Machine Learning Methods for Statistical Decision Making

GEORGE V. MOUSTAKIDES
UNIVERSITY OF PATRAS, GREECE

Two - Part Presentation

- Decision Making (Hypothesis Testing - Detection)
- Parameter Estimation

Goal: Consider the two well known problems under
a pure data-driven setup

Decision Making: Outline

- Hypothesis Testing
  - Mathematical Formulation
  - Data Driven Approach
  - The Consistency Property

- Designing Consistent Test
  - Optimization Problems with Consistent Solutions
  - Data Driven Implementation

- Detection in Time Series
  - I.i.d. processes
  - Markov processes

Hypothesis Testing

Mathematical Formulation

For a random vector X we assume the following two hypotheses

$$\mathsf{H}_0: \quad X \sim \mathsf{f}_0(X), \;\; \mathbb{P}(\mathsf{H}_0)$$
$$\mathsf{H}_1: \quad X \sim \mathsf{f}_1(X), \;\; \mathbb{P}(\mathsf{H}_1)$$

For every X need to decide if it comes from H_0 or H_1

Decide using a *Decision Function* $\mathsf{D}(X) \in \{0, 1\}$

Would like to optimize $\mathsf{D}(X)$

Plethora of applications in diverse scientific fields!!!

Bayesian Approach

Minimize decision error probability

$$\min_{\mathsf{D}}\Big\{\mathbb{P}(\mathsf{D}=1|\mathsf{H}_0)\mathbb{P}(\mathsf{H}_0)+\mathbb{P}(\mathsf{D}=0|\mathsf{H}_1)\mathbb{P}(\mathsf{H}_1)\Big\}$$

$$\frac{\mathsf{f}_1(X)}{\mathsf{f}_0(X)}\underset{\mathsf{H}_0}{\overset{\mathsf{H}_1}{\gtreqless}}\frac{\mathbb{P}(\mathsf{H}_0)}{\mathbb{P}(\mathsf{H}_1)}\quad\equiv\quad\frac{\mathsf{f}_1(X)\mathbb{P}(\mathsf{H}_1)}{\mathsf{f}_0(X)\mathbb{P}(\mathsf{H}_0)}\underset{\mathsf{H}_0}{\overset{\mathsf{H}_1}{\gtreqless}}1$$

For $\omega(\mathsf{r})$ strictly increasing

$$\mathsf{r}(X)\underset{\mathsf{H}_0}{\overset{\mathsf{H}_1}{\gtreqless}}1\quad\equiv\quad\omega\big(\mathsf{r}(X)\big)\underset{\mathsf{H}_0}{\overset{\mathsf{H}_1}{\gtreqless}}\omega(1),\qquad \mathsf{r}(X)=\frac{\mathsf{f}_1(X)\mathbb{P}(\mathsf{H}_1)}{\mathsf{f}_0(X)\mathbb{P}(\mathsf{H}_0)}$$

Neyman-Pearson Approach

$$\mathsf{H}_0: \quad X \sim \mathsf{f}_0(X), \quad \mathbb{P}(\mathsf{H}_0)$$
$$\mathsf{H}_1: \quad X \sim \mathsf{f}_1(X), \quad \mathbb{P}(\mathsf{H}_1)$$

Maximize detection probability $\mathbb{P}(\mathsf{D}=1|\mathsf{H}_1)$

subject to false alarm probability constraint $\mathbb{P}(\mathsf{D}=1|\mathsf{H}_0) \leq \alpha$

$$\frac{\mathsf{f}_1(X)}{\mathsf{f}_0(X)} \underset{\mathsf{H}_0}{\overset{\mathsf{H}_1}{\gtreqless}} \lambda, \quad \mathbb{P}\left(\frac{\mathsf{f}_1(X)}{\mathsf{f}_0(X)} \geq \lambda \Big| \mathsf{H}_0\right) = \alpha$$

For $\omega(\mathsf{r})$ strictly increasing

$$\omega\big(\mathsf{r}(X)\big) \underset{\mathsf{H}_0}{\overset{\mathsf{H}_1}{\gtreqless}} \eta, \quad \mathbb{P}\Big(\omega\big(\mathsf{r}(X)\big) \geq \eta \Big| \mathsf{H}_0\Big) = \alpha, \qquad \mathsf{r}(X) = \frac{\mathsf{f}_1(X)}{\mathsf{f}_0(X)}$$

Data Driven Approach

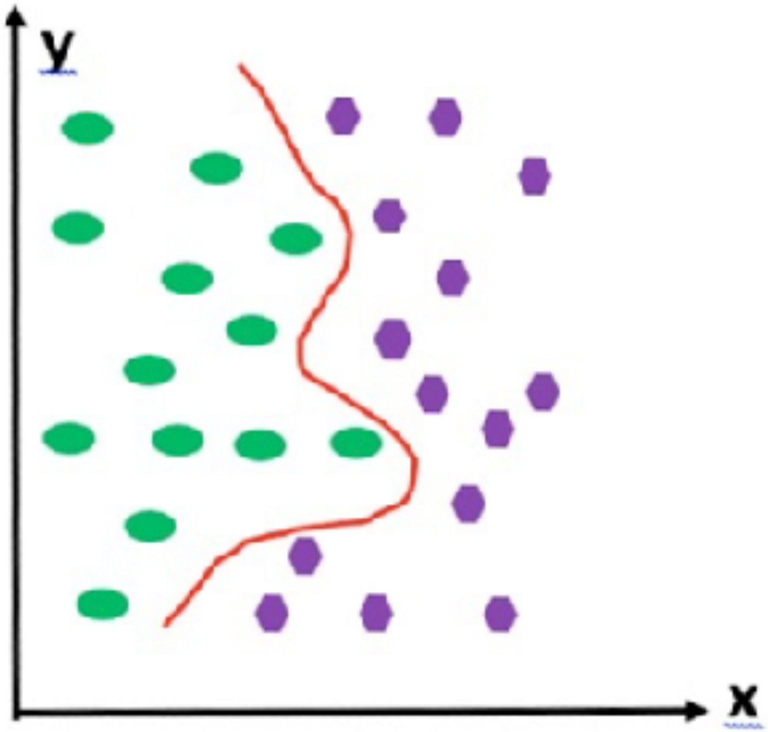
$$\mathsf{H}_0: \quad X \sim \mathsf{f}_0(X), \quad \mathbb{P}(\mathsf{H}_0)$$
$$\mathsf{H}_1: \quad X \sim \mathsf{f}_1(X), \quad \mathbb{P}(\mathsf{H}_1)$$

$$X_1^0 \;\; X_2^0 \;\; \ldots \;\; X_{n_0}^0$$
$$X_1^1 \;\; X_2^1 \;\; \ldots \;\; X_{n_1}^1$$

Sampled from f_0

Sampled from f_1

$$\mathbb{P}(\mathsf{H}_i) \approx \frac{n_i}{n_0 + n_1}$$



Design border to separate the two datasets

What is the best border ?

$$\text{All } X: \frac{\mathsf{f}_1(X)\mathbb{P}(\mathsf{H}_1)}{\mathsf{f}_0(X)\mathbb{P}(\mathsf{H}_0)} = 1$$

Instead of a “border”, design a decision like function $\mathsf{v}(X)$

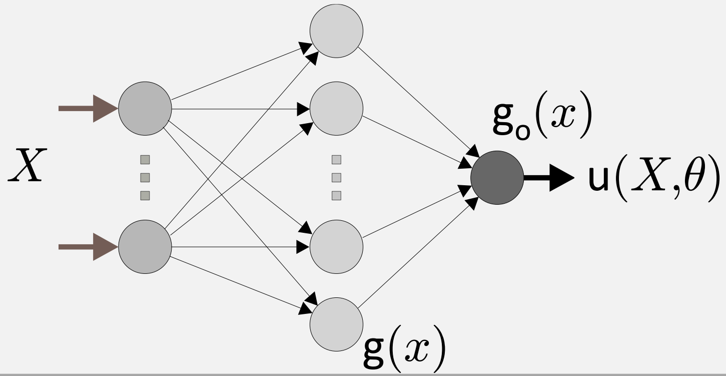
$$\mathsf{v}(X) = \begin{cases} -1 & \text{when } X \text{ from } \mathsf{H}_0 \\ 1 & \text{when } X \text{ from } \mathsf{H}_1. \end{cases}$$

Designing a function $\mathsf{v}(X)$ when information is in the form of data is challenging.

We replace $\mathsf{v}(X)$ with a neural network $\mathsf{u}(X,\theta)$ and select parameters θ

Cybenko 1989 (universal approximation)

For sufficiently large neural network $\mathsf{u}(X,\theta)$ we can find suitable parameters θ such that we can approximate arbitrarily close any function $\mathsf{v}(X)$



$$|\mathsf{v}(X) - \mathsf{u}(X,\theta)| \leq \epsilon$$

Select a neural network configuration $\mathsf{u}(X,\theta)$ and optimize network parameters θ by defining the distance

$$\mathsf{J}(\theta) = \frac{1}{n_0 + n_1} \left\{ \sum_{i=1}^{n_0} \Big(-1 - \mathsf{u}(X_i^0, \theta) \Big)^2 + \sum_{j=1}^{n_1} \Big(1 - \mathsf{u}(X_j^1, \theta) \Big)^2 \right\}$$

and solving the optimization problem

$$\min_{\theta} \mathsf{J}(\theta) \;\Rightarrow\; \theta_{\mathsf{o}} \;\Rightarrow\; \mathsf{u}(X, \theta_{\mathsf{o}})$$

We use the resulting function $\mathsf{u}(X,\theta_{\mathsf{o}})$ to make a decision for any new data X as follows:

$$\mathsf{u}(X, \theta_{\mathsf{o}}) \underset{\mathsf{H}_0}{\overset{\mathsf{H}_1}{\gtreqless}} 0$$

Works “well”!! Why??

Understanding using Asymptotic Analysis

$$n_0, n_1 \to \infty, \qquad \mathsf{u}(X,\theta) \to \mathsf{v}(X)$$

$$\mathsf{J}(\theta) = \frac{n_0}{n_0+n_1} \frac{1}{n_0} \sum_{i=1}^{n_0} \Big(1 + \mathsf{u}(X_i^0,\theta)\Big)^2 + \frac{n_1}{n_0+n_1} \frac{1}{n_1} \sum_{j=1}^{n_1} \Big(1 - \mathsf{u}(X_j^1,\theta)\Big)^2$$

$$\mathsf{J}(\mathsf{v}) = \mathbb{P}(\mathsf{H}_0)\mathbb{E}_0\left[\Big(1+\mathsf{v}(X)\Big)^2\right] + \mathbb{P}(\mathsf{H}_1)\mathbb{E}_1\left[\Big(1-\mathsf{v}(X)\Big)^2\right]$$

$$\min_\theta \mathsf{J}(\theta) \to \min_{\mathsf{v}} \mathsf{J}(\mathsf{v})$$

$$\theta_o \Rightarrow \mathsf{u}(X,\theta_o) \approx \mathsf{v}_o(X)$$

$$\mathbb{E}_1\left[\Big(1-\mathsf{v}(X)\Big)^2\right] = \mathbb{E}_0\left[\Big(1-\mathsf{v}(X)\Big)^2 \frac{\mathsf{f}_1(X)}{\mathsf{f}_0(X)}\right]$$

$$\mathsf{J}(\mathsf{v}) = \mathbb{P}(\mathsf{H}_0)\mathbb{E}_0\left[\Big(1+\mathsf{v}(X)\Big)^2 + \mathsf{r}(X)\Big(1-\mathsf{v}(X)\Big)^2\right] \qquad \mathsf{r}(X) = \frac{\mathsf{f}_1(X)\mathbb{P}(\mathsf{H}_1)}{\mathsf{f}_0(X)\mathbb{P}(\mathsf{H}_0)}$$

minimize for each fixed X

$$\mathsf{v}_\mathsf{o}(X) = \frac{\mathsf{r}(X)-1}{\mathsf{r}(X)+1} = \omega\big(\mathsf{r}(X)\big), \quad \text{where } \omega(\mathsf{r}) = \frac{\mathsf{r}-1}{\mathsf{r}+1} \text{ strictly increasing}$$

Test equivalent to Bayes: $\mathsf{v}_\mathsf{o}(X) = \omega\big(\mathsf{r}(X)\big) \underset{\mathsf{H}_0}{\overset{\mathsf{H}_1}{\gtreqless}} \omega(1) = 0 \quad \Rightarrow \quad \mathsf{u}(X, \theta_\mathsf{o}) \underset{\mathsf{H}_0}{\overset{\mathsf{H}_1}{\gtreqless}} 0$

Equivalence in the limit

Consistency (with respect to the Bayes test)

$$\min_{\theta} \mathsf{J}(\theta) = \min_{\theta} \left\{ \sum_{i=1}^{n_0} \Big(1 + \mathsf{u}(X_i^0, \theta)\Big)^2 + \sum_{j=1}^{n_1} \Big(1 - \mathsf{u}(X_j^1, \theta)\Big)^2 \right\}$$

$$\Rightarrow \ \theta_{\mathsf{o}} \ \Rightarrow \ \mathsf{u}(X, \theta_{\mathsf{o}})$$

$$\min_{\mathsf{v}} \mathsf{J}(\mathsf{v}) = \min_{\mathsf{v}} \left\{ \mathbb{P}(\mathsf{H}_0)\mathbb{E}_0 \left[\Big(1 + \mathsf{v}(X)\Big)^2 \right] + \mathbb{P}(\mathsf{H}_1)\mathbb{E}_1 \left[\Big(1 - \mathsf{v}(X)\Big)^2 \right] \right\}$$

$$\Rightarrow \ \mathsf{v}_{\mathsf{o}}(X) \ = \ \omega\big(\mathsf{r}(X)\big)$$

Expect: $\mathsf{u}(X, \theta_{\mathsf{o}}) \approx \omega\big(\mathsf{r}(X)\big)$

Optimum Test: $\omega\big(\mathsf{r}(X)\big) \underset{\mathsf{H}_0}{\overset{\mathsf{H}_1}{\gtreqless}} \omega(1)$, Close to Optimum: $\mathsf{u}(X, \theta_{\mathsf{o}}) \underset{\mathsf{H}_0}{\overset{\mathsf{H}_1}{\gtreqless}} \omega(1)$

Develop data driven methods for estimation of $\omega\big(\mathsf{r}(X)\big)$ for other $\omega(\mathsf{r})$

Consistent tests eventually prevail over inconsistent tests

Designing Consistent Test

$$\mathsf{r}(X) \underset{\mathsf{H}_0}{\overset{\mathsf{H}_1}{\gtreqless}} 1 \;\equiv\; \omega_1\big(\mathsf{r}(X)\big) \underset{\mathsf{H}_0}{\overset{\mathsf{H}_1}{\gtreqless}} \omega_1(1) \;\equiv\; \omega_2\big(\mathsf{r}(X)\big) \underset{\mathsf{H}_0}{\overset{\mathsf{H}_1}{\gtreqless}} \omega_2(1)$$

$$\mathsf{u}(X,\theta_{\mathsf{o}}) \underset{\mathsf{H}_0}{\overset{\mathsf{H}_1}{\gtreqless}} 1 \;\not\equiv\; \mathsf{u}_1(X,\theta_1) \underset{\mathsf{H}_0}{\overset{\mathsf{H}_1}{\gtreqless}} \omega_1(1) \;\not\equiv\; \mathsf{u}_2(X,\theta_2) \underset{\mathsf{H}_0}{\overset{\mathsf{H}_1}{\gtreqless}} \omega_2(1)$$

For function $\omega(\mathsf{r})$ can we define cost

$$\mathsf{J}(\mathsf{v}) = \mathbb{P}(\mathsf{H}_0)\mathbb{E}_0\big[\phi\big(\mathsf{v}(X)\big)\big] + \mathbb{P}(\mathsf{H}_1)\mathbb{E}_1\big[\psi\big(\mathsf{v}(X)\big)\big]$$

$$\mathsf{J}(\mathsf{v}) = \mathbb{P}(\mathsf{H}_0)\mathbb{E}_0\Big[\big(1+\mathsf{v}(X)\big)^2\Big] + \mathbb{P}(\mathsf{H}_1)\mathbb{E}_1\Big[\big(1-\mathsf{v}(X)\big)^2\Big]$$

so that $\min_{\mathsf{v}} \mathsf{J}(\mathsf{v}) \;\Rightarrow\; \mathsf{v}_{\mathsf{o}}(X) = \omega\big(\mathsf{r}(X)\big)$?

THEOREM: Select strictly increasing function $\omega(\mathsf{r})$ and strictly negative function $\rho(z)$. Define

$$\psi'(z) = \rho(z), \quad \phi'(z) = -\omega^{-1}(z)\rho(z)$$

then the solution of the optimization problem

$$\min_{\mathsf{v}} \mathsf{J}(\mathsf{v}) = \min_{\mathsf{v}} \left\{ \mathbb{P}(\mathsf{H}_0)\mathbb{E}_0\left[\phi\big(\mathsf{v}(X)\big)\right] + \mathbb{P}(\mathsf{H}_1)\mathbb{E}_1\left[\psi\big(\mathsf{v}(X)\big)\right] \right\}$$

satisfies $\mathsf{v}_o(X) = \arg\min_{\mathsf{v}} \mathsf{J}(\mathsf{v}) = \omega\big(\mathsf{r}(X)\big)$

Same optimal solution $\omega(\mathsf{r}(X))$ for all functions $\rho(z) < 0$

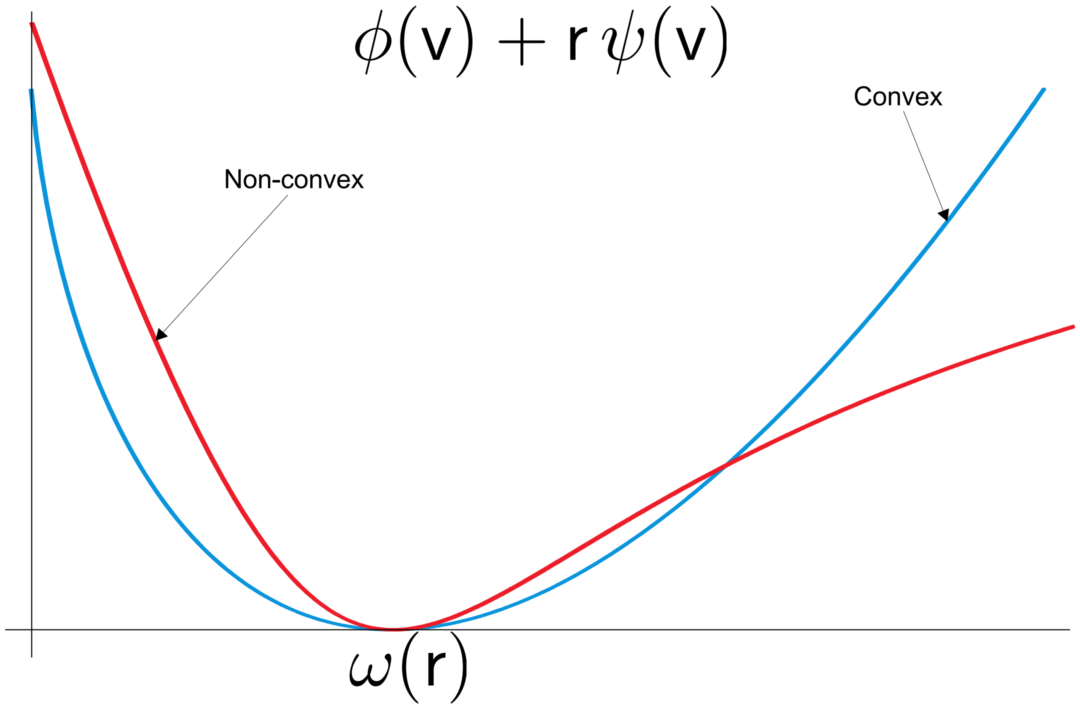
Proof (highlights): Apply change of measures in the second part

$$\mathsf{J}(\mathsf{v}) = \mathbb{P}(\mathsf{H}_0)\mathbb{E}_0\left[\phi\big(\mathsf{v}(X)\big)\right] + \mathbb{P}(\mathsf{H}_1)\mathbb{E}_1\left[\psi\big(\mathsf{v}(X)\big)\right]$$
$$= \mathbb{P}(\mathsf{H}_0)\mathbb{E}_0\left[\phi\big(\mathsf{v}(X)\big) + \mathsf{r}(X)\psi\big(\mathsf{v}(X)\big)\right]$$

therefore

$$\min_{\mathsf{v}} \mathsf{J}(\mathsf{v})$$
$$= \min_{\mathsf{v}} \mathbb{P}(\mathsf{H}_0)\mathbb{E}_0\left[\phi\big(\mathsf{v}(X)\big) + \mathsf{r}(X)\psi\big(\mathsf{v}(X)\big)\right]$$

minimize for each fixed X



Examples of functions

A: $\omega(\mathsf{r}) = \mathsf{r} \in \mathbb{R}_+$ (likelihood ratio)

$$\rho(z) = -1, z \geq 0 \;\; \Rightarrow \;\; \phi(z) = \frac{z^2}{2}, \;\; \psi(z) = -z$$

Mean Square

B: $\omega(\mathsf{r}) = \log(\mathsf{r}) \in \mathbb{R}$ (log-likelihood ratio)

$$\rho(z) = -e^{-0.5z} \;\; \Rightarrow \;\; \phi(z) = 2e^{0.5z}, \;\; \psi(z) = 2e^{-0.5z}$$

Exponential

C: $\omega(\mathsf{r}) = \dfrac{\mathsf{r}}{\mathsf{r}+1} \in [0,1]$ (posterior probability)

$$\rho(z) = -\frac{1}{z}, z \in [0,1] \;\; \Rightarrow \;\; \phi(z) = -\log(1-z), \;\; \psi(z) = -\log(z)$$

Cross Entropy

Data Driven Implementation

$$\mathsf{J}(\mathsf{v}) = \mathbb{P}(\mathsf{H}_0)\mathbb{E}_0\left[\phi\big(\mathsf{v}(X)\big)\right] + \mathbb{P}(\mathsf{H}_1)\mathbb{E}_1\left[\psi\big(\mathsf{v}(X)\big)\right]$$

$$\mathsf{J}(\theta) = \frac{1}{n_0+n_1}\left\{\sum_{i=1}^{n_0}\phi\big(\mathsf{u}(X_i^0,\theta)\big) + \sum_{j=1}^{n_1}\psi\big(\mathsf{u}(X_j^1,\theta)\big)\right\}$$

$$\mathsf{u}(X,\theta_{\mathsf{o}}) \approx \omega\left(\frac{\mathsf{f}_1(X)\mathbb{P}(\mathsf{H}_1)}{\mathsf{f}_0(X)\mathbb{P}(\mathsf{H}_0)}\right)$$

$$\mathsf{J}(\theta) = \frac{1}{n_0}\sum_{i=1}^{n_0}\phi\big(\mathsf{u}(X_i^0,\theta)\big) + \frac{1}{n_1}\sum_{j=1}^{n_1}\psi\big(\mathsf{u}(X_j^1,\theta)\big)$$

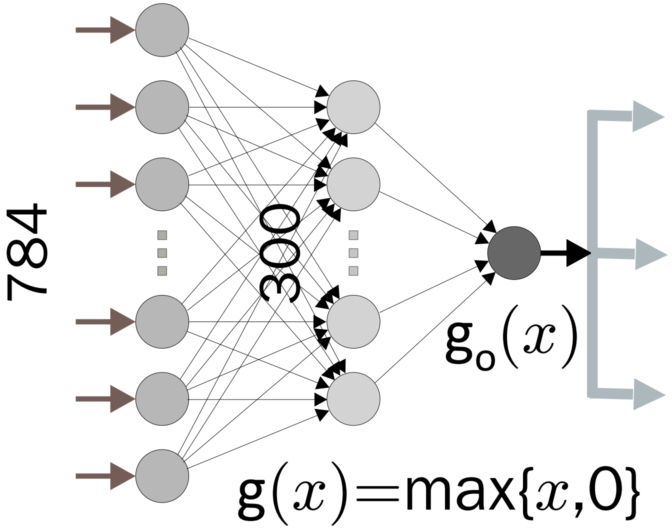
$$\mathsf{u}(X,\theta_{\mathsf{o}}) \approx \omega\left(\frac{\mathsf{f}_1(X)}{\mathsf{f}_0(X)}\right)$$

Example: Classification Problem

From dataset MNIST isolate handwritten numerals 4 and 9

Gray scale images 28 X 28 = 784 pixels. Design classifier using training data. Examine performance using testing data.

Neural network 784 X 300 X 1



Mean square $\mathrm{g_o}(x)=\max\{x,0\}$, $\mathrm{u}_1(X,\theta_1)$

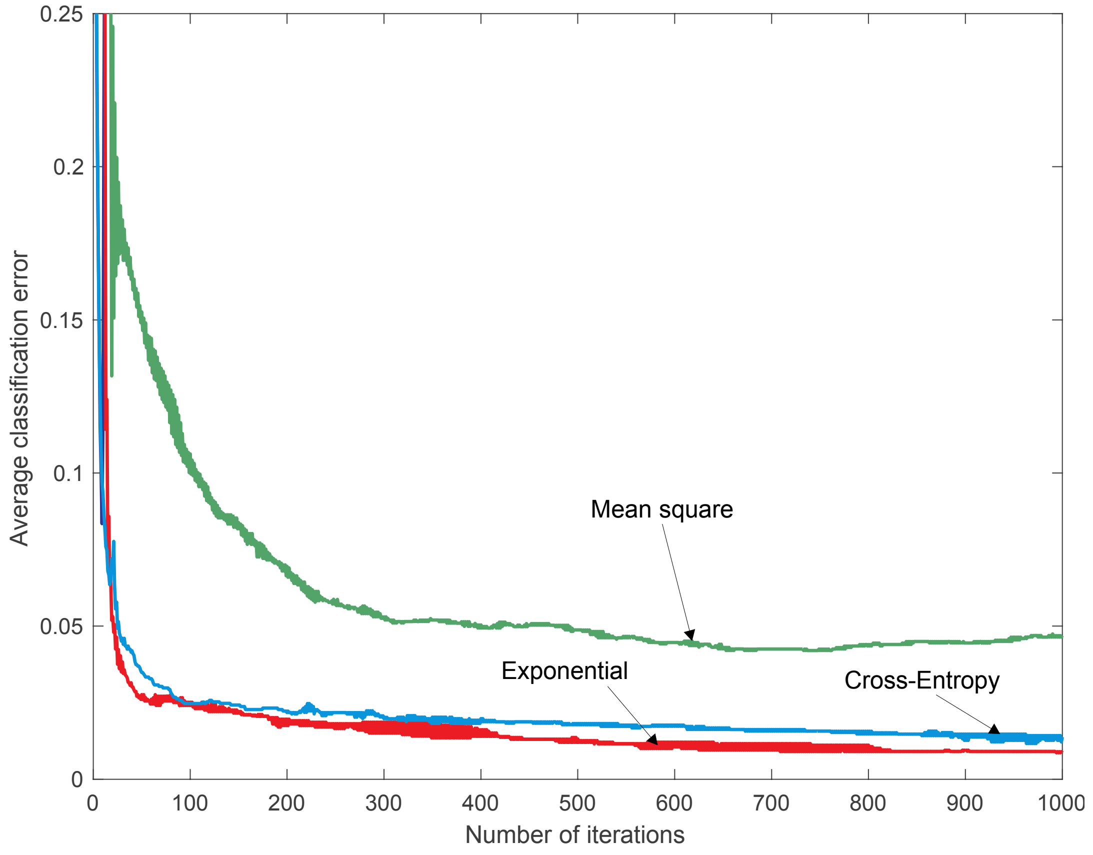
Exponential $\mathrm{g_o}(x)= x$, $\mathrm{u}_2(X,\theta_2)$

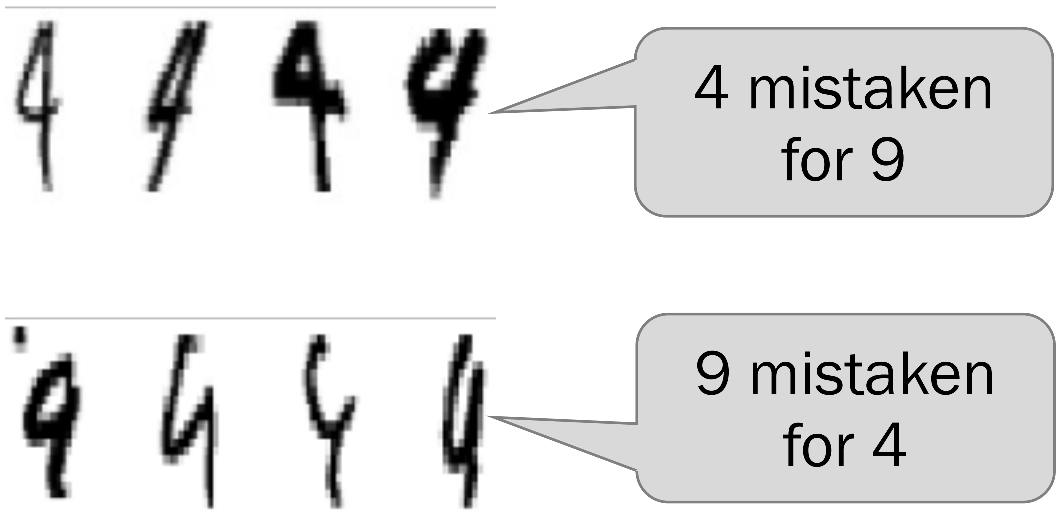
Cross entropy $\mathrm{g_o}(x)=$ sigmoid, $\mathrm{u}_3(X,\theta_3)$

# of parameters
235801

Training set: 5500 “4” and 5500 “9”. Testing set: 982 “4” and 1009 “9”

$$\mathsf{u}_1(X,\theta_1) \underset{\mathsf{H}_0}{\overset{\mathsf{H}_1}{\gtreqless}} 1, \quad \mathsf{u}_2(X,\theta_2) \underset{\mathsf{H}_0}{\overset{\mathsf{H}_1}{\gtreqless}} 0, \quad \mathsf{u}_3(X,\theta_3) \underset{\mathsf{H}_0}{\overset{\mathsf{H}_1}{\gtreqless}} \frac{1}{2}$$





Detection in Time Series

More practically interesting case: Testing of time series $\{ X_1, X_2, \ldots, X_n \}$

The whole set of measurements under H_0 or H_1

For testing we need likelihood ratio

$$\frac{f_1(X_n,\ldots,X_1)}{f_0(X_n,\ldots,X_1)} = \frac{f_1(X_n|X_{n-1},\ldots,X_1)}{f_0(X_n|X_{n-1},\ldots,X_1)} \frac{f_1(X_{n-1}|X_{n-2},\ldots,X_1)}{f_0(X_{n-1}|x_{X-2},\ldots,X_1)} \cdots \frac{f_1(X_1)}{f_0(X_1)}$$

When i.i.d. under each hypothesis $$\frac{f_1(X_n,\ldots,X_1)}{f_0(X_n,\ldots,X_1)} = \frac{f_1(X_n)}{f_0(X_n)} \cdots \frac{f_1(X_1)}{f_0(X_1)}$$

Test to be used $$\sum_{i=1}^{n} \log\left(\frac{f_1(X_i)}{f_0(X_i)}\right) \underset{H_0}{\overset{H_1}{\gtreqless}} \eta$$

Interested in estimating $\omega\big(\mathsf{r}(X)\big) = \omega\left(\frac{\mathsf{f}_1(X)}{\mathsf{f}_0(X)}\right)$

We are given training data: $\{X_1^0, \ldots, X_{n_0}^0\}$ following H_0

$\{X_1^1, \ldots, X_{n_1}^1\}$ following H_1

For each $\omega(\mathsf{r})$ of interest, minimize corresponding $\mathsf{J}(\theta)$

$$\mathsf{J}(\theta) = \frac{1}{n_0}\sum_{i=1}^{n_0}\phi\big(\mathsf{u}(X_i^0, \theta)\big) + \frac{1}{n_1}\sum_{j=1}^{n_1}\psi\big(\mathsf{u}(X_j^1, \theta)\big)$$

$$\mathsf{u}(X, \theta_{\mathsf{o}}) \approx \omega\left(\frac{\mathsf{f}_1(X)}{\mathsf{f}_0(X)}\right)$$

For $\omega(\mathsf{r}) = \mathsf{r}$, $\mathsf{u}_1(X, \theta_1) \approx \frac{\mathsf{f}_1(X)}{\mathsf{f}_0(X)}$, use $\log\big(\mathsf{u}_1(X, \theta_1)\big)$ (Mean Square)

For $\omega(\mathsf{r}) = \log(\mathsf{r})$, $\mathsf{u}_2(X, \theta_2) \approx \log\left(\frac{\mathsf{f}_1(X)}{\mathsf{f}_0(X)}\right)$, use $\mathsf{u}_2(X, \theta_2)$ (Exponential)

For $\omega(\mathsf{r}) = \frac{\mathsf{r}}{\mathsf{r}+1}$, $\mathsf{u}_3(X, \theta_3) \approx \frac{\frac{\mathsf{f}_1(X)}{\mathsf{f}_0(X)}}{\frac{\mathsf{f}_1(X)}{\mathsf{f}_0(X)} + 1}$, use $\log\left(\frac{\mathsf{u}_3(X, \theta_3)}{1 - \mathsf{u}_3(X, \theta_3)}\right)$

(Cross Entropy)

Example: Testing i.i.d. sequences

Assume $\{X_i\}$ are vectors of length 10

We would like to test 20 consecutive samples $\{X_1, \ldots, X_{20}\}$

H_0: $\mathsf{f}_0 \sim \mathcal{N}(0, I)$

H_1: $\mathsf{f}_1 \sim \mathcal{N}\big(\frac{1}{\sqrt{10}}[1 \cdots 1], 1.2I\big)$

Neural Network 10×20×1

Training data $n_0 = n_1 =$100

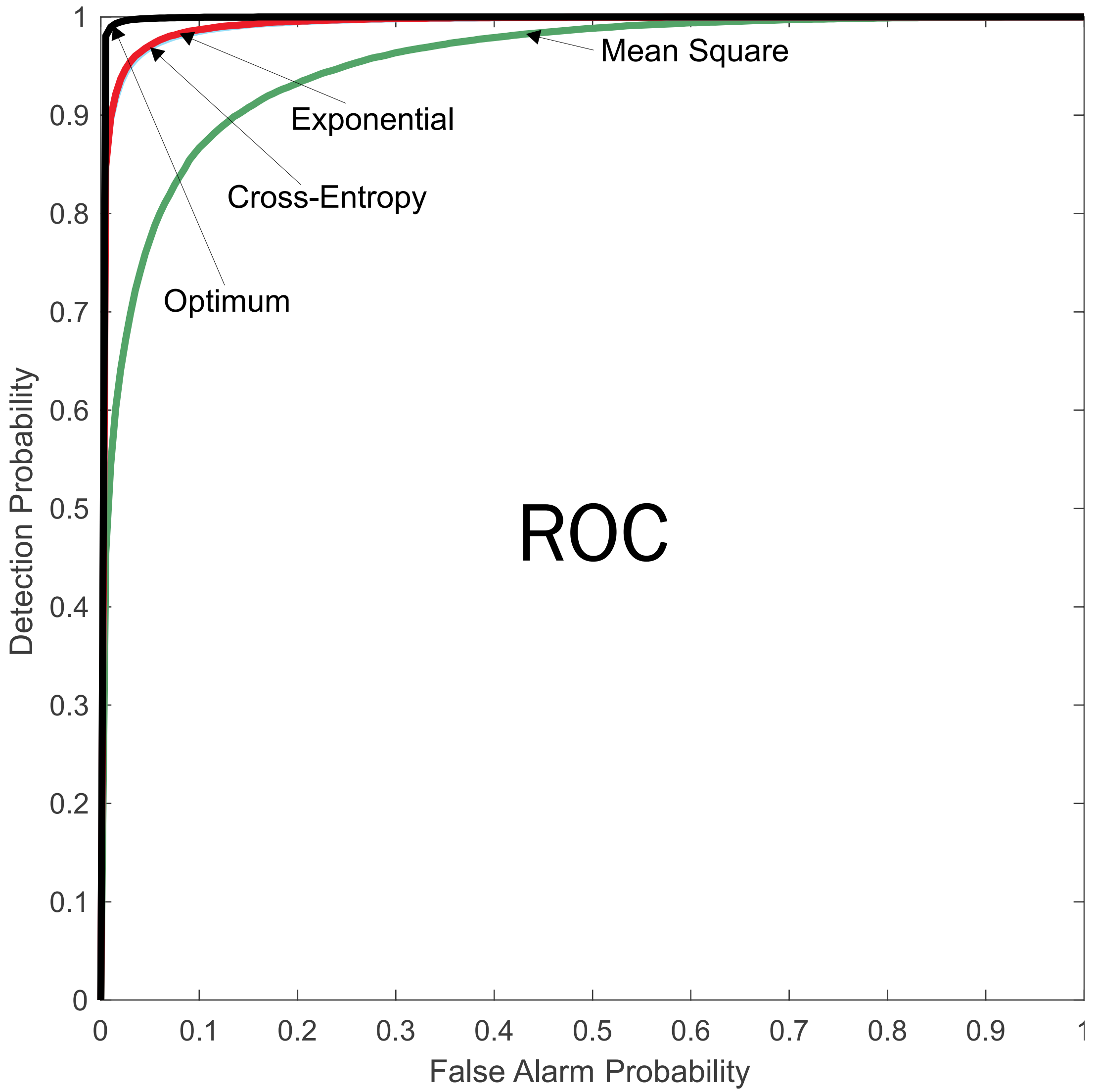
Produce $\mathsf{u}_1(X, \theta_1), \mathsf{u}_2(X, \theta_2), \mathsf{u}_3(X, \theta_3)$

We will employ $\log\big(\mathsf{u}_1(X, \theta_1)\big), \quad \mathsf{u}_2(X, \theta_2), \quad \log\left(\frac{\mathsf{u}_3(X, \theta_3)}{1 - \mathsf{u}_3(X, \theta_3)}\right)$

Testing for n =20 samples

$$\left.\begin{array}{r}\sum_{i=1}^{20} \log\big(\mathsf{u}_1(X_i,\theta_1)\big)\\ \sum_{i=1}^{20} \mathsf{u}_2(X_i,\theta_2)\\ \sum_{i=1}^{20} \log\left(\frac{\mathsf{u}_3(X_i,\theta_3)}{1-\mathsf{u}_3(X_i,\theta_3)}\right)\end{array}\right\} \overset{\mathsf{H}_1}{\underset{\mathsf{H}_0}{\gtrless}} \eta$$

We generate 100000×20 realizations from f_0 and from f_1



Testing Markovian processes

Consider Markovian processes with "memory" m

$$\frac{\mathsf{f}_1(X_n|X_{n-1},\ldots,X_1)}{\mathsf{f}_0(X_n|X_{n-1},\ldots,X_1)} = \frac{\mathsf{f}_1(X_n|X_{n-1},\ldots,X_{n-m})}{\mathsf{f}_0(X_n|X_{n-1},\ldots,X_{n-m})}$$

$$\frac{\mathsf{f}_1(X_n,\ldots,X_1)}{\mathsf{f}_0(X_n,\ldots,X_1)} = \frac{\mathsf{f}_1(X_n|X_{n-1},\ldots,X_{n-m})}{\mathsf{f}_0(X_n|X_{n-1},\ldots,X_{n-m})}\cdots\frac{\mathsf{f}_1(X_{m+1}|X_m,\ldots,X_1)}{\mathsf{f}_0(X_{m+1}|X_m,\ldots,X_1)} \times\frac{\mathsf{f}_1(X_m,\ldots,X_1)}{\mathsf{f}_0(X_m,\ldots,X_1)}$$

Can we estimate likelihood ratio of conditional densities?

a) Through data dynamics (classical)

b) Directly (proposed)

Classical Approach

Most common model, Autoregressive

$$X_t = A_1^i X_{t-1} + \cdots + A_m^i X_{t-m} + W_t, \quad i = 0, 1$$

$$X_t = G_i(X_{t-1}, \ldots, X_{t-m}, \theta^i) + W_t$$

Use training data $\{X_1^0, \ldots, X_{n_0}^0\}$ and $\{X_1^1, \ldots, X_{n_1}^1\}$ to solve

$$\min_{\theta^i} \sum_{t=1}^{n_i} \Big(X_t^i - G_i(X_{t-1}^i, \ldots, X_{t-m}^i, \theta^i) \Big)^2 \quad \Rightarrow \quad \theta_{\mathsf{o}}^i$$

$$W_t^i = X_t^i - G_i(X_{t-1}^i, \ldots, X_{t-m}^i, \theta_{\mathsf{o}}^i), \quad \Sigma_i = \frac{1}{n_i} \sum_{t=1}^{n_i} W_t^i (W_t^i)^\intercal$$

Assume $\{W_t^i\}$ i.i.d. Gaussian $\mathcal{N}(0, \Sigma_i)$

X_t given $\{X_{t-1}, \ldots, X_{t-m}\}$ under hypothesis H_i

Gaussian with mean $G_i(X_{t-1}, \ldots, X_{t-m}, \theta_{\mathsf{o}}^i)$ and covariance Σ_i

To test $\{X_1, \ldots, X_n\}$

$$W_t^i = X_t - G_i(X_{t-1}, \ldots, X_{t-m}, \theta_{\mathsf{o}}^i)$$

$$\frac{\mathsf{f}_1(X_t|X_{t-1}, \ldots, X_{t-m})}{\mathsf{f}_0(X_t|X_{t-1}, \ldots, X_{t-m})} = \frac{e^{-\frac{1}{2}(W_t^1)^\intercal \Sigma_1^{-1} W_t^1}}{e^{-\frac{1}{2}(W_t^0)^\intercal \Sigma_0^{-1} W_t^0}} \sqrt{\frac{|\Sigma_0|}{|\Sigma_1|}}$$

Not purely data driven

Gaussian assumption arbitrary, not necessarily suitable for all data!

Proposed Approach

$$\log\left(\frac{\mathsf{f}_1(X_t|X_{t-1},\ldots,X_{t-m})}{\mathsf{f}_0(X_t|X_{t-1},\ldots,X_{t-m})}\right) = \log\left(\frac{\frac{\mathsf{f}_1(X_t,X_{t-1},\ldots,X_{t-m})}{\mathsf{f}_1(X_{t-1},\ldots,X_{t-m})}}{\frac{\mathsf{f}_0(X_t,X_{t-1},\ldots,X_{t-m})}{\mathsf{f}_0(X_{t-1},\ldots,X_{t-m})}}\right) =$$

$$\log\left(\frac{\mathsf{f}_1(X_t,X_{t-1},\ldots,X_{t-m})}{\mathsf{f}_0(X_t,X_{t-1},\ldots,X_{t-m})}\right) - \log\left(\frac{\mathsf{f}_1(X_{t-1},\ldots,X_{t-m})}{\mathsf{f}_0(X_{t-1},\ldots,X_{t-m})}\right)$$

$$\mathsf{u}_{m+1}(X_t,\ldots,X_{t-m},\theta_{m+1}) \qquad \mathsf{u}_m(X_t,\ldots,X_{t-m+1},\theta_m)$$

$$\log\left(\frac{\mathsf{f}_1(X_t|X_{t-1},\ldots,X_{t-m})}{\mathsf{f}_0(X_t|X_{t-1},\ldots,X_{t-m})}\right) \approx$$

$$\mathsf{u}_{m+1}(X_t,\ldots,X_{t-m},\theta_{m+1}) - \mathsf{u}_m(X_{t-1},\ldots,X_{t-m},\theta_m)$$

Example: Testing Markov sequences

Scalar observations $\{x_1, \ldots, x_n\}$

$w_t \sim \mathcal{N}(0,1)$, i.i.d.

$\mathsf{H}_0 : x_t = w_t$

$\mathsf{H}_1 : x_t = \text{sign}(x_{t-1})\sqrt{|x_{t-1}|} + w_t$

$\mathsf{u}_2(x_t, x_{t-1}, \theta_2) : \ 2 \times 20 \times 1$

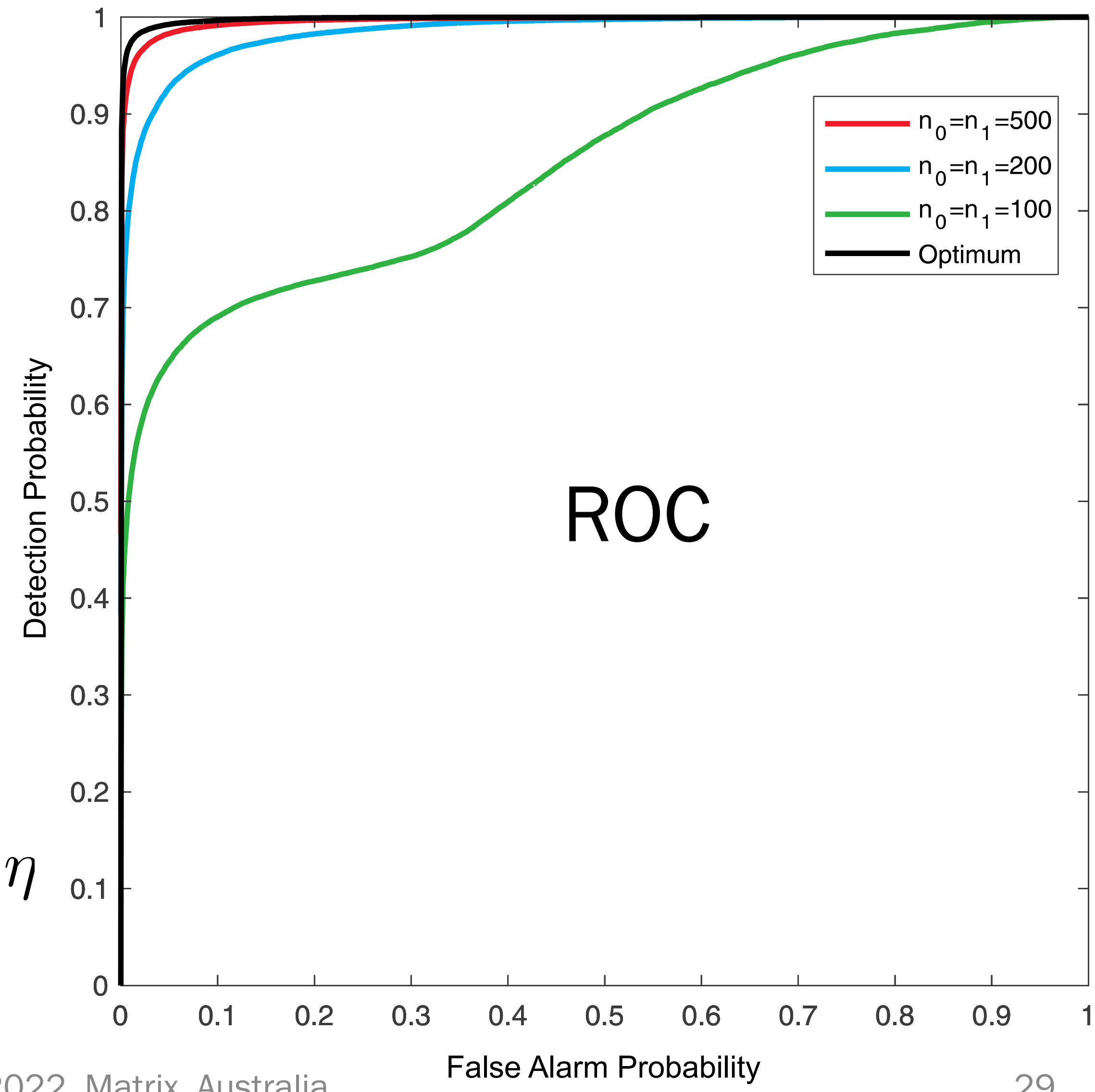
$\mathsf{u}_1(x_t, \theta_1) : \ 1 \times 10 \times 1$ (Exponential)

Training data $n_0 = n_1 = 100, 200, 500$

Testing $n = 20$, i.e. $\{x_1, \ldots, x_{20}\}$

$$\sum_{t=2}^{20} \mathsf{u}_2(x_t, x_{t-1}, \theta_2) - \sum_{t=2}^{19} \mathsf{u}_1(x_t, \theta_1) \underset{\mathsf{H}_0}{\overset{\mathsf{H}_1}{\gtreqless}} \eta$$

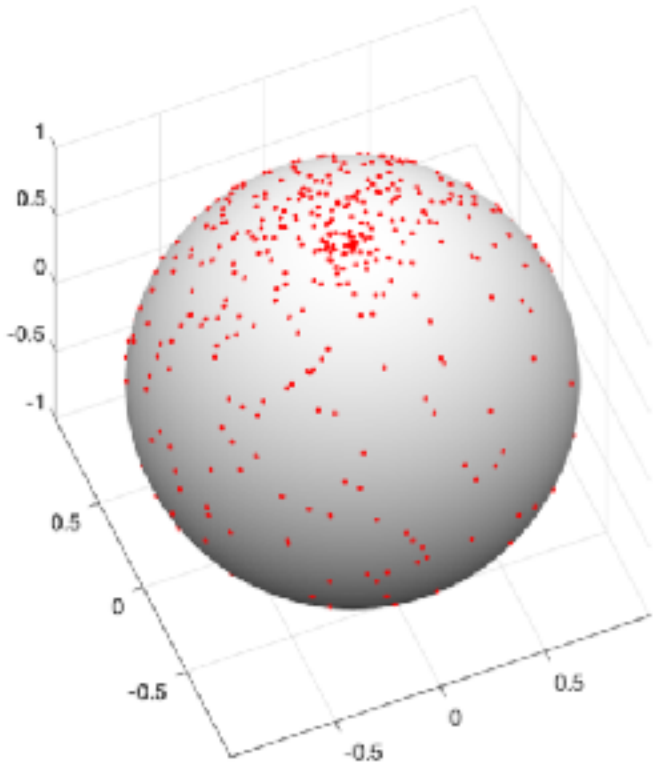
100000×20 samples from H_0 and H_1



Parameter Estimation: Outline

- Probability density vs Generative model
    - Inverse problems

Probability Density vs Generative Model



Points in N-D space can be random and lie on a lower dimensional surface (manifold)

Example red points on sphere (2-D in 3-D space)

Points are random with coordinates $[y_1,\ y_2,\ y_3]$ related through a deterministic equation

To lie on a sphere of radius r : $y_1^2 + y_2^2 + y_3^2 = r^2$

$$\mathsf{f}(y_1, y_2, y_3) = \delta(y_1^2 + y_2^2 + y_3^2 - r^2)\,\mathsf{h}(y_1, y_2)$$

Generative model: Data are representable as $Y=\mathsf{G}(Z)$, $Z\sim\mathsf{h}(Z)$. Many datasets satisfy

$$\mathsf{dim}(Y) \gg \mathsf{dim}(Z)$$

To design $\mathsf{G}(Z)$ we assume existence of training set $\{Y_1,\dots,Y_n\}$

Approximate Generator with neural network $\mathsf{G}(Z,\theta)$
Define second neural network the Discriminator $\mathsf{D}(X,\vartheta)$

$$\min_{\theta}\max_{\vartheta}\left\{\frac{1}{n}\sum_{t=1}^{n}\phi(\mathsf{D}(Y_t,\vartheta))+\frac{1}{m}\sum_{t=1}^{m}\psi\Big(\mathsf{D}\big(\mathsf{G}(Z_t,\theta),\vartheta\big)\Big)\right\}$$

$$\Rightarrow \theta_{\mathsf{o}} \Rightarrow \mathsf{G}(Z,\theta_{\mathsf{o}})$$

Generative
Adversarial
Network

Generator $\mathsf{G}(Z,\theta_{\mathsf{o}})$ when applied to realizations of Z yields samples following closely the density of $\{Y_1,\dots,Y_n\}$

Example

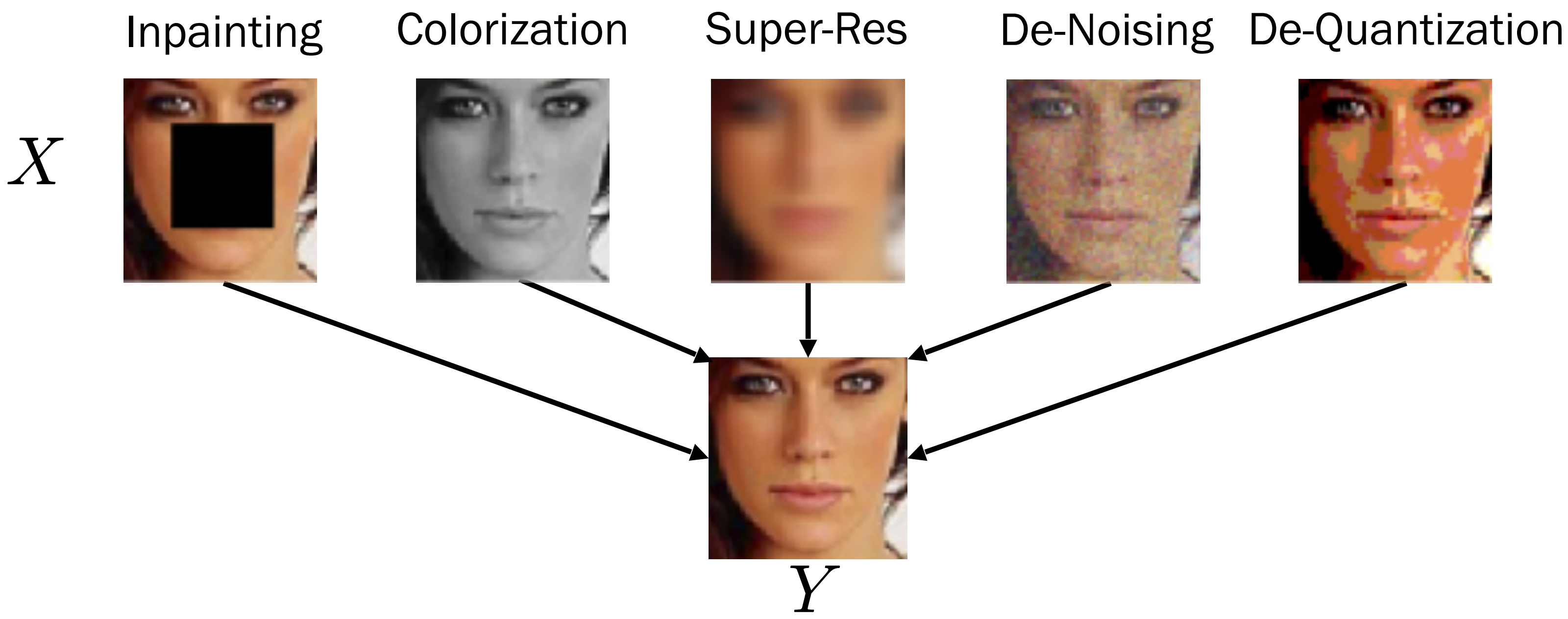
HD-CelebA (30 000 high definition images 1024 X 1024 of celebrities)




Extremely hard to control convergence of the adversarial problem

NVIDIA used progressive growing of GANs (4X4), (8X8),....,(1024X1024)

Design of $Y=\mathrm{G}(Z,\theta)$ where Z is Gaussian vector of length 500



$X = \mathsf{T}(Y) + W$

Instead of estimating Y from X, since $Y = \mathsf{G}(Z)$, we first estimate Z and then recover Y from $Y = \mathsf{G}(Z)$

Instead of estimating 3×10^6 variables from X, we only estimate 500 (vector Z)

Y: Follows generative model $\mathsf{G}(Z)$

Z: Input is Gaussian with mean 0 and covariance identity

X: Measurement is vector of length N

We can estimate input Z by solving the optimization problem

$$\min_Z \left\{ \log\left(\|X - \mathsf{T}\big(\mathsf{G}(Z)\big)\|^2 \right) + \frac{1}{N}\|Z\|^2 \right\}$$

$$\Rightarrow \hat{Z} \Rightarrow \hat{Y} = \mathsf{G}(\hat{Z})$$

Optimization problem is an outcome of rigorous analysis based on Statistical estimation theory where probability densities are replaced by generative models

Blurring with 3 X 3 mask

Colorization (green channel)

De-Quantization

3 levels per RGB channel, 27 colors