Letter Recognition — Setup

Ling Li
Learning Systems Group, Caltech
October 16, 2001
Data Summary

• Letter Image Recognition Data (denoted by $\mathcal{D}$).

• $N = |\mathcal{D}| = 15997$ samples.

• 16 integer features (range 0–15).

• 26 classes (A–Z, 26 capital letters).

$\mathcal{D}$ can be normalized by its mean and standard deviation.

![Distribution of samples and mean/standard deviation of features]
Nearest-Neighbor

Divide $\mathcal{D}$ into two parts: $\mathcal{D}^{(1)} = \mathcal{D}_{1 \sim n}$ and $\mathcal{D}^{(2)} = \mathcal{D}_{n+1 \sim N}$.

Use $\mathcal{D}^{(1)}$ as the training set and $\mathcal{D}^{(2)}$ the validation set. The in-sample leave-one-out errors (to estimate the oos errors) and the validation errors were plot.

- Runtime: $\sim 4$ min/run (Matlab, 1G Hz PIII).
- 3 or 5-neighbor: similar curves, higher errors when $n$ is small or large.
- If normalized data is used, the error usually increased by $0.5 \sim 0.8$ percents.
Neural Networks

For this problem, Schwenk et al. used a 16-70-50-26 architecture, in “Boosting Neural Networks.” With boosting, they got error rate 1.5%.

I tried casually:

- train input: scaled to $-0.95 \sim 0.95$.
- class $i$ output: 0.99 for the $i^{th}$ output, $-0.99$ for rest.
- runtime: $\sim 2$ sec/epoch (16-70-50-26) and $\sim 0.5$ sec/epoch (16-60-26)
- momentum was adopted.
- error: within 10 epochs, reduced to $\sim 0.076$. ($\sqrt{2} \times 0.076 \approx 0.39$)