Linear regression, Logistic regression, and Generalized Linear Models

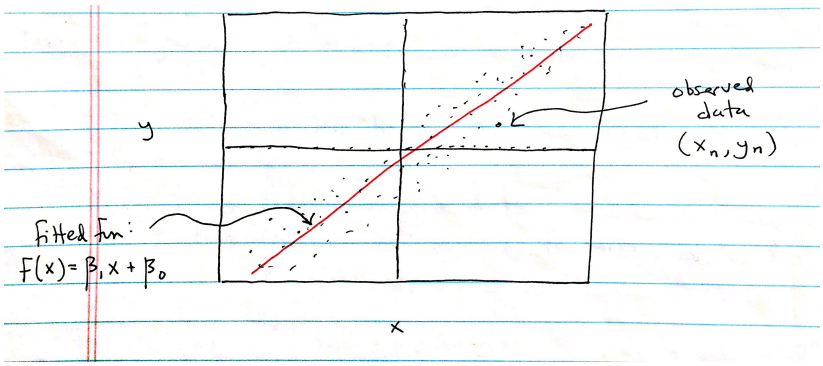
David M. Blei
Columbia University

December 2, 2015

1 Linear Regression

One of the most important methods in statistics and machine learning is linear regression. Linear regression helps solve the problem of predicting a real-valued variable y, called the *response*, from a vector of inputs x, called the *covariates*.

The goal is to predict y from x with a linear function. Here is a picture.



Here are some examples.

- Given my mother's height, what is my shoe size?
- Given the stock price today, what will it be tomorrow?
- Given today's precipitation, what will it be in a week?
- Others? Where have you seen linear regression?

In linear regression there are p covariates; we fit a linear function to predict the response,

$$f(x) = \beta_0 + \sum_{i=1}^{p} \beta_i x_i. \tag{1}$$

The vector β contains the p *coefficients*; β_0 is the *intercept*. (Note the intercept can be seen as a coefficient for a special covariate that is always equal to one.)

This set-up is less limiting than you might imagine. The covariates can be flexible:

- Any feature of the data
- Transformations of the original features, $x_2 = \log x_1$ or $x_2 = \sqrt{x_1}$.
- A basis expansion, e.g., $x_2 = x_1^2$ and $x_3 = x_1^3$. Here linear regression fits a polynomial, rather than a line.
- Indicator functions of qualitative covariates, e.g., 1[The subject has brown hair].
- *Interactions* between covariates, e.g., $x_3 = x_1 x_2$.

Its simplicity and flexibility makes linear regression one of the most important and widely used statistical prediction methods. There are papers, books, and sequences of courses devoted to linear regression.

1.1 Fitting a regression

We fit a linear regression to covariate/response data. Each data point is a pair (x, y), where x are the covariates and y is the response. Given a data set $\mathcal{D} = \{(x_n, y_n)\}_{n=1}^N$, our goal is to find the coefficients β that best predict y_{new} from x_{new}. For simplicity, assume that x_n is a scalar and the intercept β_0 is zero. (In general we can assume $\beta_0 = 0$ by centering the response variables before analyzing them.) There is only one coefficient β.

A reasonable approach is to consider the squared Euclidean distance between each *fitted response* $f(x_n) = \beta x_n$ and the true response y_n, $\epsilon_n = (y_n - \beta x_n)^2$. Choose β that

minimizes the sum of the distances over the data,

$$\hat{\beta} = \arg\min \frac{1}{2} \sum_{n=1}^{N} (y_n - \beta x_n)^2 \tag{2}$$

This has a closed-form solution, both in the one-covariate case and the more general setting with p covariates. (We get back to this later.)

[In the picture, point out the distances between predicted values and true responses.]

Given a fitted coefficient $\hat{\beta}$, we plug it into the regression equation to predict the new response,

$$\hat{y}_{\text{new}} = \hat{\beta} x_{\text{new}}. \tag{3}$$

1.2 Linear regression as a probabilistic model

Linear regression can be interpreted as a probabilistic model,

$$y_n \mid x_n \sim N(\beta^\top x_n, \sigma^2). \tag{4}$$

For each response this is like putting a Gaussian "bump" around a mean, which is a linear function of the covariates. This is a *conditional model*; the inputs are not modeled with a distribution.

[Draw the graphical model, training and testing]

The parameters of a linear regression are its coefficients, and we fit them with maximum likelihood.[1] We observe a training set of covariate/response pairs $\mathcal{D} = \{(x_n, y_n)\}$; we want to find the parameter β that maximizes the conditional log likelihood, $\sum \log p(y_n \mid x_n, \beta)$.

Finding this MLE is equivalent to minimizing the residual sum of squares, described above. The conditional log likelihood is

$$\mathcal{L}(\beta; \mathbf{x}, \mathbf{y}) = \sum_{n=1}^{N} -\frac{1}{2} \log 2\pi\sigma^2 - \frac{1}{2}(y_n - \beta^\top x_n)^2 / 2\sigma^2. \tag{5}$$

[1] We will add priors later on.

Optimizing with respect to β reveals

$$\hat{\beta} = \arg\max -\frac{1}{2\sigma^2} \sum_{n=1}^{N} (y_n - \beta^\top x_n)^2 \tag{6}$$

This is equivalent to minimizing the residual sum of squares.

Return to the single covariate case. The derivative is

$$\frac{d\mathcal{L}}{d\beta} = \frac{1}{2\sigma^2} \sum_{n=1}^{N} (y_n - \beta x_n) x_n \tag{7}$$

In this case the optimal can be found analytically,

$$\hat{\beta} = \frac{\sum_{n=1}^{N} y_n x_n}{\sum_{n=1}^{N} x_n^2}. \tag{8}$$

This is the empirical covariance between the covariate and response divided by the empirical variance of the covariate. (There is also an analytic solution when we have multiple covariates, called the *normal equation*. We won't discuss it here.)

Foreshadow: Modern regression problems are high dimensional, which means that the number of covariates p is large. In practice statisticians *regularize* their models, veering away from the MLE solution to one where the coefficients have smaller magnitude. (This is where priors come in.) In the next lecture, we will discuss why regularization is good.

Still in the probabilistic framework, let's turn to prediction. The conditional expectation is $\mathbb{E}[y_{\text{new}} \mid x_{\text{new}}]$. This is simply the mean of the corresponding Gaussian,

$$\mathbb{E}[y_{\text{new}} \mid x_{\text{new}}] = \hat{\beta}^\top x_{\text{new}}. \tag{9}$$

Notice the variance σ^2 does not play a role in prediction.

With this perspective on prediction, we can rewrite the derivative of the conditional log likelihood,

$$\frac{d\mathcal{L}}{d\beta} = \sum_{n=1}^{N} (y_n - \mathbb{E}[y \mid x_n, \beta]) x_n. \tag{10}$$

This form will come up again later.

In the more general case, there are p covariates. The derivative is

$$\frac{d\mathcal{L}}{d\beta_i} = \sum_{n=1}^{N}(y_n - \mathbb{E}[Y \mid x_n, \beta])x_{ni}. \tag{11}$$

This is intuitive. When we are fitting the data well, the derivative is close to zero. If both the error (also called the *signed residual*) and covariate are large then we will move the corresponding coefficient.

2 Logistic regression

We have seen that linear regression corresponds to this graphical model.

[Draw the graphical model, training and testing]

We can use the same machinery to do classification. Consider binary classification, where each data point is in one of two classes $y_n \in \{0, 1\}$. If we used linear regression to model this data, then

$$y_n \mid x_n \sim N(\beta^\top x_n, \sigma^2). \tag{12}$$

This is not appropriate for binary classification.

Q: Why not? The reason is that y_n needs to be zero or one, i.e., a Bernoulli random variable. (In some set-ups, it makes more sense for the classes to be $\{-1, 1\}$. Not here.)

We model $p(y = 1 \mid x)$ as a Bernoulli whose parameter is a function of x,

$$p(y = 1 \mid x) = \mu(x)^y(1 - \mu(x))^{1-y}. \tag{13}$$

What form should $\mu(x)$ take?

Let's go back to our line of thinking around linear regression. We can think of linear regression as a Gaussian whose mean is a function of x, specifically, $\mu(x) = \beta^\top x_n$. Is this appropriate here? No, because we need $\mu(x) \in (0, 1)$.

Rather, we use the *logistic function*,

$$\mu(x) = \frac{1}{1 + e^{-\eta(x)}}. \tag{14}$$

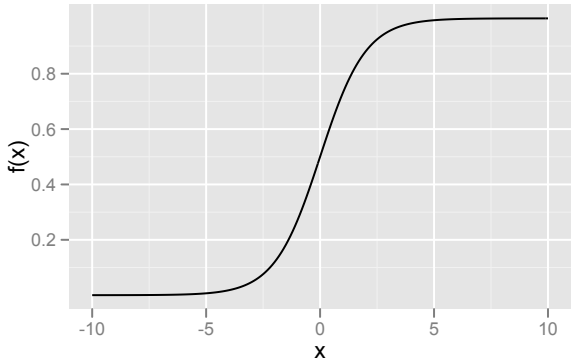
This function maps the reals to a value in $(0, 1)$.

- Q: What happens when $\eta(x) = -\infty$?
  A: $\mu(x) = 0$

- Q: What happens when $\eta(x) = +\infty$?
  A: $\mu(x) = 1$

- Q: What happens when $\eta(x) = 0$?
  A: $\mu(x) = 1/2$.

Here is a plot of the logistic function



(We can make it steeper by premultiplying the exponent by a constant; we can change the midpoint by adding an intercept to the exponent.)

This specifies the model,

$$y \sim \text{Bernoulli}(\mu(\beta^\top x)). \tag{15}$$

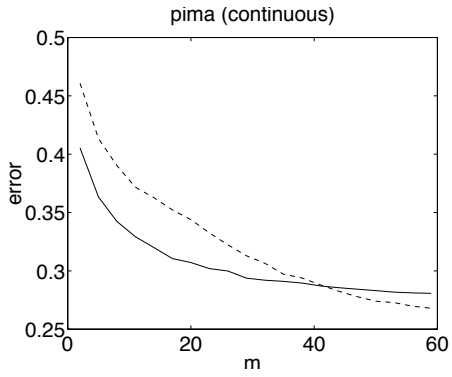
Note that the graphical model is identical to linear regression.

Important: *The covariates enter the probability of the response through a linear combination with the coefficients.* That linear combination is then passed through a function to make a quantity that is appropriate as a parameter to the distribution of the response. In the plot, the x axis is $\beta^\top x$. In this way, the covariates and coefficients control the probability distribution of the response.

Optional: Generative vs. discriminative modeling. As for linear regression we always assume that x_n is observed. Contrasting this with naive Bayes classification, this is a

discriminative model. The folk wisdom, made more formal by Ng and Jordan (2002), is that discriminative models give better performance as more data are observed. Of course "more" must take into account dimension and number of data points. (For example, 100 1-dimensional points is "more" than 100 1000-dimensional points.)

Here is an example picture from their paper:



The paper includes many such pictures, for many data sets.

2.1 Fitting logistic regression with maximum likelihood

Our data are $\{(x_n, y_n)\}$ pairs, where x_n are covariates (as for linear regression) and y_n is a binary response (e.g., email features and spam/not spam). We fit the coefficients of logistic regression by maximizing the *conditional likelihood*,

$$\hat{\beta} = \arg\max_{\beta} \sum_{n=1}^{N} \log p(y_n \mid x_n, \beta). \tag{16}$$

The objective is

$$\mathcal{L} = \sum_{n=1}^{N} y_n \log \mu(\beta^\top x_n) + (1 - y_n) \log(1 - \mu(\beta^\top x_n)). \tag{17}$$

Define $\eta_n \triangleq \beta^\top x_n$ and $\mu_n \triangleq \mu(\eta_n)$, but do not lose sight that both depend on β. Define

$\mathcal{L}_n$ to be the nth term in the conditional likelihood,

$$\mathcal{L}_n = y_n \log \mu_n + (1 - y_n) \log(1 - \mu_n) \tag{18}$$

The full likelihood is $\mathcal{L} = \sum_{n=1}^{N} \mathcal{L}_n$.

Use the chain rule to calculate the derivative of the conditional likelihood,

$$\frac{d\mathcal{L}}{d\beta_i} = \sum_{n=1}^{N} \frac{d\mathcal{L}_n}{d\mu_n} \frac{d\mu_n}{d\beta_i}. \tag{19}$$

The first term is

$$\frac{d\mathcal{L}_n}{d\mu_n} = \frac{y_n}{\mu_n} - \frac{(1 - y_n)}{1 - \mu_n} \tag{20}$$

We use the chain rule again to compute the second term. The derivative of the logistic with respect to its argument η is

$$\frac{d\mu(\eta)}{d\eta} = \mu(\eta)(1 - \mu(\eta)). \tag{21}$$

(This is an exercise.) Now apply the chain rule

$$\begin{aligned} \frac{d\mu_n}{d\beta_i} &= \frac{d\mu_n}{d\eta_n} \frac{d\eta_n}{d\beta_i} && (22) \\ &= \mu_n(1 - \mu_n) x_{ni} && (23) \end{aligned}$$

With this reasoning, the derivative of each term is

$$\begin{aligned} \frac{d\mathcal{L}_n}{d\beta_i} &= \left(\frac{y_n}{\mu_n} - \frac{1 - y_n}{1 - \mu_n} \right) \mu_n(1 - \mu_n) x_{ni} && (24) \\ &= ((y_n(1 - \mu_n) - (1 - y_n)\mu_n) x_{ni} && (25) \\ &= (y_n - y_n\mu_n - \mu_n + y_n\mu_n) x_{ni} && (26) \\ &= (y_n - \mu_n) x_{ni} && (27) \end{aligned}$$

So, the full derivative is

$$\frac{d\mathcal{L}}{d\beta_i} = \sum_{n=1}^{N} (y_n - \mu(\beta^\top x_n)) x_{ni} \tag{28}$$

Logistic regression algorithms fit the objective with gradient methods, such as stochastic

gradient ascent. Nice closed-form solutions, like the normal equations in linear regression, are not available. But the likelihood is convex; there is a unique solution.

Note that $\mathbb{E}[y_n \mid x_n] = p(y_n \mid x_n) = \mu(\beta^\top x_n)$. Recall the *linear regression* derivative. (Warning: In the equation below y_n is real valued.)

$$\frac{d\mathcal{L}}{d\beta_i} = \sum_{n=1}^{N} (y_n - \beta^\top x_n) x_{ni} \tag{29}$$

Further recall that in linear regression, $\mathbb{E}[y_n \mid x_n] = \beta^\top x_n$.

Both the linear regression and logistic regression derivatives have the form

$$\sum_{n=1}^{N} (y_n - \mathbb{E}[y \mid x_n, \beta]) x_{ni} \tag{30}$$

(Something is happening here... more later.)

Linear separators and the margin. Suppose there are two covariates and two classes.

[Draw a plane with "+" and "-" labeled points, separable.]

Q: What kind of classification boundary does logistic regression create?

- Q: When does $p(+ \mid x) = 1/2$? A: When $\eta(x) = 0$
- Q: Where does $\beta^\top x_n = 0$? A: A line in covariate space.
- Q: What happens when $\beta^\top x < 0$? A: $p(+ \mid x) < 1/2$
- Q: What happens when $\beta^\top x > 0$? A: $p(+ \mid x) < 1/2$

So, the boundary around 1/2 occurs where $\beta^\top x = 0$. Thus, logistic regression finds a *linear separator*. (Many of you have seen SVMs. They also find a linear separator, but non-probabilistic.)

Furthermore, those of you who know about SVMs know about the *margin*. Intuitively, SVMs don't care about points that are easy to classify; rather, they try to separate the points that are difficult.

Loosely, the same idea applies here (Hastie et al., 2009).

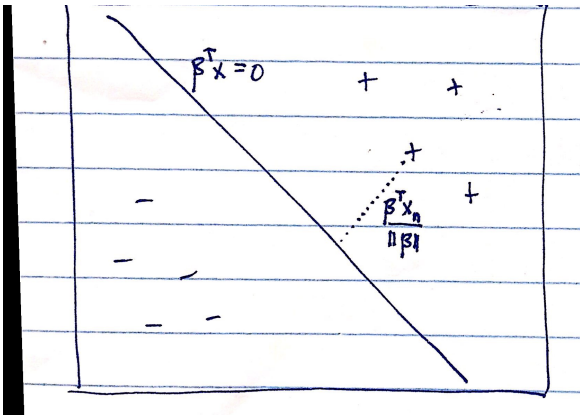
- The argument to the logistic is $\beta^\top x_n$. It is the distance to the separator scaled by $||\beta||$.

- What happens to the likelihood of a data point as we get further away from the separator?

  See the plot of the logistic function. Suppose we are moving the boundary that changes a (correctly classified) point from being 1 inch away to being 2 inches away. This will have a large positive effect on the likelihood. Now suppose that same moves changes another point from being 100 inches away to 101 inches away. The relative change in likelihood is much smaller. (The reason: the shape of the logistic function.)

- This means that the probability of the data changes most near the line. Thus, when we maximize likelihood, we focus on the margin.

Aside: Separable data. Here is some separable data:



Consider $\beta' = \frac{\beta}{||\beta||}$. Hold β' fixed and consider the likelihood as a function of the scaling, $c \triangleq ||\beta||$. If we increase c then (a) the separating plane does not change and (b) the likelihood goes up. Thus, the MLE for separable data is not well-defined.

3 Generalized linear models

Linear regression and logistic regression are both *linear models*. The coefficient β enters the distribution of y_n through a linear combination of x_n. The difference is in the type of the response. In linear regression the response is real valued; in logistic regression the response is binary.

Linear and logistic regression are instances for a more general class of models, *generalized*

linear models (GLMs) (McCullagh and Nelder, 1989). The idea is to use a general exponential family for the response distribution. In addition to real and binary responses, GLMs can handle categorical, positive real, positive integer, and ordinal responses.

The idea behind logistic and linear regression: The conditional expectation of y_n depends on x_n through a function of a linear relationship,

$$\mathbb{E}\left[y_n \mid x_n, \beta\right] = f(\beta^\top x_n) = \mu_n$$

In linear regression, f is the identity; in logistic regression, f is the logistic. Finally, these methods endow y_n with a distribution that depends on μ_n. In linear regression, the distribution is a Gaussian; in logistic regression, it is a Bernoulli.

GLMs generalize this idea with an exponential family. The generic GLM has the following form,

$$\begin{aligned} p(y_n \mid x_n) &= \log h(y_n) \exp\{\eta_n^\top y_n - a(\eta_n)\} && (31) \\ \eta_n &= \psi(\mu_n) && (32) \\ \mu_n &= f(\beta^\top x_n) && (33) \end{aligned}$$

Note:

- The input x_n enters the model through $\beta^\top x_n$.
- The conditional mean μ_n is a function $f(\beta^\top x_n)$. It is called the *response function* or *link function*.
- The response y_n has conditional mean μ_n.
- Its natural parameter is denoted $\eta_n = \psi(\mu_n)$.

GLMs let us build probabilistic predictors of many kinds of responses.

There are two choices to make in a GLM: the distribution of the response y and the function that gives us its mean $f(\beta^\top x)$. The distribution is usually determined by the form of y (e.g., Gaussian for real, Poisson for integer, Bernoulli for binary). The response function is somewhat constrained—it must give a mean in the right space for the distribution of y—but also offers some freedom. For example, we can use a logistic function or a probit function

when modeling binary data—both map the reals to $(0, 1)$.

Recall our discussion of the mean parameter and natural parameter in an exponential family, specifically, that there is a 1-1 relationship between the mean $\mathbb{E}[Y]$ and the natural parameter η. We denote the mapping from the mean to the natural parameter by $\psi(\mu) = \eta$; we denote the mapping from the natural parameter to the mean by $\psi^{-1}(\eta) = \mathbb{E}[Y]$.

We can use ψ^{-1} as a response function—this is called the *canonical response function*. When we use the canonical response function, then the natural parameter *is* $\beta^\top x_n$,

$$p(y_n \mid x_n) = h(y_n) \exp\{(\beta^\top x_n)^\top t(y_n) - a(\eta_n)\}. \tag{34}$$

The logistic function (for binary response) and identity function (for real response) are examples of canonical response functions.

3.1 Fitting a GLM

We fit GLM methods with gradient optimization. It suffices to investigate the likelihood and its derivative.

The data are input/response pairs $\{(x_n, y_n)\}$. The conditional likelihood is

$$\mathcal{L}(\beta; \mathcal{D}) = \sum_{n=1}^{N} h(y_n) + \eta_n^\top t(y_n) - a(\eta_n), \tag{35}$$

and recall that η_n is a function of β and x_n (via f and ψ).

Define each term to be $\mathcal{L}_n$. The gradient is

$$\begin{aligned}
\nabla_\beta \mathcal{L} &= \sum_{n=1}^{N} \nabla_{\eta_n} \mathcal{L}_n \nabla_\beta \eta_n && (36)\\
&= \sum_{n=1}^{N} (t(y_n) - \nabla_{\eta_n} a(\eta_n)) \nabla_\beta \eta_n && (37)\\
&= \sum_{n=1}^{N} (t(y_n) - \mathbb{E}[Y \mid x_n, \beta])(\nabla_{\mu_n} \eta_n)(\nabla_{\beta^\top x_n} \mu_n) x_n && (38)
\end{aligned}$$

In a canonical GLM, $\eta_n = \beta^\top x_n$ and $\nabla_\beta \eta_n = x_n$. Therefore,

$$\nabla_\beta \mathcal{L} = \sum_{n=1}^{N} (t(y_n) - \mathbb{E}\left[Y \mid x_n, \beta\right]) x_n \tag{39}$$

Recall that the logistic and linear regression derivatives had this form. These are examples of generalized linear models with canonical links.

On small data, fitting GLM's is easy in R: use the `glm` command.

References

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning*. Springer, 2 edition.

McCullagh, P. and Nelder, J. A. (1989). *Generalized Linear Models*. London: Chapman and Hall.

Ng, A. and Jordan, M. (2002). On discriminative versus generative classifiers: A comparison of logistic regression and naive Bayes. In *Advances in Neural Information Processing Systems 14*.