Shanni Lam
Math189R SU20
Homework 7
Thursday, July 9, 2020

Feel free to work with other students, but make sure you write up the homework and code on your own (no copying homework *or* code; no pair programming). Feel free to ask students or instructors for help debugging code or whatever else, though.

The starter files for problem 2 can be found under the Resource tab on course website. The plot for problem 2 generated by the sample solution has been included in the starter files for reference. Please print out all the graphs generated by your own code and submit them together with the written part, and make sure you upload the code to your Github repository.

1 (Murphy 11.3 - EM for Mixtures of Bernoullis) Show that the M step for ML estimation of a mixture of Bernoullis is given by

$$\mu_{kj} = \frac{\sum_i r_{ik} x_{ij}}{\sum_i r_{ik}}.$$

Show that the M step for MAP estimation of a mixture of Bernoullis with a $\beta(a, b)$ prior is given by

$$\mu_{kj} = \frac{\left(\sum_i r_{ik} x_{ij}\right) + a - 1}{\left(\sum_i r_{ik}\right) + a + b - 2}.$$

a) The complete data log likelihood is given by $\ell(\boldsymbol{\mu}) = \sum_i \sum_k r_{ik} log \mathbb{P}(\mathbf{x}_i|\boldsymbol{\theta_k})$
$= \sum_i \sum_k r_{ik} \sum_j x_{ik} log \boldsymbol{\mu_{kj}} + (1 - \mathbf{x}_{ij}) log(1 - \boldsymbol{\mu_{kj}})$
where i is the datapoint index, k is the component, and j is the dimension index of the D dimensional bit vectors.
Differentiating with respect to $\boldsymbol{\mu_{kj}}$ and setting it equal to 0 to set up the optimality condition, we have

$\frac{\partial}{\partial \boldsymbol{\mu_{kj}}} = \sum_i r_{ik}(\frac{\mathbf{x}_{ij}}{\mu_{kj}} - \frac{1-\mathbf{x}_{ik}}{1-\boldsymbol{\mu_{kj}}}) = \sum_i r_{ik}(\frac{\mathbf{x}_{ij} - \mathbf{x}_{ij}\boldsymbol{\mu_{kj}} + \boldsymbol{\mu_{kj}}\mathbf{x}_{jk} - \boldsymbol{\mu_{kj}}}{\boldsymbol{\mu_{kj}}(1-\boldsymbol{\mu_{kj}})}) = \sum_i r_{ik}(\frac{\mathbf{x}_{ij} - \boldsymbol{\mu_{kj}}}{\boldsymbol{\mu_{kj}}(1-\boldsymbol{\mu_{kj}})})$
$= \frac{1}{\boldsymbol{\mu_{kj}}(1-\boldsymbol{\mu_{kj}})} \sum_i r_{ik}(\mathbf{x}_{ij} - \boldsymbol{\mu_{kj}}) = 0$. Note that we took out the denominator as a constant in front because it has no terms relating to i.

This gives the optimality condition $\sum_i r_{ik}(\mathbf{x}_{ij} - \boldsymbol{\mu_{kj}}) = 0 \implies \sum_i r_{ik}\mathbf{x}_{ij} - \sum_i r_{ik}\boldsymbol{\mu_{kj}} = 0 \implies$ $\sum_i r_{ik}\mathbf{x}_{ij} = \boldsymbol{\mu_{kj}} \sum_{\boldsymbol{i}} \boldsymbol{r_{ik}}$. Finally, isolate for $\boldsymbol{\mu_{kj}}$, so $\boldsymbol{\mu_{kj}} = \frac{\sum_i r_{ik}\mathbf{x}_{ij}}{\sum_i r_{ik}}$, as desired.

b) The complete data log likelihood plus the log prior (ignoring the π terms as we are maximizing without regard to them) is given by

$\ell(\boldsymbol{\mu}) = \sum_i \sum_k r_{ik} log \mathbb{P}(\mathbf{x}_i|\boldsymbol{\mu_k}) + log \mathbb{P}(\boldsymbol{\mu_k})$
$= \sum_i \sum_k r_{ik}(\sum_j \mathbf{x}_{ij} log \boldsymbol{\mu_{kj}} + (1 - \mathbf{x}_{ij}) log(1 - \boldsymbol{\mu_{kj}})) + (a-1) log(\boldsymbol{\mu_{kj}}) + (b-1) log(1 - \boldsymbol{\mu_{kj}})$.

Taking derivatives and setting them equal to 0, we have $\frac{\partial \ell}{\partial \boldsymbol{\mu}} = \sum_i (\frac{r_{ik}\mathbf{x}_{ij}+a-1}{\boldsymbol{\mu_{kj}}} - \frac{r_{ik}(1-\mathbf{x}_{ij})+b-1}{1-\boldsymbol{\mu_{kj}}})$
$= \frac{1}{\boldsymbol{\mu_{kj}}(1-\boldsymbol{\mu_{kj}})} \sum_i r_{ik}\mathbf{x}_{ij} - r_{ik}\boldsymbol{\mu_{kj}} + a - 1 - \boldsymbol{\mu_{kj}}a + \boldsymbol{\mu_{kj}} - \boldsymbol{\mu_{kj}}b + \boldsymbol{\mu_{kj}}$
$= \frac{1}{\boldsymbol{\mu_{kj}}(1-\boldsymbol{\mu_{kj}})} [\sum_i r_{ik}\mathbf{x}_{ij} - (\sum_i r_{ik} + a + b - 2)\boldsymbol{\mu_{kj}} + a - 1] = 0.$

The constant in front can be eliminated since the equation is equated to 0. Then, $\sum_i r_{ik}\mathbf{x}_{ij} + a - 1 = (\sum_i r_{ik} + a + b - 2)\boldsymbol{\mu_{kj}}$, which gives the desired result. Isolating, we get the optimality condition that $\boldsymbol{\mu_{kj}} = \frac{\left(\sum_i r_{ik}\mathbf{x}_{ij}\right)+a-1}{(\sum_i r_{ik})+a+b-2}$, as desired.
Note that if $a = b = 1$, we arrive at the original maximum likelihood estimate from part a, $\boldsymbol{\mu_{kj}} = \frac{\sum_i r_{ik}\mathbf{x}_{ij}}{\sum_i r_{ik}}$, which makes sense because $\beta(1,1)$ is a uniform distribution over $[0,1]$, so these parameters cause the situation that there is no prior.

■

2 (Lasso Feature Selection) In this problem, we will use the online news popularity dataset we used in hw2pr3. In the starter code, we have already parsed the data for you. However, you might need internet connection to access the data and therefore successfully run the starter code.

First, ignoring undifferentiability at $x = 0$, take $\frac{\partial |x|}{\partial x} = \text{sign}(x)$. Using this, show that $\nabla \|\mathbf{x}\|_1 = \text{sign}(\mathbf{x})$ where sign is applied elementwise. Derive the gradient of the ℓ_1 regularized linear regression objective

$$\text{minimize: } \|A\mathbf{x} - \mathbf{b}\|_2^2 + \lambda \|\mathbf{x}\|_1$$

Then, implement a gradient descent based solution of the above optimization problem for this data. Produce the convergence plot (objective vs. iterations) for a non-trivial value of λ. In the same figure (and different axes) produce a 'regularization path' plot. Detailed more in section 13.3.4 of Murphy, a regularization path is a plot of the optimal weight on the y axis at a given regularization strength λ on the x axis. Armed with this plot, provide an ordered list of the top five features in predicting the log-shares of a news article from this dataset (with justification).

Wrap the gradient descent step with a threshold function

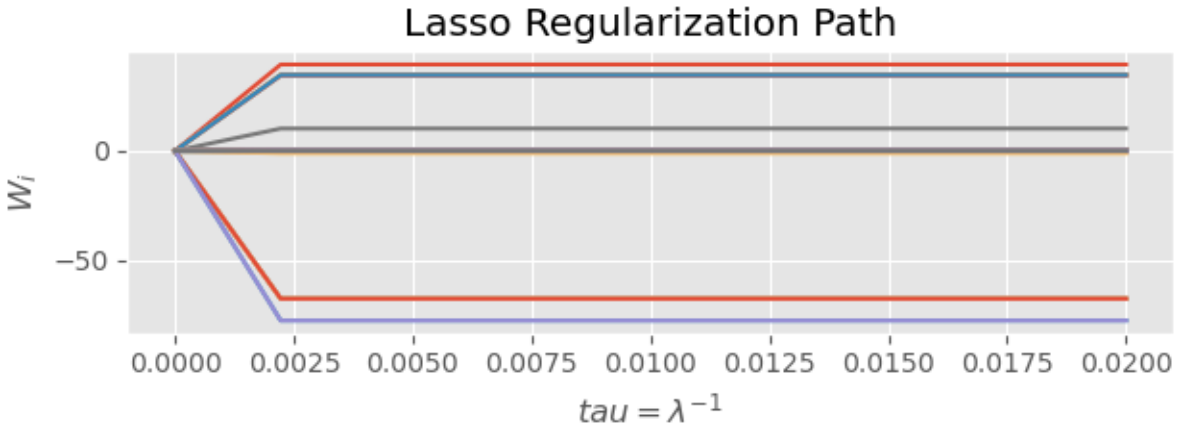
$$proj_\gamma(\mathbf{x})_i = \begin{cases} \mathbf{x}_i - \gamma, & \mathbf{x}_i > \gamma \\ 0, & |\mathbf{x}_i| \leq \gamma \\ \mathbf{x}_i + \gamma, & x_i < -\gamma \end{cases}$$

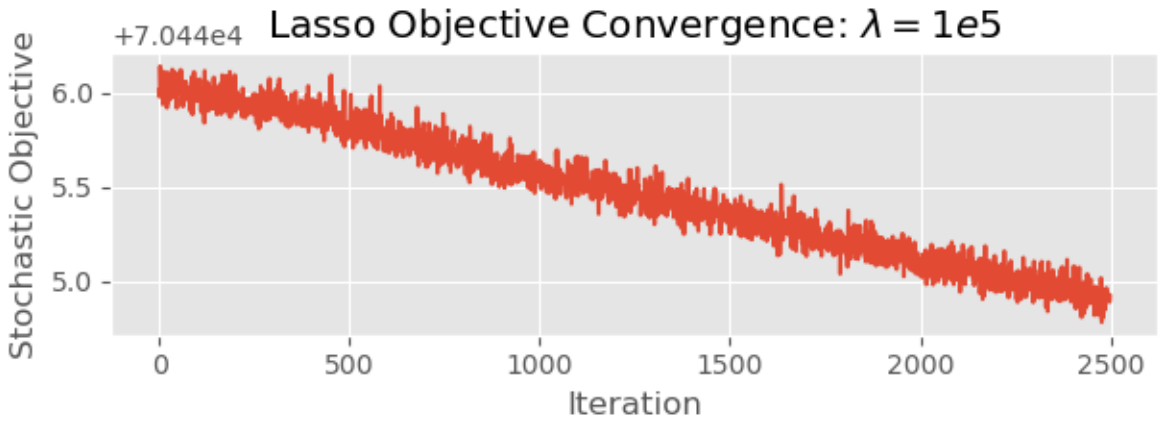
so that each iteration $\mathbf{x}_{i+1} = prox_\gamma(\mathbf{x}_i - \gamma \nabla f(\mathbf{x}_i))$, where γ is the learning rate.

$\frac{\partial ||\mathbf{x}||_1}{\partial \mathbf{x}_i} = \frac{\partial \sum |\mathbf{x}_i|}{\partial \mathbf{x}_i} = sign(\mathbf{x}_i)$. It follows that $\nabla ||\mathbf{x}||_1 = sign(\mathbf{x}_i)$.
Then, $\nabla(||A\mathbf{x} - \mathbf{b}||_2^2 + \lambda ||\mathbf{x}||_1) = \nabla(\mathbf{x}^T A^T A\mathbf{x} + \mathbf{b}^T\mathbf{b} + \lambda ||\mathbf{x}||_1 = 2A^T A\mathbf{x} - 2\mathbf{b}^T A + \lambda sign(\mathbf{x}))$ since we are given the partial derivative of any absolute value of a variable is its sign.

See the plots below: the lasso regularization path and lasso objective plot of convergence vs. iteration for a nontrivial λ correspondingly. Note that since we instantiated our weights with the least squares estimate, we can see our lasso objective not moving significantly, although if we look at sparsity over time, it increases.

The top 5 important features in predicting the log-shares of a news article from this dataset are ['timedelta', 'weekday_is_wednesday', 'weekday_is_thursday', 'weekday_is_friday', 'weekday_is_saturday'].


Lasso Regularization Path
W_i
0
−50
0.0000
0.0025
0.0050
0.0075
0.0100
0.0125
0.0150
0.0175
0.0200
$tau = \lambda^{-1}$


Lasso Objective Convergence: $\lambda = 1e5$
+7.044e4
Stochastic Objective
6.0
5.5
5.0
0
500
1000
1500
2000
2500
Iteration

■