Lecture 2. Linear Regression and Classification

- Linear Regression
  - Assumption: output depends on inputs linearly
    - Means that assuming a feature with a positive weight, the larger the feature the larger the output
  - Linear model: $y = w_1x_1 + w_2x_2 + \dots w_nx_n + b = w^Tx + b$
  - w_i = weights
  - b = bias
    - Means that if all features are 0, what is the output?
- Objective function
  - Goal of LR: choose weights and ~~ar~~ bias to make model fit observed data
  - Quality measure for some given model : Least square loss
- Linear Model
  - Linearity assumption: the target (price) can be expressed as a weighted sum of features (area, age,):
    - Weights influence each feature on the prediction of target
    - Bias: b
    - Affine transformation: a linear transfo-rmation of features via a weighted sum, combined with a translation via the added bias
- Overall Loss Function
  - Training aims to find parameters that minimise total loss across all traini-ng examples

- Ridge Regression
  - Loss function with regularisations
  - Essentially just involves modifications to loss function
  - λ: regularisation number
- Linear Regression from Scratch
  1) Generate synthetic data
  2) Read the data
  3) Initialise model parameters
  4) Define the model
  5) Define the optimisation ~~algorithmx~~ algorithm
  6) Train
  7) Estimate errors
- Training (I)
  - Initialise parameters
  - Repeat until done:
    - Compute gradient
    - Update parameters
  - Each epoch iterates through the entire dataset
- Training (II)
  - Generate predictions by calling net(x) and calculate the loss, l (the forward propagation)
  - Calculate gradients by running the backpropagation
  - Update the model parameters by invoking our optimiser
  - For good measure:
    - Compute loss after each epoch
    - Print it to monitor progress

- Softmax Regression for Classification Problem
  - Predict categories such as cat, dog and cow
- Network Architecture
  - Multiple outputs, one per class
  - Multiple linear models, one per class
- Softmax Operation
  - Interpret the outputs of the model as probabilities
    - Any output $\hat{y}_j$ is interpreted as the probability that a given item is in class j
    - Choose the class with the largest output value as our prediction $\operatorname{argmax}_j y_j$
    - The probabilities should be non-ve and sum to 1
  - For prediction we pick the most likely class
  - Although softmax is a nonlinear function, the outputs of softmax regression are still defined by an affine transformation of input features
  - $\therefore$ SR is a linear model
- Loss Function
  - Measure the quality of the predicted probabilities
  - Maximum likelihood estimation
  - Cross-Entropy loss
- Softmax and Derivatives
  - The derivative is the difference betwe

- en the probability assigned by the model and elements in the one-hot label vector
- Similar to what happens in regression where the gradient was the differenc between the observation and est.

- Implementation of Softmax Regression
  - Given in book