Input-Dependent Estimation of Generalization Error under Covariate Shift

Masashi Sugiyama
Tokyo Institute of Technology, Tokyo, Japan

Klaus-Robert Müller, Matthias Krauledat
Fraunhofer FIRST, Berlin,
University of Potsdam, Germany

Pradeep Shenoy
Fraunhofer FIRST, Berlin,
U Washington, Seattle
Overview

- Introduction to model selection
- Covariate Shift
- Applications
  - Regression
  - Brain Computer Interfacing
- Conclusion
Regression Problem

From \( \{(x_i, y_i)\}_{i=1}^{n} \), obtain \( \hat{f}(x) \) such that it is as close to \( f(x) \) as possible.
Typical Method of Learning

- Linear regression model

\[ \hat{f}(x) = \sum_{i=1}^{p} \alpha_i \varphi_i(x) \]

- Ridge regression

\[
\min_{\{\alpha_i\}} \left[ \sum_{i=1}^{n} \left( \hat{f}(x) - y_i \right)^2 + \lambda \sum_{i=1}^{n} \alpha_i^2 \right]
\]

\[ \lambda : \text{Ridge parameter} \]
Model Selection

Choice of the model affects Heavily the learned function

(Model refers to, e.g., the ridge parameter $\lambda$)
Ideal Model Selection

Determine the model $\lambda$ such that a certain generalization error $J$ is minimized.

$$\hat{\lambda} = \arg\min_{\lambda} J(\lambda)$$
Practical Model Selection

However, the generalization error $J$ can not be directly calculated since it includes unknown learning target function

Determine the model $\lambda$ such that an estimator $\hat{J}$ of the generalization error is minimized.

$$\hat{\lambda} = \arg\min_{\lambda} \hat{J}(\lambda)$$

We want to have an accurate estimator $\hat{J}$

(not true for Bayesian model selection using evidence)
Two Approaches to Estimating Generalization Error (1)

Try to obtain unbiased estimators

\[ \mathbb{E}[\hat{J}(\lambda) - J(\lambda)] = 0 \]
\[ \approx 0 \]
\[ \rightarrow 0 \]

- CP (Mallows, 1973)
- Cross-Validation
- Akaike Information Criterion (Akaike, 1974)

Interested in typical-case performance
Two Approaches to Estimating Generalization Error (2)

Try to obtain probabilistic upper bounds

$$J(\lambda) \leq \hat{J}(\lambda)$$

with probability $1 - \eta$

- VC-bound (Vapnik & Chervonenkis, 1974)
- Span bound (Chapelle & Vapnik, 2000)
- Concentration bound (Bousquet & Elisseeff 2001) etc.

Interested in worst-case performance
Popular Choices of Generalization Measure

- **Risk**
  \[
  R(\hat{f}) = \int \ell (\hat{f}(x), y) p(x, y) dx dy
  \]
  e.g., \( \ell (\hat{y}, y) = (\hat{y} - y)^2 \)

- **Kullback-Leibler divergence**
  \[
  KL(p, \hat{p}) = \int p(x, y) \log \frac{p(y|x)}{\hat{p}(y|x)} dx dy
  \]

  \( p(y|x) \): Target density
  \( \hat{p}(y|x) \): Learned density
Concerns in Existing Methods

- The used approximation often requires a large (infinite) number of training examples for justification (asymptotic approximation). They do not work with small samples.

- Generalization measure should be integrated over $p(x, y)$, from which training examples $\{(x_i, y_i)\}_{i=1}^{n}$ are drawn. They cannot be used for transduction (estimating error at a point of interest).
Our Interests

We are interested in

- Estimating the generalization error with accuracy guaranteed for small (finite) samples
- Estimating the transduction error (the error at a point of interest)
- Investigating the role of unlabeled samples (samples $\{x_i\}_{i=1}^{n}$ without output sample values $\{y_i\}_{i=1}^{n}$)
Our Generalization Measure

- \( H \): Functional Hilbert space
- We assume \( f, \hat{f} \in H \)

\[
J(\hat{f}) = \| \hat{f} - f \|^2
\]

\( \| \cdot \| \): Norm in the function space \( H \)

\( J = 0 \iff \hat{f} = f \)
We are interested in typical performance so we estimate the expected generalization error over the noise:

\[ EJ = E\|\hat{f} - f\|^2 \]

We do not take expectation over input points \( \{x_i\}_{i=1}^n \) Data-dependent!

We do not assume \( x_i^{\text{train}}, x_i^{\text{test}} \sim i.i.d. p(x) \) Advantageous in active learning!
Bias / Variance Decomposition

$$EJ = E||\hat{f} - f||^2$$

$$= ||E\hat{f} - f||^2 + E||\hat{f} - E\hat{f}||^2$$

Bias

Variance

RKHS $H$

$E: \text{Expectation over noise}$

$\sigma^2: \text{Noise variance}$

$X^*: \text{Adjoint of } X$

$\hat{f} = Xy$

We want to estimate the bias!
Tricks for Estimating Bias

Suppose we have a linear operator $X_u$ that gives an unbiased estimate $\hat{f}_u$ of $f$.

$$E\hat{f}_u = f : \hat{f}_u = X_u y$$

$E$ : Expectation over noise $y = (y_1, y_2, \ldots, y_n)^T$

We use $\hat{f}_u$ for estimating the bias of $\hat{f}$.
Unbiased Estimator of Bias

RKHS $H$

$Xy = \hat{f}$

Rough estimate

$\hat{f}_u = X_u y$

$Bias = \|E\hat{f} - f\|^2$

$= \|\hat{f} - \hat{f}_u\|^2 - 2\langle X_0 Af, X_0 \epsilon \rangle - \|X_0 \epsilon\|^2$

$\hat{Bias} = \|\hat{f} - \hat{f}_u\|^2 - 0 - \sigma^2 \text{tr} (X_0 X_0^*)$

$E \hat{Bias} = Bias$
Subspace Information Criterion (SIC)

\[ SIC = \| \hat{f} - \hat{f}_u \|^2 - \sigma^2 \text{tr} (X_0 X_0^*) + \sigma^2 \text{tr} (XX^*) \]

\[ X_0 = X - X_u \]

SIC is an unbiased estimator of the generalization error with finite samples

\[ \mathbb{E} SIC = \mathbb{E} \| \hat{f} - f \|^2 \]

\( \mathbb{E} \) : Expectation over noise
Obtaining Unbiased Estimate

We need $X_u$ that gives an unbiased estimate of learning target $f(x)$.

$$E\hat{f}_u = f : \hat{f}_u = X_u y$$

$X_u$ exists if and only if

$$\{K(x, x_i)\}_{i=1}^n$$ span the entire space $H$.

When this is satisfied, $X_u$ is given by $X_u = A^\dagger$.

We can enjoy all the features!

(Unlabeled samples, transductive inference etc.)
Preparation for Covariate Shift setting: Standard Regression Problem

- Learning target function: \( f(x) \)

- Training examples:
  \[ \{(x_i, y_i) \mid y_i = f(x_i) + \epsilon_i\}_{i=1}^n \]

- Test input:
  \[ \{t_i \mid t_i \sim i.i.d. p_t(x)\}_{i=1}^m \]

- Goal: Obtain approximation \( \hat{f}(x) \) that minimizes expected error for test inputs (or generalization error)

\[
J = \int \left( \hat{f}(t) - f(t) \right)^2 p_t(t) dt
\]
Training Input Distribution

- Common assumption:

  Training input \( \{x_i\}_{i=1}^n \) follows the same distribution as test input:

  \[ x_i \overset{i.i.d.}{\sim} p_t(x) \]

- Here, we suppose distributions are different.

  \[ x_i \overset{i.i.d.}{\sim} p_x(x) \]
  \[ t_i \overset{i.i.d.}{\sim} p_t(x) \]
  \[ p_x(x) \neq p_t(x) \]

Covariate shift
Is covariate shift important to investigate?

Yes! It often happens in reality.

- Interpolation / extrapolation
- Active learning (experimental design)
- Classification from imbalanced data
Asymptotically unbiased if model is correct.
Asymptotically biased for misspecified models.
Need to reduce bias.
Weighted Least Squares for Covariate Shift (Shimodaira, 2000)

\[
\min_{\alpha} \left[ \sum_{i=1}^{n} \frac{p_t(x_i)}{p_x(x_i)} \left( \hat{f}(x_i) - y_i \right)^2 \right]
\]

\[p_x(x), p_t(x) : \text{Assumed known and strictly positive}\]

- Asymptotically unbiased for misspecified models.
- Can have large variance.
- Need to reduce variance.
\( \lambda \)-Weighted Least Squares

\[
\min_{\alpha} \left[ \sum_{i=1}^{n} \left( \frac{p_t(x_i)}{p_x(x_i)} \right)^{\lambda} \left( \hat{f}(x_i) - y_i \right)^2 \right]
\]

- Large bias, Small variance
- (Intermediate)
- Small bias, Large variance

\( \lambda = 0 \)

\( \lambda = 0.5 \)

\( \lambda = 1 \)

\( \lambda \) should be chosen appropriately!

(Model Selection)
Generalization Error Estimation under Covariate Shift

- \( \lambda \) is determined so that (estimated) generalization error is minimized.
- However, standard methods such as cross-validation is heavily biased.
- **Goal**: Derive better estimator.
Setting

- i.i.d. noise with mean 0 and variance $\sigma^2$
- Linear regression model:

$$\hat{f}(\mathbf{x}) = \sum_{i=1}^{p} \alpha_i \varphi_i(\mathbf{x})$$

- $\lambda$-weighted least squares:

$$\min_{\alpha} \left[ \sum_{i=1}^{n} \left( \frac{p_t(x_i)}{p_x(x_i)} \right)^\lambda \left( \hat{f}(x_i) - y_i \right)^2 \right]$$

$$\hat{\alpha} = L y$$

$$L = (X^\top D^\lambda X)^{-1} X^\top D^\lambda$$

$$\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_p)^\top$$

$$y = (y_1, y_2, \ldots, y_n)^\top$$

$$X_{i,j} = \varphi_j(\mathbf{x}_i)$$

$$D = \text{diag} \left( \frac{p_t(x_i)}{p_x(x_i)} \right)$$
Decomposition of Generalization Error

\[ J = \int \left( \hat{f}(x) - f(x) \right)^2 p_t(x) dx \]
\[ = \| \hat{f} - f \|^2_{L_2(p_t)} \]
\[ = \| \hat{f} \|^2_{L_2(p_t)} - 2 \langle \hat{f}, f \rangle_{L_2(p_t)} + \| f \|^2_{L_2(p_t)} \]

Accessible Estimated Constant (ignored)

We estimate \[ Z \equiv \langle \hat{f}, f \rangle_{L_2(p_t)} \]
Orthogonal Decomposition of Learning Target Function

\[ f(x) = g(x) + r(x) \]

\[ \langle \varphi_i, r \rangle_{L_2(p_t)} = 0 \]

\[ g(x) = \sum_{i=1}^{p} \alpha_i^* \varphi_i(x) \]

\( \alpha^* \): Optimal parameter

\[ Z = \langle \hat{f}, f \rangle_{L_2(p_t)} = \langle \hat{f}, g \rangle_{L_2(p_t)} = \langle U \hat{\alpha}, \alpha^* \rangle \]

\[ U_{i,j} = \langle \varphi_i, \varphi_j \rangle_{L_2(p_t)} \]
Suppose we have

- \( L_u \), which gives linear unbiased estimator of \( \alpha^* \)
  \[ \mathbb{E}_\epsilon L_u y = \alpha^* \]
- \( \sigma_u^2 \): Unbiased estimator of noise variance
  \[ \mathbb{E}_\epsilon \sigma_u^2 = \sigma^2 \]

Then we have an unbiased estimator of \( \mathbb{E}_\epsilon Z \):

\[ \hat{Z} \equiv \langle ULY, L_u y \rangle - \sigma_u^2 \text{tr}(ULL_u^T) \]

But \( L_u, \sigma_u^2 \) are not always available.

Use approximations instead
Approximations of $L_u, \sigma_u^2$

1. $\hat{L}_u = (X^\top DX)^{-1} X^\top D$

2. $\hat{\sigma}_u^2 = \frac{\| y - H y \|^2}{n - p}$

$H = X(X^\top X)^{-1} X^\top$

$\mathbb{E}_\epsilon L_u y = \alpha^*$

$\mathbb{E}_\epsilon \sigma_u^2 = \sigma^2$

- If model is correct,
  $\mathbb{E}_\epsilon \hat{L}_u y = \alpha^*$
  $\mathbb{E}_\epsilon \hat{\sigma}_u^2 = \sigma^2$

- If model is misspecified,
  $\mathbb{E}_\epsilon \hat{L}_u y \to \alpha^*$
  $\mathbb{E}_\epsilon \hat{\sigma}_u^2 \not\to \sigma^2$ (as $n \to \infty$)
$\hat{J} = \langle ULy, Ly \rangle - 2\langle ULy, \hat{L}_u y \rangle + 2\sigma_u^2 \text{tr}(ULL\hat{L}_u^\top)$

Bias: $B_\epsilon = \mathbb{E}_\epsilon[\hat{J} - J] + C$

- If model is correct,
  $B_\epsilon = 0$

- If model is almost correct,
  $B_\epsilon = O(\delta) \quad \delta = \max\{r(x_i)\}$

- If model is misspecified,
  $B_\epsilon = O_p(n^{-\frac{1}{2}})$
Simulation (Toy)

\[
\min_{\alpha} \left[ \sum_{i=1}^{n} \left( \frac{p_t(x_i)}{p_x(x_i)} \right)^\lambda \left( \hat{f}(x_i) - y_i \right)^2 \right]
\]

\[
\lambda = 0 \quad \lambda = 0.5 \quad \lambda = 1
\]
Results

10-fold cross-validation

True generalization error

Proposed estimator
Simulation (Abalone from DELVE)

- Estimate the age of abalones from 7 physical measurements.
- We add bias to 4th attribute (weight of abalones).
- Training and test input densities are estimated by standard kernel density estimator.

\[ \hat{f}(x) = \alpha_1 + \sum_{i=1}^{7} \alpha_{i+1} x^{(i)} \]
Generalization Error Estimation

\( n = 50 \)

- True gen error
- 10CV
- Proposed

\( n = 200 \)

\( n = 800 \)

Mean over 300 trials
## Test Error After Model Selection

### Extrapolation in 4\textsuperscript{th} attribute

<table>
<thead>
<tr>
<th>n</th>
<th>50</th>
<th>200</th>
<th>800</th>
</tr>
</thead>
<tbody>
<tr>
<td>OPT</td>
<td>9.86 ± 4.27</td>
<td>7.40 ± 1.77</td>
<td>6.54 ± 1.34</td>
</tr>
<tr>
<td>Proposed</td>
<td>11.67 ± 5.74</td>
<td>7.95 ± 2.15</td>
<td>6.77 ± 1.40</td>
</tr>
<tr>
<td>10CV</td>
<td>10.88 ± 5.05</td>
<td>8.06 ± 1.91</td>
<td>7.24 ± 1.37</td>
</tr>
</tbody>
</table>

### Extrapolation in 6\textsuperscript{th} attribute

<table>
<thead>
<tr>
<th>n</th>
<th>50</th>
<th>200</th>
<th>800</th>
</tr>
</thead>
<tbody>
<tr>
<td>OPT</td>
<td>9.04 ± 4.04</td>
<td>6.76 ± 1.68</td>
<td>6.05 ± 1.25</td>
</tr>
<tr>
<td>Proposed</td>
<td>10.67 ± 6.19</td>
<td>7.31 ± 2.24</td>
<td>6.20 ± 1.33</td>
</tr>
<tr>
<td>10CV</td>
<td>10.15 ± 4.95</td>
<td>7.42 ± 1.81</td>
<td>6.68 ± 1.25</td>
</tr>
</tbody>
</table>
Classification

Figure 4.7 Training input density function $p_x(x)$ (left) and test input density function $p_t(x)$ (right) for binary classification with imbalanced data.

From Sugiyama & Müller in Press
Brain Computer Interfacing
Conclusions

- Covariate shift: Training and test input distributions are different
- Ordinary LS: Biased
- Weighted LS: Unbiased but large variance.
- $\lambda$-WLS: Model selection needed.
- Cross-validation: Biased
- Proposed generalization error estimator:
  - Exactly unbiased (correct models)
  - Asymptotically unbiased (misspecified models)
- Applications
  - Regression & Classification
  - BCI. Online Adaptation?
Future Issue: Shifting distributions within experiment