REFERENCES

Alejano, L. R., Alonso, E., Rodriguez-Dono, A., & Fernandez-Manin, G. (2010). Application of the convergence-confinement method to tunnels in rock masses exhibiting Hoek–Brown strain-softening behaviour. *International Journal of Rock Mechanics and Mining Sciences*, *47*(1), 150–160.

Amitrano, D., & Helmstetter, A. (2006). Brittle creep, damage, and time to failure in rocks. *Journal of Geophysical Research: Solid Earth*, *111*(B11).

Bairagi, V., & Munot, M. V. (2019). *Research methodology: A practical and scientific approach*. CRC Press.

Baranwal, A. K., Bhaskar, A., Mishra, S., Satyanarayana, G. v, & Kumar, T. (n.d.-a). *Identification of suitable parameters for mine slope monitoring by Movement and Surveying RADAR: A case study at Sukinda Chromite mine, M/s Tata steel*.

Baranwal, A. K., Bhaskar, A., Mishra, S., Satyanarayana, G. v, & Kumar, T. (n.d.-b). *Identification of suitable parameters for mine slope monitoring by Movement and Surveying RADAR: A case study at Sukinda Chromite mine, M/s Tata steel*.

Bennett, J. D. (1982). *Relationship between work place and worker characteristics and severity of injuries in US underground bituminous coal mines, 1975-1981*. https://www.osti.gov/biblio/5328604

Bieniawski, Z. T. (1979). The geomechanics classification in rock engineering applications. *4th ISRM Congress*.

Brady, B. H. G., & Brown, E. T. (2006). *Rock mechanics: for underground mining*. Springer science & business media.

Broadbent, C. D., & Zavodni, Z. M. (1982). Influence of rock structure on stability, Stability in Surface Mining. *Society of Mining Engineers*, *3*, 30–35.

Brox, D., & Newcomen, W. (2003). Utilizing strain criteria to predict highwall stability performance. *10th ISRM Congress*.

Bui, X. N., Nguyen, H., Choi, Y., Nguyen-Thoi, T., Zhou, J., & Dou, J. (2020). Prediction of slope failure in open-pit mines using a novel hybrid artificial intelligence model based on decision tree and evolution algorithm. *Scientific Reports*, *10*(1), 1–17. https://doi.org/10.1038/s41598-020-66904-y

Cabrejo, A., & Harries, N. (2012). *Effective Slope Monitoring for Open Cut Coal Mines*.

Cahill M, J. & L. (2006). Ground control at Leinster nickel operations. *Journal of the Southern African Institute of Mining and Metallurgy*, *106*(7), 471–478.

Cahill M, J. & L. (2006). Ground control at Leinster nickel operations. *Journal of the Southern African Institute of Mining and Metallurgy*, *106*(7), 471–478.

Calder, P. N., & Blackwell, G. (1980). Investigation of a complex rock slope displacement at Brenda Mines. *CIM Bull.;(Canada)*, *73*(820).

Carlà, T., Farina, P., Intrieri, E., Botsialas, K., & Casagli, N. (2017). On the monitoring and early-warning of brittle slope failures in hard rock masses: Examples from an open-pit mine. *Engineering Geology*, *228*, 71–81.

Carlà, T., Farina, P., Intrieri, E., Ketizmen, H., & Casagli, N. (2018). Integration of ground-based radar and satellite InSAR data for the analysis of an unexpected slope failure in an open-pit mine. *Engineering Geology*, *235*, 39–52.

Carlà, T., Intrieri, E., di Traglia, F., Nolesini, T., Gigli, G., & Casagli, N. (2017). Guidelines on the use of inverse velocity method as a tool for setting alarm thresholds and forecasting landslides and structure collapses. *Landslides*, *14*, 517–534.

Carlà, T., Intrieri, E., Farina, P., & Casagli, N. (2017a). A new method to identify impending failure in rock slopes. *International Journal of Rock Mechanics and Mining Sciences*, *93*, 76–81.

Carlà, T., Intrieri, E., Farina, P., & Casagli, N. (2017b). A new method to identify impending failure in rock slopes. *International Journal of Rock Mechanics and Mining Sciences*, *93*(January), 76–81. https://doi.org/10.1016/j.ijrmms.2017.01.015

Chung, C.-C., & Lin, C.-P. (2011). High concentration suspended sediment measurements using time domain reflectometry. *Journal of Hydrology*, *401*(1–2), 134–144.

Crosta, G. B., & Agliardi, F. (2003). Failure forecast for large rock slides by surface displacement measurements. *Canadian Geotechnical Journal*, *40*(1), 176–191.

Cruden, D. M., & Masoumzadeh, S. (1987). Accelerating creep of the slopes of a coal mine. *Rock Mechanics and Rock Engineering*, *20*(2), 123–135.

Dash, A. K. (2019). Analysis of accidents due to slope failure in Indian opencast coal mines. *Current Science*, *117*(2), 304–308.

Day, A. P., & Seery, J. M. (2007). Monitoring of a Large Wall Failure at Tom Price Iron Orre Mine. *Slope Stability 2007: Proceedings of the 2007 International Symposium on Rock Slope Stability in Open Pit Mining and Civil Engineering*, 333–340.

Dick, G. J., Eberhardt, E., Cabrejo-Liévano, A. G., Stead, D., & Rose, N. D. (2015a). Development of an early-warning time-of-failure analysis methodology for open-pit mine slopes utilizing ground-based slope stability radar monitoring data. *Canadian Geotechnical Journal*, *52*(4), 515–529.

Dick, G. J., Eberhardt, E., Cabrejo-Liévano, A. G., Stead, D., & Rose, N. D. (2015b). Development of an early-warning time-of-failure analysis methodology for open-pit mine slopes utilizing ground-based slope stability radar monitoring data. *Canadian Geotechnical Journal*, *52*(4), 515–529. https://doi.org/10.1139/cgj-2014-0028

Dok, A., Fukuoka, H., Katsumi, T., & Inui, T. (n.d.). *Tertiary Creep Reproduction in Back-Pressure-Controlled Test to Understand the Mechanism and Final Failure Time of Rainfall–Induced Landslides*.

Dowding, C., Su, M. B., & O'Connor, K. (1989). Measurement of rock mass deformation with grouted coaxial antenna cables. *Rock Mechanics and Rock Engineering*, *22*, 1–23.

Eberhardt, E., Stead, D., & Coggan, J. S. (2004). Numerical analysis of initiation and progressive failure in natural rock slopes—the 1991 Randa rockslide. *International Journal of Rock Mechanics and Mining Sciences*, *41*(1), 69–87.

Frizzell, M. R., Nguyen, N. M., Goldberg, L. H., Parikh, S. A., & Sinai, M. J. (2020). Heparin-induced bullous hemorrhagic dermatosis: A report of 3 cases. *JAAD Case Reports*, *6*(10), 1065–1068.

Fukuzono, T. (1984). A method for predicting the failure time of a sandy soil slope using the inverse number of velocity. *Proceedings of the 23rd Meeting of Japan Landslide Society*, 80–81.

FUKUZONO, T. (1985a). A method to predict the time of slope failure caused by rainfall using the inverse number of velocity of surface displacement. *Landslides*, *22*(2), 8-13_1.

FUKUZONO, T. (1985b). A method to predict the time of slope failure caused by rainfall using the inverse number of velocity of surface displacement. *Landslides*, *22*(2), 8-13_1.

FUKUZONO, T. (1985c). A new method for predicting the failure time of a slope. *Proceedings of 4^< Th> International Conference and Field Workshop on Landslide., 1985*, 145–150.

Girard, J. M. (2001a). Assessing and Monitoring Open Pit Mine Highwalls. *Proceedings of the 32nd Annual Institute on Mining Health, Safety and Research*, 159–171.

Girard, J. M. (2001b). *Assessing and monitoring open pit mine highwalls*.

Girard, J. M., & McHugh, E. (2000). *Detecting problems with mine slope stability*.

Glastonbury, J., & Fell, R. (2002). *Report on the analysis of the deformation behaviour of excavated rock slopes*. University of New South Wales, School of Civil and Environmental Engineering.

Haala, N., Peter, M., Kremer, J., & Hunter, G. (2008). Mobile LiDAR mapping for 3D point cloud collection in urban areas—A performance test. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci*, *37*, 1119–1127.

Harries, N., & Holmstrom, M. (2007). The use of slope stability radar in monitoring slopes and managing slope instability hazards. *Queensland Roads*, *4*.

Harries, N. J., & Cabrejo, A. G. L. (2010a). Deformation Response of Coal Mine Slopes-Implications For Slope Hazard Management Using Evacuation Based On Slope Monitoring. *44th US Rock Mechanics Symposium and 5th US-Canada Rock Mechanics Symposium*.

Harries, N. J., & Cabrejo, A. G. L. (2010b). Deformation Response of Coal Mine Slopes-Implications For Slope Hazard Management Using Evacuation Based On Slope

Monitoring. *44th US Rock Mechanics Symposium and 5th US-Canada Rock Mechanics Symposium*.

Harries, N., Noon, D., & Rowley, K. (2006a). Case Studies of Slope Stability Radar in Open Cut Mines. *International Symposium on Stability of Rock Slopes in Open Pit Mining and Civil Engineering, The South African Institute of Mining and Metallurgy, Symposium Series 44*, 335–342.

Harries, N., Noon, D., & Rowley, K. (2006b). Case studies of slope stability radar used in open cut mines. *Stability of Rock Slopes in Open Pit Mining and Civil Engineering Situations*, 335–342.

Harries, N., Noon, D., & Rowley, K. (2006c). Case studies of slope stability radar used in open cut mines. *Stability of Rock Slopes in Open Pit Mining and Civil Engineering Situations*, 335–342.

Hebblewhite, M., Munro, R. H., & Merrill, E. H. (2009). Trophic consequences of postfire logging in a wolf–ungulate system. *Forest Ecology and Management*, *257*(3), 1053–1062.

Heim, A. (1932). *Bergsturz und menschenleben* (Issue 20). Fretz & Wasmuth.

Hoek, E., & Bray, J. W. (1981). *Rock Slope Engineering. London: The Institution of Mining and Metallurgy*. Maney Publishing.

Hoek, E., Rippere, K. H., & Stacey, P. F. (2000). Large scale slope designs–a review of the state of the art. *Slope Stability in Surface Mining*, 3–10.

Hwang, S. G., Guevarra, I. F., & Yu, B. O. (2009). Slope failure prediction using a decision tree: A case of engineered slopes in South Korea. *Engineering Geology*, *104*(1–2), 126–134. https://doi.org/10.1016/j.enggeo.2008.09.004

Intrieri, E., Carlà, T., & Gigli, G. (2019). Forecasting the time of failure of landslides at slope-scale: A literature review. *Earth-Science Reviews*, *193*, 333–349.

Intrieri, E., & Gigli, G. (2016). Landslide forecasting and factors influencing predictability. *Natural Hazards and Earth System Sciences*, *16*(12), 2501–2510.

Jäggli, M. (1928). *Il cataclima tellurico al Motto d'Arbino*. Grassi.

Juang, C. H., Lee, D. H., & Sheu, C. (1992). Mapping slope failure potential using fuzzy sets. *Journal of Geotechnical Engineering*, *118*(3), 475–494.

Kane, W. F., & Beck, T. J. (2000). Instrumentation practice for slope monitoring. *Engineering Geology Practice in Northern California. Association of Engineering Geologists Sacramento and San Francisco Sections*.

Kasmer, O., & Ulusay, R. (2006). Stability of spoil piles at two coal mines in Turkey: geotechnical characterization and design considerations. *Environmental & Engineering Geoscience*, *12*(4), 337–352.

Kayesa, G. (2006a). Prediction of slope failure at Letlhakane Mine with the Geomos Slope Monitoring System. *Proceedings of the International Symposium on Stability of Rock Slopes in Open Pit Mining and Civil Engineering Situations, Series S44*.

Kayesa, G. (2006b). Prediction of slope failure at Letlhakane mine with the geomos slope monitoring system. *International Symposium on Stability of Rock Slopes in Open Pit Mining and Civil Engineering*, 605–622.

Kemeny, J. (2003). The time-dependent reduction of sliding cohesion due to rock bridges along discontinuities: a fracture mechanics approach. *Rock Mechanics and Rock Engineering*, *36*, 27–38.

Kothari, U. C., & Momayez, M. (2018). New approaches to monitoring, analyzing and predicting slope instabilities. *Journal of Geology and Mining Research*, *10*(1), 1–14.

Krause, T. R., Seymour, K. J., & Sloat, K. C. M. (1999). Long-term evaluation of a behavior-based method for improving safety performance: a meta-analysis of 73 interrupted time-series replications. *Safety Science*, *32*(1), 1–18. https://doi.org/https://doi.org/10.1016/S0925-7535(99)00007-7

Lajtai, E. Z., & Bielus, L. P. (1986). Stress corrosion cracking of Lac du Bonnet granite in tension and compression. *Rock Mechanics and Rock Engineering*, *19*, 71–87.

Lajtai, E. Z., & Schmidtke, R. H. (1986). Delayed failure in rock loaded in uniaxial compression. *Rock Mechanics and Rock Engineering*, *19*, 11–25.

Liang, W., Zhao, Y., Xu, S., & Dusseault, M. B. (2008). Dissolution and seepage coupling effect on transport and mechanical properties of glauberite salt rock. *Transport in Porous Media*, *74*, 185–199.

Lin, H.-M., Chang, S.-K., Wu, J.-H., & Juang, C. H. (2009). Neural network-based model for assessing failure potential of highway slopes in the Alishan, Taiwan Area: Pre- and post-earthquake investigation. *Engineering Geology*, *104*(3–4), 280–289.

Little, M. J. (2006a). Slope monitoring strategy at PPRust open pit operation. *Proceedings of the International Symposium on Stability of Rock Slopes in Open Pit Mining and Civil Engineering*, 211–230.

Little, M. J. (2006b). Slope monitoring strategy at PPRust open pit operation. *Proceedings of the International Symposium on Stability of Rock Slopes in Open Pit Mining and Civil Engineering*, 211–230.

Little, M. J. (2006c). The benefit to open pit rock slope design of geotechnical databases. *The South African Institute of Mining and Metallurgy International Symposium on Stability of Rock Slopes in Open Pit Mining and Civil Engineering.-P*, 97–116.

Main, I. G. (2000). A damage mechanics model for power-law creep and earthquake aftershock and foreshock sequences. *Geophysical Journal International*, *142*(1), 151–161.

McHugh, E. L., Long, D. G., & Sabine, C. (1900). *Applications of ground-based radar to mine slope monitoring*.

Mencl, V. (1966). Mechanics of landslides with non-circular slip surfaces with special reference to the Vaiont slide. *Géotechnique*, *16*(4), 329–337.

Mercer, K. G. (2006). *Investigation into the time dependent deformation behaviour and failure mechanisms of unsupported rock slopes based on the interpretation of observed deformation behaviour*. University of the Witwatersrand.

Merrill, I., & SKOLNIK, E. D. (1990). *RADAR HANDBOOK.* MCGRAW HILL BOOK Company.

Moretto, S., Bozzano, F., Esposito, C., Mazzanti, P., & Rocca, A. (2017). Assessment of landslide pre-failure monitoring and forecasting using satellite SAR interferometry. *Geosciences*, *7*(2), 36.

Mufundirwa, A., Fujii, Y., & Kodama, J. (2010a). A new practical method for prediction of geomechanical failure-time. *International Journal of Rock Mechanics and Mining Sciences*, *47*(7), 1079–1090.

Mufundirwa, A., Fujii, Y., & Kodama, J. (2010b). A new practical method for prediction of geomechanical failure-time. *International Journal of Rock Mechanics and Mining Sciences*, *47*(7), 1079–1090. https://doi.org/https://doi.org/10.1016/j.ijrmms.2010.07.001

Okagbue, C. O. (1987). Stability of waste spoils in an area strip mine—geological and geotechnical considerations. *Earth Surface Processes and Landforms*, *12*(3), 289–300.

Okubo, S., Fukui, K., & Nishimatsu, Y. (1997). Local safety factor applicable to wide range of failure criteria. *Rock Mechanics and Rock Engineering*, *30*(4), 223–227.

Osasan, K. S. (2012). *Open-cast mine slope deformation and failure mechanisms interpreted from slope radar monitoring*.

Peltzer, G., Rosen, P., Rogez, F., Werner, C., Fieldings, E., & Hudnut, K. (1997). Crustal deformation studies using synthetic aperture radar (SAR) interferometry. *ERS SAR Interferometry*, *406*, 17.

Petley, D. N. (2004). The evolution of slope failures: mechanisms of rupture propagation. *Natural Hazards and Earth System Sciences*, *4*(1), 147–152.

Petley, D. N., Bulmer, M. H., & Murphy, W. (2002). Patterns of movement in rotational and translational landslides. *Geology*, *30*(8), 719–722.

Petley, D. N., Higuchi, T., Petley, D. J., Bulmer, M. H., & Carey, J. (2005). Development of progressive landslide failure in cohesive materials. *Geology*, *33*(3), 201–204.

Read, J., & Stacey, P. (2009a). *Guidelines for open pit slope design*.

Read, J., & Stacey, P. (2009b). *Guidelines for open pit slope design*.

Reeves, B. A., Stickley, G. F., Noon, D. A., & Longstaff, I. D. (2000). Developments in monitoring mine slope stability using radar interferometry. *IGARSS 2000. IEEE 2000 International Geoscience and Remote Sensing Symposium. Taking the Pulse of the Planet: The Role of Remote Sensing in Managing the Environment. Proceedings (Cat. No. 00CH37120)*, *5*, 2325–2327.

Reeves, B., Noon, D., Stickley, G., & Longstaff, D. (1997). Monitoring rock slope deformation by radar interferometry. *Proceedings of the Workshop on Applications of Radio Science WARS'97*, 119–123.

Rose, N. D., & Hungr, O. (2007a). Forecasting potential rock slope failure in open pit mines using the inverse-velocity method. *International Journal of Rock Mechanics and Mining Sciences*, *44*(2), 308–320.

Rose, N. D., & Hungr, O. (2007b). Forecasting potential rock slope failure in open pit mines using the inverse-velocity method-case examples. *1st Canada-US Rock Mechanics Symposium*.

Saito, M. (1965). Forecasting the time of occurrence of a slope failure. *Proc. 6 Th Int. Conf. Soil Mechanics and Foundation Eng.*, 537–541.

Saito, M. (1969). Forecasting time of slope failure by tertiary creep. *Proceedings of the 7th International Conference on Soil Mechanics and Foundation Engineering, Mexico City, Mexico*, *2*, 677–683.

Saito, M., & Uezawa, H. (1961). Failure of soil due to creep, paper presented at Fifth International Conference on Soil Mechanics and Foundation Engineering. *Int. Soc. of Soil Mech. and Geotech. Eng., Montreal, Que., Canada*.

Sakellariou, M. G., & Ferentinou, M. D. (2005). A study of slope stability prediction using neural networks. *Geotechnical & Geological Engineering*, *23*(4), 419–445.

Sharon, R., & Eberhardt, E. (2020). *Guidelines for Slope Performance Monitoring*. CSIRO PUBLISHING.

Simmons, J. & S. (2006). Composite failure mechanisms in coal measures' rock masses-myths and reality. *Journal of the Southern African Institute of Mining and Metallurgy*, *106*(7), 459–469.

Singhroy, V., & Molch, K. (2004). Characterizing and monitoring rockslides from SAR techniques. *Advances in Space Research*, *33*(3), 290–295.

Sjöberg, J. (1996). *Large scale slope stability in open pit mining: a review*.

Sjöberg, J., & Norstrom, U. (2000). Failure mechanisms for high slopes in hard rock. *Slope Stability in Surface Mining*, 71–80.

Stacey, T. R. (2007). Slope stability in high stress and hard rock conditions. *Slope Stability 2007: Proceedings of the 2007 International Symposium on Rock Slope Stability in Open Pit Mining and Civil Engineering*, 187–200.

Ulusay, R., Gökçeoğlu, C., Sönmez, H., & Tuncay, E. (2001). Causes, mechanism and environmental impacts of instabilities at Himmetoğlu coal mine and possible remedial measures. *Environmental Geology*, *40*(6).

Upasna, C. K., & Moe, M. (2018). New approaches to monitoring, analyzing and predicting slope instabilities. *Journal of Geology and Mining Research*, *10*(1), 1–14. https://doi.org/10.5897/jgmr2017.0272

Varnes, D. J. (1983). Time-deformation relations in creep to failure of earth materials. *Proceedings, 7^< Th> Southeast Asian Geotechnical Conference, Hong Kong, 1983*, *2*, 107–130.

Vaziri, A., Moore, L., & Ali, H. (2010). Monitoring systems for warning impending failures in slopes and open pit mines. *Natural Hazards*, *55*, 501–512.

Venter, J., Kuzmanovic, A., & Wessels, S. D. N. (2013). An evaluation of the CUSUM and inverse velocity methods of failure prediction based on two open pit instabilities in the Pilbara. *Proceedings of the 2013 International Symposium on Slope Stability in Open Pit Mining and Civil Engineering*, 1061–1076.

Voight, B., & Kennedy, B. A. (1979). Slope failure of 1967–1969, Chuquicamata mine, Chile. In *Developments in Geotechnical Engineering* (Vol. 14, pp. 595–632). Elsevier.

Wang, S. (1981). On the mechanism and process of slope deformation in an open pit mine. *Rock Mechanics*, *13*, 145–156.

Way, L. W., Stewart, L., Gantert, W., Liu, K., Lee, C. M., Whang, K., & Hunter, J. G. (2003). Causes and prevention of laparoscopic bile duct injuries: analysis of 252 cases from a human factors and cognitive psychology perspective. *Annals of Surgery*, *237*(4), 460–469. https://doi.org/10.1097/01.SLA.0000060680.92690.E9

Wyllie, D. C., & Mah, C. (2004). *Rock slope engineering*. CRC Press.

Xu, N., Tang, C., Li, H., Dai, F., Ma, K., Shao, J., & Wu, J. (2012). Excavation-induced microseismicity: microseismic monitoring and numerical simulation. *Journal of Zhejiang University SCIENCE A*, *13*(6), 445–460.

Zavodni, Z. M. (2000). Time-dependent movements of open-pit slopes. *Slope Stability in Surface Mining*, 81–87.

Zavodni, Z. M., & Broadbent, C. D. (1978). Slope failure kinematics. *19th US Symposium on Rock Mechanics (USRMS)*.

Zavodni, Z. M., & Broadbent, C. D. (1980). *Slope Failure Kinematics*. Bulletin of the Canadian Institute of Mining.

APPENDIX-A

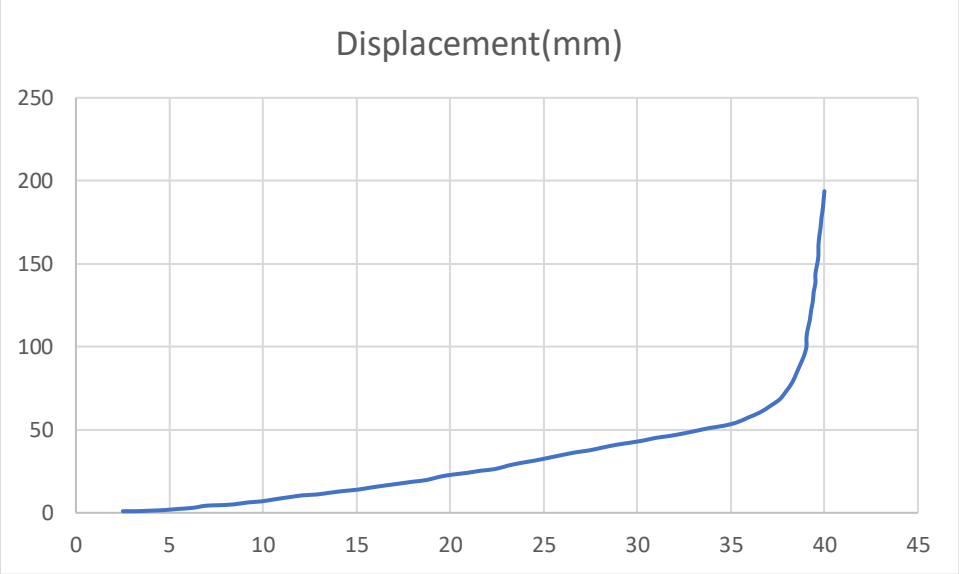
The datasets have been shared in the form of graphs due to space constraints, as the data has thousands of rows for each monitoring set generated using the Slope Stability Radar (SSR). Therefore, only two of the digitized datasets have been shared below and all the other data sets either raw data from the South Eastern Coalfields Limited mines or digitized datasets have been presented through graphs.

Digitized Data: For all the digitized datasets, displacement(mm) data is tabulated with respect to time, where time is measured as number of days.

Dataset 1: Australian Mine (Glastonbury and Fell 2002)

TIME(Days)	Displacement(mm)
2.508930878	0.982519997
3.225704803	0.979887218
3.987351145	1.307668132
4.726598302	1.635531319
5.421121136	2.294137296
6.227639895	2.952331901
6.944710008	4.272012084
7.70635635	4.599792998
8.44564053	5.092945306
9.185072805	6.247254094
9.924394008	6.905695522
10.66386331	8.225293431
11.3808964	9.379684494
12.12032867	10.53399328
12.85961285	11.02714559
13.62144431	12.18137211
14.33844037	13.17047405
15.0777986	13.9942046
15.8172679	15.31380251
16.53430099	16.46819357
17.29613245	17.62242008
18.03556472	18.77672887
18.75256079	19.76583081
19.44737981	21.74674975
20.16444992	23.06642994
20.90384517	24.0554496
21.64331447	25.37504751
22.40510891	26.36398491
23.12236414	28.51011069
23.8395083	30.16036912
24.53417922	31.48013157
25.25132338	33.13039
25.96846754	34.78064842
37.62359661	68.45684629
37.96065806	73.24899666
38.27539438	78.3S7180755
38.50060801	83.82552577
38.70338542	89.11403715
38.90616284	94.40254853
39.04181675	100.021885
39.0430015	105.3111368
39.1113838	110.6001418
39.22463853	116.2195606
39.2931319	122.0044329
39.38391339	127.2933557
39.43000759	133.0783103
39.52078908	138.3672331
39.52201085	143.8217741
39.61290341	149.6065642
39.68124869	154.7302801
39.68254451	160.5153993
39.72867573	166.4656431
39.79705803	171.7546481
39.8431152	177.3743136
39.91149751	182.6633187
39.95755468	188.2829842
40.00357482	193.7373606

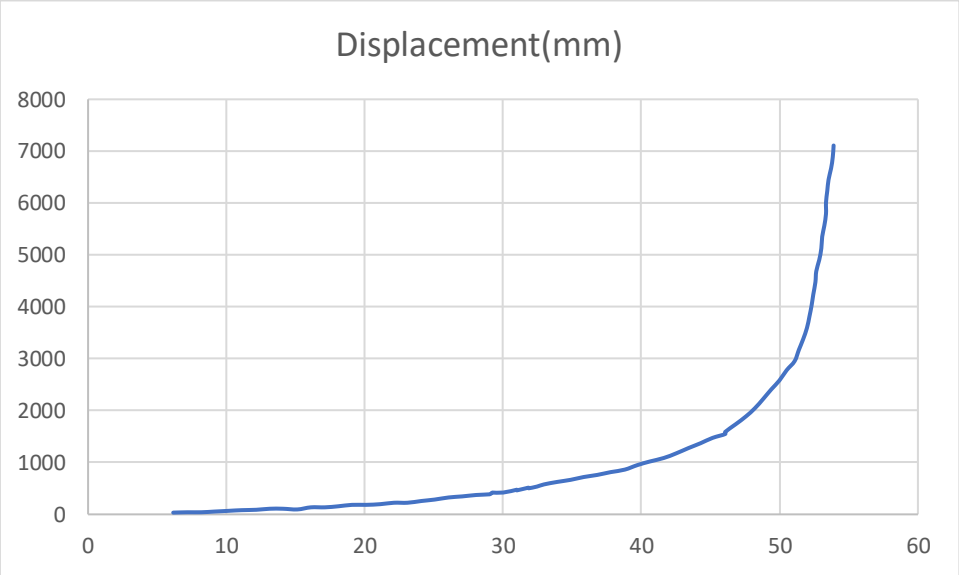
26.70801089	36.43082457
27.42504398	37.58521563
28.11978895	39.23555633
28.83693311	40.88581475
29.59880159	42.20533039
30.3158717	43.52501057
31.01061668	45.17535127
31.75004895	46.32966006
32.46715609	47.81462936
33.18430025	49.46488779
33.87904522	51.11522848
34.61851452	52.43482639
35.31333354	54.41574533
35.96361336	57.55385266
36.56905778	60.52683542
37.10748976	64.3265106



Dataset 2: Agoyama slide (Saito 1979)

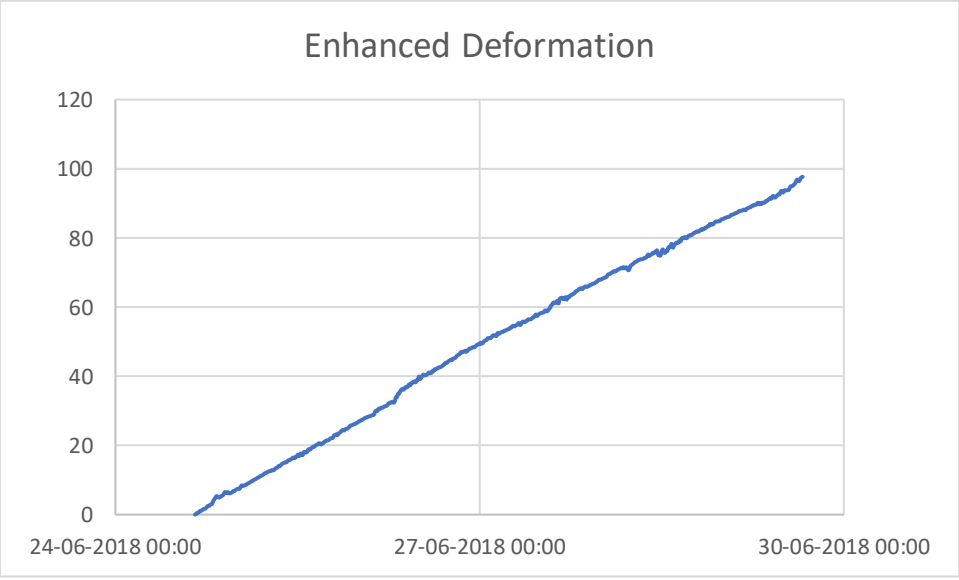
Time(Days)	Displacement(mm)	45.14548239	1473.684211
6.156202144	33.95585739	46.06431853	1548.387097
7.166921899	40.74702886	46.06431853	1589.134126
8.177641654	40.74702886	46.83001531	1738.539898
9.157733538	54.32937182	47.56508423	1887.945671
10.13782542	67.91171477	48.20826953	2044.142615
11.1179173	81.49405772	48.82082695	2227.504244
12.15926493	88.2852292	49.40275651	2410.865874
13.13935681	108.6587436	49.98468606	2580.645161
14.08882083	108.6587436	50.50535988	2777.589134
15.13016845	95.07640068	51.08728943	2954.159593
16.11026034	135.8234295	51.39356815	3171.47708
17.12098009	135.8234295	51.69984686	3375.212224

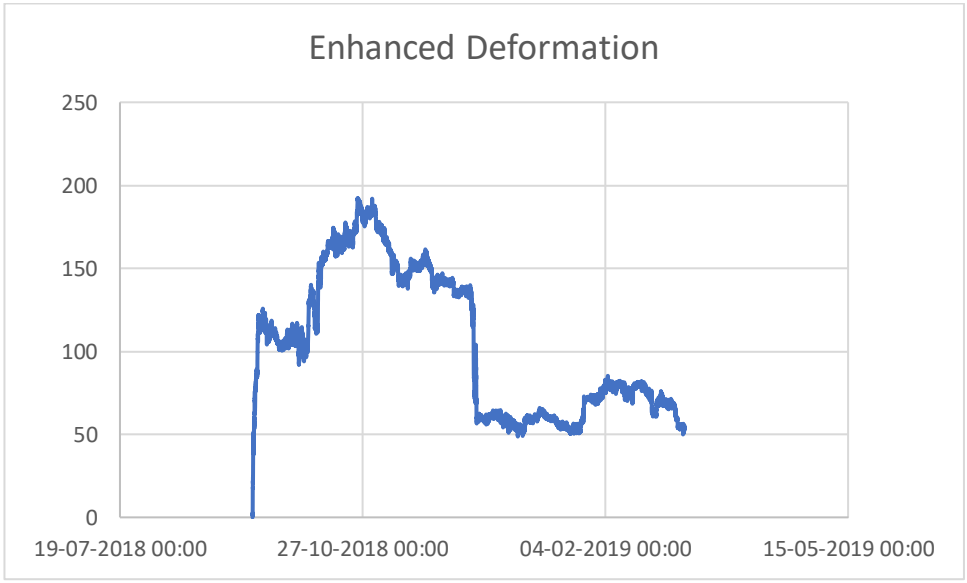
18.10107198	156.196944	51.9754977	3592.529711
19.08116386	183.3616299	52.15926493	3816.63837
20.09188361	183.3616299	52.31240429	4027.164686
21.13323124	196.9439728	52.43491577	4237.691002
22.11332312	224.1086587	52.58805513	4475.382003
23.09341501	224.1086587	52.64931087	4685.908319
24.10413476	258.0645161	52.86370597	4903.225806
25.05359877	285.229202	53.01684533	5127.334465
26.06431853	325.9762309	53.07810107	5351.443124
27.04441041	346.3497453	53.23124043	5561.96944
28.05513017	373.5144312	53.35375191	5792.86927
29.03522205	387.0967742	53.35375191	6010.186757
29.15773354	407.4702886	53.44563553	6234.295416
29.21898928	421.0526316	53.53751914	6451.612903
29.21898928	421.0526316	53.72128637	6668.93039
30.01531394	421.0526316	53.84379786	6886.247878
30.99540582	475.3820034	53.9050536	7110.356537
30.99540582	475.3820034		
30.99540582	468.5908319		
30.99540582	461.7996604		
31.85298622	516.1290323		
31.85298622	516.1290323		
31.85298622	502.5466893		
32.46554364	536.5025467		
32.95558959	577.2495756		
33.9050536	624.7877759		
34.88514548	665.5348048		
35.83460949	719.8641766		
36.81470138	760.6112054		
37.79479326	814.9405772		
38.80551302	862.4787776		
39.72434916	950.7640068		
40.61255743	1018.675722		
41.62327718	1086.587436		
42.51148545	1174.872666		
43.39969372	1276.740238		
44.28790199	1371.816638		

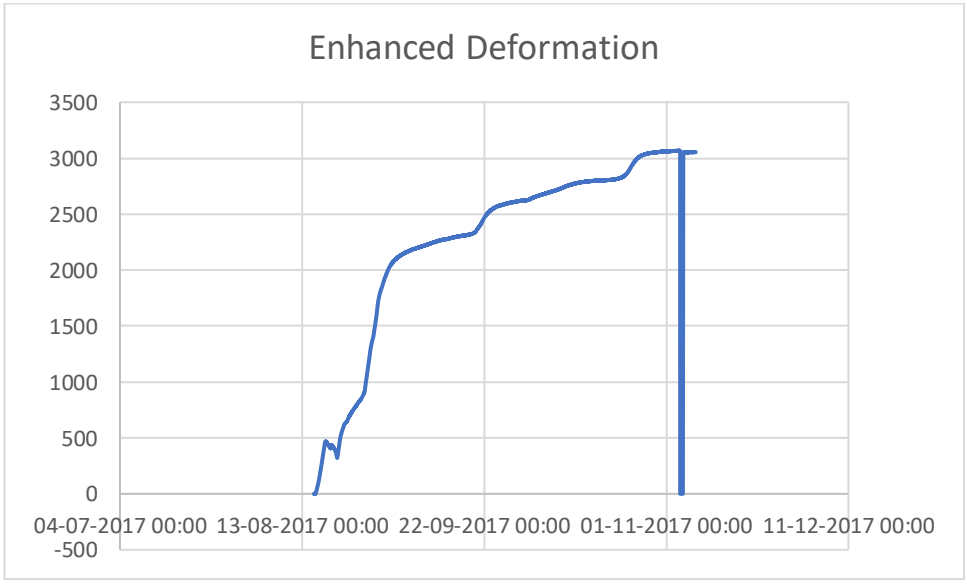


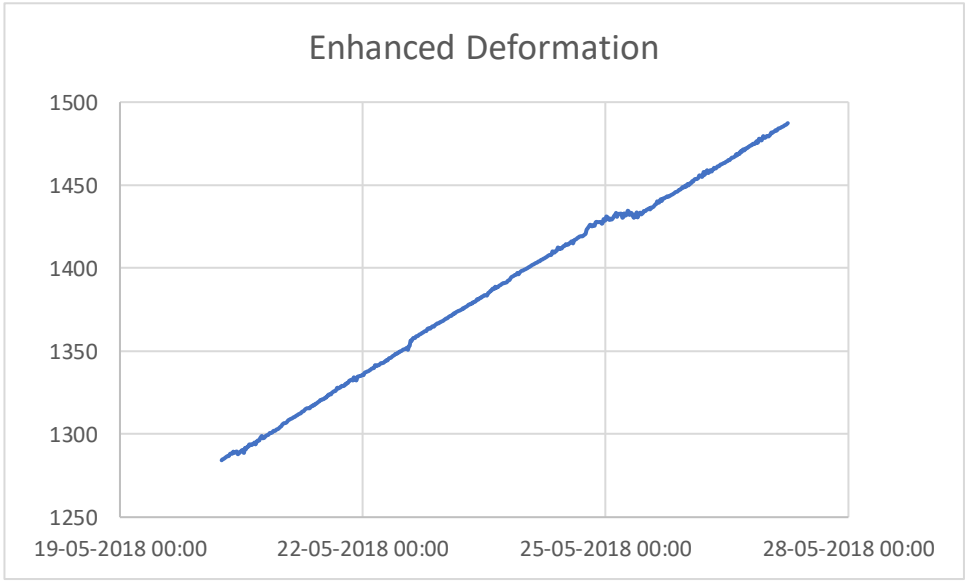
Filed Data: All the field data are collected using the SSR Viewer software and represents the deformation data captured by the SSR on field. The SSR measures deformation approximately every 15 minutes and the deformation is measured in mm. The following graphs are a direct representation of the enhanced deformation(mm) vs time(date-time), measured every 15 minutes.

Dataset 1: Kusmunda Mines, South Eastern Coalfields Limited.

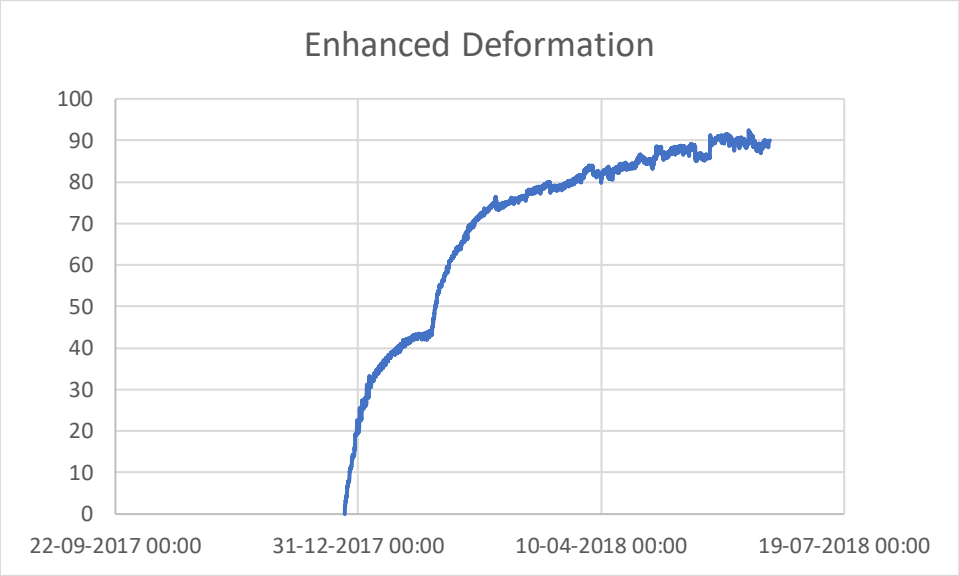


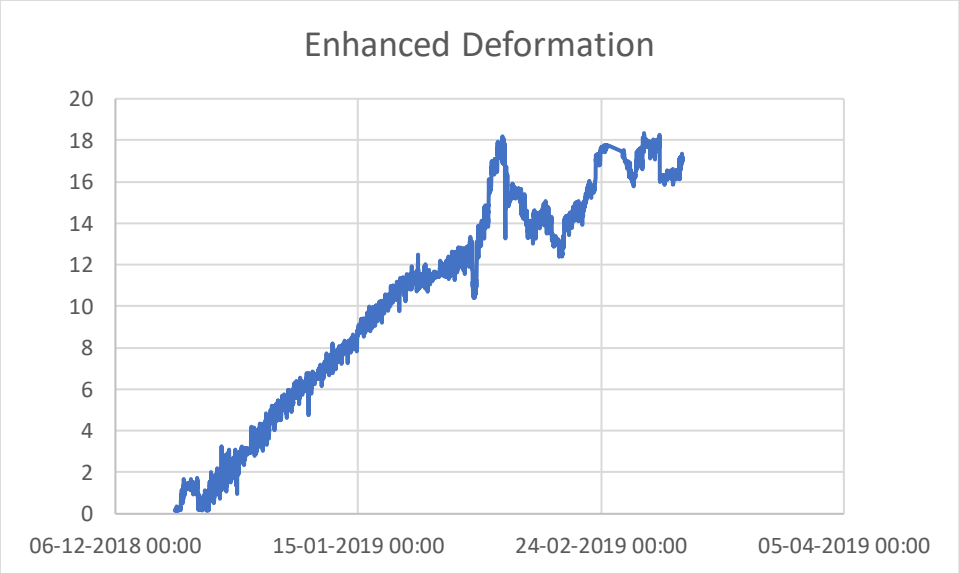

Enhanced Deformation
250
200
150
100
50
0
19-07-2018 00:00
27-10-2018 00:00
04-02-2019 00:00
15-05-2019 00:00

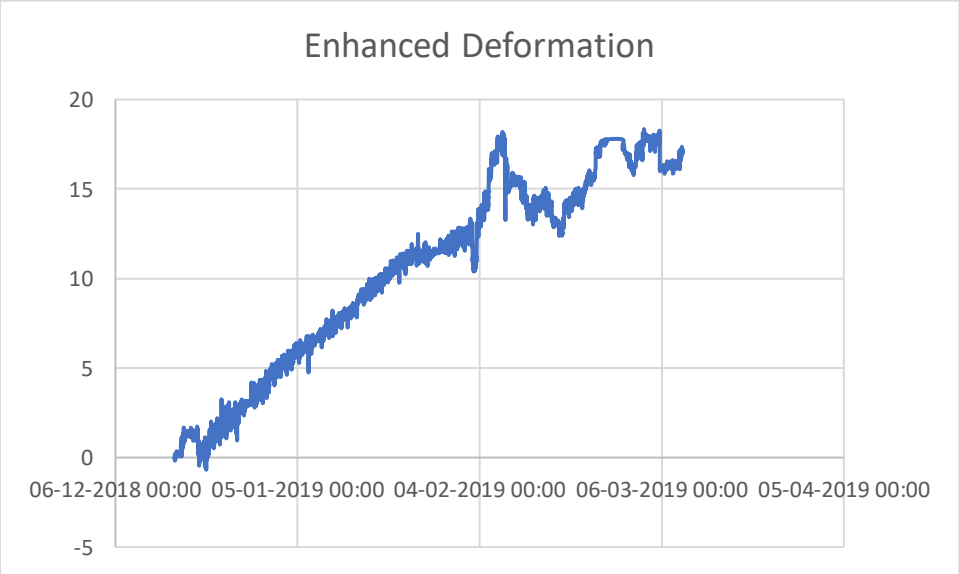

Enhanced Deformation
3500
3000
2500
2000
1500
1000
500
0
-500
04-07-2017 00:00
13-08-2017 00:00
22-09-2017 00:00
01-11-2017 00:00
11-12-2017 00:00

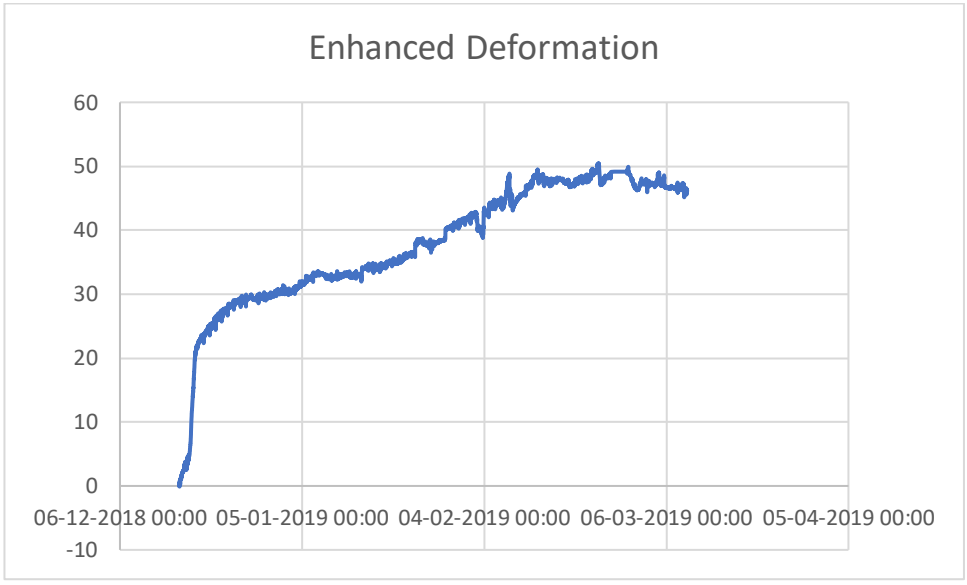

Enhanced Deformation
1500
1450
1400
1350
1300
1250
19-05-2018 00:00
22-05-2018 00:00
25-05-2018 00:00
28-05-2018 00:00

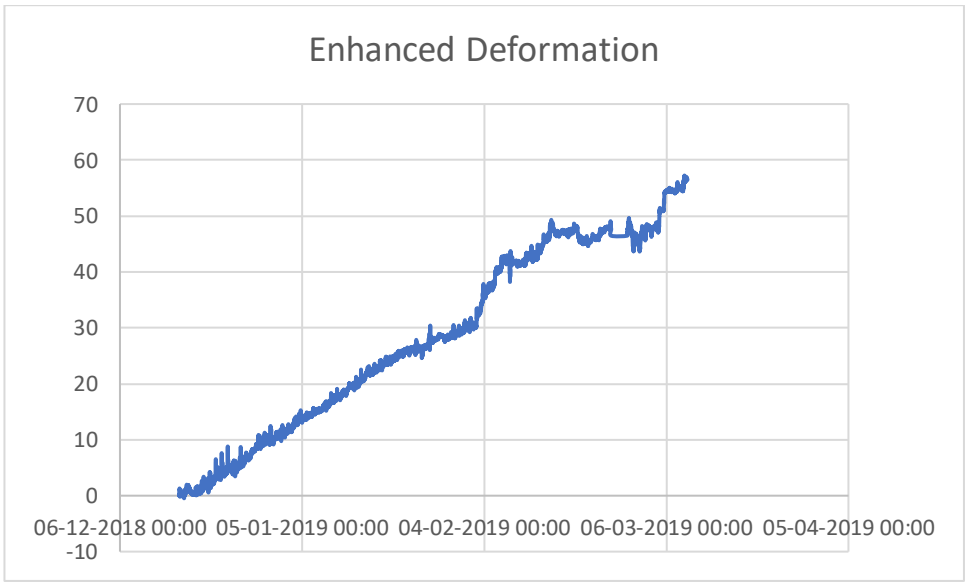
Dataset 2: Gevra Mines, South Eastern Coalfields Limited.

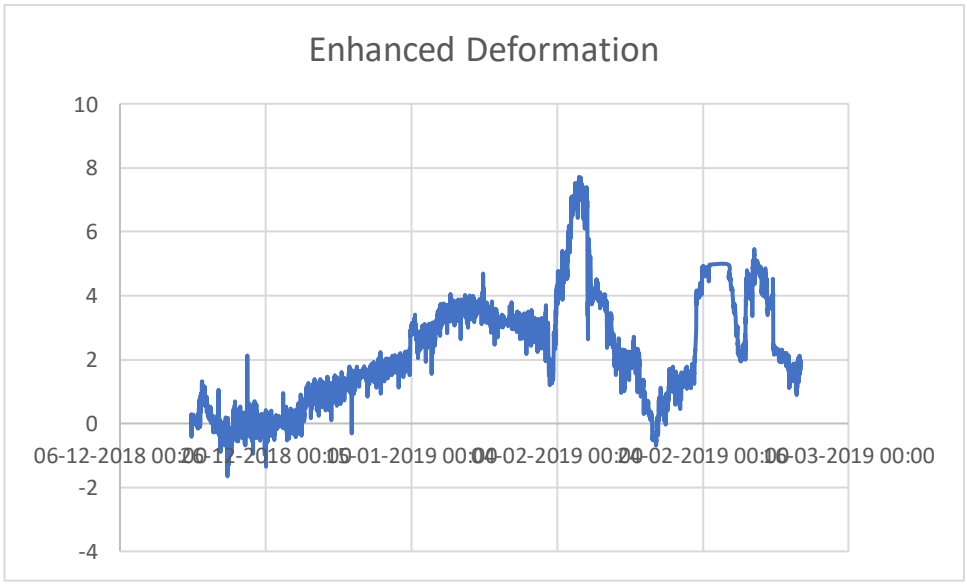


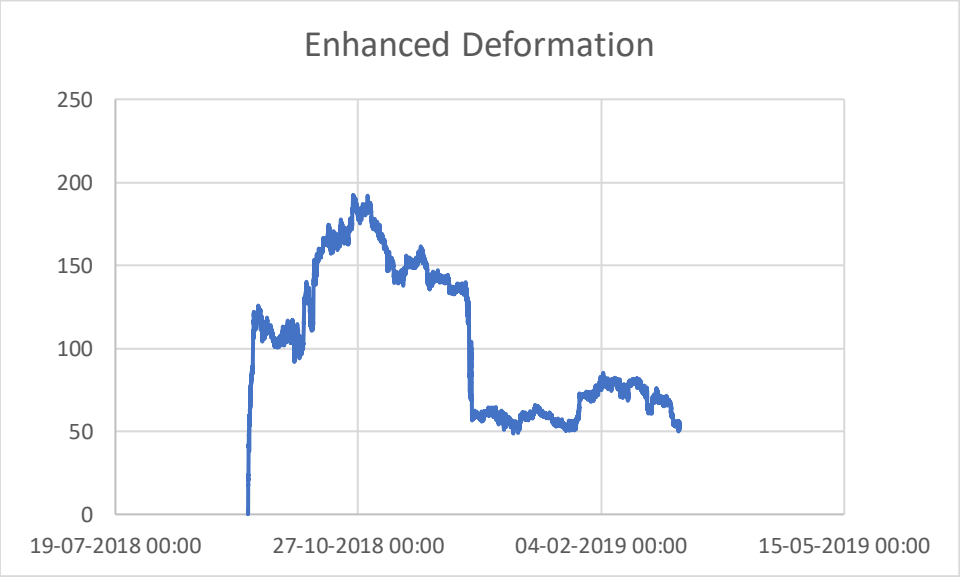


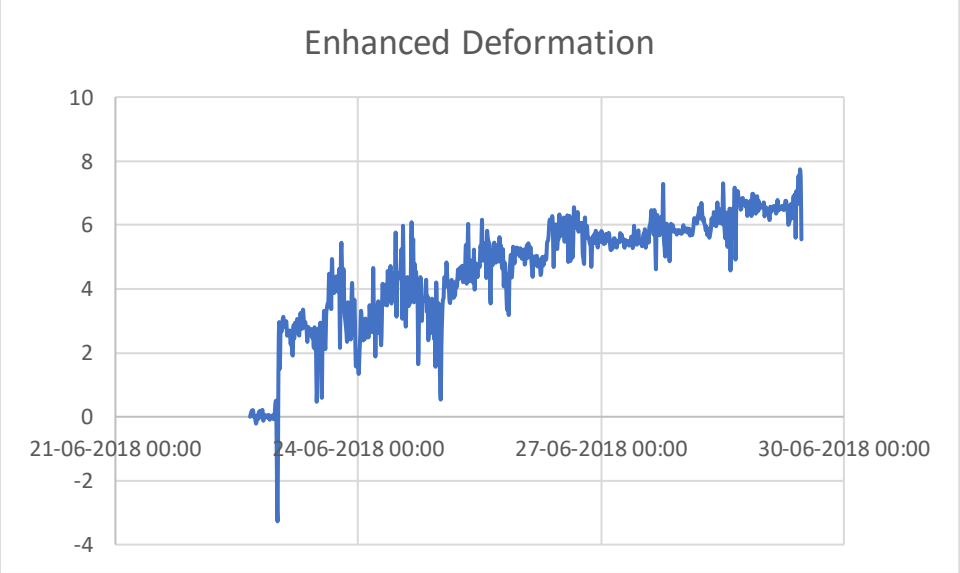


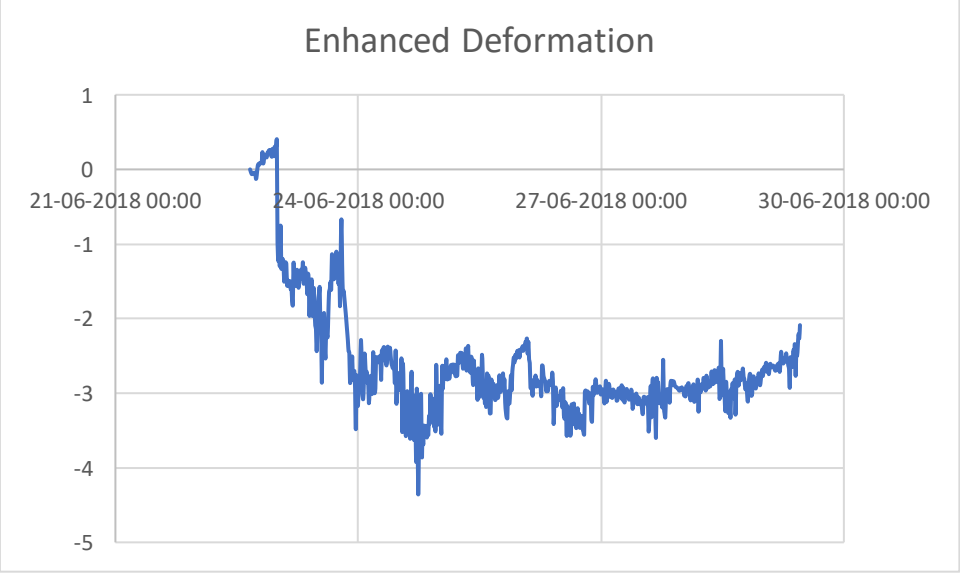

Enhanced Deformation
60
50
40
30
20
10
0
-10
06-12-2018 00:00
05-01-2019 00:00
04-02-2019 00:00
06-03-2019 00:00
05-04-2019 00:00

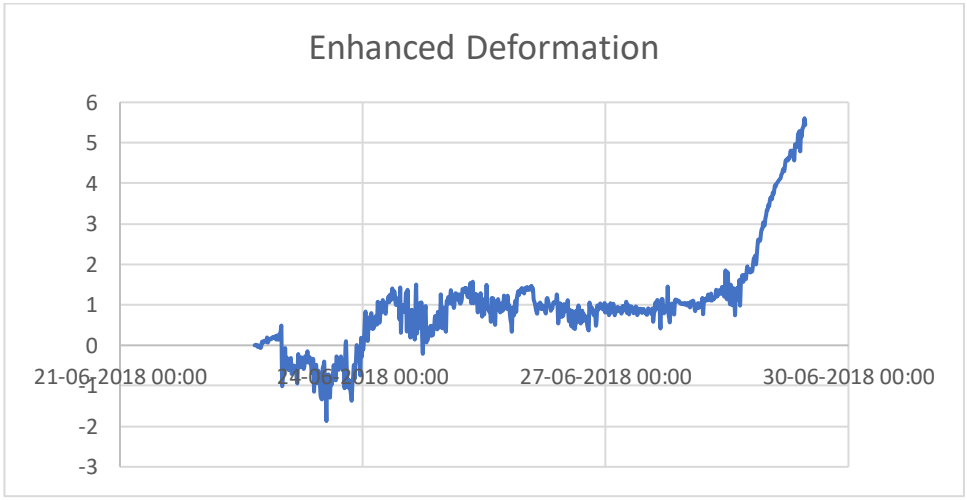

Enhanced Deformation
70
60
50
40
30
20
10
0
-10
06-12-2018 00:00
05-01-2019 00:00
04-02-2019 00:00
06-03-2019 00:00
05-04-2019 00:00


Enhanced Deformation
10
8
6
4
2
0
-2
-4
06-12-2018 00:00
05-01-2019 00:00
04-02-2019 00:00
06-03-2019 00:00

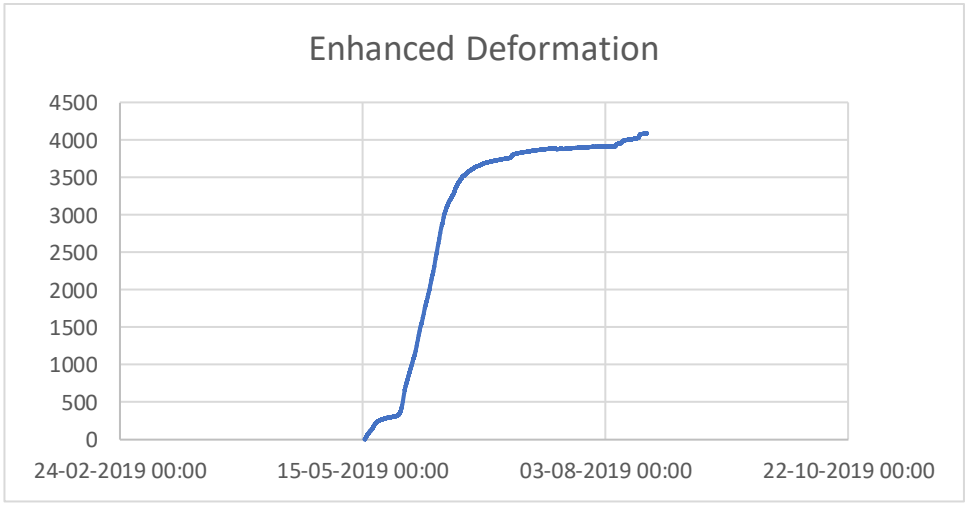

Enhanced Deformation
250
200
150
100
50
0
19-07-2018 00:00
27-10-2018 00:00
04-02-2019 00:00
15-05-2019 00:00

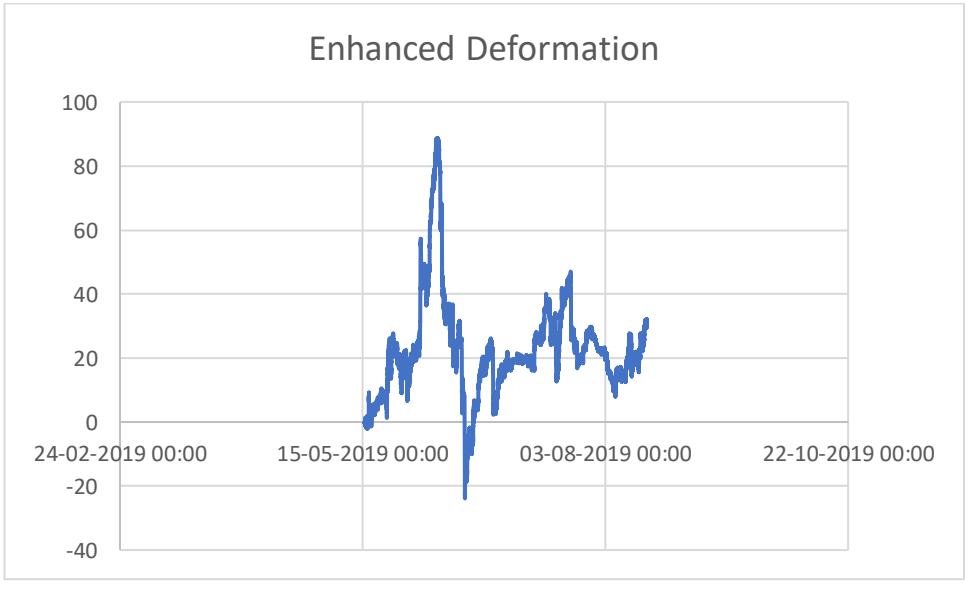

Enhanced Deformation
10
8
6
4
2
0
-2
-4
21-06-2018 00:00
24-06-2018 00:00
27-06-2018 00:00
30-06-2018 00:00


Enhanced Deformation
1
0
-1
-2
-3
-4
-5
21-06-2018 00:00
24-06-2018 00:00
27-06-2018 00:00
30-06-2018 00:00



Dataset 3: Dipka Mines, South Eastern Coalfields Limited.





The code we used for data analysis and finding the onset of failure is shared below;

```
clc;
kus1=readtable('Data 1_Kusmunda Old V1.xlsx');
[r, c]=size(kus1);
kus1
kus1.EnhancedDeformation
dispdiff=diff(kus1.EnhancedDeformation);
N=datenum(kus1.Time);
timediff1=diff(N)*24*60;
vec=[];
for rng=1:r
vec(rng)=rng;
end
timecorr=vec' %transpose of raw matrix
timediff=diff(vec')
newdatatab=table(timediff, dispdiff);
newdatatab
filttabdata=newdatatab(newdatatab.dispdiff>=0,:)
v=zeros(length(timediff),1);
for i=1:length(timediff)
    v(i)=(dispdiff(i)/timediff(i));
end
size(v)
veltbl=table(timediff, v);
veltblpos=veltbl(veltbl.v>=0,:);
[rn, cn]=size(veltblpos);
timecorrval=(timecorr((2:rn+1),:));
figure
plot(timecorrval,veltblpos.v)
velocity=veltblpos.v;
fnltbl=table(timecorrval,velocity)
fnltbl
[row,col]=size(fnltbl);
row,col
%condition 0
co=fnltbl.velocity;
for i=1:row
if co(i)>0
cco=true;
chkrow=i; %for checking if code is running or not;
else
cco=false;
end
end
chkrow
if chkrow==row
fprintf('proceed to condition 1');
% condition 1
    ci=diff(fnltbl.velocity);
lnth_ci=length(ci);
condi=false;
for j=1:lnth_ci-3
cnt=0;
for indi=j:j+3
if ci(indi)>0
```

```
cnt=cnt+1;
               ci(indi)
else
cnt=cnt+0;
end

if cnt>=3
condi=true;
else
condi=false;
end
end
if condi==true
fprintf('condition 1 is satisfied for j=indi-3= %d\n',indi-3);
a_diff_tbl_cnt=0;
t_c=indi-3+9
indpplt=[]
for b=1:5
%figure
%hold on
               x=fnltbl.timecorrval(indi-4+b:t_c-1+b,:);
               y=fnltbl.velocity(indi-4+b:t_c-1+b,:);
pplt=polyfit(x, y, 2);
%x1=fnltbl.timecorrval(t_c:t_c,:);
%y1=fnltbl.velocity(t_c-10:t_c,:);
%pplt1=polyfit(x1, y1, 2);
%hold off
for indplt=1
indpplt(b,:)=pplt(1,1);
end
end
           t_c-1
           indi-4
a_tbl=table(indpplt)
a_chk_cnt=0;
for a_chk=1:4
if a_tbl.indpplt(a_chk)>0
a_chk_cnt=a_chk_cnt+1;
else
a_chk_cnt=a_chk_cnt+0;
end
end
if a_chk_cnt>=3
fprintf('criteria 2 satisfied. Proceeding to criteria 3');
a_diff_tbl_data=diff(a_tbl.indpplt);
a_diff_tbl=table(a_diff_tbl_data)
a_diff_tbl_cnt=0;
for a_diff_cond=1:4
if a_diff_tbl.a_diff_tbl_data(a_diff_cond)>0
a_diff_tbl_cnt=a_diff_tbl_cnt+1;
else
a_diff_tbl_cnt=a_diff_tbl_cnt+0;
end
end
if a_diff_tbl_cnt>=3
fprintf('Acceleration phase reached at velocity index %d and deformation index
%d\n',t_c-9,t_c-9+1);
fprintf('So the time is ');
                 tm=t_c-9+1;
```

```
                kus1.Time(tm)
break;
else
fprintf('condition 3 is not satisfied \n');
end
else
fprintf('criteria 2 is not satisfied properly \n');
end
else
fprintf('condition 1 is not satisfied for j=indi-3= %d\n', indi-3);
end
end
else
fprintf('condition 1 not satisfied\n');
end

for b=1:4
    figure
    hold on
    x=fnltbl.timecorrval(t_c-9+b:t_c+b,:);
    y=fnltbl.velocity(t_c+b-9:t_c+b,:);
pplt=polyfit(x, y, 2);
    f=polyval(pplt,x);
    plot(x,y,'o',x,f,'-')
    hold off
end
figure
for b=1:4
%figure
    hold on
    x=fnltbl.timecorrval(t_c-10+b:t_c+b-1,:);
    y=fnltbl.velocity(t_c+b-10:t_c+b-1,:);
pplt=polyfit(x, y, 2);
    f1=polyval(pplt,x);
    plot(x,y)
    hold off
end
figure
for b=1:1
%figure
    hold on
    x=fnltbl.timecorrval(t_c-10+b:t_c+b-1+4,:);
    y=fnltbl.velocity(t_c+b-10:t_c+b-1+4,:);
pplt=polyfit(x, y, 2);
    f1=polyval(pplt,x);
    plot(x,y,'o',x,f1,'-')
    hold off
end

for b=1:4
    x=fnltbl.timecorrval(t_c-10+b:t_c+b-1,:);
    y=fnltbl.velocity(t_c-10+b:t_c+b-1,:);
pplt_fit=polyfit(x, y, 2)
end
```

Output

```
fnltbl =  5845×2 table
```

time	velocity
2	0.14824
3	0.47646
4	0.22452
5	0.16517
6	0.084181
7	0.48999
8	0.058977
:	:
5832	0.019
5833	0.108
5834	0.149
5835	0.14
5836	0.131
5837	0.115
5838	0.007
5839	0.157
5840	0.186
5841	0.1
5842	0.076
5843	0.162
5844	0.199
5845	0.079
5846	0.145

All the stages were satisfied at the 36^{th} index of time to give the onset of acceleration.

Below is the code in MATLAB with the results of a particular dataset proving the above observations where n= number of monitoring data for parabolic study or curve fitting and d= number of data points needed for the model to run in every stage. Here we also discuss the percentage limit as ratios.

Checking with a d=5, 3/5 configuration:

```
crit5_35 = 2*ones(length(A)-5,1);
for j = 1:length(A)-5
    vj = v(j:j+5);
    aj = A(j:j+5);
    if sum(vj(1:5)>0)<5
        continue
    end
    crit5_35(j) = crit5_35(j)+1;
    if sum(diff(vj)>0)<3
        continue
    end
    crit5_35(j) = crit5_35(j)+1;
    if sum(aj(1:5)>0)<3
        continue
    end
    crit5_35(j) = crit5_35(j)+1;
    if sum(diff(aj)>0)<3
        continue
    end
end
```

Checking with a d=5, 4/5 configuration:

```
crit5_45 = 2*ones(length(A)-5,1);
for j = 1:length(A)-5
    vj = v(j:j+5);
    aj = A(j:j+5);
    if sum(vj(1:5)>0)<5
        continue
    end
    crit5_45(j) = crit5_45(j)+1;
    if sum(diff(vj)>0)<4
        continue
    end
    crit5_45(j) = crit5_45(j)+1;
    if sum(aj(1:5)>0)<4
        continue
    end
    crit5_45(j) = crit5_45(j)+1;
    if sum(diff(aj)>0)<4
        continue
    end
end
```

Plotting different check configuration results:

```
figure(2)
hold on
grid on
plot(t(1:end-13),crit4_34,'-',t(1:end-14),crit5_35,'-',t(1:end-
14),crit5_45,'-')
xlabel('Time (days)');
ylabel('Stage');
legend('d=4, 3/4', 'd=5, 3/5', 'd=5, 4/5');
%ylim([0 4]);
yticks([ 2 3 4 5]);
```

Checking with a d=6, 4/6 configuration:

```
crit6_46 = 2*ones(length(A)-6,1);
for j = 1:length(A)-6
    vj = v(j:j+6);
    aj = A(j:j+6);
    if sum(vj(1:6)>0)<6
        continue
    end
    crit6_46(j) = crit6_46(j)+1;
    if sum(diff(vj)>0)<4
        continue
    end
    crit6_46(j) = crit6_46(j)+1;
    if sum(aj(1:6)>0)<4
        continue
    end
    crit6_46(j) = crit6_46(j)+1;
    if sum(diff(aj)>0)<4
        continue
    end
end
```

Checking with a d=6, 5/6 configuration:

```
crit6_56 = 2*ones(length(A)-6,1);
for j = 1:length(A)-6
    vj = v(j:j+6);
    aj = A(j:j+6);
    if sum(vj(1:6)>0)<6
        continue
    end
    crit6_56(j) = crit6_56(j)+1;
    if sum(diff(vj)>0)<5
        continue
    end
    crit6_56(j) = crit6_56(j)+1;
    if sum(aj(1:6)>0)<5
        continue
    end
    crit6_56(j) = crit6_56(j)+1;
    if sum(diff(aj)>0)<5
        continue
    end
end
```

Plotting different check configuration results:

```
figure(3)
hold on
grid on
plot(t(1:end-13),crit4_34,'-',t(1:end-15),crit6_46,'-',t(1:end-
15),crit6_56,'-')
xlabel('Time (days)');
ylabel('Stage');
legend('d=4, 3/4', 'd=6, 4/6', 'd=6, 5/6');
%ylim([0 4]);
yticks([ 2 3 4 5]);
```

Finding a parabolic fitting for different values of the 'n' parameter. Repeating the procedure for finding parabolic fitting parameters from the beginning:

```
N = [8 9 11 12];
AA = zeros(length(v)-7,1);

for k = 1:length(N)
    for j = 1:length(v)-N(k)+1
        p = polyfit(t(j:j+N(k)-1),v(j:j+N(k)-1),2);
        AA(j,k) = p(1);
    end
end
```

Applying a d=4, 3/4 configuration check for each model with different 'n' value. n=8:

```
crit8 = 2*ones(length(AA)-4,1);
for j = 1:length(crit8)
    vj = v(j:j+4);
    aj = AA(j:j+4,1);
    if sum(vj(1:4)>0)<4
        continue
    end
    crit8(j) = crit8(j)+1;
    if sum(diff(vj)>0)<3
        continue
    end
    crit8(j) = crit8(j)+1;
    if sum(aj(1:4)>0)<3
        continue
    end
    crit8(j) = crit8(j)+1;
    if sum(diff(aj)>0)<3
        continue
    end
end
```

n=9:

```
crit9 = 2*ones(length(AA)-5,1);
for j = 1:length(crit9)
    vj = v(j:j+4);
```

```
    aj = AA(j:j+4,2);
    if sum(vj(1:4)>0)<4
        continue
    end
    crit9(j) = crit9(j)+1;
    if sum(diff(vj)>0)<3
        continue
    end
    crit9(j) = crit9(j)+1;
    if sum(aj(1:4)>0)<3
        continue
    end
    crit9(j) = crit9(j)+1;
    if sum(diff(aj)>0)<3
        continue
    end
end
```

n=11:

```
crit11 = 2*ones(length(AA)-7,1);
for j = 1:length(crit11)
    vj = v(j:j+4);
    aj = AA(j:j+4,3);
    if sum(vj(1:4)>0)<4
        continue
    end
    crit11(j) = crit11(j)+1;
    if sum(diff(vj)>0)<3
        continue
    end
    crit11(j) = crit11(j)+1;
    if sum(aj(1:4)>0)<3
        continue
    end
    crit11(j) = crit11(j)+1;
    if sum(diff(aj)>0)<3
        continue
    end
end
```

n=12:

```
crit12 = 2*ones(length(AA)-8,1);
for j = 1:length(crit12)
    vj = v(j:j+4);
    aj = AA(j:j+4,4);
    if sum(vj(1:4)>0)<4
        continue
    end
    crit12(j) = crit12(j)+1;
    if sum(diff(vj)>0)<3
        continue
    end
    crit12(j) = crit12(j)+1;
    if sum(aj(1:4)>0)<3
        continue
    end
```

```
    crit12(j) = crit12(j)+1;
    if sum(diff(aj)>0)<3
        continue
    end
end
```

Plotting the check results for all models with different 'n' value:

```
figure(4)
hold on
grid on
plot(t(1:end-11),crit8,t(1:end-12),crit9,t(1:end-13),crit4_34,t(1:end-
14),crit11,t(1:end-15),crit12)
xlabel('Time (days)');
ylabel('Stage');
legend('n=8', 'n=9','n=10','n=11','n=12');
%ylim([0 4]);
yticks([ 2 3 4 5]);

figure(5)
hold on
grid on
yyaxis left
plot(T,S)
ylabel('Displacement (mm)');
yyaxis right
plot(T(3:end),ac)
ylabel('Acceleration (mm/d^2)');
xlabel('Time (days)');
```

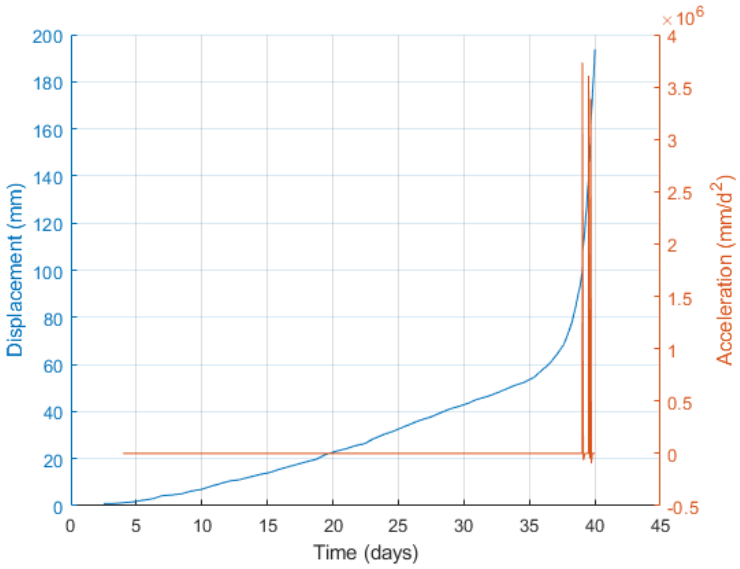


Figure 4. 1 : Plot showing slope behaviour used in the parametric study

```
hold on
grid on
S1 = S(2:end);
```

```
h                                                                                =
plot(t(crit4_34==2),S1(crit4_34==2),'o',t(crit4_34==3),S1(crit4_34==3),'o',t
(crit4_34==4),S1(crit4_34==4),'o',t(crit4_34==5),S1(crit4_34==5),'o');
h(4).Color = [0.4660 0.6740 0.1880];
legend('Stage 2','Stage 3', 'Stage 4','Stage 5');
ylabel('Displacement (mm)');
xlabel('Time (days)');
```

The codes were further modified using the proposed algorithm. These codes are below;

Code 1:

```
close all; clear; clc;

[data] = readmatrix('V1.xlsx');
S = data(:,2);
t = data(:,1);

dS = diff(S);
dt = diff(t);
t = cumsum(dt);
v = dS./dt;

% Find a parabolic fitting for each dataset
A = zeros(length(v)-9,1);
B = zeros(length(v)-9,1);
C = zeros(length(v)-9,1);

for j = 1:length(v)-9
    p = polyfit(t(j:j+9),v(j:j+9),2);
    A(j) = p(1);
    B(j) = p(2);
    C(j) = p(3);
end

% Find a power law fitting for each dataset:
alpha = zeros(length(v)-9,1);
beta = zeros(length(v)-9,1);
for j = 1:length(v)-9
    q = polyfit(log(t(j:j+9)),log(v(j:j+9)),1);
    alpha(j) = exp(q(2));
    beta(j) = q(1);
%     q = fit(t(j:j+9),v(j:j+9),'power1');
%     alpha(j) = q.a;
%     beta(j) = q.b;
end

%Plot the data:
figure(1)
subplot (2,2,1)
f = 200;
fplot(@(t) A(f)*t^2+B(f)*t+C(f),[t(f)-dt(1),t(f+9)+dt(1)]);
hold on
grid on
fplot(@(t) alpha(f)*t^beta(f),[t(f)-dt(1),t(f+9)+dt(1)]);
plot(t(f:f+9),v(f:f+9),'o')
xlabel('Time (days)');
ylabel('Displacement rate (mm/day)');
```

```
legend('Parabolic fitting','Power law fitting', 'Displacement rate data');
title('Dataset #1');

subplot (2,2,2)
f = 520;
fplot(@(t) A(f)*t^2+B(f)*t+C(f),[t(f)-dt(1),t(f+9)+dt(1)]);
hold on
grid on
fplot(@(t) alpha(f)*t^beta(f),[t(f)-dt(1),t(f+9)+dt(1)]);
plot(t(f:f+9),v(f:f+9),'o')
xlabel('Time (days)');
ylabel('Displacement rate (mm/day)');
legend('Parabolic fitting','Power law fitting', 'Displacement rate data');
title('Dataset #2');

subplot (2,2,3)
f = 600;
fplot(@(t) A(f)*t^2+B(f)*t+C(f),[t(f)-dt(1),t(f+9)+dt(1)]);
hold on
grid on
fplot(@(t) alpha(f)*t^beta(f),[t(f)-dt(1),t(f+9)+dt(1)]);
plot(t(f:f+9),v(f:f+9),'o')
xlabel('Time (days)');
ylabel('Displacement rate (mm/day)');
legend('Parabolic fitting','Power law fitting', 'Displacement rate data');
title('Dataset #3');

subplot (2,2,4)
f = 800;
fplot(@(t) A(f)*t^2+B(f)*t+C(f),[t(f)-dt(1),t(f+9)+dt(1)]);
hold on
grid on
fplot(@(t) alpha(f)*t^beta(f),[t(f)-dt(1),t(f+9)+dt(1)]);
plot(t(f:f+9),v(f:f+9),'o')
xlabel('Time (days)');
ylabel('Displacement rate (mm/day)');
legend('Parabolic fitting','Power law fitting', 'Displacement rate data');
title('Dataset #4');
set(gcf,'position',[300,150,1000,800])

crit4_34 = zeros(length(A)-4,1);
for j = 1:length(A)-4
    vj = v(j:j+4);
    aj = A(j:j+4);
    if sum(vj(1:4)>0)<4
        continue
    end
    crit4_34(j) = crit4_34(j)+1;
    if sum(diff(vj)>0)<3
        continue
    end
    crit4_34(j) = crit4_34(j)+1;
    if sum(aj(1:4)>0)<3
        continue
    end
    crit4_34(j) = crit4_34(j)+1;
    if sum(diff(aj)>0)<3
        continue
    end
```

```
end

crit5_35 = zeros(length(A)-5,1);
for j = 1:length(A)-5
    vj = v(j:j+5);
    aj = A(j:j+5);
    if sum(vj(1:5)>0)<5
        continue
    end
    crit5_35(j) = crit5_35(j)+1;
    if sum(diff(vj)>0)<3
        continue
    end
    crit5_35(j) = crit5_35(j)+1;
    if sum(aj(1:5)>0)<3
        continue
    end
    crit5_35(j) = crit5_35(j)+1;
    if sum(diff(aj)>0)<3
        continue
    end
end

crit5_45 = zeros(length(A)-5,1);
for j = 1:length(A)-5
    vj = v(j:j+5);
    aj = A(j:j+5);
    if sum(vj(1:5)>0)<5
        continue
    end
    crit5_45(j) = crit5_45(j)+1;
    if sum(diff(vj)>0)<4
        continue
    end
    crit5_45(j) = crit5_45(j)+1;
    if sum(aj(1:5)>0)<4
        continue
    end
    crit5_45(j) = crit5_45(j)+1;
    if sum(diff(aj)>0)<4
        continue
    end
end

figure(2)
hold on
grid on
plot(t(1:end-13),crit4_34,'-',t(1:end-14),crit5_35,'-',t(1:end-
14),crit5_45,'-')
xlabel('Time (days)');
ylabel('Criterion');
legend('d=4, 3/4', 'd=5, 3/5', 'd=5, 4/5');
%ylim([0 4]);
yticks([ 0 1 2 3 4]);

crit6_46 = zeros(length(A)-6,1);
for j = 1:length(A)-6
    vj = v(j:j+6);
    aj = A(j:j+6);
```

```
    if sum(vj(1:6)>0)<6
        continue
    end
    crit6_46(j) = crit6_46(j)+1;
    if sum(diff(vj)>0)<4
        continue
    end
    crit6_46(j) = crit6_46(j)+1;
    if sum(aj(1:6)>0)<4
        continue
    end
    crit6_46(j) = crit6_46(j)+1;
    if sum(diff(aj)>0)<4
        continue
    end
end

crit6_56 = zeros(length(A)-6,1);
for j = 1:length(A)-6
    vj = v(j:j+6);
    aj = A(j:j+6);
    if sum(vj(1:6)>0)<6
        continue
    end
    crit6_56(j) = crit6_56(j)+1;
    if sum(diff(vj)>0)<5
        continue
    end
    crit6_56(j) = crit6_56(j)+1;
    if sum(aj(1:6)>0)<5
        continue
    end
    crit6_56(j) = crit6_56(j)+1;
    if sum(diff(aj)>0)<5
        continue
    end
end

figure(3)
hold on
grid on
plot(t(1:end-13),crit4_34,'-',t(1:end-15),crit6_46,'-',t(1:end-
15),crit6_56,'-')
xlabel('Time (days)');
ylabel('Criterion');
legend('d=4, 3/4', 'd=6, 4/6', 'd=6, 5/6');
%ylim([0 4]);
yticks([ 0 1 2 3 4]);

N = [8 9 11 12];
AA = zeros(length(v)-7,1);

for k = 1:length(N)
    for j = 1:length(v)-N(k)+1
        p = polyfit(t(j:j+N(k)-1),v(j:j+N(k)-1),2);
        AA(j,k) = p(1);
    end
end
```

```
crit8 = zeros(length(AA)-4,1);
for j = 1:length(crit8)
    vj = v(j:j+4);
    aj = AA(j:j+4,1);
    if sum(vj(1:4)>0)<4
        continue
    end
    crit8(j) = crit8(j)+1;
    if sum(diff(vj)>0)<3
        continue
    end
    crit8(j) = crit8(j)+1;
    if sum(aj(1:4)>0)<3
        continue
    end
    crit8(j) = crit8(j)+1;
    if sum(diff(aj)>0)<3
        continue
    end
end

crit9 = zeros(length(AA)-5,1);
for j = 1:length(crit9)
    vj = v(j:j+4);
    aj = AA(j:j+4,2);
    if sum(vj(1:4)>0)<4
        continue
    end
    crit9(j) = crit9(j)+1;
    if sum(diff(vj)>0)<3
        continue
    end
    crit9(j) = crit9(j)+1;
    if sum(aj(1:4)>0)<3
        continue
    end
    crit9(j) = crit9(j)+1;
    if sum(diff(aj)>0)<3
        continue
    end
end

crit11 = zeros(length(AA)-7,1);
for j = 1:length(crit11)
    vj = v(j:j+4);
    aj = AA(j:j+4,3);
    if sum(vj(1:4)>0)<4
        continue
    end
    crit11(j) = crit11(j)+1;
    if sum(diff(vj)>0)<3
        continue
    end
    crit11(j) = crit11(j)+1;
    if sum(aj(1:4)>0)<3
        continue
    end
    crit11(j) = crit11(j)+1;
    if sum(diff(aj)>0)<3
```

```
        continue
    end
end

crit12 = zeros(length(AA)-8,1);
for j = 1:length(crit12)
    vj = v(j:j+4);
    aj = AA(j:j+4,4);
    if sum(vj(1:4)>0)<4
        continue
    end
    crit12(j) = crit12(j)+1;
    if sum(diff(vj)>0)<3
        continue
    end
    crit12(j) = crit12(j)+1;
    if sum(aj(1:4)>0)<3
        continue
    end
    crit12(j) = crit12(j)+1;
    if sum(diff(aj)>0)<3
        continue
    end
end

figure(4)
hold on
grid on
plot(t(1:end-11),crit8,t(1:end-12),crit9,t(1:end-13),crit4_34,t(1:end-
14),crit11,t(1:end-15),crit12)
xlabel('Time (days)');
ylabel('Criterion');
legend('n=8', 'n=9','n=10','n=11','n=12');
%ylim([0 4]);
yticks([ 0 1 2 3 4]);

figure(5)
hold on
grid on
S1 = S(2:end);

h                                                                    =
plot(t(crit4_34==0),S1(crit4_34==0),'.',t(crit4_34==1),S1(crit4_34==1),'.',t
(crit4_34==2),S1(crit4_34==2),'.',t(crit4_34==3),S1(crit4_34==3),'.');
h(4).Color = [0.4660 0.6740 0.1880];
legend('Criterion 0','Criterion 1', 'Criterion 2','Criterion 3');
ylabel('Displacement (mm)');
xlabel('Time (days)');

def_crit = ones(length(A)-4,1);
for j = 1:length(def_crit)
    vj = v(j:j+4);
    aj = A(j:j+4);
    sj = S(j:j+4);
    if sum(sj>0)<5
        continue
    end
    def_crit(j) = def_crit(j)+1;
    if sum(vj(1:4)>0)<4
```

```
        continue
    end
    def_crit(j) = def_crit(j)+1;
    if sum(diff(vj)>0)<3
        continue
    end
    def_crit(j) = def_crit(j)+1;
    if sum(aj(1:4)>0)<3
        continue
    end
    def_crit(j) = def_crit(j)+1;
    if sum(diff(aj)>0)<3
        continue
    end
end

figure(6)
plot(t(1:end-13),def_crit)
hold on
grid on
xlabel('Time (days)');
ylabel('Criterion');
title('5 criteria check')
yticks([ 1 2 3 4 5]);

figure(7)
h                                                                    =
plot(t(def_crit==1),S1(def_crit==1),'.',t(def_crit==2),S1(def_crit==2),'.',.
..

t(def_crit==3),S1(def_crit==3),'.',t(def_crit==4),S1(def_crit==4),'.',t(def_
crit==5),S1(def_crit==5),'.');
grid on
legend('Criterion 1', 'Criterion 2','Criterion 3','Criterion 4','Criterion
5');
ylabel('Displacement (mm)');
xlabel('Time (days)');
```

Code 2:

```
close all; clear; clc;

%Load the data:
[data] = readmatrix('V1.xlsx','Sheet','Velocityover120');
v = data(10:end,2);
t = data(10:end,1);

dt = diff(t);
t = [0;cumsum(dt)];

% Find a parabolic fitting for each dataset
A = zeros(length(v)-9,1);
B = zeros(length(v)-9,1);
C = zeros(length(v)-9,1);

for j = 1:length(v)-9
    p = polyfit(t(j:j+9),v(j:j+9),2);
    A(j) = p(1);
```

```
    B(j) = p(2);
    C(j) = p(3);
end

% Find a power law fitting for each dataset:
alpha = zeros(length(v)-9,1);
beta = zeros(length(v)-9,1);
for j = 1:length(v)-9
    q = polyfit(log(t(j:j+9)),log(v(j:j+9)),1);
    alpha(j) = exp(q(2));
    beta(j) = q(1);
%     q = fit(t(j:j+9),v(j:j+9),'power1');
%     alpha(j) = q.a;
%     beta(j) = q.b;
end

%Plot the data:
figure(1)
subplot (2,2,1)
f = 50;
fplot(@(t) A(f)*t^2+B(f)*t+C(f),[t(f)-dt(1),t(f+9)+dt(1)]);
hold on
grid on
fplot(@(t) alpha(f)*t^beta(f),[t(f)-dt(1),t(f+9)+dt(1)]);
plot(t(f:f+9),v(f:f+9),'o')
xlabel('Time (days)');
ylabel('Displacement rate (mm/day)');
legend('Parabolic fitting','Power law fitting', 'Displacement rate data');
title('Dataset #1');

subplot (2,2,2)
f = 100;
fplot(@(t) A(f)*t^2+B(f)*t+C(f),[t(f)-dt(1),t(f+9)+dt(1)]);
hold on
grid on
fplot(@(t) alpha(f)*t^beta(f),[t(f)-dt(1),t(f+9)+dt(1)]);
plot(t(f:f+9),v(f:f+9),'o')
xlabel('Time (days)');
ylabel('Displacement rate (mm/day)');
legend('Parabolic fitting','Power law fitting', 'Displacement rate data');
title('Dataset #2');

subplot (2,2,3)
f = 150;
fplot(@(t) A(f)*t^2+B(f)*t+C(f),[t(f)-dt(1),t(f+9)+dt(1)]);
hold on
grid on
fplot(@(t) alpha(f)*t^beta(f),[t(f)-dt(1),t(f+9)+dt(1)]);
plot(t(f:f+9),v(f:f+9),'o')
xlabel('Time (days)');
ylabel('Displacement rate (mm/day)');
legend('Parabolic fitting','Power law fitting', 'Displacement rate data');
title('Dataset #3');

subplot (2,2,4)
f = 200;
fplot(@(t) A(f)*t^2+B(f)*t+C(f),[t(f)-dt(1),t(f+9)+dt(1)]);
hold on
grid on
```

```
fplot(@(t) alpha(f)*t^beta(f),[t(f)-dt(1),t(f+9)+dt(1)]);
plot(t(f:f+9),v(f:f+9),'o')
xlabel('Time (days)');
ylabel('Displacement rate (mm/day)');
legend('Parabolic fitting','Power law fitting', 'Displacement rate data');
title('Dataset #4');
set(gcf,'position',[300,150,1000,800])

crit4_34 = zeros(length(A)-4,1);
for j = 1:length(A)-4
    vj = v(j:j+4);
    aj = A(j:j+4);
    if sum(vj(1:4)>0)<4
        continue
    end
    crit4_34(j) = crit4_34(j)+1;
    if sum(diff(vj)>0)<3
        continue
    end
    crit4_34(j) = crit4_34(j)+1;
    if sum(aj(1:4)>0)<3
        continue
    end
    crit4_34(j) = crit4_34(j)+1;
    if sum(diff(aj)>0)<3
        continue
    end
end

crit5_35 = zeros(length(A)-5,1);
for j = 1:length(A)-5
    vj = v(j:j+5);
    aj = A(j:j+5);
    if sum(vj(1:5)>0)<5
        continue
    end
    crit5_35(j) = crit5_35(j)+1;
    if sum(diff(vj)>0)<3
        continue
    end
    crit5_35(j) = crit5_35(j)+1;
    if sum(aj(1:5)>0)<3
        continue
    end
    crit5_35(j) = crit5_35(j)+1;
    if sum(diff(aj)>0)<3
        continue
    end
end

crit5_45 = zeros(length(A)-5,1);
for j = 1:length(A)-5
    vj = v(j:j+5);
    aj = A(j:j+5);
    if sum(vj(1:5)>0)<5
        continue
    end
    crit5_45(j) = crit5_45(j)+1;
    if sum(diff(vj)>0)<4
```

```
        continue
    end
    crit5_45(j) = crit5_45(j)+1;
    if sum(aj(1:5)>0)<4
        continue
    end
    crit5_45(j) = crit5_45(j)+1;
    if sum(diff(aj)>0)<4
        continue
    end
end

figure(2)
hold on
grid on
plot(t(1:end-13),crit4_34,'-',t(1:end-14),crit5_35,'-',t(1:end-
14),crit5_45,'-')
xlabel('Time (days)');
ylabel('Criterion');
legend('d=4, 3/4', 'd=5, 3/5', 'd=5, 4/5');
%ylim([0 4]);
yticks([ 0 1 2 3 4]);

crit6_46 = zeros(length(A)-6,1);
for j = 1:length(A)-6
    vj = v(j:j+6);
    aj = A(j:j+6);
    if sum(vj(1:6)>0)<6
        continue
    end
    crit6_46(j) = crit6_46(j)+1;
    if sum(diff(vj)>0)<4
        continue
    end
    crit6_46(j) = crit6_46(j)+1;
    if sum(aj(1:6)>0)<4
        continue
    end
    crit6_46(j) = crit6_46(j)+1;
    if sum(diff(aj)>0)<4
        continue
    end
end

crit6_56 = zeros(length(A)-6,1);
for j = 1:length(A)-6
    vj = v(j:j+6);
    aj = A(j:j+6);
    if sum(vj(1:6)>0)<6
        continue
    end
    crit6_56(j) = crit6_56(j)+1;
    if sum(diff(vj)>0)<5
        continue
    end
    crit6_56(j) = crit6_56(j)+1;
    if sum(aj(1:6)>0)<5
        continue
    end
```

```
    crit6_56(j) = crit6_56(j)+1;
    if sum(diff(aj)>0)<5
        continue
    end
end

figure(3)
hold on
grid on
plot(t(1:end-13),crit4_34,'-',t(1:end-15),crit6_46,'-',t(1:end-
15),crit6_56,'-')
xlabel('Time (days)');
ylabel('Criterion');
legend('d=4, 3/4', 'd=6, 4/6', 'd=6, 5/6');
%ylim([0 4]);
yticks([ 0 1 2 3 4]);

N = [8 9 11 12];
AA = zeros(length(v)-7,1);

for k = 1:length(N)
    for j = 1:length(v)-N(k)+1
        p = polyfit(t(j:j+N(k)-1),v(j:j+N(k)-1),2);
        AA(j,k) = p(1);
    end
end

crit8 = zeros(length(AA)-4,1);
for j = 1:length(crit8)
    vj = v(j:j+4);
    aj = AA(j:j+4,1);
    if sum(vj(1:4)>0)<4
        continue
    end
    crit8(j) = crit8(j)+1;
    if sum(diff(vj)>0)<3
        continue
    end
    crit8(j) = crit8(j)+1;
    if sum(aj(1:4)>0)<3
        continue
    end
    crit8(j) = crit8(j)+1;
    if sum(diff(aj)>0)<3
        continue
    end

end

crit9 = zeros(length(AA)-5,1);
for j = 1:length(crit9)
    vj = v(j:j+4);
    aj = AA(j:j+4,2);
    if sum(vj(1:4)>0)<4
        continue
    end
    crit9(j) = crit9(j)+1;
    if sum(diff(vj)>0)<3
        continue
```

```
    end
    crit9(j) = crit9(j)+1;
    if sum(aj(1:4)>0)<3
        continue
    end
    crit9(j) = crit9(j)+1;
    if sum(diff(aj)>0)<3
        continue
    end

end

crit11 = zeros(length(AA)-7,1);
for j = 1:length(crit11)
    vj = v(j:j+4);
    aj = AA(j:j+4,3);
    if sum(vj(1:4)>0)<4
        continue
    end
    crit11(j) = crit11(j)+1;
    if sum(diff(vj)>0)<3
        continue
    end
    crit11(j) = crit11(j)+1;
    if sum(aj(1:4)>0)<3
        continue
    end
    crit11(j) = crit11(j)+1;
    if sum(diff(aj)>0)<3
        continue
    end

end

crit12 = zeros(length(AA)-8,1);
for j = 1:length(crit12)
    vj = v(j:j+4);
    aj = AA(j:j+4,4);
    if sum(vj(1:4)>0)<4
        continue
    end
    crit12(j) = crit12(j)+1;
    if sum(diff(vj)>0)<3
        continue
    end
    crit12(j) = crit12(j)+1;
    if sum(aj(1:4)>0)<3
        continue
    end
    crit12(j) = crit12(j)+1;
    if sum(diff(aj)>0)<3
        continue
    end

end

figure(4)
hold on
grid on
```

```
plot(t(1:end-11),crit8,t(1:end-12),crit9,t(1:end-13),crit4_34,t(1:end-
14),crit11,t(1:end-15),crit12)
xlabel('Time (days)');
ylabel('Criterion');
legend('n=8', 'n=9','n=10','n=11','n=12');
%ylim([0 4]);
yticks([ 0 1 2 3 4]);

% figure(5)
% hold on
% grid on
% plot(t,v)
```

Code 3:

```
close all; clear; clc;

% Load the data:
[data] = readmatrix('Fig7.csv','NumHeaderLines',1);
S = data(:,2);
T = data(:,1);

% Find the velocity and acceleration:
dS = diff(S);
dt = diff(T);
t = cumsum(dt);
v = dS./dt;
ac = diff(v)./dt(2:end);

% Find a parabolic fitting for each dataset
A = zeros(length(v)-9,1);
B = zeros(length(v)-9,1);
C = zeros(length(v)-9,1);
% Iterate through each i...i+9 points to obtain the constants for parabolic
% fitting
for j = 1:length(v)-9
    p = polyfit(t(j:j+9),v(j:j+9),2);
    A(j) = p(1);
    B(j) = p(2);
    C(j) = p(3);
end

% Find a power law fitting for each dataset:
alpha = zeros(length(v)-9,1);
beta = zeros(length(v)-9,1);
for j = 1:length(v)-9
    q = polyfit(log(t(j:j+9)),log(v(j:j+9)),1);
    alpha(j) = exp(q(2));
    beta(j) = q(1);
%     q = fit(t(j:j+9),v(j:j+9),'power1');
%     alpha(j) = q.a;
%     beta(j) = q.b;
end

%Plot the data:
figure(1)
subplot (2,2,1)
```

```
f = 35;
fplot(@(t) A(f)*t^2+B(f)*t+C(f),[t(f)-dt(1),t(f+9)+dt(1)]);
hold on
grid on
fplot(@(t) alpha(f)*t^beta(f),[t(f)-dt(1),t(f+9)+dt(1)]);
plot(t(f:f+9),v(f:f+9),'o')
xlabel('Time (days)');
ylabel('Displacement rate (mm/day)');
legend('Parabolic fitting','Power law fitting', 'Displacement rate data');
title('Dataset #1');

subplot (2,2,2)
f = 40;
fplot(@(t) A(f)*t^2+B(f)*t+C(f),[t(f)-dt(1),t(f+9)+dt(1)]);
hold on
grid on
fplot(@(t) alpha(f)*t^beta(f),[t(f)-dt(1),t(f+9)+dt(1)]);
plot(t(f:f+9),v(f:f+9),'o')
xlabel('Time (days)');
ylabel('Displacement rate (mm/day)');
legend('Parabolic fitting','Power law fitting', 'Displacement rate data');
title('Dataset #2');

subplot (2,2,3)
f = 52;
fplot(@(t) A(f)*t^2+B(f)*t+C(f),[t(f)-dt(1),t(f+9)+dt(1)]);
hold on
grid on
fplot(@(t) alpha(f)*t^beta(f),[t(f)-dt(1),t(f+9)+dt(1)]);
plot(t(f:f+9),v(f:f+9),'o')
xlabel('Time (days)');
ylabel('Displacement rate (mm/day)');
legend('Parabolic fitting','Power law fitting', 'Displacement rate data');
title('Dataset #3');

subplot (2,2,4)
f = 60;
fplot(@(t) A(f)*t^2+B(f)*t+C(f),[t(f)-dt(1),t(f+9)+dt(1)]);
hold on
grid on
fplot(@(t) alpha(f)*t^beta(f),[t(f)-dt(1),t(f+9)+dt(1)]);
plot(t(f:f+9),v(f:f+9),'o')
xlabel('Time (days)');
ylabel('Displacement rate (mm/day)');
legend('Parabolic fitting','Power law fitting', 'Displacement rate data');
title('Dataset #4');
set(gcf,'position',[300,150,1000,800])

crit4_34 = 2*ones(length(A)-4,1);
for j = 1:length(A)-4
    vj = v(j:j+4);
    aj = A(j:j+4);
    if sum(vj(1:4)>0)<4
        continue
    end
    crit4_34(j) = crit4_34(j)+1;
    if sum(diff(vj)>0)<3
        continue
    end
```

```
    crit4_34(j) = crit4_34(j)+1;
    if sum(aj(1:4)>0)<3
        continue
    end
    crit4_34(j) = crit4_34(j)+1;
    if sum(diff(aj)>0)<3
        continue
    end
end

crit5_35 = 2*ones(length(A)-5,1);
for j = 1:length(A)-5
    vj = v(j:j+5);
    aj = A(j:j+5);
    if sum(vj(1:5)>0)<5
        continue
    end
    crit5_35(j) = crit5_35(j)+1;
    if sum(diff(vj)>0)<3
        continue
    end
    crit5_35(j) = crit5_35(j)+1;
    if sum(aj(1:5)>0)<3
        continue
    end
    crit5_35(j) = crit5_35(j)+1;
    if sum(diff(aj)>0)<3
        continue
    end
end

crit5_45 = 2*ones(length(A)-5,1);
for j = 1:length(A)-5
    vj = v(j:j+5);
    aj = A(j:j+5);
    if sum(vj(1:5)>0)<5
        continue
    end
    crit5_45(j) = crit5_45(j)+1;
    if sum(diff(vj)>0)<4
        continue
    end
    crit5_45(j) = crit5_45(j)+1;
    if sum(aj(1:5)>0)<4
        continue
    end
    crit5_45(j) = crit5_45(j)+1;
    if sum(diff(aj)>0)<4
        continue
    end
end

figure(2)
hold on
grid on
plot(t(1:end-13),crit4_34,'-',t(1:end-14),crit5_35,'-',t(1:end-
14),crit5_45,'-')
xlabel('Time (days)');
ylabel('Stage');
```

```
legend('d=4, 3/4', 'd=5, 3/5', 'd=5, 4/5');
%ylim([0 4]);
yticks([ 2 3 4 5]);

crit6_46 = 2*ones(length(A)-6,1);
for j = 1:length(A)-6
    vj = v(j:j+6);
    aj = A(j:j+6);
    if sum(vj(1:6)>0)<6
        continue
    end
    crit6_46(j) = crit6_46(j)+1;
    if sum(diff(vj)>0)<4
        continue
    end
    crit6_46(j) = crit6_46(j)+1;
    if sum(aj(1:6)>0)<4
        continue
    end
    crit6_46(j) = crit6_46(j)+1;
    if sum(diff(aj)>0)<4
        continue
    end
end

crit6_56 = 2*ones(length(A)-6,1);
for j = 1:length(A)-6
    vj = v(j:j+6);
    aj = A(j:j+6);
    if sum(vj(1:6)>0)<6
        continue
    end
    crit6_56(j) = crit6_56(j)+1;
    if sum(diff(vj)>0)<5
        continue
    end
    crit6_56(j) = crit6_56(j)+1;
    if sum(aj(1:6)>0)<5
        continue
    end
    crit6_56(j) = crit6_56(j)+1;
    if sum(diff(aj)>0)<5
        continue
    end
end

figure(3)
hold on
grid on
plot(t(1:end-13),crit4_34,'-',t(1:end-15),crit6_46,'-',t(1:end-
15),crit6_56,'-')
xlabel('Time (days)');
ylabel('Stage');
legend('d=4, 3/4', 'd=6, 4/6', 'd=6, 5/6');
%ylim([0 4]);
yticks([ 2 3 4 5]);

N = [8 9 11 12];
AA = zeros(length(v)-7,1);
```

```
for k = 1:length(N)
    for j = 1:length(v)-N(k)+1
        p = polyfit(t(j:j+N(k)-1),v(j:j+N(k)-1),2);
        AA(j,k) = p(1);
    end
end

crit8 = 2*ones(length(AA)-4,1);
for j = 1:length(crit8)
    vj = v(j:j+4);
    aj = AA(j:j+4,1);
    if sum(vj(1:4)>0)<4
        continue
    end
    crit8(j) = crit8(j)+1;
    if sum(diff(vj)>0)<3
        continue
    end
    crit8(j) = crit8(j)+1;
    if sum(aj(1:4)>0)<3
        continue
    end
    crit8(j) = crit8(j)+1;
    if sum(diff(aj)>0)<3
        continue
    end
end

crit9 = 2*ones(length(AA)-5,1);
for j = 1:length(crit9)
    vj = v(j:j+4);
    aj = AA(j:j+4,2);
    if sum(vj(1:4)>0)<4
        continue
    end
    crit9(j) = crit9(j)+1;
    if sum(diff(vj)>0)<3
        continue
    end
    crit9(j) = crit9(j)+1;
    if sum(aj(1:4)>0)<3
        continue
    end
    crit9(j) = crit9(j)+1;
    if sum(diff(aj)>0)<3
        continue
    end
end

crit11 = 2*ones(length(AA)-7,1);
for j = 1:length(crit11)
    vj = v(j:j+4);
    aj = AA(j:j+4,3);
    if sum(vj(1:4)>0)<4
        continue
    end
    crit11(j) = crit11(j)+1;
    if sum(diff(vj)>0)<3
```

```
        continue
    end
    crit11(j) = crit11(j)+1;
    if sum(aj(1:4)>0)<3
        continue
    end
    crit11(j) = crit11(j)+1;
    if sum(diff(aj)>0)<3
        continue
    end
end

crit12 = 2*ones(length(AA)-8,1);
for j = 1:length(crit12)
    vj = v(j:j+4);
    aj = AA(j:j+4,4);
    if sum(vj(1:4)>0)<4
        continue
    end
    crit12(j) = crit12(j)+1;
    if sum(diff(vj)>0)<3
        continue
    end
    crit12(j) = crit12(j)+1;
    if sum(aj(1:4)>0)<3
        continue
    end
    crit12(j) = crit12(j)+1;
    if sum(diff(aj)>0)<3
        continue
    end
end

figure(4)
hold on
grid on
plot(t(1:end-11),crit8,t(1:end-12),crit9,t(1:end-13),crit4_34,t(1:end-
14),crit11,t(1:end-15),crit12)
xlabel('Time (days)');
ylabel('Stage');
legend('n=8', 'n=9','n=10','n=11','n=12');
%ylim([0 4]);
yticks([ 2 3 4 5]);

figure(5)
hold on
grid on
yyaxis left
plot(T,S)
ylabel('Displacement (mm)');
yyaxis right
plot(T(3:end),ac)
ylabel('Acceleration (mm/d^2)');
xlabel('Time (days)');

figure(6)
hold on
grid on
S1 = S(2:end);
```

```
h                                                                          =
plot(t(crit4_34==2),S1(crit4_34==2),'o',t(crit4_34==3),S1(crit4_34==3),'o',t
(crit4_34==4),S1(crit4_34==4),'o',t(crit4_34==5),S1(crit4_34==5),'o');
h(4).Color = [0.4660 0.6740 0.1880];
legend('Stage 2','Stage 3', 'Stage 4','Stage 5');
ylabel('Displacement (mm)');
xlabel('Time (days)');
```

Code 4:

```
close all; clear; clc;

% Load the data:
[data] = readmatrix('Fig9.csv','NumHeaderLines',1);
S = data(:,2);
T = data(:,1);

% Find the velocity and acceleration:
dS = diff(S);
dt = diff(T);
t = cumsum(dt);
v = dS./dt;
ac = diff(v)./dt(2:end);

% Find a parabolic fitting for each dataset
A = zeros(length(v)-9,1);
B = zeros(length(v)-9,1);
C = zeros(length(v)-9,1);
% Iterate through each i...i+9 points to obtain the constants for parabolic
% fitting
for j = 1:length(v)-9
    p = polyfit(t(j:j+9),v(j:j+9),2);
    A(j) = p(1);
    B(j) = p(2);
    C(j) = p(3);
end

% Find a power law fitting for each dataset:
alpha = zeros(length(v)-9,1);
beta = zeros(length(v)-9,1);
for j = 1:length(v)-9
    q = polyfit(log(t(j:j+9)),log(v(j:j+9)),1);
    alpha(j) = exp(q(2));
    beta(j) = q(1);
%     q = fit(t(j:j+9),v(j:j+9),'power1');
%     alpha(j) = q.a;
%     beta(j) = q.b;
end

%Plot the data:
figure(1)
subplot (2,2,1)
f = 35;
fplot(@(t) A(f)*t^2+B(f)*t+C(f),[t(f)-dt(1),t(f+9)+dt(1)]);
hold on
grid on
fplot(@(t) alpha(f)*t^beta(f),[t(f)-dt(1),t(f+9)+dt(1)]);
plot(t(f:f+9),v(f:f+9),'o')
xlabel('Time (days)');
```

```
ylabel('Displacement rate (mm/day)');
legend('Parabolic fitting','Power law fitting', 'Displacement rate data');
title('Dataset #1');

subplot (2,2,2)
f = 40;
fplot(@(t) A(f)*t^2+B(f)*t+C(f),[t(f)-dt(1),t(f+9)+dt(1)]);
hold on
grid on
fplot(@(t) alpha(f)*t^beta(f),[t(f)-dt(1),t(f+9)+dt(1)]);
plot(t(f:f+9),v(f:f+9),'o')
xlabel('Time (days)');
ylabel('Displacement rate (mm/day)');
legend('Parabolic fitting','Power law fitting', 'Displacement rate data');
title('Dataset #2');

subplot (2,2,3)
f = 52;
fplot(@(t) A(f)*t^2+B(f)*t+C(f),[t(f)-dt(1),t(f+9)+dt(1)]);
hold on
grid on
fplot(@(t) alpha(f)*t^beta(f),[t(f)-dt(1),t(f+9)+dt(1)]);
plot(t(f:f+9),v(f:f+9),'o')
xlabel('Time (days)');
ylabel('Displacement rate (mm/day)');
legend('Parabolic fitting','Power law fitting', 'Displacement rate data');
title('Dataset #3');

subplot (2,2,4)
f = 60;
fplot(@(t) A(f)*t^2+B(f)*t+C(f),[t(f)-dt(1),t(f+9)+dt(1)]);
hold on
grid on
fplot(@(t) alpha(f)*t^beta(f),[t(f)-dt(1),t(f+9)+dt(1)]);
plot(t(f:f+9),v(f:f+9),'o')
xlabel('Time (days)');
ylabel('Displacement rate (mm/day)');
legend('Parabolic fitting','Power law fitting', 'Displacement rate data');
title('Dataset #4');
set(gcf,'position',[300,150,1000,800])

crit4_34 = 2*ones(length(A)-4,1);
for j = 1:length(A)-4
    vj = v(j:j+4);
    aj = A(j:j+4);
    if sum(vj(1:4)>0)<4
        continue
    end
    crit4_34(j) = crit4_34(j)+1;
    if sum(diff(vj)>0)<3
        continue
    end
    crit4_34(j) = crit4_34(j)+1;
    if sum(aj(1:4)>0)<3
        continue
    end
    crit4_34(j) = crit4_34(j)+1;
    if sum(diff(aj)>0)<3
        continue
```

```
    end
end

crit5_35 = 2*ones(length(A)-5,1);
for j = 1:length(A)-5
    vj = v(j:j+5);
    aj = A(j:j+5);
    if sum(vj(1:5)>0)<5
        continue
    end
    crit5_35(j) = crit5_35(j)+1;
    if sum(diff(vj)>0)<3
        continue
    end
    crit5_35(j) = crit5_35(j)+1;
    if sum(aj(1:5)>0)<3
        continue
    end
    crit5_35(j) = crit5_35(j)+1;
    if sum(diff(aj)>0)<3
        continue
    end
end

crit5_45 = 2*ones(length(A)-5,1);
for j = 1:length(A)-5
    vj = v(j:j+5);
    aj = A(j:j+5);
    if sum(vj(1:5)>0)<5
        continue
    end
    crit5_45(j) = crit5_45(j)+1;
    if sum(diff(vj)>0)<4
        continue
    end
    crit5_45(j) = crit5_45(j)+1;
    if sum(aj(1:5)>0)<4
        continue
    end
    crit5_45(j) = crit5_45(j)+1;
    if sum(diff(aj)>0)<4
        continue
    end
end

figure(2)
hold on
grid on
plot(t(1:end-13),crit4_34,'-',t(1:end-14),crit5_35,'-',t(1:end-
14),crit5_45,'-')
xlabel('Time (days)');
ylabel('Stage');
legend('d=4, 3/4', 'd=5, 3/5', 'd=5, 4/5');
%ylim([0 4]);
yticks([ 2 3 4 5]);

crit6_46 = 2*ones(length(A)-6,1);
for j = 1:length(A)-6
    vj = v(j:j+6);
```

```
    aj = A(j:j+6);
    if sum(vj(1:6)>0)<6
        continue
    end
    crit6_46(j) = crit6_46(j)+1;
    if sum(diff(vj)>0)<4
        continue
    end
    crit6_46(j) = crit6_46(j)+1;
    if sum(aj(1:6)>0)<4
        continue
    end
    crit6_46(j) = crit6_46(j)+1;
    if sum(diff(aj)>0)<4
        continue
    end
end

crit6_56 = 2*ones(length(A)-6,1);
for j = 1:length(A)-6
    vj = v(j:j+6);
    aj = A(j:j+6);
    if sum(vj(1:6)>0)<6
        continue
    end
    crit6_56(j) = crit6_56(j)+1;
    if sum(diff(vj)>0)<5
        continue
    end
    crit6_56(j) = crit6_56(j)+1;
    if sum(aj(1:6)>0)<5
        continue
    end
    crit6_56(j) = crit6_56(j)+1;
    if sum(diff(aj)>0)<5
        continue
    end
end

figure(3)
hold on
grid on
plot(t(1:end-13),crit4_34,'-',t(1:end-15),crit6_46,'-',t(1:end-
15),crit6_56,'-')
xlabel('Time (days)');
ylabel('Stage');
legend('d=4, 3/4', 'd=6, 4/6', 'd=6, 5/6');
%ylim([0 4]);
yticks([ 2 3 4 5]);

N = [8 9 11 12];
AA = zeros(length(v)-7,1);

for k = 1:length(N)
    for j = 1:length(v)-N(k)+1
        p = polyfit(t(j:j+N(k)-1),v(j:j+N(k)-1),2);
        AA(j,k) = p(1);
    end
end
```

```
crit8 = 2*ones(length(AA)-4,1);
for j = 1:length(crit8)
    vj = v(j:j+4);
    aj = AA(j:j+4,1);
    if sum(vj(1:4)>0)<4
        continue
    end
    crit8(j) = crit8(j)+1;
    if sum(diff(vj)>0)<3
        continue
    end
    crit8(j) = crit8(j)+1;
    if sum(aj(1:4)>0)<3
        continue
    end
    crit8(j) = crit8(j)+1;
    if sum(diff(aj)>0)<3
        continue
    end
end

crit9 = 2*ones(length(AA)-5,1);
for j = 1:length(crit9)
    vj = v(j:j+4);
    aj = AA(j:j+4,2);
    if sum(vj(1:4)>0)<4
        continue
    end
    crit9(j) = crit9(j)+1;
    if sum(diff(vj)>0)<3
        continue
    end
    crit9(j) = crit9(j)+1;
    if sum(aj(1:4)>0)<3
        continue
    end
    crit9(j) = crit9(j)+1;
    if sum(diff(aj)>0)<3
        continue
    end
end

crit11 = 2*ones(length(AA)-7,1);
for j = 1:length(crit11)
    vj = v(j:j+4);
    aj = AA(j:j+4,3);
    if sum(vj(1:4)>0)<4
        continue
    end
    crit11(j) = crit11(j)+1;
    if sum(diff(vj)>0)<3
        continue
    end
    crit11(j) = crit11(j)+1;
    if sum(aj(1:4)>0)<3
        continue
    end
    crit11(j) = crit11(j)+1;
```

```
    if sum(diff(aj)>0)<3
        continue
    end
end
crit12 = 2*ones(length(AA)-8,1);
for j = 1:length(crit12)
    vj = v(j:j+4);
    aj = AA(j:j+4,4);
    if sum(vj(1:4)>0)<4
        continue
    end
    crit12(j) = crit12(j)+1;
    if sum(diff(vj)>0)<3
        continue
    end
    crit12(j) = crit12(j)+1;
    if sum(aj(1:4)>0)<3
        continue
    end
    crit12(j) = crit12(j)+1;
    if sum(diff(aj)>0)<3
        continue
    end
end
Figure
hold on
grid on
plot(t(1:end-11),crit8,t(1:end-12),crit9,t(1:end-13),crit4_34,t(1:end-
14),crit11,t(1:end-15),crit12)
xlabel('Time (days)');
ylabel('Stage');
legend('n=8', 'n=9','n=10','n=11','n=12');
%ylim([0 4]);
yticks([ 2 3 4 5]);
Figure
hold on
grid on
yyaxis left
plot(T,S)
ylabel('Displacement (mm)');
yyaxis right
plot(T(3:end),ac)
ylabel('Acceleration (mm/d^2)');
xlabel('Time (days)');
Figure
hold on
grid on
S1 = S(2:end);
h                                                                                =
plot(t(crit4_34==2),S1(crit4_34==2),'o',t(crit4_34==3),S1(crit4_34==3),'o',t
(crit4_34==4),S1(crit4_34==4),'o',t(crit4_34==5),S1(crit4_34==5),'o');
h(4).Color = [0.4660 0.6740 0.1880];
legend('Stage 2','Stage 3', 'Stage 4','Stage 5');
ylabel('Displacement (mm)');
xlabel('Time (days)');
```

APPENDIX-C

(A) PhD research paper in journal

1. Masood M.M., Verma T., Seervi V., "Development of an Algorithm for the Prediction of Slope Failure in Surface Mines", Journal of The Institution of Engineers (India) Series D, 2023.

2. Masood M.M., Verma T., Yugaraju G., "Developing an Algorithm for the Identification of Onset of Failure of Mine Slopes in Surface Mines", Journal of Mines, Metals & Fuels, 2023.

3. Masood M.M., Yugaraju G., Verma T., "Slope Monitoring and Failure Prediction Techniques in Mines: A Review", Journal of Mines, Metals & Fuels, 2022.