Machine Learning
Regularization and Binary Data

Rajesh Ranganath
- What happens if there are too many features?
- What if the outcome variable is binary?
Crohn’s Disease

Goal: Understand Role of Genetics in Severity of Crohn’s Disease
How to answer this question?

[ Kerras+ 2017 ]
How to answer this question?

[Figure S2: Population structure inferred from the TGP data set using the TRUCkTRUERG algorithm at three settings for the number of populations $K$. The visualization of the reds in the Figure shows patterns consistent with the major geographical regions. Some of the clusters identify a specific region (e.g. red for Africa) while others represent admixture between regions (e.g. green for Europeans and Central/South Americans). The presence of clusters that are shared between different regions demonstrates the more continuous nature of the structure. The new cluster from $K=7$ to $K=8$ matches structure differentiating between American groups. For $K=9$, the new cluster is unpopulated.]

[Gopalan+ 2016]
How to answer this question?

Measure their Crohn’s disease severity risk
How to answer this question?

- How many people?

- A high estimate of prevalence is .2% of the US population

- Approximately 600K people in the US

- 10K people would be a broad study
How to answer this question?

How many genetic variants?

> 10 million
How to answer this question?

How many genetic variants?

> 10 million
Math to find importance?

Use Linear Regression!
Math to find importance?

Use Linear Regression!

$$\min_{\theta} \frac{1}{n} \sum_{i=1}^{n} (\theta^T x_i - y_i)^2$$
Math to find importance?

Use Linear Regression!

$$\min_{\theta} \frac{1}{n} \sum_{i=1}^{n} (\theta^\top x_i - y_i)^2$$

- How do we get feature importances?

- Need to standardize $\theta_i$. How?
  - One way: subtract mean and divide by standard deviation.
Math to find importance?

Use Linear Regression!

\[
\min_{\theta} \frac{1}{n} \sum_{i=1}^{n} (\theta^\top x_i - y_i)^2
\]

- How do we get feature importances?
- Just \( \theta_i \) doesn’t work. Why?
Math to find importance?

Use Linear Regression!

\[
\min_{\theta} \frac{1}{n} \sum_{i=1}^{n} (\theta^\top x_i - y_i)^2
\]

- How do we get feature importances?
- Just \( \theta_i \) doesn’t work. Why?
- Need to standardize \( \theta_i \). How?
Math to find importance?

Use Linear Regression!

$$
\min_{\theta} \frac{1}{n} \sum_{i=1}^{n} (\theta^\top x_i - y_i)^2
$$

- How do we get feature importances?
- Just $\theta_i$ doesn’t work. Why?
- Need to standardize $\theta_i$. How?
- One way: subtract mean and divide by standard deviation
What happens when we do linear regression?

Number of features is greater than number of data points
What happens when we do linear regression?

Number of features is greater than number of data points

Lots of solutions! Given by

\[ \mathbf{X}\theta = y \]

Solutions span \( p - n \) subspace (under linear independence of \( \mathbf{x}_i \))

*Need to restrict model complexity in some way*
How do we restrict linear regression?
Model Complexity in Linear Regression

- Throw away variables
- Combine variables
- Sequentially add variables one at a time
- Restrict the parameters of linear regression
Norms

The $\alpha$-norm of a vector:

$$||\theta||_\alpha = \left( \sum_{i=1}^{p} |\theta_i|^\alpha \right)^{\frac{1}{\alpha}}$$

Common norms included

- $\alpha = 2$: Euclidean norm
- $\alpha = 1$: Manhattan distance
- $\alpha = \infty$: Maximum norm

Norms of difference give you a type of distance
Regularization

Add a function to the minimization

\[ \mathcal{L}(\theta) = \sum_{i=1}^{n} (\theta^\top x_i - y_i)^2 + \lambda \|\theta\|_2^2 \]

Penalizes the norm of the parameters

Called

- Ridge regression
- L2 regression
Solution

We take the derivative and set it equal to zero

$$\nabla \mathcal{L} = \sum_{i=1}^{n} 2(\theta^\top x_i - y_i)x_i + 2\lambda \theta$$
Solution

We take the derivative and set it equal to zero

\[ \nabla \mathcal{L} = \sum_{i=1}^{n} 2(\theta^\top x_i - y_i)x_i + 2\lambda \theta \]

Setting it equal to zero

\[ \theta = \left( \left( \sum_{i=1}^{n} x_i x_i^\top \right) + \lambda I \right)^{-1} \sum_{i=1}^{n} y_i x_i \]

Or in matrix form

\[ \theta = (X^\top X + \lambda I)^{-1} X^\top y \]
\[ \theta = \left( \left( \sum_{i=1}^{n} x_i x_i^\top \right) + \lambda I \right)^{-1} \sum_{i=1}^{n} y_i x_i \]

- When \( \lambda \) large

\[ \left( \left( \sum_{i=1}^{n} x_i x_i^\top \right) + \lambda I \right)^{-1} \]

is small on diagonal

- When \( \lambda \) small

\[ \left( \left( \sum_{i=1}^{n} x_i x_i^\top \right) + \lambda I \right)^{-1} \]

Looks close to least squares. Single solution!

- When \( n \) is large converges to least squares
Understanding l2-regression

Consider a general regularized minimization problem

$$\min_{\theta} f(\theta) + \lambda g(\theta)$$
Understanding l2-regression

Consider a general regularized minimization problem

$$\min_{\theta} f(\theta) + \lambda g(\theta)$$

Now consider an alternative formulation

$$\min_{\theta} f(\theta)
\quad g(\theta) \leq M$$

Directly limits $g(\theta)$ rather than penalizing
Understanding l2-regression

\[
\min_{\theta} f(\theta) + \lambda g(\theta)
\]

Suppose for a fixed \( \lambda \), the optimal \( \theta \) is \( \theta^* \)
Understanding l2-regression

$$\min_{\theta} f(\theta) + \lambda g(\theta)$$

Suppose for a fixed $\lambda$, the optimal $\theta$ is $\theta^*$

Then in

$$\min_{\theta} f(\theta)$$

$$g(\theta) \leq M$$

set $M = g(\theta^*)$
Understanding l2-regression

Now suppose that there \( \hat{\theta} \neq \theta^* \) is the minimum of

\[
\min_{\theta} f(\theta) \\
g(\theta) \leq M
\]

with \( M = g(\theta^*) \). That is,

\[
f(\hat{\theta}) < f(\theta^*)
\]

Putting this together with the constraint gives

\[
f(\hat{\theta}) + \lambda g(\hat{\theta}) < f(\theta^*) + \lambda g(\theta^*)
\]

This contradicts the minimality of \( \theta^* \)
Understanding l2-regression

Consider the ridge regression problem

$$\min_{\theta} \sum_{i=1}^{n} (\theta^\top x_i - y_i)^2 + \lambda \|\theta\|_2^2$$
Understanding l2-regression

Consider the ridge regression problem

$$\min_{\theta} \sum_{i=1}^{n} (\theta^\top x_i - y_i)^2 + \lambda \|\theta\|_2^2$$

for every $\lambda$ is equivalent to

$$\min_{\theta} \sum_{i=1}^{n} (\theta^\top x_i - y_i)^2$$

$$\|\theta\|_2^2 \leq M$$

for some $M$.

L2 regularized ridge regression constrains parameters to a ball
Figure 3.4 Plot of the contours of the unregularized error function (blue) along with the constraint region (3.30) for the quadratic regularizer $q = 2$ on the left and the lasso regularizer $q = 1$ on the right, in which the optimum value for the parameter vector $\mathbf{w}$ is denoted by $\mathbf{w}^\star$.

The lasso gives a sparse solution in which $w_1^\star = 0$.

For the remainder of this chapter we shall focus on the quadratic regularizer (3.27) both for its practical importance and its analytical tractability.

### 3.1.5 Multiple outputs

So far, we have considered the case of a single target variable $t$. In some applications, we may wish to predict $K > 1$ target variables, which we denote collectively by the target vector $\mathbf{t}$. This could be done by introducing a different set of basis functions for each component of $\mathbf{t}$, leading to multiple, independent regression problems.

However, a more interesting, and more common, approach is to use the same set of basis functions to model all of the components of the target vector so that

$$ y(x, \mathbf{w}) = \mathbf{W}^\top \phi(x) \quad (3.31) $$

where $y$ is a $K$-dimensional column vector, $\mathbf{W}$ is an $M \times K$ matrix of parameters, and $\phi(x)$ is an $M$-dimensional column vector with elements $\phi_j(x)$, with $\phi_0(x) = 1$ as before. Suppose we take the conditional distribution of the target vector to be an isotropic Gaussian of the form

$$ p(\mathbf{t}|x, \mathbf{W}, \beta) = N(\mathbf{t}|\mathbf{W}^\top \phi(x), \beta^{-1}I) \quad (3.32) $$

If we have a set of observations $t_1, \ldots, t_N$, we can combine these into a matrix $\mathbf{T}$ of size $N \times K$ such that the $n$th row is given by $\mathbf{t}_n^\top$. Similarly, we can combine the input vectors $x_1, \ldots, x_N$ into a matrix $\mathbf{X}$. The log likelihood function is then given by

$$ \ln p(\mathbf{T}|\mathbf{X}, \mathbf{W}, \beta) = \sum_{n=1}^{N} \ln N(\mathbf{t}_n|\mathbf{W}^\top \phi(x_n), \beta^{-1}I) = NK \ln (\beta^2 \pi) - \beta^2 \sum_{n=1}^{N} \| \mathbf{t}_n - \mathbf{W}^\top \phi(x_n) \|^2. \quad (3.33) $$

[Bishop 2006]
Why do the level sets look like circles?
3.4 Shrinkage Methods

Coefficients

-0.2 0.0 0.2 0.4 0.6

\[ \text{df}(\lambda) \]

FIGURE 3.8. Profiles of ridge coefficients for the prostate cancer example, as the tuning parameter \( \lambda \) is varied. Coefficients are plotted versus \( \text{df}(\lambda) \), the effective degrees of freedom. A vertical line is drawn at \( \text{df} = 5.0 \), the value chosen by cross-validation.
Why not penalize with other norms?
L1 vs L2 norm

Leads to sparsity because of bigger penalties near zero
Regularization

Add a function to the minimization

\[ \mathcal{L}(\theta) = \sum_{i=1}^{n} (\theta^\top x_i - y_i)^2 + \lambda ||\theta||_1 \]

Penalizes the norm of the parameters

Called
- Lasso regression
- L1 regression
L1 Regularization: How do we solve?

Add a function to the minimization

\[ \mathcal{L}(\theta) = \sum_{i=1}^{n} (\theta^\top x_i - y_i)^2 + \lambda ||\theta||_1 \]

The absolute value is not differentiable. Options
- Can use quadratic programming
- Custom solver?
An Alternative View

$$\min_{\theta} \sum_{i=1}^{n} (\theta^\top x_i - y_i)^2 + \lambda ||\theta||_1$$
An Alternative View

$$\min_{\theta} \sum_{i=1}^{n}(\theta^\top x_i - y_i)^2 + \lambda \|\theta\|_1$$

for every $\lambda$ is equivalent to

$$\min_{\theta} \sum_{i=1}^{n}(\theta^\top x_i - y_i)^2$$

$$\|\theta\|_1 \leq M$$

for some $M$.

L1 regularization constrains parameters to a diamond
Figure 3.4 Plot of the contours of the unregularized error function (blue) along with the constraint region (3.30) for the quadratic regularizer \( q = 2 \) on the left and the lasso regularizer \( q = 1 \) on the right, in which the optimum value for the parameter vector \( w \) is denoted by \( w^\star \).

The lasso gives a sparse solution in which \( w_1^\star = 0 \).

For the remainder of this chapter we shall focus on the quadratic regularizer (3.27) both for its practical importance and its analytical tractability.

3.1.5 Multiple outputs

So far, we have considered the case of a single target variable \( t \). In some applications, we may wish to predict \( K > 1 \) target variables, which we denote collectively by the target vector \( t \). This could be done by introducing a different set of basis functions for each component of \( t \), leading to multiple, independent regression problems.

However, a more interesting, and more common, approach is to use the same set of basis functions to model all of the components of the target vector so that

\[
y(x, w) = W^T \phi(x) \tag{3.31}
\]

where \( y \) is a \( K \)-dimensional column vector, \( W \) is an \( M \times K \) matrix of parameters, and \( \phi(x) \) is an \( M \)-dimensional column vector with elements \( \phi_j(x) \), with \( \phi_0(x) = 1 \) as before. Suppose we take the conditional distribution of the target vector to be an isotropic Gaussian of the form

\[
p(t|x, W, \beta) = N(t|W^T \phi(x), \beta^{-1} I) \tag{3.32}
\]

If we have a set of observations \( t_1, \ldots, t_N \), we can combine these into a matrix \( T \) of size \( N \times K \) such that the \( n \)th row is given by \( t_T_n \). Similarly, we can combine the input vectors \( x_1, \ldots, x_N \) into a matrix \( X \). The log likelihood function is then given by

\[
\ln p(T|X, W, \beta) = N \sum_{n=1}^{N} \ln N(t_n|W^T \phi(x_n), \beta^{-1} I) = NK \ln \left( \frac{\beta^{-2}}{2\pi} \right) - \frac{\beta^2}{2} N \sum_{n=1}^{N} ||t_n - W^T \phi(x_n)||^2. \tag{3.33}
\]
72 3. Linear Methods for Regression

The region for ridge regression is the disk $\beta_1^2 + \beta_2^2 \leq t$, while for lasso it is the diamond $|\beta_1| + |\beta_2| \leq t$. Both methods find the first point where the elliptical contours hit the constraint region. Unlike the disk, the diamond has corners; if the solution occurs at a corner, then it has an $\alpha$-parameter $\beta_j$ equal to zero. When $p > 2$, the diamond becomes a rhomboid, and has many corners, flat edges and faces; there are many more opportunities for the estimated parameters to be zero.

We can generalize ridge regression and the lasso, and view them as Bayes estimates. Consider the criterion

$$\hat{\beta} = \text{arg min}_{\beta} \left\{ \sum_{i=1}^{N} \left( y_i - \beta_0 - \sum_{j=1}^{p} x_{ij} \beta_j \right)^2 + \lambda p \sum_{j=1}^{p} |\beta_j|^q \right\}$$

(3.53)

for $q \geq 0$. The contours of constant value of $\sum_j |\beta_j|^q$ are shown in Figure 3.12, for the case of two inputs.

Thinking of $|\beta_j|^q$ as the log-prior density for $\beta_j$, the area of these equi-contours of the prior distribution of the parameters. The value $q = 0$ corresponds to variable subset selection, as the penalty simply counts the number of nonzero parameters; $q = 1$ corresponds to the lasso, while $q = 2$ or ridge regression. Notice that for $q \leq 1$, the prior is not uniform in direction, but concentrates more mass in the coordinate directions. The prior corresponding to the $q = 1$ case is an independent double exponential (or Laplace) distribution for each input, with density $(1/2\tau) \exp(-|\beta|/\tau)$ and $\tau = 1/\lambda$.

The case $q = 1$ (lasso) is the smallest $q$ such that the constraint region is convex; non-convex constraint regions make the optimization problem more difficult.

In this view, the lasso, ridge regression and best subset selection are Bayes estimates with different priors. Note, however, that they are derived as posterior modes, that is, maximizers of the posterior. It is more common to use the mean of the posterior as the Bayes estimate. Ridge regression is also the posterior mean, but the lasso and best subset selection are not.

Looking again at the criterion (3.53), we might try using other values of $q$ besides 0, 1, or 2. Although one might consider estimating $q$ from the data, our experience is that it is not worth the effort for the extra variance incurred. Values of $q \in (1, 2)$ suggest a compromise between the lasso and ridge regression. Although this is the case, with $q > 1$, $|\beta_j|^q$ is differentiable at 0, and so does not share the ability of lasso ($q = 1$) for $q = 4$ $q = 2$ $q = 1$ $q = 0.5$ $q = 0.1$

[Elements of Statistical Learning]
FIGURE 3.10. Profiles of lasso coefficients, as the tuning parameter \( t \) is varied. Coefficients are plotted versus \( s = t / \sum_p |\hat{\beta}_j| \). A vertical line is drawn at \( s = 0 \), the value chosen by cross-validation. Compare Figure 3.8 on page 65; the lasso profiles hit zero, while those for ridge do not. The profiles are piece-wise linear, and so are computed only at the points displayed; see Section 3.4.4 for details.
Understanding the L1 regularization

- Tries to pick individual features

- With a bunch of correlated features, what happens?
Understanding the L1 regularization

- Tries to pick individual features

- With a bunch of correlated features, what happens?

- Selects one

- This could harm prediction as a linear projection of the features might be more stable

Can we combine L1 and L2?
$\ell_1 + \ell_2$
For $\alpha \in [0, 1]$

$$\mathcal{L}(\theta) = \sum_{i=1}^{n} (\theta^\top x_i - y_i)^2 + \lambda (\alpha \|\theta\|_1 + (1 - \alpha) \|\theta\|_2^2)$$

Called the elastic net

- Used in many machine learning libraries
- Works well in practice
Recall

What if $f$ is not linear

$$\min_{f} \mathbb{E}_{p(x,y)}[(y - f(x))^2]$$
Recall

What if $f$ is not linear

$$
\min_f \mathbb{E}_{p(x,y)}[(y - f(x))^2]
$$

$$
= \min_f \mathbb{E}_{p(x)}[(\mathbb{E}[y|x] - f(x))^2] + \mathbb{E}_{p(x)}[\mathbb{V} \mathbb{a}r(y|x)]
$$

*The optimal $f$ is the conditional expectation*
Where’s overfitting? Where’s the analysis loose?
Need to consider where the training data came from

\[ \mathcal{D} = (x_1, y_1) \ldots (x_n, y_n) \sim p \]

*The training data set is a random variable!*
The training data set is a random variable!

- Imagine wanting to predict heights from diet
The training data set is a random variable!

- Imagine wanting to predict heights from diet

- Step 1? Get Data
The training data set is a random variable!

- Imagine wanting to predict heights from diet

- Step 1? Get Data

- How? A person doing a random survey of people’s heights and diet
  Gets $\mathcal{D}_1$

- Another doing a random survey of people’s heights and diet
  Gets $\mathcal{D}_2$

- $\mathcal{D}_1$ and $\mathcal{D}_2$ are random because the full population was not surveyed

- $\mathcal{D}_1$ and $\mathcal{D}_2$ are drawn from the same distribution
Refine our Analysis

Include data randomness

\[
\min_f \mathbb{E}_{p(x,y), \mathcal{D} \sim p} [(y - f(x; \mathcal{D}))^2]
\]
Refine our Analysis

Include data randomness

\[
\min_f \mathbb{E}_{p(x,y), \mathcal{D} \sim p}[(y - f(x; \mathcal{D}))^2] = \min_f \mathbb{E}_{p(x), \mathcal{D} \sim p}[(\mathbb{E}[y | x] - f(x; \mathcal{D}))^2] + \mathbb{E}_{p(x)}[\text{Var}(y | x)]
\]
Refine our Analysis

Include data randomness

\[
\min_f \mathbb{E}_{p(x, y), \mathcal{D} \sim p} [(y - f(x; \mathcal{D}))^2] \\
= \min_f \mathbb{E}_{p(x), \mathcal{D} \sim p} [(\mathbb{E}[y | x] - f(x; \mathcal{D}))^2] + \mathbb{E}_{p(x)}[\text{Var}(y | x)] \\
= \min_f \mathbb{E}_{p(x), \mathcal{D} \sim p} [(\mathbb{E}[y | x] - \mathbb{E}_{\mathcal{D} \sim p}[f(x); \mathcal{D}] + \mathbb{E}_{\mathcal{D} \sim p}[f(x); \mathcal{D}] - f(x; \mathcal{D}))^2] \\
+ \mathbb{E}_{p(x)}[\text{Var}(y | x)]
\]
Refine our Analysis

Include data randomness

\[
\begin{align*}
\min_f \mathbb{E}_{p(x,y), \mathcal{D} \sim p} [ (y - f(x; \mathcal{D}))^2 ] \\
= \min_f \mathbb{E}_{p(x), \mathcal{D} \sim p} [ (\mathbb{E}[y | x] - f(x; \mathcal{D}))^2 ] + \mathbb{E}_{p(x)}[\text{Var}(y | x)] \\
= \min_f \mathbb{E}_{p(x), \mathcal{D} \sim p} [ (\mathbb{E}[y | x] - \mathbb{E}_{\mathcal{D} \sim p}[f(x); \mathcal{D}] + \mathbb{E}_{\mathcal{D} \sim p}[f(x); \mathcal{D}] - f(x; \mathcal{D}))^2 ] \\
+ \mathbb{E}_{p(x)}[\text{Var}(y | x)] \\
= \min_f \mathbb{E}_{p(x), \mathcal{D} \sim p} [ (\mathbb{E}[y | x] - \mathbb{E}_{\mathcal{D} \sim p}[f(x); \mathcal{D}])^2 ] \\
+ 2(\mathbb{E}[y | x] - \mathbb{E}_{\mathcal{D} \sim p}[f(x); \mathcal{D}])(\mathbb{E}_{\mathcal{D} \sim p}[f(x); \mathcal{D}] - f(x; \mathcal{D})) \\
+ (\mathbb{E}_{\mathcal{D} \sim p}[f(x); \mathcal{D}] - f(x; \mathcal{D}))^2] \\
+ \mathbb{E}_{p(x)}[\text{Var}(y | x)]
\end{align*}
\]
Refine our Analysis

Continuing

$$\begin{align*}
\min_f \mathbb{E}_{p(x), \mathcal{D} \sim p} [(y - f(x; \mathcal{D}))^2] \\
= \min_f \mathbb{E}_{p(x), \mathcal{D} \sim p} \left[ \left( \mathbb{E}[y | x] - \mathbb{E}_{\mathcal{D} \sim p}[f(x); \mathcal{D}] \right)^2 \right. \\
+ 2(\mathbb{E}[y | x] - \mathbb{E}_{\mathcal{D} \sim p}[f(x); \mathcal{D}])(\mathbb{E}_{\mathcal{D} \sim p}[f(x); \mathcal{D}] - f(x; \mathcal{D})) \\
+ (\mathbb{E}_{\mathcal{D} \sim p}[f(x); \mathcal{D}] - f(x; \mathcal{D}))^2 \\
+ \mathbb{E}_{p(x)}[\text{Var}(y | x)]
\end{align*}$$
Refine our Analysis

Continuing

\[
\begin{align*}
\min_f & \mathbb{E}_{p(x), \mathcal{D} \sim p} [(y - f(x; \mathcal{D}))^2] \\
= & \min_f \mathbb{E}_{p(x), \mathcal{D} \sim p} [ (\mathbb{E}[y | x] - \mathbb{E}_{\mathcal{D} \sim p}[f(x); \mathcal{D}])^2 \\
+ & 2(\mathbb{E}[y | x] - \mathbb{E}_{\mathcal{D} \sim p}[f(x); \mathcal{D}]) (\mathbb{E}_{\mathcal{D} \sim p}[f(x); \mathcal{D}] - f(x; \mathcal{D})) \\
+ & (\mathbb{E}_{\mathcal{D} \sim p}[f(x); \mathcal{D}] - f(x; \mathcal{D}))^2 ] \\
+ & \mathbb{E}_{p(x)}[\text{Var}(y | x)] \\
= & \min_f \mathbb{E}_{p(x), \mathcal{D} \sim p} [ (\mathbb{E}[y | x] - \mathbb{E}_{\mathcal{D} \sim p}[f(x); \mathcal{D}])^2 \\
+ & (\mathbb{E}_{\mathcal{D} \sim p}[f(x); \mathcal{D}] - f(x; \mathcal{D}))^2 ] \\
+ & \mathbb{E}_{p(x)}[\text{Var}(y | x)]
\end{align*}
\]
Refine our Analysis

\[
\min_f \mathbb{E}_{p(x,y), D \sim p} [(y - f(x; D))^2]
\]

\[
= \min_f \mathbb{E}_{p(x), D \sim p} 
\left[ (\mathbb{E}[y | x] - \mathbb{E}_{D \sim p}[f(x); D])^2 + (\mathbb{E}_{D \sim p}[f(x); D] - f(x; D))^2 \right] 
+ \mathbb{E}_{p(x,y)}[\text{Var}(y | x)]
\]

\[
= \min_f \mathbb{E}_{p(x)} 
\left[ (\mathbb{E}[y | x] - \mathbb{E}_{D \sim p}[f(x); D])^2 \right] 
+ \mathbb{E}_{D \sim p} 
\left[ (\mathbb{E}_{D \sim p}[f(x); D] - f(x; D))^2 \right] 
+ \mathbb{E}_{p(x)}[\text{Var}(y | x)]
\]

Bias

Variance

Noise
\[
\min_f \mathbb{E}_{p(x)} \left[ (\mathbb{E}[y \mid x] - \mathbb{E}_{D \sim p}[f(x) \mid D])^2 \right] \\
\underbrace{\text{Bias}} \\
+ \mathbb{E}_{D \sim p} \left[ (\mathbb{E}_{D \sim p}[f(x) \mid D] - f(x) \mid D)^2 \right] \\
\underbrace{\text{Variance}} \\
+ \mathbb{E}_{p(x)}[\text{Var}(y \mid x)] \\
\underbrace{\text{Noise}}
\]

- Bias accounts for model mismatch of average predictor
- Variance accounts for finite data
- Noise accounts for \( y \) not being a deterministic function of \( x \)
Get two house price data sets $\mathcal{D}^1$, $\mathcal{D}^2$

- Small models tend to have more bias
- Big models tend to have more variance
Get two house price data sets $\mathcal{D}^1$, $\mathcal{D}^2$

- Small models tend to underfit
- Big models tend to overfit
Get two house price data sets $\mathcal{D}^1$, $\mathcal{D}^2$

- Bias is like underfitting
- Variance is like overfitting
Changing Gears: A New Problem
Crohn’s Disease

Goal: Understand Role of Genetics in Getting Crohn’s Disease
Crohn’s Disease

- Here the $y$ for each person is $\{0, 1\}$
- 0 means they do not have Crohn’s Disease
- 1 means they do have Crohn’s Disease

Called a *classification* problem
Maximum Likelihood

Principle of Maximum Likelihood

- Tries to maximize likelihood of the data
- Has some nice efficiency properties with growing
Maximum Likelihood

Likelihood function measures “probability” of data

\[ \ell(x_i; \theta) \]

When data are independent, we get

\[ \prod_i \ell(x_i; \theta) \]

Generally work with logs for stability

\[ \sum_i \log \ell(x_i; \theta) \]

and stochastic optimization for speed
Maximum Likelihood

Suppose we flipped a biased coin many times and got samples

\[ x_i \in 0, 1 \]

Now we want to compute the maximum likelihood estimate for the probability of heads of the coin.

*How do we start?*
Maximum Likelihood

1. We need a distribution over coin flips?
Maximum Likelihood

1. We need a distribution over coin flips?

\[ x_i \sim \text{Bernoulli}(p) = p^{x_i} (1 - p)^{1-x_i} \]
Maximum Likelihood

1. We need a distribution over coin flips?

\[ x_i \sim \text{Bernoulli}(p) = p^{x_i} (1 - p)^{1-x_i} \]

Why not another distribution?
Maximum Likelihood

1. We need a distribution over coin flips?

\[ x_i \sim \text{Bernoulli}(p) = p^{x_i} (1 - p)^{1-x_i} \]

Why not another distribution?

2. Write down the log-likelihood of the observations

\[ L(p) = \sum_{i=1}^{n} \log(p^{x_i} (1 - p)^{1-x_i}) \]

Maximize by taking derivatives and setting to zero.

What happens when you only observe zeros?
Maximum Likelihood

1. We need a distribution over coin flips?

   \[ x_i \sim \text{Bernoulli}(p) = p^{x_i} (1 - p)^{1-x_i} \]

   Why not another distribution?

2. Write down the log-likelihood of the observations

   \[ \mathcal{L}(p) = \sum_{i=1}^{n} \log(p^{x_i} (1 - p)^{1-x_i}) \]
Maximum Likelihood

1. We need a distribution over coin flips?

   \[ x_i \sim \text{Bernoulli}(p) = p^{x_i} (1 - p)^{1-x_i} \]

   Why not another distribution?

2. Write down the log-likelihood of the observations

   \[ \mathcal{L}(p) = \sum_{i=1}^{n} \log(p^{x_i} (1 - p)^{1-x_i}) \]

3. Maximize by taking derivatives and setting to zero
Maximum Likelihood

1. We need a distribution over coin flips?

\[ x_i \sim \text{Bernoulli}(p) = p^{x_i}(1 - p)^{1-x_i} \]

Why not another distribution?

2. Write down the log-likelihood of the observations

\[ \mathcal{L}(p) = \sum_{i=1}^{n} \log(p^{x_i}(1 - p)^{1-x_i}) \]

3. Maximize by taking derivatives and setting to zero

What happens when you only observe zeros?
Maximum Likelihood

1. Need a distribution over each observed data point
2. Write down the log-likelihood of the observations
3. Maximize the log-likelihood (by gradients)
ML for Binary Outcome Linear Regression

1. Need a distribution over each observed data point
ML for Binary Outcome Linear Regression

1. Need a distribution over each observed data point

\[ p(y_i = 1 | x_i; \theta) = \frac{1}{1 + \exp(-\theta^\top x_i)} := \sigma(\theta^\top x_i) \]

\[ \sigma \text{ is the sigmoid or logistic function maps from } \mathbb{R} \text{ to } (0, 1) \]
ML for Binary Outcome Linear Regression

1. Need a distribution over each observed data point

\[
p(y_i = 1 | x_i; \theta) = \frac{1}{1 + \exp(-\theta^\top x_i)} := \sigma(\theta^\top x_i)
\]

\(\sigma\) is the sigmoid or logistic function maps from \(\mathbb{R}\) to \((0, 1)\)

2. Write down log-likelihood of the \(n\) observations
ML for Binary Outcome Linear Regression

1. Need a distribution over each observed data point

\[ p(y_i = 1 \mid x_i; \theta) = \frac{1}{1 + \exp(-\theta^\top x_i)} := \sigma(\theta^\top x_i) \]

\( \sigma \) is the sigmoid or logistic function maps from \( \mathbb{R} \) to \((0, 1)\)

2. Write down log-likelihood of the \( n \) observations

\[
\sum_{i=1}^{n} \log(p(y_i \mid x_i; \theta)) = \sum_{i=1}^{n} y_i \log \sigma(\theta^\top x_i) + (1 - y_i) \log(1 - \sigma(\theta^\top x_i))
\]
ML for Binary Outcome Linear Regression

1. Need a distribution over each observed data point

\[ p(y_i = 1 \mid x_i; \theta) = \frac{1}{1 + \exp(-\theta^\top x_i)} := \sigma(\theta^\top x_i) \]

\( \sigma \) is the sigmoid or logistic function maps from \( \mathbb{R} \) to \((0, 1)\)

2. Write down log-likelihood of the \( n \) observations

\[
\sum_{i=1}^{n} \log(p(y_i \mid x_i; \theta)) = \sum_{i=1}^{n} y_i \log \sigma(\theta^\top x_i) + (1 - y_i) \log(1 - \sigma(\theta^\top x_i))
\]

3. Maximize with respect to \( \theta \)

*Called logistic regression*
Maximize

$$\nabla_\theta \mathcal{L} = \nabla_\theta \sum_{i=1}^{n} y_i \log \sigma(\theta^\top x_i) + (1 - y_i) \log(1 - \sigma(\theta^\top x_i))$$

$$= \sum_{i=1}^{n} y_i \frac{\sigma'(\theta^\top x_i)x_i}{\sigma(\theta^\top x_i)} + (y_i - 1) \frac{\sigma'(\theta^\top x_i)x_i}{1 - \sigma(\theta^\top x_i)}$$
Maximize

\[
\nabla_\theta \mathcal{L} = \nabla_\theta \sum_{i=1}^{n} y_i \log \sigma(\theta^\top x_i) + (1 - y_i) \log(1 - \sigma(\theta^\top x_i))
\]

\[
= \sum_{i=1}^{n} y_i \frac{\sigma'(\theta^\top x_i) x_i}{\sigma(\theta^\top x_i)} + (y_i - 1) \frac{\sigma'(\theta^\top x_i) x_i}{1 - \sigma(\theta^\top x_i)}
\]

Use

\[
\sigma'(a) = \sigma(a)(1 - \sigma(a))
\]
Maximize

\[ \nabla_\theta \mathcal{L} = \nabla_\theta \sum_{i=1}^{n} y_i \log \sigma(\theta^\top x_i) + (1 - y_i) \log(1 - \sigma(\theta^\top x_i)) \]

\[ = \sum_{i=1}^{n} y_i \frac{\sigma'(\theta^\top x_i)x_i}{\sigma(\theta^\top x_i)} + (y_i - 1) \frac{\sigma'(\theta^\top x_i)x_i}{1 - \sigma(\theta^\top x_i)} \]

Use

\[ \sigma'(a) = \sigma(a)(1 - \sigma(a)) \]

Then

\[ \nabla_\theta \mathcal{L} = \sum_{i=1}^{n} x_i (y_i - y_i \sigma(\theta^\top x_i) + (y_i - 1)\sigma(\theta^\top x_i)) \]

\[ = \sum_{i=1}^{n} x_i (y_i - \sigma(\theta^\top x_i)) \]
Maximize

\[ \nabla_\theta \mathcal{L} = \nabla_\theta \sum_{i=1}^{n} y_i \log \sigma(\theta^\top x_i) + (1 - y_i) \log(1 - \sigma(\theta^\top x_i)) \]

\[ = \sum_{i=1}^{n} y_i \frac{\sigma'(\theta^\top x_i) x_i}{\sigma(\theta^\top x_i)} + (y_i - 1) \frac{\sigma'(\theta^\top x_i) x_i}{1 - \sigma(\theta^\top x_i)} \]

Use

\[ \sigma'(a) = \sigma(a)(1 - \sigma(a)) \]

Then

\[ \nabla_\theta \mathcal{L} = \sum_{i=1}^{n} x_i (y_i - y_i \sigma(\theta^\top x_i) + (y_i - 1) \sigma(\theta^\top x_i)) \]

\[ = \sum_{i=1}^{n} x_i (y_i - \sigma(\theta^\top x_i)) \]

Can we set it equal to zero?
\[ \nabla_\theta \mathcal{L} = \sum_{i=1}^{n} x_i (y_i - y_i \sigma(\theta^\top x_i) + (y_i - 1)\sigma(\theta^\top x_i)) \]

\[ = \sum_{i=1}^{n} x_i (y_i - \sigma(\theta^\top x_i)) \]

Use gradient or stochastic gradient ascent to maximize.

Is there a more intuitive form?
\[ \nabla_{\theta} \mathcal{L} = \sum_{i=1}^{n} x_i (y_i - y_i \sigma(\theta^T x_i) + (y_i - 1) \sigma(\theta^T x_i)) \]

\[ = \sum_{i=1}^{n} x_i (y_i - \sigma(\theta^T x_i)) \]

Use gradient or stochastic gradient ascent to maximize.

Is there a more intuitive form?

\[ \nabla_{\theta} \mathcal{L} = \sum_{i=1}^{n} x_i (y_i - E_{\theta}[y_i | x_i]) \]

Most gradients can seen as balancing various terms
Understanding Logistic Regression

We could have used other functions to map real $\mathbf{x}^\top \boldsymbol{\theta}$

- CDF of Gaussian maps from reals to (0, 1)
- CDF of Student-T maps similarly

Why use logistic regression?
All machine learning methods use assumptions to generalize

Important to understand those assumptions
Log odds. Start with logit function

$$logit(p) = \log\left(\frac{p}{1-p}\right)$$

Compute logit of $p(y = 1 \mid x)$

$$logit(p(y = 1 \mid x)) = \log\left(\frac{\frac{1}{1+\exp(-\theta^\top x_i)}}{\frac{\exp(-\theta^\top x_i)}{1+\exp(-\theta^\top x_i)}}\right) = -\log(\exp(-\theta^\top x)) = \theta^\top x$$

Logistic regression is linear in the logit of probabilities

Measure under which importance should be reported
Probabilistic Assumptions
Probabilistic Assumptions

Start with the logistic distribution

\[ p(a) = \frac{1}{4} \text{sech}^2 \left( \frac{x}{2} \right) = \frac{1}{(\exp(x/2) + \exp(-x/2))^2} \]

Logistic regression has a noise model

\[ \epsilon_i \sim \text{logistic} \]
\[ y_i = \theta \top x_i + \epsilon_i > 0 \]

- Noise comes from missing features under correct model
- Noise is logistic
Probabilistic Assumptions

Start with the logistic distribution

\[ p(a) = \frac{1}{4} \text{sech}^2 \left( \frac{x}{2} \right) = \frac{1}{(\exp(x/2) + \exp(-x/2))^2} \]

Logistic regression has a noise model

\[ \epsilon_i \sim \text{logistic} \]
\[ y_i = \theta^\top x_i + \epsilon_i > 0 \]

- Can use lots of distributions for noise
- Normal noise yields \textit{probit model}. Popular in economics
Probabilistic Assumptions

Start with the logistic distribution

\[ p(a) = \frac{1}{4} \text{sech}^2 \left( \frac{x}{2} \right) = \frac{1}{(\exp(x/2) + \exp(-x/2))^2} \]

Logistic regression has a noise model

\[ \epsilon_i \sim \text{logistic} \]
\[ y_i = \theta^\top x_i + \epsilon_i > 0 \]

- Has heavier tails than the normal distribution
- Fits “outliers” less
Regularization

If $p > n$, can overfit like linear regression. Can regularize!
Regularization

If $p > n$, can overfit like linear regression. Can regularize!

$$\sum_{i=1}^{n} y_i \log \sigma(\theta^\top x_i) + (1 - y_i) \log(1 - \sigma(\theta^\top x_i)) - \lambda \|\theta\|_2^2$$
Regularization

If $p > n$, can overfit like linear regression. Can regularize!

$$
\sum_{i=1}^{n} y_i \log \sigma(\theta^T x_i) + (1 - y_i) \log(1 - \sigma(\theta^T x_i)) - \lambda ||\theta||_1
$$

Does L1 give sparsity here?
Crohn’s Disease

Goal: Understand Role of Genetics in the Number of Crohn’s Related Doctor Visits
- $x_i$ are features
- $y_i$ are now counts

Do we have to derive a new model again for this?
- $x_i$ are features
- $y_i$ are now counts

Do we have to derive a new model again for this?

We need a distribution for $p(y | x)$. Can use exponential families
Exponential Families

Exponential families contain many popular distributions

- Bernoulli: Binary values
- Normal: Real values
- Gamma: Positive values
- Categorical: Multiple types
Exponential Families

Exponential family distribution

\[ p(x) = h(x) \exp(\eta^\top t(x) - a(\eta)) \]

- \( h \): Base measure
- \( \eta \): Parameters
- \( t(x) \): Sufficient statistics
- \( a(\eta) \): log-normalizer

\[ a(\eta) = \log \int h(x) \exp(\eta^\top t(x)) dx \]
Important property of exponential family

\[ \nabla a(\eta) = \nabla \log \int h(x) \exp(\eta^\top t(x)) dx \]
Important property of exponential family

$$\nabla a(\eta) = \nabla \log \int h(x) \exp(\eta^T t(x)) dx$$

$$= \frac{\nabla_\eta \int h(x) \exp(\eta^T t(x)) dx}{\int h(x) \exp(\eta^T t(x)) dx}$$
Important property of exponential family

\[ \nabla a(\eta) = \nabla \log \int h(x) \exp(\eta^\top t(x)) dx \]

\[ = \frac{\nabla_{\eta} \int h(x) \exp(\eta^\top t(x)) dx}{\int h(x) \exp(\eta^\top t(x)) dx} \]

\[ = \frac{\int h(x) \nabla_{\eta} \exp(\eta^\top t(x)) dx}{\int h(x) \exp(\eta^\top t(x)) dx} \]
Important property of exponential family

\[ \nabla a(\eta) = \nabla \log \int h(x) \exp(\eta^\top t(x)) dx \]

\[ = \frac{\nabla_\eta \int h(x) \exp(\eta^\top t(x)) dx}{\int h(x) \exp(\eta^\top t(x)) dx} \]

\[ = \frac{\int h(x) \nabla_\eta \exp(\eta^\top t(x)) dx}{\int h(x) \exp(\eta^\top t(x)) dx} \]

\[ = \frac{\int h(x) t(x) \exp(\eta^\top t(x)) dx}{\exp(\log(\int h(x) \exp(\eta^\top t(x)) dx))} \]
Important property of exponential family

\[ \nabla a(\eta) = \nabla \log \int h(x) \exp(\eta^\top t(x)) \, dx \]

\[ = \frac{\nabla_\eta \int h(x) \exp(\eta^\top t(x)) \, dx}{\int h(x) \exp(\eta^\top t(x)) \, dx} \]

\[ = \int h(x) \nabla_\eta \exp(\eta^\top t(x)) \, dx \]

\[ = \frac{\int h(x) t(x) \exp(\eta^\top t(x)) \, dx}{\exp(\log(\int h(x) \exp(\eta^\top t(x)) \, dx))} \]

\[ = \frac{\int h(x) t(x) \exp(\eta^\top t(x)) \, dx}{\exp(a(\eta))} \]
Important property of exponential family

\[ \nabla a(\eta) = \nabla \log \int h(x) \exp(\eta^\top t(x)) dx \]

\[ = \frac{\nabla \eta \int h(x) \exp(\eta^\top t(x)) dx}{\int h(x) \exp(\eta^\top t(x)) dx} \]

\[ = \frac{\int h(x) \nabla \eta \exp(\eta^\top t(x)) dx}{\int h(x) \exp(\eta^\top t(x)) dx} \]

\[ = \frac{\int h(x) t(x) \exp(\eta^\top t(x)) dx}{\exp(\log(\int h(x) \exp(\eta^\top t(x)) dx))} \]

\[ = \frac{\int h(x) t(x) \exp(\eta^\top t(x)) dx}{\exp(a(\eta))} \]

\[ = \int h(x) t(x) \exp(\eta^\top t(x) - a(\eta)) dx \]
Important property of exponential family

\[ \nabla a(\eta) = \nabla \log \int h(x) \exp(\eta^\top t(x))dx \]

\[ = \frac{\nabla_{\eta} \int h(x) \exp(\eta^\top t(x))dx}{\int h(x) \exp(\eta^\top t(x))dx} \]

\[ = \frac{\int h(x) \nabla_{\eta} \exp(\eta^\top t(x))dx}{\int h(x) \exp(\eta^\top t(x))dx} \]

\[ = \frac{\int h(x) t(x) \exp(\eta^\top t(x))dx}{\exp(\log(\int h(x) \exp(\eta^\top t(x))dx))} \]

\[ = \frac{\int h(x) t(x) \exp(\eta^\top t(x))dx}{\exp(a(\eta))} \]

\[ = \int h(x) t(x) \exp(\eta^\top t(x) - a(\eta))dx \]

\[ = \mathbb{E}[t(x)] \]
Build regression models with exponential families
Build regression models with exponential families

- Use exponential families for conditional distribution

\[ p(y_i \mid x_i) \sim \text{Expfam} \]

- Maximum likelihood to estimate parameters
More basic. How do you design conditional distributions?

\[ p(y | x) \sim \text{Normal}(\mu, \sigma) \]
More basic. How do you design conditional distributions?

\[ p(y \mid x) \sim \text{Normal}(\mu, \sigma) \]

Mean and variance functions of the conditioning variable

\[
\begin{align*}
\mu(x_i) &= f_\theta(x_i) \\
\sigma(x_i) &= \log(1 + \exp(g_\theta(x_i)))
\end{align*}
\]
Generalized Linear Models

\[ p(y_i | x_i) \sim \text{Expfam} \]
Generalized Linear Models

\[ p(y_i | x_i) \sim \text{Expfam} \]

The parameter is the natural parameter. One option

\[ \eta_i = \theta^\top x_i \]
Generalized Linear Models

\[ p(y_i \mid x_i) \sim \text{Expfam} \]

The parameter is the natural parameter. One option

\[ \eta_i = \theta^\top x_i \]

Learn parameters by maximizing likelihood

\[ \mathcal{L}(\theta) = \sum_i \log(h(y_i)) \exp(\eta_i^\top t(y_i) - a(\eta_i))) \]
Generalized Linear Models

\[ p(y_i | x_i) \sim \text{Expfam} \]

The parameter is the natural parameter. One option

\[ \eta_i = \theta^\top x_i \]

Learn parameters by maximizing likelihood

\[ \mathcal{L}(\theta) = \sum_i \log(h(y_i) \exp(\eta_i^\top t(y_i) - a(\eta_i))) \]

Parameters can be generalized with link functions
What happens with a Normal with fixed variance $\sigma = 1$?
What happens with a Normal with fixed variance $\sigma = 1$?

$$p(x) = \frac{1}{\sqrt{2\pi}} \exp\left(\frac{-(x - \mu)^2}{2}\right)$$

- $h_\sigma$
  $$\frac{1}{\sqrt{2\pi}} \exp\left(\frac{-x^2}{2}\right)$$
- $t(x) = x$
- $a(\mu) = \frac{\mu^2}{2}$
- $\eta = \mu$
\[ p(x) = \frac{1}{\sqrt{2\pi}} \exp\left(\frac{-(x - \mu)^2}{2}\right) \]

- \( \eta = \mu \)

Use \( \eta_i = \theta^\top x_i \). What is this?
\[ p(x) = \frac{1}{\sqrt{2\pi}} \exp\left(\frac{-(x - \mu)^2}{2}\right) \]

- \( \eta = \mu \)

Use \( \eta_i = \theta^\top x_i \). What is this?

Generalized linear model with Normal is linear regression
What happens with a Bernoulli?
What happens with a Bernoullii?

\[ p(x) = \exp(\eta x - \log(1 + \exp(\eta))) \]
What happens with a Bernoulli?

Set $\eta_i = \theta^\top x_i$

$$p(y_i | x_i) = \exp(\theta^\top x_i y_i - \log(1 + \exp(\theta^\top x_i)))$$
What happens with a Bernoulli?

Set $\eta_i = \theta^\top x_i$

$$p(y_i \mid x_i) = \exp(\theta^\top x_i y_i - \log(1 + \exp(\theta^\top x_i)))$$

Compute $\mathbb{E}[y_i \mid x_i] = p(y \mid x_i)$

$$\nabla_\eta \log(1 + \exp(\theta^\top x_i)) = \sigma(\theta^\top x_i)$$

*It’s logistic regression*
Regularization

Learn parameters by maximizing likelihood

\[ \mathcal{L}(\theta) = \sum_i \log(h(y_i) \exp(\eta_i^\top t(y_i) - a(\eta_i))) \]

- Can regularize generalized linear models
- Use $L2$, $L1$, or elastic net
Complex models can be regularized by parameter size
L1 regularization finds sparse solutions
Logistic regression is for binary outcomes
Linear and logistic regression are examples of GLMs
GLMs plus regularization allow regression on all kinds of data