Estimating mutual information for high-dimensional sparse relationships

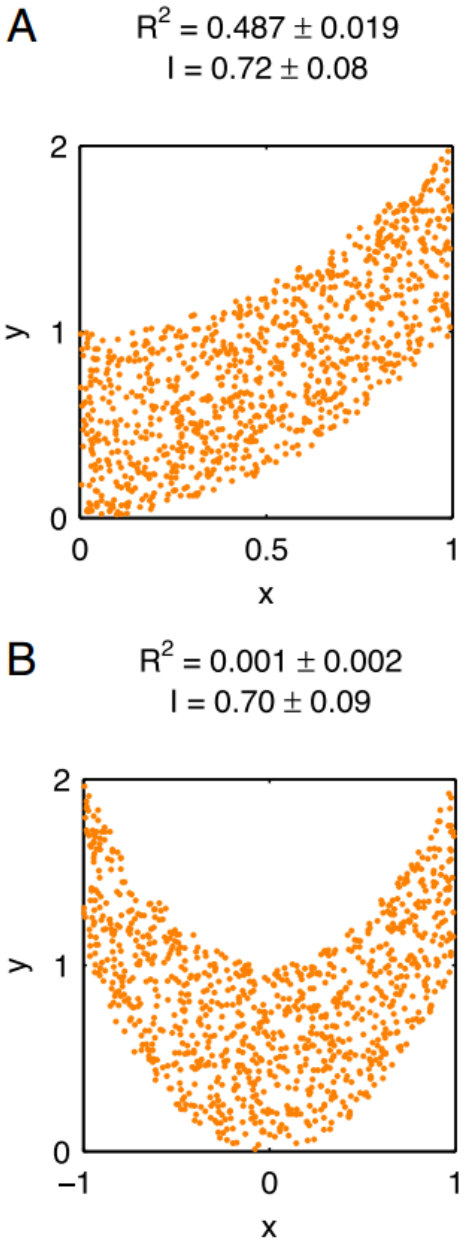
Charles Zheng

Stanford University

January 11, 2017

(Joint work with Yuval Benjamini.)

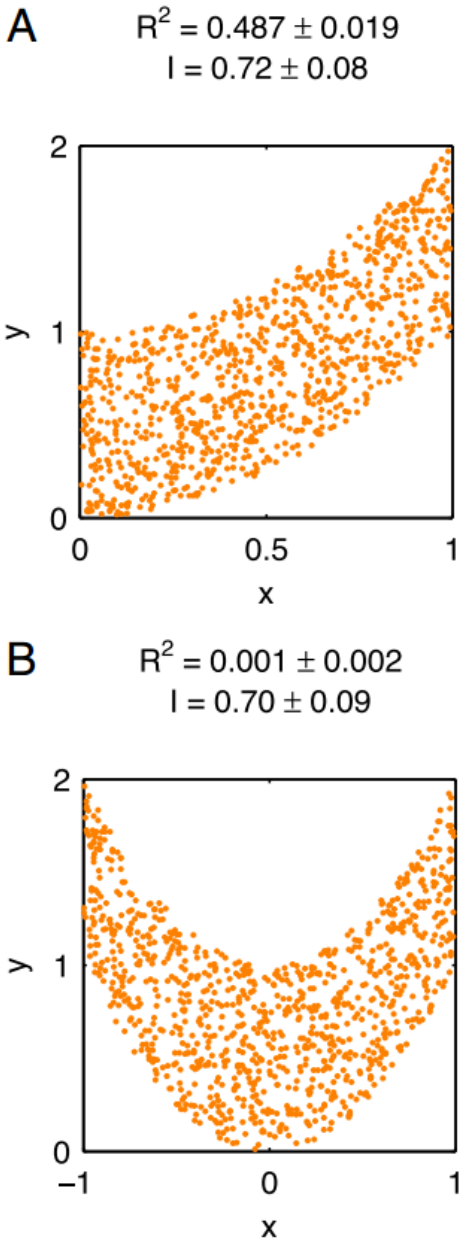
Overview



Mutual information $I(\vec{X}; \vec{Y})$

- measures dependence between two random vectors, $\vec{X}$ and $\vec{Y}$

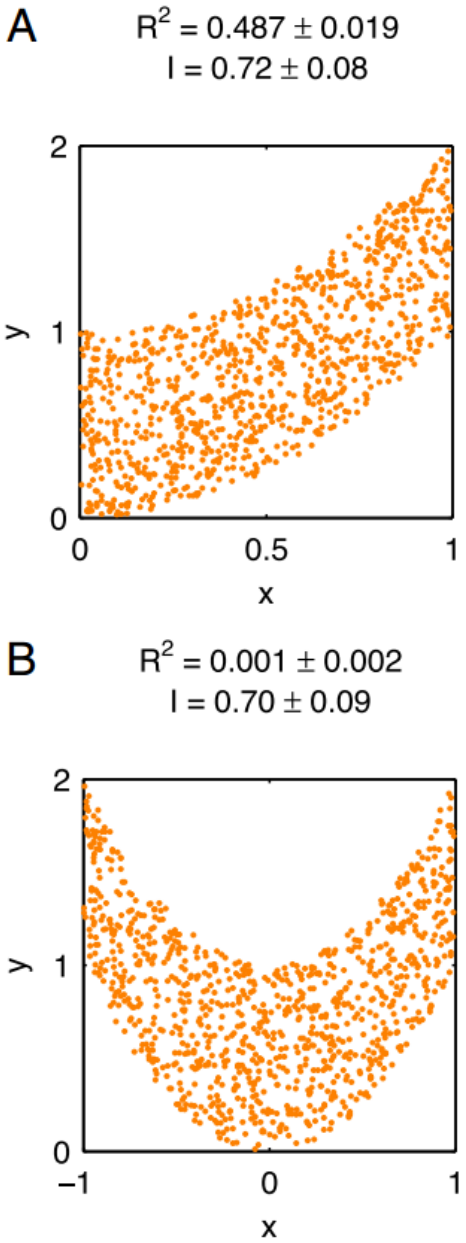
Overview



Mutual information $I(\vec{X}; \vec{Y})$

- measures dependence between two random vectors, $\vec{X}$ and $\vec{Y}$
- applies to nonlinear and multidimensional relationships (unlike correlation)

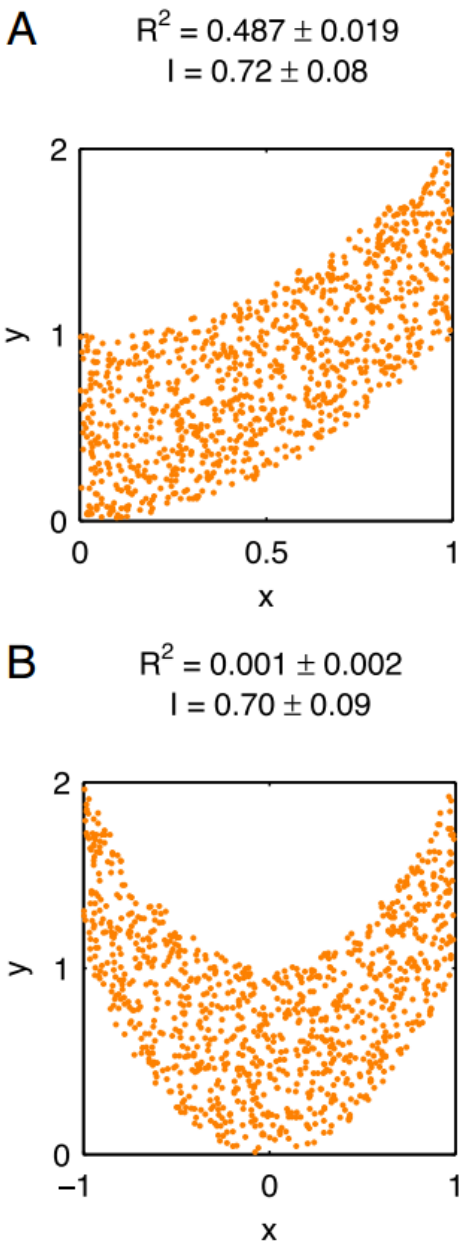
Overview



Mutual information $I(\vec{X}; \vec{Y})$

- measures dependence between two random vectors, $\vec{X}$ and $\vec{Y}$
- applies to nonlinear and multidimensional relationships (unlike correlation)
- is *difficult to estimate* in high dimensions

Overview

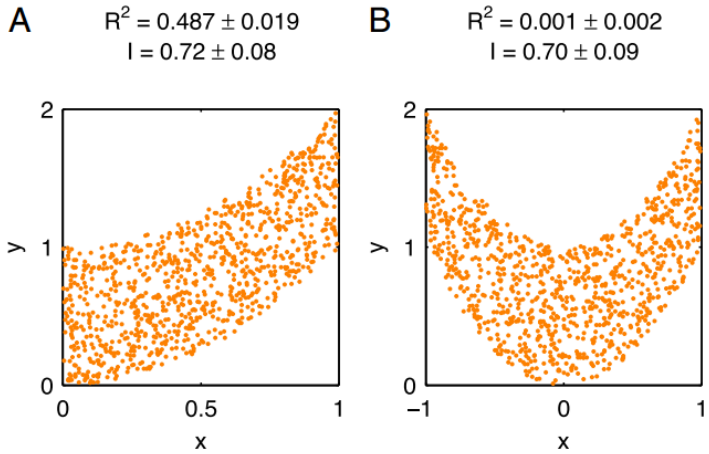


Mutual information $I(\vec{X}; \vec{Y})$

- measures dependence between two random vectors, $\vec{X}$ and $\vec{Y}$
- applies to nonlinear and multidimensional relationships (unlike correlation)
- is *difficult to estimate* in high dimensions

This work combines *machine learning* (sparse estimation) with *information theory* to obtain better estimates of mutual information.

Mutual information $I(X; Y)$



Introduced in Shannon's 1948 paper, "A mathematical theory of communication"

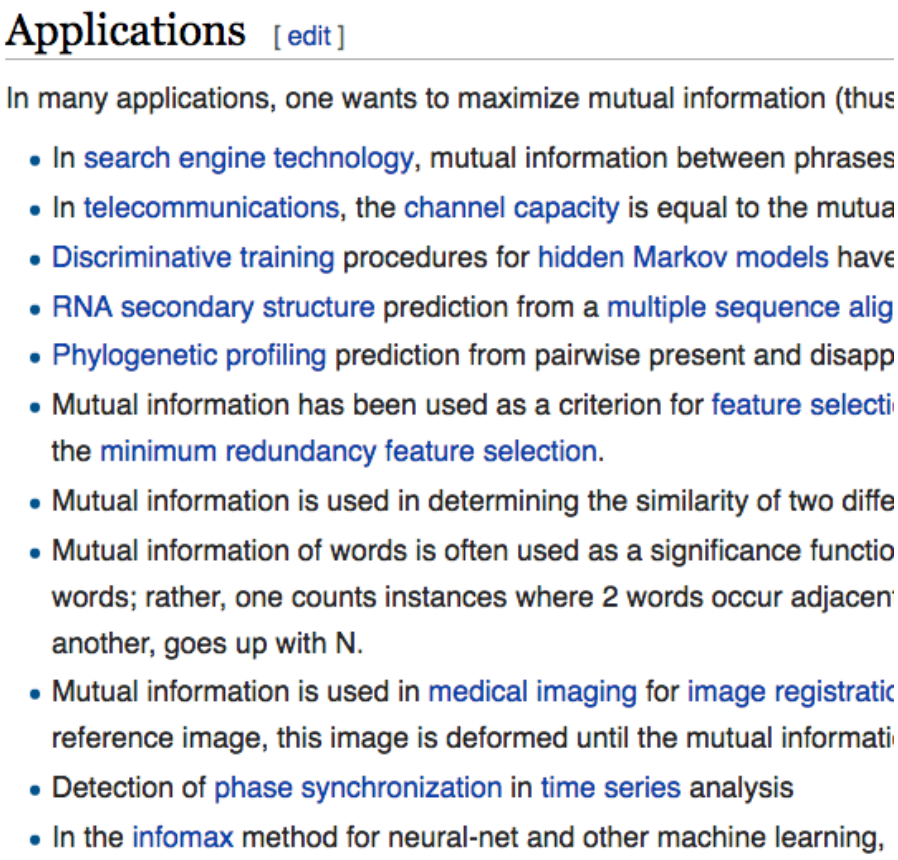
Image credit Kinney et al. 2014.

Applications of $I(X;Y)$

Mutual information has since been applied to many areas outside of information theory

Applications of $I(X;Y)$

Mutual information has since been applied to many areas outside of information theory



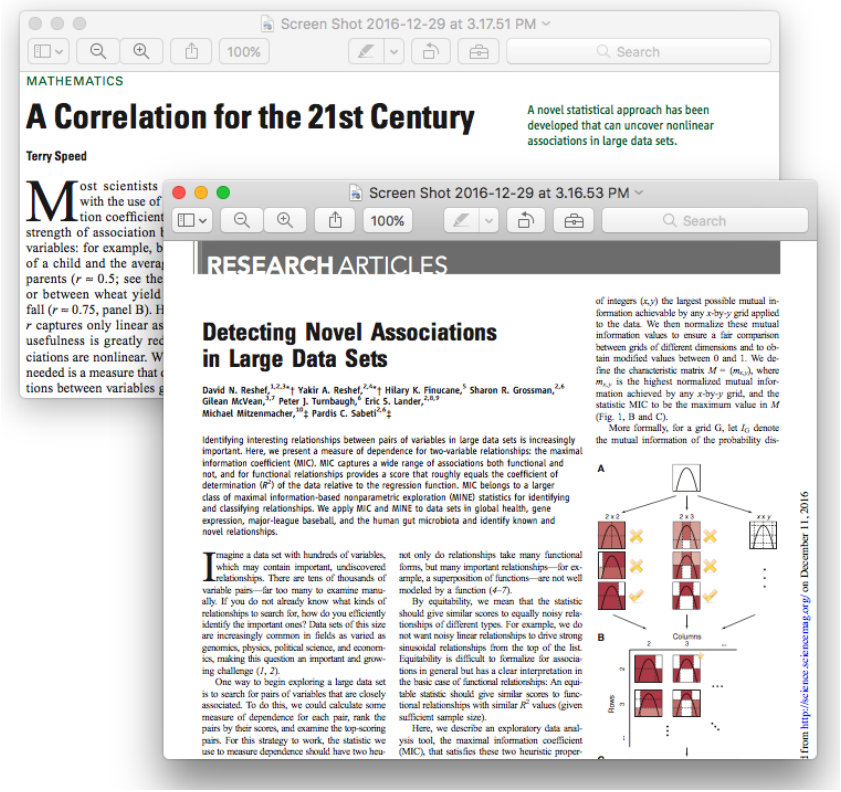
Applications [edit]

In many applications, one wants to maximize mutual information (thus

- In search engine technology, mutual information between phrases
- In telecommunications, the channel capacity is equal to the mutua
- Discriminative training procedures for hidden Markov models have
- RNA secondary structure prediction from a multiple sequence alig
- Phylogenetic profiling prediction from pairwise present and disapp
- Mutual information has been used as a criterion for feature selecti the minimum redundancy feature selection.
- Mutual information is used in determining the similarity of two diffe
- Mutual information of words is often used as a significance functio words; rather, one counts instances where 2 words occur adjacen another, goes up with N.
- Mutual information is used in medical imaging for image registratic reference image, this image is deformed until the mutual informati
- Detection of phase synchronization in time series analysis
- In the infomax method for neural-net and other machine learning,

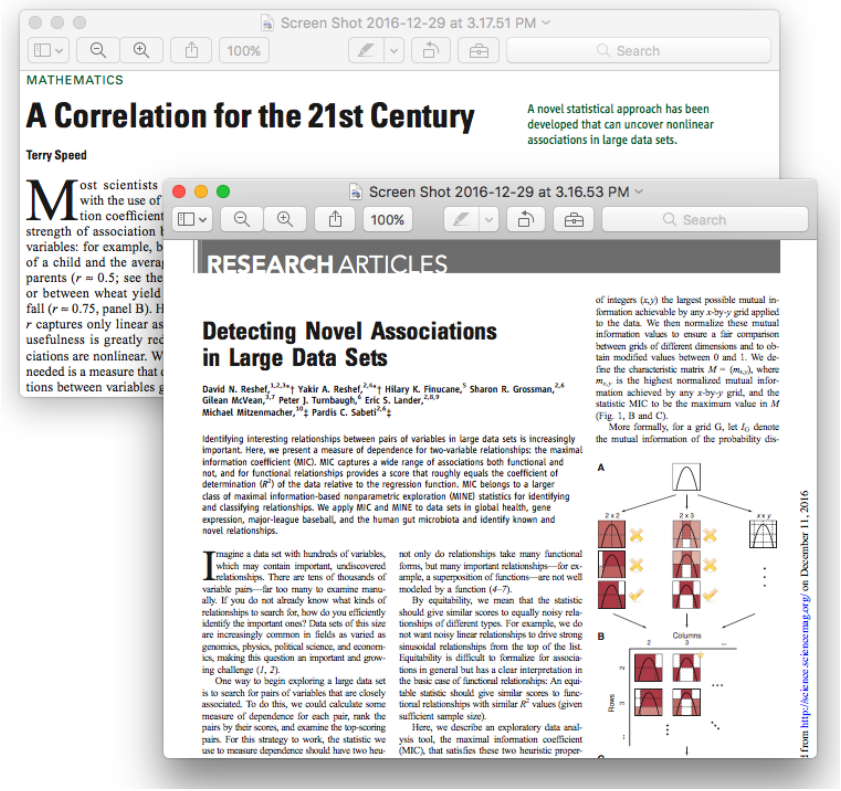
Engineering, biology, computer science, physics, medicine

Comparing $I(X; Y)$ with Pearson correlation



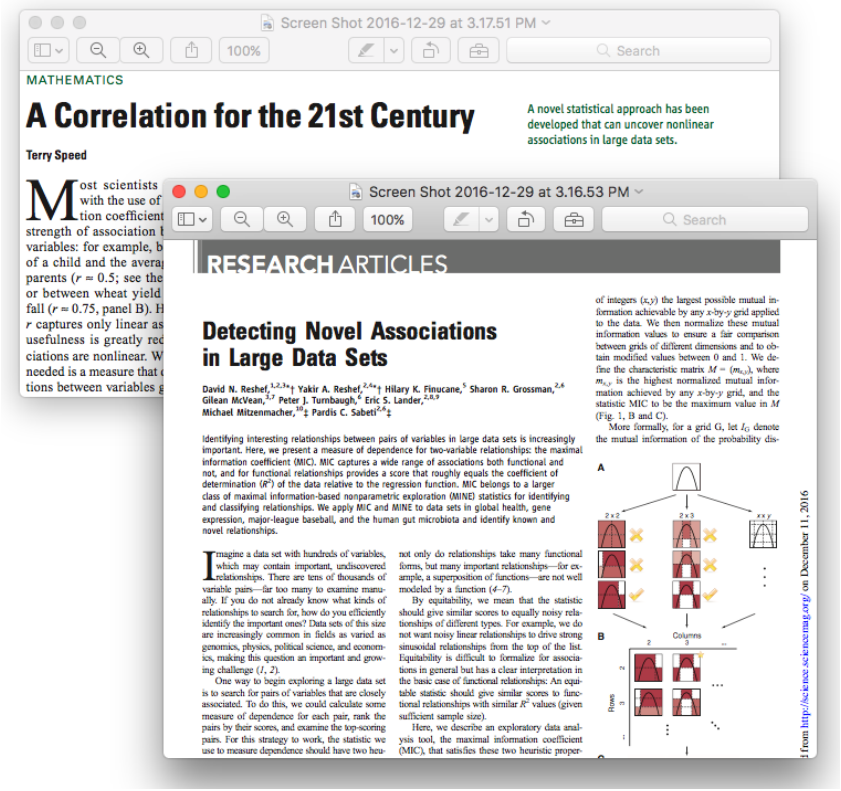
- In many applications scientists are interested in *dependence*, not *correlation* (Reshef et al. 2011, Speed 2011).

Comparing $I(X; Y)$ with Pearson correlation



- In many applications scientists are interested in *dependence*, not *correlation* (Reshef et al. 2011, Speed 2011).
- Only mutual information (and derived quantities) measures dependence directly

Comparing $I(X; Y)$ with Pearson correlation



- In many applications scientists are interested in *dependence*, not *correlation* (Reshef et al. 2011, Speed 2011).
- Only mutual information (and derived quantities) measures dependence directly

Problems with mutual information

- Hard to interpret (compared to R^2)
- Hard to estimate (compared to R^2)

Can we make $I(X;Y)$ easier to interpret?

- If (X, Y) have a bivariate normal distribution with correlation ρ, then

$$I(X;Y) = \frac{-1}{2}\ln(1-\rho^2)$$

Can we make $I(X;Y)$ easier to interpret?

- If (X, Y) have a bivariate normal distribution with correlation ρ, then

$$I(X;Y) = \frac{-1}{2}\ln(1-\rho^2)$$

- Define the "Shannon correlation"

$$\text{Cor}_{Shannon}(X,Y) = \sqrt{1-e^{-2I(X;Y)}}$$

Can we make $I(X; Y)$ easier to interpret?

- If (X, Y) have a bivariate normal distribution with correlation ρ, then

$$I(X; Y) = \frac{-1}{2} \ln(1 - \rho^2)$$

- Define the "Shannon correlation"

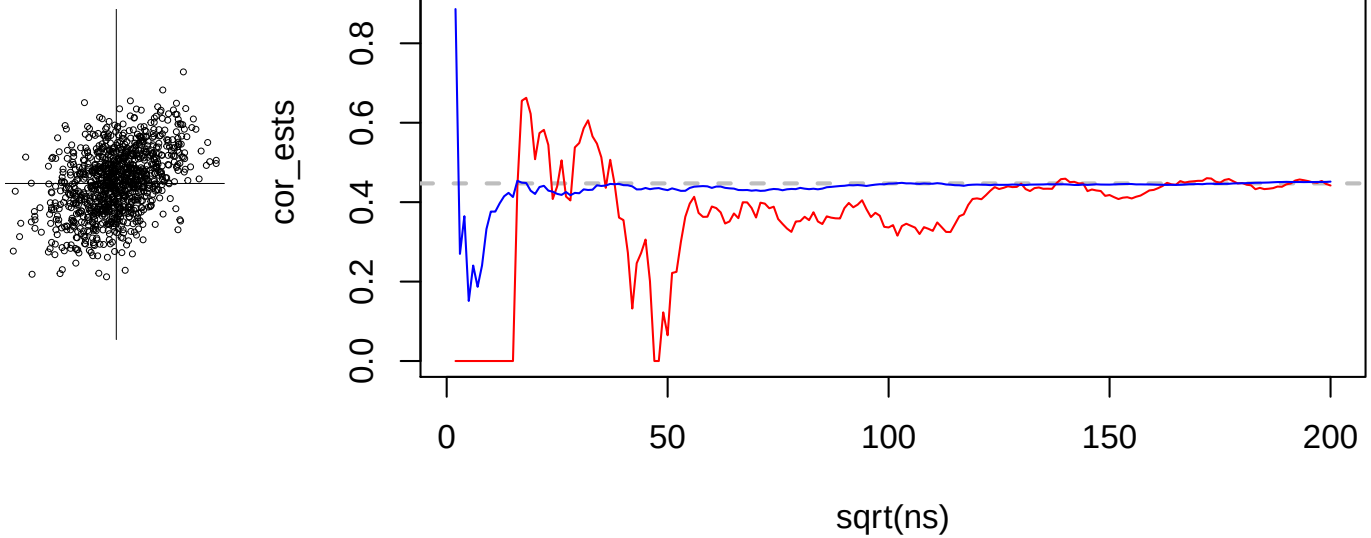
$$\text{Cor}_{Shannon}(X, Y) = \sqrt{1 - e^{-2I(X;Y)}}$$

- Then $\text{Cor}_{Shannon}(X, Y) \in [0, 1]$.
- For (X, Y) bivariate normal,

$$\text{Cor}_{Pearson}(X, Y) = \text{Cor}_{Shannon}(X, Y)$$

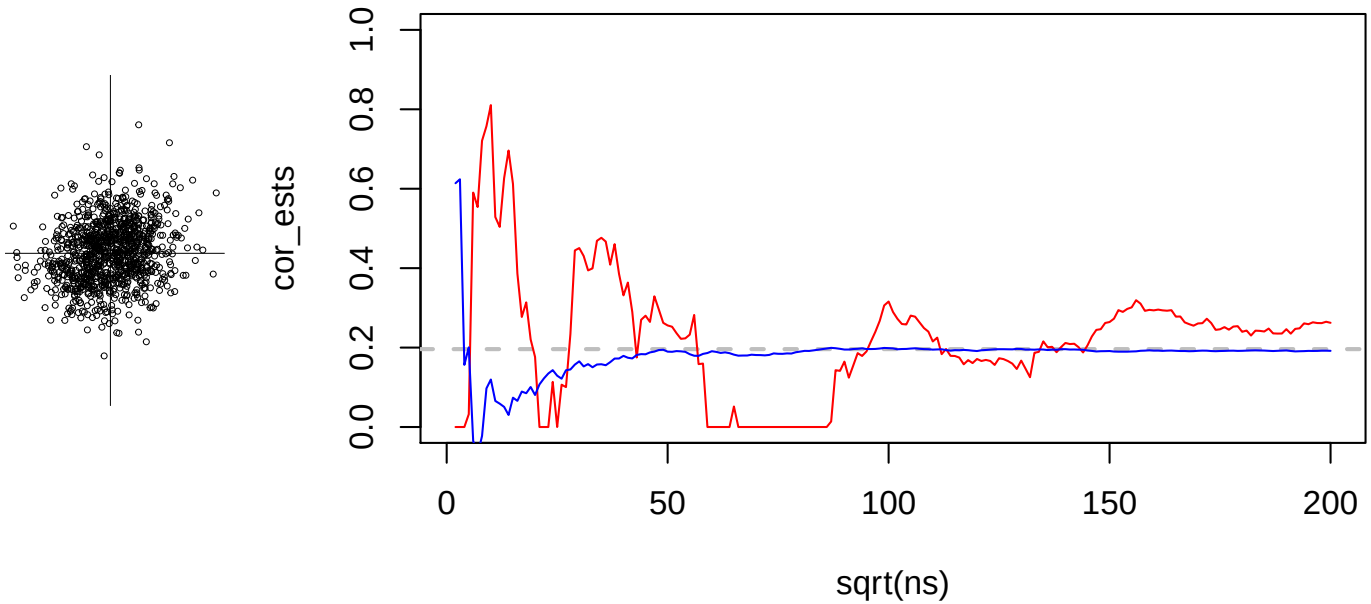
Difficulty of estimating $I(X; Y)$

Example with $\text{Cor}_{Pearson}(X, Y) = \text{Cor}_{Shannon}(X, Y) = 0.44$.



Difficulty of estimating $I(X; Y)$

Example with $\text{Cor}_{Pearson}(X, Y) = \text{Cor}_{Shannon}(X, Y) = 0.2$.



How to estimate $I(X; Y)$

Suppose we observe pairs $(X_i, Y_i)_{i=1}^n$ iid from density $p(x, y)$

- Definition of mutual information:

$$I(X; Y) = \int \log \left(\frac{p(x, y)}{p(x)p(y)} \right) p(x, y) dx dy$$

How to estimate $I(X; Y)$

Suppose we observe pairs $(X_i, Y_i)_{i=1}^n$ iid from density $p(x, y)$

- Definition of mutual information:

$$I(X; Y) = \int \log \left(\frac{p(x, y)}{p(x)p(y)} \right) p(x, y) dx dy$$

- Kernel density estimate approaches estimate $p(x, y)$ (Beirlant et al. 2001, Ivanov and Rozhkova 1981)
- Nearest neighbor estimators rely on distance-based computations (Mnatsakanov et al. 2008, Goria et al. 2005, Singh et. al. 2003)

Problems in high dimensions

- Density estimation is known to have *exponential complexity* with respect to dimensionality.
  - E.g. to get the same precision, you need 10 observations for univariate X, Y but 1000 for trivariate $\vec{X}, \vec{Y}$.

Problems in high dimensions

- Density estimation is known to have *exponential complexity* with respect to dimensionality.
  - E.g. to get the same precision, you need 10 observations for univariate X, Y but 1000 for trivariate $\vec{X}, \vec{Y}$.
- Many applications with high-dimensional X, Y.
  - Gene expression time series
  - Functional magnetic resonance imaging

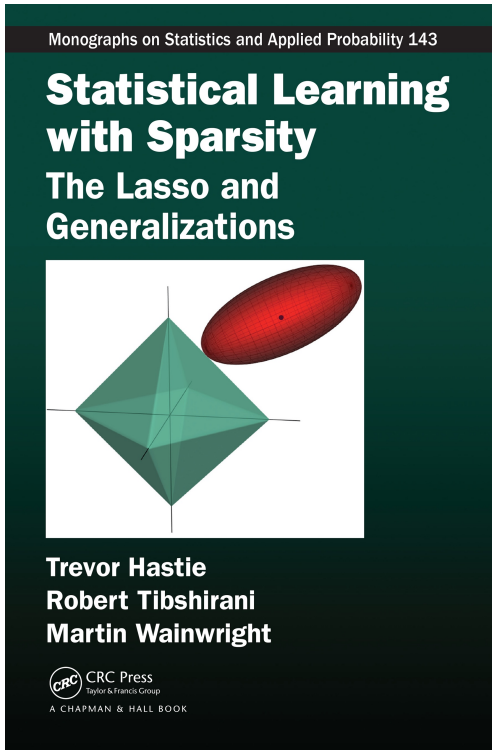
Problems in high dimensions

- Density estimation is known to have *exponential complexity* with respect to dimensionality.
  - E.g. to get the same precision, you need 10 observations for univariate X, Y but 1000 for trivariate $\vec{X}, \vec{Y}$.
- Many applications with high-dimensional X, Y.
  - Gene expression time series
  - Functional magnetic resonance imaging
- One approach is to assume joint multivariate normality of X, Y, but this reduces mutual information to a linear statistic.

Problems in high dimensions

- Density estimation is known to have *exponential complexity* with respect to dimensionality.
  - E.g. to get the same precision, you need 10 observations for univariate X, Y but 1000 for trivariate $\vec{X}, \vec{Y}$.
- Many applications with high-dimensional X, Y.
  - Gene expression time series
  - Functional magnetic resonance imaging
- One approach is to assume joint multivariate normality of X, Y, but this reduces mutual information to a linear statistic.
- Other approaches: binning (Bialek et al. 1991, Paninski 2003), confusion matrix of a classifier (Treves 1997, Quiroga et al. 2009)

First idea: Use sparsity!



- Suppose that $Y \approx f(X) + \epsilon$, where f depends *sparsely* on X.
- Can we exploit the sparsity to obtain an estimate of $I(X; Y)$?

Second idea: link prediction accuracy to mutual information

- If $I(X;Y) > 0$, then X carries information about Y and vice-versa.

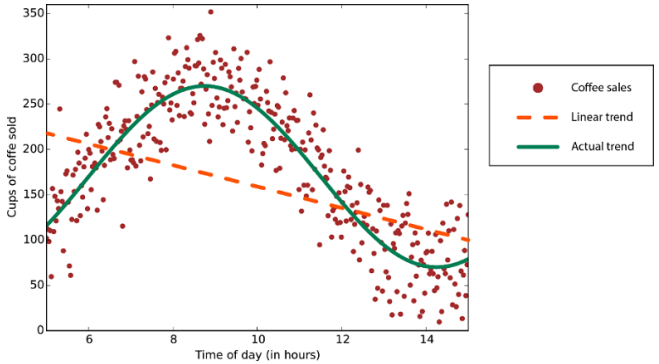
Second idea: link prediction accuracy to mutual information

- If $I(X;Y) > 0$, then X carries information about Y and vice-versa.
- Therefore, we can *predict* Y from X (or X from Y)

Second idea: link prediction accuracy to mutual information

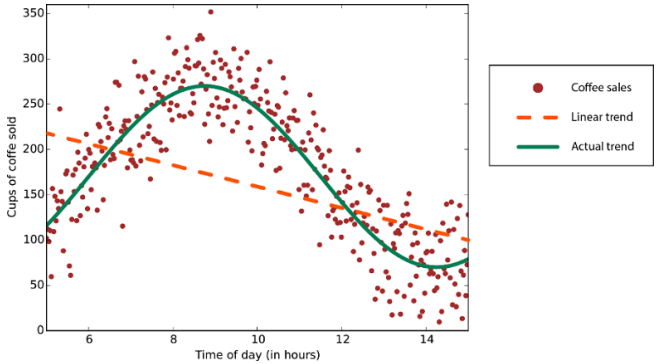
- If $I(X;Y) > 0$, then X carries information about Y and vice-versa.
- Therefore, we can *predict* Y from X (or X from Y)
- We know that often *prediction accuracy* implies a lower bound for *mutual information* (e.g. Fano 1952)

Background: Regression



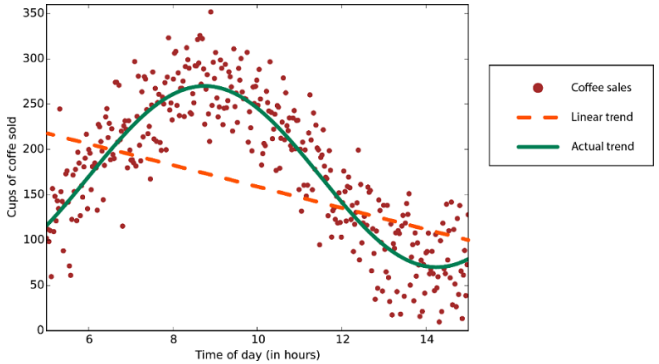
- Suppose you observe $(\vec{X}^{(i)}, Y^{(i)})_{i=1}^{n}$ where $Y^{(i)} = f(\vec{X}^{(i)}) + \epsilon$, where f is an unknown function and ϵ is noise. (Also, assume $\mathbf{E}[\epsilon] = 0$.)

Background: Regression



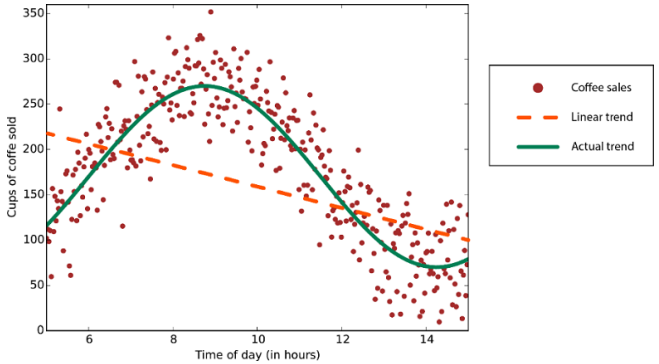
- Suppose you observe $(\vec{X}^{(i)}, Y^{(i)})_{i=1}^n$ where $Y^{(i)} = f(\vec{X}^{(i)}) + \epsilon$, where f is an unknown function and ϵ is noise. (Also, assume $\mathbf{E}[\epsilon] = 0$.)
- The goal in regression is to recover the unknown function f.

Background: Regression



- Suppose you observe $(\vec{X}^{(i)}, Y^{(i)})_{i=1}^{n}$ where $Y^{(i)} = f(\vec{X}^{(i)}) + \epsilon$, where f is an unknown function and ϵ is noise. (Also, assume $\mathbf{E}[\epsilon] = 0$.)
- The goal in regression is to recover the unknown function f.
- In *linear regression*, we assume f is linear.

Background: Regression



- Suppose you observe $(\vec{X}^{(i)}, Y^{(i)})_{i=1}^{n}$ where $Y^{(i)} = f(\vec{X}^{(i)}) + \epsilon$, where f is an unknown function and ϵ is noise. (Also, assume $\mathbf{E}[\epsilon] = 0$.)
- The goal in regression is to recover the unknown function f.
- In *linear regression*, we assume f is linear.
- if we do not assume a particular form for f, we can use *nonparametric regression*.

Background: Sparse regression

- When $\vec{X}$ is high dimensional, classical regression techniques perform poorly.

Background: Sparse regression

- When $\vec{X}$ is high dimensional, classical regression techniques perform poorly.
- If the true function f only depends on a small number of components in $\vec{X}$, we can still do well if we use *sparse* regression methods.

Background: Sparse regression

- When $\vec{X}$ is high dimensional, classical regression techniques perform poorly.
- If the true function f only depends on a small number of components in $\vec{X}$, we can still do well if we use *sparse* regression methods.

	Classical	*Sparse*
Linear	Ordinary Least-Squares (Gauss 1975?)	Elastic net (Zou 2008)
Nonpar.	LOWESS (Cleveland 1979)	Random forests (Breiman 2001)

Our proposal

Suppose we observe pairs $(X_i, Y_i)_{i=1}^n$ iid from density $p(x, y)$.

1. Estimate a (sparse) regression model for $\mathbf{E}[y|x]$.
2. Assess the *prediction accuracy* of the model using *identification risk*
3. Use the identification risk to obtain a lower bound for the mutual information $I(X; Y)$

Multiple-response regression

- Pairs $(x_i, y_i)_{i=1}^n$, where X is p-dimensional and Y is q-dimensional.
- Data matrices $\boldsymbol{X}_{n\times p}$, $\boldsymbol{Y}_{n\times q}$.
- For each column of Y, fit sparse model $Y^{(i)} \approx X^T\beta^{(i)} + \epsilon$, e.g. by using elastic net (Zou 2008),

$$\hat{\beta}^{(i)} = \text{argmin}_\beta ||\boldsymbol{X}^T\beta^{(i)} - Y^{(i)}||^2 + \lambda_2||\beta^{(i)}||_2^2 + \lambda_1||\beta^{(i)}||_1$$

- Or, fit a *random forest* model for each column of Y (Breiman 2001)

Regression vs Identification loss

- Independent *test set* $(x_i^*, y_i^*)_{i=1}^k$.
- Use model to predict $\hat{y}_i^* = (x_i^*)^T \hat{B}$ for $i = 1, \ldots, k$.

Two ways to evaluate the predictive accuracy of the regression model:

- Regression (mean squared-error) loss:

$$\text{MSE} = \frac{1}{k} \sum_{i=1}^{k} ||y_i^* - \hat{y}_i^*||^2.$$

- Identification loss (Kay 2008):

$$\text{IdLoss}_k = \frac{1}{k} \sum_{i=1}^{k} (1 - I\{\hat{y}_i^* \text{ is nearest neighbor of } y_i^*\}).$$

[note: point out that while idloss was introduced by Kay, that we are the first to consider theory, and add slide about 1d example/robustness]

Identification loss and mutual information

- Define the identification risk as the expected identification loss

$$\text{IdRisk}_k = \mathbf{E}[\text{IdLoss}_k]$$

- Define the Bayes risk as the identification risk given the *true* model parameters. Hence,

$$\text{BayesRisk}_k \leq \text{IdRisk}_k.$$

- **Theorem.** (Z., Benjamini 2016) There exists a function g_k such that

$$I(X;Y) \geq g_k(\text{BayesRisk}_k).$$

- Resulting estimator:

$$\hat{I}_{IdLoss}(X;Y) = g_k(\text{IdLoss}_k).$$

Cross-validated loss

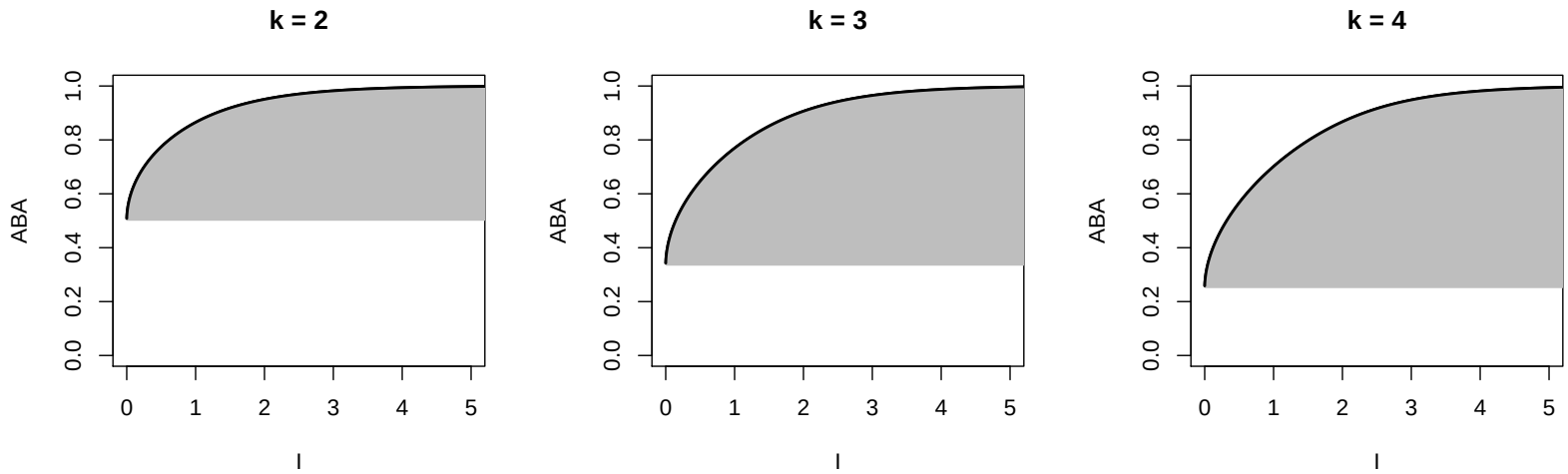
Leave-k-out cross-validation (LkoCV) can be used for both squared-error loss and identification loss.

- Start with a dataset $(x_i, y_i)_{i=1}^N$.
- Let $n = N - k$. Consider all $\binom{N}{k}$ partitions of the dataset into a test set $(\boldsymbol{X}, \boldsymbol{Y})$ and training set $(\boldsymbol{X}^*, \boldsymbol{Y}^*)$.
- For each partition, compute the loss.
- Define the LkoCV loss as the average loss over $\binom{N}{k}$ partitions.

Computational note. One can subsample to avoid computing all $\binom{N}{k}$ partitions. In particular, if $m = N/k$, then one can use m-fold cross-validation which uses m partitions that have disjoint test sets.

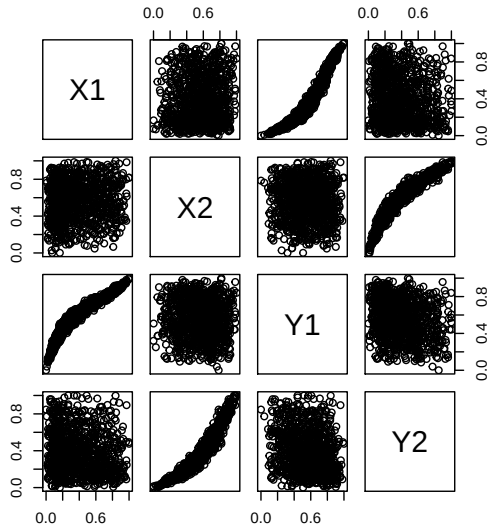
Functions

Illustration of $C_k = g_k^{-1}$



As information increases, the maximal identification risk goes to 0. [note: pictures need to be rotated]

Simulation



- Generate data: $(Y_1, Y_2) = f(X_1, X_2, \epsilon)$ where f is nonlinear.
- Add extra noise dimensions $X_3, X_4, \ldots$.
- $n = 1000$.
- Compare Nearest-Neighbor estimator (Mnatsakov et al, 2008, implemented in `FNN`) with our method using *Random Forest*.

Simulation Results

True $I(X; Y) = 4.615$.

Extra dim	NN	RF $k = 10$	RF $k = 20$
0	**4.445**	3.989	3.924
1	3.040	**3.645**	3.610
2	1.773	**3.249**	3.182

Section 2

Theory

Functional formulation

Bayes identification risk $\text{BayesRisk}_k[p(x,y)]$ and mutual information $\text{I}[p(x,y)]$ are both *functionals* of $p(x,y)$.

$$\text{BayesAcc}_k[p(x,y)] = \frac{1}{k} \int p_X(x_1) \dots p_X(x_k) \max_{i=1}^{k} p(y|x_i) dx_1 \dots dx_k dy.$$

$$\text{I}[p(x,y)] = \int p(x,y) \log \frac{p(x,y)}{p(x)p(y)} dxdy.$$

where $\text{BayesAcc}_k = 1 - \text{BayesRisk}_k$.

Problem formulation

Take $\iota > 0$, and fix $k \in \{2, 3, ...\}$. Let $p(x, y)$ be a joint density (where (X, Y) could be random vectors of any dimensionality.) Supposing

$$\mathsf{I}[p(x, y)] \leq \iota,$$

then can we find an upper bound, $g_k^{-1}(\iota)$, on $\mathsf{BayesAcc}_k[p(x, y)]$?

Proof outline

1. Reduce problem to optimization over univariate densities.
2. Define the Lagrangian functional

$$\mathcal{L}[q(x)] = -\text{BayesAcc}_k[q(x)] + \lambda \int_0^1 q(x)dx + \nu \text{I}[q(x)]$$

   which maps the univariate density $q(x)$ to a real number.
3. Compute the functional derivative of the Lagrangian

$$\nabla \mathcal{L}[q](x) = -t^{k-1} + \lambda + \nu(1 + \log q(x))$$

4. Set $\nabla \mathcal{L}[q](x) = 0$, yielding

$$q^*(t) = \alpha e^{\beta t^{k-1}}.$$

5. Check that local minimizer is global minimizer.

Result

Theorem. For any $\iota > 0$, there exists $\beta_\iota \geq 0$ such that defining

$$q_\beta(t) = \frac{\exp[\beta t^{k-1}]}{\int_0^1 \exp[\beta t^{k-1}]},$$

we have

$$\int_0^1 q_{\beta_\iota}(t) \log q_{\beta_\iota}(t) dt = \iota.$$

Then,

$$\sup_{I(X;Y)=\iota} \text{BayesAcc}_k = \int_0^1 q_{\beta_\iota}(t) t^{k-1} dt = g_k^{-1}(\iota).$$

Section 3

Conclusion

Application to gene expression time series

to be contd

Related work and future directions

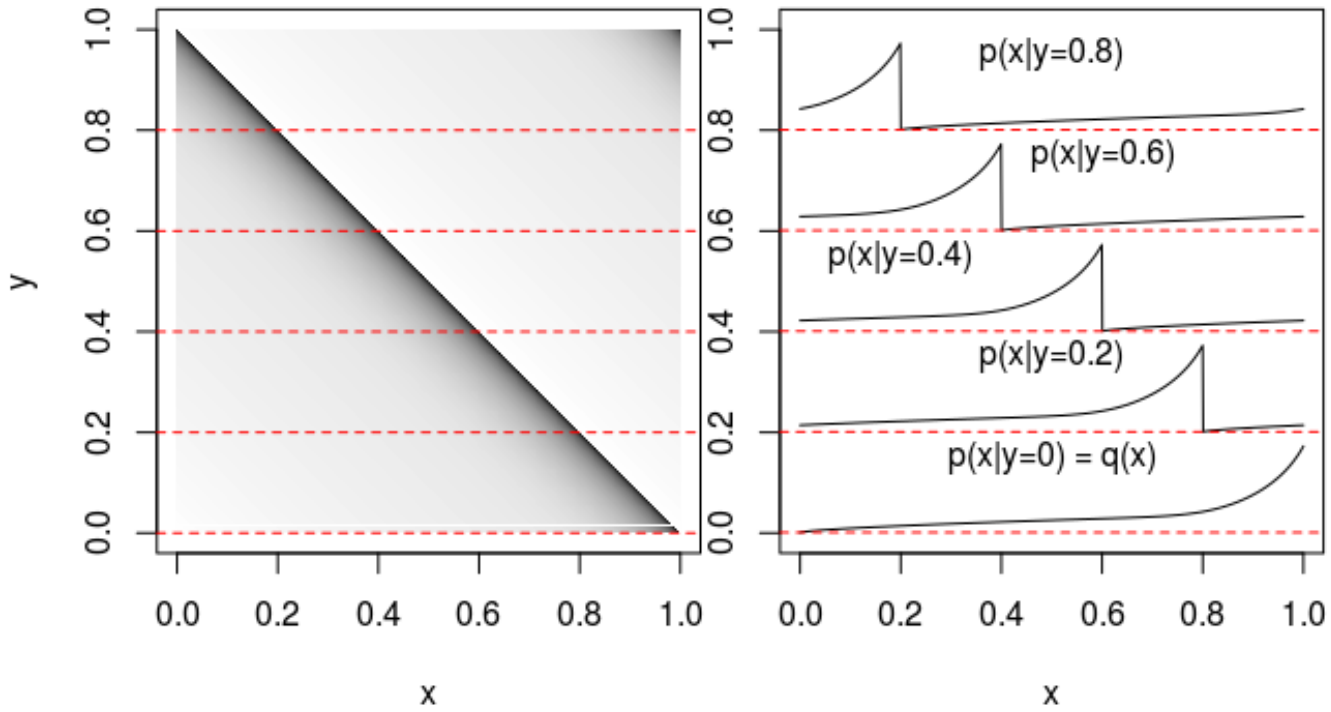
to be contd

Section 4

The End

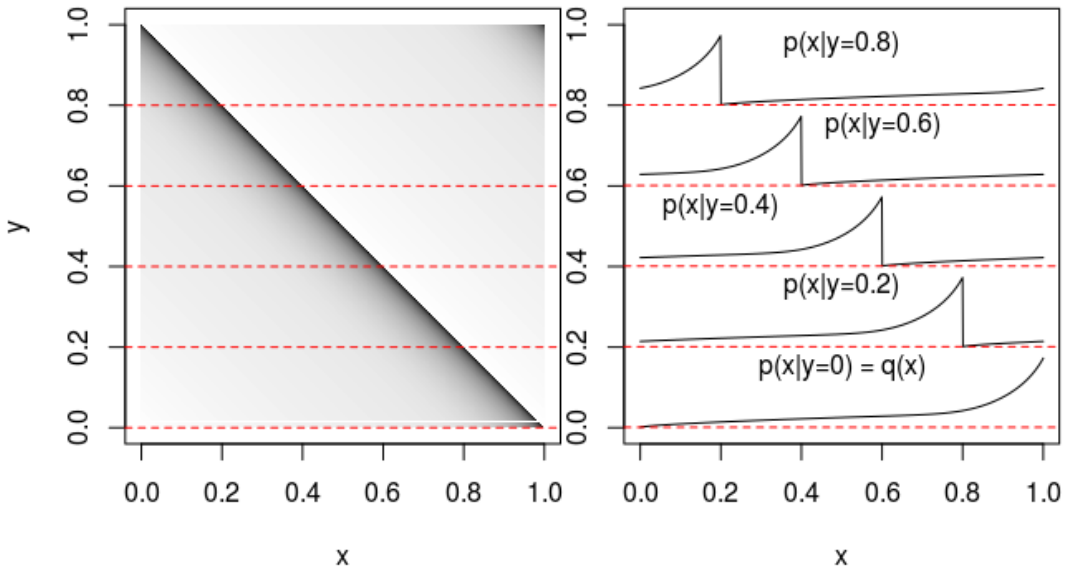
Reduced Problem

Rather than show the whole proof, we consider a simplified problem to illustrate the methods.



Actually, the simplified problem is equivalent to the full problem and we get the same answer (but this is non-trivial).

Reduced Problem



- $p(x, y)$ on unit square with uniform marginals.
- The conditional distributions $p(x|y)$ are just "shifted" copies of a common density, $q(x)$, on $[0, 1]$

$$p(x|y) = q(x - y + I\{x < y\})$$

- Furthermore, $q(x)$ is increasing in x.

Simplified formulae

The information and average Bayes error can be written in terms of $q(x)$.

$$\mathrm{I}[p(x,y)] = \int_0^1 q(x) \log q(x) dx$$

$$\mathrm{BayesAcc}_k[p(x,y)] = \int_{[0,1]^k} \max_{i=1}^{k} q(x_i) dx_1 \cdots dx_k$$

Simplified formulae

Overload the notation and "redefine" information and average Bayes error as functionals of $q(x)$.

$$\mathsf{I}[q(x)] \stackrel{def}{=} \int_0^1 q(x) \log q(x) dx$$

$$\mathsf{BayesAcc}_k[q(x)] \stackrel{def}{=} \frac{1}{k} \int_{[0,1]^k} \max_{i=1}^{k} q(x_i) dx_1 \cdots dx_k$$

Optimization problem

We now pose the question: how do we find $q(x)$ which maximizes $\text{BayesAcc}_k[q(x)]$ subject to $\text{I}[q(x)] \leq \iota$?

- *Domain of the optimization*: Recall that $q(x)$ satisfies $q(x) \geq 0$, $\int_0^1 q(x)dx = 1$, and is increasing in x. Let $\mathcal{Q}$ denote the space of functions on $[0,1] \to [0,\infty)$ which are increasing in x.
- *Constraints*: We have two remaining constraints, $\text{I}[q(x)] \leq \iota$ and $\int_0^1 q(x)dx = 1$.

Hence the problem is

$$\text{maximize}_{q(x)\in\mathcal{Q}} \ \text{BayesAcc}_k[q(x)] \text{ subject to } \int_0^1 q(x)dx = 1 \text{ and } \text{I}[q(x)] \leq \iota.$$

Optimization problem

$$\text{maximize}_{q(x)\in\mathcal{Q}}\ \text{BayesAcc}_k[q(x)] \text{ subject to } \int_0^1 q(x)dx = 1 \text{ and } \mathsf{I}[q(x)] \leq \iota.$$

- Does a solution exist? *Yes*, because the space of measures with density $q(x)$ satisfying $\mathsf{I}[q(x)] \leq \iota$ is tight, and both the constraints and objective are continuous wrt to the topology of weak convergence.
- Given a solution $q^*(x)$ exists, there exist Lagrange multipliers $\lambda \in \mathbb{R}$ and $\nu > 0$ such that q^* minimizes

$$\begin{aligned}\mathcal{L}[q(x)] &= -\text{BayesAcc}_k[q(x)] + \lambda \int_0^1 q(x)dx + \nu \mathsf{I}[q(x)] \\ &= \int_0^1 (-t^{k-1} + \lambda + \nu \log q(x))q(x)dx.\end{aligned}$$

Functional derivatives

- Taylor explansions are a useful trick for computing functional derivatives
- We can compute the functional derivative of $\mathcal{L}[q(x)]$ by writing

$$\begin{aligned}
&\mathcal{L}[q(x) + \epsilon\xi(x)] \\
&= \int_0^1 (-t^{k-1} + \lambda + \nu \log(q(x) + \epsilon\xi(x)))(q(x) + \epsilon\xi(x))dx. \\
&\approx \int (q(x)+\epsilon\xi(x))(-t^{k-1} + \lambda + \nu\{\log q(x) + \frac{\epsilon\xi(x)}{q(x)}\})dx \\
&\approx \mathcal{L}[q(x)]+ \int_0^1 (-t^{k-1} + \lambda + \nu(1 + \log q(x))\epsilon\xi(x)dx.
\end{aligned}$$

- Hence

$$\nabla\mathcal{L}[q](x) = -t^{k-1} + \lambda + \nu(1 + \log q(x))$$

Variational magic!

Suppose we set the functional derivative to 0,

$$0 = \nabla \mathcal{L}[q](t) = -t^{k-1} + \lambda + \nu + \nu \log q(t).$$

Then we conclude that the optimal $q^*(t)$ takes the form

$$q^*(t) = \alpha e^{\beta t^{k-1}}$$

for some $\alpha > 0$, $\beta > 0$.
From the constraint $\int q(t)dt = 1$, we get

$$q_\beta(t) = \frac{e^{\beta t^{k-1}}}{\int e^{\beta t^{k-1}} dt}.$$

Result

Theorem. For any $\iota > 0$, there exists $\beta_\iota \geq 0$ such that defining

$$q_\beta(t) = \frac{\exp[\beta t^{k-1}]}{\int_0^1 \exp[\beta t^{k-1}]},$$

we have

$$\int_0^1 q_{\beta_\iota}(t) \log q_{\beta_\iota}(t) dt = \iota.$$

Then,

$$\sup_{I(X;Y)=\iota} \text{BayesAcc}_k = \int_0^1 q_{\beta_\iota}(t) t^{k-1} dt = g_k^{-1}(\iota).$$