Regularization combats the effects of noise by putting a leash on the algorithm.

\[ E_{\text{aug}}(h) = E_{\text{in}}(h) + \lambda \Omega(h) \]

- \( \Omega(h) \rightarrow \) smooth, simple \( h \)
- noise is rough, complex.

Different regularizers give different results
- can choose \( \lambda \), the amount of regularization.

\[ \lambda = 0 \quad \lambda = 0.0001 \quad \lambda = 0.01 \quad \lambda = 1 \]

Overfitting \( \rightarrow \) Underfitting

Optimal \( \lambda \) balances approximation and generalization, bias and variance.

Validation: A Sneak Peek at \( E_{\text{out}} \)

Validation goes directly for the jugular:

\[ E_{\text{val}}(g) = E_{\text{in}}(g) + \text{overfit penalty} \]

In-sample estimate of \( E_{\text{out}} \) is the Holy Grail of learning from data.
The Validation Set

1. Remove $K$ points from $\mathcal{D}$

\[ \mathcal{D} = D_{\text{train}} \cup D_{\text{val}} \]

2. Learn using $D_{\text{train}} \rightarrow g^*$

3. Test $g^*$ on $D_{\text{val}} \rightarrow E_{\text{val}}$

4. Use error $E_{\text{val}}$ to estimate $E_{\text{out}}(g^*)$

$E_{\text{val}}$ is an estimate for $E_{\text{out}}(g^*)$

\[ E_{\text{val}}(g) = E_{\text{val}}(g^*) \]

\[ E_{\text{val}} = \frac{1}{K} \sum_{k=1}^{K} e_k \]

\[ \text{Var}[E_{\text{val}}] = \frac{1}{K^2} \sum_{k=1}^{K} \text{Var}[e_k] \]

Rule of thumb: $K^* = \frac{N}{4}$

Choosing $K$

Expected $E_{\text{val}}$

Size of Validation Set, $K$

10 20 30

$E_{\text{val}}$ depends on $K$

Smaller $K$ = more reliable $E_{\text{val}}$?
Restoring $D$

Primary goal: output best hypothesis. $g$ was trained on all the data.

Secondary goal: estimate $E_{\text{out}}(g)$. $g$ is behind closed doors.

$E_{\text{val}}(g) \leq E_{\text{in}}(g) + O\left(\sqrt{d_{\text{vc}} N \log N}\right)$

$E_{\text{out}}(g) \leq E_{\text{out}}(g) \leq E_{\text{val}}(g) + O\left(\frac{1}{\sqrt{K}}\right)$

Learning curve is decreasing (a practical truth, not a theorem)

Unbiased error bar depends on $g$.

$E_{\text{val}}(g)$ usually wins as an estimate for $E_{\text{out}}(g)$, especially when the learning curve is not steep.

Model Selection

The most important use of validation

$\mathcal{H}_1 \rightarrow \mathcal{H}_2 \rightarrow \mathcal{H}_3 \rightarrow \mathcal{H}_M$

$D_{\text{train}} \rightarrow \mathcal{H}_1 \rightarrow \mathcal{H}_2 \rightarrow \mathcal{H}_3 \rightarrow \mathcal{H}_M$

Validation Estimate for $(\mathcal{H}_1, g_1)$

The most important use of validation
Validation Estimate for \((\mathcal{H}_1, g_1)\)

The most important use of validation

\[
\begin{align*}
\mathcal{H}_1 & \quad \mathcal{H}_2 & \quad \mathcal{H}_3 & \quad \cdots & \quad \mathcal{H}_M \\
D_{\text{train}} & \quad \downarrow & \quad \downarrow & \quad \downarrow & \quad \downarrow \\
\mathcal{G}_1 & \quad \mathcal{G}_2 & \quad \mathcal{G}_3 & \quad \cdots & \quad \mathcal{G}_M \\
D_{\text{val}} & \quad \downarrow & \quad \downarrow & \quad \downarrow & \quad \downarrow \\
E_1 & \quad E_2 & \quad E_3 & \quad \cdots & \quad E_M
\end{align*}
\]

Compute Validation Estimates for All Models

The most important use of validation

\[
\begin{align*}
\mathcal{H}_1 & \quad \mathcal{H}_2 & \quad \mathcal{H}_3 & \quad \cdots & \quad \mathcal{H}_M \\
D_{\text{train}} & \quad \downarrow & \quad \downarrow & \quad \downarrow & \quad \downarrow \\
\mathcal{G}_1 & \quad \mathcal{G}_2 & \quad \mathcal{G}_3 & \quad \cdots & \quad \mathcal{G}_M \\
D_{\text{val}} & \quad \downarrow & \quad \downarrow & \quad \downarrow & \quad \downarrow \\
E_1 & \quad E_2 & \quad E_3 & \quad \cdots & \quad E_M
\end{align*}
\]

Pick The Best Model According to Validation Error

\[
\begin{align*}
\mathcal{H}_1 & \quad \mathcal{H}_2 & \quad \mathcal{H}_3 & \quad \cdots & \quad \mathcal{H}_M \\
D_{\text{train}} & \quad \downarrow & \quad \downarrow & \quad \downarrow & \quad \downarrow \\
\mathcal{G}_1 & \quad \mathcal{G}_2 & \quad \mathcal{G}_3 & \quad \cdots & \quad \mathcal{G}_M \\
D_{\text{val}} & \quad \downarrow & \quad \downarrow & \quad \downarrow & \quad \downarrow \\
E_1 & \quad E_2 & \quad E_3 & \quad \cdots & \quad E_M
\end{align*}
\]

\[E_{\text{val}}(g_{m^*}) \text{ is not Unbiased For } E_{\text{out}}(g_{m^*})\]

… because we choose one of the \(M\) finalists.

\[E_{\text{out}}(g_{m^*}) \leq E_{\text{val}}(g_{m^*}) + O\left(\sqrt{\frac{\ln M}{K}}\right)\]

VC error bar for selecting a hypothesis from \(M\) using a data set of size \(K\).
Restoring $\mathcal{D}$

$$
\mathcal{H}_1 \quad \mathcal{H}_2 \quad \mathcal{H}_3 \quad \cdots \quad \mathcal{H}_M
$$

$$
g_1 \quad g_2 \quad g_3 \quad \cdots \quad g_M
$$

Model with best $g$ also has best $g^*$
We can find model with best $g^*$ using validation

$\leftarrow$ leap of faith
$\leftarrow$ true modulo $E_{val}$ error bar

Comparing $E_{in}$ and $E_{val}$ for Model Selection

Application to Selecting $\lambda$

Which regularization parameter to use?
$$
\lambda_1, \lambda_2, \ldots, \lambda_M.
$$

This is a special case of model selection over $M$ models,

$$
(\mathcal{H}, \lambda_1) \quad (\mathcal{H}, \lambda_2) \quad (\mathcal{H}, \lambda_3) \quad \cdots \quad (\mathcal{H}, \lambda_M)
$$

$$
g_1 \quad g_2 \quad g_3 \quad \cdots \quad g_M
$$

Picking a model amounts to chosing the optimal $\lambda$

The Dilemma When Choosing $K$

Validation relies on the following chain of reasoning,

$$
E_{out}(g) \approx E_{out}(g^*) \approx E_{val}(g^*)
$$

(small $K$) \quad (large $K$)
Can we get away with $K = 1$?

Yes, almost!

The Leave One Out Error ($K = 1$)

$\mathbb{E}[e_1] = \mathbb{E}_{\text{out}}(\mathcal{F})$

... but it is a wild estimate

The Leave One Our Errors

$E_{\text{cv}} = \frac{1}{N} \sum_{n=1}^{N} e_n$

Cross Validation is Unbiased

Theorem. $E_{\text{cv}}$ is an unbiased estimate of $\bar{\mathbb{E}}_{\text{out}}(N - 1)$.

Expected $E_{\text{cv}}$ when learning with $N - 1$ points.
Reliability of $E_{cv}$

$e_n$ and $e_m$ are not independent.

- $e_n$ depends on $g_n$, which was trained on $(x_m, y_m)$.
- $e_m$ is evaluated on $(x_m, y_m)$.

$E_{cv}$ can be used for model selection just as $E_{val}$, for example to choose $\lambda$. 

Cross Validation is Computationally Intensive

$N$ epochs of learning each on a data set of size $N - 1$.

- Analytic approaches, for example linear regression with weight decay:
  \[
  w_{reg} = (Z^T Z + \lambda I)^{-1} Z^T y
  \]

- 10-fold cross validation:

  \[
  H(\lambda) = Z(Z^T Z + \lambda I)^{-1} Z^T y
  \]

Digits Problem: ‘1’ Versus ‘Not 1’

- 5th order polynomial transform:
  - 20 dimensional non-linear feature space

Average Intensity

\[
\begin{align*}
\mathbf{x} &= (1, x_1, x_2) \\
\mathbf{z} &= (1, x_1, x_2, x_1^2, x_2^2, x_1^3, x_2^3, x_1^4, x_2^4, x_1^5, x_2^5, x_1^6, x_2^6, x_1^7, x_2^7, x_1^8, x_2^8, x_1^9, x_2^9, x_1^{10}, x_2^{10})
\end{align*}
\]
Validation Wins In the Real World

\[ E_{in} = 0\% \]
\[ E_{out} = 2.5\% \]

\[ E_{in} = 0.8\% \]
\[ E_{out} = 1.5\% \]