

# Learning gradients: prescriptive models

Sayan Mukherjee

Department of Statistical Science  
Institute for Genome Sciences & Policy  
Department of Computer Science  
Duke University

May 11, 2007

## Relevant papers

- Learning Coordinate Covariances via Gradients. Sayan Mukherjee, Ding-Xuan Zhou; Journal of Machine Learning Research, 7(Mar):519–549, 2006.
- Estimation of Gradients and Coordinate Covariation in Classification. Sayan Mukherjee, Qiang Wu; Journal of Machine Learning Research, 7(Nov):2481–2514, 2006.
- Learning Gradients and Feature Selection on Manifolds. Sayan Mukherjee, Qiang Wu, Ding-Xuan Zhou; Annals of Statistics, submitted.
- Learning Gradients: simultaneous regression and inverse regression. Mauro Maggioni, Sayan Mukherjee, Qiang Wu; Journal of Machine Learning Research, in preparation.

## Table of contents

- 1 Regression and inverse regression
  - Learning gradients: a justification for inverse regression
- 2 Estimating gradients
  - Nonparametric kernel model
  - Convergence of estimate
- 3 Break for pictures
- 4 Gradients on (Riemannian) manifolds
- 5 Dimension reduction
- 6 More pictures
- 7 Graphical models
- 8 Open problems

## Generative vs. predictive modelling

Given data =  $\{Z_i = (x_i, y_i)\}_{i=1}^n$  with  $Z_i \stackrel{iid}{\sim} \rho(X, Y)$ .

$X \in \mathcal{X} \subset \mathbb{R}^p$  and  $Y \subset \mathbb{R}$  and  $p \gg n$ .

## Generative vs. predictive modelling

Given data =  $\{Z_i = (x_i, y_i)\}_{i=1}^n$  with  $Z_i \stackrel{iid}{\sim} \rho(X, Y)$ .

$X \in \mathcal{X} \subset \mathbb{R}^p$  and  $Y \subset \mathbb{R}$  and  $p \gg n$ .

Two options

- 1 discriminative or regression  $Y|X$
- 2 generative  $X|Y$  (sometimes called inverse regression)

## Motivation and related work

Data generated by measuring thousands of variables lies on or near a low-dimensional manifold.

## Motivation and related work

Data generated by measuring thousands of variables lies on or near a low-dimensional manifold.

Manifold learning: LLE, ISOMAP, Laplacian Eigenmaps, Hessian Eigenmaps.

## Motivation and related work

Data generated by measuring thousands of variables lies on or near a low-dimensional manifold.

Manifold learning: LLE, ISOMAP, Laplacian Eigenmaps, Hessian Eigenmaps.

Simultaneous dimensionality reduction and regression: SIR, MAVE, SAVE.

# Regression

Given  $X \in \mathcal{X} \subset \mathbb{R}^p$  and  $Y \subset \mathbb{R}$  and  $p \gg n$  and  $\rho(X, Y)$  we want  $Y|X$ .

A natural idea

$$f_r(x) = \arg \min[\text{var}(f)] = \arg \min \mathbb{E}_Y (Y - f(X))^2,$$

and  $f_r(x) = \mathbb{E}_Y[Y|x]$  provides a summary of  $Y|X$ .

## Inverse regression

Given  $X \in \mathcal{X} \subset \mathbb{R}^p$  and  $Y \subset \mathbb{R}$  and  $p \gg n$  and  $\rho(X, Y)$  we want  $X|Y$ .

$\Omega = \text{cov}(X|Y)$  provides a summary of  $X|Y$ .

# Inverse regression

Given  $X \in \mathcal{X} \subset \mathbb{R}^p$  and  $Y \subset \mathbb{R}$  and  $p \gg n$  and  $\rho(X, Y)$  we want  $X|Y$ .

$\Omega = \text{cov}(X|Y)$  provides a summary of  $X|Y$ .

- 1  $\Omega_{jj}$  – relevance of variable with respect to label
- 2  $\Omega_{ij}$  – covariation with respect to label

# Learning gradients

Given data =  $\{Z_i = (x_i, y_i)\}_{i=1}^n$  with  $Z_i \stackrel{iid}{\sim} \rho(X, Y)$ .

We will simultaneously estimate  $f_r(x)$  and  $\nabla f_r = \left(\frac{\partial f_r}{\partial x^1}, \dots, \frac{\partial f_r}{\partial x^p}\right)^T$ .

# Learning gradients

Given data =  $\{Z_i = (x_i, y_i)\}_{i=1}^n$  with  $Z_i \stackrel{iid}{\sim} \rho(X, Y)$ .

We will simultaneously estimate  $f_r(x)$  and  $\nabla f_r = \left(\frac{\partial f_r}{\partial x^1}, \dots, \frac{\partial f_r}{\partial x^p}\right)^T$ .

- 1 regression:  $f_r(x)$
- 2 inverse regression: outer product of gradients

$$\Gamma = \mathbb{E}[\nabla f_r \otimes \nabla f_r] \text{ or}$$

$$\Gamma_{ij} = \left\langle \frac{\partial f_r}{\partial x^i}, \frac{\partial f_r}{\partial x^j} \right\rangle.$$

## Linear case

We start with the linear case

$$y = w \cdot x + \varepsilon, \quad \varepsilon \stackrel{iid}{\sim} \text{No}(0, \sigma^2).$$

$$\Sigma_X = \text{cov}(X), \quad \sigma_Y^2 = \text{var}(Y).$$

$$\Omega = \text{cov}(X|Y) = \frac{1}{\sigma_Y^2} \Sigma_X \Gamma \Sigma_X.$$

## Linear case

We start with the linear case

$$y = w \cdot x + \varepsilon, \quad \varepsilon \stackrel{iid}{\sim} \text{No}(0, \sigma^2).$$

$$\Sigma_X = \text{cov}(X), \quad \sigma_Y^2 = \text{var}(Y).$$

$$\Omega = \text{cov}(X|Y) = \frac{1}{\sigma_Y^2} \Sigma_X \Gamma \Sigma_X.$$

$\Gamma$  and  $\Omega$  are equivalent modulo rotation and scale.

## Nonlinear case

For smooth  $f(x)$

$$y = f(x) + \varepsilon, \quad \varepsilon \stackrel{iid}{\sim} \text{No}(0, \sigma^2).$$

$\Omega = \text{cov}(X|Y)$  not so clear.

## Nonlinear case

Partition into sections and compute local quantities

$$\mathcal{X} = \bigcup_{i=1}^{\mathcal{I}} \mathcal{X}_i$$

$$\Omega_i = \text{cov}(X_{\mathcal{X}_i} | Y_{\mathcal{X}_i})$$

$$\Sigma_i = \text{cov}(X_{\mathcal{X}_i})$$

$$\sigma_i^2 = \text{var}(Y_{\mathcal{X}_i}).$$

## Nonlinear case

Partition into sections and compute local quantities

$$\mathcal{X} = \bigcup_{i=1}^{\mathcal{I}} \mathcal{X}_i$$

$$\Omega_i = \text{cov}(X_{\mathcal{X}_i} | Y_{\mathcal{X}_i})$$

$$\Sigma_i = \text{cov}(X_{\mathcal{X}_i})$$

$$\sigma_i^2 = \text{var}(Y_{\mathcal{X}_i}).$$

$$\Gamma = \sum_{i=1}^{\mathcal{I}} \sigma_i^2 \Sigma_i^{-1} \Omega_i \Sigma_i^{-1}.$$

## Taylor expansion and gradients

Given  $D = (Z_1, \dots, Z_n)$  the variance of  $f$  may be approximated as

$$\text{Var}_n(f) = \sum_{i,j=1}^n w_{ij} \left[ y_i - f(x_j) - \nabla f(x_j) \cdot (x_i - x_j) \right]^2,$$

$w_{ij}$  ensures the locality of  $x_i \approx x_j$ .

Regression and inverse regression

**Estimating gradients**

Break for pictures

Gradients on (Riemannian) manifolds

Dimension reduction

More pictures

Graphical models

Open problems

Nonparametric kernel model

Convergence of estimate

## Penalized loss estimator

$$\hat{f}(x) = \arg \min_{f \in \text{bs}} [\text{error on data} + \text{smoothness of function}]$$

## Penalized loss estimator

$$\hat{f}(x) = \arg \min_{f \in \text{bs}} [\text{error on data} + \text{smoothness of function}]$$

$$\text{error on data} = L(f, \text{data}) = \text{Var}_n(f)^2$$

$$\text{smoothness of function} = \|f\|_K^2 = \int |f'(x)|^2 dx$$

$$\text{big function space} = \text{reproducing kernel Hilbert space} = \mathcal{H}_K$$

## Penalized loss estimator

$$\hat{f}(x) = \arg \min_{f \in \mathcal{H}_K} [L(f, \text{data}) + \lambda \|f\|_K^2]$$

The kernel:  $K : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$  e.g.  $K(u, v) = e^{(-\|u-v\|^2)}$ .

The RKHS

$$\mathcal{H}_K = \overline{\left\{ f \mid f(x) = \sum_{i=1}^{\ell} \alpha_i K(x, x_i), \quad x_i \in \mathcal{X}, \alpha_i \in \mathbb{R}, \ell \in \mathbb{N} \right\}}.$$

## Gradient estimate

Nonparametric model

$$(f_D, \vec{f}_D) := \arg \min_{f, \vec{f} \in \mathcal{H}_K^{p+1}} \left[ \sum_{i,j=1}^n w_{ij} \left( y_j - f(x_i) - \vec{f}(x_i) \cdot (x_j - x_i) \right)^2 + \lambda_1 \|f\|_K^2 + \lambda_2 \|\vec{f}\|_K^2 \right],$$

$\mathcal{H}_K^p$  is the space of  $p$  functions  $\vec{f} = (f_1, \dots, f_p)$  where  $f_i \in \mathcal{H}_K$ ,  $\|\vec{f}\|_K^2 = \sum_{i=1}^p \|f_i\|_K^2$ , and  $\lambda_1, \lambda_2 > 0$ .

## Computational efficiency

The computation requires fewer than  $n^2$  parameters and is  $O(n^6)$  time and  $O(pn)$  memory

$$f_D(x) = \sum_{i=1}^n a_{i,D} K(x_i, x), \quad \vec{f}_D(x) = \sum_{i=1}^n c_{i,D} K(x_i, x)$$

with  $a_D = (a_{1,D}, \dots, a_{n,D}) \in \mathbb{R}^n$  and  $c_D = (c_{1,D}, \dots, c_{n,D})^T \in \mathbb{R}^{np}$ .

# Consistency

## Theorem

*Under mild regularity conditions on the distribution and corresponding density, with probability  $1 - \delta$*

$$\|\vec{f}_D - \nabla f\|_{\rho_X} \leq C \log\left(\frac{1}{\delta}\right) n^{-1/p}$$
$$\|f - \hat{f}\|_{\rho_X} \leq C \log\left(\frac{1}{\delta}\right) n^{-1/p}.$$

## Linear example

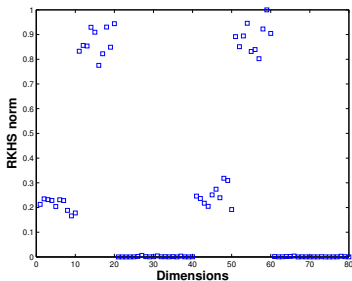
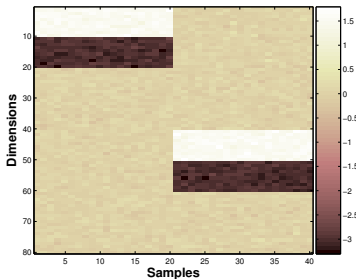
Samples from class  $-1$  were drawn from

$$\begin{aligned}x^j &\sim \text{No}(1.5, 1), \text{ for } j = 1, \dots, 10, \\x^j &\sim \text{No}(-3, 1), \text{ for } j = 11, \dots, 20, \\x^j &\sim \text{No}(0, \sigma_{\text{noise}}), \text{ for } j = 21, \dots, 80,\end{aligned}$$

Samples from class  $+1$  were drawn from

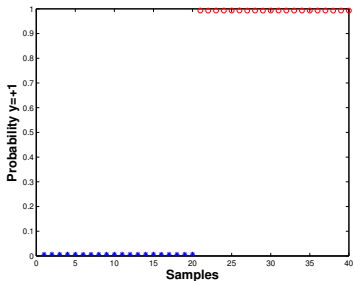
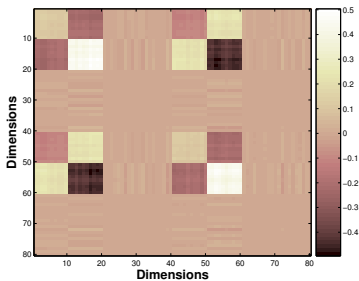
$$\begin{aligned}x^j &\sim \text{No}(1.5, 1), \text{ for } j = 41, \dots, 50, \\x^j &\sim \text{No}(-3, 1), \text{ for } j = 51, \dots, 60, \\x^j &\sim \text{No}(0, \sigma_{\text{noise}}), \text{ for } j = 1, \dots, 40, 61, \dots, 80.\end{aligned}$$

## Linear example



- Regression and inverse regression
- Estimating gradients
- Break for pictures
- Gradients on (Riemannian) manifolds
- Dimension reduction
- More pictures
- Graphical models
- Open problems

# Linear example



## Nonlinear example

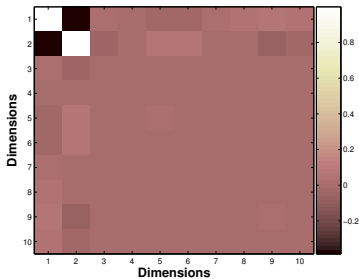
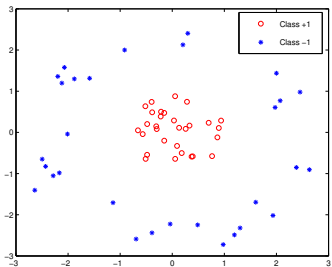
Samples from class +1 were drawn from

$$(x^1, x^2) = (r \sin(\theta), r \cos(\theta)), \text{ where } r \sim U[0, 1] \text{ and } \theta \sim U[0, 2\pi],$$
$$x^j \sim \text{No}(0.0, .2), \text{ for } j = 3, \dots, 200,$$

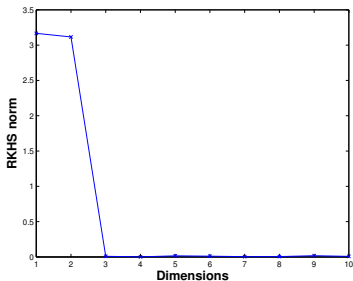
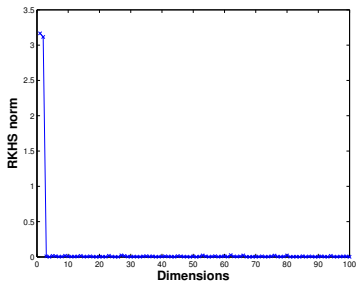
Samples from class -1 were drawn from

$$(x^1, x^2) = (r \sin(\theta), r \cos(\theta)), \text{ where } r \sim U[2, 3] \text{ and } \theta \sim U[0, 2\pi],$$
$$x^j \sim \text{N}(0.0, .2), \text{ for } j = 3, \dots, 200.$$

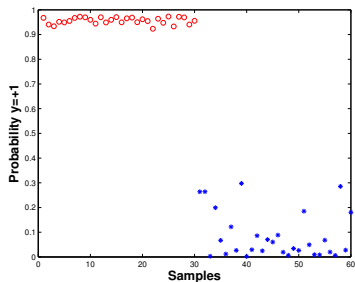
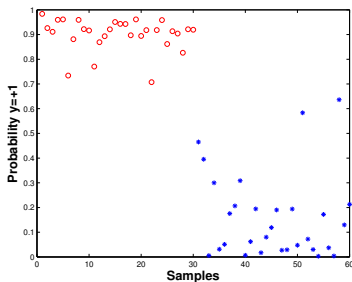
## Nonlinear example



## Nonlinear example



## Nonlinear example



## Restriction to a manifold

Assume the data is concentrated on a manifold  $\mathcal{M} \subset \mathbb{R}^p$  with  $\mathcal{M} \in \mathbb{R}^d$  and there exists an isometric embedding  $\varphi : \mathcal{M} \rightarrow \mathbb{R}^p$ .

## Restriction to a manifold

Assume the data is concentrated on a manifold  $\mathcal{M} \subset \mathbb{R}^p$  with  $\mathcal{M} \in \mathbb{R}^d$  and there exists an isometric embedding  $\varphi : \mathcal{M} \rightarrow \mathbb{R}^p$ .

Given a smooth orthonormal vector field  $\{e_1, \dots, e_d\}$  we can define the gradient on the manifold  $\nabla_{\mathcal{M}} f = (e_1 f, \dots, e_d f)$ .

## Restriction to a manifold

Assume the data is concentrated on a manifold  $\mathcal{M} \subset \mathbb{R}^p$  with  $\mathcal{M} \in \mathbb{R}^d$  and there exists an isometric embedding  $\varphi : \mathcal{M} \rightarrow \mathbb{R}^p$ .

Given a smooth orthonormal vector field  $\{e_1, \dots, e_d\}$  we can define the gradient on the manifold  $\nabla_{\mathcal{M}} f = (e_1 f, \dots, e_d f)$ .

For  $q \in U \subset \mathcal{M}$  a chart  $\mathbf{u} : U \rightarrow \mathbb{R}^d$  satisfying  $\frac{\partial}{\partial u^i}(q) = e_i(q)$  exists.

The Taylor expansion on the manifold around  $q$

$$f(q') \approx f(q) + \nabla_{\mathcal{M}} f(q) \cdot (\mathbf{u}(q') - \mathbf{u}(q)) \text{ for } q' \approx q.$$

## A problem with all manifold methods

The Taylor expansion on the manifold around  $q$

$$f(q') \approx f(q) + \nabla_{\mathcal{M}} f(q) \cdot (\mathbf{u}(q') - \mathbf{u}(q)) \text{ for } q' \approx q.$$

## A problem with all manifold methods

The Taylor expansion on the manifold around  $q$

$$f(q') \approx f(q) + \nabla_{\mathcal{M}} f(q) \cdot (\mathbf{u}(q') - \mathbf{u}(q)) \text{ for } q' \approx q.$$

$\{(q_i, y_i)\}_{i=1}^n \in \mathcal{M} \times Y$  are drawn from the manifold but neither  $\mathcal{M}$  nor a local expression of  $\mathcal{M}$  are given. Also we have only the image  $x_i = \varphi(q_i) \in \mathbb{R}^p$ .

## The standard solution

The Taylor expansion on the manifold around  $q$

$$f(q') \approx f(q) + \nabla_{\mathcal{M}} f(q) \cdot (\mathbf{u}(q') - \mathbf{u}(q)) \text{ for } q' \approx q.$$

## The standard solution

The Taylor expansion on the manifold around  $q$

$$f(q') \approx f(q) + \nabla_{\mathcal{M}} f(q) \cdot (\mathbf{u}(q') - \mathbf{u}(q)) \text{ for } q' \approx q.$$

The Taylor expansion on the manifold around  $x$  in terms of  $f \circ \varphi^{-1} \in \mathbb{R}^p$

$$(f \circ \varphi^{-1})(u) - (f \circ \varphi^{-1})(x) \approx \nabla(f \circ \varphi^{-1})(x) \cdot (u - x) \text{ for } u \approx x.$$

## The standard solution

The Taylor expansion on the manifold around  $q$

$$f(q') \approx f(q) + \nabla_{\mathcal{M}} f(q) \cdot (\mathbf{u}(q') - \mathbf{u}(q)) \text{ for } q' \approx q.$$

The Taylor expansion on the manifold around  $x$  in terms of  $f \circ \varphi^{-1} \in \mathbb{R}^p$

$$(f \circ \varphi^{-1})(u) - (f \circ \varphi^{-1})(x) \approx \nabla(f \circ \varphi^{-1})(x) \cdot (u - x) \text{ for } u \approx x.$$

Due to this equivalence  $\vec{f}_D \approx d\varphi \circ \nabla_{\mathcal{M}} f$

## Improved rate of convergence

### Theorem

*Under mild regularity conditions on the distribution and corresponding density, with probability  $1 - \delta$*

$$\begin{aligned}\|(d\varphi)^* \vec{f}_D - \nabla_{\mathcal{M}} f\|_{\rho_X} &\leq C \log\left(\frac{1}{\delta}\right) n^{-1/d} \\ \|f - \hat{f}\|_{\rho_X} &\leq C \log\left(\frac{1}{\delta}\right) n^{-1/d},\end{aligned}$$

*where  $(d\varphi)^*$  is the dual of the map  $d\varphi$ .*

## Sensitive features

### Proposition

Let  $f$  be a smooth function on  $\mathbb{R}^p$  with gradient  $\nabla f$ . The sensitivity of the function  $f$  along a (unit normalized) direction  $u$  is  $\|u \cdot \nabla\|_K$ .

The  $d$  most sensitive features are those  $\{u_1, \dots, u_d\}$  that are orthogonal and maximize  $\|u_i \cdot \nabla f\|_K$ . A spectral decomposition of  $\Gamma$  is used to compute these features, the eigenvectors corresponding to the  $d$  top eigenvalues.

## Projection of data

The matrix

$$\hat{\Gamma} = \vec{f}_D \otimes \vec{f}_D = c_D^T K c_D,$$

is an empirical estimate of  $\Gamma$ .

## Projection of data

The matrix

$$\hat{\Gamma} = \vec{f}_D \otimes \vec{f}_D = c_D^T K c_D,$$

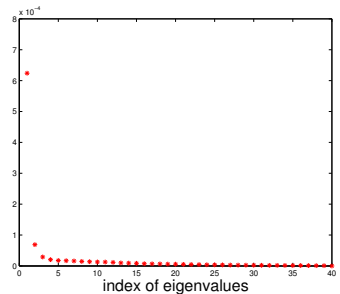
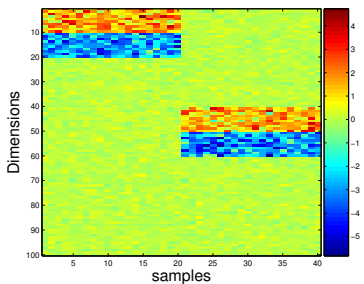
is an empirical estimate of  $\Gamma$ .

Geometry is preserved by projection onto top  $k$ -eigenvectors.

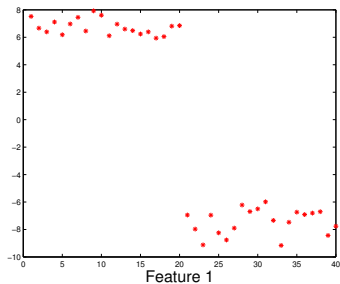
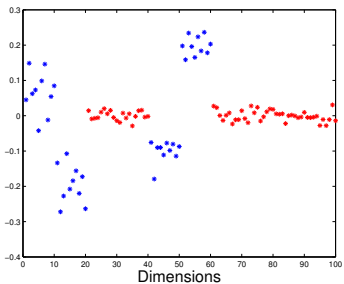
No need to compute the  $p \times p$  matrix, method is  $O(n^2 p + n^3)$  time and  $O(p \times n)$  memory.

- Regression and inverse regression
- Estimating gradients
- Break for pictures
- Gradients on (Riemannian) manifolds
- Dimension reduction
- More pictures
- Graphical models
- Open problems

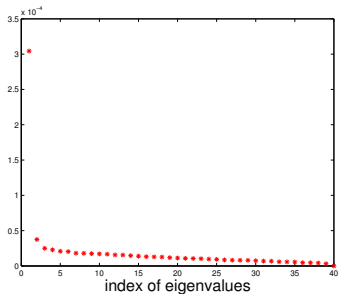
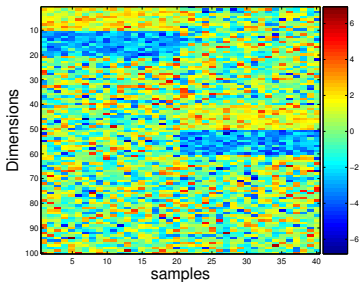
# Linear example



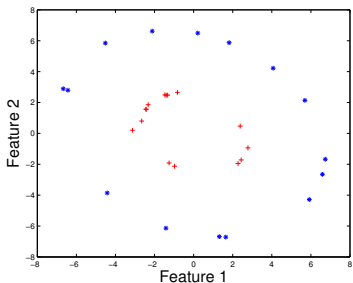
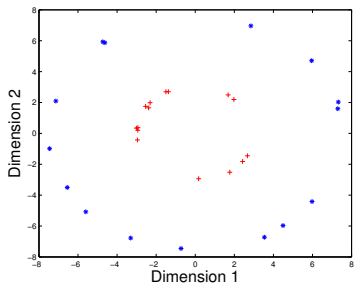
# Linear example



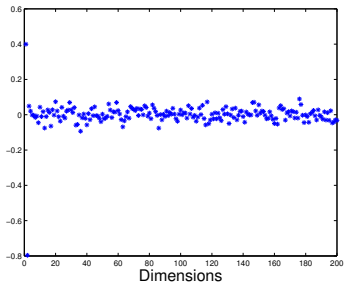
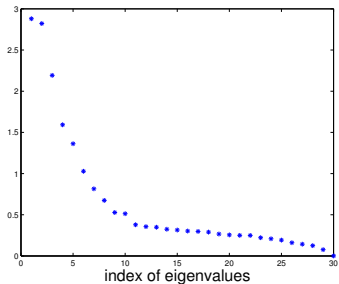
## Linear example



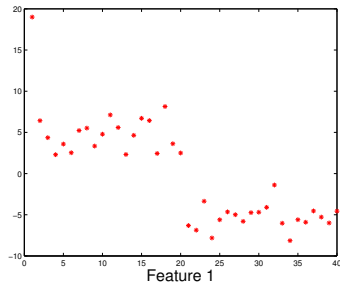
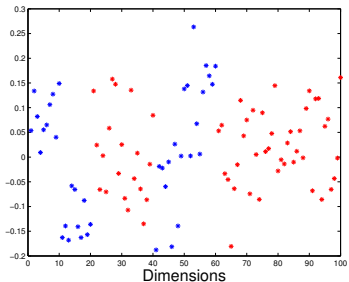
## Nonlinear example



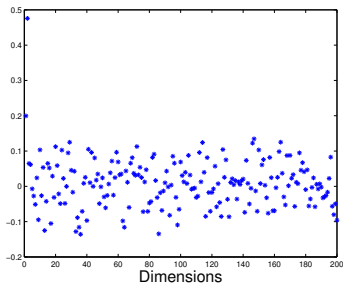
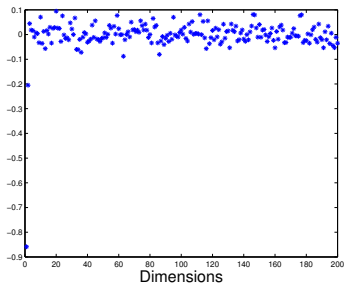
## Nonlinear example



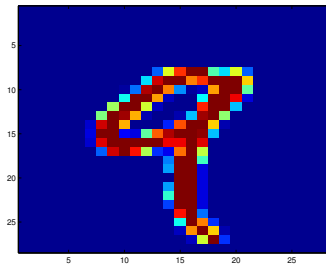
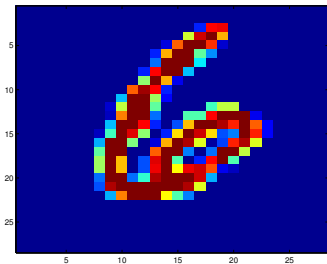
# Nonlinear example



## Nonlinear example

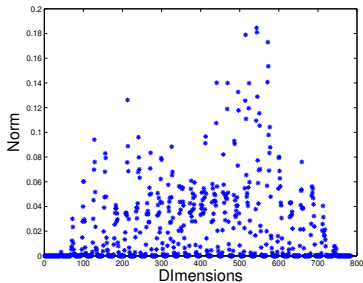
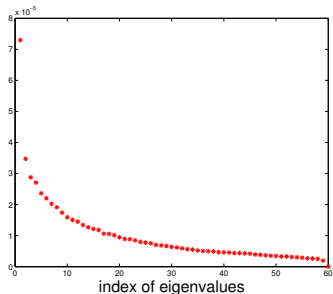


## Digits: "6" vs. "9"



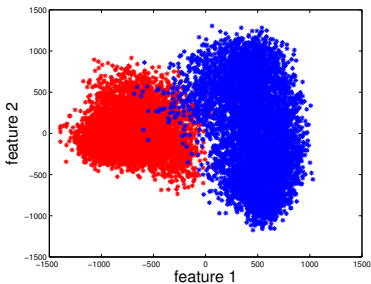
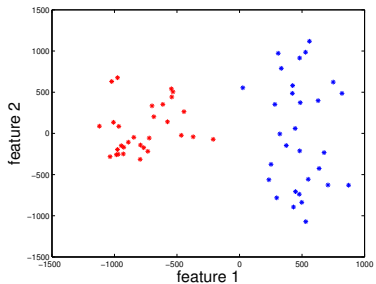
Regression and inverse regression  
Estimating gradients  
Break for pictures  
Gradients on (Riemannian) manifolds  
Dimension reduction  
More pictures  
Graphical models  
Open problems

## Digits: "6" vs. "9"



Regression and inverse regression  
Estimating gradients  
Break for pictures  
Gradients on (Riemannian) manifolds  
Dimension reduction  
More pictures  
Graphical models  
Open problems

## Digits: "6" vs. "9"

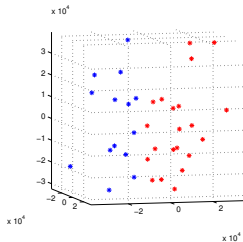
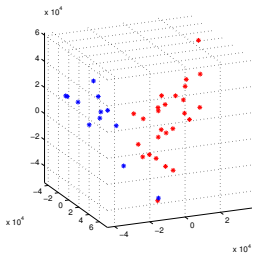


# Leukemia

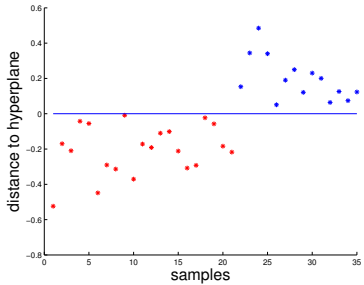
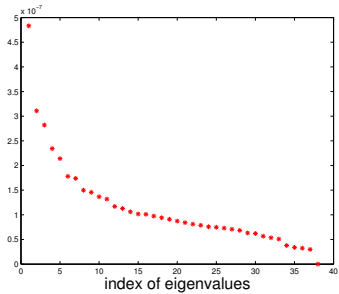
48 samples of AML, 25 samples of ALL,  $p = 7,129$ .

Dataset split into a training set of 38 samples and a test set of 35 samples.

# Leukemia



# Leukemia



## Gauss-Markov graphical models

Give a multivariate normal distribution with covariance matrix  $C$   
the matrix  $P = C^{-1}$  is the conditional independence matrix

$P_{ij} = \text{dependence of } i \leftrightarrow j \mid \text{all other variables.}$

## Gauss-Markov graphical models

Give a multivariate normal distribution with covariance matrix  $C$   
the matrix  $P = C^{-1}$  is the conditional independence matrix

$P_{ij}$  = dependence of  $i \leftrightarrow j$  | all other variables.

Set  $C = \hat{\Gamma}$ . However, many values near zero and how to truncate.

## Multiscale graphical models

Assume for now  $\hat{\Gamma}$  has all positive entries and is normalized to be a Markov matrix.

$$T = \Lambda^{-1/2} \hat{\Gamma} \Lambda^{-1/2},$$

where  $\Lambda$  is the eigenvalue matrix for  $\hat{\Gamma}$ .

## Multiscale graphical models

Note the harmonic expansion

$$(I - T)^{-1} = \sum_{k=1}^{\infty} T^k = \prod_{k=1}^{\infty} (I + T^{2^k}),$$

where  $k$  is path-length in the first inequality

$T^k_{ij}$  = probability  $i \rightarrow j$  in path-length  $k$ .

$(I + T^{2^k})$  factorizes  $(I - T)^{-1}$  into low rank matrices with fewer entries.

## Multiscale graphical models

Note the harmonic expansion

$$(I - T)^{-1} = \sum_{k=1}^{\infty} T^k = \prod_{k=1}^{\infty} (I + T^{2^k}),$$

where  $k$  is path-length in the first inequality

$T^k_{ij}$  = probability  $i \rightarrow j$  in path-length  $k$ .

$(I + T^{2^k})$  factorizes  $(I - T)^{-1}$  into low rank matrices with fewer entries.

This is the idea of diffusion wavelets.

## Summary and toy example

Key quantities

$$\begin{aligned} C &\Leftrightarrow T \\ C^{-1} &\Leftrightarrow (I - T)^{-1}. \end{aligned}$$

so  $\prod (I + T^{2^k})$  factorizes  $C^{-1}$ .

## Summary and toy example

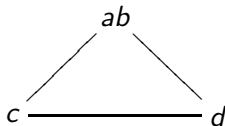
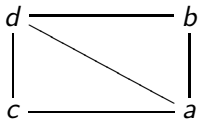
$X_{abcd}$

$$\hat{\Gamma} = \begin{pmatrix} 5 & 3 & -1.5 & 0 \\ 3 & 5 & -0.5 & 2 \\ -1.5 & -0.5 & 5 & -2.5 \\ 0 & 2 & -2.5 & 5 \end{pmatrix}.$$

## Summary and toy example

$X_{abcd}$

$$\hat{\Gamma} = \begin{pmatrix} 5 & 3 & -1.5 & 0 \\ 3 & 5 & -0.5 & 2 \\ -1.5 & -0.5 & 5 & -2.5 \\ 0 & 2 & -2.5 & 5 \end{pmatrix}.$$



## General covariance matrices

Math goes through as above. Interpretation is slightly different.  
Path coefficients (Wright 1921) and path regression (Tukey 1954)

$$X_1 = c_{12}X_2 + c_{13}X_3 + \dots + c_{1m}X_m$$

where  $c_{12}$  is the contribution of  $X_2$  to  $X_1$ .

## General covariance matrices

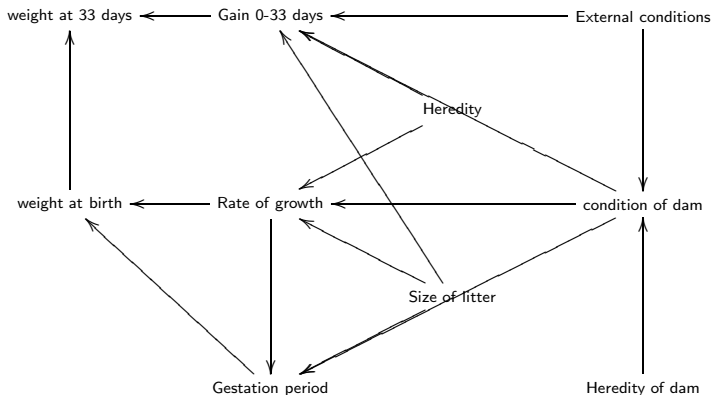
Math goes through as above. Interpretation is slightly different.  
Path coefficients (Wright 1921) and path regression (Tukey 1954)

$$X_1 = c_{12}X_2 + c_{13}X_3 + \dots + c_{1m}X_m$$

where  $c_{12}$  is the contribution of  $X_2$  to  $X_1$ .

Matrix  $\Gamma$  can be thought of as matrix of path coefficients. Powers of  $k$  compute path coefficients of length  $k$  as in Markov case.

## An early graphical model



## Discussion

Lots of work left:

- Semi-supervised setting.

## Discussion

Lots of work left:

- Semi-supervised setting.
- Multi-task setting.

## Discussion

Lots of work left:

- Semi-supervised setting.
- Multi-task setting.
- Bayesian formulation.

## Discussion

Lots of work left:

- Semi-supervised setting.
- Multi-task setting.
- Bayesian formulation.
- Nonlinear projections – diffusion maps.

## Discussion

Lots of work left:

- Semi-supervised setting.
- Multi-task setting.
- Bayesian formulation.
- Nonlinear projections – diffusion maps.
- Noise on-off manifolds.