Target Detection Fix: Search All CSV Rows Including Pre-Header Rows

🎯 Problem Statement

Issue: Target detection failed when the target row appeared BEFORE the header row in the CSV file.

Scenario

In `data (3).csv`:
- **Row 0**: `"TARGET is Hawthorns, Horncastle Road, Goulceby, LOUTH  LN11 9WB"` (TARGET row)
- **Row 1**: Empty row
- **Row 2**: `Date,Address,Postcode,Type,Tenure,...` (HEADER row)
- **Row 3+**: Data rows

Root Cause

1. CSV parser's `detectHeaders()` correctly identified row 2 as the header
2. `normalizeData()` started processing from `headerRowIndex + 1` (row 3), skipping rows 0-2
3. Target row (row 0) was never parsed into the properties array
4. `findTarget()` searched only the properties array and failed to find the target

✅ Solution

Modified the CSV parsing and target detection flow to capture and search ALL rows including pre-header rows.

Architecture Changes

1. CSV Parser (`src/utils/csvParser.js`)

- **Modified `parseCSV()`** to return an object with:
- `normalizedData`: Array of normalized property objects (post-header rows)
- `preHeaderRows`: Array of raw row arrays (pre-header rows)
- `headerMapping`: Column index to standard header mapping

- **Added `normalizePreHeaderRow(row)`** function:

- Converts raw CSV row array to normalized property object
- Detects target indicators using regex patterns:
  - `"TARGET is ..."`
  - `"TARGET = ..."`
  - `"target: ..."`
  - etc.
- Extracts address and postcode from target indicator strings
- Sets `isTarget = 1` flag when target indicator found

2. Target Finder (`src/utils/targetFinder.js`)

- **Modified `findTarget(properties, preHeaderRows)`** :
- Now accepts optional `preHeaderRows` parameter
- **Step 1**: Search pre-header rows FIRST
    - Normalizes each raw row using `normalizePreHeaderRow()`
    - Checks for target indicators
    - Marks candidates with `source: 'pre-header'`
- **Step 2**: Search normalized properties
    - Uses existing target detection logic
    - Marks candidates with `source: 'normalized'`
- **Removal Logic**: Only removes target from properties array if `source === 'normalized'`

3. Main Script (`src/main.js`)

- Updated to handle new `parseCSV()` return format:

  ```
  javascript
    const parseResult = parseCSV(csvContent);
    const properties = parseResult.normalizedData;
    const preHeaderRows = parseResult.preHeaderRows;
  ```

- Passes `preHeaderRows` to `findTarget()` :

  ```
  javascript
    const { target, comparables } = findTarget(properties, preHeaderRows);
  ```

🧪 Test Results

Tested with 3 CSV files to ensure both scenarios work:

Test 1: `data (3).csv` - Target BEFORE Header ✅

```
Row 0: TARGET is Hawthorns, Horncastle Road, Goulceby, LOUTH  LN11 9WB
Row 1: (empty)
Row 2: Date,Address,Postcode,... (HEADER)
Row 3+: Data rows

Result:
✓ Pre-header rows: 2
✓ Target found from pre-header at index -1
✓ Address: "Hawthorns, Horncastle Road, Goulceby, LOUTH"
✓ Postcode: "LN11 9WB"
✓ Comparables: 50 (properties array unchanged)
```

Test 2: `output (22).csv` - Target AFTER Header ✅

```
Row 0: Date of sale,Address,Postcode,... (HEADER)
Row 1: EPC Lookup row
Row 2: 7 Fernbank Close,DN9 3PT,,,,,,,,,,,,,,,,,,,,1,, (TARGET with isTarget=1)
Row 3+: Data rows

Result:
✓ Pre-header rows: 0
✓ Target found from normalized at index 1
✓ Address: "7 Fernbank Close"
✓ Postcode: "DN9 3PT"
✓ Comparables: 19 (target removed from properties)
```

Test 3: `output (23).csv` - Target AFTER Header ✅

```
Row 0: Date of sale,Address,Postcode,... (HEADER)
Row 1: EPC Lookup row
Row 2: 7 Fernbank Close,DN9 3PT,,,,,,,,,,,,,,,,,,,,1,, (TARGET with isTarget=1)
Row 3+: Data rows

Result:
✓ Pre-header rows: 0
✓ Target found from normalized at index 1
✓ Address: "7 Fernbank Close"
✓ Postcode: "DN9 3PT"
✓ Comparables: 19 (target removed from properties)
```

📋 Implementation Details

Pre-Header Row Normalization Logic

The `normalizePreHeaderRow()` function handles two target detection patterns:

Pattern 1: Target Indicator with Address

```
// Example: "TARGET is Hawthorns, Horncastle Road, Goulceby, LOUTH  LN11 9WB"
const TARGET_PATTERNS = [
    /target\s+is\s+(.+)/i,
    /target\s*=\s*(.+)/i,
    /target\s*:\s*(.+)/i,
    // ... more patterns
];
```

- Extracts address/postcode data after the target indicator
- Parses UK postcode using regex: `/\b([A-Z]{1,2}\d{1,2}[A-Z]?\s?\d[A-Z]{2})\b/i`
- Removes postcode from address field
- Sets `isTarget = 1`

Pattern 2: Structured Row with isTarget Flag

```
// Example: ["", "7 Fernbank Close", "DN9 3PT", ..., "1", ...]
```

- Detects postcode pattern in cells
- Detects address (comma-separated or multi-word strings)
- Detects `"1"` in late columns (isTarget flag position)
- Sets `isTarget = 1` if flag found

🔍 Code Changes Summary

Files Modified

1. ✅ `src/utils/csvParser.js` (104 lines added)
   - Modified `parseCSV()` return format
   - Added `normalizePreHeaderRow()` function
   - Exported `normalizePreHeaderRow` in module.exports

2. ✅ `src/utils/targetFinder.js` (29 lines modified)
   - Added `preHeaderRows` parameter to `findTarget()`
   - Added pre-header row search logic (Step 1)
   - Added source tracking (`pre-header` vs `normalized`)
   - Modified removal logic to check source

3. ✅ `src/main.js` (12 lines modified)
   - Updated to destructure `parseResult` from `parseCSV()`
   - Extract `normalizedData` and `preHeaderRows`
   - Pass `preHeaderRows` to `findTarget()`

4. ✅ `test-target-detection-all-rows.js` (NEW)
   - Comprehensive test script for all scenarios
   - Tests 3 CSV files with different structures
   - Validates target detection and comparable count

🎉 Benefits

1. **Backwards Compatible**: Still works with CSVs where target is after header
2. **Robust**: Handles various target indicator formats (“TARGET is”, “TARGET =”, etc.)
3. **Smart Parsing**: Automatically extracts address and postcode from target strings
4. **Well-Tested**: All 3 test scenarios pass with comprehensive logging
5. **Clear Logging**: Detailed logs show which source (pre-header vs normalized) the target came from

---

📦 Commit Information

Branch: `fix/target-detection-all-rows`
Commit: `e8a0756`
Date: December 4, 2025

Pull Request

Create PR: https://github.com/CliveCaseley/soldcomp-analyser2/pull/new/fix/target-detection-all-rows

---

✅ Testing Instructions

Run the test script:

```
cd /home/ubuntu/github_repos/soldcomp-analyser2
node test-target-detection-all-rows.js
```

Expected output:

```
ALL TESTS COMPLETED
✅ TEST 1 PASSED (data (3).csv)
✅ TEST 2 PASSED (output (22).csv)
✅ TEST 3 PASSED (output (23).csv)
```

📝 Version

Fix Version: v2.3
Description: Pre-Header Row Target Detection

🔗 Related Issues

- Resolves: Target detection fails when target row appears before header row
- Related to: Smart Header Detection (v2.2) - builds on header detection improvements