Generalization for Streaming Data

Michael Spece

Departments of Machine Learning and Statistics
Carnegie Mellon University

June 11, 2015

Learning Game/Decision Theoretic Setup

Fix $T \in \mathbb{Z}_+$

- Environment generates T observations $\mathbf{y} := y_1, \cdots, y_T$
- Learner estimates $\hat{x}(\mathbf{y})$

Definition of Generalization

Generalization error (a measure of overfitting)

$$\mathop{\mathbb{E}}_{\mathbf{y}} \left| \frac{1}{T} \sum_{t=1}^{T} \ell(\hat{x}(\mathbf{y}), y_t) - \mathop{\mathbb{E}}_{y_0} \ell(\hat{x}(\mathbf{y}), y_0) \right|$$

Online Learning Refinement (Online to Batch Conversion)

A specific way of computing the estimate (compute it online):

Fix $T \in \mathbb{Z}_+$

- For $t \in \{1, \cdots, T\}$
  - Environment generates y_t
  - Learner "instantaneously" estimates $\hat{x}'_t(y_1, \cdots, y_t)$
- Learner estimates $\hat{x} := \overline{\hat{\mathbf{x}}'}$

Void for Generalizing from Streaming Data

Drawbacks of batch perspective for streaming data

- Final estimate is not equal to the last sequential estimation
- Empirical risk is not equal to actual loss suffered under sequential estimation
- Given the definition of generalization error, restricts the notion of cumulative loss to mean

Solution

Fix $T \in \mathbb{Z}_+$

- Environment generates a single observation $\mathbf{y} := (y_1, \cdots, y_T)$
- Learner estimates $\hat{\mathbf{x}}(\mathbf{y})$

Generalization error becomes

$$\mathop{\mathbb{E}}_{\mathbf{y}} \left| \ell(\hat{\mathbf{x}}(\mathbf{y}), \mathbf{y}) - \mathop{\mathbb{E}}_{\mathbf{y_0}} \ell(\hat{\mathbf{x}}(\mathbf{y}), \mathbf{y_0}) \right|$$

Online Learning Refinement

Compute estimate online

Fix $T \in \mathbb{Z}_+$

- Environment generates $\mathbf{y}$
- For $t \in \{1, \cdots, T\}$
  - Learner "instantaneously" estimates $\hat{x}_t(y_1, \cdots, y_t)$
- Learner estimates $\hat{\mathbf{x}} := (\hat{x}_1, \cdots, \hat{x}_T)$

Summary

Generalization error is a measure of overfitting

Applying to streaming data (one vector-valued observation, vector-valued estimation), generalization error becomes

$$\mathop{\mathbb{E}}_{\mathbf{y}} \left| \ell(\hat{\mathbf{x}}(\mathbf{y}), \mathbf{y}) - \mathop{\mathbb{E}}_{\mathbf{y_0}} \ell(\hat{\mathbf{x}}(\mathbf{y}), \mathbf{y_0}) \right|$$

Features	Expanded Applications
Preserves ordering of estimations	Dynamic models
Single loss	Non-convex cumulative losses
Minimal assumptions	Non-stationary data

Bounding

Given an online learning algorithm, one can attempt to show that the algorithm generalizes by bounding its generalization error

Certain functional forms and regularity conditions entail a martingale bound

Example functional form:

$$\ell(\hat{x}(\mathbf{y}), \mathbf{y}) = B(\ell'_1(\hat{x}_1, y_2), \cdots, \ell'_{T-1}(\hat{x}_{T-1}, y_T))^1$$

Example regularity conditions: B nonnegative, subadditive, and (for better rates) smooth

[1] This form appears in Rahklin et al. 2010.

Implication

Martingale bound is in the form of a supremum of the norms of martingale difference sequences (MDSs)

Under regularity conditions, the supremum grows sublinearly in T, i.e. generalization holds.

Generality to which results hold

More general results can simplify notation

Algorithmic Analysis

Generalization error can be computed for simulated data or, with additional assumptions, estimated from data

Does generalization error help explain the improved performance of online forecasters?