Introduction to Machine Learning

Komal Teru
Mila, McGill University

Topics to cover today

- **Machine learning fundamentals**
- **Python fundamentals**
- **Pytorch fundamentals (with an end-to-end training and evaluation regime)**

Machine learning fundamentals

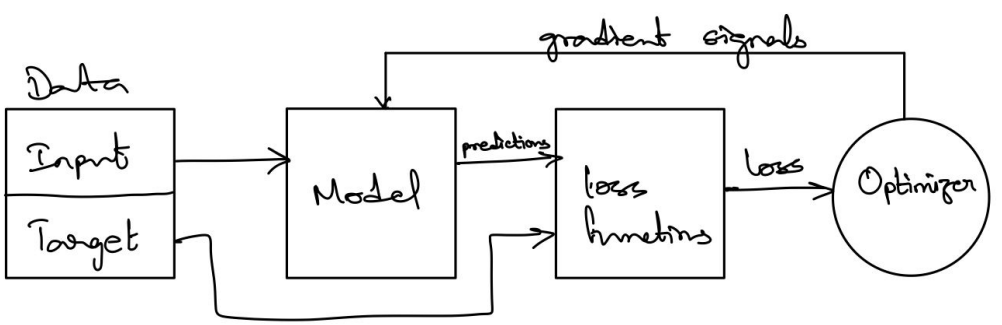
Central idea: **Tweak the model parameters to minimize a chosen objective.**

Hope is that minimizing this objective on training data will generalize to unseen data.

Machine learning fundamentals

Key ingredients of ML pipeline

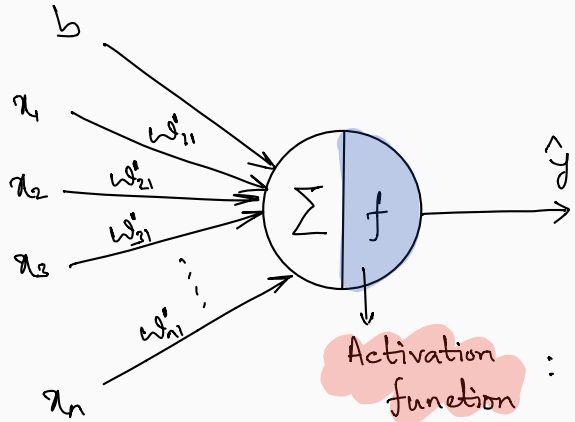
- Data : image, text, speech, graph etc.
- Model : MLP, CNN, RNN, etc.
- Loss function : MSE, cross-entropy, etc.
- Optimization algorithm (optimizer) : SGD, Adam, Adagrad, etc.



Machine learning fundamentals

Model: Multi layer perceptron

Perceptron as a computational block : Single scalar output



$$\hat{y} = f\left(\sum_{i=1}^{n} x_i w'_i + b\right)$$

Let $W_j^i = (w_{ij}^i, w_{ij}^i, w_{3j}^i, \ldots, w_{nj}^i)$

Activation function : Non-linear function;

Most popular: ReLU, Sigmoid

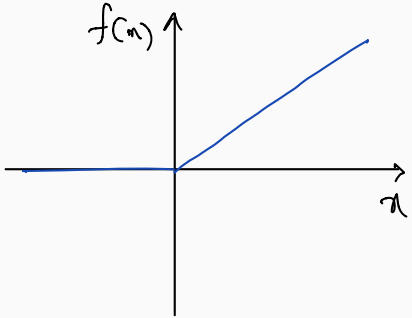
Machine learning fundamentals

Model: Multi layer perceptron

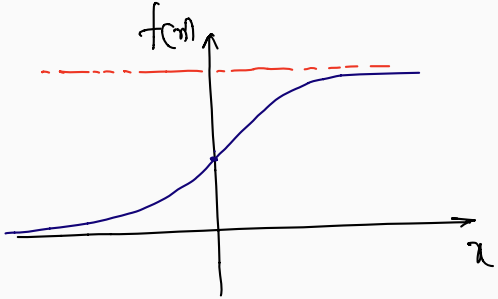
Perceptron as a computational block

Activation functions

ReLU: $f(x) = \begin{cases} x & \text{if } x > 0 \\ 0 & \text{if } x = 0 \end{cases}$



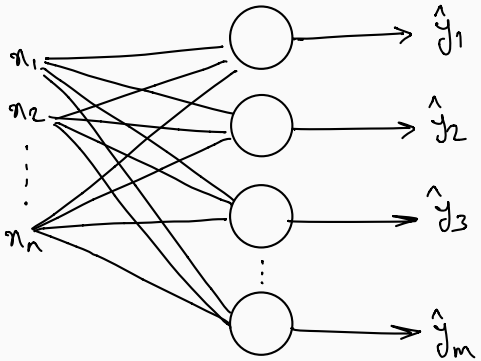
Sigmoid (σ): $f(x) = \frac{1}{1 + e^{-x}}$



Machine learning fundamentals

Model: Multi layer perceptron

Many perceptrons stacked together to form a layer : Multiple outputs



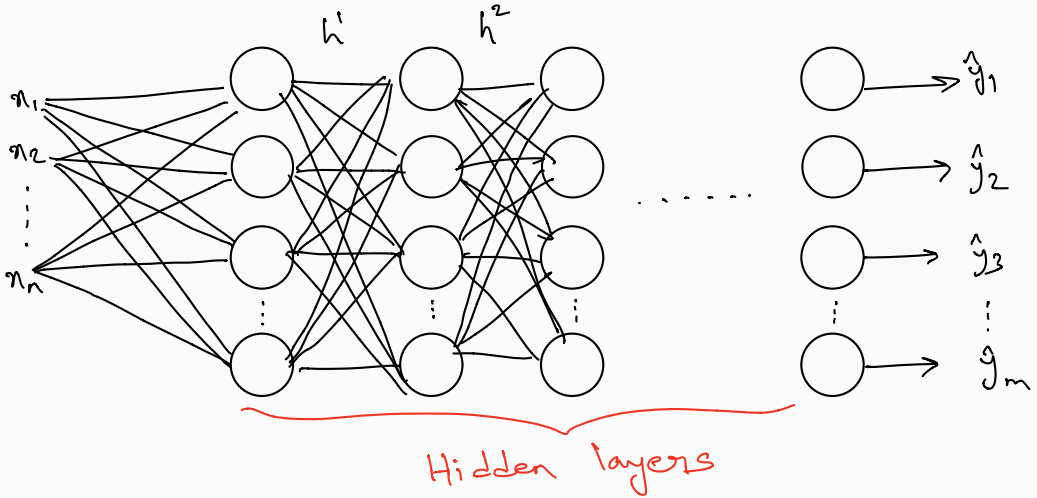
$$\hat{y}_j = f\left(\sum_{i=1}^{\hat{}} x_i w'_{ij} + b'_j\right)$$

Let $\vec{\hat{y}} = (\hat{y}_1, \hat{y}_2 \ldots, \hat{y}_m)$

Machine learning fundamentals

Model: Multi layer perceptron

Many perceptron layers stacked together



Machine learning fundamentals

Loss function

Mean squared error **(for regression task)**

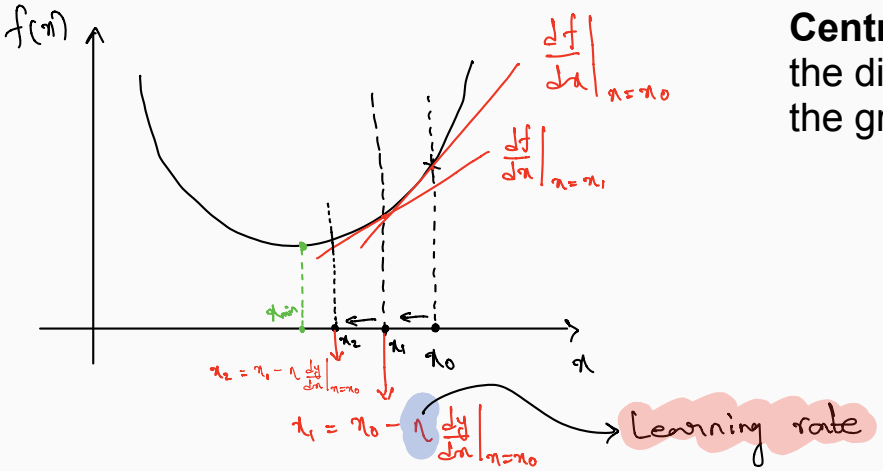
$$MSE = \frac{1}{N} \sum_{i=1}^{N} (\hat{y}_i - y_i)^2$$

Demo:http://neuralnetworksanddeeplearning.com/chap3.html#softmax

Machine learning fundamentals

Optimization algorithm

Stochastic gradient descent (SGD): Minimize a given parameterized function.

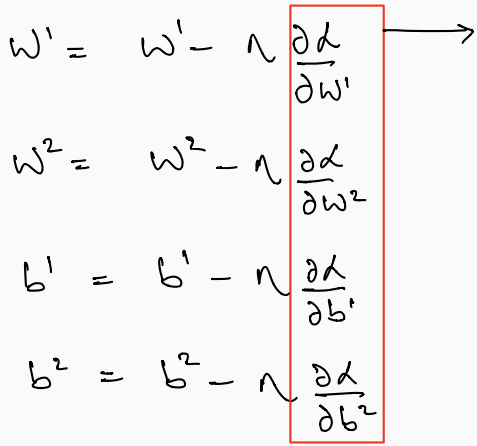


Central idea: Move in the direction opposite to the gradient.

Machine learning fundamentals

Optimization algorithm

Stochastic gradient descent (SGD): Minimize a given parameterized function.



→ Efficient way to compute this is by back-propagation

Machine learning fundamentals

Terminology

Mini-batches: Computing loss over the whole training dataset is computationally intensive. Sample **mini-batches** of data and update the model parameters for each mini-batch.

Epochs: One complete pass through the training dataset is counted as one **epoch.**

Hyper-parameters: Parameters which are not optimized by SGD, eg. learning rate. In the simplest case, these are hand tuned.

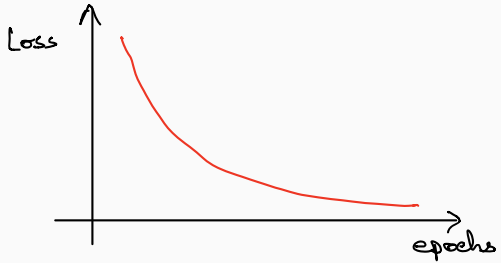
Machine learning fundamentals

Terminology

Data splits:

- Training data: Used to learn model parameters
- Validation data: Used to tune hyper-parameters
- Test data: Used to evaluate model parameters

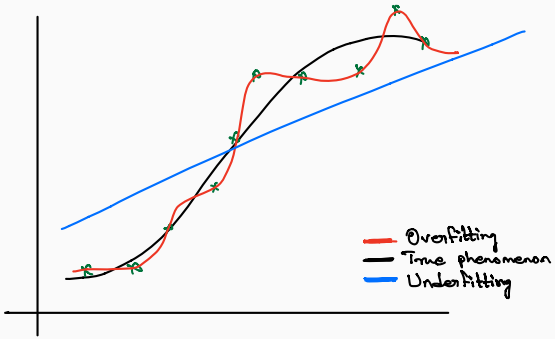
Learning curves: The plots of loss v/s epochs

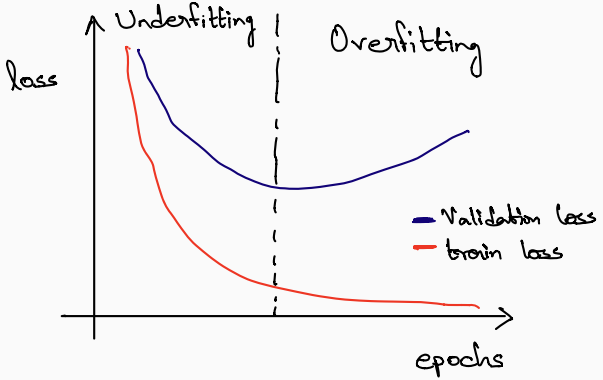


Machine learning fundamentals

Terminology

Overfitting/Underfitting: The hope was that minimizing lodd on training data would generalize to unseen data. That is not always the case.



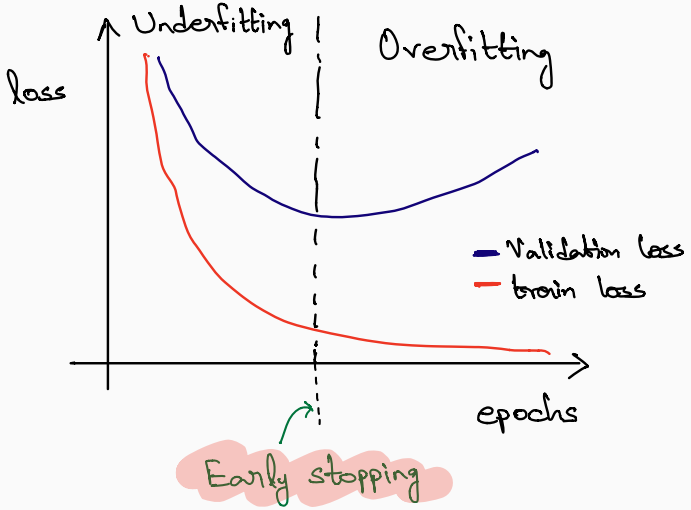


Machine learning fundamentals

Early stopping

Overfitting/Underfitting:

An easy way to mitigate overfitting.



Machine learning fundamentals

End-to-end pytorch demo

Link : https://colab.research.google.com/drive/1gBkCfQuhOgrW9L38hqGO-IMZxVBNw6f5