

README: Dataset for Optimal Intensity Measure Selection for Jacket-Type Offshore Platforms

Overview

This dataset contains the primary Incremental Dynamic Analysis (IDA) results for a jacket-type offshore platform located in the Persian Gulf. The data were generated using OpenSees with a 3D finite element model including fiber-section beam-column elements and a beam-on-nonlinear-Winkler foundation (BNWF) for soil-pile-structure interaction.

The dataset includes 91 far-field ground motion records. For each record, the maximum inter-story drift ratio (MIDR) is reported across multiple scaling levels, along with the corresponding spectral acceleration at the fundamental period ($Sa(T1)$).

File Structure

- `IDA\_Data.csv` - Main dataset in wide format.
- `Optimal\_IM\_Selection\_Jacket\_Article.m` - MATLAB script for statistical analysis (optional supplement).

Data Dictionary (for `IDA\_Data.csv`)

The file is structured in **wide format**:

- **Rows**: Each row represents a specific **intensity level** (scaling factor). The first row after the header corresponds to the lowest intensity ($Sa(T1) \approx 0.025g$), and subsequent rows correspond to higher intensities.
- **Columns**: The columns are organized as pairs for each record (Rec 1 to Rec 91):
 - `RecN\_Drift` - Maximum inter-story drift ratio (MIDR) in percent (%) for record N at that intensity level.
 - `RecN\_Sa(T1)` - Spectral acceleration at the fundamental period (in g) for record N at that intensity level.

Note: Empty cells indicate non-convergence in the numerical analysis at that particular intensity level.

Methodology

The platform is a three-leg jacket structure originally built in the 1970s. The numerical model was developed in OpenSees. The fundamental period ($T1$) was obtained from eigenvalue analysis. For a complete description of the modeling assumptions, scaling procedures, and the full list of 57 candidate IMs studied, please refer to the associated publication (currently under review in *Ships and Offshore Structures*).

Contact

For questions regarding this dataset, please contact Amin Ghannadial at [ایمیل خود] را اینجا وارد کنید.

License

This dataset is distributed under the Creative Commons CC BY 4.0 license.

Rec 1		Rec 2		Rec 3		Rec 4		Rec 5
Drift	Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)	Drift
0.000986	0.025	0.00121	0.025	0.000825	0.025	0.000846	0.025	0.001157
0.0018	0.05	0.002184	0.05	0.001597	0.05	0.00163	0.05	0.002053
0.003475	0.1	0.004534	0.1	0.002903	0.1	0.003201	0.1	0.004141
0.005156	0.15	0.006816	0.15	0.004402	0.15	0.004804	0.15	0.006368
0.008424	0.225	0.010144	0.225	0.006191	0.225	0.006877	0.225	0.00911
0.011345	0.3	0.013576	0.3	0.008418	0.3	0.009057	0.3	0.012738
0.015639	0.4	0.017514	0.4	0.011292	0.4	0.012563	0.4	0.018044
0.019371	0.5	0.022274	0.5	0.014466	0.5	0.015718	0.5	0.023894
0.024446	0.625	0.028451	0.625	0.018148	0.625	0.019596	0.625	0.033492
0.02992	0.75	0.036049	0.75	0.022639	0.75	0.023996	0.75	0.045044
0.03796	0.9	0.045178	0.9	0.027799	0.9	0.029083	0.9	0.059396
0.046179	1.05	0.054928	1.05	0.03276	1.05	0.035501	1.05	0.073147
0.056485	1.225	0.067337	1.225	0.037891	1.225	0.044332	1.225	0.085145
0.068236	1.4	0.080277	1.4	0.045052	1.4	0.053656	1.4	0.092726
0.084108	1.6	0.089897	1.533333	0.055827	1.6	0.065771	1.6	0.098282
0.099201	1.8	0.096491	1.622222	0.068217	1.8	0.078481	1.8	0.099616
0.100275	1.816667	0.097884	1.641975	0.078601	1.95	0.0889	1.95	0.100506
0.102148	1.85	0.098983	1.655144	0.085322	2.05	0.096613	2.05	0.101516
0.110959	1.95	0.099949	1.663923	0.090391	2.116667	0.098225	2.072222	0.108696
0.136458	2.25	0.099972	1.669776	0.093118	2.161111	0.098965	2.087037	
		0.100252	1.681481	0.096135	2.190741	0.099162	2.096914	
		0.109839	1.8	0.097065	2.210494	0.100519	2.116667	
				0.097921	2.223663	0.111426	2.25	
				0.098771	2.232442			
				0.100666	2.25			

Sa(T1)	Rec 6		Rec 7		Rec 8		Rec 9		Sa(T1)
	Drift	Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)	
0.025	0.001376	0.025	0.001016	0.025	0.001269	0.025	0.001085	0.025	
0.05	0.002286	0.05	0.001927	0.05	0.002298	0.05	0.001902	0.05	
0.1	0.00422	0.1	0.003601	0.1	0.004226	0.1	0.003899	0.1	
0.15	0.006701	0.15	0.005461	0.15	0.006418	0.15	0.005926	0.15	
0.225	0.010431	0.225	0.008254	0.225	0.009871	0.225	0.008989	0.225	
0.3	0.014138	0.3	0.011082	0.3	0.013597	0.3	0.011817	0.3	
0.4	0.020878	0.4	0.015235	0.4	0.018535	0.4	0.015947	0.4	
0.5	0.028299	0.5	0.019599	0.5	0.024429	0.5	0.021098	0.5	
0.625	0.041335	0.625	0.024682	0.625	0.033024	0.625	0.028673	0.625	
0.75	0.060745	0.75	0.031005	0.75	0.044876	0.75	0.037519	0.75	
0.9	0.082575	0.85	0.041653	0.9	0.062979	0.9	0.050452	0.9	
1.05	0.096324	0.916667	0.054283	1.05	0.083029	1.05	0.064973	1.05	
1.166667	0.097876	0.921605	0.071006	1.225	0.089668	1.088889	0.080964	1.225	
1.244444	0.099724	0.931481	0.089653	1.4	0.093307	1.114815	0.099716	1.4	
1.296296	0.106721	0.961111	0.094851	1.444444	0.095439	1.132099	0.100363	1.404938	
1.307819	0.125066	1.05	0.098654	1.474074	0.097126	1.143621	0.101063	1.414815	
1.315501			0.099693	1.480658	0.098203	1.151303	0.105164	1.444444	
1.330864			0.101379	1.493827	0.098223	1.156424	0.113824	1.533333	
1.4			0.106276	1.533333	0.101196	1.166667	0.14138	1.8	
			0.14125	1.8	0.135956	1.4			

Rec 10		Rec 11		Rec 12		Rec 13		Rec 14	
Drift	Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)	Drift	
0.001374	0.025	0.001177	0.025	0.001345	0.025	0.000983	0.025	0.001012	
0.002381	0.05	0.002209	0.05	0.002409	0.05	0.001687	0.05	0.00188	
0.00477	0.1	0.004401	0.1	0.0046	0.1	0.00348	0.1	0.003453	
0.007183	0.15	0.006736	0.15	0.007167	0.15	0.005194	0.15	0.005321	
0.010673	0.225	0.010178	0.225	0.010495	0.225	0.007908	0.225	0.008299	
0.014863	0.3	0.013926	0.3	0.014764	0.3	0.011652	0.3	0.011241	
0.027051	0.4	0.01943	0.4	0.021455	0.4	0.016816	0.4	0.015102	
0.047098	0.5	0.025693	0.5	0.030121	0.5	0.023256	0.5	0.019387	
0.053362	0.527778	0.036675	0.625	0.043628	0.625	0.035703	0.625	0.025953	
0.057747	0.546296	0.051108	0.75	0.062012	0.75	0.051338	0.75	0.035077	
0.058981	0.550412	0.070935	0.9	0.081814	0.85	0.072904	0.9	0.049105	
0.060759	0.558642	0.09835	1.05	0.096749	0.916667	0.095599	1.05	0.067063	
0.067428	0.583333	0.099116	1.054321	0.099488	0.931481	0.097333	1.062963	0.082827	
0.10066	0.75	0.100611	1.062963	0.099734	0.934774	0.098309	1.071605	0.094551	
		0.105933	1.088889	0.101565	0.941358	0.099163	1.077366	0.097483	
		0.119516	1.166667	0.105902	0.961111	0.099901	1.081207	0.099356	
		0.161516	1.4	0.125115	1.05	0.100703	1.088889	0.100744	
						0.111135	1.166667	0.102868	
						0.140012	1.4	0.120344	

[illegible]

Rec 19		Rec 20		Rec 21		Rec 22		Rec 23
Drift	Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)	Drift
0.000984	0.025	0.001118	0.025	0.000988	0.025	0.00099	0.025	0.001135
0.001925	0.05	0.002051	0.05	0.001888	0.05	0.00182	0.05	0.002335
0.003493	0.1	0.00379	0.1	0.003554	0.1	0.003479	0.1	0.00452
0.005279	0.15	0.005872	0.15	0.005238	0.15	0.005227	0.15	0.006645
0.008064	0.225	0.009008	0.225	0.007923	0.225	0.008067	0.225	0.009734
0.010789	0.3	0.012023	0.3	0.010531	0.3	0.010985	0.3	0.012928
0.014793	0.4	0.017403	0.4	0.013186	0.4	0.015527	0.4	0.017425
0.019328	0.5	0.023059	0.5	0.016437	0.5	0.02074	0.5	0.021953
0.025105	0.625	0.030932	0.625	0.020635	0.625	0.027315	0.625	0.02781
0.030138	0.75	0.039391	0.75	0.024812	0.75	0.03507	0.75	0.033949
0.034678	0.9	0.050915	0.9	0.029812	0.9	0.046424	0.9	0.041562
0.040985	1.05	0.061886	1.05	0.03506	1.05	0.058609	1.05	0.049236
0.046134	1.225	0.075405	1.225	0.042204	1.225	0.071981	1.225	0.068371
0.061969	1.4	0.087507	1.4	0.051474	1.4	0.082203	1.4	0.092017
0.08074	1.6	0.097161	1.533333	0.062135	1.6	0.089335	1.533333	0.094842
0.098169	1.8	0.098894	1.562963	0.072022	1.8	0.093725	1.622222	0.096848
0.099664	1.816667	0.098918	1.569547	0.081855	2.025	0.09579	1.681481	0.098085
0.099996	1.827778	0.100622	1.582716	0.090998	2.25	0.098872	1.720988	0.099108
0.100985	1.835185	0.102587	1.622222	0.097261	2.416667	0.099601	1.747325	0.099678
0.102304	1.85	0.114158	1.8	0.099971	2.527778	0.099285	1.764883	0.100706
0.109431	1.95			0.100268	2.552469	0.099858	1.776589	0.100791
0.13636	2.25			0.103095	2.601852	0.099427	1.784393	0.120563
				0.107036	2.75	0.101547	1.8	

Rec 24		Rec 25		Rec 26		Rec 27		
Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)
0.025	0.001017	0.025	0.001067	0.025	0.000927	0.025	0.000924	0.025
0.05	0.00184	0.05	0.001922	0.05	0.00177	0.05	0.001752	0.05
0.1	0.003596	0.1	0.003446	0.1	0.003298	0.1	0.003286	0.1
0.15	0.005534	0.15	0.005026	0.15	0.005135	0.15	0.005078	0.15
0.225	0.008104	0.225	0.007534	0.225	0.007732	0.225	0.007581	0.225
0.3	0.011106	0.3	0.010094	0.3	0.010346	0.3	0.010628	0.3
0.4	0.01468	0.4	0.013114	0.4	0.013976	0.4	0.014211	0.4
0.5	0.018142	0.5	0.016937	0.5	0.0174	0.5	0.017777	0.5
0.625	0.023111	0.625	0.021258	0.625	0.021707	0.625	0.022142	0.625
0.75	0.028964	0.75	0.02555	0.75	0.026326	0.75	0.026861	0.75
0.9	0.036528	0.9	0.030901	0.9	0.032544	0.9	0.033209	0.9
1.05	0.043944	1.05	0.035742	1.05	0.039488	1.05	0.040564	1.05
1.4	0.065448	1.4	0.041255	1.225	0.047873	1.225	0.05745	1.4
1.8	0.093037	1.8	0.047535	1.4	0.055495	1.4	0.077969	1.8
1.85	0.095969	1.85	0.054607	1.6	0.074627	1.8	0.085926	1.95
1.883333	0.097579	1.883333	0.060581	1.8	0.096749	2.25	0.090848	2.05
1.905556	0.099486	1.890741	0.075339	2.25	0.1	2.305556	0.094182	2.116667
1.92037	0.10001	1.905556	0.088296	2.75	0.099442	2.317901	0.096308	2.161111
1.930247	0.103013	1.95	0.094146	2.933333	0.101705	2.326132	0.097775	2.190741
1.936831	0.125314	2.25	0.09675	3.055556	0.100846	2.342593	0.09893	2.210494
1.95			0.098138	3.137037	0.105348	2.416667	0.099916	2.223663
2.25			0.099589	3.191358	0.12033	2.75	0.099917	2.232442
			0.100246	3.203429			0.100671	2.25
			0.100886	3.227572				
			0.103084	3.3				

Rec 28		Rec 29		Rec 30		Rec 31		Rec 32
Drift	Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)	Drift
0.001052	0.025	0.000711	0.025	0.000996	0.025	0.001285	0.025	0.000841
0.00193	0.05	0.001363	0.05	0.001799	0.05	0.00246	0.05	0.001674
0.003593	0.1	0.002374	0.1	0.003371	0.1	0.004638	0.1	0.003175
0.00565	0.15	0.003496	0.15	0.004893	0.15	0.006836	0.15	0.004945
0.008274	0.225	0.005262	0.225	0.007108	0.225	0.01054	0.225	0.007368
0.010998	0.3	0.006769	0.3	0.009423	0.3	0.014512	0.3	0.009329
0.014681	0.4	0.00884	0.4	0.013171	0.4	0.020532	0.4	0.012055
0.018339	0.5	0.011012	0.5	0.016971	0.5	0.026913	0.5	0.015276
0.023395	0.625	0.013927	0.625	0.021325	0.625	0.035425	0.625	0.019633
0.029117	0.75	0.017166	0.75	0.025409	0.75	0.043562	0.75	0.024672
0.036735	0.9	0.020994	0.9	0.030296	0.9	0.065634	1.05	0.030662
0.045164	1.05	0.024539	1.05	0.034977	1.05	0.092115	1.4	0.03787
0.064583	1.4	0.028092	1.225	0.041078	1.225	0.094542	1.444444	0.046237
0.085868	1.8	0.03252	1.4	0.048685	1.4	0.099093	1.474074	0.054021
0.092882	1.95	0.037722	1.6	0.068188	1.8	0.099419	1.493827	0.061996
0.097072	2.05	0.04283	1.8	0.093752	2.25	0.101123	1.506996	0.07004
0.098425	2.072222	0.048851	2.025	0.096045	2.305556	0.102359	1.533333	0.083806
0.098324	2.087037	0.054808	2.25	0.09826	2.342593	0.123716	1.8	0.094207
0.099025	2.096914	0.060953	2.5	0.099388	2.367284			0.097733
0.100197	2.116667	0.067174	2.75	0.101039	2.383745			0.098201
0.107246	2.25	0.079665	3.3	0.103033	2.416667			0.098731
		0.094038	3.9	0.121572	2.75			0.100271
		0.096801	3.972222					0.101642
		0.098964	4.02037					0.114424
		0.098622	4.052469					
		0.099786	4.073868					
		0.096329	4.088134					
		0.101533	4.116667					
		0.108408	4.55					

Rec 33		Rec 34		Rec 35		Rec 36		
Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)
0.025	0.001033	0.025	0.000795	0.025	0.000843	0.025	0.001007	0.025
0.05	0.001975	0.05	0.001572	0.05	0.001654	0.05	0.001943	0.05
0.1	0.00386	0.1	0.003156	0.1	0.003243	0.1	0.003575	0.1
0.15	0.005996	0.15	0.004638	0.15	0.005005	0.15	0.005337	0.15
0.225	0.009085	0.225	0.006544	0.225	0.007289	0.225	0.008302	0.225
0.3	0.0116	0.3	0.008298	0.3	0.009449	0.3	0.01097	0.3
0.4	0.015886	0.4	0.010709	0.4	0.012436	0.4	0.014765	0.4
0.5	0.020469	0.5	0.013445	0.5	0.015572	0.5	0.018326	0.5
0.625	0.02711	0.625	0.0171	0.625	0.019585	0.625	0.024096	0.625
0.75	0.034304	0.75	0.020566	0.75	0.024389	0.75	0.030248	0.75
0.9	0.058565	1.05	0.025083	0.9	0.03078	0.9	0.048677	1.05
1.05	0.093121	1.4	0.029928	1.05	0.037516	1.05	0.074085	1.4
1.225	0.096662	1.444444	0.036658	1.225	0.0458	1.225	0.084277	1.533333
1.4	0.099609	1.474074	0.043591	1.4	0.053469	1.4	0.091043	1.622222
1.6	0.100069	1.480658	0.060084	1.8	0.062003	1.6	0.096269	1.681481
1.8	0.101356	1.493827	0.081414	2.25	0.068395	1.8	0.099372	1.720988
2.25	0.104461	1.533333	0.089419	2.416667	0.078985	2.25	0.100032	1.729767
2.75	0.122918	1.8	0.095224	2.527778	0.085429	2.75	0.101154	1.747325
2.933333			0.099245	2.601852	0.09589	2.933333	0.105176	1.8
2.974074			0.099806	2.618313	0.096758	2.974074		
3.001235			0.100356	2.629287	0.097373	3.001235		
3.019342			0.102045	2.651235	0.09699	3.019342		
3.055556			0.10775	2.75	0.099088	3.031413		
3.3					0.102308	3.055556		
					0.115643	3.3		

Rec 37		Rec 38		Rec 39		Rec 40		Rec 41
Drift	Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)	Drift
0.000892	0.025	0.001137	0.025	0.001402	0.025	0.001831	0.025	0.001206
0.001724	0.05	0.002134	0.05	0.002559	0.05	0.001841	0.05	0.002074
0.003202	0.1	0.004363	0.1	0.005094	0.1	0.003394	0.1	0.004378
0.004904	0.15	0.006787	0.15	0.00758	0.15	0.005991	0.15	0.006074
0.007583	0.225	0.010378	0.225	0.011083	0.225	0.010981	0.225	0.00958
0.009789	0.3	0.01386	0.3	0.014514	0.3	0.012418	0.3	0.01295
0.013366	0.4	0.017924	0.4	0.019085	0.4	0.02851	0.5	0.016464
0.016789	0.5	0.022061	0.5	0.02407	0.5	0.062956	0.75	0.020678
0.021084	0.625	0.035933	0.75	0.030742	0.625	0.064539	0.761111	0.026212
0.025664	0.75	0.061909	1.05	0.037052	0.75	0.066903	0.768519	0.031333
0.032789	0.9	0.075588	1.166667	0.060775	1.05	0.069141	0.783333	0.055077
0.040584	1.05	0.085035	1.244444	0.095254	1.4	0.084468	0.85	0.096942
0.06315	1.4	0.091366	1.296296	0.098559	1.444444	0.14199	1.05	0.098644
0.094657	1.8	0.094676	1.330864	0.098546	1.454321			0.100996
0.099314	1.85	0.09814	1.353909	0.098408	1.460905			0.104565
0.099971	1.861111	0.098501	1.359031	0.101197	1.474074			0.116386
0.100667	1.868519	0.100319	1.369273	0.102996	1.533333			0.146879
0.102628	1.883333	0.104716	1.4	0.123355	1.8			
0.107524	1.95							
0.132738	2.25							

Rec 42		Rec 43		Rec 44		Rec 45		
Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)
0.025	0.0019	0.05	0.001034	0.025	0.001163	0.025	0.000925	0.025
0.05	0.0057	0.15	0.001957	0.05	0.002208	0.05	0.001767	0.05
0.1	0.0117	0.3	0.003742	0.1	0.004554	0.1	0.003595	0.1
0.15	0.02128	0.5	0.005753	0.15	0.007177	0.15	0.005129	0.15
0.225	0.0366	0.75	0.008934	0.225	0.010716	0.225	0.007323	0.225
0.3	0.054	1.05	0.011784	0.3	0.014033	0.3	0.009911	0.3
0.4	0.0885	1.4	0.016139	0.4	0.019055	0.4	0.013517	0.4
0.5	0.1299	1.8	0.021477	0.5	0.024247	0.5	0.016881	0.5
0.625			0.030032	0.625	0.031935	0.625	0.021068	0.625
0.75			0.037012	0.75	0.041067	0.75	0.025636	0.75
1.05			0.055354	1.05	0.052295	0.9	0.032316	0.9
1.4			0.089033	1.4	0.063024	1.05	0.03847	1.05
1.404938			0.093856	1.444444	0.072009	1.225	0.046298	1.225
1.414815			0.097232	1.474074	0.077218	1.4	0.054359	1.4
1.444444			0.09888	1.493827	0.083261	1.8	0.073303	1.8
1.533333			0.099857	1.506996	0.088338	2.25	0.095708	2.25
1.8			0.10109	1.515775	0.090018	2.416667	0.098268	2.305556
			0.103105	1.533333	0.093399	2.527778	0.099954	2.342593
			0.130361	1.8	0.095585	2.601852	0.100664	2.350823
					0.097385	2.651235	0.101075	2.367284
					0.098478	2.684156	0.103692	2.416667
					0.098578	2.706104	0.119315	2.75
					0.09951	2.720736		
					0.097157	2.730491		
					0.100259	2.75		

Rec 46		Rec 47		Rec 48		Rec 49		Rec 50	
Drift	Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)	Drift	
0.000986	0.025	0.001248	0.025	0.001073	0.025	0.001008	0.025	0.001314	
0.001938	0.05	0.002372	0.05	0.001929	0.05	0.00183	0.05	0.002402	
0.003862	0.1	0.004777	0.1	0.003986	0.1	0.003672	0.1	0.004827	
0.006015	0.15	0.007354	0.15	0.00586	0.15	0.005278	0.15	0.007505	
0.00923	0.225	0.011461	0.225	0.008748	0.225	0.008339	0.225	0.011233	
0.011753	0.3	0.016002	0.3	0.011143	0.3	0.011125	0.3	0.015219	
0.015768	0.4	0.021512	0.4	0.014991	0.4	0.015005	0.4	0.020906	
0.020804	0.5	0.027743	0.5	0.018992	0.5	0.018791	0.5	0.026625	
0.036446	0.75	0.050414	0.75	0.029622	0.75	0.02377	0.625	0.044553	
0.06452	1.05	0.078879	1.05	0.042788	1.05	0.028765	0.75	0.074294	
0.077902	1.166667	0.09127	1.166667	0.048462	1.166667	0.035444	0.9	0.086547	
0.087118	1.244444	0.098328	1.244444	0.051122	1.244444	0.042475	1.05	0.092444	
0.09394	1.296296	0.099437	1.250206	0.05289	1.296296	0.059794	1.4	0.096925	
0.097963	1.330864	0.100585	1.261728	0.055235	1.330864	0.082298	1.8	0.097983	
0.099035	1.338546	0.103714	1.296296	0.056899	1.353909	0.09093	1.95	0.099252	
0.099737	1.343667	0.112872	1.4	0.057349	1.369273	0.096129	2.05	0.099295	
0.10084	1.353909			0.057251	1.379515	0.097642	2.072222	0.100406	
0.106695	1.4			0.058838	1.386344	0.098705	2.087037	0.102729	
				0.480002	1.4	0.099192	2.096914		
						0.100539	2.116667		
						0.107592	2.25		

Sa(T1)	Rec 51		Rec 52		Rec 53		Rec 54		Sa(T1)
	Drift	Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)	
0.025	0.001134	0.025	0.000943	0.025	0.000696	0.025	0.000682	0.025	
0.05	0.002068	0.05	0.001825	0.05	0.001391	0.05	0.001378	0.05	
0.1	0.004198	0.1	0.003673	0.1	0.002591	0.1	0.002566	0.1	
0.15	0.006327	0.15	0.005333	0.15	0.00385	0.15	0.003839	0.15	
0.225	0.009385	0.225	0.008215	0.225	0.005437	0.225	0.005841	0.225	
0.3	0.012686	0.3	0.01109	0.3	0.006838	0.3	0.007747	0.3	
0.4	0.017082	0.4	0.014953	0.4	0.009674	0.4	0.010644	0.4	
0.5	0.021414	0.5	0.019166	0.5	0.011958	0.5	0.012668	0.5	
0.75	0.027372	0.625	0.024352	0.625	0.015218	0.625	0.016361	0.625	
1.05	0.03439	0.75	0.030605	0.75	0.018843	0.75	0.020059	0.75	
1.166667	0.052117	1.05	0.038624	0.9	0.023215	0.9	0.024566	0.9	
1.244444	0.078498	1.4	0.045921	1.05	0.027804	1.05	0.029376	1.05	
1.296296	0.089635	1.533333	0.053144	1.225	0.033773	1.225	0.033789	1.225	
1.330864	0.09728	1.622222	0.059023	1.4	0.039495	1.4	0.036765	1.4	
1.338546	0.098953	1.641975	0.064965	1.8	0.046106	1.6	0.041318	1.6	
1.343667	0.100368	1.655144	0.086642	2.25	0.051593	1.8	0.04636	1.8	
1.353909	0.102112	1.681481	0.095381	2.416667	0.057805	2.025	0.050024	2.025	
1.4	0.113603	1.8	0.096999	2.453704	0.063517	2.25	0.054286	2.25	
			0.097192	2.461934	0.077182	2.75	0.057223	2.5	
			0.159547	2.478395	0.091056	3.3	0.059485	2.75	
			0.100497	2.527778	0.097214	3.5	0.067051	3.3	
			0.111547	2.75	0.09997	3.633333	0.08588	3.9	
					0.100345	3.662963	0.092209	4.116667	
					0.10318	3.722222	0.095634	4.261111	
					0.107148	3.9	0.099367	4.357407	
							0.099742	4.378807	
							0.102054	4.421605	
							0.105531	4.55	

Rec 55		Rec 56		Rec 57		Rec 58		Rec 59	
Drift	Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)	Drift	
0.001416	0.025	0.000831	0.025	0.000892	0.025	0.000992	0.025	0.001205	
0.002485	0.05	0.001622	0.05	0.001755	0.05	0.001738	0.05	0.002211	
0.004962	0.1	0.002984	0.1	0.003273	0.1	0.003495	0.1	0.004249	
0.007474	0.15	0.004438	0.15	0.004633	0.15	0.005623	0.15	0.006661	
0.011315	0.225	0.006714	0.225	0.006785	0.225	0.007938	0.225	0.010265	
0.01595	0.3	0.009003	0.3	0.009027	0.3	0.010382	0.3	0.014535	
0.035446	0.5	0.012035	0.4	0.011941	0.4	0.014009	0.4	0.029031	
0.079933	0.75	0.015022	0.5	0.014394	0.5	0.018385	0.5	0.066209	
0.086462	0.783333	0.018572	0.625	0.018237	0.625	0.025575	0.625	0.084268	
0.0882	0.790741	0.022518	0.75	0.022242	0.75	0.034143	0.75	0.087371	
0.089383	0.795679	0.027685	0.9	0.027485	0.9	0.056855	1.05	0.090172	
0.091222	0.798971	0.033152	1.05	0.033068	1.05	0.096286	1.4	0.099401	
0.093289	0.805556	0.040743	1.225	0.040251	1.225	0.098028	1.414815	0.127218	
0.101938	0.85	0.048849	1.4	0.047663	1.4	0.100319	1.424691		
0.150376	1.05	0.068352	1.8	0.056793	1.6	0.101756	1.444444		
		0.090225	2.25	0.065167	1.8	0.114623	1.533333		
		0.09788	2.416667	0.081571	2.25	0.155394	1.8		
		0.099437	2.453704	0.093585	2.75				
		0.099811	2.461934	0.097152	2.933333				
		0.100179	2.478395	0.098832	3.055556				
		0.102304	2.527778	0.099795	3.082716				
		0.112717	2.75	0.099878	3.100823				
				0.099529	3.112894				
				0.10016	3.137037				
				0.102838	3.3				

Rec 60		Rec 61		Rec 62		Rec 63		
Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)
0.025	0.001324	0.025	0.001374	0.025	0.001089	0.025	0.00121	0.025
0.05	0.002181	0.05	0.002409	0.05	0.001939	0.05	0.002145	0.05
0.1	0.004883	0.1	0.004662	0.1	0.003441	0.1	0.003866	0.1
0.15	0.007532	0.15	0.007111	0.15	0.005407	0.15	0.005996	0.15
0.225	0.016466	0.3	0.015394	0.3	0.008178	0.225	0.009794	0.225
0.3	0.050414	0.5	0.040933	0.5	0.011185	0.3	0.013632	0.3
0.5	0.051464	0.503086	0.042904	0.509259	0.015691	0.4	0.028099	0.5
0.75	0.053684	0.509259	0.043643	0.511317	0.019124	0.5	0.061855	0.75
0.85	0.060036	0.527778	0.044636	0.515432	0.023759	0.625	0.068996	0.783333
0.857407	0.076805	0.583333	0.047896	0.527778	0.030658	0.75	0.07278	0.805556
0.872222	0.129125	0.75	0.064024	0.583333	0.037517	0.9	0.07528	0.810494
0.916667			0.114652	0.75	0.045402	1.05	0.077638	0.82037
1.05					0.063847	1.4	0.084955	0.85
					0.08438	1.8	0.162813	1.05
					0.091891	1.95		
					0.09676	2.05		
					0.098426	2.072222		
					0.098327	2.087037		
					0.099937	2.096914		
					0.100549	2.116667		
					0.109367	2.25		

Rec 64		Rec 65		Rec 66		Rec 67		Rec 68
Drift	Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)	Drift
0.001055	0.025	0.00092	0.025	0.001284	0.025	0.001072	0.025	0.000876
0.001688	0.05	0.001497	0.05	0.002208	0.05	0.001755	0.05	0.001535
0.003016	0.1	0.002501	0.1	0.004298	0.1	0.003281	0.1	0.003246
0.004696	0.15	0.005512	0.15	0.006053	0.15	0.005028	0.15	0.004793
0.007691	0.225	0.012587	0.3	0.008717	0.225	0.007617	0.225	0.007192
0.011907	0.3	0.021304	0.5	0.010897	0.3	0.010001	0.3	0.009437
0.020616	0.5	0.031651	0.583333	0.017756	0.4	0.013449	0.4	0.012075
0.042533	0.75	0.033863	0.601852	0.023598	0.5	0.016694	0.5	0.016726
0.053783	0.85	0.034668	0.605967	0.041572	0.75	0.02016	0.625	0.022233
0.055328	0.857407	0.036496	0.614198	0.07105	1.05	0.024279	0.75	0.028587
0.058402	0.872222	0.041348	0.638889	0.084404	1.166667	0.04136	1.05	0.047477
0.067567	0.916667	0.070341	0.75	0.094012	1.244444	0.087796	1.4	0.070061
0.099575	1.05			0.096733	1.261728	0.090375	1.414815	0.079169
				0.097259	1.273251	0.092577	1.424691	0.09237
				0.098947	1.280933	0.096476	1.444444	0.099872
				0.099194	1.286054	0.116123	1.533333	0.100956
				0.101265	1.296296	0.168222	1.8	0.107879
				0.109252	1.4			0.120431

Rec 69		Rec 70		Rec 71		Rec 72		
Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)
0.025	0.000864	0.025	0.001187	0.025	0.000974	0.025	0.00084	0.025
0.05	0.001541	0.05	0.001979	0.05	0.001758	0.05	0.001549	0.05
0.1	0.003088	0.1	0.004032	0.1	0.003023	0.1	0.002753	0.1
0.15	0.004395	0.15	0.006754	0.15	0.004664	0.15	0.004413	0.15
0.225	0.006436	0.225	0.009588	0.225	0.006818	0.225	0.007431	0.225
0.3	0.008415	0.3	0.01268	0.3	0.009425	0.3	0.0097	0.3
0.4	0.011544	0.4	0.019653	0.4	0.015219	0.4	0.013366	0.4
0.5	0.014674	0.5	0.028354	0.5	0.020593	0.5	0.018336	0.5
0.625	0.018617	0.625	0.059814	0.75	0.028445	0.625	0.036115	0.75
0.75	0.023065	0.75	0.092023	1.05	0.037771	0.75	0.070068	1.05
1.05	0.035713	1.05	0.096054	1.088889	0.056441	1.05	0.08745	1.166667
1.4	0.067111	1.4	0.096873	1.114815	0.081199	1.4	0.088458	1.175309
1.533333	0.08069	1.533333	0.099707	1.132099	0.091104	1.533333	0.090015	1.18107
1.622222	0.08816	1.622222	0.100956	1.13594	0.097735	1.622222	0.09278	1.192593
1.681481	0.094121	1.681481	0.101582	1.143621	0.099561	1.628807	0.10118	1.244444
1.69465	0.097373	1.720988	0.103186	1.166667	0.100802	1.641975	0.132435	1.4
1.720988	0.098604	1.747325	0.134143	1.4	0.102256	1.681481		
1.8	0.099377	1.753178			0.113595	1.8		
	0.100912	1.764883						
	0.102765	1.8						

Rec 73		Rec 74		Rec 75		Rec 76		Rec 77
Drift	Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)	Drift
0.00069	0.025	0.001306	0.025	0.001454	0.025	0.001115	0.025	0.000977
0.001399	0.05	0.002048	0.05	0.002559	0.05	0.00201	0.05	0.001802
0.002435	0.1	0.004532	0.1	0.004752	0.1	0.003744	0.1	0.003189
0.003646	0.15	0.006359	0.15	0.00743	0.15	0.005679	0.15	0.004956
0.005715	0.225	0.010575	0.225	0.010795	0.225	0.008887	0.225	0.00782
0.007175	0.3	0.015773	0.3	0.015569	0.3	0.012248	0.3	0.010756
0.010545	0.4	0.032513	0.5	0.029873	0.5	0.016358	0.4	0.013193
0.012483	0.5	0.065425	0.75	0.063675	0.75	0.020506	0.5	0.017289
0.015498	0.625	0.084097	0.85	0.084201	0.85	0.037361	0.75	0.023294
0.019004	0.75	0.086329	0.857407	0.089411	0.872222	0.065959	1.05	0.029803
0.02352	0.9	0.089126	0.872222	0.090487	0.87716	0.081014	1.166667	0.05317
0.028564	1.05	0.097817	0.916667	0.091611	0.880453	0.092239	1.244444	0.08999
0.047437	1.4	0.122416	1.05	0.094071	0.887037	0.094651	1.261728	0.097212
0.083453	1.8			0.101298	0.916667	0.096041	1.273251	0.098151
0.099366	1.95			0.136847	1.05	0.097286	1.280933	0.098267
0.100065	1.961111					0.098048	1.286054	0.100949
0.102432	1.983333					0.100506	1.296296	0.102747
0.107933	2.05					0.116458	1.4	0.110054
0.125754	2.25							

[illegible]

Rec 82		Rec 83		Rec 84		Rec 85		Rec 86
Drift	Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)	Drift	Sa(T1)	Drift
0.000991	0.025	0.000987	0.025	0.001138	0.025	0.000913	0.025	0.001006
0.001808	0.05	0.001817	0.05	0.001839	0.05	0.001822	0.05	0.001861
0.003231	0.1	0.003935	0.1	0.003336	0.1	0.003176	0.1	0.003711
0.005323	0.15	0.0059	0.15	0.004868	0.15	0.004851	0.15	0.005638
0.010175	0.225	0.009038	0.225	0.007372	0.225	0.007395	0.225	0.008425
0.013811	0.3	0.012855	0.3	0.011714	0.3	0.00946	0.3	0.011572
0.025573	0.5	0.017768	0.4	0.024449	0.5	0.011593	0.4	0.015399
0.056505	0.75	0.022433	0.5	0.05751	0.75	0.011636	0.5	0.018729
0.057344	0.753704	0.034061	0.75	0.063634	0.783333	0.014498	0.625	0.023752
0.05914	0.761111	0.072521	1.05	0.066242	0.790741	0.017448	0.75	0.028948
0.062778	0.783333	0.093372	1.166667	0.068958	0.805556	0.023626	0.9	0.035982
0.078348	0.85	0.094895	1.175309	0.077942	0.85	0.031553	1.05	0.044199
0.135247	1.05	0.09538	1.18107	0.125027	1.05	0.041305	1.225	0.063183
		0.098194	1.192593			0.051594	1.4	0.085829
		0.108181	1.244444			0.070377	1.8	0.094867
		0.138744	1.4			0.096083	2.25	0.096448
						0.098346	2.268519	0.099178
						0.099689	2.280864	0.100111
						0.098524	2.289095	0.101607
						0.102638	2.305556	0.113877
						0.115025	2.416667	
						0.145586	2.75	

[illegible]

Rec 91

Drift	Sa(T1)
0.000872	0.025
0.00169	0.05
0.003153	0.1
0.004844	0.15
0.0075	0.225
0.010078	0.3
0.013565	0.4
0.016975	0.5
0.021438	0.625
0.026669	0.75
0.034204	0.9
0.042841	1.05
0.066888	1.4
0.098745	1.8
0.099732	1.816667
0.100841	1.827778
0.103081	1.85
0.111458	1.95
0.135347	2.25

```

function results = Optimal_IM_Selection_Jacket_Article(dataFile, outputDir)
% OPTIMAL_IM_SELECTION_JACKET_ARTICLE
% -----

    if nargin < 1 || isempty(dataFile)
        createArticleTemplate('Article_IM_Data_Template.csv');
        fprintf('Created Article_IM_Data_Template.csv\n');
        fprintf('Fill it with the actual OpenSees/IDA results and run again.\n');
        results = [];
        return
    end

    if nargin < 2 || isempty(outputDir)
        outputDir = 'Article_Results';
    end

    if ~isfile(dataFile)
        error('Input file not found: %s', dataFile);
    end
    if ~isfolder(outputDir)
        mkdir(outputDir);
    end

    alpha = 0.05;
    selectedIMs = [16, 25, 30];
    imNames = articleIMNames();

    D = readtable(dataFile, 'TextType', 'string', ...
        'VariableNamingRule', 'preserve');

    required = ["RecordID", "Mw", "R_km", "ScaleFactor", "MIDR"];
    for j = 1:57
        required(end+1) = sprintf("IM%02d", j); %#ok<AGROW>
    end

    variableNames = string(D.Properties.VariableNames);
    missing = required(~ismember(required, variableNames));
    if ~isempty(missing)
        error('Missing required columns: %s', strjoin(missing, ', '));
    end

    recordID = string(D("RecordID"));
    Mw = toNumeric(D("Mw"));
    Rkm = toNumeric(D("R_km"));
    scaleFactor = toNumeric(D("ScaleFactor"));
    MIDR = toNumeric(D("MIDR"));

    if any(~isfinite(MIDR) | MIDR <= 0)
        error('MIDR values must be finite and strictly positive.');
```

```

nRows = height(D);
IM = nan(nRows,57);
for j = 1:57
    IM(:,j) = toNumeric(D.(sprintf('IM%02d',j)));
end

% -----
% Fit all 57 scalar PSDMs
% -----
N = zeros(57,1);
a = nan(57,1);
b = nan(57,1);
R2 = nan(57,1);
beta = nan(57,1);
zeta = nan(57,1);
pMagnitude = nan(57,1);
pDistance = nan(57,1);
pScale = nan(57,1);
residuals = nan(nRows,57);

for j = 1:57
    fit = fitPSDM(IM(:,j), MIDR);
    N(j) = fit.N;
    a(j) = fit.a;
    b(j) = fit.b;
    R2(j) = fit.R2;
    beta(j) = fit.beta;
    zeta(j) = fit.beta / max(abs(fit.b), eps);

    if fit.N >= 3
        residuals(fit.mask,j) = fit.residual;
        pMagnitude(j) = regressionSlopePValue( ...
            Mw(fit.mask), fit.residual);
        pDistance(j) = regressionSlopePValue( ...
            Rkm(fit.mask), fit.residual);
        pScale(j) = regressionSlopePValue( ...
            scaleFactor(fit.mask), fit.residual);
    end
end

end

Summary = table((1:57)', string(imNames(:)), N, a, b, R2, beta, zeta, ...
    pMagnitude, pDistance, pScale, ...
    pMagnitude >= alpha, pDistance >= alpha, pScale >= alpha, ...
    'VariableNames', { ...
        'IM_No', 'IM_Name', 'N', 'a', 'b_Practicality', 'R2', ...
        'beta_Efficiency', 'zeta_Proficiency', ...
        'p_Magnitude', 'p_Distance', 'p_ScaleFactor', ...
        'Sufficient_Magnitude', 'Sufficient_Distance', 'Scaling_Robust'});

```

```

% Rankings are provided for convenient comparison; the individual
% article criteria remain available in the preceding columns.
Summary.Rank_Efficiency = ascendingRank(Summary.beta_Efficiency);
Summary.Rank_Practicality = descendingRank(abs(Summary.b_Practicality));
Summary.Rank_Proficiency = ascendingRank(Summary.zeta_Proficiency);
Summary.Rank_Sufficiency = descendingRank( ...
    min(Summary.p_Magnitude, Summary.p_Distance));
Summary.Rank_Scaling = descendingRank(Summary.p_ScaleFactor);
Summary.Mean_Rank = mean([ ...
    Summary.Rank_Efficiency, ...
    Summary.Rank_Practicality, ...
    Summary.Rank_Proficiency, ...
    Summary.Rank_Sufficiency, ...
    Summary.Rank_Scaling], 2, 'omitnan');

Ranking = sortrows(Summary, ...
    {'Mean_Rank','beta_Efficiency'}, {'ascend','ascend'});

writetable(Summary, fullfile(outputDir,'Article_IM_Summary.csv'));
writetable(Ranking, fullfile(outputDir,'Article_IM_Ranking.csv'));

ResidualTable = table(recordID,'VariableNames',{'RecordID'});
for j = 1:57
    ResidualTable.(sprintf('Residual_IM%02d',j)) = residuals(:,j);
end
writetable(ResidualTable, fullfile(outputDir,'Article_PSDM_Residuals.csv'));

% -----
% Article figures
% -----
createRegressionComparisonFigure(IM, MIDR, Summary, outputDir);
createMetricFigure((1:57)', beta, ...
    'Efficiency', '\beta', fullfile(outputDir,'Fig4_Efficiency.png'));
createMetricFigure((1:57)', b, ...
    'Practicality', 'b', fullfile(outputDir,'Fig5_Practicality.png'));
createMetricFigure((1:57)', zeta, ...
    'Proficiency', '\zeta', fullfile(outputDir,'Fig6_Proficiency.png'));

for imNo = selectedIMs
    createSufficiencyFigure( ...
        Mw, Rkm, residuals(:,imNo), imNo, imNames{imNo}, outputDir);
end

createScalingFigure( ...
    scaleFactor, residuals, selectedIMs, imNames, outputDir);

% -----
% Collapse fragility curves, if capacity columns exist
% -----
[FragilitySummary, FragilityCurves] = ...

```

```

        calculateOptionalFragility(D, recordID, imNames, outputDir);

% -----
% Benchmark comparison to values stated in the manuscript
% -----
Benchmark = articleBenchmarks(Summary);
writetable(Benchmark, fullfile(outputDir,'Article_Benchmark_Check.csv'));

fprintf('\nTop 10 IMs using the combined diagnostic ranking:\n');
disp(Ranking(1:min(10,height(Ranking)), ...
    {'IM_No','IM_Name','beta_Efficiency','b_Practicality', ...
    'R2','zeta_Proficiency','p_Magnitude','p_Distance', ...
    'p_ScaleFactor','Mean_Rank'}));

fprintf('\nKey article IMs:\n');
disp(Summary(ismember(Summary.IM_No,[16 23 25 29 30 41 55 56]),:));

fprintf('\nResults saved in: %s\n', outputDir);

results = struct();
results.Summary = Summary;
results.Ranking = Ranking;
results.Residuals = ResidualTable;
results.FragilitySummary = FragilitySummary;
results.FragilityCurves = FragilityCurves;
results.BenchmarkCheck = Benchmark;
end

% =====
% PSDM
% =====
function fit = fitPSDM(IM, MIDR)
    mask = isfinite(IM) & IM > 0 & isfinite(MIDR) & MIDR > 0;
    x = log(IM(mask));
    y = log(MIDR(mask));

    fit = struct('N',numel(y),'a',NaN,'b',NaN,'R2',NaN, ...
        'beta',NaN,'residual',[],'mask',mask);

    if numel(y) < 3 || std(x) <= eps
        return
    end

    X = [ones(numel(x),1), x];
    coefficient = X \ y;
    yHat = X * coefficient;
    e = y - yHat;

    SSE = sum(e.^2);

```

```

    SST = sum((y-mean(y)).^2);

    fit.a = exp(coefficient(1));
    fit.b = coefficient(2);
    fit.R2 = 1 - SSE/max(SST,eps);
    fit.beta = sqrt(SSE/(numel(y)-2));
    fit.residual = e;
end

% =====
% SUFFICIENCY AND ROBUSTNESS
% =====
function p = regressionSlopePValue(x,y)
    mask = isfinite(x) & isfinite(y);
    x = x(mask);
    y = y(mask);
    n = numel(y);

    if n < 3 || std(x) <= eps
        p = NaN;
        return
    end

    X = [ones(n,1),x];
    coefficient = X \ y;
    residual = y-X*coefficient;
    variance = sum(residual.^2)/(n-2);
    covariance = variance * pinv(X'*X);
    standardError = sqrt(max(covariance(2,2),0));

    if standardError <= eps
        p = NaN;
        return
    end

    tValue = abs(coefficient(2)/standardError);
    degreesFreedom = n-2;

    % Exact two-sided Student-t probability, without Statistics Toolbox.
    p = betainc(degreesFreedom/(degreesFreedom+tValue^2), ...
        degreesFreedom/2, 0.5);
end

% =====
% OPTIONAL FRAGILITY ANALYSIS
% =====
function [FragilitySummary, FragilityCurves] = ...
    calculateOptionalFragility(D, recordID, imNames, outputDir)

```

```

variableNames = string(D.Properties.VariableNames);
capacityColumns = strings(57,1);
available = false(57,1);

for j = 1:57
    capacityColumns(j) = sprintf("Capacity_IM%02d",j);
    available(j) = ismember(capacityColumns(j),variableNames);
end

FragilitySummary = table();
FragilityCurves = table();

if ~any(available)
    fprintf(['No Capacity_IMxx columns were found; ', ...
            'fragility curves were skipped.\n']);
    return
end

summaryRows = [];
curveRows = [];

for j = find(available)'
    capacityAll = toNumeric(D.(capacityColumns(j)));

    % One collapse capacity per record. Repeated IDA rows are reduced
    % using the first finite capacity for each RecordID.
    uniqueIDs = unique(recordID,'stable');
    capacity = nan(numel(uniqueIDs),1);
    for k = 1:numel(uniqueIDs)
        idx = recordID == uniqueIDs(k) & ...
              isfinite(capacityAll) & capacityAll > 0;
        values = capacityAll(idx);
        if ~isempty(values)
            capacity(k) = values(1);
        end
    end
    capacity = capacity(isfinite(capacity) & capacity > 0);

    if numel(capacity) < 3
        continue
    end

    lnCapacity = log(capacity);
    medianCapacity = exp(mean(lnCapacity));
    betaCapacity = std(lnCapacity,0);

    imGrid = logspace( ...
        log10(max(min(capacity)*0.25,eps)), ...
        log10(max(capacity)*2.0), 250)';

```

```

probability = standardNormalCDF( ...
    log(imGrid/medianCapacity)/max(betaCapacity,eps));
normalizedIM = imGrid/medianCapacity;

summaryRows = [summaryRows; ...
    table(j,string(imNames{j}),numel(capacity), ...
        medianCapacity,betaCapacity, ...
        'VariableNames',{'IM_No','IM_Name','N_Capacity', ...
        'Median_Capacity','Beta_ln_Capacity'})); %#ok<AGROW>

curveRows = [curveRows; ...
    table(repmat(j,numel(imGrid),1), ...
        repmat(string(imNames{j}),numel(imGrid),1), ...
        imGrid,normalizedIM,probability, ...
        'VariableNames',{'IM_No','IM_Name','IM', ...
        'Normalized_IM','Probability_of_Collapse'})); %#ok<AGROW>
end

if isempty(summaryRows)
    fprintf('Capacity columns existed but contained insufficient data.\n');
    return
end

FragilitySummary = summaryRows;
FragilityCurves = curveRows;

writetable(FragilitySummary, ...
    fullfile(outputDir,'Article_Fragility_Summary.csv'));
writetable(FragilityCurves, ...
    fullfile(outputDir,'Article_Fragility_Curves.csv'));

% Figure 11 focuses on article-highlighted IMs.
focus = [23 25 30 55];
focus = focus(ismember(focus,FragilitySummary.IM_No));
if ~isempty(focus)
    f = figure('Visible','off','Color','w');
    hold on
    for imNo = focus
        rows = FragilityCurves.IM_No == imNo;
        plot(FragilityCurves.Normalized_IM(rows), ...
            FragilityCurves.Probability_of_Collapse(rows), ...
            'LineWidth',1.5, ...
            'DisplayName',sprintf('IM%d - %s', ...
            imNo,imNames{imNo}));
    end
    grid on
    xlabel('Normalized IM');
    ylabel('Probability of Collapse');
    title('Collapse Fragility Curves');
    xlim([0 2]);

```

```

        ylim([0 1]);
        legend('Location','best','Interpreter','none');
        saveFigure(f,fullfile(outputDir,'Fig11_Fragility_Curves.png'));
        close(f)
    end
end

```

```

function p = standardNormalCDF(z)
    p = 0.5*(1+erf(z/sqrt(2)));
end

```

```

% =====
% ARTICLE FIGURES
% =====

```

```

function createRegressionComparisonFigure(IM,MIDR,Summary,outputDir)
    f = figure('Visible','off','Color','w', ...
        'Position',[100 100 850 900]);

```

```

    imList = [30 25];
    figureTitles = ["IM30: Best IM","IM25: Weak IM"];

```

```

    for k = 1:2
        imNo = imList(k);
        mask = isfinite(IM(:,imNo)) & IM(:,imNo)>0 & ...
            isfinite(MIDR) & MIDR>0;
        x = log(IM(mask,imNo));
        y = log(MIDR(mask));

        subplot(2,1,k)
        scatter(x,y,16,'filled');
        hold on
        if numel(x) >= 3
            coefficient = [ones(numel(x),1),x]\y;
            xLine = linspace(min(x),max(x),200)';
            yLine = coefficient(1)+coefficient(2)*xLine;
            plot(xLine,yLine,'LineWidth',1.6);
            equationText = sprintf( ...
                'ln(MIDR) = %.4f + %.4f ln(IM), R^2 = %.4f', ...
                coefficient(1),coefficient(2),Summary.R2(imNo));
            text(min(x)+0.05*rangeSafe(x), ...
                max(y)-0.10*rangeSafe(y),equationText, ...
                'Interpreter','none');
        end
    end

```

```

    grid on
    xlabel(sprintf('ln(IM%d)',imNo));
    ylabel('ln(MIDR)');
    title(figureTitles(k));
end

```

```

    saveFigure(f,fullfile(outputDir,'Fig3_Regression_IM30_IM25.png'));
    close(f)
end

```

```

function createMetricFigure(x,y,titleText,yLabelText,fileName)
    f = figure('Visible','off','Color','w');
    plot(x,y,'-d','LineWidth',1.25,'MarkerSize',5);
    grid on
    xlabel('IM No. ');
    ylabel(yLabelText);
    title(titleText);
    xlim([1 57]);
    saveFigure(f,fileName);
    close(f)
end

```

```

function createSufficiencyFigure(Mw,Rkm,residual,imNo,imName,outputDir)
    f = figure('Visible','off','Color','w','Position',[100 100 900 420]);

    subplot(1,2,1)
    regressionScatter(Mw,residual,'Magnitude','Residual');
    title(sprintf('IM%d (%s): Magnitude',imNo,imName), ...
        'Interpreter','none');

    subplot(1,2,2)
    regressionScatter(Rkm,residual,'Distance (km)','Residual');
    title(sprintf('IM%d (%s): Distance',imNo,imName), ...
        'Interpreter','none');

    saveFigure(f,fullfile(outputDir, ...
        sprintf('Sufficiency_IM%02d.png',imNo)));
    close(f)
end

```

```

function createScalingFigure(scaleFactor,residuals,imList,imNames,outputDir)
    f = figure('Visible','off','Color','w','Position',[100 100 900 900]);

    for k = 1:numel(imList)
        imNo = imList(k);
        subplot(numel(imList),1,k)
        regressionScatter(scaleFactor,residuals(:,imNo), ...
            'Scale Factor','Residual');
        title(sprintf('IM%d (%s)',imNo,imNames{imNo}), ...
            'Interpreter','none');
    end
end

```

```

    saveFigure(f,fullfile(outputDir,'Fig10_Scaling_Robustness.png'));
    close(f)
end

```

```

function regressionScatter(x,y,xLabelText,yLabelText)
    mask = isfinite(x) & isfinite(y);
    x = x(mask);
    y = y(mask);

    scatter(x,y,16,'filled');
    hold on
    grid on
    xlabel(xLabelText);
    ylabel(yLabelText);

    if numel(y) >= 3 && std(x)>eps
        coefficient = [ones(numel(x),1),x]\y;
        xLine = linspace(min(x),max(x),200)';
        yLine = coefficient(1)+coefficient(2)*xLine;
        plot(xLine,yLine,'LineWidth',1.5);
        p = regressionSlopePValue(x,y);
        text(min(x)+0.05*rangeSafe(x), ...
            max(y)-0.10*rangeSafe(y), ...
            sprintf('slope = %.4g, p = %.4g',coefficient(2),p));
    end
end

```

```

function saveFigure(f,fileName)
    try
        exportgraphics(f,fileName,'Resolution',300);
    catch
        saveas(f,fileName);
    end
end

```

```

function r = rangeSafe(x)
    r = max(x)-min(x);
    if ~isfinite(r) || r<=0
        r = 1;
    end
end

```

```

% =====
% BENCHMARK CHECK
% =====
function B = articleBenchmarks(Summary)

```

```

% Values explicitly stated in the manuscript text. These are comparison
% targets only; they never replace calculated results.
IM_No = [23;25;29;30;35;41;48;51;55;56];
Metric = [ ...
    "b"; ...
    "beta"; ...
    "beta"; ...
    "beta"; ...
    "beta"; ...
    "beta"; ...
    "beta"; ...
    "b"; ...
    "beta"];
Reported = [ ...
    1.536; ...
    1.054; ...
    0.3502; ...
    0.2646; ...
    0.3538; ...
    0.3493; ...
    0.9764; ...
    0.8222; ...
    0.389; ...
    0.3574];

Calculated = nan(size(Reported));
for i = 1:numel(IM_No)
    if Metric(i) == "b"
        Calculated(i) = Summary.b_Practicality(IM_No(i));
    else
        Calculated(i) = Summary.beta_Efficiency(IM_No(i));
    end
end

Absolute_Difference = abs(Calculated-Reported);
B = table(IM_No,Metric,Reported,Calculated,Absolute_Difference);
end

% =====
% RANKING
% =====
function rank = ascendingRank(x)
    x = x(:);
    rank = nan(size(x));
    finite = isfinite(x);
    finiteIndices = find(finite);
    [~,order] = sort(x(finite),'ascend');
    rank(finiteIndices(order)) = (1:sum(finite))';

```

```

    rank(~finite) = sum(finite)+1;
end

```

```

function rank = descendingRank(x)
    x = x(:);
    rank = nan(size(x));
    finite = isfinite(x);
    finiteIndices = find(finite);
    [~,order] = sort(x(finite),'descend');
    rank(finiteIndices(order)) = (1:sum(finite))';
    rank(~finite) = sum(finite)+1;
end

```

```

% =====
% INPUT HELPERS
% =====
function x = toNumeric(value)
    if isnumeric(value)
        x = double(value);
    else
        x = str2double(string(value));
    end
    x = x(:);
end

```

```

function createArticleTemplate(fileName)
    n = 91;
    T = table("GM"+compose("%03d",(1:n)'), ...
        nan(n,1),nan(n,1),nan(n,1),nan(n,1), ...
        'VariableNames',{'RecordID','Mw','R_km','ScaleFactor','MIDR'});

    for j = 1:57
        T.(sprintf('IM%02d',j)) = nan(n,1);
    end

    % Optional capacity columns are included in the template. Delete them
    % if fragility curves are not part of the submitted data package.
    for j = 1:57
        T.(sprintf('Capacity_IM%02d',j)) = nan(n,1);
    end

    writetable(T,fileName);
end

```

```

function names = articleIMNames()
    names = { ...

```

'PGA', ...
'RMSa', ...
'SMA', ...
'Arias Intensity', ...
'D5-75', ...
'D5-95', ...
'Characteristic Intensity', ...
'EDA', ...
'A95', ...
'MIV', ...
'Destructiveness Potential', ...
'a_rs', ...
'CAV', ...
'Riddell-Garcia Acceleration', ...
'PGV', ...
'RMSv', ...
'SED', ...
'v_rs', ...
'Velocity Intensity', ...
'Fajfar Intensity', ...
'SMV', ...
'Medium-Period IM', ...
'Riddell-Garcia Velocity', ...
'PGD', ...
'Displacement Intensity', ...
'RMSd', ...
'd_rs', ...
'Riddell-Garcia Displacement', ...
'Sa(T1)', ...
'Cordova Intensity', ...
'EPA', ...
'ASI', ...
'Average Spectral Acceleration', ...
'MPMR', ...
'Two-Period Intensity', ...
'EPV', ...
'Housner Intensity', ...
'VSI', ...
'Spectral Velocity', ...
'EPD', ...
'Spectral Displacement', ...
'PGV/PGA', ...
'PGD/PGV', ...
'PGV^2/PGA', ...
'PGD^2/RMSd', ...
'Combined IM_1', ...
'Damage Index', ...
'Cosenza-Manfredi Intensity', ...
'Impulsivity Index', ...
'Mean Period', ...

```
'Predominant Period (Sa)', ...  
'Predominant Period (Sv)', ...  
'IEPA', ...  
'IEPV', ...  
'SIR', ...  
'VWBI', ...  
'CAV_STD'};
```

```
end
```