Lecture 3. Multilayer Perceptions

- Outline
  - Multilayer fully connected neural netw-orks
  - Model selection, overfitting and undefitting
  - Regularisation methods: weight decay
- Limitations of Linear models
  - Assumption of monotonicity: outputs increase ~~parp~~ proportionally with any feature
  - Limited capacity: can only separate linearly separable data
    - A data set is linearly separable if for each class there is a hyper plane to separate this class from the others
  - To increase model capacity, one needs to introduce non-linear models
- From Single Layer to Multiple Layers
  - To overcome limitations of linear models one can introduce hidden layers
- From Linear to Nonlinear
  - Activation function σ is applied to

each hidden unit following the linear transformation
- With activation func.s in place, no longer going to collapse our MLP into a linear model
- To build more general MLPs, we can continue stacking such hidden layers

- Universal Approximators
  - MLPs are universal approximators which means they can approximate nonlinear functions
- Activation Functions
  - Decide whether or not a neuron should be activated
- ReLU Function
  - $\text{ReLU}(x) = \max(x, 0)$
  - Retains only non-negative elements;
  - Sets corresponding activations to 0
  - Derivative: 0 when input is -ve, and 1 when input is +ve
  - Not differentiable when input is 0;
  - In practice, set derivative to 0
  - Variants: parameterised ReLU (pReLU)
    - $\text{pReLU}(x) = \max(x, 0) + \alpha \min(x, 0)$
    - α: hyperparameter, some algo.s learn it
- Sigmoid Functions
  - $\text{sigmoid}(x) = \frac{1}{1 + \exp(-x)}$, $\exp(-x) = e^{-x}$
  - $\frac{d}{dx} \text{sigmoid}(x) = \frac{\exp(-x)}{(1 + \exp(-x))^2} = \text{sigmoid}(x)(1 - \text{sigmoid}(x))$

- Tanh Function
  - Similar output to sigmoid
  - $\tanh(x) = \frac{1 - \exp(-2x)}{1 + \exp(-2x)}$
  - $\frac{d}{dx}\tanh(x) = 1 - \tanh^2(x)$
- Summary
  - MLP adds one or multiple fully-connected hidden layers between the input and output layers and transforms the output of the hidden layer via an activation function
  - Commonly used Activation Functions:
    - ReLU
    - pReLU
    - sigmoid
    - tanh → far from centre learns slower ($\frac{d}{dx}$ towards 0)
  - With each of these activation functions MLPs with a single hidden layer are universal approximators
- Underfitting, Overfitting, Model Selection
  - Goal of ML is to disover patterns that *generalise* from training data to unseen data
  - In learning of the models, rely on training data only
  - Training data is usually a sample of the data with some underlying distribution
  - Overfitting: danger of fitting training data too well that causes failure to generalise
  - Regularisation: techniques to combat overfitting

- Models feature some hyperparameters
  - hyperparameters: parameters which define learning framework
  - parameter of the model: parameters which define a model in a general learning framework
  - Weights of linear regression or neural networks are parameters
  - Learning rate, epoch number, regularisation number, number of neurons, are hyper-parameters
- In practice, split available data into subsets:
  - training set
  - validation set
- Training set is used to learn parameters of model
- validation set is used to select the hyper parameters
- Training Error and Generalisation Error
  - training error is the error of our model calculated on the training set
  - generalisation error is the expectation of our model's error when we apply the model to an infinite stream of additional data samples drawn from the same underlying data distribution as our original sample
  - We estimate generalisation error by applying our model to an independent test set

- Stastical Learning Theory
  - i.i.d. assumption: both the training data and the test data are drawn independently from identical distros
  - Simple models and abundant data: we expect generalisation error to resemble training error
- Common Factors of Generalisation
  1) Number of tunable parameters
  2) Values taken by parameters
     - When weights take a wider range of values, model is more likely to overfit
  3) Number of training examples
- Model Section
  - Process to select model after evaluating several candidates
  - Validation dataset is used if a large amount of data is available
  - Use cross-validation if data is small
- Underfitting or Overfitting
  - Generalisation gap: difference between training error and validation error
  - Underfitting: substantial training error, small generalisation gap
    - Model is too ~~small~~ simple to reduce training error
  - Overfitting: small training error, large generalisation gap
- Weight Decay
  - Simple regularisation method
  - Same as ridge regression

"Do you understand your options and how they work?"
- Theory question philosophy
- Use norms of weight vector to measure complexity of linear model
- Add its norm as a penalty term in the loss function
- Least squared error plus L_2 norm penalty : ridge regression
- $\lambda = 0$: reduces the LSE
- $\lambda > 0$: restricts size of $\|w\|$

• Implementation
- Why called "weight decay"
  • Given penalty term alone, the optimisation algorithm decays the weight at each step of training

• Summary
- Regularisation is a common method for dealing with overfitting.
- It adds a penalty term to the loss function on the training set to reduce complexity of the learned model
- One particular choice for simplification is weight decay using an L_2 penalty
- This leads to weight decay in the seuccessive steps of the learning algorithm.