Deduplicating Large-Scale Databases

Samuel L. Ventura[1]
Rebecca Nugent[1] Erica R.H. Fuchs[2]

1 Department of Statistics, Carnegie Mellon University
2 Engineering & Public Policy Department, Carnegie Mellon University

October 9, 2013

USPTO Deduplication Example

Which inventor records from the US Patent & Trademark Office (USPTO) database correspond to the same unique individuals?

Last	First	Middle	City	St	Assignee
...	...	...	...	...	...
Millar	David	A.	Stanford	CA	Stanford University
Miller	Dave	A.	Fair Haven	NJ	UNC
Miller	David	A.B.	Stanford	CA	Stanfrod University
Miller	David	Andrew	Stanford	CA	Lucent Technologies
Miller	David	Andrew	Fair Haven	NJ	Lucent Technologes
Miller	David	B.	Los Angeles	CA	Agilent Technologies
Miller	David	D.	Billerca	MA	Lucent Technologies
...	...	...	...	...	...

USPTO: 8 million patents, multiple inventors per patent

Our Deduplication Approach

How to find the probability that each record pair matches?

- Linear combination of similarity scores? e.g.:
  $P(M) = 0.35 * \mathit{last} + 0.25 * \mathit{first} + 0.25 * \mathit{DOB} + 0.15 * \mathit{address}$
- Based on labeled/training data?
  Where does labeled/training data come from?

Labeled Inventor Records

How would YOU create labeled USPTO inventor records?

Labeled Inventor Records

1. Stats: Unique inventors approximated via some simple record linkage procedure (exact matching, etc)
2. EPP: Inventor contact information obtained from various sources (e.g. professional societies like IEEE)
3. EPP: Researchers contact inventors, request their resume/CV and a list of all patents
4. Stats: For each contacted inventor, generate a list of "potential matches" – records with field information similar enough to be considered for clerical review
5. EPP: Manually review 100,000 potentially matching records, labeling each one with an ID number corresponding to one of the contacted inventors

Are we done? How could we improve?

Labeled Inventor Records

6. Stats: Run simple sanity checks on the human labels

7. Stats: Compare pairs of records with similarity scores.
Also compare the IDs, so that we get a table like this:

ID_1	ID_2	Last	First	Mid	City	St	Assignee	Co-Inv	Class	Match
1	4	0.93	1.00	0.75	1.00	1	0.50	1	1	Yes
1	7	0.93	1.00	0.00	0.42	0	0.50	0	1	No
4	7	1.00	1.00	0.00	0.42	0	1.00	0	0	No

Pairwise comparisons of 98,762 labeled inventor records

- $\approx$ **100 million record-pairs**, labeled as Match or Non-Match
- Compare records with similarity scores for each field

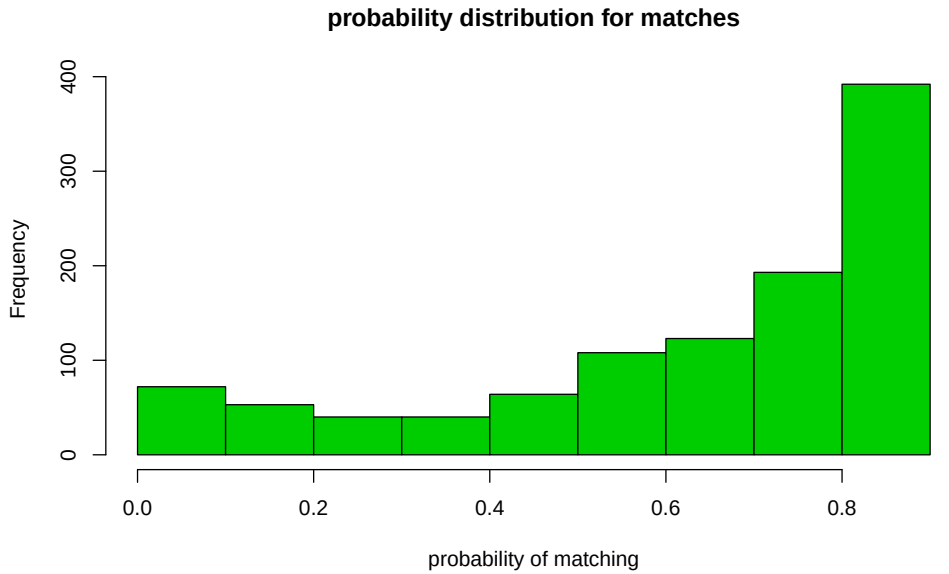
Checking for mistakes with $\hat{p}_{ij}$

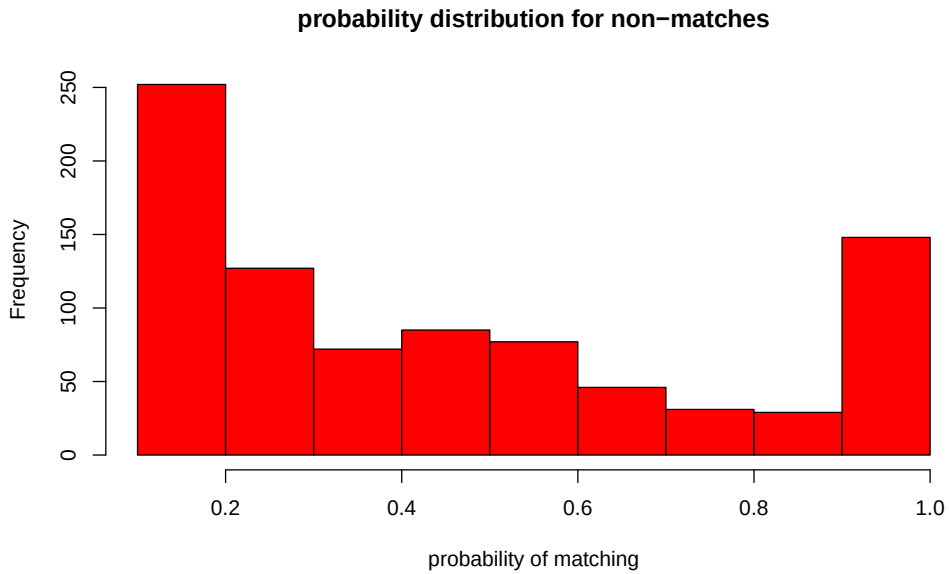
$\hat{p}_{ij}$: The **probability** that records i, j match

- Train a **classification** model on labeled data
- Use the model to **predict** whether record-pairs match
- Result: **Pairwise matching probability** for any record-pair

Compare predicted probabilities to clerk's match/non-match labels

Checking for mistakes with $\hat{p}_{ij}$





Labeled Inventor Records

8. Stats: Train match/non-match classification models

9. Stats: Compare predicted probabilities from models to the match/non-match labels to identify potential mistakes

10. EPP: Go though the potential mistakes, identify if they are human errors

Results: Labeled records that we can trust!

Classification Performance

We apply this approach to deduplicate USPTO inventors

Deduplication Method	False Neg. (%)	False Pos. (%)
Lai et al (2009)[2]	8.39	4.13
Linear Discriminant Analysis	8.48	1.64
Quadratic Discriminant Analysis	3.19	1.62
Classification Trees	2.23	2.49
Logistic Regression	1.68	1.64
Random Forests	0.62	0.74

Classification approaches outperform heuristic approaches

2 Lai et al (2009): A heuristic approach for deduplicating USPTO inventors

Too Much Data?

Is it possible to have too much training data?

What computational issues arise when training datasets are large?

Is 100,000 labeled records "too much" data?

Storing Similarity Scores

To reduce the number of calculations, for each field k:

- Let N_k = the number of unique values in field k
- Compute all $\binom{N_k}{2}$ similarity scores

For large values of N_k, calculations can take up to several days and several gigabytes of storage space

Comparison Field	# of Unique Values	Storage Space (MB)
Last	62,903	5500
First	25,045	700
Middle	4,269	19
Suffix	104	<1
City	25,711	1100
State	55	<1
Country	291	<1
Assignee	26,610	2000

Forest of Random Forests

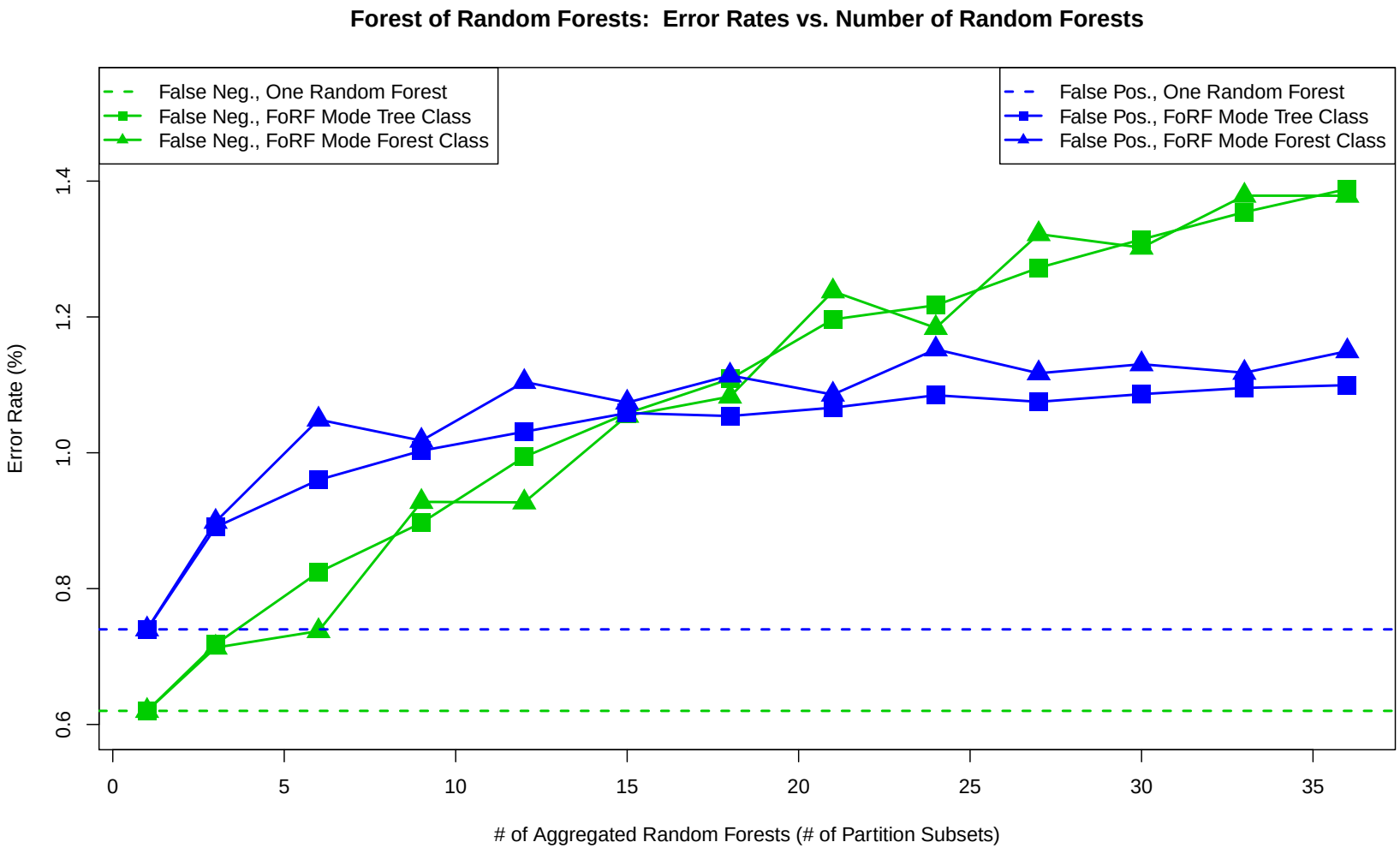
Issue: Difficult to train single classifier on 100 million observations

Solution: **Train multiple, smaller classifiers and aggregate**

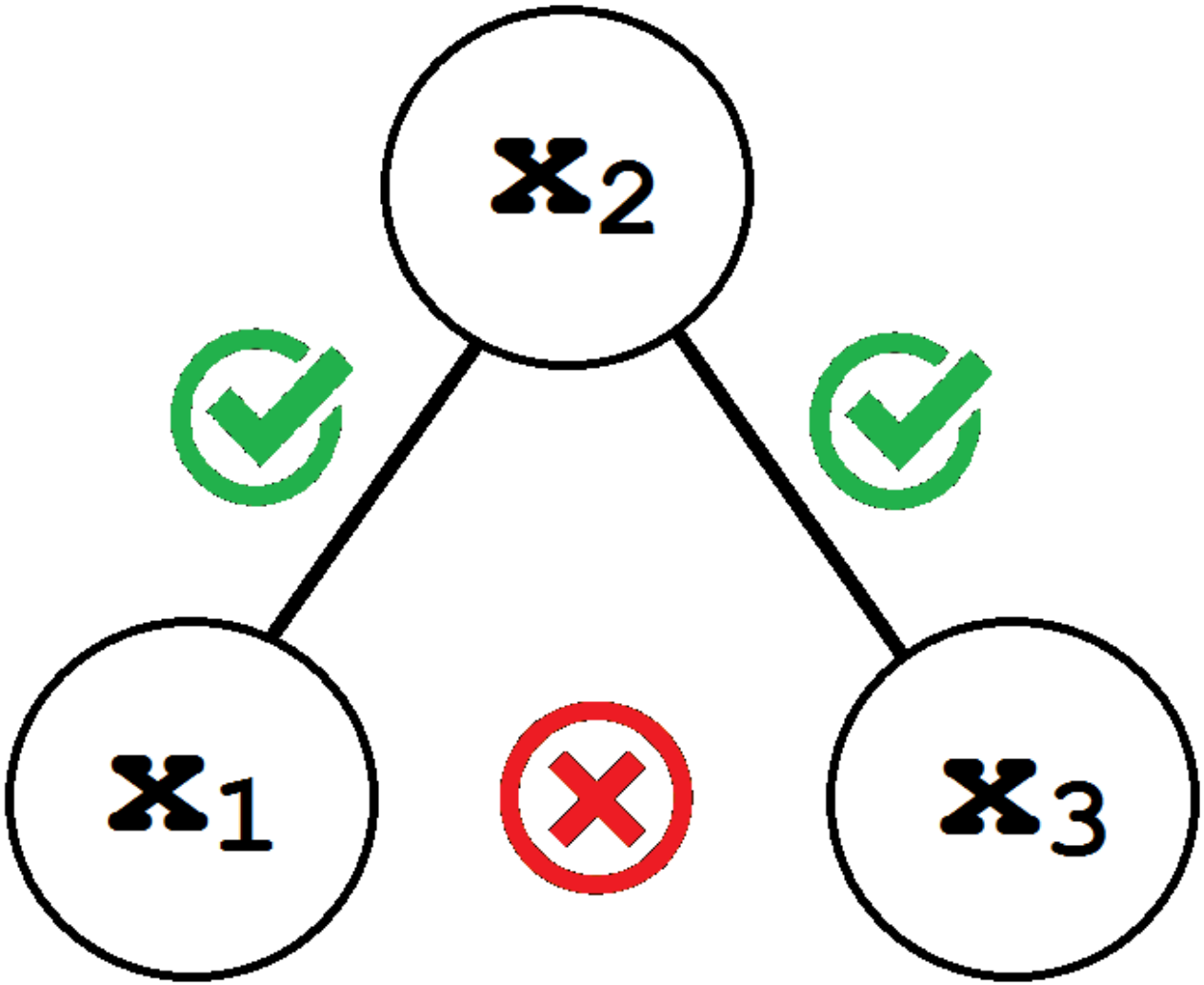
Forest of Random Forests (FoRF) algorithm:

1. Split the training data into R disjoint random subsets
2. Let F_r be the random forest trained on subset $r = 1, ..., R$
3. Let $F^{rand} = \{F_r\}_{r=1}^{R}$ be the forest of random forests with random subsets
4. Aggregate predictions of each F_r when predicting

Error Rates vs. Number of FoRF Partition Subsets



Linking Records to Unique Entities



Which records are duplicated?

Linking Records to Unique Entities – Clustering

Solution: **Cluster records using pairwise distances**

Create distance matrix D using predicted probabilities ($\hat{p}_{ij}$)

- If n records, then D is $n \times n$
- $D[i,j] = d_{ij} = h(\hat{p}_{ij})$
- h: monotonic inverse function of $\hat{p}_{ij}$

  e.g. $h(\hat{p}_{ij}) = 1 - \hat{p}_{ij}$, $h(\hat{p}_{ij}) = e^{-\hat{p}_{ij}}$, $h(\hat{p}_{ij}) = 1/(1 + \hat{p}_{ij})$

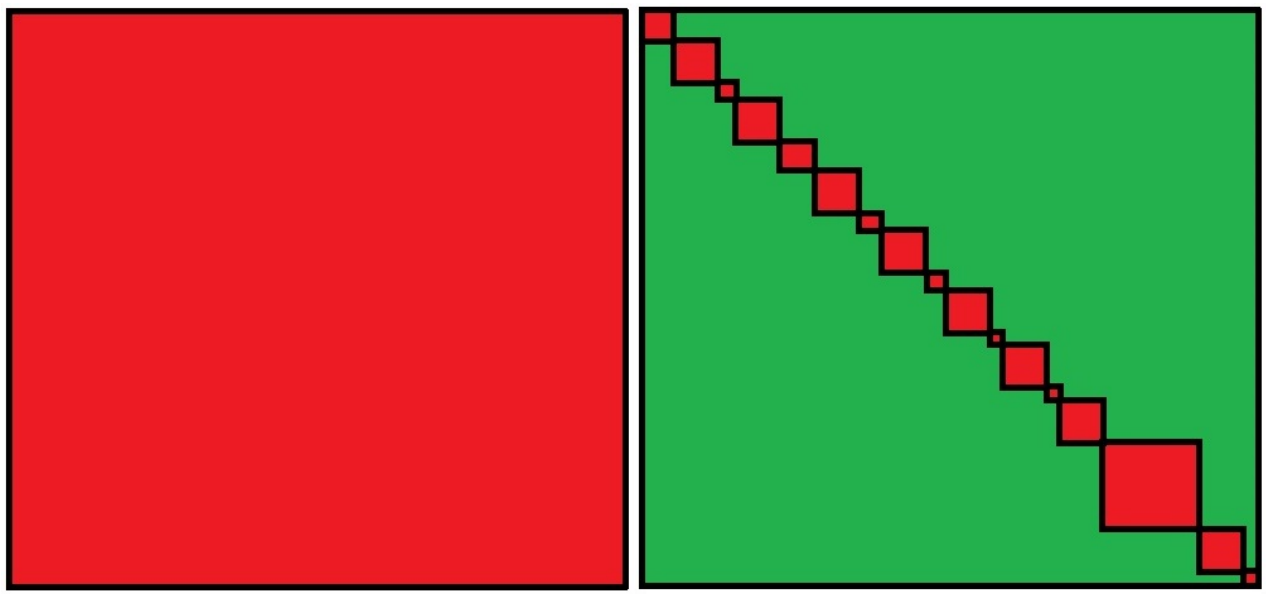
Duplicated records assigned to same cluster

Blocking in Large Scale Deduplication Problems

Deduplication: Compare all pairs of n records – $\binom{n}{2}$ comparisons

- 8 million USPTO records $\Longrightarrow$ **32 trillion comparisons**
- 300 million Census records $\Longrightarrow$ **45 quadrillion comparisons**

Common Solution: Blocking (only compare records within blocks)



False negative errors from blocking?

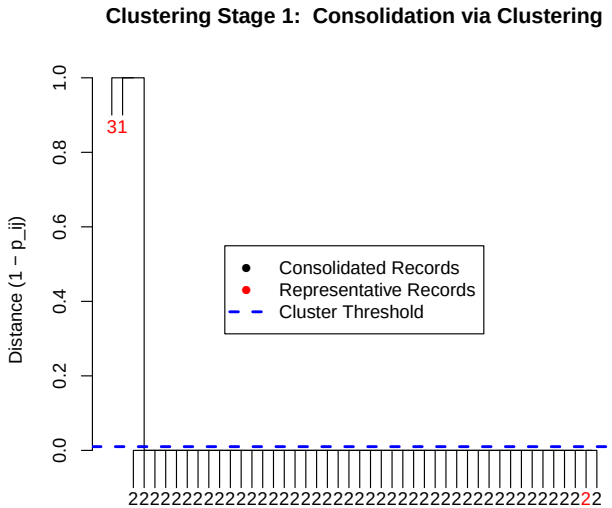
Blocked Hierarchical Clustering

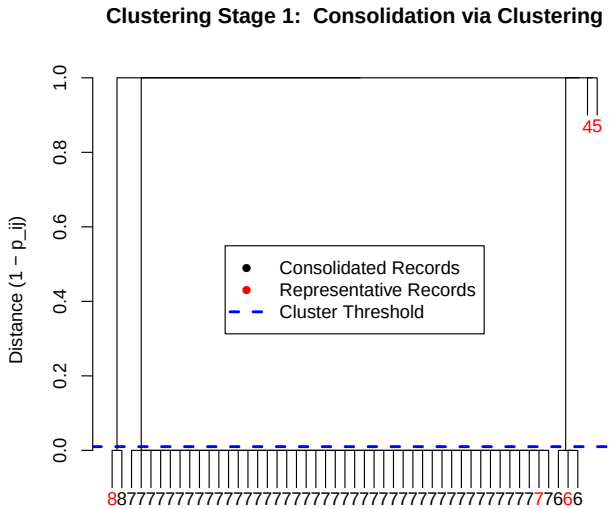
1. Partition the data into blocks of records X_b
   (e.g. blocks of records which share the same last name)

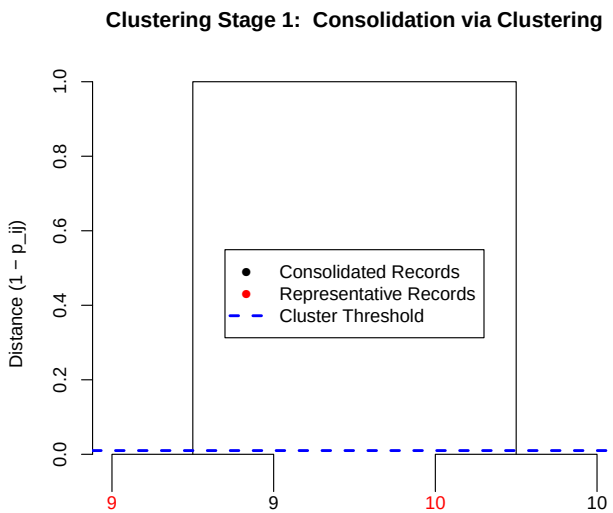
2. Within each block of records X_b:
   - 2.1 Calculate D_b using the $\hat{p}_{ij}$s from classification
   - 2.2 Build the single-linkage hierarchical clustering tree using D_b
   - 2.3 Cut the tree at a level corresponding to $\hat{p}_{ij} = \tau_1$
     to identify clusters of records
   - 2.4 Find each block-cluster's "representative record":
     record with highest mean within-cluster probability of matching
   - 2.5 Consolidate duplicated records within clusters

3. Repeat 2.1 – 2.3 with the resulting representative records

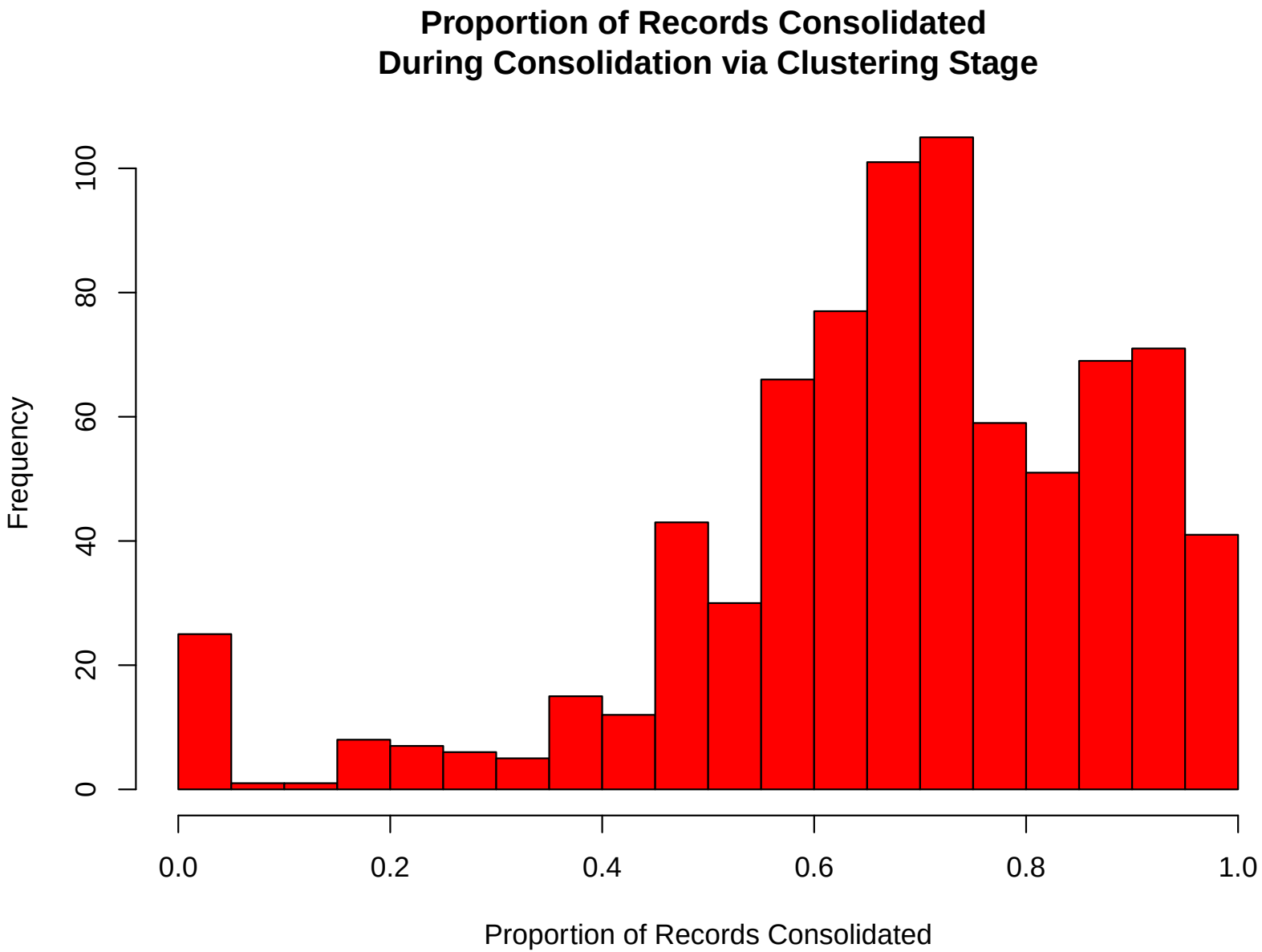
Identify Representative Records



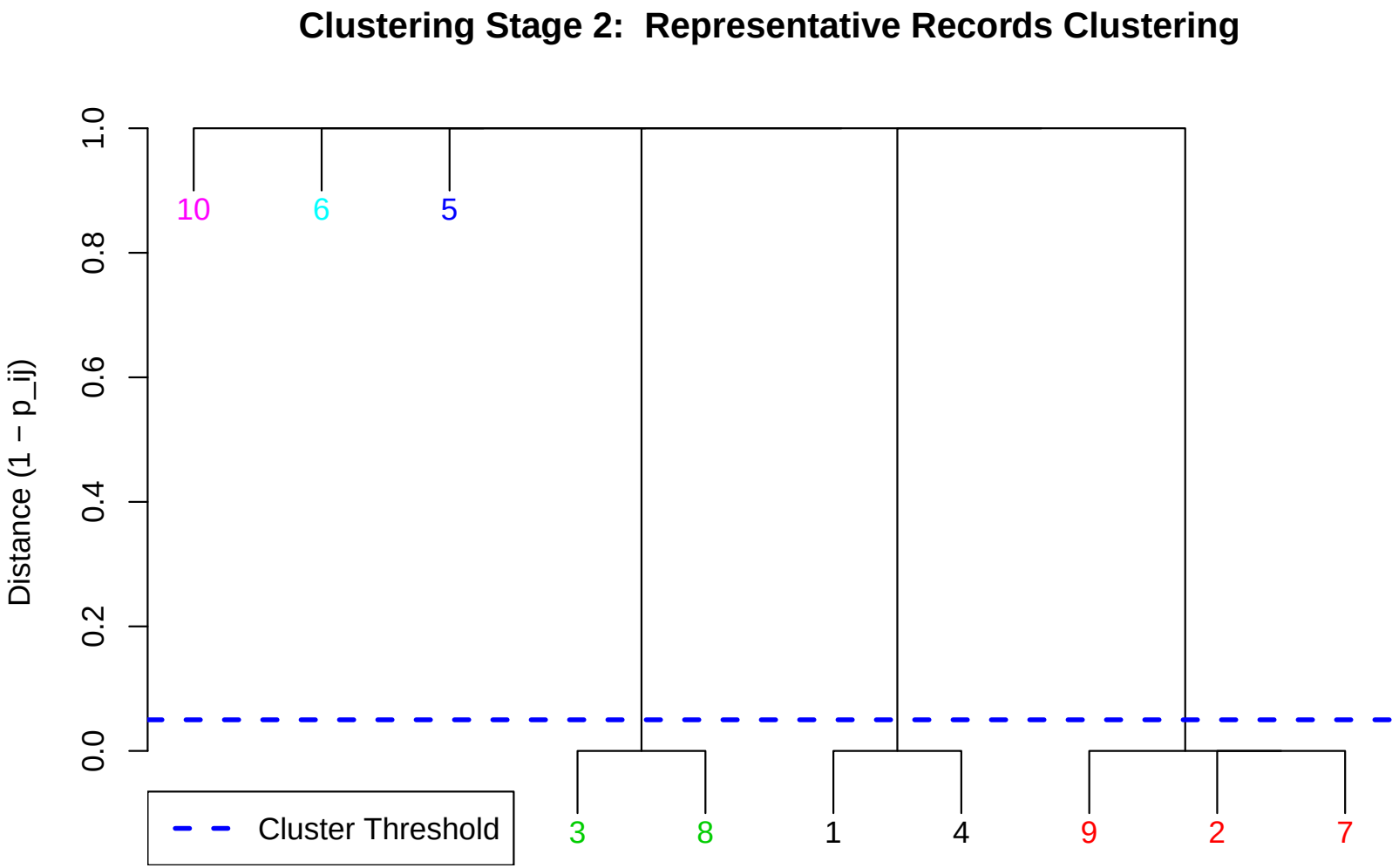




Consolidate Duplicated Records



Cluster Representative Records



Results

Last	First/Mid	City	St/Co	Assignee	ID
De Groot	Edwin	Saratoga	CA	NA	1
de Groot	Edwin	Saratoga	CA	Philips Lumileds Lighting	4
de Groot	P.	Middletown	CT	Zygo Corporation	5
de Groot	Paul	Grenoble	FR	Thomson CSF	6
De Groot	Peter J	Middletown	CT	Zygo Corporation	2
de Groot	Peter J.	Bethel	CT	The Perkin-Elmer	7
deGroot	Peter J.	Middletown	CT	Boeing	9
De Groot	Wilhelmus	Palo Alto	CA	Silicon Light Machines	3
de Groot	Wilhelmus A.	Palo Alto	CA	QUALCOMM MEMS	8
deGroot	Wilhemus A.	Rocky River	OH	S3 Incorporated	10

Comparing to One-Stage Hierarchical Clustering

Relatively uncommon last name (DeGroot)

Clustering Algorithm	Records	Run-Time (sec)	Comparisons
Blocked Hierarchical Clustering	101	0.32	2,316
Standard Hierarchical Clustering	101	0.41	5,050

Very common last name (Miller)

Clustering Algorithm	Records	Run-Time (sec)	Comparisons
Blocked Hierarchical Clustering	944	58.81	363,927
Standard Hierarchical Clustering	944	75.21	445,096

Both approaches yield the same (correct!) deduplication results

Comparing to One-Stage Hierarchical Clustering

Small dataset, many unique last names

Clustering Algorithm	Records	Run-Time (sec)	Comparisons
Blocked Hierarchical Clustering	426	5.56	731
Standard Hierarchical Clustering	426	10.42	90,525

Small/moderate dataset, many unique last names

Clustering Algorithm	Records	Run-Time (sec)	Comparisons
Blocked Hierarchical Clustering	1,657	41.34	7,217
Standard Hierarchical Clustering	1,657	384.27	1,371,996

Moderate dataset, many unique last names

Clustering Algorithm	Records	Run-Time (sec)	Comparisons
Blocked Hierarchical Clustering	3,821	197.42	23,028
Standard Hierarchical Clustering	3,821	4019.8	7,298,110

USPTO Deduplication Example

How did our classification + clustering approach perform for the David Miller(s) example?

Last	First	Middle	City	St	Assignee	True ID	Our ID
...	...	...	...	...	...	...	...
Millar	David	A.	Stanford	CA	Stanford University	1001	1
Miller	Dave	A.	Fair Haven	NJ	UNC	1001	1
Miller	David	A.B.	Stanford	CA	Stanfrod University	1001	1
Miller	David	Andrew	Stanford	CA	Lucent Technologies.	1001	1
Miller	David	Andrew	Fair Haven	NJ	Lucent Technologes	1001	1
Miller	David	B.	Los Angeles	CA	Agilent Technologies	1001	1
Miller	David	D.	Billerca	MA	Lucent Technologies	1002	2
...	...	...	...	...	...	...	

Single Linkage = Enforcing Transitivity

Enforcing Transititivy of Pairwise Matches:

- Compare $\hat{p}_{ij}$ to some matching threshold, p^*
- $\hat{p}_{ij} \geq p^* \implies$ Match
- $\hat{p}_{ij} < p^* \implies$ Non-Match
- Chain together pairwise matches

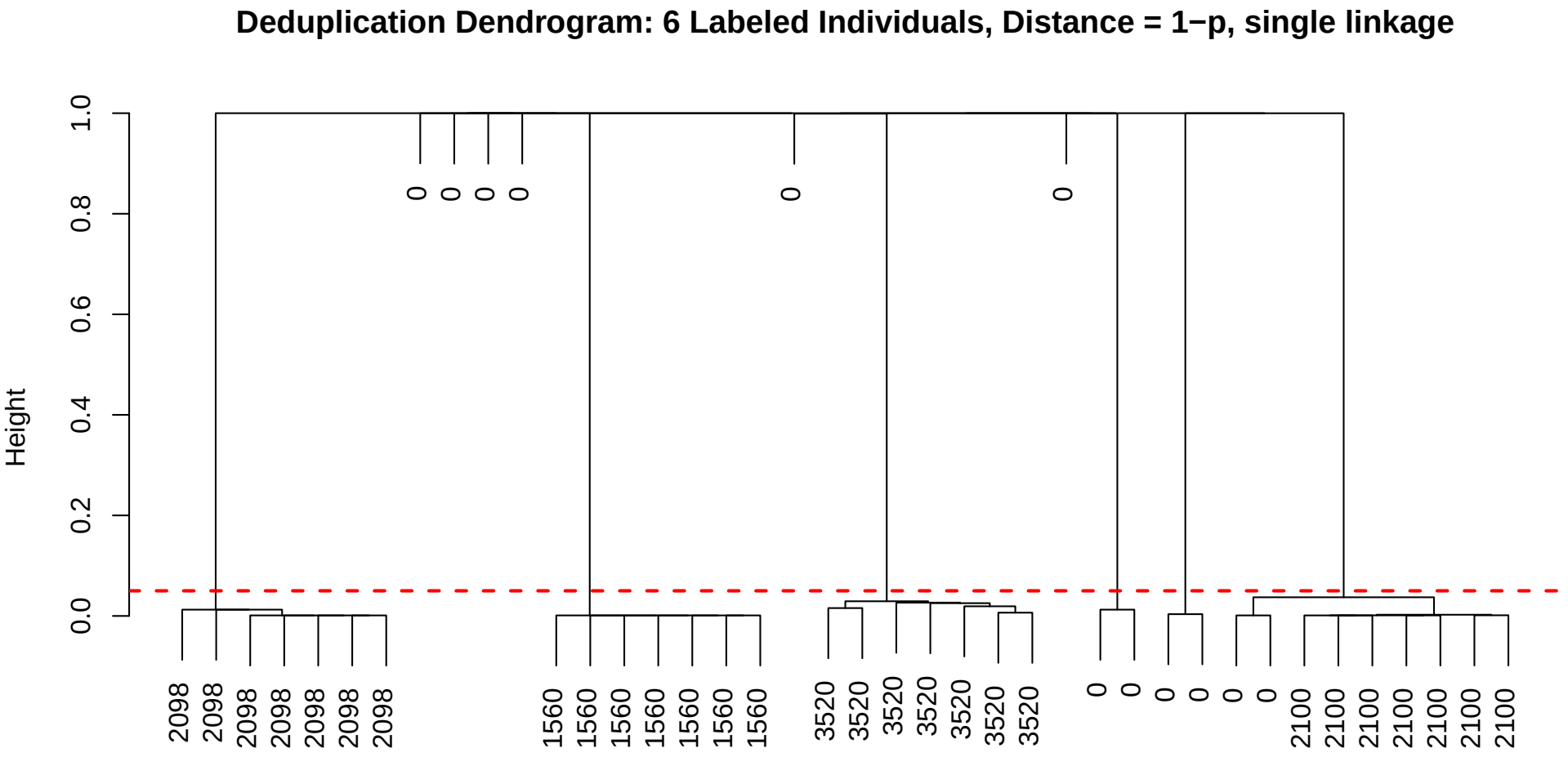
Single Linkage:

- Chain together pairs of observations with lowest $h(\hat{p}_{ij})$, where $h(\hat{p}_{ij})$ is the distance metric
- Cut the tree at some distance threshold, $h(p^*)$

Enforcing transitivity at p^* = Cutting single linkage tree at $h(p^*)$

Clustering for Deduplication: Single Linkage

Single linkage: More susceptible to false positive errors



Inventor 2100 (and friends)

Last	First	Mid	City	St	Assignee	True ID	Our ID
Zarian	James	R.	Corona Del Mar	CA	Lumenyte	2100	1
Zarian	James	R.	Newport Beach	CA	Lumenyte	2100	1
Zarian	Jashmid	J.	Woodland Hills	CA	Lumenyte	0	1
Zarian	Jashmid	NA	Woodland Hills	CA	Lumenyte	0	1
Zara	Michael	NA	Vienna	VA	Duke Univ.	0	2
Zara	Michael	NA	Vienna	VA	GW Univ.	0	2

Michael Zara:

- Correctly separated from James Zarian
- Correctly linked from Duke to George Washington?

Jashmid Zarian:

- Incorrectly linked to James Zarian
- Correctly linked across middle name differences?
- James's father?