Information Theoretic Model Validation by Approximate Optimization

Joachim M. Buhmann

Computer Science Department, ETH Zurich
Overview

- Motivation of information theory for optimization
- Approximation capacity of a cost function
- Examples
  - Binary symmetric channel
  - Cluster validation
  - Role based access control
  - Robust SVD
- Conclusion and outlook
What is the central challenge of pattern recognition?

- I) Finding the “right” model? II) Validating a model?
- **Hypothesis:** Validation of pattern recognition models is the fundamental challenge!
  
  ➞ Algorithmic search for PR models should prefer **noise tolerant** and **expressive** models over brittle, simplistic ones! *(stability vs informativeness)*
  
  ➞ **Information theory** enables us to measure the context sensitive information content of models!
Information Theory & Pattern Recognition

- IT-Components
  - Code vectors $\subseteq \{\text{strings}\} = \text{hypothesis class}$
  - Noisy channel
  - Decoder: minimize Hamming distance

- Criterion for error free communication
  $\Rightarrow$ mutual information

- Pattern Recognition elements
  - Approximation sets $\subseteq \text{hypothesis class}$
  - Noisy optimization problem
  - Decoding by approximate optimization of test instance

- model validation based on guaranteed approximations
  $\Rightarrow$ mutual information
Examples of hypothesis classes

- **Partitions or clusterings**: compactness/connectivity costs

- **Trees or dendrograms**: partitions with ultrametricity; Tree depth? # leaves?

- **Graphical models**: structured probability models; # nodes/edges?

The hypothesis class is much smaller than the data space!
Code problems define approximation sets

- **IT:** Space of strings is partitioned by code vectors
- **PR:** Hypothesis class is partitioned by code problems

![Code vectors and approximation set diagram]
Pattern recognition as optimization

- **Given**: data $X \in \mathcal{X}$ in data (input) space $\mathcal{X}$
- **Goal**: Learn structure from data, i.e., interpret data relative to a hypothesis class
- **Hypothesis class** $\mathcal{C}$ with hypotheses (solutions)

$$c : \mathcal{X} \rightarrow \mathbb{K} \quad (\text{e.g., } \mathbb{B}^n \text{ or } \{1, \ldots, k\}^n)$$

$$X \mapsto c(X)$$

- **Cost function** to define a partial order on $\mathcal{C}$

$$R : \mathcal{C} \times \mathcal{X} \rightarrow \mathbb{R}_{\geq 0}$$

$$(c, X) \mapsto R(c, X)$$
Symmetries of the Learning Problem

Assume (!) that the cost function $R$ is equivariant under the transformations

$$\Sigma = \{ \sigma : R(c, X) = R(\sigma \circ c, \sigma \circ X) \}$$

Minimizer:

$$c^\perp(X) = \arg\min_{c \in C} R(c, X)$$

$$c^\perp(\sigma \circ X) = \sigma \circ c^\perp(X)$$

Approximation set:

$$c \in C_\gamma(X) \equiv \{ c : R(c, X) \leq R(c^\perp, X) + \gamma \}$$
How to generate code problems?

1. Combinatorial optimization problems: \textit{permutation} of combinatorial components, e.g., vertices in graphs

2. Localization problems: \textit{shifts} of data

3. Orientation problems (PCA, SVD): \textit{rotations}
Ex.: Graph Cut - Clustering in two groups

- **Graph** representation: vertices denote objects
  edges express (dis)similarities

- **Hypothesis class:** all cuts of a graph
Code problem generation for Graph Cut

graph cut code problems

graph cut test problem
Coding with Graph Cut approximation sets

define a set of code problems

problem generator PG

\[ R(\cdot, X^{(1)}) \]

\[ R(\cdot, X^{(1)}) \]

\( \{\sigma_1, \ldots, \sigma_{2n\rho}\} \)

sender

receiver

\( \sigma \circ X^{(1)} \)
Communication by approximation sets

1. Sender sends a permutation index $\sigma_s$ to problem generator.

2. Problem generator sends a new problem with permuted indices to receiver without revealing $\sigma_s$.

3. Receiver identifies the permutation $\hat{\sigma}$ by comparing approximation sets.

\[
R(\cdot, \sigma_s \circ \mathbf{X}^{(2)}), \text{ s.t. } \mathbf{X}^{(1)}, \mathbf{X}^{(2)} \sim P(\mathbf{X})
\]
Communication Process

- Receiver has to compare sets of hypotheses $C_\gamma(X^{(1)})$ of training instance (code problem) with approximate clusterings $C_\gamma(X^{(2)})$ of the test data.

- Define a mapping $\psi : C(X^{(1)}) \rightarrow C(X^{(2)})$

- Decoding by overlap maximization $(\tilde{X}^{(2)} := \sigma_s \circ X^{(2)})$

$$\hat{\sigma} = \arg \max_{\sigma} \left| \psi \circ C_\gamma(\sigma \circ X^{(1)}) \cap C_\gamma(\tilde{X}^{(2)}) \right|$$
Error Events and Approximation Capacity

- Sender selects transformation \( \sigma_s \Rightarrow C_\gamma(\sigma_s \circ X^{(1)}) \)

- Joint approximation sets

\[
\Delta C_j = \psi \circ C_\gamma(\sigma_j \circ X^{(1)}) \cap C_\gamma(\sigma_s \circ X^{(2)})
\]

for \( 1 \leq j \leq 2^{n\rho}, \ j \neq s \)

- Error events:

\[
|\Delta C_j| \geq |\Delta C_s| \text{ for } 1 \leq j \leq 2^{n\rho}, \ j \neq s
\]
Error Probability

- Conditional error

\[ P(\hat{\sigma} \neq \sigma_s | \sigma_s) = P(\max_{j \neq s} |\Delta C_j| > |\Delta C_s| | \sigma_s) \]

Union bound

\[ \leq \sum_{j \neq s} P(|\Delta C_j| > |\Delta C_s| | \sigma_s) \]

Assume random transformations \( \sigma \in \Sigma \)

\[ \leq 2^{n\rho} P(\Delta C \neq s > |\Delta C_s| | \sigma_s) \]

\[ = 2^{n\rho} \mathbb{E}_{X(1,2)} \mathbb{E}_{\sigma \neq s} \left[ \mathbb{I}\{|\Delta C \neq s| \geq |\Delta C_s|\} | \sigma_s \right] \]

The random transformation statistically decouples the two approximation sets in \( \Delta C \neq s \)
Bounding of error

\[ \mathbb{E}_{\sigma \neq s} \left[ \mathbb{I}\{ |\Delta C_{\neq s}| \geq |\Delta C_s| \} \right] \leq \frac{1}{|\{\sigma \neq s\}|} \sum_{\{\sigma \neq s\}} \frac{|\Delta C_{\neq s}|}{|\Delta C_s|} \]

(a) Bound on indicator function \( \mathbb{I}\{ x \geq a \} \leq x/a \)

(b) Averaging over random transformation

(c) Definition of mutual information

\[ \exp (-n I_{\gamma}(\sigma \neq s, \hat{\sigma})) \]
Condition of vanishing total error

\[
\lim_{n \to \infty} P(\hat{\sigma} \neq \sigma_s | \sigma_s) = 0 \quad \text{yields}
\]

- Rate is bounded by mutual information

\[
\rho \log 2 < \frac{1}{n} \log \frac{|\{\sigma \neq s\}|}{|C_1^{(1)}| |C_2^{(2)}|} \log \frac{|C_2^{(2)}|}{|C_2^{(2)}|} - \log \frac{|C_2^{(2)}|}{\Delta C_s} = 1
\]

\[
\equiv I_2^{(1)}(\sigma \neq s, \hat{\sigma})
\]

- Lower bound: generalize Fano’s inequality to ASC (work in progress)
Model Selection by Maximization of Approximation Capacity

- Optimize the communication channel w.r.t. approximation quality $\gamma$ ($\beta$), topology and metric of solution space, cost function $R(\ldots)$, transfer function $\psi$. 

\[
R(\cdot, \sigma_s \circ X^{(2)}), \text{s.t. } X^{(1)}, X^{(2)} \sim P(X)
\]
Ex.: Binary Coding

- Hypothesis class: set of binary strings
  \[ \xi^{(1)} = (\xi_1^{(1)}, \xi_2^{(1)}, \ldots, \xi_n^{(1)}), \xi^{(2)} \in \{-1, 1\}^n \]

- Costs: \[ R(s, \xi^{(1)}) = \sum_{i=1}^{n} \mathbb{I}_{\{s_i \neq \xi_i^{(1)}\}} \]

- Mutual information: \( (\delta = \frac{1}{n}|\{i : \xi_i^{(1)} \neq \xi_i^{(2)}\}|) \)
  \[ I_\beta = \ln 2 + (1 - \delta) \ln \cosh \beta - \ln(\cosh \beta + 1) \]
  \[ (*) = \ln 2 + (1 - \delta) \ln(1 - \delta) + \delta \ln \delta \]
  if \((*) \frac{dI_\beta}{d\beta} = 0\) holds

- ASC for Hamming distance yield capacity of binary symmetric channel!
2d Mixture Model Estimation

Experimental Setting:
2 source Gaussians, 
n=10000, d=2, \( \Delta \mu = 2 \)

Generalization error
\[ \mathbb{E} \left[ R(c^{(1)}, X^{(2)}) \right] / n \]

Approximation Capacity
Gibbs sampling with 4 clusters

Experimental Setting:
\( n=500, d=100, \) 2 source Gaussians
ordered phase, up to 4 estimated Gaussians

\[
\mathbb{E} \left[ R(c^{(1)}, X^{(2)}) \right]/n
\]
High Dimensional Density Estimation


Phase Diagram:
mixture of 2 Gaussians

n=500; d=100, 500, 3000; \( \alpha := \frac{n}{d} \)

Overlap: \( r = u_0^{-1} \langle \Delta \mu, \Delta \mu_0 \rangle \)

Approximation Capacity

\( \alpha = 5 \)

\( \alpha = 1 \)

\( \alpha = 1/6 \)
Denoising Binary Matrices by rank-k approximation

Boolean matrix with 40% random entries

Continuous rank-k approximation

$X_5 = U_5 S_5 V_5$

Rounding as approximation

$g(X_k) = \text{round}(X_k)$

$X = USV$
Maximum of approximation capacity selects optimal rank $k$

- Integrate over variations of the signal matrix $U$.

$$I_\beta(\sigma_j, \hat{\sigma}) = \frac{1}{n} \log \frac{\left\{|\{\sigma_j\}\| \Delta C_\beta(X^{(1)}, X^{(2)})\right\}}{|C_\beta(X^{(1)})||C_\beta(X^{(2)})|}$$
Role-Based Access Control

- **Given:** Binary user permission matrix

- **Discretionary Access-Control:**
  Direct Assignments of users to permissions

- **Role-Based Access Control (RBAC):** Permissions are granted via roles

Diagrams showing direct user-permission assignments, role-permission assignments, and user-role assignments.
Role-Mining for RBAC

- **Role-Mining**: Given a user-permission assignment matrix $X$, find a set of roles $U$ and assignments $Z$ such that

$$X \approx U \otimes Z$$

- **Multi Assignment Clustering**: generative approach including noise model, inference with DA
Synthetic Data: Parameter Accuracy vs. Approximation Capacity

MAC: More accurate estimators for centroids, it yields higher approximation capacity than SAC.
Real-World Data: Prediction Error vs. Approximation Capacity

- **Generalization**: Can roles predict permissions of new users?
  1. Use few permissions (20%) to determine role set
  2. Predict hidden/missing permissions (80%).
- Centroids with maximal capacity yield minimal generalization error
Conclusion

- **Quantization**: Noise quantizes mathematical structures (hypothesis classes) => symbols
- These symbols can be used for **coding**!
- Optimal error free coding scheme determines **approximation capacity** of a model class.
  => Bounds for robust optimization.
  => **Quantization** of hypothesis class measures structure specific information in data.
Future Work

- **Generalization**: replace approximation sets based on cost functions by smoothed outputs of algorithms (“smoothed generalization”)
- **Model reduction** in dynamical systems: quantize sets of ODEs or PDEs (systems biology)
- Relate **statistical complexity**, i.e. the approximation capacity, to algorithmic or computational complexity.
Philosophical speculations

- We experience a **paradigm shift from model driven reasoning to algorithm dominated reasoning** (Bernard Chazelle “The Algorithm: Idiom of Modern Science”)

=> **model validation** more essential than modeling since modeling can be algorithmically formulated as exploration of model space.

- **Ceterum censeo**: The coupling of **statistical complexity** and **algorithmic complexity** should be reconsidered in the light of **statistical learning theory** and information theory.