Case Study 3: fMRI Prediction

LASSO Regression

Machine Learning/Statistics for Big Data
CSE599C1/STAT592, University of Washington
Emily Fox
February 19th, 2013

©Emily Fox 2013

fMRI Prediction Task

- Goal: Predict word stimulus from fMRI image

Can we read your brain?

Classifier (logistic regression, kNN, …) → HAMMER or HOUSE
Typical Stimuli

Each stimulus repeated several times

Zero-Shot Classification

- **Goal:** Classify words not in the training set
- **Challenges:**
  - Cost of fMRI recordings is high
  - Can’t get recordings for every word in the vocabulary

Never showed “giraffe” in scanner

| Classifier (logistic regression, kNN, …) | HAMMER or HOUSE |
Semantic Features

Semantic feature values: “celery”
- 0.8368, eat
- 0.3461, taste
- 0.3153, fill
- 0.2430, see
- 0.1145, clean
- 0.0600, open
- 0.0586, smell
- 0.0286, touch
...  
- 0.0000, drive
- 0.0000, wear
- 0.0000, lift
- 0.0000, break
- 0.0000, ride

Semantic feature values: “airplane”
- 0.8673, ride
- 0.2891, see
- 0.2851, say
- 0.1689, near
- 0.1228, open
- 0.0883, hear
- 0.0771, run
- 0.0749, lift
...  
- 0.0049, smell
- 0.0010, wear
- 0.0000, taste
- 0.0000, rub
- 0.0000, manipulate

Zero-Shot Classification

- From training data, learn two mappings:
  - S: input image \(\rightarrow\) semantic features
  - L: semantic features \(\rightarrow\) word

- Can use “cheap” co-occurrence data to help learn L

![Features of word](image)

![Classifier](image)

HAMMER or HOUSE
fMRI Prediction Subtask

- **Goal:** Predict semantic features from fMRI image

```
20,000 voxels

Features of word

\[ x^i \rightarrow y^i \]

Let's consider each semantic feature separately

Learning $S$: \( \Theta \) images $\rightarrow$ semantic features

\[
\begin{bmatrix}
x_1^i \\
x_2^i \\
\vdots \\
x_d^i \\
\end{bmatrix}_{20,000} \rightarrow \begin{bmatrix}
y_1^i \\
y_2^i \\
\vdots \\
y_d^i \\
\end{bmatrix}_{d = \# of semantic features}
\]
```

Ridge Regression

- Ameliorating issues with overfitting:
  - New objective:
    
    \[
    \min_{\beta} \sum_{i=1}^{n} (y_i - (\beta_0 + \beta^T x_i))^2 + \lambda \beta^T \beta 
    \]
    
    \[
    \min_{\beta} \text{RSS}(\beta) \quad \text{s.t.} \quad \|\beta\|_2^2 \leq S
    \]
    
    Reformulate:
    
    \[
    F(\beta) = \frac{1}{2} \beta^T (X^T X) \beta - \beta^T (X^T y) + \text{const.} + \frac{1}{2} \lambda \beta^T \beta 
    \]
    
    \[
    F(\beta) = \frac{1}{2} \text{RSS}(\beta) + \text{const.} + \frac{1}{2} \lambda \beta^T (X^T X + \lambda I) \beta
    \]
    
    \[
    \hat{\beta}_{\text{ridge}} = (X^T X + \lambda I)^{-1} (X^T y)
    \]

- Penatilization of weights = "regularization"
Variable Selection

- Ridge regression: Penalizes large weights

- What if we want to perform “feature selection”? 
  - E.g., Which regions of the brain are important for word prediction?
  - Can’t simply choose predictors with largest coefficients in ridge solution
  - Computationally impossible to perform “all subsets” regression

- Try new penalty: Penalize non-zero weights
  - Penalty:
    \[ \| \beta \|_1 = \sum_j | \beta_j | \]
  - Leads to sparse solutions
  - Just like ridge regression, solution is indexed by a continuous param \( \lambda \)

LASSO Regression

- **LASSO**: least absolute shrinkage and selection operator

- New objective:
  \[
  \min_{\beta} \sum_{i=1}^N (y_i - (B_0 + \beta^T x_i))^2 + \lambda \| \beta \|_1
  \]

  \[= \min_{\beta} \text{RSS}(\beta) \quad \text{s.t.} \quad \| \beta \|_1 \leq B \]
Soft Thresholding

To see why LASSO results in sparse solutions, look at conditions that must hold at optimum.

- L1 penalty $\|\beta\|_1$ is not differentiable whenever $\beta_j = 0$
- Look at subgradient…
Subgradients of Convex Functions

- Gradients lower bound convex functions:
  \[ f(y) \geq f(x) + \nabla f(x) \cdot (y-x) \]

- Gradients are unique at \( x \) if function differentiable at \( x \)

- Subgradients: Generalize gradients to non-differentiable points:
  - Any plane that lower bounds function:
    \[ \nabla \in \partial f(x) \quad \text{Subgradient} \]
    \[ f(y) \geq f(x) + \nabla \cdot (y-x) \]

Soft Thresholding

- Gradient of RSS term:
  \[ \frac{\partial}{\partial \beta_j} \text{RSS} = \sum_{i=1}^{N} (y_i - \beta_j x_i)^2 \]

- Subgradient of full objective:
  \[ \frac{\partial}{\partial \beta_j} F(\beta) = (a_j \beta_j - c_j) + \lambda \beta_j \| \beta \|_1 \]
  \[ = \begin{cases} 
  a_j \beta_j - c_j - \lambda & \beta_j < 0 \\
  [-c_j - \lambda, -c_j + \lambda] & \beta_j = 0 \\
  a_j \beta_j - c_j + \lambda & \beta_j > 0
  \end{cases} \]
Soft Thresholding

- Set subgradient = 0:
  \[ \partial_{\beta_j} F(\beta) = \begin{cases} 
  a_j \beta_j - c_j - \lambda & \beta_j < 0 \\
  - c_j - \lambda, -c_j + \lambda & \beta_j = 0 \\
  a_j \beta_j - c_j + \lambda & \beta_j > 0 
  \end{cases} \]

  \[ \text{If } \beta_j < 0 \]
  \[ a_j \beta_j - c_j - \lambda = 0 \Rightarrow \beta_j = \frac{c_j + \lambda}{a_j} < 0 \Rightarrow c_j < -\lambda \quad \text{strong neg. corr., then } \beta_j < 0 \]

  \[ \text{If } \beta_j > 0 \]
  \[ a_j \beta_j - c_j + \lambda = 0 \Rightarrow \beta_j = \frac{c_j - \lambda}{a_j} > 0 \Rightarrow c_j > \lambda \quad \text{strong pos. corr., then } \beta_j > 0 \]

  \[ \text{If } \beta_j = 0 \quad -\lambda < c_j < \lambda \quad \text{otherwise, } \beta_j = 0 \]

- The value of \( c_j = 2 \sum_{i=1}^{N} x_i^i(y_i - \beta_{-j}^i x_{-j}) \) constrains \( \beta_j \)
Recall: Ridge Coefficient Path

Typical approach: select $\lambda$ using cross validation (CV)

Now: LASSO Coefficient Path

Again, each $\lambda$ indexes a diff. soln

There are only a few critical values of $\lambda$ where new cov. enter model

Sols are sparse for any given $\lambda$
LASSO Example

<table>
<thead>
<tr>
<th>Term</th>
<th>Least Squares</th>
<th>Ridge</th>
<th>Lasso</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intercept</td>
<td>2.465</td>
<td>2.452</td>
<td>2.468</td>
</tr>
<tr>
<td>lcavol</td>
<td>0.680</td>
<td>0.420</td>
<td>0.533</td>
</tr>
<tr>
<td>lweight</td>
<td>0.263</td>
<td>0.238</td>
<td>0.169</td>
</tr>
<tr>
<td>age</td>
<td>-0.141</td>
<td>-0.046</td>
<td></td>
</tr>
<tr>
<td>lbph</td>
<td>0.210</td>
<td>0.162</td>
<td>0.002</td>
</tr>
<tr>
<td>svi</td>
<td>0.305</td>
<td>0.227</td>
<td>0.094</td>
</tr>
<tr>
<td>lcp</td>
<td>-0.288</td>
<td>0.000</td>
<td></td>
</tr>
<tr>
<td>gleason</td>
<td>-0.021</td>
<td>0.040</td>
<td></td>
</tr>
<tr>
<td>pgg45</td>
<td>0.267</td>
<td>0.133</td>
<td></td>
</tr>
</tbody>
</table>

Debiasing

Original (D = 4096, number of nonzeros = 160)

L1 reconstruction (K0 = 1024, lambda = 0.0516, MSE = 0.0027)

Debiased (MSE = 3.26e-05)

Minimum norm solution (MSE = 0.0292)

From Kevin Murphy textbook

1. Use LASSO to find support
2. Run regression just w/ the selected cov. ⇒ removes bias for this model

LS est.

all coeff. shrunk ⇒ bias

Some people:

⇒ all coeff. shrunk ⇒ bias

⇒ 1. Use LASSO to find support

⇒ 2. Run regression just w/ the selected cov. ⇒ removes bias for this model

LS est.
Sparsistency

- Typical Statistical Consistency Analysis:
  - Holding model size \( p \) fixed, as number of samples \( N \) goes to infinity, estimated parameter goes to true parameter
    \[ \hat{\theta} \rightarrow \theta^* \]
  - Here we want to examine \( p \gg N \) domains
  - Let both model size \( p \) and sample size \( N \) go to infinity!
    - Hard case: \( N = k \log p \)
    \[ N \text{ grows slowly relative to } p \]

Rescale LASSO objective by \( N \):

\[ \min_{\beta} \frac{1}{N} \text{RSS}(\beta) + \lambda_N \sum_j |\beta_j| \]

Theorem (Wainwright 2008, Zhao and Yu 2006, ...):
- Under some constraints on the design matrix \( X \), if we solve the LASSO regression using
  \[ \lambda_N \geq \frac{2c_1}{\delta} \sqrt{\frac{2\delta^2 \log p}{N}} \]

Then for some \( c_1>0 \), the following holds with at least probability

\[ 1 - 4 \exp\left(-c_1 N \lambda_N^2\right) \rightarrow 1 \]

- The LASSO problem has a unique solution with support contained within the true support
  \[ S(\hat{\beta}) \subseteq S(\beta^*) \]
- If \[ \min_{j \in S(\beta^*)} |\beta_j^*| > c_2 \lambda_N \text{ for some } c_2>0 \], then
  \[ S(\hat{\beta}) = S(\beta^*) \]
fMRI Prediction Results

- Leave-one-out-cross-validation
  - Learn ridge coefficients using 59 fMRI images
  - Predict semantic features of heldout image
  - Compare whether very large set of possible other words

![Graph showing fMRI prediction results.]

**Figure 2:** The mean and median rank accuracies across nine participants for two different semantic feature sets. Both the original 60 fMRI words and a set of 946 nouns were considered.

Table 2: The top-five predicted words for a novel fMRI image taken for the word in bold (all fMRI images taken from participant P1). The number in the parentheses contains the rank of the correct word selected from 946 concrete nouns in English.

<table>
<thead>
<tr>
<th>Rank</th>
<th>Foot</th>
<th>Screwdriver</th>
<th>Train</th>
<th>Truck</th>
<th>Color</th>
<th>House</th>
<th>Pants</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>fox</td>
<td>pin</td>
<td>mass</td>
<td>jeep</td>
<td>artichoke</td>
<td>hotel</td>
<td>clothing</td>
</tr>
<tr>
<td>2</td>
<td>feet</td>
<td>nail</td>
<td>marsh</td>
<td>truck</td>
<td>grape</td>
<td>school</td>
<td>vest</td>
</tr>
<tr>
<td>3</td>
<td>nail</td>
<td>wrench</td>
<td>bus</td>
<td>sedan</td>
<td>cabbage</td>
<td>factory</td>
<td>dress</td>
</tr>
<tr>
<td>4</td>
<td>bone</td>
<td>dagger</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>gorilla</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Acknowledgements

- Some material in this lecture was based on slides provided by:
  - Tom Mitchell – fMRI
  - Rob Tibshirani – LASSO
  - Ryan Tibshirani – Fused LASSO