SYSTEMS PLANNING & ANALYSIS

Distribution Free Uncertainty for CERs

William King (wking@spa.com)

Shaun Irvin (sirvin@spa.com)

15 May 2024

Understanding Prediction Intervals

- What are they?
  - A range of values likely to contain the true value of a new observation, with a certain level of confidence; provide an estimate of uncertainty surrounding the prediction
  - Based on a fitted model and account for variation in:
    - Response variable; Standard Estimating Error (SEE, $\sqrt{MSE}$ or $SPE \; * \; \hat{y}$) is usually an unbiased estimator of this variation
    - Estimating regression parameters: usually a function of the distance between independent variables and their means, scaled by their covariance

- Classical formula for calculating Prediction Intervals for linear regression:

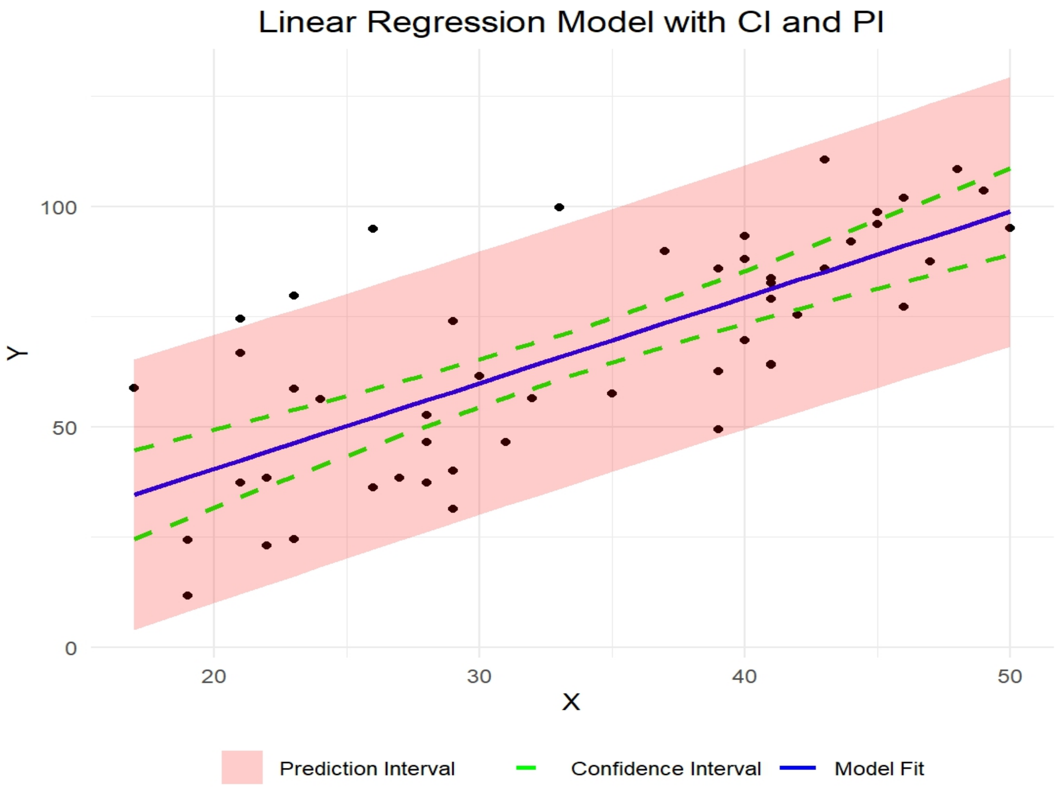
$$\hat{y} \pm \; t_{\frac{\alpha}{2},n-p-1} \; \times \sqrt{MSE\left(1+\frac{1}{n}+\frac{(x \; - \; \bar{x})^2}{\sum(x_i-\bar{x})^2}\right)}$$

where $MSE \; = \frac{1}{n-p-1}\sum_{i=1}^{n}(y_i-\hat{y}_i)^2$

- Assumptions:
  - Linearity, Independent errors, Normal errors, Equal error variance

Prediction Intervals Example

- How are they different from Confidence Intervals?
  - Prediction Interval - range for a predicted value for a given input: $\hat{\boldsymbol{y}} = \boldsymbol{x_0\beta} + \boldsymbol{\varepsilon}$
  - Confidence Interval - range for the ***expected*** predicted value ($E[\varepsilon] = 0$): $\hat{\boldsymbol{y}} = \boldsymbol{x_0\beta}$
  - Prediction Intervals have a wider range since they account for error in the prediction



CI : $\hat{y} \pm t_{\frac{\alpha}{2},n-p-1} \times \sqrt{MSE\left(\frac{1}{n} + \frac{(x-\bar{x})^2}{\sum(x_i-\bar{x})^2}\right)}$

Vs.

PI : $\hat{y} \pm t_{\frac{\alpha}{2},n-p-1} \times \sqrt{MSE\left(1 + \frac{1}{n} + \frac{(x-\bar{x})^2}{\sum(x_i-\bar{x})^2}\right)}$

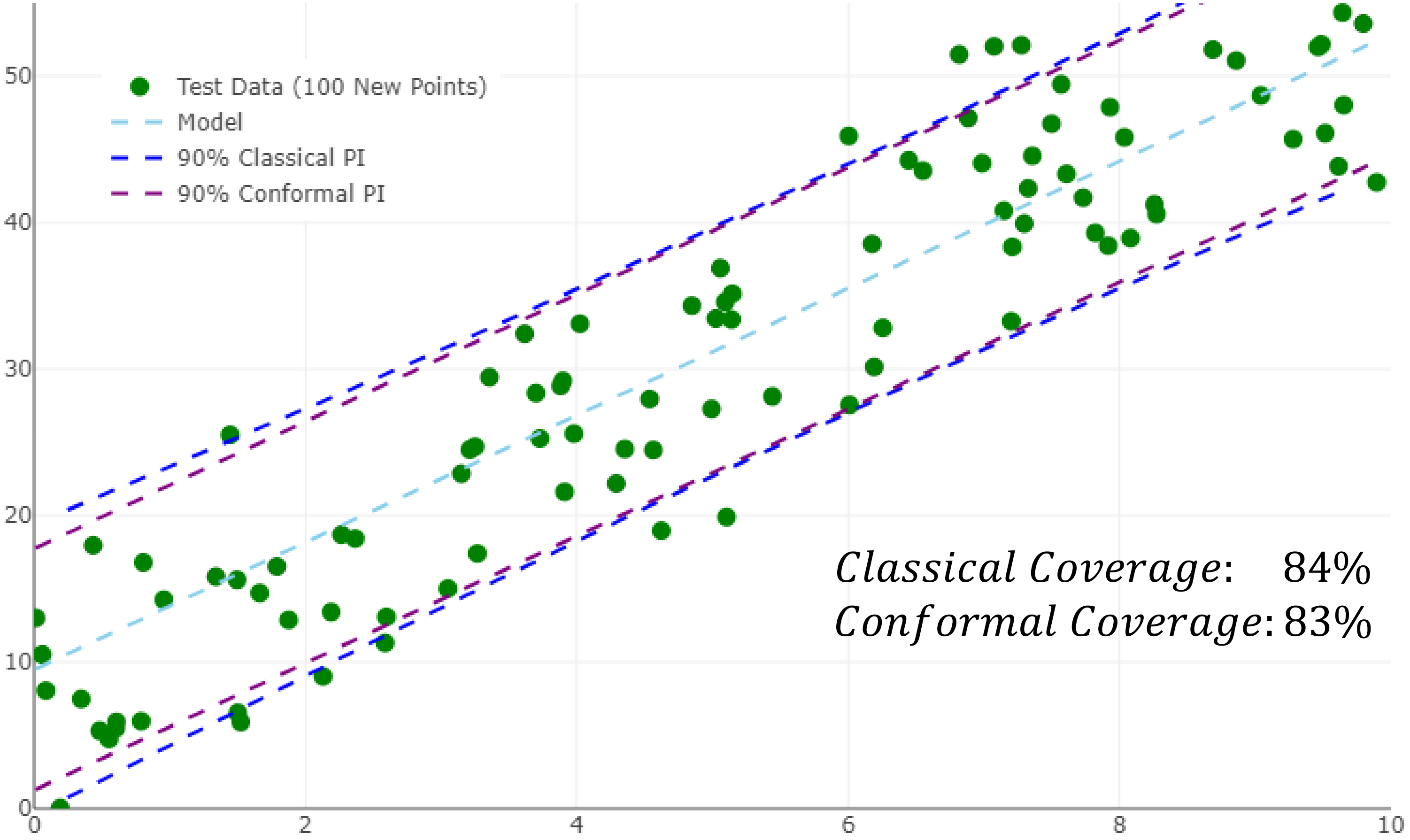
Extra Error Term!

Conformal Prediction

- Conformal Prediction is a technique that generates prediction intervals with rigorous statistical coverage guarantees and without distributional assumptions
  - – Applies to any machine learning algorithm or "black-box" model
  - – Applies to both regression and classification problems
  - – Only requires the exchangeability of data (a weaker assumption than independence)
    - Classical: Linearity, Independent errors, Normal errors, Equal error variance (L I N E)
    - Conformal: Independent errors (-- I – --)
- For regression problems, conformal prediction only requires access to a trained model (i.e., CER) and unseen calibration data that was not used for training:
  1. Use the previously trained model to predict the unseen calibration data and calculate residuals for the calibration data
  2. Find the percentile ($\hat{q}_{1-\alpha}$) of the calibration residuals corresponding to your level of significance
  3. Apply that calibration residual percentile to generate intervals around new predictions ($\hat{y} \pm \hat{q}_{1-\alpha}$)
- Remarkably, this simple process yields statistical coverage guarantees given the exchangeability of the underlying data

Conformal Intuition – How does this work?

- Suppose we have some training data (●), some calibration data (●), and a new data point we need to predict (▲)



Classical Coverage: 84%

Conformal Coverage: 83%

Why use conformal prediction?

- Conformal prediction exploits calibration data to produce a realistic representation of how the model performs on new, unseen data (e.g., the new system you are about to estimate)
  - Essentially, conformal prediction exploits the information we gain from applying the model to labeled data unseen at the time of training, to estimate the error we can expect when we apply the model to unlabeled data (i.e., where the response is unknown)
  - Regardless of how well (or how poorly) your model fits the underlying training data, the calibration data still provides an unbiased assessment of how well the model can generalize against completely new data
- Suppose you were handed a “black-box” cost model or CER. How would you go about measuring if you were using the model correctly? How would you go about understanding the predictive uncertainty of using such a model?
  - If you know what the outputs should be for a particular set of inputs, and the model approximates the outputs, you are likely using the model correctly
  - If you know many of these (input, output) pairs, you can know how well the model will perform against the new program you are about to estimate
  - Conformal prediction provides a statistically sound framework to answer these questions

How else might we quantify uncertainty?

- A naïve approach would apply the conformal procedure previously described with residuals on the training data acting as calibration data:
  - $PI: \hat{f}(X_i) \pm$ the $(1-\alpha)*100$ percentile of $\left|Y_1 - \hat{f}(X_1)\right|, \dots, \left|Y_n - \hat{f}(X_n)\right|$
  - Leads to be artificially narrow prediction intervals with overfitting (i.e., when the model performs well on training data, but not on unseen data)
    - Depends on you having access to the underlying training data
  - This approach has NO statistical coverage guarantees
  - This approach does NOT account for variability of residuals across the input space
- A slightly better approach would be to use Leave-One-Out Cross-Validation (LOOCV, a.k.a. "jackknife") residuals on the training data acting as calibration data:
  - $PI: \hat{f}(X_i) \pm$ the $(1-\alpha)*100$ percentile of $\left|Y_1 - \hat{f}_{-1}(X_1)\right|, \dots, \left|Y_n - \hat{f}_{-n}(X_n)\right|$
  - Leads to slightly wider prediction intervals that are more robust than the naïve approach
    - Depends on you having access to the underlying training data
  - This approach still has NO statistical coverage guarantees
  - This approach still does NOT account for variability of residuals across the input space

Conformal Variants

- Full Conformal Prediction (Visual):
  - PI: $\left\{y : \left|y - \hat{f}_y(x_{n+1})\right| \leq Q_{1-\alpha}(R_1, \dots, R_n, R_{n+1})\right\}$
    - Where $\hat{f}_y$ is the model trained as if (x_{n+1}, y) were a new data point, $R_i = \left|y_i - \hat{f}_y(x_i)\right|$ and $Q_{1-\alpha}$ is the $(1-\alpha) * 100$ percentile of the residuals
  - Does not require a calibration dataset, but requires re-fitting the model for every possible value of y whenever a new prediction needs to be made
    - Since this is infeasible in practice, usually a finite grid of y-values are selected and evaluated, but this can be very computationally expensive even with small datasets
  - This approach has strong statistical coverage guarantees ($\geq 1 - \alpha$)
- Split Conformal Prediction (what we have previously discussed, Visual):
  - Partition data into training (size m) and calibration (size $n - m$) sets:
  - PI: $\hat{f}_{train}(x_{n+1}) \pm Q_{1-\alpha}\left(R_1^C, \dots, R_{n-m}^C\right)$
    - Where $\hat{f}_{train}$ is the model trained on the m training data points, $R_i^C = \left|y_i - \hat{f}_{train}(x_i)\right|$ for all i in the calibration set, and $Q_{1-\alpha}$ is defined as above
  - Requires sacrificing data to the calibration set, but only needs to be fit once
    - Calibration data can be hard to come by (nearly 1000 calibration data points are needed to achieve coverage between 88-92% at a 90% confidence level)
  - This approach has strong statistical coverage guarantees ($\geq 1 - \alpha$)

Conformal Variants

- CV+ for K-fold Cross-Validation (CV+, Visual)
  - Partition data into K non-overlapping subsets: $S_1, \dots, S_k$
  - $PI: \left[Q_\alpha\left(\hat{f}_{-S_{k(i)}}(x_{n+1}) - R_i^{CV}\right), Q_{1-\alpha}\left(\hat{f}_{-S_{k(i)}}(x_{n+1}) + R_i^{CV}\right)\right]$
    - Where $\hat{f}_{-S_{k(i)}}$ is the model trained with the k-th subset removed, $k(i)$ indicates the subset that includes the i-th data point, $R_i^{CV} = \left|y_i - \hat{f}_{-S_{k(i)}}(x_i)\right|$ is the absolute value of the out-of-fold residual, and Q_α is defined as before
  - Does not require a separate calibration data set and only requires fitting subsets of the data K times
    - The out-of-fold residuals stand in proxy for the calibration dataset, since they are unseen at the time each model is trained during cross-validation
    - If you are already performing cross-validation, then you are already training these models and calculating their out-of-fold residuals
      - The only extra things you need to do is to save each $\hat{f}_{-S_{k(i)}}$ model and the association of out-of-fold residuals to subsets $k(i)$
    - Note: CV+ where $K = n$ is called the Jackknife+ (a form of Leave-One-Out cross-validation)
  - Less strong than Full or Split conformal, but CV+ offers a statistical coverage guarantee
    - CV+ sacrifices some of the statistical coverage guarantee but doesn't require a separate calibration dataset, and doesn't involve fitting the model infinitely many times

Conformal Method Comparison

Variant	Training Cost	Calibration Data	Coverage Guarantee	Empirical Coverage	Notes
Full	∞	No	$\geq 1-\alpha$	$\approx 1-\alpha$	Infeasible even with small datasets
Split	1	Yes	$\geq 1-\alpha$	$\approx 1-\alpha$	Good when you have lots of calibration data or a computationally expensive model; stronger statistical guarantees than CV+
K-fold CV+	K	No	$\geq 1-2\alpha$	$\gtrapprox 1-\alpha$	Good when you have less data, or a very complex model; substantially computationally cheaper than Full conformal, but more costly than split conformal

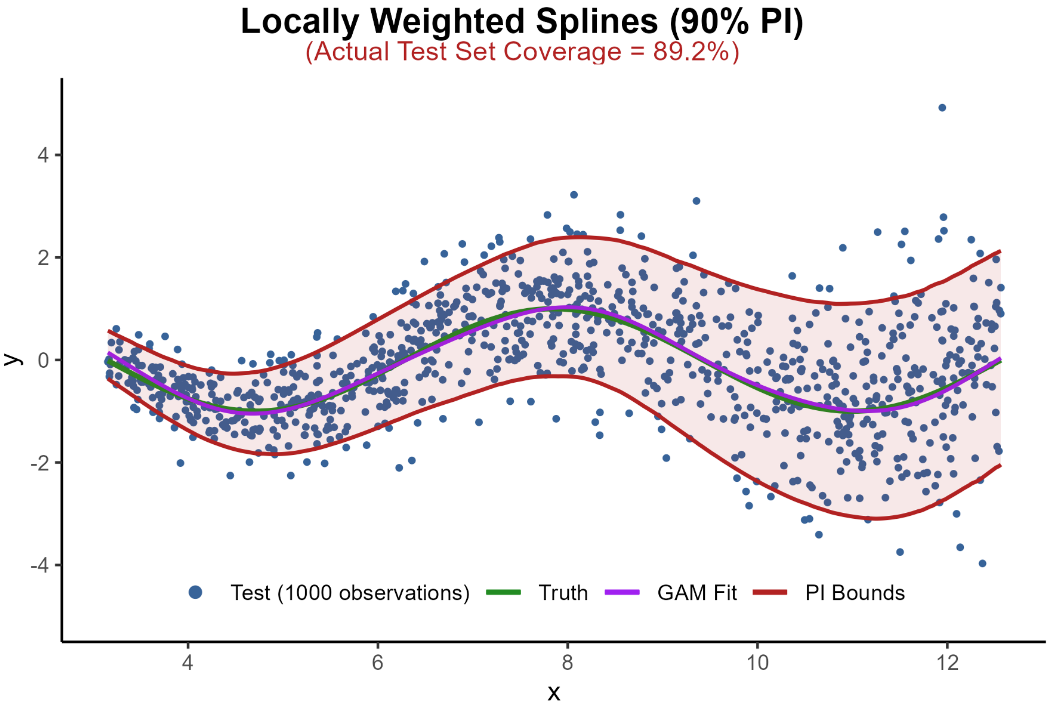
- K-Fold CV+ offers a balance between the computational cost of Full Conformal and the calibration data size requirements for Split Conformal
- If you're already performing cross-validation to evaluate your models, CV+ is essentially computationally free (you just need to save the sub-models and residuals you are already calculating)
- All three methods are distribution free!

Locally Weighted CV+

- The conformal methods previously discussed tend to generate prediction intervals with constant width
- This behavior makes sense with additive errors, but not with the multiplicative errors we tend to see with cost data
- Luckily, conformal prediction works with any non-conformity measure (Visual)
  - Up to this point, we used the absolute value of the calibration residuals as a non-conformity measure
  - Scaling the absolute value of the residuals by an estimate of the residual spread is still a valid non-conformity measure that preserves the statistical coverage guarantees
  - Previously, we defined $R_i^{CV} = \left| y_i - \hat{f}_{-S_{k(i)}}(x_i) \right|$, now we consider $R_i^{LW} = \frac{\left| y_i - \hat{f}_{-S_{k(i)}}(x_i) \right|}{\hat{\rho}_{-S_{k(i)}}(x_i)}$
    - Where $\hat{\rho}_{-S_{k(i)}}(x_i)$ is the estimate of the conditional mean absolute deviation of the residuals from $\hat{f}_{-S_{k(i)}}$
      - This involves fitting two models at each step of cross-validation: one model to predict the response as usual, and one model to predict the absolute value of the in-bag residuals
    - $PI: \left[Q_\alpha \left(\hat{f}_{-S_{k(i)}}(x_{n+1}) - R_i^{LW} * \hat{\rho}_{-S_{k(i)}}(x_i) \right), Q_{1-\alpha} \left(\hat{f}_{-S_{k(i)}}(x_{n+1}) + R_i^{LW} * \hat{\rho}_{-S_{k(i)}}(x_i) \right) \right]$

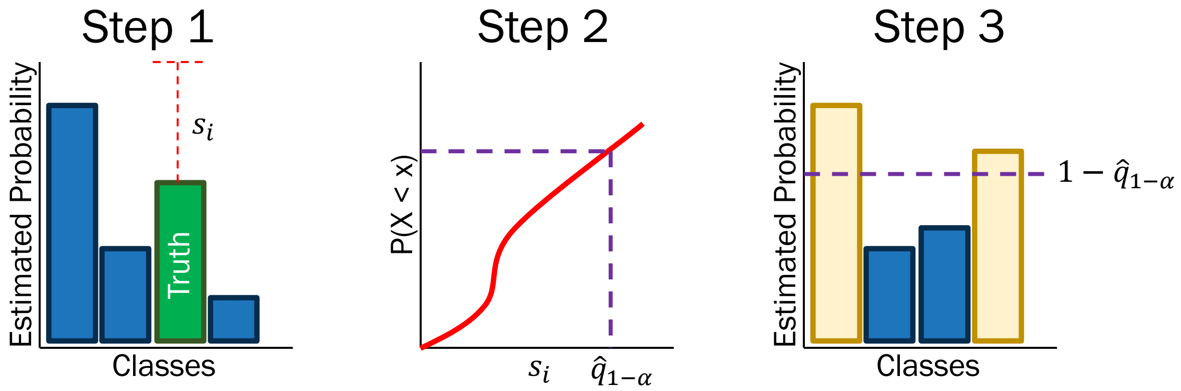
Applications to Regression

- Create datasets to train/calibrate a ML model using CV+ method
- Fit a RF model on training data and plot predictions of test data
- Use CV+ method to determine 90% Prediction Interval bounds
- Explore other CV+ variants
  - Locally Weighted
  - Generalized Additive Model (GAM) using splines



Conformal Prediction for Classification

- In a classification setting, conformal prediction produces prediction sets, that are guaranteed to contain the true label with some measure of statistical certainty (where larger prediction sets indicate more uncertainty)
- For classification problems, conformal prediction only requires access to a trained model that can output estimated class probabilities and unseen calibration data that was not used for training:
  1. Predict $class\ \widehat{probabilities}$ for the unseen calibration data and compute calibration scores as $s_i = 1 - class\ \widehat{probabilities}_i$ for the true classes
  2. Find the percentile ($\hat{q}_{1-\alpha}$) of the calibration scores corresponding to your level of significance
  3. Form predictive sets with all classes whose estimated probability is greater than $1 - \hat{q}_{1-\alpha}$



Graphic adapted from Angelopoulos and Bates, 2022

Applications to Classification

- Suppose we are interested in predicting a satellite's mission given its orbital characteristics (Perigee, Apogee and Eccentricity)
- Collect 5464 data points and randomly split data into training (4675 data points), calibration (520 points points) and testing (269 data points)
- Train a Random Forest model and tune hyperparameters with cross-validation
  - Model has 10-Fold CV accuracy of 87%
- Apply the model to estimate class probabilities on the calibration data, and calculate calibration scores against the true classes
  - For $\alpha = 10\%$, Obtain $q_{0.9} = 0.806$, which implies we'll include any class into our predictive sets whose estimated probability greater than $1 - q_{0.9} = 0.194$
    - Performing this procedure on the testing data results in a 91% coverage (meaning 91% of the predictive sets created following this rule contained the true label in the testing data)
  - Some examples from the Testing Data:

True Class	Predictive Set	Covered?
'Communications'	'Communications'	Yes
'Tech Development'	'Space Science', 'Tech Development'	Yes
'Communications'	'Earth Obs', 'Space Science', 'Tech Development'	No

Future Research

- Hierarchical classification for WBS normalization
  - Applying conformal prediction to hierarchical classification for WBS normalization can direct human intervention to elements with large prediction sets (i.e., where there the algorithm is highly uncertain)
- Application of Conformal Prediction in time-series data
  - Developing a modified conformal prediction technique that incorporates adjustments for autocorrelation and data trends
  - Impacts on economic forecasting, stock market predictions, weather forecasting, etc.
- Conformal Prediction in High-Dimensional data settings
  - Integrating conformal prediction with principal component analysis (PCA) to reduce dimensionality and handle situations where the number of variables is large and the number of observations is low
- Many packages in R and Python to facilitate conformal prediction
  - MAPIE (Model Agnostic Prediction Interval Estimator) for Python
  - tidymodels for R
  - Awesome Conformal Prediction for a curated list of “videos, tutorials, books, papers, PhD and MSc theses, articles and open-source libraries”

Conclusions

- Conformal prediction enables distribution free uncertainty with any machine learning algorithm
  - Only requirement is the exchangeability of the data (a weaker form of the i.i.d. assumption we make with classical approaches)
  - We get a rigorous statistical coverage guarantee regardless of how well the underlying model fits the data
  - As we embrace more accurate regression techniques that are less interpretable than classical approaches, we don’t have to sacrifice predictive uncertainty
- CV+ is a conformal technique that balances computational cost with the need for lots of calibration data for regression problems
  - If you’re already performing cross-validation, CV+ is basically computationally free
  - CV+ offers guaranteed coverage of at least $1 - 2 * \alpha$ with empirical coverage often close to $1 - \alpha$

Backup

References

- *Predictive Inference with the Jackknife+*
  - Barber, Candes, Ramdas and Tibshirani, 2021, The Annuals of Statistics
- *Distribution-Free Predictive Inference for Regression*
  - Lei, G'Sell, Rinaldo, Tibshirani and Wasserman, 2018, Journal of the American Statistical Association
- *A Tutorial on Conformal Prediction*
  - Shafer and Vovk, 2008, Journal of Machine Learning Research
- *A Gentle Introduction to Conformal Prediction and Distribution-Free Uncertainty Quantification*
  - Angelopoulos and Bates, 2022, arXiv:2107.07511v6

Full Conformal Example

Data

y	x
y1	x1
y2	x2
y3	x3
y4	x4
y5	x5
y6	x6
y7	x7
y8	x8
y9	y9
y10	x10

Augmented Data

y	x
y1	x1
y2	x2
y3	x3
y4	x4
y5	x5
y6	x6
y7	x7
y8	x8
y9	y9
y10	x10
y10	x11

$\hat{f}_{y10}$

Augmented Data Residuals

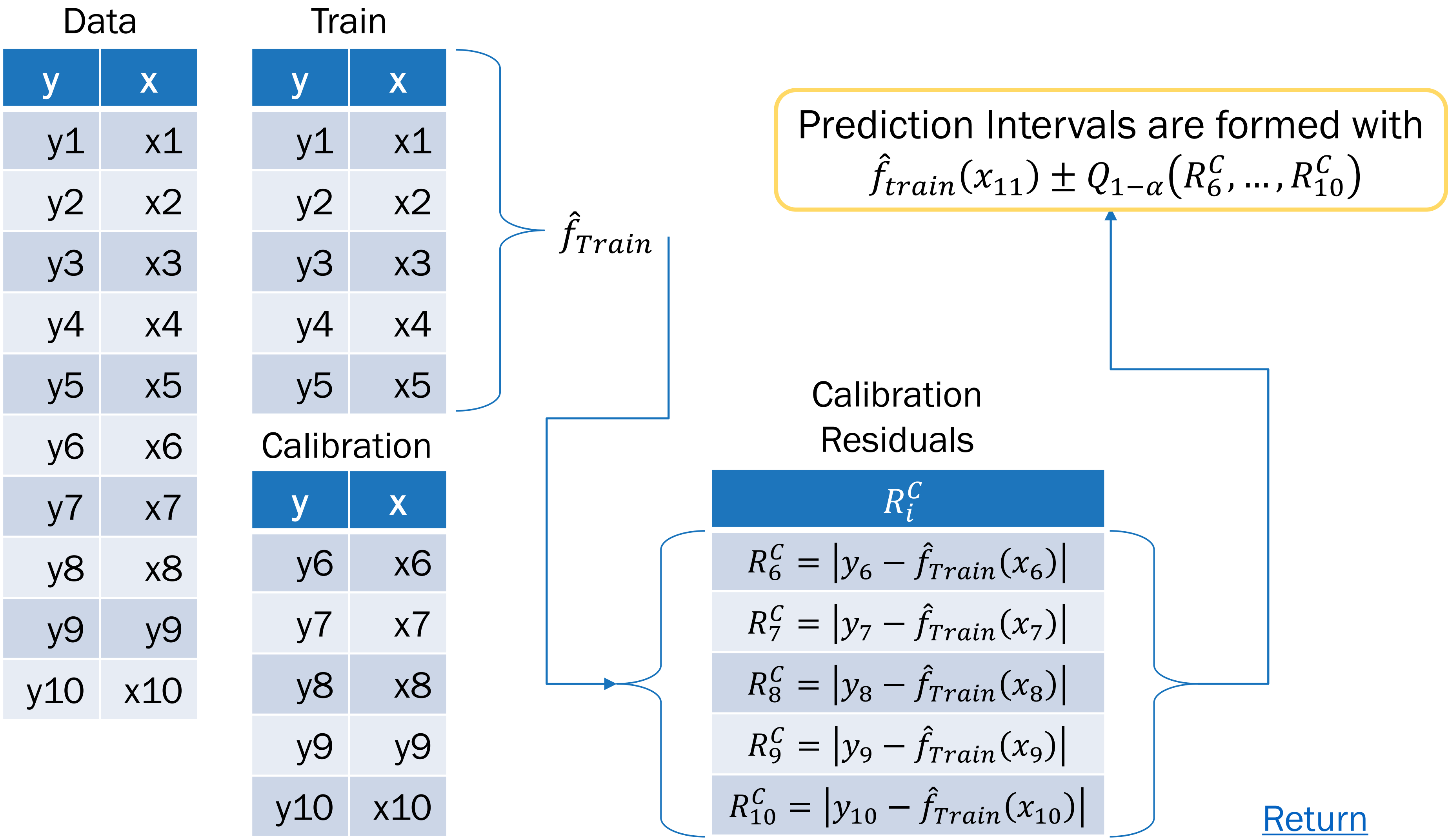
R_i
$R_1 = \lvert y_1 - \hat{f}_{y10}(x_1)\rvert$
$R_2 = \lvert y_2 - \hat{f}_{y10}(x_2)\rvert$
$R_3 = \lvert y_3 - \hat{f}_{y10}(x_3)\rvert$
$R_4 = \lvert y_4 - \hat{f}_{y10}(x_4)\rvert$
$R_5 = \lvert y_5 - \hat{f}_{y10}(x_5)\rvert$
$R_6 = \lvert y_6 - \hat{f}_{y10}(x_6)\rvert$
$R_7 = \lvert y_7 - \hat{f}_{y10}(x_7)\rvert$
$R_8 = \lvert y_8 - \hat{f}_{y10}(x_8)\rvert$
$R_9 = \lvert y_9 - \hat{f}_{y10}(x_9)\rvert$
$R_{10} = \lvert y_{10} - \hat{f}_{y10}(x_{10})\rvert$
$R_{11} = \lvert y_{10} - \hat{f}_{y10}(x_{11})\rvert$

y10 is in the PI iff

$\boldsymbol{R_{11} \leq Q_{1-\alpha}(R_1, \dots, R_{11})}$

Return

Split Conformal Example



Data

y	x
y1	x1
y2	x2
y3	x3
y4	x4
y5	x5
y6	x6
y7	x7
y8	x8
y9	y9
y10	x10

Train

y	x
y1	x1
y2	x2
y3	x3
y4	x4
y5	x5

$\hat{f}_{Train}$

Calibration

y	x
y6	x6
y7	x7
y8	x8
y9	y9
y10	x10

Calibration Residuals

R_i^C
$R_6^C = \lvert y_6 - \hat{f}_{Train}(x_6) \rvert$
$R_7^C = \lvert y_7 - \hat{f}_{Train}(x_7) \rvert$
$R_8^C = \lvert y_8 - \hat{f}_{Train}(x_8) \rvert$
$R_9^C = \lvert y_9 - \hat{f}_{Train}(x_9) \rvert$
$R_{10}^C = \lvert y_{10} - \hat{f}_{Train}(x_{10}) \rvert$

Prediction Intervals are formed with
$\hat{f}_{train}(x_{11}) \pm Q_{1-\alpha}\left(R_6^C, \ldots, R_{10}^C\right)$

Return

CV+ Example

Return

Data

y	x
y1	x1
y2	x2
y3	x3
y4	x4
y5	x5
y6	x6
y7	x7
y8	x8
y9	y9
y10	x10

Fold 1

y	x
y1	x1
y2	x2
y3	x3
y4	x4
y5	x5

$\hat{f}_{-S_2}$

Fold 2

y	x
y6	x6
y7	x7
y8	x8
y9	y9
y10	x10

$\hat{f}_{-S_1}$

CV+ Residuals

R_i^{CV}
$R_1^{CV} = \lvert y_1 - \hat{f}_{-S_1}(x_1) \rvert$
$R_2^{CV} = \lvert y_2 - \hat{f}_{-S_1}(x_2) \rvert$
$R_3^{CV} = \lvert y_3 - \hat{f}_{-S_1}(x_3) \rvert$
$R_4^{CV} = \lvert y_4 - \hat{f}_{-S_1}(x_4) \rvert$
$R_5^{CV} = \lvert y_5 - \hat{f}_{-S_1}(x_5) \rvert$

R_i^{CV}
$R_6^{CV} = \lvert y_6 - \hat{f}_{-S_2}(x_6) \rvert$
$R_7^{CV} = \lvert y_7 - \hat{f}_{-S_2}(x_7) \rvert$
$R_8^{CV} = \lvert y_8 - \hat{f}_{-S_2}(x_8) \rvert$
$R_9^{CV} = \lvert y_9 - \hat{f}_{-S_2}(x_9) \rvert$
$R_{10}^{CV} = \lvert y_{10} - \hat{f}_{-S_2}(x_{10}) \rvert$

CV+ Predictions

Low	High
$\hat{f}_{-S_1}(x_{11}) - R_1^{CV}$	$\hat{f}_{-S_1}(x_{11}) + R_1^{CV}$
$\hat{f}_{-S_1}(x_{11}) - R_2^{CV}$	$\hat{f}_{-S_1}(x_{11}) + R_2^{CV}$
$\hat{f}_{-S_1}(x_{11}) - R_3^{CV}$	$\hat{f}_{-S_1}(x_{11}) + R_3^{CV}$
$\hat{f}_{-S_1}(x_{11}) - R_4^{CV}$	$\hat{f}_{-S_1}(x_{11}) + R_4^{CV}$
$\hat{f}_{-S_1}(x_{11}) - R_5^{CV}$	$\hat{f}_{-S_1}(x_{11}) + R_5^{CV}$
$\hat{f}_{-S_2}(x_{11}) - R_6^{CV}$	$\hat{f}_{-S_2}(x_{11}) + R_6^{CV}$
$\hat{f}_{-S_2}(x_{11}) - R_7^{CV}$	$\hat{f}_{-S_2}(x_{11}) + R_7^{CV}$
$\hat{f}_{-S_2}(x_{11}) - R_8^{CV}$	$\hat{f}_{-S_2}(x_{11}) + R_8^{CV}$
$\hat{f}_{-S_2}(x_{11}) - R_9^{CV}$	$\hat{f}_{-S_2}(x_{11}) + R_9^{CV}$
$\hat{f}_{-S_2}(x_{11}) - R_{10}^{CV}$	$\hat{f}_{-S_2}(x_{11}) + R_{10}^{CV}$
α quantile of these values is the low bound of the PI	$1 - \alpha$ quantile of these values is the high bound of the PI

Locally Weighted CV+ Example

Return

Fold 2

y	x
y6	x6
y7	x7
y8	x8
y9	x9
y10	x10

$\hat{f}_{-S_2}$

Residuals

R_i
$R_6 = \lvert y_6 - \hat{f}_{-S_1}(x_6)\rvert$
$R_7 = \lvert y_7 - \hat{f}_{-S_1}(x_7)\rvert$
$R_8 = \lvert y_8 - \hat{f}_{-S_1}(x_8)\rvert$
$R_9 = \lvert y_9 - \hat{f}_{-S_1}(x_9)\rvert$
$R_{10} = \lvert y_{10} - \hat{f}_{-S_1}(x_{10})\rvert$

$\hat{\rho}_{-S_2}(x_i)$

R_i^{LW}
$R_6^{LW} = \lvert y_6 - \hat{f}_{-S_2}(x_6)\rvert / \hat{\rho}_{-S_2}(x_6)$
$R_7^{LW} = \lvert y_7 - \hat{f}_{-S_2}(x_7)\rvert / \hat{\rho}_{-S_2}(x_7)$
$R_8^{LW} = \lvert y_8 - \hat{f}_{-S_2}(x_8)\rvert / \hat{\rho}_{-S_2}(x_8)$
$R_9^{LW} = \lvert y_9 - \hat{f}_{-S_2}(x_9)\rvert / \hat{\rho}_{-S_2}(x_9)$
$R_{10}^{LW} = \lvert y_{10} - \hat{f}_{-S_2}(x_{10})\rvert / \hat{\rho}_{-S_2}(x_{10})$

LW CV+
Predictions

Low	High
$\hat{f}_{-S_1}(x_{11}) - R_1^{LW} * \hat{\rho}_{-S_1}(x_{11})$	$\hat{f}_{-S_1}(x_{11}) + R_1^{LW} * \hat{\rho}_{-S_1}(x_{11})$
$\hat{f}_{-S_1}(x_{11}) - R_2^{LW} * \hat{\rho}_{-S_1}(x_{11})$	$\hat{f}_{-S_1}(x_{11}) + R_2^{LW} * \hat{\rho}_{-S_1}(x_{11})$
$\hat{f}_{-S_1}(x_{11}) - R_3^{LW} * \hat{\rho}_{-S_1}(x_{11})$	$\hat{f}_{-S_1}(x_{11}) + R_3^{LW} * \hat{\rho}_{-S_1}(x_{11})$
$\hat{f}_{-S_1}(x_{11}) - R_4^{LW} * \hat{\rho}_{-S_1}(x_{11})$	$\hat{f}_{-S_1}(x_{11}) + R_4^{LW} * \hat{\rho}_{-S_1}(x_{11})$
$\hat{f}_{-S_1}(x_{11}) - R_5^{LW} * \hat{\rho}_{-S_1}(x_{11})$	$\hat{f}_{-S_1}(x_{11}) + R_5^{LW} * \hat{\rho}_{-S_1}(x_{11})$
$\hat{f}_{-S_2}(x_{11}) - R_6^{LW} * \hat{\rho}_{-S_2}(x_{11})$	$\hat{f}_{-S_2}(x_{11}) + R_6^{LW} * \hat{\rho}_{-S_2}(x_{11})$
$\hat{f}_{-S_2}(x_{11}) - R_7^{LW} * \hat{\rho}_{-S_2}(x_{11})$	$\hat{f}_{-S_2}(x_{11}) + R_7^{LW} * \hat{\rho}_{-S_2}(x_{11})$
$\hat{f}_{-S_2}(x_{11}) - R_8^{LW} * \hat{\rho}_{-S_2}(x_{11})$	$\hat{f}_{-S_2}(x_{11}) + R_8^{LW} * \hat{\rho}_{-S_2}(x_{11})$
$\hat{f}_{-S_2}(x_{11}) - R_9^{LW} * \hat{\rho}_{-S_2}(x_{11})$	$\hat{f}_{-S_2}(x_{11}) + R_9^{LW} * \hat{\rho}_{-S_2}(x_{11})$
$\hat{f}_{-S_2}(x_{11}) - R_{10}^{LW} * \hat{\rho}_{-S_2}(x_{11})$	$\hat{f}_{-S_2}(x_{11}) + R_{10}^{LW} * \hat{\rho}_{-S_2}(x_{11})$
α quantile of these values is the low bound of the PI	$1 - \alpha$ quantile of these values is the high bound of the PI