Machine Learning Applications for Finance

Classification

Hao Xing

Classification

- Qualitative variables takes values in an unordered set $\mathcal{C}$ such as

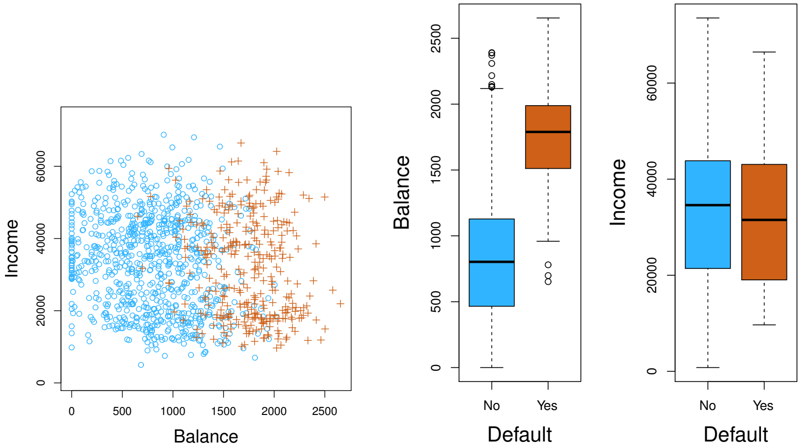
  credit card transaction $\in \{normal, fraudulent\}$

- Given a feature vector X and a qualitative response Y taking values in the set $\mathcal{C}$, the classification task is to build a function $C(X)$ and use it to predict Y

- Often we are more interested in estimating the probability that X belongs to each category in $\mathcal{C}$

  For example, it is more valuable to have an estimate the probability that a credit card transaction is fraudulent or not, than a classification fraudulent or not.

Example: Credit Card Default



Logistic regression

Let $Y = 1$ to indicate default

$$p(X) = Pr(Y = 1|X)$$

We want to use $X =$balance to predict default. Logistic regression uses the form

$$p(X) = \frac{e^{\beta_0 + \beta_1 X}}{1 + e^{\beta_0 + \beta_1 X}}.$$

No matter what values β_0, β_1 or X takes, $p(X) \in (0, 1)$

Rearrangement gives

$$\log\left(\frac{p(X)}{1 - p(X)}\right) = \beta_0 + \beta_1 X.$$

This monotone transformation is called the log odds or logit transformation of $p(X)$.

Maximum likelihood

We use maximum likelihood to estimate the parameters

$$\ell(\beta_0, \beta_1) = \prod_{i:y_i=1} p(x_i) \prod_{i:y_i=0} (1 - p(x_i)).$$

This likelihood gives the probability of the observed zeros and ones in the data. We pick β_0 and β_1 to maximize the likelihood of the observed data.

In R we use the glm function to fit linear regression models

	Coefficient	Std. Error	Z-statistic	P-value
Intercept	-10.6513	0.3612	-29.5	< 0.0001
balance	0.0055	0.0002	24.9	< 0.0001

Making predictions

What is our estimated probability of default for someone with a balance of $1000?

$$\hat{p}(X) = \frac{e^{\hat{\beta}_0+\hat{\beta}_1 X}}{1+e^{\hat{\beta}_0+\hat{\beta}_1 X}} = \frac{e^{-10.6513+0.0055\times 1000}}{1+e^{-10.6513+0.0055\times 1000}} = 0.006$$

With a balance of $2000?

$$\hat{p}(X) = \frac{e^{\hat{\beta}_0+\hat{\beta}_1 X}}{1+e^{\hat{\beta}_0+\hat{\beta}_1 X}} = \frac{e^{-10.6513+0.0055\times 2000}}{1+e^{-10.6513+0.0055\times 2000}} = 0.586$$

Let's do it again, using student as the predictor

	Coefficient	Std. Error	Z-statistic	P-value
Intercept	-3.5041	0.0707	-49.55	< 0.0001
student [Yes]	0.4049	0.1150	3.52	0.0004

$$\hat{Pr}(\text{default}|\text{student} = \text{yes}) = \frac{e^{-3.5041+0.4049\times 1}}{1 + e^{-3.5041+0.4049\times 1}} = 0.0431$$

$$\hat{Pr}(\text{default}|\text{student} = \text{no}) = \frac{e^{-3.5041+0.4049\times 0}}{1 + e^{-3.5041+0.4049\times 0}} = 0.0292$$

Logistic regression with several variables

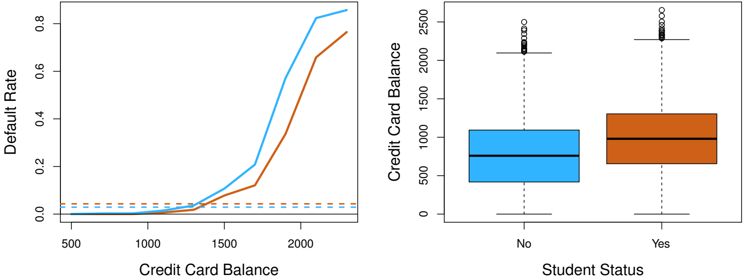
$$\log\left(\frac{p(X)}{1-p(X)}\right) = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$$

$$p(X) = \frac{e^{\beta_0+\beta_1 X_1+\cdots\beta_p X_p}}{1+e^{\beta_0+\beta_1 X_1+\cdots\beta_p X_p}}$$

	Coefficient	Std. Error	Z-statistic	P-value
Intercept	-10.8690	0.4923	-22.08	< 0.0001
balance	0.0057	0.0002	24.74	< 0.0001
income	0.0030	0.0082	0.37	0.7115
student [Yes]	-0.6468	0.2362	-2.74	0.0062

Why is coefficient for student negative, while it was positive before?

Confounding



- Students tend to have higher balances than non-students so their marginal default rate is higher than for non-students
- But for each level of balance, students default less than non-students
- Multiple logistic regression can tease this out

Logistic regression with more than two classes

It is easily generalized to more than two classes

One version (used in the R package glmnet) has the symmetric form

$$\Pr(Y = k|X) = \frac{e^{\beta_{0k}+\beta_{1k}X_1+\cdots\beta_{pk}X_p}}{\sum_{\ell=1}^{K} e^{\beta_{0\ell}+\beta_{1\ell}X_1+\cdots+\beta_{p\ell}X_p}}.$$

Here there is a linear function for each class

Multiclass logistic regression is also referred to as multinomial regression

Optimization in logistic regression

Let $h(x^{(i)}, \theta) = P(y = 1|x^{(i)}, \theta) = \frac{e^{-\theta^\top x^{(i)}}}{1+e^{-\theta^\top x^{(i)}}}$, where θ represents a vector of parameters. Then

$$P(y|x^{(i)}, \theta) = h(x^{(i)}, \theta)^{y^{(i)}}(1 - h(x^{(i)}, \theta))^{1-y^{(i)}}$$

The likelihood of observations $\{(x^{(i)}, y^{(i)}); i = 1, \dots, m\}$ is

$$L(\theta) = \prod_{i=1}^{m} h(x^{(i)}, \theta)^{y^{(i)}}(1 - h(x^{(i)}, \theta))^{1-y^{(i)}}.$$

Hence the (negative) average log-likelihood is

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m}\left[y^{(i)} \log h(x^{(i)}, \theta) + (1 - y^{(i)}) \log(1 - h(x^{(i)}, \theta))\right].$$

Gradient descent

For the parameter θ_j,

$$\text{Repeat} \quad \theta_j \leftarrow \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta) = \theta_j - \frac{\alpha}{m} \sum_{i=1}^{m} (h(x^{(i)}, \theta), y^{(i)}) x_j^{(i)}.$$

A vectorized implementation is

$$\theta \leftarrow \theta - \frac{\alpha}{m} X^\top (H(X, \theta) - Y).$$

Here α is the so called learning rate.

Bayes theorem

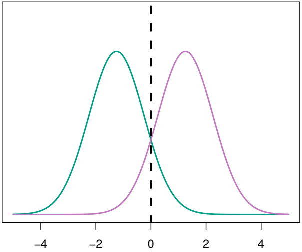
$$Pr(Y = k|X = x) = \frac{Pr(X = x|Y = k) \cdot Pr(Y = k)}{Pr(X = x)}$$

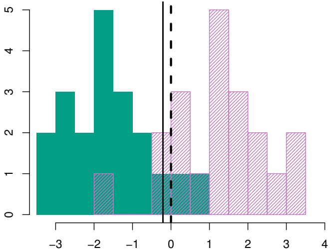
One writes this slightly differently for discriminant analysis

$$Pr(Y = k|X = x) = \frac{\pi_k f_k(x)}{\sum_{l=1}^{K} \pi_l f_l(x)}, \quad \text{where}$$

- $f_k(x) = Pr(X = x|Y = k)$ is the density for X in class k. Here we will use normal densities for each class
- $\pi_k = Pr(Y = k)$ is the marginal or prior probability for class k

Classify to the highest density





Example with $\mu_1 = -1.5$, $\mu_2 = 1.5$, $\pi_1 = \pi_2 = 0.5$ and $\sigma^2 = 1$.

The decision boundary is the dash line in the middle. The right is classified as pink; the left is classified as green.

Why discriminant analysis?

- When the classes are well-separated, the parameter estimated for the logistic regression model are surprising unstable. Linear discriminant analysis does not suffer this problem.
- If n (number of observations) is small and the distribution of the predictors X is approximately normal in each class, the linear discriminant model is again more stable than the logistic regression model
- Linear discriminant analysis is popular when there are more than two response classes

Linear discriminant analysis when $p = 1$

The Gaussian density:

$$f_k(x) = \frac{1}{\sqrt{2\pi}\sigma_k} e^{-\frac{1}{2}\left(\frac{x-\mu_k}{\sigma_k}\right)^2}$$

For linear discriminant analysis, we assume that all the $\sigma_k = \sigma$ are the same

Plugging into Bayes formula, $p_k(x) = Pr(Y = k|X = x)$ is

$$p_k(x) = \frac{\pi_k \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\left(\frac{x-\mu_k}{\sigma}\right)^2}}{\sum_{l=1^K} \pi_l \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\left(\frac{x-\mu_l}{\sigma}\right)^2}}$$

To classify at the value $X = x$, we need to see which $p_k(x)$ is the largest. This is equivalent to the largest discriminant score

$$\delta_k(x) = x\frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + \log(\pi_k),$$

which is a linear function of x

Estimating the parameters

Use the training date

$$\hat{\pi}_k = \frac{n_k}{n}$$

$$\hat{\mu}_k = \frac{1}{n_k} \sum_{i:y_i=k} x_i$$

$$\begin{aligned}\hat{\sigma}^2 =& \frac{1}{n-K} \sum_{k=1}^{K} \sum_{i:y_i=k} (x_i - \hat{\mu}_k)^2 \\ =& \sum_{k=1}^{K} \frac{n_k - 1}{n-K} \cdot \hat{\sigma}_k^2\end{aligned}$$

where $\hat{\sigma}_k^2 = \frac{1}{n_k-1} \sum_{i:y_i=k} (x_i - \hat{\mu}_k)^2$ is the usual formula for the estimated variance in the k-th class

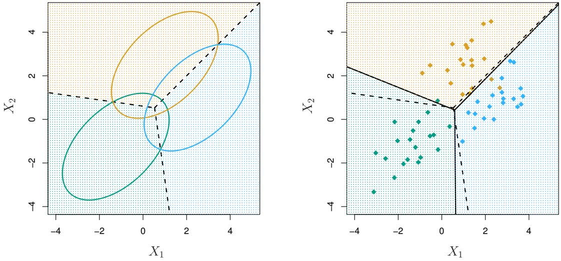
Linear Discriminant analysis when $p > 1$

$$\text{Density:} \quad f(x) = \frac{1}{(2\pi)^{p/2}} e^{-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)}$$

$$\text{Discriminant function: } \delta_k = x^T\Sigma^{-1}\mu_k - \frac{1}{2}\mu_k^T\Sigma^{-1}\mu_k + \log \pi_k$$

still a linear function in x

Example: $p = 2$, $K = 3$



The dashed lines are known as the Bayes decision boundaries (if we know the try density in each class). The solid lines are estimated from the data

From $\delta_k(x)$ to probabilities

Once we have estimated $\hat{\delta}_k(x)$, we can turn these into estimates for class probabilities

$$\widehat{Pr}(Y = k|X = x) = \frac{e^{\hat{\delta}_k(x)}}{\sum_{l=1}^{K} e^{\hat{\delta}_l(x)}}$$

So classifying to the largest $\hat{\delta}_k(x)$ amounts to classifying to the class for which $\widehat{Pr}(Y = k|X = x)$ is the largest

LDA on credit data

		True	Default	Status
		No	Yes	Total
Predicted	No	9644	252	9896
Default	Yes	23	81	104
Status	Total	9667	333	10000

$(23 + 252)/10000$ error - a 2.75% misclassification rate (not so bad!)

- This is training error, and we may be overfitting
- If we always classify as No, we would have make $333/10000$ error, or only 3.33%
- Of the true No's, we make $23/9667 = 0.2\%$ error, of the true Yes's, we make $252/333 = 75.7\%$ error!

Types of errors

False positive rate: The fraction of negative examples that classified as positive - 0.2% in example

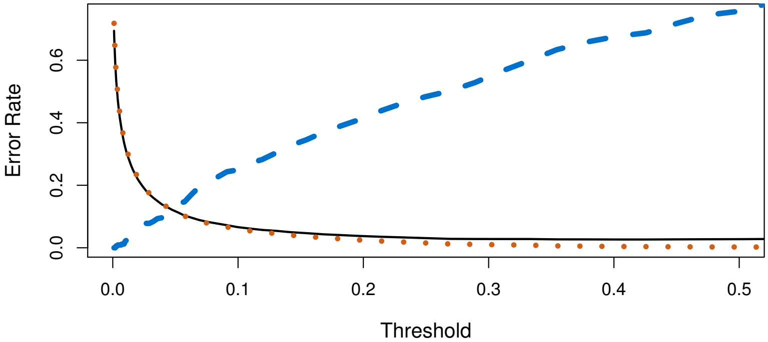
Flase negative rate: The fraction of positive examples that are classified as negative - 75.7% in example

We produced this table by classifying to class Yes if

$$\hat{Pr}(Default = Yes|Balance, Student) \geq \text{threshold}$$

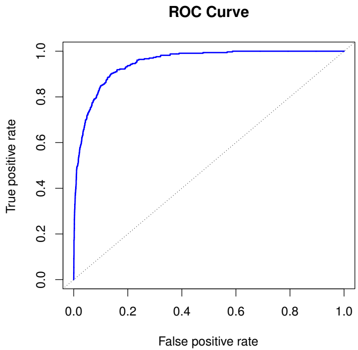
where the threshold is in $[0, 1]$ and we can vary threshold

Varying the threshold



In order to reduce the false negative rate, we may want to reduce the threshold to 0.1 or less.

ROC curve



The ROC plot displays both errors simultaneously

The diagonal is a random classification, 50-50 chances

Sometimes we use the AUC or area under the curve to summarize the overall performance. Higher AUC is good

Other forms of Discriminant Analysis

$$Pr(Y = k|X = x) = \frac{\pi_k f_k(x)}{\sum_{l=1}^{K} \pi_l f_l(x)}$$

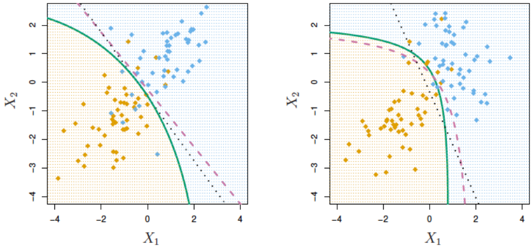
When $f_k(x)$ are Gussian densities, with the same covariance matrix Σ in each class, this leads to linear discriminant analysis. By changing the forms for $f_k(x)$, we get different classifiers

- With Gaussian but different Σ_k in each class, we get quadratic discriminant analysis
- With $f_k(x) = \prod_{j=1}^{p} f_{jk}(x_j)$ (conditional independence model) in each class we get naive Bayes. For Gaussian, this means the Σ_k are diagonal
- Many other forms, by proposing specific density models for $f_k(x)$, including nonparametric approaches

Quadratic Discriminant Analysis

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

because the Σ_k are different, the quadratic terms matter.



Naive Bayes

Assumes features are independent in each class
Useful when p is large, and so multivariate methods like QDA and even LDA break down.

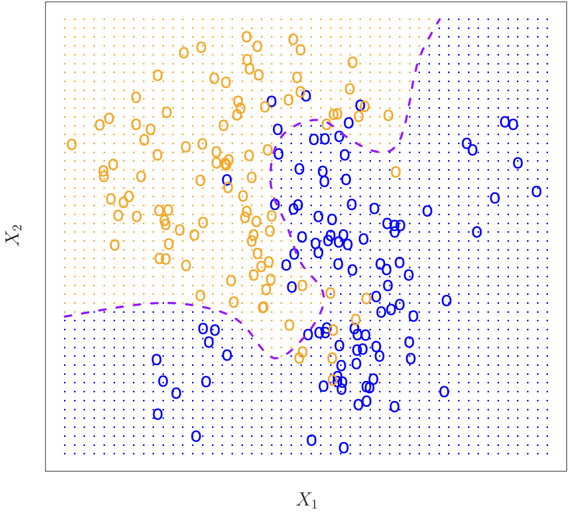
- Gaussian naive Bayes assumes each Σ_k is diagonal:

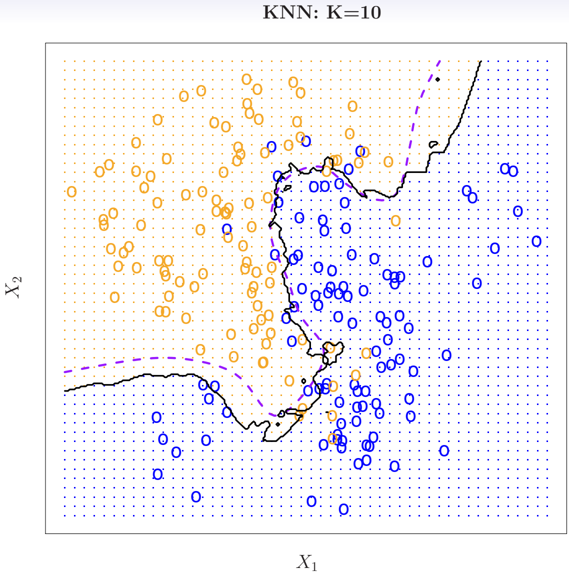
$$\delta_k(x) \propto \log \left[\pi_k \prod_{j=1}^{p} f_{kj}(x_j)\right] = -\frac{1}{2} \sum_{j=1}^{p} \frac{x_j - \mu_{kj}^2}{\sigma_{kj}^2} + log \pi_k$$

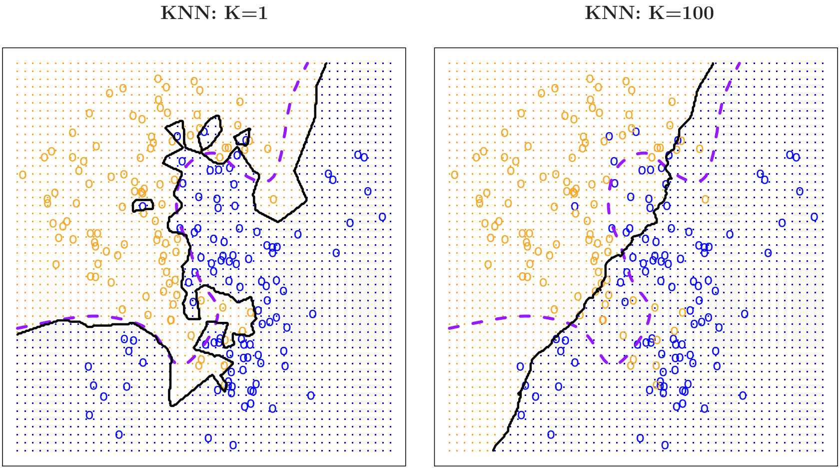
- can use for mixed feature vectors (qualitative and quantitative). If X_j is qualitative, replace $f_{kj}(x_j)$ with probability mass function over discrete categories.

Despite strong assumptions, naive Bayes often produces good classification result

K-nearest neighbors in 2-dim




KNN: K=10
X_1
X_2


KNN: K=1
KNN: K=100

Training errors and test errors