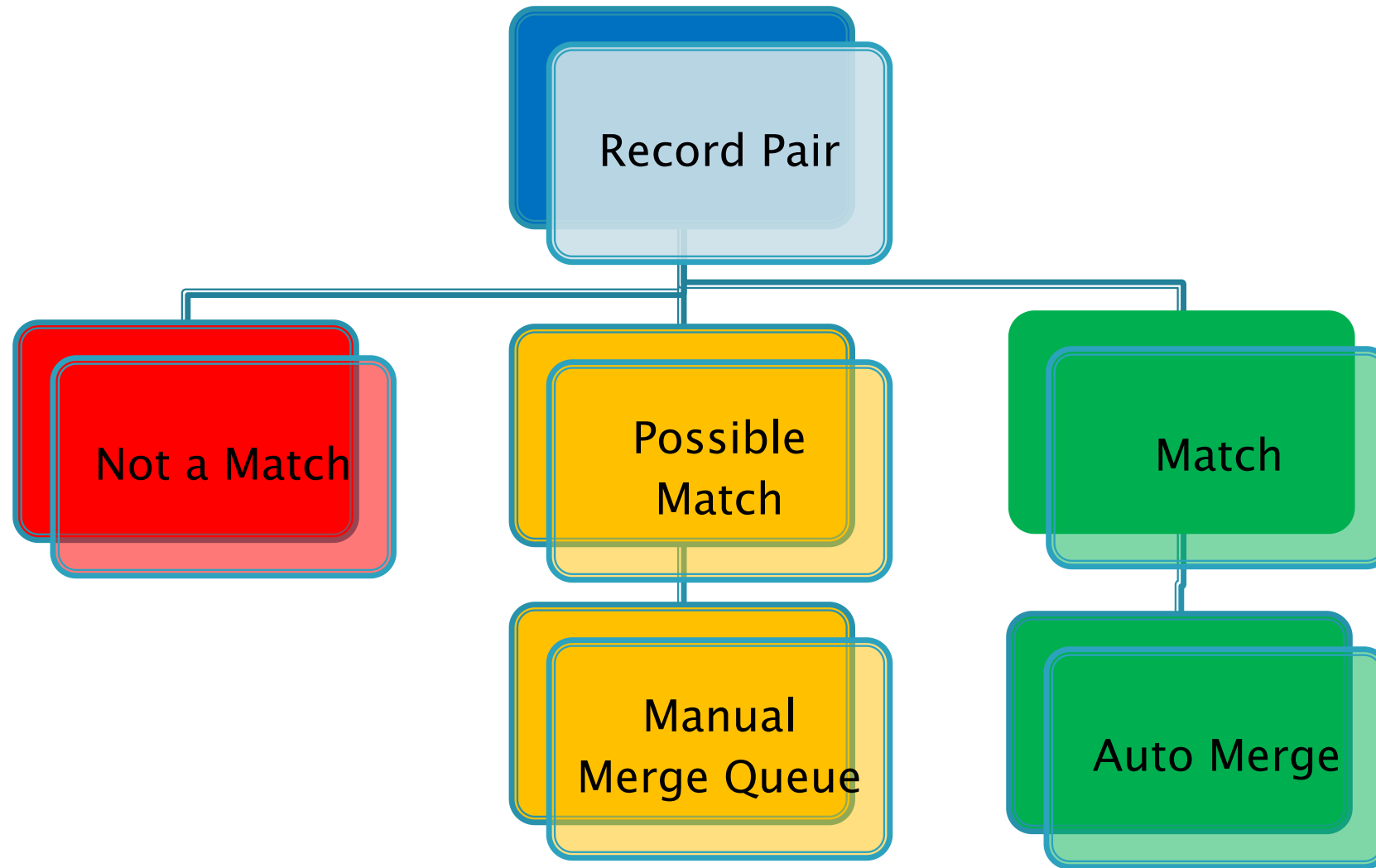# Patient Deduplication Analysis:
A case study in identifying issues with the Massachusetts patient deduplication algorithm using a standalone tool

Sam Belton
Strategic Solutions Group
sbelton @ssg-llc.com
MIIS IT Project Team
Production Support

Agenda

1. Current MIIS deduplication methods
2. Methodology and tools for analysis
3. Evaluation of high priority categories
4. Detailed review of individual match cases
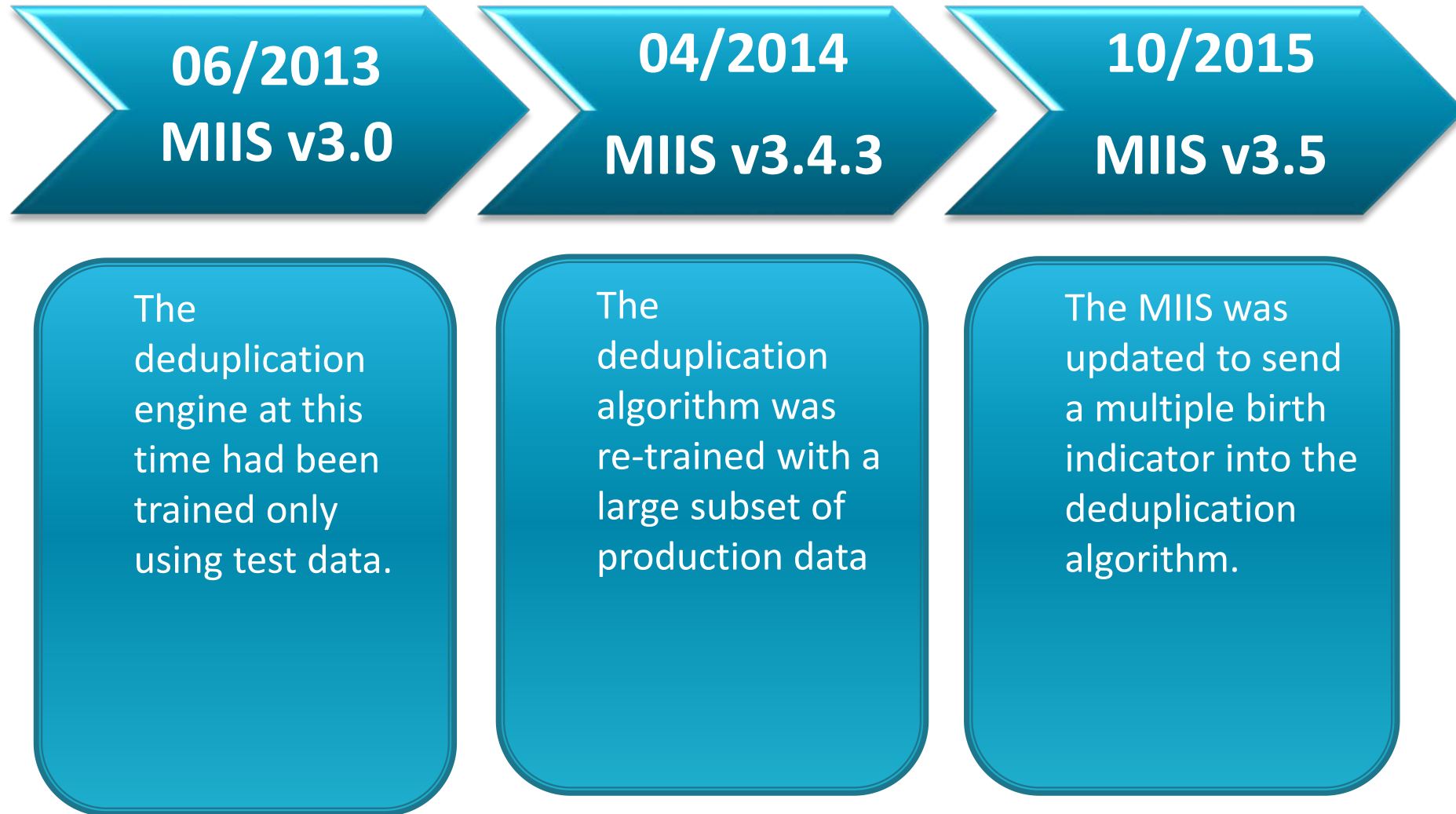5. Outcomes and updates
6. Future considerations

SSG

Massachusetts IIS Patient Deduplication

- The Massachusetts IIS (MIIS) currently uses a weight based algorithm that has been trained using both test data as well as production data.

- Currently it is very hard to judge if there are any widespread issues.

- Before this investigation we would only become aware of issues if they were reported by end users to the MIIS support team.

Timeline of Patient Deduplication in the MIIS

06/2013 MIIS v3.0

The deduplication engine at this time had been trained only using test data.

04/2014 MIIS v3.4.3

The deduplication algorithm was re-trained with a large subset of production data

10/2015 MIIS v3.5

The MIIS was updated to send a multiple birth indicator into the deduplication algorithm.

SSG
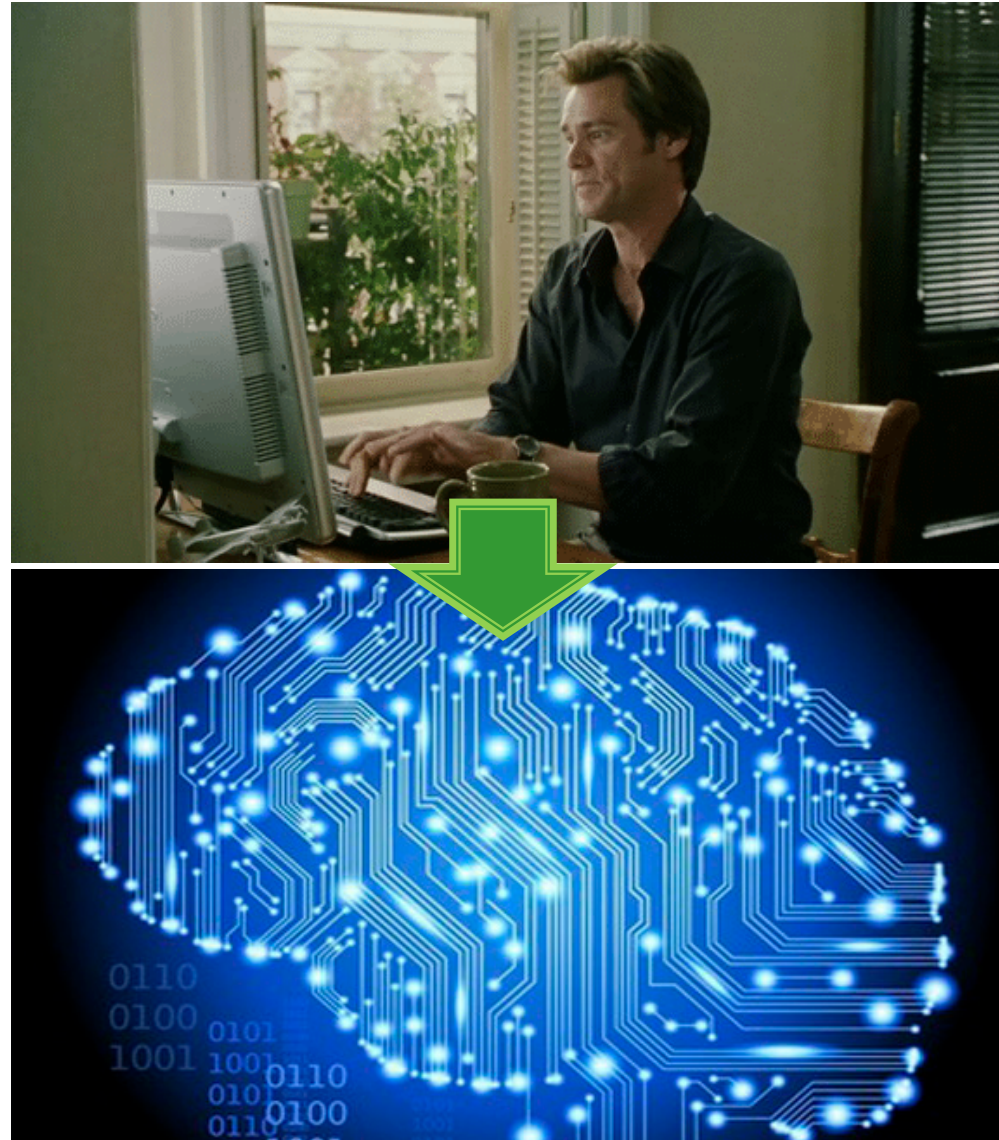
Problem Statement

We wanted to solve two issues:

1) Fix any over deduping that may be occurring
2) Reduce the number of records in the Manual Merge Queue to get to a more manageable size.

   - At the time the queue had over 400k pairs
   - This would take 3 FTEs over 10 months to complete by hand and would cost over $200k
   - Without an adjustment to our dedup algorithm the queue will continue to grow.
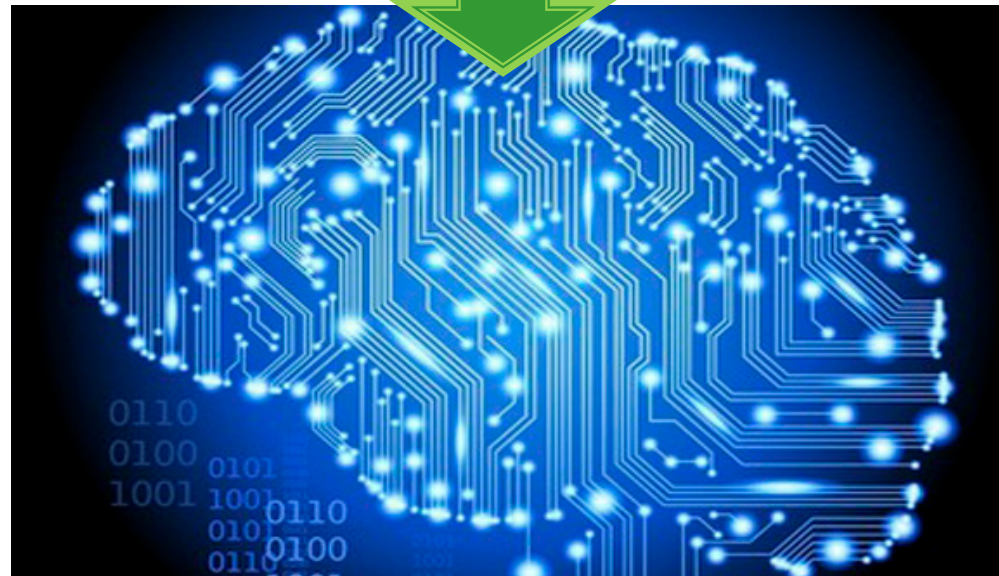
Patients in MIIS
By Category
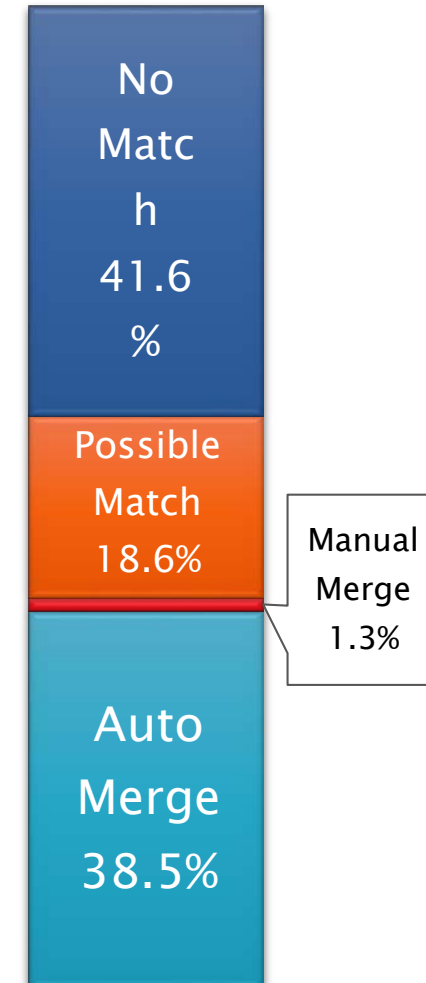
No Match
41.6%

Possible Match
18.6%

Manual Merge
1.3%

Auto Merge
38.5%

Concept for Lantern Person Matching Tool

- Develop a tool to do this analysis work automatically.
- The tool will sit outside of the IIS application, accessing the data via a DB connection.
- Configurable to be used with any record system.
- Our tool codenamed "Lantern", would use deep learning technology (a form of AI) to learn patterns and accurately determine the probability that any two pairs are a match.
- Trained with Massachusetts production data.
- For any pair of records analyzed, a predicted matching probability would be produced.

Concept for Lantern Person Matching Tool

- Develop a tool to do this analysis work automatically.
- The tool will sit outside of the IIS application, accessing the data via a DB connection.
- Configurable to be used with any record system.
- Our tool codenamed "Lantern", would use deep learning technology (a form of AI) to learn patterns and accurately determine the probability that any two pairs are a match.
- Trained with Massachusetts production data.
- For any pair of records analyzed, a predicted matching probability would be produced.

Categories of Interest
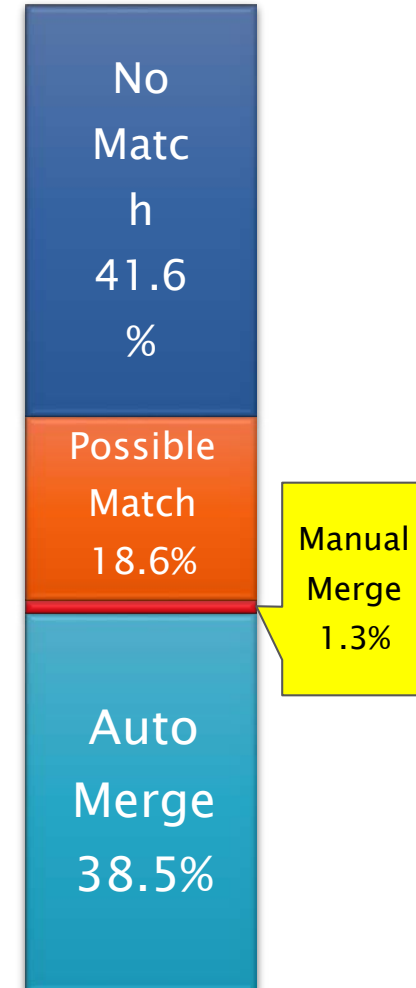
1) Auto Merges with low probability score

Patients in MIIS
By Category

No Match 41.6%

Possible Match 18.6%

Manual Merge 1.3%

Auto Merge 38.5%

Categories of Interest

1) Auto Merges with low probability score
2) Manual Merges with low probability score

Patients in MIIS
By Category

No Match 41.6%

Possible Match 18.6%

Manual Merge 1.3%

Auto Merge 38.5%
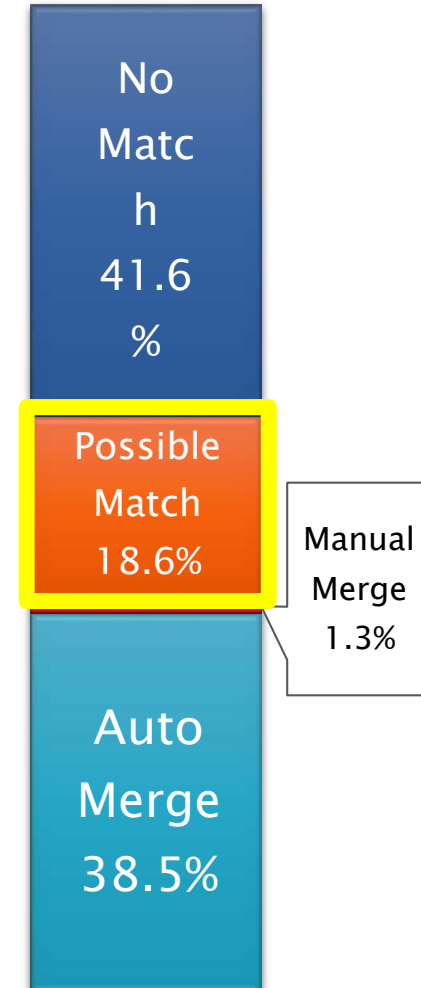
SSG
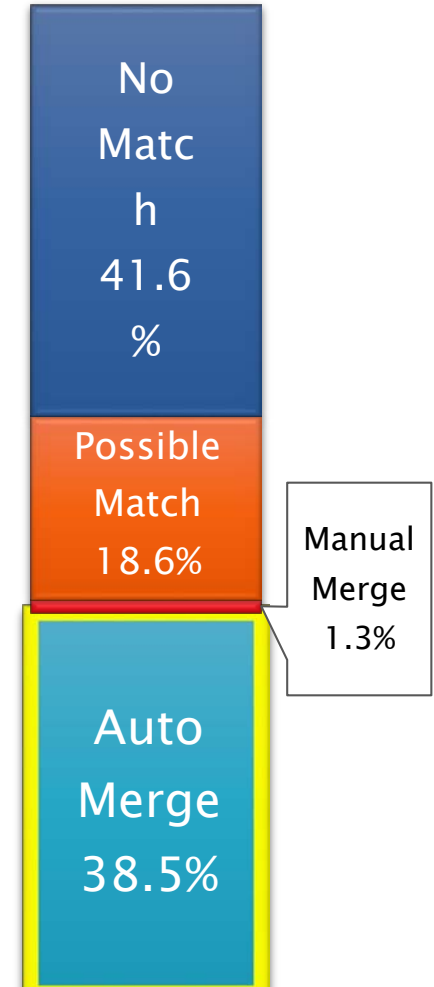
Categories of Interest
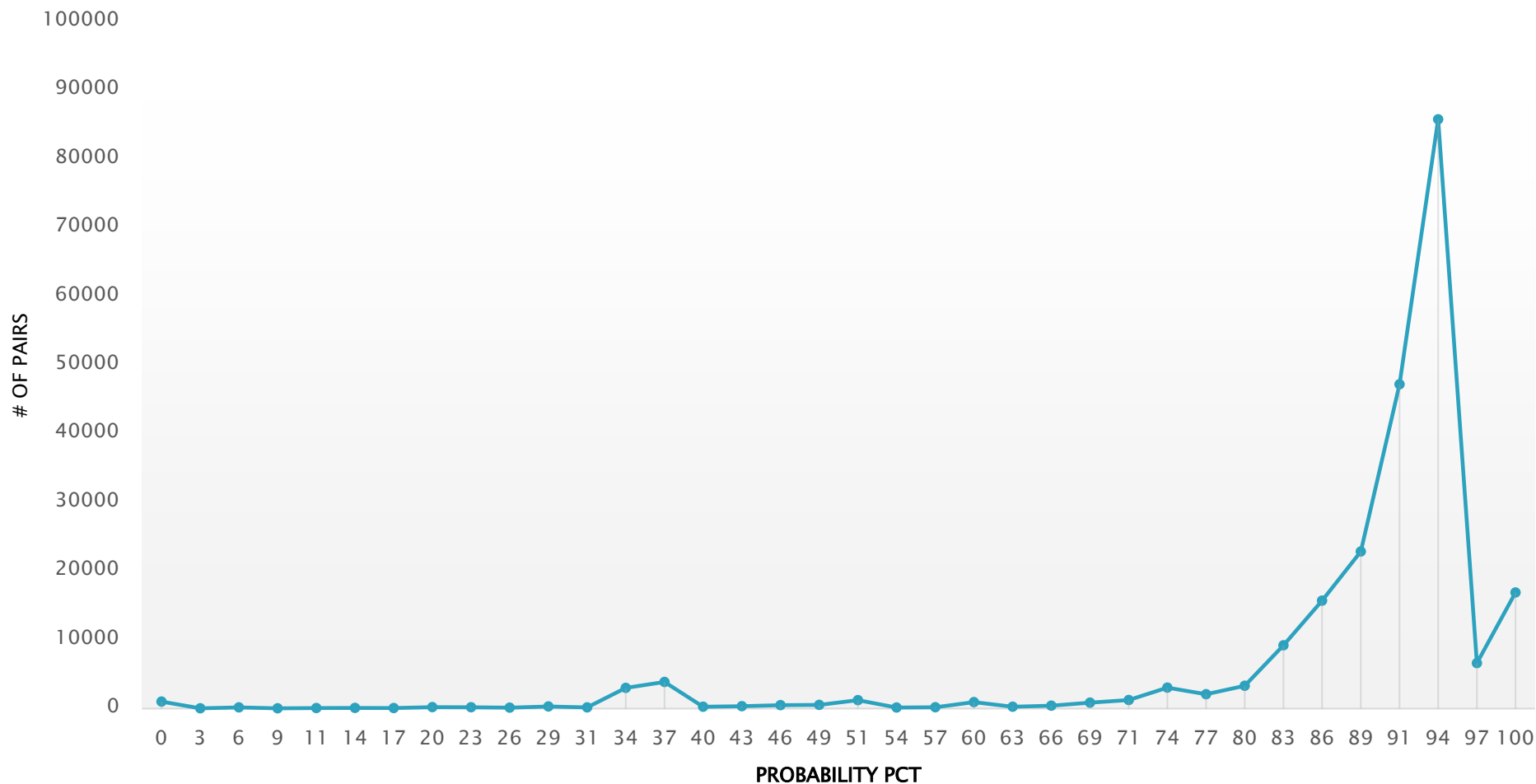
1) Auto Merges with low probability score
2) Manual Merges with low probability score
3) Possible matches on the manual resolution queue with a low probability score
4) Possible matches on the manual resolution queue with a high probability score that could auto merge.

Patients in MIIS
By Category

No Match
41.6 %

Possible Match
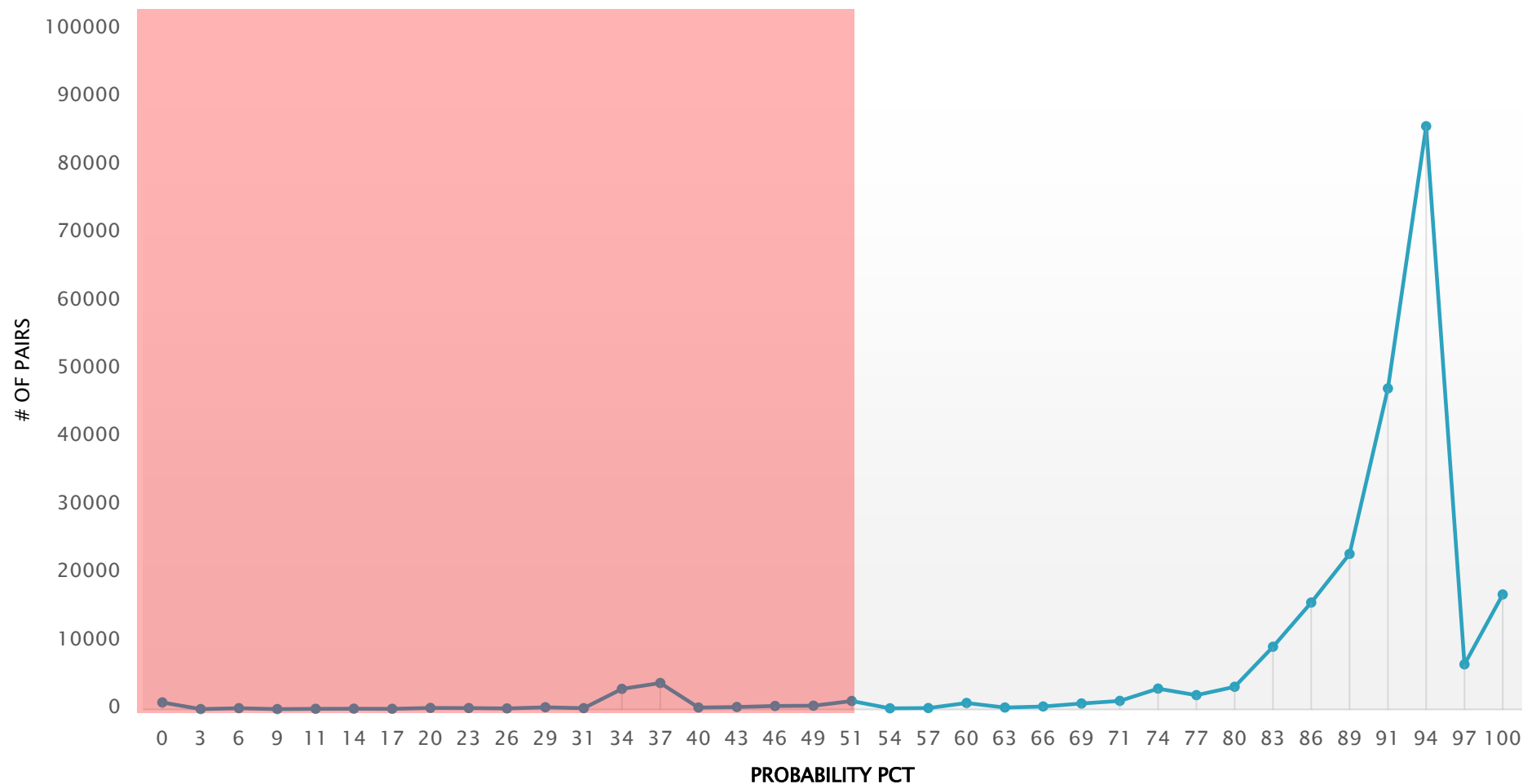18.6%

Manual Merge
1.3%

Auto Merge
38.5%

Probabilistic Analysis of Auto Merges

Lantern Determined Probability of
Pairs that Auto Merged

Probabilistic Analysis of Auto Merges

Lantern Determined Probability of
Pairs that Auto Merged

Merges with a low score

Auto Merges:

▸ Only 65 of the 200k auto merges had a score under 50%

▸ All 65 occurred before the system was re-trained with production data.

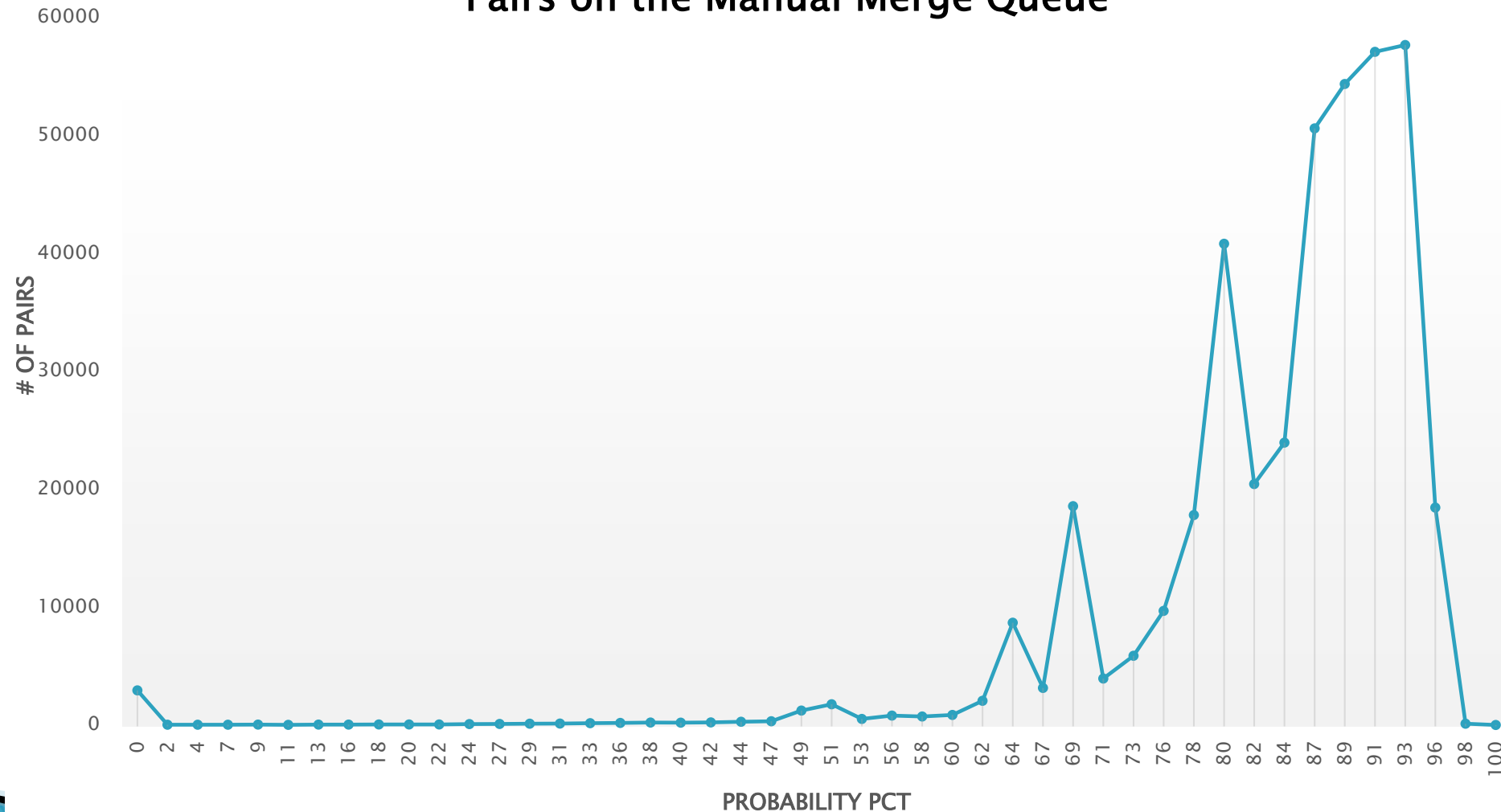**Manual Merges:**

▸ Just 5 of 18k manual merges were deemed to be incorrect.

▸ The underlying issue here is that the records were put on the queue.

SSG

Lantern Determined Probability of
Pairs on the Manual Merge Queue

Probabilistic Analysis of Manual Merge Queue

Lantern Determined Probability of
Pairs on the Manual Merge Queue
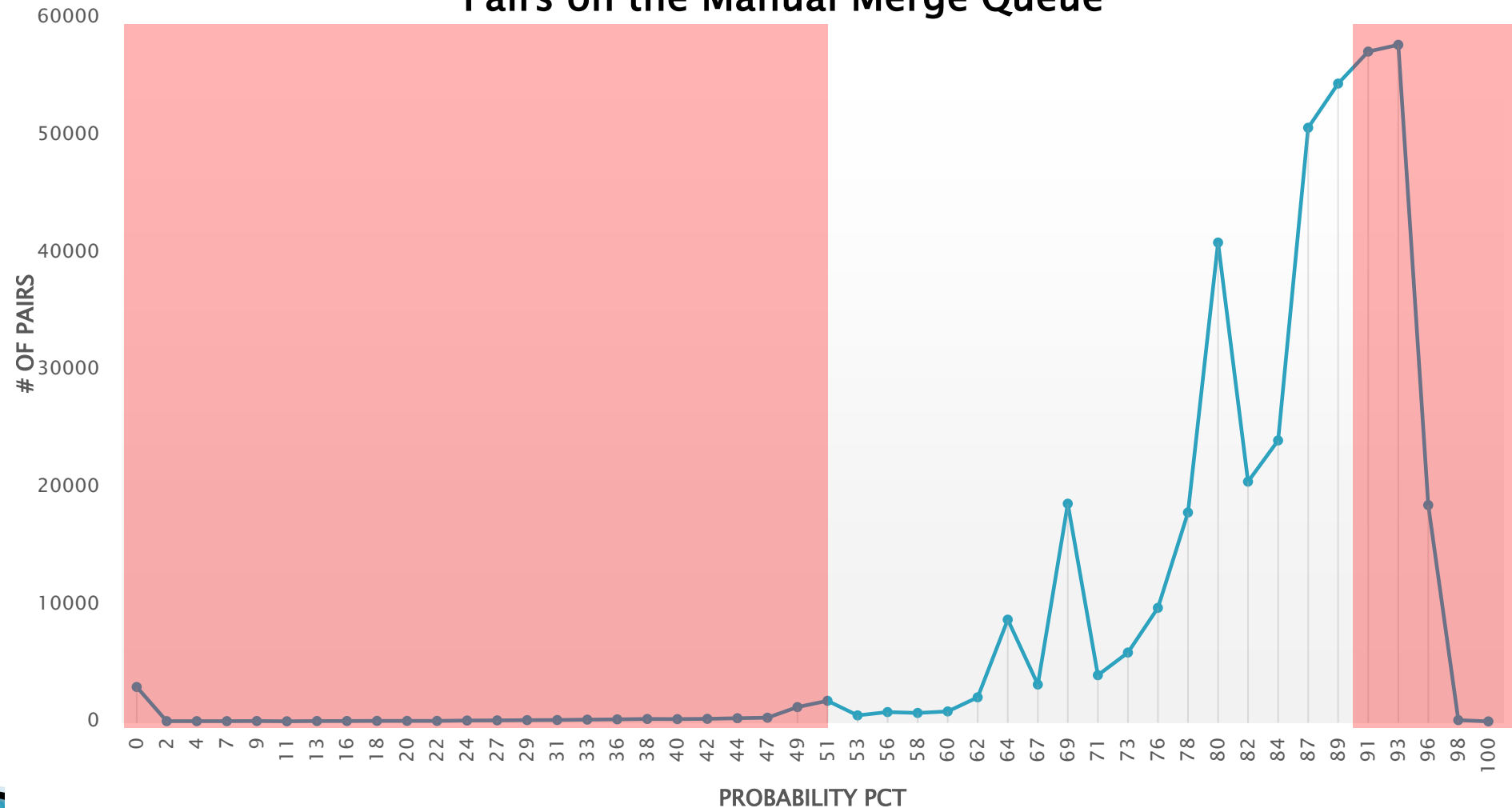
Possible matches with a low score

- Over half of all patients with low probability had at least one patient with a hyphenated last name
- Other fields on the records were not an exact match either, resulting in a low probability from our tool.
- A deeper investigation is needed.

Patients with < 30% Probability Marked as a Possible Match

Last Name Includes Hyphen 55%

Last Name Does Not Include Hyphen 45%
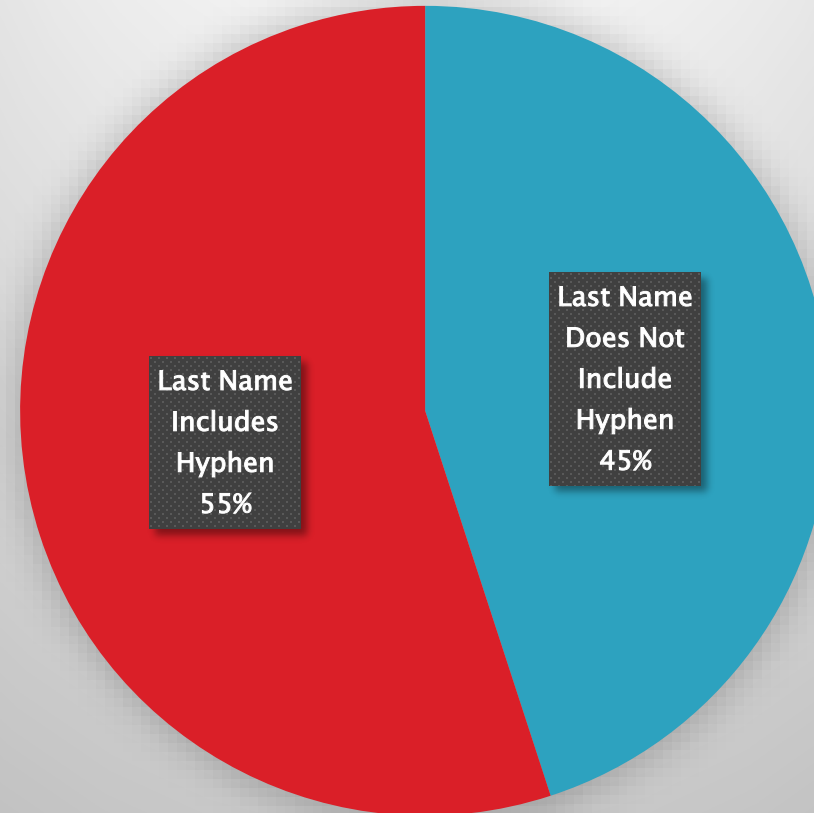
Individual Case Analysis

Case # 1: "The Hyphen Issue"

- MIIS Given Result: Possible Match
- Preferred Result: Not a Match

Field	Patient A	Patient B
Last Name	Sample-One	Single
Middle Name	Middy	
First Name	Sam	Seth
Gender	Male	Male
Date of Birth	01/01/2001	01/01/2001
Street Address	4 Oak Street	2 Maple Street
City	Boston	Worchester
State	MA	MA

Case # 1: "The Hyphen Issue"

- A code review was necessary to determine why patients with hyphenated last names were being marked as a possible match.
- We quickly isolated one section of the code that runs in the dedup engine.

```java
public double score(Patient patientA, Patient patientB) {
    String a = patientA.getValue(fieldName);
    String b = patientB.getValue(fieldName);
    if (a.contains("-") || b.contains("-")) {
        String[] partsA = a.split("\\-");
        String[] partsB = a.split("\\-");
        for (String partA : partsA) {
            for (String partB : partsB) {
```

Individual case analysis

Case # 1: "The Hyphen Issue" – SOLVED

▸ A bug was found where we parse hyphenated names and compare each piece to the other patient record.

```java
public double score(Patient patientA, Patient patientB) {
    String a = patientA.getValue(fieldName);
    String b = patientB.getValue(fieldName);
    if (a.contains("-") || b.contains("-")) {
        String[] partsA = a.split("\\-");
        String[] partsB = a.split("\\-");
        for (String partA : partsA) {
            for (String partB : partsB) {
```

▸ Fixing this bug would result in a 4% reduction in the manual review queue.

Possible Matches with a high score > 90 %

DOB Match, Last or First Not Matching
7%

DOB,L,F,G Match; Address given & Not Matching
11%

DOB,L,F,G Match Address Not Matching
84%

DOB,Last,First Match; Gender Not Matching
9%

DOB,L,F,G Match; Address Blank
73%

- ■ DOB,Last,First Match; Gender Not Matching
- ■ DOB Match, Last or First Not Matching
- ■ DOB, L, F, G Match; Address given & Not Matching
- ■ DOB, L, F, G Match; Address Blank
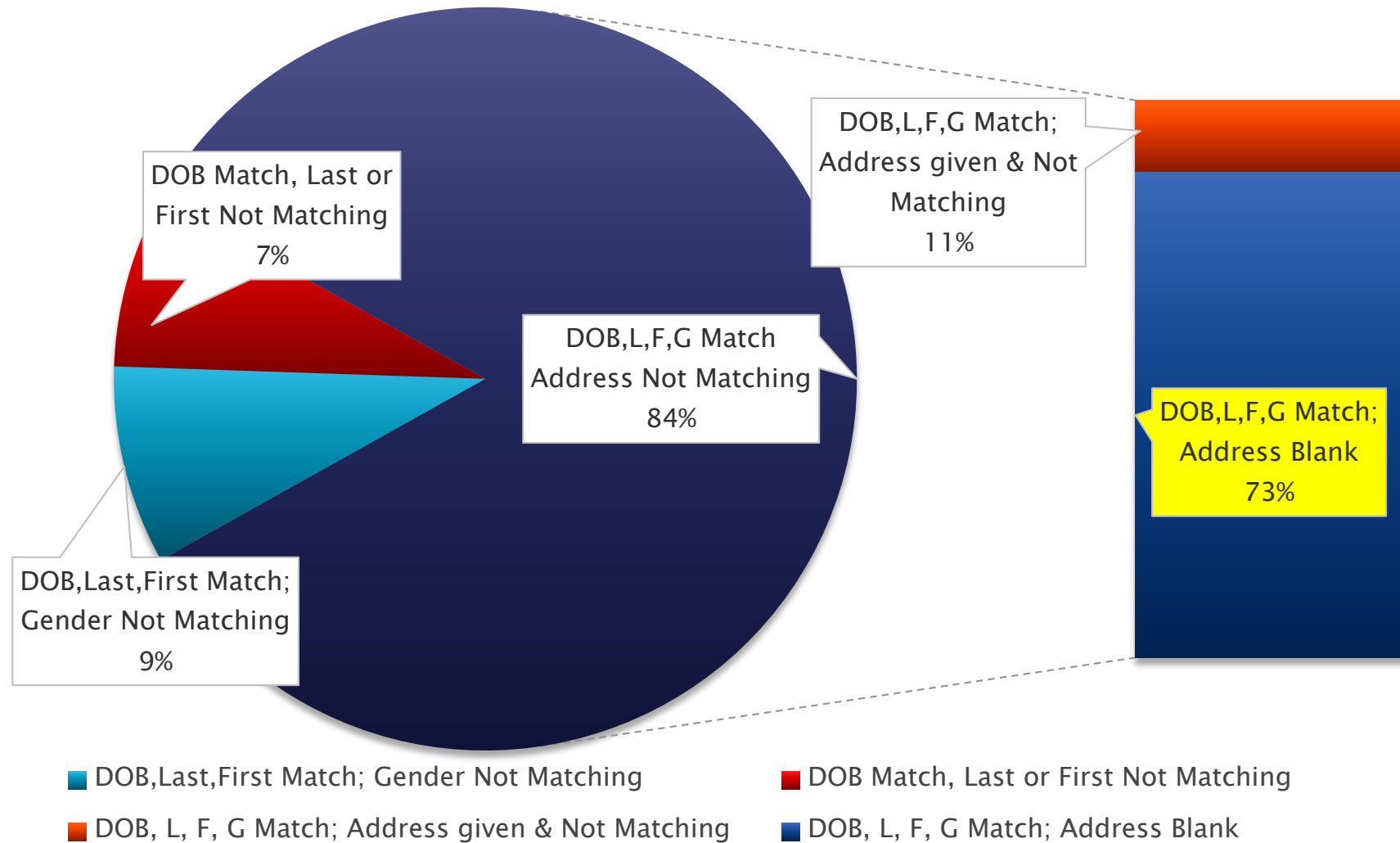
Individual case analysis

Case # 2: "The missing address issue"

▸ MIIS Given Result: Possible Match
▸ Preferred Result: Exact Match

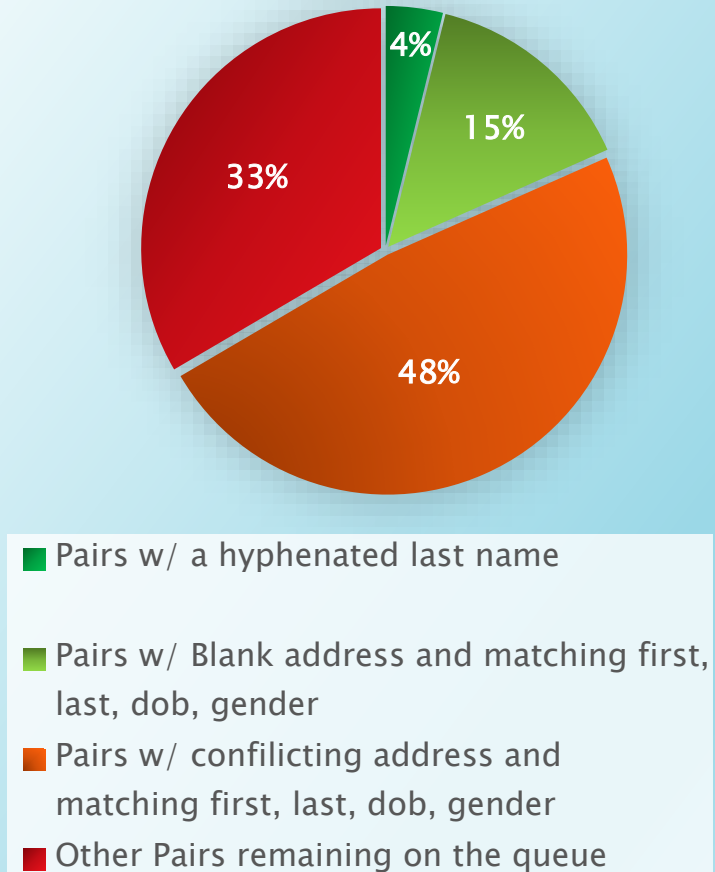| Field	Patient A	Patient B
Last Name	Sample	Sample
Middle Name	Mid	M
First Name	Sam	Sam
Gender	Male	Male
Date of Birth	01/01/2001	01/01/2001
Street Address	4 Oak Street	
City	Boston	
State	MA	

Individual case analysis

Case # 2: "The missing address issue"

▸ This case was not as clear cut as the hyphen issue, there was no bug discovered in the code.

▸ The way the MIIS Deduplication algorithm is trained, there is a level of skepticism if addresses do not match, this includes blank addresses.

▸ This issue can be resolved through retraining of the MIIS system to "teach" the algorithm that a comparison to a blank address should not deduct from the matching probability.

▸ With retraining, we expect the manual review queue will be reduced by 15%.

How much can we reduce the queue

- Our bug fix for the hyphen would reduce the size of the queue by 4%.
- By matching patients with blank addresses and the same First, Last, Gender and DOB we can reduce the total queue by an additional 15%.
- We have also identified that 48% of the queue matches on First, Last, DOB and Gender but has a conflicting address. Retraining for this we could reduce the total queue by 67%

Manual Merge Queue

4%
15%
33%
48%

- Pairs w/ a hyphenated last name
- Pairs w/ Blank address and matching first, last, dob, gender
- Pairs w/ confilicting address and matching first, last, dob, gender
- Other Pairs remaining on the queue

What did we learn?

- By utilizing the Lantern Person Matching Tool, we were able to quickly identify specific areas we needed to improve. This was done without the painstaking task of a manual review of a large amount of data.

- We have identified ways to reduce the size of the queue by **67%, removing 272,858 pairs.**

- Our tool identified 70 merges were incorrect and needed a data fix, avoiding critical downstream errors for these patients.

- Remember our 3 hypothetical employees going through the queue by hand?
  - We were able to identify ways to remove 67% of the queue.
  - To complete this by hand would have cost approx. $140K
  - We were able to execute this analysis to efficiently reduce the size of the queue with a cost savings of over $125k

How can this technology help you?

1. Stand alone tool can be plugged into any database to both learn from the data set and analyze the data
   a) More advanced training: incorporating data from other IIS registries to use a larger training set will make the tool more detailed and accurate.
   b) AI Technology likes larger data sets, the more data the better!!

Thank You

∴∴SSG

Questions?
Comments?
Suggestions?

Sam Belton
Strategic Solutions Group
sbelton@ssg-llc.com
MIIS IT Project Team
Production Support