**Bias-Variance Analysis**

Let $\mathcal{X}$ be a set (or space) of objects and let $\rho$ be a fixed probabiity distribution (or density) on $\mathcal{X} \times R$. In other worlds $\rho$ is a probability density on pairs $\langle x, y \rangle$ with $x \in \mathcal{X}$ and $y \in R$. We now consider an arbitrary space of prediction functions $f_w : \mathcal{X} \to R$ with $w \in R^D$. For example, we might have $f_w(x) = w \cdot \Phi(x)$ where $\Phi : \mathcal{X} \to R$ is a feature map. But we might also have some other arbitrary function such as the following "neural network".

$$f_w(x) = w_1 s(w_s \Phi_2(x) + w_3 \Phi_3(x)) + w_4 s(w_5 \Phi_3(x) + w_6 \Phi_4(x)) \quad (1)$$

$$s(z) = \frac{1}{1 + e^{-z}} \quad (2)$$

The point is that $f_w$ can be any function parameterized by a vector $w$ of $D$ parameters. The set of functions of the form $f_w$ can be viewed as a space of predictor functions. We can now define the following.

$$f^*(x) = \mathrm{E}\left[y|x\right] = \mathrm{E}_{y \sim \rho(\cdot|x)}\left[y\right] \quad (3)$$

$$w^* = \operatorname*{argmin}_w \mathrm{E}_{\langle x, y \rangle \sim \rho}\left[(f_w(x) - y)^2\right] \quad (4)$$

$$(5)$$

The function $f^*$ is the best predictor possible (for square loss) over all function of $\mathcal{X}$ and $f_{w^*}$ is the best predictor (for square loss) in the space of predictors of the form $f_w$. Typically $f^*$ is not equal to any function of the form $f_w$. Now consider an arbitrary predictor $f$. We define the square loss of $f$ as follows.

$$\mathcal{L}_2(f) = \mathrm{E}_{\langle x, y \rangle \sim \rho}\left[(f(x) - y)^2\right] \quad (6)$$

The first step of bias-variance analysis is the following expression for $\mathcal{L}_2(f)$.

$$\begin{aligned}
\mathcal{L}_2(f) &= \mathrm{E}_{\langle x, y \rangle \sim \rho}\left[(f(x) - y)^2\right] \\
&= \mathrm{E}_{\langle x, y \rangle \sim \rho}\left[((f(x) - f^*(x)) - (y - f^*(x)))^2\right] \\
&= \mathrm{E}_{\langle x, y \rangle \sim \rho}\left[(f(x) - f^*(x))^2 - 2(f(x) - f^*(x))(y - f^*(x)) + (y - f^*(x))^2\right] \\
&= \mathrm{E}_{x \sim \rho}\left[(f(x) - f^*(x))^2\right] \\
&\quad -2\mathrm{E}_{x \sim \rho}\left[(f(x) - f^*(x))\mathrm{E}_{y \sim \rho(\cdot|x)}\left[y - f^*(x)\right]\right] \\
&\quad +\mathrm{E}_{\langle x, y \rangle \sim \rho}\left[(y - f^*(x))^2\right] \\
&= \mathrm{E}_{x \sim \rho}\left[(f(x) - f^*(x))^2\right] + \mathrm{E}_{x \sim \rho}\left[\mathrm{E}_{y \sim \rho(\cdot|x)}\left[(y - f^*(x))^2\right]\right] \quad (7)
\end{aligned}$$

Note that the second term in the left hand side of (7) does not depend on $f$. The second term can be interpreted as the average over the choice of $x$ of the variance of $y$ given $x$, i.e., the variance of $y$ when we repeatedly draw different

values of $y$ from the conditional distribution on $y$ given $x$. We will call this the noise term — we can interpret $y$ as being equal to $f^*(x)$ plus zero mean noise.

Note that the noise term does not depend on $f$. This implies that $w^*$ can be defined equivalently as follows.

$$w^* = \operatorname*{argmin}_w \mathrm{E}_{x\sim\rho}\left[(f_w(x) - f^*(x))^2\right] \tag{8}$$

Now we consider learning from a sample $D = \langle x_1, y_1\rangle, \ldots, \langle x_N, y_N\rangle$. We assume that the sample is drawn IID from $\rho$, i.e., each pair $\langle x_t, y_t\rangle$ is drawn independently from $\rho$. So the training data itself is a random variable. We are already considering an arbitrary parameterized space of predictors. We now consider an arbitrary learning algorithm $A$ which takes as input training data $D$ and produces as output a setting $A(D)$ of the predictor parameters. We now consider the expected generalization loss when we use learning algorithm $A$.

$$\begin{aligned}\mathcal{L}_2(A) &= \mathrm{E}_{D\sim\rho^N}\left[\mathcal{L}_2(f_{A(D)})\right] \\ &= \mathrm{E}_{D\sim\rho^N,\langle x,y\rangle\sim\rho}\left[(f_{A(D)}(x) - y)^2\right]\end{aligned}$$

We also define the following "average prediction" on input $x$ under learning algorithm $A$.

$$\overline{f}_A(x) = \mathrm{E}_{D\sim\rho^N}\left[f_{A(D)}(x)\right]$$

The full bias-variance analysis is to rewrite $\mathcal{L}_2(A)$ as follows.

$$\begin{aligned}\mathcal{L}_2(A) &= \mathrm{E}_{D\sim\rho^N,\langle x,y\rangle\sim\rho}\left[(f_{A(D)}(x) - y)^2\right] \\ &= \mathrm{E}_{D\sim\rho^N,\langle x,y\rangle\sim\rho}\left[((f_{A(D)}(x) - \overline{f}_A(x)) - (y - \overline{f}_A(x)))^2\right] \\ &= \mathrm{E}_{D\sim\rho^N,\langle x,y\rangle\sim\rho}\left[(f_{A(D)}(x) - \overline{f}_A(x))^2 - 2(f_{A(D)}(x) - \overline{f}_A(x))(y - \overline{f}_A(x)) + (y - \overline{f}_A(x))^2\right] \\ &= \mathrm{E}_{x\sim\rho}\left[(f_{A(D)}(x) - \overline{f}_A(x))^2\right] \\ &\quad -2\mathrm{E}_{\langle x,y\rangle\sim\rho}\left[(y - \overline{f}_A(x))\mathrm{E}_{D\sim\rho^N}\left[f_{A(D)}(x) - \overline{f}_A(x)\right]\right] \\ &\quad +\mathrm{E}_{\langle x,y\rangle\sim\rho}\left[(y - \overline{f}_A(x))^2\right] \\ &= \mathrm{E}_{x\sim\rho}\left[(f_{A(D)}(x) - \overline{f}_A(x))^2\right] + \mathcal{L}_2(\overline{f}_A) \\ \\ &= \mathrm{E}_{x\sim\rho,D\sim\rho^N}\left[(f_{A(D)}(x) - \overline{f}_A(x))^2\right] \\ &\quad +\mathrm{E}_{x\sim\rho}\left[(\overline{f}_A(x) - f^*(x))^2\right] \\ &\quad +\mathrm{E}_{\langle x,y\rangle\sim\rho}\left[(y - f^*(x))^2\right]\end{aligned} \tag{9}$$

Equation (9) gives the full bias-variance analysis. The first term in the left hand is the average over drawing $x$ of the variance over drawing traning data of $f_{A(D)}(x)$. This is called the variance term. The second term is a squared distance between the function $\overline{f}_A$ and the optimal function $f^*$. This is called

the bias term. The third term does not depend on $A$ at all and is just the average variance of noise added to $y$ at a given $x$. This is called the noise term.

It should be noted that in general the function $\overline{f}_A$ is different from optimal function $f_{w^*}$ in the parameterized space of functions. In fact if the space of parameterized functions is nonconvex, then $\overline{f}_A$ may not be definable by any setting of the parameters.

As the number of parameters is increased we typically have that the bias term decreases while the variance term increases. Hence there is a bias-variance trade off.

# 1 Bias-Variance for $K$-Nearest Neighbor

For a give sample $D$ and point $x$ let $N_k(x)$ be the set of times $t$ such that $x_t$ is one of the $K$ nearest neighbors of $x$ over all the training values $x_t$. We define the $K$-nearest neighbor predictor as follows.

$$f_D(x) = \frac{1}{K} \sum_{t \in N_K(x)} y_t \tag{10}$$

Although the nearest neighbor rule is non-parametric, the bias-variance analysis still applies when we measure the performance of $f_D$ with square loss. For $K = 1$ the bias is very small — the expected value of $f_D(x)$ should be near $f^*$ since the one nearest neighbor should be near $x$. But the variance of one nearest neighbor is very large. As we increase $K$ the variance becomes smaller because we are averaging over more training points for each prediction. However, as $K$ increases the bias eventually becomes large because we are using points that are far from $x$. For $K = N$ we simply predict the mean value of $y$. This has quite low variance but the bias is large as $f_D(x)$ now ignores $x$.

# 2 Linear Learning of Linear Predictors

Consider the class of linear predictors defined by $f_w(x) = w \cdot \Phi(x)$ for some feature map $\Phi$ and weight vector $w$ with $w, \Phi(x) \in R^D$. Again we assume a given distribution $\rho$ on $\mathcal{X} \times R$. For linear predictors the optimal parameter setting $w^*$, as defined by (4), can be written as follows.

$$\begin{aligned} w^* &= \Gamma^{-1}\overline{\beta} \\ \overline{\beta} &= \mathrm{E}_{\langle x, y\rangle \sim \rho}\left[y\Phi(x)\right] \\ \Gamma &= \mathrm{E}_{x \sim \rho}\left[\Phi(x)\Phi^T(x)\right] \end{aligned}$$

We now assume that the matrix $\Gamma$ is given by God (or perhaps by a vast sample of $x$ only). We then consider the following "linear" learning algorithm $A$.

$$A(D) = \Gamma^{-1}\hat{\beta} \tag{11}$$

$$\hat{\beta} = \frac{1}{N}\sum_{t=1}^{N} y_t \Phi(x_t) \tag{12}$$

In general we can define $\overline{w}$ to be the average value of $A(D)$, i.e. the expectation over drawing $D$ of the parameter vector $A(D)$. For the special case of linear learning of linear predictors we have the following.

$$\overline{f}_A(x) = f_{\overline{w}}(x) = f_{w^*}(x) \tag{13}$$

In this case the bias term can be written as follows.

$$\text{bias} = \mathrm{E}_{x\sim\rho}\left[(w^* \cdot \Phi(x) - f^*(x))^2\right]$$

So for linear learning of linear predictors the bias can be interpreted as a square distance between $f_{w^*}$ and the ideal function $f^*$. If $\mathcal{X}$ is finite then this is literally a squared distance in a finite dimensional vector space. So the bias becomes literally a squared distance between $f_{w^*}$ and $f^*$. Note that if we add new fetaures this distance cannot increase. The bias will typically be reduced as we add new features.

We can also show that as new features are added the variance is non-decreasing. To do this we will work in the coordinate system that is the eigenvectors of $\Gamma$. In this coordiante system we have the following.

$$w_i^* = \frac{\overline{\beta}_i}{\lambda_i}$$

$$A(D)_i = \frac{\hat{\beta}_i}{\lambda_i}$$

$$\lambda_i = \mathrm{E}_{x\sim D}\left[\Phi_i^2(x)\right]$$

The variance can now be written as follows.

$$
\begin{aligned}
\text{variance} &= \mathrm{E}_{x\sim\rho, D\sim\rho^N}\left[(A(D)\cdot\Phi(x) - w^*\cdot\Phi(x))^2\right] \\
&= \mathrm{E}_{x\sim\rho, D\sim\rho^N}\left[((A(D) - w^*)\cdot\Phi(x))^2\right] \\
&= \mathrm{E}_{x\sim\rho, D\sim\rho^N}\left[\left(\sum_i (A(D)_i - w_i^*)\Phi_i(x)\right)^2\right] \\
&= \mathrm{E}_{x\sim\rho, D\sim\rho^N}\left[\sum_{i,j} (A(D)_i - w_i^*)(A(D)_j - w_j^*)\Phi_i(x)\Phi_j(x)\right] \\
&= \sum_{i,j} \mathrm{E}_{D\sim\rho^N}\left[(A(D)_i - w_i^*)(A(D)_j - w_j^*)\mathrm{E}_{x\sim\rho}\left[\Phi_i(x)\Phi_j(x)\right]\right] \\
&= \sum_i \mathrm{E}_{D\sim\rho^N}\left[(A(D)_i - w_i^*)^2\lambda_i\right] \\
&= \sum_i \mathrm{E}_{D\sim\rho^N}\left[(\hat{\beta}_i - \overline{\beta}_i)^2/\lambda_i\right] \\
&= \sum_{i=1}^{D} \frac{\sigma_i^2}{N} \qquad (14) \\
\sigma_i^2 &= \mathrm{E}_{\langle x, y\rangle\sim\rho}\left[\left(\frac{y\Phi_i(x)}{\sqrt{\lambda_i}} - \frac{\overline{\beta}_i}{\sqrt{\lambda_i}}\right)^2\right]
\end{aligned}
$$

If we add a feature that is linearly independent of the existing feaures, then this the new feature can always be written as a linear combination of the previous eigenvectors plus a new eigenvector orthogonal to the previous ones. Hence a new feature simply adds another term to (14) so we get that variance can only increase when a new feature is added. Note that for linear learning of linear predictors, the bias is independent of the size of the training sample, but the variance decreases linearly with $N$ (for a fixed feature set). Also, if $y$ and $\Phi_i(x)$ are both bounded then we get that $\sigma_i^2$ is bounded and the variance of the learning algorithm is no larger than $O(D/N)$.