Comparison with Gaussian Naïve Bayes

Gaussian Naïve Bayes vs. Logistic Regression

Generative $p(x|y)$ $p(y)$

Discriminative $p(y|x)$ =

Set of Gaussian Naïve Bayes parameters (feature variance independent of class label)

Set of Logistic Regression parameters

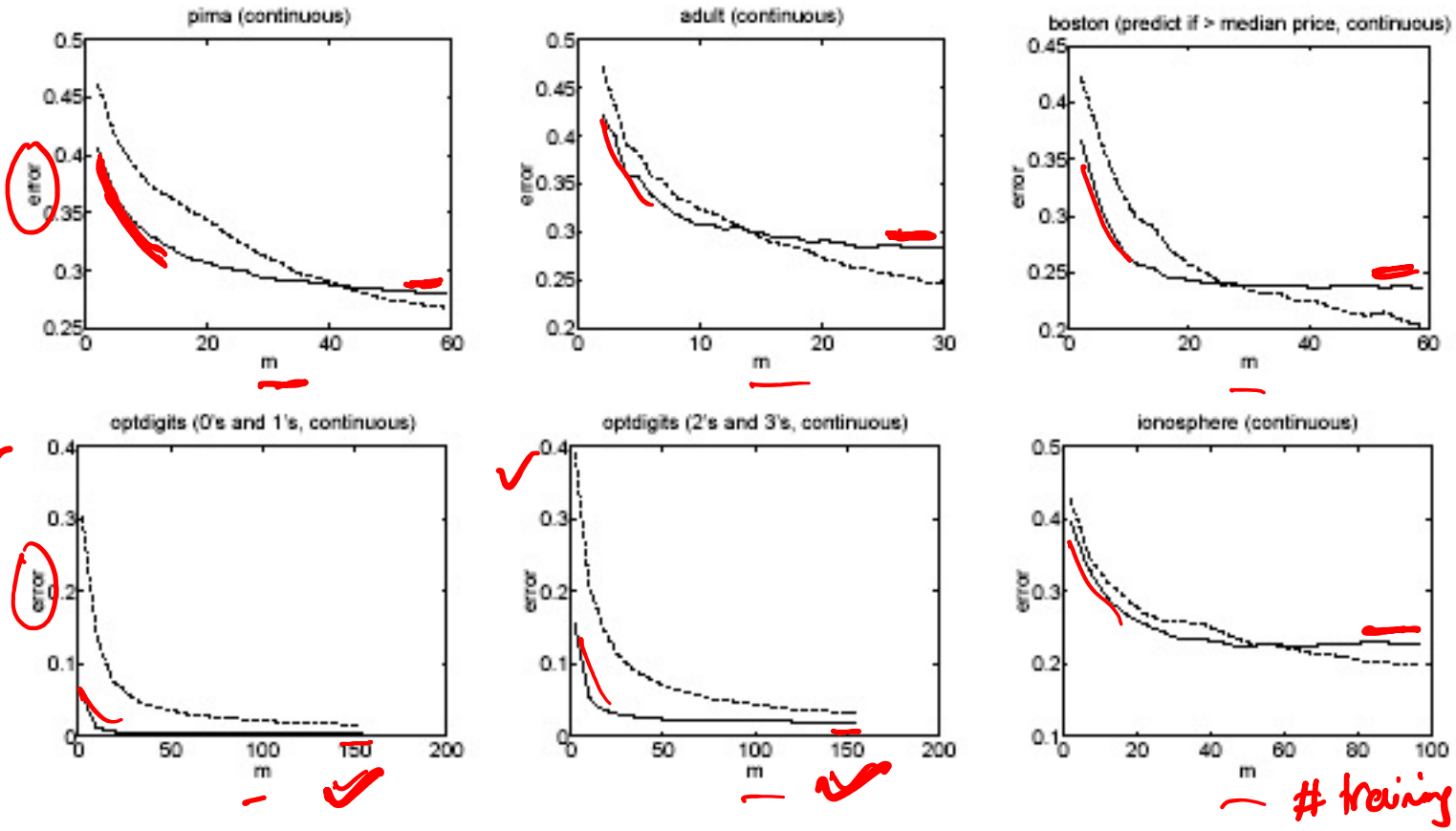
$w_0, w_1, \ldots w_d$ =

μ_y, $\Sigma_y = \begin{bmatrix} \sigma_1^{2(y)} & & 0 \\ & \ddots & \\ 0 & & \sigma_d^{2(y)} \end{bmatrix}$ → ind of y ← then linear decision boundary

- Representation equivalence (both yield linear decision boundaries)
  - **But only in a special case!!!** (GNB with class-independent variances)
  - **LR makes no assumptions about** P(**X**|Y) **in learning**!!!
  - **Optimize different functions (MLE/MCLE) or (MAP/MCAP)! Obtain different solutions**

Experimental Comparison (Ng-Jordan'01)

UCI Machine Learning Repository 15 datasets, 8 continuous features, 7 discrete features



More in Paper…

—— Naïve Bayes ----- Logistic Regression

Gaussian Naïve Bayes vs. Logistic Regression

- If conditional independence assumption holds, then GNB has lower large sample error
- If conditional independence assumption DOES NOT hold, then GNB has higher large sample error
  - But if converges faster (to its higher large sample error) as the parameter estimates are not coupled

What you should know

- LR is a linear classifier ✓
- LR optimized by maximizing conditional likelihood or conditional posterior
  - no closed-form solution
  - concave ! global optimum with gradient ascent
- Gaussian Naïve Bayes with class-independent variances representationally equivalent to LR
  - Solution differs because of objective (loss) function
- In general, NB and LR make different assumptions
  - NB: Features independent given class ! assumption on P(**X**|Y)
  - LR: Functional form of P(Y|**X**), no assumption on P(**X**|Y)
- Convergence rates
  - GNB (usually) needs less data
  - LR (usually) gets to better solutions in the limit

Linear Regression

Aarti Singh

Machine Learning 10-315
Sept 20, 2021

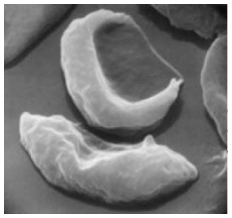
Supervised Learning Tasks

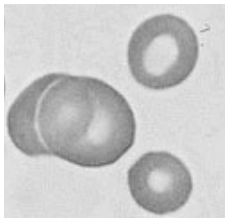
Classification

X = Document

Sports
Science
News

Y = Topic



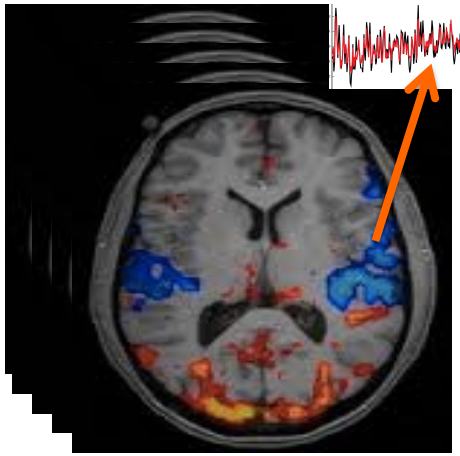


X = Cell Image

Anemic cell
Healthy cell

Y = Diagnosis

Regression

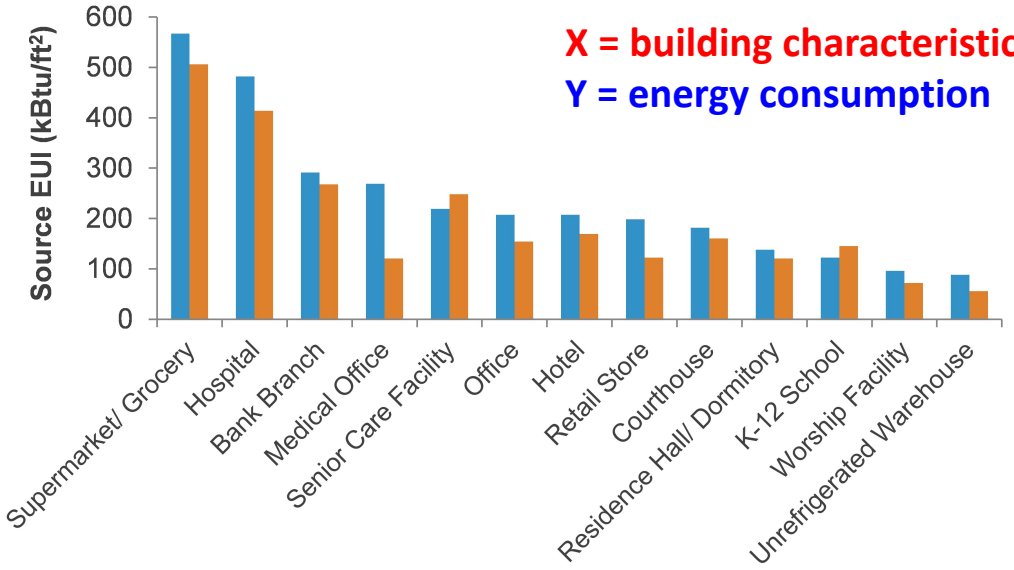


X = Brain Scan

Y = Age of a subject

Regression Tasks

Estimating Energy Usage



energystar.gov

Estimating Contamination

Performance Measures

0/1

$\text{loss}(Y, f(X))$ - Measure of closeness between true label *Y* and prediction *f*(*X*)

Don't just want label of one test data (e.g. cell image), but any cell image $X \in \mathcal{X}$

$$(X, Y) \sim P_{XY}$$

Given a cell image drawn randomly from the collection of all cell images, how well does the predictor perform on average?

$$\text{Risk } R(f) \equiv \mathbb{E}_{XY}\left[\text{loss}(Y, f(X))\right]$$

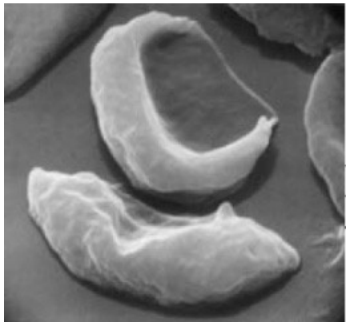
0/1

$E[1_{f(X) \neq Y}]$

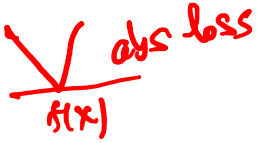
$= P(f(X) \neq Y)$

Performance Measures

Performance Measure: Risk $R(f) \equiv \mathbb{E}_{XY}\left[\text{loss}(Y, f(X))\right]$



	$\text{loss}(Y, f(X))$	Risk $R(f)$
"Anemic cell"	$1_{\{f(X) \neq Y\}}$	$P(f(X) \neq Y)$
	0/1 loss	Probability of Error
Share Price "$ 24.50"	$(f(X) - Y)^2$	$\mathbb{E}[(f(X) - Y)^2]$
	square loss	Mean Square Error
		MSE



Empirical Risk Minimization

Optimal predictor:

$$f^* = \arg\min_f \mathbb{E}[(f(X) - Y)^2]$$

$= \mathbb{E}_{XY}$

Empirical Minimizer:

$$\widehat{f}_n = \arg\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} (f(X_i) - Y_i)^2$$

Empirical mean

Law of Large Numbers:

$$\frac{1}{n} \sum_{i=1}^{n} [\text{loss}(Y_i, f(X_i))] \xrightarrow{n \to \infty} \mathbb{E}_{XY} [\text{loss}(Y, f(X))]$$

Restrict class of predictors

Optimal predictor: $$f^* = \arg\min_f \mathbb{E}[(f(X) - Y)^2]$$

Empirical Minimizer: $$\widehat{f}_n = \arg\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} (f(X_i) - Y_i)^2$$

Class of predictors

- Why?

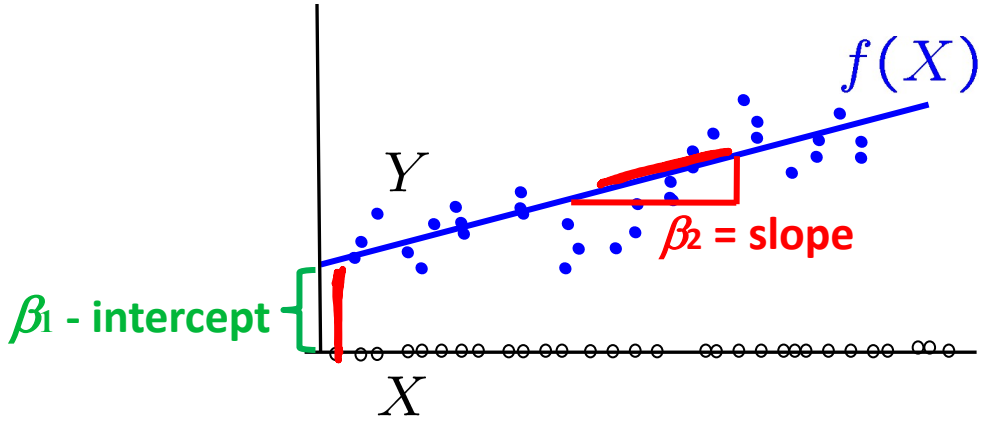
$\mathcal{F}$ -
- Class of Linear functions
- Class of Polynomial functions
- Class of nonlinear functions

Linear Regression

$$\hat{f}_n^L = \arg\min_{f \in \mathcal{F}_L} \frac{1}{n} \sum_{i=1}^{n} (f(X_i) - Y_i)^2$$

Least Squares Estimator

$\mathcal{F}_L$ - Class of Linear functions



Uni-variate case:

$$f(X) = \beta_1 + \beta_2 X$$

Multi-variate case:

p features

$$f(X) = f(X^{(1)}, \ldots, X^{(p)}) = \beta_1 X^{(1)} + \beta_2 X^{(2)} + \cdots + \beta_p X^{(p)}$$

$$= X\beta \qquad \text{where} \quad X = [X^{(1)} \ldots X^{(p)}], \quad \beta = [\beta_1 \ldots \beta_p]^T$$

Linear Regression (Matrix-vector form)

$$\hat{f}_n^L = \arg\min_{f \in \mathcal{F}_L} \frac{1}{n} \sum_{i=1}^{n} (f(X_i) - Y_i)^2 \qquad f(X_i) = X_i \beta$$

$$\hat{\beta} = \arg\min_{\beta} \frac{1}{n} \sum_{i=1}^{n} (X_i \beta - Y_i)^2 \qquad \hat{f}_n^L(X) = X\hat{\beta}$$

$$= \arg\min_{\beta} \frac{1}{n} (\mathbf{A}\beta - \mathbf{Y})^T (\mathbf{A}\beta - \mathbf{Y})$$

$$\mathbf{A} = \begin{bmatrix} X_1 \\ \vdots \\ X_n \end{bmatrix} = \begin{bmatrix} X_1^{(1)} & \dots & X_1^{(p)} \\ \vdots & \ddots & \vdots \\ X_n^{(1)} & \dots & X_n^{(p)} \end{bmatrix} \qquad \mathbf{Y} = \begin{bmatrix} \mathbf{Y}_1 \\ \vdots \\ \mathbf{Y}_n \end{bmatrix}$$

Linear Regression

$$\hat{\beta} = \arg\min_{\beta} \frac{1}{n}(\mathbf{A}\beta - \mathbf{Y})^T(\mathbf{A}\beta - \mathbf{Y}) = \arg\min_{\beta} J(\beta)$$

$$J(\beta) = (\mathbf{A}\beta - \mathbf{Y})^T(\mathbf{A}\beta - \mathbf{Y})$$

- Poll

Is the objective convex in β?

A) Convex, quadratic in β
B) Non-convex, A may not be positive semi definite
C) Depends on conditioning (ratio of max:min eigenvalues) of A^TA
D) Convex, A^TA is positive semi definite

Linear Regression

$$\beta = \begin{bmatrix} \beta_1 \\ \vdots \\ \beta_p \end{bmatrix}$$

$$\hat{\beta} = \arg\min_{\beta} \frac{1}{n}(\mathbf{A}\beta - \mathbf{Y})^T(\mathbf{A}\beta - \mathbf{Y}) = \arg\min_{\beta} J(\beta)$$

$y^T A = z^T$ Rule 1

$$J(\beta) = (\mathbf{A}\beta - \mathbf{Y})^T(\mathbf{A}\beta - \mathbf{Y})$$

$$= \beta^T A^T A \beta - Y^T A \beta - \beta^T A^T Y + Y^T Y$$

$$\frac{\partial J(\beta)}{\partial \beta}\bigg|_{\hat{\beta}} = 0$$

$$\frac{\partial J}{\partial \beta} = A^T A + A^T A \beta - 2A^T Y$$

$$= 2A^T A \beta - 2A^T Y$$

$$A^T A \hat{\beta} = A^T Y$$

$\beta^T z$

$\frac{\partial z^T \beta}{\partial \beta} = z$: Rule 1

$\frac{\partial \beta^T Q \beta}{\partial \beta} = (Q + Q^T)\beta$: Rule 2

$Q = I$

$\beta_{1\times 1}$

Linear regression solution satisfies Normal Equations

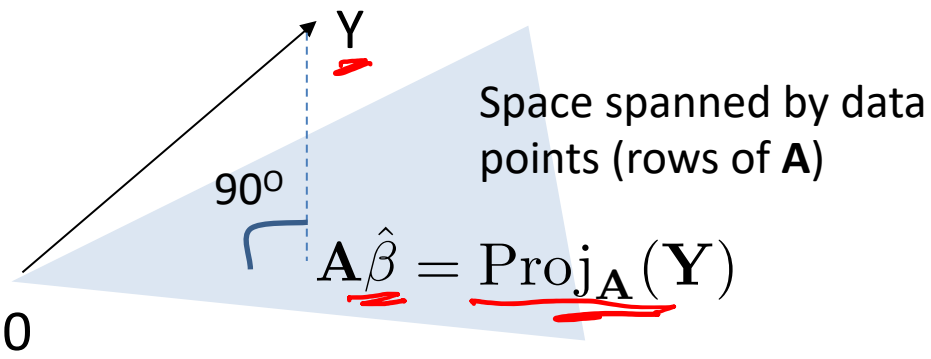
$$(\mathbf{A}^T\mathbf{A})\hat{\beta} = \mathbf{A}^T\mathbf{Y}$$

p x p p x1 p x1

If $(\mathbf{A}^T\mathbf{A})$ is invertible,

$$\hat{\beta} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{Y} \qquad \hat{f}_n^L(X) = X\hat{\beta}$$

Predicted labels for training points $\mathbf{A}\hat{\beta} = \mathrm{Proj}_{\mathbf{A}}(\mathbf{Y})$



Linear regression solution satisfies Normal Equations

$$(\mathbf{A}^T\mathbf{A})\hat{\beta} = \mathbf{A}^T\mathbf{Y}$$

p x p p x1 p x1

If $(\mathbf{A}^T\mathbf{A})$ is invertible,

$$\hat{\beta} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{Y} \qquad \hat{f}_n^L(X) = X\hat{\beta}$$

Later: When is $(\mathbf{A}^T\mathbf{A})$ invertible ?

Now: What if $(\mathbf{A}^T\mathbf{A})$ is invertible but expensive (p very large)?

Gradient Descent

Even when $(\mathbf{A}^T\mathbf{A})$ is invertible, might be computationally expensive if **A** is huge.

$$\widehat{\beta} = \arg\min_{\beta} \frac{1}{n}(\mathbf{A}\beta - \mathbf{Y})^T(\mathbf{A}\beta - \mathbf{Y}) = \arg\min_{\beta} J(\beta)$$

Since J(β) is convex, move along negative of gradient

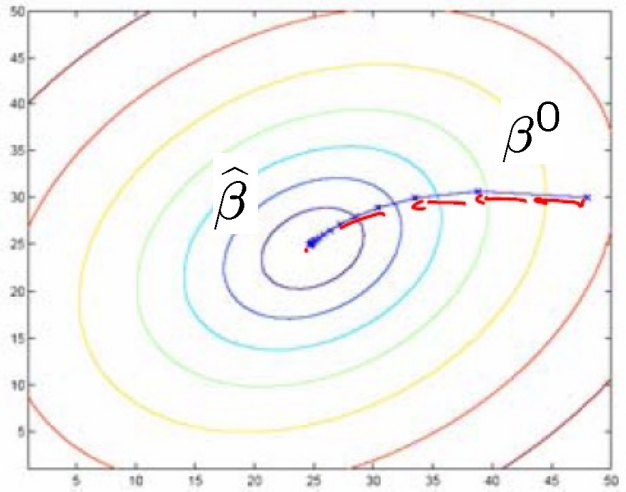
Initialize: β^0

Update: $\beta^{t+1} = \beta^t - \frac{\alpha}{2}\frac{\partial J(\beta)}{\partial \beta}\Big|_t$

step size

$$= \beta^t - \alpha\, \mathbf{A}^T(\mathbf{A}\beta^t - Y)$$

0 if $\widehat{\beta} = \beta^t$



Stop: when some criterion met e.g. fixed # iterations, or $\frac{\partial J(\beta)}{\partial \beta}\Big|_{\beta^t} < \varepsilon$.

Least Square solution satisfies Normal Equations

$$\left.\frac{\partial J(\beta)}{\partial \beta}\right|_{\widehat{\beta}} = 0 \qquad \text{gives} \qquad (\mathbf{A}^T\mathbf{A})\widehat{\beta} = \mathbf{A}^T\mathbf{Y}$$

p x p p x1 p x1

If $(\mathbf{A}^T\mathbf{A})$ is invertible,

1) If dimension p not too large, analytical solution:

$$\widehat{\beta} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{Y} \qquad \widehat{f}_n^L(X) = X\widehat{\beta}$$

2) If dimension p is large, computing inverse is expensive $O(p^3)$
Gradient descent since objective is convex ($\mathbf{A}^T\mathbf{A} \succeq 0$)

$$\begin{aligned}\beta^{t+1} &= \beta^t - \frac{\alpha}{2}\left.\frac{\partial J(\beta)}{\partial \beta}\right|_t \\ &= \beta^t - \alpha\ \mathbf{A}^T(\mathbf{A}\beta^t - Y)\end{aligned}$$