Neural networks

Fraida Fund

Contents

In this lecture 1
From linear to non-linear 2
Review: learning non-linear decision boundaries from linear classifiers 2
Using multiple logistic regressions? 2
Model of example two-stage classifier (1) 2
Model of example two-stage classifier (2) 2
Illustration of two-stage classifier 2
Training the two-stage classifier 2

In this lecture

From linear to non-linear

Review: learning non-linear decision boundaries from linear classifiers

- Logistic regression - using basis functions
- SVM - using kernel
- Decision tree - AdaBoost uses multiple linear classifiers (decision stumps)

Using multiple logistic regressions?

TODO example notebook

Step 1: Classify into small number of linear regions. Each output from step 1 is a linear classifier with soft decision.

Step 2: Predict class label. Output is weighted average of step 1 weights

Model of example two-stage classifier (1)

First step (*hidden layer*):

- Take $N_H = 4$ linear discriminants.

$$\begin{bmatrix} z_{H,1} = w_{H,1}^T x + b_{H,1} \\ \cdots \\ z_{H,N_H} = w_{H,N_H}^T x + b_{H,N_H} \end{bmatrix}$$

- Each makes a soft decision: $u_{H,m} = g(z_{H,m}) = \frac{1}{1+e^{-z_{H,m}}}$

Model of example two-stage classifier (2)

Second step (*output layer*):

- Linear discriminant using output of previous stage as features:

$$z_o = w_o^T u_H + b_o$$

- Soft decision:

$$u_o = g(z_o) = \frac{1}{1 + e^{-z_o}}$$

Illustration of two-stage classifier

Training the two-stage classifier

- From final stage: $z_o = F(\mathbf{x}, \theta)$ where parameters $\theta = (\mathbf{W}_H, \mathbf{W}_o, b_H, b_o)$
- Given training data $(\mathbf{x}_i, y_i), i = 1, \ldots, N$ and loss function $L(\theta) := -\sum_{i=1}^{N} \ln P(y_i|\mathbf{x}_i, \theta)$
- Choose parameters to minimize loss: $\hat{\theta} = \text{argmin}_\theta \, L(\theta)$ ## Neural networks

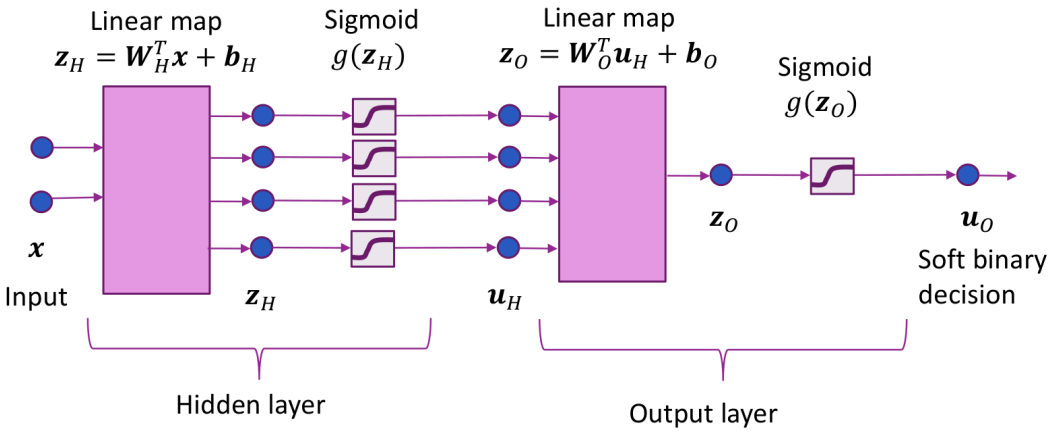

Figure 1: Two-stage classifier