IntroML
ML. 3. (Linear) Classification

DataLab CSIC

Objectives and schedule

General concepts and basic algos in (linear) classification

- Discriminant functions
- Probabilistic discriminative approaches
- Probabilistic generative approaches

Contents

- Key concepts: Likelihood optimisation for classification, Bayesian classification, Stochastic gradient descent, multiple class, metrics, glms (beyond logistic regression)
- Later chapters more on non-linear classification

ISLR 4, ESL 4, Bishop 4

Lab

- LDA, QDA
- Full logistic regression example (incl. Regularisation)
- Text classification (multiple classifiers)

Classification. Three broad approaches

Objective

Divide input space into decision regions so that each input is associated to one (and only one) class

Methods (in this chapter). Decision surfaces are linear in inputs

- Discriminant function
- Probabilistic generative model
- Probabilistic discriminative model

Classification. Three broad approaches (as usual with blurry frontiers)

Objective

Divide input space into decision regions so that each input is associated to one (and only one) class

Methods (in this chapter). Decision surfaces are linear in inputs

- Discriminant function
- Probabilistic generative model
- Probabilistic discriminative model

A decision analytic perspective

K classes. Prior probability for class i π_i

Given class i, feature distribution $f(x|\theta_i) = f_i(x)$

Given x, posterior probability for class i $p_i(x) = \frac{\pi_i f_i(x)}{\sum_{i=1}^{K} \pi_i f_i(x)} \propto \pi_i f_i(x)$

Utility for assigning to class j when actually in class i u_{ij}

Maximise expected utility $\arg\max_j \sum_{i=1}^{K} u_{ij} \frac{\pi_i f_i(x)}{\sum \pi_i f_i(x)} = \arg\max_j \sum_{i=1} u_{ij} \pi_i f_i(x)$

Discriminant functions

Discriminant functions

A function that takes an input x and assigns it to one of the clases

No direct assessment of class probability given input x

LDA

Feature density given class is normal

Same variance for all clases

$$p_i(x) = \frac{1}{\sqrt{2\pi}\,\sigma_i} \exp\left(-\frac{1}{2}\frac{(x-\mu_i)^2}{\sigma_i^2}\right)$$

$$\sigma_1^2 = \sigma_2^2 = \dots = \sigma_K^2$$

$$\max \; p_i(x) = \frac{\pi_i \frac{1}{\sqrt{2\pi}\,\sigma} \exp\left(-\frac{1}{2\sigma^2}(x-\mu_i)^2\right)}{\sum_{l=1}^{K} \pi_l \frac{1}{\sqrt{2\pi}\,\sigma} \exp\left(-\frac{1}{2\sigma^2}(x-\mu_l)^2\right)} \propto \pi_i \exp\left(-\frac{1}{2\sigma^2}(x-\mu_i)^2\right) \propto \pi_i \exp\left(-\frac{1}{2\sigma^2}(\mu_i^2 - 2\mu_i x)\right)$$

$$\max \; x\frac{\mu_i}{\sigma^2} - \frac{\mu_i^2}{2\sigma^2} + \log(\pi_i)$$

LDA (cont)

K=2, Equal prior probs

Class 1 if

$$2x(\mu_1-\mu_2) > \mu_1^2-\mu_2^2$$

Boundary

$$x = \frac{\mu_1^2-\mu_2^2}{2(\mu_1-\mu_2)} = \frac{\mu_1+\mu_2}{2}$$

Gen,Discriminant function

$$\hat{\delta}_i(x) = x\,\frac{\hat{\mu}_i}{\hat{\sigma}^2} - \frac{\hat{\mu}_i^2}{2\hat{\sigma}^2} + \log(\hat{\pi}_i)$$

$$\hat{\mu}_i = \frac{1}{n_i}\sum_{y_j=i} x_j$$

$$\hat{\sigma}^2 = \frac{1}{n-K}\sum_{k=1}^{K}\sum_{y_j=k}(x_j-\hat{\mu}_k)$$

$$\hat{\pi}_i = \frac{n_i}{n}$$

QDA

Feature density given class is normal

$$X \mid i \sim N_i(\mu_i, \Sigma_i)$$

Specific variance for each class

$$\Sigma_i$$

Discriminant function

$$\delta_i(x) = -\frac{1}{2} x^T \Sigma_i^{-1} x + x^T \Sigma_i^{-1} \mu_i - \frac{1}{2} \mu_i^T \Sigma_i^{-1} \mu_i - \frac{1}{2} \log |\Sigma_i| + \log \pi_i$$

$$\hat{\Sigma}_i, \hat{\mu}_i, \hat{\pi}_i \rightarrow \operatorname{argmax} \delta_i(x)$$

LDA vs QDA

With p predictors

Algo	Pars	Flexibility	Variance	Bias
LDA	p(p+1)/2	Less	Smaller	Bigger
QDA	K(p(p+1))/2	More	Bigger	Smaller

Regularised discriminant analysis

$$\Sigma_i(\alpha) = \alpha\hat{\Sigma}_i + (1-\alpha)\hat{\Sigma}$$

$$\hat{\Sigma}(\gamma) = \gamma\hat{\Sigma} + (1-\gamma)\hat{\sigma}^2 I$$

Computations in discriminant analysis

$\Sigma = U D U^T$

U ORTHONORMAL

D DIAGONAL POSITIVE EIGENVALUES

$$(x-\hat{\mu})^T \Sigma^{-1} (x-\hat{\mu}) = (U^T(x-\mu))^T D^{-1} (U^T(x-\mu))$$

$$\log|\Sigma| = \sum \log d_{ii}$$

1. $X^* \longleftarrow D^{-1/2} U^T X$
2. CLASSIFY NEW INSTANCE TO THE CLOSEST CENTROID, MODULO π_i

Probabilistic discriminative models

Probabilistic discriminative models

Compute class probabilities given input, typically through a parametric model

Logistic regression

Data

x_i: EXPLANATORY

$(x_i, y_i), i = 1, \dots, n \qquad y_i \in \{0, 1\}$

Model

$$y_i \mid \theta_i \sim Ber(\theta_i) \qquad \text{logit}\, \theta_i = \log \frac{\theta_i}{1-\theta_i} = \alpha + \beta x_i$$

$$\theta_i = \text{logit}^{-1}(\alpha + \beta x_i) = \frac{e^{\alpha+\beta x_i}}{1+e^{\alpha+\beta x_i}} = \frac{1}{1+e^{-(\alpha+\beta x_i)}}$$

Logistic regression. Likelihood formulation

$$\ell(\alpha,\beta \mid (x,y)) = \left[\prod_{i:y_i=1} \theta_i(x_i)\right]\left[\prod_{i:y_i=0} (1-\theta_i(x_i))\right] = \prod_{i=1}^{n}\left[\theta_i(x_i)^{y_i}(1-\theta_i(x_i))^{1-y_i}\right]$$

$$\log\left(\ell(\alpha,\beta \mid (x,y)\right) = \sum_{y_i=1} \log\theta_i(x_i) + \sum_{y_i=0} \log\left(1-\theta_i(x_i)\right)$$

$$= \sum_{i=1}^{n} y_i \log(\theta(x_i)) + (1-y_i)\log\left(1-\theta_i(x_i)\right)$$

Logistic regression. Evaluating the likelihood

Some little tricks

$$\frac{\text{logit}^{-1} z}{1-\text{logit}^{-1} z} = \frac{\frac{1}{1+e^{-z}}}{1-\frac{1}{1+e^{-z}}} = \frac{1}{e^{-z}} = e^{z} \qquad \log\left(\frac{\text{logit}^{-1} z}{1-\text{logit}^{-1} z}\right) = z$$

$$1-\text{logit}^{-1}(z) = 1-\frac{1}{1+e^{-z}} = \frac{e^{-z}}{1+e^{-z}} = \frac{1}{1+e^{z}}$$

$$\log\left(1-\text{logit}^{-1}(z)\right) = -\log\left(1+e^{z}\right) = -\text{log1p}\left(e^{z}\right)$$

$$\log\left(\ell(\alpha,\beta \mid (x,y)\right) = \sum_{y_i=1}(\alpha+\beta x_i) - \sum_{y_i=0}\text{log1p}\left(\exp(\alpha+\beta x_i)\right)$$

$$= \sum y_i(\alpha+\beta x_i) - (1-y_i)\,\text{log1p}\left(\exp(\alpha+\beta x_i)\right)$$

Logistic regression. Gradient descent

Basic approach

$$(\alpha, \beta)_{i+1} = (\alpha, \beta)_i - \eta_i \nabla \log(\ell(\alpha,\beta))$$

$$\nabla \log(\ell(\alpha,\beta)) = \begin{pmatrix} \sum_{y_i=1} 1 - \sum_{y_i=0} \frac{\partial}{\partial \alpha} \log 1p(\exp(\alpha + \beta x_i)) \\ \sum_{y_i=1} x_i - \sum_{y_i=0} \frac{\partial}{\partial \beta} \log 1p(\exp(\alpha + \beta x_i)) \end{pmatrix} \quad O(n)$$

Newton-Raphson

$$(\alpha, \beta)_{i+1} = (\alpha, \beta)_i - H^{-1} \nabla \log(\ell(\alpha,\beta))$$

Iterative reweighted least squares

Logistic regression. Stochastic gradient descent

$$f(w) = \frac{1}{n}\sum_{i=1}^{n} f_i(w)$$

$$w^{t+1} = w^t - \eta_t \nabla_w f(w)$$

$$= w^t - \eta_t \left(\frac{1}{n}\sum_{i=1}^{n} \nabla_w f_i(w)\right) \qquad O(n)$$

SAMPLE MINIBATCH $B\{i_1,\dots,i_B\} \subset \{1,\dots,n\}$

$$w^{t+1} = w^t - \eta_t \frac{1}{B}\sum_{i\in B} f_i(w^t) \qquad O(B)$$

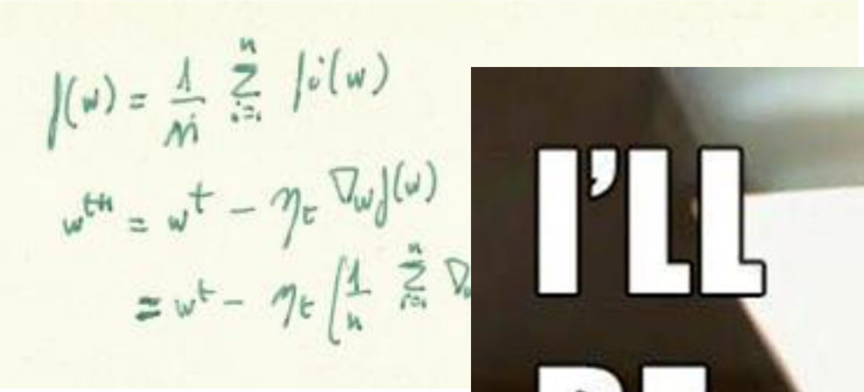
When n is super-large

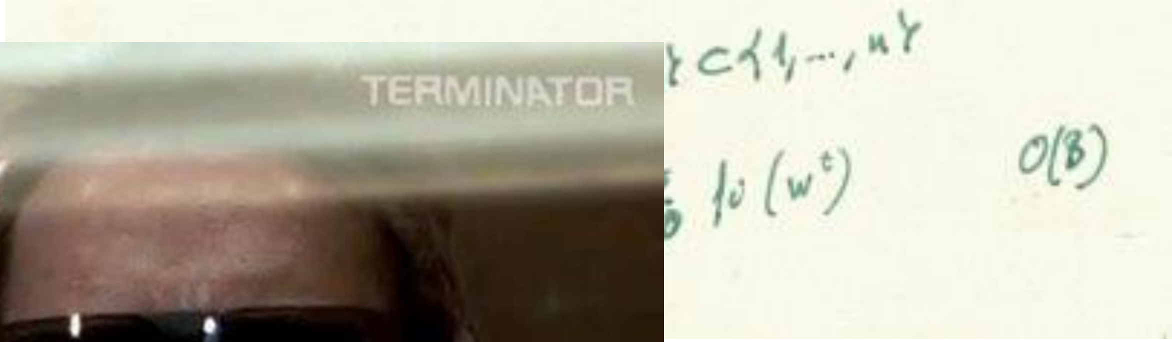
Robbins Monro (1954) stoch. approx.

$$\sum \eta_t = \infty \quad \sum \eta_t^2 < \infty \Rightarrow |f(w^t) - f^*| = O(1/t)$$

Fits in memory, smaller complexity, Parallelise, Escape local optima

Logistic regression. Stochastic gradient descent





super-large

Robbins Monro (

Fits in memory, smaller complexity, Parallelise, Escape local optima

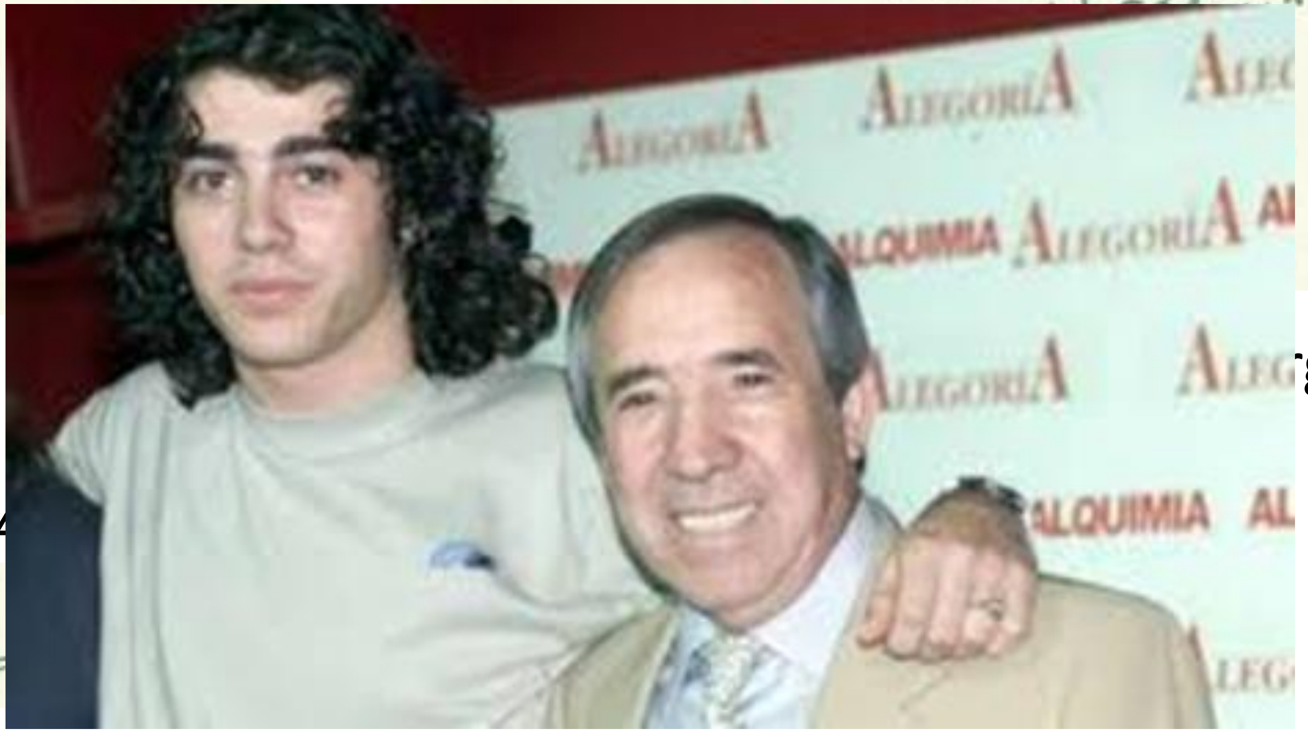
Logistic regression. Stochastic gradient descent

$$J(w) = \frac{1}{m} \sum_{i=1}^{n} J_i(w)$$

$$w^{t+1} = w^t - \eta_t \nabla_w J(w)$$

$$= w^t - \eta_t \left(\frac{1}{n} \sum_{i=1}^{n} \nabla_w J_i(w) \right)$$

$O(B)$



rge

Robbins Monro (1954

$\sum \eta_t =$

Fits in memory, smaller complexity, Parallelise, Escape local optima

Logistic regression. Interpretation

Odds ratio

$$\theta(x) = \frac{e^{\alpha+\beta x}}{1+e^{\alpha+\beta x}} \qquad \text{ODDS} \quad \frac{\theta(x)}{1-\theta(x)} = e^{\alpha+\beta x} \in (0,\infty)$$

$$\log \frac{\theta(x)}{1-\theta(x)} = \alpha+\beta x \qquad x \rightarrow x+1 \Rightarrow \text{LOG-ODDS} + \beta$$

$$\Downarrow$$

$$\text{ODDS} * e^{\beta}$$

$$OR = \frac{\exp(\alpha+\beta(x+1))}{\exp(\alpha+\beta x)} = e^{\beta}$$

Odds Ratio	X vs Y
1	No association
>1	Bigger output probability
<1	Smaller output probability

Logistic regression. Prediction

Just plug-in estimators to assess probabilities

$$\hat{\alpha}, \hat{\beta}$$

$$Pr(Y=1 \mid \tilde{x}) = \frac{e^{\hat{\alpha}+\hat{\beta}\tilde{x}}}{1+e^{\hat{\alpha}+\hat{\beta}\tilde{x}}}$$

Cut-off level

$$Pr(Y=1 \mid \tilde{x}) \geq k \longrightarrow 1$$

Logistic regression with regulariser

$$\max \sum_{i=1}^{n} \left[y_i \log p(x_i|\beta) + (1-y_i) \log\left(1-p(x_i|\beta)\right) \right] - \lambda \sum_j \beta_j^2$$

$$\max \sum_{i=1}^{n} \left[y_i \log p(x_i|\beta) + (1-y_i) \log\left(1-p(x_i|\beta)\right) \right] - \lambda \left| \sum_i \beta_i \right|$$

Bayesian logistic regression

Likelihood
Generic prior
Generic posterior

$$p(\alpha,\beta|y,x) \propto p(\alpha,\beta)\prod_{i=1}^{n}\left[p(x_i|\beta)^{y_i}\,(1-p(x_i|\beta))^{1-y_i}\right]$$

If few parameters, numerical integration
    else MCMC or Laplace approximations

Multiclass logistic regression

OR with K classes

$$\log \frac{P(1|x)}{P(K|x)} = \beta_{10} + \beta_1^t x \quad \Big| \quad \log \frac{P(2|x)}{P(K|x)} = \beta_{20} + \beta_2^t x \quad \Big| \cdots \Big| \quad \log \frac{P(K-1|x)}{P(K|x)} = \beta_{(K-1)0} + \beta_{K-1}^t X$$

Probabilities

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

Parameters

$$\theta = (\beta_{10}, \beta_1^t; \dots; \beta_{(K-1)0}, \beta_{K-1}^t)$$

Log-likelihood

$$\ell(\theta) = \sum_{i=1}^{N} \log p_{g_i}(x_i|\theta) \qquad p_k(x_i|\theta) = Pr(k|x_i, \theta)$$

Probit regression

Close to logistic regression

$$Pr(1|x) = \Phi(\beta_0 + \beta_1 x) \qquad \Phi \leftrightarrow N(0,1)$$

$$\Phi(z) = \frac{1}{2}\left\{1 + \mathrm{erf}\left(\frac{z}{\sqrt{2}}\right)\right\}$$

glms

Exponential family

$$p(y|\eta) = b(y)\exp(\eta T(y))\, h(\eta)$$

e.g., Bernoulli

$$\theta^y (1-\theta)^{1-y} = 1\cdot\left(\frac{\theta}{1-\theta}\right)^y (1-\theta)$$

$$= 1\cdot \exp\left(\left(\log\frac{\theta}{1-\theta}\right) y\right)\cdot(1-\theta)$$

glm

GLM

$$y \,|\, x; \theta \sim \text{ExpFamily}(\eta) \quad h_\theta(x) = E(y|x,\theta) \quad \eta = \theta^t x$$

Probabilistic generative models

Probabilistic generative models

Estimate class probabilities and input distribution given class

Compute class probability given input via Bayes formula

Naive Bayes

Variational autoencoders

Naive Bayes

$$\Pr(k|x) = \frac{\pi_k f_k(x)}{\sum_i \pi_i f_i(x)}$$

NB ass.

$$f_k(x) = \prod_{j=1}^{p} f_{kj}(x_j)$$

Naive Bayes. Parameter estimation

Continuous

$$X_j | k \sim N(\mu_{kj}, \sigma_{kj}^2)$$

Counts

$$Pr(X_j = x | k) = \frac{n_{kj} + 1}{n_k + h}$$

Categorical

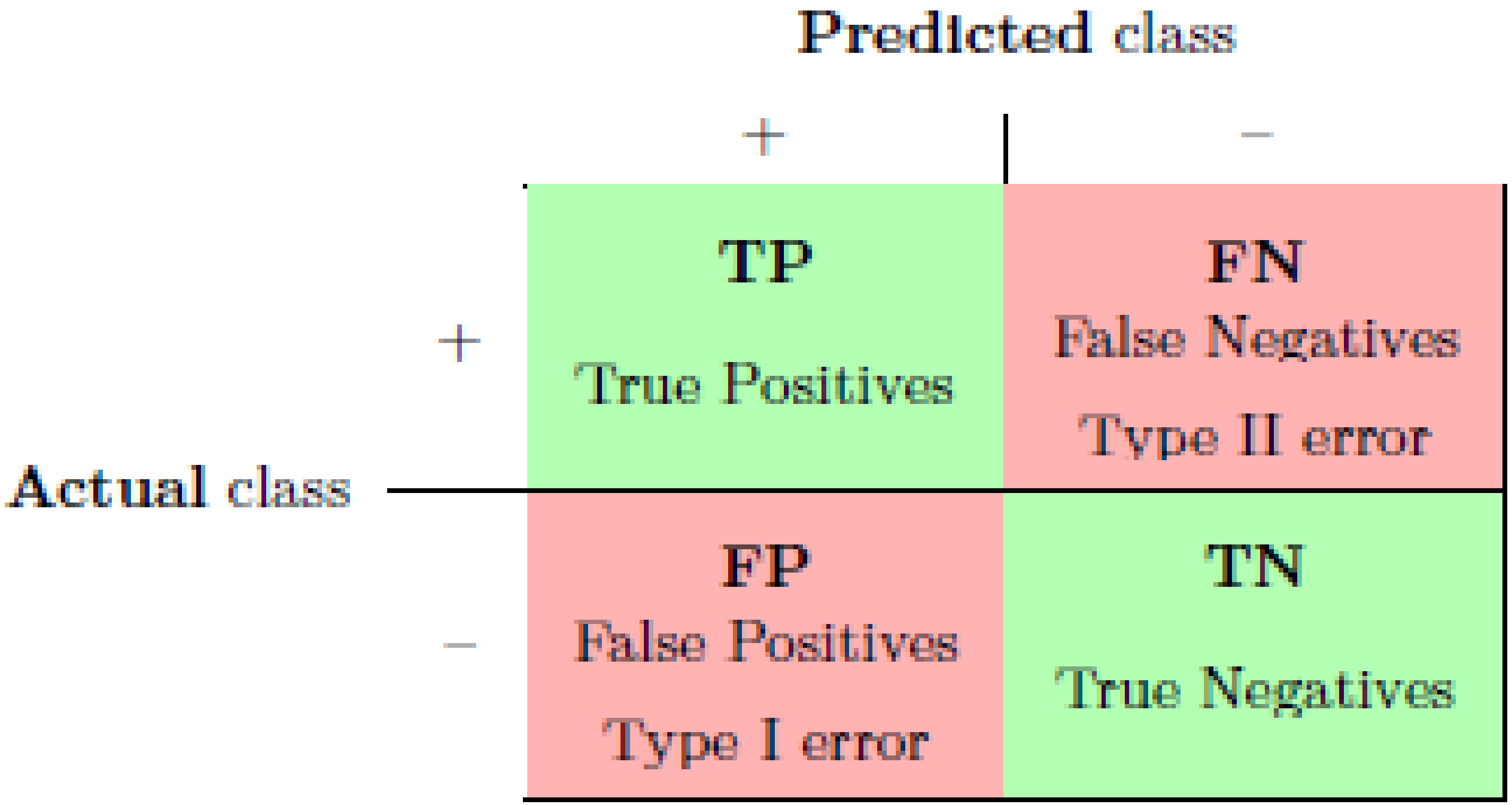
$$Pr(X_j = x | k) = \frac{\sum^{n} I(x_{ij} = x)\, I(Y = y_k)}{\sum I(Y = y_k)}$$

Classification metrics

Ok… but how do we assess classifier performance???

(More in labs!!!)

Confusion matrix



Confusion matrix

		Predicted class +	Predicted class −
Actual class	+	**TP** True Positives	**FN** False Negatives Type II error
	−	**FP** False Positives Type I error	**TN** True Negatives

Metric	**Formula**	**Interpretation**
Accuracy	$\frac{TP + TN}{TP + TN + FP + FN}$	Overall performance of model
Precision	$\frac{TP}{TP + FP}$	How accurate the positive predictions are
Recall Sensitivity	$\frac{TP}{TP + FN}$	Coverage of actual positive sample
Specificity	$\frac{TN}{TN + FP}$	Coverage of actual negative sample

Imbalanced problems

Accuracy not sufficient in imbalanced problems

- Example. CTR (click-through rate, click 1, no-click 0)
  - About 10^8 ads (observations), only 80000 clicks
  - A model classifying always as 0, accuracy of 99.92%

Unequal class distribution inherent in

Fraud detection
Anomaly detection
Credit default prediction
Conversion prediction
Intrusion detection….

Imbalanced problems

F1 score	$\frac{2TP}{2TP + FP + FN}$	Hybrid metric useful for unbalanced classes

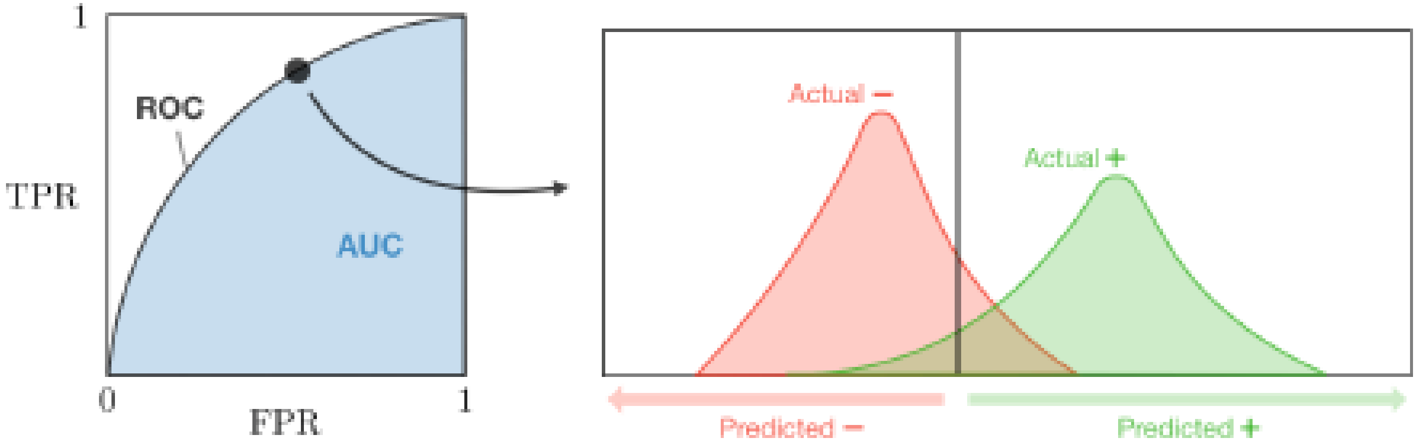
Oversampling SMOTE Synthetic Minority Oversampling Tech

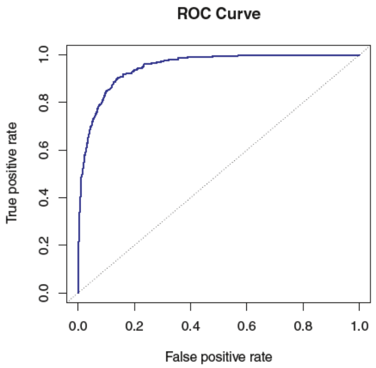
Undersampling

Bayesian methods

ROC (receiver operating characteristics)
AUC (area under curve)

Metric	Formula	Equivalent
True Positive Rate TPR	$\frac{TP}{TP + FN}$	Recall, sensitivity
False Positive Rate FPR	$\frac{FP}{TN + FP}$	1-specificity





More generally, need to take into account costs

Fraud, no fraud

Cannabinoids. Decisions entail researching or not researching drug

Calibration

Discrimination vs Calibration

The Achilles heel of predictive analytics

https://bmcmedicine.biomedcentral.com/articles/10.1186/s12916-019-1466-7

When you say probability of A is 70% you are right 70% of the times

Back to the decision analytic perspective…

K classes. Prior probability for class i

$$\pi_i$$

Given class i, feature distribution

$$f(x|\theta_i) = f_i(x)$$

Given x, posterior probability for class i

$$p_i(x) = \frac{\pi_i f_i(x)}{\sum_{i=1}^{K} \pi_i f_i(x)} \propto \pi_i f_i(x)$$

Utility for assigning to class j when actually in class i

$$u_{ij}$$

Maximise expected utility

$$\arg\max_j \sum_{i=1}^{K} u_{ij} \frac{\pi_i f_i(x)}{\sum \pi_i f_i(x)} = \arg\max_j \sum_{i=1} u_{ij} \pi_i f_i(x)$$

Classification. To be seen

C+R. Decision trees, random forests, boosting

C (+R). Support vector machines

R +C. Perceptrons. Neural networks. Deep neural nets

See you next week

introml@icmat.es

Stuff at

https://datalab-icmat.github.io/courses_stats.html