1.7 Uncertainty model

This section corresponds to box labelled "Uncertainty model" in Figure 1.4. It is envisioned with the following question in mind:

> Given a new configuration with input parameters $\mathbf{X} = (X_1, X_2, \ldots, X_m)$ and prediction $\mathbf{Y} = (Y_1, Y_2, \ldots, Y_q)$, where $\mathbf{Y} = F(\mathbf{X})$, what is the expected uncertainty (error) associated to the prediction $\mathbf{Y}$?

To the aim of answering this question, the main tasks are:

- To build an **uncertainty model**.
- To **validate** such model.

The expected uncertainty translates into (for instance) $q = 6$ confidence intervals $(CI_1, CI_2, \ldots, CI_6)$, such that the k-th predicted reserve factor Y_k has a confidence interval $CI_k = [Y_k^{min}, Y_k^{max}]$. Other types of uncertainty quantifications are possible (standard deviation, credible interval, etc).

Importantly, there are various sources of uncertainty that need to be combined:

1. **Global error distribution** (cfr. section 1.4). This is, the distribution of error in the test set, $P(E)$.
2. **Local-output error distribution** (cfr. section 1.6). This is, the distribution of the error in the test set, conditioned to a specific region of the prediction, $P(E|\mathbf{Y})$.
3. **Local-input error distribution** (cfr. section 1.5). This is, the distribution of the error in the test set, conditioned to a specific region of the input space (error bias characterization), $P(E|\mathbf{X})$.

Our uncertainty model will combine contributing factors (1), (2) and (3) into an effective uncertainty.

Each part (1-3) of the process will need to be **validated**. To this aim, we split the test set into a calibration set $\mathcal{E}^{\text{cal}}$ and a test set[6] $\mathcal{E}^{\text{test}}$ (80/20 random split), estimate the uncertainty

[6]Not to be confused with $\mathcal{S}^{\text{test}}$. Here we denote by "test set" the empirical error distribution sampled from

model on the calibration set and evaluate it by computing its **coverage** (*i.e.* the percentage of configurations whose true output belongs to the confidence interval region) in the test set.

1.7.1 Global uncertainty model based on $P(E)$

After a sanity check that the dataset under analysis (here, the full calibration set) fulfils the minimum size requirements (threshold is set at 300 points for each output variable $\mathbf{Y}_i$) we proceed with:

A. Global Uncertainty Model (GUM) computation

The interval $[P2.5, P97.5]$ (or in general, the interval $[a, b]$ if other percentiles are chosen) of $P(E)$ is estimated in the calibration set, where the lower bound P2.5 and the upper bound P97.5 are not raw estimates but the mean of a bootstrap analysis of the lower and upper percentiles respectively. The bootstrapping procedure is implemented as described in algorithm 3. This procedure is repeated for all m output variables. Illustration of the GUM is provided in Table 1.20.

Algorithm 3: Bootstrapped uncertainty CI

Data: Calibration set $\mathcal{E}^{\text{cal}}$
Result: Bootstrapped 95% CI [P2.5,P97.5]
Initialize empty arrays $P_{2.5}$ and $P_{97.5}$;
for $i = 1$ **to** 100 **do**
  Build calibration set i by randomly sampling (with replacement) a total number of points equal to the size of the calibration set;
  Compute lower ($P_{2.5}^{(i)}$) and upper ($P_{97.5}^{(i)}$) percentiles;
  Add $P_{2.5}^{(i)}$ to $P_{2.5}$;
  Add $P_{97.5}^{(i)}$ to $P_{97.5}$;
end
$P_{2.5\_mean} \leftarrow mean(P_{2.5})$;
$P_{97.5\_mean} \leftarrow mean(P_{97.5})$;
return $P_{2.5\_mean}$ and $P_{97.5\_mean}$;

B. Global Uncertainty Model (GUM) evaluation

The (bootstrapped) error coverage is estimated in the test set. This is done by computing the

the points belonging to $\mathcal{S}^{\text{test}}$.

Table 1.20: Global Uncertainty Model: Confidence Interval (mean and extremes) for output variables "RF Forced Crippling" (left) and "RF Column Buckling" (right). Header showing $[y_{min}, y_{max}]$ in the calibration set.

	[0.75, 4.99]
Mean	0.001859
CI	[-0.5366, 0.1018]

	[0.9069, 4.99]
Mean	0.002292
CI	[-0.5339, 0.1076]

percentage of points in the test set whose error lies inside the chosen uncertainty model. The bootstrap procedure is carried out as described in algorithm 4.

Algorithm 4: Bootstrapped coverage

Data: Uncertainty model (CI), test set $\mathcal{E}^{\text{test}}$
Result: Mean and confidence interval of error coverage
Initialize empty array *error_coverages*;
for $i = 1$ **to** 100 **do**
  Build resampled test set i by randomly sampling (with replacement) a total number of points equal to the size of the test set;
  Compute error coverage in resampled test set i;
  Add error coverage to *error_coverages*;
end
Sort *error_coverages* in increasing order;
Calculate mean, 2.5th percentile, and 97.5th percentile of *error_coverages*;
return Mean, 2.5th percentile, and 97.5th percentile;

The output is the bootstrap error coverage and its 95% confidence interval. The criteria for a correctly calibrated error coverage is that it is correctly calibrated if (b-a)% (by default, 95%) lies inside the bootstrapped error coverage confidence interval.

This procedure is repeated for all m output variables. For a graphical interpretation of results, cfr. Figure 1.26. The six GUM's CIs are given in Table 1.21.

Table 1.21: Global Uncertainty Model. The six confidence intervals (bootstrapped) defining the GUM alongside their coverage –in the calibration set– are showed.

Coverage table

	RF Forced Crippling	RF Column Buckling	RF In Plane	RF Net Tension	RF Pure Compression	RF Shear Panel Failure
Coverage	98.89	99.07	98.92	98.9	99.12	99
CI	[98.61, 99.13]	[98.77, 99.31]	[98.57, 99.21]	[98.8, 98.99]	[98.8, 99.37]	[98.8, 99.17]

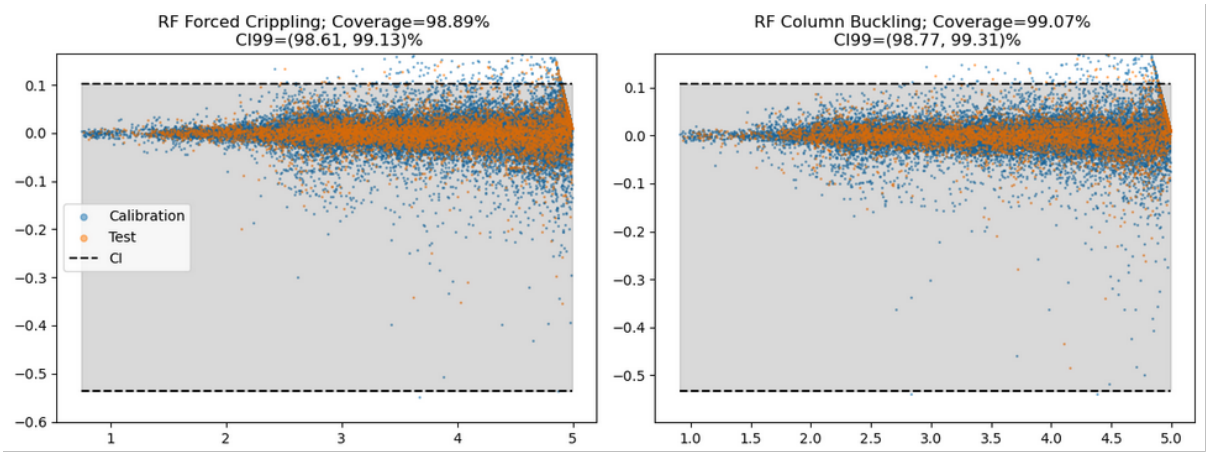


Figure 1.26: Global Uncertainty Model: error coverage. The coverage is higher than 95% for both output variables –for convenience just two output variables are depicted– what means the calibration criteria is successfully met.

1.7.2 Local uncertainty model based on $P(E|\mathbf{Y})$

A. Output variable binning

In order to be able to computationally afford this analysis, the output space $\mathbf{Y}$ must be binned.

In the first place, a scatter plot of the error of each point in the calibration set is plotted for illustration as a function of the true output y_t in Figure 1.27.

The calibration set is then partitioned into a total of equispaced n_{bins} bins according to the range of the true output y_t.

For illustration, a total of n_{bins} violin plots (one per bin, concatenated in a single plot where the "x axis" is the bin number) is plotted in Figure 1.28, where each violin plot describes the error density of the points in the respective bin.

B. Local Output Uncertainty Model (LOUM) computation

First, a sanity check that the number of points in each bin is above a certain threshold (namely, 300 points) assures statistical robustness for this next part.

For those bins for which the minimum-size requirement is not met, the bin is given the Global Uncertainty Model (GUM).

For those bins for which the minimum-size requirement is met, the code computes an uncertainty model in terms of the bootstrap [P2.5, P97.5] (or a general [a,b]) using algorithm 3, and assigns to the bin the resulting local uncertainty model (vid. Table 1.22).

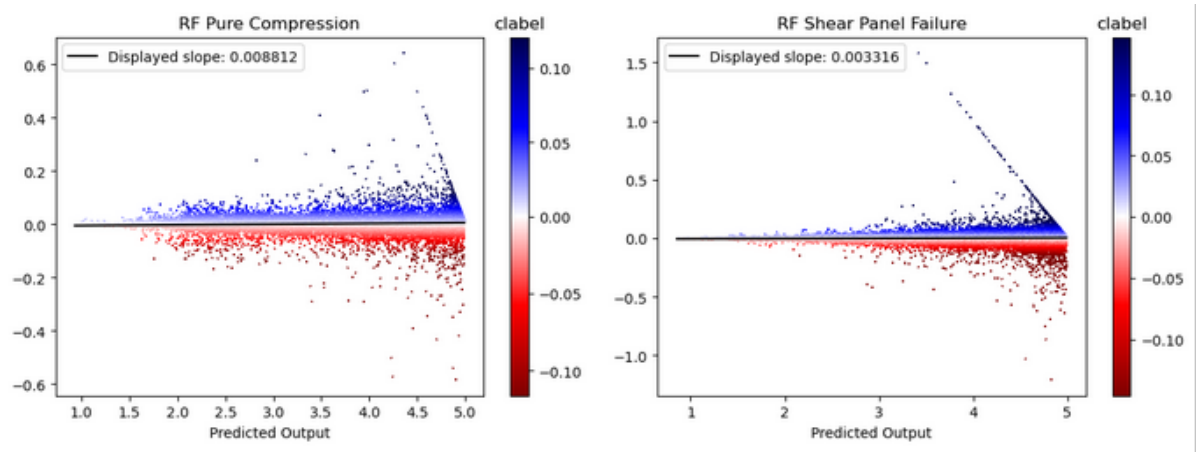


Figure 1.27: Local Uncertainty Model based on $P(E|\mathbf{Y})$: Scatter plot of the residue as a function of the predicted output $\hat{y}$, for output variables "RF Pure Compression" and "RF Shear Panel Failure".

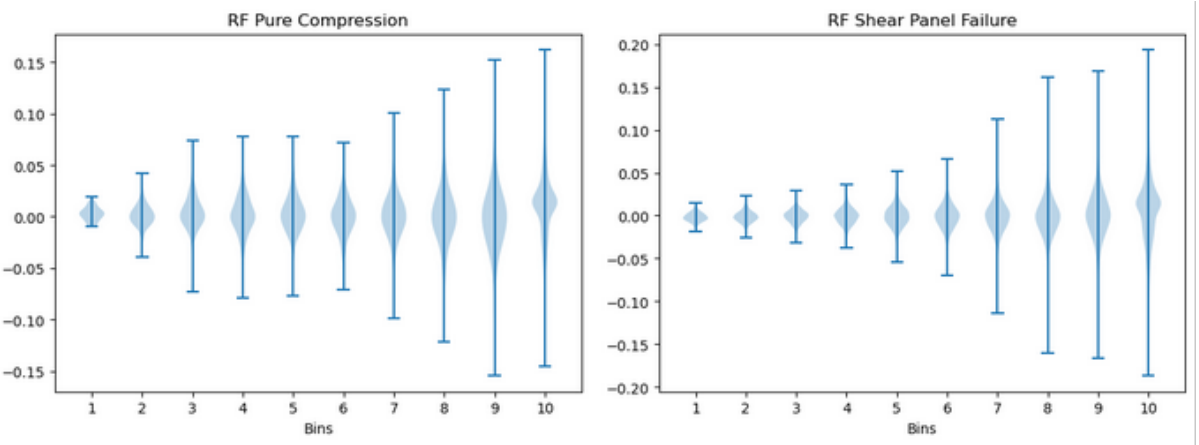


Figure 1.28: Local Uncertainty Model based on $P(E|\mathbf{Y})$: Violin plots of the residue, for binned output variables "RF Pure Compression" and "RF Shear Panel Failure".

This procedure is repeated for all m output variables.

C. Local Output Uncertainty Model (LOUM) evaluation

The bootstrapped error coverage in the test set associated to the LOUM vector is computed with algorithm 4, with the nuance that for each point in the test set, we need to find, based on the point's error, to which bin of the output variable it belongs, and then it is checked whether such error is within the corresponding bootstrapped interval [P2.5, P97.5].

The output is the bootstrap error coverage and its 95% confidence interval. The criteria for a correctly calibrated error coverage is that it is correctly calibrated if (b-a)% (by default, 95%) lies inside the bootstrapped error coverage confidence interval.

Table 1.22: Local Output Uncertainty Model: Bootstrapped CIs for each of the ten bins into which the output variable "RF Forced Crippling" has been binned. The mean of each CI is also given, as well as the bin extrema $[\hat{y}_{min}, \hat{y}_{max}]$ for every bin (in the first row).

Input: RF Forced Crippling; Output: RF Forced Crippling

	[0.75, 1.174]	**[1.174, 1.598]**	**[1.598, 2.022]**	**[2.022, 2.446]**	**[2.446, 2.87]**	**[2.87, 3.294]**	**[3.294, 3.718]**	**[3.718, 4.142]**	**[4.142, 4.566]**	**[4.566, 4.99]**
Mean	-0.000885	-0.0001975	0.0001137	0.001315	0.0004612	-0.0002439	0.0006223	0.001116	0.001307	0.00512
CI	[-0.02095, 0.01634]	[-0.03995, 0.02743]	[-0.06692, 0.02547]	[-0.1572, 0.05091]	[-0.2539, 0.07597]	[-0.1987, 0.0708]	[-0.4765, 0.09805]	[-0.4279, 0.1017]	[-0.321, 0.1269]	[-0.4837, 0.1203]

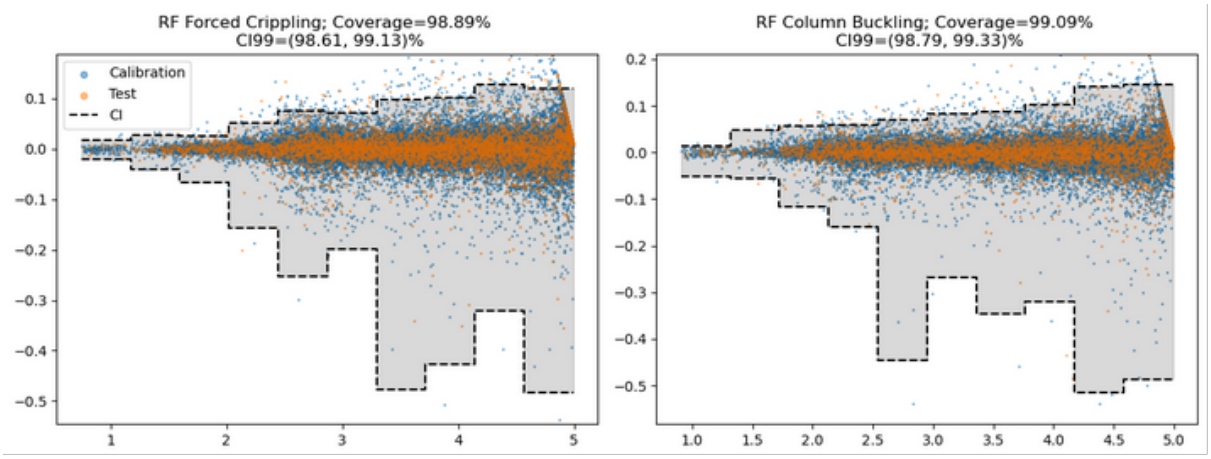


Figure 1.29: LOUM coverage in the validation set. For both output variables coverage is above the minimum 95% threshold.

This procedure is repeated for all m output variables.

A scatter plot of the error of each point in the calibration set is displayed in Figure 1.29, along with the uncertainty model interval (the upper and lower percentiles emphasized in blue and red respectively). This LOUM shows up as a piece-wise constant upper curve and a piece-wise constant lower curve. LOUM coverage for all the six Reserve Factors can be found in Table 1.23.

Table 1.23: LOUM coverage. Coverage in the validation set is displayed in the first row, whereas the coverage bootstrapped CIs are displayed in the second row.

Input: dp; Output: RF Forced Crippling		
	0.000000	**0.900000**
Mean	0.001506	0.002707
CI	[-0.5033, 0.09891]	[-0.5199, 0.1104]

1.7.3 Local uncertainty model based on $P(E|\mathbf{X})$

A. Input variable binnarization

In order to be able to computationally afford LIUM calculation, it is needed to bin the input space in the first place.

For illustration, a scatter plot of the error of each point in the calibration set as a function of x_i is displayed in Figure 1.30 (just two input numerical and two categorical variables have been selected for convenience).

For each continuous input variable x_i the calibration set is partitioned into a total of equispaced n_{bins} bins according to the range of input variable x_i.

For illustration, in Figure 1.31(a) n_{bins} number of violin plots (one per bin, concatenated in a single plot where the "x axis" is the bin number) are plotted, where each violin plot describes the error density of the points in the respective bin.

Binning is exclusive to continuous numerical variables. For illustration, in Figure 1.31(b) and Figure 1.31(c) m_i violin plots (one per each possible value that the corresponding input variable x_i can take) are plotted, where each violin plot describes the error density of the points whose input variable x_i takes the same value.

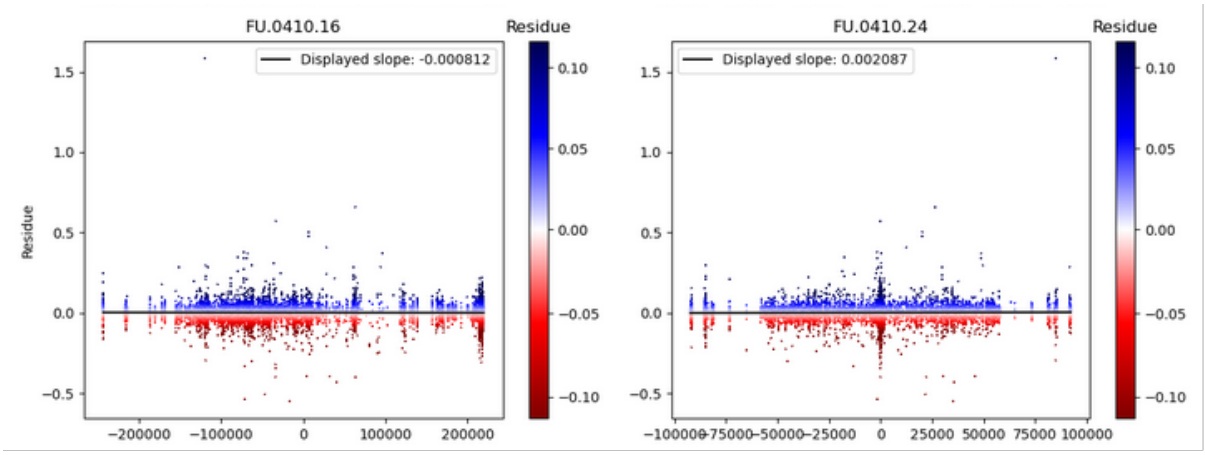
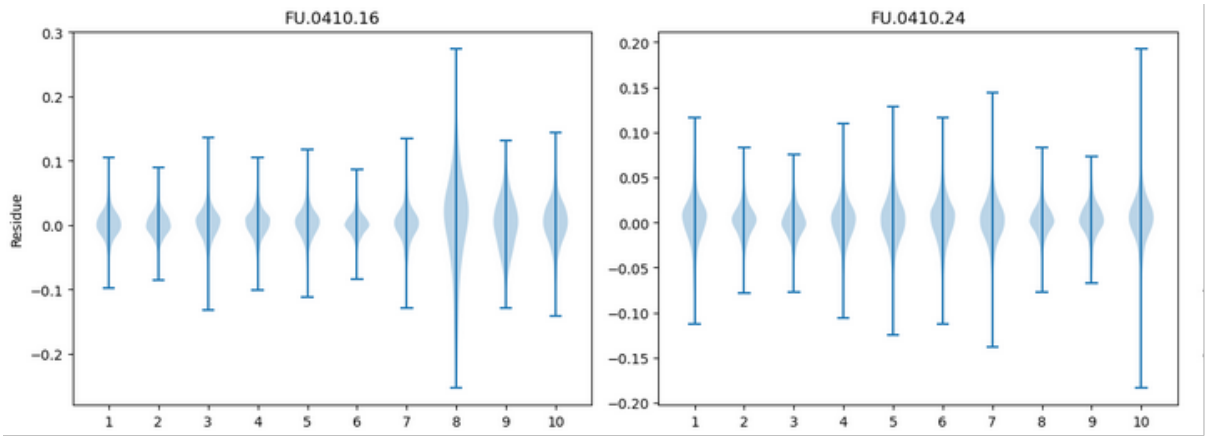


Figure 1.30: LIUM: Scatter plot of the residue of "RF Forced Crippling" as a function of the input x (left: x is "FU.0410.16". Right: x is "FU.0410.24").

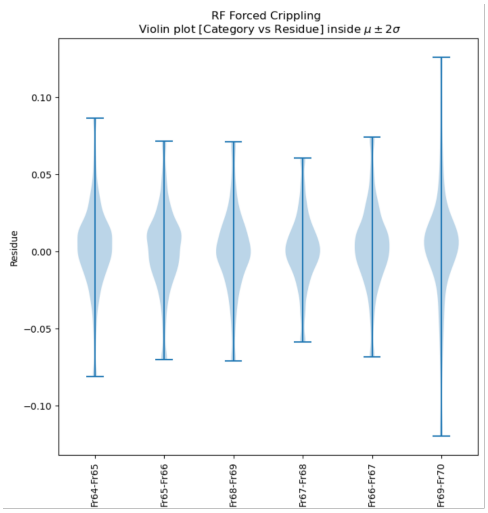
B. Local Input Uncertainty Model (LIUM) computation

For each continuous input variable x_i that has been discretized into n_{bins} bins:

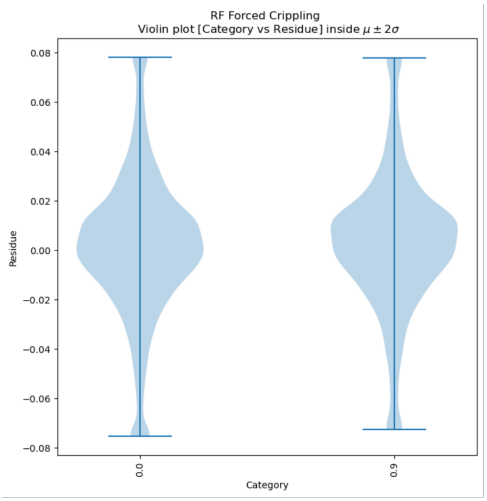
- It is checked whether the number of points inside each bin fulfils the minimum bin size (set at 300 points) requirement.
- For those bins for which the minimum bin size requirement is not met, the bin is given the Global Uncertainty Model (GUM).
- For those bins for which the minimum bin size requirement is met, an uncertainty model is computed in terms of the bootstrapped interval [P2.5, P97.5] (or a general [a,b]) with algorithm 3, and the resulting local uncertainty model is assigned to the bin.



(a) Input variables: "FU.0410.16" (left), "FU.0410.24" (right)



(b) Input variable: "Frame"



(c) Input variable: "dp"

Figure 1.31: Violin plots showing the residue of output variable "RF Forced Crippling". (a): Numerical inputs (binned). (b) and (c): categorical inputs.

- The output (illustration of it is provided in Table 1.24)is the resulting vector of bootstraps [P2.5, P97.5] associated to the input variable x_i.

For each categorical input variable x_i:

- The code checks whether the number of points inside each category fulfils the minimum bin size requirement (the minimum bin size requirement is just a name, note nevertheless that categorical variables have not been binned).
- For those categories for which the minimum bin size requirement is not met, the category is given the Global Uncertainty Model (GUM).

Table 1.24: LIUM: Bootstrapped CIs for binned input variable "FU.0430.26" (bins' extrema displayed as column names). Residue corresponds to output variable "RF Shear Panel Failure".

	Input: FU.0430.26; Output: RF Shear Panel Failure							
	[-1.656e+04, -1.325e+04]	**[-1.325e+04, -9938]**	**[-9938, -6625]**	**[-6625, -3313]**	**[-3313, 0]**	**[0, 3313]**	**[3313, 6625]**	**[6625, 9938]**
Mean	-0.003271	0.007046	-0.001071	0.001169	0.001253	0.0003371	-0.0001739	0.002927
CI	[-0.3023, 0.1606]	[-0.179, 0.1424]	[-0.2358, 0.1031]	[-0.6774, 0.09636]	[-0.6403, 0.1062]	[-1.116, 0.1228]	[-0.2585, 0.1092]	[-0.1927, 0.175]

Table 1.25: LIUM: Bootstrapped CIs for categorical input variable "dp" (possible values that the variable can take are displayed as column names). Residue corresponds to "RF Shear Panel Failure" output variable.

	Input: dp; Output: RF Forced Crippling	
	0.000000	**0.900000**
Mean	0.001506	0.002707
CI	[-0.5033, 0.09891]	[-0.5199, 0.1104]

- For those categories for which the minimum bin size requirement is met, the code computes an uncertainty model in terms of the bootstrap [P2.5, P97.5] (or a general [a,b]) with algorithm 3, and assigns to the category the resulting local uncertainty model.

- The output (illustration of it is provided in Table 1.25)is the resulting vector of bootstraps [P2.5, P97.5] (with length equal to the number of categories of the categorical variable x_i) associated to the input variable x_i.

This procedure is repeated for all m output variables.

C. Local Input Uncertainty Model (LIUM) evaluation

For each input variable:

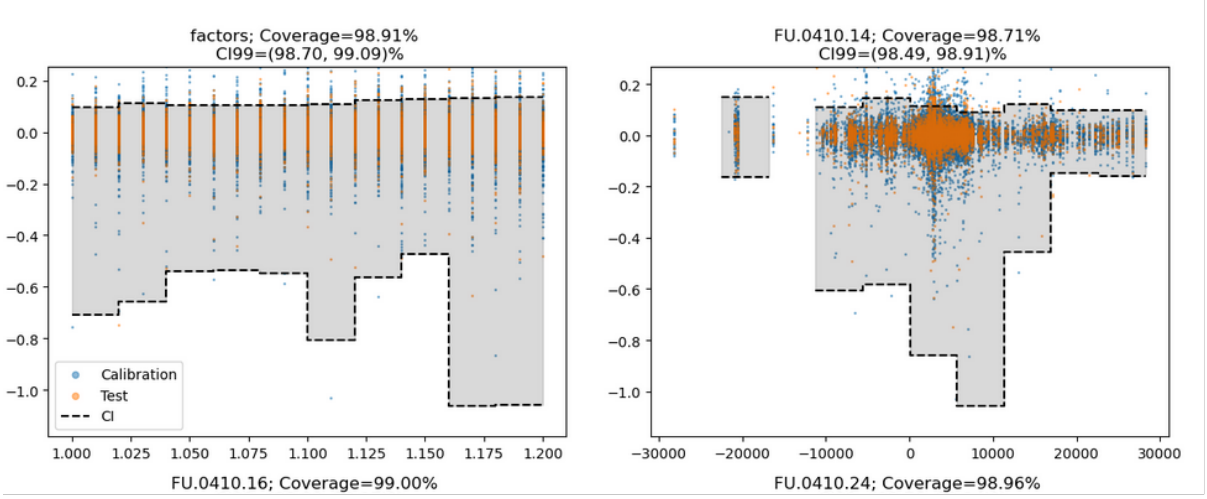
- The bootstrap error coverage in the test set associated to the LIUM vector is computed with algorithm 4, with the nuance that for each point in the test set, we need to find, based on the point's error, to which bin (or category) of the input variable it belongs, and then check whether such error is within the corresponding bootstrapped interval [P2.5, P97.5].

- The output is the bootstrap error coverage and its 95% confidence interval. The criteria for

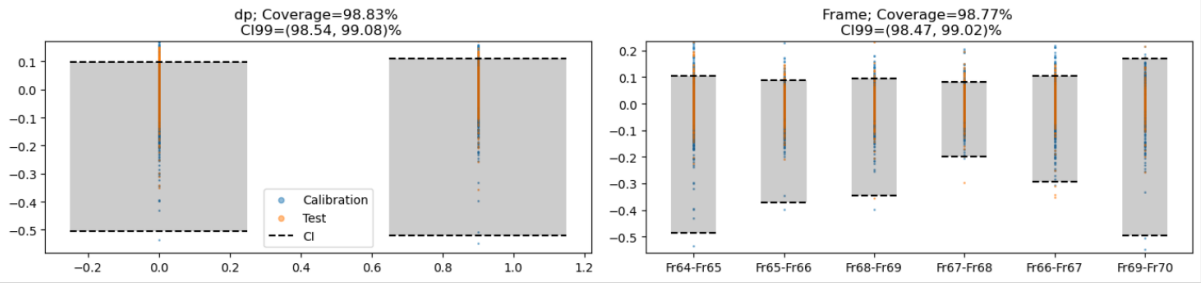
a correctly calibrated error coverage is that it is correctly calibrated if (b-a)% (by default, 95%) lies inside the bootstrapped error coverage confidence interval.

This procedure is repeated for all m output variables.

For illustration, a scatter plot of the error of each point in the calibration set is displayed in Figure 1.32, along with the uncertainty model interval (the upper and lower percentiles emphasized in blue and red respectively). This LIUM shows up as a piece-wise constant upper curve and a piece-wise constant lower curve.



(a) Numerical inputs: "FU.0410.14" (left), "FU.0410.15" (right). White slices correspond to bins which have not fulfilled the minimum bin size requirements. For such bins, the GUM is used instead of LIUM.



(b) Categorical inputs: "dp" (left), "Frame" (right)

Figure 1.32: LIUM coverage on the evaluation set. Residue of the output variable "RF Forced Crippling". The grey area represents the uncertainty CI calculated by the LIUM for the corresponding bin.

1.7.4 Full uncertainty model (FUM)

The FUM is a conservative uncertainty model that consistently takes, for a given point of the test set $(\mathbf{X}, y)$, the largest uncertainty

$$\text{FUM}((\mathbf{X}, y)) = \max\{\text{GUM}, \text{LOUM}(y), \text{LIUM}(x_1), \text{LIUM}(x_2), ..., \text{LIUM}(x_n)\}$$

Accordingly (vid. Figure 1.33),

- For each point in the test set with input variables $(x_1, ..., x_n)$ and output prediction y, the the larger out of all the intervals in the set

  $$\{\text{GUM}, \text{LOUM}(y), \text{LIUM}(x_1), \text{LIUM}(x_2), ..., \text{LIUM}(x_n)\}$$

  is selected and it is checked whether the error associated to this point lies inside the corresponding interval.

- The error coverage is computed, accordingly.

- The bootstrap error coverage is computed, as per previous sections, and the final output is the bootstrap error coverage and its 95% confidence interval, along with a markdown message about whether the FUM is well calibrated.

Bibliography

[1] P. Bijlaard. "On the Buckling of Stringer Panels Including Forced Crippling". In: *Journal of the Aeronautical Sciences* 22.7 (1955), pp. 491–501 (cit. on p. 1).

[2] F. P. Preparata and M. I. Shamos. "Convex Hulls: Basic Algorithms". In: *Computational Geometry: An Introduction*. New York, NY: Springer New York, 1985, pp. 95–149. ISBN: 978-1-4612-1098-6. DOI: 10.1007/978-1-4612-1098-6_3. URL: https://doi.org/10.1007/978-1-4612-1098-6_3 (cit. on p. 2).

[3] D. Barrett et al. "Measuring abstract reasoning in neural networks". In: *Proceedings of the 35th International Conference on Machine Learning*. Ed. by J. Dy and A. Krause. Vol. 80. Proceedings of Machine Learning Research. PMLR, Oct. 2018, pp. 511–520. URL: https://proceedings.mlr.press/v80/barrett18a.html (cit. on p. 3).

[4] B. M. Lake and M. Baroni. "Still not systematic after all these years: On the compositional skills of sequence-to-sequence recurrent networks". In: *CoRR* abs/1711.00350 (2017). arXiv: 1711.00350. URL: http://arxiv.org/abs/1711.00350 (cit. on p. 3).

[5] D. Saxton et al. "Analysing Mathematical Reasoning Abilities of Neural Models". In: *CoRR* abs/1904.01557 (2019). arXiv: 1904.01557. URL: http://arxiv.org/abs/1904.01557 (cit. on p. 3).

[6] T. Ebert, J. Belz, and O. Nelles. "Interpolation and extrapolation: Comparison of definitions and survey of algorithms for convex and concave hulls". In: *2014 IEEE Symposium on Computational Intelligence and Data Mining (CIDM)*. IEEE. 2014, pp. 310–314 (cit. on p. 3).

[7] W.-Y. Loh, C.-W. Chen, and W. Zheng. "Extrapolation errors in linear model trees". In: *ACM Transactions on Knowledge Discovery from Data (TKDD)* 1.2 (2007), 6–es (cit. on p. 3).

[8] P. Klesk. "Construction of a Neurofuzzy Network Capable of Extrapolating (and Interpolating) With Respect to the Convex Hull of a Set of Input Samples in R". In: *IEEE Transactions on Fuzzy Systems* 16.5 (2008), pp. 1161–1179. DOI: 10.1109/TFUZZ.2008.924337 (cit. on p. 3).

[9] R. Balestriero, J. Pesenti, and Y. LeCun. "Learning in high dimension always amounts to extrapolation". In: *arXiv preprint arXiv:2110.09485* (2021) (cit. on pp. 3, 4, 10).

[10] S. Marsland. *Machine Learning: An Algorithmic Perspective*. 2nd ed. Boca Raton, USA: Chapman & Hall/CRC, 2015 (cit. on pp. 3, 7).

[11] I. Bárány and Z. Füredi. "On the shape of the convex hull of random points". In: *Probability theory and related fields* 77 (1988), pp. 231–240 (cit. on p. 3).

[12] L. Bonnasse-Gahot. "Interpolation, extrapolation, and local generalization in common neural networks". In: *arXiv preprint arXiv:2207.08648* (2022) (cit. on pp. 4, 10, 13).

[13] H. Hotelling. "Analysis of a complex of statistical variables into principal components." In: *Journal of educational psychology* 24.6 (1933), p. 417 (cit. on p. 9).

[14] B. Rosner and D. Grove. "Use of the Mann–Whitney U-test for clustered data". In: *Statistics in medicine* 18.11 (1999), pp. 1387–1400 (cit. on p. 13).

[15] R. Velez Ibarrola and A. Garcia Perez. *Calculo de probabilidades y Estadistica Matematica*. 1st ed. Madrid, Spain: Universidad Nacional de Educacion a Distancia, 1994 (cit. on p. 15).

[16] D. Zhang. "A coefficient of determination for generalized linear models". In: *The American Statistician* 71.4 (2017), pp. 310–316 (cit. on p. 20).

[17] J. D. Jobson. *Applied multivariate data analysis: regression and experimental design*. Springer Science & Business Media, 2012 (cit. on pp. 20, 37).

[18] G. Chen, J. R. Gott, and B. Ratra. "Non-Gaussian Error Distribution of Hubble Constant Measurements". In: *Publications of the Astronomical Society of the Pacific* 115.813 (2003), p. 1269 (cit. on p. 22).

[19] P. Pernot, B. Huang, and A. Savin. "Impact of non-normal error distributions on the benchmarking and ranking of Quantum Machine Learning models". In: *Machine Learning: Science and Technology* 1.3 (2020), p. 035011 (cit. on p. 22).

[20] D. Smyl et al. "Learning and correcting non-Gaussian model errors". In: *Journal of Computational Physics* 432 (2021), p. 110152 (cit. on p. 22).

[21] L. Chai et al. "Using generalized Gaussian distributions to improve regression error modeling for deep learning-based speech enhancement". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 27.12 (2019), pp. 1919–1931 (cit. on p. 22).

[22] M. C. Jones and A. Pewsey. "Sinh-arcsinh distributions". In: *Biometrika* 96.4 (2009), pp. 761–780 (cit. on pp. 27, 28).

[23] B. Rosner. "Percentage points for a generalized ESD many-outlier procedure". In: *Technometrics* 25.2 (1983), pp. 165–172 (cit. on p. 27).

[24] B. Efron. "Bootstrap methods: another look at the jackknife". In: *Breakthroughs in statistics: Methodology and distribution*. Springer, 1992, pp. 569–593 (cit. on p. 29).

[25] E. B. Wilson. "Probable inference, the law of succession, and statistical inference". In: *Journal of the American Statistical Association* 22.158 (1927), pp. 209–212 (cit. on p. 32).

[26] L. Taylor and G. Nitschke. "Improving deep learning with generic data augmentation". In: *2018 IEEE symposium series on computational intelligence (SSCI)*. IEEE. 2018, pp. 1542–1547 (cit. on p. 35).

[27] T. K. Kim. "Understanding one-way ANOVA using conceptual figures". In: *Korean journal of anesthesiology* 70.1 (2017), pp. 22–26 (cit. on p. 37).

[28] B. R. Kirkwood and J. A. Sterne. *Essential medical statistics*. John Wiley & Sons, 2010 (cit. on p. 38).

[29] Y. Fujikoshi. "Two-way ANOVA models with unbalanced data". In: *Discrete Mathematics* 116.1-3 (1993), pp. 315–334 (cit. on p. 41).

[30] O. Montenbruck and E. Gill. *Satellite Orbits: Models, Methods and Applications*. 1st ed. Wessling, Germany: Springer, 2001 (cit. on p. 72).

[31] K. Yamanaka and F. Ankersen. "New State Transition Matrix for Relative Motion on an Arbitrary Elliptical Orbit". In: *Journal of Guidance, Control & Dynamics* 25.1 (2002), pp. 60–66. DOI: 10.2514/2.4875 (cit. on p. 73).

[32] D.-W. Gim and K. Alfriend. "State Transition Matrix of Relative Motion for the Perturbed Noncircular Reference Orbit". In: *Journal of Guidance Control and Dynamics* 26 (Nov. 2003), pp. 956–971. DOI: 10.2514/2.6924 (cit. on p. 73).

[33] S.-S. Exchange. *Practical Uses of an STM*. 2019. URL: https://space.stackexchange.com/questions/32916 (cit. on p. 74).

[34] ESA. *Precise Orbit Determination*. 2011. URL: https://gssc.esa.int/navipedia/index.php/Precise_Orbit_Determination (cit. on p. 74).

[35] C. Chao and F. Hoots. *Applied Orbit Perturbation and Maintenance*. Aerospace Press, 2018. ISBN: 9781523123346 (cit. on p. 79).

[36] W. E. Wiesel. *Modern Astrodynamics*. 2nd ed. Beavercreek, Ohio: Aphelion Press, 2010 (cit. on p. 83).

[37] M. Eckstein, C. Rajasingh, and P. Blumer. "Colocation Strategy and Collision Avoidance for the Geostationary Satellites at 19 Degrees West". In: *CNES International Symposium on Space Dynamics*. Vol. 25. Oberpfaffenhofen, Germany: DLR GSOC, Nov. 1989, pp. 60–66 (cit. on p. 84).

[38] S. D'Amico and O. Montenbruck. "Proximity Operations of Formation-Flying Spacecraft Using an Eccentricity/Inclination Vector Separation". In: *Journal of Guidance, Control & Dynamics* 29.3 (2006), pp. 554–563. DOI: 10.2514/1.15114 (cit. on pp. 84, 90, 91).

[39] D.-W. Gim and K. T. Alfriend. "Satellite Relative Motion Using Differential Equinoctial Elements". In: *Celestial Mechanics and Dynamical Astronomy* 92.4 (2005), pp. 295–336. DOI: 10.1007/s10569-004-1799-0 (cit. on pp. 85, 86).

[40] S. D'Amico. *Relative Orbital Elements as Integration Constants of Hill's Equations*. TN 05-08. Oberpfaffenhofen, Germany: Deutsches Zentrum für Luft- und Raumfahrt (DLR), 2005 (cit. on pp. 85, 90).

[41] H. Schaub. "Relative Orbit Geometry Through Classical Orbit Element Differences". In: *Journal of Guidance, Control & Dynamics* 27.5 (2004), pp. 839–848. DOI: 10.2514/1.12595 (cit. on pp. 85, 90).

[42] G. Gaias, C. Colombo, and M. Lara. "Accurate Osculating/Mean Orbital Elements Conversions for Spaceborne Formation Flying". In: (Feb. 2018). https://www.researchgate.net/publication/340378956_Accurate_OsculatingMean_Orbital_Elements_Conversions_for_Spaceborne_Formation_Flying (cit. on p. 86).

[43] T. Vincent Peters and R. Noomen. "Linear Cotangential Transfers and Safe Orbits for Elliptic Orbit Rendezvous". In: *Journal of Guidance, Control & Dynamics* 44.4 (2021), pp. 732–748. DOI: 10.2514/1.G005152 (cit. on pp. 92, 93).

[44] R. H. Lopes, I. Reid, and P. R. Hobson. "The two-dimensional Kolmogorov-Smirnov test". In: (2007).

[45] G. Schoups and J. A. Vrugt. "A formal likelihood function for parameter and predictive inference of hydrologic models with correlated, heteroscedastic, and non-Gaussian errors". In: *Water Resources Research* 46.10 (2010).

[46] B. D. Tapley, B. E. Schutz, and G. H. Born. *Statistical Orbit Determination*. 1st ed. Amsterdam, Netherlands: Elsevier, 2004.

[47] K. T. Alfriend and Srinivas. *Spacecraft Formation Flying*. 1st ed. Oxford, United Kingdom: Elsevier, 2010.

[48] R. H. Battin. *An Introduction to the Mathematics and Methods of Astrodynamics, Revised Edition*. 1st ed. Reston, Virginia: American Institute of Aeronautics and Astronautics, 1999.

[49] D. Brouwer and G. M-Clemence. *Methods of Celestial Mechanics*. Brackley Square House, London: Academic Press, 1961.

[50] H. Schaub and J. L. Junkins. *Analytical Mechanics of Space Systems*. Reston, VA: AIAA Education Series, Oct. 2003. DOI: 10.2514/4.861550.

[51] W. Fehse. *Automated Rendezvous and Docking of Spacecraft*. Cambridge Aerospace Series. Cambridge: Cambridge University Press, 2003. DOI: 10.1017/CBO9780511543388.

[52] P. K. S. Dennis D. McCarthy. *Time: From Earth Rotation to Atomic Physics*. Wiley-VCH, 2009. ISBN: 3527407804; 9783527407804.

[53] K. Wakker. *Fundamentals of Astrodynamics*. Jan. 2015. ISBN: 978-94-6186-419-2.

[54] A. H. Nayfeh. *Perturbation Methods*. Weinheim, Germany: Wiley-VCH, 2004.

[55] W. M. Kaula. *Theory of Satellite Geodesy*. Mineola, New York: Dover Publications, 2013.

[56] F. Tisserand. *Traité de Mécanique Céleste*. Vol. 1. Paris, 1889.

[57] J. Sullivan, S. Grimberg, and S. D'Amico. "Comprehensive Survey and Assessment of Spacecraft Relative Motion Dynamics Models". In: *Journal of Guidance, Control & Dynamics* 40.8 (2017), pp. 1837–1859. DOI: 10.2514/1.G002309.

[58] H. Schaub and K. T. Alfriedn. "Hybrid Cartesian and Orbit Element Feedback Law for Formation Flying Spacecraft". In: *Journal of Guidance, Control & Dynamics* 25.2 (2002), pp. 387–393. DOI: 10.2514/2.4893.

[59] N. Capitaine. "The Celestial Pole Coordinates". In: *Celestial Mechanics and Dynamical Astronomy* 48 (1990), pp. 127–143.

[60] D. D. McCarthy. *IERS Conventions (1992)*. 21. Paris, France: Central Bureau of IERS - Observatoire de Paris, 1996.

[61] J. Williams. *LVLH Transformations*. https://degenerateconic.com/uploads/2015/03/lvlh.pdf. 2014.

[62] H. Fiedler. *Analysis of TerraSAR-L Cartwheel Constellations*. https://elib.dlr.de/22345/. Oberpfaffenhofen, Germany: Deutsches Zentrum für Luft- und Raumfahrt (DLR), Nov. 2003.

[63] G. W. Hill. "Researches in the Lunar Theory". In: *American Journal of Mathematics* 1.2 (1878), pp. 129–147. DOI: 10.2307/2369304.

[64] W. H. Clohessy and R. S. Wiltshire. "Terminal Guidance System for Satellite Rendezvous". In: *Journal of the Aerospace Sciences* 27.9 (1960), pp. 653–658. DOI: 10.2514/8.8704.

[65] R. Broucke. "On the Matrizant of the Two-Body Problem". In: *Astronomy and Astrophysics* 6 (June 1970), p. 173.

[66] J. Tschauner and P. Hempel. "Optimale Beschleunigungsprogramme fÃ¼r das Rendezvous-Manöver". In: *Astronautica Acta* 10 (1964), pp. 296–307.

[67] T. Carter. "State Transition Matrices for Terminal Rendezvous Studies: Brief Survey and New Example". In: *Journal of Guidance Control and Dynamics* 21 (Jan. 1998), pp. 148–155. DOI: 10.2514/2.4211.

[68] O. Montenbruck, M. Kirschner, and S. D'Amico. "E/I-vector separation for safe switching of the GRACE formation". In: *Aerospace Science and Technology* 10 (2006), pp. 628–635. DOI: 10.1016/j.ast.2006.04.001.

[69] G. Gaias, J.-S. Ardaens, and C. Colombo. "Precise line-of-sight modelling for angles-only relative navigation". In: *Advances in Space Research* 67.11 (2021). Satellite Constellations and Formation Flying, pp. 3515–3526. DOI: 10.1016/j.asr.2020.05.048.

[70] G. Gaias, C. Colombo, and M. Lara. "Analytical Framework for Precise Relative Motion in Low Earth Orbits". In: *Journal of Guidance, Control, and Dynamics* 43.5 (2020), pp. 915–927. DOI: 10.2514/1.G004716.

[71] D. Brouwer. "Solution of the problem of artificial satellite theory without drag". In: *Astronomical Journal* 64.5 (Nov. 1959), p. 378. DOI: 10.1086/107958.

[72] R. H. Lyddane. "Small eccentricities or inclinations in the Brouwer theory of the artificial satellite". In: *Astronomical Journal* 68 (Oct. 1963), p. 555. DOI: 10.1086/109179.

[73] A. Deprit. "Canonical transformations depending on a small parameter". In: *Celestial Mechanics* 1.1 (Mar. 1969), pp. 12–30. DOI: 10.1007/BF01230629.

[74] A. Deprit. "Delaunay Normalisations". In: *Celestial Mechanics* 26.1 (Jan. 1982), pp. 9–21. DOI: 10.1007/BF01233178.

[75] G. Hori. "Theory of General Perturbation with Unspecified Canonical Variable". In: *Publications of the Astronomical Society of Japan* 18 (Jan. 1966), pp. 287–296.

[76] Y. Chihabi and S. Ulrich. "Spacecraft Formation Guidance Law using a State Transition Matrix With Gravitational, Drag and Third-Body Perturbations". In: Jan. 2020. DOI: 10.2514/6.2020-1460.

[77] G. Gaias, J. Ardaens, and O. Montenbruck. "Model of J2 Perturbed Satellite Relative Motion with Time-Varying Differential Drag". In: *Celestial Mechanics and Dynamical Astronomy* (2015).

[78] A. Koenig, T. Guffanti, and S. D'Amico. "New State Transition Matrices for Relative Motion of Spacecraft Formations in Perturbed Orbits". In: Sept. 2016. DOI: 10.2514/6.2016-5635.

[79] A. Biria and R. Russell. "A Satellite Relative Motion Model Including J2 and J3 via Vinti's Intermediary". In: Feb. 2016.

[80] K. Alfriend and H. Yan. "An Orbital Elements Based Approach to the Nonlinear Formation Flying Problem". In: Toulouse, France, Feb. 2016.

[81] Mathworks. *Convert complex diagonal form into real diagonal form*. 2022. URL: https://www.mathworks.com/help/matlab/ref/cdf2rdf.html.

[82] M. R. Delgado. *Lecture notes: Basics of Orbital Mechanics I*. Apr. 2008.

[83] F. G. Nievinski. *subtightplot*. https://www.mathworks.com/matlabcentral/fileexchange/39664-subtightplot. 2013.

[84] J. C. Lansey. *linspecer*. https://www.mathworks.com/matlabcentral/fileexchange/42673-beautiful-and-distinguishable-line-colors-colormap. 2015.

[85] Jan. *WindowAPI*. https://www.mathworks.com/matlabcentral/fileexchange/31437-windowapi. 2013.

[86] T. Davis. *Arrow3*. https://www.mathworks.com/matlabcentral/fileexchange/14056-arrow3. 2022.

[87] E. Duenisch. *latexTable*. https://www.mathworks.com/matlabcentral/fileexchange/44274-latextable. 2016.