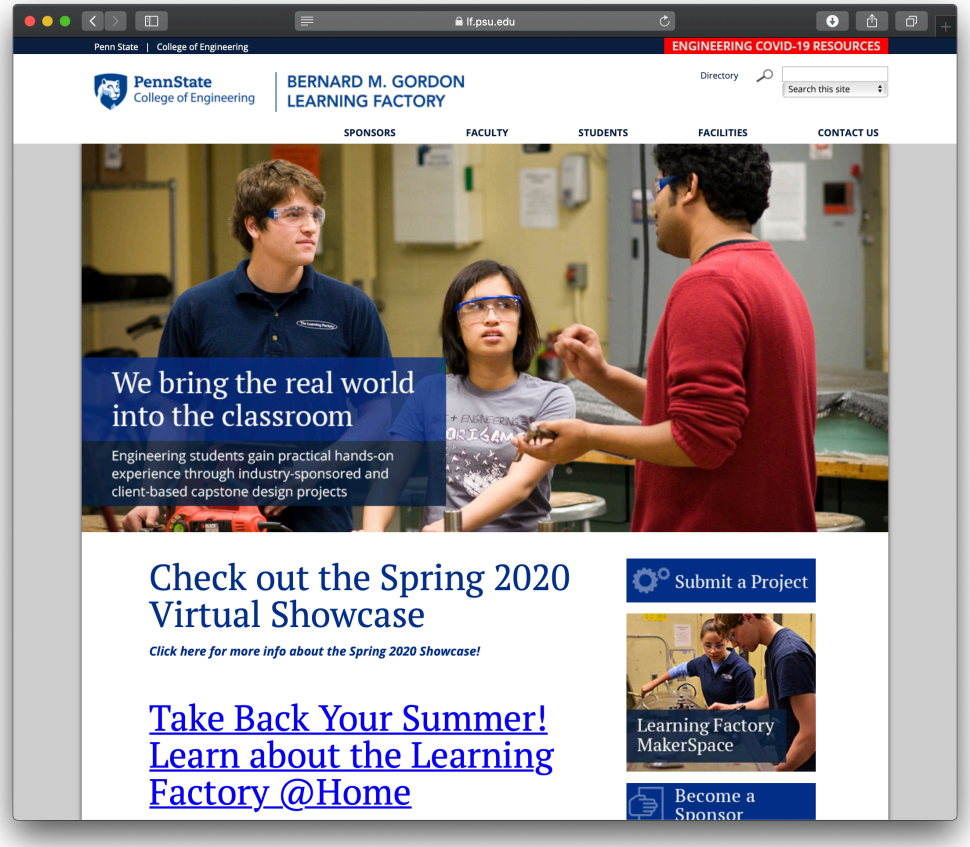
Generating an Interpretable Surrogate Model for Predicting Damage Accumulation

Chris McComb, Zhenghui Sha, Binyang Song,, Faez Ahmed

The Bernard M. Gordon Learning Factory is one of Penn State's makerspaces.

You can learn more about the Learning Factory on the website or through recent publications.



ELSEVIER
CrossMark
Available online at www.sciencedirect.com
ScienceDirect
Procedia Manufacturing 45 (2020) 528–533
Procedia MANUFACTURING
www.elsevier.com/locate/procedia

10th Conference on Learning Factories, CLF2020

The role of digital prototyping tools in learning factories

Jacob Nelson[a], Andrew Berlin[a], Jessica Menold[a,*], Matthew Parkinson[a]

[a]*The Pennsylvania State University, University Park 16802, USA*

ELSEVIER
CrossMark
Available online at www.sciencedirect.com
ScienceDirect
Procedia Manufacturing 45 (2020) 534–539
Procedia MANUFACTURING
www.elsevier.com/locate/procedia

10th Conference on Learning Factories, CLF2020

Broadening participation in learning factories through Industry 4.0

Daniel R. Spillane[a], Jessica Menold[a], Matthew B. Parkinson[a,*]

[a]*The Pennsylvania State University, University Park 16802, USA*

Several of the machines at the learning factory are outfitted with sensors for predictive maintenance.

KCF technologies

These sensors measure lots of information and also compute a damage accumulation measure.

Measured Values

v_p, peak velocity

v_{RMS}, RMS velocity

a_p, peak acceleration

T, ambient temperature

Derived Values

Damage Accumulation (related to fatigue, and useful for identifying predictive maintenance opportunities)

We want to simulate future usage scenarios to assess damage, but the sensors only work in realtime.

How will our maintenance schedule be impacted by intermittent use during COVID?

If we train machine operators in specific techniques, can it reduce damage over time?

What will happen if we make the Learning Factory open 24 hours?

What is the ROI on buying additional machines?

That's where you come in!

Your mission, should you choose to accept it, is to create a ***interpretable*** surrogate model for damage accumulation that can be used for these “what if” scenarios.

Your model must make accurate predictions, but it must also be interpretable.

You must explain which variables are most influential, to what extent they affect damage, and why they are important. This should be based on sound theories and underlying physics.

You will have data for 5 different machines.

× 1

× 3

× 1

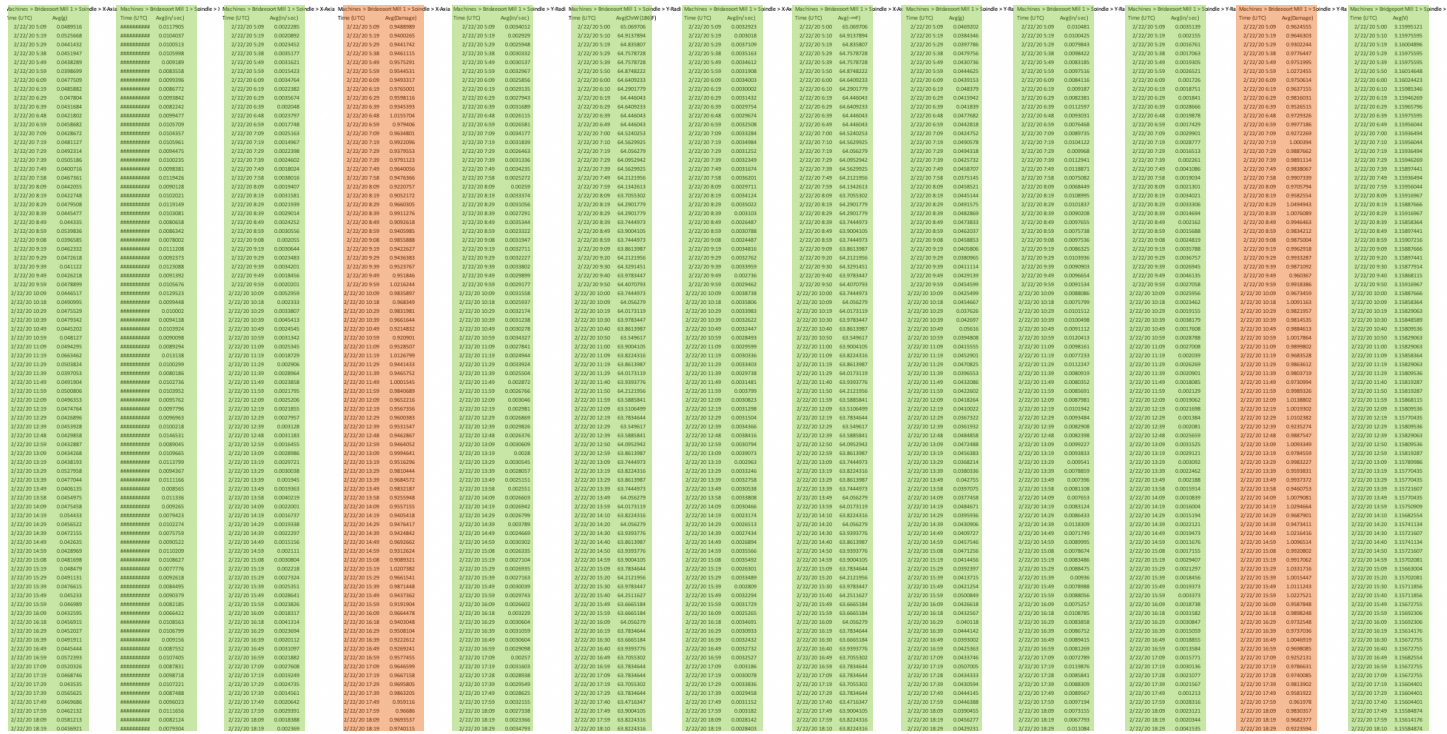
For each machine, you will have four files:

- [machine name]week1-train.csv
- [machine name]week2-train.csv
- [machine name]week3-test.csv
- [machine name]week3-submit.csv

You will have two weeks of labeled data for training (5 machines x 2 weeks each = 10 files).

Every file has 11 independent variables (physical measurements)

Each file has approximately 1000 rows, indicating readings taken every ~10 minutes over the week



Every file has 2 dependent variables (derived damage accumulation)

Let’s look at one of these variables

D	E
Machines > Bridgeport Mill 1 > Spindle > X-Axial > Peak Velocity	
Time (UTC)	Avg(in/sec)
2/22/20 5:09	0.0117905
2/22/20 5:19	0.0104037
2/22/20 5:29	0.0100513
2/22/20 5:38	0.0105998
2/22/20 5:49	0.009189
2/22/20 5:59	0.0083558
2/22/20 6:09	0.0099396
2/22/20 6:19	0.0086772
2/22/20 6:29	0.0093842
2/22/20 6:39	0.0082242
2/22/20 6:48	0.0099477

The header identifies the name of the variable

The left column provides a time stamp for each reading

The right column provides the values of the readings

Be careful – sometimes readings are not recorded. You'll need to account for this!

Every reading is separately timestamped to help with verification

I	J	K	L	M	N	O	P	Q	R
	Machines > Bridgeport Mill 1 > Spindle > X-Axial >			Machines > Bridgeport Mill 1 > Spindle > Y-Radial			Machines > Bridgeport Mill 1 > Spindle > Y-Radial >		
	Time (UTC)	Avg(Damage)		Time (UTC)	Avg(in/sec)		Time (UTC)	Avg(ChrW(186)F)	
	2/29/20 3:09	0.9457199		2/29/20 3:19	0.0029823		2/29/20 3:00	65.8102188	
	2/29/20 3:19	0.9097668		2/29/20 3:29	0.0035575		2/29/20 3:09	65.6153351	
	2/29/20 3:29	0.9444627		2/29/20 3:39	0.0028986		2/29/20 3:19	65.5374008	
	2/29/20 3:39	0.9422111		2/29/20 3:50	0.002899		2/29/20 3:29	65.4984337	
	2/29/20 3:50	0.9866141		2/29/20 3:59	0.0028412		2/29/20 3:39	65.4594666	
	2/29/20 3:59	0.9836155		2/29/20 4:09	0.0031633		2/29/20 3:50	65.1866039	
	2/29/20 4:09	0.9304277		2/29/20 4:19	0.0030401		2/29/20 3:59	65.1086731	
	2/29/20 4:19	0.911431		2/29/20 4:29	0.0030853		2/29/20 4:10	65.1086731	
	2/29/20 4:29	0.9392434		2/29/20 4:39	0.0031206		2/29/20 4:20	64.9137894	
	2/29/20 4:39	0.9291499		2/29/20 4:49	0.0024175		2/29/20 4:30	64.7189056	
	2/29/20 4:49	0.9240439					2/29/20 4:40	64.7578728	
							2/29/20 4:50	64.6409233	

Uneven column lengths indicate missing readings

You will have data for 5 different machines.

× 1

× 3

× 1

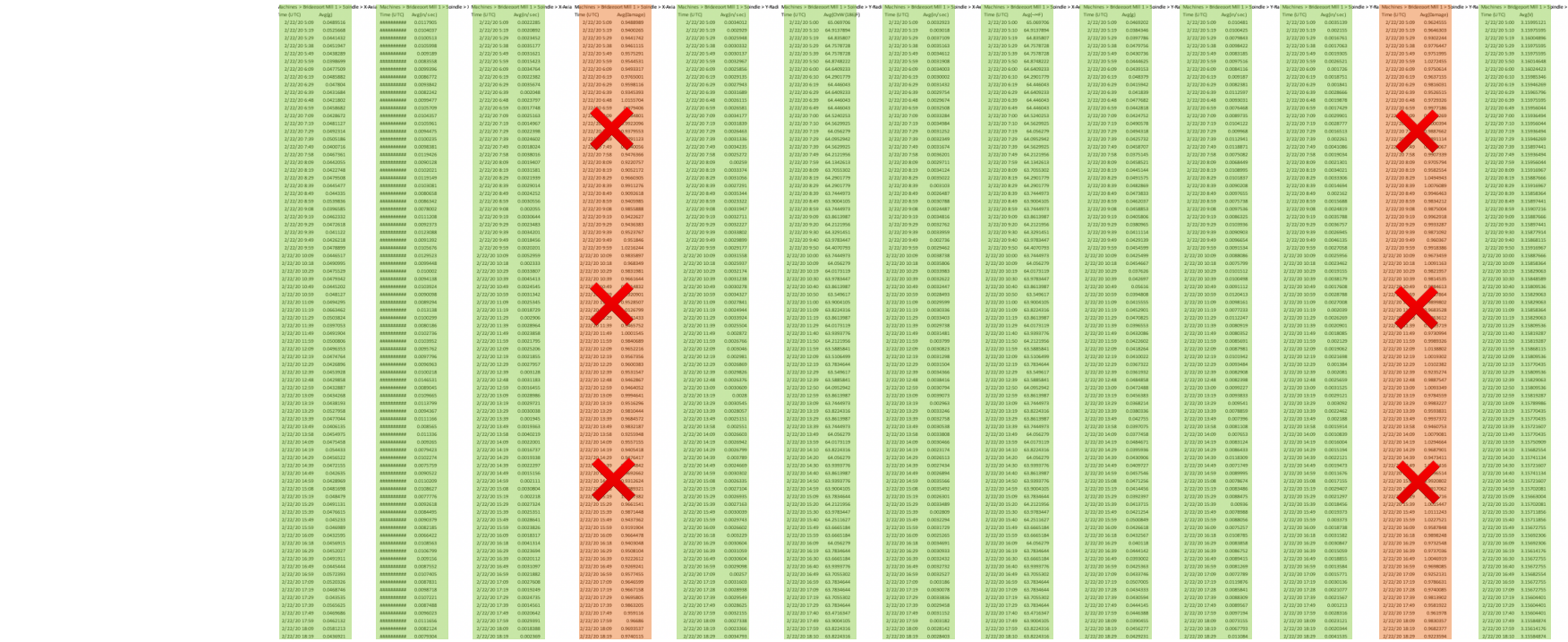
For each machine, you will have four files:

- [machine name]week1-train.csv
- [machine name]week2-train.csv
- [machine name]week3-test.csv
- [machine name]week3-submit.csv

Use each **[machine name]week3-test.csv** to run your model, and use the outputs to populate **[machine name]week3-submit.csv**

[machine name]week3-test.csv

Formatted like training data, but no labels (no damage accumulation values)



Use each **[machine name]week3-test.csv** to run your model, and then format the outputs to populate **[machine name]week3-submit.csv**

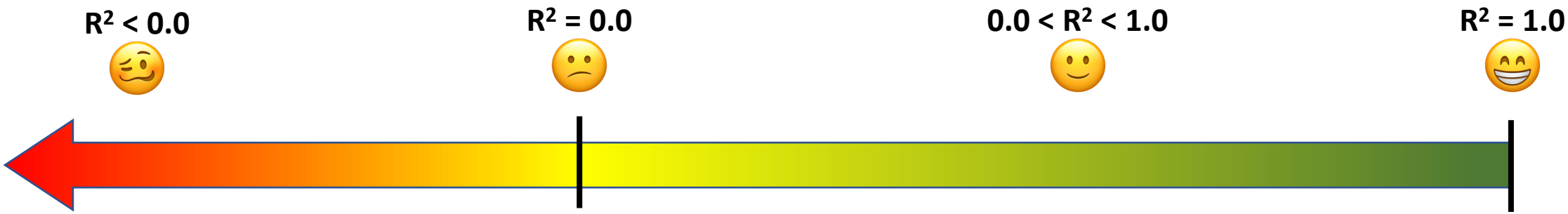
[machine name]week3-submit.csv

Do not edit the first row or the first column

Time (UTC)	Machines > Bridgeport Mill 3 > Spindle > X-Axial > Damage Accumulation	Machines > Bridgeport Mill 3 > Spindle > Y-Radial > Damage Accumulation
2/29/20 5:00		
2/29/20 5:09		
2/29/20 5:19		
2/29/20 5:29		
2/29/20 5:39		
2/29/20 5:49		
2/29/20 5:59		
2/29/20 6:10		
2/29/20 6:19		
2/29/20 6:29		
2/29/20 6:39		
2/29/20 6:49		
2/29/20 6:59		
2/29/20 7:09		
2/29/20 7:20		
2/29/20 7:29		
2/29/20 7:39		
2/29/20 7:49		
2/29/20 7:59		
2/29/20 8:09		
2/29/20 8:19		

Populate the empty cells with your model's predictions

We will compare your submission to ground truth values and calculate an average R^2 score.



A value less than 0 is possible, but not good. I have hope that you won't end up here!

A value of 0 means that you model does as well as an average predictor

Values between 0 and 1 are typical

A value of 1.0 means that your model is perfect!

You will also be scored on Technical Approach, Creativity/Innovation, and Overall Presentation.

Category	Criteria	Scoring
Technical Approach (35%) *Methods and algorithms*	• Requirement analysis and problem formulation • Literature review and exploration of ideas • The development and design of the idea • Readiness of the idea and the approach	Excellent (31-35 pts) Very good (24-30 pts) Good (17-24 pts) Limited (9-16 pts) Poor (1-8 pts)
Creativity and innovation (20%) *New or unique approaches*	• The technology breaks new ground • The project makes a profound break from established design • The project adds a major departure from established design • The code adds a new twist on established design • The chosen technology and design is already deeply established	Excellent (17-20 pts) Very good (13-16 pts) Good (19-12 pts) Limited (5-8 pts) Poor (1-4 pts)
Results (35%) *Output performance*	• This will be evaluated automatically based on results that the team submits. • Specifically, we will use an R-squared metric over testing set	Team with the best performance (35 points) Team with the second-best performance (24 points) Team with the third-best performance (18 points) Teams at fourth and fifth ranks (10 points) All others (3 points)
Overall Presentation (10%) *Organization, structure, and message conveying*	• Title, headings, labels: Appropriate size, location, spelling, and content • The demonstration of teamwork • Structure and Clarity • Boarder impact of the idea to ME subfields	Excellent (9-10 pts) Very good (7-8 pts) Good (5-6 pts) Limited (3-4 pts) Poor (1-2 pts)

Thank you!

Questions?