LINEAR REGRESSION

Buckle up...

Data Matrix

N the number of samples

p the number of features

$$\boldsymbol{X} \in \mathbb{R}^{N \times p}$$

$$\boldsymbol{X} = \begin{pmatrix} x_{00} & \cdots & x_{0p} \\ \vdots & \ddots & \vdots \\ x_{N0} & \cdots & x_{Np} \end{pmatrix}$$

Linear Model

We want to find some linear function which takes a single observation

$$X^T = (X_0, X_1, \dots, X_p)$$

and makes a prediction using a function of the form

$$f(X) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p$$

$$f(X) = \beta_0 + \sum_{j=1}^{p} X_j \beta_j$$

Linear Model

This can be written as a vector product:

$$f(X) = \beta_0 + \sum_{j=1}^{p} X_j \beta_j$$

is the same as

$$f(X) = \beta_0 + X^T \cdot \beta$$

Linear Regression

We wish to minimize the residual sum of squares:

$$RSS = \sum_{i=1}^{N} \left(y_i - f(x_i)\right)^2$$

$$RSS = \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} X_j \beta_j \right)^2$$

Linear Regression

Why do we choose to minimize this?

1. This is a reasonable measure of distance:
    - The quantity $\left(y_i - f(x_i)\right)^2$ is always positive, as required for a metric
    - The distance from the set of true observations might as well be the sum of all the distances (ie. the distance from each prediction to the true observation, summed)
2. If we assume that our experimental errors are Gaussian then minimizing the RSS is equivalent to finding the most likely model under this assumption.
    - We will discuss this more later

Linear Regression

How to find the least-squares solution?

Write the RSS as a matrix-vector equation:

$$RSS = \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2$$

$$RSS = (\boldsymbol{y} - \boldsymbol{X}\beta)^T (\boldsymbol{y} - \boldsymbol{X}\beta)$$

Linear Regression

To minimize RSS, take the usual approach of minimization:

$$\frac{\partial RSS}{\partial \beta} = \frac{\partial\{(\boldsymbol{y} - \boldsymbol{X}\beta)^T(\boldsymbol{y} - \boldsymbol{X}\beta)\}}{\partial \beta}$$

$$0 = -2\,\boldsymbol{X}^T(\boldsymbol{y} - \boldsymbol{X}\beta)$$

$$0 = \boldsymbol{X}^T\boldsymbol{y} - \boldsymbol{X}^T\boldsymbol{X}\beta$$

$$\boldsymbol{X}^T\boldsymbol{X}\beta = \boldsymbol{X}^T\boldsymbol{y}$$

$$\hat{\beta} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{y}$$

$$\hat{\boldsymbol{y}} = \boldsymbol{X}\hat{\beta} = \boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{y}$$

Some Matrix Rules

$$(\mathbf{AB})^T = \mathbf{B}^T\mathbf{A}^T$$

$$\frac{\partial \mathbf{x}^T\mathbf{a}}{\partial \mathbf{x}} = \frac{\partial \mathbf{a}^T\mathbf{x}}{\partial \mathbf{x}} = \mathbf{a}$$

$$\frac{\partial \mathbf{a}^T\mathbf{X}\mathbf{b}}{\partial \mathbf{X}} = \mathbf{a}\mathbf{b}^T$$

$$\frac{\partial \mathbf{a}^T\mathbf{X}^T\mathbf{b}}{\partial \mathbf{X}} = \mathbf{b}\mathbf{a}^T$$

$$\frac{\partial \mathbf{a}^T\mathbf{X}\mathbf{a}}{\partial \mathbf{X}} = \frac{\partial \mathbf{a}^T\mathbf{X}^T\mathbf{a}}{\partial \mathbf{X}} = \mathbf{a}\mathbf{a}^T$$

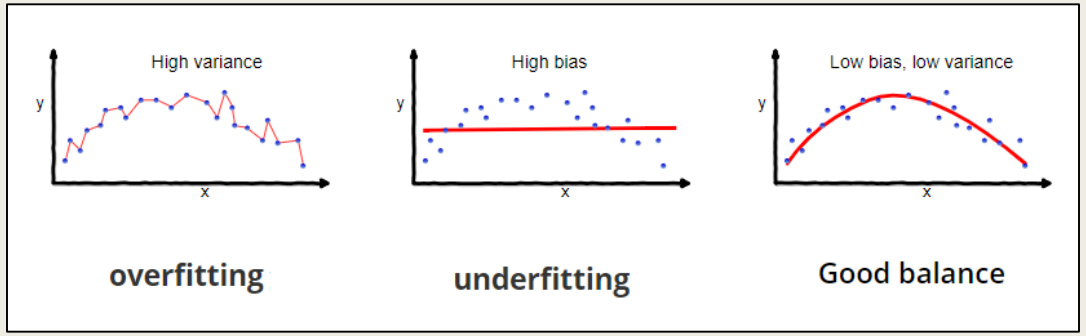
$$\frac{\partial \mathbf{b}^T\mathbf{X}^T\mathbf{X}\mathbf{c}}{\partial \mathbf{X}} = \mathbf{X}(\mathbf{b}\mathbf{c}^T + \mathbf{c}\mathbf{b}^T)$$

$$\frac{\partial}{\partial \mathbf{A}}(\mathbf{x} - \mathbf{A}\mathbf{s})^T\mathbf{W}(\mathbf{x} - \mathbf{A}\mathbf{s}) = -2\mathbf{W}(\mathbf{x} - \mathbf{A}\mathbf{s})\mathbf{s}^T$$

Linear Regression

Comments on least squares:

- The least squares solution is unique only if $\boldsymbol{X}$ is of full rank (all columns linearly independent)
- Gauss-Markov theorem: the least squares estimate is the minimum variance estimate amongst all unbiased linear estimates



- Least squares solutions often have high variance but low bias
- It can be difficult to interpret which terms in the model are most important

Ridge Regression

- When there are many correlated variables in a linear regression problem then the coefficients can be highly variable
  - The reason for this is that huge coefficients are balanced by equally small coefficients on correlated variables
- Solution: add a penalty for large coefficients

$$\hat{\beta}_{ridge} = argmin_{\beta} \left\{ RSS + \lambda \sum_{j=1}^{p} \beta_j^2 \right\}$$

Ridge Regression

- Now the criterion for the solutions is to minimize $\Omega(\lambda)$:

$$\Omega(\lambda) = (\boldsymbol{y} - \boldsymbol{X}\beta)^T(\boldsymbol{y} - \boldsymbol{X}\beta) + \lambda\beta^T\beta$$

$$\frac{\partial\Omega}{\partial\beta} = \frac{\partial\{(\boldsymbol{y} - \boldsymbol{X}\beta)^T(\boldsymbol{y} - \boldsymbol{X}\beta)\}}{\partial\beta} + \frac{\partial(\lambda\beta^T\beta)}{\partial\beta}$$

$$0 = -2\,\boldsymbol{X}^T(\boldsymbol{y} - \boldsymbol{X}\beta) + 2\lambda\beta$$

$$(\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I})\beta = \boldsymbol{X}^T\boldsymbol{y}$$

$$\hat{\beta} = (\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I})^{-1}\boldsymbol{X}^T\boldsymbol{y}$$

$$\hat{\boldsymbol{y}} = \boldsymbol{X}\hat{\beta} = \boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I})^{-1}\boldsymbol{X}^T\boldsymbol{y}$$

Ridge Regression

- Another way to understand why this works is to looks at the SVD of the solution:

Recall that the SVD of a matrix is:

$$\boldsymbol{X} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T$$

with the properties that $\boldsymbol{D}$ is a diagonal matrix and $\boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{I}, \boldsymbol{V}\boldsymbol{V}^T = \boldsymbol{I}$

$$\hat{\boldsymbol{y}} = \boldsymbol{X}\hat{\beta} = \boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I})^{-1}\boldsymbol{X}^T\boldsymbol{y}$$

$$\boldsymbol{X}\hat{\beta} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T(\boldsymbol{V}\boldsymbol{D}^T\boldsymbol{U}^T\boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T + \lambda\boldsymbol{I})^{-1}\boldsymbol{V}\boldsymbol{D}^T\boldsymbol{U}^T\boldsymbol{y}$$

$$\boldsymbol{X}\hat{\beta} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T(\boldsymbol{D}^2 + \lambda\boldsymbol{I})^{-1}\boldsymbol{V}\boldsymbol{D}^T\boldsymbol{U}^T\boldsymbol{y}$$

$$\boldsymbol{X}\hat{\beta} = \boldsymbol{U}\boldsymbol{D}(\boldsymbol{D}^2 + \lambda\boldsymbol{I})^{-1}\boldsymbol{V}^T\boldsymbol{V}\boldsymbol{D}^T\boldsymbol{U}^T\boldsymbol{y}$$

$$\boldsymbol{X}\hat{\beta} = \boldsymbol{U}\boldsymbol{D}(\boldsymbol{D}^2 + \lambda\boldsymbol{I})^{-1}\boldsymbol{D}\boldsymbol{U}^T\boldsymbol{y}$$

$$\boldsymbol{X}\hat{\beta} = \sum_{j=1}^{p} \boldsymbol{u}_j \frac{d_j^2}{d_j^2 + \lambda} \boldsymbol{u}_j^T\boldsymbol{y}$$

Ridge Regression

- Compare this to the least squares decomposition:

Ridge regression $\boldsymbol{X}\hat{\beta}_{ridge} = \sum_{j=1}^{p} \boldsymbol{u}_j \frac{d_j^2}{d_j^2+\lambda} \boldsymbol{u}_j^T \boldsymbol{y}$

Least squares regression $\boldsymbol{X}\hat{\beta}_{LS} = \boldsymbol{U}\boldsymbol{U}^T \boldsymbol{y} = \sum_{j=1}^{p} \boldsymbol{u}_j \boldsymbol{u}_j^T \boldsymbol{y}$

- So we see that ridge regression shrinks the coefficients in the basis of $\boldsymbol{U}$
- Recall from PCA theory that this basis is related to components of maximum variance, and it appears λ shrinks components with small variance fastest
- Ridge regression is also known as $\mathbb{L}_2$ - regularization, Tikhonov regularization, or weight-decay in training neural nets

The Lasso

- Also known as $\mathbb{L}_1$ – regularization or basis pursuit
- The goal is the same as in ridge regression, but the penalty is different. Now we write the objective as

$$\hat{\beta}_{ridge} = argmin_{\beta} \left\{ \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij} \beta_j \right)^2 \right\}, \text{ subject to } \sum_{j=1}^{p} |\beta_j| \leq t$$

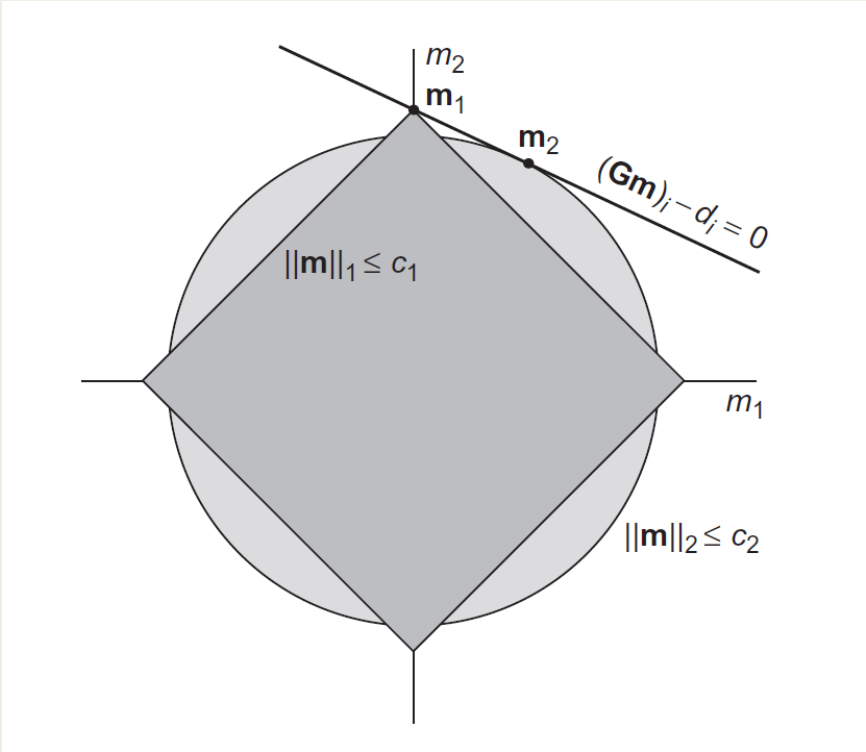
This can also be written in *Lagrangian form*:

$$\hat{\beta}_{ridge} = argmin_{\beta} \left\{ \frac{1}{2} RSS + \lambda \sum_{j=1}^{p} |\beta_j| \right\}$$

The Lasso

- The Lasso is non-linear in $\boldsymbol{y}$ so there is no closed-form solution. The optimal choice of β is found by iterative optimization procedures
- The advantage of the Lasso is that some coefficients can actually be set to 0 by the optimization procedure (as opposed to ridge regression, which only sends them to small values). This gives a clear interpretation of which terms are important.

Visualization of $\mathbb{L}_2$ vs $\mathbb{L}_1$ regularization



Regression using derived inputs

There are some methods of pre-processing our data which can achieve similar effects in reducing the model terms:

- PCA: we can find a regression along principle components instead of the original axes the data was given in.

  Compute the SVD $\boldsymbol{X} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T$ and define $\boldsymbol{Z} = \boldsymbol{U}\tilde{\boldsymbol{D}}$ for $\tilde{\boldsymbol{D}} \in \mathbb{R}^{N\times M}$, $M < p$.

  Now find the least squares solution β for $\boldsymbol{y} = \boldsymbol{Z}\beta^T$

- Partial least squares (PLS): similar to PCA in that we construct new components and perform the regression in a truncated version of this space.

  Compute $\phi_{1j} = \langle \boldsymbol{x}_j, \boldsymbol{y} \rangle$ for each $j \in \{1, \dots p\}$

  Construct the derived input $\boldsymbol{z}_1 = \sum_j \phi_{1j}\boldsymbol{x}_j$

  Compute the coefficient $\hat{\theta}_1 = \langle \boldsymbol{z}_1, \boldsymbol{y} \rangle / \langle \boldsymbol{z}_1, \boldsymbol{z}_1 \rangle$

  Orthogonalize the remaining $\boldsymbol{x}_{j+1}, \boldsymbol{x}_{j+2}$, … with respect to $\boldsymbol{z}_1$

  Repeat the above procedure for all $\phi_{m,j}$ and $\boldsymbol{x}_j$ that are left

Linear Methods of Classification

Linear Discriminant Analysis

Suppose we have a set of classes $G = \{1,2, \dots k, \dots K\}$ and we wish to compute the probability that observation $X = x$ is in class $G = k$. This can be written using Bayes theorem as:

$$P(G = k|X = x) = \frac{P_k(x)P_G(k)}{\sum_{j=1}^{K} P_j(x)P_G(j)}$$

We assert that all classes are distributed as multivariate Gaussians of the form

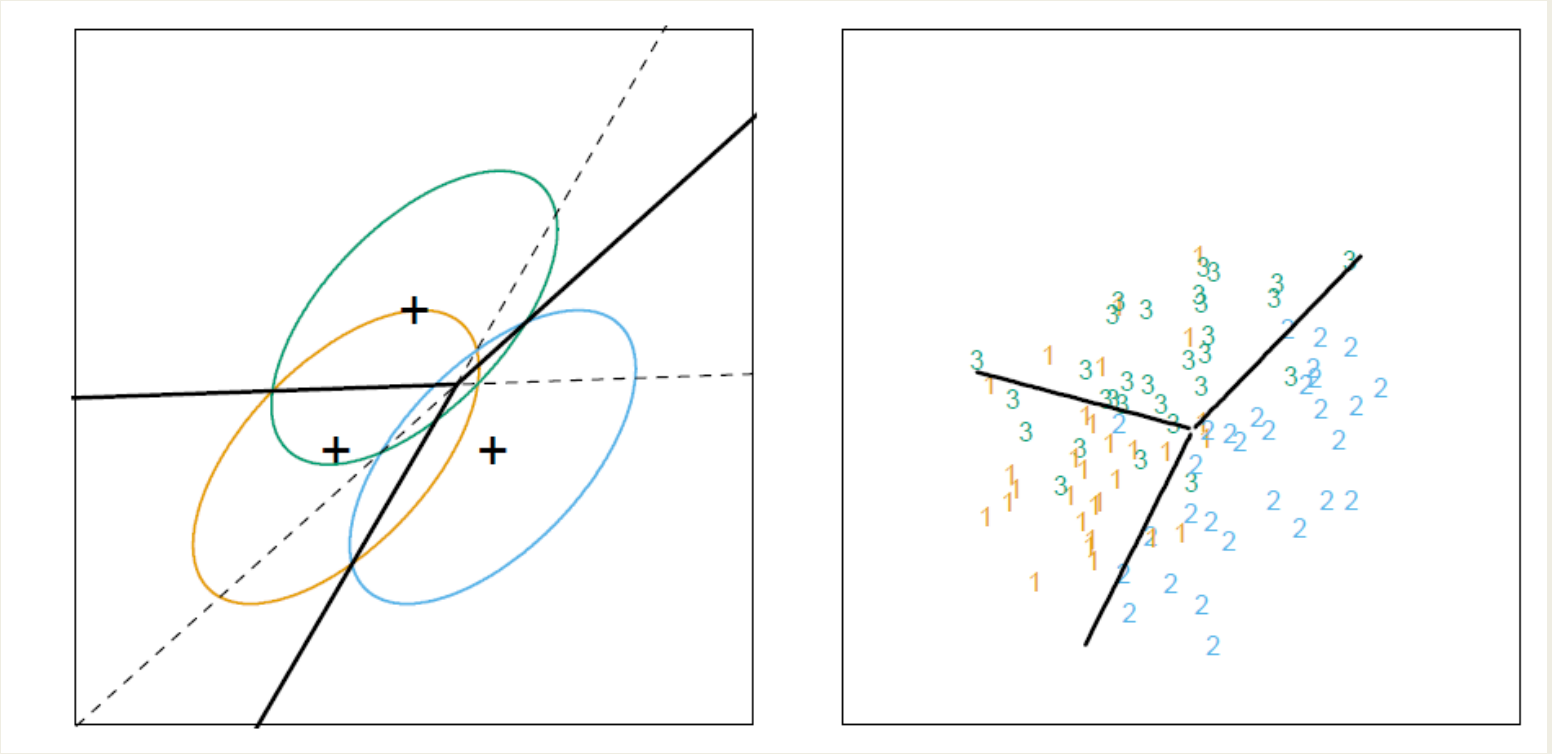
$$P_j(x) = \frac{1}{(2\pi)^{p/2}|\mathbf{\Sigma}_\mathrm{k}|^{1/2}} e^{-\frac{1}{2}(x-\mu_k)^T \mathbf{\Sigma}_\mathrm{k}^{-1}(\mathrm{x}-\mu_k)}$$

If we assume $\mathbf{\Sigma}_\mathrm{k}$ are equal for all *k* then we can compare the log-ratio of probabilities:

$$\log\frac{P(G = k|X = x)}{P(G = l|X = x)} = \log\frac{P_G(k)}{P_G(l)} + \log\frac{P_k(x)}{P_l(x)}$$

Linear Methods of Classification

Linear Discriminant Analysis



Logistic Regression

- We wish to model posterior log-probabilities of classes using linear functions in *x*, subject to the constraint that these functions sum to 1 and remain in [0,1]

$$\log \frac{\Pr(G=1|X=x)}{\Pr(G=K|X=x)} = \beta_{10} + \beta_1^T x$$
$$\log \frac{\Pr(G=2|X=x)}{\Pr(G=K|X=x)} = \beta_{20} + \beta_2^T x$$
$$\vdots$$
$$\log \frac{\Pr(G=K-1|X=x)}{\Pr(G=K|X=x)} = \beta_{(K-1)0} + \beta_{K-1}^T x.$$

Logistic Regression

- The consequence of this formulation is that

$$\Pr(G = k|X = x) = \frac{\exp(\beta_{k0} + \beta_k^T x)}{1 + \sum_{\ell=1}^{K-1} \exp(\beta_{\ell 0} + \beta_\ell^T x)},$$

$$\Pr(G = K|X = x) = \frac{1}{1 + \sum_{\ell=1}^{K-1} \exp(\beta_{\ell 0} + \beta_\ell^T x)},$$

Quick Derivation for 2 classes:

$$\log \frac{P(G = 1|X = x)}{P(G = 2|X = x)} = \beta_{10} + \beta_{11} x$$

$$e^{\beta_{10} + \beta_{11} x} = \frac{P(G = 1|X = x)}{P(G = 2|X = x)} = \frac{P(G = 1|X = x)}{1 - P(G = 1|X = x)}$$

$$P(G = 1|X = x) = \frac{e^{\beta_{10} + \beta_{11} x}}{1 + e^{\beta_{10} + \beta_{11} x}}$$

Logistic Regression

- Consider the two-class example. Use an indicator function to code the classes:

$$y_i = \begin{cases} 1 \text{ if } x_i \text{ is in class 1} \\ 0 \text{ if } x_i \text{ is in class 2} \end{cases}$$

We can write the log-likelihood of a set of N observations as a function of the parameters $\beta = \{\beta_{10},\ \beta_1\}$

$$\ell(\beta) = \sum_i \log p_k(x_i;\beta) = \sum_{i=1}^{N} \{y_i \log p(x_i;\beta) + (1-y_i)\log(1-p(x_i;\beta))\}$$

$$\ell(\beta) = \sum_{i=1}^{N} \{y_i\, \beta^T x_i - \log(1+e^{\beta^T x_i})\}$$

Logistic Regression

- Find the maximum likelihood estimate of β by taking the derivative

$$\frac{\partial \ell(\beta)}{\partial \beta} = 0 = \sum_{i=1}^{N} x_i(y_i - p(x_i;\beta))$$

- Typically this set of p equations for β is solved using the Newton-Raphson method

$$\beta^{new} = \beta^{old} - \left(\frac{\partial^2 \ell(\beta)}{\partial \beta \partial \beta^T}\right)^{-1} \frac{\partial \ell(\beta)}{\partial \beta}$$

If we define $\boldsymbol{W}$ a $N \times N$ diagonal matrix such that $w_{ii} = p(x_i;\beta^{old})(1 - p(x_i;\beta^{old}))$ then the Newton-Raphson step becomes

$$\beta^{new} = (\boldsymbol{X}^T\boldsymbol{W}\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{W}(\boldsymbol{X}\beta^{old} + \boldsymbol{W}^{-1}(\boldsymbol{y} - \boldsymbol{p}))$$

Note: this is also known as the iteratively reweighted least squares (IRLS) algorithm

That’s all!

- For more details, I recommend *The Elements of Statistical Learning* by Hastie, Tibshirani, and Friedman