Cross Validation

Andrew Nobel

March, 2020

Stochastic Framework for Classification, revisited

Recall: Two components

- Population: generic pair $(X, Y) \in \mathcal{X} \times \{0, 1\}$ (unobserved)
- Sample: Observations $(X_1, Y_1), \ldots, (X_n, Y_n)$ iid $\sim (X, Y)$

Qu: How do we use observations?

- Training: build a classification rule from data
- Testing: assess the performance of a rule
- Validation: select among competing rules or methods

Red flag: Same observations used for more than one task

Rules vs. Procedures

Observations $D_n = (X_1, Y_1), \ldots, (X_n, Y_n)$

1. *Classification rule* is a map $\phi : \mathcal{X} \to \{0, 1\}$
   - $\phi(x)$ predicts class label of x
2. *Classification procedure* is a map $\phi_n : \mathcal{X} \times (\mathcal{X} \times \{0, 1\})^n \to \{0, 1\}$
   - $\hat{\phi}_n(x) = \phi(x : D_n)$ predicts class label of x *based on* D_n

Key point

- If data D_n is fixed then $\hat{\phi}_n$ is a classification rule
- Different data sets yield different classification rules

Example

Two classification procedures: $\phi_n =$ LDA and $\psi_n =$ LogReg

Two data sets: $D_n^a = \{(x_i^a, y_i^a) : 1 \leq i \leq n\}$ and $D_n^b = \{(x_i^b, y_i^b) : 1 \leq i \leq n\}$

1. *Same data, different method*: $\hat{\phi}_n^a(x) = \phi_n(x, D_n^a)$ and $\hat{\psi}_n^a(x) = \psi_n(x, D_n^a)$
   - Two rules. How good are they? Which is better?

2. *Same method, different data*: $\hat{\phi}_n^a(x) = \phi_n(x, D_n^a)$ and $\hat{\phi}_n^b(x) = \phi_n(x, D_n^b)$
   - Two rules. How different are they? Do they perform differently?

Deeper Questions

1. **Stability**
   - How sensitive is the rule $\hat{\phi}_n = \phi_n(x, D_n)$ to the data D_n?
   - Does a small change in one of the data points yields a big change in $\hat{\phi}n$?

2. **Aggregation and Averaging**
   - How can we combine a family of rules to get a better one?
   - How can we average a family of rules?

Risk of Rules and Procedures

1. Classification rule $\phi : \mathcal{X} \to \{0, 1\}$
   - Risk $R(\phi) = \mathbb{P}(\phi(X) \neq Y)$
2. Classification procedure $\phi_n : \mathcal{X} \times (\mathcal{X} \times \{0, 1\})^n \to \{0, 1\}$
   - Conditional risk $R(\hat{\phi}_n) = \mathbb{P}(\phi_n(X : D_n) \neq Y \mid D_n)$
   - Overall risk $\mathbb{E}R(\hat{\phi}_n) = \mathbb{P}(\phi_n(X : D_n) \neq Y)$

Importance

- Means of assessing performance
- Way to compare, select among procedures
- Way to assess the difficulty of the classification problem

Estimating the Risk of Rules and Procedures

Problem: Risk measures depend on unknown distribution of (X, Y)

One solution

- Replace probabilities and expectations by averages over observations
- Appeal to the law of large numbers and probability inequalities

Training Error

Definition: Given observations $D_n = (X_1, Y_1), \ldots, (X_n, Y_n)$ the *training error* of a rule $\phi : \mathcal{X} \to \{0, 1\}$ is

$$\hat{R}_n(\phi) = \frac{1}{n} \sum_{i=1}^{n} \mathbb{I}(\phi(X_i) \neq Y_i)$$

Fact: If ϕ is fixed then

- $\mathbb{E}[\hat{R}_n(\phi)] = R(\phi)$ and $\mathrm{Var}(\hat{R}_n(\phi)) = n^{-1} R(\phi)(1 - R(\phi))$
- $\hat{R}_n(\phi) \sim n^{-1}$ Bin$(n, R(\phi))$
- $\hat{R}_n(\phi) \to R(\phi)$ in probability as n tends to infinity

Training Error, cont.

Fact: (Chebyshev) If ϕ is fixed then for every $t > 0$

$$\Pr\left(|\hat{R}_n(\phi) - R(\phi)| > t\right) \;\leq\; \frac{R(\phi)(1 - R(\phi))}{n\,t^2}$$

Fact: (Hoeffding) If ϕ is fixed then for every $t > 0$

$$\Pr\left(|\hat{R}_n(\phi) - R(\phi)| > t\right) \;\leq\; 2\exp\{-2nt^2\}$$

Problems Arising from Double Use of Training Data

Suppose the rule $\hat{\phi}_n$ is obtained from training data D_n

Question: Is $\hat{R}_n(\hat{\phi}_n)$ a good estimate of the conditional risk $R(\hat{\phi}_n)$?

Answer: No!

Root of the problem: $\hat{\phi}_n$ and $\hat{R}_n$ based on *same* observations D_n

- By design, $\hat{\phi}_n$ is fit to D_n
- $\hat{\phi}_n$ should perform better on D_n than an independent set D'_n
- Expect $\hat{R}_n(\hat{\phi}_n)$ to *underestimate* $R(\hat{\phi}_n)$

Example: Training error of 1-NN rules is always zero!

Training and Test Sets

1. Split observations $(X_1, Y_1), \ldots, (X_{n+m}, Y_{n+m})$ into two disjoint groups
   - Training set $D_n = (X_1, Y_1), \ldots, (X_n, Y_n)$
   - Test set $D_m = (X_{n+1}, Y_{n+1}), \ldots, (X_{n+m}, Y_{n+m})$

2. Use *training set* D_n to construct a classification rule $\hat{\phi}_n(x) = \phi_n(x : D_n)$

3. Assess performance of $\hat{\phi}_n$ via its average error rate on *test set* D_m

$$\hat{R}_m(\hat{\phi}_n) = m^{-1} \sum_{j=1}^{m} \mathbb{I}(\hat{\phi}_n(X_{n+j}) \neq Y_{n+j})$$

Training and Test Sets, cont.

Fact: The sets D_n and D_m are independent. Moreover,

- $\mathbb{E}[\hat{R}_m(\hat{\phi}_n) \,|\, D_n] = \mathbb{P}(\hat{\phi}_n(X) \neq Y \,|\, D_n) = R(\hat{\phi}_n)$
- For each $t > 0$,

$$\mathbb{P}\left(|\hat{R}_m(\hat{\phi}_n) - R(\hat{\phi}_n)| > t \,|\, D_n\right) \;\leq\; \frac{R(\hat{\phi}_n)(1 - R(\hat{\phi}_n))}{m\, t^2}$$

Downside: When data is hard to come by or expensive to obtain, splitting into training and test sets is a luxury, and not always feasible.

Cross Validation

Ingredients

- Observations $D_N = (X_1, Y_1), \ldots, (X_N, Y_N)$ iid $\sim (X, Y)$
- Fold-count $k \geq 2$ such that $N = km$. Usually $2 \leq k \leq 10$
- Classification procedure ϕ for data sets of size $(k-1)m$

Idea

- Split observations into equal size chunks
- Use each chunk to test rule produced from rest of the observations
- Average resulting error rates

Cross Validation

1. Divide $D_N = \tilde{D}_1 \cup \cdots \cup \tilde{D}_k$ into k *folds* each with m points
   - $\tilde{D}_\ell = \{(X_i, Y_i) : m(\ell - 1) < i \leq m\ell\}$
   - $\tilde{D}_{\setminus \ell} = \bigcup_{j \neq \ell} \tilde{D}_j$

2. Let $\hat{\phi}^\ell(x) = \phi(x : \tilde{D}_{\setminus \ell})$ be the rule derived from training set $\tilde{D}_{\setminus \ell}$

3. Evaluate error rate of $\hat{\phi}^\ell$ using test set $\tilde{D}_\ell$

$$\hat{R}^\ell(\hat{\phi}^\ell) = m^{-1} \sum_{m(\ell-1)<i\leq m\ell} \mathbb{I}(\hat{\phi}^\ell(X_i) \neq Y_i)$$

4. k-fold cross validated risk of procedure ϕ given by

$$\hat{R}^{\text{k-CV}}(\phi) = k^{-1} \sum_{\ell=1}^{k} \hat{R}^\ell(\hat{\phi}^\ell)$$

Overview of Cross Validation

For each fold $\ell = 1, \ldots, k$

- $\tilde{D}_{\setminus \ell}$ used to train $\hat{\phi}^{\ell}$
- $\tilde{D}_{\ell}$ used to test $\hat{\phi}^{\ell}$
- Test error is $\hat{R}^{\ell}(\hat{\phi}^{\ell})$

CV risk $\hat{R}^{\text{k-CV}}(\phi)$ is just the average of the fold-wise test errors $\hat{R}^{\ell}(\hat{\phi}^{\ell})$

Upshot: $\hat{R}^{\text{k-CV}}(\phi)$ is an estimate of the *expected* error $\mathbb{E}R(\phi)$ of the classification procedure ϕ. It does not estimate conditional error.