University of California, Berkeley

Lecture 10 CS189: Regression, Logistic Regression, Gradient Descent

Oscar Ortega

July 16, 2021

1 ISL: Chapter 4 - Classification

Recall overview of Classification: note that just in the regression setting, in the classification setting we have a set of training observations $(x_1, y_i), ..., (x_n, y_n)$ that we can use to build a classifier.

Why not just perform Linear regression to build such classifiers?

2 Example of when this is bad:

Situation: trying to predict the medical condition of a patient in the emergency room on the basis of her symptoms:

$$\text{Let } Y = \{1 \text{ if stroke }, 2 \text{ if drug overdose }, 3 \text{ if epileptic seizure }\}$$

Note here that because this set of symptoms is not necessarily ordered, performing Linear Regression on a set of people exhibiting symptoms would imply ordering of the response variable Y.

Also note that in the case where we are trying to classify between two classes:

$$Y = \{0 \text{ if stroke:} 1 \text{ if drug overdose}\}$$

And if we created a linear classifier for this data, it would be very sensitive to outliers in the data, shifting the boundary of stroke versus no stroke.

3 Logistic Regression

The above approach does suggest a way to perform regression on this classified data:

The Logistic Model: note how in the prior example, we basically were computing the following probability for X:

$$p(X) = b_0 + b_1 X$$

we transform our function into the following:

$$p(x) = \frac{e^{b_0 b_1 X}}{1 + e^{b_0 + b_1 X}}$$

we perform maximum likelihood estimation like before to produce the best fit model.

Define the **odds**:

$$\text{odds}(x) = \frac{p(x)}{1 - p(x)} = e^{b_0 + b_1(x)}$$

$$\leftarrow = b_0 + b_1 X$$

note that after we take logs:

$$\ln\left(\frac{p(x)}{1 - p(X)}\right) = b_0 + b_1 X$$

known as **logit or log-odds**

Because we are no longer dealing with linear function of X. we can no longer state that a one unit change in b_1 will correspond to a unit of change in X. however, if b_1 is positive then increasing x will be associated with decreasing p(X).

note that the sigmoid function is more S shaped

4 4.3.2: Estimating the Regression Coefficients

We determine the optimal b_0 and b_1 based on the available training data by using the general method of **maximum likelihood estimation**

Definition:

Likelihood function:

$$\mathscr{L}(B_o, B_1) = \prod_{i:y_i=1} p(X_i) \prod_{i':y'_i=0} (1 - p(x'_i))$$

we maximize the function with respect to these parameters.

Least squares is a form of maximum likelihood estimation.

5 MAKING PREDICTIONS

recall the general form of the formulation for $\hat{p}(X)$

$$\hat{p}(X) = \frac{e^{b_0+b_1X}}{1+e^{b_0+b_1X}}$$

6 MULTIPLE LOGISTIC REGRESSION

If we consider the problem of predicting a binary response using multiple features ('predictors'), we can extend the definition of the logit to $x \in \mathbf{R}^d$ where d is the number of predictors.

$$\ln(\frac{p(x)}{1-p(x)}) = a + b'^T x : x, b' \in \mathbf{R}^d : a \in \mathbf{R}$$

Confounding Results:
Because it is the case the logit is a linear function of x, it can be the case that the linear combination of optimal features produced by the maximum likelihood estimate might yield contradictory results.

7 LECTURE: REGRESSION AKA FITTING CURVES TO DATA

Up until this point we have been primarily concerned with classification: Regression: given point x, predict a numerical value, what is the probability that a point is correct?

-Choose form of regression fn: $h(x;p)$ with paramters p, (h = hypothesis. -Like decision fn in classification -Chose a cost fn (objective fn) -usually based on a loss fn; e.g risk = expected loss Some regression fns:

1. Linear: $h(x;w,\alpha) = w^T x + \alpha$
2. polynomial
3. logistic: $h(x;w,\alpha) = \sigma(w^T x + \alpha)$ recall $\sigma(\alpha) = \frac{1}{1+e^{-\alpha}}$

Some loss fns: let z be prediction h(x); y be true value.

1. $L(z,y) = (z-y)^2$ squared error
2. $L(z,y) = |z-y|$ absolute error
3. $L(z,y) = -yln(z) - (1-y)ln(1-z)$ logistic loss, aka cross-entropy loss $y \in [0,1], z \in (0,1)$

Some cost functions:

1. $J(h) = \frac{1}{n}\sum_i L(h(x_i), y_i)$ mean loss
2. $J(h) = \max_i L(h(x_i), y_i)$ maximum loss

3. $J(h) = \sum_i w_i L(h(x_i), y_i)$ weighted sum
4. $J(h) = \frac{1}{n}\sum_i L(h(x_i), y_i) + \lambda \|w\|_2$ l_2 loss
5. $J(h) = \frac{1}{n}\sum_i L(h(x_i), y_i) + \lambda \|w\|_1$ l_1 loss

Some famous regressions methods:

Least - Squares linear regr: quad. cost, can minimize with calc

Weighted least square linear:quad. cost, can also min with calc

Ridge Regression: quad cost, can also min with calc

Logistic Regression: convex cost, can minimize with grad. desc

Lasso: quad program

Least absolute deviations: linear program

Chebyshev Criterion: linear program

8 Least-Squares Linear Regression

Find w, a that minimized $\sum_i (x_i^T w + a - y_i)^2$

Convention: $X \in \mathbf{R}^{n,d}$ **design matrix** of sample pts.
$y \in \mathbf{R}^n$, labels. Usually n > d. Recall fictitious trick: rewrite $h(x) = x^T w + \alpha$

$$\begin{bmatrix} x_1 & x_2 & 1 \end{bmatrix}^T \begin{bmatrix} w_1 & w_2 & \alpha \end{bmatrix}$$

corresponds to finding $min_w \|Xw - y\|^2$ this is known as the residual sum of squares. Recall the following:
$X^T X w = X^T y \rightarrow$ the normal equations.
If $X^T X$ is singular, this means problem is underconstrained. We use a linear solver to find $w^* = (X^T X)^{-1} X^T y$ Usually use Cholesky factorization. recall the x terms define the pseudo inverse of X.

$$X^\dagger X = (X^T X)^{-1} X^T X = I$$

Also observe the predicted values of y are $\hat{y}_i = w^T x_i \rightarrow \hat{y} = Xw = XX^\dagger y = Hy$ where H is called the **Hat Matrix** because it puts the hat on y.
We can interpret the optimization as both a calculus optimization and a projection onto the range of the matrix X. Note that if we rewrite the normal equations. Error from y is minimized when your perpendicular to the column space of x.

$$X^T (Xw - y) = 0$$

Note how these are just the normal equations.
Advantages under this formulation:

- easy to compute just solve a linear system.

- unique stable solution.

Disadvantages under this formulation:

- Very sensitive to outliers because errors are squared.
- fails if $X^T X$ is singular.

9 Logistic Regression

Fits probabilities. $y \in (0,1)$ Usually used for classification. The input $y_i's$ can be probabilities, but most apps they're all 0 or 1.
QDA, LDA: generative models logistic regression: discriminative model with X, w using the fictition dimension we minimize the following:

$$J = -\sum_i (y_i \ln \sigma(X_i^T w) + (1-y_i)\ln(1-\sigma(X_i^T w)$$

$$\sigma'(\gamma) = \sigma(\gamma)(1-\sigma(\gamma))$$

$$= -\sum_i (\frac{y_i}{\sigma_i}\sigma'_i - \frac{1-y_i}{1-\sigma_i}\sigma'_i)$$

where $\sigma_i = \sigma(X_i^T w)$

$$-\sum_i (\frac{y_i}{\sigma_i} - \frac{1-y_i}{1-\sigma_i}\sigma_i(1-\sigma_i)X_i$$

$$-\sum_i (y_i - \sigma_i)X_i$$

$-X^T(y-\sigma(Xw))$ where $\sigma(Xw)_i = \sigma_i$ The stochastic gradient descent rule becomes the following:

$$w \leftarrow w + \epsilon(y_i - \sigma(X_i^T w))X_i$$

this will work best if we shuffle pts in random order, process one by one: For very large n, sometimes converges before we visit all points: Starting from w = 0 works well in practice.