Data Clone Detection and Visualization in Spreadsheets
written by Hermans, Felienne, Ben Sedee, Martin Pinzger, and Arie van Deursen
Very broad context: Spreadsheet errors are incredibly common
A vigilant reader, who combed through the backup of today’s Consumer Credit G.19 statement points out a flagrant and obvious error in the Fed’s data. While luckily the data impact is not major (at most $4 billion, which in our day and age is a pithy 50% of Goldman’s FICC trading desk bonus), the implication that the Fed does not check its work in something as critical as one of the core data series (or at least it used to be until a few machines took over the market, to whom, as today indicated, a record credit contraction somehow ended up being a positive event) is very, very troubling.

The original, Fed-hosted excel file with the backing data of the actual G.19 statement can be found here. We welcome all readers to compare cells AC 804 through AC 809, which is the data for "Consumer Revolving Credit Owned by Nonfinancial Business, Not Seasonally Adjusted" for the months June through November of 2009, and to compare it with data in cells AC 792 through AC 797, which is comparable data for the months June through November of 2008. These are identical and very much wrong! So, dear Fed auditors, while you obviously are very highly overpaid for your error-proofing work, can you please tell us what the real Consumer Credit number for November is?
TransAlta Says Clerical Snafu Costs It $24 Million

Tuesday, June 3, 2003 19:06 EDT

By Cameron French

TORONTO (Reuters) - TransAlta Corp. said on Tuesday it will take a $24 million charge to earnings after a bidding snafu landed it more U.S. power transmission hedging contracts than it bargained for, at higher prices than it wanted to pay.

TransAlta, Canada's top investor-owned power generator, said it submitted the erroneous bid to the New York Independent System Operator for May 2003 transmission congestion contracts. The ISO manages the state's power transmission system and the contracts hedge the cost of transmission.

But the company's computer spreadsheet contained mismatched bids for the contracts, it said.

"It was literally a cut-and-paste error in an Excel spreadsheet that we did not detect when we did our final sorting and ranking bids prior to submission," TransAlta chief executive Steve Snyder said in a conference call.

"I am clearly disappointed over this event. The important thing is to learn from it, which we've done."

As New York ISO rules did not allow for a reversal of the bids, the contracts went ahead.

The pretax charge, to be taken in the second quarter, translates into 11 Canadian cents a share. The average earnings estimate for TransAlta among analysts polled by First Call is 16 cents a share.
One major problem...
... in the context of spreadsheets

... think about formula(s|e)
The ideal solution

How to Link Cells in Different Excel Spreadsheets

Source File  Source Sheet  Source Cell
...which is rather hard to implement.
This paper's alternative -

Use the *force* to solve errors in spreadsheets.
This paper's alternative -

On a serious note...

Data clone detection and visualization
Related work

Clone detection in CODE
Random fact

Paper on clones and spreadsheets accepted into the “Tiny transactions on computer science”, v1, so if you don't have the patience to read the whole thing, you can read this version, where the abstract is bigger than the body.
Research questions

1. How often do data clones occur in spreadsheets?

2. What is the impact of data clones on spreadsheet quality?

3. Does our approach to detect and visualize data clones in spreadsheets support users in finding and understanding data clones?
Some specific terminology...

“Clone” - tuples of a formula cell and a constant cell that contain the same value. There IS a possibility of coincidence when it comes to clone “detection”

“Clone cluster” - a group of neighboring cells that are all “clones”

*Near miss* clone clusters – clusters that largely have clones, but also have cells that differ from each other.
Five Step Algorithm

1. Cell Classification
2. Lookup Creation
3. Pruning
4. Cluster Finding
5. Cluster Matching

Parameters

• Step Size
• Match Percentage
• Minimal Cluster Size
• Minimal Different Values
Step 1: Cell Classification

- Only cells with numbers
- Orange = data cells (constants)
- Pink = formula cells
Step 2: Lookup Creation

- Inverted index of cell values and locations
  - 0.4101 -> {Eff4!B28}
  - 0.1156 -> {Eff4!D25, Problem Data!B10}
Step 3: Pruning

- Remove cells that do not have both a formula and constant entry

- 0.4101 -> {sheet2!B28}
- 0.1156 -> {sheet2!D25, sheet1!B10}
Step 4: Cluster Finding

- Examine neighboring cells
- Uses **StepSize** parameter – how close neighbors have to be
- If in the lookup table – expand the cluster
- Result: list of formula clusters and constant clusters
- **MinimalClusterSize** parameter defines size. Small clusters may be “uninteresting”
Step 5: Cluster Matching

- Formula clusters are matched with constant clusters
- Uses \textbf{MatchPercentage} parameter – allows for near-miss clones
- \textbf{MinimalDifferentValues} parameter – clusters containing just one or two
Visualization

- In sheet
- Data Flow Diagram
  - Dashed arrow indicates a clone dependency
  - Solid arrow indicates a formula dependency

### Covariance Matrix

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>std</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>stdv</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.34</td>
<td>1</td>
<td>0.1156</td>
<td>0.032</td>
<td>0.0384</td>
</tr>
<tr>
<td>0.3606</td>
<td>2</td>
<td>0.032</td>
<td>0.13</td>
<td>0.048</td>
</tr>
<tr>
<td>0.3842</td>
<td>3</td>
<td>0.0384</td>
<td>0.048</td>
<td>0.1476</td>
</tr>
<tr>
<td>0.4104</td>
<td>4</td>
<td>0.0448</td>
<td>0.056</td>
<td>0.0672</td>
</tr>
</tbody>
</table>

The source of these values is Eff4!C25xF28

### Variance/Covariance Matrix:

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>std</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>stdv</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>A</td>
<td>0.1156</td>
<td>0.032</td>
<td>0.0384</td>
<td>0.0448</td>
</tr>
<tr>
<td>B</td>
<td>0.0320</td>
<td>0.1300</td>
<td>0.0480</td>
<td>0.0560</td>
</tr>
<tr>
<td>C</td>
<td>0.0384</td>
<td>0.0480</td>
<td>0.1476</td>
<td>0.0672</td>
</tr>
<tr>
<td>D</td>
<td>0.0448</td>
<td>0.0560</td>
<td>0.0672</td>
<td>0.1684</td>
</tr>
</tbody>
</table>
Evaluation

• Quantitative analysis
  • How often do data clones occur in spreadsheets? (R1)

• Qualitative analysis
  • Do detected data clones actually pose a threat to spreadsheet quality?
Quantitative Analysis – Overview

• Goal
  • How often do data clones occur in spreadsheets? (R1)
  • First evaluation of the algorithms’ performance
  • Find good parameters

• Evaluation idea
  • Take sample spreadsheets and analyze with algorithm
  • Calculate precision (no recall calculated)
Quantitative Analysis – Method

• Dataset
  • EUSES corpus (real-life spreadsheets)
  • Those 1711 spreadsheets, that contained formulas

• Precision
  • Determine false positives: Do headers indicate same conceptual data?
  • Calculate Precision
    \[
    \frac{\text{# of spreadsheets with verified clones}}{\text{# of spreadsheets with found clones}}
    \]
Quantitative Analysis – Method

• Dataset
  • EUSES corpus (real-life spreadsheets)
  • Those 1711 spreadsheets, that contained formulas

• Precision
  • Determine false positives: Do headers indicate same conceptual data?
  • Calculate Precision
    \% of spreadsheets with verified clones/# of spreadsheets with found clones = #
Quantitative Analysis – Findings

• Frequency of clones
  • Around 5% of all spreadsheets contain clones

• Performance
  • 3 hours, 49 minutes to analyze all 1711 spreadsheets
  • 8.1 seconds per file

<table>
<thead>
<tr>
<th>Minimal Size</th>
<th>Minimal Different Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>86</td>
</tr>
<tr>
<td>6</td>
<td>77</td>
</tr>
<tr>
<td>7</td>
<td>70</td>
</tr>
<tr>
<td>8</td>
<td>64</td>
</tr>
<tr>
<td>9</td>
<td>60</td>
</tr>
<tr>
<td>10</td>
<td>54</td>
</tr>
<tr>
<td>11</td>
<td>49</td>
</tr>
<tr>
<td>12</td>
<td>47</td>
</tr>
<tr>
<td>13</td>
<td>41</td>
</tr>
</tbody>
</table>
Quantitative Analysis – Findings

- Parameters

<table>
<thead>
<tr>
<th>Minimal Size</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>54.8%</td>
<td>59.1%</td>
<td>63.7%</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>6</td>
<td>54.2%</td>
<td>59.2%</td>
<td>62.9%</td>
<td>70.1%</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>7</td>
<td>53.8%</td>
<td>59.1%</td>
<td>62.5%</td>
<td>69.5%</td>
<td>70.9%</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>8</td>
<td>56.1%</td>
<td>60.2%</td>
<td>63.6%</td>
<td>70.1%</td>
<td>71.6%</td>
<td>72.9%</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>9</td>
<td>56.6%</td>
<td>60.6%</td>
<td>64.3%</td>
<td>71.2%</td>
<td>72.9%</td>
<td>74.6%</td>
<td>81.7%</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>10</td>
<td>55.1%</td>
<td>58.6%</td>
<td>62.3%</td>
<td>69.7%</td>
<td>71.4%</td>
<td>73.3%</td>
<td>80%</td>
<td>79.2%</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>11</td>
<td>56.3%</td>
<td>57.7%</td>
<td>60.9%</td>
<td>68.3%</td>
<td>70.2%</td>
<td>71.4%</td>
<td>78.4%</td>
<td>77.6%</td>
<td>78.3%</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>12</td>
<td>56.6%</td>
<td>58.1%</td>
<td>60.6%</td>
<td>67.8%</td>
<td>69.6%</td>
<td>70.9%</td>
<td>78%</td>
<td>77.1%</td>
<td>77.8%</td>
<td>81%</td>
<td>-</td>
</tr>
<tr>
<td>13</td>
<td>56.9%</td>
<td>57.4%</td>
<td>61%</td>
<td>66.7%</td>
<td>69.2%</td>
<td>70.6%</td>
<td>76.6%</td>
<td>75.6%</td>
<td>76.7%</td>
<td>80%</td>
<td>80%</td>
</tr>
</tbody>
</table>
Qualitative Analysis – Overview

• Goal
  • What is the impact of data clones on spreadsheet quality? (R2)
  • Does the approach to detect and visualize data clones in spreadsheets support...

• Evaluation idea
  • Analyze spreadsheet with algorithm
  • Present results to spreadsheet owners and ask questions
Qualitative Analysis – Method

• Datasets
  • 31 spreadsheet files of a foodbank (with corrupt results)
  • 1 spreadsheet file for research project’s budget

• Questions
  • Is this a true positive?/Did you copy this data?
  • Did this clone lead to errors or problems?
  • Could this clone be replaced by a formula link?
Qualitative Analysis – Findings

• Performance
  • 3 hours, 10 minutes to analyze the 31 foodbank files (6 minutes per file)
  • 145 clones (61 of them near-miss clones), only 1 false positive (< 1%)

• 3 seconds to analyze the budget file
• 8 exact clones, no false positives
Qualitative Analysis – Findings

• Impact of clones
  • In the foodbank files, near-misses were data actually should have matched
  • Causing results to be corrupt
  • Clones did not lead to errors in the case of budget file

• Impact of analysis
  • Foodbank: Restore correctness of results
  • Increase awareness of clones’ impact
Questions

• How else can this tool be used besides finding clones?
• By seeing near-misses, wrong data can be easily found and allow the user to correct data.

• Would this method have prevented the "copy and paste" errors cited in the introduction for TransAlta (24M) and the Federal Reserve (4B)?
• I'm skeptical. There is no evidence that the source of the data were cells with formulas - which is the only way this tool would detect the clone.

• This method requires a lot of manual verification to confirm that cells are real clones. Can the need for a human be reduced?
• It is unclear that a human can ever be completely removed from verifying clones. The authors point out that using meta data such as headers and/or genealogy like source control could increase the confidence of a cell being a clone.
Questions

• Does this method work on a Google doc spreadsheet or OpenOffice?
• Currently the code appears to be specific to Microsoft Excel, however the algorithm should be able to be ported to work on any spreadsheet.

• Is the performance acceptable?
• Run time depends on the complexity of the spreadsheets, some took as long as 3+ hours while others took just seconds. However the manual inspection is where most of the time is required.

• Could this tool be expanded to actually create links once a user verifies that a cell is a true clone?
• I see no reason why it could not. It seems that would be a great enhancement and help save the end user from themselves.