Intro to Machine Learning Notes

Basic Introduction (very surface level)

- Process of training a model to make useful predictions/generate content from data
- Major categories include: supervised, unsupervised, reinforcement learning, generative AI
- Supervised – give the model data with the 'correct answers' and let it learn
    - Regression – numeric value output; Classification – probability of being a certain category of data
- Unsupervised – no 'correct' answers given for the data, tries to spot patterns
    - Clustering – demarcate the data into natural groups that form to try to learn the pattern
- Reinforcement learning – rewards or penalties based on success/failure
- Generative AI – creating content (text, image, video etc.) based on input
- Best datasets for supervised learning are large and highly diverse

Linear Regression

In ML context, relationship between features (input) and a label (output) is supervised learning. The features can be represented as vector $\mathbf{x}$ and the labels as $\mathbf{y}$ – the data comes as $\mathbf{X} = (\mathbf{x_1}, \mathbf{x_2}, \ldots)$.

- Hypothesis – $y_i = \mathbf{w}^T\mathbf{x}_i + b$ – the $\mathbf{w}$ vector is the weight, coefficients of the model
- Loss function – you can choose a function we want to minimise between the model and the observed data – usually this is the MSE function, can be others
- $\mathbf{w} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$ is the optimal solution analytically – best linear unbiased estimator (Gauss-Markov Theorem)
- In order to minimise loss, usually have to use gradient descent method
- $\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} - \eta\nabla_{\mathbf{w}}\mathbf{L}(\mathbf{w},b)$
- Iteratively reduce the error for $\mathbf{w}$, same for b. η is the learning rate – hyperparameter
- Hyperparameters are the parameters in the model that you set – learning rate for example, needs to be right to ensure quick convergence, but not too high to fail to converge
- Batch size – number of samples the model looks at before updating weights – stochastic gradient descent has batch size 1, mini-batch SGD is batch size between 1 and N

Logistic Regression

- Similar idea, except this time the linear model is $z_i = \mathbf{w}^T\mathbf{x}_i + b$, and now the labels are given by the logistic function

$$y_i = \frac{1}{1 + e^{-z_i}}$$

- This takes values in the range (0,1)
- Loss function considered is often the log loss function instead
- Regularization is v important for this regression in particular – handles overfitting of data, especially w/ lots of noise or features
- L2 regularization – extra term in the loss function that limits very large coefficients etc.

Neural Networks

- Family of model architectures that find non-linear models in data – this is done in hidden layers, which are extra layers between the input and the output. The nodes between these layers are called neurons
- The output for each node from one layer to the next is determined by the activation function – this can be linear or other common ones include logistic, ReLU, tanh, etc.
- Then each individual neuron can be calculated using that layer's weights:

$$z_i^{(l)} = \sum_j w_{ij}^{(l)} a_j^{(l-1)} + b_j^{(l)}$$

Intro to Machine Learning Notes

- Most common training method – backpropagation – calculate losses working backwards through the network (use chain rule and usually SGD
- Backpropagation usually has issues with vanishing, exploding gradients, need to choose activation functions appropriately and need regularization
- Types: Feedforward Neural Network (FNN) – simplest type, no loops or feedback; Convolutional Neural Network (CNN) – use convolutional layers, specifically designed for grid like data e.g. images; Recurrent Neural Networks (RNN) – used for sequential data e.g. time series, has loops, variants include Long Term Short Memory (LSTM)