

回归

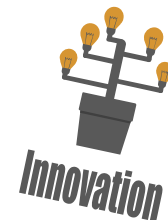
Regression

郝 奇

南京大学 天文与空间科学学院

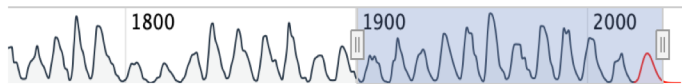
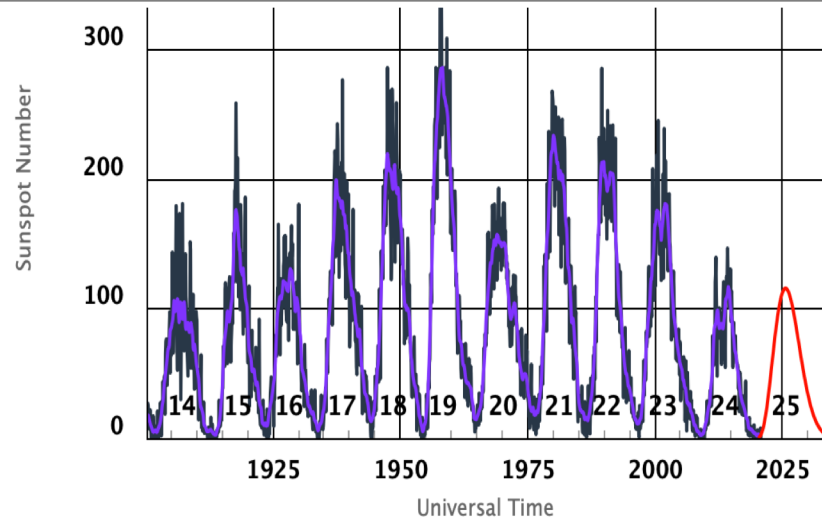
Application of Machine Learning in Astronomy

机器学习在天文中的应用



ISES Solar Cycle Sunspot Number Progression

Contents



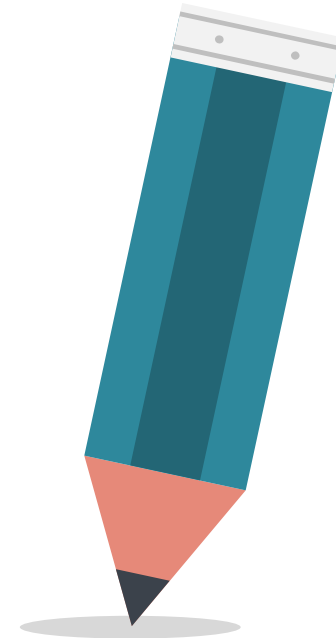
◆ Monthly Values — Smoothed Monthly Values — Predicted Values
Space Weather Prediction Center

01 What is Regression

02 Example Application

What is Regression

01



1. What is Regression

Regression is a **statistical method** used in finance, investing, and other disciplines that attempts to determine the strength and character of the relationship between one dependent variable (usually denoted by Y) and a series of other variables (known as independent variables).

Regression is a **machine learning algorithm** that can be trained to predict real numbered outputs; like temperature, stock price, etc. Regression is based on a hypothesis that can be linear, quadratic, polynomial, non-linear, etc.

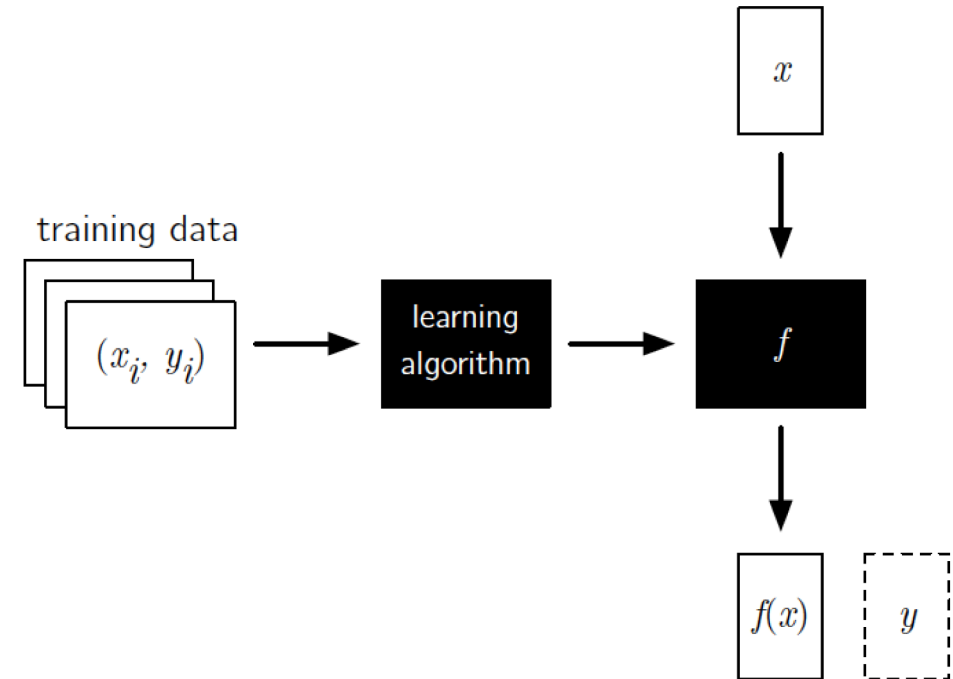
1. What is Regression

Last Lecture

Training set: $\mathcal{D} = \{\langle \mathbf{x}^{[i]}, \mathbf{y}^{[i]} \rangle, i = 1, \dots, n\}$,

Unknown function: $\mathbf{f}(\mathbf{x}) = \mathbf{y}$

Hypothesis: $\mathbf{h}(\mathbf{x}) = \hat{\mathbf{y}}$



Training: a learning algorithm takes a set of example input-output pairs, $\{(\mathbf{x}_1; \mathbf{y}_1); \dots (\mathbf{x}_n; \mathbf{y}_n)\}$, and returns a function $\mathbf{f}$ (the 'hypothesis'); the goal is for $\mathbf{f}(\mathbf{x})$ to recover the true label $\mathbf{y}$, for each example, and on future examples.

Testing: we check how well $\mathbf{f}$ predicts on a set of test examples, $\{(\mathbf{x}'_1; \mathbf{y}'_1); \dots (\mathbf{x}'_m; \mathbf{y}'_m)\}$, by measuring how well $\mathbf{f}(\mathbf{x}')$ matches $\mathbf{y}$.

1. What is Regression

Stock Market Forecast

$$f \left(\text{img} \right) = \text{Tomorrow average Stock indexes}$$

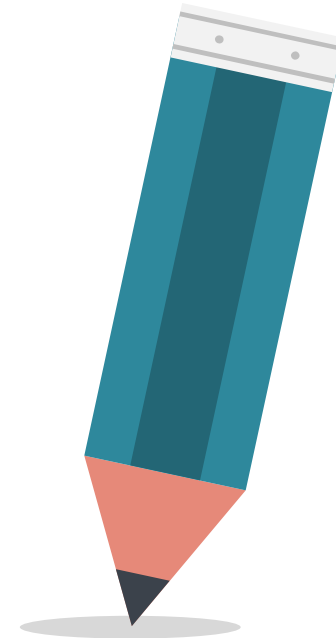

Output a scaler

Self-driving

$$f \left(\text{img} \right) = \text{Steering wheel angle}$$


Example Application

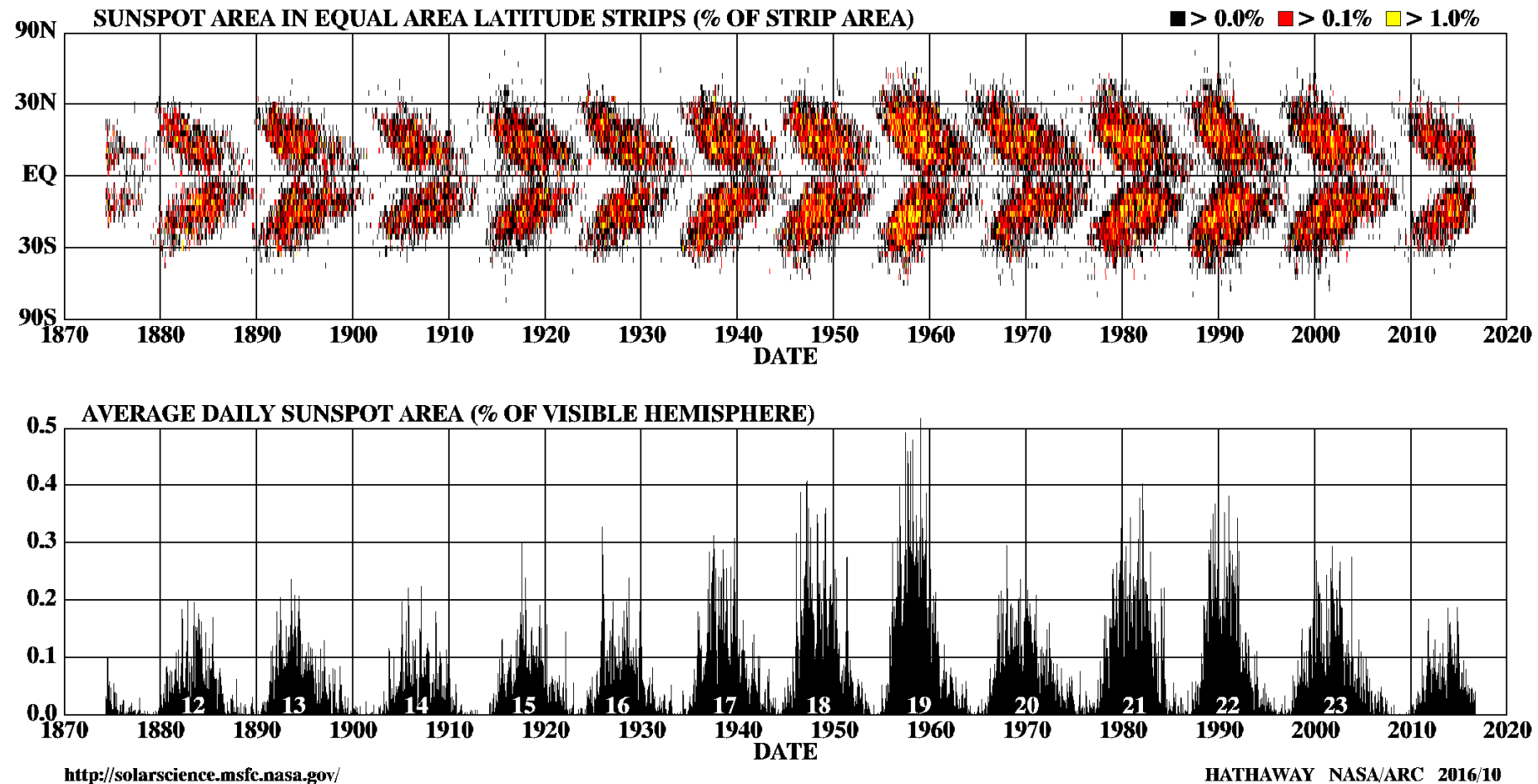
02



2. Example Application

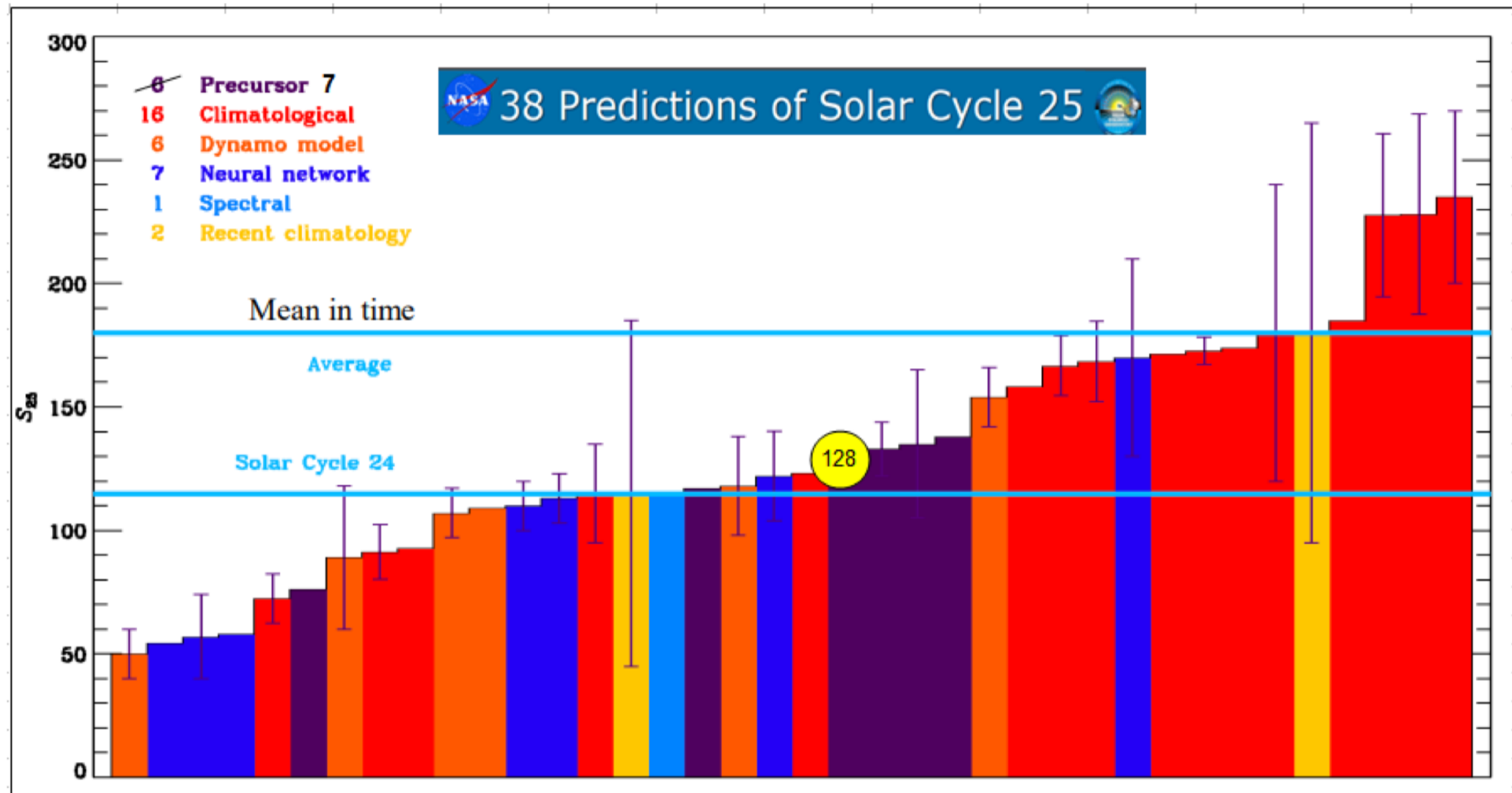
2.1 Solar Cycles

DAILY SUNSPOT AREA AVERAGED OVER INDIVIDUAL SOLAR ROTATIONS



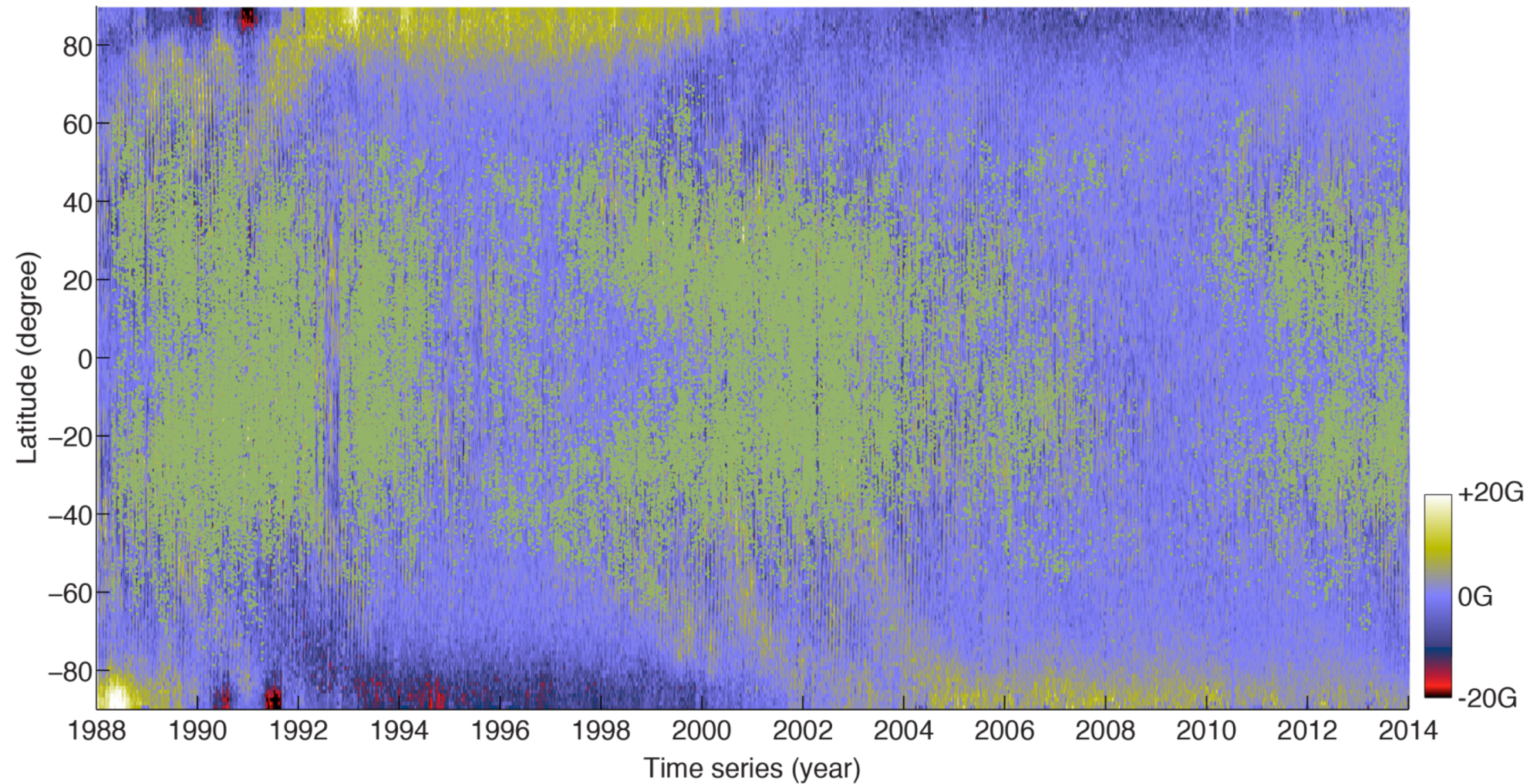
2. Example Application

2.1 Solar Cycles



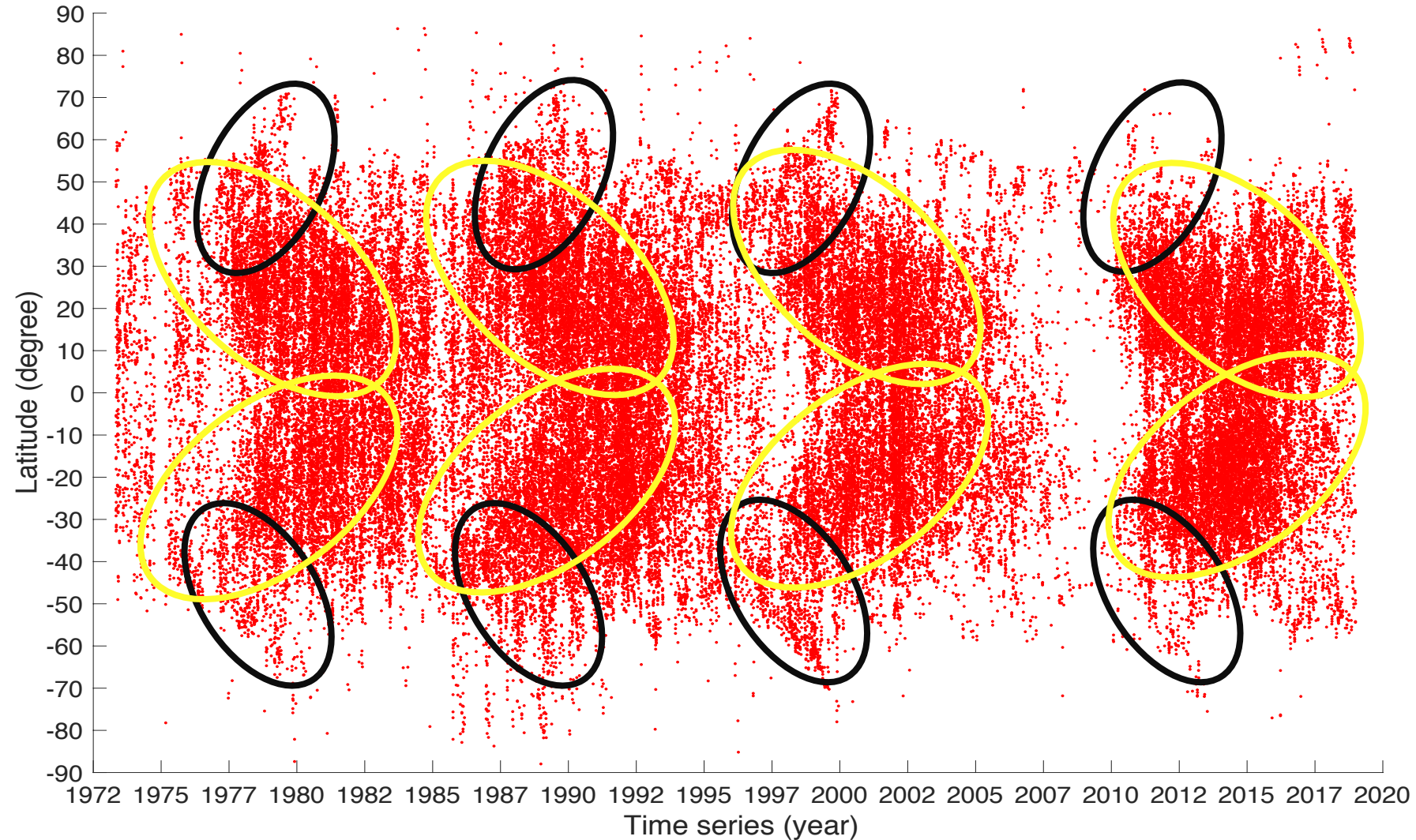
2. Example Application

2.2 Filament Butterfly Diagram



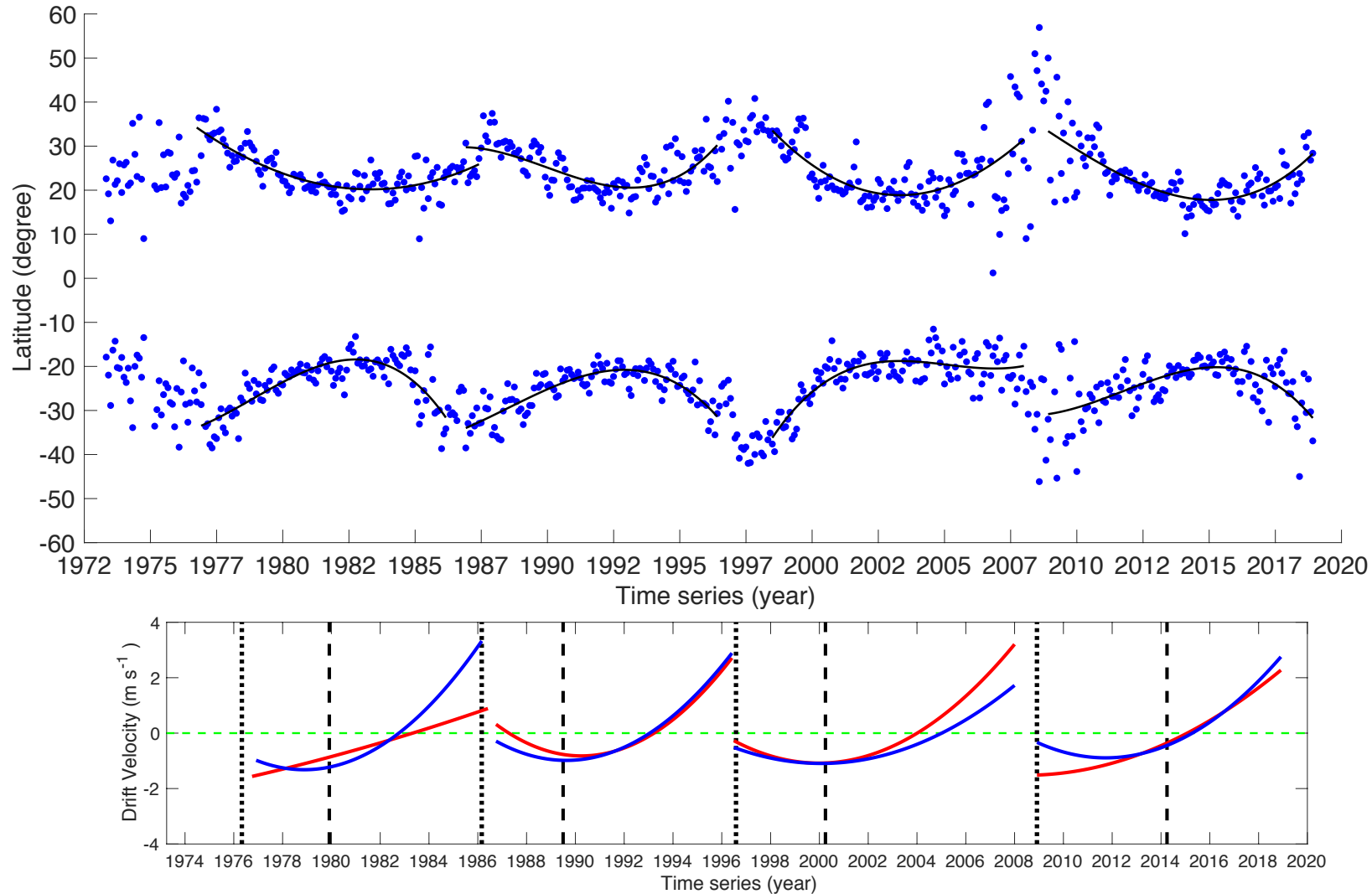
2. Example Application

2.2 Filament Butterfly Diagram



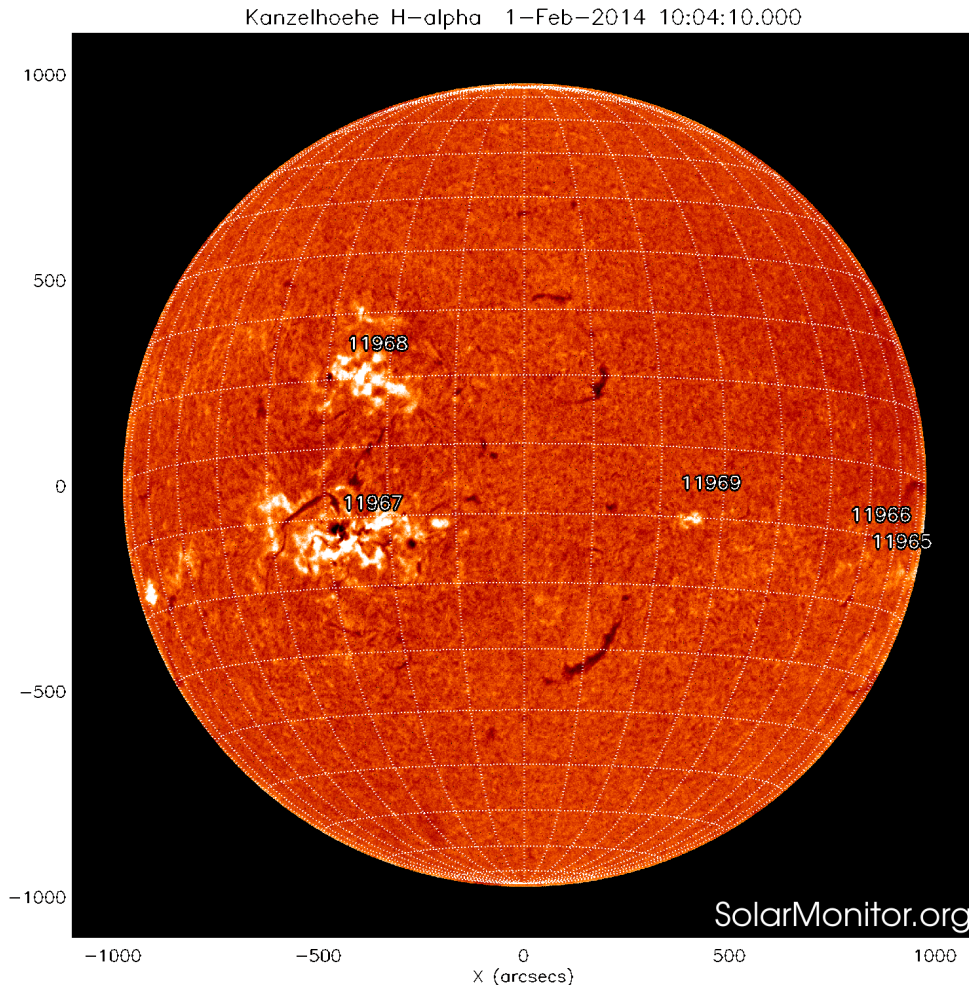
2. Example Application

2.3 Filament drift velocity



2. Example Application

2.4 Estimating the filament average location in a certain date



$f(\text{date}) = \text{filament location (latitude)}$

$$f(x) = y$$

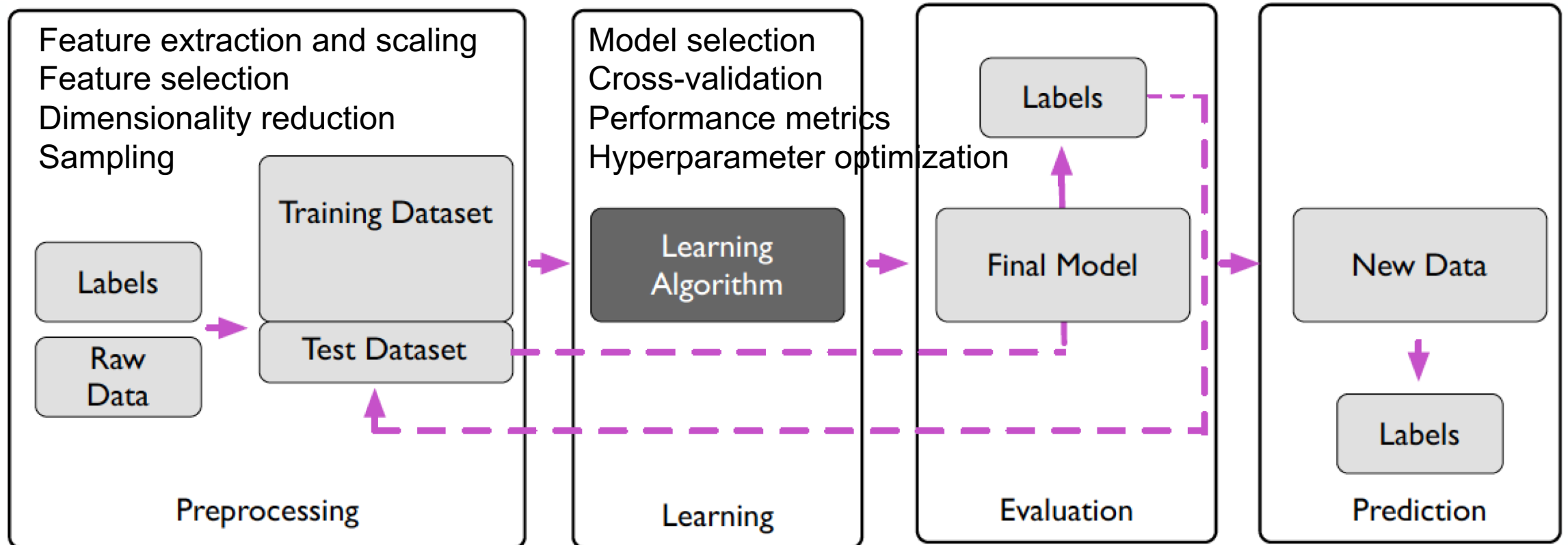
$$f(\text{"2014.02.01"}) = 30^\circ$$

2. Example Application

2.5 Training processing

Last Lecture

Supervised Learning Workflow -- More detailed overview



2. Example Application

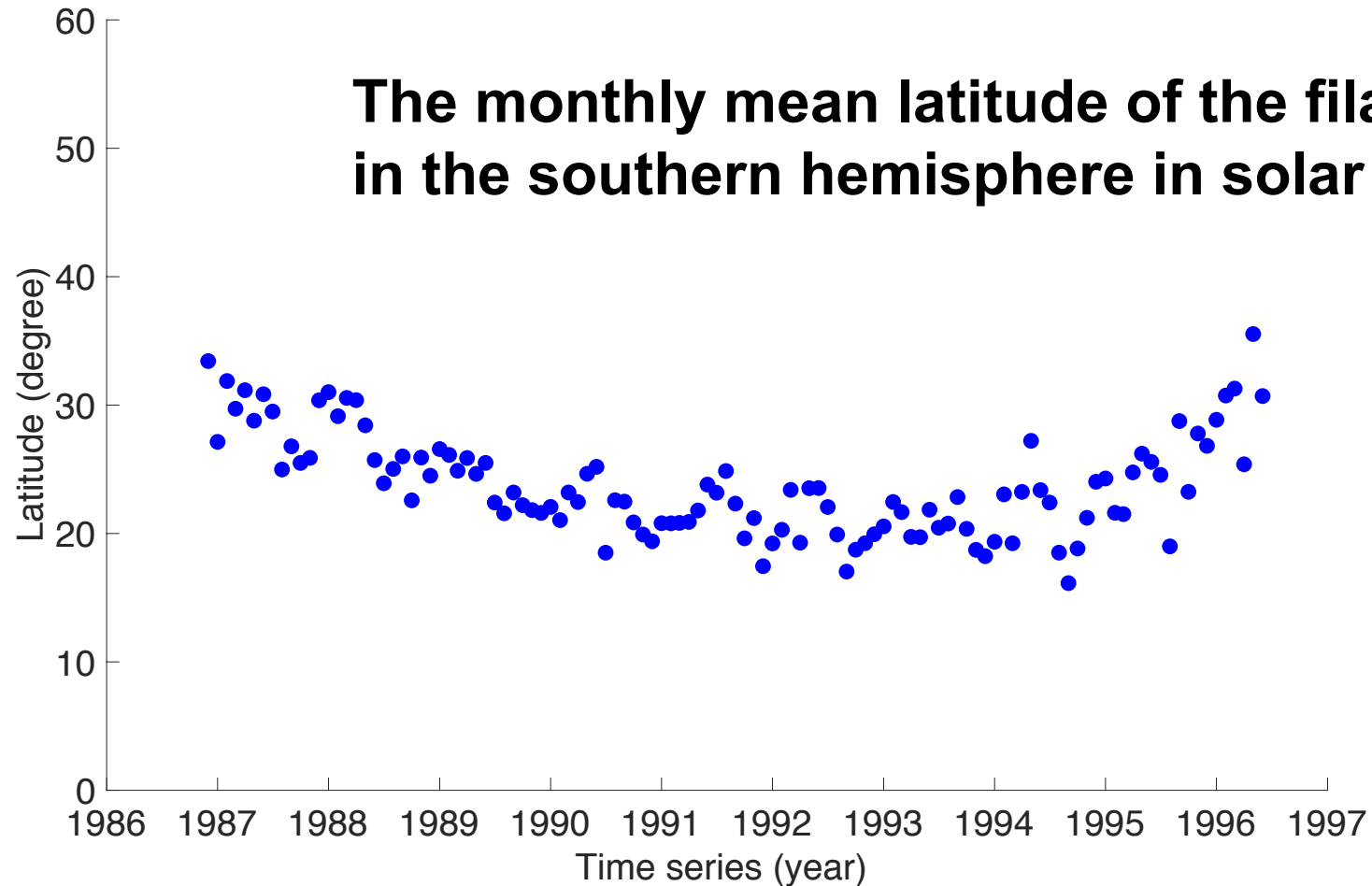
2.5 Training processing



2. Example Application

2.5 Training processing

Training Data



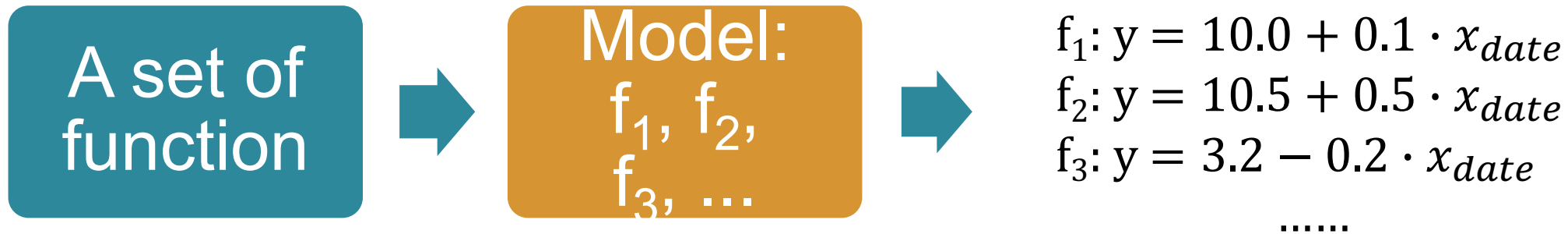
2. Example Application

2.5 Training processing

Step1 Model

$f(\text{date})$ = the filament average latitude

$$y = b + w \cdot x_{\text{date}}$$

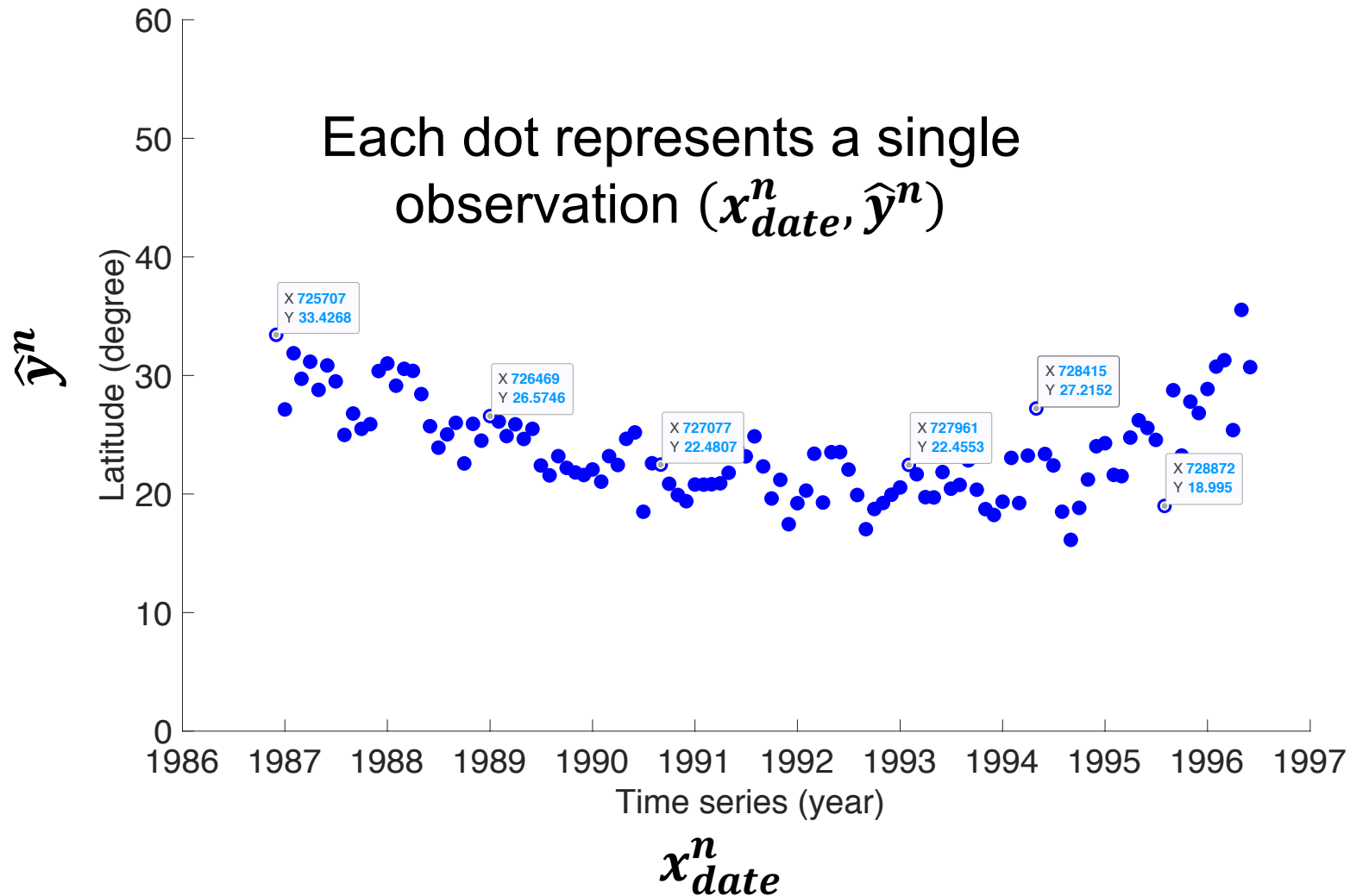


$$\text{Linear model: } y = b + \sum w_i x_i$$

where w_i is the **weight**, x_i is the feature and b is the **bias**.

2. Example Application

2.5 Training processing



Training Data

$$(x_{date}^1, \hat{y}^1)$$

$$(x_{date}^2, \hat{y}^2)$$

$$(x_{date}^3, \hat{y}^3)$$

⋮

⋮

$$(x_{date}^{115}, \hat{y}^{115})$$

2. Example Application

2.5 Training processing

Step2 Goodness of Function

Last Lecture

Loss function: Often used synonymously with cost function; sometimes also called error function. In some contexts the loss for a single data point, whereas the cost function refers to the overall (average or summed) loss over the entire dataset. Sometimes also called empirical risk.

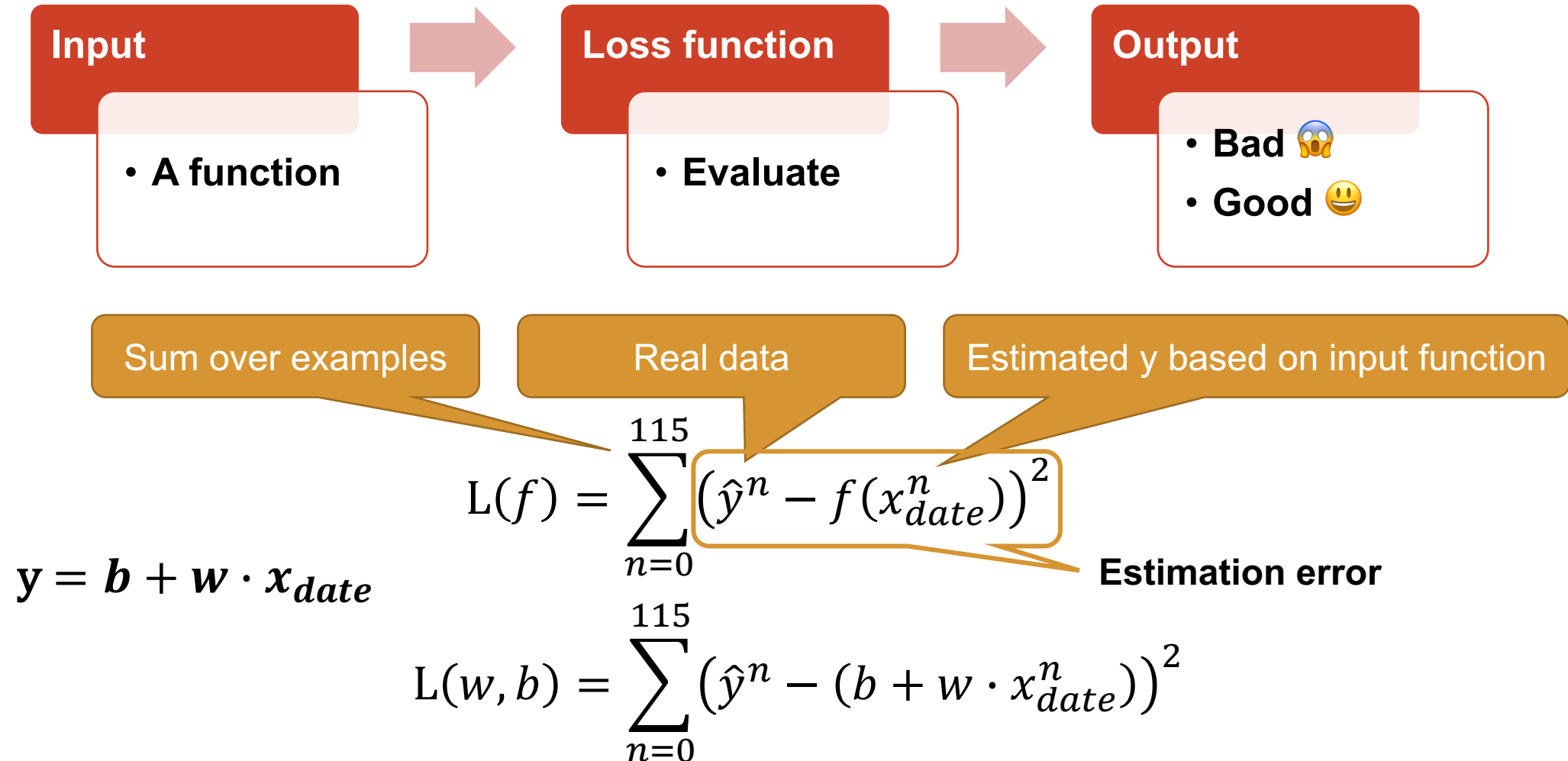
$$L(\hat{y}, y) = \begin{cases} 0 & \text{if } \hat{y} = y \\ 1 & \text{if } \hat{y} \neq y \end{cases}$$

$$ERR_{\mathcal{D}\text{test}} = \frac{1}{n} \sum_{i=1}^n L(\hat{y}^{[i]}, y^{[i]})$$

2. Example Application

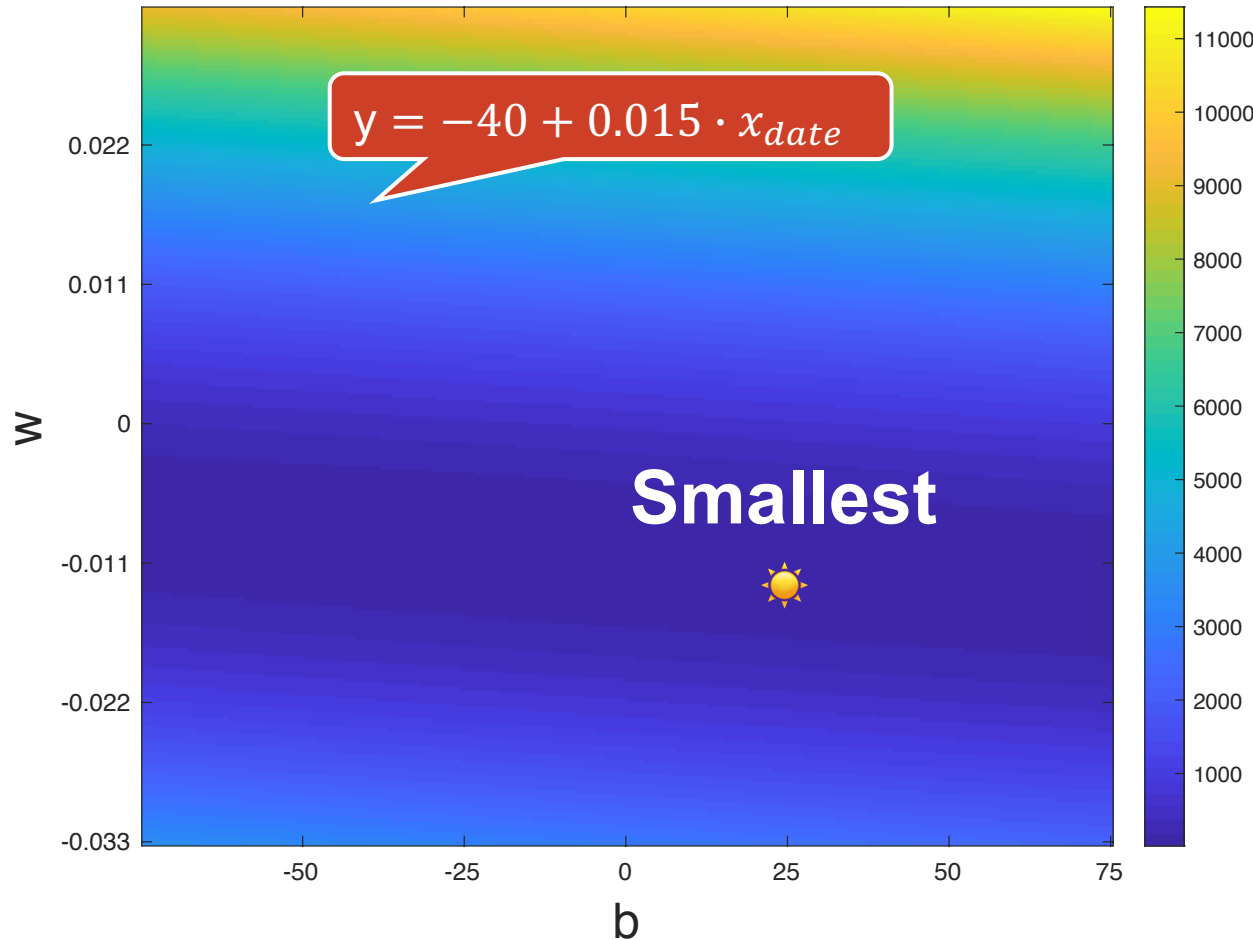
2.5 Training processing

Step2 Goodness of Function



2. Example Application

2.5 Training processing



Step2 Goodness of Function

$$L(w, b) = \sum_{n=0}^{115} (\hat{y}^n - (b + w \cdot x_{date}^n))^2$$

Each point in the figure represents a function;

The color represents $L(w, b)$;

2. Example Application

2.5 Training processing

Step3 Pick the best function

$$L(w, b) = \sum_{n=0}^{115} (\hat{y}^n - (b + w \cdot x_{date}^n))^2$$

How to pick the “BEST” function?

$$f^* = \arg \min_f L(f)$$

$$w^*, b^* = \arg \min_{w, b} L(w, b)$$

$$w^*, b^* = \arg \min_{w, b} \sum_{n=0}^{115} (\hat{y}^n - (b + w \cdot x_{date}^n))^2$$

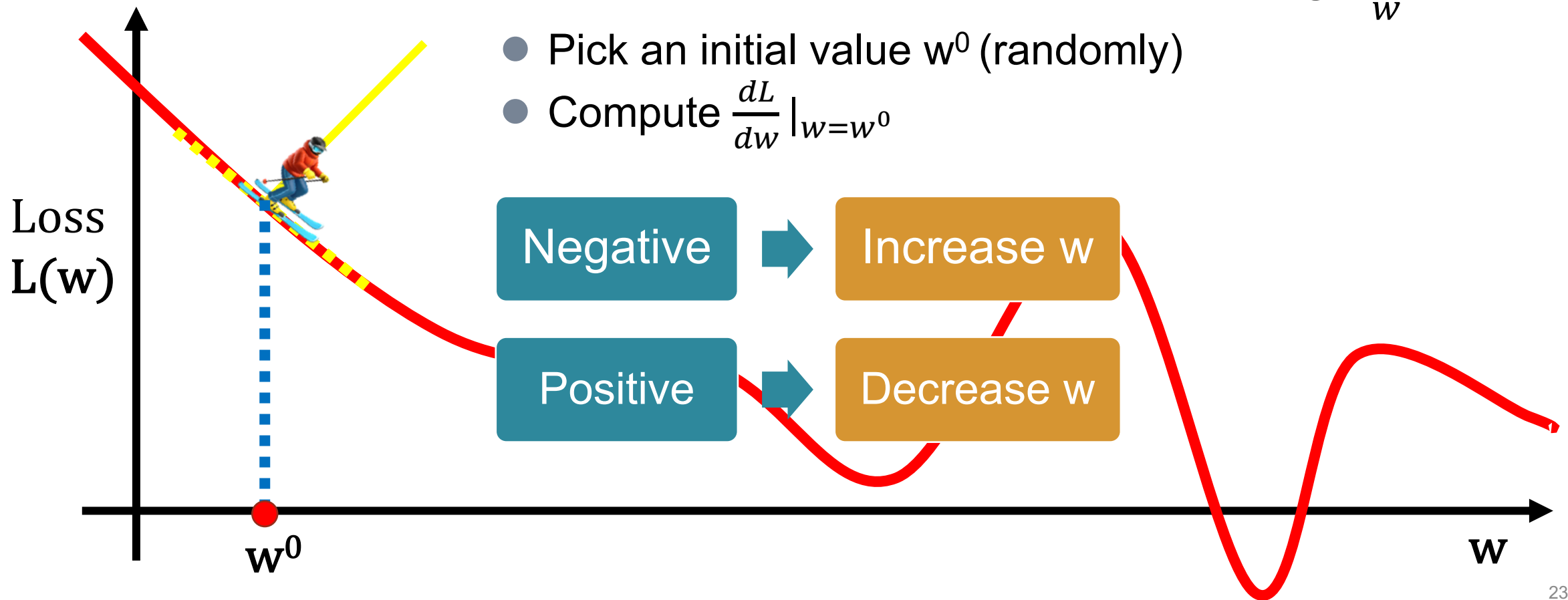
Gradient
Descent

2. Example Application

2.5 Training processing

Step3 Gradient descent

Consider loss function $L(w)$ with one parameter w : $w^* = \arg \min_w L(w)$

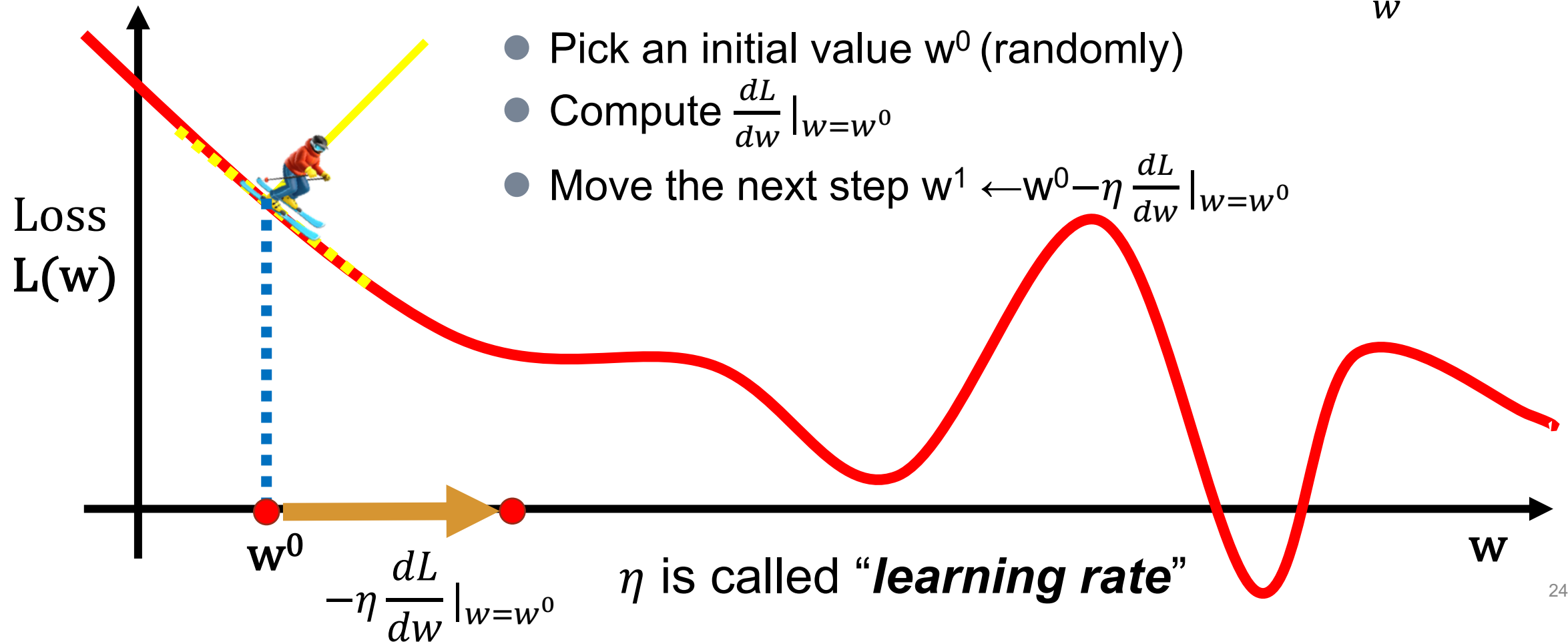


2. Example Application

2.5 Training processing

Step3 Gradient descent

Consider loss function $L(w)$ with one parameter w : $w^* = \arg \min_w L(w)$

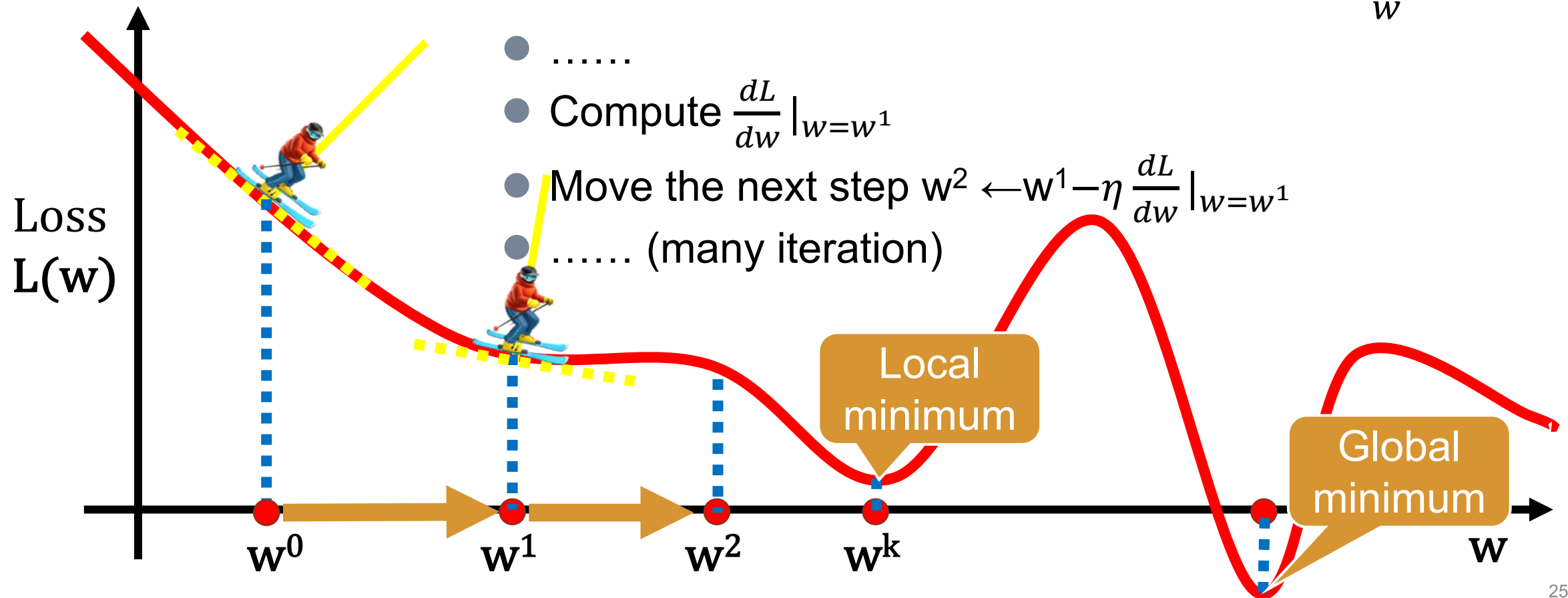


2. Example Application

2.5 Training processing

Step3 Gradient descent

Consider loss function $L(w)$ with one parameter w : $w^* = \arg \min_w L(w)$



2. Example Application

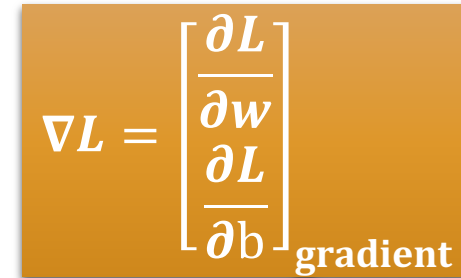
2.5 Training processing

How about two parameters?

- Pick the initial values w^0, b^0 (randomly)
- Compute $\frac{\partial L}{\partial w} \big|_{w=w^0, b=b^0}, \frac{\partial L}{\partial b} \big|_{w=w^0, b=b^0}$
- Move the next step $w^1 \leftarrow w^0 - \eta \frac{\partial L}{\partial w} \big|_{w=w^0, b=b^0}, b^1 \leftarrow b^0 - \eta \frac{\partial L}{\partial b} \big|_{w=w^0, b=b^0}$
- Compute $\frac{\partial L}{\partial w} \big|_{w=w^1, b=b^1}, \frac{\partial L}{\partial b} \big|_{w=w^1, b=b^1}$
- Move the next step $w^2 \leftarrow w^1 - \eta \frac{\partial L}{\partial w} \big|_{w=w^1, b=b^1}, b^2 \leftarrow b^1 - \eta \frac{\partial L}{\partial b} \big|_{w=w^1, b=b^1}$

Step3 Gradient descent

$$w^*, b^* = \arg \min_{w, b} L(w, b)$$


$$\nabla L = \begin{bmatrix} \frac{\partial L}{\partial w} \\ \frac{\partial L}{\partial b} \end{bmatrix}_{\text{gradient}}$$

2. Example Application

2.5 Training processing

Step3 Gradient descent

Consider this statement:

When we solving:

$\theta^* = \arg \min_{\theta} L(\theta)$ by gradient descent

Each time we update the parameters, we obtain θ that makes $L(\theta)$ smaller, *i.e.*:

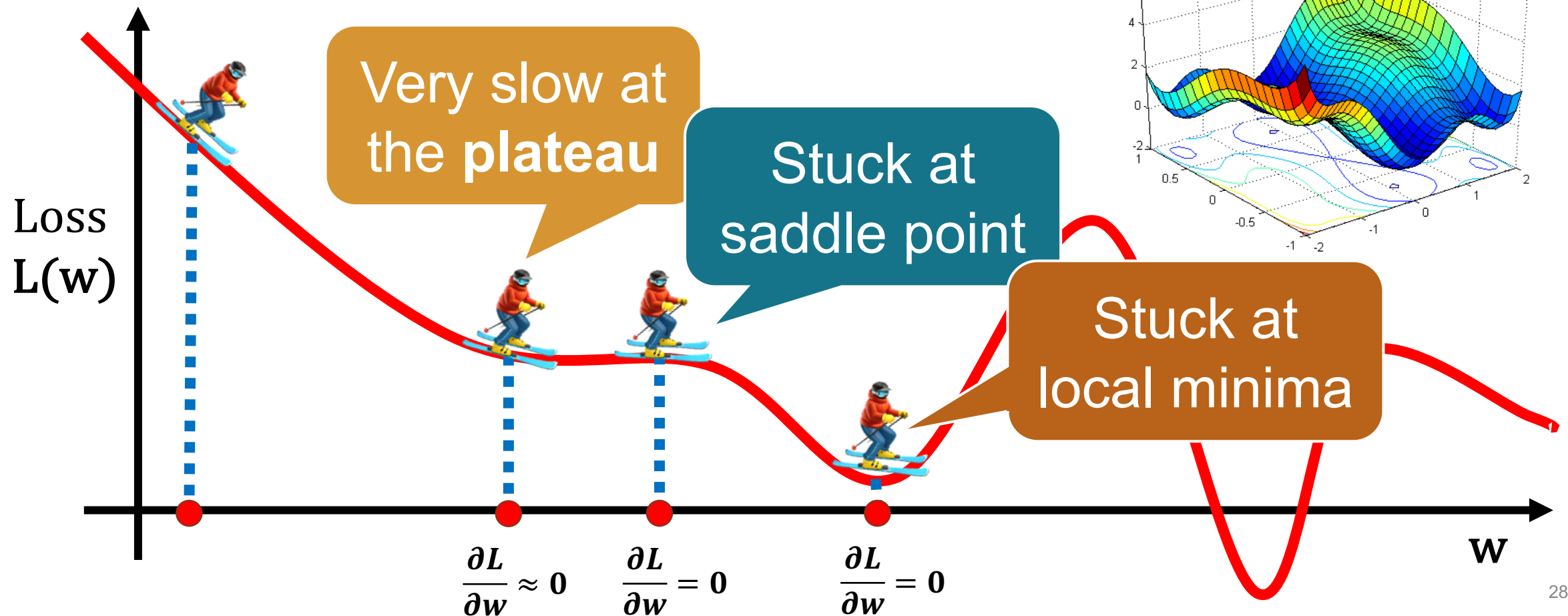
$$L(\theta^0) > L(\theta^1) > L(\theta^2) > L(\theta^3) > \dots > L(\theta^n)$$

Is it correct ?

2. Example Application

2.5 Training processing

Step3 Gradient descent



2. Example Application

2.5 Training processing

Step3 Gradient descent

Formulation of $\partial L / \partial w$ and $\partial L / \partial b$

$$L(w, b) = \sum_{n=0}^{115} (\hat{y}^n - (b + w \cdot x_{date}^n))^2$$

$$\frac{\partial L}{\partial w} = ?$$

$$\frac{\partial L}{\partial b} = ?$$

2. Example Application

2.5 Training processing

Step3 Gradient descent

Formulation of $\partial L / \partial w$ and $\partial L / \partial b$

$$L(w, b) = \sum_{n=0}^{115} (\hat{y}^n - (b + w \cdot x_{date}^n))^2$$

$$\frac{\partial L}{\partial w} = \sum_{n=0}^{115} 2(\hat{y}^n - (b + w \cdot x_{date}^n))(-x_{date}^n)$$

$$\frac{\partial L}{\partial b} = ?$$

2. Example Application

2.5 Training processing

Step3 Gradient descent

Formulation of $\partial L / \partial w$ and $\partial L / \partial b$

$$L(w, b) = \sum_{n=0}^{115} (\hat{y}^n - (b + w \cdot x_{date}^n))^2$$

$$\frac{\partial L}{\partial w} = \sum_{n=0}^{115} 2(\hat{y}^n - (b + w \cdot x_{date}^n))(-x_{date}^n)$$

$$\frac{\partial L}{\partial b} = \sum_{n=0}^{115} -2(\hat{y}^n - (b + w \cdot x_{date}^n))$$

2. Example Application

2.5 Training processing

How about the results?

$$y = b + w \cdot x_{date}$$

$$w = -0.0011$$

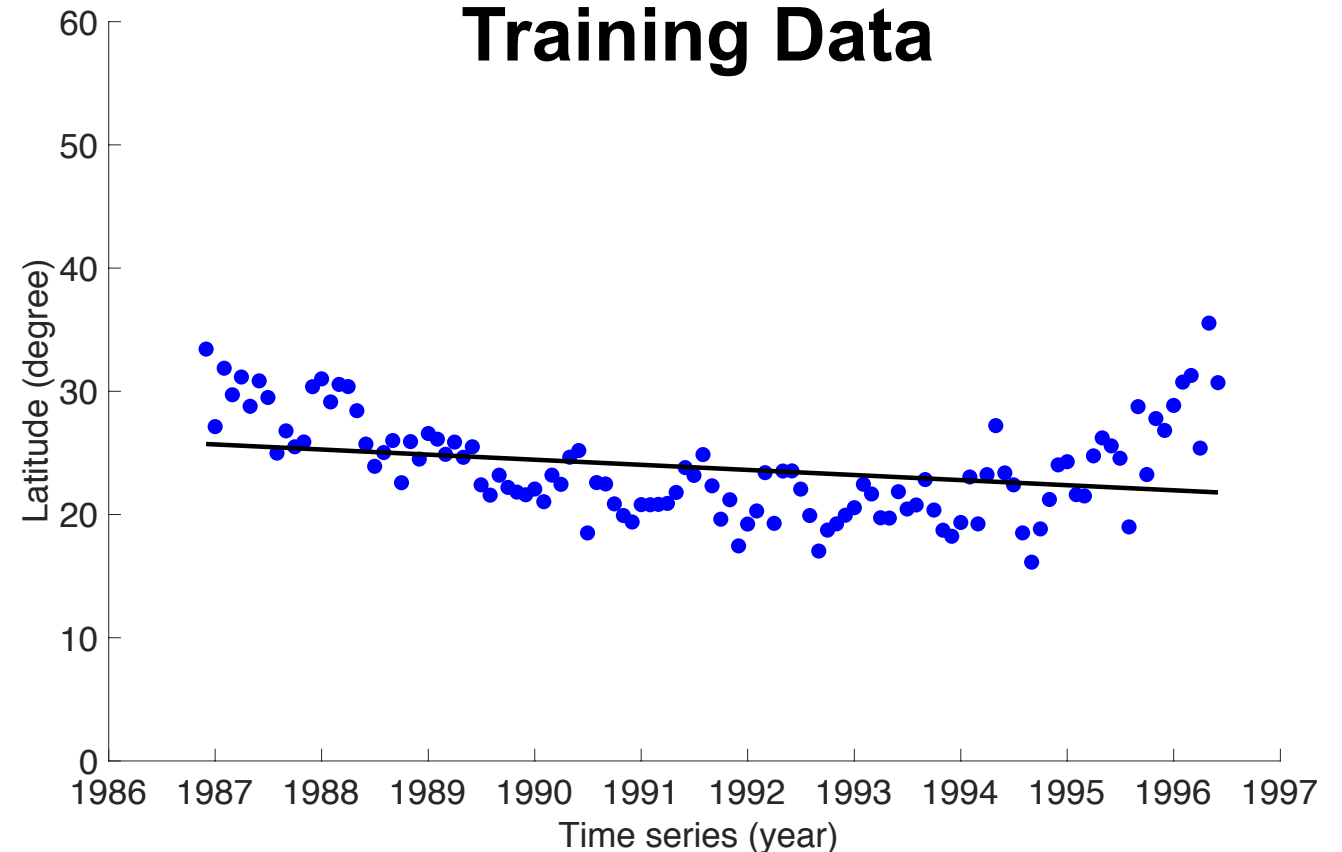
$$b = 847.59$$

Average Error on training data:

$$E = \frac{1}{110} \sum_{n=1}^{110} e^n = 2.95$$

Step3 Pick the best function

Training Data



2. Example Application

2.5 Training processing

Step3 Pick the best function

How about the results? --**Generalization**

$$y = b + w \cdot x_{date}$$

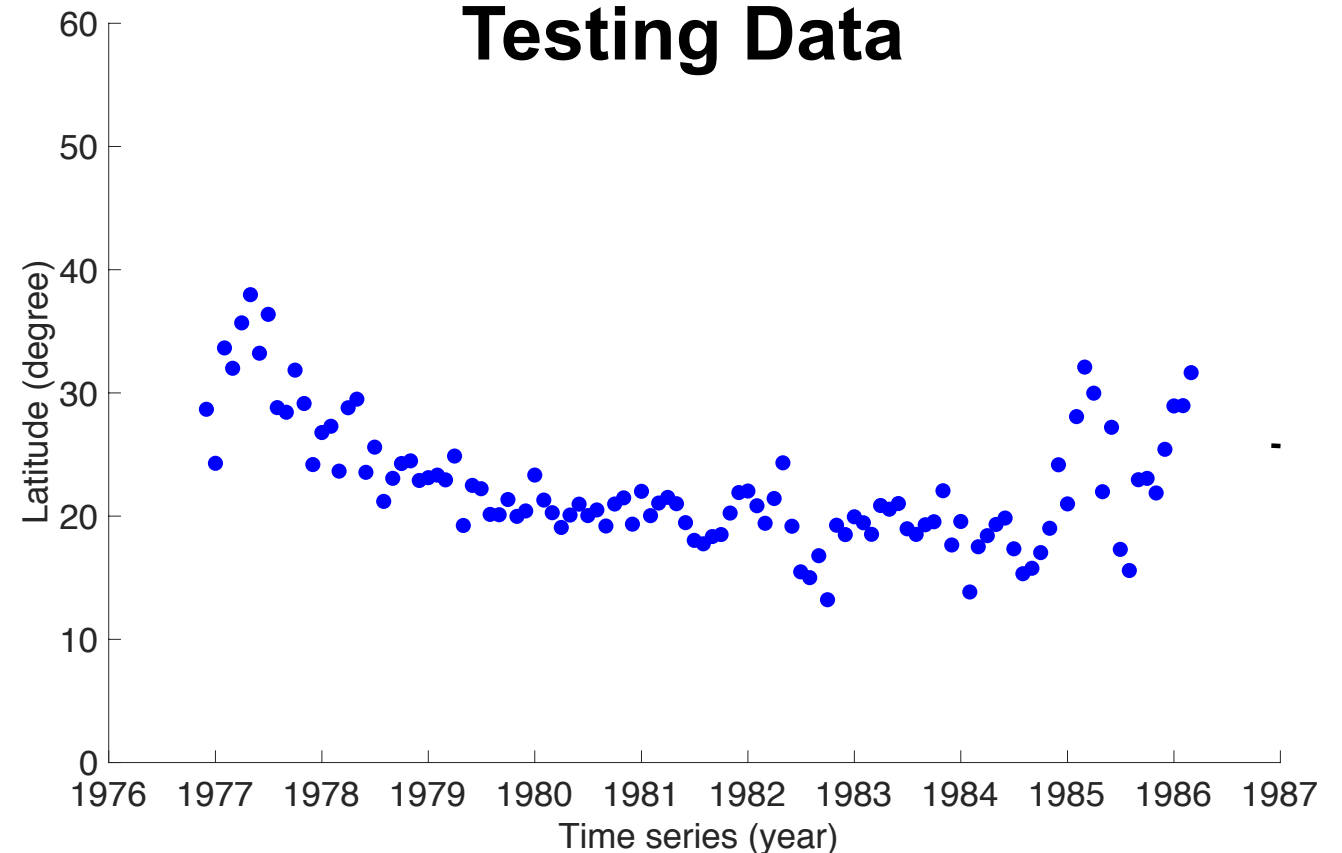
$$w = -0.0011$$

$$b = 847.59$$

Average Error on testing data:

$$E = \frac{1}{112} \sum_{n=1}^{112} e^n = ?$$

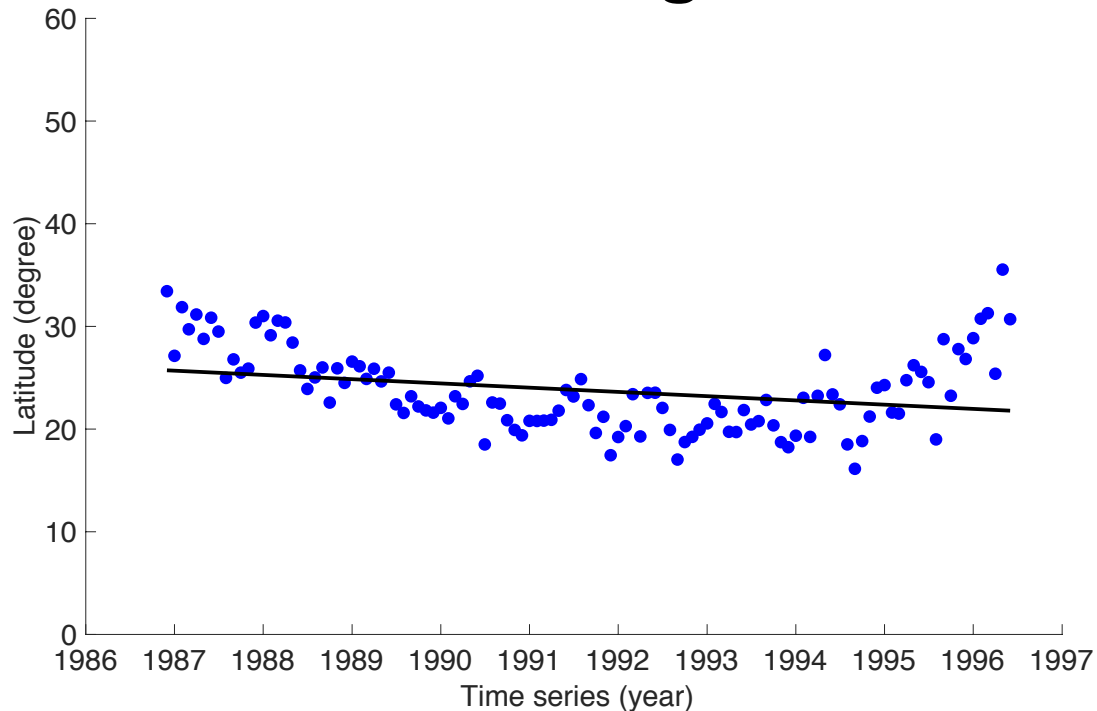
Testing Data



2. Example Application

2.5 Training processing

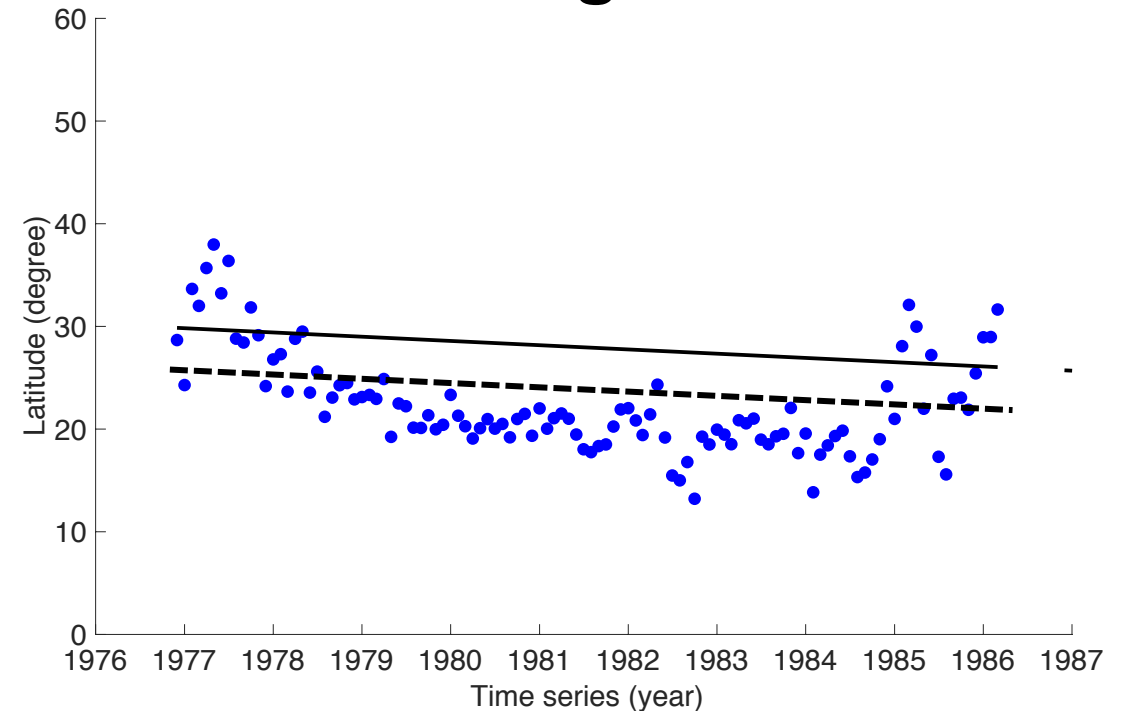
Training Data



$$y = b + w \cdot x_{date}$$
$$w = -0.0011$$
$$b = 847.59$$

Step3 Pick the best function

Testing Data

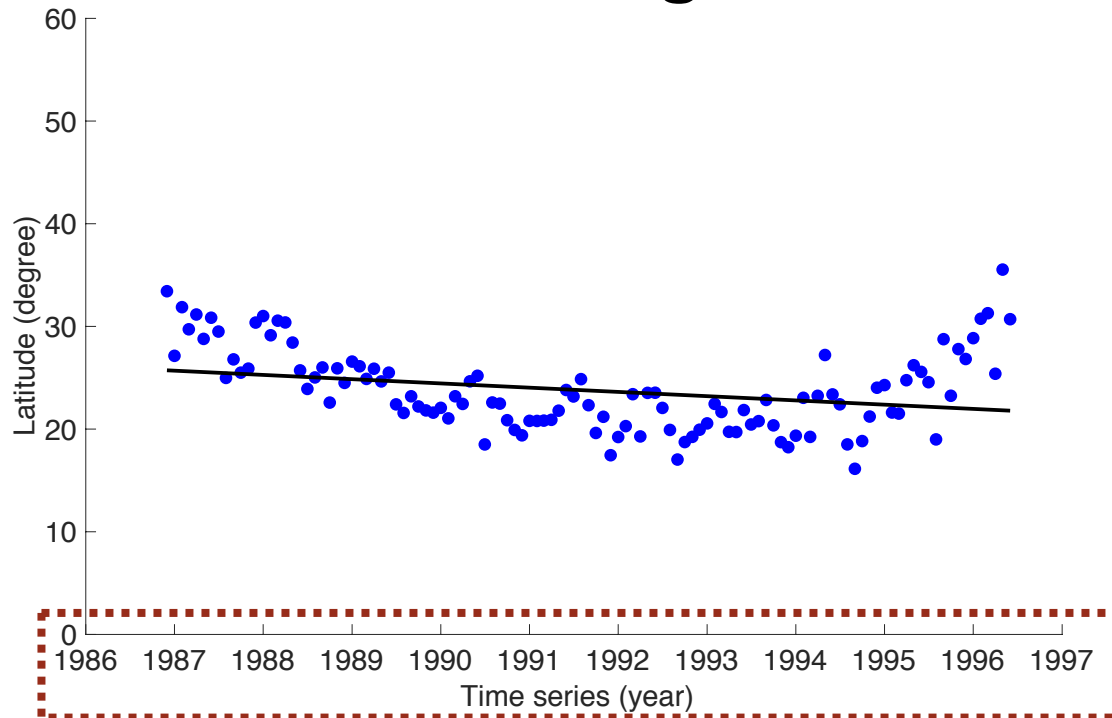


What happened ?????

2. Example Application

2.5 Training processing

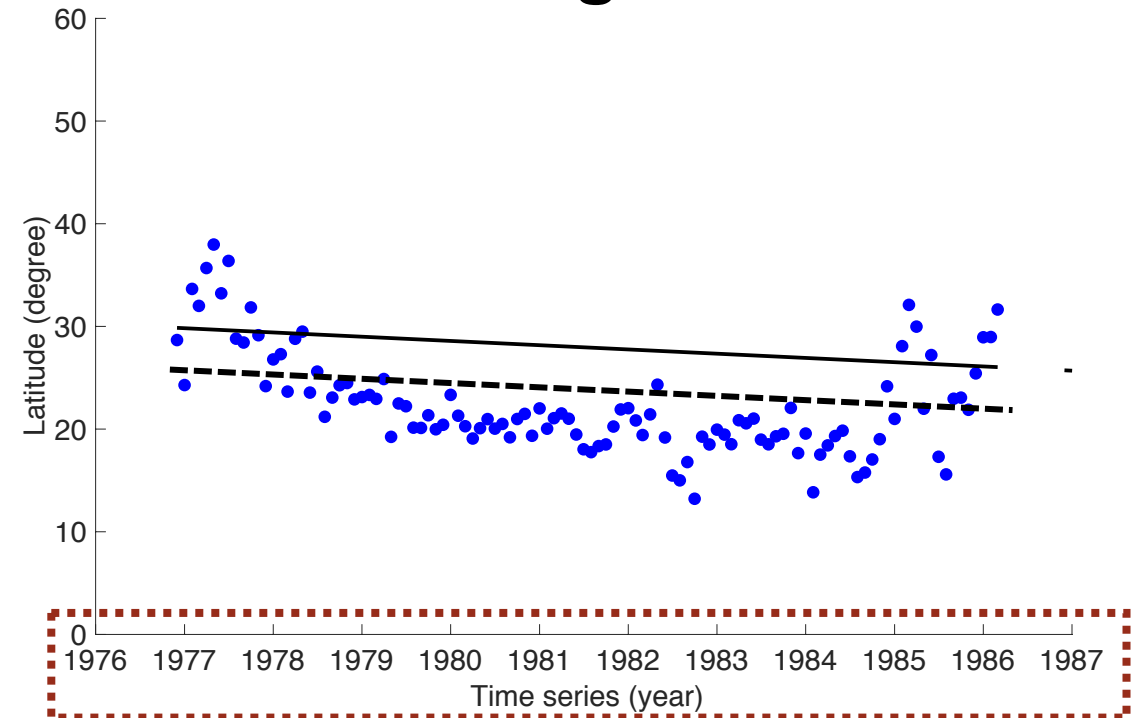
Training Data



$$y = b + w \cdot x_{date}$$
$$w = -0.0011$$
$$b = 847.59$$

Step3 Pick the best function

Testing Data



What happened ?????

2. Example Application

2.5 Training processing

Again Step3 Pick the best function

How about the results?

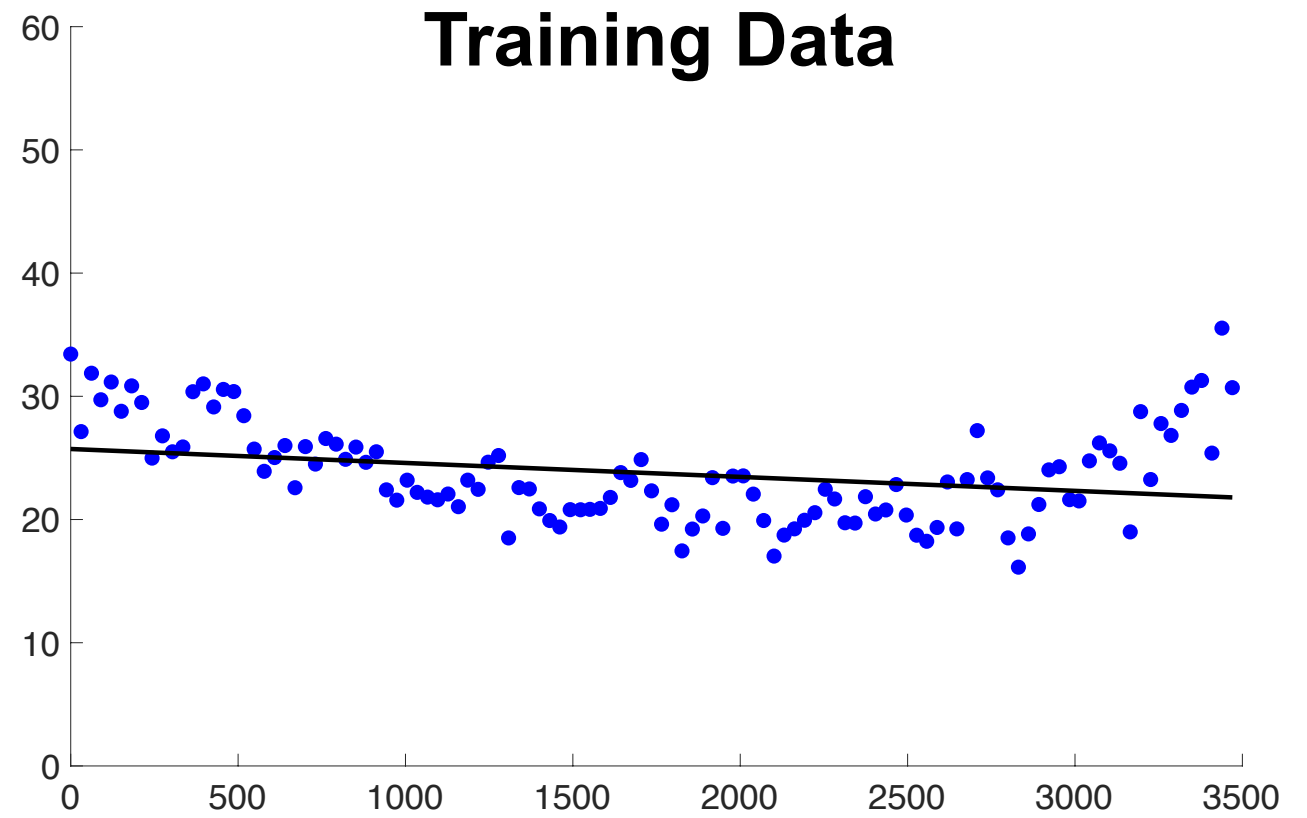
$$y = b + w \cdot x_{date}$$

$$w = -0.0011$$

$$b = 25.72$$

Average Error on training data:

$$E = \frac{1}{110} \sum_{n=1}^{110} e^n = 2.95$$



2. Example Application

2.5 Training processing

Again Step3 Pick the best function

How about the results? --Generalization

$$y = b + w \cdot x_{date}$$

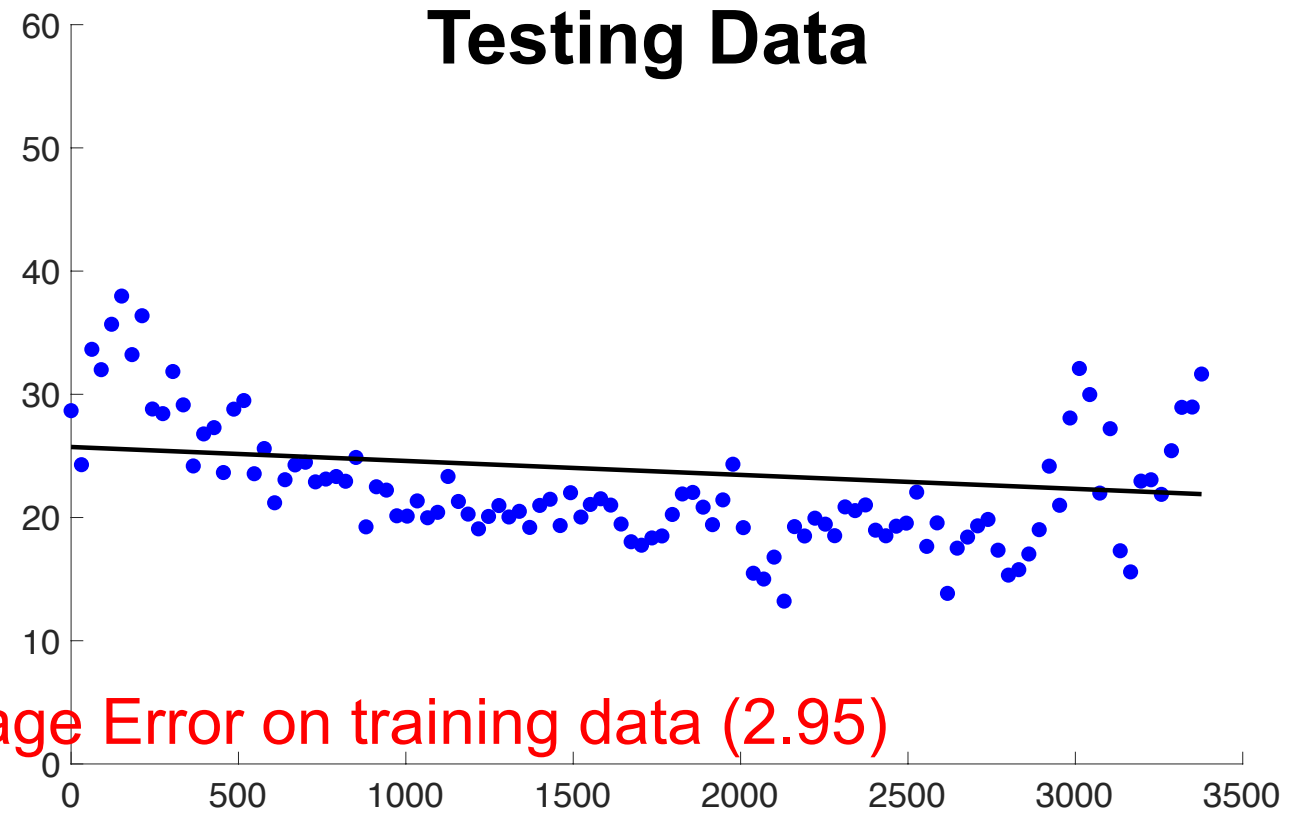
$$w = -0.0011$$

$$b = 25.72$$

Average Error on testing data:

$$E = \frac{1}{112} \sum_{n=1}^{112} e^n = 4.05 \quad > \text{Average Error on training data (2.95)}$$

Actually, we really care about the error on the new data (testing data)



2. Example Application

2.5 Training processing

How can we do better ?

2. Example Application

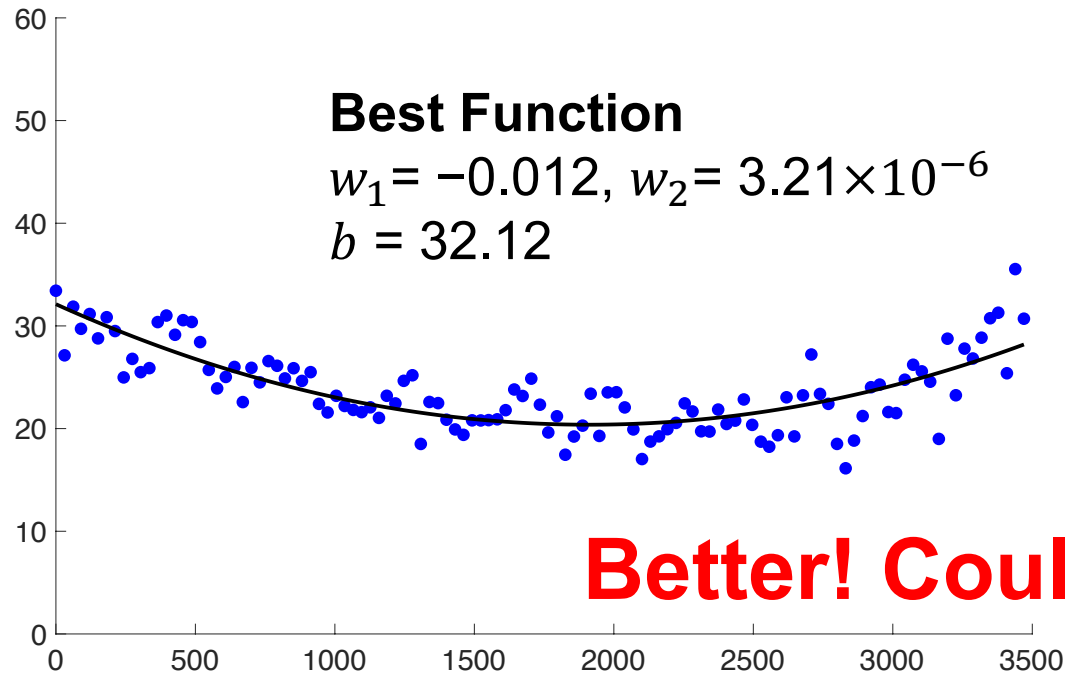
2.5 Training processing

$$y = b + w_1 \cdot x_{date} + w_2 \cdot (x_{date})^2$$

Best Function

$$w_1 = -0.012, w_2 = 3.21 \times 10^{-6}$$

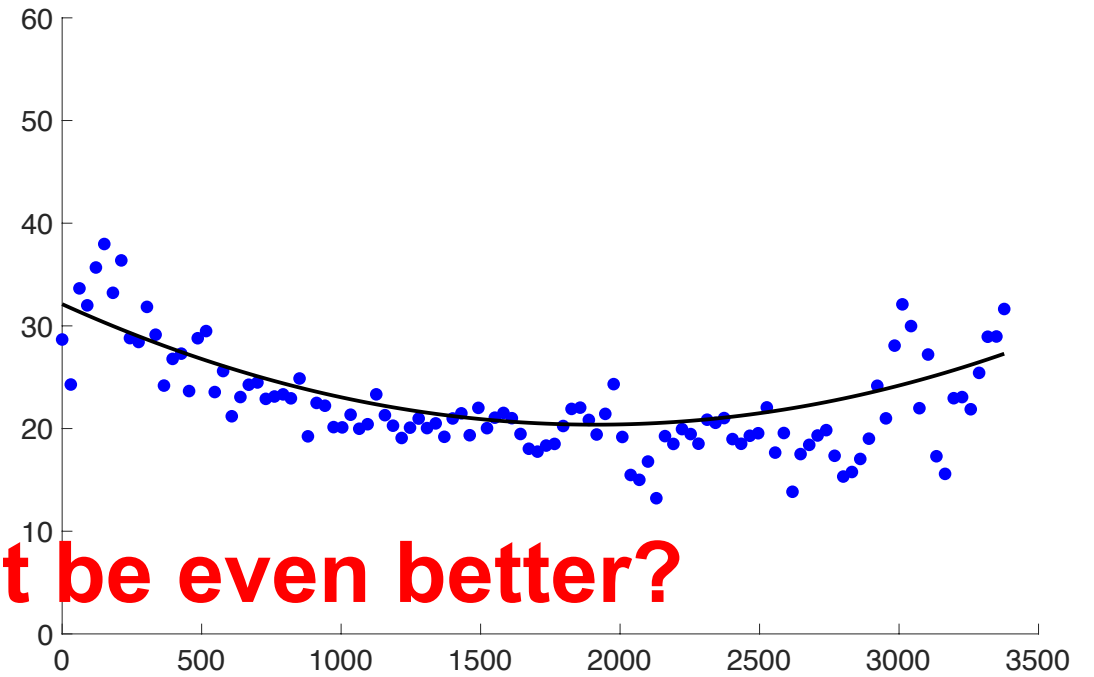
$$b = 32.12$$



Better! Could it be even better?

Average Error on **training data**: 1.84

Pick another Model



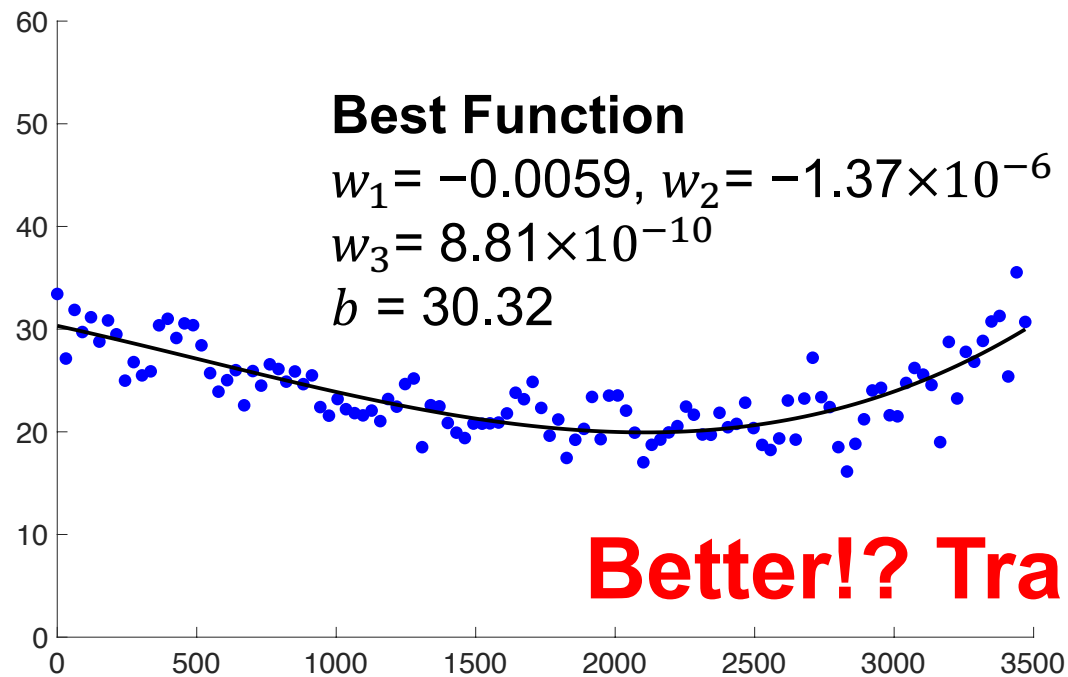
Average Error on **testing data**: 2.64

2. Example Application

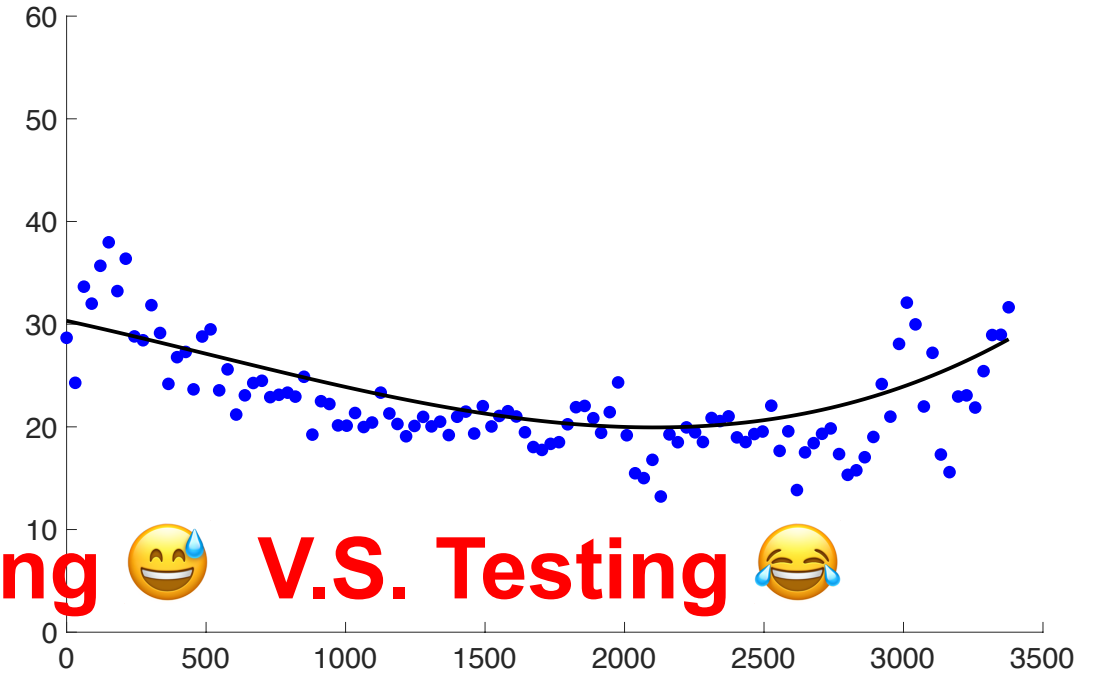
2.5 Training processing

$$y = b + w_1 \cdot x_{date} + w_2 \cdot (x_{date})^2 + w_3 \cdot (x_{date})^3$$

Pick another Model



Better!? Training 😓 **V.S. Testing** 😂



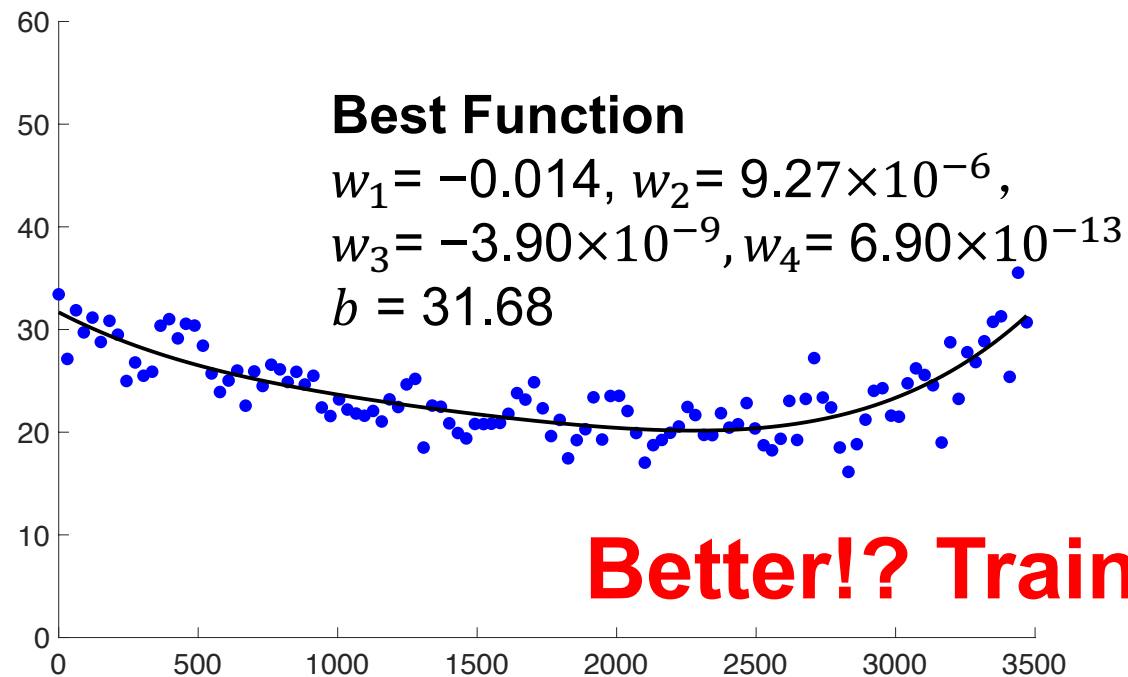
Average Error on **training data**: 1.77

Average Error on **testing data**: 2.67

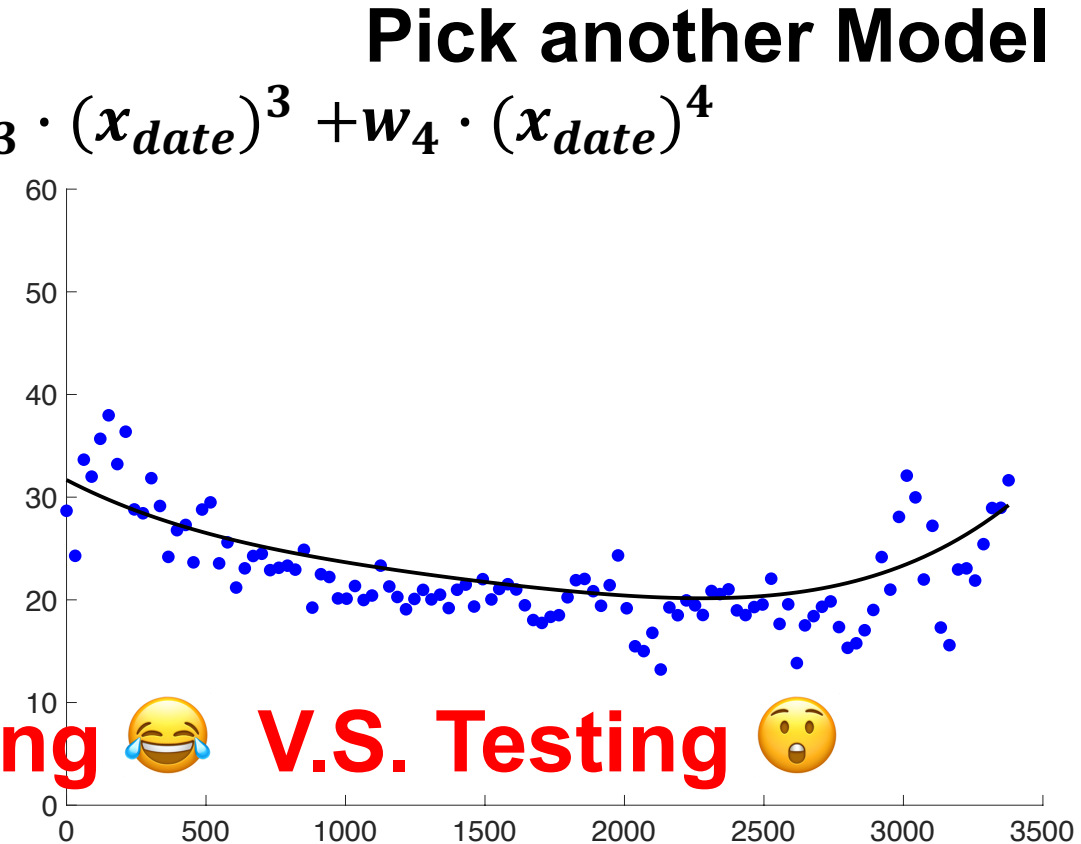
2. Example Application

2.5 Training processing

$$y = b + w_1 \cdot x_{date} + w_2 \cdot (x_{date})^2 + w_3 \cdot (x_{date})^3 + w_4 \cdot (x_{date})^4$$



Better!? Training 😂 V.S. Testing 😬



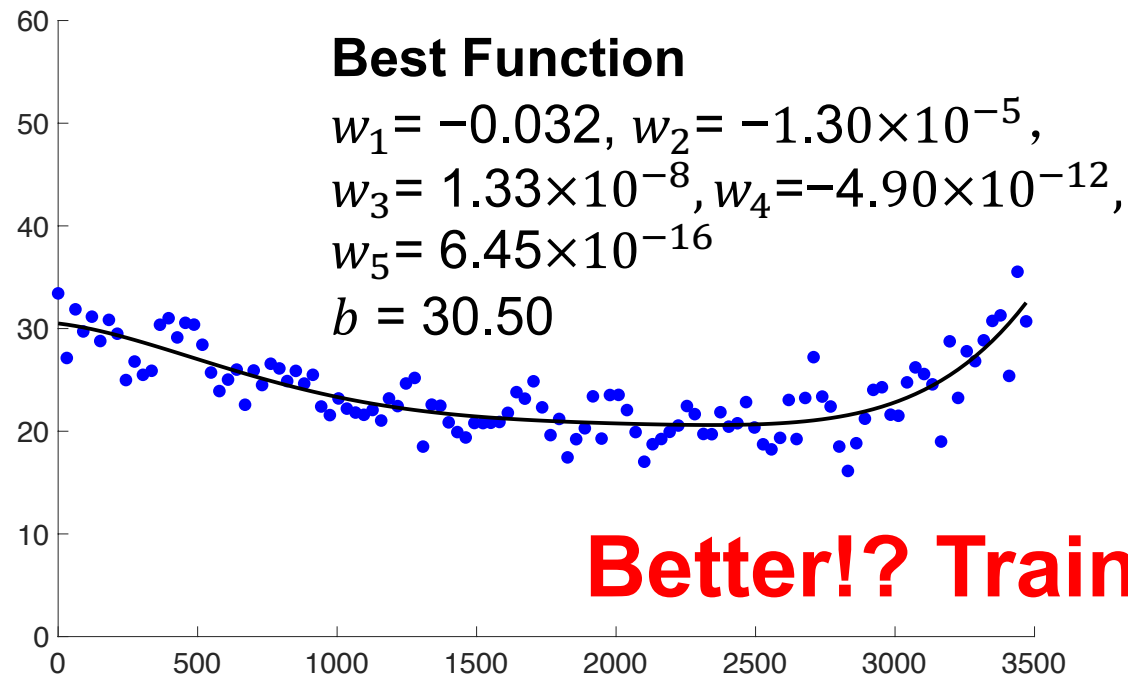
Average Error on **training data**: 1.76

Average Error on **testing data**: 2.60

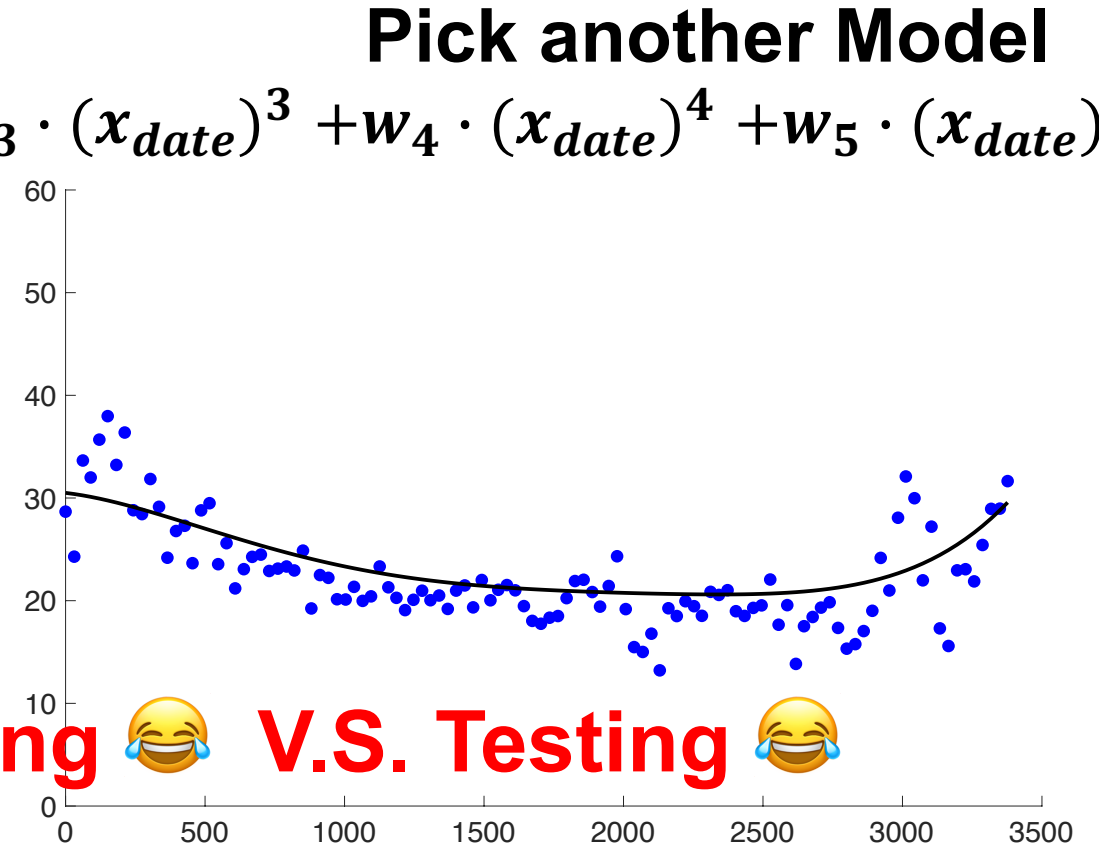
2. Example Application

2.5 Training processing

$$y = b + w_1 \cdot x_{date} + w_2 \cdot (x_{date})^2 + w_3 \cdot (x_{date})^3 + w_4 \cdot (x_{date})^4 + w_5 \cdot (x_{date})^5$$



Better!? Training 😂 V.S. Testing 😂



Average Error on **training data**: 1.75

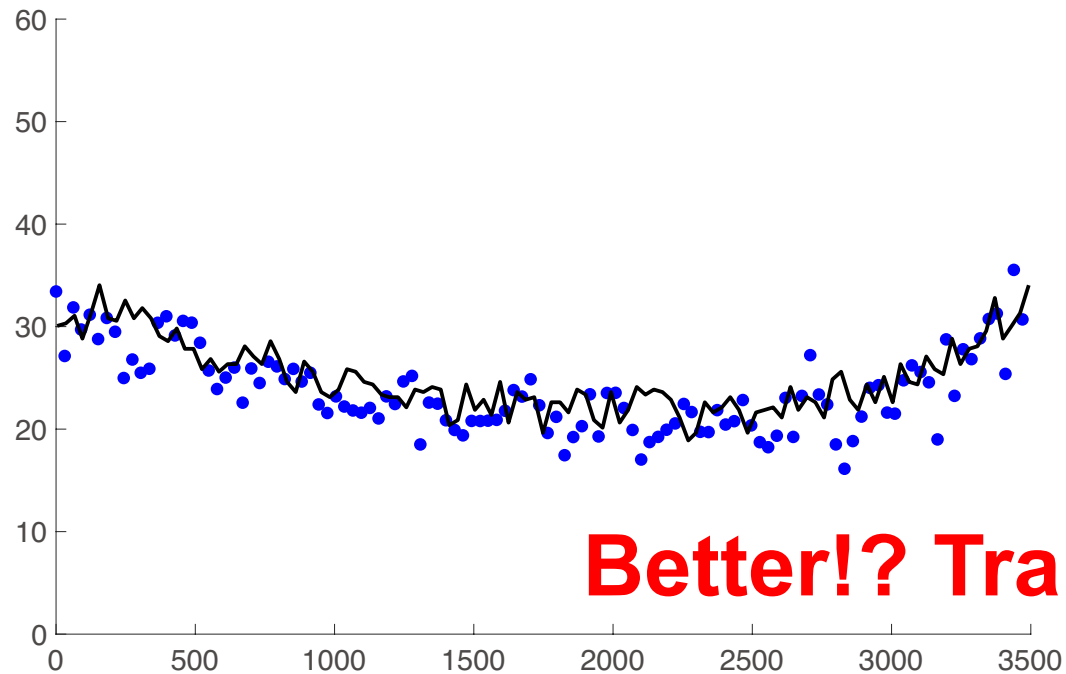
Average Error on **testing data**: 2.63

2. Example Application

2.5 Training processing

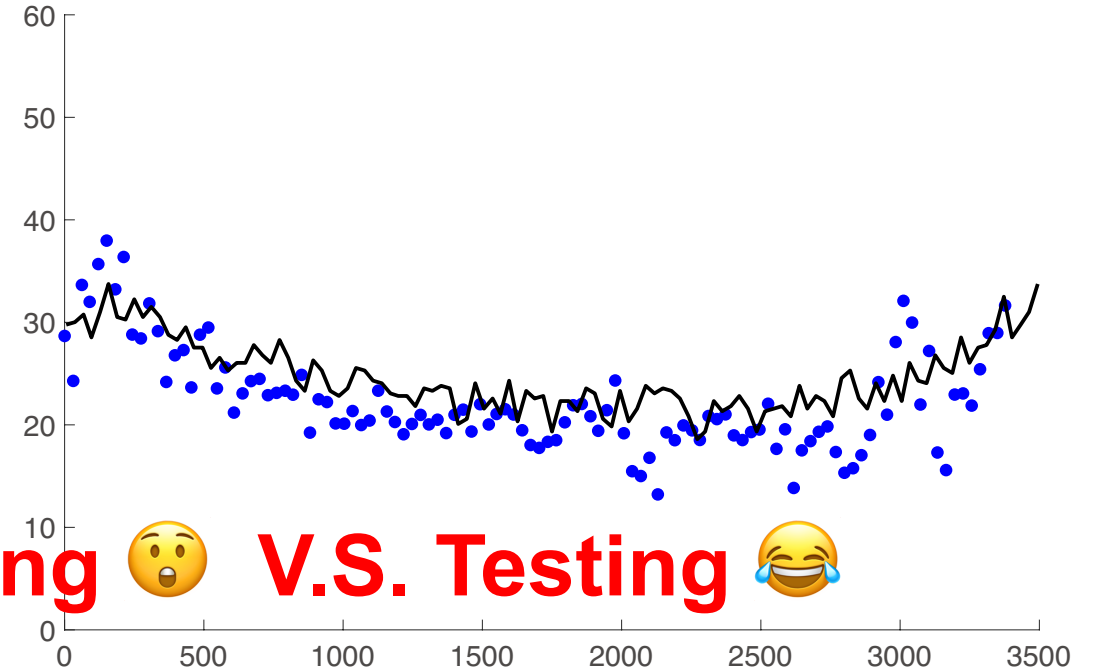
$$y = b + w_1 \cdot x_{date} + w_2 \cdot (x_{date})^2 + \dots + w_{10} \cdot (x_{date})^{10} + w_{11} \cdot (x_{date})^{11}$$

Pick another Model



Better!? Training 😬 **V.S. Testing** 😂

Average Error on **training data**: 1.72



Average Error on **testing data**: 3.38

2. Example Application

2.5 Training processing

Model Selection

If we can truly find the best function, a more complex model yields lower error on training data.

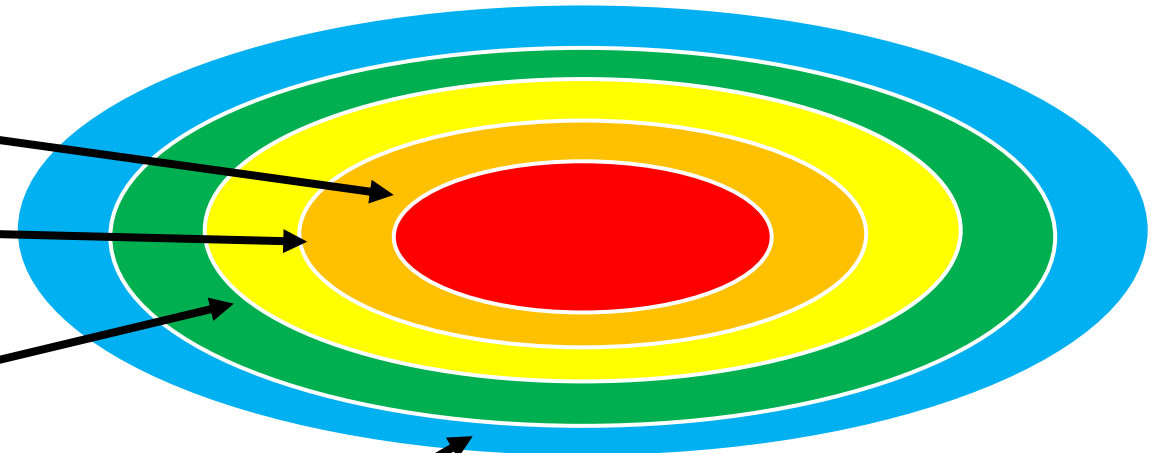
$$y = b + w \cdot x_{date}$$

$$y = b + w_1 \cdot x_{date} + w_2 \cdot (x_{date})^2$$

$$y = b + w_1 \cdot x_{date} + w_2 \cdot (x_{date})^2 + w_3 \cdot (x_{date})^3$$

$$y = b + w_1 \cdot x_{date} + w_2 \cdot (x_{date})^2 + w_3 \cdot (x_{date})^3 + w_4 \cdot (x_{date})^4$$

$$y = b + w_1 \cdot x_{date} + w_2 \cdot (x_{date})^2 + w_3 \cdot (x_{date})^3 + w_4 \cdot (x_{date})^4 + w_5 \cdot (x_{date})^5$$

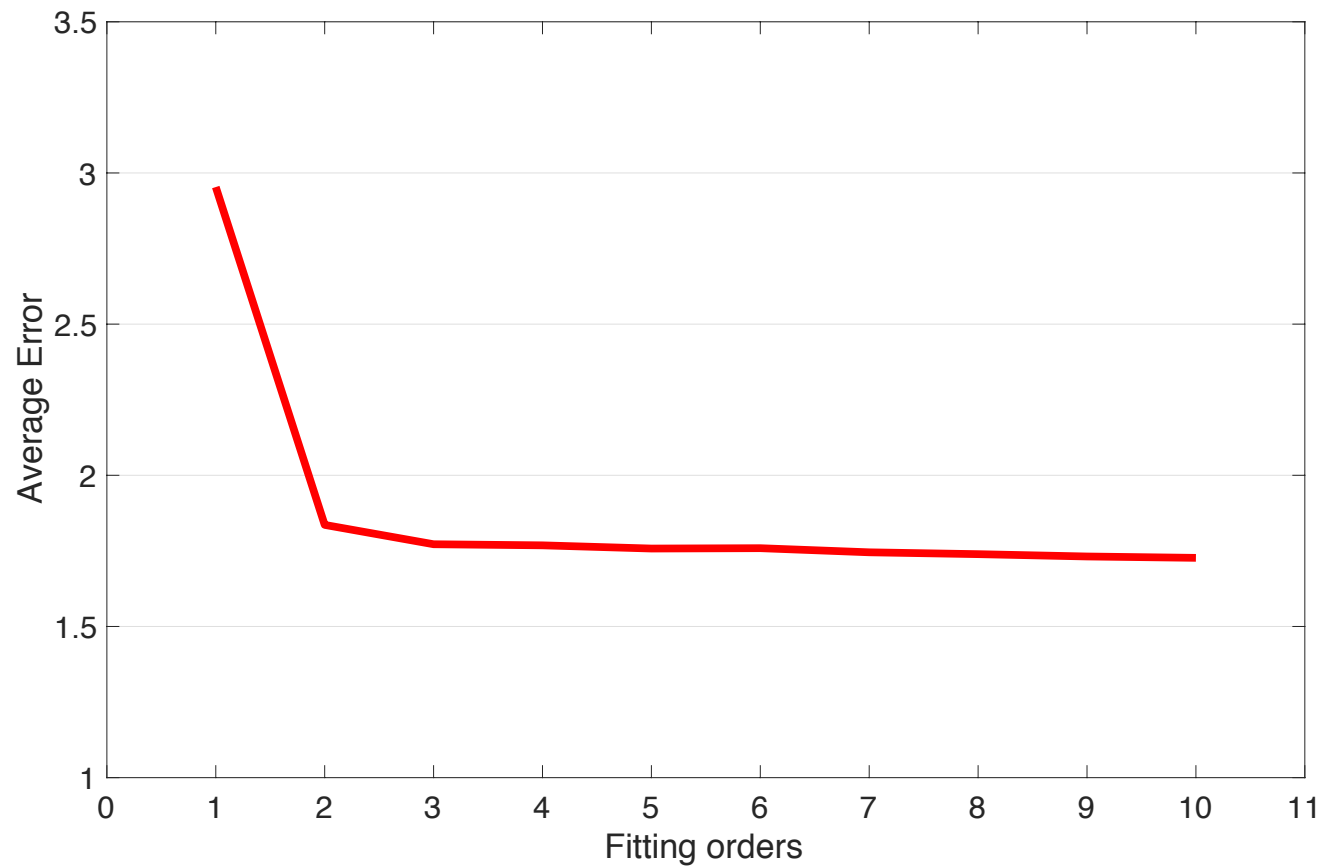


2. Example Application

2.5 Training processing

Model Selection

If we can truly find the best function, a more complex model yields lower error on training data.



2. Example Application

2.5 Training processing

Model Selection



A more complex model does not always lead to better performance on **testing data**!

2. Example Application

2.5 Training processing

Back to Step2 Goodness of Function

Regularization — Adding a penalty on model complexity

$$y = b + \sum w_i x_i$$

$$L = \sum_n \left(\hat{y}^n - \left(b + \sum w_i x_i \right) \right)^2 + \lambda \sum (w_i)^2$$

The functions with smaller w_i are better

Smooth functions are preferred 🤔 **WHY?**

$$w_i \Delta x_i + y = b + \sum w_i x_i + \Delta x_i$$

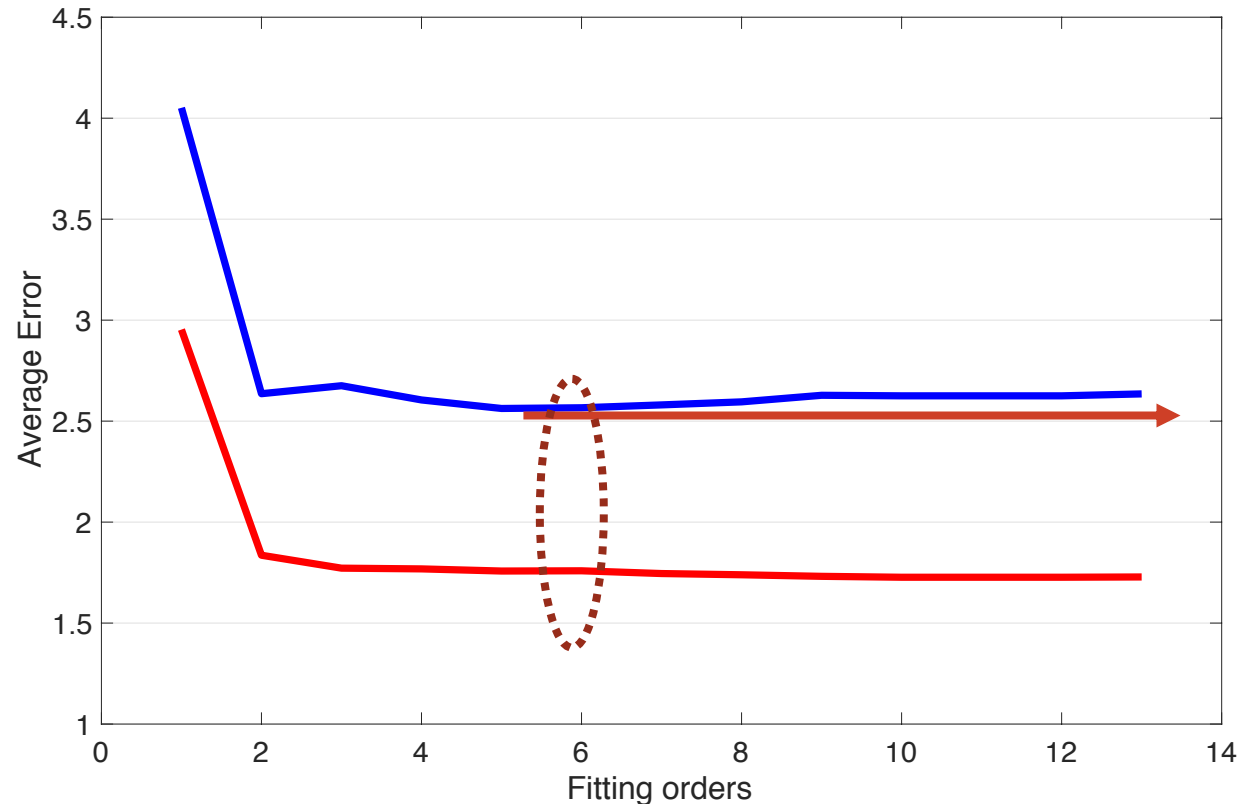
Smother function is more likely to be correct due to that it is not sensitive to noise.

2. Example Application

2.5 Training processing

Back to Step2 Goodness of Function

Regularization



Training Data

Testing Data

How smooth ?

Select λ obtaining
the best model

Larger λ , considering the training error less; prefer smooth function, but do not be too smooth

2. Example Application

2.5 Training processing

Back to Step2 Goodness of Function

Regularization — Adding a penalty on model complexity

$$L = \sum_n \left(\hat{y}^n - \left(b + \sum w_i x_i \right) \right)^2 + \lambda \sum (w_i)^2$$

L2-norm

Ridge regression

$$L = \sum_n \left(\hat{y}^n - \left(b + \sum w_i x_i \right) \right)^2 + \lambda \sum |w_i|$$

L1-norm

Lasso regression

Least absolute shrinkage and selection operator

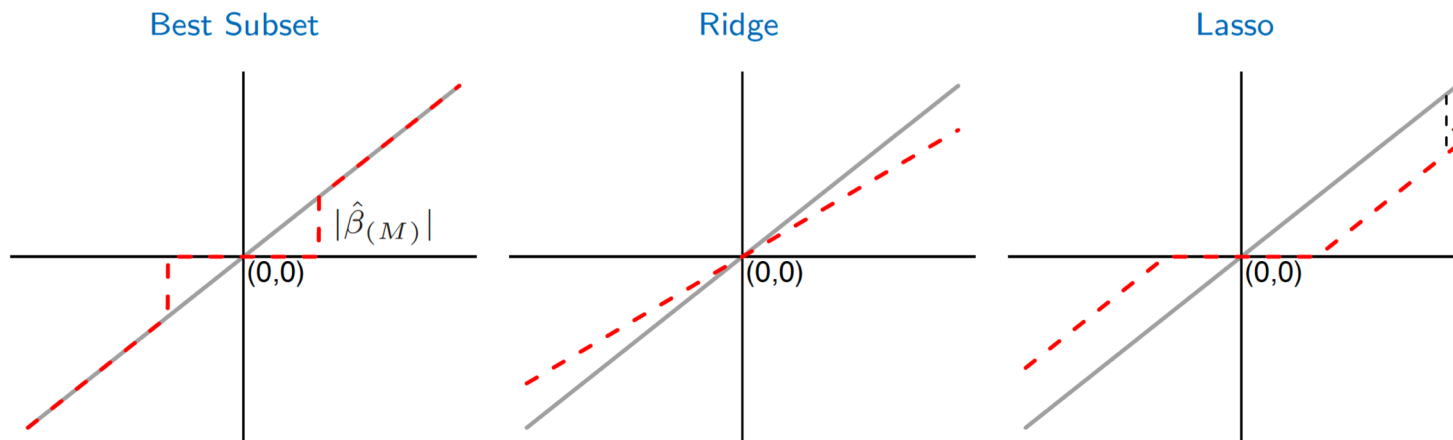
2. Example Application

2.5 Training processing

Back to Step2 Goodness of Function

Regularization

Estimator	Formula
Best subset (size M)	$\hat{\beta}_j \cdot I(\hat{\beta}_j \geq \hat{\beta}_{(M)})$
Ridge	$\hat{\beta}_j / (1 + \lambda)$
Lasso	$\text{sign}(\hat{\beta}_j)(\hat{\beta}_j - \lambda)_+$



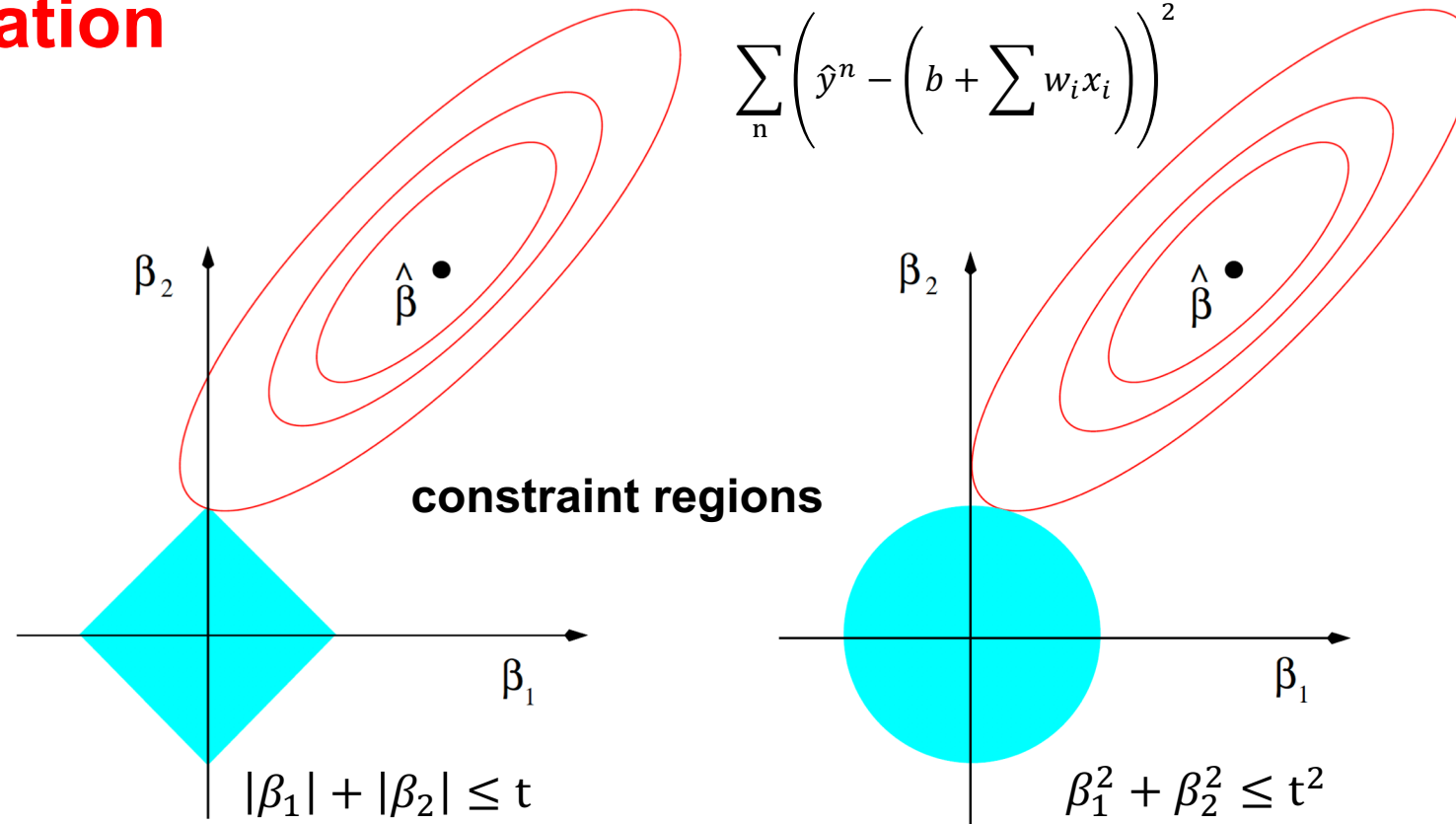
Estimators of β_j in the case of orthonormal columns of $\mathbf{X}$. M and λ are constants chosen by the corresponding techniques; sign denotes the sign of its argument (± 1), and x_+ denotes “positive part” of x . Below the table, estimators are shown by broken red lines. The 45° line in gray shows the unrestricted estimate for reference.

2. Example Application

2.5 Training processing

Regularization

Back to Step2 Goodness of Function



Lasso regression

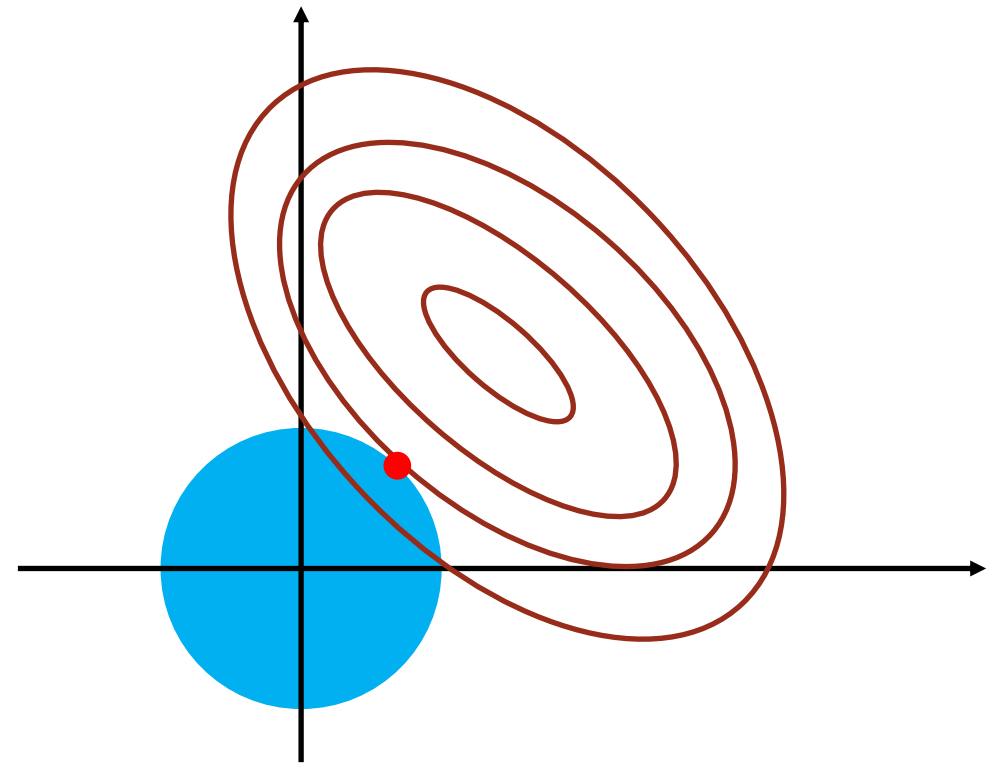
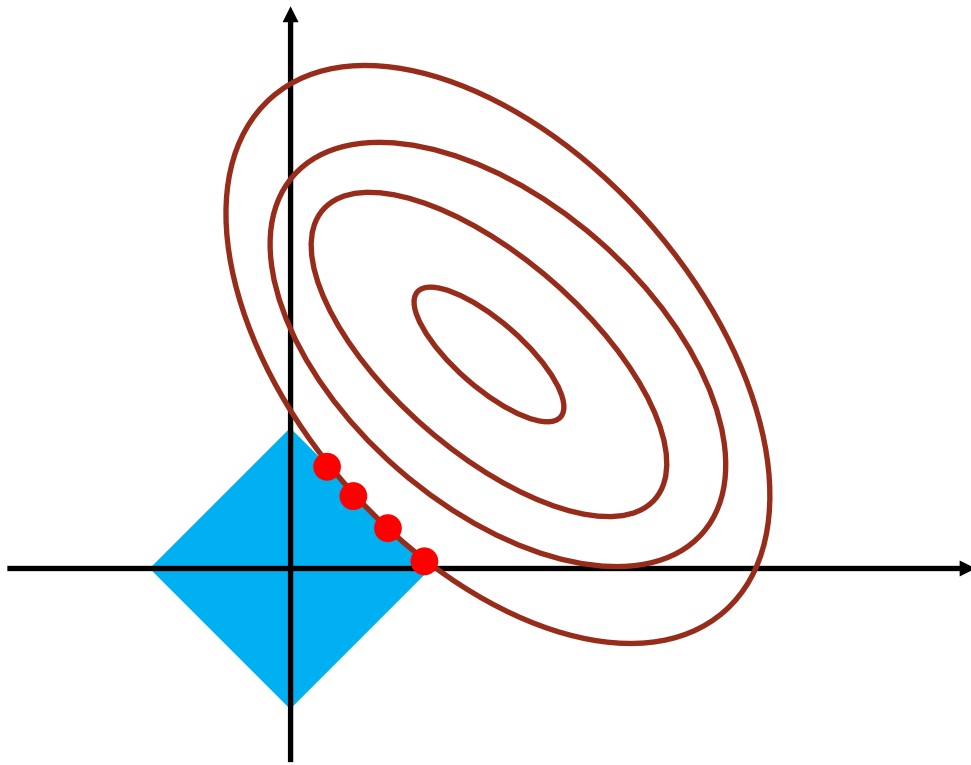
Ridge regression

2. Example Application

2.5 Training processing

Regularization

Back to Step2 Goodness of Function



L1 regularization encourages zero coefficients
The solution of L1 is unstable

2. Example Application

2.5 Training processing

Back to Step2 Goodness of Function

Regularization

L1 or L2, how to choose?

- L1 tends to shrink coefficients to zero whereas L2 tends to shrink coefficients evenly.
- L1 is useful for feature selection, as we can drop any variables associated with coefficients that go to zero.
- L2 is useful when you have collinear/codependent features.

2. Example Application

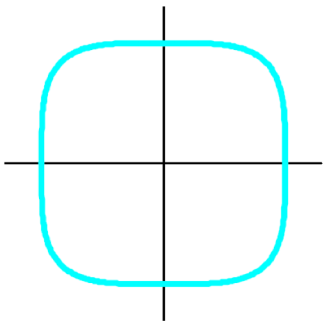
2.5 Training processing

Back to Step2 Goodness of Function

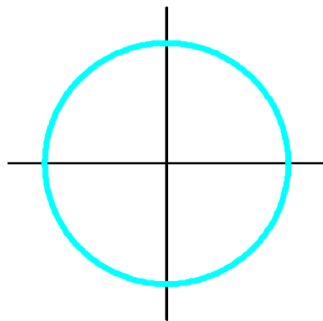
Regularization

$$L = \sum_n \left(\hat{y}^n - \left(b + \sum w_i x_i \right) \right)^2 + \lambda \sum |w_i|^q \quad q \geq 0$$

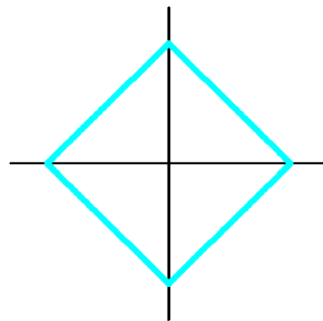
$q = 4$



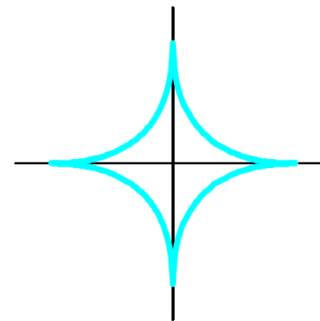
$q = 2$



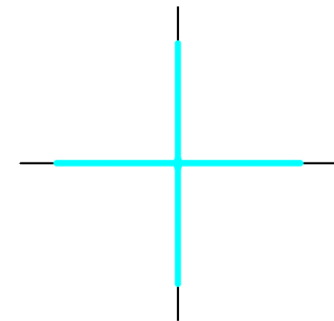
$q = 1$



$q = 0.5$



$q = 0.1$



For $q \geq 1$, the cost function is convex and has a unique minimum. The solution can be obtained by quadratic optimization.

For $q < 1$, the problem is not convex, and obtaining the global minimum is more difficult.

2. Example Application

2.5 Training processing

Back to Step2 Goodness of Function

Regularization

$$L = \sum_n \left(\hat{y}^n - \left(b + \sum w_i x_i \right) \right)^2 + \lambda \sum (w_i)^2 \quad L = \sum_n \left(\hat{y}^n - \left(b + \sum w_i x_i \right) \right)^2 + \lambda \sum |w_i|$$

Elastic-net regression

$$L = \sum_n \left(\hat{y}^n - \left(b + \sum w_i x_i \right) \right)^2 + \lambda \sum (\alpha (w_i)^2 + (1 - \alpha) |w_i|)$$

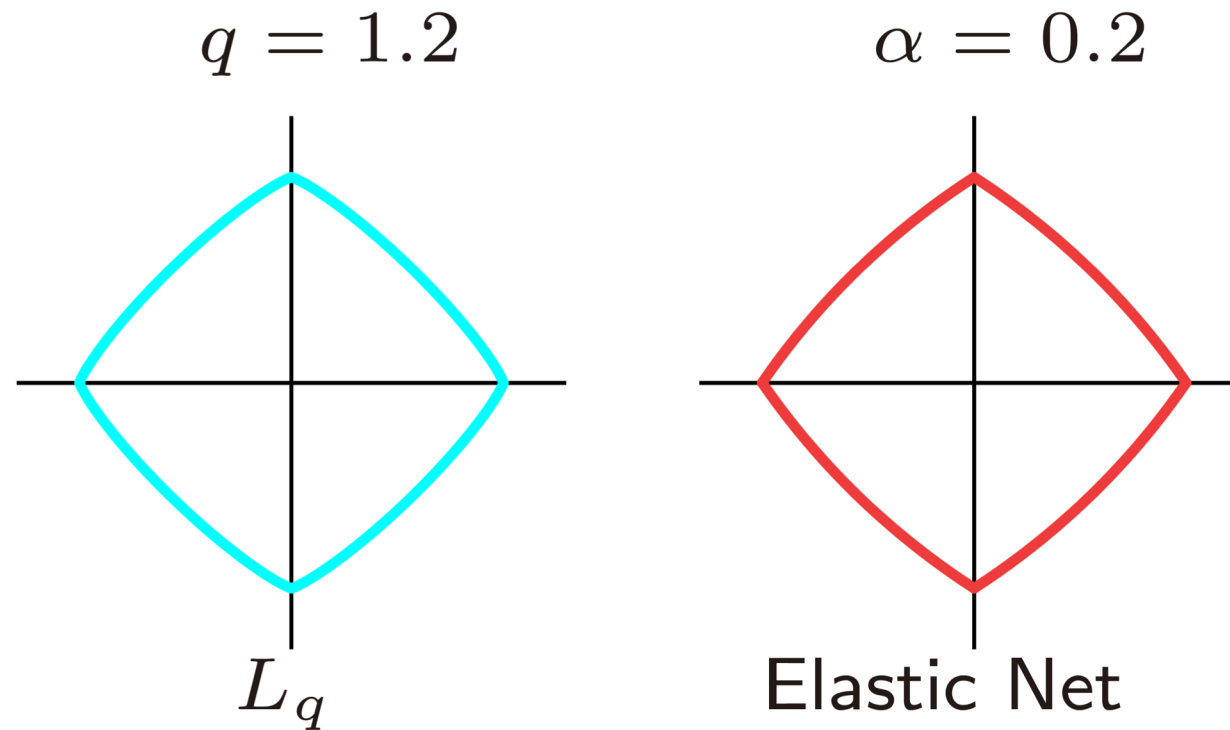
The elastic-net selects variables like the lasso, and shrinks together the coefficients of correlated predictors like ridge.

2. Example Application

2.5 Training processing

Regularization

Back to Step2 Goodness of Function



Although visually very similar, the elastic-net has sharp (non-differentiable) corners, while the $q = 1.2$ penalty does not.

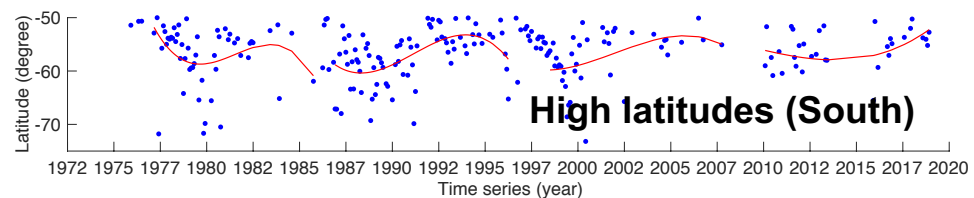
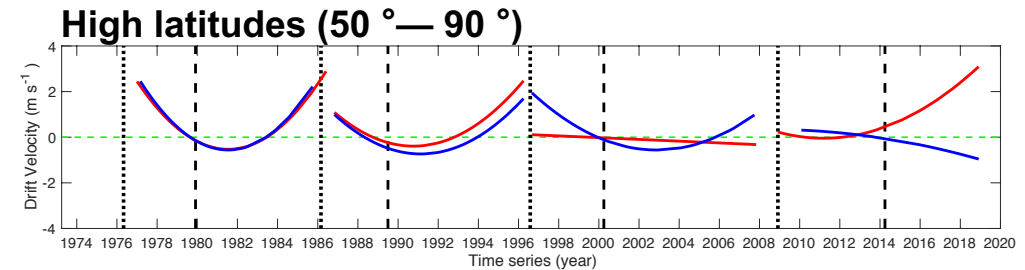
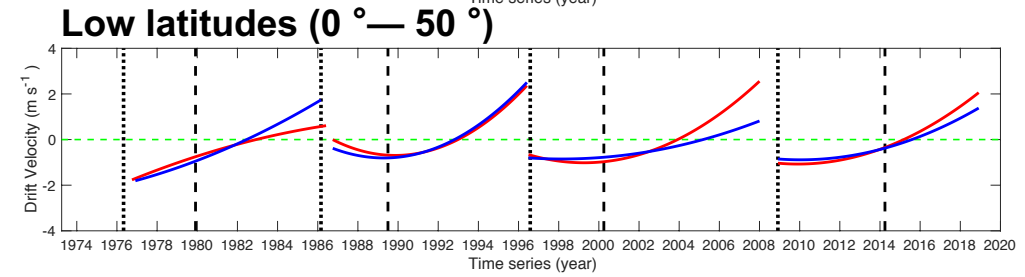
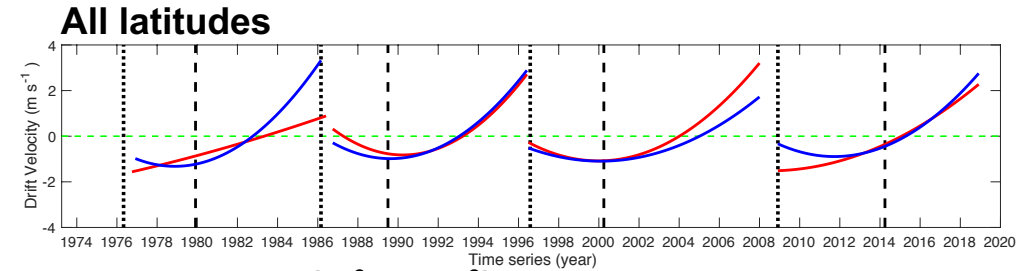
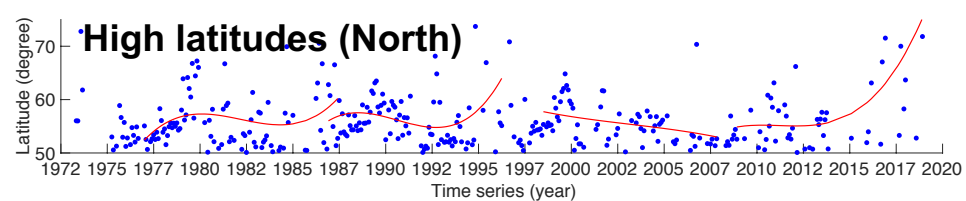
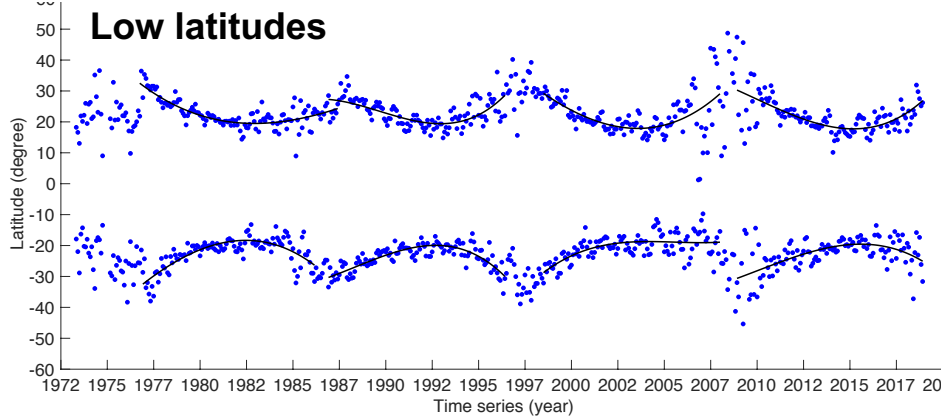
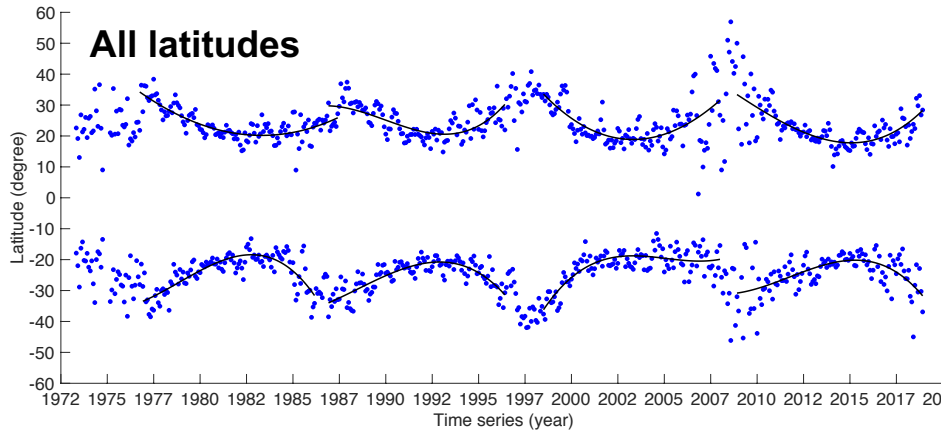
2. Example Application

2.5 Training processing

Is there any hidden factors not considered in the previous model?

2. Example Application

2.5 Training processing



2. Example Application

2.5 Training processing

Redesign the Model

$$y = b + w \cdot x_{date} \quad \longrightarrow \quad y = b + \sum w_i x_i$$

x_{lat} = latitude categories of x



If x_{lat} = Low latitude: $y = b_1 + w_1 \cdot x_{date}$
If x_{lat} = High latitude: $y = b_2 + w_2 \cdot x_{date}$



2. Example Application

2.5 Training processing

Redesign the Model

$$y = b + \sum w_i x_i$$

$$y = b_1 \cdot \delta(x_{lat} = \text{Low latitude}) + w_1 \cdot \delta(x_{lat} = \text{Low latitude}) \cdot x_{date} \\ + b_2 \cdot \delta(x_{lat} = \text{High latitude}) + w_2 \cdot \delta(x_{lat} = \text{High latitude}) \cdot x_{date}$$

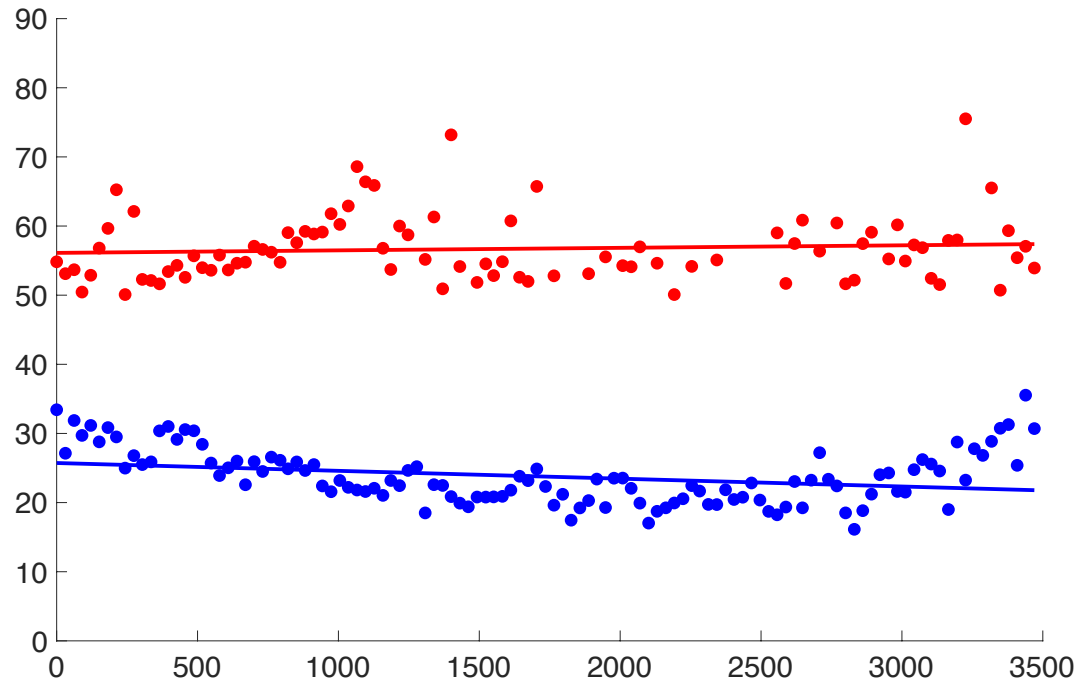
$$\delta(x_{lat} = \text{Low latitude}) \begin{cases} = 1 & \text{If } x_{lat} = \text{Low latitude} \\ = 0 & \text{otherwise} \end{cases}$$

If $x_{lat} = \text{Low latitude}$

$$y = b_1 + w_1 \cdot x_{date}$$

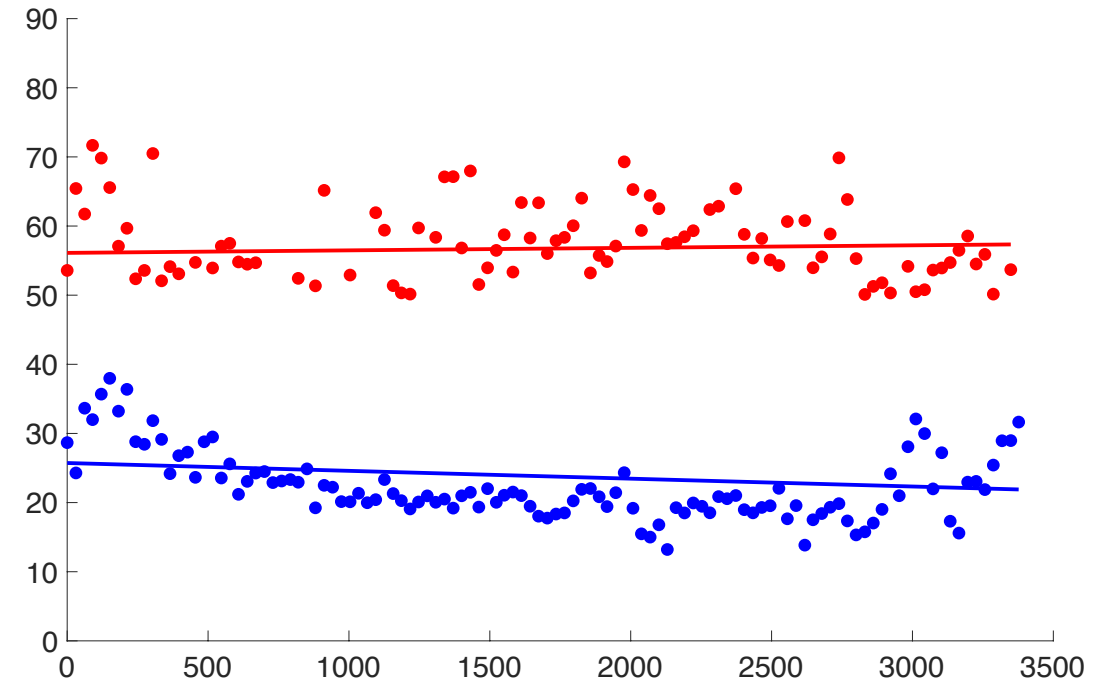
2. Example Application

2.5 Training processing



Average Error on **training data**: 3.27

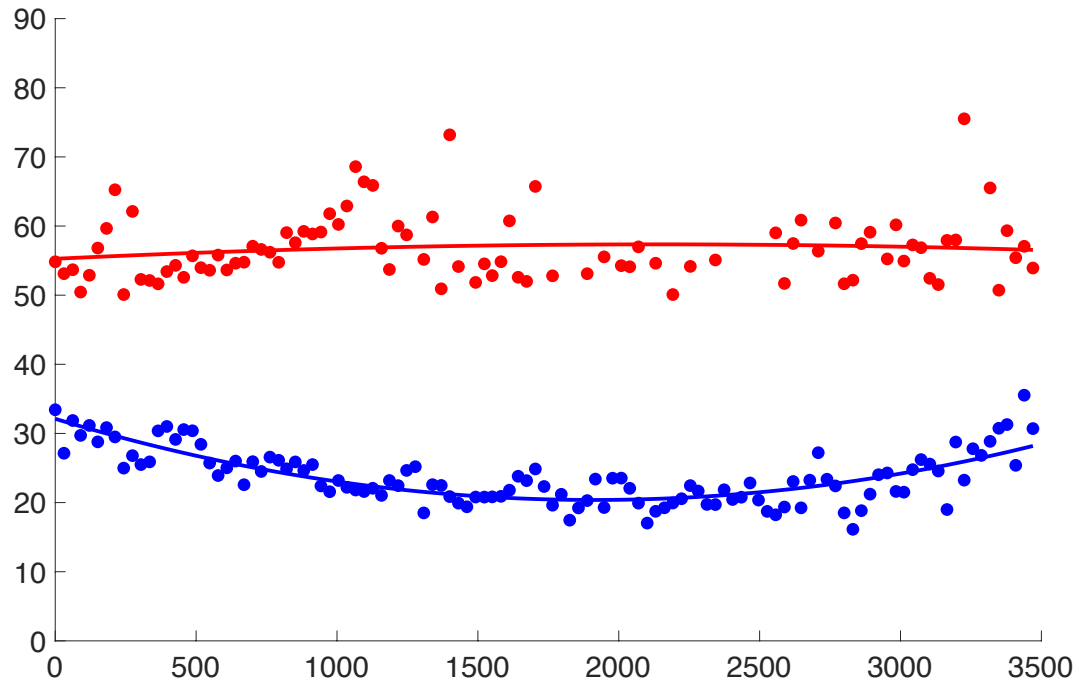
Redesign the Model



Average Error on **testing data**: 4.20

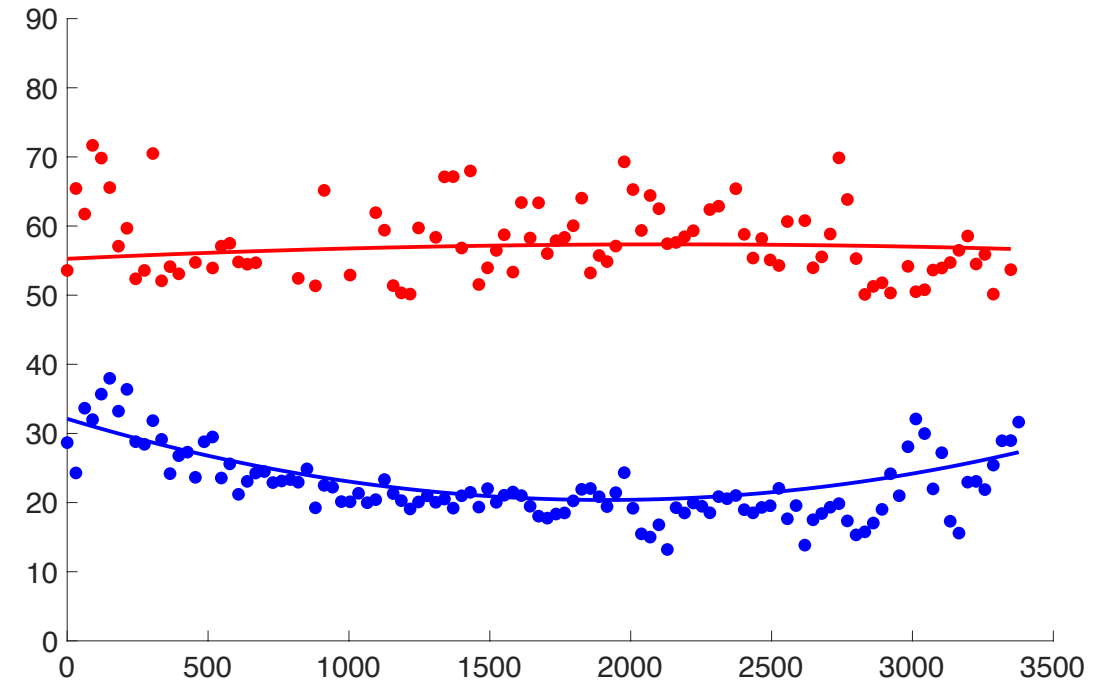
2. Example Application

2.5 Training processing



Average Error on **training data**: 2.70

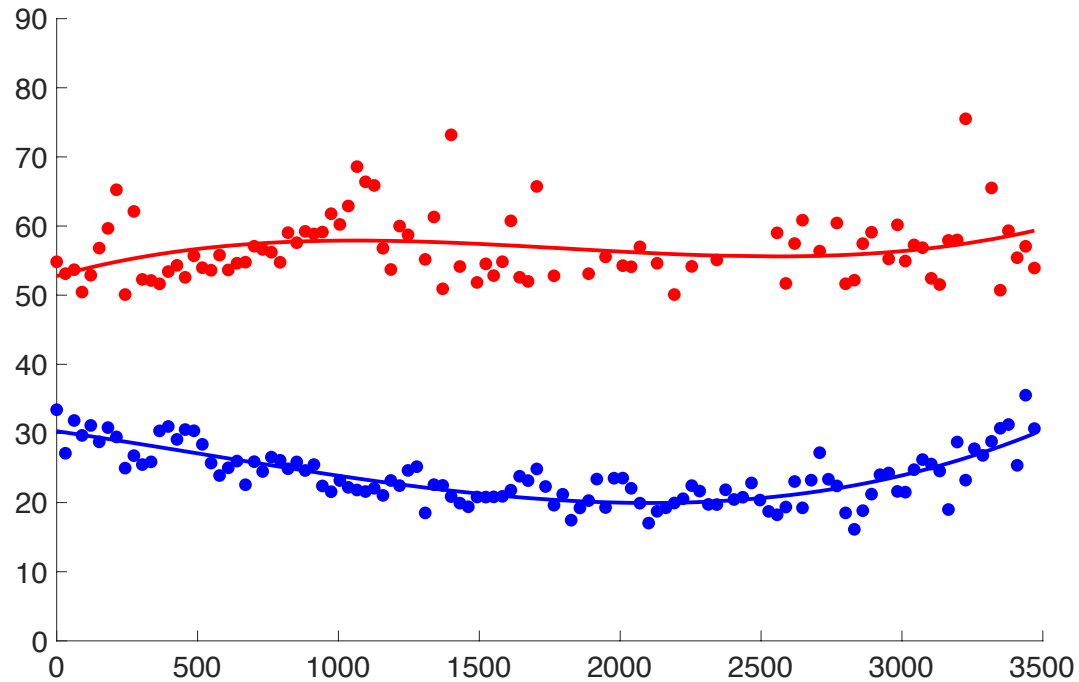
Redesign the Model



Average Error on **testing data**: 3.45

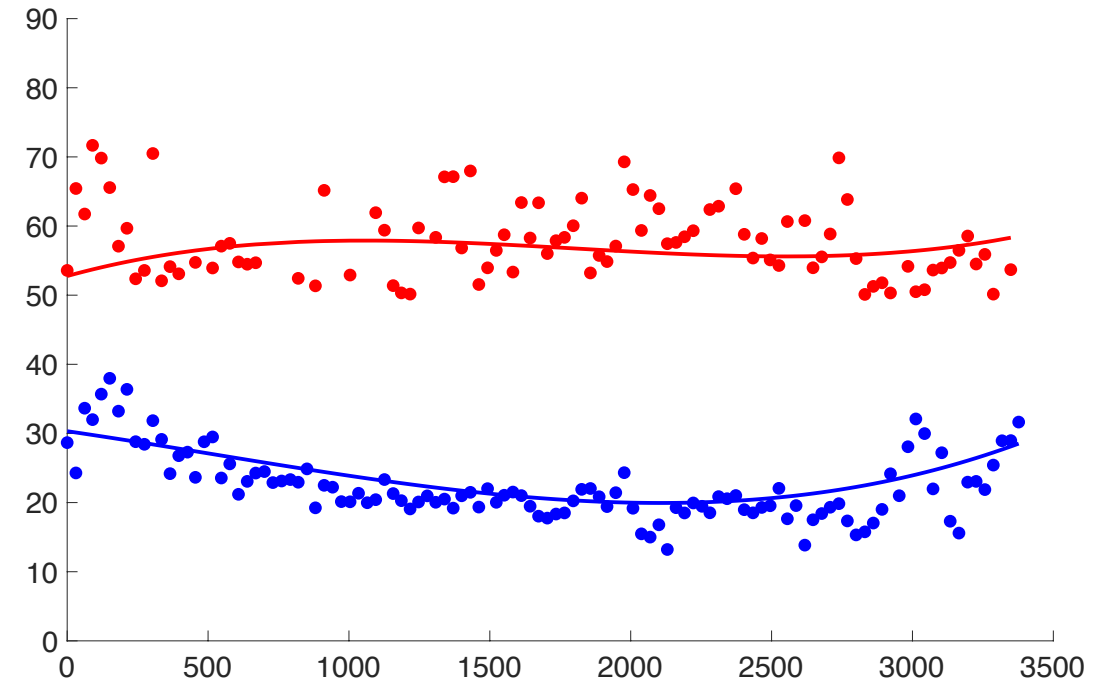
2. Example Application

2.5 Training processing



Average Error on **training data**: 2.63

Redesign the Model



Average Error on **testing data**: 3.61

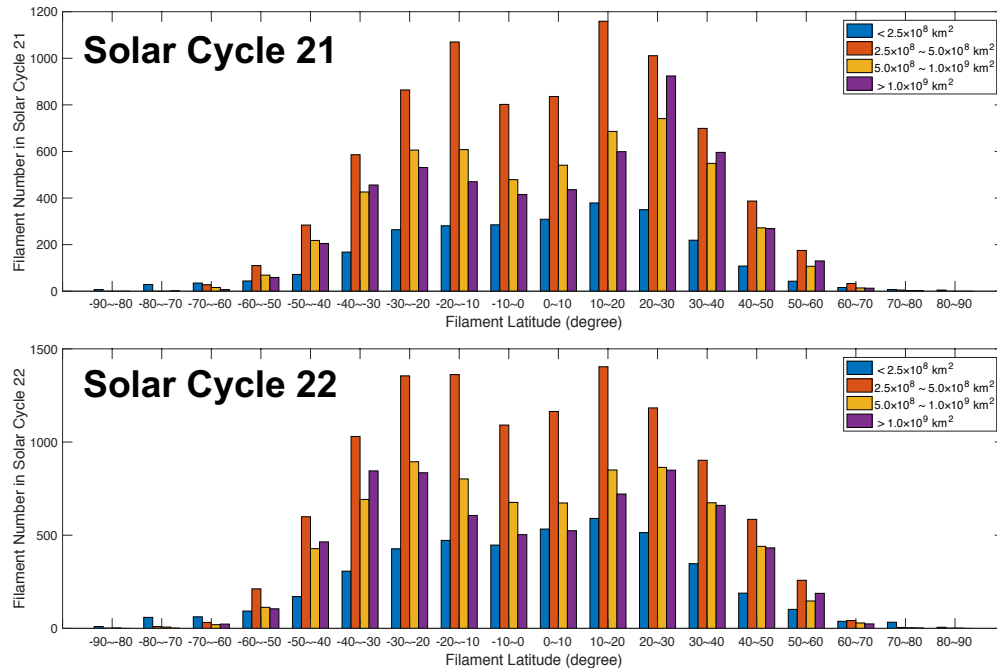
2. Example Application

2.5 Training processing

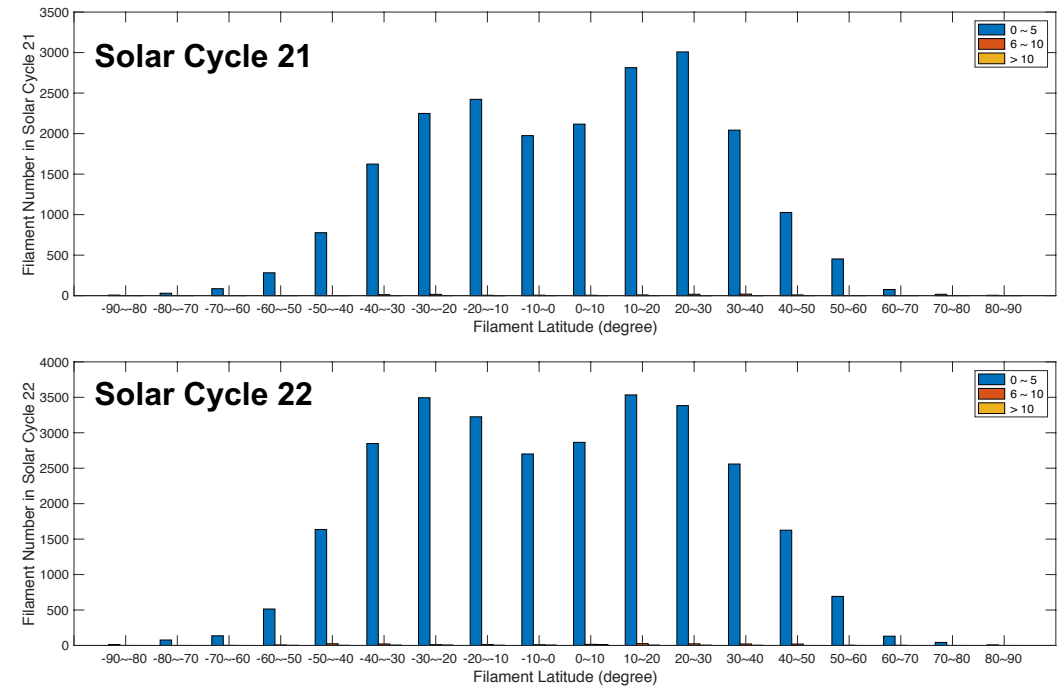
Is there any more hidden factors not considered in the previous model?

2. Example Application

2.5 Training processing



Area ??



Barb number ??

Regression

Hao, Qi

School of Astronomy and Space Science

THANKS

