Optimization

Optimization

- A network needs to have an input and an output
  - Two definitions for output:
    - Approximated output (“Predicted output” in TensorFlow syntax)
    - Labeled, target output
      - Target output calculated: model-free
      - Target output given: use labeled data
  - How close the predicted/approximated is from the labeled/target output is a **loss function,** and its value is the **loss value**
  - **Optimization**: using mathematical operations to reduce the loss value using gradients.

Optimization

- Loss functions (to calculate loss):
  - Take two parameters: Approximated/predicted output and target output
  - Some commonly used loss functions are:
    - MSELoss: (mean square error):
      - Typical for regression and approximation problems
      - Calculates differences between target and predicted/approximated outputs

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(y^i - \hat{y}^i\right)^2$$

    - Categorical Cross-Entropy: Typical for classification/categories problems

$$CEL = -\frac{1}{n}\sum_{i=1}^{N}\sum_{j=1}^{M} y_{ij} \cdot logp_{ij} \; or \; -\frac{1}{n}\sum_{i=1}^{N} log_{P_{model}}\left[y_i \in C_{yi}\right]$$

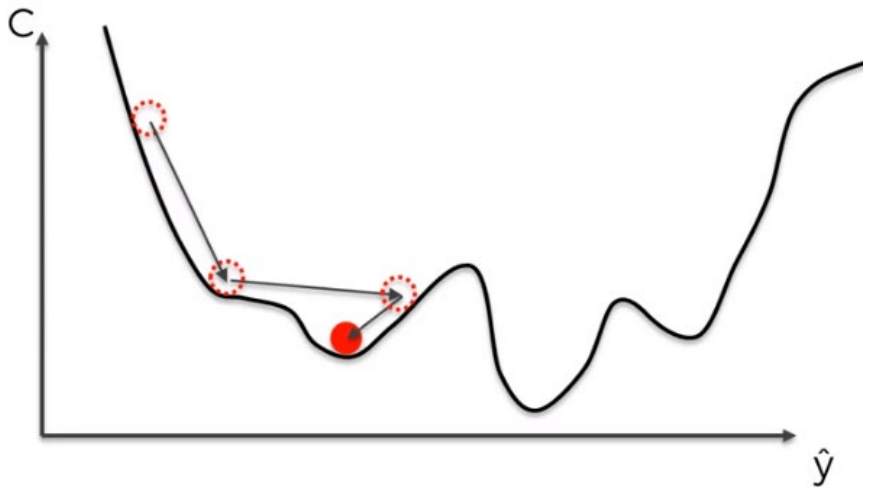
Optimization

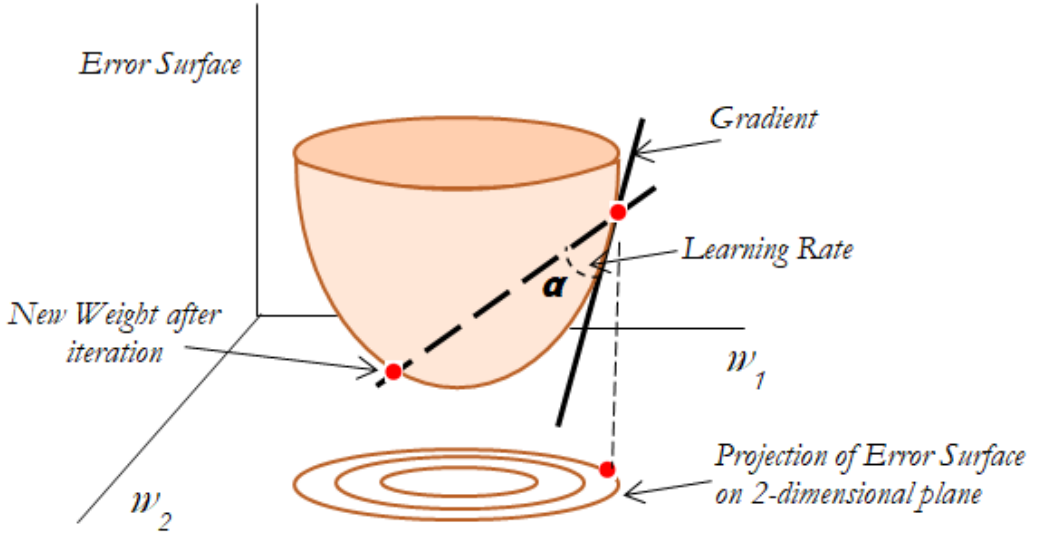
- Now that we have a loss value, we need to optimize to improve the model in training.
  - Gradients: vector that denotes the fastest increase and direction

- The optimization process:
  - Take the gradients of model parameters
    - Weights and biases
  - Change these parameters in order to decrease the loss value
  - Once the parameters are recalculated:
    - Update the parameters through **backpropagation**

Optimization

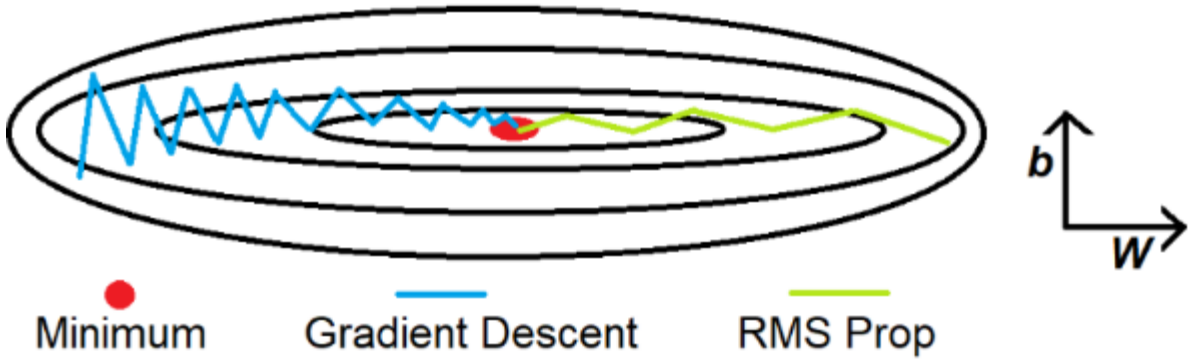
- Typical optimization methods:
  - SGD: stochastic gradient descent
    - Stochastic: randomly choose from the batch
    - **Variable parameter: learning rate**





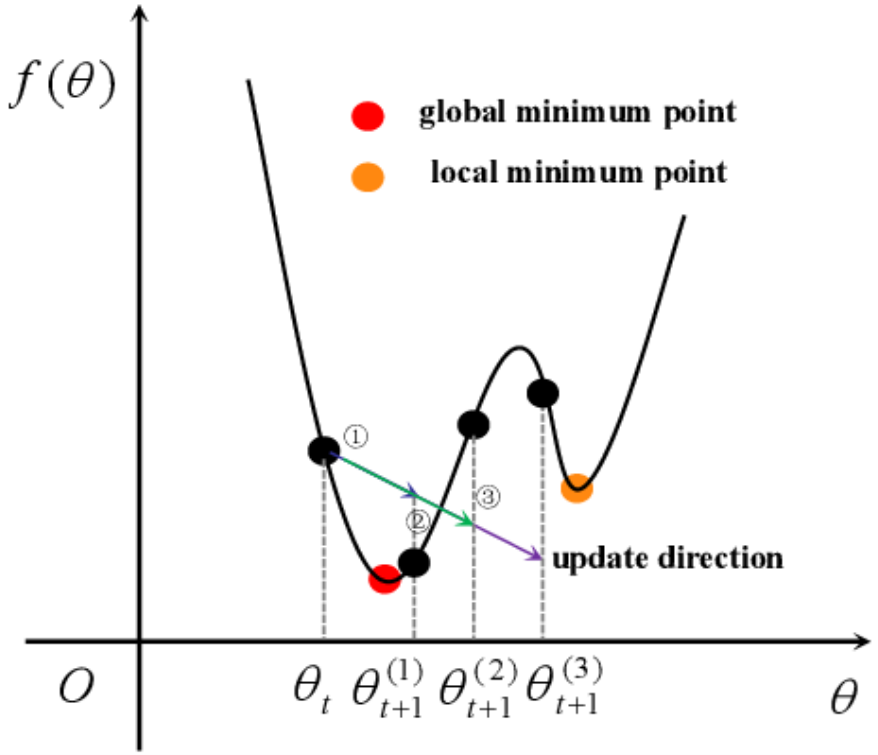
Optimization

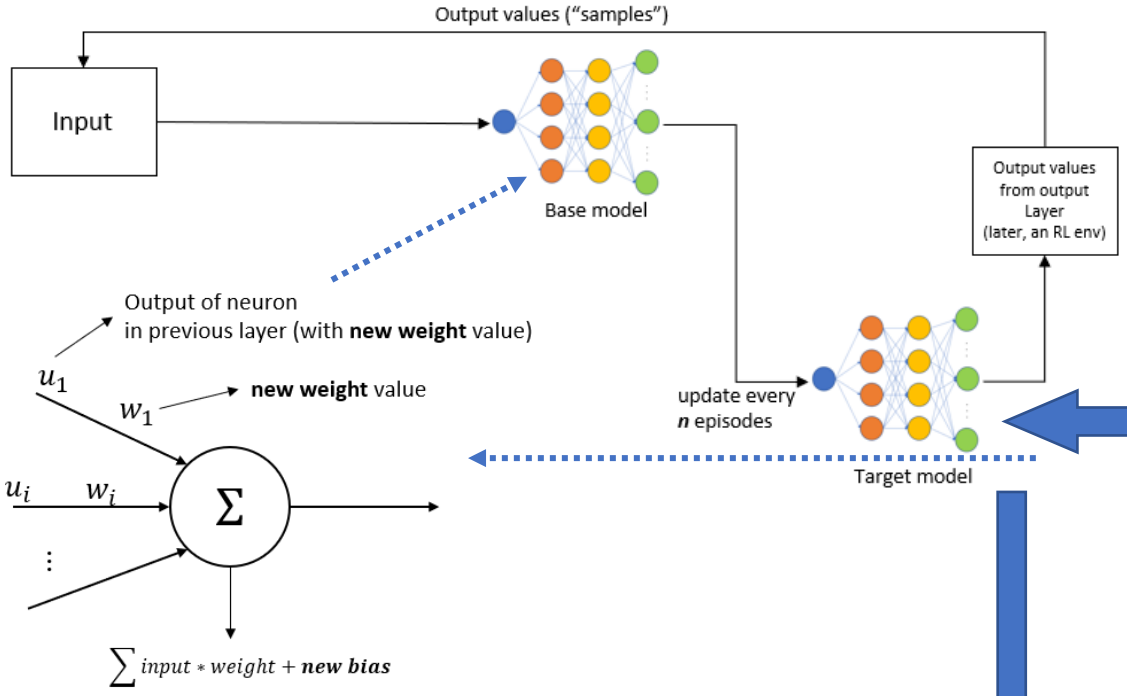
- RMSProp (Root mean squared propagation):
  - Based on AdaGrad (adaptive gradient)
    - Calculation of the step size (learning rate, LR) is “automatic”. Adapts the LR according to each parameter
    - Can slow down the process
  - RMSProp: uses a decay for learning rate using moving average (or weighted average)



Optimization

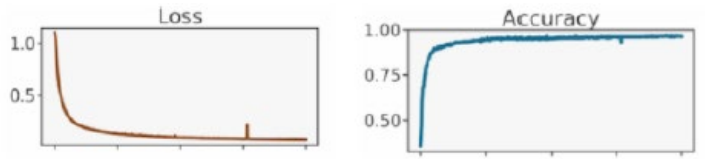
- Adam (adaptive moment)
  - Combination of AdaGrad and RMSProp
  - The learning rate is adapted in two calculations (moments):
    - Uses the average of the second moment (through the exponential moving average of gradients and square gradients)
    - Controlled decay: beta 1 & beta 2





backpropagate

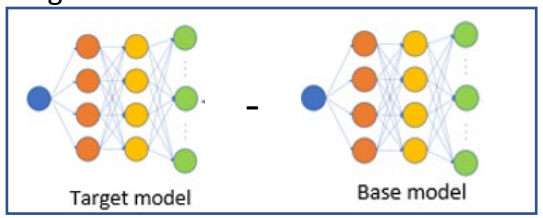
Loss (and accuracy) improved.
Update parameters now!



Using an optimization method

Using a loss function

Loss value =



Loss values are bad: Optimize!

THE UNIVERSITY OF
SOUTHERN MISSISSIPPI®