

Contents

- Unique code
- Matrix - Part 1
- Part 2 Step 1: Use pi group formulas from Part 1 to calculate all pi group data from the provided measurements
- Part 2 Step 2 Identify (and isolate) your independent and dependent
- Part 2 Step 3 Normalize
- Part 2 step 4 Multilinear
- Part 2 step 5 exponential fit
- Part 2 step 6 power fit
- Part 2 step 7 nonlinear fitting
- Check
- Part 2 step 8 rescale
- Part 3
- Part 4

Unique code

```
key = 859098551;
Project_data = ProjectDataGenerator(key);
disp(Project_data);
```

["Measured Parameter Units"]}	{ 1×10 table}
["Measurements"]}	{200×10 table}
["Basis Parameters"]}	{1×3 string }
["Dependent Parameter"]}	{ 'Pressure 1' }

Matrix - Part 1

```
matrix = [0, 1, 1, 1, 1, 0, 0, 0, 0, 0;% M
          2, -3, 0, -1, -1, 0, 0, 1, 0, 0; % L
          0, 0, 0, -3, -2, 1, 1, -1, 0, 0; % T
          0, 0, 0, 0, 0, 0, 0, 0, -1, -1]; % Θ
rrefmatrix = rref(matrix);
data_table = Project_data{2,2};
data = table2array(data_table);
dataInOrder = data(:,[1,3,6,2,4,5,7,8,9,10]);
disp(rrefmatrix);
```

Columns 1 through 7						
1.0000	0	1.5000	0	0.3333	0.3333	0.3333
0	1.0000	1.0000	0	0.3333	0.3333	0.3333
0	0	0	1.0000	0.6667	-0.3333	-0.3333
0	0	0	0	0	0	0
Columns 8 through 10						
0.1667	0	0				
-0.3333	0	0				
0.3333	0	0				
0	1.0000	1.0000				

Part 2 Step 1: Use pi group formulas from Part 1 to calculate all pi group data from the provided measurements

```
pi4 = dataInOrder(:,3).^1 ./ dataInOrder(:,4); % (M^0 L^0 T^1 theta^0)/pi4
pi5 = ((dataInOrder(:,1).^0.3333).*(dataInOrder(:,2).^0.3333).*(dataInOrder(:,3).^0.6667))./dataInOrder(:,5); % (M^0.333 L^0.3333 T^0.667 theta^0)/pi5
pi6 = ((dataInOrder(:,1).^0.3333).*(dataInOrder(:,2).^0.3333).*(dataInOrder(:,3).^~0.3333))./dataInOrder(:,6);
pi7 = ((dataInOrder(:,1).^0.3333).*(dataInOrder(:,2).^0.3333).*(dataInOrder(:,3).^0.3333))./dataInOrder(:,7);
pi8 = ((dataInOrder(:,1).^0.1667).*(dataInOrder(:,2).^~0.3333).*(dataInOrder(:,3).^0.3333))./dataInOrder(:,8);
pi9 = dataInOrder(:,4).^1./dataInOrder(:,9);
pi10 = dataInOrder(:,4).^1./dataInOrder(:,10);

A = [pi4 pi5 pi6 pi7 pi8 pi9 pi10];

disp('pi4 pi5 pi6 pi7 pi8 pi9 pi10');
disp(A);
```

pi4 pi5 pi6 pi7 pi8 pi9 pi10
1.0e+07 *

0.0000	0.0000	0.0000	0.0000	0.0000	0.3476	2.0381
0.0000	0.0000	0.0000	0.0000	0.0000	0.5829	1.5846
0.0000	0.0000	0.0000	0.0000	0.0000	1.2859	0.4178
0.0000	0.0000	0.0000	0.0000	0.0000	0.1208	0.2702
0.0000	0.0000	0.0000	0.0000	0.0000	0.2263	1.3314
0.0000	0.0000	0.0000	0.0000	0.0000	0.8077	1.1943
0.0000	0.0000	0.0000	0.0000	0.0000	0.9492	1.5496
0.0000	0.0000	0.0000	0.0000	0.0000	0.5275	0.6731
0.0000	0.0000	0.0000	0.0000	0.0000	1.3263	2.7404
0.0000	0.0000	0.0000	0.0000	0.0000	1.3489	1.7983
0.0000	0.0000	0.0000	0.0000	0.0000	3.5504	4.3282
0.0000	0.0000	0.0000	0.0000	0.0000	2.5491	2.9181
0.0000	0.0000	0.0000	0.0000	0.0000	0.3316	1.0220
0.0000	0.0000	0.0000	0.0000	0.0000	0.6298	4.3334
0.0000	0.0000	0.0000	0.0000	0.0000	0.2431	0.5906
0.0000	0.0000	0.0000	0.0000	0.0000	0.5700	4.6677
0.0000	0.0000	0.0000	0.0000	0.0000	0.6733	2.8362
0.0000	0.0000	0.0000	0.0000	0.0000	0.6207	0.2890
0.0000	0.0000	0.0000	0.0000	0.0000	0.6097	0.3323
0.0000	0.0000	0.0000	0.0000	0.0000	1.1230	0.7750
0.0000	0.0000	0.0000	0.0000	0.0000	3.0410	1.3995
0.0000	0.0000	0.0000	0.0000	0.0000	3.2514	6.5111
0.0000	0.0000	0.0000	0.0000	0.0000	0.8085	2.4474
0.0000	0.0000	0.0000	0.0000	0.0000	0.3843	0.9315
0.0000	0.0000	0.0000	0.0000	0.0000	0.3933	2.5023
0.0000	0.0000	0.0000	0.0000	0.0000	0.4502	0.8945
0.0000	0.0000	0.0000	0.0000	0.0000	0.4018	0.3384
0.0000	0.0000	0.0000	0.0000	0.0000	2.6970	0.7422
0.0000	0.0000	0.0000	0.0000	0.0000	0.6038	2.2942
0.0000	0.0000	0.0000	0.0000	0.0000	0.5568	1.2606
0.0000	0.0000	0.0000	0.0000	0.0000	1.2814	2.1241
0.0000	0.0000	0.0000	0.0000	0.0000	0.5112	1.4838
0.0000	0.0000	0.0000	0.0000	0.0000	1.2932	2.1442
0.0000	0.0000	0.0000	0.0000	0.0000	0.2421	0.6023
0.0000	0.0000	0.0000	0.0000	0.0000	2.5457	2.7918
0.0000	0.0000	0.0000	0.0000	0.0000	1.0951	0.9747
0.0000	0.0000	0.0000	0.0000	0.0000	0.6490	0.3907
0.0000	0.0000	0.0000	0.0000	0.0000	0.7257	1.5204
0.0000	0.0000	0.0000	0.0000	0.0000	0.7041	0.7154
0.0000	0.0000	0.0000	0.0000	0.0000	0.9483	1.0686
0.0000	0.0000	0.0000	0.0000	0.0000	0.7114	0.7316
0.0000	0.0000	0.0000	0.0000	0.0000	2.1881	1.2669
0.0000	0.0000	0.0000	0.0000	0.0000	1.7985	3.2011
0.0000	0.0000	0.0000	0.0000	0.0000	1.0899	0.4901
0.0000	0.0000	0.0000	0.0000	0.0000	0.9265	1.6748
0.0000	0.0000	0.0000	0.0000	0.0000	0.5171	0.8779
0.0000	0.0000	0.0000	0.0000	0.0000	1.2206	0.4998
0.0000	0.0000	0.0000	0.0000	0.0000	0.1558	0.1641
0.0000	0.0000	0.0000	0.0000	0.0000	1.1220	1.2695
0.0000	0.0000	0.0000	0.0000	0.0000	0.3080	0.7765
0.0000	0.0000	0.0000	0.0000</			

0.0000	0.0000	0.0000	0.0000	0.0000	0.5473	0.3453
0.0000	0.0000	0.0000	0.0000	0.0000	0.2921	2.3212
0.0000	0.0000	0.0000	0.0000	0.0000	0.8459	2.4877
0.0000	0.0000	0.0000	0.0000	0.0000	1.8005	2.8746
0.0000	0.0000	0.0000	0.0000	0.0000	1.1161	0.7914
0.0000	0.0000	0.0000	0.0000	0.0000	1.6144	0.6648
0.0000	0.0000	0.0000	0.0000	0.0000	0.4844	2.7198
0.0000	0.0000	0.0000	0.0000	0.0000	0.6391	0.6972
0.0000	0.0000	0.0000	0.0000	0.0000	0.1431	0.3840
0.0000	0.0000	0.0000	0.0000	0.0000	0.6462	0.5219
0.0000	0.0000	0.0000	0.0000	0.0000	1.0749	2.7577
0.0000	0.0000	0.0000	0.0000	0.0000	1.8059	1.3424
0.0000	0.0000	0.0000	0.0000	0.0000	1.2054	2.0824
0.0000	0.0000	0.0000	0.0000	0.0000	1.0625	0.9277
0.0000	0.0000	0.0000	0.0000	0.0000	1.3100	0.8498
0.0000	0.0000	0.0000	0.0000	0.0000	0.2354	0.4022
0.0000	0.0000	0.0000	0.0000	0.0000	0.1665	0.5055
0.0000	0.0000	0.0000	0.0000	0.0000	2.7361	4.1465
0.0000	0.0000	0.0000	0.0000	0.0000	0.9954	0.3919
0.0000	0.0000	0.0000	0.0000	0.0000	0.3077	0.2669
0.0000	0.0000	0.0000	0.0000	0.0000	0.4347	0.6653
0.0000	0.0000	0.0000	0.0000	0.0000	1.6862	0.7062
0.0000	0.0000	0.0000	0.0000	0.0000	1.5509	0.8975
0.0000	0.0000	0.0000	0.0000	0.0000	2.8731	1.7721
0.0000	0.0000	0.0000	0.0000	0.0000	0.7977	1.1339
0.0000	0.0000	0.0000	0.0000	0.0000	0.2367	0.3041
0.0000	0.0000	0.0000	0.0000	0.0000	1.8845	2.6897
0.0000	0.0000	0.0000	0.0000	0.0000	0.7170	0.7487
0.0000	0.0000	0.0000	0.0000	0.0000	0.4697	0.9339
0.0000	0.0000	0.0000	0.0000	0.0000	0.7433	2.4781
0.0000	0.0000	0.0000	0.0000	0.0000	3.3196	0.8931
0.0000	0.0000	0.0000	0.0000	0.0000	0.4429	0.3304
0.0000	0.0000	0.0000	0.0000	0.0000	0.2954	0.8544
0.0000	0.0000	0.0000	0.0000	0.0000	3.4773	0.8874
0.0000	0.0000	0.0000	0.0000	0.0000	0.9119	4.4743
0.0000	0.0000	0.0000	0.0000	0.0000	0.9711	2.9968
0.0000	0.0000	0.0000	0.0000	0.0000	0.3189	0.5537
0.0000	0.0000	0.0000	0.0000	0.0000	2.6030	3.8328
0.0000	0.0000	0.0000	0.0000	0.0000	0.8129	0.5552
0.0000	0.0000	0.0000	0.0000	0.0000	0.3905	0.9198
0.0000	0.0000	0.0000	0.0000	0.0000	0.2959	0.3880
0.0000	0.0000	0.0000	0.0000	0.0000	0.4622	2.7390
0.0000	0.0000	0.0000	0.0000	0.0000	0.5672	2.3260
0.0000	0.0000	0.0000	0.0000	0.0000	1.2561	1.0188
0.0000	0.0000	0.0000	0.0000	0.0000	0.1957	1.0854
0.0000	0.0000	0.0000	0.0000	0.0000	1.7773	1.0179
0.0000	0.0000	0.0000	0.0000	0.0000	0.3102	1.0849
0.0000	0.0000	0.0000	0.0000	0.0000	1.0281	3.3796
0.0000	0.0000	0.0000	0.0000	0.0000	0.7053	0.3545
0.0000	0.0000	0.0000	0.0000	0.0000	2.1739	2.6380
0.0000	0.0000	0.0000	0.0000</			

0.0000	0.0000	0.0000	0.0000	0.0000	0.4399	0.5734
0.0000	0.0000	0.0000	0.0000	0.0000	2.6924	4.2743
0.0000	0.0000	0.0000	0.0000	0.0000	0.2567	0.5004
0.0000	0.0000	0.0000	0.0000	0.0000	0.9601	0.5388
0.0000	0.0000	0.0000	0.0000	0.0000	1.0366	1.6394
0.0000	0.0000	0.0000	0.0000	0.0000	0.1964	0.9650
0.0000	0.0000	0.0000	0.0000	0.0000	1.4489	1.3220
0.0000	0.0000	0.0000	0.0000	0.0000	0.3245	0.6578
0.0000	0.0000	0.0000	0.0000	0.0000	0.3823	0.9842
0.0000	0.0000	0.0000	0.0000	0.0000	0.9213	0.3073
0.0000	0.0000	0.0000	0.0000	0.0000	0.7533	2.8151
0.0000	0.0000	0.0000	0.0000	0.0000	0.5943	1.6959
0.0000	0.0000	0.0000	0.0000	0.0000	0.2326	1.1738
0.0000	0.0000	0.0000	0.0000	0.0000	0.5313	0.6012
0.0000	0.0000	0.0000	0.0000	0.0000	2.1895	1.4008
0.0000	0.0000	0.0000	0.0000	0.0000	4.0854	5.5567
0.0000	0.0000	0.0000	0.0000	0.0000	0.8157	0.7177
0.0000	0.0000	0.0000	0.0000	0.0000	0.5547	1.2802
0.0000	0.0000	0.0000	0.0000	0.0000	0.4327	1.4836
0.0000	0.0000	0.0000	0.0000	0.0000	2.0519	1.7352
0.0000	0.0000	0.0000	0.0000	0.0000	1.4284	0.3182
0.0000	0.0000	0.0000	0.0000	0.0000	0.6217	0.3382
0.0000	0.0000	0.0000	0.0000	0.0000	0.6375	1.3703
0.0000	0.0000	0.0000	0.0000	0.0000	1.7284	0.7023
0.0000	0.0000	0.0000	0.0000	0.0000	0.5474	0.3324
0.0000	0.0000	0.0000	0.0000	0.0000	0.3266	0.3801
0.0000	0.0000	0.0000	0.0000	0.0000	1.0239	1.7067
0.0000	0.0000	0.0000	0.0000	0.0000	0.3578	0.5244

Part 2 Step 2 Identify (and isolate) your independent and dependent

```

maxPi4 = max(pi4);
maxPi5 = max(pi5);
maxPi6 = max(pi6);
maxPi7 = max(pi7);
maxPi8 = max(pi8);
maxPi9 = max(pi9);
maxPi10 = max(pi10);

B = [maxPi4 maxPi5 maxPi6 maxPi7 maxPi8 maxPi9 maxPi10];

disp('maxpi4 maxpi5 maxpi6 maxpi7 maxpi8 maxpi9 maxpi10');
disp(B);

```

```

maxpi4 maxpi5 maxpi6 maxpi7 maxpi8 maxpi9 maxpi10
1.0e+07 *
0.0000 0.0000 0.0000 0.0000 0.0000 4.8240 6.5111

```

Part 2 Step 3 Normalize

```

normPi4 = pi4./maxPi4;
normPi5 = pi5./maxPi5;
normPi6 = pi6./maxPi6;
normPi7 = pi7./maxPi7;
normPi8 = pi8./maxPi8;
normPi9 = pi9./maxPi9;
normPi10 = pi10./maxPi10;

C = [normPi4 normPi5 normPi6 normPi7 normPi8 normPi9 normPi10];
disp('normPi4 normPi5 normPi6 normPi7 normPi8 normPi9 normPi10');
disp(C);

```

```

normPi4 normPi5 normPi6 normPi7 normPi8 normPi9 normPi10
0.4080 0.1884 0.3633 0.1886 0.2124 0.0721 0.3130
0.0991 0.0966 0.2445 0.6226 0.3772 0.1208 0.2434
0.0818 0.1759 0.0727 0.3665 0.2944 0.2666 0.0642
0.7686 0.1198 0.1654 0.7844 0.3626 0.0250 0.0415
0.0861 0.0289 0.0244 0.1691 0.1414 0.0469 0.2045
0.0794 0.0669 0.1322 0.1095 0.3663 0.1674 0.1834
0.2702 0.2600 0.0981 0.2290 0.3417 0.1968 0.2380
0.1870 0.1015 0.0916 0.0639 0.2264 0.1094 0.1034
0.1605 0.2935 0.0584 0.3083 0.0984 0.2749 0.4209
0.0745 0.0299 0.0159 0.1056 0.0550 0.2796 0.2762
0.2175 0.3865 0.1615 0.0520 0.3267 0.7360 0.6648
0.1607 0.1762 0.0567 0.2549 0.1123 0.5284 0.4482
0.0740 0.1165 0.0540 0.3375 0.1128 0.0687 0.1570
0.0838 0.1751 0.0961 0.0884 0.2359 0.1306 0.6655

```

0.2544	0.2397	0.2066	0.5309	0.2820	0.0504	0.0907
0.1484	0.2080	0.2731	0.0826	0.5784	0.1182	0.7169
0.2034	0.0830	0.0980	0.3118	0.2563	0.1396	0.4356
0.5480	0.2285	0.3038	0.1424	0.2607	0.1287	0.0444
0.3211	0.0546	0.1114	0.2365	0.2261	0.1264	0.0510
0.1140	0.3285	0.1115	0.2748	0.1225	0.2328	0.1190
0.1000	0.0847	0.0786	0.3138	0.3609	0.6304	0.2149
0.0382	0.1426	0.0626	0.2806	0.1466	0.6740	1.0000
0.1417	0.0435	0.3263	0.1462	0.7870	0.1676	0.3759
0.1703	0.1350	0.1563	0.3193	0.1053	0.0797	0.1431
0.0989	0.1224	0.0257	0.1103	0.0815	0.0815	0.3843
0.5593	0.0781	0.4781	0.1005	0.2237	0.0933	0.1374
0.5630	0.1922	0.2228	0.0972	0.3372	0.0833	0.0520
0.0421	0.1648	0.1919	0.1362	0.2820	0.5591	0.1140
0.2093	0.2064	0.0421	0.3311	0.1457	0.1252	0.3524
0.1219	0.1604	0.0811	0.4475	0.0860	0.1154	0.1936
0.1387	0.4383	0.0779	0.2558	0.0917	0.2656	0.3262
0.0679	0.1391	0.0432	0.1318	0.4078	0.1060	0.2279
0.1704	0.2651	0.0881	0.5720	0.4627	0.2681	0.3293
0.1637	0.1639	0.0852	1.0000	0.0867	0.0502	0.0925
0.2086	0.3897	0.4165	0.1612	0.5988	0.5277	0.4288
0.2258	0.0421	0.1797	0.1647	0.1930	0.2270	0.1497
0.2326	0.0606	0.0905	0.2807	0.1356	0.1345	0.0600
0.2725	0.2030	0.6731	0.1180	0.5763	0.1504	0.2335
0.1057	0.2445	0.2470	0.3350	0.2757	0.1460	0.1099
0.2309	0.1159	0.0004	0.4066	0.1214	0.1966	0.1641
0.2784	0.0688	0.7677	0.1460	0.2089	0.1475	0.1124
0.0784	0.0854	0.0485	0.1631	0.4820	0.4536	0.1946
0.1671	0.2560	0.5424	0.2169	0.7000	0.3728	0.4916
0.0775	0.3036	0.0517	0.3500	0.1695	0.2259	0.0753
0.0827	0.2937	0.0596	0.3936	0.0940	0.1921	0.2572
0.2172	0.0471	0.0826	0.3220	0.4891	0.1072	0.1348
0.3768	0.1035	0.0672	0.2559	0.2544	0.2530	0.0768
0.3547	0.0538	0.2404	0.2102	0.2212	0.0323	0.0252
0.0302	0.1490	0.0160	0.3881	0.0536	0.2326	0.1950
0.1230	0.0319	0.0414	0.1432	0.1094	0.0639	0.1193
0.5298	0.1014	0.0508	0.4253	0.1012	0.1551	0.0369
0.1333	0.1126	0.0650	0.5011	0.1841	0.0839	0.0503
0.3578	0.2590	0.0389	0.1945	0.1676	0.0476	0.1023
0.3154	0.0759	0.1896	0.1024	0.5109	0.1153	0.1689
0.0857	0.2302	0.1517	0.4126	0.1768	0.1270	0.1727
0.6249	0.0949	0.6729	0.1901	0.3013	0.1012	0.2423
0.0524	0.0660	0.0397	0.1989	0.2158	0.0980	0.2338
0.1554	0.2053	0.5981	0.0476	0.1394	0.3070	0.2383
0.0330	0.0807	0.0555	0.1970	0.4295	0.2133	0.1707
0.2736	0.1366	0.2117	0.1155	0.4122	0.1313	0.0650
0.6368	0.1434	0.0701	0.1214	0.1447	0.0536	0.1427
0.0265	0.0615	0.0178	0.1596	0.1457	0.3337	0.1808
0.2164	0.0654	0.0745	0.3679	0.2502	0.0393	0.0501
0.7803	0.1238	0.0649	0.3736	0.1837	0.1017	0.2491
0.4332	0.1371	0.1093	0.0631	0.1413	0.2126	0.2798
0.1342	0.0904	0.0497	0.3118	0.1519	0.2829	0.7759
0.6192	0.6592	0.1558	0.5331	0.0806	0.0260	0.0301
0.1793	0.0336	0.0592	0.2921	0.2336	0.0619	0.0488
0.1253	0.0292	0.0283	0.2198	0.0706	0.1194	0.2105
0.2178	0.4313	0.5895	0.2190	0.6979	0.1194	0.3862
0.0828	0.0582	0.0485	0.3586	0.2734	0.2254	0.1445
0.1827	0.1649	0.2401	0.1121	0.2384	0.5563	0.1800
0.0495	0.0827	0.0724	0.1421	0.3436	0.3420	0.7783
0.1713	0.1731	0.8094	0.1798	0.5633	0.8040	0.3482
0.1530	0.4104	0.9096	0.0781	0.4316	0.6581	0.7914
0.1962	0.0608	0.0935	0.1365	0.1360	0.1364	0.0779
0.2477	0.0645	0.2306	0.0754	0.5298	0.1919	0.0684
0.5516	0.1945	0.1835	0.3063	0.1590	0.1732	0.1289
0.3842	0.1826	0.0624	0.1268	0.1415	0.0869	0.0412
0.2809	0.1899	0.2628	0.2754	0.4864	0.1613	0.0794
0.1062	0.1041	0.0607	0.1918	0.3722	0.1338	0.1873
0.1289	0.0905	0.0741	0.5392	0.2804	0.4065	0.3864
0.2591	0.6768	0.4982	0.0911	0.3478	0.0848	0.1003
0.1580	0.1305	0.1054	0.2059	0.5640	0.0304	0.0799
0.0902	0.1523	0.1415	0.1459	0.1069	0.2197	0.2199
0.1690	0.0760	0.1966	0.3596	0.4578	0.1356	0.2798
0.2011	0.0666	0.0285	0.1025	0.3076	0.1135	0.0530
0.3442	0.2836	0.8392	0.2085	0.2501	0.0606	0.3565
0.0574	0.1089	0.0218	0.1790	0.1740	0.1753	0.3821
0.1729	0.2184	0.9353	0.0941	0.7406	0.3732	0.4415
0.0730	0.0973	0.1230	0.5621	0.3103	0.2314	0.1215
0.1450	0.1436	0.0667	0.6569	0.1355	0.3347	0.1021
0.0774	0.0846	0.0717	0.2055	0.3402	0.1004	0.4177
0.1528	0.0466	0.0255	0.2204	0.1650	0.1325	0.1071
1.0000	0.6507	0.9757	0.1237	0.6884	0.0297	0.0590
0.2629	0.4261	0.3715	0.3554	0.6335	0.1340	0.0802
0.1351	0.0771	0.2743	0.1588	0.4972	0.2228	0.4235
0.0879	0.0987	0.0437	0.1734	0.2029	0.3744	0.2062
0.4014	0.0888	0.0010	0.2643	0.1536	0.2499	0.3198
0.1839	0.0790	0.0955	0.1278	0.1180	0.2203	0.1424

0.0675	0.3793	0.1307	0.5159	0.3852	0.2716	0.1304
0.5542	0.4944	0.3536	0.1182	0.4006	0.0488	0.0617
0.2430	0.1261	0.0555	0.1480	0.4063	0.0345	0.0776
0.0763	0.0502	0.0217	0.1012	0.0578	0.5672	0.6368
0.5820	0.3100	0.6900	0.2505	1.0000	0.2063	0.0602
0.1249	0.0465	0.0238	0.3780	0.0542	0.0638	0.0410
0.2946	1.0000	0.5977	0.1838	0.2072	0.0901	0.1022
0.3314	0.4943	0.2492	0.0831	0.3424	0.3495	0.1085
0.0603	0.1237	0.0406	0.1145	0.0843	0.3215	0.1378
0.0353	0.1520	0.0095	0.2406	0.0637	0.5956	0.2722
0.2236	0.0531	0.0571	0.3938	0.3904	0.1654	0.1741
0.2520	0.3570	0.0922	0.1747	0.2786	0.0491	0.0467
0.2931	0.3829	0.9433	0.1039	0.4849	0.3906	0.4131
0.3134	0.2487	0.1840	0.1759	0.4116	0.1486	0.1150
0.3178	0.1773	0.7843	0.0804	0.5256	0.0974	0.1434
0.1392	0.2594	0.0661	0.3706	0.1010	0.1541	0.3806
0.0506	0.0422	0.0507	0.1610	0.1547	0.6881	0.1372
0.2687	0.0882	0.2090	0.1922	0.4850	0.0918	0.0507
0.6225	0.1897	0.0356	0.3301	0.0850	0.0612	0.1312
0.0819	0.1603	0.0319	0.3576	0.0997	0.7208	0.1363
0.0500	0.1587	0.1087	0.1283	0.1182	0.1890	0.6872
0.0980	0.0523	0.0561	0.1305	0.0840	0.2013	0.4603
0.5045	0.1563	0.0585	0.1810	0.3594	0.0661	0.0850
0.2028	0.5061	0.1251	0.1317	0.3116	0.5396	0.5887
0.1583	0.0694	0.0159	0.4270	0.0888	0.1685	0.0853
0.2255	0.0891	0.1495	0.1820	0.3439	0.0810	0.1413
0.1868	0.0963	0.0238	0.2472	0.2204	0.0613	0.0596
0.2518	0.3455	0.2099	0.1336	0.5466	0.0958	0.4207
0.3400	0.2287	0.4527	0.1240	0.5777	0.1176	0.3572
0.0622	0.0872	0.0913	0.1711	0.0829	0.2604	0.1565
0.1616	0.1205	0.0945	0.2640	0.1360	0.0406	0.1667
0.1223	0.1787	0.2235	0.1238	0.2556	0.3684	0.1563
0.7074	0.1402	0.3384	0.1448	0.5789	0.0643	0.1666
0.0883	0.0865	0.3814	0.2286	0.1574	0.2131	0.5191
0.1529	0.0280	0.0559	0.5708	0.2016	0.1462	0.0545
0.0960	0.1320	0.1287	0.0743	0.0966	0.4506	0.4052
0.0352	0.0312	0.0476	0.8297	0.1042	0.5538	0.1103
0.1506	0.0858	0.0610	0.1057	0.1063	0.1487	0.1468
0.8221	0.3720	0.5461	0.3667	0.5499	0.0404	0.0462
0.2341	0.3580	0.3974	0.1628	0.3368	0.3183	0.4860
0.0738	0.2474	0.0748	0.2437	0.2400	0.6239	0.1951
0.3335	0.1021	0.1085	0.0989	0.2669	0.1562	0.0706
0.1238	0.0772	0.0210	0.6948	0.0936	0.0601	0.0930
0.2570	0.2996	0.2432	0.4993	0.2555	0.3616	0.3272
0.4997	0.1491	1.0000	0.0979	0.5592	0.0463	0.1039
0.0315	0.1489	0.0438	0.5739	0.1209	1.0000	0.8361
0.2185	0.1864	0.0687	0.1336	0.1151	0.1336	0.1854
0.1532	0.1122	0.4196	0.0690	0.6832	0.5550	0.4319
0.0675	0.3527	0.0563	0.7643	0.2505	0.2347	0.1991
0.3334	0.1344	0.2217	0.5220	0.2802	0.1376	0.1128
0.1027	0.2712	0.2063	0.1586	0.2880	0.1421	0.2350
0.2410	0.2684	0.0380	0.4304	0.1825	0.1170	0.0474
0.1394	0.3527	0.3087	0.1868	0.2369	0.2380	0.2785
0.3759	0.1278	0.0751	0.2093	0.2679	0.2460	0.3419
0.1364	0.1216	0.2151	0.0933	0.7015	0.2518	0.3157
0.0649	0.0586	0.0397	0.1054	0.1271	0.1939	0.6897
0.0981	0.0258	0.0247	0.3121	0.0672	0.1257	0.0687
0.1561	0.1326	0.0780	0.1623	0.4342	0.1524	0.3625
0.1635	0.4313	0.0739	0.2119	0.0664	0.1356	0.1155
0.1199	0.2174	0.0834	0.1724	0.1412	0.0720	0.0902
0.8287	0.1208	0.1045	0.2704	0.1712	0.1112	0.0533
0.3610	0.1469	0.3827	0.3634	0.5518	0.1419	0.2664
0.1992	0.0226	0.1021	0.2296	0.2082	0.0353	0.0566
0.1401	0.4082	0.6807	0.1970	0.2445	0.4119	0.4800
0.3478	0.0581	0.0539	0.2090	0.1180	0.1111	0.1370
0.5347	0.5337	0.2020	0.1587	0.4410	0.2159	0.2087
0.1285	0.0591	0.1134	0.2761	0.0855	0.5902	0.4113
0.0573	0.1295	0.1016	0.3397	0.3518	0.6547	0.3921
0.3876	0.4588	0.4842	0.2281	0.2432	0.0858	0.4776
0.5632	0.3070	0.5382	0.1184	0.7526	0.1550	0.0519
0.1173	0.1297	0.2140	0.3756	0.1293	0.1338	0.5689
0.4107	0.0849	0.3846	0.0786	0.3173	0.1050	0.0535
0.4086	0.1876	0.5242	0.1254	0.2236	0.0912	0.0881
0.0540	0.1293	0.0630	0.1125	0.2400	0.5581	0.6565
0.1668	0.0529	0.0159	0.1397	0.1106	0.0532	0.0769
0.3383	0.6460	0.2420	0.1344	0.1911	0.1990	0.0828
0.3116	0.0802	0.3730	0.2026	0.7399	0.2149	0.2518
0.7352	0.2517	0.5527	0.1664	0.1389	0.0407	0.1482
0.4094	0.1073	0.9306	0.1161	0.1624	0.3004	0.2030
0.1192	0.0796	0.0524	0.3542	0.1490	0.0673	0.1010
0.1718	0.0170	0.0321	0.1048	0.1775	0.0792	0.1512
0.8633	0.2467	0.0567	0.2562	0.2200	0.1910	0.0472
0.0649	0.0379	0.0513	0.3614	0.3444	0.1562	0.4324
0.2293	0.8722	0.2983	0.1956	0.2292	0.1232	0.2605
0.4098	0.1757	0.0465	0.3845	0.0939	0.0482	0.1803
0.7768	0.0589	0.1311	0.1704	0.4363	0.1101	0.0923

0.2014	0.4596	0.5542	0.1555	0.1316	0.4539	0.2151
0.0825	0.0517	0.5128	0.0966	0.4973	0.8469	0.8534
0.1832	0.1147	0.1122	0.6020	0.2932	0.1691	0.1102
0.1333	0.1264	0.9073	0.1413	0.7033	0.1150	0.1966
0.1133	0.0360	0.0141	0.6971	0.0489	0.0897	0.2279
0.0521	0.0379	0.0288	0.3443	0.0789	0.4253	0.2665
0.0930	0.0247	0.0164	0.1271	0.1487	0.2961	0.0489
0.2928	0.0324	0.1079	0.0956	0.5674	0.1289	0.0519
0.0358	0.1146	0.0134	0.1409	0.0596	0.1322	0.2105
0.1377	0.1088	0.1048	0.7231	0.0962	0.3583	0.1079
0.3755	0.0393	0.0546	0.3030	0.1303	0.1135	0.0511
0.5754	0.0447	0.0866	0.2355	0.2155	0.0677	0.0584
0.2634	0.1475	0.0632	0.4136	0.1032	0.2122	0.2621
0.1302	0.1944	0.0294	0.2081	0.0790	0.0742	0.0805

Part 2 step 4 Multilinear

```
IndepData = [pi4 pi6 pi7 pi8 pi9 pi10];
DepData = pi5;
linear = fitlm(IndepData,DepData,"linear");
linearfit = linear.Coefficients.Estimate;
```

```
disp(linearfit);
```

```
1.0e-05 *

    0.0003
    0.2146
    0.8450
    0.0606
   -0.0000
    0.0000
    0.0000
```

Part 2 step 5 exponential fit

```
initialFit = fitlm(log(IndepData),DepData,"linear");
initialGuess = [exp(initialFit.Coefficients.Estimate(1)); initialFit.Coefficients.Estimate(2:end)];
expfunction = @(b,x)exp(b(1))*exp(x*b(2:end));
Exponential = fitnlm(IndepData,DepData,expfunction,initialGuess);
ExpoFit = Exponential.Coefficients.Estimate;
disp('exponential fit');
```

```
disp(ExpoFit);
```

Warning: Some columns of the Jacobian are effectively zero at the solution, indicating that the model is insensitive to some of its parameters. That may be because those parameters are not present in the model, or otherwise do not affect the predicted values. It may also be due to numerical underflow in the model function, which can sometimes be avoided by choosing better initial parameter values, or by rescaling or recentering. Parameter estimates may be unreliable.

```
exponential fit
   -19.3219
    311.0037
    821.2725
     79.6478
     -0.0029
     0.0000
     0.0000
```

Part 2 step 6 power fit

```
initialpfit = fitlm(log(IndepData), log(DepData), "linear");
initialpGuess = [exp(initialpfit.Coefficients.Estimate(1)); initialpfit.Coefficients.Estimate(2:end)];
powerfunction = @(b,x)exp(b(1)) * prod(x.^(b(2:end)'), 2);
power = fitnlm(IndepData,DepData,powerfunction,initialpGuess);
powerfit = power.Coefficients.Estimate;
disp('powerfit');
```

```
disp(powerfit);
```

```
powerfit
   -13.8664
     0.1454
     0.3569
     0.1205
```

```
-0.1546
0.0266
0.0092
```

Part 2 step 7 nonlinear fitting

```
linearfunction = @(b,x)b(1)+sum(x.*b(2:end)',2);
nonlinear = fitnlm(IndepData,DepData,linearfunction,linearfit);
nonlinearfit = nonlinear.Coefficients.Estimate;
disp('nonlinearfit')
disp(nonlinearfit);
```

Warning: The Jacobian at the solution is ill-conditioned, and some model parameters may not be estimated well (they are not identifiable). Use caution in making predictions.

```
nonlinearfit
1.0e-05 *

0.0003
0.2146
0.8450
0.0606
-0.0000
0.0000
0.0000
```

Check

```
disp(linearfit);
disp(nonlinearfit);
```

```
1.0e-05 *

0.0003
0.2146
0.8450
0.0606
-0.0000
0.0000
0.0000

1.0e-05 *

0.0003
0.2146
0.8450
0.0606
-0.0000
0.0000
0.0000
```

Part 2 step 8 rescale

```
idepscale = [maxPi4, maxPi6,maxPi7,maxPi8,maxPi9,maxPi10];
depScale = maxPi5;

%linear
linearscale = linear.Coefficients.Estimate; %rescale multilinear
linearscale(1) = linearscale(1)* depScale;
linearscale(2:end) = linearscale(2:end)./idepscale'; %transpose

disp(linearscale);

%power @(b,x)b(1) * prod(x.^(b(2:end)')), 2)
powerscale = power.Coefficients.Estimate;
powerscale(1) = powerscale(1) * depScale * prod(depScale .^ powerscale(2:end));

disp(powerscale);

%exponential
expscale = Exponential.Coefficients.Estimate;
expscale(1) = expscale(1) * depScale;
expscale(2:end) = expscale(2:end) ./ idepscale';

disp(expscale);
```



```

0.0000
0.0010
0.0078
0.0002
-0.0000
0.0000
0.0000

-0.0000
0.1454
0.3569
0.1205
-0.1546
0.0266
0.0092

1.0e+05 *

-0.0000
1.4044
7.6091
0.3266
-0.0000
0.0000
0.0000

```

Part 3

```

linearAIC = linear.ModelCriterion.AIC;%-7.05723e+3
expoAIC = Exponential.ModelCriterion.AIC;%smallest -7.0522e+3
powerAIC = power.ModelCriterion.AIC; %~-7.0680e+3

D = [linearAIC expoAIC powerAIC];
disp('linearAIC expoAIC powerAIC');
disp(D);

% coefficients and pvalue
expoCoef = Exponential.Coefficients.Estimate;
expoPvalue = Exponential.Coefficients.pValue;

disp('Pvalue');
disp(expoPvalue);

% Bonfeeroni aphpa = 0.01
alpha = 0.01;
numIndep = length(expoCoef) - 1; %6
newAlpha = alpha / numIndep; % 0.0017
critical = find(expoPvalue(2:end) < newAlpha); %first three pi groups are significant and based on p-value the first three group is smaller than 0.001
disp('Threshold');
disp(newAlpha);
disp('Critical groups');
disp(critical);
% pi5 is dependent, pi4 pi6 pi7 is significant dependent, the first zero is
% C0

```

```

linearAIC expoAIC powerAIC
1.0e+03 *

-7.0572 -7.0522 -7.0680

Pvalue
0
0
0
0
0.4665
0.8034
0.4362

Threshold
0.0017

Critical groups
1
2
3

```

Part 4

Creating 10x scales is feasible,engths are multiplied by 10, areas by

```
%100, and volumes by 1000, to maintaince the independent group the model will accurately replicate the system's behavior at the new scale
%wind tunnel is as good example as a controlled system
%use first observation as example,area:1.09523914600291e-8, velocity:0.59393ms, Mass: 4.33736239010938e-4
%power 2.12187274812354×10^5, force: 46.3306886748899, Mass, power, and force scale with volume and related properties
%Velocity scales directly with the length scale,density and thermal conductivity, no scaling factor is applied unless specified, so they remain unchan
```
