(Introduction to) Neural Computation

Bias vs. Variance

Peter Tiño
School of Computer Science
University of Birmingham
Curse of finite samples

Problem: Recovering functions over infinite domains using finite samples

Additional problem: The sample we are given may be noisy

\( X \subseteq \mathbb{R}^n \) - input space
\( Y \subseteq \mathbb{R}^m \) - output space

Nature/Environment: Joint distribution \( p(X,Y) \) over \( X \times Y \).
Generating finite samples

\[ p(x, y) = p(y|x) \cdot p(x) \]

For each data item \( i = 1, 2, \ldots, N \):
1. Generate an input \( x^i \) from \( p(x) \)
2. Given the input \( x^i \), generate the associated output \( y^i \) from \( p(y|x^i) \)

Generated sample (training set):
\[ T = \{(x^1, y^1), (x^2, y^2), \ldots, (x^N, y^N)\}. \]
Degrees of freedom in learning with a fixed learning machine

Noisy environment:
We may generate infinitely many different (training) samples $\mathcal{T}$ of size $N$.
Imagine that each sample $\mathcal{T}$ is a realisation of a random variable $T$ over $(X \times Y)^N$. $T$ is distributed according to $p_T$.

Sensitivity to parameter initialization:
On the same sample the learning mechanism can come up with many different answers, depending on parameter initialization.
Imagine that each initial setting parameters $\mathbf{w}$ is a realisation of a random variable $W$ over the parameter space. $W$ is distributed according to $p_W$. 
A thought experiment

Imagine we trained our learning model on all possible (or a huge number of) samples $\mathcal{T}$ of the same size $N$.

Imagine further that, given a sample $\mathcal{T}$, we trained our learning model starting using all possible (or a huge number of) initial parameter settings $\mathbf{w}$.

We are then given a single test input $\mathbf{x} \in \mathcal{X}$ and check the model prediction $\hat{y} = f(\mathbf{x}) \in \mathcal{Y}$ in each training run.
Questions to be asked

1. **Bias:**
   How accurate (on average) are the model predictions?
   The ‘desired’ (mean) response from the environment:
   \[
   \mu(x) = E_P(y|x)[y]
   \]

2. **Variance:**
   How spread out are the model predictions?
   Mean of the squared fluctuations of model predictions \( f(x) \) around their ‘center of gravity’, \( E_{P_T,P_W}[f(x)] \).
Mean squared error at \( x \)

Notation:

\[
E_{PT, PW}[\cdot] = E_{T, W}[\cdot]
\]

Mean squared error at \( x \):

\[
E_{T, W}[(f(x) - \mu(x))^2] = \text{Bias} + \text{Variance} \\
= (E_{T, W}[f(x)] - \mu(x))^2 \\
+ E_{T, W}[(f(x) - E_{T, W}[f(x)])^2]
\]
Bias/Variance dilemma

Simple models of low complexity
- high bias, small variance
- potentially rubbish, but stable predictions (w.r.t. different samples $\mathcal{T}$ and initial parameters $\mathbf{w}$)

Flexible models of high complexity
- small bias, high variance
- over-complex models can be always massaged to exactly explain the observed training data

What is the right level of model complexity?
The problem of model selection
Model selection

Heuristic approaches
- Optimal Brain Damage
- Optimal Brain Surgeon
- ...

Evolutionary approaches
- Evolutionary neural networks
- Evolutionary grammar induction
- ...

Peter Tiño
Model selection (Continued)

Information theory and coding approaches
- Akaike Information Criterion
- Bayesian Information Criterion
- Minimum Message length
- ...

Sampling approaches
- Early stopping
- Cross-Validation
- ...