The Kardashian Kernel

David F. Fouhey

Sublime and Distinguished Grand Poobah,

CMU, Karlsruhe Inst. of Technology, Kharkiv Polytechnic Inst.

Daniel Maturana

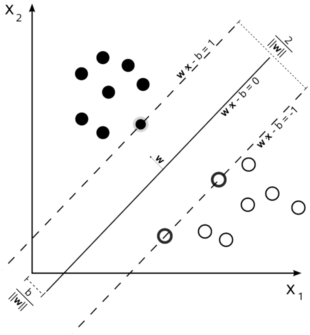
Distinguished Appointed Lecturer of Keeping it Real,

CMU, KAIST, Kyushu Inst. of Technology

Outline

Motivation

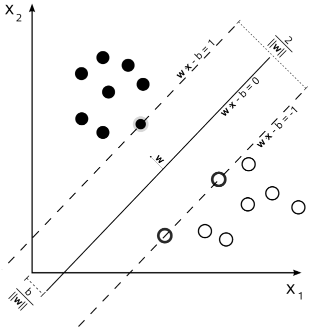
- Kernel machines are popular
  - Have fancy math
  - They work well



Motivation

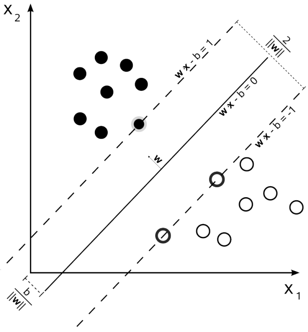
- Kernel machines are popular
  - Have fancy math
  - They work well
- The Kardashians are popular

  - **(TODO)**



Motivation

- Kernel machines are popular
  - Have fancy math
  - They work well
- The Kardashians are popular
  - (**TODO**)
- Why not combine them?



Related work

- Kronecker product
- Krylov subspace methods
- Kolmogorov axioms
- Kalman Filters
- Kent distribution
- Karhunen-Loève Transform
- Keypoint retrieval w/ K-d tree search
- Kriging (AKA Gaussian process regression)
- Kohonen maps (AKA Self-Organizing Maps)
- K-grams
- K-folds
- K-armed bandits
- ...

Related work

- Kronecker product
- Krylov subspace methods
- Kolmogorov axioms
- Kalman Filters
- Kent distribution
- Karhunen-Loève Transform
- Keypoint retrieval w/ K-d tree search
- Kriging (AKA Gaussian process regression)
- Kohonen maps (AKA Self-Organizing Maps)
- K-grams
- K-folds
- K-armed bandits
- ...
- Our approach: provably k-optimal, as our paper has significantly more k's and substantially more pictures of the Kardashians

Outline

Definitions

- Let $\mathcal{X}$ be an instance space.

Definitions

- Let $\mathcal{X}$ be an instance space.
- The Kardashian Kernel is an inner product operator $K_K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$.

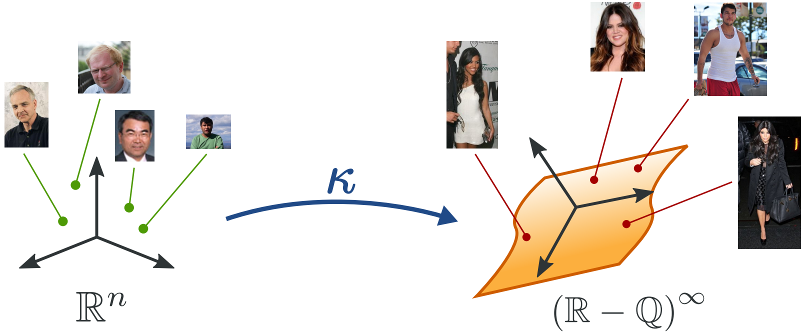
Definitions

- Let $\mathcal{X}$ be an instance space.
- The Kardashian Kernel is an inner product operator $K_K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$.
- Kernel trick (Mercer): $K_K(x, x') = \kappa(x)^T \kappa(x)$, with $\kappa : \mathcal{X} \to \mathfrak{K}$.

Definitions

- Let $\mathcal{X}$ be an instance space.
- The Kardashian Kernel is an inner product operator $K_K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$.
- Kernel trick (Mercer): $K_K(x, x') = \kappa(x)^T \kappa(x)$, with $\kappa : \mathcal{X} \to \mathfrak{K}$.
- can leverage the Kardashian Feature space without suffering the Kurse of Dimensionality.

The Kardashian Kernel Trick



$\kappa : \mathbb{R}^n \rightarrow Span\{$  ,  $\}$

$$K_K(x, x') = \kappa(x)^T \kappa(x')$$

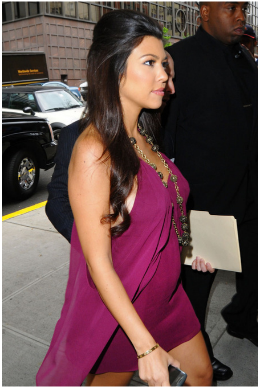
On Some Issues Raised by the Kardashian Kernel

On Reproducing Kardashian Kernels

- Does K_K define a Reproducing Kernel Hilbert Space (RKHS)? i.e. are the Kardashians Reproducing Kernels?

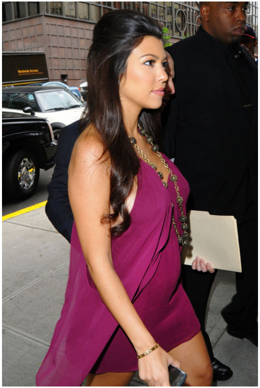
On Reproducing Kardashian Kernels

- Does K_K define a Reproducing Kernel Hilbert Space (RKHS)? i.e. are the Kardashians Reproducing Kernels?
- Only proven for case of Kourtney



On Reproducing Kardashian Kernels

- Does K_K define a Reproducing Kernel Hilbert Space (RKHS)? i.e. are the Kardashians Reproducing Kernels?
- Only proven for case of Kourtney
- But prominent bloggers argue that it is also true for Kim



On Divergence Functionals

Crucial question: does the space induced by κ have structure that is advantageous to minimizing the f-divergences?

Theorem

$$\min_{w} = \frac{1}{n}\sum_{i=1}^{n}\langle w, \kappa(x_i)\rangle - \frac{1}{n}\sum_{j=1}^{n}\log\langle w, \kappa(y_j)\rangle + \frac{\lambda_n}{2}||w||_{\mathfrak{K}}^2$$

Proof.

Obvious by the use of the Jensen-Jenner Inequality. $\square$

Outline

Kardashian SVM problem setting

Regular Support Vector Machines (SVMs) are boring. We propose to solve the following optimization problem, which is subject to the Kardashian-Karush-Kuhn-Tucker (**KKKT**) Conditions:

$$\min_{\mathbf{w},\xi,\mathbf{b}} \frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{n}\xi_i$$

Kardashian SVM problem setting

Regular Support Vector Machines (SVMs) are boring. We propose to solve the following optimization problem, which is subject to the Kardashian-Karush-Kuhn-Tucker (**KKKT**) Conditions:

$$\min_{\mathbf{w},\xi,\mathbf{b}} \frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{n}\xi_i$$

such that

$$\begin{aligned} y_i(\mathbf{w}^T\kappa(\mathbf{x}_i) - \mathbf{b}) \geq 1 - \xi_i & \quad 1 \leq i \leq n \\ \xi_i \geq 0 & \quad 1 \leq i \leq n \\ \zeta_j = 0 & \quad 1 \leq j \leq m. \end{aligned}$$

Learning algorithm

- Standard approach: Kuadratic Programming (KP)

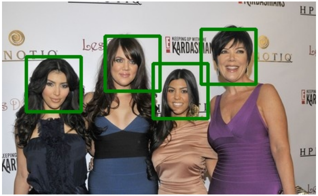
Learning algorithm

- Standard approach: Kuadratic Programming (KP)
- But see Kurvature of optimization manifold

Learning algorithm

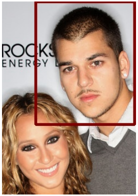
- Standard approach: Kuadratic Programming (KP)
- But see Kurvature of optimization manifold
- Take advantage of geometry: Konvex-Koncave Procedure (KKP)

Experiment: Kardashian or Cardassian?



(a) Kardashian - (l. to r.) Kim, Khloé, Kourtney, Kris

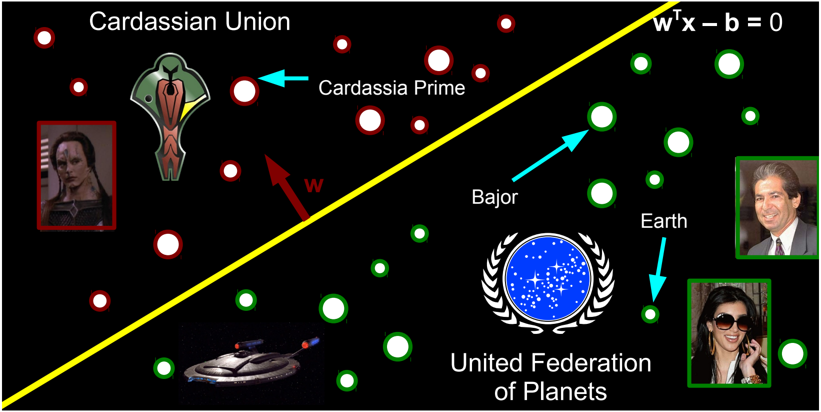
(b) Cardassian - (l. to r.) Gul Dukat, Elim Garak



(c) A failure case (or is it?): Kardashian - Rob

Our "Kardashian or Cardassian" dataset.

Schematic for Kardashian or Cardassian SVM



In the feature space $\mathfrak{K}$ induced by κ, the decision boundary between Cardassian and Kardashian lies approximately 5 light years from Cardassia Prime.

Graph Kardashiancian

Graph Kardashiancian

- The Graph Laplacian ℓ

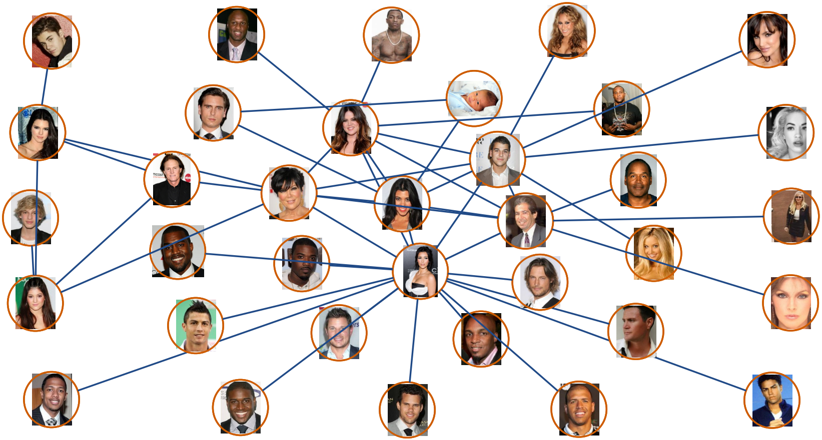
$$\ell_{i,j} := \begin{cases} \deg(v_i) & \text{if } i = j \\ -1 & \text{if } i \neq j \text{ and } v_i \text{ is adjacent to } v_j \\ 0 & \text{otherwise.} \end{cases}$$

Graph Kardashiancian

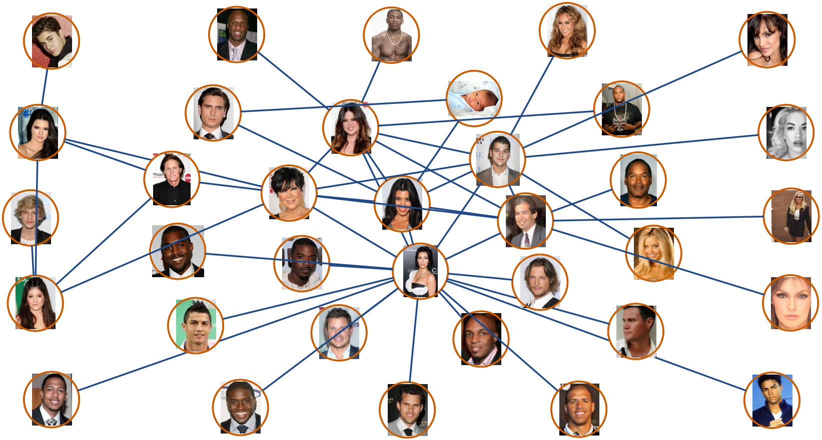
- The Graph Kardashiancian $\mathcal{K}$

$$\mathcal{K}_{i,j} := \begin{cases} \deg(v_i) & \text{if } i = j \\ -\kappa & \text{if } i \neq j \text{ and } v_i \text{ is Kardashian-adjacent to } v_j \\ 0 & \text{otherwise.} \end{cases}$$

Graph Kardashiancian



Graph Kardashiancian



- Application: KardashianRank

Kardashian Kopula

Kardashian Kopula

- Powerful generalization of the Gaussian Copula

$$c_{\Sigma}(u) = \frac{1}{\sqrt{\det \Sigma}} \exp\left(-\frac{1}{2} \Phi^{-1}(u)^{T} \left(\Sigma^{-1} - \mathbf{I} \right) \Phi^{-1}(u) \right)$$

Kardashian Kopula

- Powerful generalization of the Gaussian Copula

$$c_{\Sigma}^{K}(u) = \frac{1}{\sqrt{\det \Sigma}} \exp\left(-\frac{1}{2}\mathbf{K}^{-1}(u)^{T}\left(\Sigma^{-1} - \mathbf{I}\right)\mathbf{K}^{-1}(u)\right)$$

Kardashian Kopula

- Powerful generalization of the Gaussian Copula

$$c_{\Sigma}^{K}(u) = \frac{1}{\sqrt{\det \Sigma}} \exp\left(-\frac{1}{2} \mathbf{K}^{-1}(u)^{T} \left(\Sigma^{-1} - \mathbf{I} \right) \mathbf{K}^{-1}(u) \right)$$

- Video illustrating the Kardashian Kopula (featuring rapper Ray J) may be found in the supplementary material

Outline

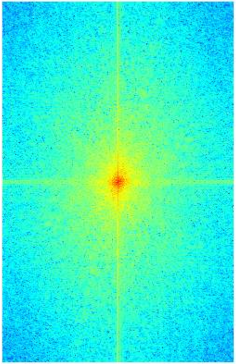
Celebrity-based Machine Learning

We have exhausted Kardashianity, but currently working on:

Celebrity-based Machine Learning

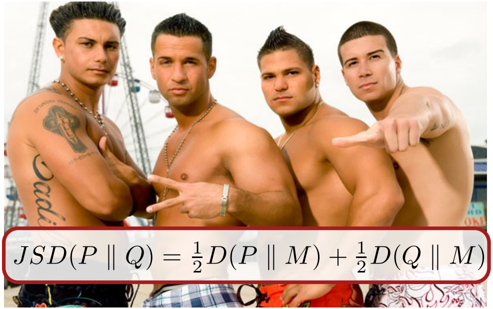
The Tila Tequila Transform (T_{T_T})

$$\xrightarrow{T_{T_T}(\mathcal{I})}$$



Celebrity-based Machine Learning

The Jensen-Shannon-Jersey-Shore (JS^2) divergence



$$JSD(P \parallel Q) = \frac{1}{2}D(P \parallel M) + \frac{1}{2}D(Q \parallel M)$$

A powerful generalization of The Kardashian-Kulback-Leibler (KKL) divergence

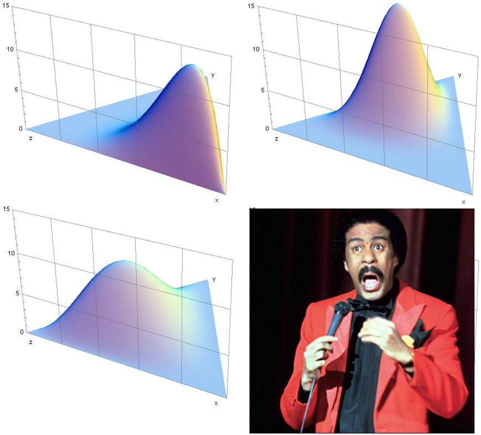
Celebrity-based Machine Learning

Jamie Lee Curtis Regularization

$$\min_{\beta(t)} \left(||y - \sum_{l=1}^{L} \mathbf{X}\beta(t)_l||_2^2 + \lambda||\beta(t) - \beta(t - 24h)||_2 \right)$$

Celebrity-based Machine Learning

The Richard Pryor Prior



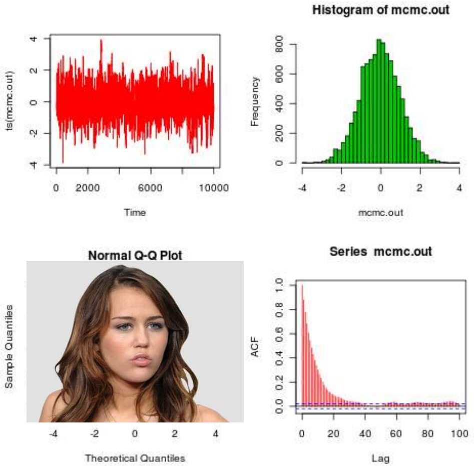
Celebrity-based Machine Learning

The Carrie Fisher Information Matrix

$\mathcal{I}(\theta) = \mathrm{E}\Big[\Big(\frac{\partial}{\partial\theta}\log$  $\Big)^2\Big|\,\theta\Big]$

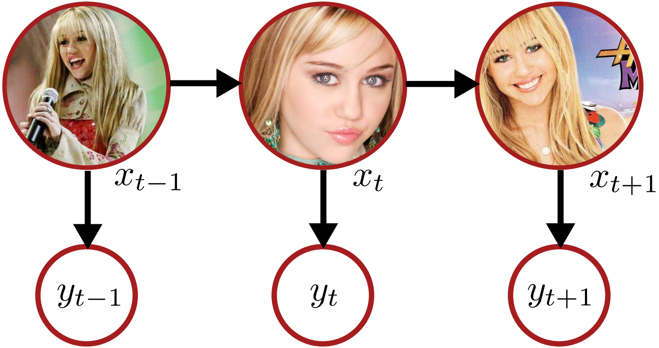
Celebrity-based Machine Learning

Miley Cyrus Markov Chain Monte Carlo (*MCMCMC*) methods for inference



Celebrity-based Machine Learning

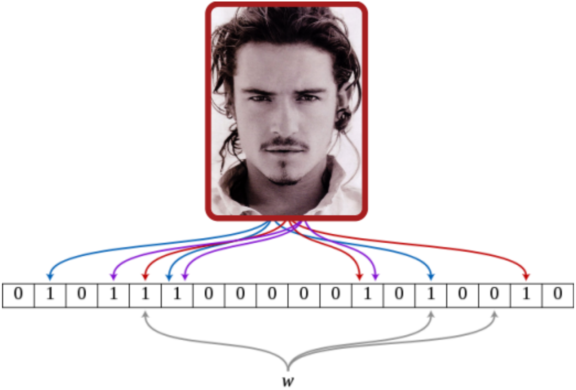
Hannah Montana Hidden Markov Models (*HMHMHMM*).



Train with MCMCMC for best of both worlds!

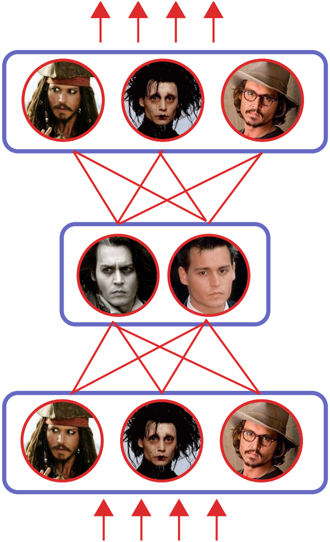
Celebrity-based Machine Learning

The Orlando Bloom Filter



Celebrity-based Machine Learning

Johnny Depp Belief Nets (*JDBN*s)



Thank you