	26	33	35	36	37	39	42	45	47	54	57	61	68	450

ages of employees (US)



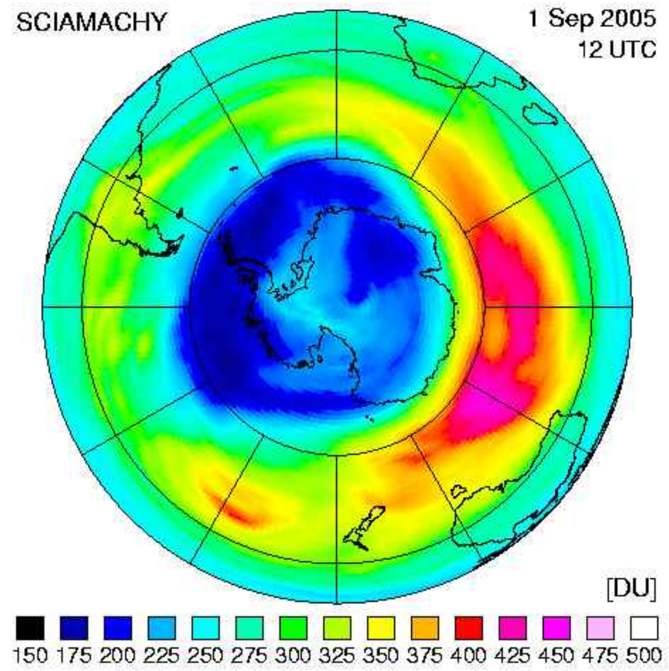
- median 37
- mean 58.52632
- variance 9252.041

Adapted from Joe Hellerstein's 2012 CS 194 Guest Lecture

Data Cleaning Makes Everything Okay?

The appearance of a hole in the earth's ozone layer over Antarctica, first detected in 1976, was so unexpected that scientists didn't pay attention to what their instruments were telling them; they thought their instruments were malfunctioning.

National Center for Atmospheric Research



In fact, the data were rejected as unreasonable by data quality control algorithms

Dirty Data Problems

- From Stanford Data Integration Course:
  1) parsing text into fields (separator issues)
  2) Naming conventions: ER: NYC vs New York
  3) Missing required field (e.g. key field)
  4) Different representations (2 vs Two)
  5) Fields too long (get truncated)
  6) Primary key violation (from un- to structured or during integration
  7) Redundant Records (exact match or other)
  8) Formatting issues – especially dates
  9) Licensing issues/Privacy/ keep you from using the data as you would like?

Conventional Definition of Data Quality

- Accuracy
    - The data was recorded correctly.
- Completeness
    - All relevant data was recorded.
- Uniqueness
    - Entities are recorded once.
- Timeliness
    - The data is kept up to date.
        - Special problems in federated data: time consistency.
- Consistency
    - The data agrees with itself.

Adapted from Ted Johnson's SIGMOD 2003 Tutorial

Problems …

- Unmeasurable
  - Accuracy and completeness are extremely difficult, perhaps impossible to measure.
- Context independent
  - No accounting for what is important. E.g., if you are computing aggregates, you can tolerate a lot of inaccuracy.
- Incomplete
  - What about interpretability, accessibility, metadata, analysis, etc.
- Vague
  - The conventional definitions provide no guidance towards practical improvements of the data.

Adapted from Ted Johnson's SIGMOD 2003 Tutorial

Finding a modern definition

- We need a definition of data quality which
  - Reflects the **use** of the data
  - Leads to **improvements in processes**
  - Is **measurable** (we can define metrics)

- First, we need a better understanding of how and where data quality problems occur
  - The data quality continuum

Adapted from Ted Johnson's SIGMOD 2003 Tutorial

Meaning of Data Quality (2)

- There are many types of data, which have different uses and typical quality problems
  - Federated data
  - High dimensional data
  - Descriptive data
  - Longitudinal data
  - Streaming data
  - Web (scraped) data
  - Numeric vs. categorical vs. text data

Adapted from Ted Johnson's SIGMOD 2003 Tutorial

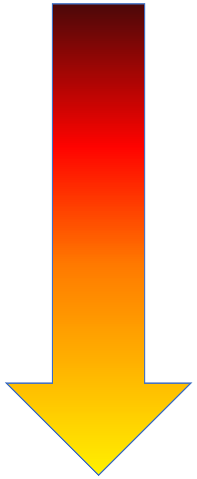
Meaning of Data Quality (2)

- There are many uses of data
  - Operations
  - Aggregate analysis
  - Customer relations …
- Data Interpretation : the data is useless if we don't know all of the *rules* behind the data.
- Data Suitability : Can you get the answer from the available data
  - Use of proxy data
  - Relevant data is missing

Adapted from Ted Johnson's SIGMOD 2003 Tutorial

The Data Quality Continuum

- Data and information is not static, it flows in a data collection and usage process
  - Data gathering
  - Data delivery
  - Data storage
  - Data integration
  - Data retrieval
  - Data mining/analysis



Adapted from Ted Johnson's SIGMOD 2003 Tutorial

Data Gathering

- How does the data enter the system?
- Sources of problems:
  - Manual entry
  - No uniform standards for content and formats
  - Parallel data entry (duplicates)
  - Approximations, surrogates – SW/HW constraints
  - Measurement or sensor errors.

Adapted from Ted Johnson's SIGMOD 2003 Tutorial

Data Gathering - Solutions

- Potential Solutions:
    - Preemptive:
        - Process architecture (build in integrity checks)
        - Process management (reward accurate data entry, data sharing, data stewards)
    - Retrospective:
        - Cleaning focus (duplicate removal, merge/purge, name & address matching, field value standardization)
        - Diagnostic focus (automated detection of glitches).

Adapted from Ted Johnson's SIGMOD 2003 Tutorial

Data Delivery

- Destroying or mutilating information by inappropriate pre-processing
  - Inappropriate aggregation
  - Nulls converted to default values
- Loss of data:
  - Buffer overflows
  - Transmission problems
  - No checks

Adapted from Ted Johnson's SIGMOD 2003 Tutorial

Data Delivery - Solutions

- Build reliable transmission protocols
  - Use a relay server
- Verification
  - Checksums, verification parser
  - Do the uploaded files fit an expected pattern?
- Relationships
  - Are there dependencies between data streams and processing steps
- Interface agreements
  - Data quality commitment from the data stream supplier.

Adapted from Ted Johnson's SIGMOD 2003 Tutorial

Data Storage

- You get a data set. What do you do with it?
- Problems in physical storage
  - Can be an issue, but terabytes are cheap.
- Problems in logical storage
  - Poor metadata.
    - Data feeds are often derived from application programs or legacy data sources. What does it mean?
  - Inappropriate data models.
    - Missing timestamps, incorrect normalization, etc.
  - Ad-hoc modifications.
    - Structure the data to fit the GUI.
  - Hardware / software constraints.
    - Data transmission via Excel spreadsheets, Y2K

Adapted from Ted Johnson's SIGMOD 2003 Tutorial

Data Storage - Solutions

- Metadata
  - Document and publish data specifications.
- Planning
  - Assume that everything bad will happen.
  - Can be very difficult.
- Data exploration
  - Use data browsing and data mining tools to examine the data.
    - Does it meet the specifications you assumed?
    - Has something changed?

Adapted from Ted Johnson's SIGMOD 2003 Tutorial

Data Retrieval

- Exported data sets are often a view of the actual data. Problems occur because:
    - Source data not properly understood.
    - Need for derived data not understood.
    - Just plain mistakes.
        - Inner join vs. outer join
        - Understanding NULL values
- Computational constraints
    - E.g., too expensive to give a full history, we'll supply a snapshot.
- Incompatibility
    - Ebcdic? Unicode?

Adapted from Ted Johnson's SIGMOD 2003 Tutorial

Data Mining and Analysis

- What are you doing with all this data anyway?
- Problems in the analysis.
  - Scale and performance
  - Confidence bounds?
  - Black boxes and dart boards
  - Attachment to models
  - Insufficient domain expertise
  - Casual empiricism

Adapted from Ted Johnson's SIGMOD 2003 Tutorial

Retrieval and Mining - Solutions

- Data exploration
  - Determine which models and techniques are appropriate, find data bugs, develop domain expertise.
- Continuous analysis
  - Are the results stable? How do they change?
- Accountability
  - Make the analysis part of the feedback loop.

Adapted from Ted Johnson's SIGMOD 2003 Tutorial

Data Quality Constraints

- Many data quality problems can be captured by *static* constraints based on the schema.
  - Nulls not allowed, field domains, foreign key constraints, etc.
- Many others are due to problems in workflow, and can be captured by *dynamic* constraints
  - E.g., orders above $200 are processed by Biller 2
- The constraints follow an 80-20 rule
  - A few constraints capture most cases, thousands of constraints to capture the last few cases.
- Constraints are measurable. Data Quality Metrics?

Adapted from Ted Johnson's SIGMOD 2003 Tutorial

Data Quality Metrics

- We want a measurable quantity
  - Indicates what is wrong and how to improve
  - Realize that DQ is a messy problem, no set of numbers will be perfect
- Types of metrics
  - Static vs. dynamic constraints
  - Operational vs. diagnostic
- Metrics should be *directionally correct* with an improvement in use of the data.
- A very large number metrics are possible
  - Choose the most important ones.

Adapted from Ted Johnson's SIGMOD 2003 Tutorial

Examples of Data Quality Metrics

- Conformance to schema
  - Evaluate constraints on a snapshot.
- Conformance to business rules
  - Evaluate constraints on changes in the database.
- Accuracy
  - Perform inventory (expensive), or use proxy (track complaints). Audit samples?
- Accessibility
- Interpretability
- Glitches in analysis
- Successful completion of end-to-end process

Adapted from Ted Johnson's SIGMOD 2003 Tutorial

Technical Approaches

- We need a multi-disciplinary approach to attack data quality problems
  - No one approach solves all problem
- Process management
  - Ensure proper procedures
- Statistics
  - Focus on analysis: find and repair anomalies in data.
- Database
  - Focus on relationships: ensure consistency.
- Metadata / domain expertise
  - What does it mean? Interpretation

Adapted from Ted Johnson's SIGMOD 2003 Tutorial

Data cleaning for structured data

Detect and ***repair*** errors in a structured dataset

University of Chicago, *Cicago*, IL

Data cleaning for structured data

Detect and ***repair*** errors in a structured dataset

University of Chicago, *Cicago*, IL

1. Detect — University of Chicago, *Cicago*, IL

Data cleaning for structured data

Detect and ***repair*** errors in a structured dataset

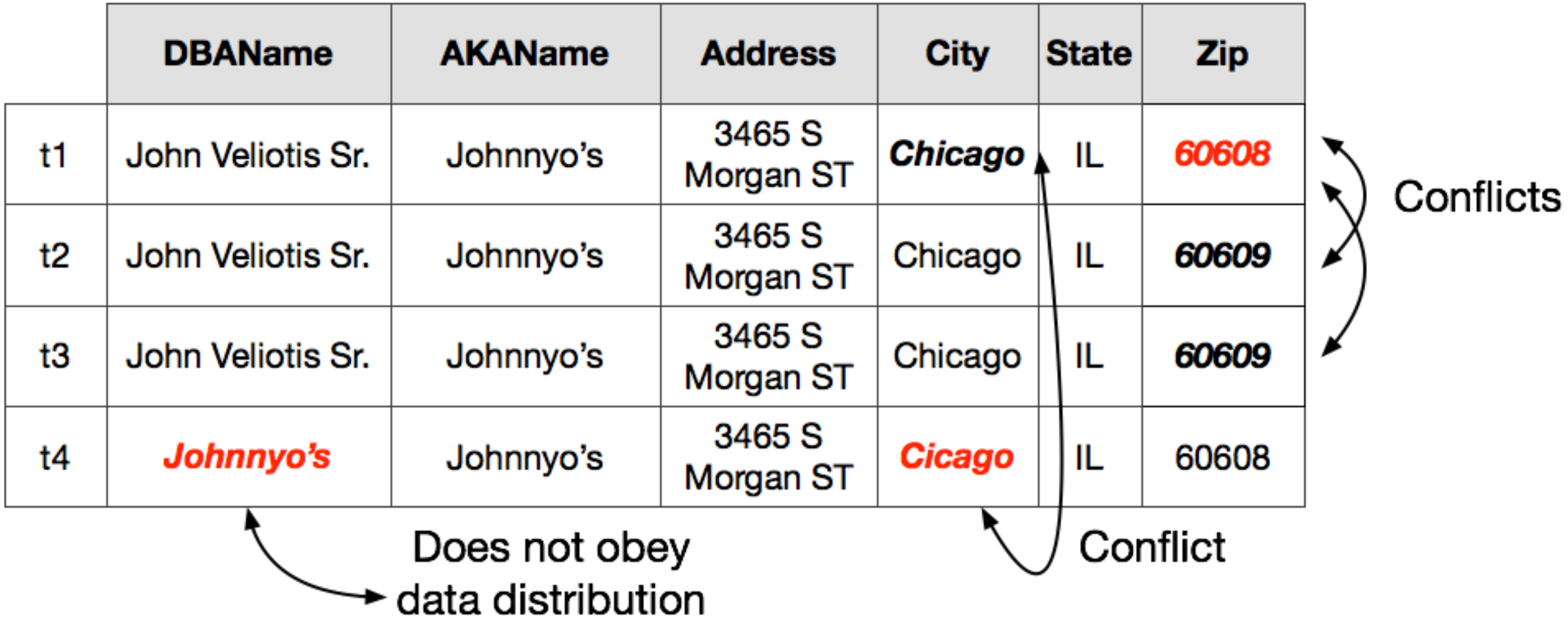
University of Chicago, *Cicago*, IL

1. Detect University of Chicago, *Cicago*, IL

2. Repair University of Chicago, *Chicago*, IL

A simple example

Chicago’s food inspection dataset



	DBAName	AKAName	Address	City	State	Zip
t1	John Veliotis Sr.	Johnnyo’s	3465 S Morgan ST	***Chicago***	IL	***60608***
t2	John Veliotis Sr.	Johnnyo’s	3465 S Morgan ST	Chicago	IL	***60609***
t3	John Veliotis Sr.	Johnnyo’s	3465 S Morgan ST	Chicago	IL	***60609***
t4	***Johnnyo’s***	Johnnyo’s	3465 S Morgan ST	***Cicago***	IL	60608

Conflicts

Does not obey data distribution

Conflict

Detect and ***repair*** errors in a structured dataset

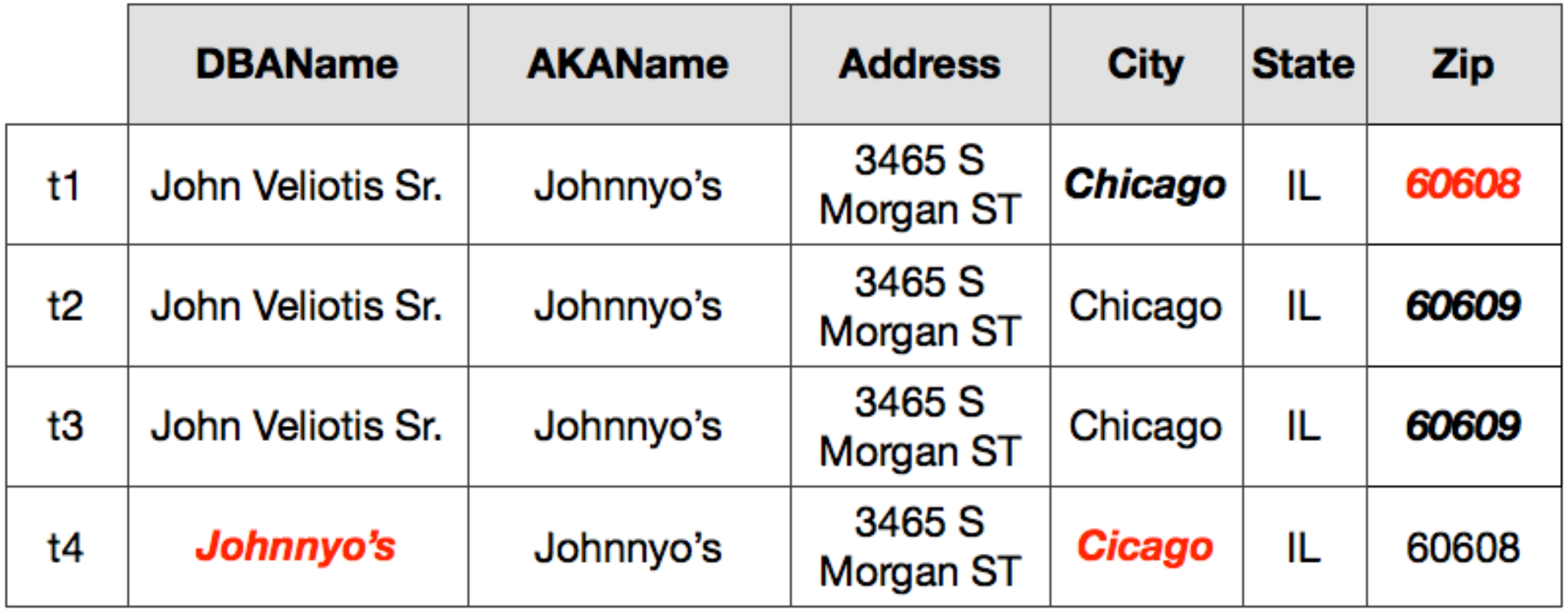
Constraints and minimality

Functional dependencies

$$\text{c1: DBAName} \rightarrow \text{Zip}$$

$$\text{c2: Zip} \rightarrow \text{City, State}$$

$$\text{c3: City, State, Address} \rightarrow \text{Zip}$$



	DBAName	AKAName	Address	City	State	Zip
t1	John Veliotis Sr.	Johnnyo's	3465 S Morgan ST	***Chicago***	IL	***60608***
t2	John Veliotis Sr.	Johnnyo's	3465 S Morgan ST	Chicago	IL	***60609***
t3	John Veliotis Sr.	Johnnyo's	3465 S Morgan ST	Chicago	IL	***60609***
t4	***Johnnyo's***	Johnnyo's	3465 S Morgan ST	***Cicago***	IL	60608

Bohannon et al., 2005, 2007; Kolahi and Lakshmanan , 2005; Bertossi et al., 2011; Chu et al., 2013; 2015 Fagin et al., 2015

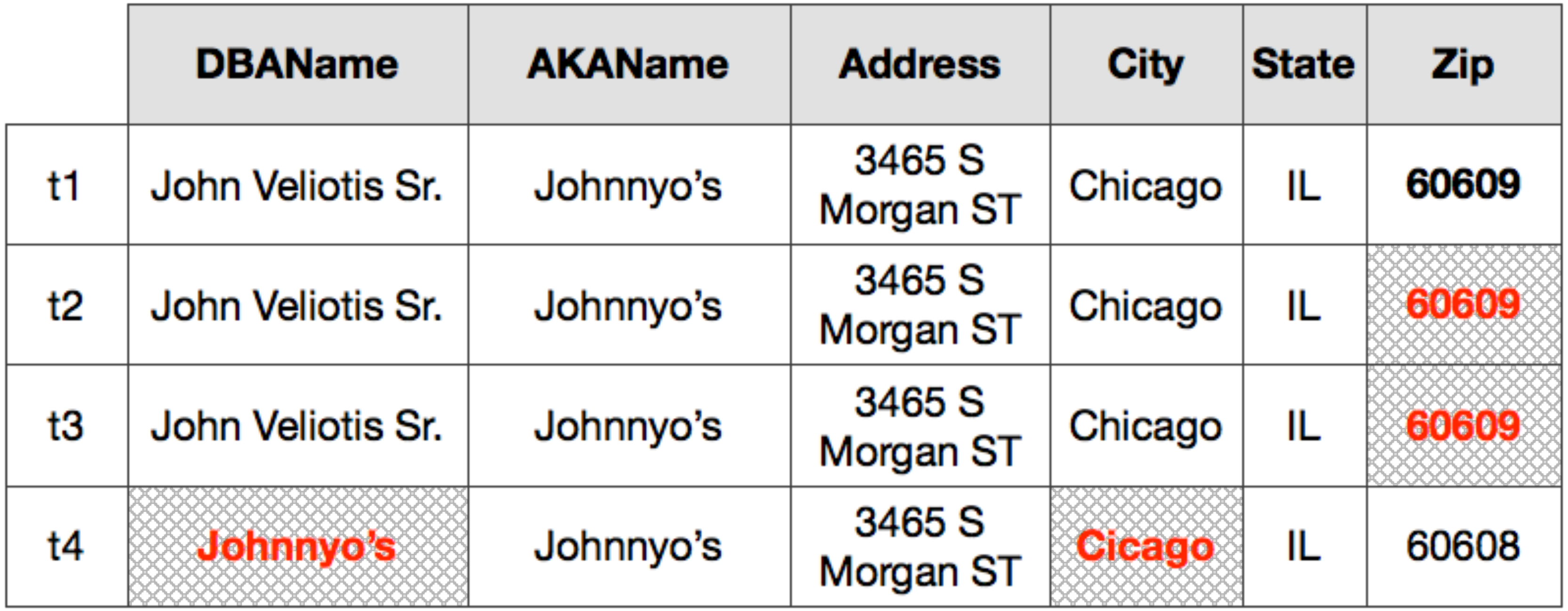
Constraints and minimality

Functional dependencies

c1: DBAName → Zip

c2: Zip → City, State

c3: City, State, Address → Zip



	DBAName	AKAName	Address	City	State	Zip
t1	John Veliotis Sr.	Johnnyo's	3465 S Morgan ST	Chicago	IL	**60609**
t2	John Veliotis Sr.	Johnnyo's	3465 S Morgan ST	Chicago	IL	**60609**
t3	John Veliotis Sr.	Johnnyo's	3465 S Morgan ST	Chicago	IL	**60609**
t4	**Johnnyo's**	Johnnyo's	3465 S Morgan ST	**Cicago**	IL	60608

Action: Fewer erroneous than correct cells; perform minimum number of changes to satisfy all constraints

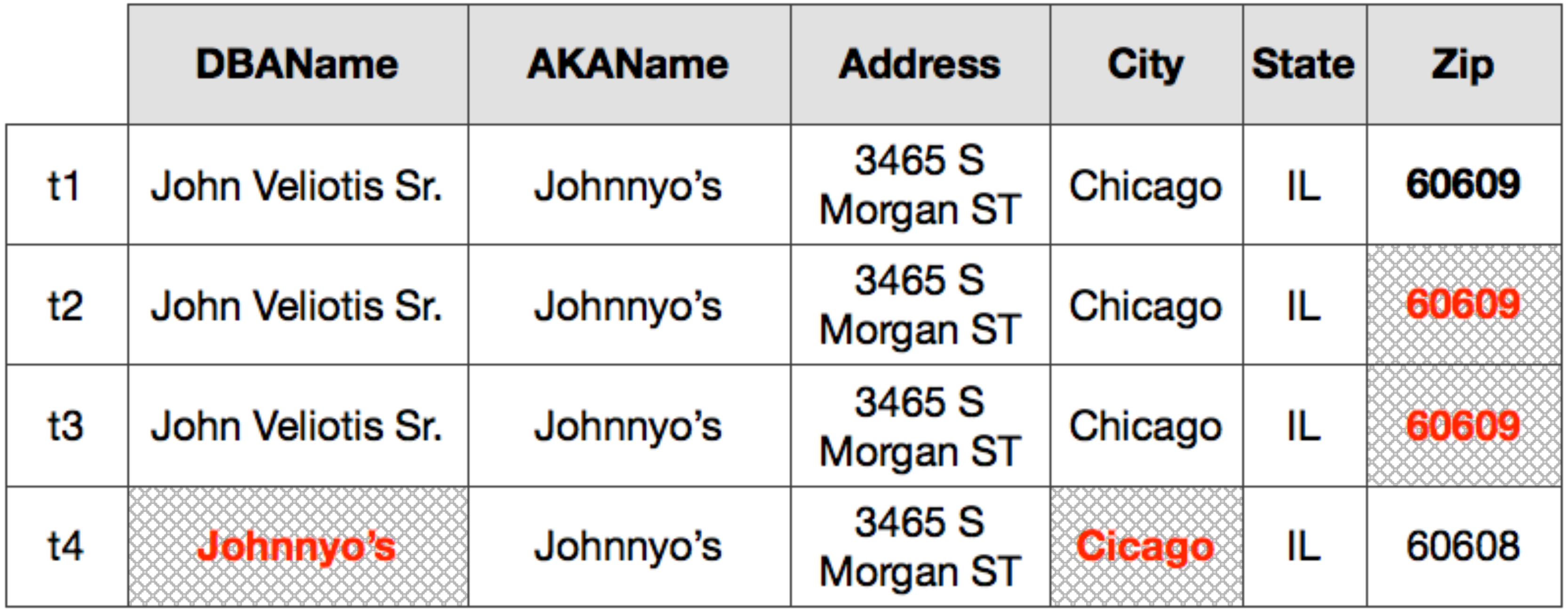
Constraints and minimality

Functional dependencies

c1: DBAName → Zip

c2: Zip → City, State

c3: City, State, Address → Zip



	DBAName	AKAName	Address	City	State	Zip
t1	John Veliotis Sr.	Johnnyo’s	3465 S Morgan ST	Chicago	IL	**60609**
t2	John Veliotis Sr.	Johnnyo’s	3465 S Morgan ST	Chicago	IL	**60609**
t3	John Veliotis Sr.	Johnnyo’s	3465 S Morgan ST	Chicago	IL	**60609**
t4	**Johnnyo’s**	Johnnyo’s	3465 S Morgan ST	**Cicago**	IL	60608

Error; correct zip code is 60608

Does not fix errors and introduces new ones.

External information

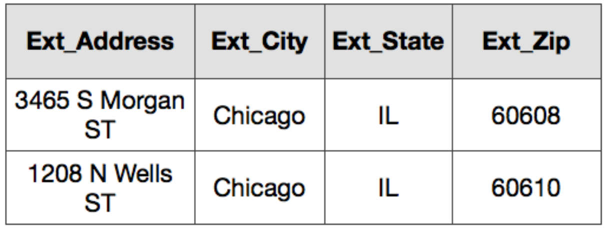
Matching dependencies

$$\text{m1: } \text{Zip} = \text{Ext\_Zip} \rightarrow \text{City} = \text{Ext\_City}$$

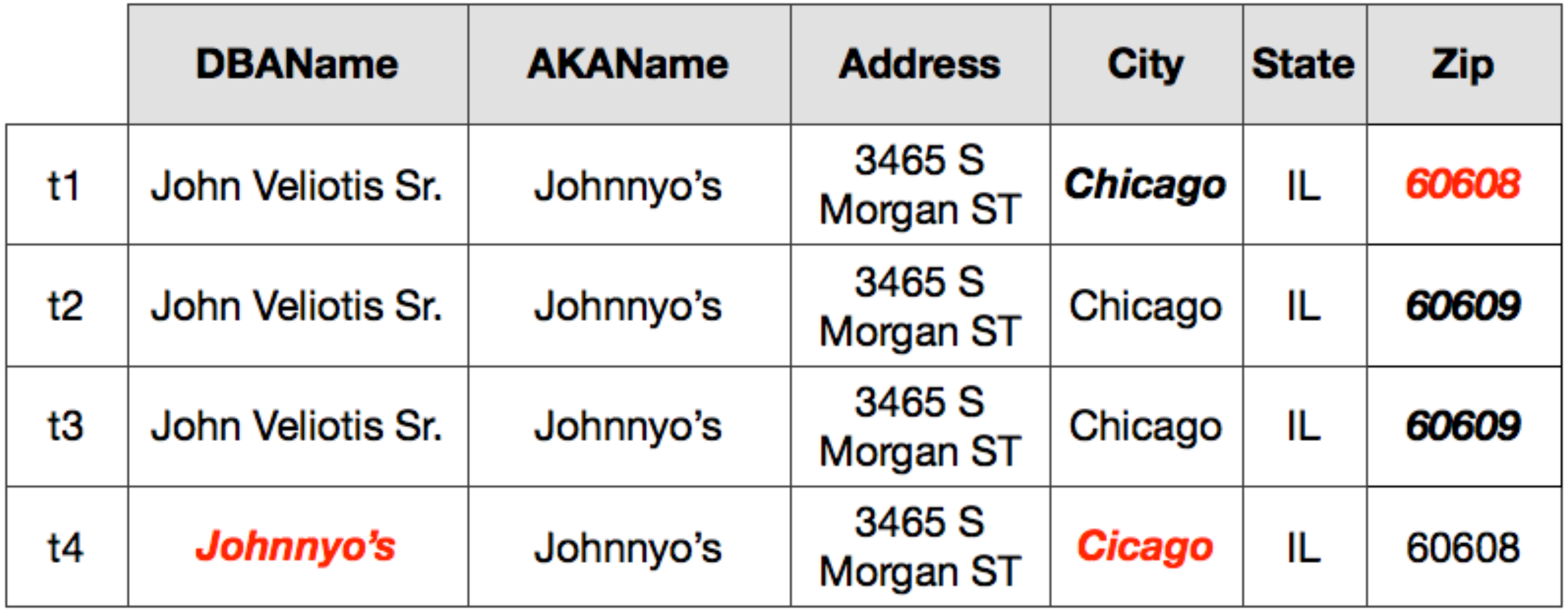
$$\text{m2: } \text{Zip} = \text{Ext\_Zip} \rightarrow \text{State} = \text{Ext\_State}$$

$$\text{m3: } \text{City} = \text{Ext\_City} \wedge \text{State} = \text{Ext\_State} \wedge \wedge \text{Address} = \text{Ext\_Address} \rightarrow \text{Zip} = \text{Ext\_Zip}$$

External list of addresses



Ext_Address	Ext_City	Ext_State	Ext_Zip
3465 S Morgan ST	Chicago	IL	60608
1208 N Wells ST	Chicago	IL	60610



	DBAName	AKAName	Address	City	State	Zip
t1	John Veliotis Sr.	Johnnyo's	3465 S Morgan ST	***Chicago***	IL	***60608***
t2	John Veliotis Sr.	Johnnyo's	3465 S Morgan ST	Chicago	IL	***60609***
t3	John Veliotis Sr.	Johnnyo's	3465 S Morgan ST	Chicago	IL	***60609***
t4	***Johnnyo's***	Johnnyo's	3465 S Morgan ST	***Cicago***	IL	60608

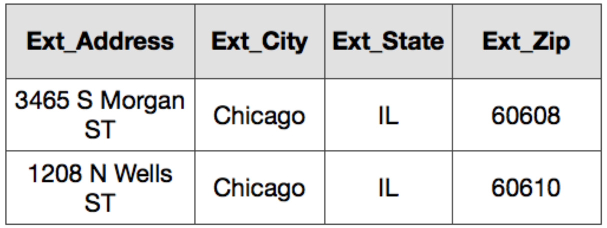
Fan et al., 2009; Bertossi et al., 2010; Chu et al., 2015

External information

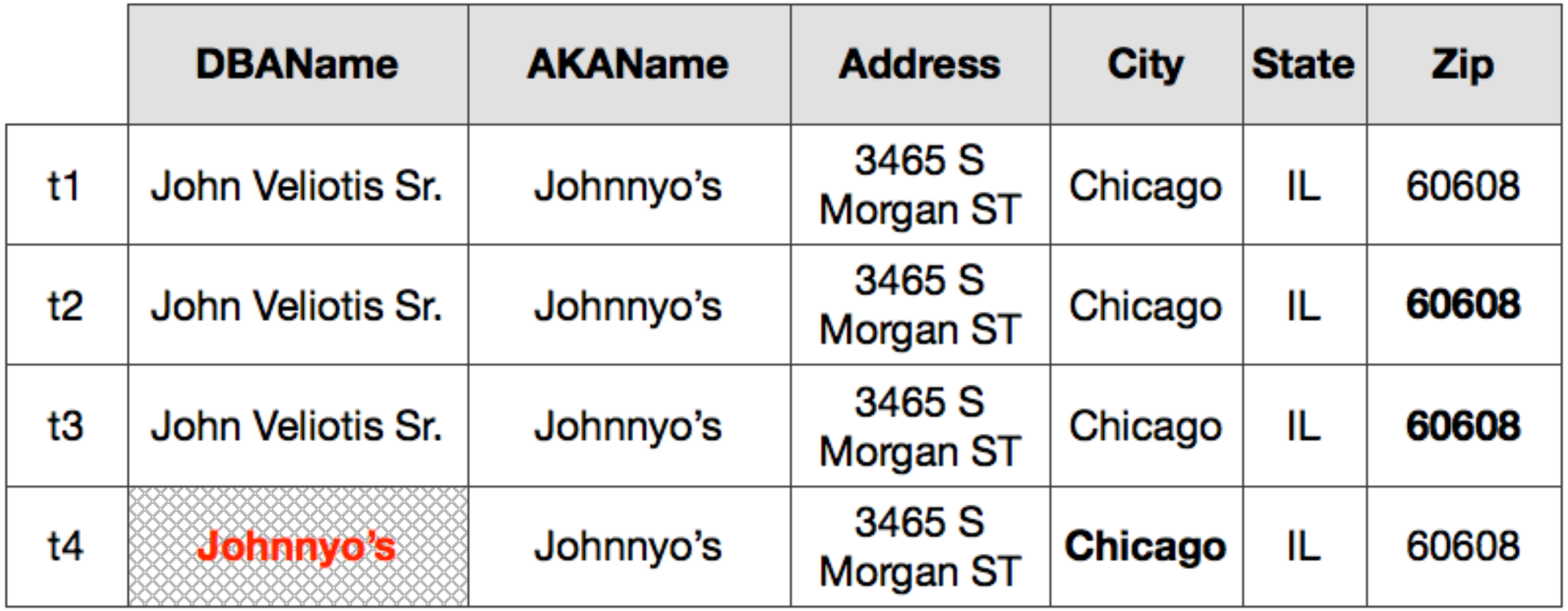
Matching dependencies

$$\begin{aligned}
&\text{m1: } \text{Zip} = \text{Ext\_Zip} \rightarrow \text{City} = \text{Ext\_City}\\
&\text{m2: } \text{Zip} = \text{Ext\_Zip} \rightarrow \text{State} = \text{Ext\_State}\\
&\text{m3: } \text{City} = \text{Ext\_City} \wedge \text{State} = \text{Ext\_State} \wedge\\
&\qquad \wedge\ \text{Address} = \text{Ext\_Address} \rightarrow \text{Zip} = \text{Ext\_Zip}
\end{aligned}$$

External list of addresses



Ext_Address	Ext_City	Ext_State	Ext_Zip
3465 S Morgan ST	Chicago	IL	60608
1208 N Wells ST	Chicago	IL	60610



	DBAName	AKAName	Address	City	State	Zip
t1	John Veliotis Sr.	Johnnyo's	3465 S Morgan ST	Chicago	IL	60608
t2	John Veliotis Sr.	Johnnyo's	3465 S Morgan ST	Chicago	IL	**60608**
t3	John Veliotis Sr.	Johnnyo's	3465 S Morgan ST	Chicago	IL	**60608**
t4	**Johnnyo's**	Johnnyo's	3465 S Morgan ST	**Chicago**	IL	60608

Action: Map external information to input dataset using matching dependencies and repair disagreements

External information

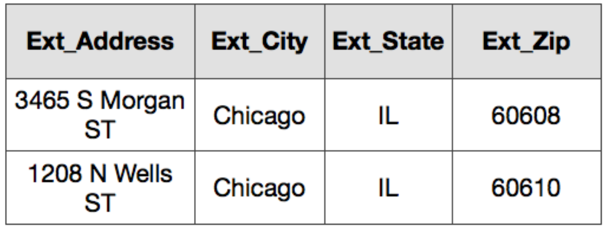
Matching dependencies

m1: $\text{Zip} = \text{Ext\_Zip} \rightarrow \text{City} = \text{Ext\_City}$

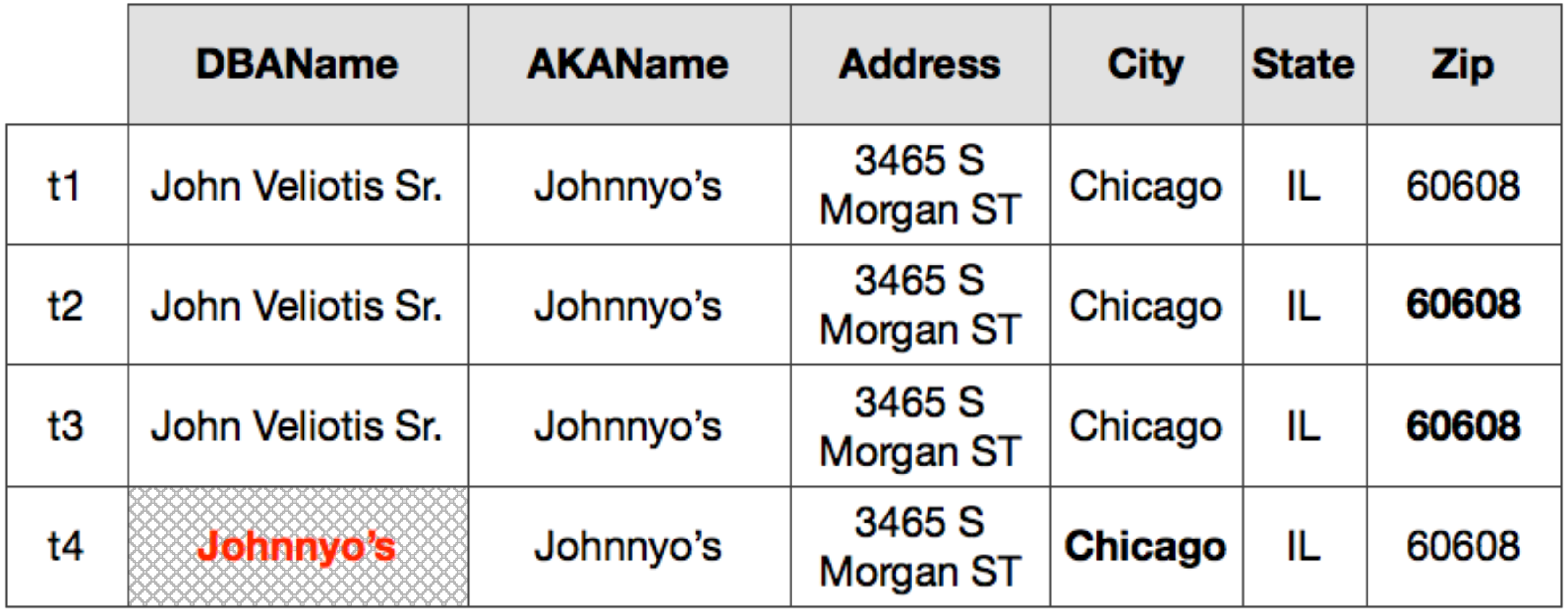
m2: $\text{Zip} = \text{Ext\_Zip} \rightarrow \text{State} = \text{Ext\_State}$

m3: $\text{City} = \text{Ext\_City} \wedge \text{State} = \text{Ext\_State} \wedge$
$\wedge\, \text{Address} = \text{Ext\_Address} \rightarrow \text{Zip} = \text{Ext\_Zip}$

External list of addresses



Ext_Address	Ext_City	Ext_State	Ext_Zip
3465 S Morgan ST	Chicago	IL	60608
1208 N Wells ST	Chicago	IL	60610



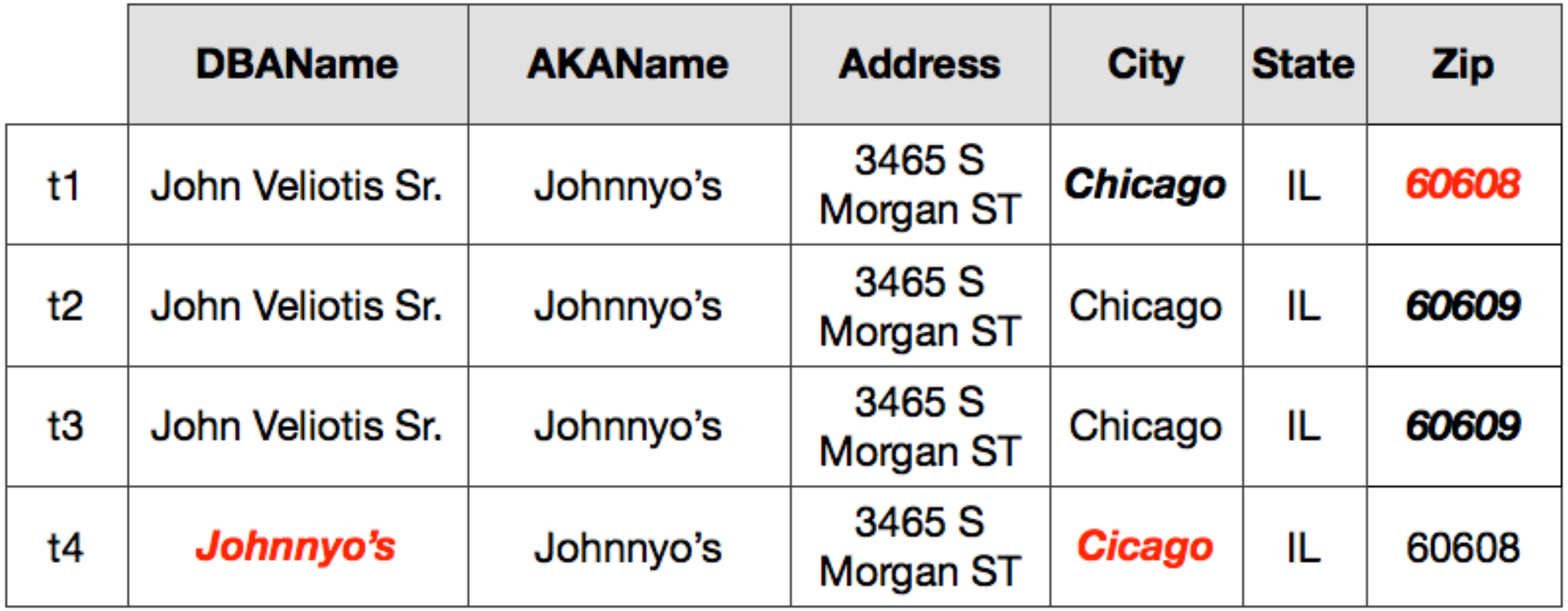
	DBAName	AKAName	Address	City	State	Zip
t1	John Veliotis Sr.	Johnnyo's	3465 S Morgan ST	Chicago	IL	60608
t2	John Veliotis Sr.	Johnnyo's	3465 S Morgan ST	Chicago	IL	**60608**
t3	John Veliotis Sr.	Johnnyo's	3465 S Morgan ST	Chicago	IL	**60608**
t4	**Johnnyo's**	Johnnyo's	3465 S Morgan ST	**Chicago**	IL	60608

External dictionaries may have limited coverage or not exist altogether

Quantitative statistics

Reason about co-occurrence of values across cells in a tuple

Estimate the distribution governing each attribute



	DBAName	AKAName	Address	City	State	Zip
t1	John Veliotis Sr.	Johnnyo's	3465 S Morgan ST	***Chicago***	IL	***60608***
t2	John Veliotis Sr.	Johnnyo's	3465 S Morgan ST	Chicago	IL	***60609***
t3	John Veliotis Sr.	Johnnyo's	3465 S Morgan ST	Chicago	IL	***60609***
t4	***Johnnyo's***	Johnnyo's	3465 S Morgan ST	***Cicago***	IL	60608

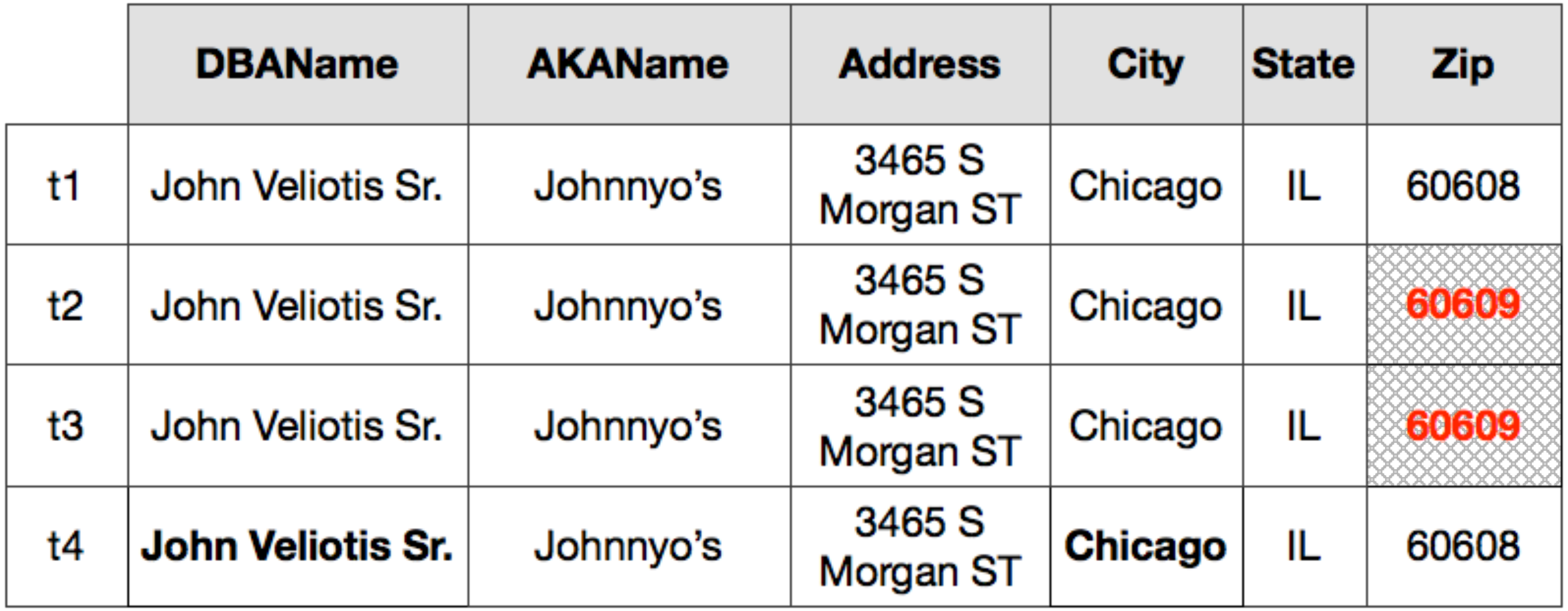
Example: Chicago co-occurs with IL

Hellerstein, 2008; Mayfield et al., 2010; Yakout et al., 2013

Quantitative statistics

Reason about co-occurrence of values across cells in a tuple

Estimate the distribution governing each attribute



	DBAName	AKAName	Address	City	State	Zip
t1	John Veliotis Sr.	Johnnyo’s	3465 S Morgan ST	Chicago	IL	60608
t2	John Veliotis Sr.	Johnnyo’s	3465 S Morgan ST	Chicago	IL	**60609**
t3	John Veliotis Sr.	Johnnyo’s	3465 S Morgan ST	Chicago	IL	**60609**
t4	**John Veliotis Sr.**	Johnnyo’s	3465 S Morgan ST	**Chicago**	IL	60608

Again, fails to repair the wrong zip code

Let’s combine everything

Constraints and minimality

	DBAName	AKAName	Address	City	State	Zip
t1	John Veliotis Sr.	Johnnyo’s	3465 S Morgan ST	Chicago	IL	**60609**
t2	John Veliotis Sr.	Johnnyo’s	3465 S Morgan ST	Chicago	IL	**60609**
t3	John Veliotis Sr.	Johnnyo’s	3465 S Morgan ST	Chicago	IL	**60609**
t4	**Johnnyo’s**	Johnnyo’s	3465 S Morgan ST	**Cicago**	IL	60608

External data

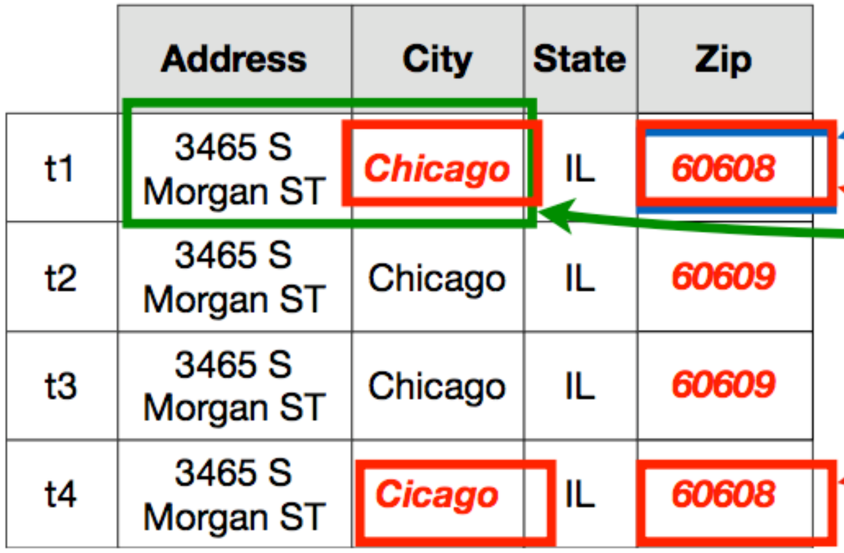
	DBAName	AKAName	Address	City	State	Zip
t1	John Veliotis Sr.	Johnnyo’s	3465 S Morgan ST	Chicago	IL	60608
t2	John Veliotis Sr.	Johnnyo’s	3465 S Morgan ST	Chicago	IL	**60608**
t3	John Veliotis Sr.	Johnnyo’s	3465 S Morgan ST	Chicago	IL	**60608**
t4	**Johnnyo’s**	Johnnyo’s	3465 S Morgan ST	**Chicago**	IL	60608

Quantitative statistics

	DBAName	AKAName	Address	City	State	Zip
t1	John Veliotis Sr.	Johnnyo’s	3465 S Morgan ST	Chicago	IL	60608
t2	John Veliotis Sr.	Johnnyo’s	3465 S Morgan ST	Chicago	IL	**60609**
t3	John Veliotis Sr.	Johnnyo’s	3465 S Morgan ST	Chicago	IL	**60609**
t4	**John Veliotis Sr.**	Johnnyo’s	3465 S Morgan ST	**Chicago**	IL	60608

Different solutions suggest different repairs

A probabilistic model for data repairs



	Address	City	State	Zip
t1	3465 S Morgan ST	Chicago	IL	60608
t2	3465 S Morgan ST	Chicago	IL	60609
t3	3465 S Morgan ST	Chicago	IL	60609
t4	3465 S Morgan ST	Cicago	IL	60608

Each cell is a random variable

Value co-occurences capture data statistics

Constraints introduce correlations

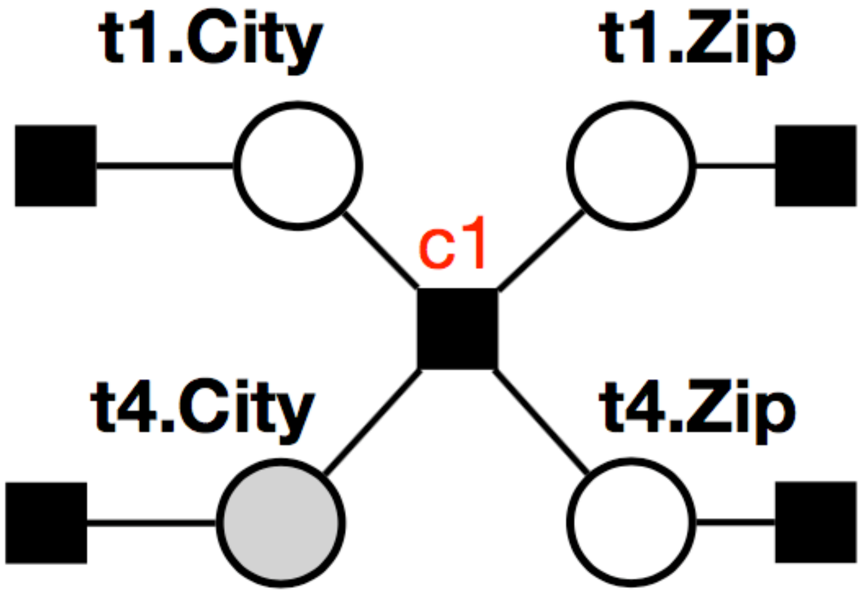
$$c1:\ \mathrm{Zip} \rightarrow \mathrm{City}$$

“Address= 3465 S Morgan St”

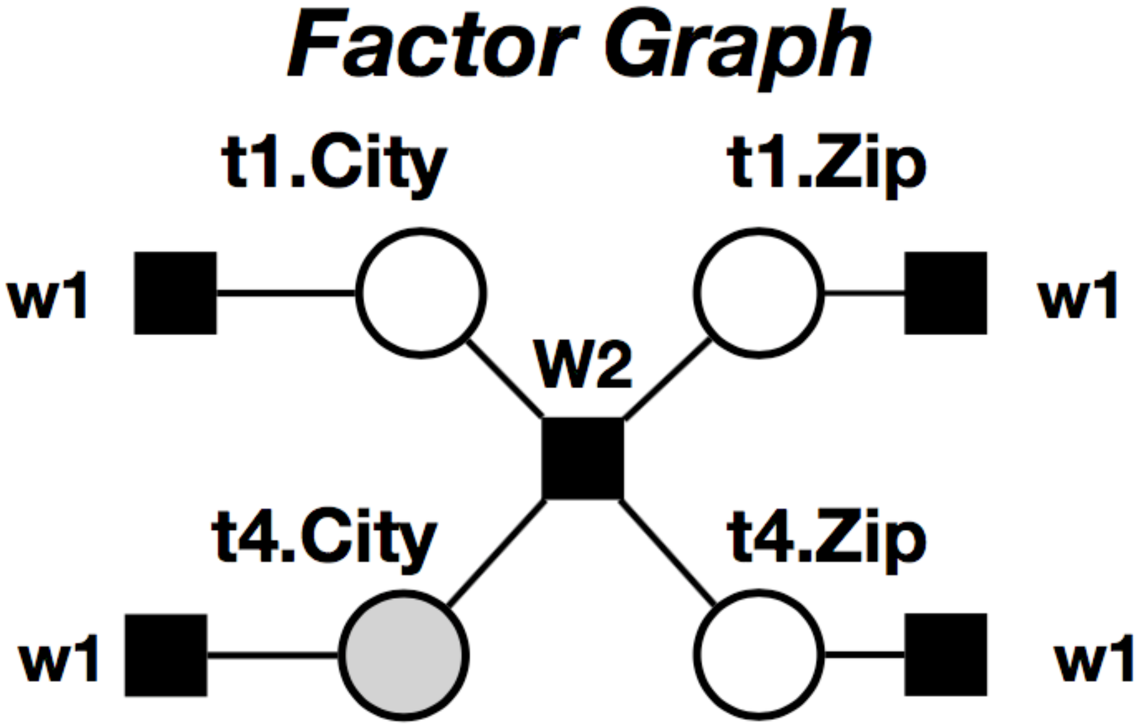
○ : Unknown (to be inferred) RV

● : Observed (fixed) RV

■ : Factor (encodes correlations)



Learning the model



HoloClean automatically generates a factor graph that captures:

- Co-occurences
- Correlations due to constraints
- Evidence due to external data

Exponential family (canonical form)

$$\mathbf{w} = (w_1, w_2, \ldots, w_s)^T$$

$$P(x|w) = \exp\left(\sum_{i=1}^{s} w_i T_i(x) - A(\mathbf{w})\right)$$

Repairing is a learning and inference problem:
Learn parameters w (use SGD) and infer the marginal distribution for unknown variables (use Gibbs sampling)