Record Linkage: Theory and Practice

William E. Yancey

william.e.yancey@census.gov

U.S. Census Bureau

U S C E N S U S B U R E A U

Introduction

- Definition

Introduction

- Definition
- Terminology

US CENSUS BUREAU

Introduction

- Definition
- Terminology
- Uses

Introduction

- Definition
- Terminology
- Uses
- Context

USCENSUSBUREAU

Definition

- A procedure to find pairs of records in two files that represent the same entity

US CENSUS BUREAU

Definition

- A procedure to find pairs of records in two files that represent the same entity
- When both files are the same file, the procedure is to find duplicate records

U S C E N S U S B U R E A U

Terminology

- Matched records: both records represent the same entity in truth

U S C E N S U S B U R E A U

Terminology

- Matched records: both records represent the same entity in truth
- Linked records: Both records are identified by record linkage procedure as probably representing the same entity

U S C E N S U S B U R E A U

Uses

- Updating and deduplicating a survey frame

U S C E N S U S B U R E A U

Uses

- Updating and deduplicating a survey frame
- Merging two files for microdata anlysis

U S C E N S U S B U R E A U

Uses

- Updating and deduplicating a survey frame
- Merging two files for microdata anlysis
- Determine confidentiality of microdata

U S C E N S U S B U R E A U

Uses

- Updating and deduplicating a survey frame
- Merging two files for microdata anlysis
- Determine confidentiality of microdata
- Measure a population by capture-recapture

Capture-Recapture

- Let A, B be independent random samples of sample space S

$$x_{11} = |A \cap B| \quad x_{10} = |A \setminus B|$$
$$x_{01} = |B \setminus A| \quad x_{00} = |S \setminus (A \cup B)|$$

U S C E N S U S B U R E A U

Capture-Recapture

- Let A, B be independent random samples of sample space S

$$x_{11} = |A \cap B| \quad x_{10} = |A \setminus B|$$
$$x_{01} = |B \setminus A| \quad x_{00} = |S \setminus (A \cup B)|$$

- Then

$$\hat{x}_{00} = E[x_{00}] = \frac{x_{1+}x_{+1}}{x_{11}}$$

U S C E N S U S B U R E A U

Capture-Recapture, Cont.

- Take two independent surveys of a region and estimate the number of people missed.

U S C E N S U S B U R E A U

Capture-Recapture, Cont.

- Take two independent surveys of a region and estimate the number of people missed.
- Note accuracy of estimate depends on accuracy of x_{11}, as determined by record linkage

U S C E N S U S B U R E A U

Capture-Recapture, Cont.

- Take two independent surveys of a region and estimate the number of people missed.
- Note accuracy of estimate depends on accuracy of x_{11}, as determined by record linkage
- Y.M.M. Bishop,S.E. Fienberg, P.W. Holland, *Discrete Multivariate Analysis, Theory and Priactice*, Chapter 6. MIT Press, 1975

U S C E N S U S B U R E A U

Record Linkage Basics

- Context

U S C E N S U S B U R E A U

Record Linkage Basics

- Context
- Deterministic Record Linkage

U S C E N S U S B U R E A U

Record Linkage Basics

- Context
- Deterministic Record Linkage
- Probabilistic Record Linkage

US CENSUS BUREAU

Record Linkage Basics

- Context
- Deterministic Record Linkage
- Probabilistic Record Linkage
- Not Statistical Matching

Record Linkage Basics

- Context
- Deterministic Record Linkage
- Probabilistic Record Linkage
- Not Statistical Matching
- Need for Automated Record Linkage

U S C E N S U S B U R E A U

Context

- Files have records of fixed length with fields of fixed length and position (or in a database with retrievable individual fields)

USCENSUSBUREAU

Context

- Files have records of fixed length with fields of fixed length and position (or in a database with retrievable individual fields)
- Not a search algorithm

Deterministic Record Linkage

- Records are linked when

U S C E N S U S B U R E A U

Deterministic Record Linkage

- Records are linked when
  - They agree exactly on all matching fields

Deterministic Record Linkage

- Records are linked when
  - They agree exactly on all matching fields
  - Or on predetermined portion of fields

U S C E N S U S B U R E A U

Probabilistic Record Linkage

- Assign a probabilistic weighting to record pairs

Probabilistic Record Linkage

- Assign a probabilistic weighting to record pairs
- Accepts record pairs with sufficiently high weights as linked pairs

U S C E N S U S B U R E A U

Not Statistical Matching

- Statistical matching: Bring together pairs of records with statistically similar characteristics, not necessarily representing the same entity

U S C E N S U S B U R E A U

Not Statistical Matching

- Statistical matching: Bring together pairs of records with statistically similar characteristics, not necessarily representing the same entity
- Usually for two files that represent different samples of a population

U S C E N S U S B U R E A U

Not Statistical Matching

- Statistical matching: Bring together pairs of records with statistically similar characteristics, not necessarily representing the same entity
- Usually for two files that represent different samples of a population
- Older practice than exact matching (deterministic or probabilistic)

U S C E N S U S B U R E A U

Need for Automated Record Linkage

Clerical matching is:

U S C E N S U S B U R E A U

Need for Automated Record Linkage

Clerical matching is:

- expensive

Need for Automated Record Linkage

Clerical matching is:

- expensive
- slow

U S C E N S U S B U R E A U

Need for Automated Record Linkage

Clerical matching is:

- expensive
- slow
- error prone

U S C E N S U S B U R E A U

Need for Automated Record Linkage

Clerical matching is:

- expensive
- slow
- error prone

	Clerical	1988	1990
Computer match proportion	0%	70%	75%
# clerks	3000	600	200
#months	6	1.5	1.5
False match rate	5%	0.5%	0.2%

U S C E N S U S B U R E A U

Rec. Link. Theory: Fellegi & Sunter

- Basic Definitions and Notation

Rec. Link. Theory: Fellegi & Sunter

- Basic Definitions and Notation
- Agreement Patterns

Rec. Link. Theory: Fellegi & Sunter

- Basic Definitions and Notation
- Agreement Patterns
- Example Comparison Space

U S C E N S U S B U R E A U

Rec. Link. Theory: Fellegi & Sunter

- Basic Definitions and Notation
- Agreement Patterns
- Example Comparison Space
- Conditional Probabilities

Rec. Link. Theory: Fellegi & Sunter

- Basic Definitions and Notation
- Agreement Patterns
- Example Comparison Space
- Conditional Probabilities
- Linkage Rule

U S C E N S U S B U R E A U

Rec. Link. Theory: Fellegi & Sunter

- Basic Definitions and Notation
- Agreement Patterns
- Example Comparison Space
- Conditional Probabilities
- Linkage Rule
- Error Rates

Rec. Link. Theory: Fellegi & Sunter

- Basic Definitions and Notation
- Agreement Patterns
- Example Comparison Space
- Conditional Probabilities
- Linkage Rule
- Error Rates
- Clerical Region

U S C E N S U S B U R E A U

Rec. Link. Theory: Fellegi & Sunter

- Basic Definitions and Notation
- Agreement Patterns
- Example Comparison Space
- Conditional Probabilities
- Linkage Rule
- Error Rates
- Clerical Region
- Fundamental Theorem

U S C E N S U S B U R E A U

Rec. Link. Theory: Fellegi & Sunter

- Basic Definitions and Notation
- Agreement Patterns
- Example Comparison Space
- Conditional Probabilities
- Linkage Rule
- Error Rates
- Clerical Region
- Fundamental Theorem
- Conditional Independence Assumption

Basic Definitions and Notation

- Sets of entities A, B

U S C E N S U S B U R E A U

Basic Definitions and Notation

- Sets of entities A, B
- Corresponding files of records $\alpha(A), \beta(B)$

U S C E N S U S B U R E A U

Basic Definitions and Notation

- Sets of entities A, B
- Corresponding files of records $\alpha\left(A\right), \beta\left(B\right)$
- Sample space $\alpha\left(A\right) \times \beta\left(B\right)$

U S C E N S U S B U R E A U

Basic Definitions and Notation

- Sets of entities A, B
- Corresponding files of records $\alpha(A), \beta(B)$
- Sample space $\alpha(A) \times \beta(B)$
- Matches

$$M = \{(\alpha(a), \beta(b)) \mid a = b\}$$

Basic Definitions and Notation

- Sets of entities A, B
- Corresponding files of records $\alpha\left(A\right), \beta\left(B\right)$
- Sample space $\alpha\left(A\right) \times \beta\left(B\right)$
- Matches

$$M = \left\{\left(\alpha\left(a\right), \beta\left(b\right)\right) \mid a = b\right\}$$

- Nonmatches

$$U = \left\{\left(\alpha\left(a\right), \beta\left(b\right)\right) \mid a \neq b\right\}$$

U S C E N S U S B U R E A U

Basic Definitions and Notation

- Sets of entities A, B

U S C E N S U S B U R E A U

Basic Definitions and Notation

- Sets of entities A, B
- Corresponding files of records $\alpha\left(A\right), \beta\left(B\right)$

U S C E N S U S B U R E A U

Basic Definitions and Notation

- Sets of entities A, B
- Corresponding files of records $\alpha(A), \beta(B)$
- Sample space $\alpha(A) \times \beta(B)$

U S C E N S U S B U R E A U

Basic Definitions and Notation

- Sets of entities A, B
- Corresponding files of records $\alpha(A), \beta(B)$
- Sample space $\alpha(A) \times \beta(B)$
- Matches

$$M = \{(\alpha(a), \beta(b)) \mid a = b\}$$

Basic Definitions and Notation

- Sets of entities A, B
- Corresponding files of records $\alpha(A), \beta(B)$
- Sample space $\alpha(A) \times \beta(B)$
- Matches

$$M = \{(\alpha(a), \beta(b)) \mid a = b\}$$

- Nonmatches

$$U = \{(\alpha(a), \beta(b)) \mid a \neq b\}$$

U S C E N S U S B U R E A U

Agreement Patterns

- Comparison space

$$\alpha(A) \times \beta(B) \rightarrow \Gamma$$

Agreement Patterns

- Comparison space

$$\alpha(A) \times \beta(B) \rightarrow \Gamma$$

- Comparison vector

$$\gamma \in \Gamma$$

U S C E N S U S B U R E A U

Agreement Patterns

- Comparison space

$$\alpha(A) \times \beta(B) \rightarrow \Gamma$$

- Comparison vector

$$\gamma \in \Gamma$$

- Each component of comparison vector can take on finitely many values, typically two

U S C E N S U S B U R E A U

Agreement Patterns

- Comparison space

$$\alpha(A) \times \beta(B) \rightarrow \Gamma$$

- Comparison vector

$$\gamma \in \Gamma$$

- Each component of comparison vector can take on finitely many values, typically two

$$\begin{aligned} \gamma &= (\gamma_1, \gamma_2, \ldots, \gamma_n) \\ \gamma_i &\in \{0, 1\} \end{aligned}$$

Example Comparison Space

- Consider 3 binary comparisons

Example Comparison Space

- Consider 3 binary comparisons
  - γ_1 pair agrees on last name

U S C E N S U S B U R E A U

Example Comparison Space

- Consider 3 binary comparisons
  - γ_1 pair agrees on last name
  - γ_2 pair agrees on first name

U S C E N S U S B U R E A U

Example Comparison Space

- Consider 3 binary comparisons
  - γ_1 pair agrees on last name
  - γ_2 pair agrees on first name
  - γ_3 pair agrees on street name

U S C E N S U S B U R E A U

Example Comparison Space

- Consider 3 binary comparisons
  - γ_1 pair agrees on last name
  - γ_2 pair agrees on first name
  - γ_3 pair agrees on street name
- Sample agreement pattern

$$\gamma = (1, 0, 1)$$

Conditional Probabilities

- Probability that a record pair has agreement pattern γ, given that it is a match/nonmatch

$$\Pr(\gamma|M)$$
$$\Pr(\gamma|U)$$

U S C E N S U S B U R E A U

Conditional Probabilities

- Probability that a record pair has agreement pattern γ, given that it is a match/nonmatch

$$\Pr(\gamma|M)$$
$$\Pr(\gamma|U)$$

- Agreement ratio

$$R(\gamma) = \frac{\Pr(\gamma|M)}{\Pr(\gamma|U)}$$

US CENSUS BUREAU

Conditional Probabilities

- Probability that a record pair has agreement pattern γ, given that it is a match/nonmatch

$$\Pr(\gamma|M)$$
$$\Pr(\gamma|U)$$

- Agreement ratio

$$R(\gamma) = \frac{\Pr(\gamma|M)}{\Pr(\gamma|U)}$$

- Conditioned on the unobservable truth

U S C E N S U S B U R E A U

Linkage Rule

Designate a record pair's status based on its agreement pattern

U S C E N S U S B U R E A U

Linkage Rule

Designate a record pair's status based on its agreement pattern

- Link

Linkage Rule

Designate a record pair's status based on its agreement pattern

- Link
- Non-link

U S C E N S U S B U R E A U

Linkage Rule

Designate a record pair's status based on its agreement pattern

- Link
- Non-link
- Undecided

U S C E N S U S B U R E A U

Linkage Rule

Designate a record pair's status based on its agreement pattern

- Link
- Non-link
- Undecided

$$L : \Gamma \rightarrow \{L, N, C\}$$

Error Rates

- False match–a linked pair that is not a match

Error Rates

- False match–a linked pair that is not a match
- False nonmatch–a nonlinked pair that is a match

U S C E N S U S B U R E A U

Error Rates

- False match–a linked pair that is not a match
- False nonmatch–a nonlinked pair that is a match
- False match rate–probability that a designated link is a nonmatch

Error Rates

- False match–a linked pair that is not a match
- False nonmatch–a nonlinked pair that is a match
- False match rate–probability that a designated link is a nonmatch

$$\mu = \Pr\left(L|U\right)$$

U S C E N S U S B U R E A U

Error Rates

- False match–a linked pair that is not a match
- False nonmatch–a nonlinked pair that is a match
- False match rate–probability that a designated link is a nonmatch

$$\mu = \Pr\left(L|U\right)$$

- False nonmatch rate–probability that a designated nonlink is a match

U S C E N S U S B U R E A U

Error Rates

- False match–a linked pair that is not a match
- False nonmatch–a nonlinked pair that is a match
- False match rate–probability that a designated link is a nonmatch

$$\mu = \Pr(L|U)$$

- False nonmatch rate–probability that a designated nonlink is a match

$$\lambda = \Pr(N|M)$$

Error Rates, Cont.

If all pairs of records are designated link or nonlink

U S C E N S U S B U R E A U

Error Rates, Cont.

If all pairs of records are designated link or nonlink

	Match	Nonmatch
Link	$1-\lambda$	$\mu = \Pr(L\vert U)$
Nonlink	$\lambda = \Pr(N\vert M)$	$1-\mu$

U S C E N S U S B U R E A U

Clerical Region

- The set C of record pairs which are designated neither probable link nor probable nonlink by the linkage rule

U S C E N S U S B U R E A U

Clerical Region

- The set C of record pairs which are designated neither probable link nor probable nonlink by the linkage rule
- The match status of these pairs is left to clerical review

U S C E N S U S B U R E A U

Fundamental Theorem

- Fellegi & Sunter ("*A Theory for Record Linkage*", JASA, December,1969)

U S C E N S U S B U R E A U

Fundamental Theorem

- Fellegi & Sunter ("*A Theory for Record Linkage*", JASA, December,1969)
- Order the comparison vectors $\{\gamma^j\}$ by their agreement ratios $R\left(\gamma^j\right)$

U S C E N S U S B U R E A U

Fundamental Theorem

- Fellegi & Sunter ("*A Theory for Record Linkage*", JASA, December,1969)
- Order the comparison vectors $\{\gamma^j\}$ by their agreement ratios $R\left(\gamma^j\right)$
- Choose upper T_μ and lower T_λ cutoff values for $R\left(\gamma\right)$

U S C E N S U S B U R E A U

Fundamental Theorem, Cont.

Linkage rule:

Fundamental Theorem, Cont.

Linkage rule:

- Pairs with $R\left(\gamma^j\right) \geq T_\mu$ are designated links

US CENSUS BUREAU

Fundamental Theorem, Cont.

Linkage rule:

- Pairs with $R\left(\gamma^j\right) \geq T_\mu$ are designated links
- Pairs with $R\left(\gamma^j\right) \leq T_\lambda$ are designated nonlinks

U S C E N S U S B U R E A U

Fundamental Theorem, Cont.

Linkage rule:

- Pairs with $R\left(\gamma^{j}\right) \geq T_{\mu}$ are designated links
- Pairs with $R\left(\gamma^{j}\right) \leq T_{\lambda}$ are designated nonlinks
- Pairs with $T_{\lambda} < R\left(\gamma^{j}\right) < T_{\mu}$ are in the clerical region

US CENSUS BUREAU

Fundamental Theorem, Cont.

The error rates for this linkage rule are

U S C E N S U S B U R E A U

Fundamental Theorem, Cont.

The error rates for this linkage rule are

$$\mu = \sum_{R(\gamma^j) \geq T_\mu} \Pr\left(\gamma^j | U\right)$$

U S C E N S U S B U R E A U

Fundamental Theorem, Cont.

The error rates for this linkage rule are

$$\mu = \sum_{R(\gamma^j) \geq T_\mu} \Pr\left(\gamma^j | U\right)$$

$$\lambda = \sum_{R(\gamma^j) \leq T_\lambda} \Pr\left(\gamma^j | M\right)$$

U S C E N S U S B U R E A U

Fundamental Theorem, Cont.

- Theorem: For these error rates μ, λ, this is the optimal linkage rule, in the sense of producing the minimum size critical region

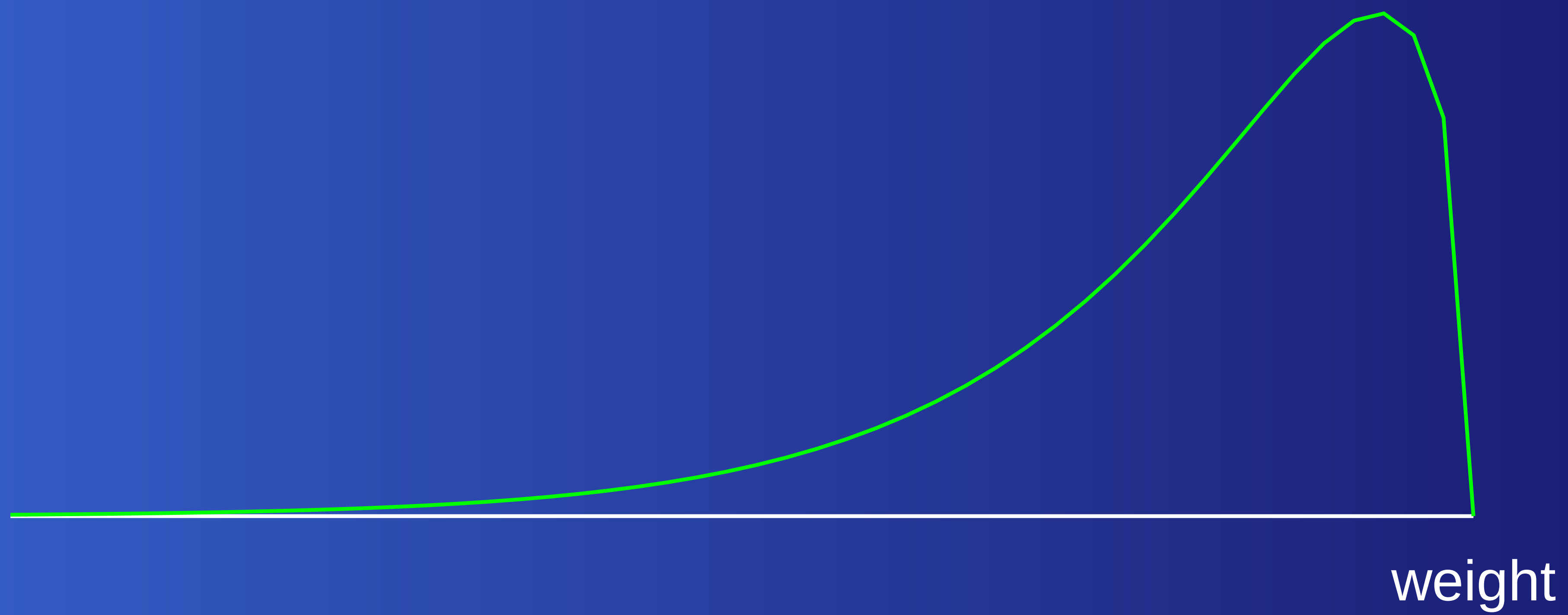
US CENSUS BUREAU

Fundamental Theorem, Cont.

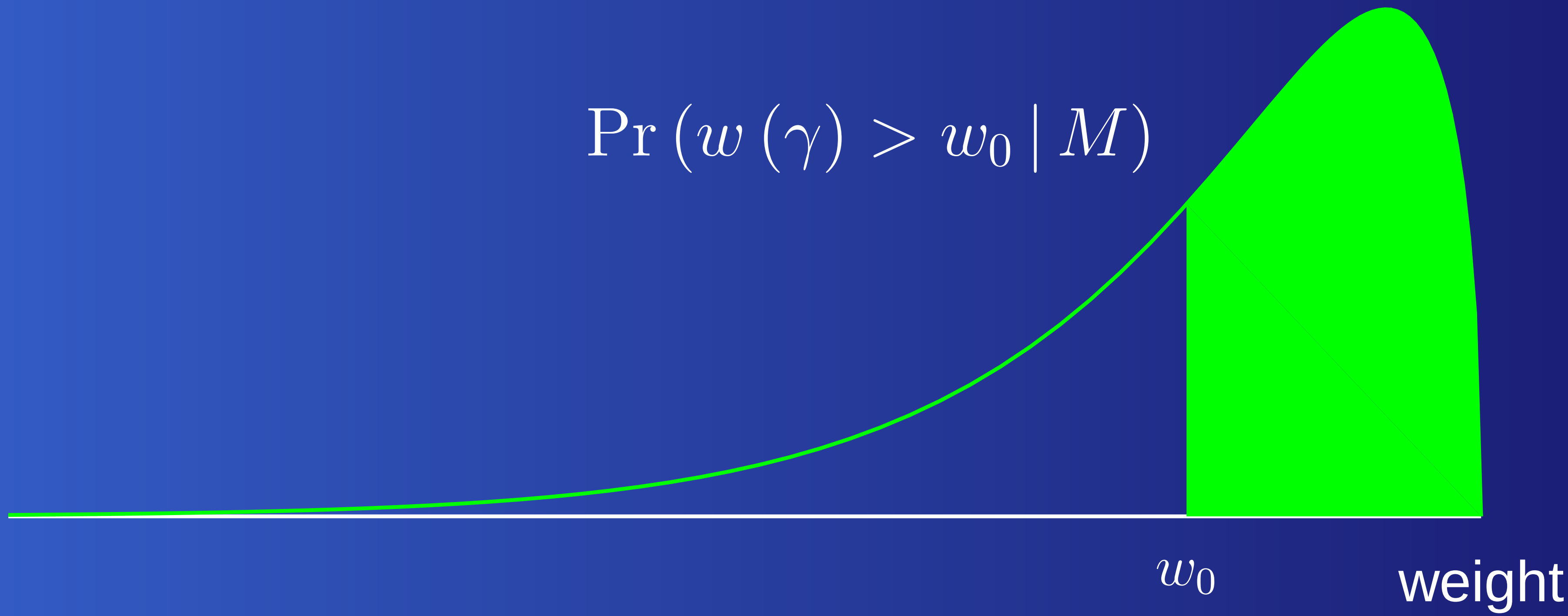
- Theorem: For these error rates μ, λ, this is the optimal linkage rule, in the sense of producing the minimum size critical region
- In other words, for a given error bound tolerance, this rule make as many linkage decisions as possible

U S C E N S U S B U R E A U

Weight Distribution for Matches

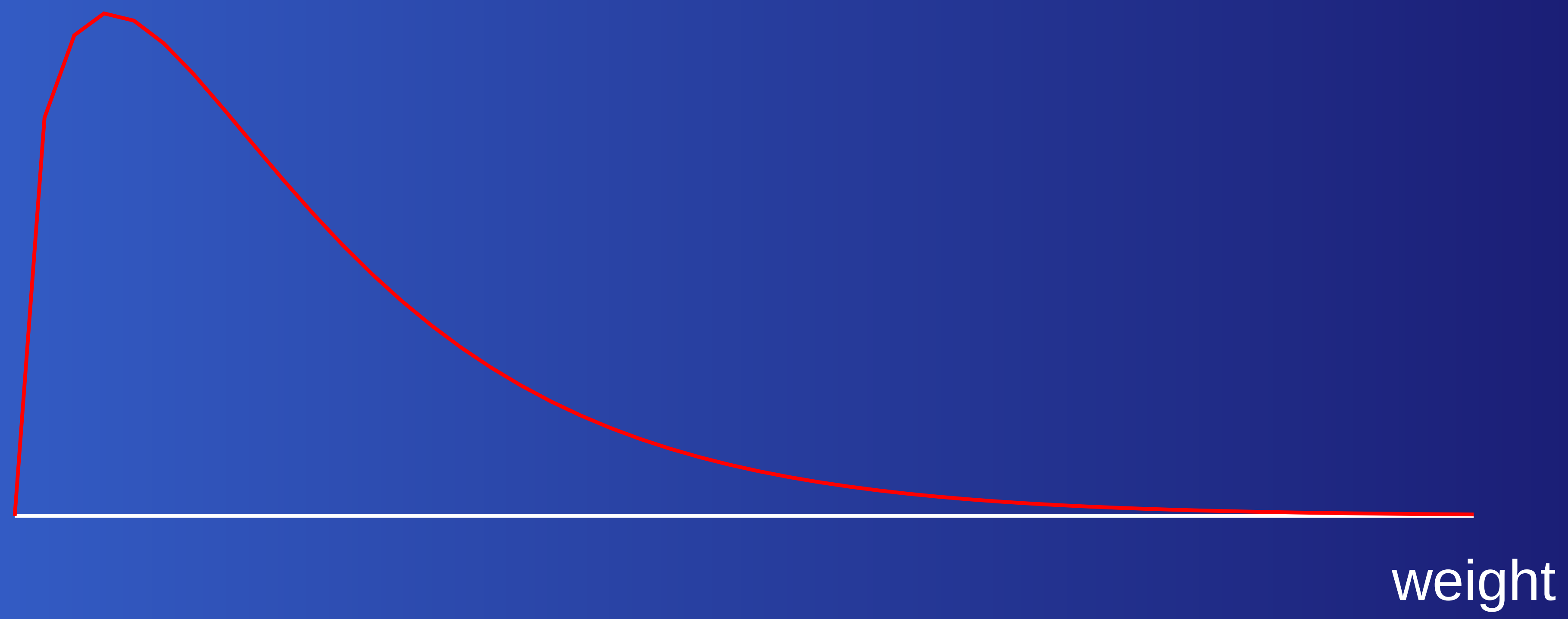


Weight Distribution for Matches



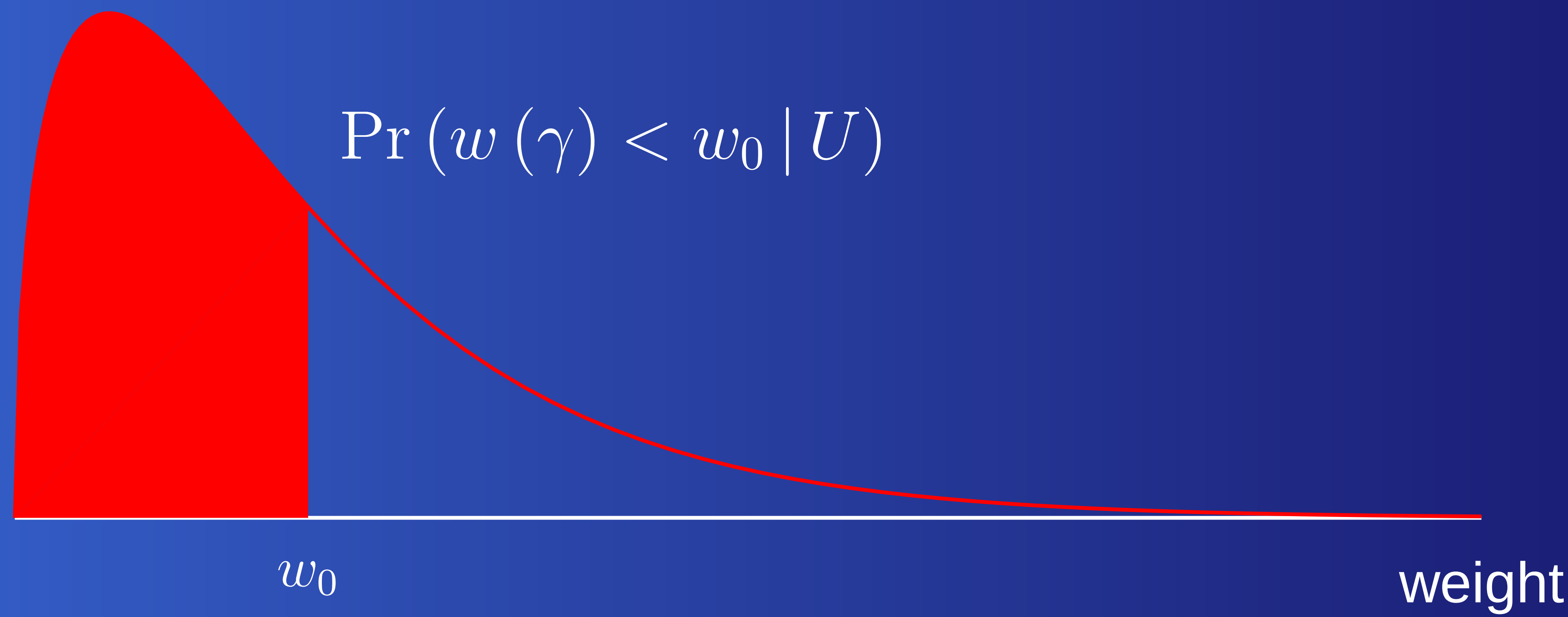
U S C E N S U S B U R E A U

Weight Distribution for Non-Matches

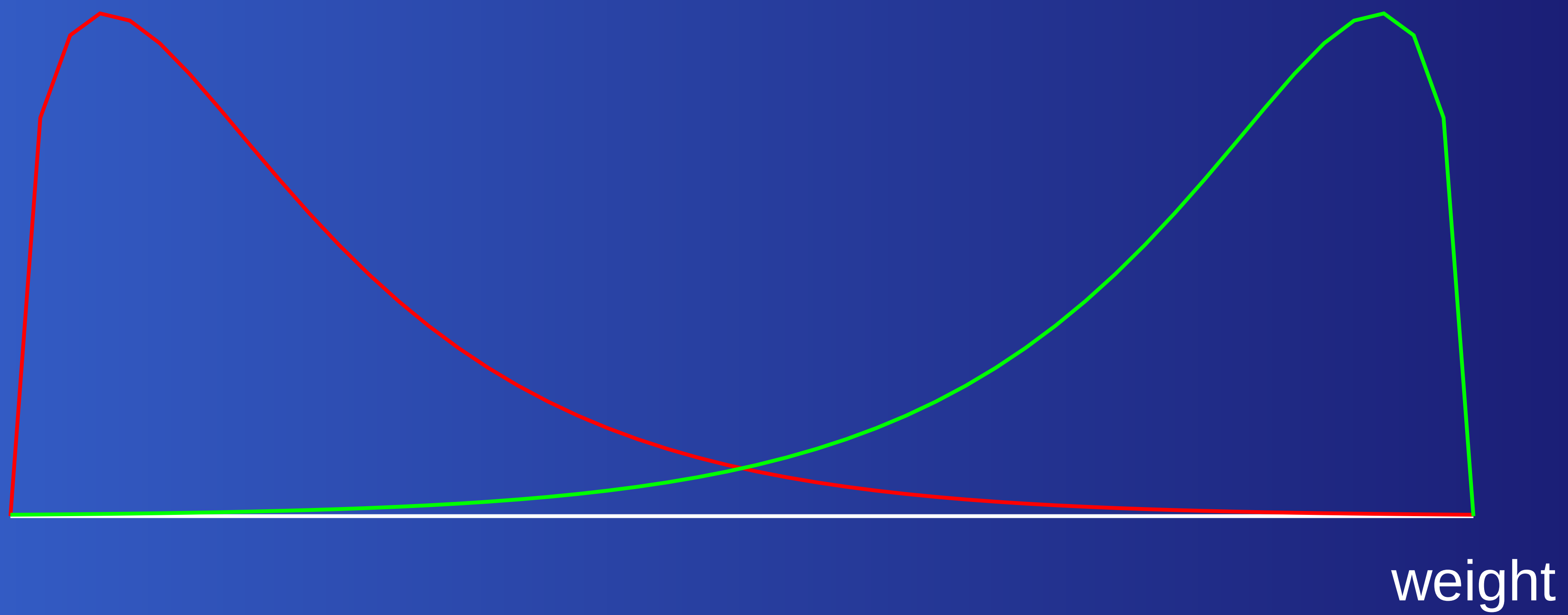


US CENSUS BUREAU

Weight Distribution for Non-Matches

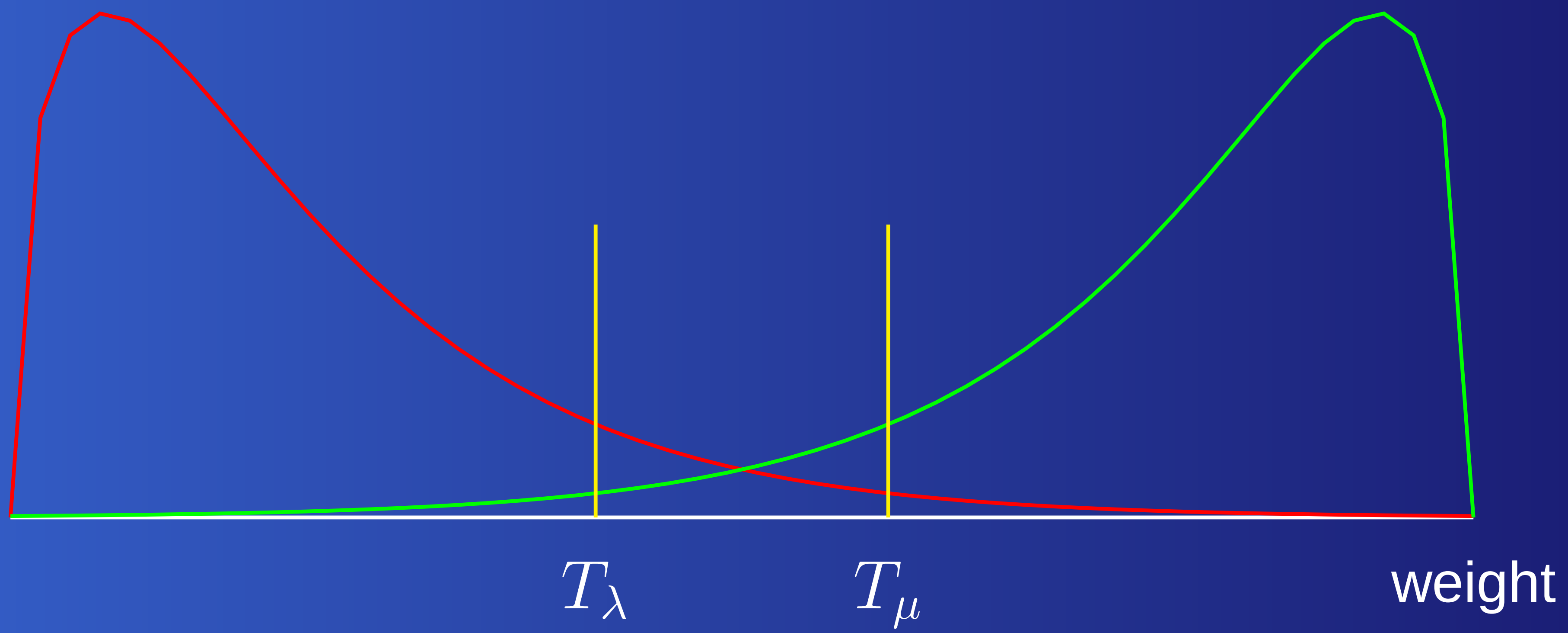


Idealized Distributions



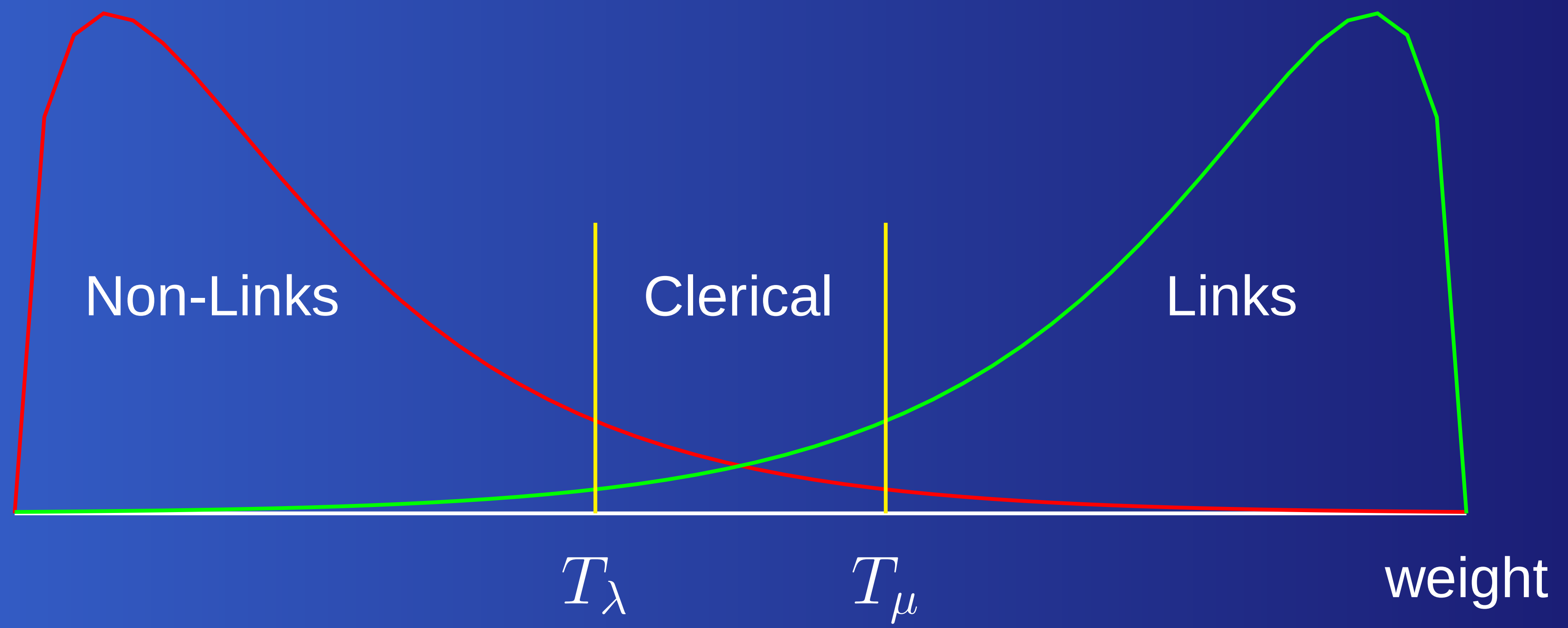
U S C E N S U S B U R E A U

Idealized Distributions

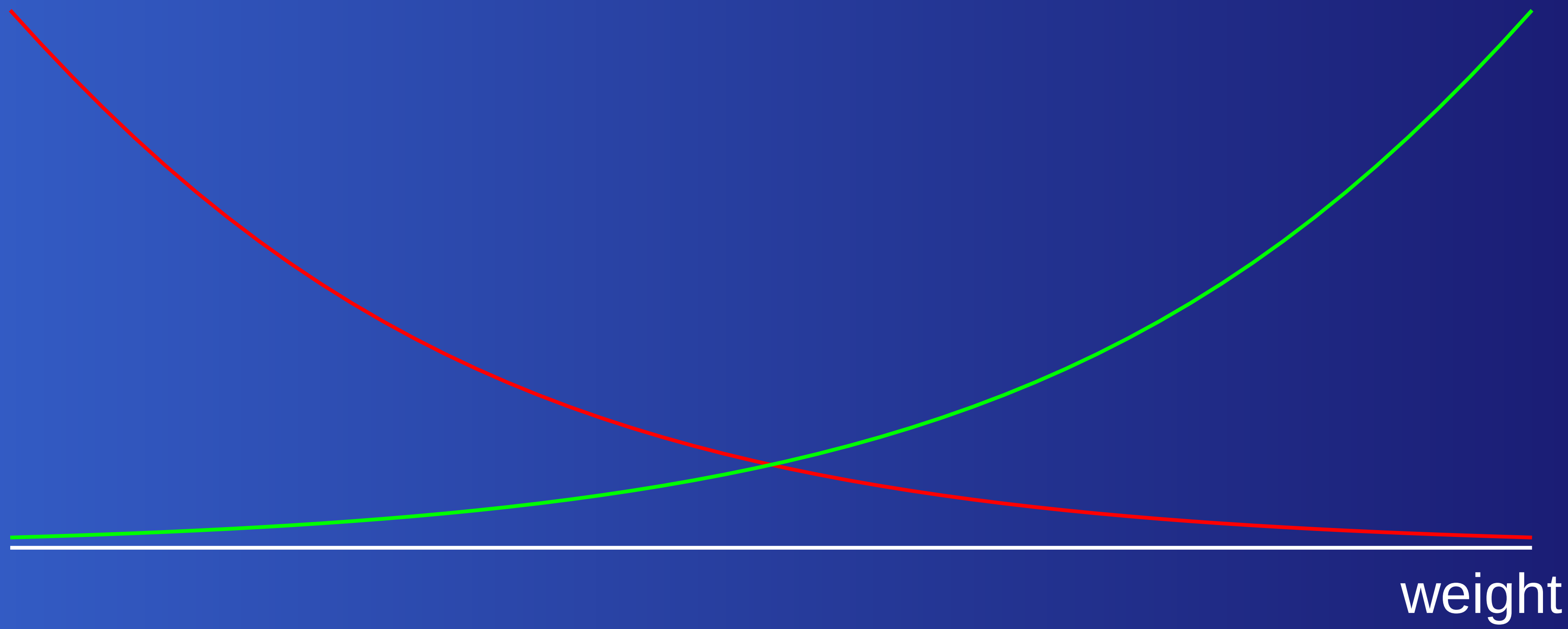


U S C E N S U S B U R E A U

Idealized Distributions

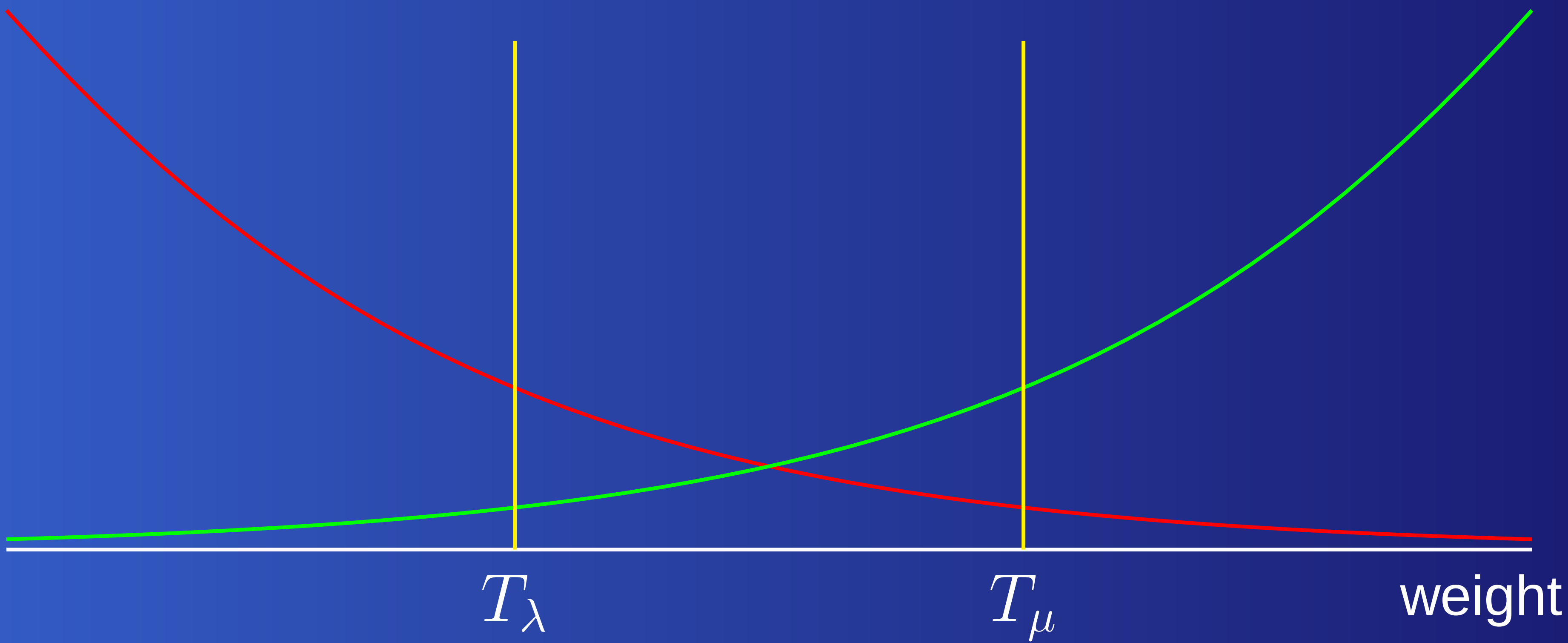


Error Rates, Clerical Review Region

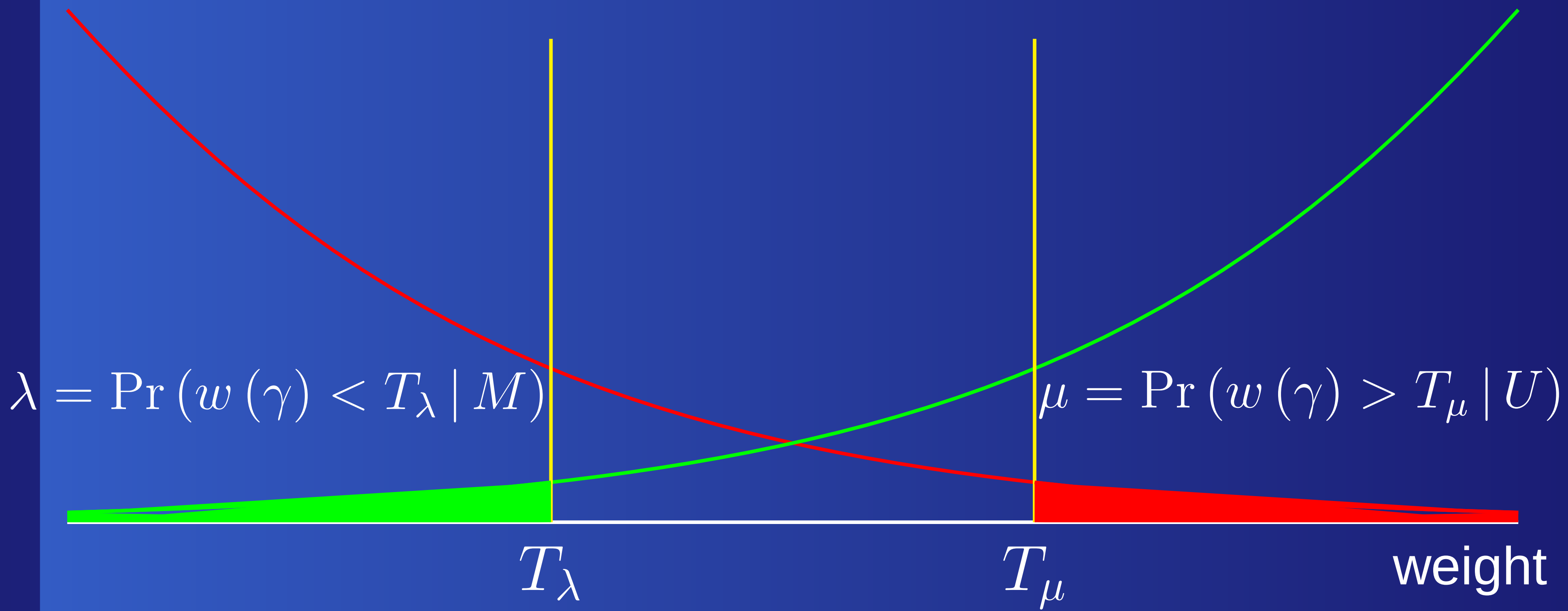


U S C E N S U S B U R E A U

Error Rates, Clerical Review Region



Error Rates, Clerical Review Region



U S C E N S U S B U R E A U

Conditional Independence Assumption

- To facilitate computation of conditional probabilities, Fellegi & Sunter assume conditional independence of comparison vector components

U S C E N S U S B U R E A U

Conditional Independence Assumption

- To facilitate computation of conditional probabilities, Fellegi & Sunter assume conditional independence of comparison vector components
- For $\gamma = (\gamma_1, \gamma_2, \ldots, \gamma_n)$, assume

$$\begin{aligned} \Pr(\gamma|M) &= \prod_{i=1}^{n} \Pr(\gamma_i|M) \\ \Pr(\gamma|U) &= \prod_{i=1}^{n} \Pr(\gamma_i|U) \end{aligned}$$

U S C E N S U S B U R E A U

Cond. Indep. Assumption, Cont.

- The factors $\Pr\left(\gamma_i|M\right), \Pr\left(\gamma_i|U\right)$ are called *marginal probabilities*

US CENSUS BUREAU

Cond. Indep. Assumption, Cont.

- The factors $\Pr(\gamma_i|M), \Pr(\gamma_i|U)$ are called *marginal probabilities*
- The ratio

$$\frac{\Pr(\gamma_i|M)}{\Pr(\gamma_i|U)}$$

determines the *distinguishing power* of the comparison γ_i

U S C E N S U S B U R E A U

Cond. Indep. Assumption, Cont.

Under conditional independence assumption, it is convenient to compute the *weight* of the comparison vector

U S C E N S U S B U R E A U

Cond. Indep. Assumption, Cont.

Under conditional independence assumption, it is convenient to compute the *weight* of the comparison vector

$$
\begin{aligned}
w(\gamma) &= \log R(\gamma) \\
&= \sum_{i=1}^{n} \frac{\log \Pr(\gamma_i | M)}{\log \Pr(\gamma_i | U)} \\
&= \sum_{i=1}^{n} w(\gamma_i)
\end{aligned}
$$

U S C E N S U S B U R E A U

Cond. Indep. Assumption, Cont.

- Motivation: Reduce the number of parameters

U S C E N S U S B U R E A U

Cond. Indep. Assumption, Cont.

- Motivation: Reduce the number of parameters
- For n binary comparisons and two conditional classes M, U, there are 2^{n+1} parameters

U S C E N S U S B U R E A U

Cond. Indep. Assumption, Cont.

- Motivation: Reduce the number of parameters
- For n binary comparisons and two conditional classes M, U, there are 2^{n+1} parameters
  - 2^n comparison vectors

Cond. Indep. Assumption, Cont.

- Motivation: Reduce the number of parameters
- For n binary comparisons and two conditional classes M, U, there are 2^{n+1} parameters
  - 2^n comparison vectors
  - 2 conditional probabilities for each vector

U S C E N S U S B U R E A U

Cond. Indep. Assumption, Cont.

- Motivation: Reduce the number of parameters
- For n binary comparisons and two conditional classes M, U, there are 2^{n+1} parameters
  - 2^n comparison vectors
  - 2 conditional probabilities for each vector
- Under conditional independence assumption, there are $2n$ parameters

U S C E N S U S B U R E A U

Cond. Indep. Assumption, Cont.

- Motivation: Reduce the number of parameters
- For n binary comparisons and two conditional classes M, U, there are 2^{n+1} parameters
  - 2^n comparison vectors
  - 2 conditional probabilities for each vector
- Under conditional independence assumption, there are $2n$ parameters
- Rationale: Given M, errors producing disagreement should be random

U S C E N S U S B U R E A U

Cond. Indep. Assumption, Cont.

- Often computable in closed form

Cond. Indep. Assumption, Cont.

- Often computable in closed form
- Can produce good decision rules even if model inaccurate

U S C E N S U S B U R E A U

Cond. Indep. Assumption, Cont.

- Often computable in closed form
- Can produce good decision rules even if model inaccurate
- Refered to as *naive Bayes* in machine learning

U S C E N S U S B U R E A U

Conditional Independence Example

- Suppose

$$\begin{aligned}
\Pr(\gamma_1 = 1 | M) &= 0.9 \\
\Pr(\gamma_2 = 1 | M) &= 0.8 \\
\Pr(\gamma_3 = 1 | M) &= 0.7
\end{aligned}$$

Conditional Independence Example

- Suppose

$$\begin{aligned} \Pr(\gamma_1 = 1|M) &= 0.9 \\ \Pr(\gamma_2 = 1|M) &= 0.8 \\ \Pr(\gamma_3 = 1|M) &= 0.7 \end{aligned}$$

- Then for $\gamma = (1, 0, 1)$,

$$\Pr(\gamma|M) = 0.9 * 0.2 * 0.7 = 0.126$$

Conditional Independence Example

- Suppose

$$\Pr\left(\gamma_1|M\right) = 0.8 \quad \Pr\left(\gamma_1|U\right) = 0.1$$
$$\Pr\left(\gamma_2|M\right) = 0.9 \quad \Pr\left(\gamma_2|U\right) = 0.3$$

U S C E N S U S B U R E A U

Conditional Independence Example

- Suppose

$$\Pr(\gamma_1|M) = 0.8 \quad \Pr(\gamma_1|U) = 0.1$$
$$\Pr(\gamma_2|M) = 0.9 \quad \Pr(\gamma_2|U) = 0.3$$

- Then

$$\frac{\Pr(\gamma_1|M)}{\Pr(\gamma_1|U)} = 8.0$$
$$\frac{\Pr(\gamma_2|M)}{\Pr(\gamma_2|U)} = 3.0$$

U S C E N S U S B U R E A U

Fellegi-Sunter Summary

- Choose conditional probability parameters

Fellegi-Sunter Summary

- Choose conditional probability parameters
- Conduct field comparisons on record pairs

U S C E N S U S B U R E A U

Fellegi-Sunter Summary

- Choose conditional probability parameters
- Conduct field comparisons on record pairs
- Classify record pairs based on weight of comparison vector

U S C E N S U S B U R E A U

Record Linkage Methodology

- Parameter estimation

Record Linkage Methodology

- Parameter estimation
  - EM Algorithm

U S C E N S U S B U R E A U

Record Linkage Methodology

- Parameter estimation
  - EM Algorithm
- Blocking

Choosing Parameters

- Informal

USCENSUSBUREAU

Choosing Parameters

- Informal
- EM Algorithm

U S C E N S U S B U R E A U

Choosing Parameters

- Informal
- EM Algorithm
- Other Methods

U S C E N S U S B U R E A U

Informal Methods

- Guess

Informal Methods

- Guess

$$0 < \Pr(\gamma|U) < \Pr(\gamma|M) < 1$$

U S C E N S U S B U R E A U

Informal Methods

- Guess

$$0 < \Pr(\gamma | U) < \Pr(\gamma | M) < 1$$

- Approximate

U S C E N S U S B U R E A U

Informal Methods

- Guess

$$0 < \Pr(\gamma|U) < \Pr(\gamma|M) < 1$$

- Approximate

$$\Pr(\gamma|U) \approx \Pr(\gamma|S)$$

Informal Methods

- Guess

$$0 < \Pr\left(\gamma|U\right) < \Pr\left(\gamma|M\right) < 1$$

- Approximate

$$\Pr\left(\gamma|U\right) \approx \Pr\left(\gamma|S\right)$$

- Iterate

Informal Methods

- Guess

$$0 < \Pr\left(\gamma|U\right) < \Pr\left(\gamma|M\right) < 1$$

- Approximate

$$\Pr\left(\gamma|U\right) \approx \Pr\left(\gamma|S\right)$$

- Iterate
  - Perform matching with current parameters

U S C E N S U S B U R E A U

Informal Methods

- Guess

$$0 < \Pr(\gamma|U) < \Pr(\gamma|M) < 1$$

- Approximate

$$\Pr(\gamma|U) \approx \Pr(\gamma|S)$$

- Iterate
  - Perform matching with current parameters
  - Review results

Informal Methods

- Guess

$$0 < \Pr(\gamma|U) < \Pr(\gamma|M) < 1$$

- Approximate

$$\Pr(\gamma|U) \approx \Pr(\gamma|S)$$

- Iterate
  - Perform matching with current parameters
  - Review results
  - Adjust parameters based on observation

U S C E N S U S B U R E A U

EM Algorithm

- Dempster, Laird, Rubin. “Maximum likelihood from incomplete data via the EM algorithm”. *Journal of the Royal Statistical Society.* SeriesB. **39**. pp. 1–39. 1977.

U S C E N S U S B U R E A U

EM Algorithm

- Dempster, Laird, Rubin. “Maximum likelihood from incomplete data via the EM algorithm”. *Journal of the Royal Statistical Society.* SeriesB. **39**. pp. 1–39. 1977.
- McLachlan, Krishnan. *The EM Algorithm and Extensions.* Wiley-Interscience. 2nd Ed. 2007.

USCENSUSBUREAU

EM Algorithm

- Dempster, Laird, Rubin. “Maximum likelihood from incomplete data via the EM algorithm”. *Journal of the Royal Statistical Society.* SeriesB. **39**. pp. 1–39. 1977.
- McLachlan, Krishnan. *The EM Algorithm and Extensions.* Wiley-Interscience. 2nd Ed. 2007.
- Maximum likelihood method

U S C E N S U S B U R E A U

EM Algorithm

- Dempster, Laird, Rubin. “Maximum likelihood from incomplete data via the EM algorithm”. *Journal of the Royal Statistical Society.* SeriesB. **39**. pp. 1–39. 1977.
- McLachlan, Krishnan. *The EM Algorithm and Extensions.* Wiley-Interscience. 2nd Ed. 2007.
- Maximum likelihood method
- Latent class

U S C E N S U S B U R E A U

EM Algorithm

- Dempster, Laird, Rubin. “Maximum likelihood from incomplete data via the EM algorithm”. *Journal of the Royal Statistical Society.* SeriesB. **39**. pp. 1–39. 1977.
- McLachlan, Krishnan. *The EM Algorithm and Extensions.* Wiley-Interscience. 2nd Ed. 2007.
- Maximum likelihood method
- Latent class
- Mixture model

U S C E N S U S B U R E A U

Likelihood Function

- $$\begin{aligned} L &= \prod_{(a,b)\in S} \Pr\left(\gamma\left(a,b\right)\right) \\ &= \prod_{j} \left(\Pr\left(\gamma^{j}|M\right)\Pr\left(M\right) + \Pr\left(\gamma^{j}|U\right)\Pr\left(U\right)\right)^{n_j} \end{aligned}$$

U S C E N S U S B U R E A U

Likelihood Function

- $$\begin{aligned} L &= \prod_{(a,b)\in S} \Pr\left(\gamma\left(a,b\right)\right) \\ &= \prod_{j} \left(\Pr\left(\gamma^j | M\right) \Pr\left(M\right) + \Pr\left(\gamma^j | U\right) \Pr\left(U\right)\right)^{n_j} \end{aligned}$$

- $$n_j = \left|\left\{\left(a,b\right) \in S \mid \gamma\left(a,b\right) = \gamma^j\right\}\right|$$

U S C E N S U S B U R E A U

Complete-data Likelihood Function

- Consider

$$\chi_j = \begin{cases} 1 & \text{if } (a,b)^j \in M \\ 0 & \text{if } (a,b)^j \in U \end{cases}$$

$$X_j = \sum_{\gamma(a,b)=\gamma^j} \chi_j\,(a,b)$$

US CENSUS BUREAU

Complete-data Likelihood Function

- Consider

$$\chi_j = \begin{cases} 1 & \text{if } (a,b)^j \in M \\ 0 & \text{if } (a,b)^j \in U \end{cases}$$

$$X_j = \sum_{\gamma(a,b)=\gamma^j} \chi_j\left(a,b\right)$$

- Then

$$L = \prod_j \left(\left(\Pr\left(\gamma^j | M\right) \Pr\left(M\right)\right)^{\overline{X}_j} \left(\Pr\left(\gamma^j | U\right) \Pr\left(U\right)\right)^{1-\overline{X}_j} \right)^{n_j}$$

U S C E N S U S B U R E A U

Expectation Step

- Given current estimates of conditional probabilities and $\Pr(M), \Pr(U)$, compute

USCENSUSBUREAU

Expectation Step

- Given current estimates of conditional probabilities and $\Pr(M), \Pr(U)$, compute

$$
\begin{aligned}
E\left(\overline{X}^j\right) &= \Pr\left(M|\gamma^j\right) \\
&= \frac{\Pr\left(\gamma^j|M\right)\Pr(M)}{\Pr\left(\gamma^j|M\right)\Pr(M) + \Pr\left(\gamma^j|U\right)\Pr(U)} \\
&= \hat{X}^j
\end{aligned}
$$

U S C E N S U S B U R E A U

Maximization Step

- Given unobserved data estimates $\hat{X}^j$, compute probabilities $\Pr\left(\gamma^j|M\right)$, $\Pr\left(\gamma^j|U\right)$, $\Pr\left(M\right)$, $\Pr\left(U\right)$ maximizing

U S C E N S U S B U R E A U

Maximization Step

- Given unobserved data estimates $\hat{X}^j$, compute probabilities $\Pr\left(\gamma^j|M\right)$, $\Pr\left(\gamma^j|U\right)$, $\Pr\left(M\right)$, $\Pr\left(U\right)$ maximizing

$$\log L =$$

$$\sum_j n_j \left(\hat{X}^j \left(\log \Pr\left(\gamma^j|M\right) + \log \Pr\left(M\right)\right) \right.$$

$$\left. + \left(1 - \hat{X}^j\right)\left(\log \Pr\left(\gamma^j|U\right) + \log \Pr\left(U\right)\right)\right)$$

U S C E N S U S B U R E A U

Max Step, Cont.

- Under conditional independence

Max Step, Cont.

- Under conditional independence

$$
\begin{aligned}
\log L = {} & \\
& \sum_j n_j \left(\sum_i \hat{X}^j \left(\log \Pr\left(\gamma_i^j | M\right) + \log \Pr\left(M\right) \right) \right. \\
& \left. + \left(1 - \hat{X}^j\right) \left(\sum_i \log \Pr\left(\gamma_i^j | U\right) + \log \Pr\left(U\right) \right) \right)
\end{aligned}
$$

Max Step, Cont.

For

$$n = \sum_j n_j$$

estimate

$$\Pr(M) = \frac{1}{n} \sum_j n_j \bar{X}^j$$

U S C E N S U S B U R E A U

Max Step, Cont.

- Let

$$k_i^j = \begin{cases} 1 & \text{if } \gamma_i^j = 1 \\ 0 & \text{if } \gamma_i^j = 0 \end{cases}$$

Max Step, Cont.

- Let

$$k_i^j = \begin{cases} 1 & \text{if } \gamma_i^j = 1 \\ 0 & \text{if } \gamma_i^j = 0 \end{cases}$$

- and estimate

$$\Pr\left(\gamma_i | M\right) = \frac{1}{n} \sum_j n_j \bar{X}^j k_i^j$$

U S C E N S U S B U R E A U

EM Algorithm

1. Initialize with probability values

U S C E N S U S B U R E A U

EM Algorithm

1. Initialize with probability values
2. Iterate

U S C E N S U S B U R E A U

EM Algorithm

1. Initialize with probability values
2. Iterate
   (a) Expectation Step

U S C E N S U S B U R E A U

EM Algorithm

1. Initialize with probability values
2. Iterate
   (a) Expectation Step
   (b) Maximization Step

EM Algorithm

1. Initialize with probability values
2. Iterate
   (a) Expectation Step
   (b) Maximization Step
3. Until convergence of likelihood function

U S C E N S U S B U R E A U

EM Algorithm Remarks

- Each EM iteration increases likelihood, so algorithm converges to a (local) maximum

U S C E N S U S B U R E A U

EM Algorithm Remarks

- Each EM iteration increases likelihood, so algorithm converges to a (local) maximum
- For this conditional independence model, convergence is efficient and generally insensitive to initial data

U S C E N S U S B U R E A U

EM Algorithm Remarks

- Each EM iteration increases likelihood, so algorithm converges to a (local) maximum
- For this conditional independence model, convergence is efficient and generally insensitive to initial data
- For latent class to be numerically detected, it must be represented by about 5% of the total record pair data

U S C E N S U S B U R E A U

EM Algorithm Remarks

- Each EM iteration increases likelihood, so algorithm converges to a (local) maximum
- For this conditional independence model, convergence is efficient and generally insensitive to initial data
- For latent class to be numerically detected, it must be represented by about 5% of the total record pair data
- Check: Do $\Pr(M), \Pr(U)$ seem reasonable?

U S C E N S U S B U R E A U

EM Remarks, Cont.

- If $\Pr(M), \Pr(U)$ are off, everything is off

EM Remarks, Cont.

- If $\Pr(M), \Pr(U)$ are off, everything is off
- We can extend problem to comparisons taking on more that 2 values

U S C E N S U S B U R E A U

EM Remarks, Cont.

- If $\Pr(M), \Pr(U)$ are off, everything is off
- We can extend problem to comparisons taking on more that 2 values
  - Creates more pattern types and probability parameters

U S C E N S U S B U R E A U

EM Remarks, Cont.

- If $\Pr(M), \Pr(U)$ are off, everything is off
- We can extend problem to comparisons taking on more that 2 values
  - Creates more pattern types and probability parameters
- Can extend algorithm to more that 2 classes

U S C E N S U S B U R E A U

EM Remarks, Cont.

- If $\Pr(M), \Pr(U)$ are off, everything is off
- We can extend problem to comparisons taking on more that 2 values
  - Creates more pattern types and probability parameters
- Can extend algorithm to more that 2 classes
  - Increases number of parameters to be estimated

U S C E N S U S B U R E A U

Blocking

- If set A contains m records and set B contains n records then $A \times B$ contains mn record pairs

U S C E N S U S B U R E A U

Blocking

- If set A contains m records and set B contains n records then $A \times B$ contains mn record pairs
- It is computationally inefficient to compare all record pairs

US CENSUS BUREAU

Blocking

- If set A contains m records and set B contains n records then $A \times B$ contains mn record pairs
- It is computationally inefficient to compare all record pairs
- In practice, just bring together record pairs that agree on some chosen features (blocking criterion)

U S C E N S U S B U R E A U

Blocking

- If set A contains m records and set B contains n records then $A \times B$ contains mn record pairs
- It is computationally inefficient to compare all record pairs
- In practice, just bring together record pairs that agree on some chosen features (blocking criterion)
- Generally repeat record linkage procedure for several different blocking criteria

U S C E N S U S B U R E A U

Blocking Criteria

- Geographic codes

U S C E N S U S B U R E A U

Blocking Criteria

- Geographic codes
- Postal or phone codes

U S C E N S U S B U R E A U

Blocking Criteria

- Geographic codes
- Postal or phone codes
- Name prefix

Blocking Criteria

- Geographic codes
- Postal or phone codes
- Name prefix
- Phonetic name codes

U S C E N S U S B U R E A U

Blocking Criteria

- Geographic codes
- Postal or phone codes
- Name prefix
- Phonetic name codes
  - Soundex

U S C E N S U S B U R E A U

Blocking Criteria

- Geographic codes
- Postal or phone codes
- Name prefix
- Phonetic name codes
  - Soundex
  - NYSIIS

U S C E N S U S B U R E A U

Blocking Criteria

- Geographic codes
- Postal or phone codes
- Name prefix
- Phonetic name codes
  - Soundex
  - NYSIIS
- Combinations

U S C E N S U S B U R E A U

Record Linkage Refinements

- String comparator

Record Linkage Refinements

- String comparator
- Third latent class

U S C E N S U S B U R E A U

Record Linkage Refinements

- String comparator
- Third latent class
- Third comparison value

U S C E N S U S B U R E A U

Record Linkage Refinements

- String comparator
- Third latent class
- Third comparison value
- One-to-one matching

U S C E N S U S B U R E A U

String Comparator

- For some comparisons (*e.g.* categorical variables), it is sufficient to assign agree/disagree

U S C E N S U S B U R E A U

String Comparator

- For some comparisons (*e.g.* categorical variables), it is sufficient to assign agree/disagree
- For string variables (*e.g.* first names, last names, street names) this is probably too restrictive

U S C E N S U S B U R E A U

String Comparator

- For some comparisons (*e.g.* categorical variables), it is sufficient to assign agree/disagree
- For string variables (*e.g.* first names, last names, street names) this is probably too restrictive
- A string comparator allows us to assign comparison values between full agreement and full disagreement

U S C E N S U S B U R E A U

String Comparator Context

- Binary comparison $\gamma \in \{0, 1\}$

U S C E N S U S B U R E A U

String Comparator Context

- Binary comparison $\gamma \in \{0, 1\}$
- Weight assignment

$$a_w = \log \frac{\Pr(\gamma = 1|M)}{\Pr(\gamma = 1|U)}$$

$$d_w = \log \frac{\Pr(\gamma = 0|M)}{\Pr(\gamma = 0|U)}$$

$$d_w < 0 < a_w$$

U S C E N S U S B U R E A U

String Comparator Context, Cont.

- For alphabet Σ, our string comparator is a *similarity function*

$$\gamma : \Sigma^* \times \Sigma^* \to [0, 1]$$

$$\gamma\left(\alpha, \beta\right) = 1 \text{ if } \alpha = \beta$$

US CENSUS BUREAU

String Comparator Context, Cont.

- For alphabet Σ, our string comparator is a *similarity function*

$$\gamma : \Sigma^* \times \Sigma^* \to [0, 1]$$
$$\gamma(\alpha, \beta) = 1 \text{ if } \alpha = \beta$$

- Weight assignment function w is an increasing interpolation function

$$w : [0, 1] \to [d_w, a_w]$$
$$w(1) = a_w$$

U S C E N S U S B U R E A U

Some String Comparator Types

- Bigram, *n*-gram

U S C E N S U S B U R E A U

Some String Comparator Types

- Bigram, *n*-gram
- Jaro-Winkler

Some String Comparator Types

- Bigram, *n*-gram
- Jaro-Winkler
- Edit distance

Bigrams

- Decompose string into a set of 2-character (contiguous) substrings

$$alphabet \rightarrow \{al, lp, ph, ha, ab, be, et\}$$

U S C E N S U S B U R E A U

Bigrams

- Decompose string into a set of 2-character (contiguous) substrings

$$alphabet \rightarrow \{al, lp, ph, ha, ab, be, et\}$$

- For alphabet of $s = |\Sigma|$ characters, record bigram counts in a vector of dimension s^2

U S C E N S U S B U R E A U

Bigrams, Cont.

- Two strings can be compared by computing the "angle" between their bigram vectors a, b

$$\cos\theta = \frac{a \cdot b}{|a|\,|b|}$$

Bigrams, Cont.

- Two strings can be compared by computing the “angle” between their bigram vectors a, b

$$\cos\theta = \frac{a \cdot b}{|a|\,|b|}$$

- Obvious generalization to *n*-grams

U S C E N S U S B U R E A U

Bigrams, Cont.

- Two strings can be compared by computing the "angle" between their bigram vectors a, b

$$\cos\theta = \frac{a \cdot b}{|a|\,|b|}$$

- Obvious generalization to *n*-grams
- Vector for *n*-gram is in s^n dimensional space

U S C E N S U S B U R E A U

Bigrams, Cont.

- Computation algorithm is fast (linear)

U S C E N S U S B U R E A U

Bigrams, Cont.

- Computation algorithm is fast (linear)
- Don’t work very well for record linkage

U S C E N S U S B U R E A U

Bigrams, Cont.

- Computation algorithm is fast (linear)
- Don’t work very well for record linkage
  - Ignores order of bigram occurrence

$$abcba \approx bcbab$$

U S C E N S U S B U R E A U

Bigrams, Cont.

- Computation algorithm is fast (linear)
- Don't work very well for record linkage
  - Ignores order of bigram occurrence

$$abcba \approx bcbab$$

  - Works better for small alphabet, long strings than *vice versa*

U S C E N S U S B U R E A U

Jaro-Winkler Comparator

- In the following, let $\alpha = (a_1, a_2, \ldots a_m)$, $\beta = (b_1, b_2, \ldots, b_n)$ be strings of lengths m, n respectively with $m \leq n$

U S C E N S U S B U R E A U

Jaro-Winkler Comparator

- In the following, let $\alpha = (a_1, a_2, \ldots a_m)$, $\beta = (b_1, b_2, \ldots, b_n)$ be strings of lengths m, n respectively with $m \le n$
- Comparator value depends on number of common characters and character "transpositions"

U S C E N S U S B U R E A U

Jaro-Winkler Comparator, Cont.

- Strings α, β have common characters a_i, b_j iff

$$\begin{aligned} a_i &= b_j \\ |i - j| &< \left\lfloor \frac{n}{2} \right\rfloor \end{aligned}$$

US CENSUS BUREAU

Jaro-Winkler Comparator, Cont.

- Strings α, β have common characters a_i, b_j iff

$$\begin{aligned} a_i &= b_j \\ |i - j| &< \left\lfloor \frac{n}{2} \right\rfloor \end{aligned}$$

- The number of transpositions is computed as the greatest integer of half of the number of out-of-order common character pairs

Jaro-Winkler Comparator, Cont.

- For string pair with c common characters and t transpositions, basis similarity score is

$$x = \frac{1}{3}\left(\frac{c}{m} + \frac{c}{n} + \frac{c-t}{c}\right)$$

Jaro-Winkler Example

- Consider the strings (b,a,r,n,e,s) and (a,n,d,e,r,s,o,n)

U S C E N S U S B U R E A U

Jaro-Winkler Example

- Consider the strings (*b,a,r,n,e,s*) and (*a,n,d,e,r,s,o,n*)
- Search range d

$$
\begin{aligned}
n &= 8 \\
d &= \left\lfloor \frac{8}{2} \right\rfloor - 1 = 3
\end{aligned}
$$

U S C E N S U S B U R E A U

Jaro-Winkler Example

- Consider the strings (*b,a,r,n,e,s*) and (*a,n,d,e,r,s,o,n*)
- Search range d

$$\begin{aligned} n &= 8 \\ d &= \left\lfloor \frac{8}{2} \right\rfloor - 1 = 3 \end{aligned}$$

- Common characters

$$(a, r, n, e, s)$$
$$(a, n, e, r, s)$$

U S C E N S U S B U R E A U

Jaro-Winkler Example, Cont.

- Five common characters with 3 out of order, so $c = 5, t = 1$

U S C E N S U S B U R E A U

Jaro-Winkler Example, Cont.

- Five common characters with 3 out of order, so $c = 5, t = 1$
- Score

$$x = \frac{1}{3}\left(\frac{5}{6} + \frac{5}{8} + \frac{4}{5}\right) = \frac{271}{360} \doteq 0.75280$$

U S C E N S U S B U R E A U

Jaro-Winkler Variations

- Similar characters

Jaro-Winkler Variations

- Similar characters
- Prefix adjustment

Jaro-Winkler Variations

- Similar characters
- Prefix adjustment
- Long suffix adjustment

U S C E N S U S B U R E A U

Similar Characters

- Attempt to compensate for common misspellings or typos

Similar Characters

- Attempt to compensate for common misspellings or typos
- List of 36 pairs of characters deemed similar (*e.g.* most vowell pairs)

U S C E N S U S B U R E A U

Similar Characters

- Attempt to compensate for common misspellings or typos
- List of 36 pairs of characters deemed similar (*e.g.* most vowell pairs)
- After common characters designated, remaining characters checked for similar pairs

U S C E N S U S B U R E A U

Similar Characters

- Attempt to compensate for common misspellings or typos
- List of 36 pairs of characters deemed similar (*e.g.* most vowell pairs)
- After common characters designated, remaining characters checked for similar pairs
- Each similar pair is scored as 0.3 of a common pair

U S C E N S U S B U R E A U

Similar Characters, Cont.

- Revised character count

$$c_s = c + 0.3s$$

U S C E N S U S B U R E A U

Similar Characters, Cont.

- Revised character count

$$c_s = c + 0.3s$$

- Adjusted comparator score

$$x_s = \frac{1}{3}\left(\frac{c_s}{m} + \frac{c_s}{n} + \frac{c-t}{c}\right)$$

Similar Characters, Cont.

- For example. *abc* and *ebc* have 2 common characters and the remaining pair (*a*,*e*) are similar, so

$$\begin{aligned} x_s &= \frac{1}{3}\left(\frac{2}{3}+\frac{2}{3}+1\right)+\frac{1}{3}\left(\frac{0.3}{3}+\frac{0.3}{3}\right) \\ &= \frac{7}{9}+\frac{1}{15} \\ &= \frac{38}{45} \end{aligned}$$

U S C E N S U S B U R E A U

Common Prefix

- Spelling mistakes tend to occur later in the string (Winkler)

U S C E N S U S B U R E A U

Common Prefix

- Spelling mistakes tend to occur later in the string (Winkler)
- Check for common prefix of up to 4 characters

U S C E N S U S B U R E A U

Common Prefix

- Spelling mistakes tend to occur later in the string (Winkler)
- Check for common prefix of up to 4 characters
- If length of common prefix is p, adjust score x by

$$x_p = x + \frac{p\,(1 - x)}{10}$$

U S C E N S U S B U R E A U

Long String Adjustment

- Adjust score for longer strings with several common characters beyond common prefix

U S C E N S U S B U R E A U

Long String Adjustment

- Adjust score for longer strings with several common characters beyond common prefix
- Conditions for using the adjustment

U S C E N S U S B U R E A U

Long String Adjustment

- Adjust score for longer strings with several common characters beyond common prefix
- Conditions for using the adjustment

1. $m \geq 5$

U S C E N S U S B U R E A U

Long String Adjustment

- Adjust score for longer strings with several common characters beyond common prefix
- Conditions for using the adjustment
  1. $m \geq 5$
  2. $c - p \geq 2$

U S C E N S U S B U R E A U

Long String Adjustment

- Adjust score for longer strings with several common characters beyond common prefix
- Conditions for using the adjustment
  1. $m \geq 5$
  2. $c - p \geq 2$
  3. $c - p \geq \frac{m-p}{2}$

U S C E N S U S B U R E A U

Long String Adjustment, Cont.

- That is,

U S C E N S U S B U R E A U

Long String Adjustment, Cont.

- That is,
  1. Both strings are at least 5 characters long

U S C E N S U S B U R E A U

Long String Adjustment, Cont.

- That is,
  1. Both strings are at least 5 characters long
  2. There are at least two common characters besides the agreeing prefix characters

U S C E N S U S B U R E A U

Long String Adjustment, Cont.

- That is,
  1. Both strings are at least 5 characters long
  2. There are at least two common characters besides the agreeing prefix characters
  3. We want the strings outside the common prefixes to be fairly rich in common characters, so that the remaining common characters are at least half of the remaining common characters of the shorter string

Long String Adjustment, Cont.

- If conditions met, then adjust score by

$$x_l = x + (1 - x)\,\frac{c - (p + 1)}{m + n - 2\,(p - 1)}$$

Long String Adjustment, Cont.

- In *barnes, anderson* example, conditions are met, so the adjusted score is

$$
\begin{aligned}
x_l &= \frac{271}{360} + \left(1 - \frac{271}{360}\right) \frac{5-1}{6+8+2} \\
&= \frac{391}{480} \\
&\doteq 0.8146
\end{aligned}
$$

U S C E N S U S B U R E A U

Jaro-Winkler Comparator

- Slower algorithm (quadratic)

Jaro-Winkler Comparator

- Slower algorithm (quadratic)
- Performs very well in tests

Edit Distance String Comparators

- The minimum number of edits required to convert sting α to string β, lengths $m \leq n$

U S C E N S U S B U R E A U

Edit Distance String Comparators

- The minimum number of edits required to convert sting α to string β, lengths $m \leq n$
  - Insert

U S C E N S U S B U R E A U

Edit Distance String Comparators

- The minimum number of edits required to convert sting α to string β, lengths $m \leq n$
  - Insert
  - Delete

U S C E N S U S B U R E A U

Edit Distance String Comparators

- The minimum number of edits required to convert sting α to string β, lengths $m \leq n$
  - Insert
  - Delete
  - Substitute

U S C E N S U S B U R E A U

Edit Distance String Comparators

- The minimum number of edits required to convert sting α to string β, lengths $m \leq n$
  - Insert
  - Delete
  - Substitute
- Dynamic programming algorithm, quadratic complexity $O(mn)$

Edit Distance Algorithm

- For α_i prefix of α of length i, β_j prefix of β of length j

U S C E N S U S B U R E A U

Edit Distance Algorithm

- For α_i prefix of α of length i, β_j prefix of β of length j
- Initialize

$$\begin{aligned} e(\alpha_i, \varepsilon) &= i \\ e(\varepsilon, \beta_j) &= j \\ e(\varepsilon, \varepsilon) &= 0 \end{aligned}$$

Edit Distance Algorithm, Cont.

- Compute

$$e\left(\alpha_i,\beta_j\right)=\min\begin{cases}e\left(\alpha_{i-1},\beta_j\right)+1\\ e\left(\alpha_i,\beta_{j-1}\right)+1\\ \begin{cases}e\left(\alpha_{i-1},\beta_{j-1}\right) & \text{if } a_i=b_j\\ e\left(\alpha_{i-1},\beta_{j-1}\right)+1 & \text{if } a_i\neq b_j\end{cases}\end{cases}$$

U S C E N S U S B U R E A U

Edit Distance Algorithm, Cont.

- Compute

$$e\left(\alpha_i,\beta_j\right) = \min \begin{cases} e\left(\alpha_{i-1},\beta_j\right)+1 \\ e\left(\alpha_i,\beta_{j-1}\right)+1 \\ \begin{cases} e\left(\alpha_{i-1},\beta_{j-1}\right) & \text{if } a_i = b_j \\ e\left(\alpha_{i-1},\beta_{j-1}\right)+1 & \text{if } a_i \neq b_j \end{cases} \end{cases}$$

- Distance

$$e = e\left(\alpha,\beta\right) = e\left(\alpha_m,\beta_n\right)$$

Edit Distance Similarity Function

- Edit distance is a metric

U S C E N S U S B U R E A U

Edit Distance Similarity Function

- Edit distance is a metric
- Similarity function

$$x_e = 1 - \frac{e}{n}$$

U S C E N S U S B U R E A U

Edit Distance Example

- For example, for *barnes, anderson*, have possible minimal edit path

$$(b, \varepsilon)\, a\, (r, n)\, (n, d)\, e\, (\varepsilon, r)\, s\, (\varepsilon, o)\, (\varepsilon, n)$$

Edit Distance Example

- For example, for *barnes, anderson*, have possible minimal edit path

$$(b, \varepsilon)\, a\, (r, n)\, (n, d)\, e\, (\varepsilon, r)\, s\, (\varepsilon, o)\, (\varepsilon, n)$$

- So

$$x_e = 1 - \frac{6}{8} = \frac{1}{4}$$

U S C E N S U S B U R E A U

Edit Distance Example

- For example, for *barnes, anderson*, have possible minimal edit path

$$(b,\varepsilon)\,a\,(r,n)\,(n,d)\,e\,(\varepsilon,r)\,s\,(\varepsilon,o)\,(\varepsilon,n)$$

- So

$$x_e = 1 - \frac{6}{8} = \frac{1}{4}$$

- Note order of characters very important

U S C E N S U S B U R E A U

Longest Common Subsequence

- Length of longest common subsequence (*lcs*)

US CENSUS BUREAU

Longest Common Subsequence

- Length of longest common subsequence (*lcs*)
- Similar dynamic programming algorithm, without substitutions

$$e\left(\alpha_i, \beta_j\right) = \min \begin{cases} e\left(\alpha_{i-1}, \beta_j\right) + 1 \\ e\left(\alpha_i, \beta_{j-1}\right) + 1 \\ e\left(\alpha_{i-1}, \beta_{j-1}\right) & \text{if } a_i = b_j \end{cases}$$

U S C E N S U S B U R E A U

LCS Similarity Function

- Similarity function

$$x_c = \frac{l}{m}$$

US CENSUS BUREAU

LCS Similarity Function

- Similarity function

$$x_c = \frac{l}{m}$$

- Example *lcs*= (a, n, e, s), similarity score

$$x_c = \frac{4}{6} = \frac{2}{3}$$

U S C E N S U S B U R E A U

Combination Similarity Function

- Compute both edit distance and *lcs*

U S C E N S U S B U R E A U

Combination Similarity Function

- Compute both edit distance and *lcs*
- Combined score

$$x_{ec} = \frac{1}{2}\left(\left(1 - \frac{e}{n}\right) + \frac{l}{m}\right)$$

U S C E N S U S B U R E A U

Combination Similarity Function

- Compute both edit distance and *lcs*
- Combined score

$$x_{ec} = \frac{1}{2}\left(\left(1 - \frac{e}{n}\right) + \frac{l}{m}\right)$$

- Example

$$x_{ec} = \frac{1}{2}\left(\frac{1}{4} + \frac{2}{3}\right) = \frac{11}{24} \doteq 0.4583$$

U S C E N S U S B U R E A U

Evaluating String Comparators

- Yancey, “Evaluating String Comparator Performance for Record Linkage,” 2005, http://www.census.gov/srd/www/byname.html

U S C E N S U S B U R E A U

Evaluating String Comparators

- Yancey, “Evaluating String Comparator Performance for Record Linkage,” 2005, http://www.census.gov/srd/www/byname.html
- Compare performance

U S C E N S U S B U R E A U

Evaluating String Comparators

- Yancey, “Evaluating String Comparator Performance for Record Linkage,” 2005, http://www.census.gov/srd/www/byname.html
- Compare performance
  - Jaro-Winkler

U S C E N S U S B U R E A U

Evaluating String Comparators

- Yancey, “Evaluating String Comparator Performance for Record Linkage,” 2005, http://www.census.gov/srd/www/byname.html
- Compare performance
  - Jaro-Winkler
  - Edit distance

U S C E N S U S B U R E A U

Evaluating String Comparators, Cont.

- Jaro-Winkler, with and without modifications

US CENSUS BUREAU

Evaluating String Comparators, Cont.

- Jaro-Winkler, with and without modifications
  - Prefix adjustment

US CENSUS BUREAU

Evaluating String Comparators, Cont.

- Jaro-Winkler, with and without modifications
  - Prefix adjustment
  - Similar characters

U S C E N S U S B U R E A U

Evaluating String Comparators, Cont.

- Jaro-Winkler, with and without modifications
  - Prefix adjustment
  - Similar characters
  - Long suffix adjustment

U S C E N S U S B U R E A U

Evaluating String Comparators, Cont.

- Edit distance

U S C E N S U S B U R E A U

Evaluating String Comparators, Cont.

- Edit distance
  - Edit distance similarity

USCENSUSBUREAU

Evaluating String Comparators, Cont.

- Edit distance
  - Edit distance similarity
  - Markov edit distance (J. Wei. “Markov Edit Distance”. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol 26, No. 3, pp. 311–321, 2004)

U S C E N S U S B U R E A U

Evaluating String Comparators, Cont.

- Edit distance
  - Edit distance similarity
  - Markov edit distance (J. Wei. "Markov Edit Distance". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol 26, No. 3, pp. 311–321, 2004)
  - With and without *lcs*

U S C E N S U S B U R E A U

Evaluating String Comparators, Cont.

- Lots of data

US CENSUS BUREAU

Evaluating String Comparators, Cont.

- Lots of data
- Truth decks from 1990 and 2000 U.S. Census

U S C E N S U S B U R E A U

Evaluating String Comparators, Cont.

- Lots of data
- Truth decks from 1990 and 2000 U.S. Census
- M: All non-identical, non-blank names from matched record pairs

U S C E N S U S B U R E A U

Evaluating String Comparators, Cont.

- Lots of data
- Truth decks from 1990 and 2000 U.S. Census
- M: All non-identical, non-blank names from matched record pairs
- U: All cross pairs of these names

U S C E N S U S B U R E A U

Results of String Comparator Evaluati

- Jaro-Winkler did well

U S C E N S U S B U R E A U

Results of String Comparator Evaluati

- Jaro-Winkler did well
  - Prefix adjustment always helps

U S C E N S U S B U R E A U

Results of String Comparator Evaluati

- Jaro-Winkler did well
  - Prefix adjustment always helps
  - Similar character adjustment generally helps a bit

U S C E N S U S B U R E A U

Results of String Comparator Evaluati

- Jaro-Winkler did well
  - Prefix adjustment always helps
  - Similar character adjustment generally helps a bit
  - Long suffix adjustment sometime helps a little

U S C E N S U S B U R E A U

Results of String Comparator Evaluati

- Adding *lcs* significantly improves edit distance and Markov edit distance

U S C E N S U S B U R E A U

Results of String Comparator Evaluati

- Adding *lcs* significantly improves edit distance and Markov edit distance
- Edit distance always better than Markov edit distance

U S C E N S U S B U R E A U

Results of String Comparator Evaluati

- Adding *lcs* significantly improves edit distance and Markov edit distance
- Edit distance always better than Markov edit distance
- Jaro-Winkler (full) comparable to edit distance/*lcs*

U S C E N S U S B U R E A U

Results of String Comparator Evaluati

- Adding *lcs* significantly improves edit distance and Markov edit distance
- Edit distance always better than Markov edit distance
- Jaro-Winkler (full) comparable to edit distance/*lcs*
  - Usually

U S C E N S U S B U R E A U

Jaro-Winkler Anomaly

- Let α, β be strings of length n with no common characters

U S C E N S U S B U R E A U

Jaro-Winkler Anomaly

- Let α, β be strings of length n with no common characters
- For Jaro-Winkler

U S C E N S U S B U R E A U

Jaro-Winkler Anomaly

- Let α, β be strings of length n with no common characters
- For Jaro-Winkler
  - $s(\alpha, \alpha\beta) = \frac{5}{6}$

U S C E N S U S B U R E A U

Jaro-Winkler Anomaly

- Let α, β be strings of length n with no common characters
- For Jaro-Winkler
  - $s\left(\alpha, \alpha\beta\right) = \frac{5}{6}$
  - In $n \geq 4$, with prefix adjustment, $s\left(\alpha, \alpha\beta\right) = \frac{9}{10}$

U S C E N S U S B U R E A U

Jaro-Winkler Anomaly

- Let α, β be strings of length n with no common characters
- For Jaro-Winkler
  - $s(\alpha, \alpha\beta) = \frac{5}{6}$
  - In $n \geq 4$, with prefix adjustment, $s(\alpha, \alpha\beta) = \frac{9}{10}$
  - $s(\beta, \alpha\beta) = 0$

U S C E N S U S B U R E A U

Jaro-Winkler Anomaly

- Let α, β be strings of length n with no common characters
- For Jaro-Winkler
  - $s(\alpha, \alpha\beta) = \frac{5}{6}$
  - In $n \geq 4$, with prefix adjustment, $s(\alpha, \alpha\beta) = \frac{9}{10}$
  - $s(\beta, \alpha\beta) = 0$
- For edit-distance/*lcs*, $s(\alpha, \alpha\beta) = s(\beta, \alpha\beta) = \frac{3}{4}$

U S C E N S U S B U R E A U

Hybrid Comparator

- Compute both Jaro-Winkler and edit distance/*lcs*

U S C E N S U S B U R E A U

Hybrid Comparator

- Compute both Jaro-Winkler and edit distance/*lcs*
- Use larger of Jaro-Winkler and (scaled) edit distance/*lcs*

U S C E N S U S B U R E A U

Hybrid Comparator

- Compute both Jaro-Winkler and edit distance/*lcs*
- Use larger of Jaro-Winkler and (scaled) edit distance/*lcs*
- Where J-W does well, hybrid does a little better than either

US CENSUS BUREAU

Hybrid Comparator

- Compute both Jaro-Winkler and edit distance/*lcs*
- Use larger of Jaro-Winkler and (scaled) edit distance/*lcs*
- Where J-W does well, hybrid does a little better than either
- Where J-W does significantly worse, hybrid does nearly as well as edit distance/*lcs*

U S C E N S U S B U R E A U

Hybrid Comparator, Cont.

- Can see some improvement in actual record linkage results

U S C E N S U S B U R E A U

Hybrid Comparator, Cont.

- Can see some improvement in actual record linkage results
- Calculation takes a long time

U S C E N S U S B U R E A U

String Comparator Summary

- String comparator improves record linkage

U S C E N S U S B U R E A U

String Comparator Summary

- String comparator improves record linkage
- String comparator takes significant amount of record linkage computation time

U S C E N S U S B U R E A U

String Comparator Summary

- String comparator improves record linkage
- String comparator takes significant amount of record linkage computation time
  - For J-W, about 30%

U S C E N S U S B U R E A U

More Than Two Latent Classes

- EM algorithm generalizes to more than 2 classes, M, U

U S C E N S U S B U R E A U

More Than Two Latent Classes

- EM algorithm generalizes to more than 2 classes, M, U
- Does U have any natural partitions?

More Than Two Latent Classes

- EM algorithm generalizes to more than 2 classes, M, U
- Does U have any natural partitions?
- For Census data

U S C E N S U S B U R E A U

More Than Two Latent Classes

- EM algorithm generalizes to more than 2 classes, M, U
- Does U have any natural partitions?
- For Census data
  - U_1, different people, same household

U S C E N S U S B U R E A U

More Than Two Latent Classes

- EM algorithm generalizes to more than 2 classes, M, U
- Does U have any natural partitions?
- For Census data
  - U_1, different people, same household
  - U_2, different people, different household

U S C E N S U S B U R E A U

More Than Two Latent Classes

- EM algorithm generalizes to more than 2 classes, M, U
- Does U have any natural partitions?
- For Census data
  - U_1, different people, same household
  - U_2, different people, different household
- Classes have to be implicit in the matching data

U S C E N S U S B U R E A U

EM for Three Classes

- Use EM to estimate $\Pr(U_1), \Pr(U_2)$, and marginal probabilities $\Pr(\gamma_i | U_1), \Pr(\gamma_i | U_2)$

U S C E N S U S B U R E A U

EM for Three Classes

- Use EM to estimate $\Pr(U_1), \Pr(U_2)$, and marginal probabilities $\Pr(\gamma_i|U_1), \Pr(\gamma_i|U_2)$
- Recombine

$$\Pr(\gamma_i|U) = \frac{\Pr(\gamma_i|U_1)\Pr(U_1) + \Pr(\gamma_i|U_2)\Pr(U_2)}{\Pr(U_1) + \Pr(U_2)}$$

U S C E N S U S B U R E A U

More Than Two Comparison Values

- Can have more than {agree, disagree}

US CENSUS BUREAU

More Than Two Comparison Values

- Can have more than {agree, disagree}
- For m comparison values, EM algorithm must estimate $2\,(m-1)$ parameters

U S C E N S U S B U R E A U

More Than Two Comparison Values

- Can have more than {agree, disagree}
- For m comparison values, EM algorithm must estimate $2\,(m-1)$ parameters
- We have used {agree, disagree, missing} when data is often missing but has distinguishing power when present

More Than Two Comparison Values

- Can have more than {agree, disagree}
- For m comparison values, EM algorithm must estimate $2\,(m-1)$ parameters
- We have used {agree, disagree, missing} when data is often missing but has distinguishing power when present
  - For example, middle initial

U S C E N S U S B U R E A U

More Than Two Compr. Values, Cont.

- Reasonability check for parameter estimation

$$\log \frac{\Pr\left(\text{blank}|M\right)}{\Pr\left(\text{blank}|U\right)} \approx 0$$

One-to-one Matching

- If both files have no duplication within them, then it is preferable to have output with each record linked to no more than one record in the other file

U S C E N S U S B U R E A U

One-to-one Matching

- If both files have no duplication within them, then it is preferable to have output with each record linked to no more than one record in the other file
- All records that are compared with each other are within a block

U S C E N S U S B U R E A U

One-to-one Matching

- If both files have no duplication within them, then it is preferable to have output with each record linked to no more than one record in the other file
- All records that are compared with each other are within a block
- Linear assignment algorithm used to find optimal one-to-one matches within a block

U S C E N S U S B U R E A U

Linear Assignment Algorithm

- For agreement weights in block

$$
\begin{array}{c|ccccc}
 & B_1 & B_2 & B_3 & \cdots & B_n \\
\hline
A_1 & w_{11} & w_{12} & w_{13} & & w_{1n} \\
A_2 & w_{21} & w_{22} & w_{23} & & w_{2n} \\
A_3 & w_{31} & w_{32} & w_{33} & & w_{3n} \\
\vdots & & & & & \\
A_n & w_{n1} & w_{n2} & w_{n3} & & w_{nn}
\end{array}
$$

USCENSUSBUREAU

Linear Assignment Algorithm

- Find permutation $\bar{\sigma}$ that maximizes

$$\sum_{i=1}^{n} w_{i,\sigma(i)}$$

Linear Assignment Algorithm

- Find permutation $\bar{\sigma}$ that maximizes

$$\sum_{i=1}^{n} w_{i,\sigma(i)}$$

- Not a greedy algorithm

U S C E N S U S B U R E A U

Linear Assignment Algorithm

- Find permutation $\bar{\sigma}$ that maximizes

$$\sum_{i=1}^{n} w_{i,\sigma(i)}$$

- Not a greedy algorithm

Father	40	$\longleftrightarrow$	Mother	39
Mother	39	$\longleftrightarrow$	Daughter	16
Daughter	16	$\longleftrightarrow$	Son	13
Son	13			

U S C E N S U S B U R E A U

Error Rates

- False Match Rate

$$\mu = \Pr\left(L \,|\, U\right) = \Pr\left(w\left(\gamma\right) < T_{\mu} \,|\, U\right)$$

U S C E N S U S B U R E A U

Error Rates

- False Match Rate

$$\mu = \Pr\left(L \,|\, U\right) = \Pr\left(w\left(\gamma\right) < T_{\mu} \,|\, U\right)$$

- False Non-match Rate

$$\lambda = \Pr\left(N \,|\, M\right) = \Pr\left(w\left(\gamma\right) > T_{\lambda} \,|\, U\right)$$

Practical Considerations

- Question: Relative to what sample space?

U S C E N S U S B U R E A U

Practical Considerations

- Question: Relative to what sample space?
  - $A \times B$

Practical Considerations

- Question: Relative to what sample space?
  - $A \times B$
  - Pairs in blocking scheme

US CENSUS BUREAU

Practical Considerations

- Question: Relative to what sample space?
  - $A \times B$
  - Pairs in blocking scheme
  - After 1-1 matching

U S C E N S U S B U R E A U

Practical Considerations

- Question: Relative to what sample space?
  - $A \times B$
  - Pairs in blocking scheme
  - After 1-1 matching
- Each step presumably filters out a lot of low-weight pairs

U S C E N S U S B U R E A U

False Non-Match Rate

- Difficult to determine as well as define

U S C E N S U S B U R E A U

False Non-Match Rate

- Difficult to determine as well as define
- May as well try to estimate number of undiscovered matches in $A \times B$

U S C E N S U S B U R E A U

False Non-Match Rate

- Difficult to determine as well as define
- May as well try to estimate number of undiscovered matches in $A \times B$
- Can try capture-recapture using *independent* blocking schemes

U S C E N S U S B U R E A U

False Match Rate

- Bellin-Rubin

U S C E N S U S B U R E A U

False Match Rate

- Bellin-Rubin
- Larsen

False Match Rate

- Bellin-Rubin
- Larsen
- Larsen, Rubin, Winkler

U S C E N S U S B U R E A U

Bellin-Rubin

- Bellin, T.R. and Rubin, D.B. (1995) “A Method for Calibrating False-Match Rates in Record Linkage,” *Journal of the American Statistical Association*, 90,pp.694–707.

U S C E N S U S B U R E A U

Bellin-Rubin

- Bellin, T.R. and Rubin, D.B. (1995) “A Method for Calibrating False-Match Rates in Record Linkage,” *Journal of the American Statistical Association*, 90,pp.694–707.
- Consider sample space without 1-1 matching

U S C E N S U S B U R E A U

Bellin-Rubin

- Bellin, T.R. and Rubin, D.B. (1995) “A Method for Calibrating False-Match Rates in Record Linkage,” *Journal of the American Statistical Association*, 90,pp.694–707.
- Consider sample space without 1-1 matching
- Model as a mixture of 2 normal distributions (Box-Cox)

U S C E N S U S B U R E A U

Bellin-Rubin

- Bellin, T.R. and Rubin, D.B. (1995) “A Method for Calibrating False-Match Rates in Record Linkage,” *Journal of the American Statistical Association*, 90,pp.694–707.
- Consider sample space without 1-1 matching
- Model as a mixture of 2 normal distributions (Box-Cox)
- M and U must be well-separated and unimodal

U S C E N S U S B U R E A U

Larsen

- Larsen, M.D. “Hierarchical Bayesian Record Linkage Theory,” Iowa State University, Statiistics Department Technical Report

USCENSUSBUREAU

Larsen

- Larsen, M.D. “Hierarchical Bayesian Record Linkage Theory,” Iowa State University, Statiistics Department Technical Report
- Estimate error rates with 1-1 matching

U S C E N S U S B U R E A U

Larsen

- Larsen, M.D. “Hierarchical Bayesian Record Linkage Theory,” Iowa State University, Statiistics Department Technical Report
- Estimate error rates with 1-1 matching
- Complicated restrained optimization

U S C E N S U S B U R E A U

Larsen

- Larsen, M.D. “Hierarchical Bayesian Record Linkage Theory,” Iowa State University, Statiistics Department Technical Report
- Estimate error rates with 1-1 matching
- Complicated restrained optimization
- Metropolis-Hastings procedure

Improved Parameter Estimates

- Recall, if we had correct parameter values (and model), under Fellegi-Sunter, error rates are known

U S C E N S U S B U R E A U

Improved Parameter Estimates

- Recall, if we had correct parameter values (and model), under Fellegi-Sunter, error rates are known
- Improve parameter estimates using training data

Extended Likelihood Function

- For unlabled sample space S and labeled training data set T, extended likelihood function

$$L = \left(\prod_{(a,b) \in S} \Pr\left(\gamma\left(a,b\right)\right) \right)^{1-\lambda} \left(\prod_{(a,b) \in T} \Pr\left(\gamma\left(a,b\right)\right) \right)^{\lambda}$$

for $0 \leq \lambda \leq 1$

U S C E N S U S B U R E A U

Extended Likelihood Function

- For unlabled sample space S and labeled training data set T, extended likelihood function

$$L = \left(\prod_{(a,b)\in S} \Pr\left(\gamma\left(a,b\right)\right) \right)^{1-\lambda} \left(\prod_{(a,b)\in T} \Pr\left(\gamma\left(a,b\right)\right) \right)^{\lambda}$$

  for $0 \leq \lambda \leq 1$

- Estimate using EM

U S C E N S U S B U R E A U

Larsen, Rubin

- Larsen, M.D. and Rubin, D.B. (2001) “Iterative Automated Record Linkage Using Mixture Models,” *Journal of the American Statistical Association* 79, pp.32–41

U S C E N S U S B U R E A U

Larsen, Rubin

- Larsen, M.D. and Rubin, D.B. (2001) “Iterative Automated Record Linkage Using Mixture Models,” *Journal of the American Statistical Association* 79, pp.32–41
- T is sample of pairs from the clerical review region that have been clerically reviewed

Winkler

- Winkler, W.E. “Automatically Estimating Record Linkage False Match Rates,” (2007) http://www.census.gov/srd/www/byname.html

U S C E N S U S B U R E A U

Winkler

- Winkler, W.E. “Automatically Estimating Record Linkage False Match Rates,” (2007) http://www.census.gov/srd/www/byname.html
- T is sample of “pseudo-truth” data: pairs with sufficiently high or sufficiently low weight

Data Preparation

- Files must have matching fields of fixed length and location

US CENSUS BUREAU

Data Preparation

- Files must have matching fields of fixed length and location
- Matching fields are compared on a character by character basis

U S C E N S U S B U R E A U

Data Preparation

- Files must have matching fields of fixed length and location
- Matching fields are compared on a character by character basis
- Unnecessary inconsistencies must be removed before matching is done

U S C E N S U S B U R E A U

Basic Preparation

- Consistently encode categorical variables

U S C E N S U S B U R E A U

Basic Preparation

- Consistently encode categorical variables
  - Sex, race

U S C E N S U S B U R E A U

Basic Preparation

- Consistently encode categorical variables
  - Sex, race
  - Date, age

U S C E N S U S B U R E A U

Basic Preparation

- Consistently encode categorical variables
  - Sex, race
  - Date, age
- Spelling standardization

U S C E N S U S B U R E A U

Basic Preparation

- Consistently encode categorical variables
  - Sex, race
  - Date, age
- Spelling standardization
  - Titles: Dr, Dr., Doctor

Basic Preparation

- Consistently encode categorical variables
  - Sex, race
  - Date, age
- Spelling standardization
  - Titles: Dr, Dr., Doctor
  - Nicknames: Bill, William

Basic Preparation

- Consistently encode categorical variables
  - Sex, race
  - Date, age
- Spelling standardization
  - Titles: Dr, Dr., Doctor
  - Nicknames: Bill, William
  - Standard words: Co, Co., Cmpny, Company

Basic Preparation, Cont.

- Identify and parse components

U S C E N S U S B U R E A U

Basic Preparation, Cont.

- Identify and parse components
  - Names: last, first

US CENSUS BUREAU

Basic Preparation, Cont.

- Identify and parse components
  - Names: last, first
  - Addresses: number, street, unit

U S C E N S U S B U R E A U

Address Parsing

- 16 W Main ST APT 16

  RR 2 BX 215

  Fuller BLDG SUITE 405

  14588 HWY 16 W

Address Parsing

- 16 W Main ST APT 16
  RR 2 BX 215
  Fuller BLDG SUITE 405
  14588 HWY 16 W

-

Pre2	Hsnm	Stnm	RR	Box	Post1	Post2	Unit1	Unit2	Bldg
W	16	Main					16		
			2	215					
								405	Fuller
	14588	HWY 16				W			

U S C E N S U S B U R E A U

Business Lists

- Much harder

Business Lists

- Much harder
- May have fewer comparison fields

USCENSUSBUREAU

Business Lists

- Much harder
- May have fewer comparison fields
  - Name

U S C E N S U S B U R E A U

Business Lists

- Much harder
- May have fewer comparison fields
  - Name
  - Address

U S C E N S U S B U R E A U

Business Lists

- Much harder
- May have fewer comparison fields
  - Name
  - Address
  - Phone

U S C E N S U S B U R E A U

Business Lists

- Much harder
- May have fewer comparison fields
  - Name
  - Address
  - Phone
- These may not be unique

Business Lists

- Much harder
- May have fewer comparison fields
  - Name
  - Address
  - Phone
- These may not be unique
- May be difficult to parse

U S C E N S U S B U R E A U

Example of Business Name Parsing

- DR John J Smith MD
  Smith DRY FRM
  Smith & Son ENTP

Example of Business Name Parsing

- DR John J Smith MD
  Smith DRY FRM
  Smith & Son ENTP

-

Pre	First	Mid	Last	Post1	Post2	Bus1	Bus2
DR	John	J	Smith	MD			
			Smith			DRY	FRM
			Smith		Son	ENTP	

U S C E N S U S B U R E A U

Two Kinds of Standardizer

- Deterministic

U S C E N S U S B U R E A U

Two Kinds of Standardizer

- Deterministic
  - Rule based

Two Kinds of Standardizer

- Deterministic
  - Rule based
- Probabilistic

Two Kinds of Standardizer

- Deterministic
  - Rule based
- Probabilistic
  - Hidden Markov model

U S C E N S U S B U R E A U

Rule-Based Standardizer

- U.S. Census Bureau software

U S C E N S U S B U R E A U

Rule-Based Standardizer

- U.S. Census Bureau software
- Based on extensive expert experience

U S C E N S U S B U R E A U

Rule-Based Standardizer

- U.S. Census Bureau software
- Based on extensive expert experience
- Created for a specific sample space

Hidden Markov Standardizer

- Adaptable to different sample spaces

Hidden Markov Standardizer

- Adaptable to different sample spaces
- Based on training data

Hidden Markov Standardizer Referenc

- P. Christen, T. Churches, J.X. Jhu. (2002) "Probabilistic Name and Address Cleaning and Standardization." *The Australasian Data Mining Workshop*. http://datamining.anu.eedu.au/projects/linkage.html

U S C E N S U S B U R E A U

Hidden Markov Standardizer Referenc

- P. Christen, T. Churches, J.X. Jhu. (2002) “Probabilistic Name and Address Cleaning and Standardization.” *The Australasian Data Mining Workshop*. http://datamining.anu.eedu.au/projects/linkage.html
- T. Churches, P. Christen, J. Lu, J.X. Zhu. (2002) “Preparation of Name and Address Data for Record Linkage Using Hidden Markov Models.” *BioMed Central Medical Informatics and Decision Making,* 2(9), http://www.biomedcentral.com/1472-6947/2/9.

Hidden Markov Standardizer Referenc

- P. Christen, T. Churches, J.X. Jhu. (2002) “Probabilistic Name and Address Cleaning and Standardization.” *The Australasian Data Mining Workshop*. http://datamining.anu.eedu.au/projects/linkage.html
- T. Churches, P. Christen, J. Lu, J.X. Zhu. (2002) “Preparation of Name and Address Data for Record Linkage Using Hidden Markov Models.” *BioMed Central Medical Informatics and Decision Making,* 2(9), http://www.biomedcentral.com/1472-6947/2/9.
- FEBRL Project (Freely Extensible Biomedical Record Linkage)

Hidden Markov Model

- Identify a finite number of hidden Markov states

USCENSUSBUREAU

Hidden Markov Model

- Identify a finite number of hidden Markov states
  - first, last1, last2, mi, prefix, suffix

U S C E N S U S B U R E A U

Hidden Markov Model

- Identify a finite number of hidden Markov states
  - first, last1, last2, mi, prefix, suffix
- Use training data to assign transition probabilities from one state to the next

U S C E N S U S B U R E A U

Hidden Markov Model

- Identify a finite number of hidden Markov states
  - first, last1, last2, mi, prefix, suffix
- Use training data to assign transition probabilities from one state to the next
- Use training data to assign probabilities for observations having given hidden state

U S C E N S U S B U R E A U

Hidden Markov Model

- Identify a finite number of hidden Markov states
  - first, last1, last2, mi, prefix, suffix
- Use training data to assign transition probabilities from one state to the next
- Use training data to assign probabilities for observations having given hidden state
  - Look-up lists

Hidden Markov Model

- Identify a finite number of hidden Markov states
  - first, last1, last2, mi, prefix, suffix
- Use training data to assign transition probabilities from one state to the next
- Use training data to assign probabilities for observations having given hidden state
  - Look-up lists
  - Coded rules

U S C E N S U S B U R E A U

Hidden Markov Model, Cont.

- Break object into component observations, assign them initial Markov states

Hidden Markov Model, Cont.

- Break object into component observations, assign them initial Markov states
  - “sir”, “mick”, “jagger”, “mbe”

U S C E N S U S B U R E A U

Hidden Markov Model, Cont.

- Break object into component observations, assign them initial Markov states
  - “sir”, “mick”, “jagger”, “mbe”
- Compute the highest probability sequence of hidden states for the given observations

Viterbi Algorithm

- Not feasible to compute probabilities for all possible paths $O\left(n^{l}\right)$

U S C E N S U S B U R E A U

Viterbi Algorithm

- Not feasible to compute probabilities for all possible paths $O\left(n^l\right)$
- Dynamic programming algorithm $O\left(nl\right)$

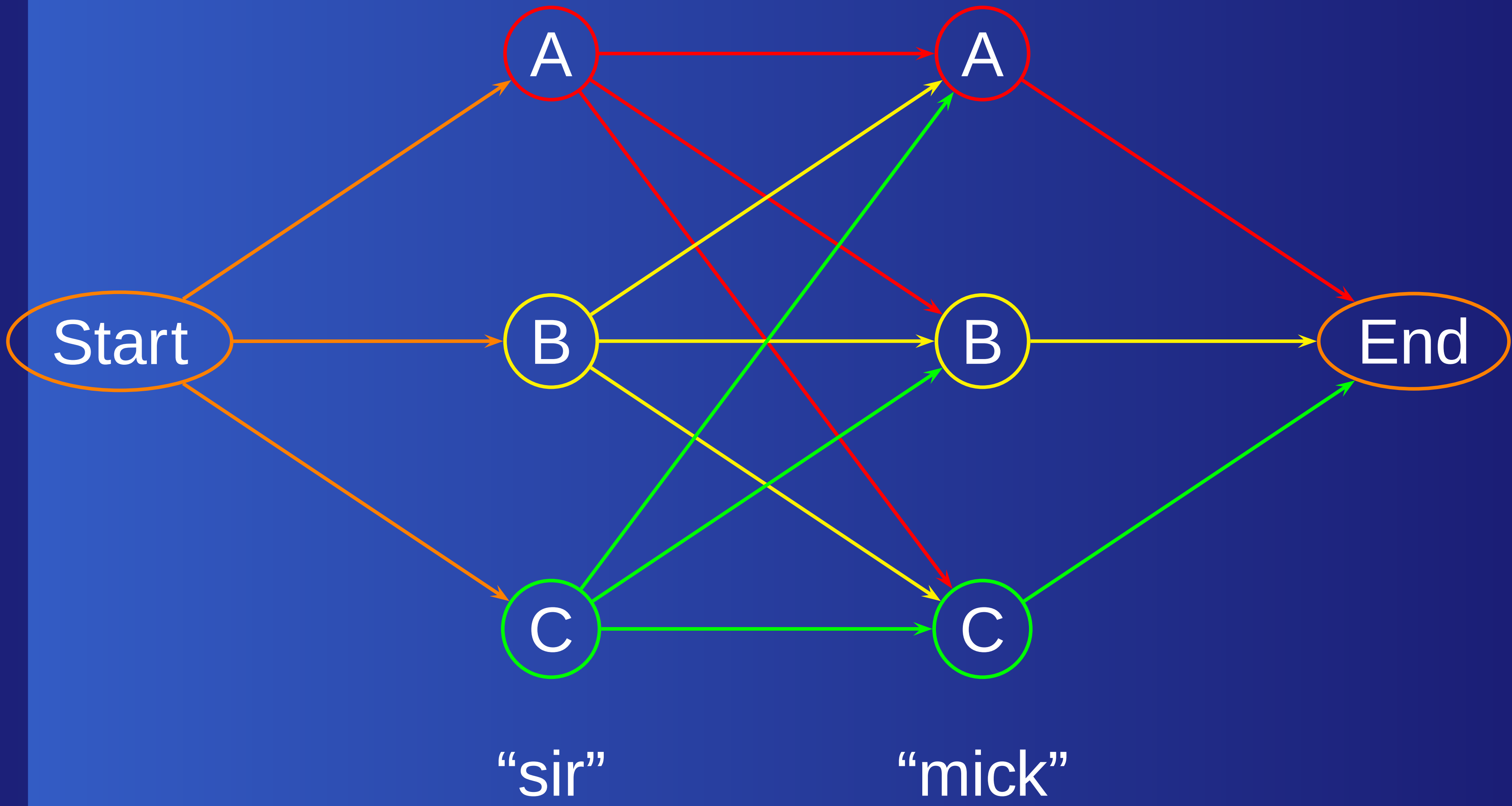
U S C E N S U S B U R E A U

Viterbi Algorithm

- Not feasible to compute probabilities for all possible paths $O\left(n^l\right)$
- Dynamic programming algorithm $O\left(nl\right)$
- Each state is arrived at by the most probable subpath (Markov property)

U S C E N S U S B U R E A U

HMM Diagram



Standardization Summary

- Much more time is likely to be spent preparing the data than performing the record linkage

U S C E N S U S B U R E A U

Standardization Summary

- Much more time is likely to be spent preparing the data than performing the record linkage
- Records that fail to be standardized will probably fail to be matched

U S C E N S U S B U R E A U

U.S. Census Bureau Software

- Matching programs

U.S. Census Bureau Software

- Matching programs
  - Matcher

U S C E N S U S B U R E A U

U.S. Census Bureau Software

- Matching programs
  - Matcher
  - Bigmatch

U S C E N S U S B U R E A U

U.S. Census Bureau Software

- Matching programs
  - Matcher
  - Bigmatch
- Auxiliary programs

USCENSUSBUREAU

U.S. Census Bureau Software

- Matching programs
  - Matcher
  - Bigmatch
- Auxiliary programs
  - Counter

U S C E N S U S B U R E A U

U.S. Census Bureau Software

- Matching programs
  - Matcher
  - Bigmatch
- Auxiliary programs
  - Counter
  - EM

U.S. Census Bureau Software

- Matching programs
  - Matcher
  - Bigmatch
- Auxiliary programs
  - Counter
  - EM
  - Standardizer

U S C E N S U S B U R E A U

Matching Programs: Matcher

- Matcher

Matching Programs: Matcher

- Matcher
  - One-to-one matching

U S C E N S U S B U R E A U

Matching Programs: Matcher

- Matcher
  - One-to-one matching
    - Files should not have duplicates

U S C E N S U S B U R E A U

Matching Programs: Matcher

- Matcher
  - One-to-one matching
    - Files should not have duplicates
  - Pre-sort files according to blocking scheme

U S C E N S U S B U R E A U

Matching Programs: Matcher

- Matcher
  - One-to-one matching
    - Files should not have duplicates
  - Pre-sort files according to blocking scheme
  - Can re-run program on residual files

U S C E N S U S B U R E A U

Matching Programs: Matcher

- Matcher
  - One-to-one matching
    - Files should not have duplicates
  - Pre-sort files according to blocking scheme
  - Can re-run program on residual files
    - Resort files according to new blocking scheme

U S C E N S U S B U R E A U

Matching Programs: Bigmatch

- Bigmatch

U S C E N S U S B U R E A U

Matching Programs: Bigmatch

- Bigmatch
  - No one-to-one matching

U S C E N S U S B U R E A U

Matching Programs: Bigmatch

- Bigmatch
  - No one-to-one matching
    - Can be used for deduplicating file

U S C E N S U S B U R E A U

Matching Programs: Bigmatch

- Bigmatch
  - No one-to-one matching
    - Can be used for deduplicating file
  - Do not pre-sort files

U S C E N S U S B U R E A U

Matching Programs: Bigmatch

- Bigmatch
  - No one-to-one matching
    - Can be used for deduplicating file
  - Do not pre-sort files
  - Can run several blocking schemes

Matching Programs: Bigmatch

- Bigmatch
  - No one-to-one matching
    - Can be used for deduplicating file
  - Do not pre-sort files
  - Can run several blocking schemes
  - Can match several files to one file

Matching Programs: Bigmatch

- Bigmatch
  - No one-to-one matching
    - Can be used for deduplicating file
  - Do not pre-sort files
  - Can run several blocking schemes
  - Can match several files to one file
  - One file must fit into memory

U S C E N S U S B U R E A U

Auxiliary Programs: Counter

- Counter program

Auxiliary Programs: Counter

- Counter program
    - Simplified matching program

Auxiliary Programs: Counter

- Counter program
  - Simplified matching program
  - Counts number of times each matching pattern occurs

Auxiliary Programs: Counter

- Counter program
  - Simplified matching program
  - Counts number of times each matching pattern occurs
  - String comparator has (high) cutoff

Auxiliary Programs: Counter

- Counter program
  - Simplified matching program
  - Counts number of times each matching pattern occurs
  - String comparator has (high) cutoff
  - Provides input for EM

U S C E N S U S B U R E A U

Auxiliary Programs: EM

- EM algorithm program

Auxiliary Programs: EM

- EM algorithm program
  - Estimates probability parameters for given file and blocking scheme

U S C E N S U S B U R E A U

Auxiliary Programs: EM

- EM algorithm program
  - Estimates probability parameters for given file and blocking scheme
  - Has 2-class and 3-class versions

U S C E N S U S B U R E A U

Auxiliary Programs, Standardizer

- Standardizer

U S C E N S U S B U R E A U

Auxiliary Programs, Standardizer

- Standardizer
  - Standardizes names and addresses

U S C E N S U S B U R E A U

Auxiliary Programs, Standardizer

- Standardizer
  - Standardizes names and addresses
  - Rule-based parsing