Lecture 8:

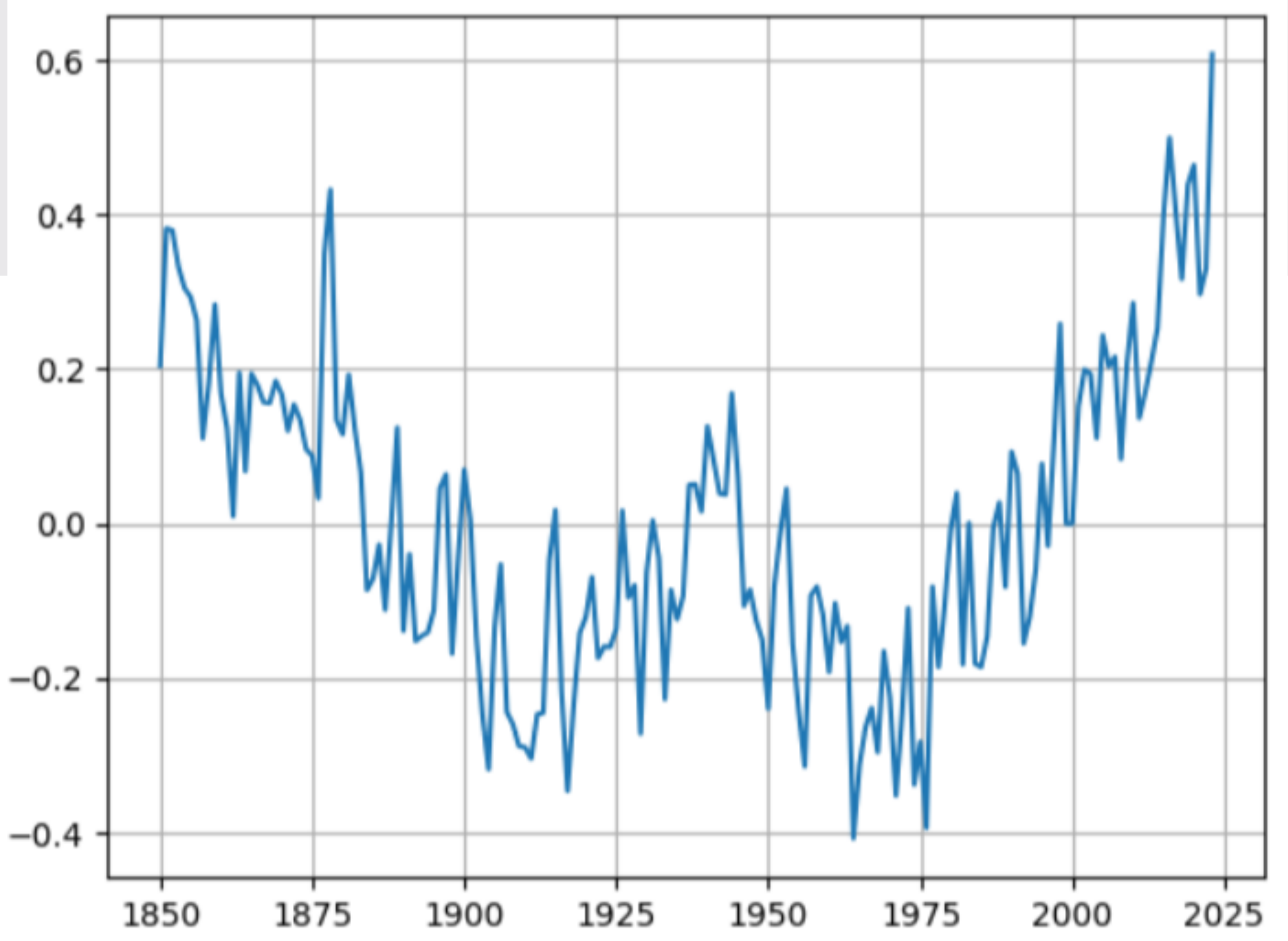
Model Selection

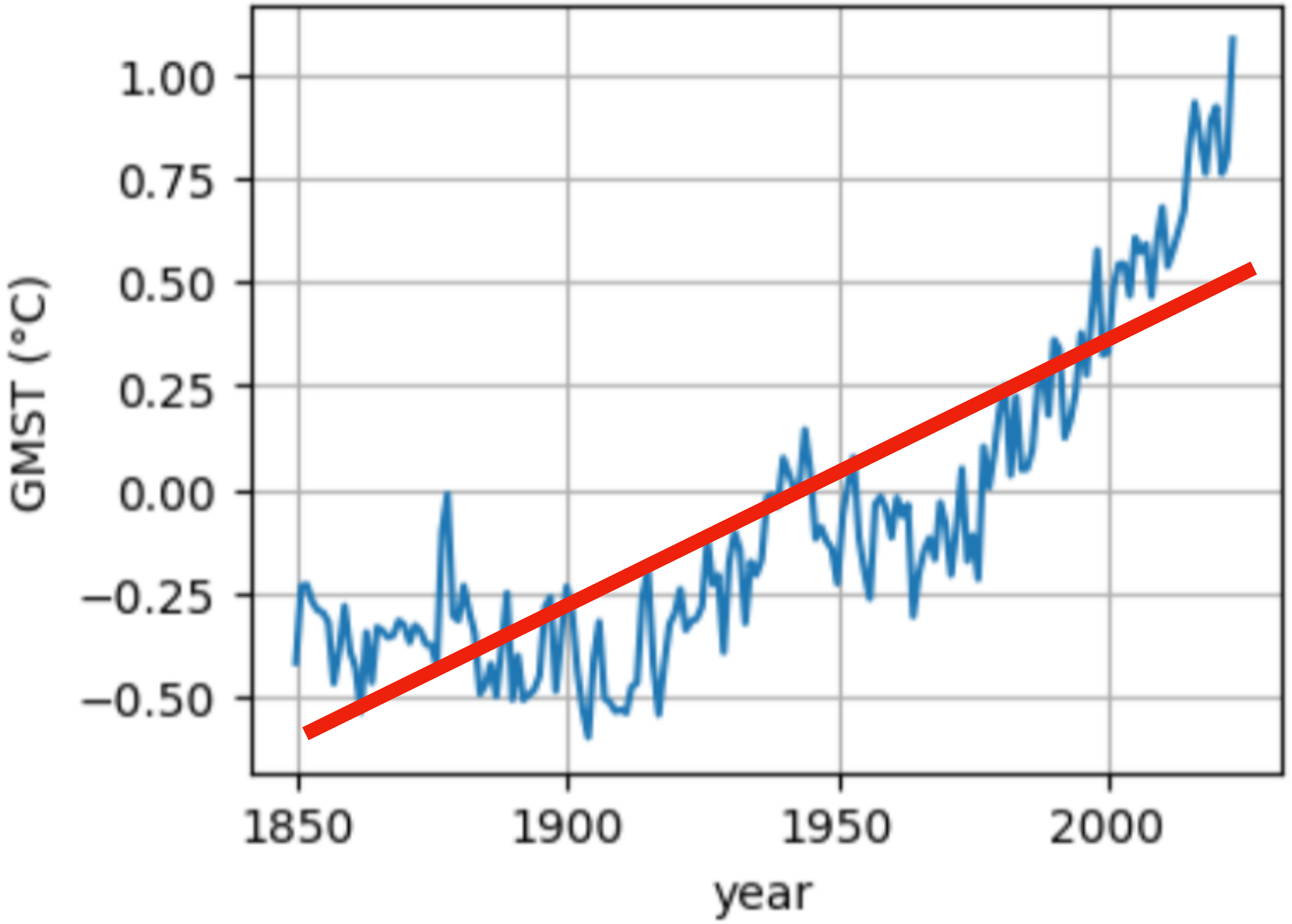
Road Map of the Statistics Part

	Lecture 5	Lecture 6	Lecture 7
Quantification Technique	Mean, variance, skewness, & kurtosis	Pearson's Correlation (Linear relationship)	Linear regression (OLS)
Uncertainty & Significance	Gaussian distribution Chi-2 distribution	r, p = scipy.stats. pearsonr(x, y)	results.summary( )
Assumptions	Data is Gaussian or follows specific types of distribution Independent Sampling	Data is Gaussian Independent Sampling	x is noise free Error is Gaussian Independent Sampling Equal err variance
Test assumptions	K-S test		Auto-correlation (Effective Sample Size)
Treatment		Bootstrapping	Block Bootstrapping

What will be covered in this lecture?

1. Fitting a quadratic function
2. Fitting a polynomial
3. What is a good model?
    - 3.1 Training, Validation, and Testing sets
    - 3.2 Cross Validation
    - 3.3 Bayesian Information Criteria
4. Interpolation vs. Extrapolation

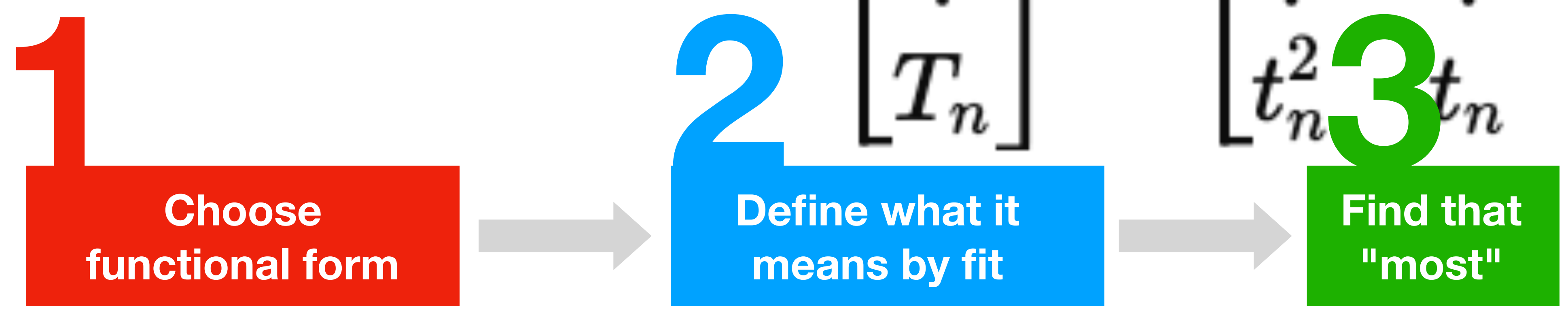


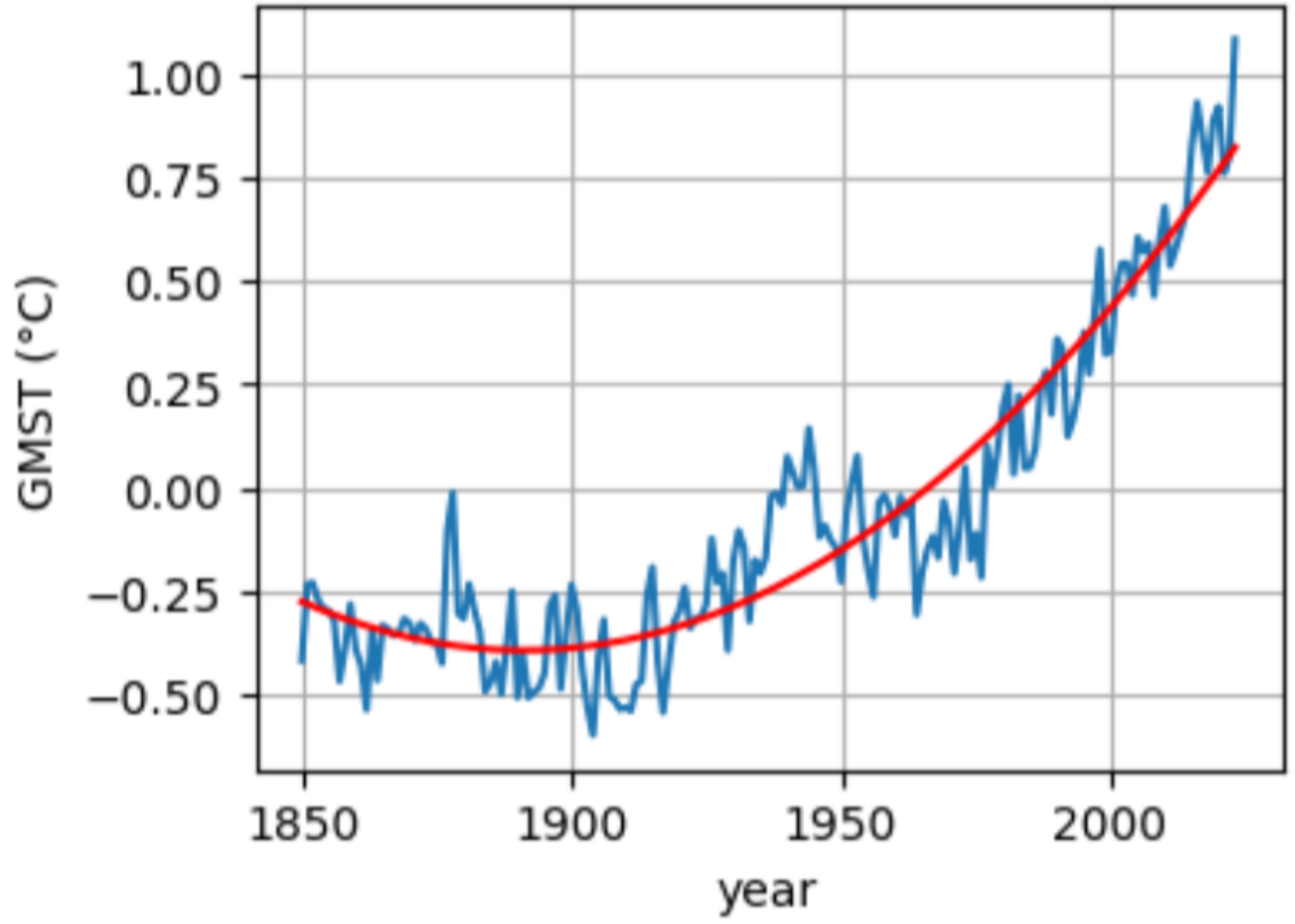


Find other functional forms: a Quadratic Function

$$T = \alpha_2 t^2 + \alpha_1 t + \alpha_0$$

$$T = \mathbf{X}\boldsymbol{\alpha} \qquad \begin{bmatrix} T_1 \\ T_2 \\ \vdots \\ T_n \end{bmatrix} = \begin{bmatrix} t_1^2 & t_1 & 1 \\ t_2^2 & t_2 & 1 \\ \vdots & \vdots & \vdots \\ t_n^2 & t_n & 1 \end{bmatrix} \begin{bmatrix} \alpha_2 \\ \alpha_1 \\ \alpha_0 \end{bmatrix}$$





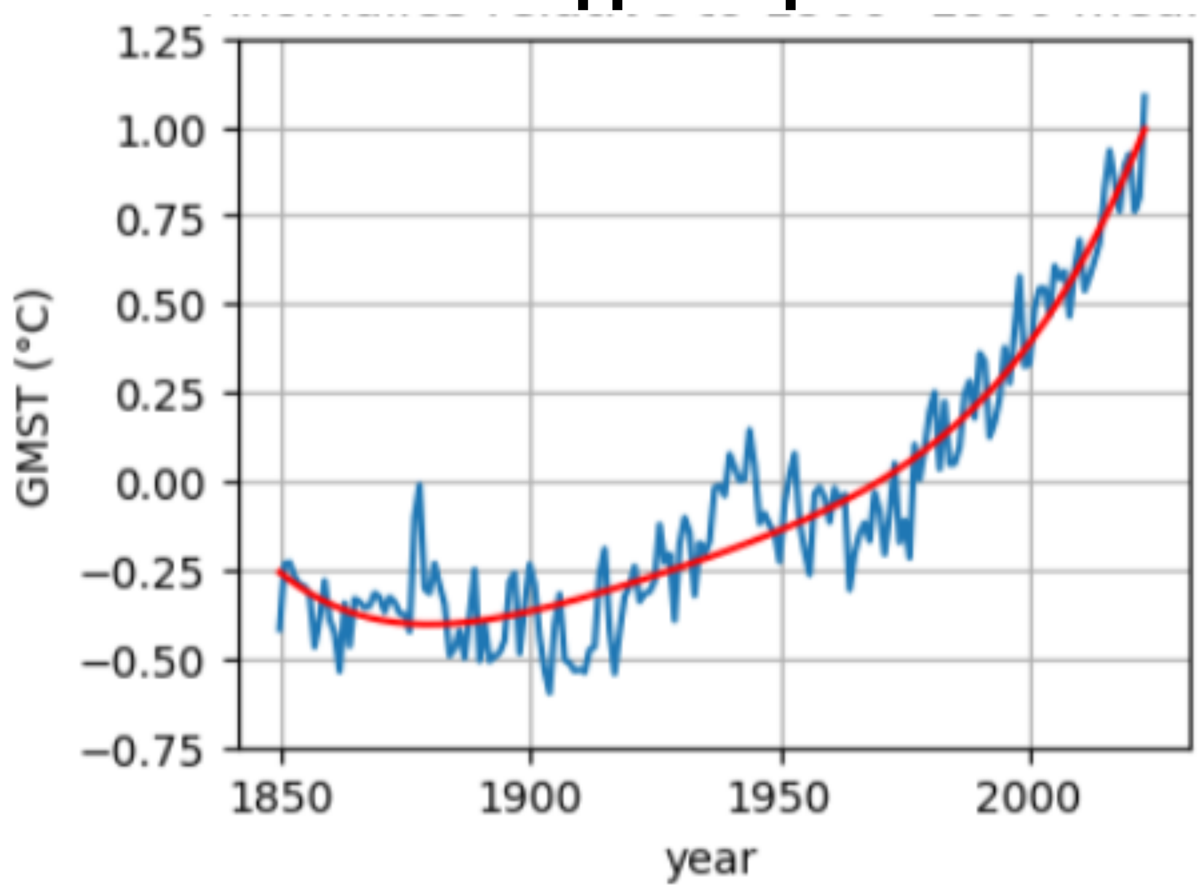

```
import statsmodels.api as sm

t = years - np.mean(years)

X = np.column_stack((t**2, t, np.ones(len(t))))

model = sm.OLS(GMST, X)

results = model.fit()

GMST_hat = results.fittedvalues
```

Find other functional forms: Polynomial

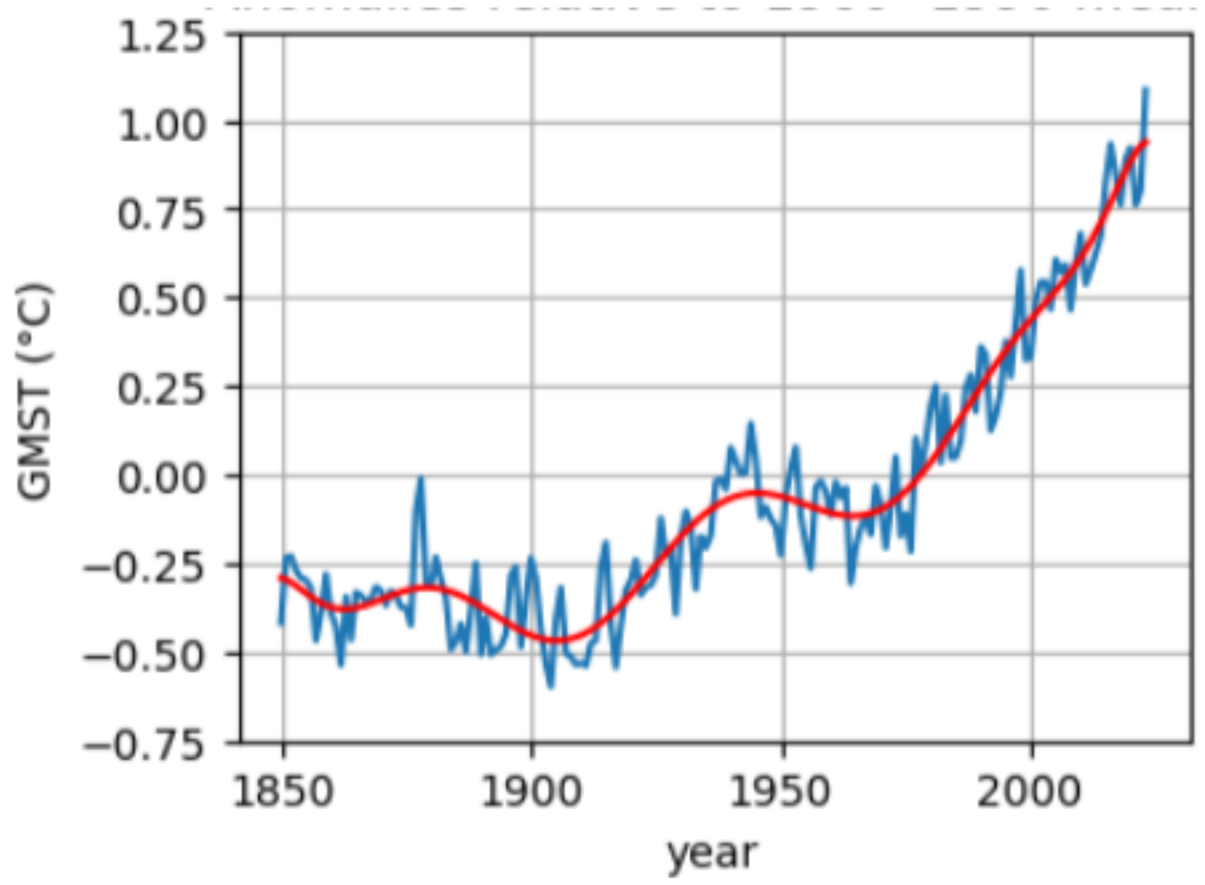
$$\mathrm{T} = \sum_{i=0}^{n} \alpha_i t^i$$

```
import statsmodels.api as sm
t = years - np.mean(years)
years_matrix = np.column_stack((t**N, ..., t**2, t, np.ones(len(t))))
model = sm.OLS(GMST, years_matrix)
results = model.fit()
GMST_hat = results.fittedvalues
```

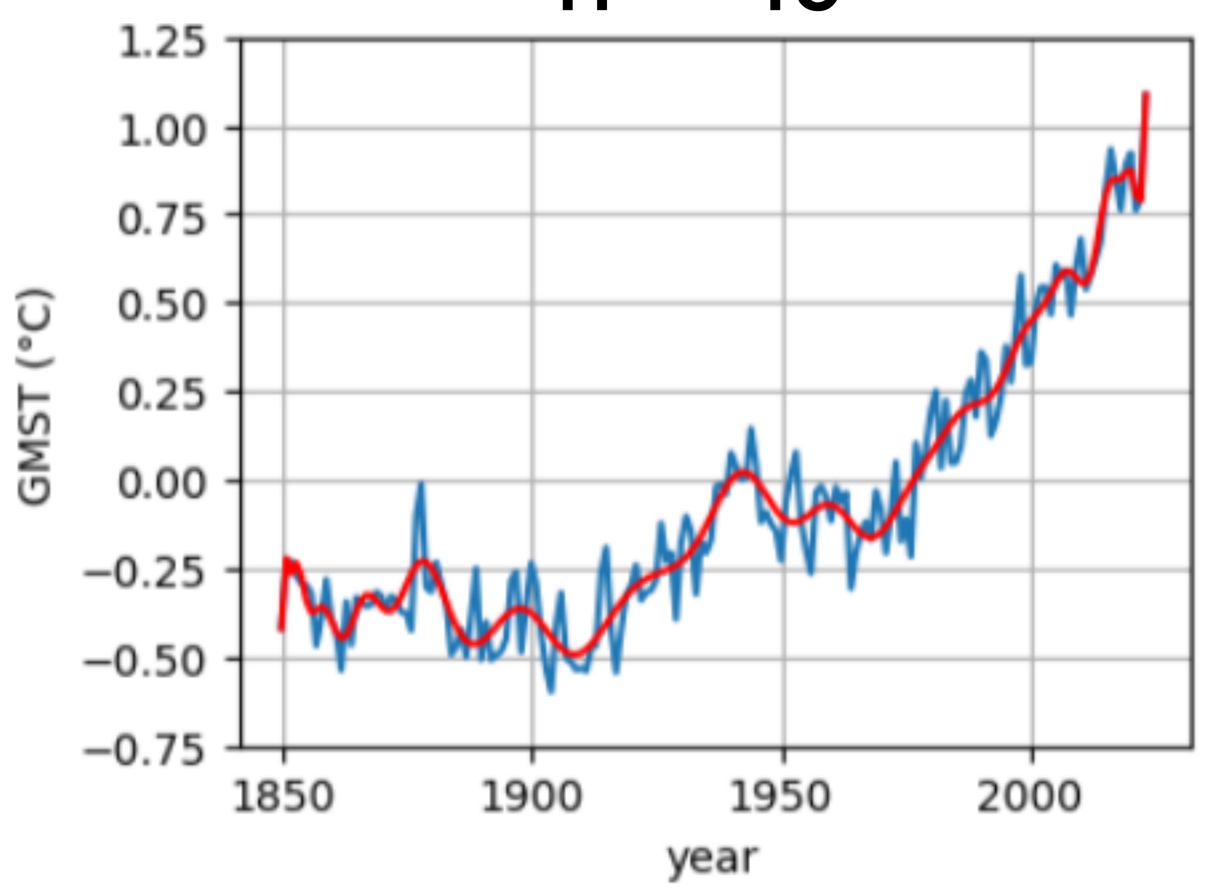
n = 4



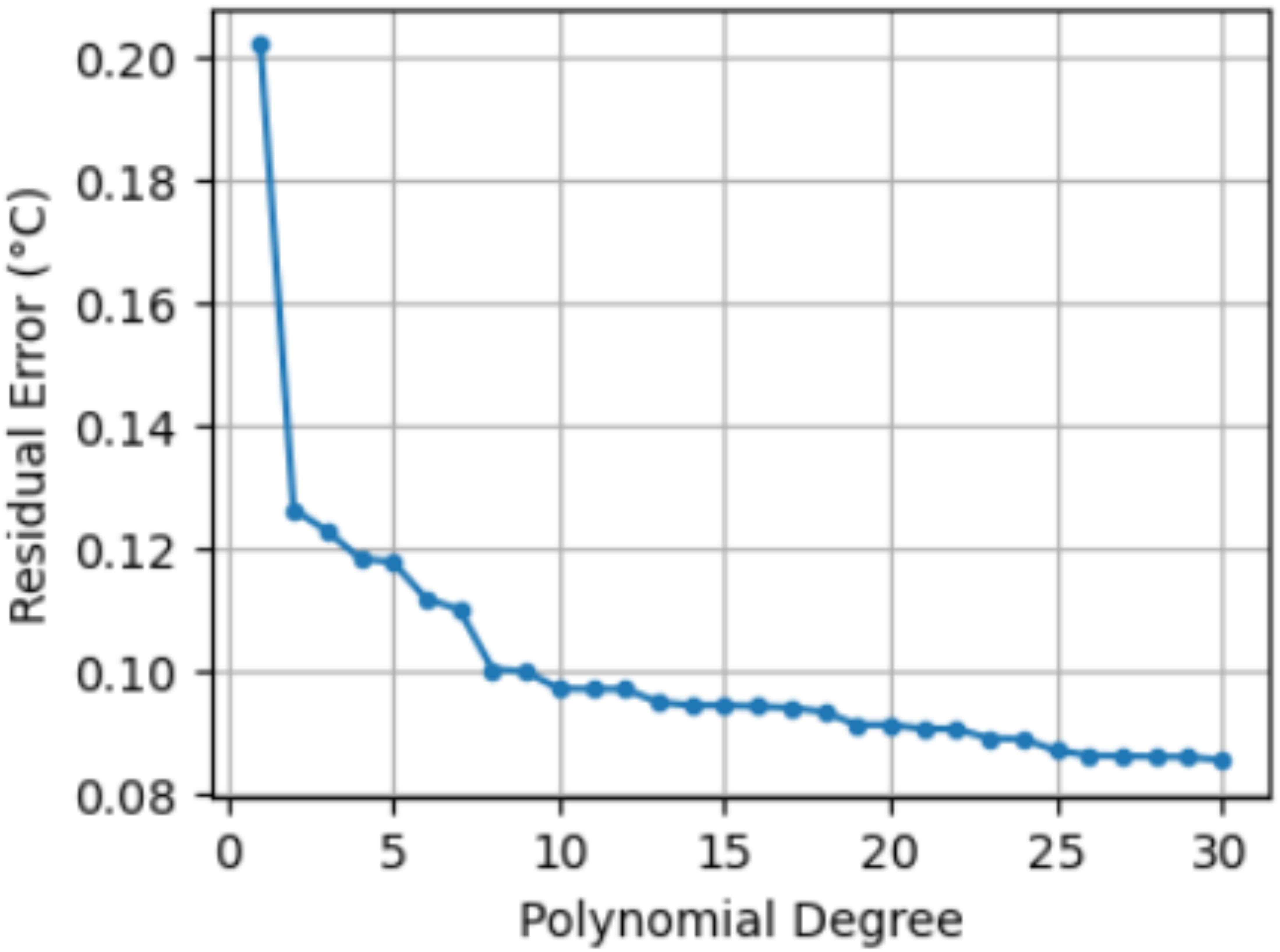
n = 10



n = 40



Residual Error Decreases with the Degree of Polynomial



But does a lower error suggests a

better model?

The purpose of building models: Make Predictions

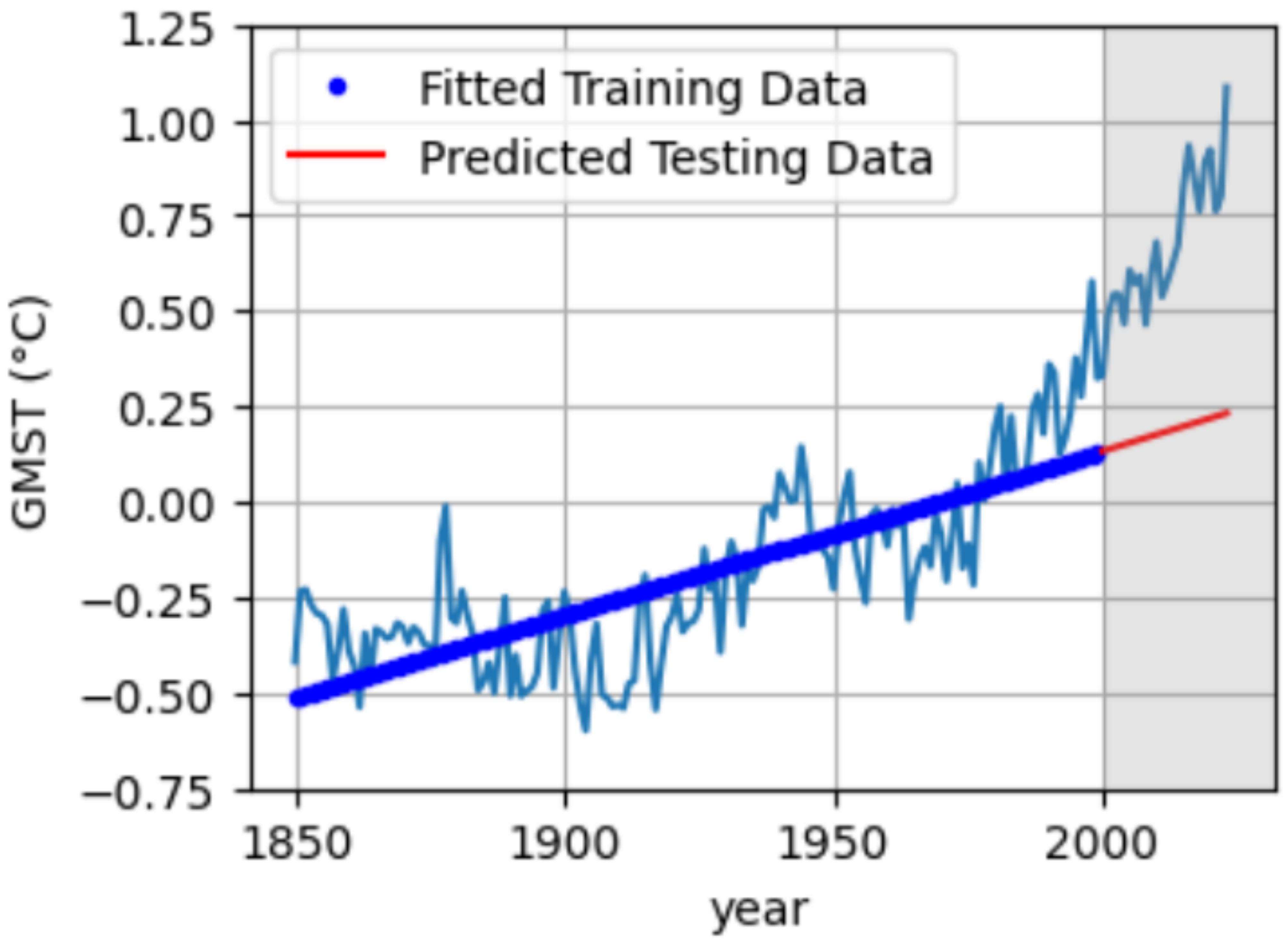
A good model is one that can

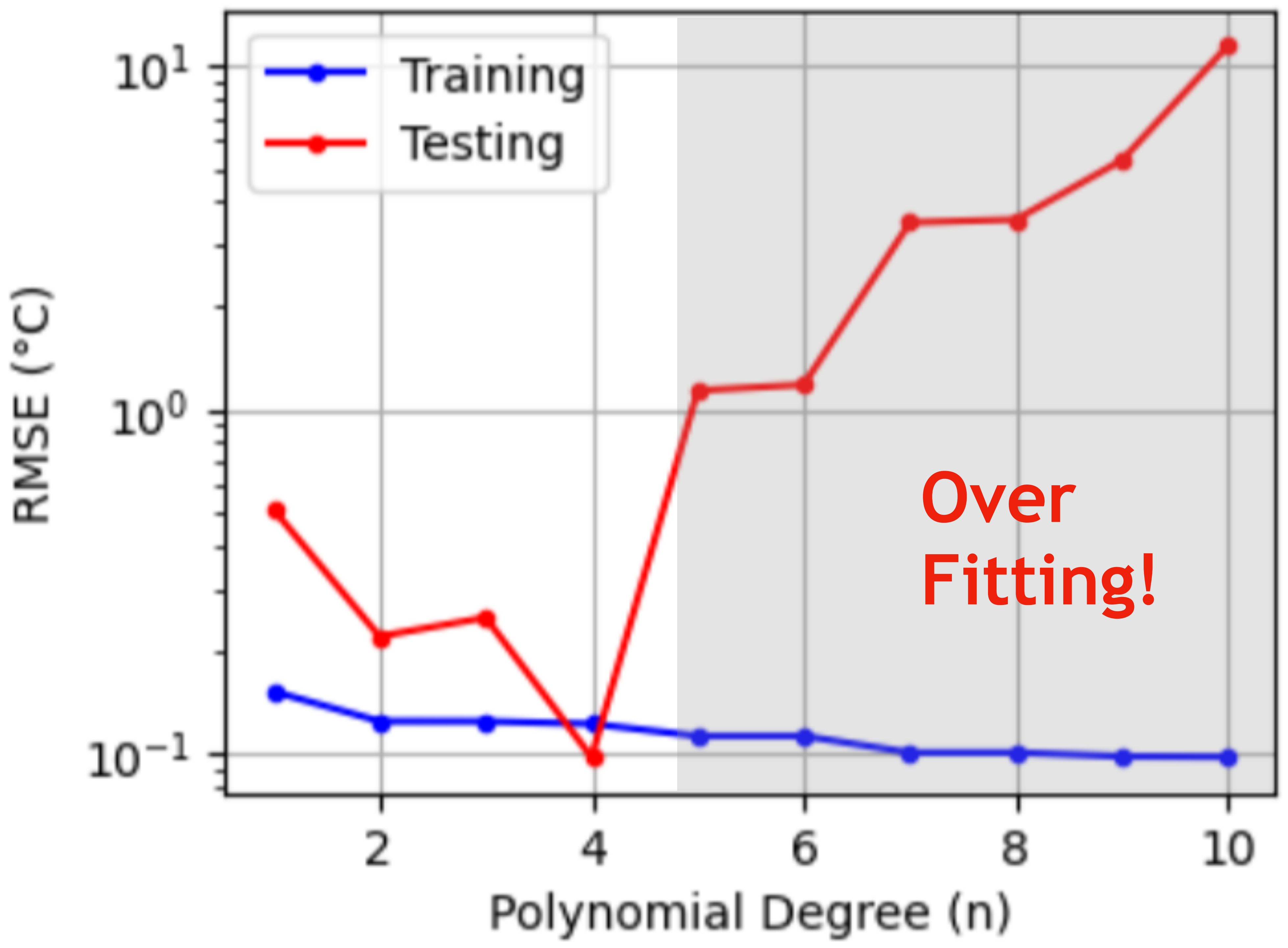
better generalise and predict unseen data.

Training Error vs. Prediction Error

Let's leave part of the data (say after 2000) out for evaluating prediction error,

and fit the model on remaining data.





Over Fitting: The regression learns noise rather than actual function form.

Account for overfitting: Training, Validation, and Testing Set

For **Large Dataset**

Fit the model

Evaluate prediction error to select the best model

Evaluate prediction error of the selected model

Account for overfitting: Cross Validation

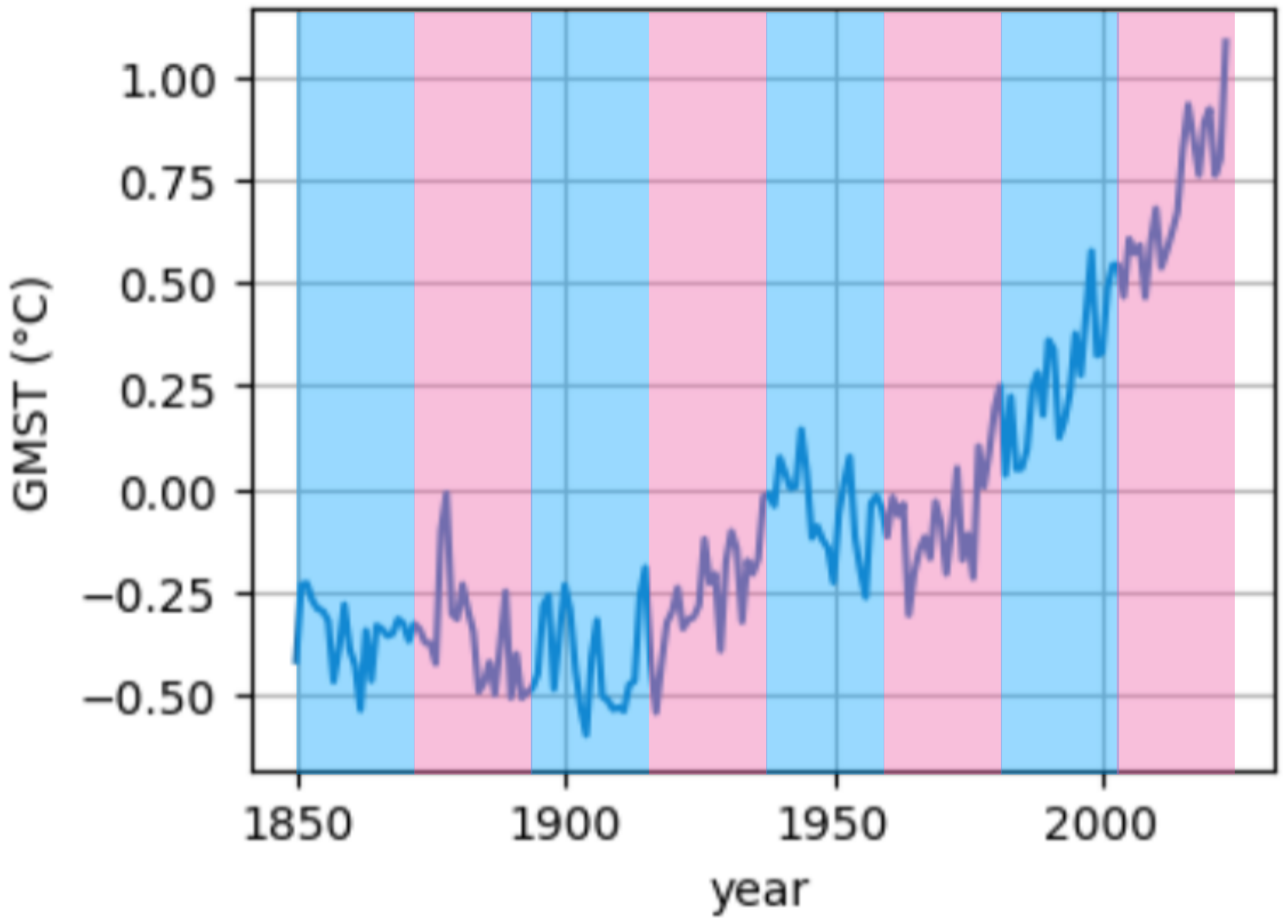
For **Small Data** but **sufficient computational resource**

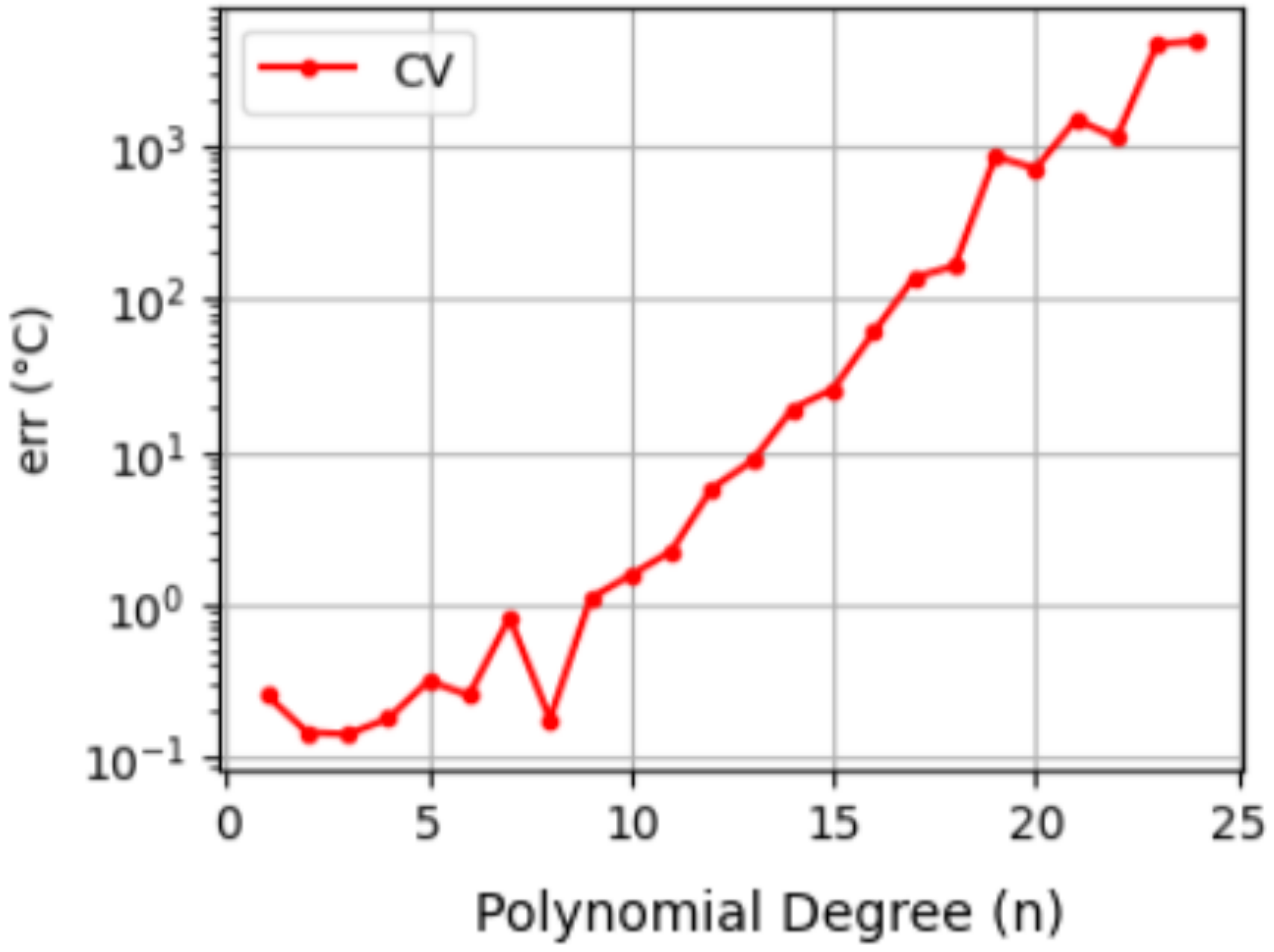
(1) Break datasets into several chunks,

(2) Leave each chunk out at a time to evaluate prediction error,

(3) Loop over all chunks and pull error estimates together
to get an averaged view of prediction error.

(4) Loop over all possible models and
select the model with the lowest prediction error.






```
N_blocks = ...

for ct_m in all possible models:

    for ct in np.arange(N_blocks):

        Fit model on remaining blocks

        Predict on the target block

        Save prediction error in an array

    Average prediction error over all blocks

Find the model with the lowest prediction error
```

Account for overfitting: Bayesian Information Criterion (BIC)

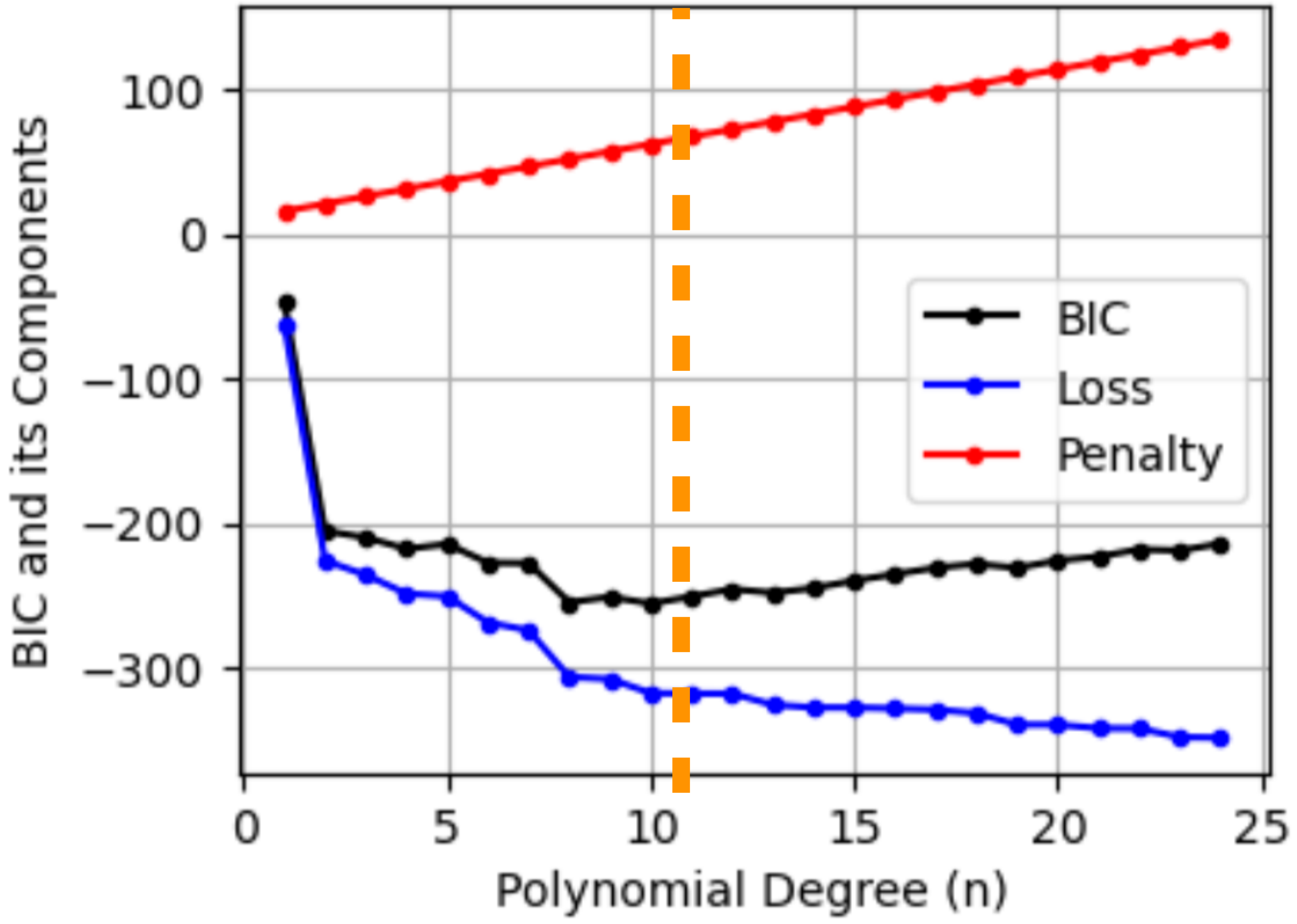
For Insufficient Computational Resource

Since complex models tend to generalise worse, we can penalise complex models by adding a penalty term to the loss function, such that these models are less preferred.

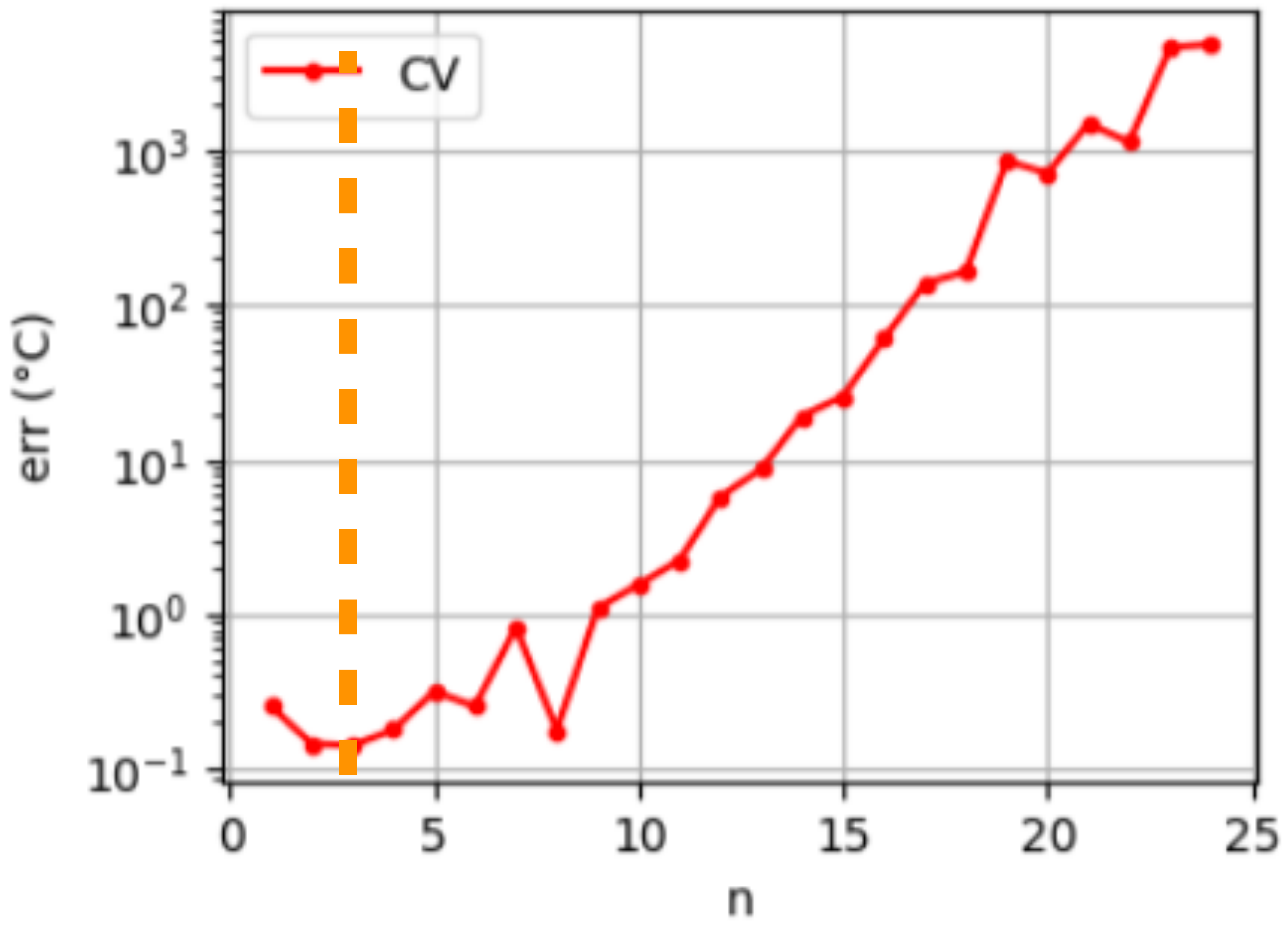
```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.714
Model:                            OLS   Adj. R-squared:                  0.712
Method:                 Least Squares   F-statistic:                     428.7
Date:                Mon, 05 Feb 2024   Prob (F-statistic):           1.40e-48
Time:                        11:34:14   Log-Likelihood:                 31.252
No. Observations:                 174   AIC:                            -58.50
Df Residuals:                     172   BIC:                            -52.19
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const         -0.0722      0.015     -4.686      0.000      -0.103      -0.042
x1             0.0064      0.000     20.706      0.000       0.006       0.007
==============================================================================
Omnibus:                        4.837   Durbin-Watson:                   0.335
Prob(Omnibus):                  0.089   Jarque-Bera (JB):                4.856
Skew:                           0.376   Prob(JB):                       0.0882
Kurtosis:                       2.679   Cond. No.                         50.2
==============================================================================
```

BIC vs. Cross Validation

BIC

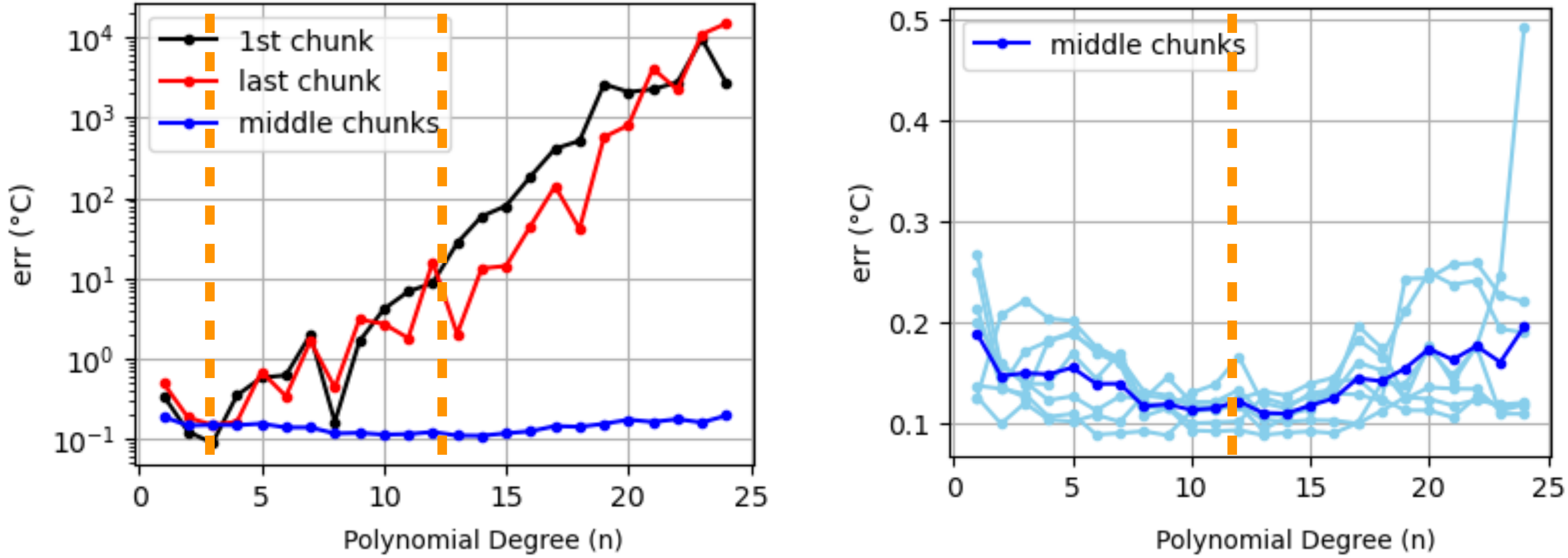


Cross Validation



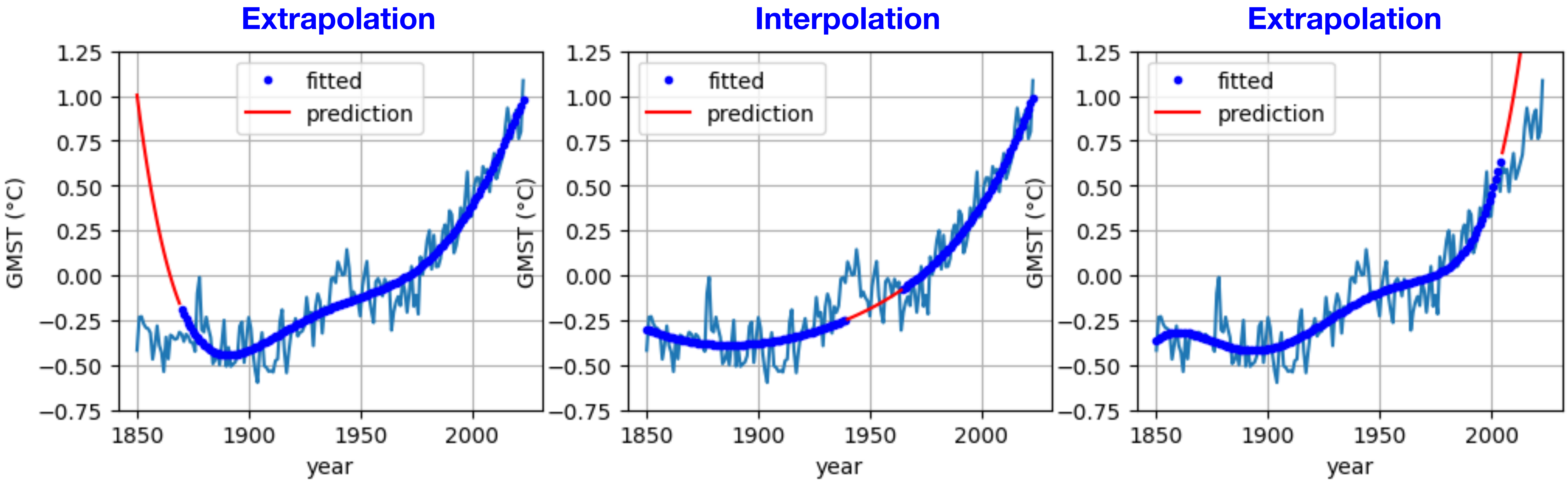
The two results do not appear to be consistent with one another...

Breaking down Cross-Validation Results



If we focus on predicting middle chunks, CV and BIC appears to be consistent.

Interpolation vs. Extrapolation



The underlying process or data distribution is **non-stationary** (the climate is changing)
Values of predictor variables associated with these chunks are not bounded by training data.

Be specifically careful when applying model to data distribution not trained upon!!!

Road Map of the Statistics Part

	Lecture 5	Lecture 6	Lecture 7	Lecture 8
Quantification Technique	Mean, variance, skewness, & kurtosis	Pearson's Correlation (Linear relationship)	Linear regression (OLS)	Model Selection
Uncertainty & Significance	Gaussian distribution Chi-2 distribution	r, p = scipy.stats. pearsonr(x, y)	results.summary( )	Training error vs. prediction error
Assumptions	Data is Gaussian or follows specific types of distribution Independent Sampling	Data is Gaussian Independent Sampling	x is noise free Error is Gaussian Independent Sampling Equal err variance	
Test assumptions	K-S test		Auto-correlation (Effective Sample Size)	
Treatment		Bootstrapping	Block Bootstrapping