Learning the Kernel?
From Multitask Learning to Collaborative Filtering

Alex Smola
Yahoo! Research, Santa Clara, CA
alex.smola.org im:alex.smola skype:smolix

with Deepay Chakrabartty, Wei Chu, Markus Weimer
Outline

• Three problems - one solution
  • Multitask learning
  • Collaborative filtering
  • Learning the kernel
• Three problems - one solution
  • Multitask learning
  • Collaborative filtering
  • Learning the kernel

• Factorization approach
  • Applications
  • Extensions (Tucker factors, data integration)
  • Optimization
Outline

• Three problems - one solution
  • Multitask learning
  • Collaborative filtering
  • Learning the kernel

• Factorization approach
  • Applications
  • Extensions (Tucker factors, data integration)
  • Optimization

• Experiments
  • User profiles
  • Webpage categorization with side information
Three problems - one solution
Learning the Kernel
Learning the Kernel

• Empirical risk (how well you do on the data)
  • Classification
  • Regression
  • Ranking
  • Graphical Models (CRF, M3M)

\[
R_{\text{emp}}[f] = \frac{1}{m} \sum_{i=1}^{m} l(y_i, f(x_i))
\]
Learning the Kernel

• Empirical risk (how well you do on the data)
  • Classification
  • Regression
  • Ranking
  • Graphical Models (CRF, M3M)

\[
R_{\text{emp}}[f] = \frac{1}{m} \sum_{i=1}^{m} l(y_i, f(x_i))
\]

• Function Regularizer
  • RK Hilbert Space norm
  • Sparsity (no kernel here)

\[
\Omega[f] = \frac{1}{2} \|f\|_{\mathcal{H}}^2
\]
Learning the Kernel

• **Empirical risk (how well you do on the data)**
  - Classification
  - Regression
  - Ranking
  - Graphical Models (CRF, M3M)

\[ R_{\text{emp}}[f] = \frac{1}{m} \sum_{i=1}^{m} l(y_i, f(x_i)) \]

• **Function Regularizer**
  - RK **Hilbert Space** norm
  - Sparsity (no kernel here)

\[ \Omega[f] = \frac{1}{2} \| f \|_{\mathcal{H}}^2 \]

• **Kernel Regularizer**
  - Convex combination
  - Wishart regularizer
  - Inverse norm regularizer

\[ \Gamma[\mathcal{H}] = \text{tr} \ K \]
• Convex optimization problem

\[
\text{minimize} \quad R_{\text{emp}}[f] = \frac{1}{m} \sum_{i=1}^{m} l(y_i, f(x_i)) + \frac{1}{2} \| f \|_H^2 + \text{tr } K
\]
Learning the Kernel

- Convex optimization problem

\[
\Omega[f] = \frac{1}{2} \|f\|^2_{\mathcal{H}} + \Gamma[\mathcal{H}] = \text{tr} K
\]

\[
\Omega'[f] := \min_{\mathcal{H}} \left[ \frac{1}{2} \|f\|^2_{\mathcal{H}} + \lambda \text{tr} K \right]
\]
Learning the Kernel

- Convex optimization problem
  - Switch to nontrivial Banach space norms
  - Computation efficient via underlying RKHS

\[
\Omega[f] = \frac{1}{2} \| f \|_\mathcal{H}^2
\]

\[
\Gamma[\mathcal{H}] = \text{tr } K
\]

\[
\Omega'[f] := \min_{\mathcal{H}} \left[ \frac{1}{2} \| f \|_\mathcal{H}^2 + \lambda \text{ tr } K \right]
\]

\[
R_{\text{emp}}[f] = \frac{1}{m} \sum_{i=1}^{m} l(y_i, f(x_i))
\]
Multitask Learning
Multitask Learning

- Many tasks, use them jointly to learn a kernel

\[ \sum_j R^j_{\text{emp}}[f_j] + \sum_j \lambda_j \Omega[f_j] + \Gamma[\mathcal{H}] \]

\[ \Omega'[f_1, \ldots, f_t] \]
Multitask Learning

- Many tasks, use them jointly to learn a kernel

\[
\sum_j R^j_{\text{emp}}[f_j] + \sum_j \lambda_j \Omega[f_j] + \Gamma[\mathcal{H}]
\]

\[
\Omega'[f_1, \ldots, f_t]
\]

- Argyriou et al. 2008, 2009

\[
\text{minimize } \sum_K f_j^\top K^{-1} f_j \text{ subject to } K \succeq 0 \text{ and } \text{tr } K \leq 1
\]
Multitask Learning

- Many tasks, use them jointly to learn a kernel

\[
\sum_j R_j^{emp} [f_j] + \sum_j \lambda_j \Omega [f_j] + \Gamma [\mathcal{H}]
\]

- Argyriou et al. 2008, 2009

\[
\min_K \sum_j f_j^{\top} K^{-1} f_j \quad \text{subject to} \quad K \succeq 0 \quad \text{and} \quad \text{tr} \, K \leq 1
\]

\[
\| [f_1, \ldots, f_t] \|_{\text{KyFan}}
\]

Chakrabarty et al. 2010
Multitask Learning

- Many tasks, use them jointly to learn a kernel

\[ \sum_j R_{\text{emp}}^j [f_j] + \sum_j \lambda_j \Omega[f_j] + \Gamma[H] \]

- Argyriou et al. 2008, 2009

\[
\minimize_K \sum_j f_j^\top K^{-1} f_j \text{ subject to } K \succeq 0 \text{ and } \text{tr } K \leq 1
\]

\[ \|[f_1, \ldots, f_t]\|_{\text{KyFan}} \]

This is the nuclear norm of collaborative filtering!
From CF to MTL
From CF to MTL

- Collaborative Filtering (Srebro et al., 2005+)

\[
\sum_{(i,j) \in S} l(Y_{ij}, F_{ij}) + \lambda \| F \|_{\text{KyFan}}
\]

Equivalent formulation

\[
\sum_{(i,j) \in S} l(U_i^\top V_j, F_{ij}) + \frac{\lambda}{2} \left[ \| U \|_{\text{Frob}}^2 + \| V \|_{\text{Frob}}^2 \right]
\]
From CF to MTL

- **Collaborative Filtering (Srebro et al., 2005+)**
  \[
  \sum_{(i,j) \in S} l(Y_{ij}, F_{ij}) + \lambda \| F \|_{\text{KyFan}}
  \]
  equivalent formulation
  \[
  \sum_{(i,j) \in S} l(U_i^T V_j, F_{ij}) + \frac{\lambda}{2} \left[ \| U \|_{\text{Frob}}^2 + \| V \|_{\text{Frob}}^2 \right]
  \]

- **Multitask Learning**
  \[
  \sum_j R_{\text{emp}}[f^j] + \lambda \| [f_1, \ldots, f_t] \|_{\text{KyFan}}
  \]
  equivalent formulation (latent factor model)
  \[
  \sum_j R_{\text{emp}}^j[U_j^T f] + \frac{\lambda}{2} \left[ \| U \|_{\text{Frob}}^2 + \| f \|_{\text{Frob}}^2 \right]
  \]
Factorization
Collaborative Filtering

\[ f(i, j) = u_i^T m_j \]
Factorization

Collaborative Filtering

\[ f(i, j) = u_i^\top m_j \]

Feature based filtering (e.g. ranking)

\[ f(i, j) = \phi_u(x_i^u)^\top M \phi_m(x_j^m) \]
\[ = [U \phi_u(x_i^u)]^\top [V \phi_m(x_j^m)] \]
Factorization

Joint Model

\[ u \rightarrow u' \rightarrow m' \rightarrow m \rightarrow r \]
Joint Model

\[ f(i, j) = \left[ U \phi_u(x_i^u) + u_i + b_u \right]^\top \left[ V \phi_m(x_j^m) + m_j + b_m \right] \]
Joint Model

\[ f(i, j) = [U \phi_u(x_i^u) + u_i + b_u]^\top [V \phi_m(x_j^m) + m_j + b_m] \]
Factorization

Joint Model

multitask collaborative filtering

\[ f(i, j) = [U \phi_u(x_i^u) + u_i + b_u]^\top [V \phi_m(x_j^m) + m_j + b_m] \]
Joint Model

\[
f(i, j) = \left[ U \phi_u(x^u_i) + u_i + b_u \right] \top \left[ V \phi_m(x^m_j) + m_j + b_m \right]
\]
Applications

- Collaborative filtering (features, IDs)
- Ranking (queries, webpages)
- Multitask learning (many categories for webpages)
- Dataset integration (different ontologies on same domain)
- Time-series prediction (stock values are correlated, balance sheet)
Tensor Factorization

user

\( u \)
\( u' \)

movie

\( m' \)
\( m \)

\( r \)
Tensor Factorization

user

movie

context
Tensor Factorization

\[ f(i, j, k) = [U \phi_u(x_i^u) + u_i + b_u]^\top [V \phi_m(x_j^m) + m_j + b_m] + [U \phi_u(x_i^u) + u_i + b_u]^\top [W \phi_c(x_k^c) + c_k + b_c] \]
Tensor Factorization

\[ f(i, j, k) = [U \phi_u(x_i^u) + u_i + b_u]^\top [V \phi_m(x_j^m) + m_j + b_m] + \]
\[ [U \phi_u(x_i^u) + u_i + b_u]^\top [W \phi_c(x_k^c) + c_k + b_c] \]
Optimization

• Stochastic gradient descent on \((i,j)\) pairs one at a time ... idiot-proof simple. But nonconvex.
User profiles
Datasets

- Click prediction
  - 3M users (d=1000), 150k documents (d=100)
  - 43M (user, document) pairs

- Page classification
  - 2.8M documents, 82 classification problems
  - Trivial problem features (e.g. “SpamOrNot-USMarket”)
Click model

(b) 

Area under ROC in Test vs. Training Subset Size

- **Red squares**: with both task/article feature
- **Blue circles**: without task feature
- **Black diamonds**: with no feature

Click model
Page classification

Area under ROC in Test vs. Training Subset Size

- Red squares: with both task/sample feature
- Blue circles: without task feature
- Black diamonds: without sample feature
Effect of dimensionality

Area under ROC in Test vs. Dimension

(a) Clicks classification problems
Summary

• Three problems - one solution
  • Multitask learning
  • Collaborative filtering
  • Learning the kernel
Summary

• Three problems - one solution
  • Multitask learning
  • Collaborative filtering
  • Learning the kernel

• Factorization approach
  • Applications
  • Extensions (Tucker factors, data integration)
  • Optimization
Summary

• Three problems - one solution
  • Multitask learning
  • Collaborative filtering
  • Learning the kernel

• Factorization approach
  • Applications
  • Extensions (Tucker factors, data integration)
  • Optimization

• Experiments
  • User profiles
  • Webpage categorization with side information