

Improving Spectrum-based Fault Localization for Spreadsheet Debugging

Elisabeth Getzner, Birgit Hofer, and Franz Wotawa
Graz University of Technology
Austria



Spreadsheets Errors???



JPMorganChase 

~ 300 million EUR damage

Spreadsheets Errors???

Paper published by
Reinhart & Rogoff

HARVARD
UNIVERSITY



Many governments
used it for decisions

Immense loss of
reputation

	A	B	I	J	K	L	M
2			Real GDP growth				
3			Debt/GDP				
4	Country	Coverage	30 or less	30 to 60	60 to 90	90 or above	30 or less
26			3,7	3,0	3,5	1,7	5,5
27	Minimum		1,6	0,3	1,3	-1,8	0,8
28	Maximum		5,4	4,9	10,2	3,6	13,3
29							
30	US	1946-2009	n.a.	3,4	3,3	-2,0	n.a.
31	UK	1946-2009	n.a.	2,4	2,5	2,4	n.a.
32	Sweden	1946-2009	3,6	2,9	2,7	n.a.	6,3
33	Spain	1946-2009	1,5	3,4	4,2	n.a.	9,9
34	Portugal	1952-2009	4,8	2,5	0,3	n.a.	7,9
35	New Zealand	1948-2009	2,5	2,9	3,9	-7,9	2,6
36	Netherlands	1956-2009	4,1	2,7	1,1	n.a.	6,4
37	Norway	1947-2009	3,4	5,1	n.a.	n.a.	5,4
38	Japan	1946-2009	7,0	4,0	1,0	0,7	7,0
39	Italy	1951-2009	5,4	2,1	1,8	1,0	5,6
40	Ireland	1948-2009	4,4	4,5	4,0	2,4	2,9
41	Greece	1970-2009	4,0	0,3	2,7	2,9	13,3
42	Germany	1946-2009	3,9	0,9	n.a.	n.a.	3,2
43	France	1946-2022	4,9	2,7	3,0	n.a.	5,2
44	Finland	1946-2023	3,8	2,4	5,5	n.a.	7,0
45	Denmark	1946-2024	3,5	1,7	2,4	n.a.	5,6
46	Canada	1946-2025	1,9	3,6	4,1	n.a.	2,2
47	Belgium	1946-2026	n.a.	4,2	3,1	2,6	n.a.
48	Austria	1946-2027	5,2	3,3	-3,8	n.a.	5,7
49	Australia	1946-2028	3,2	4,9	4,0	n.a.	5,9
50							
51			4,1	2,8	2,8	=AVERAGE(L30:L44)	
52							

Spectrum-based Fault Localization

- Suspiciousness of cell
Involvement correct / incorrect output
- + easy to use
- + low computation time
- many cells have the same ranking

Running example

	A	B	C	D	E
1		Hours	Salary	Bonus	Sum
2	Jones	17	272	26	298
3	Smith	13	208	0	208
4	Rogers	20	320	40	360
5	Total		800	66	866

	A	B	C	D	E
1		Hours	Salary	Bonus	Sum
2	Jones	17	=B2*16	=IF(B2>15; C3 /8;0)	=SUM(C2:D2)
3	Smith	13	=B3*16	=IF(B3>15; C3 /8;0)	=SUM(C3:D3)
4	Rogers	20	=B4*16	=IF(B4>15; C4 /8;0)	=SUM(C4:D4)
5	Total		=SUM(C2:C4)	=SUM(D2:D4)	=SUM(E2:E4)

$$\text{CONE}(c) = c \cup \bigcup_{c' \in \rho(c)} \text{CONE}(c')$$

$\text{CONE}(D2) = \{D2, B2, C3, B3\}$
 $\text{CONE}(D3) = \{D2, B3, C3\}$

Observation Matrix

$\text{CONE}(D2) = \{D2, B2, C3, B3\}$

$\text{CONE}(D3) = \{D2, B3, C3\}$



Cell	D2	D3	SC	Rank
B2	•		1.00	1
B3	•	•	0.71	3
C2			-	-
C3	•	•	0.71	3
D2	•		1.0	1
D3		•	-	-
Test verdict	false	true		

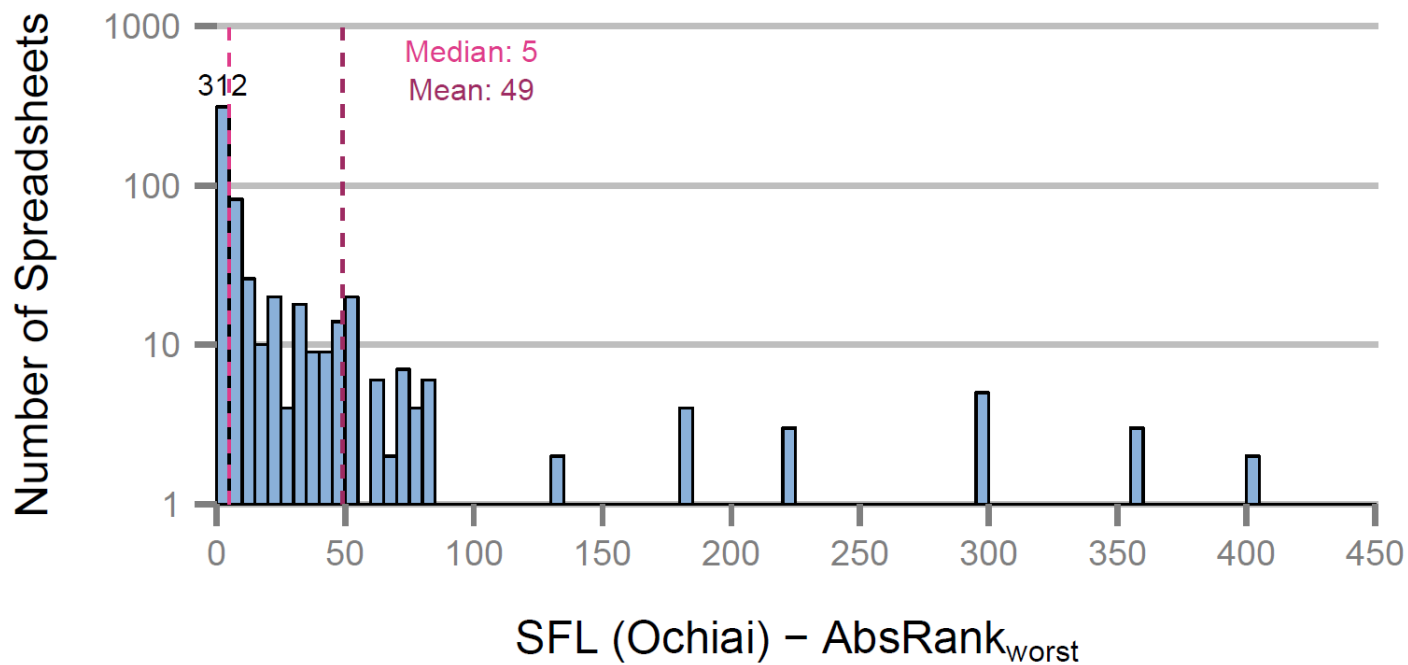
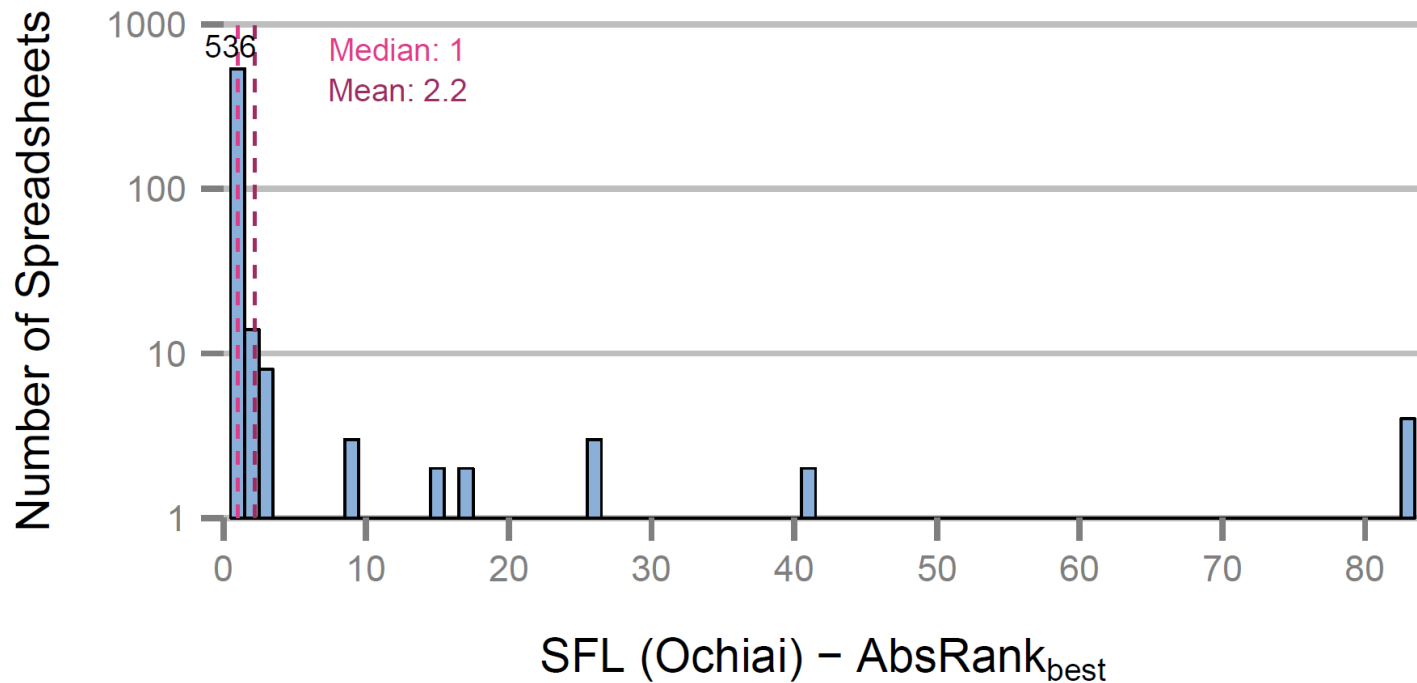
Ochiai Similarity Coefficient

$$SC_O(c) = \frac{a_{11}(c)}{\sqrt{(a_{11}(c) + a_{10}(c))(a_{11}(c) + a_{01}(c))}}$$

$$a_{11}(c) = |\{(c', d) \in O | c \in \text{CONE}(c') \wedge d = \text{false}\}|$$

$$a_{10}(c) = |\{(c', d) \in O | c \in \text{CONE}(c') \wedge d = \text{true}\}|$$

$$a_{01}(c) = |\{(c', d) \in O | c \notin \text{CONE}(c') \wedge d = \text{false}\}|$$



Our improvements

- Reduce the search space
 - Grouping
 - Dynamic slicing

Change ranking
- Prioritize cells
 - Tie breaking

Finer prioritization

Dynamic Cones

$$e = \underline{\text{IF}}(e_1, e_2, e_3)$$

$$\hat{\rho}(e) = \hat{\rho}(e_1) \cup \begin{cases} \hat{\rho}(e_2) & \text{if } \llbracket e_1 \rrbracket = \text{true} \\ \hat{\rho}(e_3) & \text{if } \llbracket e_1 \rrbracket = \text{false} \\ \emptyset & \text{otherwise} \end{cases}$$

$$\text{DYNAMICCONE}(c) = c \cup \bigcup_{c' \in \hat{\rho}(\ell(c))} \text{DYNAMICCONE}(c')$$

	A	B	C	D	E
1		Hours	Salary	Bonus	Sum
2	Jones	17	=B2*16	=IF(B2>15; C3 /8;0)	=SUM(C2:D2)
3	Smith	13	=B3*16	=IF(B3>15; C3 /8;0)	=SUM(C3:D3)
4	Rogers	20	=B4*16	=IF(B4>15; C4 /8;0)	=SUM(C4:D4)
5	Total		=SUM(C2:C4)	=SUM(D2:D4)	=SUM(E2:E4)

CONE(D2) = {D2, B2, C3, B3}

CONE(D3) = {D2, B3, C3}



DynCONE(D2) = {D2, B2, C3, B3}

DynCONE(D3) = {B3, D3}

Dynamic Cones Example

$\text{CONE}(D2) = \{D2, B2, C3, B3\}$
 $\text{CONE}(D3) = \{D2, B3, C3\}$

$\text{DynCONE}(D2) = \{D2, B2, C3, B3\}$
 $\text{DynCONE}(D3) = \{B3, D3\}$

Cell	D2	D3	SC	Rank
B2	•		1.00	1
B3	•	•	0.71	3
C2			-	-
C3	•	✗	0.71 1.00	3 1
D2	•		1.00	1
D3		•	-	-
Test verdict	false	true		

Grouping - Example

	A	B	C	D	E
1		Hours	Salary	Bonus	Sum
2	Jones	17	=B2*16	=IF(B2>15; C3 /8;0)	=SUM(C2:D2)
3	Smith	13	=B3*16	=IF(B3>15; C3 /8;0)	=SUM(C3:D3)
4	Rogers	20	=B4*16	=IF(B4>15; C4 /8;0)	=SUM(C4:D4)
5	Total		=SUM(C2:C4)	=SUM(D2:D4)	=SUM(E2:E4)

Grouping

Pre-Processing

- R1C1 ident formulas
- Neighbors
- Operate on same type of data
- Conflicting testing decisions

Post-Processing

- R1C1 ident formulas
- Neighbors
- Same similarity coefficient
- Ranking cannot be improved

Tie Breaking

- Work inspired by Xu, Debroy, Wong, and Guo:
„Ties within Fault Localization Rankings“

- Ties:
{B2, D2}, {B3,C3}

- Critical Tie:
{B2, D2}

Cell	D2	D3	SC	Rank
B2	●		1.00	1
B3	●	●	0.71	3
C2			-	-
C3	●	●	0.71	3
D2	●		1.0	1
D3		●	-	-
Test verdict	false	true		

- Position-based vs. Metric-based Strategies

Position-based Tie-breaking Strategies

- Cell Order Strategy (COS)
- Cell Distance Strategy (CDS)
- Path Length Strategy (PLS)

<i>Initial Ranking</i>	<i>COS</i>	<i>CDS</i>	<i>PLS</i>
1. B2, D2	1. D2	1. D2	1. D2
2. B3, C3	2. B2	2. B2	2. B2
3. C2, D3	3. C3	3. C3	3. C3
	4. B3	4. B3	4. B3
	5. D3	5. C2, D3	5. C2, D3
	6. C2		



Metric-based Tie-breaking Strategies

- Number of Operators (OP)
- Number of References (REF)
- Dispersion of References (DR)
- Calculation Chain: Cone Size (CS), Cone Level (CL)

<i>Initial Ranking</i>		<i>OP, REF</i>	<i>DR</i>	<i>CS, CL</i>
1. B2, D2		1. D2	1. D2	1. B2
2. B3, C3		2. B2	2. B2	2. D2
3. C2, D3		3. C3	3. B3, C3	3. B3
		4. B3	4. C2, D3	4. C3
		5. D3		5. C2
		6. C2		6. D3

Evaluation Basis

- EUSES
 - Large, diverse, well-known
 - 184 base spreadsheets, 576 faulty versions
- Payroll/Gradebook
 - Real test verdicts
 - 2 base spreadsheets, 349 faulty versions
- Info1
 - Real faults
 - 2 base spreadsheets, 119 faulty versions

Evaluation results

- Dynamic cones and Pre-processing
 - perform poorly
- Post-processing
 - improves ranking of faulty cell
- COS
 - is best tie breaker, but has the highest risk
- CDS
 - has a high tie reduction, and a positive impact
- DR
 - has a low tie reduction, but positive impact

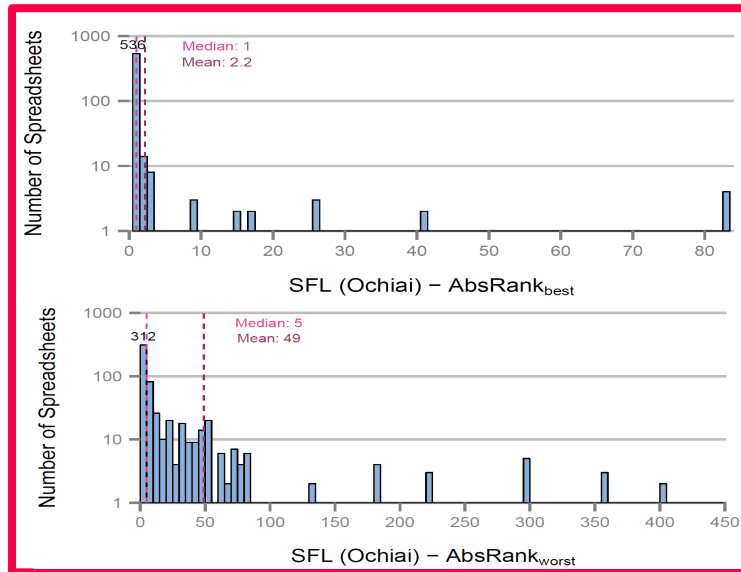
Summary

Spreadsheets Errors???



JPMorganChase 

~ 300 million EUR damage



Questions?

Our improvements

- Reduce the search space
 - Grouping
 - Dynamic slicing

Change ranking
- Prioritize cells
 - Tie breaking

Finer prioritization