Basics

General

Cauchy-Schwartz Inequality: $|u \cdot v| \leq \|u\|\|v\|$

Probability Basics

$\Pr[a \leq X \leq b] = \int_a^b f_X(x)\,dx$
Cumulative distribution function: $F_X(x) = \mathrm{P}(X \leq x) = \int_{-\infty}^{x} f_X(t)\,dt$
Bayes: $p(\theta|x) = \frac{p(x|\theta)p(\theta)}{p(x)} = p(\text{model}|\text{data}) = \frac{p(\text{data}|\text{model})p(\text{model})}{p(\text{data})}$
Posterior probability $\propto$ Likelihood $\times$ Prior probability
Markov inequality $\mathbb{P}(g(X) \geq r) \leq \dfrac{\mathbb{E}[g(X)]}{r}$
Chebyshev inequality $\mathbb{P}(|X-\mu| \geq \epsilon) \leq \frac{\sigma^2}{\epsilon^2}$
Jensen inequality $\varphi\left(\mathrm{E}[X]\right) \leq \mathrm{E}\left[\varphi(X)\right]$ where φ is a *convex* function. E.g. $log(x)$ if $x > 0$
Law of large numbers $\mathbb{P}(|\frac{S_n}{n} - \mu| \geq \epsilon) \leq \frac{\sigma^2}{n \cdot \epsilon^2} \to 0$ if $n \to 0$
Bound on disjunction: $\max \mathbb{P}(A_1), \ldots, \mathbb{P}(A_n) \leq \mathbb{P}(E) \leq \min 1, \mathbb{P}(A_1) + \cdots + \mathbb{P}(A_n)$
Glivenko-Catelli theorem: $\mathbb{P}(\sup_x |F_n(x) - F(x)| \to 0) = 1$ if $n \to 0$.

Maximization Basics

Calculus of Variations

Goal: find the maximum of a functional.
$J(x + \delta x) - J(x) = 0$
Given: functional of form $J[f] = \int_a^b L[x, f(x), f'(x)]dx$
change $\delta J = \int_a^b \frac{\delta J}{\delta f(x)} \delta f(x) dx$
functional derivative: $\frac{\delta J}{\delta f(x)} = \frac{\partial L}{\partial f} - \frac{d}{dx}\frac{\partial L}{\partial f'}$

EM-Algorithm

given: $p(\mathbf{X}, \mathbf{Z}|\theta)$ with observed variables $\mathbf{X}$ and latent variables $\mathbf{Z}$, parameters θ *goal:* maximize likelihood function $p(\mathbf{X}|\theta)$ wrt. θ.

1. choose initial setting for $\boldsymbol{\theta}^{old}$
2. **E-step:** evaluate $p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta}^{old})$
3. **M-step:** $\boldsymbol{\theta}^{new}) = \text{argmax}_\theta \mathcal{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^{old})$ where $\mathcal{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^{old}) = \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta}^{old}) ln(p(\mathbf{X}|\boldsymbol{\theta}))$
4. check if converged. Otherwise set $\boldsymbol{\theta}^{old} \leftarrow \boldsymbol{\theta}^{new}$

Information Theory

Mutual information:
measures the amount of information that can be obtained about one random variable by observing another
$I(X;Y) = \mathbb{E}_{X,Y}[SI(x,y)] = \sum_{x,y} p(x,y) \log \frac{p(x,y)}{p(x)\,p(y)}$
$I(X;Y) = H(X) - H(X|Y)$.
$I(X;Y) = I(Y;X) = H(X) + H(Y) - H(X,Y)$.
chain rule for information: $I(X;Y,Z) = I(X;Z) + I(X;Y|Z)$
$I(X;Y) = \mathbb{E}_{p(y)}[D_{\mathrm{KL}}(p(X|Y=y)\|p(X))]$.

Kullback-Leibler divergence

- quantifies coding cost describing data with probability distribution q when true distribution is p.
- KL divergence is positive semidefinite. $D_{\mathrm{KL}} \geq 0$

way of comparing two distributions: a "true"probability distribution $p(X)$, and an arbitrary probability distribution $q(X)$.
or: ünnecessary surprise". **not** symmetric!
q(X) is the distribution underlying some data, when, in reality, p(X) is the correct distribution, the Kullback–Leibler divergence is the number of average additional bits per datum necessary for compression.
$D_{\mathrm{KL}}(p(X)\|q(X)) = \sum_{x \in X} -p(x)\log q(x) - \sum_{x \in X} -p(x) \log p(x) = \sum_{x \in X} p(x) \log \frac{p(x)}{q(x)}$
continuous case: $D_{\mathrm{KL}}(P\|Q) = \int_{-\infty}^{\infty} p(x) \log \frac{p(x)}{q(x)} dx$
discrete case : $D_{\mathrm{KL}}(P\|Q) = \sum_i P(i) \log \frac{P(i)}{Q(i)}$

Dimensionality Reduction

Principal Component Analysis (PCA)

Idea

dimensionality reduction with orthogonal projection of D-dimensional data onto lower M-dimensional linear space (*principal subspace*). Variance is maximized.

Procedure

1. calculate mean $\bar{x} = \frac{1}{N}\sum_{n=1}^{N} x_n$ and covariance matrix **S** of data set
2. find the M eigenvectors corresponding to the M largest eigenvalues.
3. data projected on these eigenvectors have largest variance.

Probabilistic Principal Component Analysis (PPCA)

Idea

$\mathbf{x} = \mathbf{W}\mathbf{z} + \boldsymbol{\mu} + \boldsymbol{\epsilon}$, with $p(\mathbf{z}) = \mathcal{N}(\mathbf{z}|\mathbf{0}, \mathbf{I})$ and $\epsilon\ \mathcal{N}(0, \sigma^2\mathbf{I})$
$p(\mathbf{x}|\mathbf{z}) = \mathcal{N}(\mathbf{x}|\mathbf{W}\mathbf{z} + \boldsymbol{\mu}, \sigma^2\mathbf{I})$

Procedure

1. calculate mean $\bar{x} = \frac{1}{N}\sum_{n=1}^{N} x_n$ and covariance matrix **S** of data set
2. find the M eigenvectors corresponding to the M largest eigenvalues.
3. data projected on these eigenvectors have largest variance.

Advantages

- computationally efficient → no need to evaluate covariance matrix as intermediate step
- allows us to deal with missing data in data set

Locally Linear Embedding

N real valued vector $\vec{X}_i$ of dimension D.
Assumption: data lies close on locally linear patch of manifold.
Construct each data point form its K nearest (e.g. euclidean distance) neighbors.
Reconstruction error: $\mathcal{E}(W) = \sum_i |\vec{X}_i - \sum_j W_{ij}\vec{X}_j|$
where W_{ij} summarize the contribution of the jth data point to the ith reconstruction.
LLE Algorithm:

1. Compute the neighbors of each data point, $\vec{X}_i$.
2. Compute the weights W_{ij} that best reconstruct each data point $\vec{X}_i$ from its neighbors, minimizing the cost $\mathcal{E}(W)$ by constrained linear fits.
3. Compute the vectors $\vec{Y}_i$ best reconstructed by the weights W_{ij}, minimizing the quadratic form in $\Phi(Y) = \sum_i |\vec{X}_i - \sum_j W_{ij}\vec{Y}_j|^2$ by its bottom nonzero eigenvectors.

Maximum Entropy Inference

Maximizing entropy yields least biased inference method → maximally noncommittal wrt. missing data.

Maximum entropy distribution

Idea

Outcomes $\omega_i \in \Omega$, determine probabilities $p_i = \mathbb{P}(\omega_i)$. Given constraints/moments $\mu_i = \mathbb{E}[r_j] = \sum_{i=1}^{n} r_j(\omega_i)p_i$, where $r_j : \Omega \to \mathbb{R} j = 1, 2, ..., n$ are different functions defined over Ω.

Gibbs distribution

minimal sesitivity to changes in constraint moments μ_i
$p(x) = \frac{1}{Z}\text{exp}(-\sum_{j=1}^{m} \lambda_j r_j(x))$ where m is the number of constraints.

$Z = \int_x \text{exp}(-\sum_{j=1}^{m} \lambda_j r_j(x))dx$

Derive least sensitive distributions

1. determine all r_j from constraints.
2. calculate Z and $p(x)$ from Gibbs distribution
3. calculate all moments and set equal to constraints.

For a general cost function:

$$\mathbf{P}(\mathbf{c}) = \frac{e^{-\beta\mathcal{R}(c)}}{\sum_{c' \in \mathcal{C} e^{-\beta\mathcal{R}(c')}}} = \frac{e^{-\beta\mathcal{R}(c)}}{Z} = \text{exp}(-\beta(\mathcal{R}(c) - \mathcal{F}))$$

where $\mathcal{F} = -\frac{1}{\beta}\text{log}(Z)$ is known as „free energy“.
Free energy for any distribution: $\mathcal{F} = -\frac{1}{\beta}S(\mathbf{P}) + \mathbb{E}_{\mathbf{P}}\mathcal{R}$

Entropy

H_X of a discrete random variable X is a measure of the amount of *uncertainty* associated with the value of X when only its distribution is known
$H(X) = \mathbb{E}_X[I(x)] = -\sum_{x \in \mathbb{X}} p(x) \log p(x)$
Chain rule: $H(X,Y) = H(X) + H(Y|X)$

Maximum Entropy Clustering

Idea

Determine probabilistic centroids y_α and probabilistic assignments $P_{i\alpha}$ of i-th object to cluster α. Data $\mathcal{X}$ and labelings c are r.v.

Procedure

1. Define posterior probability distribution $P(c|\mathcal{X}, \mathcal{Y})$, $c \in \mathcal{C}$, constraint $\mathbb{E}_{P(c|\mathcal{X},\mathcal{Y})}\mathcal{R}(c, \mathcal{X}, \mathcal{Y}) = \mu$ where μ is a constant.
2. find centroid conditions and assignments $P_{i\alpha}$ by maximizing entropy wrt. $\mathcal{Y} \to \frac{\partial}{\partial \mathbf{y}_\alpha} H(P(c|\mathcal{X}, \mathcal{Y})) = \mathbf{0}$

Most agnostic $P(c|\mathcal{X}, \mathcal{Y})$ is Gibbs distribution.

$$P(c|\mathcal{X}, \mathcal{Y}) = \frac{\text{exp}(-\mathcal{R}(c, \mathcal{X}, \mathcal{Y})/T)}{\sum_{c' \in \mathcal{C}} \text{exp}(\mathcal{R}(c', \mathcal{X}, \mathcal{Y})/T)}$$

Remark: If the cost function is linear in individual costs, posterior can be factorized!

Clustering distributional data

Idea

„Close“ objects have similar feature distributions. → cluster via similarity.
Distributional data encoded in dyads (pair $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$): n different objects $\{x_i\} \in \mathcal{X}$, m possible feature values $\{y_i\} \in \mathcal{Y}$. Data is set of l observations $\mathcal{Z} = \{(x_{i(r)}, y_{j(r)})\}_{r=1}^{l}$.

Likelihood of data $\mathcal{Z}$ $\mathcal{L}(\mathcal{Z}) = \Pi_{i \leq n}\Pi_{j \leq m} P((i,j)|c(i), Q)^{l\hat{P}(i)}$
where $l\hat{P}(i) = \#$ object i exhibits value j and $\hat{P}(i)$ = empirical probability finding object i with feature value j.

Procedure

1.

Parametric distributional clustering

Density estimation is assumed to be Gaussian $g_a(y)$.
$p(y|\nu) = \sum_{a \leq s} g_a(y)p(a|\nu)$ where $p(a|\nu)$ mixture coefficients, s number of mixture components.
Remark: Often feature values are restricted to specific domains, mixture components → rectified.

Markov Chain Monte Carlo (MCMC)

Idea: sampling from a probability distribution by constructing a Markov chain that has the desired distribution as its equilibrium distribution.

Markov Chain

$P_N(x_1, ..., x_N) = p_1(x1)\prod_{t=1}^{N-1} w(x_t \to x_{t+1})$
where $\{p_1(x)\}_{x \in \mathcal{X}}$ is the initial state and $\{w(x \to y)\}_{x,y \in \mathcal{X}}$ are the transition probabilities
transition probabilities must be non-negative and normalized: $\sum_{y\ in \mathcal{X}} w(x \to y) = 1,$ for any $y \in \mathcal{X}$ *Eigenvalues:*
First eigenvalue is always equal to 1. The smaller λ_2 the faster the MCMC converges.
irreducible: It is possible to get to any state from any state
Detailed balance: $\pi_i P_{ij} = \pi_j P_{ji}$ or $p(x)(x \to y) = p(y)p(y \to x)$, implies stationarity!
Stationarity: $\pi\mathbf{P} = \pi$
Aperiodic: Chain should not get trapped in cycles

Metropolis-Hastings

Initialization: Choose an arbitrary point x_0 as first sample, choose arbitrary probability density $g(x|y)$ that suggests a candidate for the next sample value x. For Metropolis algorithm, g must be symmetric → $g(x|y) = g(y|x)$ e.g. a Gaussian distribution centered at y, → points closer to y more likely to be visited next → sequence of samples → random walk. function g → proposal density or jumping distribution.
Algorithm:

1. Generate a new sample x' from the old x according to a proposal PDF: $x' \sim q(\,|\,x)$
2. Calculate acceptance probability $A(x', x)$ according to quotient $Q(x', x) = \frac{p(x')q(x|x')}{p(x)q(x'|x)}$ $A(x', x) = min(1, Q(x', x))$
3. Accept x' as new sample with with probability $A(x', x)$, else keep x.

Disadvantages:

- samples are correlated. For independent samples one would have to take only every *n*-th sample.
- initial samples may follow a very different distribution. Solution: throw away first 1000 samples.
- for high dimensions: finding the right jumping distribution can be difficult, as the different individual dimensions behave in very different ways, and the jumping width must be "just right"for all dimensions at once to avoid excessively slow mixing → Gibbs Sampling.

Gibbs Sampling

given a multivariate distribution it is simpler to sample from a conditional distribution than to marginalize by integrating over a joint distribution.
Applicable if:

- conditional distribution of each variable is known and is easy (or at least, easier than joint distribution) to sample from
-

simulated annealing is often used to reduce the "random walk"behavior in the early part of the sampling process

Heat-Bath Algorithm

Simulated Annealing

Idea

- Markov process samples solution from solution space Ω
- cost function $\mathcal{H} : \Omega \to \mathbb{R}, \omega \in \Omega$ denotes admissible solution.
- solutions accepted/rejected according to Metropolis algorithm: decreased cost → accepted, increased cost → accepted with probability $exp(-\Delta\mathcal{H}/T)$ where $\Delta\mathcal{H} = \mathcal{H}(\omega_{new}) - \mathcal{H}(\omega_{old})$. T is the computational temperature.
- reduce gradually the temperature during search proces → force system into solution with low costs. → random walk in solution space.

Markov process with converges to equilibrium:
$\mathbf{P}^{\mathbf{Gb}}(\omega) = \text{exp}(-(\mathcal{H}(\omega) - \mathcal{F}(\mathcal{H}))/T)$
where $\mathcal{F}(\mathcal{H})) = -Tlog\sum_{\omega' \in \Omega} \text{exp}(-\mathcal{H}(\omega')/T)$

Remarks

- Temperature formally ≈ Lagrange parameter → constraint on expected costs. $\langle H \rangle = \sum_{\omega \in \Omega} P^{Gb}(\omega)\mathcal{H}(\omega)$
- Gibbs free energy related to expected cost via entropy $\mathcal{S}(P^{Gb}) = -\sum_{\omega \in \Omega} P^{Gb}(\omega)\text{log}(P^{Gb}(\omega)) = \frac{1}{T}\langle H \rangle - \frac{1}{T}\mathcal{F}(\mathcal{H})$

Deterministic Annealing

Idea

deterministic variant of *Simulated annealing*.

Information Bottleneck Method

Idea

input signal X should be efficiently encoded by e.g. cluster variable $\tilde{X}$ preserving relevant information about context variable Y as good as possible.
$I(X;\tilde{X}) - \lambda I(\tilde{X}, Y)$

Parametric Distributional Clustering

Idea

cluster set of n objects $\mathbf{o}_i$ in k groups. Assignments encoded in $M_{i,\nu}$, $M_{i,\nu} = 1$ if $\mathbf{o}_i$ assigned to cluster ν. Enforce $\sum_{\nu \leq k} M_{i,\nu} = 1$.

Graph based clustering

relations among objects not necessarily metric.

Idea

$\mathcal{O}$ is the set of vertices $\mathcal{V}$, set of edges is inferred, set of (di)similarity measures $\mathcal{D} = \{D_{ij}\}$ or $\mathcal{S} = \{S_{ij}\}$ are the weights.
Cluster defined as: $\mathcal{G}_\alpha\{\mathbf{o} \in \mathcal{O} : c(\mathbf{O}) = \alpha\}$

Correlation Clustering

Idea

agreement within cluster, disagreement between clusters maximized.
Script p. 30

Pairwise Data Clustering

cost function:

$$\mathcal{R}^{pc}(c, \mathcal{D}) = \frac{1}{2}\sum_{\nu \leq k}\left(|\mathcal{G}_\nu| \sum_{(i,j)} \in \epsilon_{\nu\nu} \frac{D_{ij}}{|\epsilon_{\nu\nu}|}\right)$$

Remarks:

- dissimilarity matrix $\mathcal{D}_{ij}$ only zero for self-dissimilarity entries, $|\mathcal{G}_\nu|, |\epsilon_{\nu\nu}|$ cardinality of cluster/edges.
- cost function invariant under symmetrization and constant/additive shifts of non-diagonal elements
- metric data can always be transformed to PC data but not always vice versa

⇒ embed given nonmetric proximity data problem in vector space without changing underlying properties. → see CSE.

Constant Shift Embedding (CSE)

Embed pairwise clustering problem into vector space.

CSE algorithm

1. given dissimilarity matrix D, calculate centralized version $D^c = QDQ$ where $Q = I_n - \frac{1}{n}e_n e_n^T$
2. calculate matrix $S^c = -1/2D^c$
3. calculate eigenvalues of S^c. If S^c isn't positive semidefinite, $\tilde{S} = S^c - \lambda_n \cdot I_n$ (subtract smallest eigenvalue from diagonal elements)
4.

Remarks:

- embedding validity shows equivalence to k-means clustering, ideas of centroids and cluster representatives can be used
- pairwise data can be denoised when transformed into vectorial data. (e.g. in preprocessing)
- minimization processes pairwise cost function or k-means problem is $\mathcal{NP}$-hard → algorithms like deterministic annealing and mean field approximation needed. For $\mathcal{R}^{km}$ it's exact, for $\mathcal{R}^{pc}$ remains an approximation.

Cut

Partitioning a graph $G(V, E)$ with nodes V and edges E into disjoint sets A, B → removing edges connecting parts. Degree of dissimilarity is computed as total weight of removed edges:
$cut(A, B) = \sum_{u \in A, v \in B} w(u, v)$
Minimum cut: optimal bipartitioning of graph that minimizes cut value.
Note: Minimum cut favors cutting small sets of isolated nodes in graph. → use normalized cut.

Normalized Cut

$Ncut(A, B) = \frac{cut(A,B)}{assoc(A,V)} + \frac{cut(A,B)}{assoc(B,V)}$ where $assoc(A, V) = \sum_{u \in A, t \in V} w(u, t)$ *Ncut(A,B)* can be seen as the disassociation between two groups.
measure for total normalized association within groups for a given partition: $Nassoc(A, B) = \frac{assoc(A,A)}{assoc(A,V)} + \frac{assoc(B,B)}{assoc(B,V)}$ where $assoc(A, A)$ are the total weights of edges connecting nodes within A, B respectively. Measures how tightly on average nodes are connected within group.
$Ncut(A, B) = 2 - Nassoc(A, B)$

Computing Optimal Partition

Grouping algorithm

Mean Field Approximation

Mean Field Theory

Approximate Gibbs distribution by neglecting correlations between stochastic variables → determine „most similar“ factorized distribution. Only consider factorized distributions: $q(\mathbf{Z}) = \prod_{i=1}^{M} q_i(\mathbf{Z}_i)$
Then minimize the Kullback-Leibler divergence to obtain best factorial approximation.
Mean field $h_{u\alpha}$ is the expected cost $\mathcal{R}(c)$ that object u assigned to cluster α

Determine mean field

1. split cost function $\mathcal{R}$ into terms that contain the object u and other terms. Term has a form like $\mathcal{R}(c) = f(u) + \mathcal{R}(c|u)$
2. take the expected value $\mathbb{E}_{\mathbf{Q}_{u \ to \alpha}}$

- chain rule for information