Variable selection and feature construction using methods related to information theory

Kari Torkkola[1]

[1] Intelligent Systems Lab, Motorola, Tempe, AZ

IJCNN 2007

Outline

1. Information Theory
   - Definitions
   - Mutual Information and Communication Channels
   - Mutual Information in Practice
2. Mutual Information and classification problems
   - Class Separability Measures
   - The Bayes Error
   - MI in Variable Selection
3. Feature Transforms based on Mutual Information
   - Maximizing Mutual Information
   - Illustrations
   - Nonlinear Transforms
   - Reducing computation
4. Further uses for Information Theoretic concepts
   - Learning Distance Metrics
   - Information Bottleneck
5. Conclusion

Outline

Outline

Outline

Outline

Why Information Theory?

- Variables or features can be understood as a “noisy channel” that conveys information about the message
- The aim would be to select or to construct features that provide as much information as possible about the “message”
- By using information theory, variable selection and feature construction can be viewed as coding and distortion problems
- Read Shannon!

Outline

Entropy

- Continuous random variable $X \in R^d$ representing available variables or observations and a discrete-valued random variable Y representing the class labels
- The uncertainty or entropy in drawing one sample of Y at random according to Shannon's definition:

$$H(Y) = E_y[\log_2 \frac{1}{p(y)}] = -\sum_y p(y) \log_2(p(y)). \quad (1)$$

- (Differential) entropy can also be written for a continuous variable as

$$H(X) = E_x[\log_2 \frac{1}{p(\mathbf{x})}] = -\int_x p(\mathbf{x}) \log_2(p(\mathbf{x})) d\mathbf{x}. \quad (2)$$

Entropy

- Continuous random variable $X \in R^d$ representing available variables or observations and a discrete-valued random variable Y representing the class labels
- The uncertainty or entropy in drawing one sample of Y at random according to Shannon's definition:

$$H(Y) = E_y[\log_2 \frac{1}{p(y)}] = -\sum_y p(y) \log_2(p(y)). \quad (1)$$

- (Differential) entropy can also be written for a continuous variable as

$$H(X) = E_x[\log_2 \frac{1}{p(\mathbf{x})}] = -\int_x p(\mathbf{x}) \log_2(p(\mathbf{x}))d\mathbf{x}. \quad (2)$$

Entropy

- Continuous random variable $X \in R^d$ representing available variables or observations and a discrete-valued random variable Y representing the class labels
- The uncertainty or entropy in drawing one sample of Y at random according to Shannon's definition:

$$H(Y) = E_y[\log_2 \frac{1}{p(y)}] = -\sum_y p(y)\log_2(p(y)). \qquad (1)$$

- (Differential) entropy can also be written for a continuous variable as

$$H(X) = E_{\mathbf{x}}[\log_2 \frac{1}{p(\mathbf{x})}] = -\int_{\mathbf{x}} p(\mathbf{x})\log_2(p(\mathbf{x}))d\mathbf{x}. \qquad (2)$$

Conditional Entropy, Mutual Information

- After having made an observation of a variable vector $\boldsymbol{x}$, the uncertainty of the class identity is defined in terms of the conditional density $p(y|\boldsymbol{x})$:

$$H(Y|X) = \int_{\boldsymbol{x}} p(\boldsymbol{x}) \left(-\sum_y p(y|\boldsymbol{x}) \log_2(p(y|\boldsymbol{x})) \right) d\boldsymbol{x}. \quad (3)$$

- Reduction in class uncertainty after having observed the variable vector $\boldsymbol{x}$ is called the mutual information between X and Y

$$I(Y, X) = H(Y) - H(Y|X) \quad (4)$$

$$= \sum_y \int_{\boldsymbol{x}} p(y, \boldsymbol{x}) \log_2 \frac{p(y, \boldsymbol{x})}{p(y)p(\boldsymbol{x})} d\boldsymbol{x} \quad (5)$$

Same as Kullback-Leibler divergence between the joint density $p(y, \boldsymbol{x})$ and its factored form $p(y)p(\boldsymbol{x})$.

Conditional Entropy, Mutual Information

- After having made an observation of a variable vector $\boldsymbol{x}$, the uncertainty of the class identity is defined in terms of the conditional density $p(y|\boldsymbol{x})$:

$$H(Y|X) = \int_{\boldsymbol{x}} p(\boldsymbol{x}) \left(- \sum_{y} p(y|\boldsymbol{x}) \log_2(p(y|\boldsymbol{x})) \right) d\boldsymbol{x}. \qquad (3)$$

- Reduction in class uncertainty after having observed the variable vector $\boldsymbol{x}$ is called the mutual information between X and Y

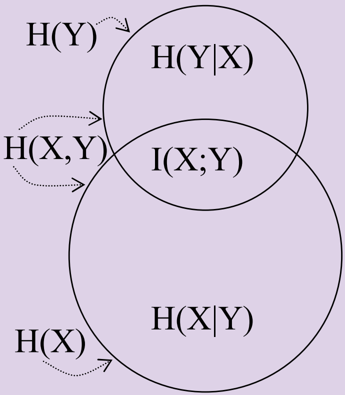
$$\begin{aligned} I(Y,X) &= H(Y) - H(Y|X) & (4) \\ &= \sum_{y} \int_{\boldsymbol{x}} p(y,\boldsymbol{x}) \log_2 \frac{p(y,\boldsymbol{x})}{p(y)p(\boldsymbol{x})} d\boldsymbol{x} & (5) \end{aligned}$$

Same as Kullback-Leibler divergence between the joint density $p(y,\boldsymbol{x})$ and its factored form $p(y)p(\boldsymbol{x})$.

Conditional Entropy, Mutual Information

- $H(X)$ and $H(Y)$ are each represented by a circle
- Joint entropy $H(X, Y)$ consists of the union of the circles
- Mutual information $I(X, Y)$ is the intersection of the circles
- $H(X, Y) = H(X) + H(Y) - I(X; Y)$

Illustration of entropies



Outline

Channel Coding

Shannon:

- Channel with input X and output Y'
- Rate of transmission of information $R = H(X) - H(X|Y') = I(X, Y')$
- The capacity of this particular (fixed) channel is defined as the maximum rate over all possible input distributions, $C = \max_{p(X)} R$
- Maximizing the rate $=$ choosing an input distribution that matches the channel (under some constraints, such as fixed power or efficiency of the channel)

Analogy to variable selection and feature construction

- Real source Y is now represented (encoded) as the available variables X
- Now the channel input distribution X is fixed
- Modify how the input is communicated to the receiver by the channel either by selecting a subset of available variables or by constructing new features $\Phi = g(X, \theta)$ where g denotes some selection or construction function, and θ represents some tunable parameters
- In Shannon's case θ was fixed but X was subject to change
- "channel" capacity can be represented as $C = \max_\theta R$ subject to some constraints, such as keeping the dimensionality of the new feature representation as a small constant

Rate-distortion theorem

- Finding the simplest representation (in terms of bits/sec) to a continuous source signal within a given tolerable upper limit of distortion
- Would not waste the channel capacity
- Solution for a given distortion D is the representation Φ that minimizes the rate $R(D) = \min_{E(d) \leq D} I(X, \Phi)$
- Combination of the two results in a loss function

$$\mathcal{L}(p(\phi|x)) = I(X, \Phi) - \beta I(\Phi, Y). \quad (6)$$

that does not require setting constraints to the dimensionality of the representation, rather it emerges as the solution
- The representation Φ can be seen as a bottleneck that extracts relevant information about Y from X (Tishby 1999)

Rate-distortion theorem

- Finding the simplest representation (in terms of bits/sec) to a continuous source signal within a given tolerable upper limit of distortion
- Would not waste the channel capacity
- Solution for a given distortion D is the representation Φ that minimizes the rate $R(D) = \min_{E(d) \leq D} I(X, \Phi)$
- Combination of the two results in a loss function

$$\mathcal{L}(p(\phi|x)) = I(X, \Phi) - \beta I(\Phi, Y). \tag{6}$$

  that does not require setting constraints to the dimensionality of the representation, rather it emerges as the solution
- The representation Φ can be seen as a bottleneck that extracts relevant information about Y from X (Tishby 1999)

Outline

Estimating Mutual Information

- Between two variables use non-parametric histogram approach (Battiti 94), but in higher dimensions any amount of data is too sparse to bin.
- Parametric class density estimates (such as Gaussians) and plug them into the definition of MI
- MI is a difference between two entropies: Entropy estimation!
  - The simplest way is the maximum likelihood estimate based on histograms
  - known to have a negative bias that can be corrected to some extent by the so-called Miller-Madow bias correction. This consists of adding $(\hat{m} - 1)/2N$ to the estimate, where $\hat{m}$ denotes an estimate of the number of bins with nonzero probability
  - this cannot be done in many practical cases, such as when the number of bins is close to the number of observations (Paninski 93)
  - Bayesian techniques can be used if some information about the underlying probability density function is available in terms of a prior (Wolpert & Wolf 95; Zaffalon 02)

Measures other than Shannon's

- Shannon derived the entropy measure axiomatically and showed that no other measure would fulfill all the axioms
- If we want to find a distribution that minimizes/maximizes the entropy or divergence, the axioms used in deriving the measure can be relaxed and still the result of the optimization is the same distribution (Kapur, 1994)
- One example is the Renyi entropy, which is defined for a discrete variable Y and for a continuous variable X as

$$H_\alpha(Y) = \frac{1}{1-\alpha} \log_2 \sum_y p(y)^\alpha; \qquad H_\alpha(X) = \frac{1}{1-\alpha} \log_2 \int_{\mathbf{x}} p(\mathbf{x})^\alpha d\mathbf{x}, \tag{7}$$

  where $\alpha > 0$, $\alpha \neq 1$, and $\lim_{\alpha \to 1} H_\alpha = H$
- Quadratic Renyi entropy is straightforward to estimate from a set of samples using the Parzen window approach

Non-Parametric Estimation of Renyi Entropy

Make use of the fact, that a convolution of two Gaussians is a Gaussian, that is,

$$\int_{\mathbf{y}} G(\mathbf{y}-\mathbf{a}_i,\Sigma_1)G(\mathbf{y}-\mathbf{a}_j,\Sigma_2)d\mathbf{y} = G(\mathbf{a}_i-\mathbf{a}_j,\Sigma_1+\Sigma_2). \quad (8)$$

Renyi entropy reduces to samplewise interactions when combined with Parzen density estimation (Principe, Fisher, and Xu, 2000).

$$\begin{aligned} H_R(Y) &= -\log \int_{\mathbf{y}} p(\mathbf{y})^2 d\mathbf{y} \\ &= -\log \frac{1}{N^2} \int_{\mathbf{y}} \left(\sum_{k=1}^{N} \sum_{j=1}^{N} G(\mathbf{y}-\mathbf{y}_k,\sigma^2 I)G(\mathbf{y}-\mathbf{y}_j,\sigma^2 I) \right) d\mathbf{y} \\ &= -\log \frac{1}{N^2} \sum_{k=1}^{N} \sum_{j=1}^{N} G(\mathbf{y}_k-\mathbf{y}_j,2\sigma^2 I). \quad (9) \end{aligned}$$

Divergence Measures

- Kullback-Leibler divergence

$$K(f,g) = \int_{\mathbf{x}} f(\mathbf{x}) \log \frac{f(\mathbf{x})}{g(\mathbf{x})} d\mathbf{x} \tag{10}$$

- Variational distance (Based on the f-divergence family)

$$V(f,g) = \int_{\mathbf{x}} |f(\mathbf{x}) - g(\mathbf{x})| d\mathbf{x}. \tag{11}$$

- Quadratic divergence

$$D(f,g) = \int_{\mathbf{x}} (f(\mathbf{x}) - g(\mathbf{x}))^2 d\mathbf{x}, \tag{12}$$

- Pinsker's inequality gives a lower bound on $K(f,g) \geq \frac{1}{2} V(f,g)^2$. Since $f(\mathbf{x})$ and $g(\mathbf{x})$ are probability density functions, both are between zero and one, and $|f(\mathbf{x}) - g(\mathbf{x})| \geq (f(\mathbf{x}) - g(\mathbf{x}))^2$, and thus $V(f,g) \geq D(f,g)$. Maximizing $D(f,g)$ thus maximizes a lower bound to $K(f,g)$.

Outline

Class Separability Measures

1. Sums of distances between data points of different classes.
2. Nonlinear functions of the distances or sums of the distances.
3. Probabilistic measures based on class conditional densities.
   - These measures may make an approximation to class conditional densities followed by some distance measure between densities (Battacharyya distance or divergence)
   - A Gaussian assumption usually needs to be made about the class-conditional densities to make numerical optimization tractable.
   - Equal class covariance assumption, although restrictive, leads to the well known Linear Discriminant Analysis (LDA), which has an analytic solution.
   - Some measures allow non-parametric estimation of the class conditional densities.
4. The Bayes error

Outline

Relation of The Bayes Error to Mutual Information

- The Bayes risk using $0/1$-loss for classification can be written as the Bayes error:

$$e_{bayes}(X) = E_x[Pr(y \neq \hat{y})] = \int_{\mathbf{x}} p(\mathbf{x}) \left(1 - \max_i (p(y_i|\mathbf{x}))\right) d\mathbf{x}, \qquad (13)$$

- An upper bound on the Bayes error (Hellman, 1970; Feder 1990)

$$e_{bayes}(X) \leq \frac{1}{2} H(Y|X) = \frac{1}{2}(H(Y) - I(Y,X)) \qquad (14)$$

- A lower bound on the error also involving conditional entropy or mutual information is given by Fano's (1961) inequality

$$e_{bayes}(X) \geq 1 - \frac{I(Y,X) + \log 2}{\log(|Y|)}, \qquad (15)$$

  where $|Y|$ refers to the cardinality of Y.
- Both bounds are minimized when the mutual information between Y and X is maximized, or when $H(Y|X)$ is minimized.
- The bounds are relatively tight, in the sense that both inequalities can be obtained with equality

Outline

Pairwise MI in variable selection

MIFS

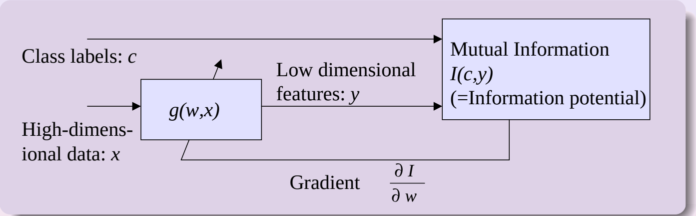
1: Set $\hat{X} = \text{argmax}_{X_i} I(Y, X_i)$;
set $\Phi \leftarrow \{\hat{X}\}$;
set $F \leftarrow \{X_1, ..., X_N\} \setminus \{\hat{X}\}$.

2: For all pairs (i, j), $X_i \in F$ and $X_j \in \Phi$
evaluate and save $I(X_i, X_j)$ unless already saved.

3: Set $\hat{X} = \text{argmax}_{X_i} \left[I(Y, X_i) - \beta \sum_{X_j \in \Phi} I(X_i, X_j) \right]$;
set $\Phi \leftarrow \Phi \cup \{\hat{X}\}$;
set $F \leftarrow F \setminus \{\hat{X}\}$,
and repeat from step 2 until $|\Phi|$ is desired.

Outline

Learning Feature Transforms by Maximizing Mutual Information Between Class Labels and Features



Express $I = I(\{\boldsymbol{y}_i, c_i\})$ in a differentiable form and perform gradient ascent (or other optimization) on $\boldsymbol{w}$, parameters of the transform g as

$$\boldsymbol{w}_{t+1} = \boldsymbol{w}_t + \eta \frac{\partial I}{\partial \boldsymbol{w}} = \boldsymbol{w}_t + \eta \sum_{i=1}^{N} \frac{\partial I}{\partial \boldsymbol{y}_i} \frac{\partial \boldsymbol{y}_i}{\partial \boldsymbol{w}}$$

1st part of the last term: information force that other samples exert to $\boldsymbol{y}_i$, 2nd part depends on the transform. If $\boldsymbol{y_i} = W\boldsymbol{x_i}$ then simply $\frac{\partial \boldsymbol{y}_i}{\partial W} = \boldsymbol{x}_i^T$.

Non-Parametric MI between Features and Labels

Labels — discrete random variable C.
Features — continuous, vector-valued Y.

Write I_T in between C and Y using the quadratic divergence:

$$\begin{aligned} I_T(C, Y) &= \sum_c \int_{\mathbf{y}} (p(c, \mathbf{y}) - p(c)p(\mathbf{y}))^2 d\mathbf{y} \\ &= \sum_c \int_{\mathbf{y}} p(c, \mathbf{y})^2 d\mathbf{y} \\ &+ \sum_c \int_{\mathbf{y}} p(c)^2 p(\mathbf{y})^2 d\mathbf{y} \\ &- 2 \sum_c \int_{\mathbf{y}} p(c, \mathbf{y}) p(c) p(\mathbf{y}) d\mathbf{y} \end{aligned} \quad (16)$$

Non-Parametric MI between Features and Labels

Using a data set of N samples and expressing class densities as their Parzen estimates with kernel width σ results in

$$
\begin{aligned}
I_T(\{\mathbf{y}_i, c_i\}) &= V_{IN} + V_{ALL} - 2V_{BTW} \\
&= \frac{1}{N^2} \sum_{p=1}^{N_c} \sum_{k=1}^{J_p} \sum_{l=1}^{J_p} G(\mathbf{y}_{pk} - \mathbf{y}_{pl}, 2\sigma^2 I) \\
&+ \frac{1}{N^2} \left(\sum_{p=1}^{N_c} \left(\frac{J_p}{N} \right)^2 \right) \sum_{k=1}^{N} \sum_{l=1}^{N} G(\mathbf{y}_k - \mathbf{y}_l, 2\sigma^2 I) \\
&- 2 \frac{1}{N^2} \sum_{p=1}^{N_c} \frac{J_p}{N} \sum_{j=1}^{J_p} \sum_{k=1}^{N} G(\mathbf{y}_{pj} - \mathbf{y}_k, 2\sigma^2 I) \qquad (17)
\end{aligned}
$$

Gradient of the Information Potential

- First, we need the derivative of the potential, or, the force between two samples as

$$\frac{\partial}{\partial \boldsymbol{y}_i} G(\boldsymbol{y}_i - \boldsymbol{y}_j, 2\sigma^2 I) = G(\boldsymbol{y}_i - \boldsymbol{y}_j, 2\sigma^2 I) \frac{(\boldsymbol{y}_j - \boldsymbol{y}_i)}{2\sigma^2}. \quad (18)$$

- With this we get for V_{IN}

$$\frac{\partial}{\partial \boldsymbol{y}_{ci}} V_{IN} = \frac{1}{N^2\sigma^2} \sum_{k=1}^{J_c} G(\boldsymbol{y}_{ck} - \boldsymbol{y}_{ci}, 2\sigma^2 I)(\boldsymbol{y}_{ck} - \boldsymbol{y}_{ci}). \quad (19)$$

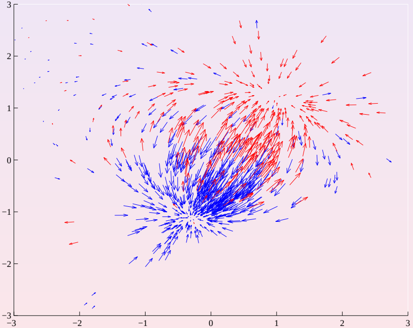
  This represents a sum of forces that other "particles" in class c exert to particle $\boldsymbol{y}_{ci}$ (direction is towards $\boldsymbol{y}_{ci}$).
- $\frac{\partial}{\partial \boldsymbol{y}_i} V_{ALL}$ represents a sum of forces that other "particles" regardless of class exert to particle $\boldsymbol{y}_{ci}$ (towards $\boldsymbol{y}_i$).
- The effect of $\frac{\partial}{\partial \boldsymbol{y}_i} V_{BTW}$ away from $\boldsymbol{y}_{ci}$, and it represents the repulsion of classes away from each other

Information Potential and Information Forces

Mutual information $I_T(\{\boldsymbol{y}_i, c_i\})$ can now be interpreted as an information potential induced by samples of data in different classes.

$\partial I/\partial \boldsymbol{y}_i$ can be interpreted as an information force that other samples exert to sample $\boldsymbol{y}_i$. It has three components:

1. Samples within a class attract each other
2. All samples attract each other
3. Samples between classes repel each other

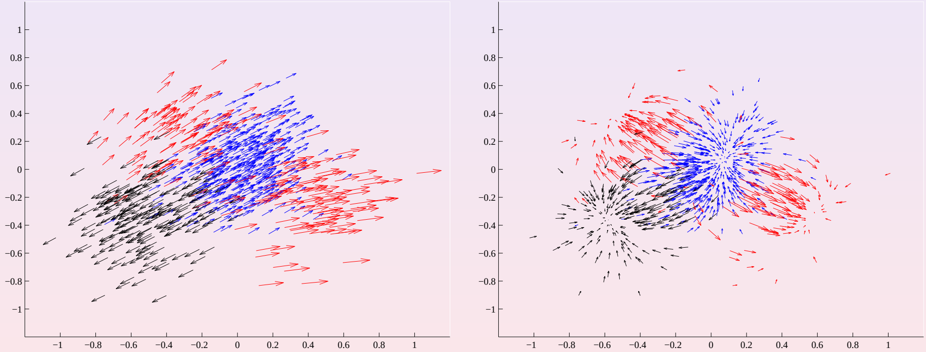


Computing $\partial I/\partial \boldsymbol{y}_i$ for all $\boldsymbol{y}_i$ requires $O(N^2)$ operations (Torkkola, 2003).

Outline

Effect of the kernel width on the forces

Three classes in three dimensions projected onto a two-dimensional subspace.

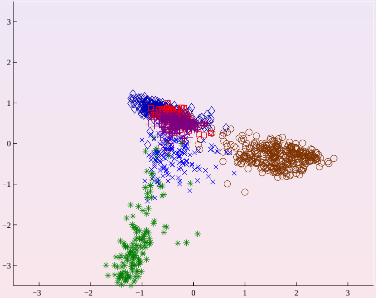


Wide kernel

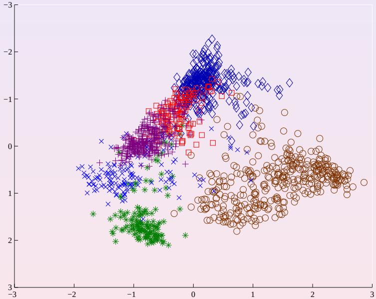
Narrow kernel

LDA vs. MMI

Landsat satellite image database from UCI repository: Six classes in 36 dimensions projected onto a two-dimensional subspace using...



LDA — compact representation of classes — on top of each other!

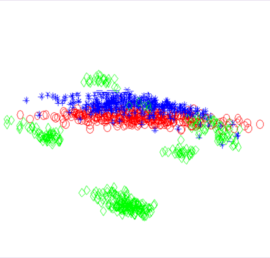


MMI — classes not as compact — need not look Gaussian — and better separated!

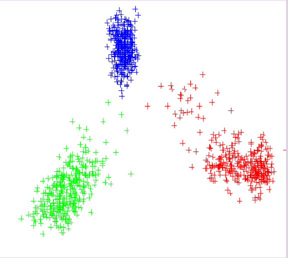
PCI vs. LDA vs. MMI

Three classes in 12 dimensions (oil pipe-flow from Aston University) projected onto a two-dimensional subspace using PCA (left), LDA or MMI with a wide kernel (middle), and MMI using a narrow kernel (right).

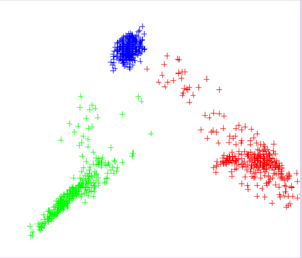
PCA



LDA/MMI wide

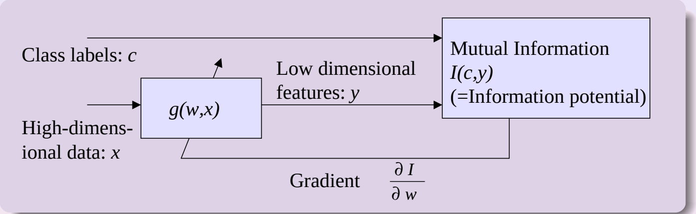


MMI narrow



Outline

Learning nonlinear transforms using MI



- Exactly the same procedure as with linear transforms. The transform g just needs to be continuous (differentiable wrt. the parameter vector $\mathbf{w}$).
- The information force remains the same, the transform-dependent part $\partial \mathbf{y}_i / \partial \mathbf{w}$ will be different.

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \eta \frac{\partial I}{\partial \mathbf{w}} = \mathbf{w}_t + \eta \sum_{i=1}^{N} \frac{\partial I}{\partial \mathbf{y}_i} \frac{\partial \mathbf{y}_i}{\partial \mathbf{w}}$$

Nonlinear transforms

Multilayer perceptrons

- Gradient of the output w.r.t. weights using (information) backpropagation
- Hidden layer activation - tanh
- Output layer activation:
  - Linear: Need orthonormalized weights in output layer
  - Tanh: Can use data-independent kernel width
- Weight initialization, partially by LDA

Radial basis function networks

- Basis functions by EM separately for each class as mixtures of diagonal-covariance Gaussians
- Two options:
  - Use MMI only to learn the output layer (this work)
  - Learn all the parameters using MMI

Outline

Stochastic gradient

- Instead of interactions between all pairs of data points, take a sample of just two.
- Samples of the same class: NO update! Samples in different classes:

$$\begin{aligned} W_{t+1} &= W_t + \eta \sum_{i=1,2} \frac{\partial I}{\partial \boldsymbol{y}_i} \frac{\partial \boldsymbol{y}_i}{\partial W} \\ &= W_t - \frac{\eta}{8\sigma^2} G(\boldsymbol{y}_1 - \boldsymbol{y}_2, 2\sigma^2 I)(\boldsymbol{y}_2 - \boldsymbol{y}_1)(\boldsymbol{x}_1^T - \boldsymbol{x}_2^T) \end{aligned} \quad (20)$$

- Full gradient using all pairs, and stochastic gradient using just one pair at a time are two ends of a spectrum: It is more desirable to take as large a random sample of the whole data set as possible, and to compute all the mutual interactions between those samples for one update of W.

Semi-Parametric Density Estimation

- Construct a Gaussian Mixture Models model in the low-dimensional output space after a random or an informed guess as the transform:

$$p(\boldsymbol{y}|c_p) = \sum_{j=1}^{K_p} h_{pj} G(\boldsymbol{y} - \boldsymbol{m}_{pj}, S_{pj}) \tag{21}$$

- The same samples are used to construct a GMM in the input space using the *same exact assignments of samples to mixture components* as the output space GMMs have. Running the EM-algorithm in the input space is now unnecessary since we know which samples belong to which mixture components.
- Now we have GMMs in both spaces and a transform mapping between the two
- Avoid operating in the high-dimensional input space altogether

Video clips

Outline

Fisher Information

Options to make relevant information more explicit:

1. Variable selection.
2. Feature construction.
   Selection/construction matrix defines a global Euclidean metric
   $$d_A^2(\mathbf{x}, \mathbf{x}') = (\mathbf{x} - \mathbf{x}')^T S^T S(\mathbf{x} - \mathbf{x}') = (\mathbf{x} - \mathbf{x}')^T A(\mathbf{x} - \mathbf{x}')$$
3. Learning a distance metric locally relevant to target (or some auxiliary variable): $A = A(\mathbf{x})$
   $$d_A^2(\mathbf{x}, \mathbf{x} + d\mathbf{x}) = d\mathbf{x}^T A(\mathbf{x}) d\mathbf{x}.$$
   Metric should reflect the divergence between conditional distributions of the target.
   $$d_J^2(\mathbf{x}, \mathbf{x} + d\mathbf{x}) = D_{KL}[p(y|\mathbf{x})||p(y|\mathbf{x} + d\mathbf{x})] = \frac{1}{2} d\mathbf{x}^T J(\mathbf{x}) d\mathbf{x}.$$
   Embedded into e.g. a clustering algorithm results in "semi-supervised" clustering that reflects the auxiliary variable (Peltonen, 2004).

Fisher Information

Options to make relevant information more explicit:

1. Variable selection.
2. Feature construction.
   Selection/construction matrix defines a global Euclidean metric

$$d_A^2(\mathbf{x},\mathbf{x}') = (\mathbf{x}-\mathbf{x}')^T S^T S(\mathbf{x}-\mathbf{x}') = (\mathbf{x}-\mathbf{x}')^T A(\mathbf{x}-\mathbf{x}')$$

3. Learning a distance metric locally relevant to target (or some auxiliary variable): $A = A(\mathbf{x})$

$$d_A^2(\mathbf{x},\mathbf{x}+d\mathbf{x}) = d\mathbf{x}^T A(\mathbf{x}) d\mathbf{x}.$$

   Metric should reflect the divergence between conditional distributions of the target.

$$d_J^2(\mathbf{x},\mathbf{x}+d\mathbf{x}) = D_{KL}[p(y|\mathbf{x})||p(y|\mathbf{x}+d\mathbf{x})] = \frac{1}{2} d\mathbf{x}^T J(\mathbf{x}) d\mathbf{x}.$$

   Embedded into e.g. a clustering algorithm results in “semi-supervised” clustering that reflects the auxiliary variable (Peltonen, 2004).

Fisher Information

Options to make relevant information more explicit:

1. Variable selection.
2. Feature construction.
   Selection/construction matrix defines a global Euclidean metric

$$d_A^2(\mathbf{x}, \mathbf{x}') = (\mathbf{x} - \mathbf{x}')^T S^T S(\mathbf{x} - \mathbf{x}') = (\mathbf{x} - \mathbf{x}')^T A(\mathbf{x} - \mathbf{x}')$$

3. Learning a distance metric locally relevant to target (or some auxiliary variable): $A = A(\mathbf{x})$

$$d_A^2(\mathbf{x}, \mathbf{x} + d\mathbf{x}) = d\mathbf{x}^T A(\mathbf{x}) d\mathbf{x}.$$

Metric should reflect the divergence between conditional distributions of the target.

$$d_J^2(\mathbf{x}, \mathbf{x} + d\mathbf{x}) = D_{KL}[p(y|\mathbf{x})||p(y|\mathbf{x} + d\mathbf{x})] = \frac{1}{2} d\mathbf{x}^T J(\mathbf{x}) d\mathbf{x}.$$

Embedded into e.g. a clustering algorithm results in “semi-supervised” clustering that reflects the auxiliary variable (Peltonen, 2004).

Fisher Information

Options to make relevant information more explicit:

1. Variable selection.
2. Feature construction.
   Selection/construction matrix defines a global Euclidean metric

$$d_A^2(\mathbf{x}, \mathbf{x}') = (\mathbf{x} - \mathbf{x}')^T S^T S(\mathbf{x} - \mathbf{x}') = (\mathbf{x} - \mathbf{x}')^T A(\mathbf{x} - \mathbf{x}')$$

3. Learning a distance metric locally relevant to target (or some auxiliary variable): $A = A(\mathbf{x})$

$$d_A^2(\mathbf{x}, \mathbf{x} + d\mathbf{x}) = d\mathbf{x}^T A(\mathbf{x}) d\mathbf{x}.$$

   Metric should reflect the divergence between conditional distributions of the target.

$$d_J^2(\mathbf{x}, \mathbf{x} + d\mathbf{x}) = D_{KL}[p(y|\mathbf{x})||p(y|\mathbf{x} + d\mathbf{x})] = \frac{1}{2} d\mathbf{x}^T J(\mathbf{x}) d\mathbf{x}.$$

   Embedded into e.g. a clustering algorithm results in “semi-supervised” clustering that reflects the auxiliary variable (Peltonen, 2004).

Fisher Information

Options to make relevant information more explicit:

1. Variable selection.
2. Feature construction.
   Selection/construction matrix defines a global Euclidean metric
   $$d_A^2(\mathbf{x}, \mathbf{x}') = (\mathbf{x} - \mathbf{x}')^T S^T S(\mathbf{x} - \mathbf{x}') = (\mathbf{x} - \mathbf{x}')^T A(\mathbf{x} - \mathbf{x}')$$
3. Learning a distance metric locally relevant to target (or some auxiliary variable): $A = A(\mathbf{x})$
   $$d_A^2(\mathbf{x}, \mathbf{x} + d\mathbf{x}) = d\mathbf{x}^T A(\mathbf{x}) d\mathbf{x}.$$
   Metric should reflect the divergence between conditional distributions of the target.
   $$d_J^2(\mathbf{x}, \mathbf{x} + d\mathbf{x}) = D_{KL}[p(y|\mathbf{x})||p(y|\mathbf{x} + d\mathbf{x})] = \frac{1}{2} d\mathbf{x}^T J(\mathbf{x}) d\mathbf{x}.$$
   Embedded into e.g. a clustering algorithm results in “semi-supervised” clustering that reflects the auxiliary variable (Peltonen, 2004).

Outline

Information Bottleneck

If X are the original data, Φ is a seeked representation, and Y variable(s) of importance, minimizing loss function

$$\mathcal{L}(p(\phi|x)) = I(X, \Phi) - \beta I(\Phi, Y)$$

leads to

IB solution

1. $p(\phi|x) = \frac{p(t)}{Z(\beta,x)} \exp(-\beta D_{KL}[p(y|x), p(y|\phi)])$
2. $p(y|\phi) = \frac{1}{p(t)} \sum_x p(y|x)p(\phi|x)p(x)$
3. $p(\phi) = \sum_x p(\phi|x)p(x)$

For example, if X are documents, Y are words, and Φ are word clusters, probability of cluster membership decays exponentially with KL-divergence between the word distributions in document x and cluster ϕ (Tishby, 1999)

Conclusion

- Shannon's seminal work showed how mutual information provides a measure of the maximum transmission rate of information through a channel
- Analogy to variable selection and feature construction with mutual information as the criterion to provide maximal information about the variable of interest
- Best thing since sliced bread for variable selection / feature construction?
- Maybe not - estimation problems from high-dimensional small(ish) data sets

Conclusion

- Shannon's seminal work showed how mutual information provides a measure of the maximum transmission rate of information through a channel
- Analogy to variable selection and feature construction with mutual information as the criterion to provide maximal information about the variable of interest
- Best thing since sliced bread for variable selection / feature construction?
- Maybe not - estimation problems from high-dimensional small(ish) data sets

Conclusion

- Shannon's seminal work showed how mutual information provides a measure of the maximum transmission rate of information through a channel
- Analogy to variable selection and feature construction with mutual information as the criterion to provide maximal information about the variable of interest
- Best thing since sliced bread for variable selection / feature construction?
- Maybe not - estimation problems from high-dimensional small(ish) data sets