Week 15: Validation

Table of Contents

Model Selection Problem

1. Model selection problem
    - Argurably the **most important** practical problem of ML
    - Given: M models $\mathcal{H}_1, \mathcal{H}_2, \dots, \mathcal{H}_\mathcal{M}$, each with corresponding algorithms $\mathcal{A}_1, \mathcal{A}_2, \dots, \mathcal{A}_\mathcal{M}$
    - Goal: Select $\mathcal{H}_{m^*}$ such that $g_{m^*} = \mathcal{A}_{m^*}(\mathcal{D})$ has **low out-of-sample error** $E_{out}(g_{m^*})$
    - Problem: unknown E_{out} due to unknown input distribution $P(X)$ and target distribution $P(y|x)$

Potential Approaches to Model Selection

1. By best E_{in}

$$m^* = \underset{1 \le m \le M}{\operatorname{argmin}}(E_m = E_{in}(\mathcal{A}_m(D)))$$

- Risk of overfitting
    - Higher-order hypothesis set **always favored** for closer fitting to training data (ϕ_{1126} over ϕ_1)
    - No regularization **always favored** over regularization for ability to generate more complex fit
- High generalization error per VC theory
    - Week 7: The VC Dimension
    - g_{m^*} achieves minimal E_{in} by computing and comparing E_{in} for **every hypothesis from every hypothesis set** $\rightarrow d_{vc} = d_{vc}(\mathcal{H}_1 \cup \mathcal{H}_2 \cup \cdots \cup \mathcal{H}_\mathcal{M})$

2. By best E_{test}

$$m^* = \underset{1 \le m \le M}{\operatorname{argmin}}(E_m = E_{test}(\mathcal{A}_m(D)))$$

- Choose the best hypothesis from each hypothesis set, then compute and compare E_{test} only for these **best candidates**
- E_{test} evaluated on a **fresh** $\mathcal{D}_{test}$
- Generalization guarantee given by finite-bin Hoeffding

$$E_{out}(g_{m^*}) \leq E_{test}(g_{m^*}) + O(\sqrt{\frac{\log M}{N_{test}}})$$

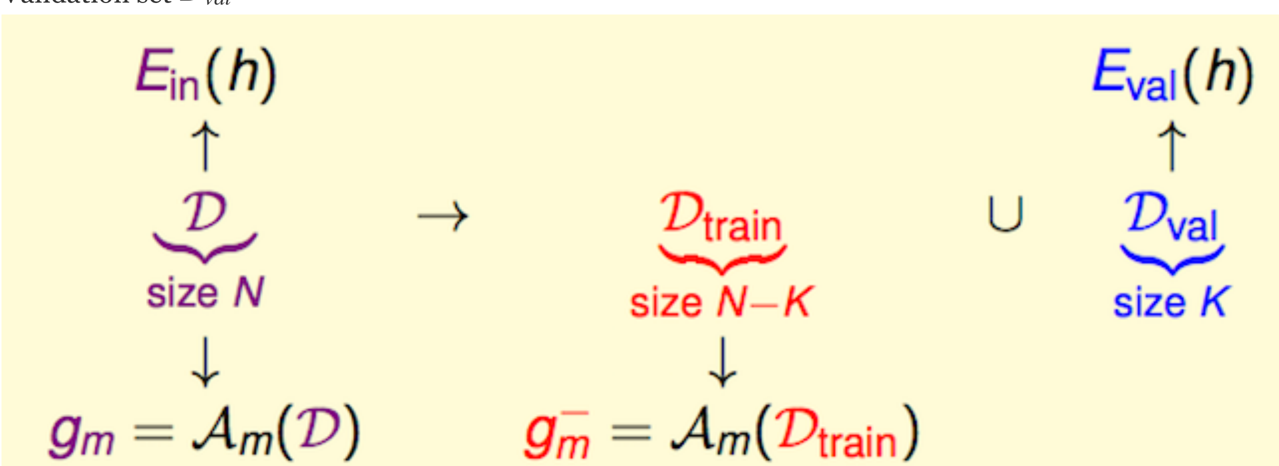
  - See Week 7
- However, testing set $\mathcal{D}_{test}$ is hard, if not impossible to obtain
  - Requires another iid sampling from target population ("locked in boss's safe")

3. By best E_{val}
   - E_{value} evaluated on an **validation set** $\mathcal{D}_{val} \subset \mathcal{D}$ previously **reserved** from available samples
     - $\mathcal{D}_{train} \cup \mathcal{D}_{val} = \mathcal{D}$
   - "Middle ground" between E_{in} and E_{test}
4. Comparing the approaches

in-sample error E_{in}	something in between: E_{val}	test error E_{test}
• calculated from $\mathcal{D}$	• calculated from $\mathcal{D}_{val} \subset \mathcal{D}$	• calculated from $\mathcal{D}_{test}$
• feasible on hand	• feasible on hand	• infeasible in boss's safe
• 'contaminated' as $\mathcal{D}$ also used by $\mathcal{A}_m$ to 'select' g_m	• 'clean' if $\mathcal{D}_{val}$ never used by $\mathcal{A}_m$ before	• 'clean' as $\mathcal{D}_{test}$ never used for selection before

Validation

1. Validation set $\mathcal{D}_{val}$



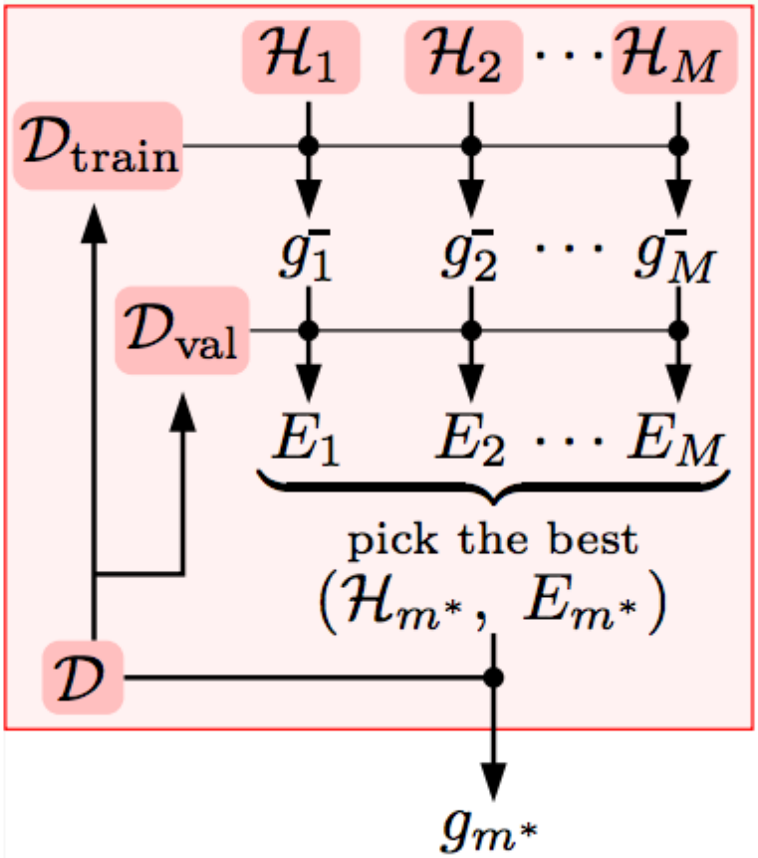
   - $\mathcal{D}_{val} \subset \mathcal{D}$
   - Provides simulation of test set with data on-hand
   - Validation set needs to be selected **at random** from available sample set $\mathcal{D}_{val}$ in order to satisfy $\mathcal{D}_{val} \overset{iid}{\sim} P(x, y)$ and provide VC guarantee between $\mathcal{D}_{val}$ and E_{out}
   - Feed only $\mathcal{D}_{train}$ to learning algorithm $\mathcal{A}_m$ to get hypothesis g, to ensure that $\mathcal{D}_{val}$ remains 'clean' and only used for validation

Model Selection by Best E_{val}

1. Choose the best model m^* as

$$m^* = \underset{1 \le m \le M}{\text{argmin}}(E_m = E_{val}(\mathcal{A}(\mathcal{D}_{train})))$$

   - The model selection process can be visualized as follows



2. The best hypothesis, g_m^- selected based on this process, has generalization guarantee (recall definition of generalization error from Week 7):

$$E_{out}(g_m^-) \le E_{val}(g_m^-) + O\left(\sqrt{\frac{\log M}{K}}\right)$$

3. Heuristic gain from training set $(N - K)$ to full input set N
   - Due to the increase in data size, applying best hypothesis $g_{m^*}^-$ learned on training set $\mathcal{D}_{train}$ onto the original input set $\mathcal{D}$ leads to a **decrease in** E_{out} $\Rightarrow$ Heuristic gain
   - Refer to "The Learing Curve" section from [Week 9: Linear Regression]()

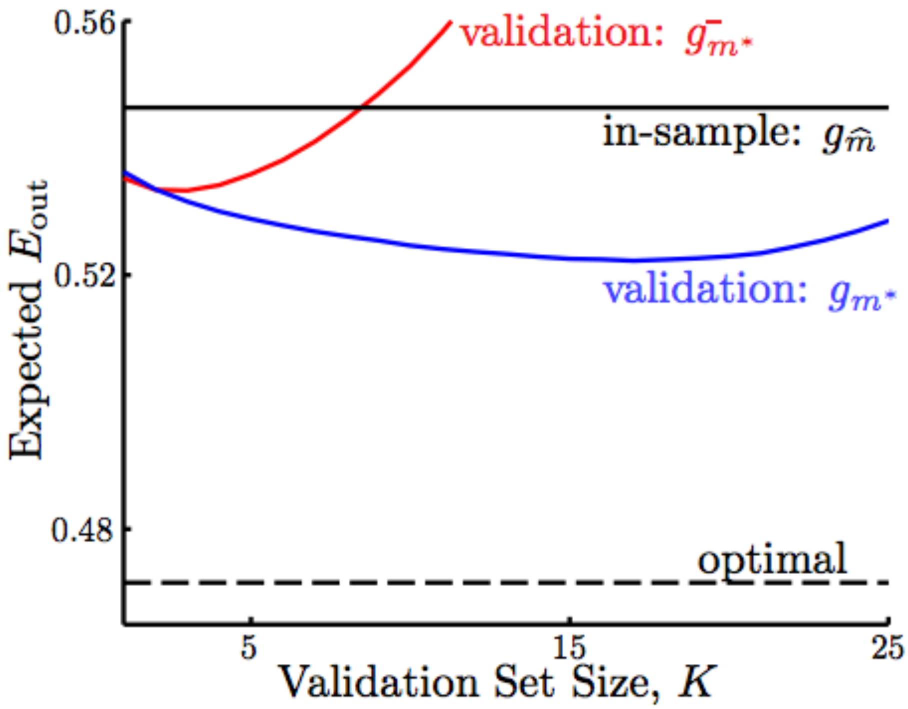
$$E_{out}(\underbrace{g_{m^*}}_{\mathcal{A}_{m^*}(\mathcal{D})}) \le E_{out}(\underbrace{g_{m^*}^-}_{\mathcal{A}_{m^*}(\mathcal{D}_{train})})$$

4. Merging inequalities from 2 and 3 gives:

$$E_{out}(g_{m^*}) \le E_{out}(g_{m^*}^-) \le E_{val}(g_{m^*}^-) + O\left(\sqrt{\frac{\log M}{K}}\right)$$

Validation in Practice

1. Compare E_{out} from different model selection mechanism
   - In-sample: Selection with E_{in}
   - Optimal: "Cheating" selection with E_{test}
   - Sub-g: Selection with E_{val} and report best hypothesis from **training set** $g^{-}_{m^*}$
   - Full-g: Selection with E_{val} and report best hypothesis from **full input set** g_{m^*}



2. Given the same input set, out-of-sample error of $g^{-}_{m^*}$ starts to deteriorate when the reserved validation set grows beyond a certain size
   - Not enough training data to produce good candidate hypothesis
   - g_{m^*} exhibits similar trend, but to lesser degree
3. The dilemma about K

$$E_{out}(g) \underset{\text{(small K)}}{\approx} E_{out}(g^{-}) \underset{\text{(large K)}}{\approx} E_{val}(e^{-})$$

   - Reasoning behind validation is to obtain a good proxy of $E_{out}(g)$ through $E_{out}(g^{-})$, which is in turn, proxied through $E_{val}(g^{-})$
   - However, first part of the proxy holds true only with small validation sets, and the second part only with large validation sets
     - Large K: **Every** $E_{val} \approx E_{out}$, but **all** g^{-} are much worse than g_m
     - Small K: **Every** $g^{-}_{m} \approx g_m$, but **all** E_{val} will be far from E_{out}
4. Practical rule of thumb for validation set size

$$K = \frac{N}{5}$$

Leave-One-Out Cross Validation

1. One-sample validation set
   - Extreme case of $K = 1$
   - Validation set consists of **one sample**, and validation error evaluated at single-sample level:

$$\mathcal{D}_{val}^{(n)} = \{(x_n, y_n)\}$$
$$E_{val}^{(n)}(g_n^-) = err(g_n^-(x_n), y_n) = e_n$$

     - Where $err(g_n^-(x_n), y_n)$ is the error measure for hypothesis g_n^-, given sample (x_n, y_n)

2. In order for the **single-sample** e_n to be a good approximation of $E_{out}(g)$ (which is evaluated on the **entire** out-of-sample set), need to take into account all possible values of $E_{val}^{(n)} \rightarrow$ average over all possible $E_{out}^{(n)}$

3. Leave-one-out **cross validation** estimate

$$E_{loocv}(\mathcal{H}, \mathcal{A}) = \frac{1}{N}\sum_{n=1}^{N} e_n = \frac{1}{N}\sum_{n=1}^{N} err(g_n^-(x_n), y_n)$$

4. Choosing learning algorithm using leave-one-out cross validation
   - Assuming $E_{loocv}(\mathcal{H}, \mathcal{A}) \approx E_{out}(g)$, minimize $E_{loocv}(\mathcal{H}, \mathcal{A})$

$$m^* = \underset{1 \le m \le M}{\operatorname{argmin}}(E_m = E_{loocv}(\mathcal{H}_m, \mathcal{A}_m))$$

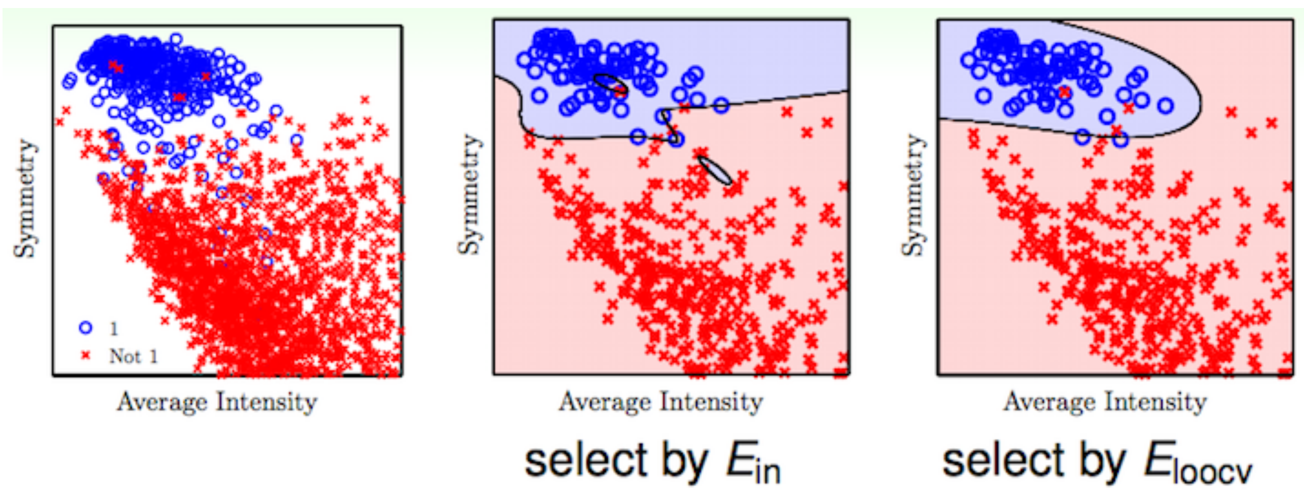
5. Theoretical guarantee of leave-one-out estimate

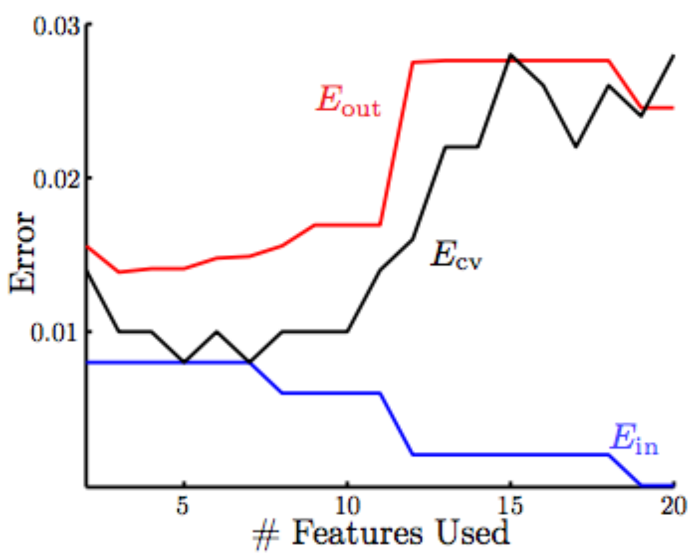
$$\begin{aligned}
\underset{\mathcal{D}}{\mathcal{E}} E_{loocv}(\mathcal{H}, \mathcal{A}) &= \underset{\mathcal{D}}{\mathcal{E}} \frac{1}{N}\sum_{n=1}^{N} e_n \\
&= \frac{1}{N}\sum_{n=1}^{N} \underset{\mathcal{D}}{\mathcal{E}} e_n \\
&= \frac{1}{N}\sum_{n=1}^{N} \underset{\mathcal{D}_n}{\mathcal{E}} \underset{x_n, y_n}{\mathcal{E}} err(g_n^-(x_n), y_n) \\
&= \frac{1}{N}\sum_{n=1}^{N} \underset{\mathcal{D}_n}{\mathcal{E}} E_{out}(g_n^-) \\
&= \frac{1}{N}\sum_{n=1}^{N} \overline{E_{out}}(N-1) = \overline{E_{out}}(N-1)
\end{aligned}$$

   - Notations
     - $\mathcal{E}$: Expected value (on a give data set)
     - $\mathcal{D}$: **Full** input sample set, comprised of $\mathcal{D}_{train}$ and $\mathcal{D}_{val} = (x_n, y_n)$
     - $\mathcal{D}_n$: Abbreviated notation for $\mathcal{D}_{train}$ in this context
   - Interpretation

        - Substitute in the definition of leave-one-out cross validation error
        - Swap order of evaluation for summation and expected value, and **decompose** expected value into two parts
            - Expected value on training set (N-1): $\underset{\mathcal{D}_n}{\mathcal{E}}$
            - Expected value on **single-sample** validation set: $\underset{x_n, y_n}{\mathcal{E}}$
        - Since g_n^- is found on **training set only**, each of the validation point is considered as **out-of-sample** from its perspective $\longrightarrow \underset{x_n, y_n}{\mathcal{E}} = E_{out}(g_n^-)$
        - Since leave-one-out cross validation will be performed on each individual sample from the input set, the expected value of $E_{out}(g_n^-)$ is essentially the **average** $E_{out}(g)$ on data set of size (N-1). The full input set, minus single sample reserved for cross-validation each time.
    - $E_{loocv}(\mathcal{H}, \mathcal{A})$ provides an **almost unbiased** estimate of $E_{out}(g)$

6. Leave-one-out cross validation in practice

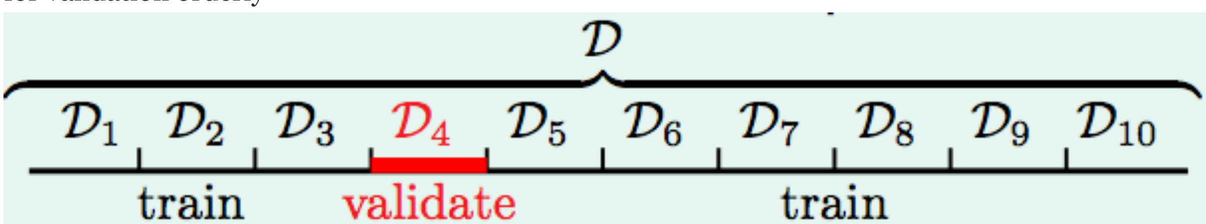




    - E_{loov} provides a much better proxy for E_{out} when used in model selection

V-fold Cross Validation

1. Disadvantages of leave-one-out estimate
    - High computation cost, **linear** to input size
        - For a set of N samples, by definition of $E_{loocv}(\mathcal{H}, \mathcal{A})$, N **additional** "trainings" are required ($N-1$ inputs each) to obtain the best model, which is not always feasible in practice
        - Except for special cases **with analytical solutions available** (e.g. linear regression), which allows for computation cost of leave-one-out estimate to decouple from input size
    - Stability
        - Validation errors calculated on **single sample** $\longrightarrow$ $E_{loocv}(\mathcal{H}, \mathcal{A})$ could fluctuate significantly if there are outliers Averaging over all samples does not necessarily offsets such fluctuations.
        - Such flucuations could potentially impact outcome of the cross-validation, depending on contents of input set
2. V-fold cross validation
    - Key idea: **Partition $\mathcal{D}$ into N parts**, and use $N-1$ parts for training, the remaining part for vaidation
        - Instead of N being the total number of samples, here N is the number of "folds"(partitions)
    - V-fold cross validation: **Random partition** of $\mathcal{D}$ into V **equal parts**. Use $V-1$ for training and 1 for validation orderly



3. V-fold cross validation estimate:

$$E_{loocv}(\mathcal{H}, \mathcal{A}) = \frac{1}{V}\sum_{v=1}^{V} E_{val}^{(v)}(g_{v}^{-})$$

4. Model selection by V-fold cross validation

$$m^* = \underset{1 \le m \le M}{\operatorname{argmin}}(E_m = E_{cv}(\mathcal{D}_m, \mathcal{A}_m))$$

5. E_{cv} provides similar guarantee for $\overline{E_{out}}(N-1)$, just in a scaled fashion (by factor of V), compared to $E_{loocv}(\mathcal{H}, \mathcal{A})$
    - Refer to proof above, replacing definition of $E_{loocv}(\mathcal{H}, \mathcal{A})$ with that of E_{cv}
    - Leave-one-out can be viewed as a special case of V-fold cross validation, with partitions of size 1
6. V-fold cross validation in practice
    - Rule of thumb: $V = 5 \text{ or } 10$
    - V-fold cross validation much more widely used than leave-one-out, and performing V-fold cross validation is generally preferred over performing a single validation if computation allows
        - More stable results from averaging E_{cv} across all partitions
7. Nature of validation
    - All training methods aim to **select** best hypothesis **within a given hypothesis set** ("Qualification")
    - All validation schemes aim to **select** best hypothesis **out of the best one from each available**

    - **hypothesis set** ("Final")
    - All testing methods aim to **evaluate** the performance of the selected hypothesis on real out-of-sample data
8. Since validation uses samples from the input/training set, it still reports **more optimistic** errors than testing, which uses out-of-sample data
    - Always **choose** hypothesis using training+validation, but **report performance** on test set