Logistic Regression Revisited

CS/DSA 5970: Machine Learning Practice
Logistic Regression Review

• Add a sigmoid non-linearity to the end of our linear model
• Sigmoid: output range from 0 to 1
  – Can interpret this as a probability
  – For classification, this can be the probability of being in the positive class
• Prior classification conversation:
  – Used the MSE cost function (mean squared differences between ground truth label and the probability)
  – Problematic because the derivative can become very flat
• MSE cost function
• Derivative of MSE wrt a particular weight
  – Show that when output is close to 0 or 1, this derivative becomes zero
  – This is particularly a problem when we are incorrect in our answer: we want to move the coefficients associated with this decision, but we can’t make much progress
  – This implies that we must wait a long time to find a solution
• Alternative: pick a new cost function that doesn’t have this problem
Log-Likelihood Cost Function

CS/DSA 5970: Machine Learning Practice
Parameter Selection for Likelihood Functions

From statistics:

• Given:
  – A set of samples drawn independently from a distribution
  – A form of distribution from which the samples are drawn (e.g., a Normal distribution)

• Find the “best” parameters that explain the set of samples
  – Typical approach: use a likelihood function
• Likelihood function for a single sample (Normal dist)
• Likelihood function for a set of independent samples
• Take the log
• Mention that we can then compute mu and sigma
Log-Likelihood For Classifiers

CS/DSA 5970: Machine Learning Practice
We can use a similar approach to talking about the “goodness” of a classifier.

The new twist: we now have two classes

- The classifier should assign a high probability to the positive examples
- And low probabilities to the negative examples
Example: Logistic Regression

CS/DSA 5970: Machine Learning Practice
Example: Logistic Regression

• SGDClassifier with ‘log’ loss:
  – Logistic regression with log likelihood loss
    (we already played with this class)

• LogisticRegression class:
  – Also uses log likelihood loss
  – Different solver than SGDClassifier
Example: Logistic Regression

Both offer regularization

- L1, L2, Elastic (must pick solver appropriately)

- SGDClassifier with ‘log’ loss:
  - Regularization parameter: alpha
  - Increase value: more regularization

- LogisticRegression class:
  - Regularization parameter: C
  - Increase value: less regularization
Code demo
Multiclass Case: Softmax

CS/DSA 5970: Machine Learning Practice
Softmax

Want to be able to handle $K > 2$ classes

- So far, the approach has been to create a set of binary classifiers and have them vote
- One vs all: need $O(K)$ classifiers
- One vs one: need $O(K^2)$ classifiers
Softmax

Approach:
• Learned function: output a score for each of K classes
• Use the softmax function to translate the scores into probabilities
• Output:
  – Can look at the probabilities directly
  – Or can pick the class with the highest probability as the predicted class
Example: Softmax
Example: Softmax

LogisticRegression class:

- Desired output can be an integer, with values encoding different classes
- Internally, the class performs one-hot encoding
Live demo