CE9010: Introduction to Data Science

Lecture 5: Gradient Descent Tricks

Semester 2 2017/18

Xavier Bresson

School of Computer Science and Engineering
Data Science and AI Research Centre
Nanyang Technological University (NTU), Singapore

NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Outline

- Feature scaling
- Gradient descent with unbalanced scaling
- Feature normalization
  - Max normalization
  - Z-scoring
- Learning rate
- Stopping condition
- Conclusion

Outline

- **Feature scaling**
- Gradient descent with unbalanced scaling
- Feature normalization
  - Max normalization
  - Z-scoring
- Learning rate
- Stopping condition
- Conclusion

Feature scaling

- Data features with similar scaling:

  Example: pixels in images

  - $x_1 = \text{pixel}_1$ (0-255)
  - $x_2 = \text{pixel}_2$ (0-255)
  - Etc

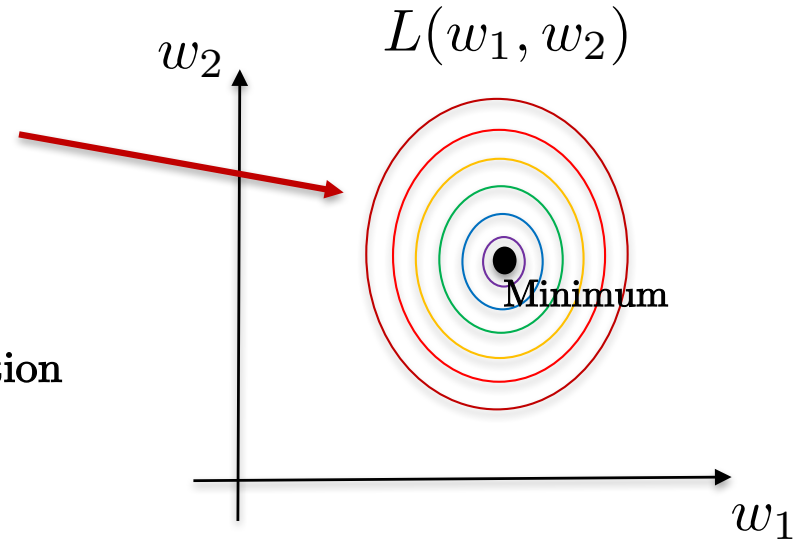
- Consequence: Loss landscape is well distributed along all variable directions.

$$L(w_1, w_2) = \frac{1}{n} \sum_{i=1}^{n} \left(w_1 x_{i(1)} + w_2 x_{i(2)} - y_i \right)^2$$

Loss function

$$f_w(x) = w_1 x_{(1)} + w_2 x_{(2)}$$

Predictive function



Loss landscape with x_1 and x_2 having similar scaling

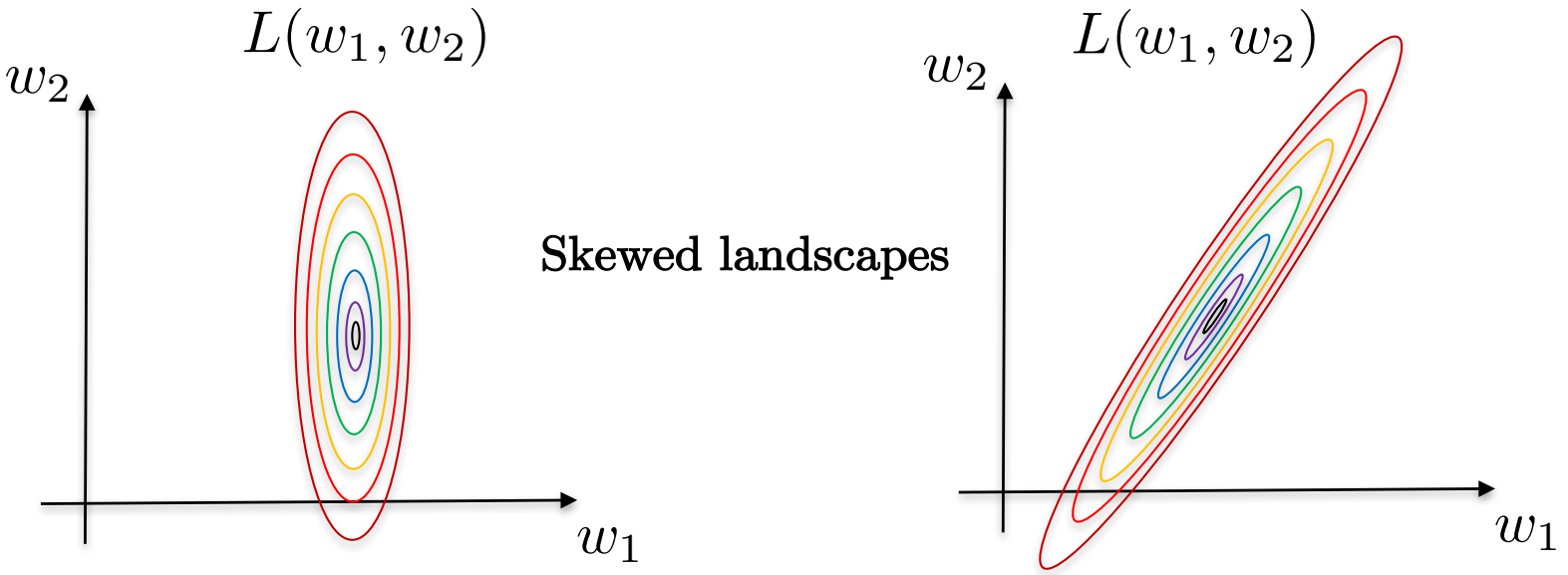
Unbalanced scaling

- Real-world data features may have different scales.

  Example: House pricing prediction

  - x_1 = house size (0-2000m^2)
  - x_2 = #rooms (1-5)

- Consequence: Loss landscape may be skew along some variable directions.



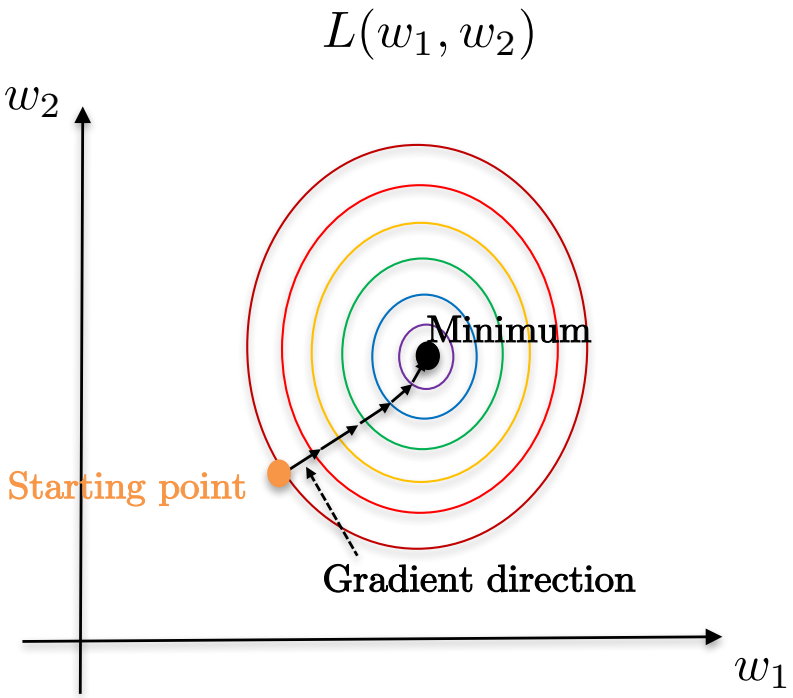
- What consequence on the gradient descent technique?

Outline

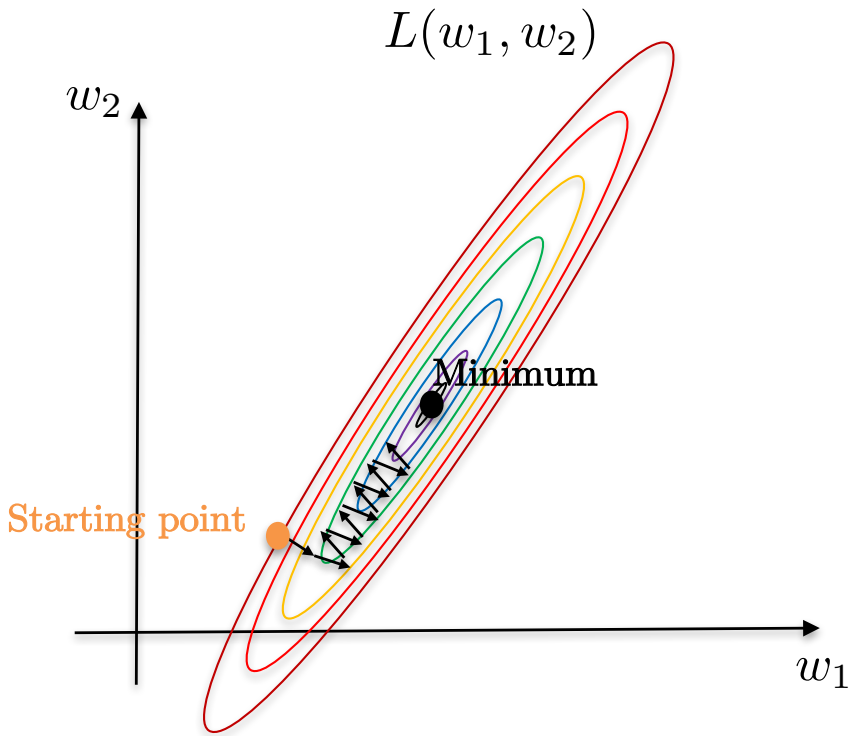
- Feature scaling
- **Gradient descent with unbalanced scaling**
- Feature normalization
  - Max normalization
  - Z-scoring
- Learning rate
- Stopping condition
- Conclusion

Gradient descent

- Reminder: GD follows the direction of the steepest descent, given by the gradient direction:



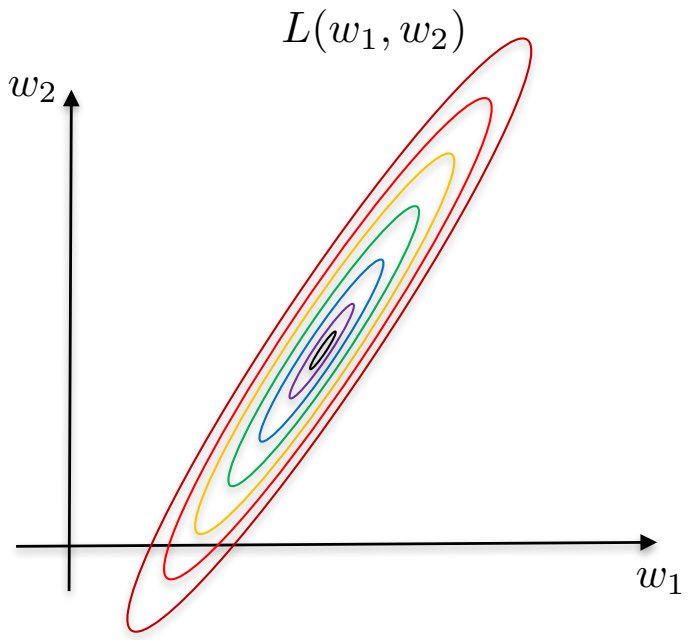
GD requires a few steps to converge



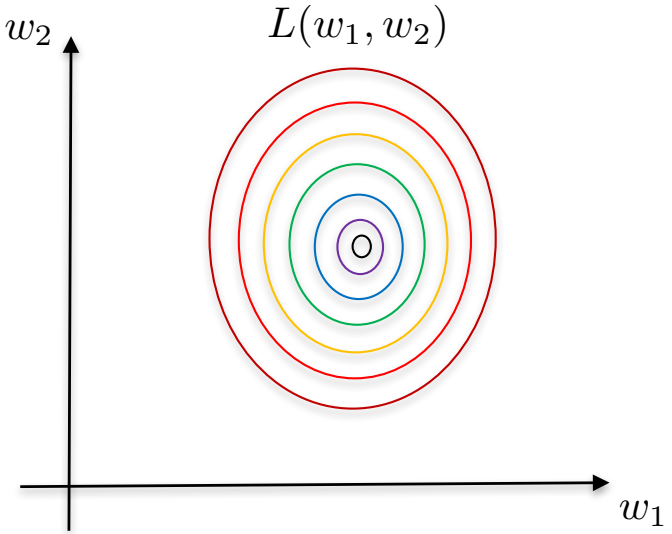
GD may require many steps to converge as its trajectory oscillates.

Normalization

- Scaling the features $x_{(i)}$ to the same order of values "flattens" the loss landscape:



Data normalization $\Rightarrow$



$x_1 =$ house size [0-2000](m^2)

$x_2 =$ #rooms [1-5]

$\Rightarrow$

$x_1 =$ house size [0,1]

$x_2 =$ #rooms [0,1]

Outline

- Feature scaling
- Gradient descent with unbalanced scaling
- **Feature normalization**
  - **Max normalization**
  - Z-scoring
- Learning rate
- Stopping condition
- Conclusion

Max normalization

- Most common data normalization: Normalize max value of x_i to 1.

$$x_i \leftarrow \frac{x_i}{\max_i x_i} \quad \Rightarrow \quad 0 \leq x_i \leq 1$$

- Example:
  - x_1 = house size 0-2000m^2 $\Rightarrow$ $x_1 = x_1/2000$
  - x_2 = #rooms 1-5 $\Rightarrow$ $x_2 = x_2/5$

- If x_i takes negative values, then

$$x_i \leftarrow \frac{x_i}{\max_i |x_i|} \quad \Rightarrow \quad -1 \leq x_i \leq 1$$

Outline

- Feature scaling
- Gradient descent with unbalanced scaling
- **Feature normalization**
  - Max normalization
  - **Z-scoring**
- Learning rate
- Stopping condition
- Conclusion

Z-scoring

- Most common statistical normalization:
  - Step 1: Center the data (zero-mean).
  - Step 2: Normalize data variance to 1 (unit-variance).

$$x_i \leftarrow \frac{x_i - \mu}{\sigma}$$

$$\mu = \text{mean}(x_i) = \frac{1}{n}\sum_{i=1}^{n} x_i$$

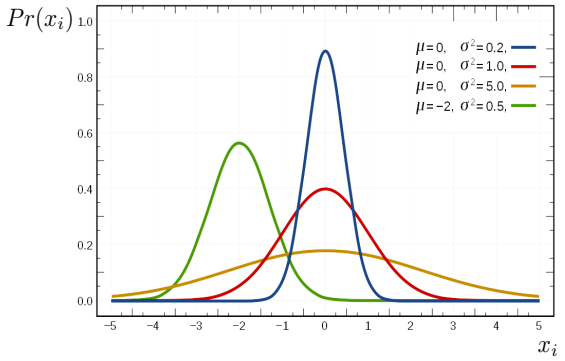
$$\sigma^2 = \text{variance}(x_i) = \frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)^2$$

$$\sigma = \text{standard deviation}(x_i) = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)^2}$$

Step 1: Center the data

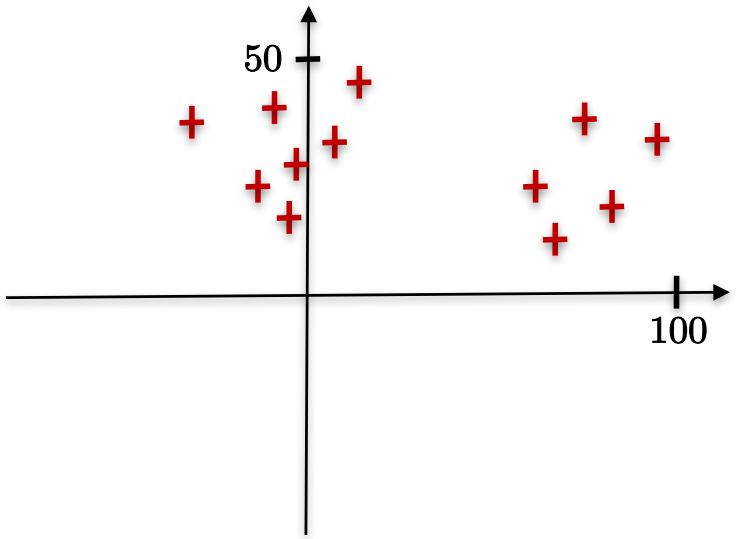
$$x_i \leftarrow x_i - \mu$$

- It is one of the most common pre-processing steps:
  - It allows comparison between data distributions (data exploration).
  - Part of data science pipeline: data acquisition, data cleaning (by exploration), data pre-processing, data analysis, decision.
  - Gradient descent works better if data is centered.
  - Step required to normalize the variance of data.



Step 1: Center the data

- Illustration:

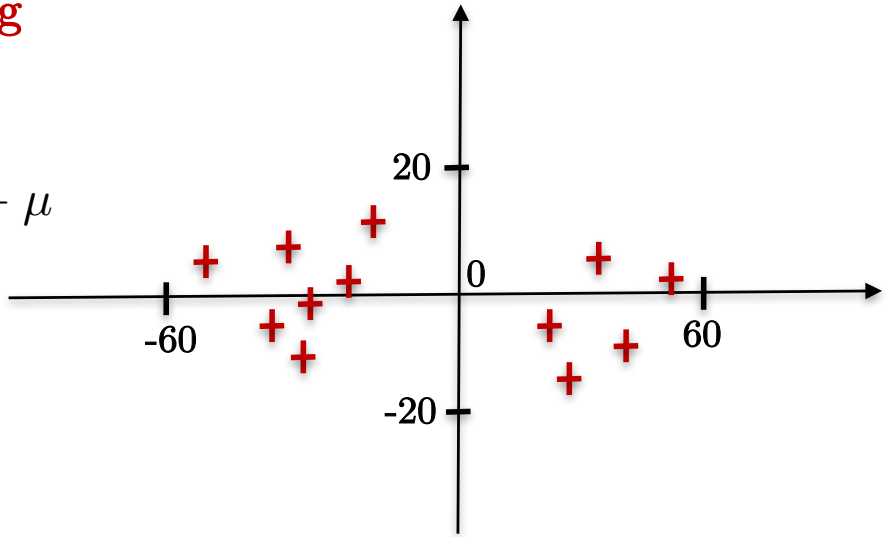


Data not centered

Data centering

$\Rightarrow$

$x_i \leftarrow x_i - \mu$



Data are centered

- Example:
  - x_1 = house size 0-2000m^2 $\Rightarrow$ $x_1 = x_1$-1000 (-1000$\leq x_1 \leq$1000)
  - x_2 = #rooms 1-5 $\Rightarrow$ $x_2 = x_2$-2.5 (-2.5$\leq x_2 \leq$2.5)

Step 2: Variance normalization

$$x_i \leftarrow \frac{x_i - \mu}{\sigma}$$

- The data variance is normalized to 1.
  - It allows easier comparison between data distributions (data exploration).
  - Sometimes, it decreases the performances of learning techniques.

Step 2: Variance normalization

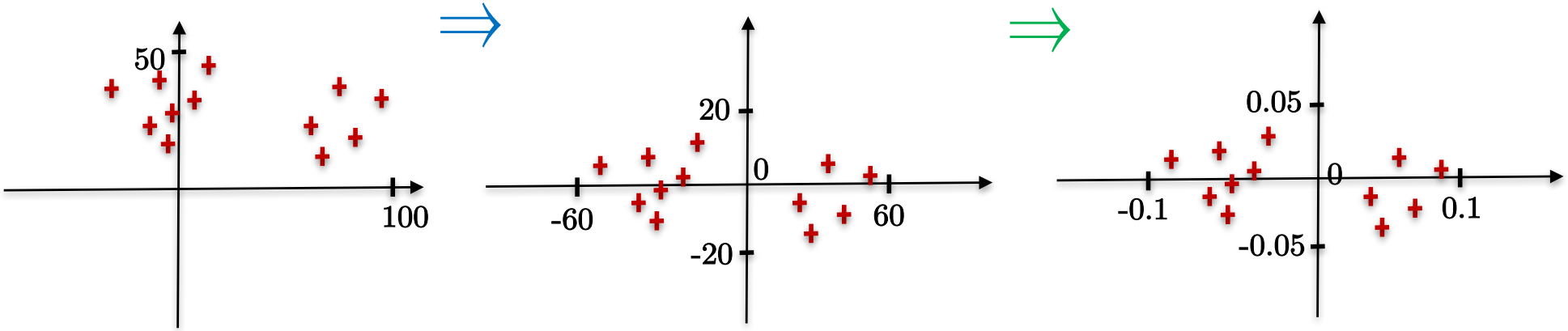
- Illustration:

Data centering

$$x_i \leftarrow x_i - \mu$$

Data Variance normalization

$$x_i \leftarrow \frac{x_i}{\sigma} \quad (\mu = 0)$$



- Example:
  - x_1 = house size 0-2000m^2 $\Rightarrow$ $x_1 = (x_1\text{-}1000)/250$ ($-0.1 \leq x_1 \leq 0.1$)
  - x_2 = #rooms 1-5 $\Rightarrow$ $x_2 = (x_2\text{-}2.5)/2$ ($-0.1 \leq x_2 \leq 0.1$)

Same scaling

Outline

- Feature scaling
- Gradient descent with unbalanced scaling
- Feature normalization
  - Max normalization
  - Z-scoring
- **Learning rate**
- Stopping condition
- Conclusion

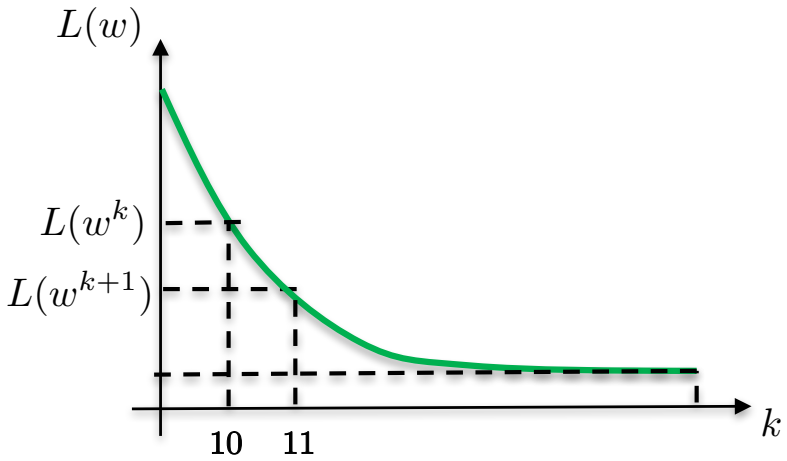
Learning rate

- Good choice of learning rate τ is essential:
  - Too small and the convergence takes lots of time.
  - Too large and the technique diverges.

- How to select a good value τ?
  - Monitor the loss decrease: The loss is guaranteed to decrease at each iteration:

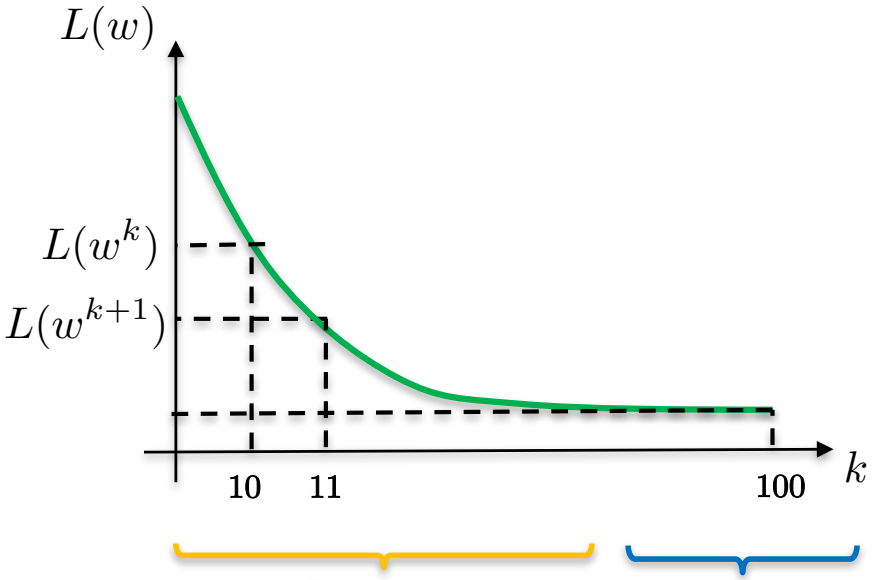
$$L(w^{k+1}) \leq L(w^k) \quad \forall k$$

k is the iteration index



  - The speed of loss decrease, a.k.a. convergence speed, should be as fast as possible.

Convergence



During the minimization:

$$w^{k+1} = w^k - \tau \frac{\partial}{\partial w} L(w^k)$$
$$L(w^{k+1}) \leq L(w^k) \quad \forall k \leq 100$$

At convergence (steady state):

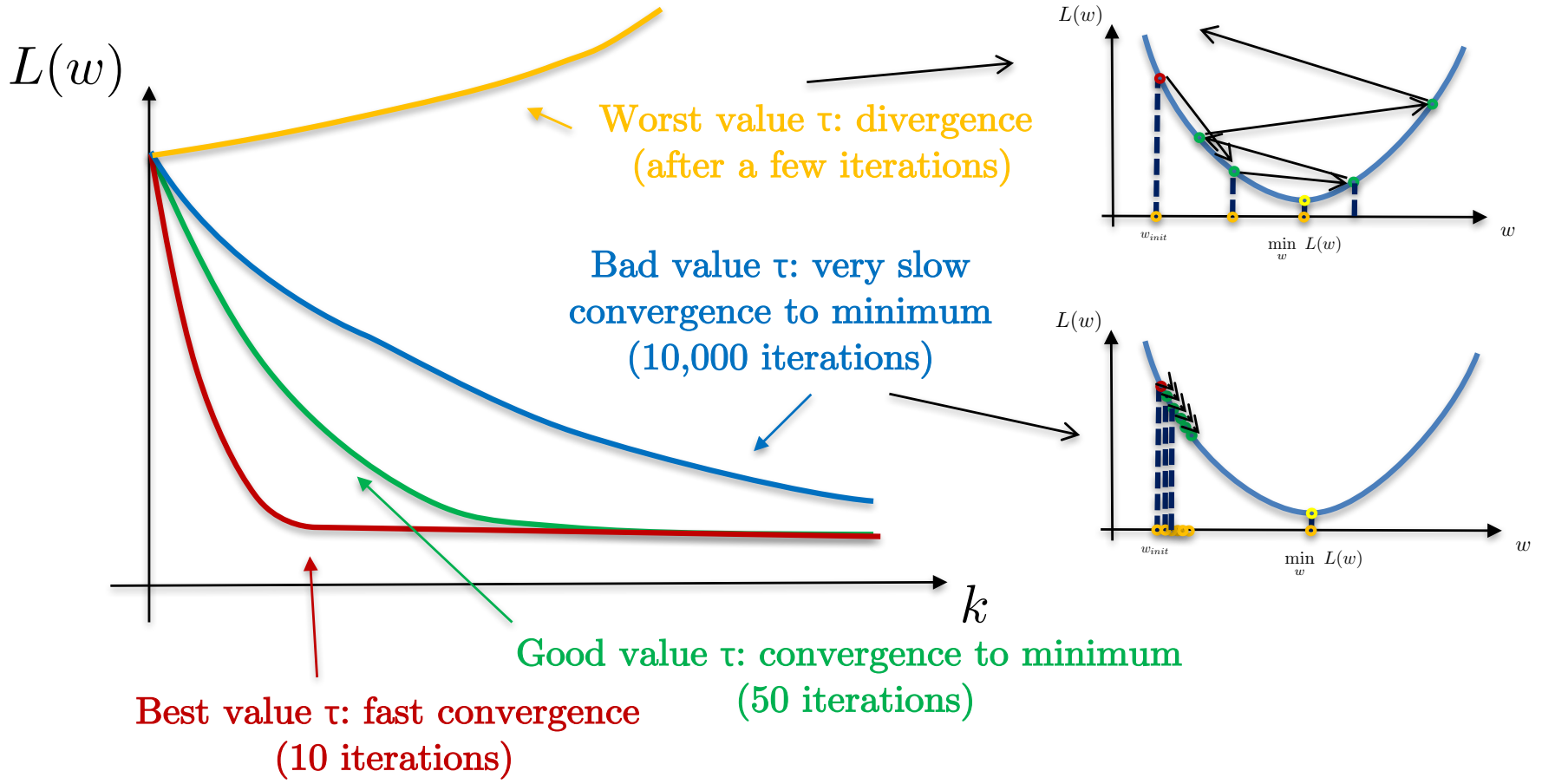
$$w^{k+1} = w^k$$
$$\cancel{w^{k+1}} = \cancel{w^k} - \tau \frac{\partial}{\partial w} L(w^k) \Rightarrow \frac{\partial}{\partial w} L(w^k) = 0$$
$$L(w^{k+1}) = L(w^k) \quad \forall k > 100$$

Definition of a minimum

- How to get to the flat region as fast as possible?

Convergence speed

- Goal: Find the best value τ that makes GD converges as fast as possible.



- Experimentally, test $\tau = 0.001 \quad 0.003 \quad 0.01 \quad 0.03 \quad 0.1 \quad 0.3 \quad 1 \quad 3 \quad 10.$

Outline

- Feature scaling
- Gradient descent with unbalanced scaling
- Feature normalization
  - Max normalization
  - Z-scoring
- Learning rate
- **Stopping condition**
- Conclusion

Stopping condition

- When to stop the iterative GD process?
  - Manually: Visual inspection of the loss decay. Stop when the steady state is reached.

    Example: Loss = 2.5 1.8 1.2 0.6 0.24 0.22 0.21 0.21 0.21 0.21

  - Automatically:
    - Mathematical, convergence is defined when

$$L(w^{k+1}) - L(w^k) \leq \varepsilon, \quad \varepsilon = 10^{-3}$$

Difficult to select

    - Data measure (like accuracy), convergence is obtained when error rate does not decrease after e.g. 10 iterations:

$$Acc(w^{k+1}) - Acc(w^k) \leq 5\%$$

Easy to select

Outline

- Feature scaling
- Gradient descent with unbalanced scaling
- Feature normalization
  - Max normalization
  - Z-scoring
- Learning rate
- Stopping condition
- **Conclusion**

Conclusion

- Data pre-processing is useful for
  - Gradient descent
  - Most data analysis techniques
- Most common data pre-processing:
  - Centering
  - Max or variance normalization
- Gradient descent technique limitations:
  - Skewed loss landscape
  - Learning rate hyper-parameter
  - Slow
- Improved GD techniques (used for neural networks):
  - Momentum, RMSprop, Adams

Questions?