Statistical Pattern Recognition

"Classical" approach

- Parametric — classical distributions
- Theoretically motivated
- Often too 'rigid' in practice

"Modern" approach

- Redundant parametrization
- Numerical optimization over training set
- Heuristic / biological motivation
- Theoretical motivation is Ripley's goal . . .

Overfitting

- A major problem with redundant methods
- ***Smoothing assumptions*** are critical
- ***Model selection*** to eliminate redundant parameters

Probabilistic data model is appropriate

- Objects independent and identically distributed
- Can model distributions parametrically

— well-behaved, known or estimable

- Learning requires a ***representative training set***

Basic Assumptions

Each object has *exactly one* class k

- Classes are discrete & unstructured

— ***not*** hierarchical, overlapping, parametric. . .

— enumerable (not too many)

- Mutually exclusive and exhaustive

— not like properties: can't have none or more than one

— no outliers (include a special class for these. . .)

Each object characterized by a *feature vector*

- ***Same*** features for each class

— no errors, missing measurements, extra information. . .

- *All* relevant information is included
- Features are informative about class

Hard, probabilistic decisions are acceptable

- Decisions may be ***wrong***

— based on ***expected loss***, maximum loss unlimited

— forced choice among equally likely classes

- Decisions are ***fragile*** — may change with

— small perturbations

— smoothing assumptions, parametric forms

— splitting or merging classes completely changes decisions

> **• Is a firm decision *really* necessary ?**
> **• Is the loss model appropriate ?**

c.f. Bayesian networks, probabilistic logics, fuzzy sets, . . .

— maintain a list of plausible alternatives

Bayesian Decision Theory

$\boxed{1}$ Given features $\boldsymbol{x}$, choose ***classifier response*** r from fixed repertoire R

$\boxed{2}$ Penalize bad responses with ***loss function*** $L(r|k)$

— depends on r and unknown true class k (maybe $\boldsymbol{x}$ too)

Bayes classifier* chooses response to *minimize expected loss* at *x

- Expectation over classes, minimization over responses

$$r_{Bayes}(\boldsymbol{x}) \equiv \arg\min_{r \in R} \sum_{k \in K} L(r|k)\, \boldsymbol{p}(k|\boldsymbol{x})$$

- Bayes rule as usual

$$\boldsymbol{p}(k|\boldsymbol{x}) = \frac{\boldsymbol{p}(\boldsymbol{x}|k)\, \boldsymbol{p}(k)}{\boldsymbol{p}(\boldsymbol{x})}$$

- Decision boundaries partition measurement space

- Best response depends *only* on ***ratios*** of ***joint*** or ***posterior likelihoods*** at $\boldsymbol{x}$

$$\frac{\boldsymbol{p}(k|\boldsymbol{x})}{\boldsymbol{p}(l|\boldsymbol{x})} = \frac{\boldsymbol{p}(k,\boldsymbol{x})}{\boldsymbol{p}(l,\boldsymbol{x})} = \frac{\boldsymbol{p}(\boldsymbol{x}|k)\, \boldsymbol{p}(k)}{\boldsymbol{p}(\boldsymbol{x}|l)\, \boldsymbol{p}(l)} \approx \text{ratio of } k\text{'s to } l\text{'s "near } \boldsymbol{x}\text{"}$$

- Equivalently, use ***differences of log likelihoods***
- Absolute probabilities $\boldsymbol{p}(\boldsymbol{x})$ are ***irrelevant !***

$\Rightarrow$ sensitive to outliers

- Local at $\boldsymbol{x}$ $\Rightarrow$ improper densities are OK

Bayes classifier is *optimal* but *fragile*

- Performance standard for other methods
- Uses ***a lot*** of information — full relative distributions
- Sensitive to outliers and modelling errors

Useful response types

- "k" — 'I think class is k'
- "DOUBT" — 'I can't decide' (any class is possible)
- "OUTLIER" — 'no class is plausible'
- "EITHER k OR l OR ..." — doubt among several classes

— many possible combinations (2^K)

— *c.f.* probabilistic logic, Dempster-Shafer, ...

Default loss function

- $L(\text{"}k\text{"}|l) = \begin{cases} 0 \text{ if } k = l \\ 1 \text{ otherwise} \end{cases}$

— Bayes response is any "k" maximizing $\boldsymbol{p}(k|\boldsymbol{x})$

— expected loss = probability of misclassification

- $L(\text{"DOUBT"}|k) = 1 - P_d$

— "DOUBT" if best class has $\boldsymbol{p}(k|\boldsymbol{x}) < P_d$

For real applications use a *realistic* loss function

Performance Measures

- Expected loss at $\boldsymbol{x}$ (average over classes k)
- Probability of ***misclassification*** / ***doubt*** at $\boldsymbol{x}$
- ***Risk*** = expected loss for class k (average over $\boldsymbol{x}$)
- ***Total risk*** = total expected loss (average over k and $\boldsymbol{x}$)
- ***Bayes risk*** = total risk of Bayes classifier

Missing Values

E.g. $\boldsymbol{x} = (\boldsymbol{y}, \boldsymbol{z})$ with $\boldsymbol{y}$ observed, $\boldsymbol{z}$ missing

1 ***Optimal "predictive" approach:*** use marginal distributions

$$\boldsymbol{p}(k|\boldsymbol{y}) \;\propto\; \boldsymbol{p}(\boldsymbol{y}|k) = \int \boldsymbol{p}(\boldsymbol{y}, \boldsymbol{z}|k)\, d\boldsymbol{z}$$

2 ***Fall back:*** plug-in estimates of missing values

— *e.g.* maximum likelihood

$$\boldsymbol{p}(\boldsymbol{y}|k) \;\sim\; \boldsymbol{p}(\boldsymbol{y}, \hat{\boldsymbol{z}}(\boldsymbol{y})|k)$$

3 ***Last resort:*** use default values, training set means

- All of these require ***prior*** distributions $\boldsymbol{p}(\boldsymbol{x}|k)$
- Nothing works if missing-ness ***changes*** $\boldsymbol{p}(\boldsymbol{x}|k)$!
- For variables whose values are often missing, can include "MISSING" as a possible value

— not clear how to model MISSING distributions

Outliers

- Include "OUTLIER" response — outliers ***do*** exist !
- They occur with some local density, just like inliers

$\Rightarrow$ include an OUTLIER class (just like other classes)

- Uninformative prior — "uniform", often improper. . .

— May also depend on ***inlier*** density, *e.g.* misreadings

— I think choice of prior no harder than for other classes, but *c.f.* Ripley

— Key ratio is ***posterior:*** outliers to k's "near $\boldsymbol{x}$"

- Loss model for $L(\text{“}k\text{”}|\text{OUTLIER}),\, L(\text{“OUTLIER”}|k)$
- Outliers sometimes ***more*** interesting than classes !

Linear & Quadratic Discriminants

- Class priors are ***Normal*** $\boldsymbol{p}(\boldsymbol{x}|k) \sim \mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$
- Log posterior likelihoods

$$\log \boldsymbol{p}(k|\boldsymbol{x}) \sim -\tfrac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k (\boldsymbol{x}-\boldsymbol{\mu}_k) + \log\left(\frac{\boldsymbol{p}(k)}{|\boldsymbol{\Sigma}_k/2\pi|^{1/2}}\right)$$

$$\log \frac{\boldsymbol{p}(k|\boldsymbol{x})}{\boldsymbol{p}(l|\boldsymbol{x})} = -\tfrac{1}{2}\boldsymbol{x}^\top(\boldsymbol{\Sigma}_k - \boldsymbol{\Sigma}_l)\boldsymbol{x} + (\boldsymbol{\Sigma}_k\boldsymbol{\mu}_k - \boldsymbol{\Sigma}_k\boldsymbol{\mu}_l)^\top \boldsymbol{x} + \text{const.}$$

- Decision boundaries ***quadratic*** (ellipsoid, hyperboloid,. . .) but ***linear*** if all classes have ***same covariance***
- Same results apply for $\boldsymbol{p}(\boldsymbol{x}|k) \sim f(\boldsymbol{x}) \exp\left(Q_k(\boldsymbol{x})\right)$ with $f \not\propto k$, Q_k linear or quadratic

— often good local model, even if globally untrue

— linear / quadratic approximation to local decision boundaries

- Quadratic model has *many more* parameters than linear one

$\Rightarrow$ parameter estimates much more variable, for small training samples linear model may be better

Computational Geometry

- Efficient algorithms for finding decision boundaries in problems with ***very many classes***:
  - — Constant $p(k)$ and $\Sigma_k \Rightarrow$ ***Voronoi diagrams***
  - — Constant $\Sigma_k \Rightarrow$ ***power diagrams***
  - — Σ_k known up to scale $\Rightarrow$ ***Apollonius model***
- *c.f.* Aurenhammer, ACM Computing Surveys 23, p 345–405, 1991

What To Estimate

Sampling paradigm models ***priors*** $p(x|k), p(k)$

- very general, but may model irrelevant detail
- handles missing values, outliers, unclassified training data

Diagnostic paradigm models ***posteriors*** $p(k|x)$

- *e.g.* logistic discriminant $\log \frac{p(k|x)}{p(l|x)}$
- allows ***importance sampling*** of training data

Discriminant paradigm models ***decision boundaries***

- classification specific, but ***not*** necessarily probabilistic
- Ripley wants to reduce it to diagnostic case

Each case uses training data to estimate parameters of an *assumed parametric form*.

Approximating the Bayes classifier

Estimate distributions from ***labelled training data*** $\mathcal{T}$

- Exactly ***what*** should we "learn", and ***how***?
- The usual problems of bias, overfitting, validation, . . .
- A ***generalization*** problem — smoothness & regularity assumptions are ***critical***

How To Estimate It

Plug-in rules — plug parameter estimates into parametric distribution (*e.g.* max. likelihood)

$$p(\ldots) \approx p(\ldots|\hat{\theta}(\mathcal{T}))$$

- ***Consistency:*** most estimators converge asymptotically to best (least false) model in family
- ***Risk-prone:*** tails often underestimated for finite training sets

Predictive approach — hedge bets by ***averaging over possible parameter values***

$$p(\ldots) \approx \int p(\ldots|\theta)\, p(\theta|\mathcal{T})\, d\theta$$

- Typically gives ***less extreme*** values, especially in tails

Heuristic approach — tune parameters for best overall performance (careful validation required!)

What counts is accurate modelling of *posterior likelihood ratios* near *decision boundaries*

- ***Not*** a standard estimation problem
- Tails of distribution often important

Models are *Always Wrong*

- Modelling errors are ***not a problem*** provided ***decision boundaries*** remain reasonable
- ***Low probability regions*** are especially vulnerable
  — rare cases, outliers have little influence on fit
- Adding parameters improves ***apparent fit***
  — but ***not*** necessarily ***performance on unseen examples***

Overfitting

- Surplus parameters fit 'noise' in training data
  — increased bias, variability, computation
  — poor ***generalization:*** worse performance on real data
- Model selection, smoothing, bias reduction techniques . . .

• Compromise between *minimal* and *least false* models
• Validate performance on *independent test data*

Neighbourhood Based Classifiers

- Estimate posteriors $\boldsymbol{p}(k|\boldsymbol{x})$ by ***counting number of training examples*** in each class "near $\boldsymbol{x}$"
- Optimal classification is most frequent class

But what does "*NEAR*" mean ?

- Many possible choices of neighbourhood
- High dimensional training data are ***always sparse***

$\Rightarrow$ large neighbourhoods needed to get significant samples

Smoothing, interpolation and parametric assumptions are critical

- They make a little data go a long way !

— sometimes *too* far . . .

Fitting Models that are wrong

- "***Least false***" model maximizes some measure of fit over model parameters $\boldsymbol{\theta}$

Maximum A Posteriori (MAP)

- Maximizes ***posterior likelihood*** of model given data & prior

$$\log \mathrm{Pr}(\mathrm{model}|\mathrm{data}) \sim \log \mathrm{Pr}(\mathrm{data}|\mathrm{model}) + \log \mathrm{Pr}(\mathrm{model}) \\ = \textstyle\sum_i \log \boldsymbol{p}(\boldsymbol{x}_i|\boldsymbol{\theta}) + \log \boldsymbol{p}(\boldsymbol{\theta})$$

Maximum Likelihood (ML)

- Ignores prior $\mathrm{Pr}(\mathrm{model})$ and maximizes

$$\log \mathrm{Pr}(\mathrm{data}|\mathrm{model}) = \sum_i \log \boldsymbol{p}(\boldsymbol{x}_i|\boldsymbol{\theta})$$

- Prior $\mathrm{Pr}(\mathrm{model})$ is significant for
  — large models / small training samples
  — rarely active parameters (small *effective* training samples)

Limit of Dense Samples

- Above formulae are for *finitely many isolated samples*
- For asymptotic limit, dense/overlapping/binned samples, average becomes expectation w.r.t. ***observed sample distribution*** $\boldsymbol{q}(\boldsymbol{x})$

$$\boxed{\frac{1}{n}\sum_i f(\boldsymbol{x}_i) \quad \longrightarrow \quad \int f(\boldsymbol{x}) \cdot \boldsymbol{q}(\boldsymbol{x})\, d\boldsymbol{x}}$$

- Up to a constant, asymptotic limit of ML sum is ***divergence***
- Use as measure of ***least false*** model

Divergence / Relative Entropy

The ***divergence*** or ***relative entropy*** of a distribution $\boldsymbol{p}$ from a reference distribution $\boldsymbol{q}$ is

$$\boxed{d(\boldsymbol{p}|\boldsymbol{q}) = \mathrm{E}_{\boldsymbol{p}}\left[\,\log\frac{\boldsymbol{p}(\boldsymbol{x})}{\boldsymbol{q}(\boldsymbol{x})}\,\right] = \int \boldsymbol{p}(\boldsymbol{x})\,\log\frac{\boldsymbol{p}(\boldsymbol{x})}{\boldsymbol{q}(\boldsymbol{x})} \;\geq 0}$$

- Usually $\boldsymbol{q}$ is a model being tested, $\boldsymbol{p}$ is the true population distribution or an observed sample distribution derived from it.
- For sample distributions $\boldsymbol{p}$, $d(\boldsymbol{p}|\boldsymbol{q})$ measures ***relative improbability*** of samples given model. For n samples

$$\boxed{\Pr(\text{samples}|\text{model}) \;\approx\; \exp\left[-n \cdot d(\boldsymbol{p}|\boldsymbol{q})\,\right]}$$

Proof: Discretize $\boldsymbol{q}$. Count samples in each cell. Use Stirling's approximation on multinomial count distribution.

- For fine discretizations (≤ 1 sample per cell) divergence reduces to ***log likelihood*** $\sum_i \log \boldsymbol{q}(\boldsymbol{x}_i) + \mathrm{const.}$
- For coarse discretizations gives standard sample binning tests

Plug-in Examples

Normal Models

- *E.g.*, linear and quadratic discriminants based on Normal density models, using estimated $\hat{\boldsymbol{\mu}}_k$ and $\hat{\boldsymbol{\Sigma}}_k$
- Fitted tails tend to be unrealistically small, even with debiased $\hat{\Sigma}$ (*i.e.* $\frac{1}{n-1}$ instead of $\frac{1}{n}$)

Multivariate t Model

- "Predictive" version of Normal with uncertain covariance

— a more realistic model when covariance is unknown

- Much fatter tails $\Rightarrow$ distant observations downweighted
- M.L. estimators for $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ are robust M-estimators

— data weighted by $\left(1 + \frac{(\boldsymbol{x}-\hat{\boldsymbol{\mu}})^\top \hat{\boldsymbol{\Sigma}}^{-1} (\boldsymbol{x}-\hat{\boldsymbol{\mu}})}{\nu}\right)^{-1}$

— use iterative procedure to estimate $\hat{\boldsymbol{\mu}}$, $\hat{\boldsymbol{\Sigma}}$

- Decision boundaries are quadratic or linear if ν's are equal

Predictive Approach

- "Classical" Bayesian approach to parameter estimation
- ***Hedge*** over uncertain parameter values

$$\boxed{\boldsymbol{p}(\cdots) = \int \boldsymbol{p}(\cdots|\boldsymbol{\theta})\,\boldsymbol{p}(\boldsymbol{\theta})\,d\boldsymbol{\theta}}$$

— distributions "smeared out" — fatter tails, flatter peaks

— integral is often intractable

— should really hedge posterior likelihood ratios ...

- Results comparable to plug in methods, ***but:***

— more conservative, probabilities less extreme

— fatter bases of tails (a key region for classification !)

— more reliable for rare events

- Prior $\boldsymbol{p}(\boldsymbol{\theta})$ often controversial

— controls variation of unconstrained parameters

Exact Predictive Results

- Gaussian with uncertain mean $\rightsquigarrow$ wider Gaussian
- Gaussian with uncertain mean & covariance (+ suitable priors) $\rightsquigarrow$ multivariate t-distribution

— still has elliptical contours, but much fatter tails

— optimal decision rules ***not*** usually quadratic

Taylor Approximation

- If estimate of $\boldsymbol{\theta}$ has mean $\hat{\boldsymbol{\theta}}$ and small covariance $\hat{\boldsymbol{\Sigma}}$

$$\boldsymbol{p}(\cdots) \;\approx\; \boldsymbol{p}(\cdots|\hat{\boldsymbol{\theta}}) + \tfrac{1}{2}\mathrm{trace}\left[\hat{\boldsymbol{\Sigma}} \cdot \frac{D^2\boldsymbol{p}}{D\boldsymbol{\theta}^2}(\cdots|\hat{\boldsymbol{\theta}})\right]$$

— fills in hollows, flattens peaks

Empirical Bayes

- Use plug in estimators for some parameters, hedge over rest

Diagnostic *vs.* Sampling Paradigm

- Under M.L. fitting, a diagnostic model $\boldsymbol{p}(k|\boldsymbol{x})$ is indistinguishable from a ***sampling paradigm*** model that ***falsely assumes a uniform overall sample distribution***

$$\boldsymbol{p}(k,\boldsymbol{x}) = \boldsymbol{p}(k|\boldsymbol{x}) \cdot \boldsymbol{p}(\boldsymbol{x}) \text{ with } \boldsymbol{p}(\boldsymbol{x}) = \sum_k \boldsymbol{p}(k,\boldsymbol{x}) = \text{const.}$$

- Essential difference between paradigms is that diagnostic ***freezes the normalization*** $Z(\boldsymbol{x}|\boldsymbol{\theta})$ while sampling leaves it free to model $\boldsymbol{p}(\boldsymbol{x})$
- Frozen d.o.f. are ***irrelevant to decisions***, but provide ***importance weighting*** for different regions of measurement space
- Weighting is already ***implicit in training samples*** (*if* they're an unbiased sample of population ...)
- But modelling it provides ***smoothing*** and reduces influence of ***large relative fluctuations*** in ***low-probability regions***

Logistic Discrimination

- Directly model ***posterior log likelihoods***

$$\log \boldsymbol{p}(k|\boldsymbol{x}) \;\approx\; g_k(\boldsymbol{x}|\boldsymbol{\theta}) - Z(\boldsymbol{x}|\boldsymbol{\theta})$$
$$\log \frac{\boldsymbol{p}(k|\boldsymbol{x})}{\boldsymbol{p}(l|\boldsymbol{x})} \;\approx\; g_k(\boldsymbol{x}|\boldsymbol{\theta}) - g_l(\boldsymbol{x}|\boldsymbol{\theta})$$

- g's have an assumed parametric form, *e.g.* linear

$$g_k(\boldsymbol{x}|\alpha_k, \boldsymbol{\beta}_k) \;\equiv\; \alpha_k + \boldsymbol{\beta}_k \cdot \boldsymbol{x}$$

- Z provides normalization $\sum_k \boldsymbol{p}(k|\boldsymbol{x}) = 1$, but is ***irrelevant*** for discrimination

$$Z(\boldsymbol{x}|\boldsymbol{\theta}) \;\equiv\; \log\left[\sum_k \exp\left(g_k(\boldsymbol{x}|\boldsymbol{\theta})\right)\right]$$

- Maximum Likelihood fit maximizes log likelihood

$$\sum_i \log \boldsymbol{p}(k_i|\boldsymbol{x}_i,\boldsymbol{\theta}) \;=\; \sum_i g_{k_i}(\boldsymbol{x}_i|\boldsymbol{\theta}) - Z(\boldsymbol{x}_i|\boldsymbol{\theta})$$

— simple (*e.g.* linear) apart from Z term

Using Unclassified Observations

- *E.g.* unlabelled training data, adaptive on-line tuning
- In ***sampling paradigm***, fit ***mixture model*** to ***total prior*** $\boldsymbol{p}(\boldsymbol{x})$

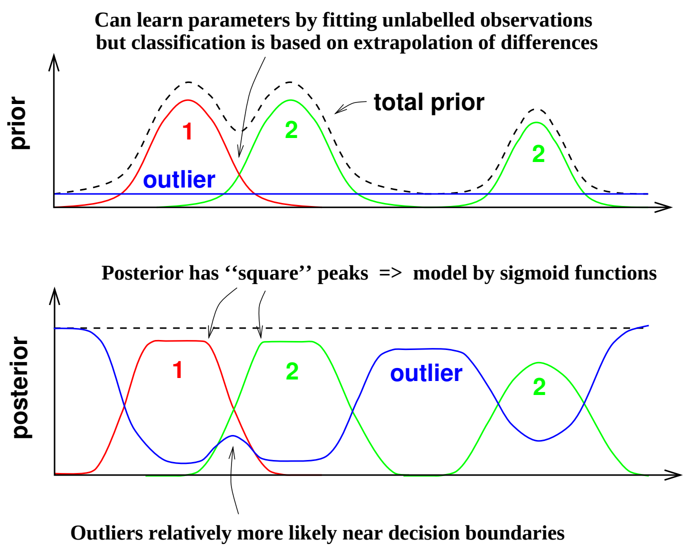
— can also use $\boldsymbol{p}(\boldsymbol{x})$ for outliers, missing values ...

- Use ***mixture components*** as ***class priors*** $\boldsymbol{p}(\boldsymbol{x}|k), \boldsymbol{p}(k)$

— a ***very strong*** parametric assumption on $\boldsymbol{p}(\boldsymbol{x}|k)$!

— overlapping tails extrapolated from peaks

— implied decision rules can be *very sensitive* to this



Estimating Mixture Distributions

- *E.g.* mixtures of Gaussians, sigmoids, radial basis functions
- Parametrization problems make fitting delicate
- ***Singularities*** — components collapse to a point or subspace
- ***Unidentifiable parameters***

— superimposed components

— infinite width or zero amplitude components

— renaming components makes no difference

Even for the simplest problems *regularization* is essential

- Limit range of spatial scales
- Limit effects of unidentifiable parameter combinations
- Prune redundant parameters when possible

Expectation-Maximization (E-M)

- A common general method for mixture estimation
- Often very slow (linear convergence)

Bias Correction

- Use theoretical bias estimates to correct training bias
- Model sensitive and usually only asymptotic
- But computationally cheap and no test set needed
- *e.g.* ***Information Criteria:*** AIC, NIC, BIC, ...

Performance Assessment

Performance on *unseen data* is what counts !

- Trained classsifiers are ***biased towards training data***
- Mean error on training examples ***underestimates*** error on unseen ones

Validation

- Estimate performance using ***independent test set***
- Simple and reliable, but wastes potential training data

Cross-validation

- Schemes that reuse subsets of training data for testing

1. Train classifier on a subset, validate on rest of data
2. Repeat for many different subsets and average results

- *e.g.* leave-one-out, jackknife, bootstrap, ...
- Estimates are unbiased but often quite variable

— training subsets are not independent

- Computationally intensive
- Linearized approximation gives closed-form bias correction formulae, asymptotically equivalent to NIC

Performance Measures

Error Rate

$$\begin{aligned} \mathrm{pmc} &= \text{probability of misclassification} \\ &\approx \text{\% of errors on test set} \end{aligned}$$

Simple approach

- Just count errors on test set

— large samples needed, especially for accurate relative estimates of low probabilities

— variability on small samples is large $\Rightarrow$ smoothing needed (*i.e.* model based assumptions)

- It helps to bias test set towards rare/high-risk events

$$\widehat{\mathrm{pmc}} = \sum_k \widehat{\mathrm{pmc}}(k) \cdot \boldsymbol{p}(k)$$

Risk Averaging

- Estimate smoothed error rate using fitted posteriors

$$\boxed{\mathrm{pmc} = 1 - \mathrm{E}_{\boldsymbol{X}}\left[\max_k \boldsymbol{p}(k|\boldsymbol{X})\right]}$$

- Plug in estimates of $\boldsymbol{p}(k|\boldsymbol{X})$, integrate for expectation
- No test set needed
- Bias can be very large, especially for plug-in estimators

$\Rightarrow$ validate estimates of $\boldsymbol{p}(k|\boldsymbol{X})$ first, using log-likelihood or Brier scores over a test set

Cross Validation

Leave-one-out

- Given n observations, train classifier on each $(n-1)$ point subset and validate on remaining observation. Average results over all n subsets

$$\boxed{\mathrm{pmc} \approx \frac{\text{\# misclassifications}}{n}}$$

Variations

- Schemes for handling importance-weighted training sets
- Use coarser subdivision: $\mathcal{O}(\frac{1}{n})$ perturbations may be too small to accurately probe $\mathcal{O}(\frac{1}{\sqrt{n}})$ variability

— coarseness reduces overlap and variability, but increases bias as training sets are smaller

— *c.f.* traditional cross comparison of two halves of data

Confusion Matrices

- Tabulate counts/probabilities of response *vs.* true class
- Shows *which* classes are being confused, but not *where*
- Smoothed (risk averaged) versions also exist

Comparing Different Classifiers

- Use ***identical*** training and test sets to reduce the ***relative variability*** of estimates
- Make comparison using statistical tests for paired/blocked data (*e.g.* McNemar, ...)

The Jackknife

- A general ***bias reduction*** scheme — cancel asymptotic bias between full and leave-one-out estimates
- If n-point estimator $\hat{\theta}$ has bias

$$\boxed{\hat{\theta} - \theta_0 = \frac{\alpha}{n} + \mathcal{O}(\frac{1}{n^2})}$$

$\Rightarrow \tilde{\theta} =$ average of leave-one-out estimates has bias $\approx \frac{\alpha}{n-1}$

$\Rightarrow$ ***Jackknifed estimate*** has bias $\mathcal{O}(\frac{1}{n^2})$

$$\boxed{n\hat{\theta} - (n-1)\tilde{\theta} = \theta_0 + \mathcal{O}(\frac{1}{n^2})}$$

- Trades ***bias*** for ***variability:*** subtract two terms of $\mathcal{O}(n)$ ...

— but some reassuring theorems exist

- Jackknife estimator for pmc estimates bias using average leave-one-out (test − training) error

— essentially same result as leave-one-out cross-validation

$$\boxed{\text{error rate} \approx \frac{\text{\# misclassifications}}{n} + \mathcal{O}(\frac{1}{n^2})}$$

The Bootstrap

- General estimation scheme based on ***random resampling*** of training set (randomized version of cross-validation)

1. Choose n samples ***with replacement*** from training set

— some examples missing, some duplicated ...

2. Train classifier on sample, measure error on missed-out points

3. Average over many samplings

- *c.f.* systematic sampling: randomization increases variability but decreases bias (?)
- Many potential uses, but not yet clear how useful it is in practice

0.632 Estimator

- Samples are ***significantly overfitted*** owing to duplicates

$\Rightarrow$ average error ϵ_0 on missed-out points is also large

- 0.632 estimator asymptotically corrects for this

$$\mathrm{pmc} \approx 0.368\,\widehat{\mathrm{pmc}} + 0.632\,\epsilon_0$$

- $1 - \frac{1}{e} \approx 0.632$ is probability for point to be in training sample, $\widehat{\mathrm{pmc}}$ = observed pmc for training examples

Biases

- Many estimators are ***biased***

— *e.g.* M.L. finds ***modes*** not ***means***

- Typical bias is $\mathcal{O}(\frac{1}{n})$ — worst for small classes

— *e.g.* M.L. ***underestimates covariance*** Σ by $\mathcal{O}(\frac{1}{n})$

M.L. $\quad \mathbf{\Sigma} \approx \frac{1}{n}\sum_i (\boldsymbol{x}_i - \boldsymbol{\mu})^2$

debiased $\quad \mathbf{\Sigma} \approx \frac{1}{n-1}\sum_i (\boldsymbol{x}_i - \boldsymbol{\mu})^2$

$$\mathbf{\Sigma}^{-1} \approx \frac{n-p-1}{n}\left[\frac{1}{n-1}\sum_i (\boldsymbol{x}_i - \boldsymbol{\mu})^2\right]^{-1}$$

- Similar results hold for estimated ***densities***

Bias Correction

- First order corrections, *e.g.* $\frac{1}{n} \to \frac{1}{n-1}$ for M.L. $\mathbf{\Sigma}$
- But predictive approach gives ***significantly*** wider tails

— *e.g.* Gaussian becomes t-distribution

- ***Rare, high risk classes*** are especially susceptible

distant decision boundaries

+ underestimated $\mathbf{\Sigma}$ & tails

$\Rightarrow$ underestimated high risk region

- Reduce bias by ***importance weighting*** of high risk and improbable data

— often implicit in diagnostic training sets

Goodness-of-Fit Biases

- Model fitting minimizes ***error function*** h over parameters $\boldsymbol{\theta}$

— *e.g.* $-$ log likelihood

- Result is ***biased towards training set***

— good ***apparent fit*** but poor ***generalization***

- Bias makes it hard to compare errors of different models
- Estimated bias (see below)

$h(\text{unseen}) - h(\text{training}) \approx p^*/n^*$

p^* = "effective # parameters" active in fit

n^* = "effective # training points" constraining them

— *I*nactive $\Rightarrow$ all data points agree on value $\Rightarrow$ low bias

— Over-active $\Rightarrow$ data points disagree strongly $\Rightarrow$ high bias

- Models with ***many, rarely active parameters*** have large bias
- *c.f.* models with free parameters for each data point (*e.g.* geometric surface fitting): constant bias/point $\Rightarrow$ need to correct d.o.f. count

Estimating Implicit Functions

$$\boldsymbol{f}(\boldsymbol{X},\boldsymbol{\theta}) = \text{ p-vector of functions with p parameters } \boldsymbol{\theta}$$

$$\boldsymbol{f}_0(\boldsymbol{\theta}) \equiv \mathrm{E}_{\boldsymbol{X}}[\,\boldsymbol{f}(\boldsymbol{X},\boldsymbol{\theta})\,]$$

$$\hat{\boldsymbol{f}}(\boldsymbol{\theta}) \equiv \tfrac{1}{n}\textstyle\sum_{i=1}^{n}\boldsymbol{f}(\boldsymbol{x}_i,\boldsymbol{\theta}) \;\approx\; \boldsymbol{f}_0(\boldsymbol{\theta}) + \text{fluctuations}$$

$$\boldsymbol{\theta}_0 \text{ solving } \boldsymbol{f}_0(\boldsymbol{\theta}_0) = \boldsymbol{0}$$

$$\hat{\boldsymbol{\theta}} \text{ near } \boldsymbol{\theta}_0 \text{ solving } \hat{\boldsymbol{f}}(\hat{\boldsymbol{\theta}}) = \boldsymbol{0}$$

$$\boldsymbol{J} \equiv \tfrac{D\boldsymbol{f}_0}{D\boldsymbol{\theta}}(\boldsymbol{\theta}_0) \;\approx\; \tfrac{D\hat{\boldsymbol{f}}}{D\boldsymbol{\theta}}(\hat{\boldsymbol{\theta}})$$

$$\boldsymbol{K} \equiv \mathrm{Var}_{\boldsymbol{X}}[\,\boldsymbol{f}\,](\boldsymbol{\theta}_0) \;\approx\; \tfrac{1}{n-1}\textstyle\sum_{i=1}^{n}(\boldsymbol{f}\cdot\boldsymbol{f}^\top)(\boldsymbol{x}_i,\hat{\boldsymbol{\theta}})$$

These give the following asymptotic distributions:

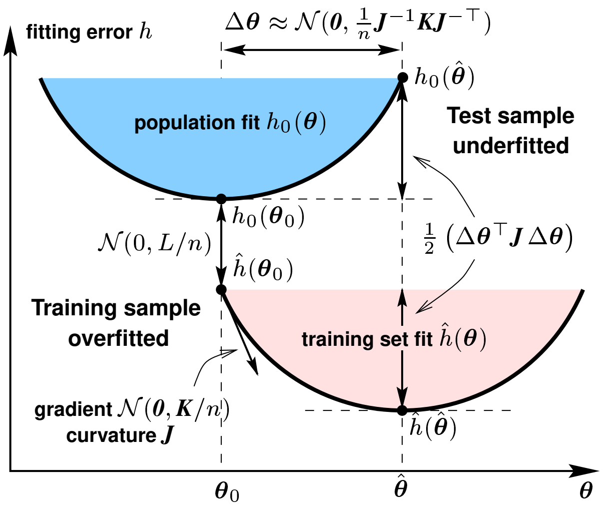
$$\hat{\boldsymbol{f}}(\boldsymbol{\theta}_0) \;\longrightarrow\; \mathcal{N}(\boldsymbol{0},\tfrac{1}{n}\boldsymbol{K})$$

$$\hat{\boldsymbol{\theta}} \;\longrightarrow\; \mathcal{N}(\boldsymbol{\theta}_0,\tfrac{1}{n}\boldsymbol{J}^{-1}\,\boldsymbol{K}\,\boldsymbol{J}^{-\top})$$

Proof

Central limits, independence of (functions of) $\boldsymbol{X}_i$, solve $\hat{\boldsymbol{f}}(\boldsymbol{\theta}) \approx \hat{\boldsymbol{f}}(\boldsymbol{\theta}_0) + \frac{D\hat{\boldsymbol{f}}}{D\boldsymbol{\theta}}\cdot\Delta\boldsymbol{\theta} = \boldsymbol{0}$ for $\Delta\boldsymbol{\theta} = (\boldsymbol{\theta}-\boldsymbol{\theta}_0)$.

Overfitting See-Saw



$\hat{\boldsymbol{\theta}}$ optimizes training set fit $\hat{h}(\boldsymbol{\theta}) \equiv \frac{1}{n}\sum_{i=1}^{n}h(\boldsymbol{x}_i,\boldsymbol{\theta})$

$\boldsymbol{\theta}_0$ optimizes population fit $h_0(\boldsymbol{\theta}) \equiv \mathrm{E}_{\boldsymbol{X}}[\,h(\boldsymbol{X},\boldsymbol{\theta})\,]$

$$\boldsymbol{J} \;\equiv\; \frac{D^2h_0}{D\boldsymbol{\theta}^2}(\boldsymbol{\theta}_0) \;\approx\; \frac{D^2\hat{h}}{D\boldsymbol{\theta}^2}(\hat{\boldsymbol{\theta}})$$

$$\boldsymbol{K} \;\equiv\; \mathrm{Var}\left[\,\frac{Dh}{D\boldsymbol{\theta}}\,\right](\boldsymbol{\theta}_0) \;\approx\; \frac{1}{n-1}\sum_{i=1}^{n}\left(\frac{Dh}{D\boldsymbol{\theta}}\cdot\frac{Dh}{D\boldsymbol{\theta}}^{\top}\right)(\boldsymbol{x}_i,\hat{\boldsymbol{\theta}})$$

$$L \;\equiv\; \mathrm{Var}[\,h\,](\boldsymbol{\theta}_0)$$

$$p^* \;\equiv\; \mathrm{trace}\left(\boldsymbol{J}^{-1}\boldsymbol{K}\right) \;=\; \text{“effective \# parameters”}$$

Asymptotic Distributions as $n \to \infty$

$$\Delta\boldsymbol{\theta} = \hat{\boldsymbol{\theta}} - \boldsymbol{\theta}_0 \longrightarrow \mathcal{N}(\boldsymbol{0},\tfrac{1}{n}\boldsymbol{J}^{-1}\,\boldsymbol{K}\,\boldsymbol{J}^{-\top})$$

$$\Delta\boldsymbol{\theta}^\top\boldsymbol{J}\,\Delta\boldsymbol{\theta} \longrightarrow p^*/n + \mathcal{O}(\sqrt{p^*/n}) \text{ fluctuations}$$

$$\sim \tfrac{1}{n}\chi^2_p \quad \text{if } \boldsymbol{J} = \boldsymbol{K}$$

- For ***true*** M.L. models $\boldsymbol{J} = \boldsymbol{K}$ and $p =$ true # parameters
- To $\mathcal{O}(n^{-3/2})$

$$\hat{h}(\boldsymbol{\theta}_0) - \hat{h}(\hat{\boldsymbol{\theta}}) = h_0(\hat{\boldsymbol{\theta}}) - h_0(\boldsymbol{\theta}_0) = \tfrac{1}{2}\Delta\boldsymbol{\theta}^\top\boldsymbol{J}\,\Delta\boldsymbol{\theta}$$

$$\hat{h}(\boldsymbol{\theta}_0) - h_0(\boldsymbol{\theta}_0) \sim \mathcal{N}(\boldsymbol{0},\tfrac{1}{n}L) \text{ (correlated with $\Delta\boldsymbol{\theta}$?)}$$

- Overfitting

$$h_0(\hat{\boldsymbol{\theta}}) - \hat{h}(\hat{\boldsymbol{\theta}}) \longrightarrow p^*/n + \mathcal{O}\left(\sqrt{(p^*+L)/n}\right)$$

$$\sim \tfrac{1}{n}(\chi^2_{p^*} + \mathcal{N}(0,L))$$

Proof

Above lemma with $\boldsymbol{f} = \frac{Dh}{D\boldsymbol{\theta}} = \boldsymbol{0}$ giving extremum of h. Expand h to 2^{nd} order at $\boldsymbol{\theta}_0$ for expectations and $\hat{\boldsymbol{\theta}}$ for sums (1^{st} order terms vanish).

Information criteria

- Bias corrections for ***goodness of fit***
- Often used for ***model selection***

Akaike Information Criterion (AIC)

$$\hat{h}(\text{unseen}) \approx \hat{h}(\text{training}) + p/n$$

- $p =$ # parameters, $n =$ # training points
- Assumes that fitted model is near true one and all parameters are active for all data points

Network Information Criterion (NIC)

$$\hat{h}(\text{unseen}) \approx \hat{h}(\text{training}) + p^*/n$$

- $p^* = \mathrm{trace}(\boldsymbol{J}^{-1}\boldsymbol{K})$ estimates ***mean # active parameters***
- Valid even if best model in family is false

Model Selection

Plug-In Approach

$\boxed{1}$ For each model estimate unbiased "quality" of fit

$\boxed{2}$ Use "best" model as if it were true one

- Any quality metric can be used

— *e.g.* log posterior likelihood

— ***performance metrics*** are best, *e.g.* total loss, pmc

- Remove overfitting bias to avoid choosing over-large models

— *e.g.* use cross-validation performance, AIC, NIC, ...

Predictive Approach

- ***Hedge over possible models*** rather than choosing just one

$$\boldsymbol{p}(k|\boldsymbol{x}) \approx \sum_m \boldsymbol{p}(k|\boldsymbol{x}, m) \cdot \boldsymbol{p}(m|\mathcal{T})$$

- Heuristically weighted averages can also be used
- Computationally expensive, but can significantly improve results
- Amounts to fitting a bigger/different parametric family ...

Complexity Penalties

- Can also ***explicitly penalize complex models***

— *e.g.* smoothing / regularization, computational convenience

- Can view penalty as a ***prior over models***

Minimum Description Length

- For any discrete distribution over models

$$\text{min.\# bits to encode choice of model} \;\approx\; -\log_2 \boldsymbol{p}(\text{model}|\mathcal{T})$$

- Turn this around and use it as a ***prior over models***

$$\boldsymbol{p}(\text{model}) \sim 2^{-(\text{\# bits to specify model})}$$

- Says that each family is roughly equally likely, and within a family $\boldsymbol{p}(\text{model}) \sim \frac{1}{\text{\# distinguishable models}}$
- In practice bit counting is usually ***very*** heuristic

Model selection using AIC / NIC

- Chosen model is asymptotically ***at least*** as big as true one
- Often slightly larger owing to statistical fluctuations in fit:

— absolute fitting errors have $\mathcal{O}(\sqrt{(L+p^*)/n})$ fluctuations

— relative errors for nested models have $\mathcal{O}(\sqrt{(\Delta L+\Delta p^*)/n})$ fluctuations

- If there is no true model in the family

— AIC tends to choose the ***biggest available one***

— NIC discounts unobservable parameters, so chooses a more reasonable one (asymptotically equivalent to cross-validation)

Searches over Model Spaces

- Exhaustive search seldom possible — too many d.o.f

$\Rightarrow$ Local, heuristic methods needed

— usually find a local extremum ... but hopefully a good one !

Incremental Methods

- Wander across space in small steps

— *e.g.* simulated annealing / Gibbs sampler

- Only requires pairwise model comparisons

— *e.g.* does adding or deleting this parameter help fit ?

Bayesian Information Criterion (BIC)

- Predictive approach to comparing ***families*** of models

— ***not*** individual models as for AIC, NIC

- Find posterior for family M by integrating out parameters $\boldsymbol{\theta}$

$$\begin{aligned}\Pr(M|\mathcal{T}) &= \int \boldsymbol{p}(M,\boldsymbol{\theta}|\mathcal{T})\,d\boldsymbol{\theta}\\ &\propto\; \boldsymbol{p}(M)\int \boldsymbol{p}(\mathcal{T}|M,\boldsymbol{\theta})\cdot\boldsymbol{p}(\boldsymbol{\theta}|M)\,d\boldsymbol{\theta}\end{aligned}$$

- "Other things being equal", wider distributions give larger results $\Rightarrow$ families with ***more uncertainty*** are preferred !
- Choice of prior is both difficult and critical

— relative probability of different families

— control variability of unobserved parameters

- Use ***saddle-point method*** to approximate $\Pr(M|\mathcal{T})$ integral
- For a weak prior with covariance $\boldsymbol{\Sigma}_0$ centred not too far from ML peak $\hat{\boldsymbol{\theta}}$, this gives

$$\begin{aligned}\log\Pr(M|\mathcal{T}) \approx \log\boldsymbol{p}(M) - L(\hat{\boldsymbol{\theta}},M|\mathcal{T})&\\ -\tfrac{1}{2}\log\left|\mathbf{I}+\boldsymbol{\Sigma}_0\cdot\tfrac{D^2L}{D\boldsymbol{\theta}^2}\right|&\end{aligned}$$

- $-L(\hat{\boldsymbol{\theta}},M|\mathcal{T})$ is the log likelihood at $\hat{\boldsymbol{\theta}}$
- $\frac{1}{2}\log|\cdot|$ is relative normalization of posterior *vs.* prior

Normal distributions — posterior inverse covariance is $\boldsymbol{\Sigma}^{-1} \equiv \boldsymbol{\Sigma}_0^{-1} + \frac{D^2L}{D\boldsymbol{\theta}^2}$

- $\frac{D^2L}{D\boldsymbol{\theta}^2}$ is $\mathcal{O}(n)$, so if it asymptotically dominates $\boldsymbol{\Sigma}_0$ over p^* dimensions ('active parameters') we might have $\frac{1}{2}\log|\cdot| \approx -\frac{p^*}{2}\log n + \ldots$

Bayesian Information Criterion

- Posterior likelihood of family M is

$$\log\Pr(M|\mathcal{T}) \approx \log\boldsymbol{p}(M) - L(\hat{\boldsymbol{\theta}},M|\mathcal{T}) - \frac{p^*}{2}\log n$$

Computational Learning Theory

- Studies algorithmic complexity of learning

Probably Almost Correct (PAC) Learning

- Provides ***probabilistic*** bounds on ***average estimator performance over all training sets***, for ***arbitrary population distributions*** within some class

— *e.g.* # training samples required to guarantee a certain pmc for almost all training sets

- ***nothing*** is guaranteed for ***specific*** training sets
- Main assumption: training & test distributions are identical
- Results depend strongly on ***assumed class of distributions***
- Current results are ***very conservative***

— mainly useful as general guidelines

— tightest results for 2 non-overlapping classes

PAC Learning

- Basic idea is that for large samples and well behaved distributions, almost all training sets are good models of true distribution

— *i.e.* within allowed class of distributions, distribution estimates converge uniformly

- ***Vapnik-Chervonenkis dimension*** of class characterizes complexity of decision boundaries
- PAC-learnable $\Longleftrightarrow$ data can be ***summarized succinctly***

— "Ockham algorithm" (summarizer) exists

Vapnik-Chervonenkis Dimension

- Measures # parameters to define decision region within family

— *i.e.* maximum number of points which can be assigned arbitrary function values by family

- *e.g.* for families of convex regions, value is predetermined for points lying inside convex hull of some other points
- In d dimensions have following VC dimensions (add one if sidedness not predefined)

— half spaces: d

— spheres: $d+1$

— ellipsoids: $d(d+1)/2$

— aligned boxes: $2d$

— convex families: # parameters to define region of family

— ***non-***convex families: VC dimension can be large . . .

Principal Components & Discriminants

- Often want to summarize high dimensional data with a low dimensional projection "plus noise" — *e.g.* images
- Linear methods find best projection directions by ***orthogonal decomposition*** (SVD, QR) of ***centred data*** or ***scatter*** matrices

Principal components

- Directions of maximum ***population variability***
- Decompose centred data matrix or total covariance

$$\mathbf{A} \equiv [\boldsymbol{x}_1 - \boldsymbol{\mu}, \ldots, \boldsymbol{x}_n - \boldsymbol{\mu}] \qquad \boldsymbol{\mu} = \tfrac{1}{n}\textstyle\sum_i \boldsymbol{x}_i$$
$$\boldsymbol{\Sigma}_T = \tfrac{1}{n-1}\mathbf{A}\ \mathbf{A}^\top = \tfrac{1}{n-1}\textstyle\sum_i (\boldsymbol{x}_i - \boldsymbol{\mu})(\boldsymbol{x}_i - \boldsymbol{\mu})^\top$$

Principal Linear Discriminants

- Directions of maximum ***inter-class separation***
- Maximize projected ***between class*** to ***within class covariance***

$$\begin{aligned}\boldsymbol{\Sigma}_W &= \tfrac{1}{n-K}\textstyle\sum_{k,\boldsymbol{x}_i\in k}(\boldsymbol{x}_i - \boldsymbol{\mu}_k)(\boldsymbol{x}_i - \boldsymbol{\mu}_k)^\top \\ \boldsymbol{\Sigma}_B &= \tfrac{1}{n-K}\textstyle\sum_k n_k(\boldsymbol{\mu}_k - \boldsymbol{\mu})(\boldsymbol{\mu}_k - \boldsymbol{\mu})^\top \\ &= \tfrac{n-1}{n-K}\boldsymbol{\Sigma}_T - \boldsymbol{\Sigma}_W\end{aligned}$$

- Decompose $\boldsymbol{\Sigma}_W$-weighted class mean matrix $\mathbf{M}$ or scatter $\mathbf{M}\mathbf{M}^\top$

$$\mathbf{M} = \boldsymbol{\Sigma}_W^{-1/2}\,[\boldsymbol{\mu}_1 - \boldsymbol{\mu}, \ldots, \boldsymbol{\mu}_k - \boldsymbol{\mu}]\ \ \mathbf{T}$$

- $\mathbf{T}$ is an arbitrary weight matrix

— traditional choice $\mathrm{diag}(\sqrt{n_k/(n-K)})$ amounts to PCA of $\mathbf{M}\mathbf{M}^\top = \boldsymbol{\Sigma}_W^{-1/2}\boldsymbol{\Sigma}_B\boldsymbol{\Sigma}_W^{-1/2}$

— might also use $\mathbf{T} = \mathbf{I}$ since $\boldsymbol{\Sigma}_W^{-1}(\boldsymbol{\mu}_k - \boldsymbol{\mu}_l)$ is what appears in linear discriminant

- $\boldsymbol{\Sigma}_W$ often nearly singular

— use PCA or truncated SVD to find $\boldsymbol{\Sigma}_W^{-1/2}$