Elements of Statistical Learning 4: Linear Methods for Classification

Fabian Hainzl

appliedAI

1.8.2019

Notation

Notation in these slides follows Elements of Statistical Learning

$G(x)$	Predictor
$p_k(x)$	$Pr(G = k \| X = x)$
π_k	$Pr(G = k)$
$\hat{f}_k(x) = \hat{\beta}_{k0} + \hat{\beta}_k^T x$	Linear model for k-th output dimension
K	Number of classes
N	Number of samples

Classification

Two approaches to supervised learning:

- Regression: Continuous output variable
- Classification: Discrete output variable

Goal of classification

- Divide input space into a collection of regions with constant classification

Linear Regression

Classification

Two approaches to supervised learning:

- Regression: Continuous output variable
- Classification: Discrete output variable

Goal of classification

- Divide input space into a collection of regions with constant classification

Linear Regression

- Linear dependence of output on the weights

Linear Classification

Classification

Two approaches to supervised learning:

- Regression: Continuous output variable
- Classification: Discrete output variable

Goal of classification

- Divide input space into a collection of regions with constant classification

Linear Regression

- Linear dependence of output on the weights

Linear Classification

- Linear decision boundaries

Decision Boundary

Decision boundaries

- Boundaries between regions of different classes
- Points of input space where several classes have same probability

Definition of decision boundary:

$$\{x : (\hat{\beta}_{k0} - \hat{\beta}_{m0}) + (\hat{\beta}_k - \hat{\beta}_m)^T x = 0\}$$

Linear classification

Linear classification:

- Decision boundaries are hyperplanes

Linear classification

Linear classification:

- Decision boundaries are hyperplanes

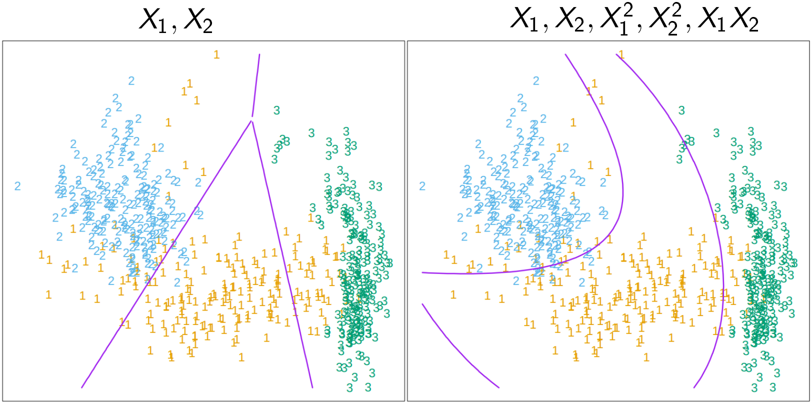
This is the case if

- posterior probability $Pr(G = k|X = x)$ is linear in x
- monotone transformation of posterior probability is linear

Generalization to non-linear decision boundaries

Augment feature space by adding squares and cross-products of features

Example for 2 dimensions:



Linear Regression for Classification

- Indicator matrix $\mathbf{Y} \in \mathbb{R}^{NxK}$ with one-hot encoded class targets in rows
- Closed-form solution for weight matrix $\mathbf{B} \in \mathbb{R}^{(p+1)xK}$:

$$\hat{\mathbf{B}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}$$

- Classification via

$$\hat{f}(x) = \hat{\mathbf{B}}[1, x^T]^T$$

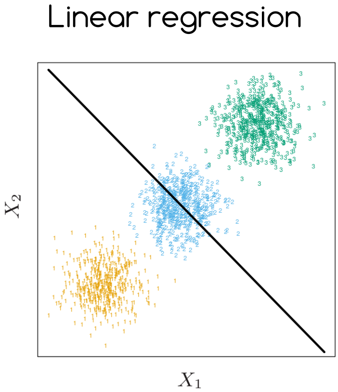
$$\hat{G}(x) = argmax_{k \in \mathcal{G}} \hat{f}_k(x)$$

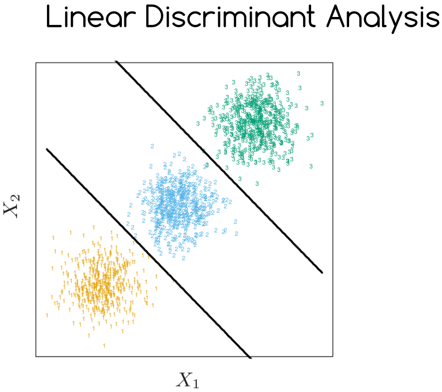
Problems with Linear Regression Approach

- Limited interpretability of $\hat{f}_k(x)$ as class probability, negative and greater 1 values possible

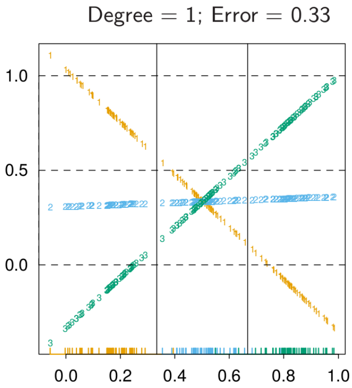
Problems with Linear Regression Approach

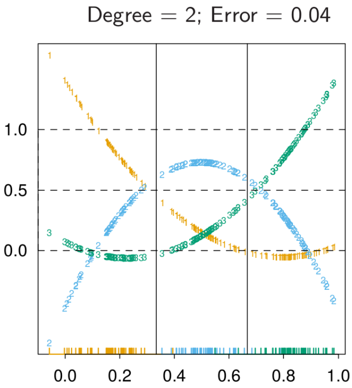
- Limited interpretability of $\hat{f}_k(x)$ as class probability, negative and greater 1 values possible
- Class masking effects





Effects of Masking





Linear Discriminant Analysis (LDA)

- Class posteriors $Pr(G|X)$ is needed for optimal classification
- With class-conditional density $f_k(x) = Pr(X = x|G = k)$ and prior $\pi_k = Pr(G = k)$,

Linear Discriminant Analysis (LDA)

- Class posteriors $Pr(G|X)$ is needed for optimal classification
- With class-conditional density $f_k(x) = Pr(X = x|G = k)$ and prior $\pi_k = Pr(G = k)$,

$$Pr(G = k|X = x) = \frac{f_k(x)\pi_k}{\sum_{j=1}^{K} f_j(x)\pi_j}$$

- LDA assumes mutliavriate Gaussians as class conditional densities with common covariance matrices $\Sigma_k = \Sigma \forall k \in K$

Estimate Parameters of Gaussian Distribution

$$f_k(x) = \frac{1}{(2\pi)^{p/2}|\Sigma_k|^{1/2}} e^{-\frac{1}{2}(x-\mu_k)^T \Sigma_k^{-1}(x-\mu_k)}$$

with

$$\hat{\pi}_k = \frac{N_k}{N}$$

$$\hat{\mu}_k = \sum_{g_i=k} \frac{x_i}{N_k}$$

$$\hat{\Sigma} = \sum_{k=1}^{K} \sum_{g_i=k} \frac{(x_i - \hat{\mu}_k)(x_i - \hat{\mu}_k)^T}{N - K}$$

Linear Discriminant Functions

- To compare two classes j and k, it sufficies to compare log ratio:

$$\log \frac{Pr(G=j|X=x)}{Pr(G=k|X=x)}$$

$$= \log \frac{\pi_j}{\pi_k} - \frac{1}{2}(\mu_j + \mu_k)^T \Sigma^{-1}(\mu_j - \mu_k) + x^T \Sigma^{-1}(\mu_k - \mu_l)$$

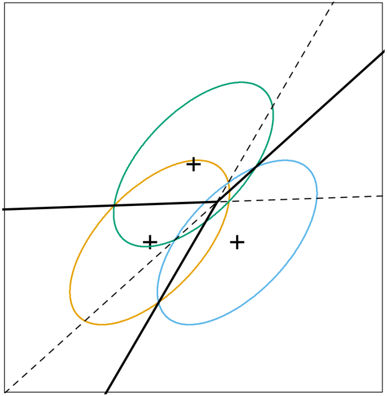
- Decision rule can be formulated as

$$G(x) = argmax_k \delta_k(x)$$

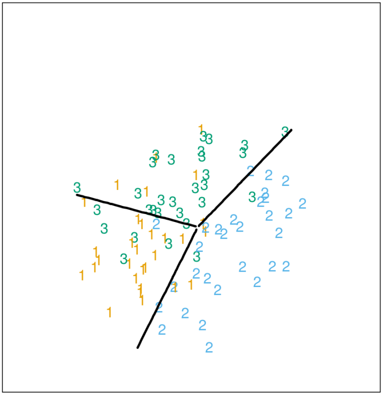
with

$$\delta_k(x) = \log \pi_k - \frac{1}{2}\mu_k^T \Sigma^{-1}\mu_k + x^T \Sigma^{-1}\mu_k$$

LDA Result

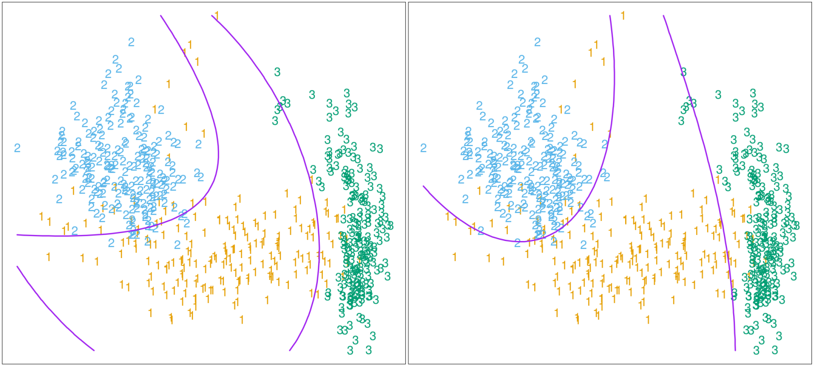


Bayes decision boundaries



LDA decision boundaries on 20 samples

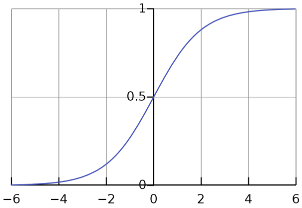
LDA vs QDA



LDA
with augmented feature
space

QDA ($\Sigma_j \neq \Sigma_k$)

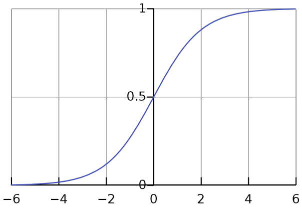
Logistic Regression



Squash network output $\mathbb{R} \rightarrow [0, 1]$ using

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

Logistic Regression



Squash network output $\mathbb{R} \to [0, 1]$ using

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

Network output is modeled as the log-odds:

$$\beta^T x = \frac{p_1}{p_0} = \frac{p_1}{1 - p_1}$$

Inverse of sigmoid function is called logit function.

Fitting Logistic Regression

- Log-likelihood for N observations:

$$L(\theta) = \sum_{i=1}^{N} \log p_{g_i}(x_i)$$

Fitting Logistic Regression

- Log-likelihood for N observations:

$$L(\theta) = \sum_{i=1}^{N} \log p_{g_i}(x_i)$$

For the binary case with the target $y_i \in \{0, 1\}$:

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

Fitting Logistic Regression

- Log-likelihood for N observations:

$$L(\theta) = \sum_{i=1}^{N} \log p_{g_i}(x_i)$$

For the binary case with the target $y_i \in \{0, 1\}$:

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

To maximize log-likelihood:

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

- Solved iteratively with IRLS (Iteratively reweighted least squares), a variant of the Newton-Raphson algorithm

Example: South African Hearth Disease

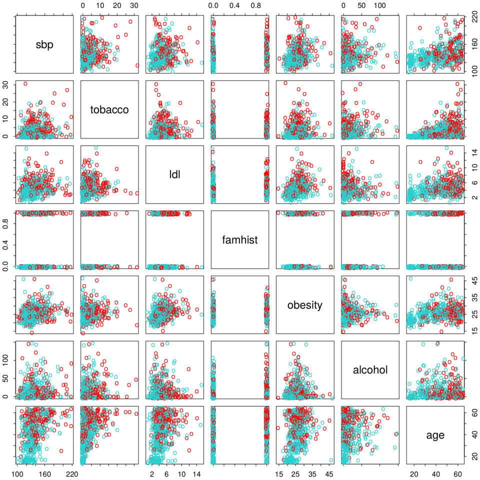
Model selection strategies:

1) Remove independent variable with least significant coefficient
2) Refit model with each variable removed, perform analysis of deviance

	Coefficient	Std. Error	Z Score
`(Intercept)`	−4.130	0.964	−4.285
`sbp`	0.006	0.006	1.023
`tobacco`	0.080	0.026	3.034
`ldl`	0.185	0.057	3.219
`famhist`	0.939	0.225	4.178
`obesity`	-0.035	0.029	−1.187
`alcohol`	0.001	0.004	0.136
`age`	0.043	0.010	4.184

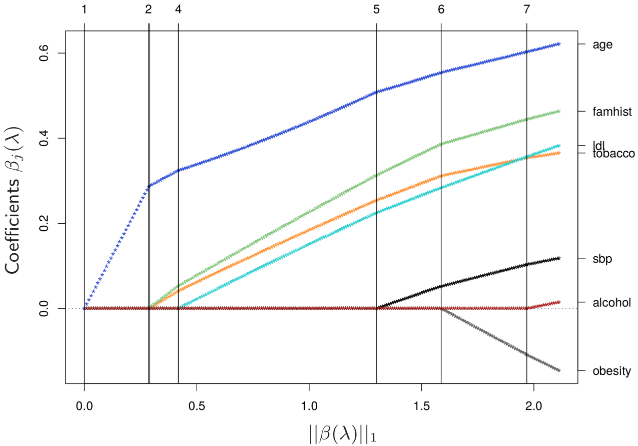
	Coefficient	Std. Error	Z score
`(Intercept)`	−4.204	0.498	−8.45
`tobacco`	0.081	0.026	3.16
`ldl`	0.168	0.054	3.09
`famhist`	0.924	0.223	4.14
`age`	0.044	0.010	4.52

Example: South African Hearth Disease



L_1 Regularized Logistic Regression

$$^{\lambda}L(\theta) = \sum_{i=1}^{N} y_i \beta^T x_i - log(1 + e^{\beta^T x_i}) - \lambda \sum_{j=1}^{p} |\beta_j|$$



Logistic Regression vs. LDA

- Mathematically identical models:
  - Log-odds are linear function of inputs in both cases
  - In LDA linearity results from Gaussian assumption, in Logistic Regression it is a modeling decision.
- Difference in fitting parameters:
  - By estimating Gaussian parameters, LDA maximizes full log-likelihood $Pr(X, G)$
  - Logistic Regression maximizes conditional likelihood $Pr(G|X)$
- Logistic Regression is 'generally felt' to be safer and more robust
- ESL claims: Similar results in practice, even if LDA assumptions are violated (e.g. qualitative predictors)