BIS557 Project Proposal

Yijun Yang, MPH’21

Problem Definition

This project will be aimed at implementing backpropagation in a three-layer neural network, and showing how to use artificial neural networks for multi-classification.

Data Source

A simulated spiral data will be used for classification. The input $X \in \mathbb{R}^{N \times 2}$ will be 2-dimension. The response will be $y \in \mathbb{R}^N$. Here, y is the category indicator. The following code shows one way to construct the simulated dataset.

Reference: Phyllotaxis - Draw flowers using mathematics

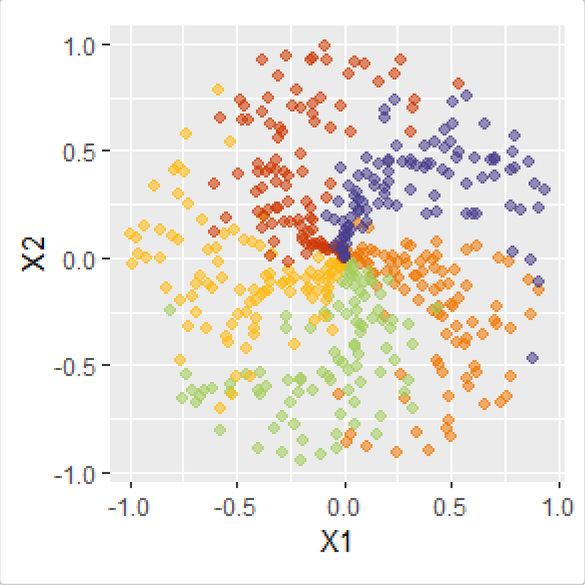
```
# number of categories
K <- 5
# number of points for each class
N <- 100
# initialize X and y
X <- matrix(0, nrow = 0, ncol = 2)
y <- c()

# simulate dataset
set.seed(2020)
for (i in 1:K) {
  r <- seq(0, 1, length.out = N)
  t <- seq(i*2*pi/5, (i+1)*2*pi/5, length.out = N) +
    rnorm(N, sd = 0.35) # add some noise
  X <- rbind(X, cbind(r*sin(t), r*cos(t)))
  y <- c(y, rep(i, N))
}

df <- as.data.frame(cbind(X, y))

# set plot images to a nice size.
# options(repr.plot.width = 15, repr.plot.height = 15)

ggplot(df, aes(x = df[,1], y = df[,2])) +
  geom_point(aes(color = as.factor(y), alpha = 0.9), size = 3, pch = 20) +
  scale_color_manual(values=c("darkorange2", "darkolivegreen3", "darkgoldenrod1", "orangered3",
  "darkslateblue")) +
  theme(legend.position = "none") +
  labs(x = "X1", y = "X2") +
  coord_fixed()
```



Methods

Activation function

In the 3-layer neural network, $ReLU$ will be used as activation function for the first 2 layers (hidden layers).

$$ReLU(x) = \begin{cases} x, & \text{if } x \geq 0 \\ 0, & \text{otherwise} \end{cases}$$

$softmax$ will be used for the last layer, since we need to make the output layer behave as a proper set of probabilities.

$$softmax(z_j^L) = \frac{e^{z_j}}{\sum_k e^{z_j}}$$

Loss function

Regularization by adding a penalty term to the loss function will be employed intended to penalize the weights and address overfitting. We can add an ℓ_2-norm as was done with ridge regression. The loss function:

$$\mathcal{L} = -\frac{1}{n}\sum_{i=1}^{n} log\ p(y = y_i|X_i) + \frac{\lambda}{2}(\|W_1\|_2^2 + \|W_2\|_2^2 + \|W_3\|_2^2)$$

Coding

The activation functions and loss functions will be differentiated for backpropagation. A backpropagation function will be written to give a list of gradients. Gradient descent (standard or SGD) will be implemented to estimate the neural network. Finally, the predicted values will be given. To intuitively show the result, `ggplot2` package will be employed to do some plotting.

Reference: *A Computational Approach to Statistical Learning*, Chapter 8.