Regression: Penalties & Priors

Lyle Ungar
Supervised learning

- Given a set of observations with labels, $y$
  - Observations
    - Web pages with “Paris” labeled “Paris, France” or “Paris Hilton”
    - Proteins labeled “apoptosis” or “signaling”
    - Patients labeled with “alzheimers” or “frontotemporal dementia”

- Generate features, $x$, for each observation

- Learn a regression model to predict $y$
  - $y = f(x) = w_0 + w_1 x_1 + w_2 x_2 + w_3 x_3 + w_4 x_4 \ldots$
  - Most of the $w_j$ are zero.
Two interpretations of regression

- **Minimize (penalized) squared error**
- **Maximize likelihood**
  - Ordinary least squares (OLS): MLE
    - Minimizes
      - A) bias
      - B) variance
      - C) bias + variance
Two interpretations of regression

- Minimize (penalized) squared error
- Maximize likelihood
  - Ridge regression: MAP
    - Minimizes
      A) bias
      B) variance
      C) bias + variance
Minimize penalized Error $||y - w \cdot x||_2^2 + \lambda ||w||_2^2$

- Minimizing the first term, representing the training error, reduces
  A) bias
  B) variance
  C) neither
Ridge regression – Bias/Variance

Minimize penalized Error

\[ ||y - w \cdot x||_2^2 + \lambda ||w||_2^2 \]

- Minimizing the second term, which can be viewed as the amount that the test error is expected to be bigger than the training error reduces

A) bias
B) variance
C) neither
Different norms, different errors

\[ y \sim N(w^T x, \sigma^2) \sim \exp(-\|y - w^T x\|_2^2/2\sigma^2) \]

- \( \text{argmax}_w p(D|w) \)  
  here: argmax\(_w p(y|w, X)\)
- Err = \(\|y - w \cdot X\|_2^2\)  
  OLS = L\(_2\) regression

\[ y \sim \exp(-\|y - w^T x\|_p^p/2\sigma^2) \]

- \( \text{argmax}_w p(D|w) \)  
  here: argmax\(_w p(y|w, X)\)
- \(\|y - w \cdot X\|_p^p\)  
  often set p=1, giving L\(_1\) regression
Different norms, different penalties

◆ Minimize penalized Error  $||y - w \cdot x||_2^2 + \lambda f(w)$

- $||w||_2^2 = \sum_j |w_j|^2$  $L_2$
- $||w||_1 = \sum_j |w_j|^1$  $L_1$
- $||w||_0 = \sum_j |w_j|^0$  $L_0$
  
  Where $|w_j|^0 = 0$ if $w_j=0$ else $|w_j|^0=1$

◆ Note that all of these encourage $w_j$ to be smaller; i.e., they *shrink* $w$. 
Feature selection for regression

- Goal: minimize error on a test set
- Approximation: minimize a penalized training set error
  - $\text{Argmin}_w (\text{Err} + \lambda \|w\|_p^p)$ where $\text{Err} = \sum_i (y_i - \sum_j w_j x_{ij})^2 = \|y - w^T X\|^2$
  - Different norms
    - $p = 2$ – “ridge regression”
      - Makes all the $w$’s a little smaller
    - $p = 1$ – “LASSO” or “LARS” (least angle regression)
      - Still convex, but drives some $w$’s to zero
    - $p = 0$ – “stepwise regression”
      - Requires search

Note the confusion in the names of the optimization method with the objective function
Argmin_{w} (Err + \lambda \|w\|_p^p)

- How to pick \lambda?

Warning: for p = 0, the above formula is not really right (here and below); it is really |y - w \cdot x|^2 + \lambda |w|_0
Different regularization priors

Argmin_w ||y - w\cdot x||_2^2 + \lambda ||w||_p^p

- **L_2**  ||w||_2^2
  - Gaussian prior: p(w) \sim \exp(-||w||_2^2/\sigma^2)

- **L_1**  ||w||_1
  - Laplace prior: roughly p(w) \sim \exp(-||w||_1/\sigma^2)

- **L_0**  ||w||_0
  - Spike and slab
Different regularization penalties

a) $L_2$

$$\text{Argmin}_w \ | |y - w \cdot x| |_2^2 + \lambda | |w| |_2^2$$

b) $L_1$

$$\text{Argmin}_w \ | |y - w \cdot x| |_2^2 + \lambda | |w| |_1$$

c) $L_0$

$$\text{Argmin}_w \ | |y - w \cdot x| |_2^2 + \lambda | |w| |_0$$

Which norm most heavily shrinks large weights?
Different regularization penalties

a) $L_2$
$$\text{Argmin}_w \| y - w \cdot x \|_2^2 + \lambda \| w \|_2^2$$

b) $L_1$
$$\text{Argmin}_w \| y - w \cdot x \|_2^2 + \lambda \| w \|_1$$

c) $L_0$
$$\text{Argmin}_w \| y - w \cdot x \|_2^2 + \lambda \| w \|_0$$

Which norm most strongly encourages weights to be set to zero?
Different regularization penalties

a) $L_2$
$$\text{Argmin}_w \ |y - w \cdot x|_2^2 + \lambda |w|_2^2$$

b) $L_1$
$$\text{Argmin}_w \ |y - w \cdot x|_2^2 + \lambda |w|_1$$

c) $L_0$
$$\text{Argmin}_w \ |y - w \cdot x|_2^2 + \lambda |w|_0$$

Which norm is scale invariant?
Different regularization penalties

Argmin$_w$ $||y - w \cdot x||^2_2 + \lambda ||w||^p_p$

- $L_2$ - Ridge regression

- $L_1$ - LASSO or LARS

- $L_0$ - “stepwise regression”

Which lead to convex optimization problems?

Warning: for $p = 0$, the above formula is not really right (here and below); it is really $|y - w \cdot x|^2_2 + \lambda |w|_0$
Solving with regularization penalties

\[ \text{Argmin}_w \ |y - w \cdot x|^2_2 + \lambda \ |w|^p \]

- **L_2**
  - \((X'X + \lambda I)^{-1} X' y\)

- **L_1**
  - Convex optimization

- **L_0**
  - Search (stepwise or streamwise)

L_1 and L_0 can handle exponentially more features than observations; L_2 cannot
L₀, L₁ and L₂ Penalties

- If the x’s have been standardized (mean zero, variance 1) then we can visualize the shrinkage:

L₂ = Ridge
sum of squares

L₁ = Lasso
sum of abs value

L₀ = “stepwise regression”
Number of features

Shrunk w
How to pick $\lambda$?

- Try a bunch of different values and see which one minimizes the (non-penalized) error on a test set
  - Cross validation
- Or use information theory for $L_0$. 
Streamwise regression

- **Initialize:**
  - `model = {}`,
  - `Err_0 = \sum_i (y_i - 0)^2 + 0`

- **For each feature `x_j`:**
  - *Try* adding the feature `x_j` to the model
  - *If*
    - `Err = \sum_i (y_i - \sum_{j \in \text{model}} w_j x_{ij})^2 + \lambda ||\text{model}||_0 < Err_{j-1}`
    - Accept new model and set `Err_j = Err`
  - *Else*
    - Keep old model and set `Err_j = Err_{j-1}`

where `||\text{model}||_0 = \# \text{ of features in the model}`
Stepwise regression

- **Initialize:**
  - `model = {}`
  - `Err_{old} = \sum_i (y_i - \theta)^2 + 0`

- **Repeat (up to $p$ times)**
  - **Try** adding each feature $x_k$ to the model
    - Pick the feature that gives the lowest error
    - `Err = min_k \sum_i (y_i - \sum_{j \in model_k} w_j x_{ij})^2 + \lambda |model_k|`
  - **If** `Err < Err_{old}`
    - Add the feature to the model
    - `Err_{old} = Err`
  - **Else** Halt
Stagewise regression

- Like stepwise, but at each iteration, keep all of the coefficients $w_j$ from the old model, and just regress the residual $r_i = y_i - \sum_{j \text{ in model}} w_j x_{ij}$ on the new candidate feature $k$.

Later: boosting
What you should know

- **$L_2$, $L_1$, $L_0$ penalties**
  - Names. How they are solved
- **Training vs. Testing**
  - Penalized error approximates test error
- **Streamwise, stepwise, stagewise regression**

$L_2 + L_1$ penalty = “Elastic net”

$$\text{argmin}_w \ | | y - wx \ | |^2_2 + \lambda_2 | |w| |^2_2 + \lambda_1 | |w| |^1$$