Sparse Coding for Motions

Leif Johnson & Joseph Cooper

2012-05-09
Outline

Linear regression

Sparse linear regression

Learning a sparse basis
Basic least squares regression

Suppose we have some noisy measurements $y$ that were generated by an unobserved state $x$ from a space spanned by the $k$ columns of $D$:

$$y \sim \mathcal{N}(Dx, \sigma^2 I) \sim \mathcal{N}\left(\sum_{j=1}^{k} x_j d_j, \sigma^2 I\right)$$

We can compute the most likely $\hat{x}$ by minimizing squared error:

$$\hat{x} = \arg \min_x \frac{1}{2} ||y - Dx||_2^2$$

Least squares by itself is prone to modeling outliers and noise.
Regularized least squares regression

To prevent overfitting, we introduce a regularization term:

$$\hat{x} = \arg\min_x \frac{1}{2} ||y - Dx||_2^2 + \lambda ||x||_\zeta$$

Different values of $\zeta$ induce different priors on $x$:

- $[\zeta = 0]$ — "L0-norm," unsolvable
- $[\zeta = 1]$ — lasso, Laplacian prior (Tibshirani, 1996)
- $[\zeta = 2]$ — ridge, Gaussian prior (Hoerl & Kennard, 1970)
- $[\zeta = 1] + [\zeta = 2]$ — elastic net (Zou & Hastie, 2005)
Why do we care about sparsity?

Suppose $k$ is large; sparsity limits “active” columns of $D$

- Helps make models easier for humans to understand
- Enables better compression

Sparsity seems to be a useful way of representing statistical properties of the natural world

So we’d like to keep $\zeta$ small to encourage sparse solutions
Sparse codes represent natural statistics efficiently.
Forward feature selection (Mallat & Zhang 1993)

Repeat for $t = 1 \ldots T$:

- Compute correlations $c = D^Tr_t$
- Find $i = \arg \max_j c_j$
- Add $c_i$ to the model
- Define $r_{t+1} \leftarrow r_t - c_id_i$

Features are selected greedily based on current residual

This is basically Matching Pursuit (Mallat & Zhang, 1993)
Least Angle Regression (Efron et al. 2004)

Repeat for $t = 1 \ldots T$:

- Compute correlations $c = D^T r_t$
- Identify “active” columns $A = \{j : |c_j| = \max_j \{|c_j|\}\}$
- Compute “equiangular” vector $u$ such that $u^T d_{A_1} = u^T d_{A_2} = \ldots$
- Compute largest $\gamma$ such that $r_t - \gamma u$ admits one additional active column
- Define $r_{t+1} \leftarrow r_t - \gamma u$

Developed by Efron, Hastie, Johnstone & Tibshirani (2004)

Same runtime complexity as OLS!
Regularization paths

Least Angle Regression (LAR) Path

Coefficients

|coef| / max|coef|
Learning a sparse basis

With Matching Pursuit, dictionary is updated based on residual
- Multiple codebook vectors cannot “share” a residual

Another way to learn is through coordinate descent
- First, compute encoding(s) given a fixed dictionary
- Then, optimize the dictionary given a fixed set of encodings
- Somewhat similar in spirit to EM
- Provable convergence, no learning rate parameter

Developed by Mairal, Bach, Ponce & Sapiro (2009)
Learning via coordinate descent (Mairal et al. 2009)

Repeat for $t = 1 \ldots T$:

- Draw a sample $x_t \sim p(x)$, and compute a sparse code:
  \[
  \alpha_t = \arg \min_\alpha \frac{1}{2} \|x_t - D_{t-1}\alpha\|_2^2 + \lambda \|\alpha\|_1
  \]

- Update running correlations:
  \[
  A_t \leftarrow A_{t-1} + \alpha_t \alpha_t^T \quad B_t \leftarrow B_{t-1} + x_t \alpha_t^T
  \]

- Then optimize $D$ given all previous $\alpha$:
  \[
  D_t = \arg \min_D \sum_{i=1}^t \frac{1}{2} \left( \text{Tr}(D^T D A_t) - \text{Tr}(D^T B_t) \right)
  \]