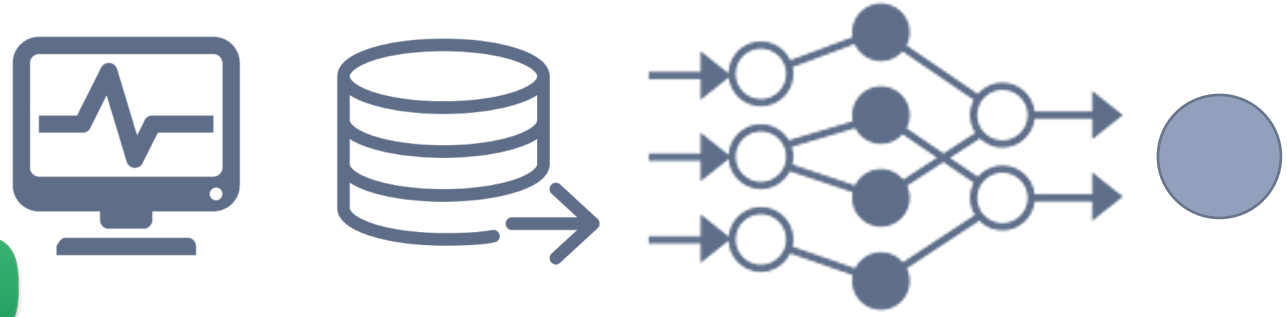
adapt

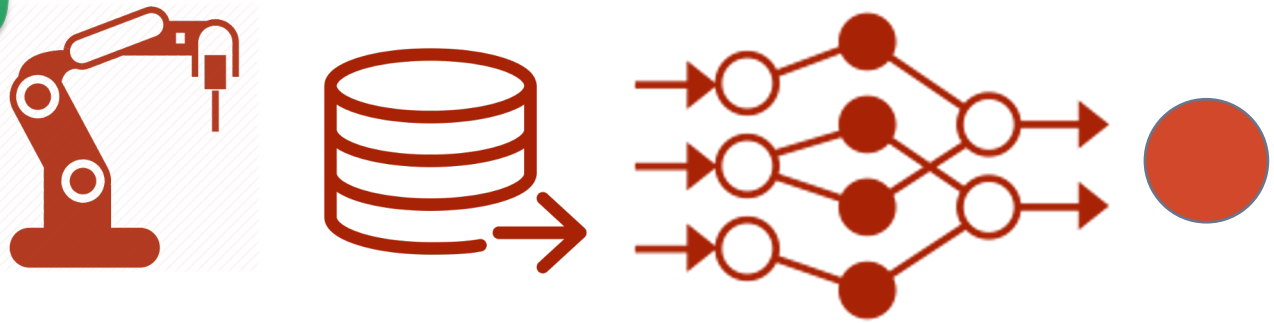
Machine Learning Framework for Sparse Data

Maximizing the utility of existing data

Model trained on simulation data

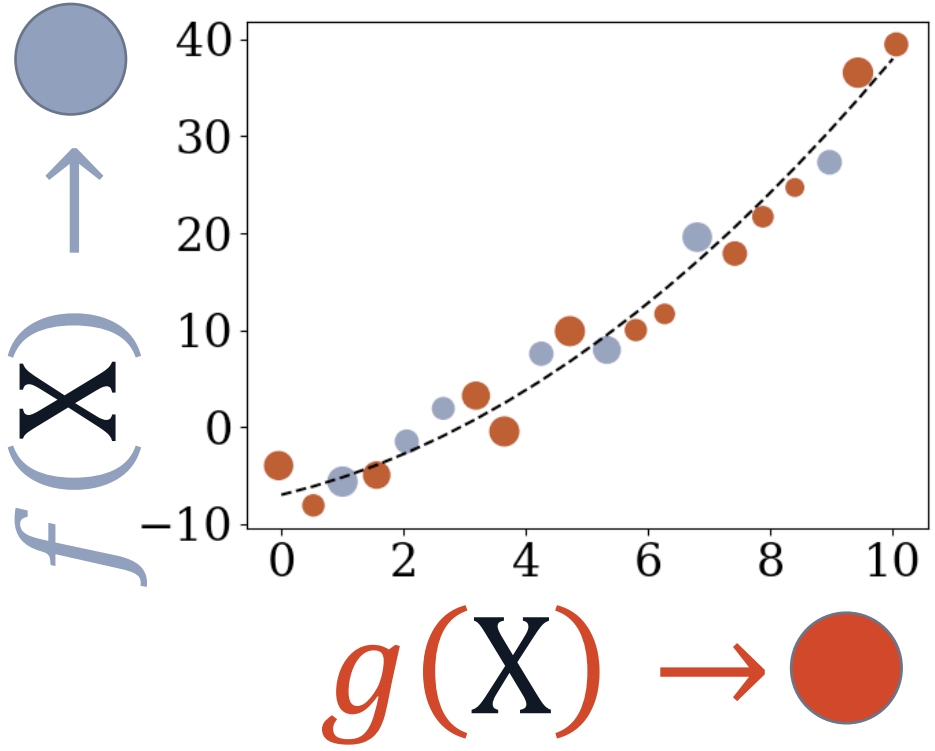


Models map build parameters to a target, such as bead width



Model trained on experimental data

Regression model to map between (●) and (●)

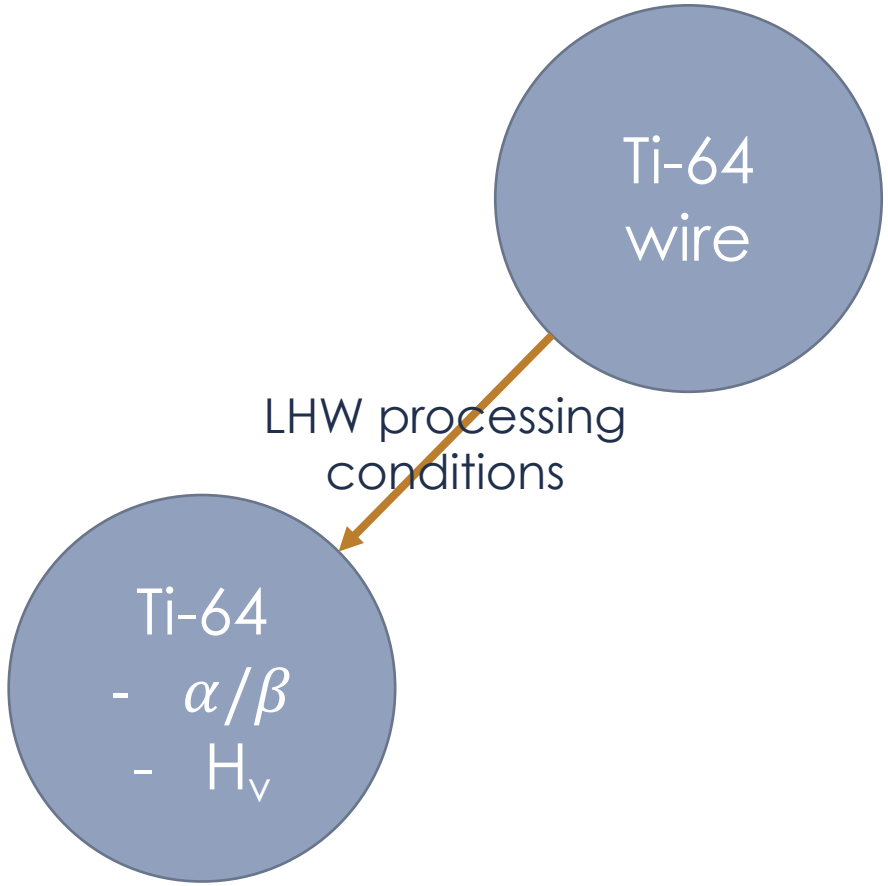


adapt

Information accumulates along the product life-cycle

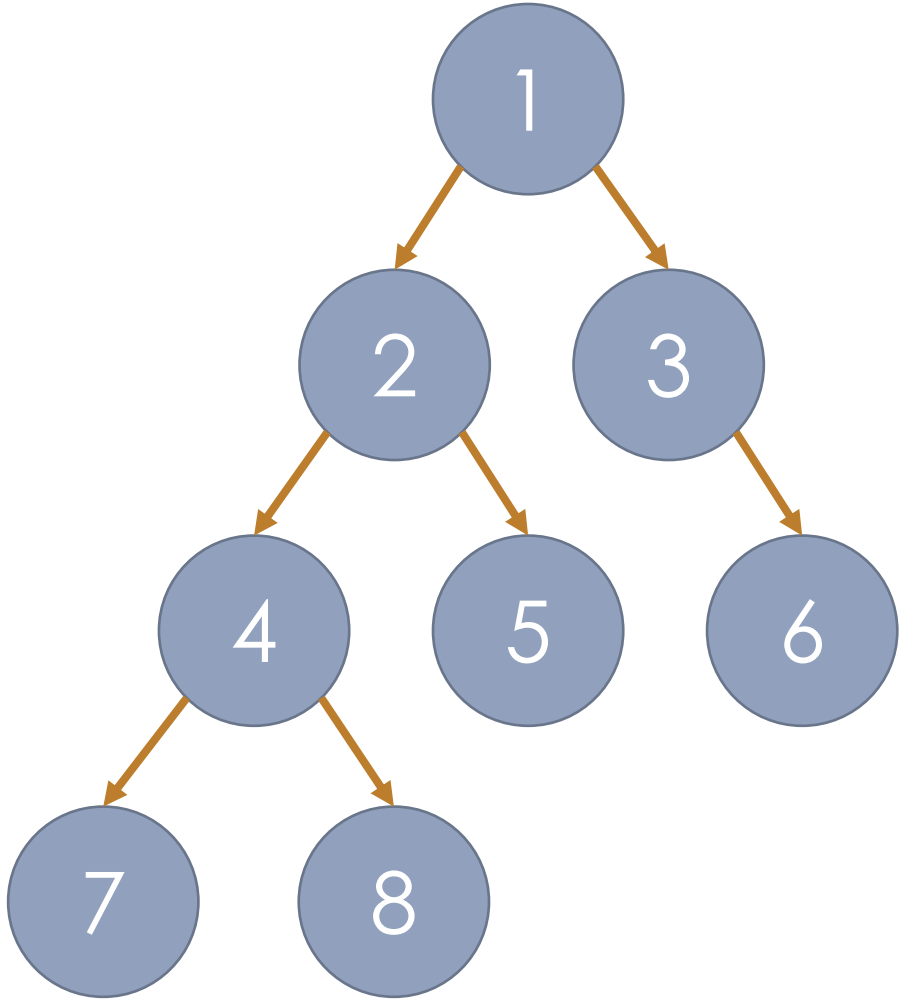
- Processes change state.
- Materials carry that state in their structure.
- Structure dictates properties.

Process information is collected asynchronously and must allow for dynamic updates



adapt

Aggregating and propagating data up and down the process hierarchy



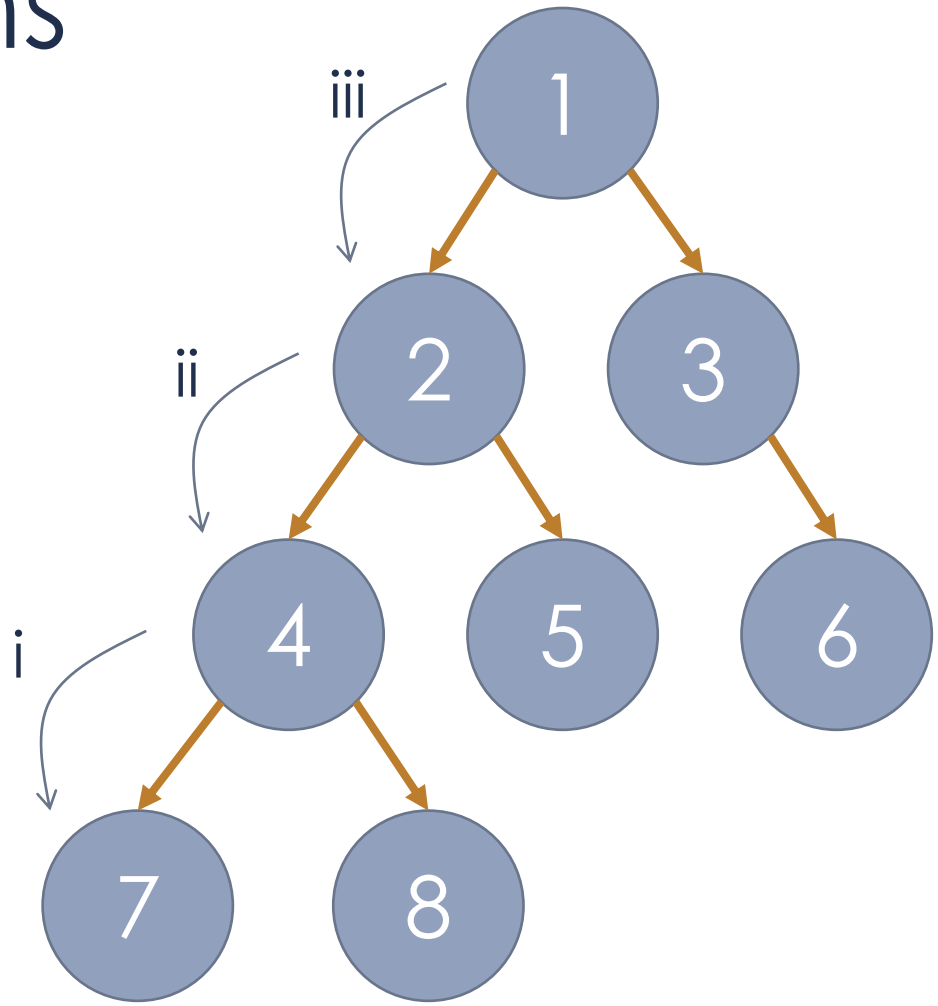
1. Data collected by LE during print
2. Samples sent to CSM
3. Samples sent to CMU
4. CSM sectioning
5. CSM Metallography
6. CMU Metallography
7. CSM tensile testing
8. CSM electron microscopy

adapt

Connectivity establishes relationships between generations

Parents pass their properties onto their children through **propagation**

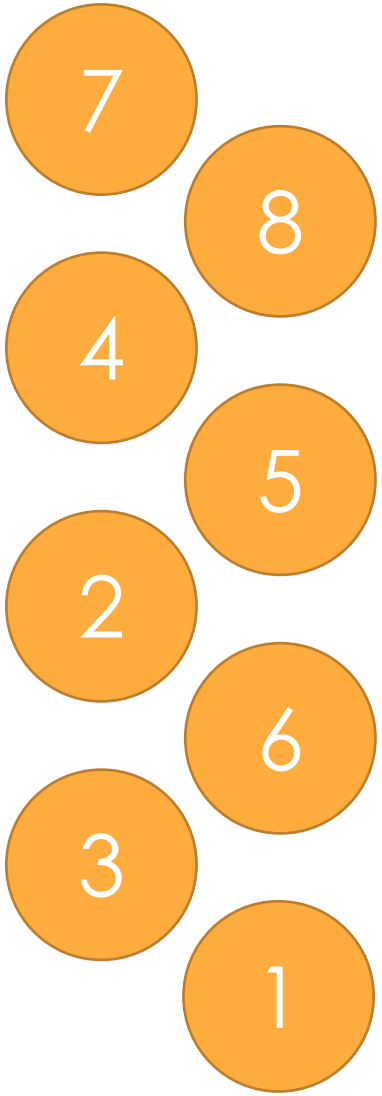
- These form the conditions (independent variables) under which the properties (structure) of the children are realized.
- Keep the property of the nearest node.

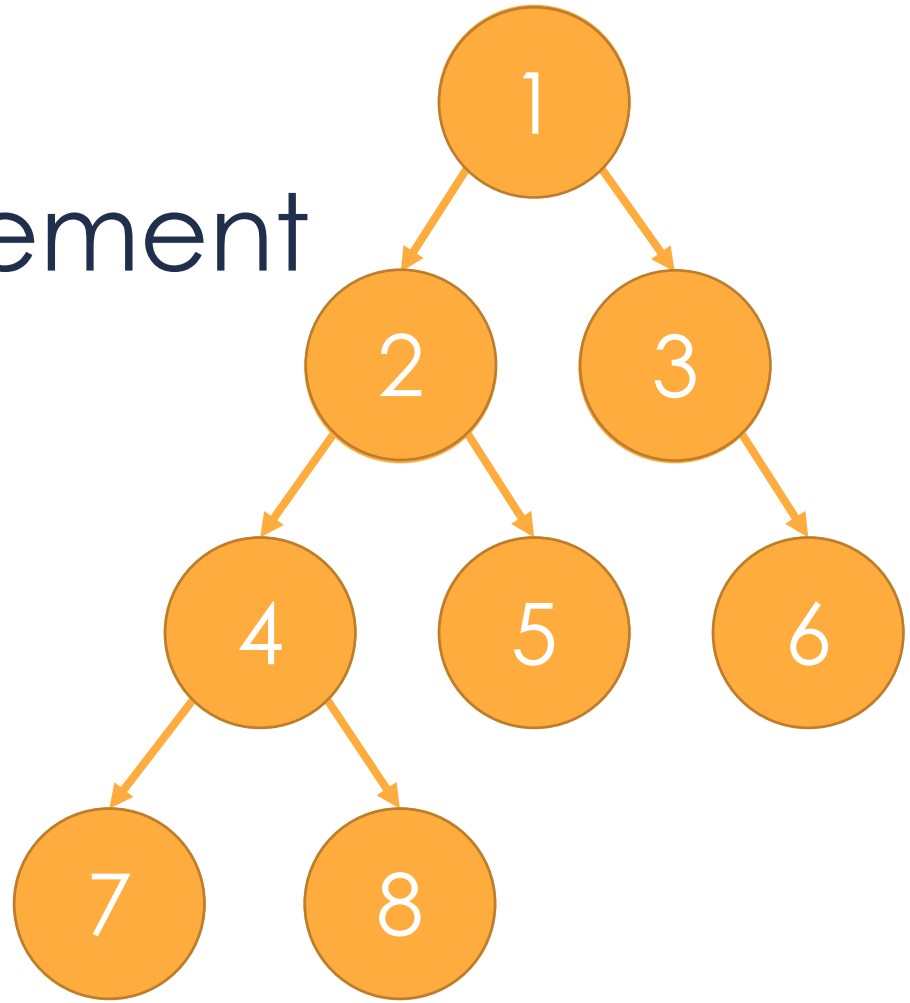


Parent initiates communication
Child accepts/rejects communication

adapt

Postorder Tree (LRN Tree) retrieves properties from nearest measurement



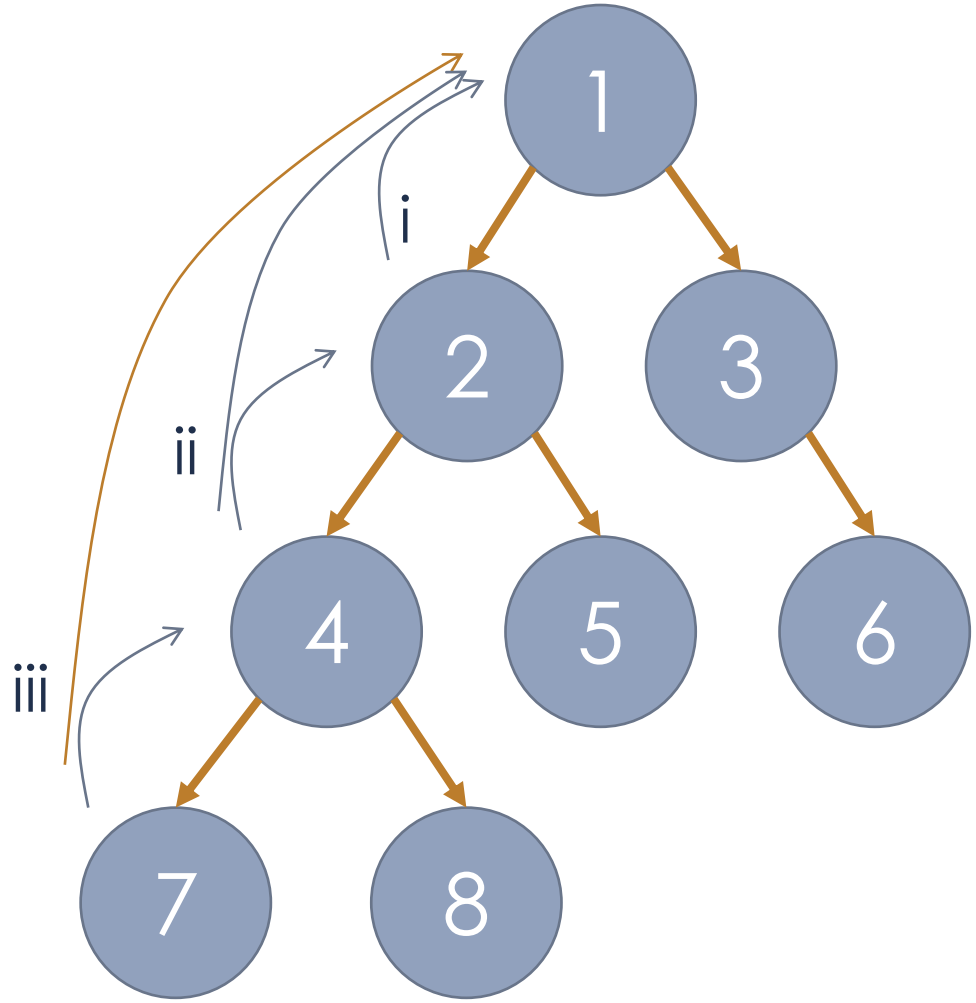


adapt

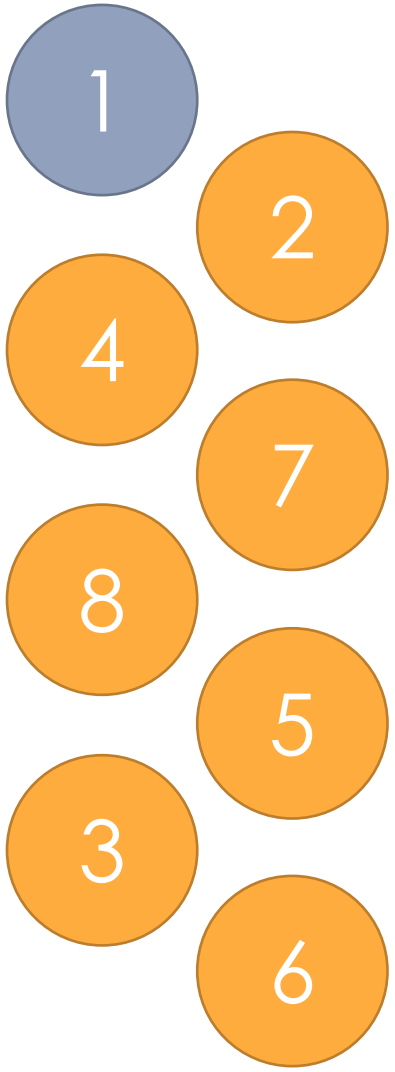
Establishing relationships between generations

Children reflect the properties of their parents through **aggregation**.

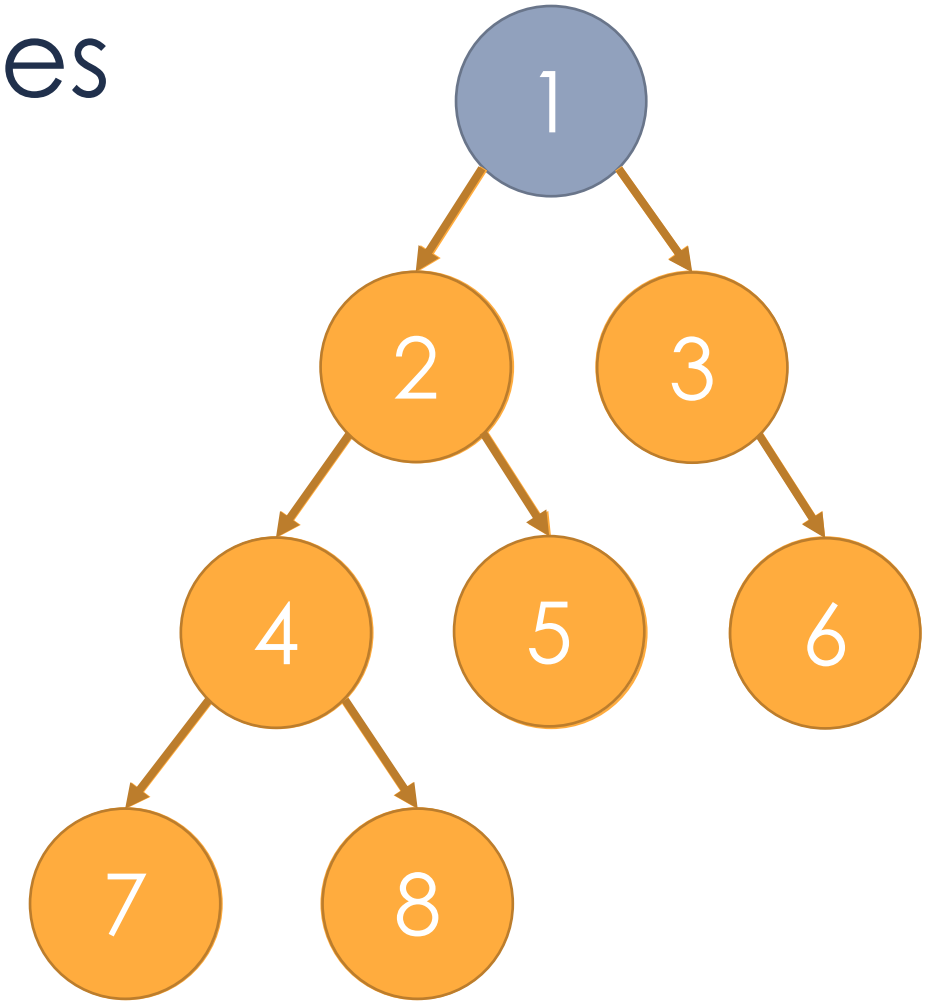
- The property (structure) of the parent is the same as the property of the child unless a process has fundamentally altered this relationship.
- The property of the parent may be a **reduction** of the properties of the children.



Parent initiates communication
Child accepts/rejects communication



Preorder Tree (NLR Tree) pushes processing information to downstream samples



Differentiating low variance measurements from unique classes

- Weld log data is collected layer-wise.
  - Multiple layers compose one part.
  - Properties are not the statistical summary of constituents, but the accumulation of state over space and time.
- Materials characterization data is collected sample-wise.
  - Multiple characterizations summarize part properties.
  - Properties of the whole are described by the statistics (mean, mode, median, max, min, and standard deviation/error) of the set of measurements.
- Downstream operators cannot reasonably be expected to know which is appropriate for any given property.

Data Preprocessing

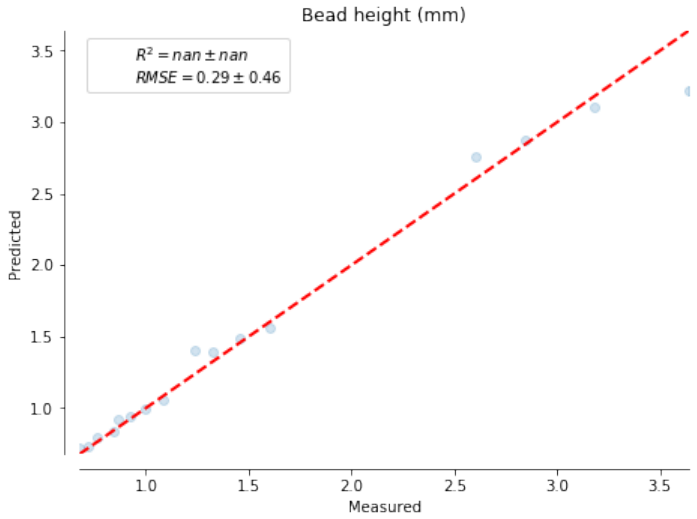
Sparse data presents a challenge for machine learning, particularly when only tens-of-records with a fully dense complement of data are present.

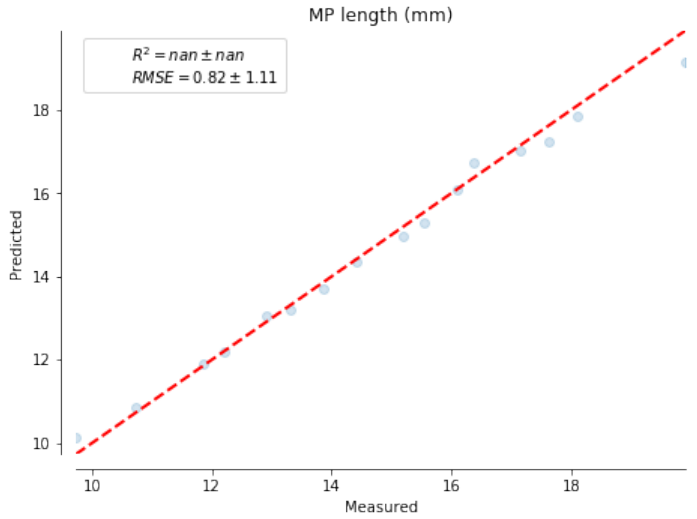
Simulation data

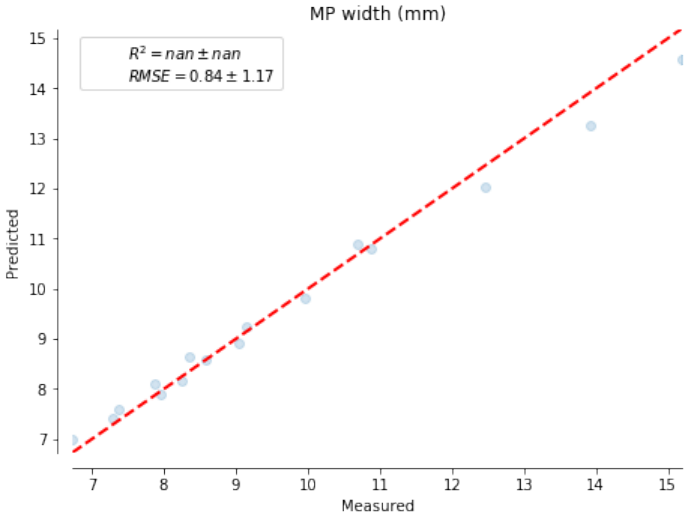
Typified by a dense input/output space, “missing” values are treated as control variables.

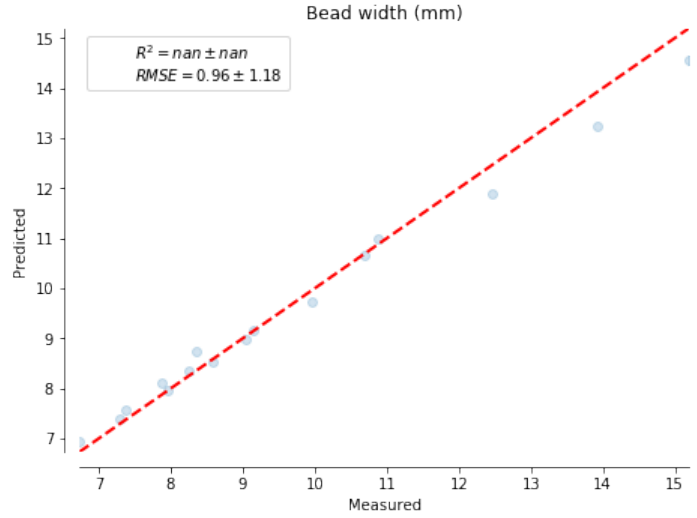
Handling missing data: Exclusion

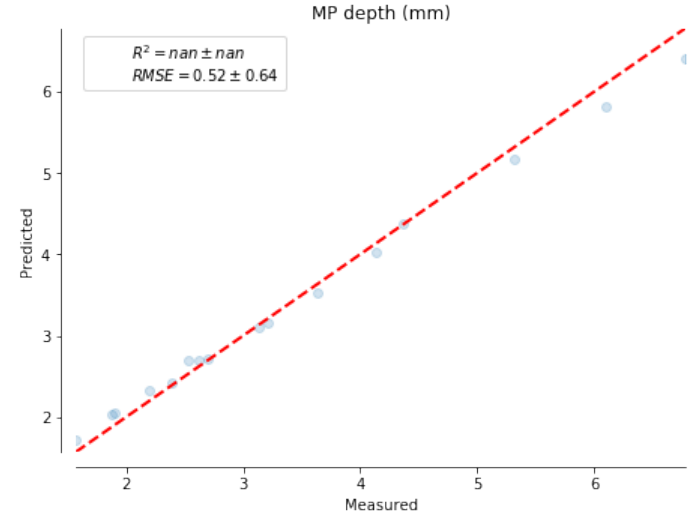
Model excludes control variables

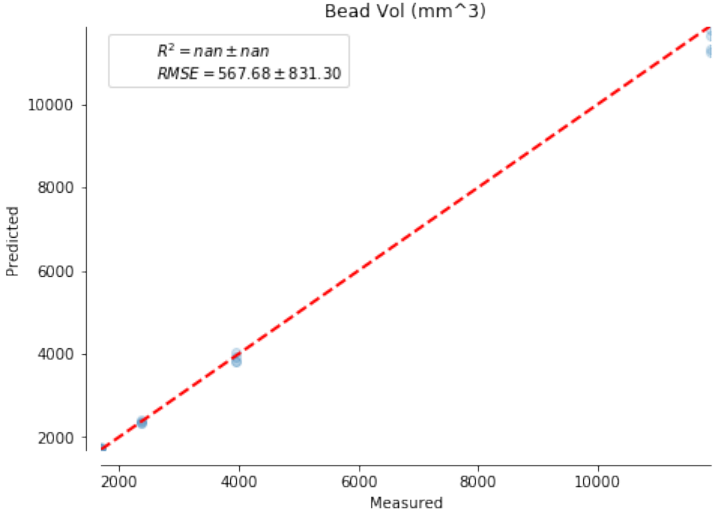












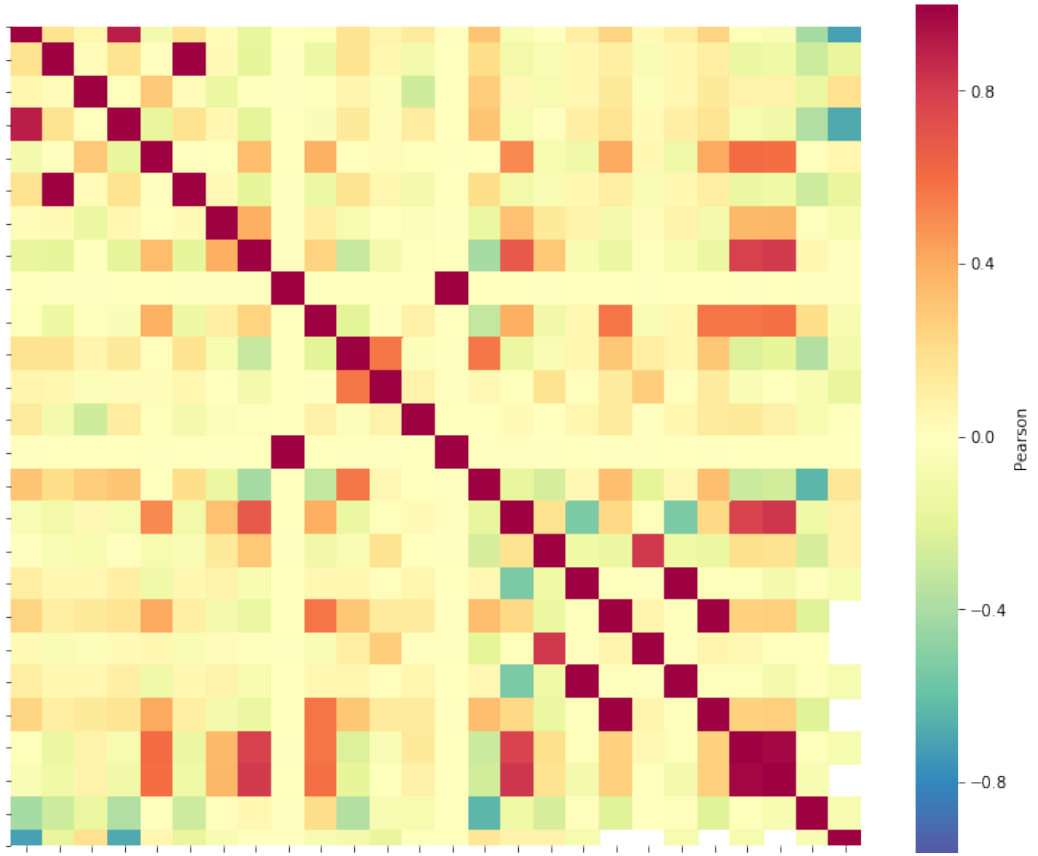
Experimental data

The asynchronous and federated nature of experimental data collection increases data sparsity.

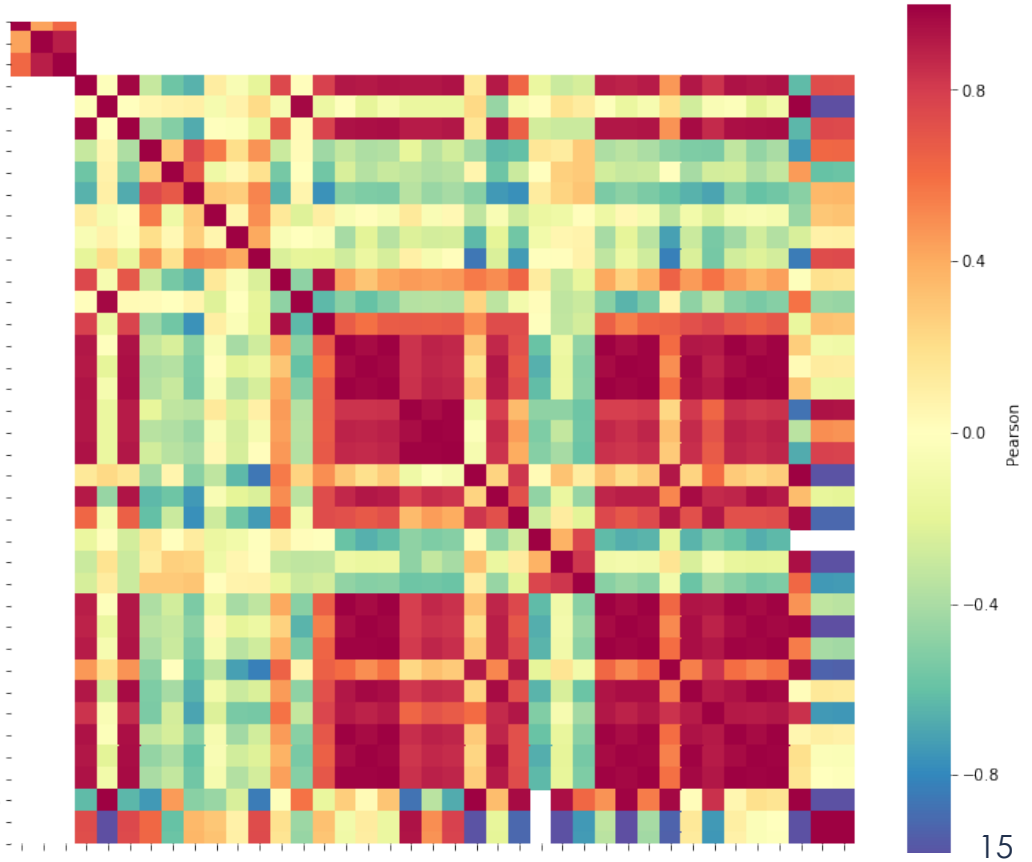
adapt

Dense correlation matrices belie missing data

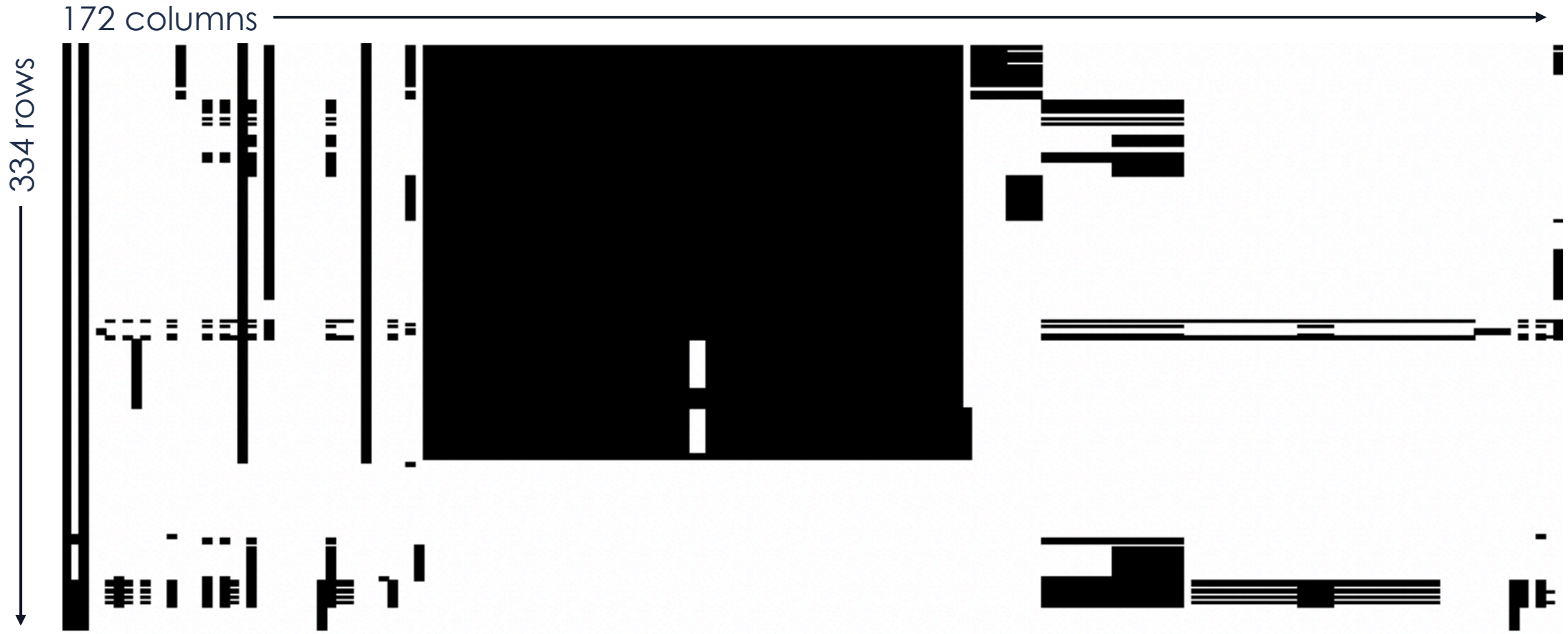
Inputs



Outputs



Data imputation decreases sparsity



17,556 entries = 30.6% coverage

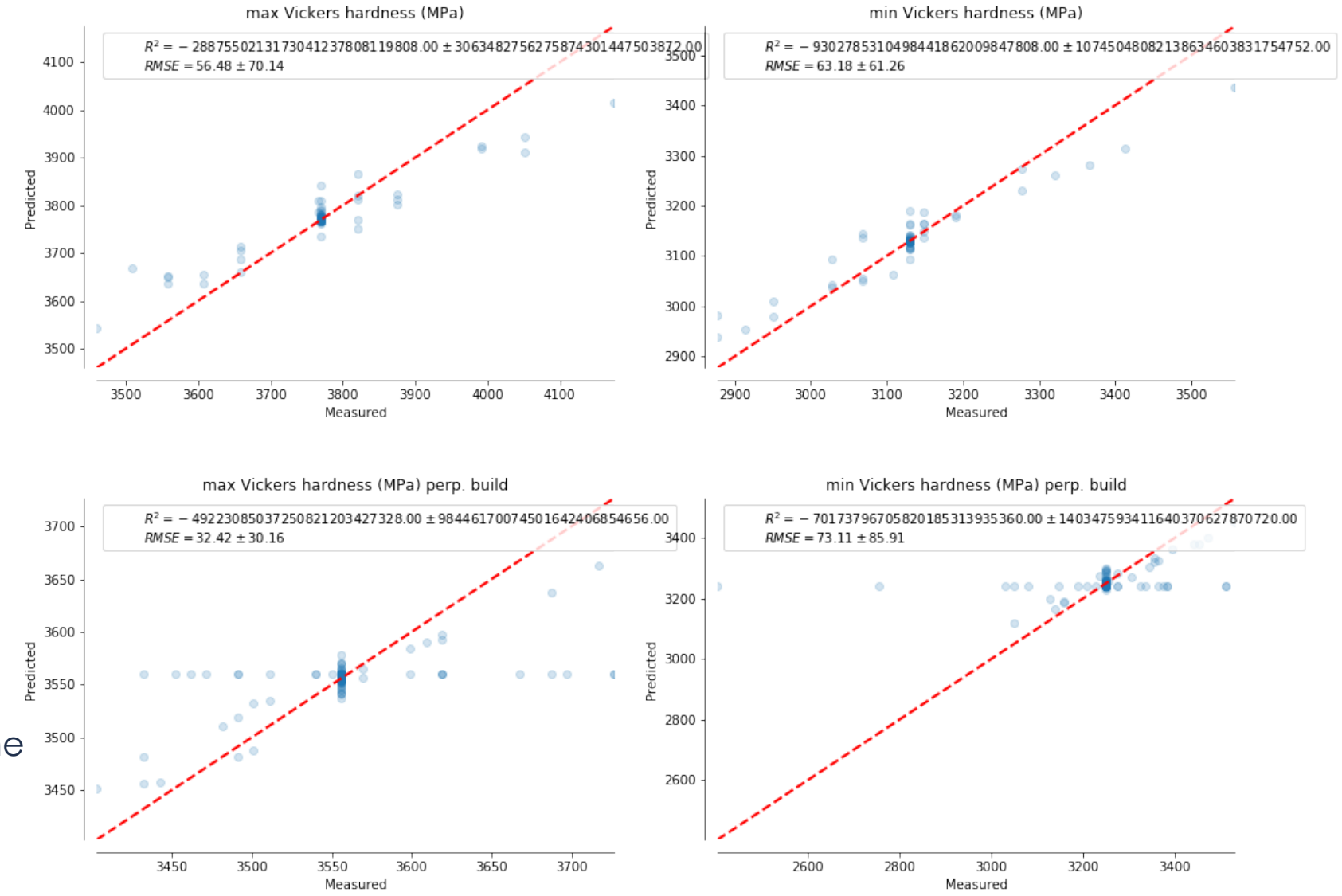
QM data set as of 4/23

Mean imputation

Variance in the response is not explained by the model since all imputed inputs are set to the same (mean) value regardless of the output.

Predicted vs. actual plot follows 1:1 line. Model does not depend on imputed values.

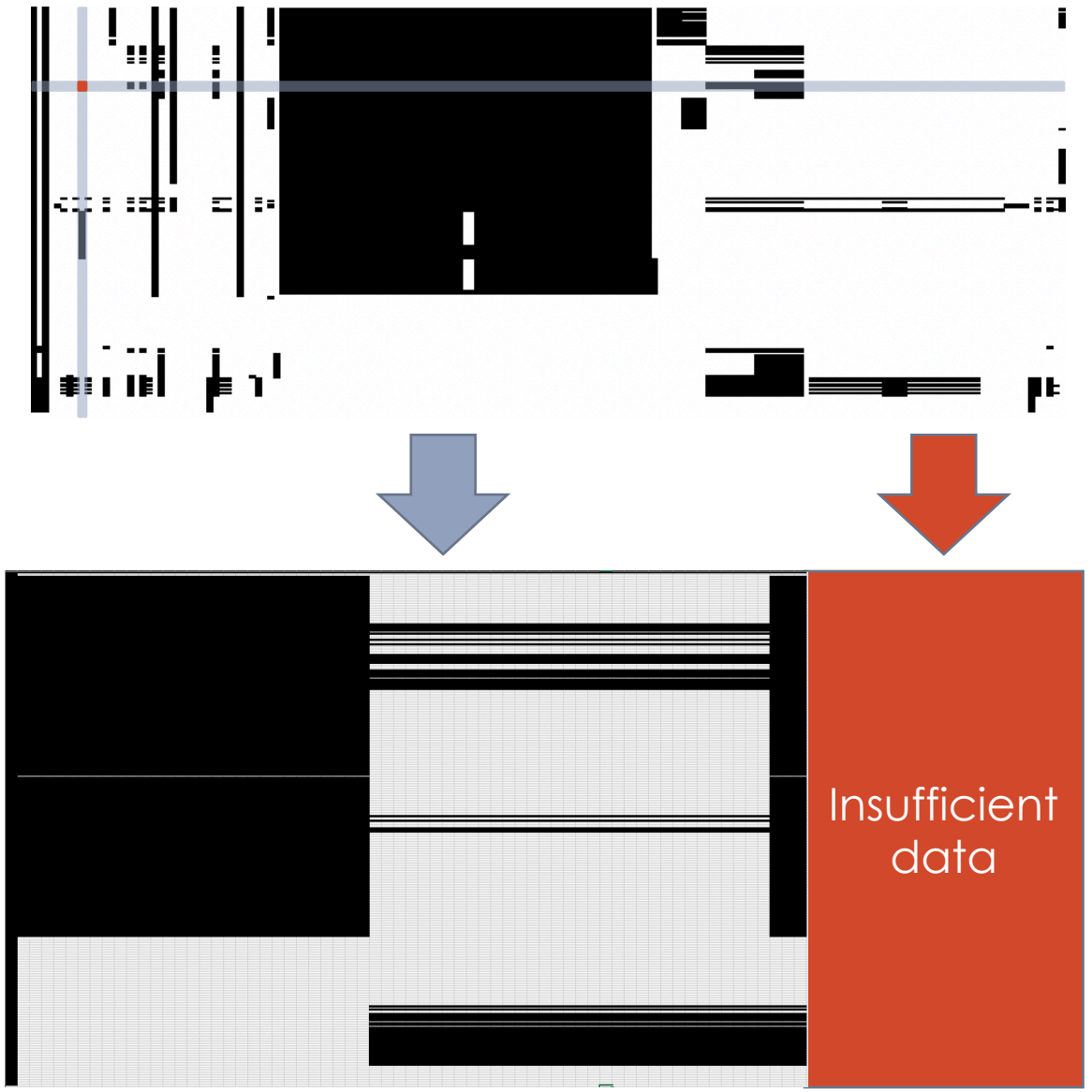
Horizontal line in the predicted vs. actual curve. Model depends on the imputed values.



Maximizing information hidden in sparse data sets

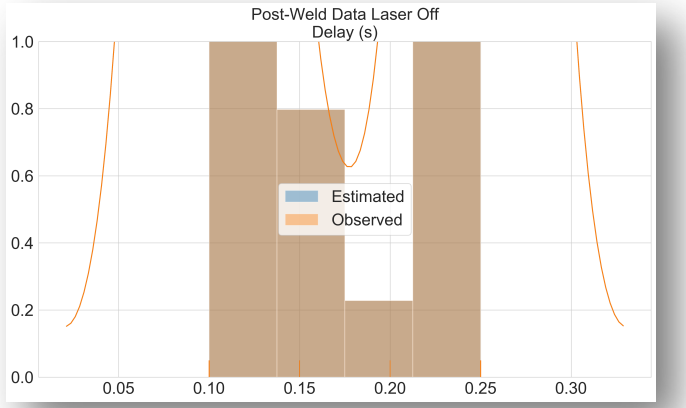
Corrletions exist between the measured response (output) and the conditions that lead to that response. Therefore, train a unique ML model for **every missing entry**.

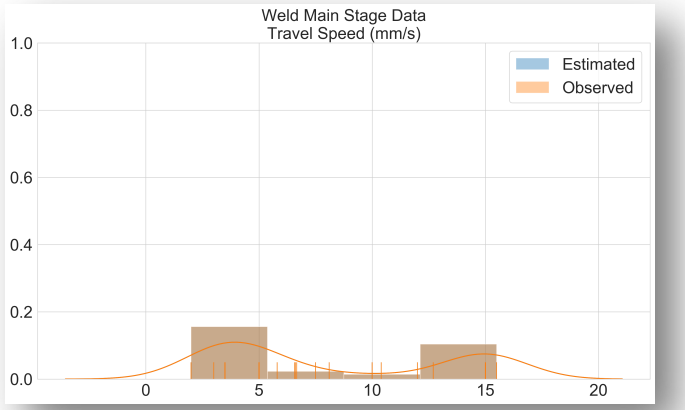
1. Select a missing entry, whether input or output (orange).
2. Choose non-empty rows and columns (from those shaded blue).
3. Exclude missing data.
4. Split, train using 5-fold CV.
5. **Fill in missing value with $\hat{y} \pm N(0, \hat{\sigma}^2)$**
6. Repeat for all 39,892 missing entries.

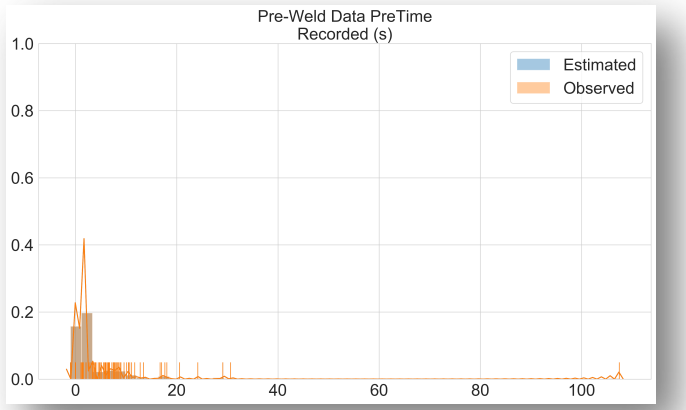


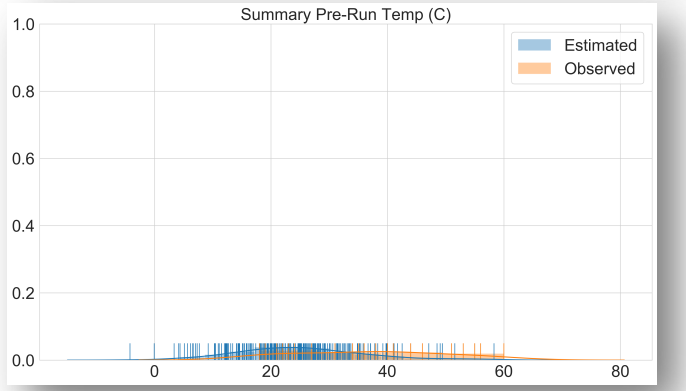
adapt

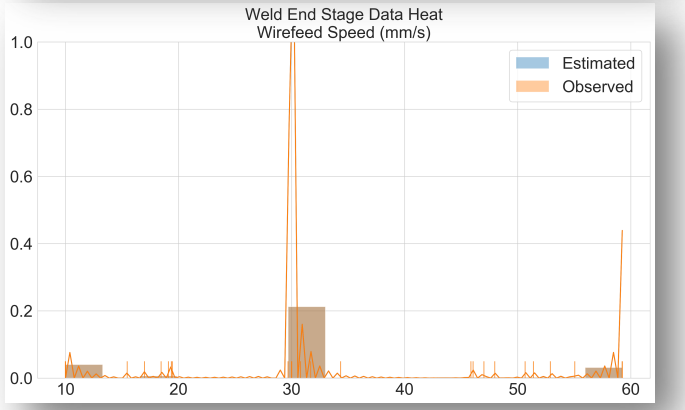
Imputation of Input Features Follows Existing Trends

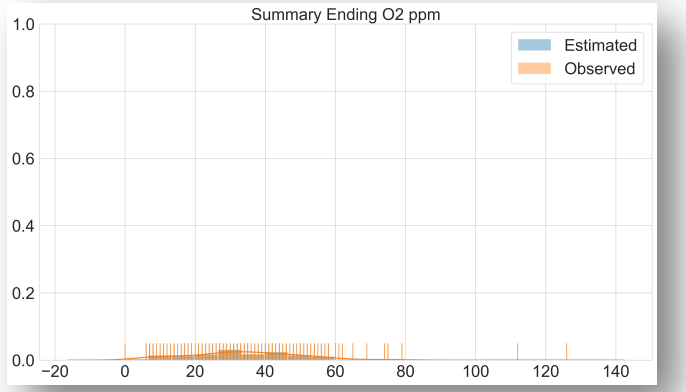


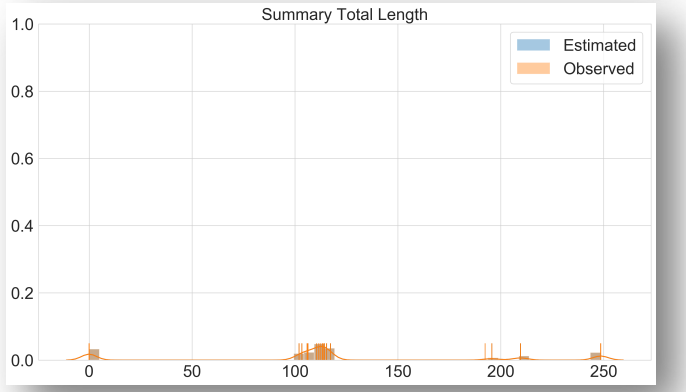












Data Preprocessing Conclusions

- Physical experiments and simulation data present distinct modeling and informatics challenges.
  - Point estimation of missing values, both inputs and outputs, across modalities.
- Data collected from physical experiments require asynchronous update.
- Handling missing data
  - Simulation data: Exclusion
  - Experimental data: RFN imputation
- Next steps
  - ML model development and hyperparameter optimization.
  - Physical experiment—simulation regression development.

ML Model Development

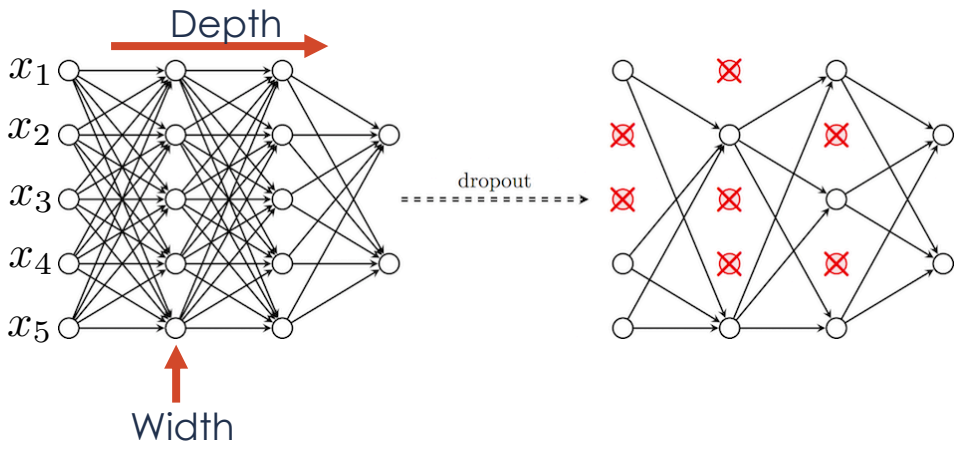
Experimental and simulation data provide information into the QM LHW process, but these data are supported on different basis sets.

Hyperparameters control model performance

As much as the data fed into the model, model hyperparameters control the usefulness, or uselessness, of a machine learning model. These hyperparameters,

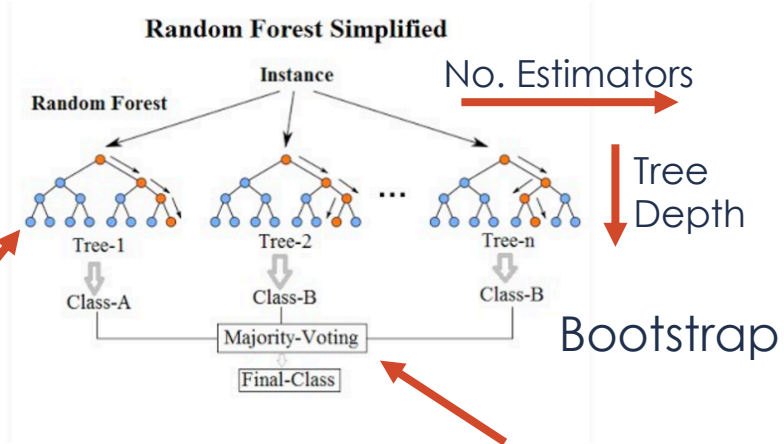
- Control model complexity,
- Reduce overfitting,
- Increase generalizability,
- Improve training efficiency,
- Differentiate classifiers from regressors, …

therefore, model hyperparameters must be **tuned**.



Neural Networks

Random Forest Networks



https://perso.mines-paristech.fr/fabien.moutarde/ES_MachineLearning/TP_convNets/drop.png
https://miro.medium.com/max/592/1*i0o8mjFfCn-uD79-F1Cqkw.png

Model Performance and the Bias-Variance Tradeoff

ML models are statistical models (*maximum likelihood estimators* or *maximum a posteriori estimators*, but we'll leave that for now) that can represent arbitrary functions.

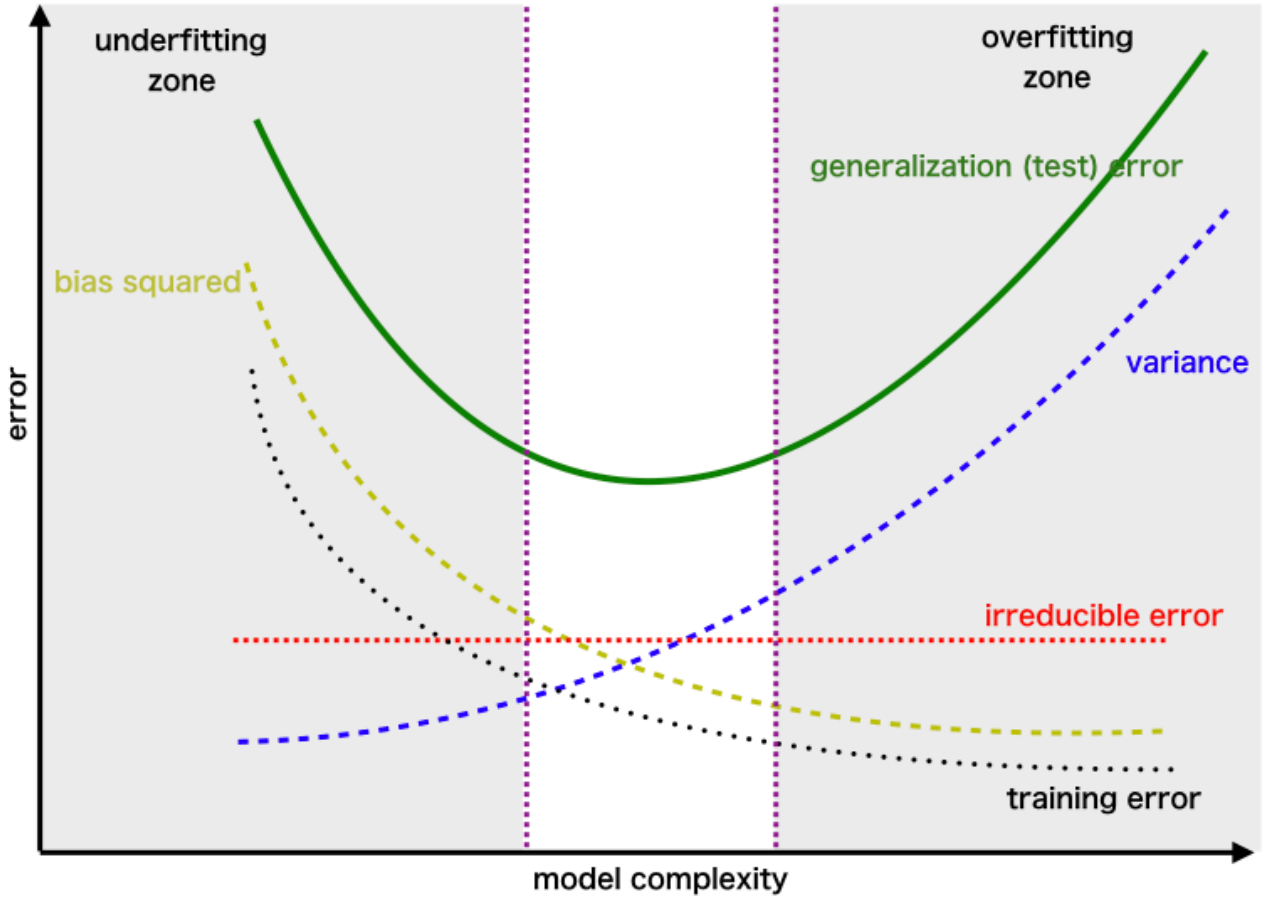
In this way, they are like Taylor Series, Polynomial Series, Fourier Series, etc.

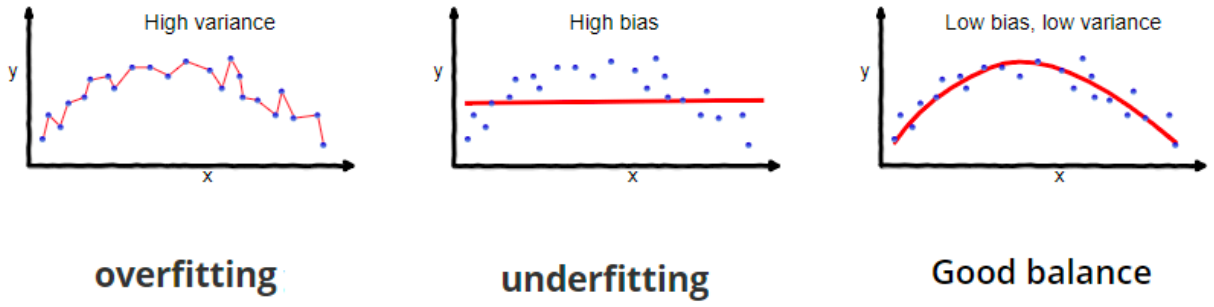
Like practical Series expansions, one must decide at what order to truncate the series,

$$f(x) = \sum_{k=1}^{\infty} a_k x^k \rightarrow \sum_{k=1}^{N} a_k x^k$$

too low: high bias

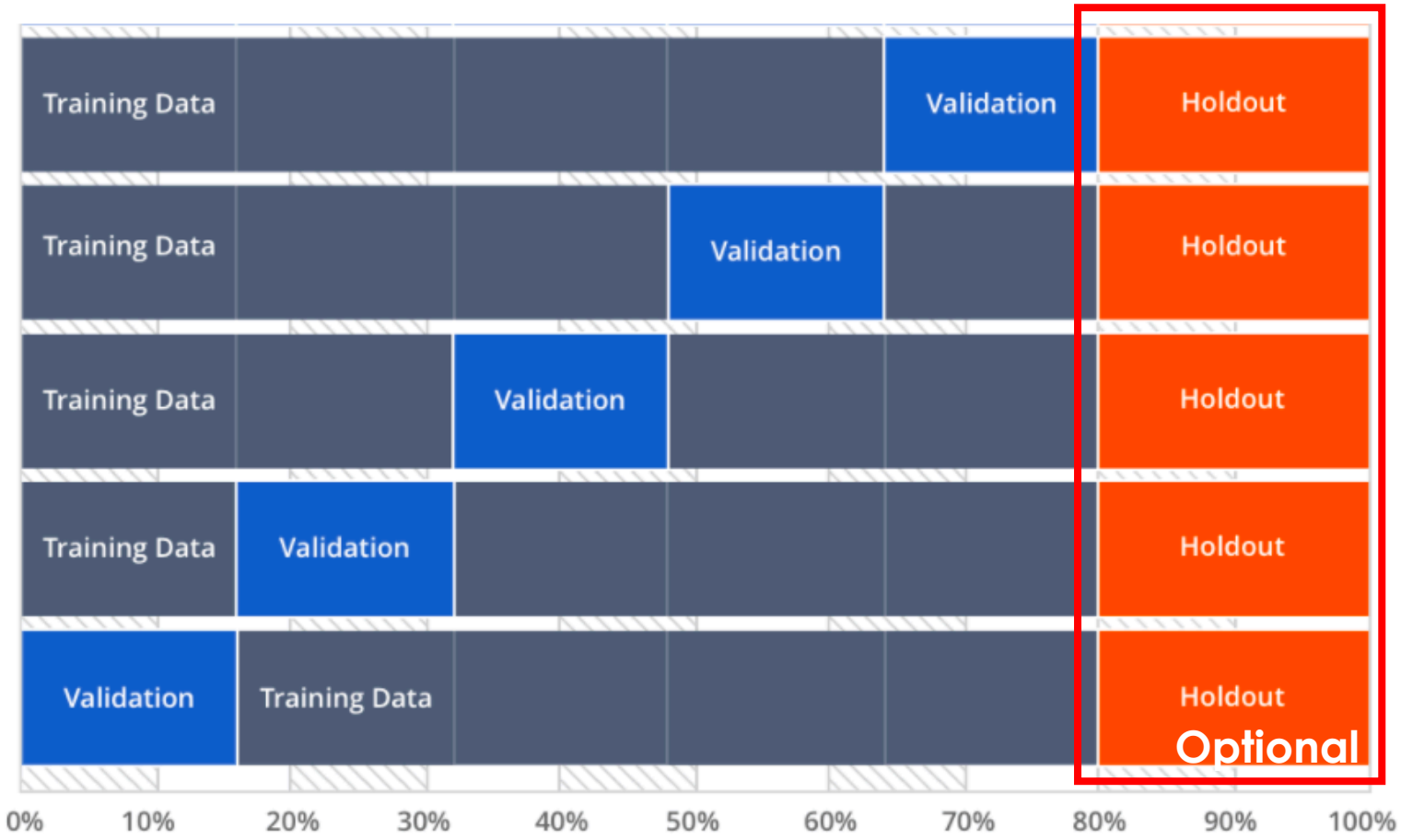
too high: high variance





adapt

Cross-Validation (CV) and GridSearch: Optimizing model hyperparameters



Cross-validation provides an **out-of-sample error estimate**: how the model will perform on new, unseen measurements. (Five iterations in this case.)

This process is repeated for each unique combination of hyperparameters (GridSearch).

The model with the lowest out-of-sample error is kept.

https://3gp10c1vpy442j63me73gy3s-wpengine.netdna-ssl.com/wp-content/uploads/2018/03/Screen-Shot-2018-03-21-at-4.26.53-PM.png

ML Model Optimization Controls Deployment Efficacy

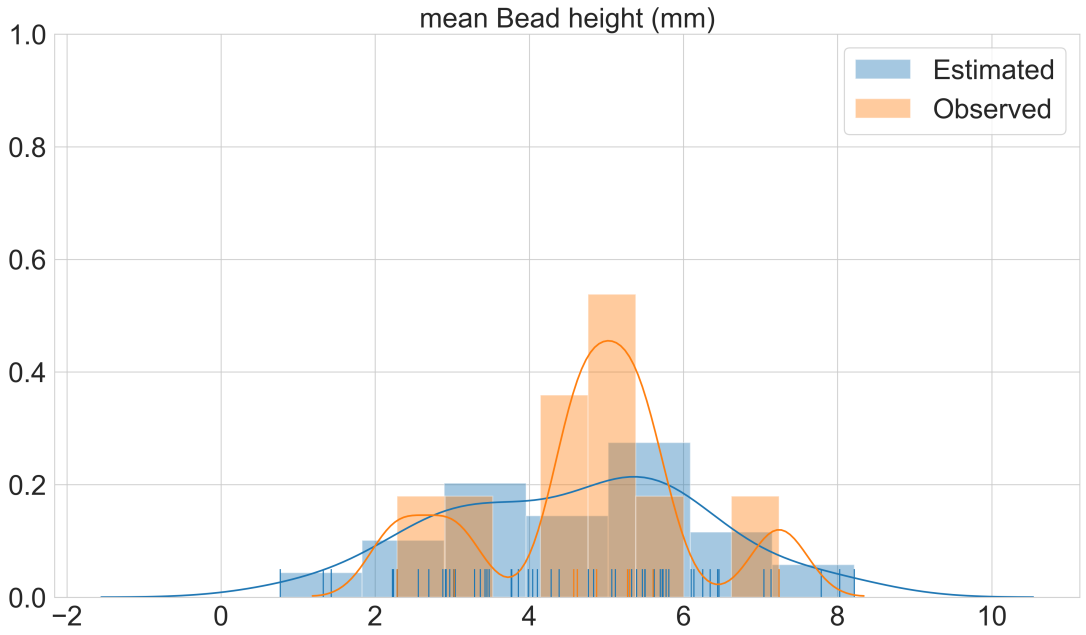
Bias-Variance Tradeoff

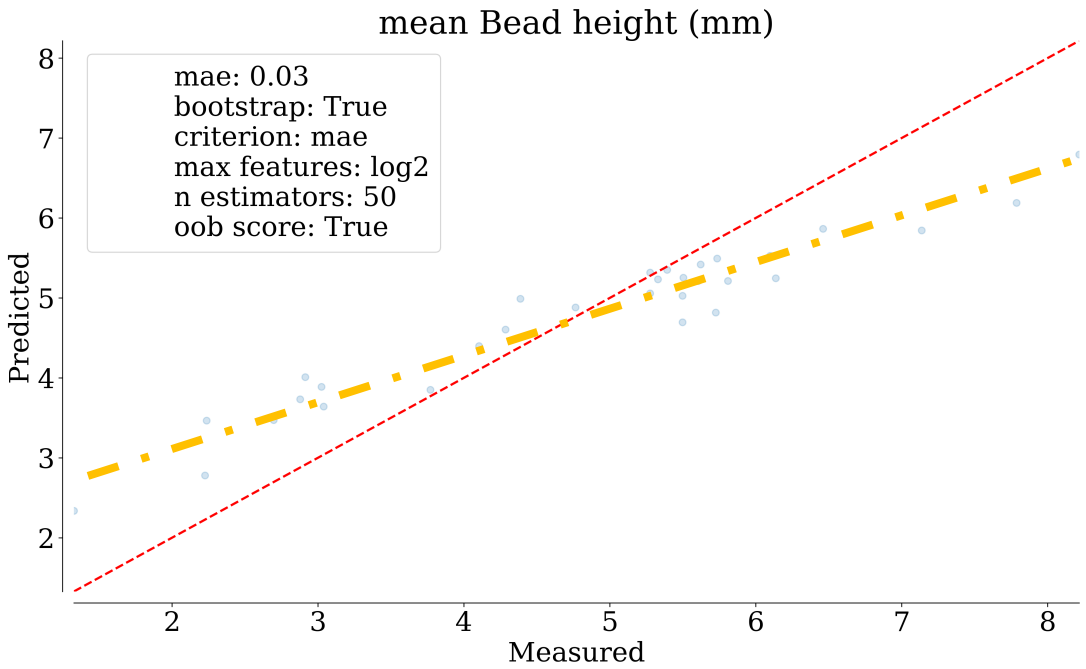
- Tuning hyperparameters.
  - Control model complexity.
  - Combat overfitting.
- Estimating out-of-sample error.
  - Bootstrapping,
  - GridSearch,
  - Cross validation.

Random Forest (RFN)-specific

- Error metrics: MAE, MSE
- Bootstrapping (features)
  - 10%, 20%, 50%, 80%
  - $\sqrt{N}$, $\log_2 N$, N
- Bagging: 67%/33% split
- 5-fold cross validation

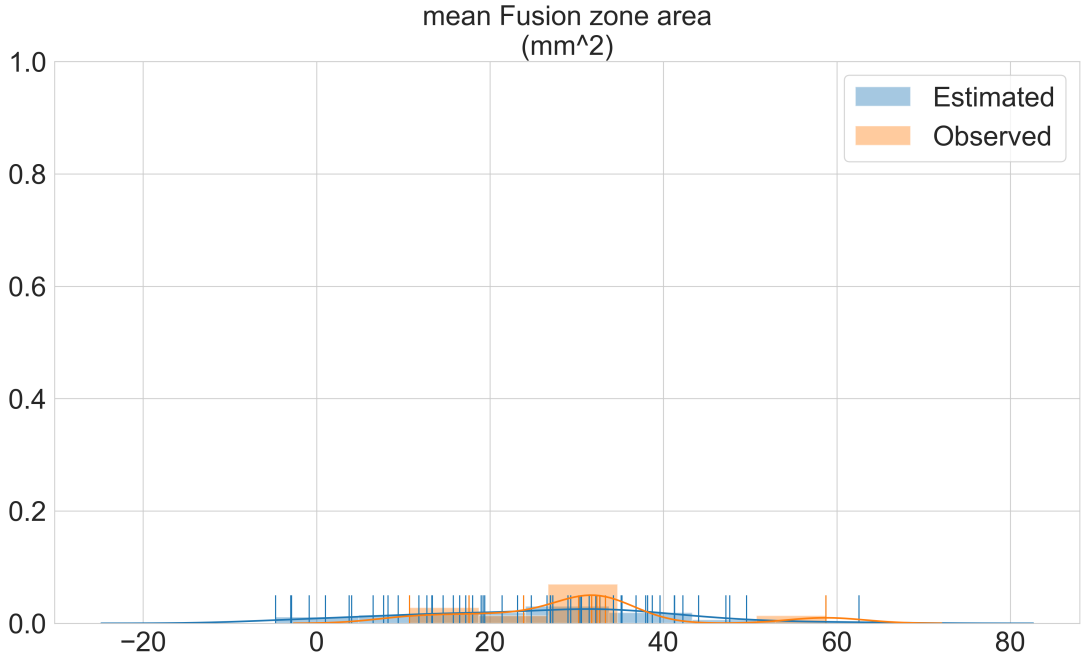
Bead Height

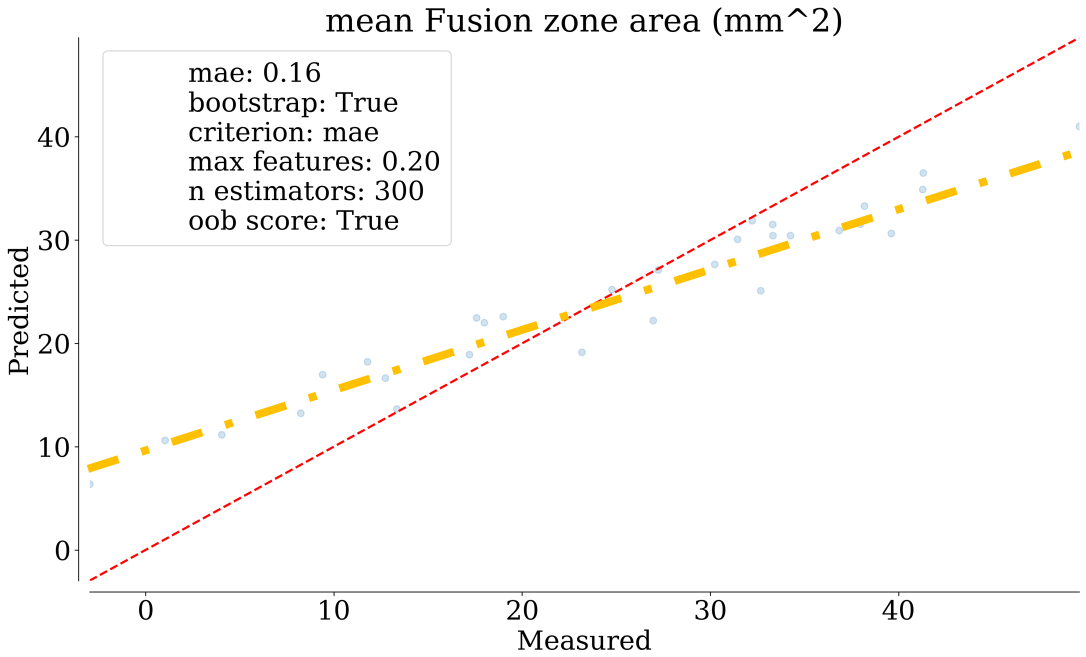




Priority: high

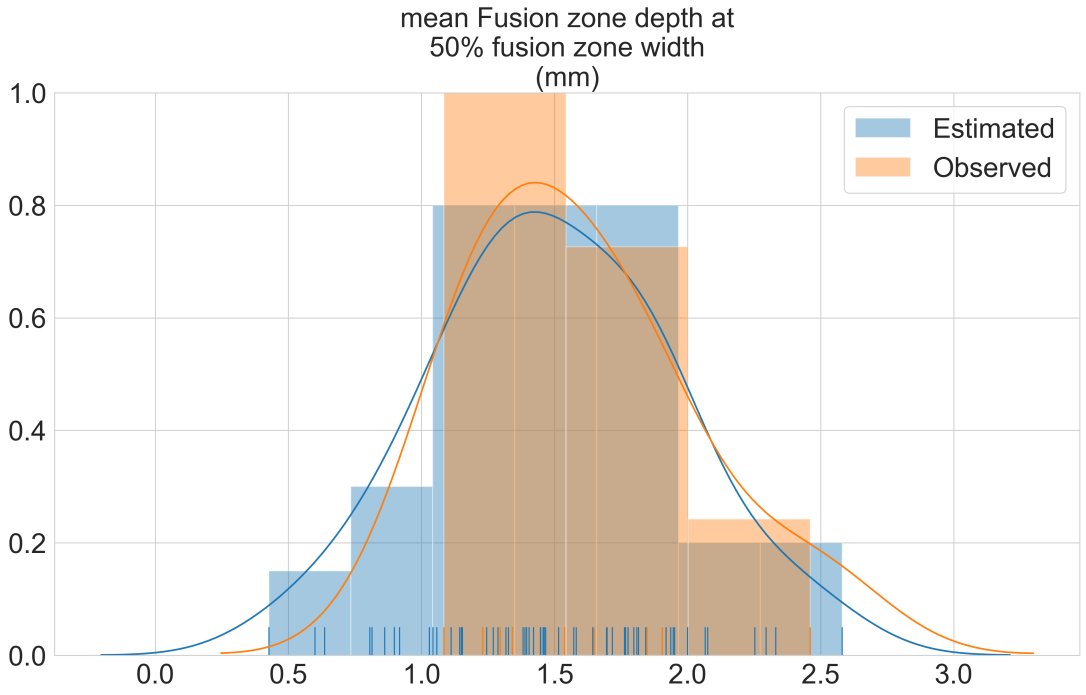
Fusion Zone Area

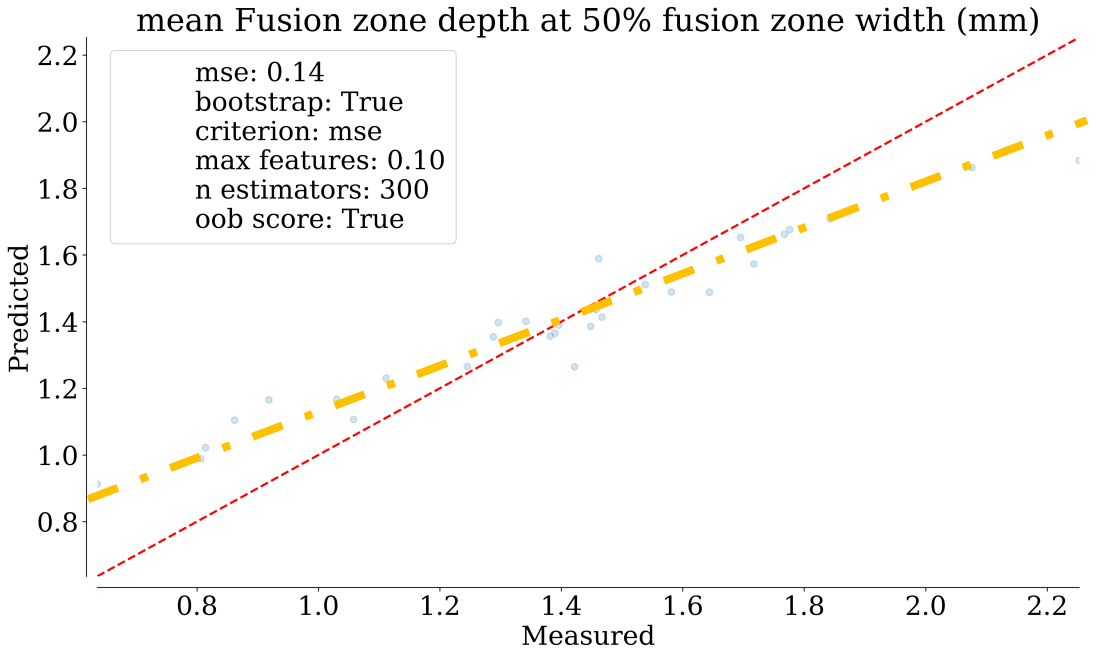




Priority: high

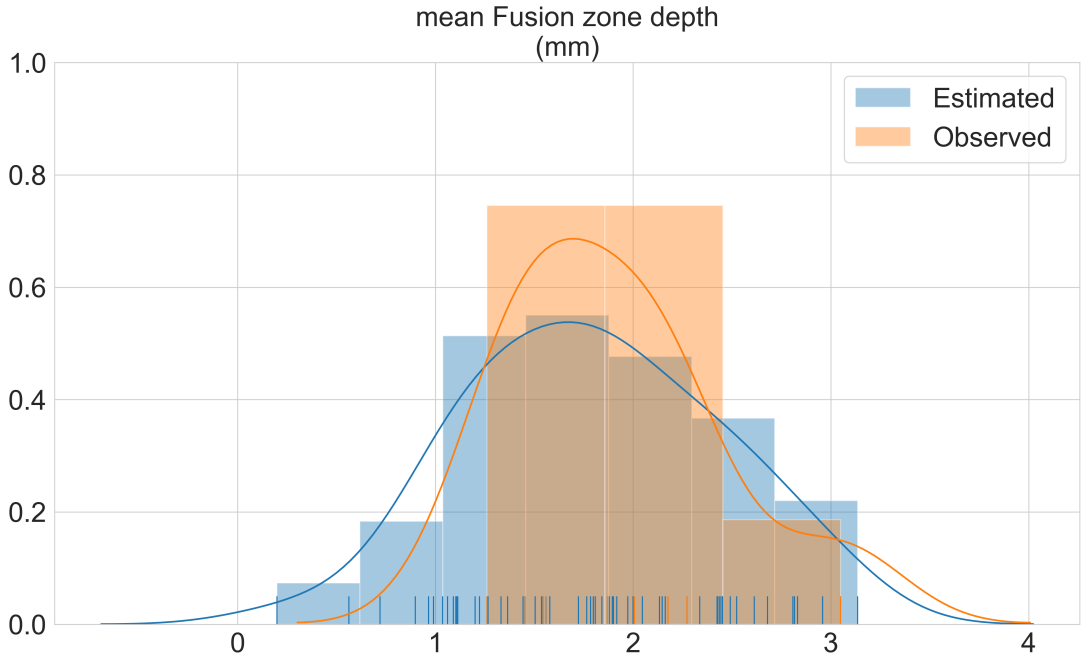
Fusion Zone Depth at 50% Bead Width

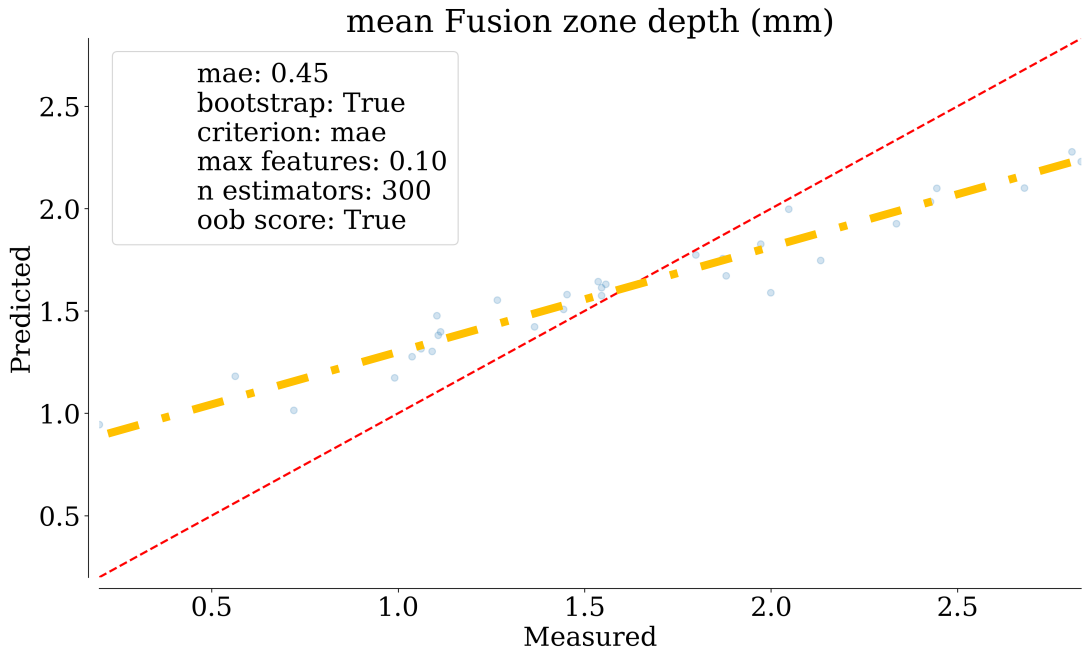




Priority: medium

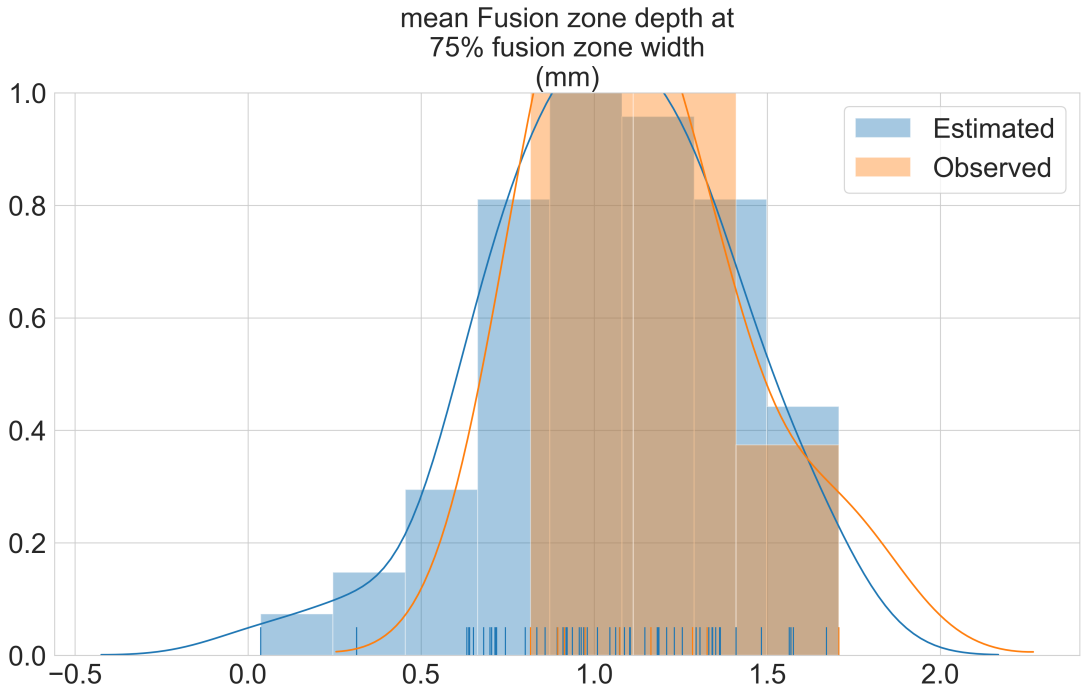
Fusion Zone Depth

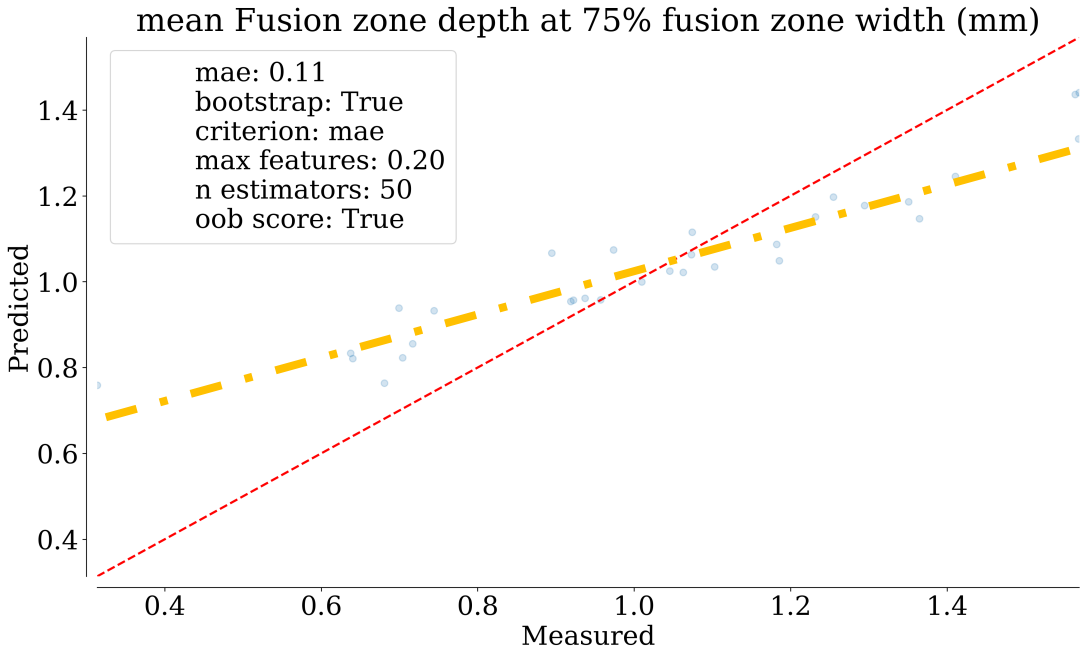




Priority: high

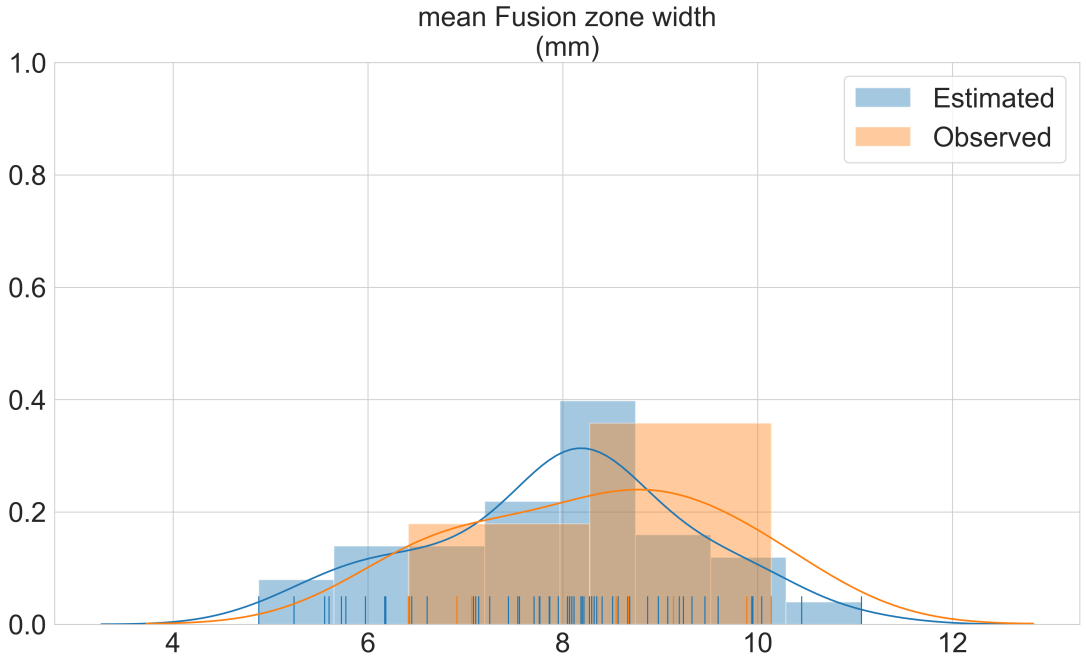
Fusion Zone Depth at 75% Bead Width

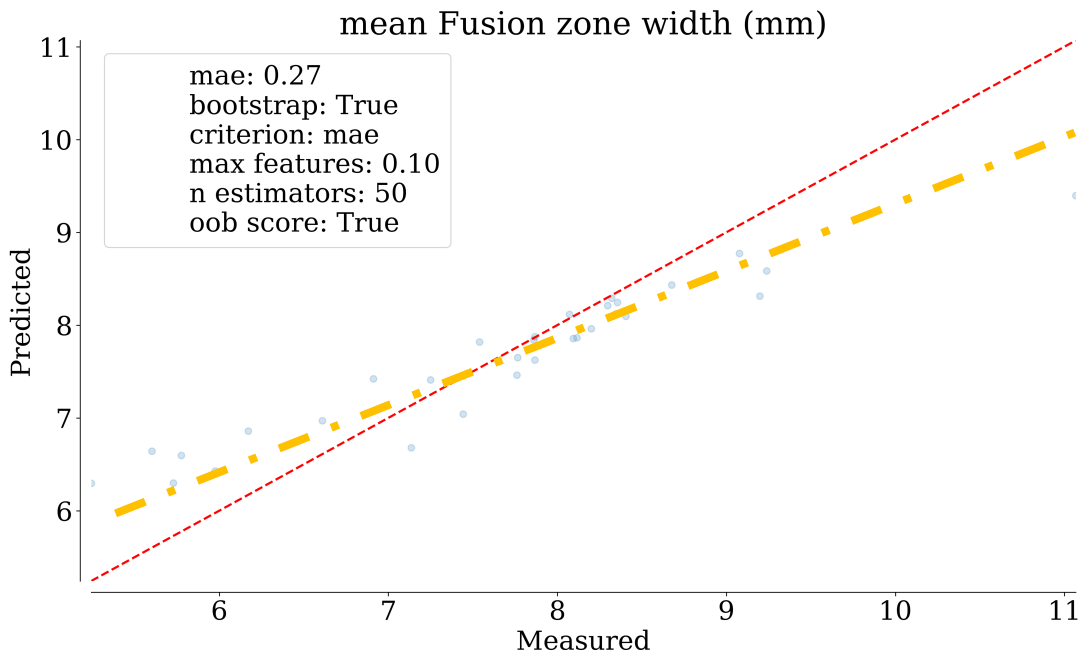




Priority: high

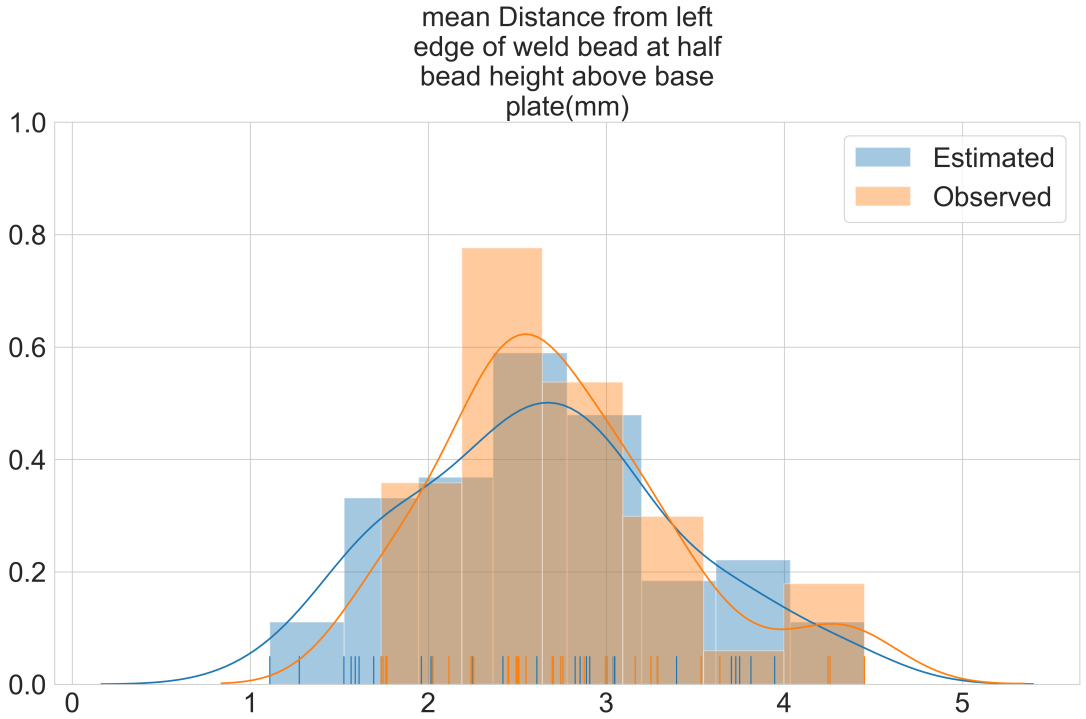
Fusion Zone Width

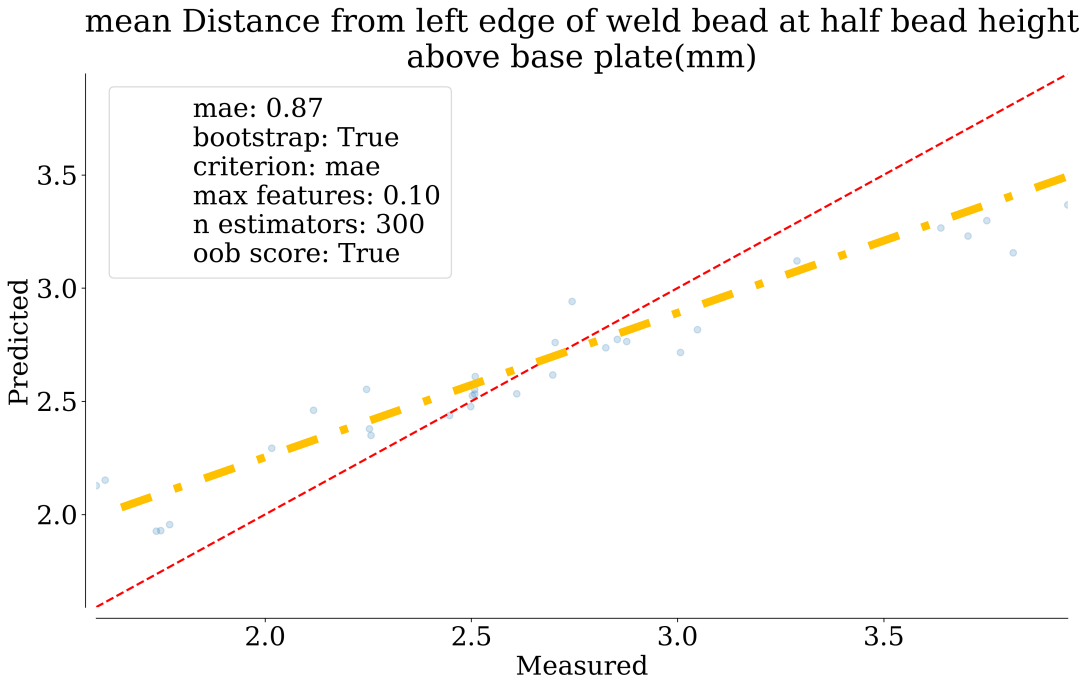




Priority: medium

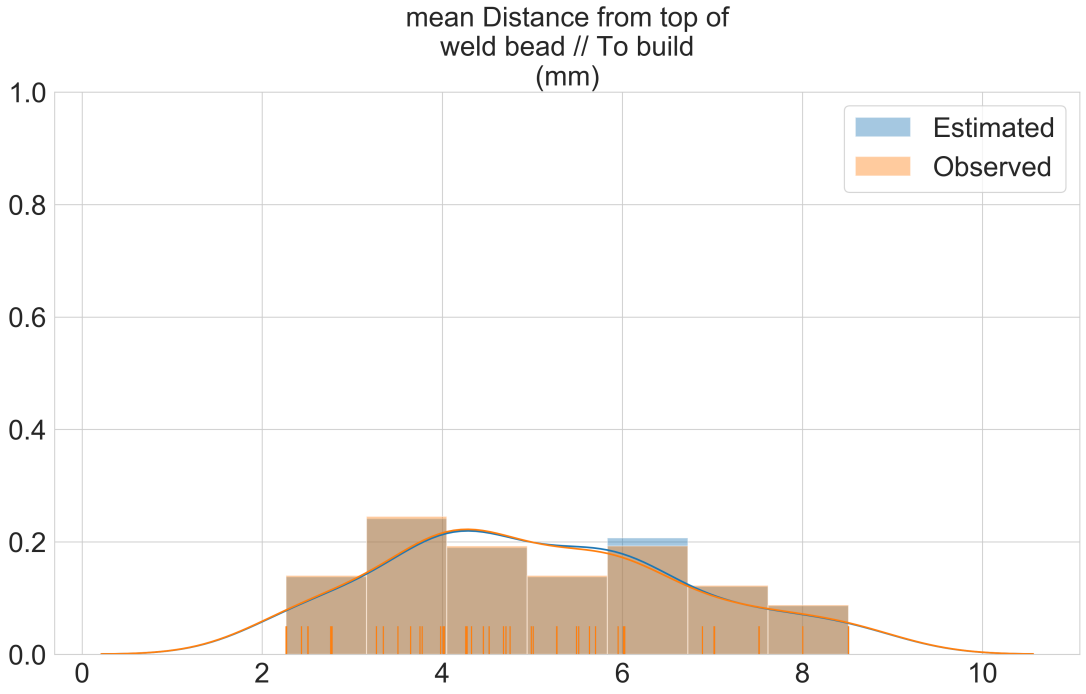
Weld Bead Left Edge at Half Bead Height

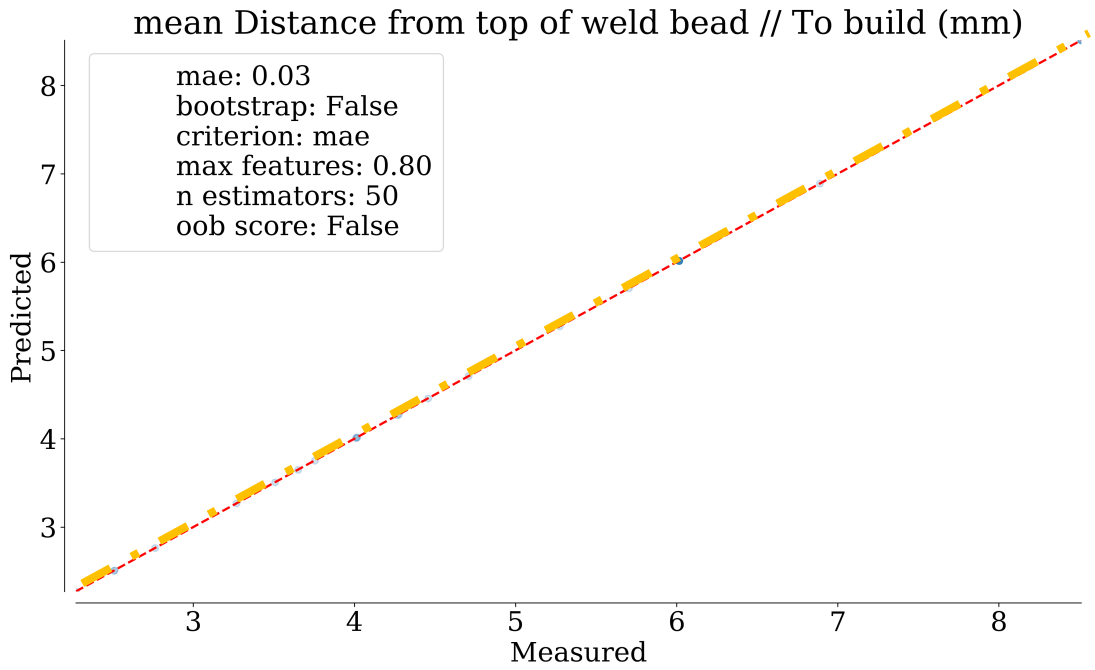




Priority: medium

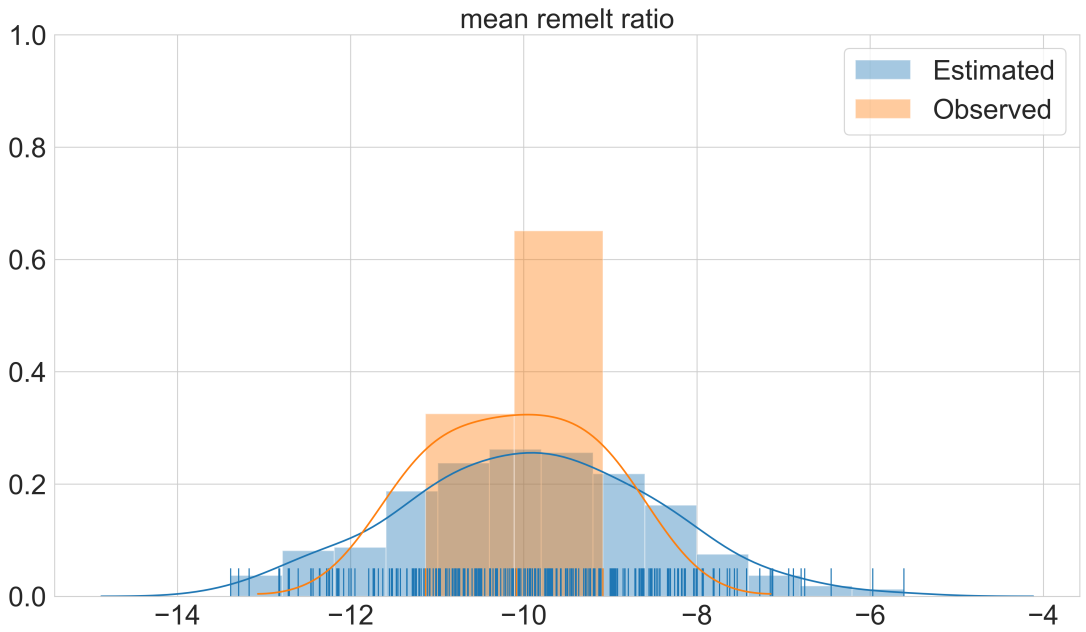
Top of Weld Bead (Parallel to Build)

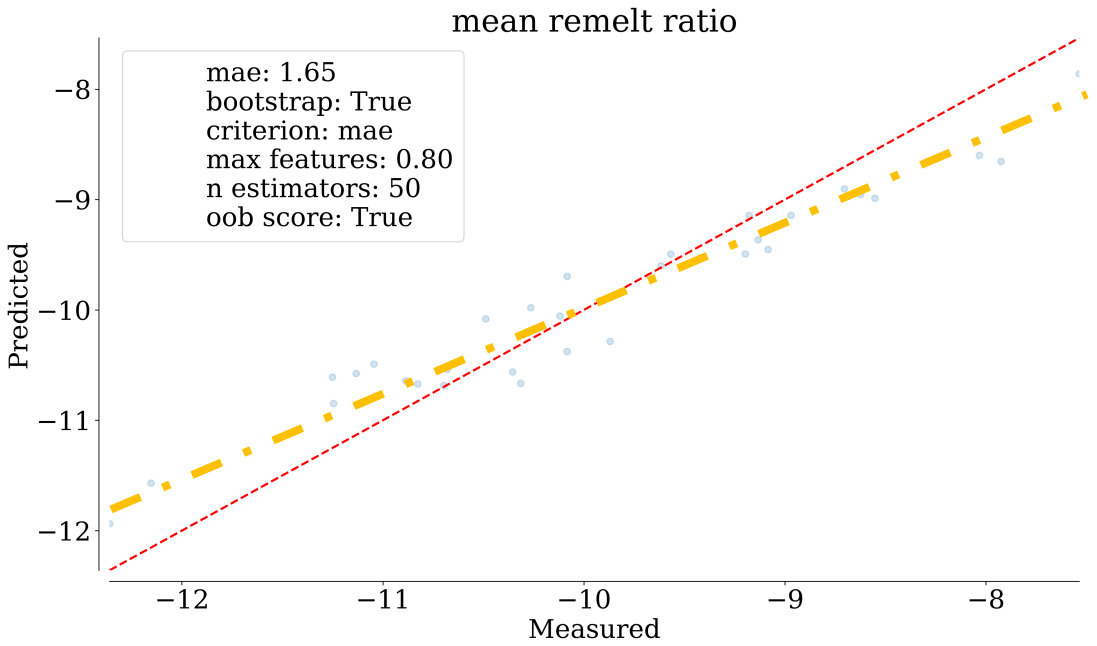




Priority: low

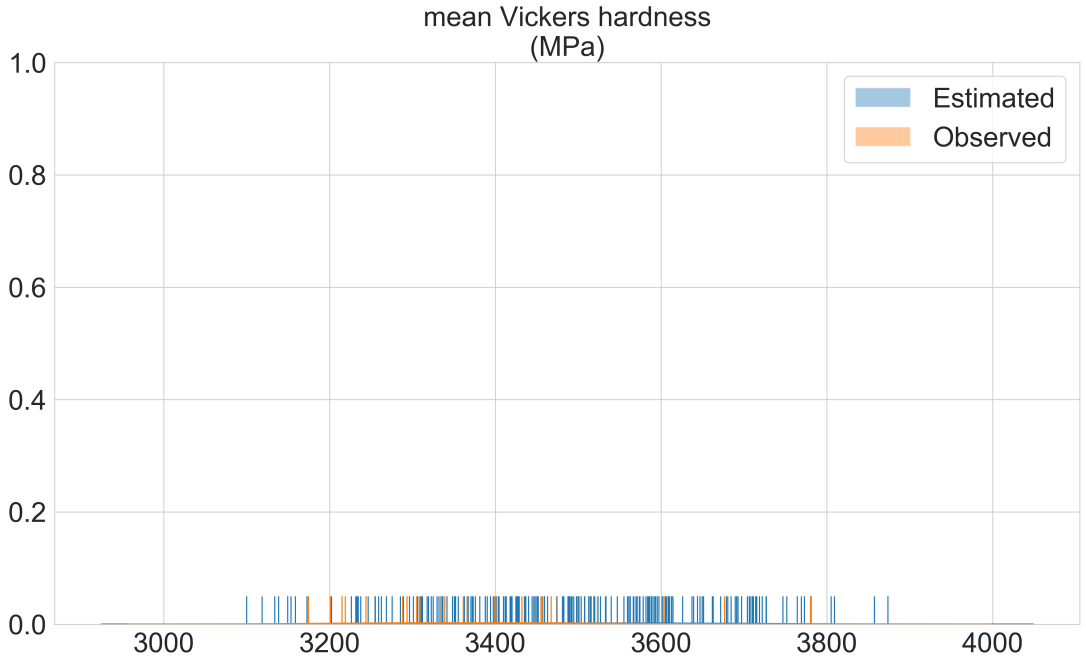
Remelt Ratio

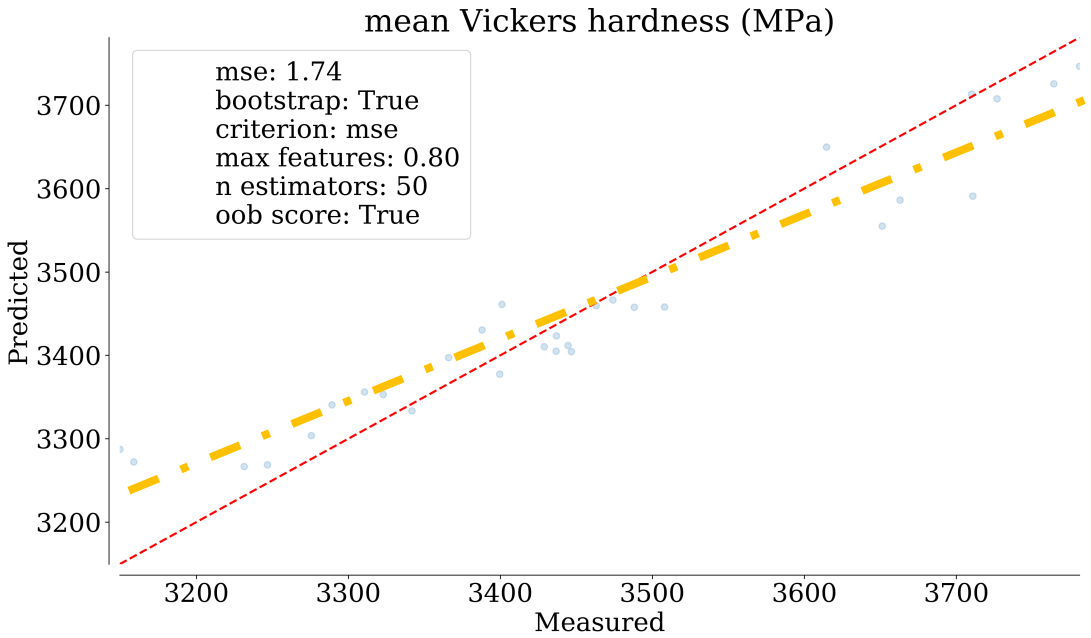




Priority: medium

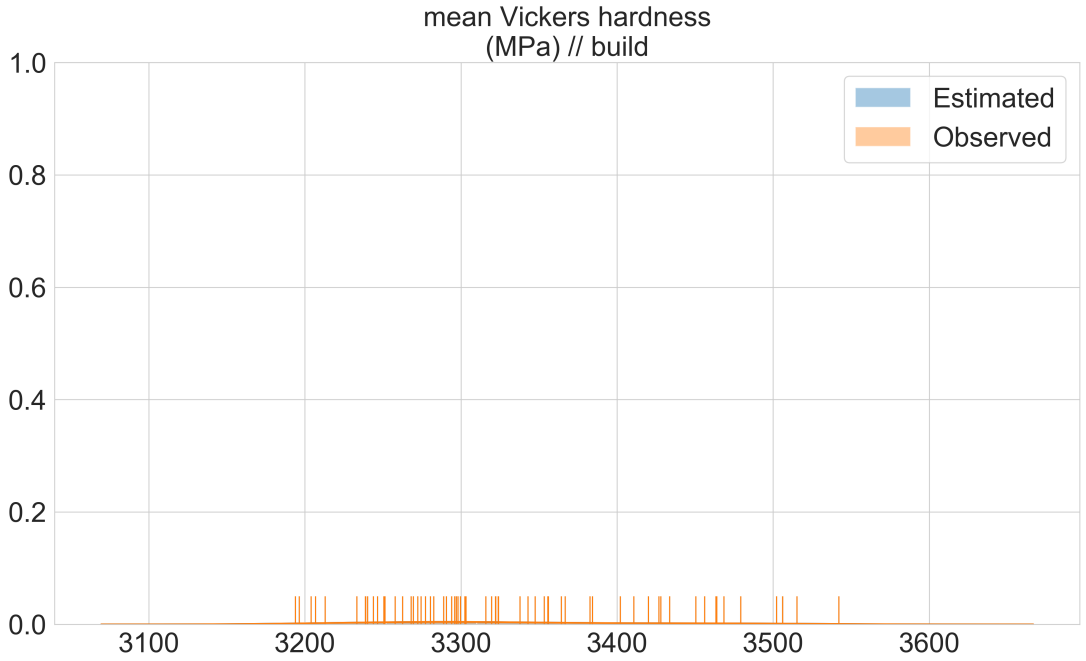
Vickers Hardness

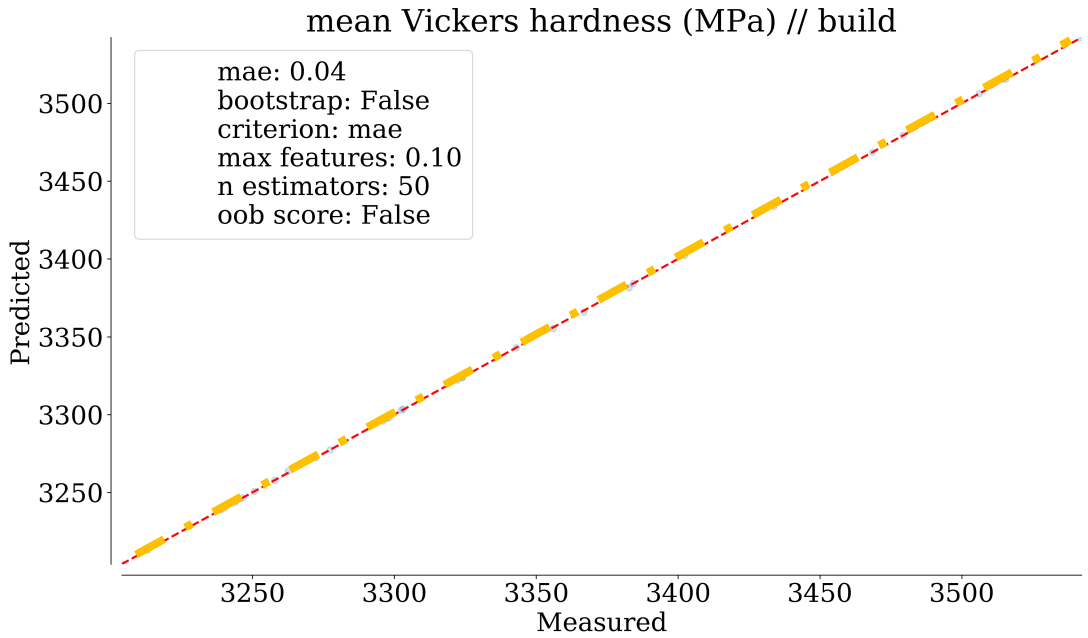




Priority: medium

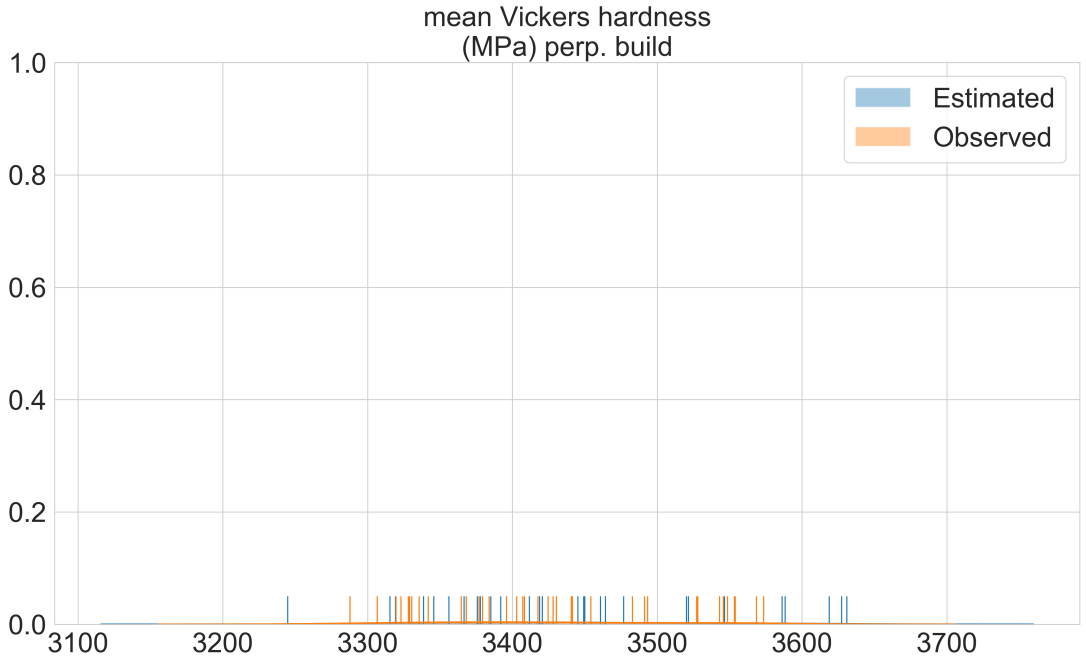
Vickers Hardness (Parallel to Build)

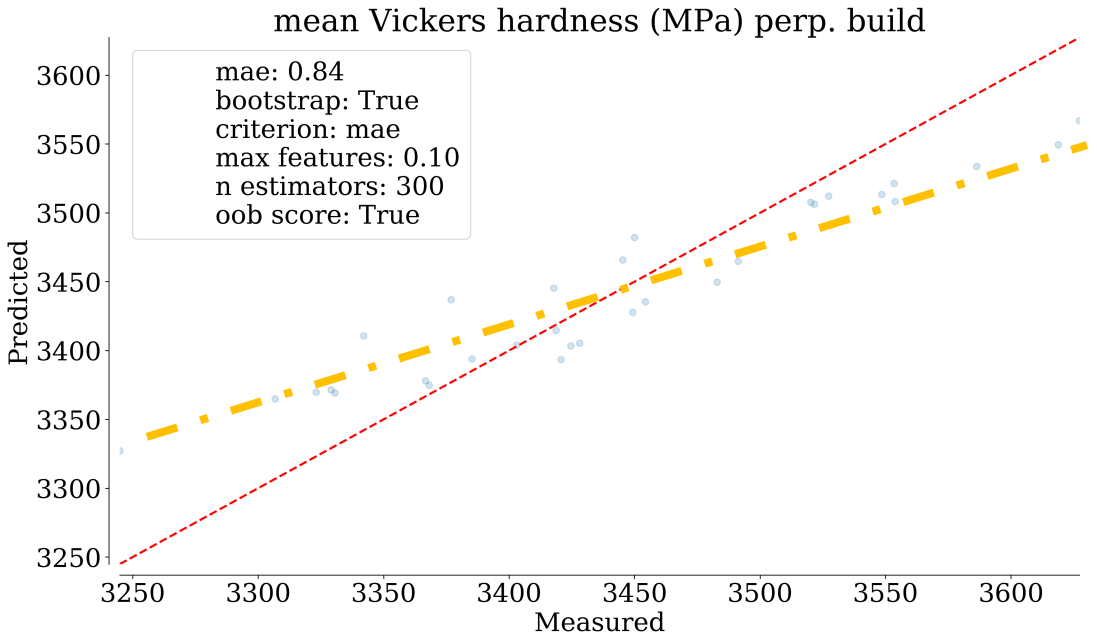




Priority: low

Vickers Hardness (Perp. to Build)



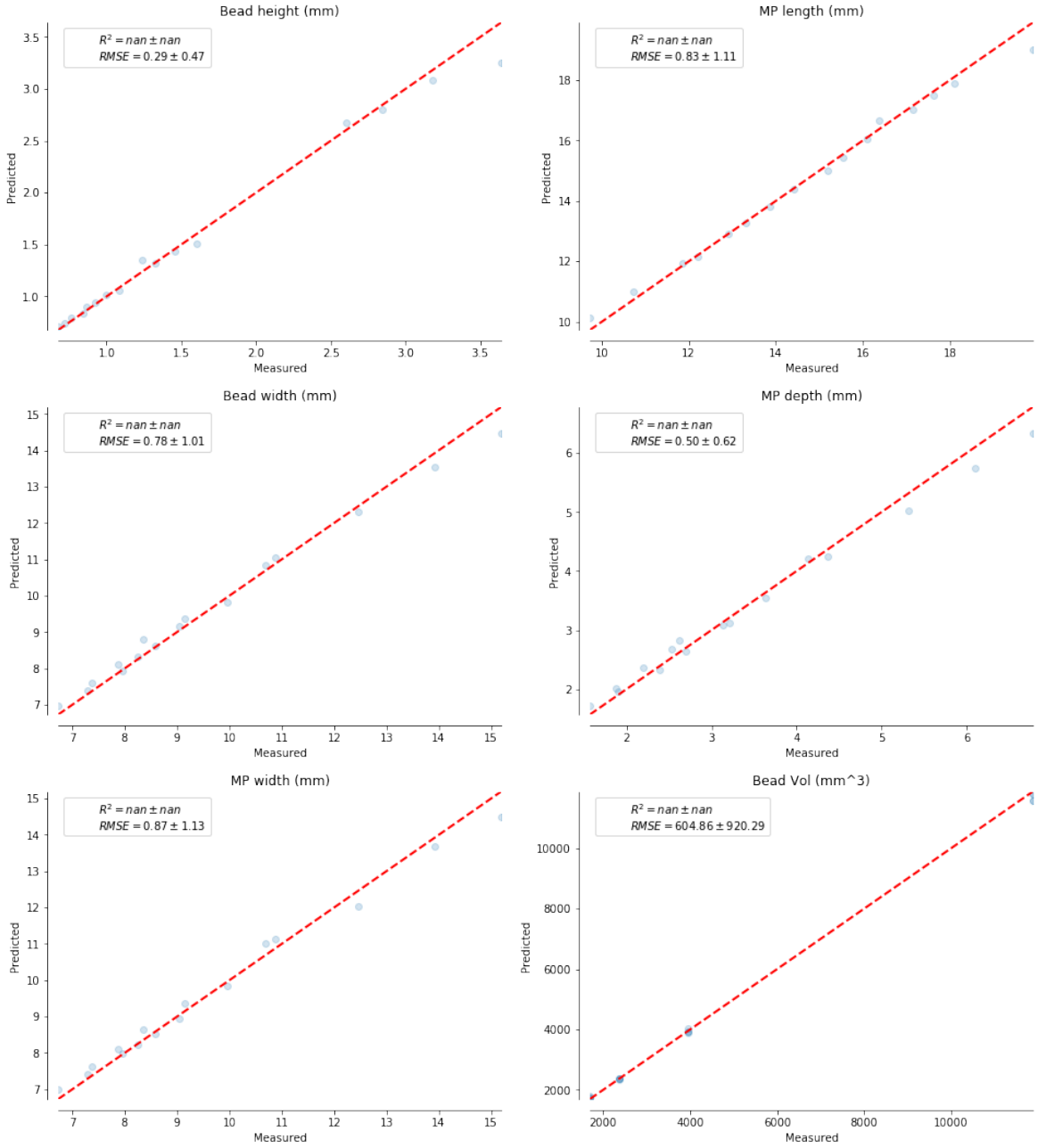


Priority: high

Estimating simulation results

Excellent predicted vs. actual agreement suggests these 16 simulation results are sufficient to capture the functional relationship between simulated build conditions and simulated results.

What we need from the simulations are those properties that correlate to *in situ* measurements (leading pyrometer temperature, trailing pyrometer temperature, melt pool temperature) and those properties that cannot be measured experimentally, e.g. subsurface temperature profile.



ML Model Development Conclusions

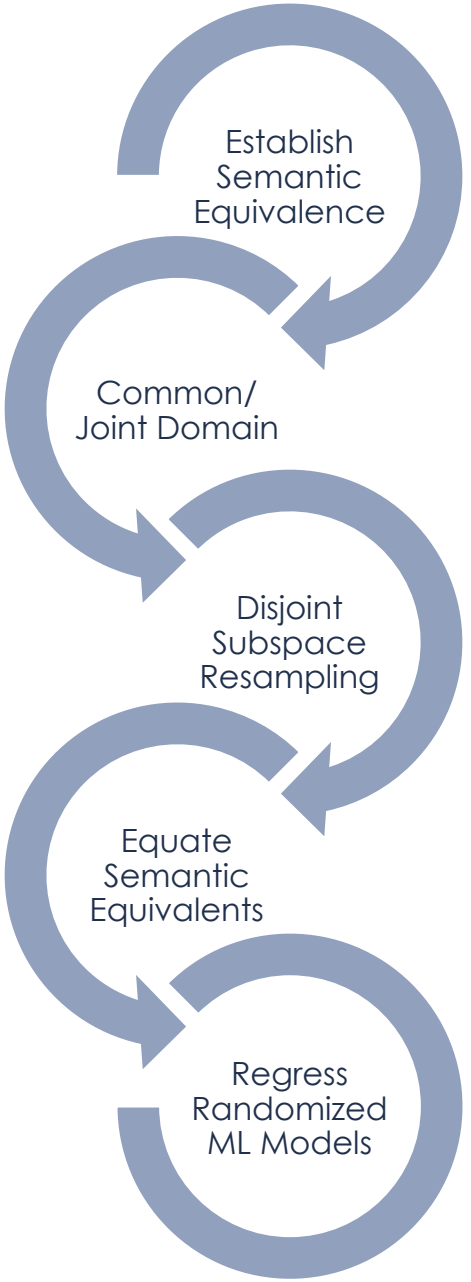
- As expected, RFN-imputed values lower R^2.
  - Markedly better model performance than other imputation methods.
- Feature data priority summary:
  - High: bead height, fusion zone area, fusion zone depth, fusion zone depth at 75% bead width, weld bead left edge at half bead height, Vickers hardness (perpendicular to build)
  - Medium: fusion zone depth at 50% bead width, fusion zone width, remelt ratio, Vickers hardness,
  - Low: top of weld bead, Vickers hardness (parallel to build)
- Simulation model results
  - Properties matching *in situ* experimental measurements
  - Properties not experimentally measureable, but critical to AM performance.

Simulation/Experiment Regression

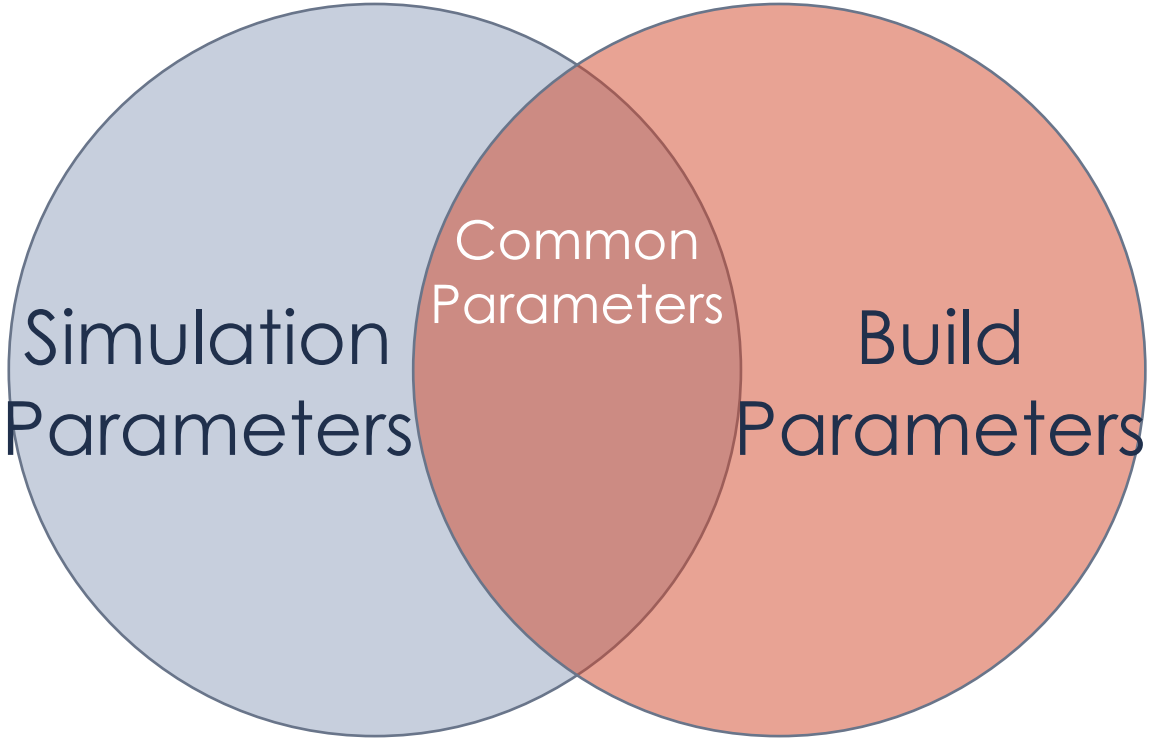
Two ML models, one trained on the experimental data and another on the simulation data, explore the common (joint) domain in the solution space. A regression maps between these solutions.

Regression process summary

- Establish semantic equivalency between experiment and simulation variables.
- Identify common domains across all equivalent variables.
- Sample each variable uniformly across the established domain.
- Ensure equivalent variables have matched values.
- Use models trained in the previous section to predict responses shared by both experimental and simulation ML models.



Binary Semantic Maps set nominal ontological equivalence (intersection)



Experimental	Simulation
Weld Main Stage Data: Laser Power (W)	Laser power (W)
Summary: Measured Weld Time	Layer Time (s)
Weld Main Stage Data: Travel Speed (mm/s)	Travel speed (mm/sec)

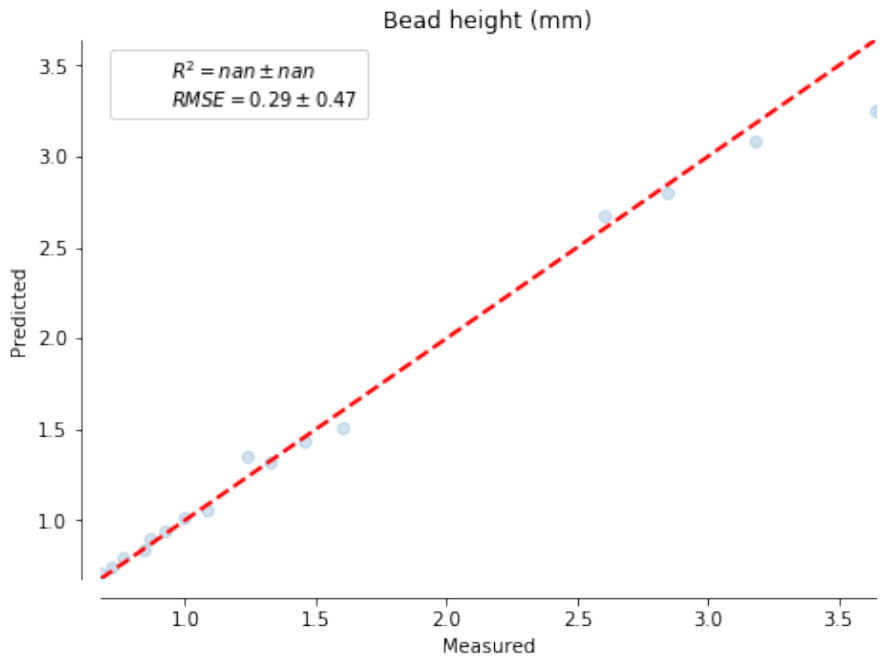
Random sampling explores unpaired model degrees of freedom (difference)

	Weld Main Stage Data: Laser Power (W)	Weld End Stage Data: Wire Power (kW)	Post-Weld Data: Laser Off Delay (s)	Weld Main Stage Data: Wire Power (kW)	Weld End Stage Data: Laser Power (W)	Post-Weld Data: Pull Out Dist (mm)	Weld Fill Stage Data: Heat Wirefeed Speed (mm/s)	Summary: Measured Weld Time	Weld Ignition Stage Data: Laser Pre-Time (s)	Weld Main Stage Data: Heat Wirefeed Speed (mm/s)	Summary: Total Length	Weld Ignition Stage Data: Laser Power (W)	Weld Fill Stage Data: Laser Power (W)	Summary: Pre-Run Temp (C)	Summary: Starting O2 ppm	Pre-Weld Data: PreTime Recorded (s)	Post-Weld Data: Roll Back Time (s)	Weld Fill Stage Data: Fill Time (s)	Weld Main Stage Data: Weave Width (mm)	Weld Main Stage Data: Travel Speed (mm/s)	Weld Fill Stage Data: Wire Power (kW)	Weld Ignition Stage Data: Wire Power (kW)	Summary: Measured Pre-Time	Weld End Stage Data: Heat Wirefeed Speed (mm/s)	Weld Ignition Stage Data: Laser Start Move Delay (s)	Summary: Ending O2 ppm
count	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000
mean	4,518.06	0.15	0.18	0.60	4,915.32	7.00	34.46	86.11	0.35	70.30	125.69	4,727.16	4,908.02	27.25	25.78	54.69	0.01	0.46	0.05	7.90	0.15	0.15	54.62	35.09	0.20	63.33
std	873.02	0.09	0.04	0.34	633.42	0.58	14.32	36.96	0.09	33.96	71.91	731.81	627.19	18.96	32.22	31.36	0.01	0.20	0.00	3.50	0.09	0.09	31.49	14.15	0.00	36.32
min	**3,002.08**	**0.00**	**0.10**	**0.00**	**3,801.94**	**6.00**	**10.04**	**21.51**	**0.20**	**10.17**	**0.10**	**3,500.95**	**3,803.00**	**(4.18)**	**(28.91)**	**(0.95)**	**0.00**	**0.10**	**0.05**	**2.02**	**0.00**	**0.00**	**(0.82)**	**10.16**	**0.20**	**0.06**
25%	3,756.93	0.08	0.14	0.30	4,368.42	6.50	21.34	54.92	0.27	40.79	62.63	4,112.81	4,382.49	9.98	(1.84)	26.97	0.01	0.28	0.05	4.84	0.08	0.07	27.35	23.40	0.20	31.41
50%	4,517.63	0.16	0.18	0.60	4,915.88	6.98	34.19	86.80	0.35	69.11	125.88	4,681.67	4,910.97	27.00	24.62	55.18	0.01	0.46	0.05	7.80	0.15	0.14	55.31	34.69	0.20	63.73
75%	5,269.73	0.23	0.22	0.88	5,465.01	7.50	46.89	117.13	0.43	97.98	189.30	5,380.11	5,444.06	43.64	54.78	81.99	0.02	0.64	0.05	10.92	0.23	0.22	83.11	47.92	0.20	94.85
max	**5,999.76**	**0.30**	**0.25**	**1.20**	**5,997.16**	**8.00**	**59.18**	**149.84**	**0.50**	**129.94**	**248.38**	**5,984.25**	**5,996.39**	**59.98**	**81.97**	**107.36**	**0.02**	**0.80**	**0.05**	**14.00**	**0.30**	**0.30**	**107.43**	**59.25**	**0.20**	**126.00**

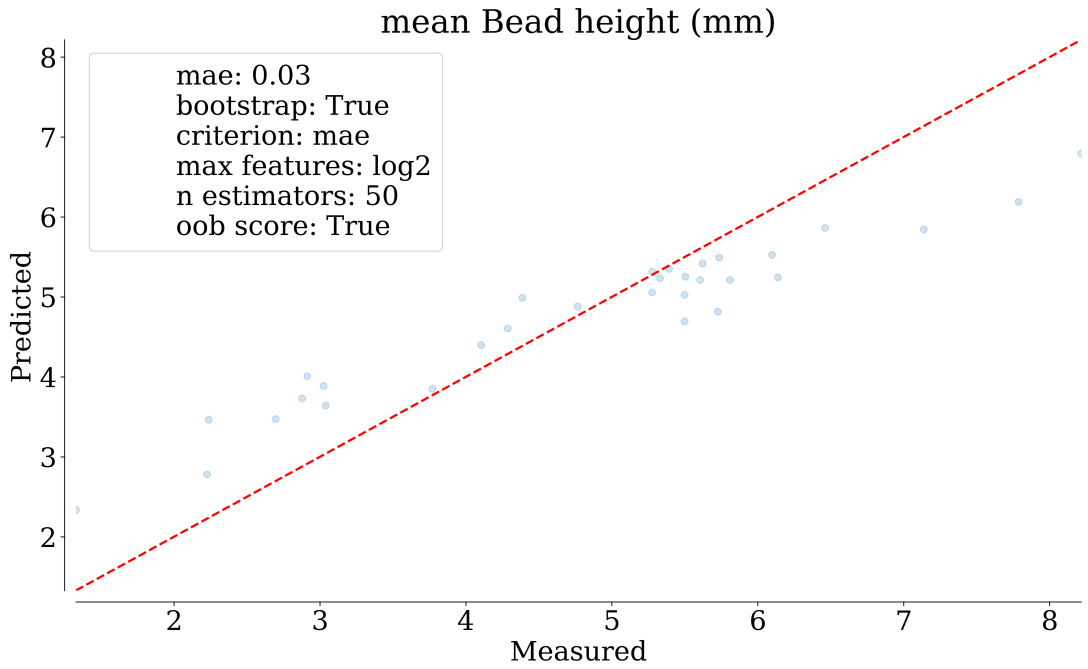
adapt

Models predict **Bead Height** from joint subspace, sample disjoint

Simulation ML Model

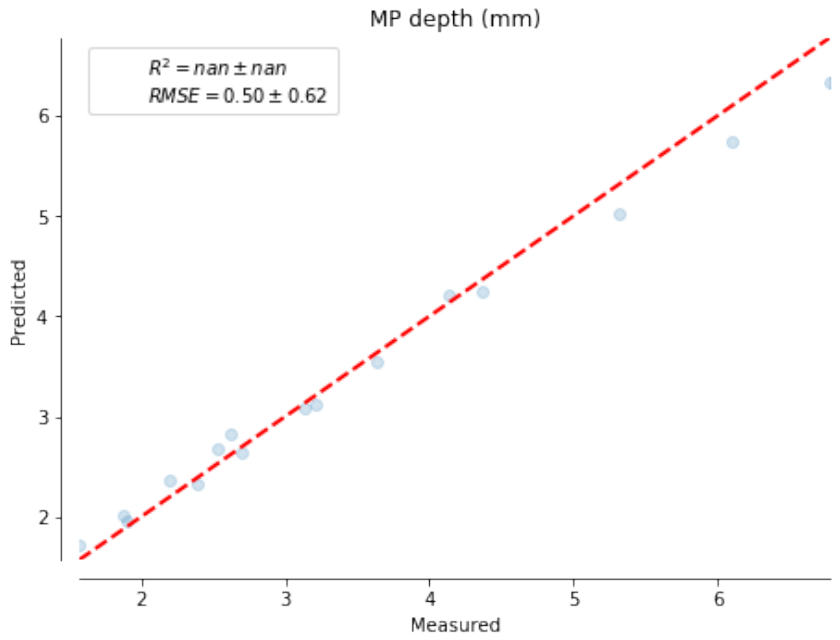


Experimental ML Model

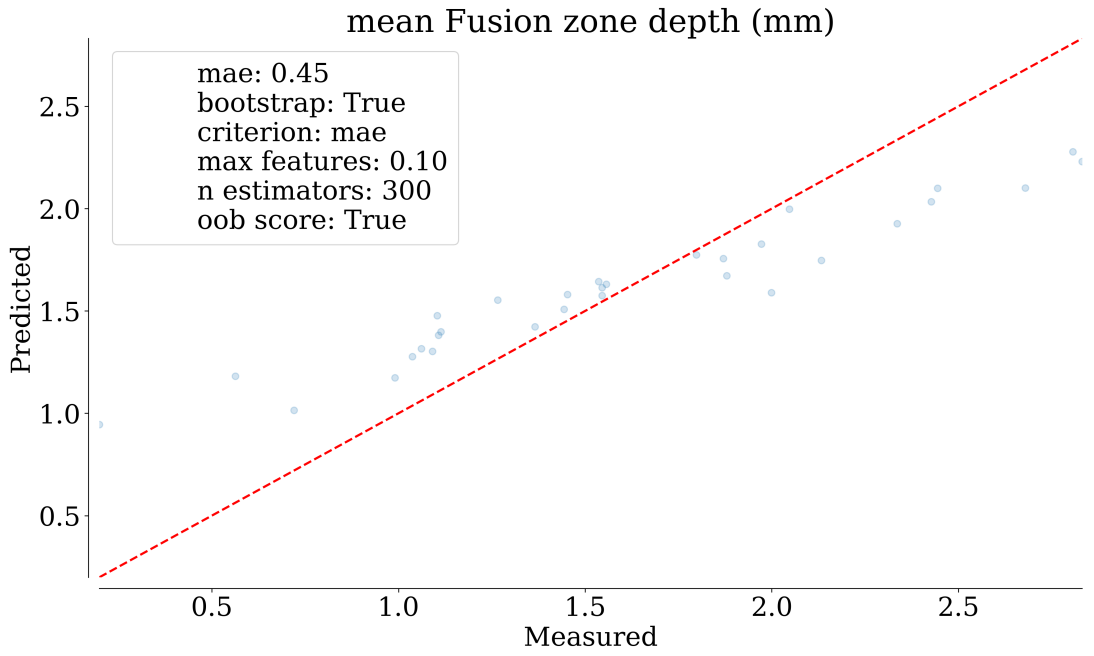


Similarly, models predict **Fusion Zone/Melt Pool Depth**

Simulation ML Model



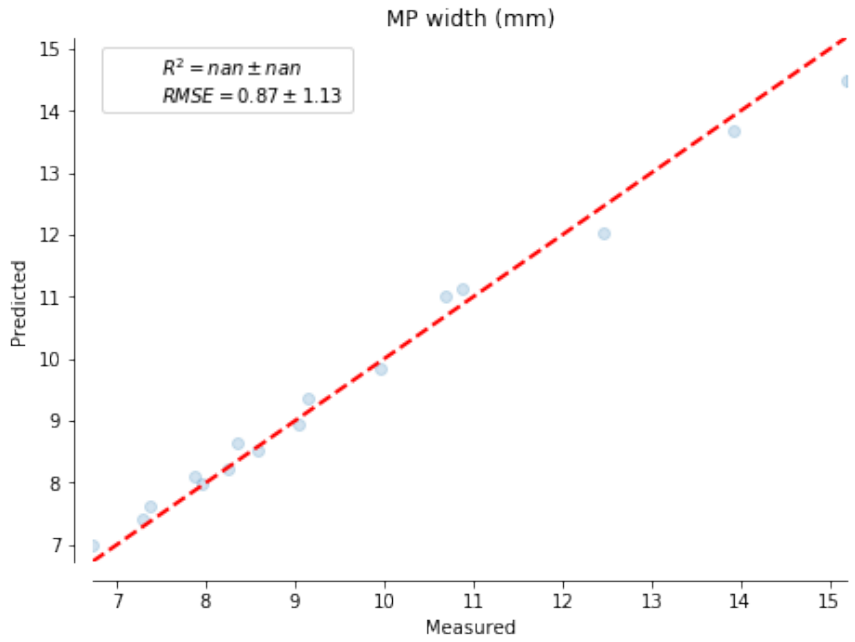
Experimental ML Model



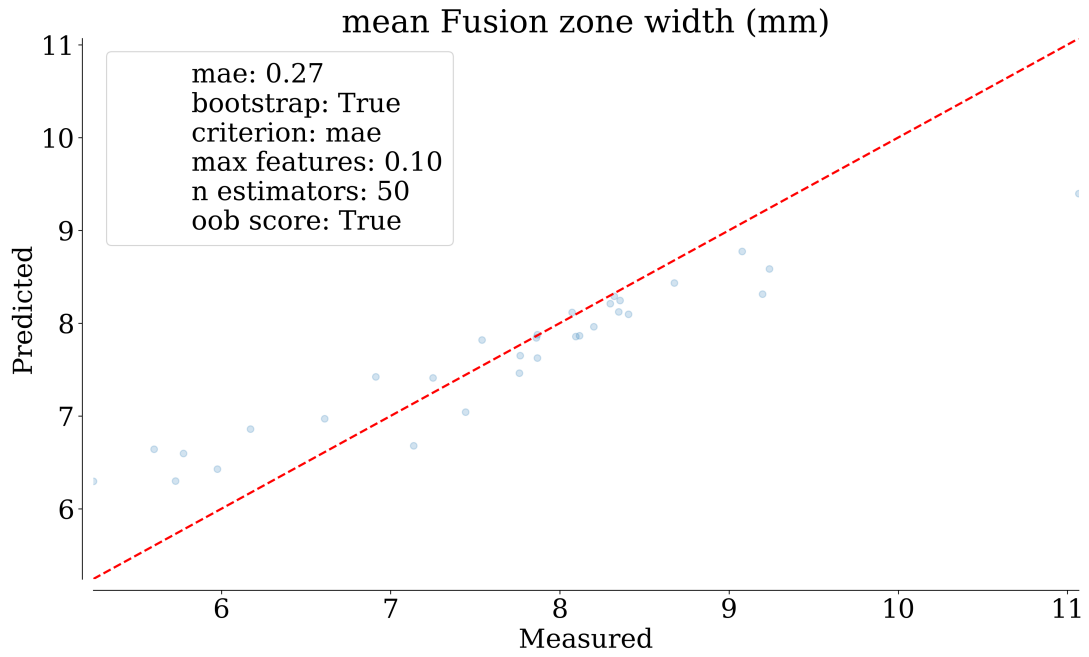
adapt

Similarly, models predict **Fusion Zone/Bead Width**

Simulation ML Model

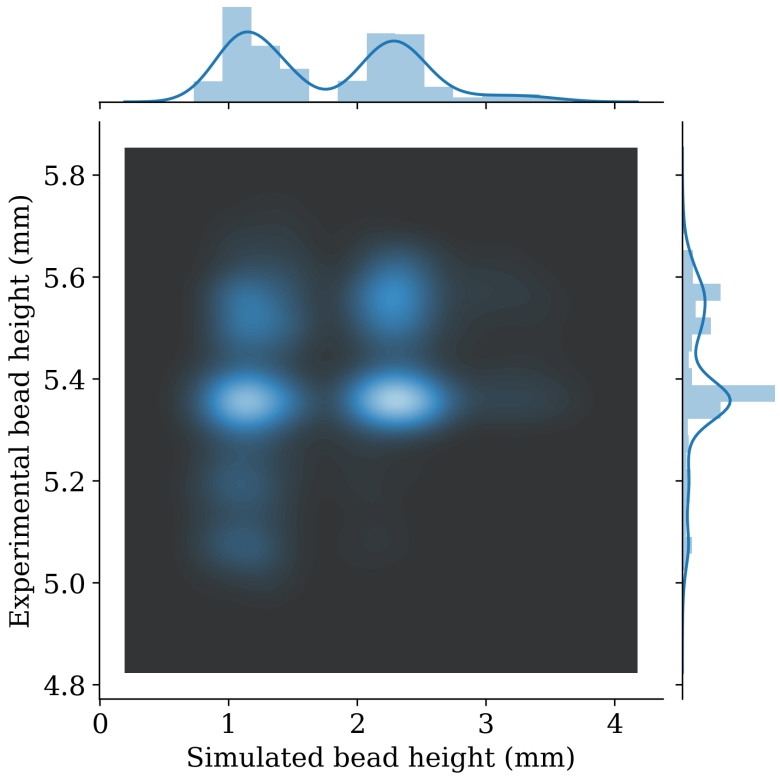


Experimental ML Model



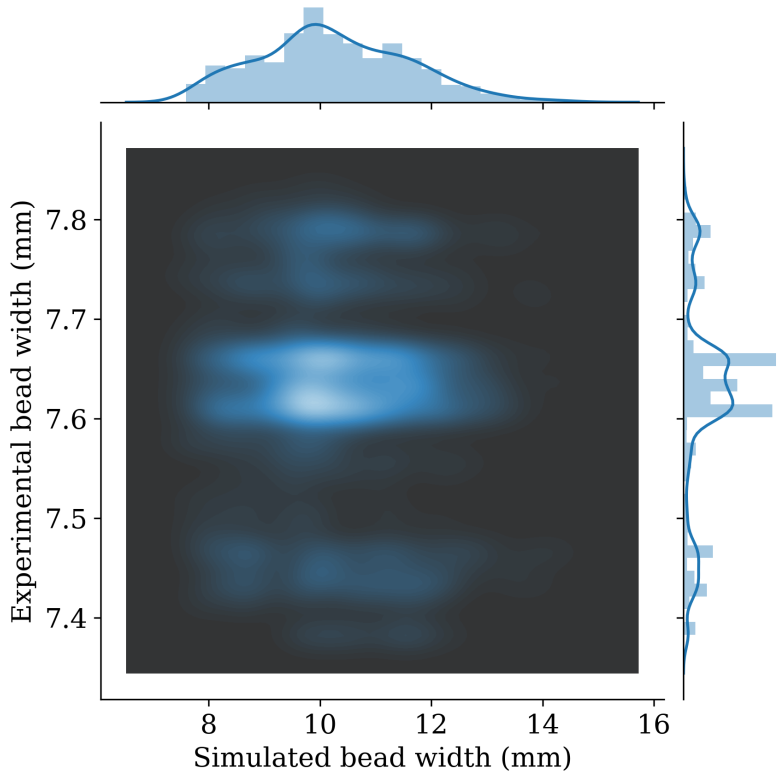
adapt

Uncontrolled variables introduce significant variability



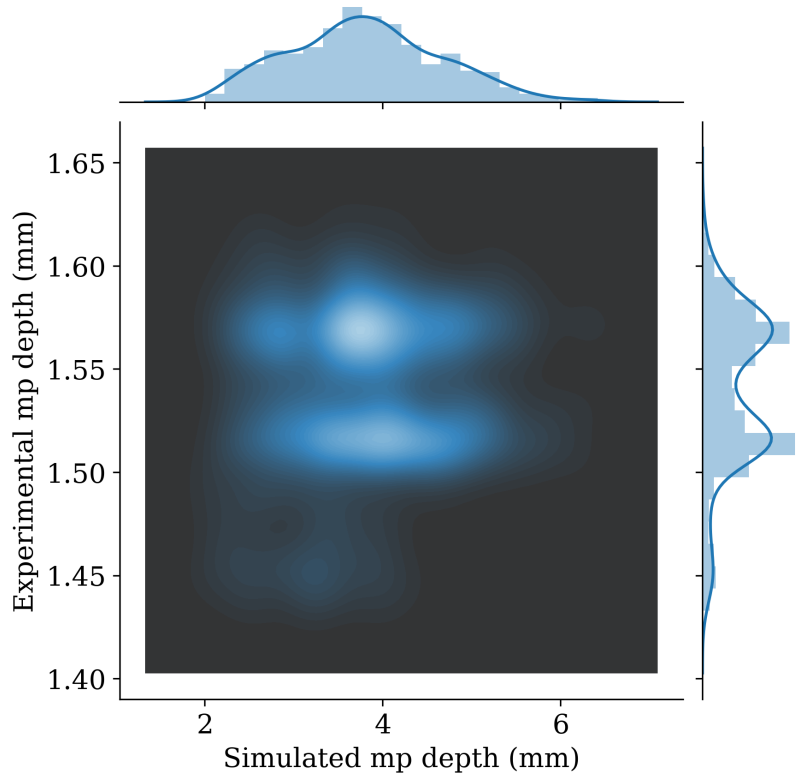
Pearson R: -0.09

R^2: -18



Pearson R: -0.07

R^2: -4.5

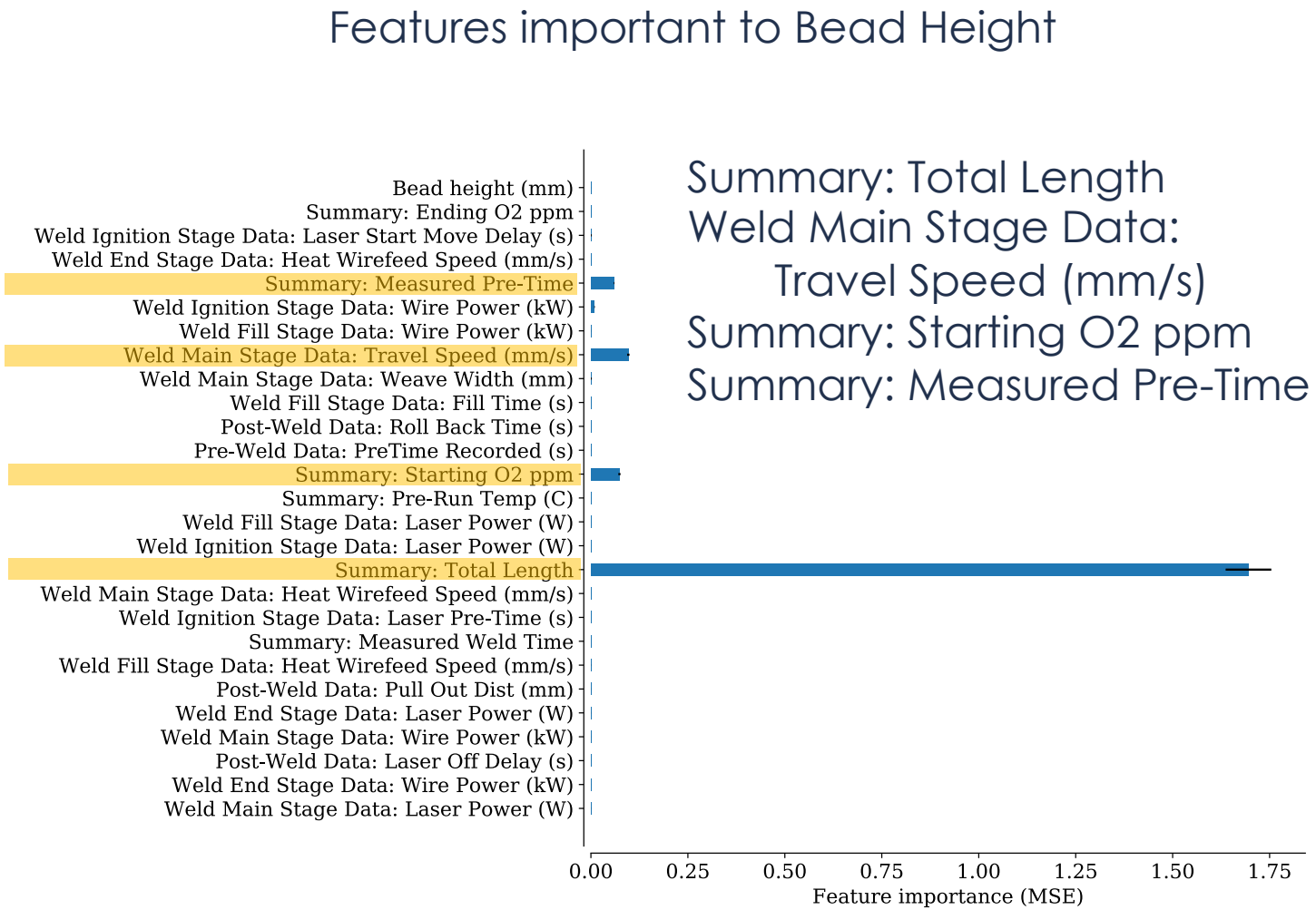


Pearson R: 0.03

R^2: -6.9

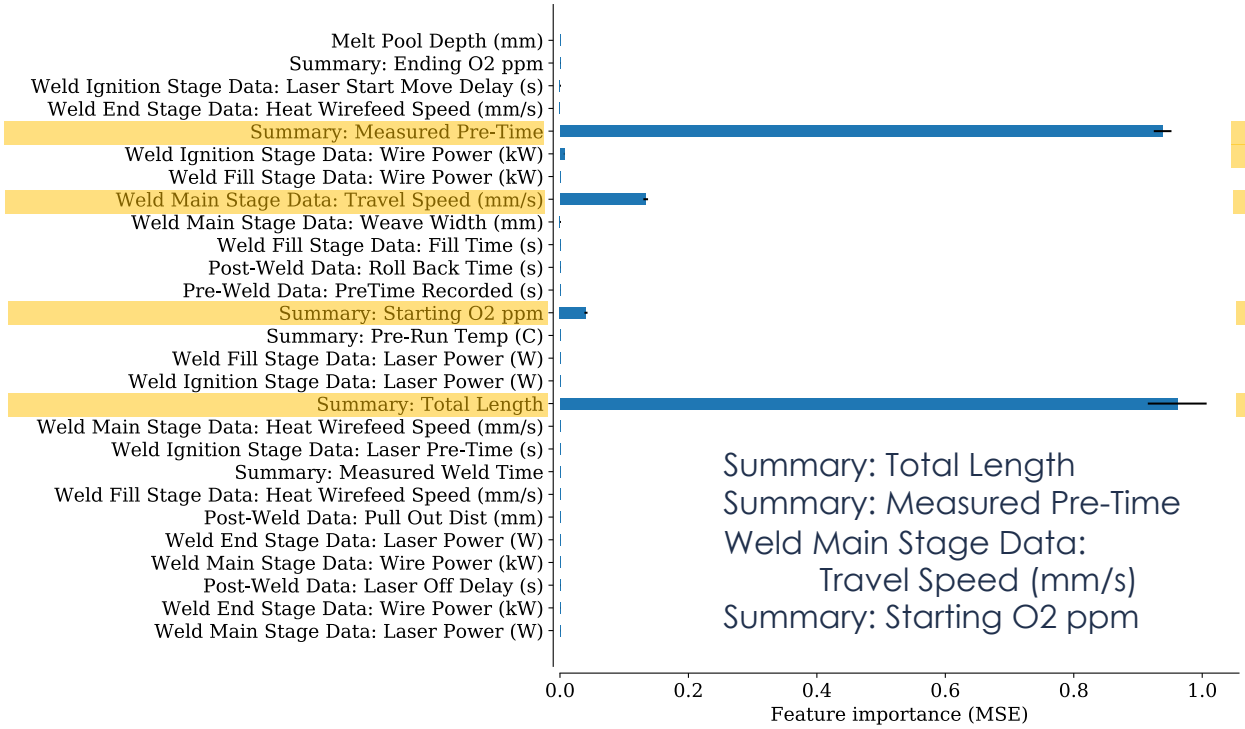
Feature importance determination

- **Establish a baseline**: The optimized model is scored (MAE, MSE, R^2) using the original data.
- **For each column, permute and retrain**: The observations of each variable (column) are permuted, the model retrained and rescored.
- **Evaluate each permutation**: If after permutating a column, the model score remains unchanged, that variable has little importance. The more the score drops, the more important the variable.



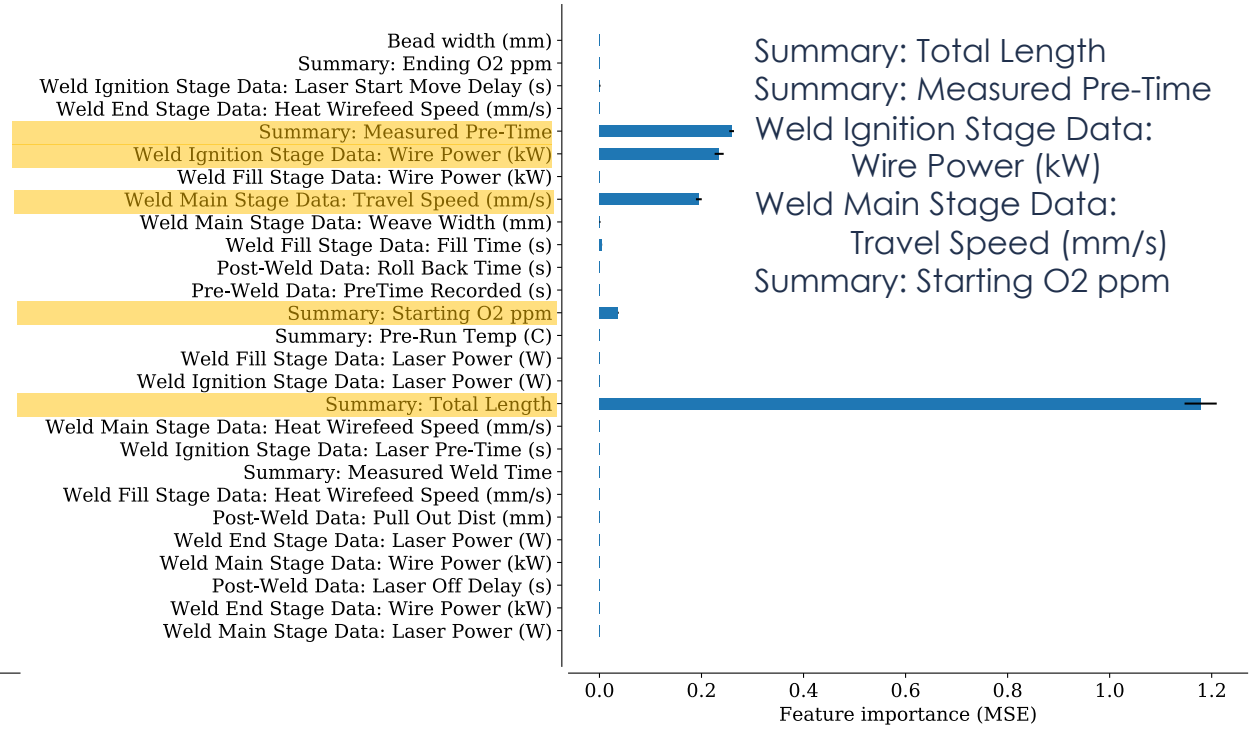
Importance of continuous variables provide direction

Melt Pool Depth



Summary: Total Length
Summary: Measured Pre-Time
Weld Main Stage Data:
Travel Speed (mm/s)
Summary: Starting O2 ppm

Bead Width



Summary: Total Length
Summary: Measured Pre-Time
Weld Ignition Stage Data:
Wire Power (kW)
Weld Main Stage Data:
Travel Speed (mm/s)
Summary: Starting O2 ppm

Significant Developments

- User-definable bijective (one-to-one) semantic mapping between experimental and simulation names.
  - Extend to surjective (many-to-one) mappings for more complex ontologies.
- Establish common domains in equivalent columns.
  - Extend beyond nominal equivalence.
- Sample equivalent point in common subspace; random, uniform sampling over disjoint spaces.
  - Extend to categorical variables.

Conclusions/Discussion Points

- No statistically significant trend exists between simulated and experimental measurements.
  - Uncontrolled variables indicate need to simulate different stages of the build process.
  - Random sampling of the disjoint subspace currently limited to uniform distribution of continuous variables. Extension to discrete variables may be necessary.
  - Semantic maps currently cannot express complex ontologies.
- Reduced possibility of a hidden correlation to underlying build conditions by regressing $y_{exp} \sim [X_{exp}, y_{sim}]$.