Table

Description automatically generated

Lecture 1

* Resurgence due to Big Data, Hardware, and Software
* Text, letter

  Description automatically generatedLecture 2
* Loss Function – Takes the error for a single training instance and applies additional properties to it (ex. squaring the error) / Cost Function - Aggregates the loss over all training instances to provide a single measure of how well the model is performing across the entire training dataset.
* Text

  Description automatically generated
* Advantages: Works well on ML Tasks, Often very scalable / Disadvantages: Might find a local minimum / Only applies to differentiable function
* Finding a good alpha – testing / adaptive learning rate / momentum
* true negative rate (TNR) / When we increase the recall rate by adjusting the classification threshold of a model, the precision rate is decreased. / high precision and high recall are what every model optimizes for. Consider the tradeoff, there is a balancing metric called F1 score that combines the two terms / FPR: The proportion of actual negative cases that the model incorrectly predicts as positive
* ROC – Receiver Operating Characteristic / TPR vs FPR / True Positive Rate = (TP/(TP+FN)) = Sensitivity/Recall / False Positive Rate = (FP/(Negatives)) = FP/(TN+FP) 1- TNR, Perfect Classifier has coordinates 0,1 / AUC = 1 / AUC = 0.5 is Random
* Logistic Regression / P(x) / 1-P(x) = e to the power of beta 0 + beta 1 times X. So, Log(P(X) / 1-P(X) = beta 0 + beta 1 times X is called the logit function / can be used for binary classification
* Problems of using linear regression for classification / Does not provide probabilities of classification / does not handle non linear decision boundaries
* Multi Layer Perceptron - An MLP is a specific type of ANN that is fully connected No matter how many layers you have, if all are linear, you can always collapse them into a single layer that is also linear. / Sigmoid (1 over 1 plus e to the minus x 1 /(1+e-x)) Pros Smooth Gradient Use for Probabilities [0,1] Cons Vanishing Gradient 🡪 Slow or never convergence / hyperbolic tangent ((ex – e-x)/ex + e-x)) [-1, 1], Pros Smooth Gradient, Cons Vanishing Gradient / ReLU (Rectified Linear Unit) (max(0,x)) [0,x] for x > 0 Pros reduces vanishing gradient, computationally efficient, Cons Dying ReLU – neurons become 0. Leaky ReLU, addresses Dying ReLU but is more complex / Softmax (softmax(xi) = exi/sum over j exj) For multiclass in final stage, Pros – outputs probabilities that sum to 1, can handle The output of the softmax is interpreted as the probability of getting each class.
* Diagram

  Description automatically generated
* Logistic cost function is log loss function
* Text, letter

  Description automatically generated
* Argmax / argmin are the arguments where function is its maximum or minimum

Lecture 3

* multiple classes unlike sigmoid, cons exponential causes numerical instability
* AND Gate – w1: 20, w2: 20, b1: -30, OR gate w1: 40, w2: 40, b1: -30, NOT gate w1:-20, b1: 10, XOR Gate w11: 1, w12: 1, b11: -1.5, b12: -0.5, w21: -1, w22: 1, b2: -0.5
* Sigmoid vanishing gradient: For values of z that are very large in magnitude (either positively or negatively), σ(z) approaches 0 or 1. This means the derivative approaches 0. Solve by using ReLU
* Exploding gradient – solve by clipping or batch normalization
* Dying ReLU problem: This issue often occurs in ANNs that use ReLU activation function when the parameters of a neuron are updated in such a way that the neuron only receives negative inputs, resulting in a zero gradient that prevents further learning.
* A picture containing text, clock

  Description automatically generated
* Batch gradient descent – each iteration, Stochastic Gradient Descent - Stochastic Gradient Descent (SGD) utilizes a single, randomly selected instance from the dataset per iteration. Can escape local minima, oscillating, potentially overshooting, explore the parameter space Mini batch is a subset of points in each step. Computational efficiency of stochastic with stability of batch
* Patience too small 🡪 stop due to noise, Patience too large 🡪 prevent early stopping, min\_delta – minimum value that is considered to be an improvement
* Done on validation set
* Adaptive learning rate, Momentum, Adam, Adam is an Adaptive learning rate that adjusts the learning rate for each parameter individually, taking into account the historical gradients.
* Feature scaling, min max (xi = (xi – min)/si – si is range max – min of xi. Simple but sensitive to outliers )and normalization (xi=(xi – mu)/sigma) less sensitive t outliers but not bounded
* Differentiability is essential for backpropagation

Lecture 4

* Simple 🡪 high bias and underfitting, complex 🡪 high variance and overfitting
* Regularization techniques
* Table

  Description automatically generated
* :
* A picture containing table

  Description automatically generated
* Patience too small 🡪 stop due to noise, Patience too large 🡪 prevent early stopping, min\_delta – minimum value that is considered to be an improvement
* Done on validation set
* Adaptive learning rate, Momentum, Adam, Adam is an Adaptive learning rate that adjusts the learning rate for each parameter individually, taking into account the historical gradients.
* Feature scaling, min max (xi = (xi – min)/si – si is range max – min of xi. Simple but sensitive to outliers )and normalization (xi=(xi – mu)/sigma) less sensitive t outliers but not bounded

Lecture 5

* Adam is an Adaptive learning rate that adjusts the learning rate for each parameter individually, taking into account the historical gradients.
* CNNs are a category of deep neural networks specialized for analyzing visual imagery. / Local Connectivity: Unlike traditional neural networks, CNNs connect each neuron to only a local region of the input data, capturing spatial relationships and local patterns. Filter of size 4x4 :16 different weights / This“patchy” operation is convolution
* Diagram

  Description automatically generated

Diagram

Description automatically generated

Padding refers to the process of adding layers of zeros (or other values) outside the original image boundaries before applying the convolution operation. / With padding, we can apply convolutions to an image and keep the output dimensions either the same or adjust them according to our needs. / Stride Definition: The stride is the number of pixels we skip when we move the filter across the input image. A stride of 1 means the filter moves one pixel at a time. Stride greater than 1 is a dimensionality reduction

Input is n x n, filter is f x f, padding is p, and stride is s 🡪 output is lb(((n + 2p -f) /s) + 1) x lb(((n + 2p -f) /s) + 1) lb is lower bounds

Pooling takes an area and reduces it to a number. Could be max value or average value, lb(((nh-f)/s)+1) x lb(((nw-f)/s) +1) x nc

Input -> Convolution / ReLU -> Pooling -> Convolution + ReLU ->Pooling -> Flatten -> Fully Connected -> Softmax

This is Feature Learning flowed by Classification (Flatten on)

Pooling pro quick dimension reduction, non parametric con potential special information loss strided convolutions pro redices dimension, captures complex patterns cons more parameters, computationally intensive

The primary purpose of the convolution operation is feature extraction.

In essence, convolution extracts features, padding preserves spatial dimensions (especially at the borders), max pooling provides spatial invariance and reduces dimensions, and strides can be used to control the reduction in dimensionality during the convolution operation.