Gaussian Mixture Copula Model

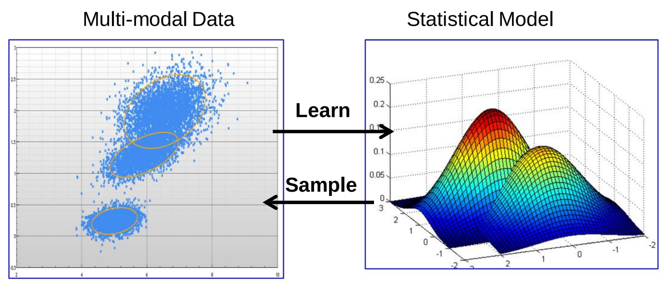
Ashutosh Tewari, Madhusudana Shashanka, Michael J. Giering
Emails: tewaria, shasham, gierinmj @utrc.utc.com

Copulas in Machine Learning (NIPS 2011)

United Technologies Research Center
United Technologies Research Center, East Hartford, CT

Continuous Mixture Models

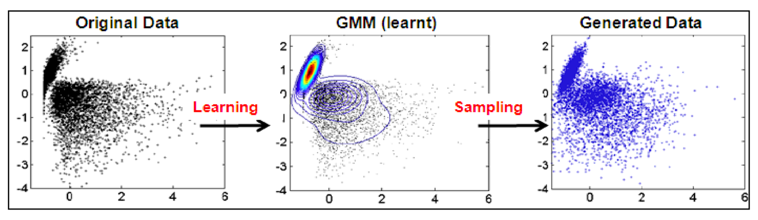
Models to explain data stemming from different populations:
1. Sensor data from engg systems.
2. Demographic data.
3. Finance.
4. Image/Video analytics.



Gaussian Mixture Models are widely used for this task !!!

- GMM can approximate any continuous PDF and scales well with the data dimension.
- GMM imposes a rigid assumption about the Gaussianity of each mode. Not a realistic assumption in several domains !!!

Example of a poor mixture of Gaussians-based generative model



Motivation: Develop a ***Copula*** based mixture model, with comparable scalability, but higher flexibility.

Copula-based Factorization of Joint Densities

In a Nutshell:

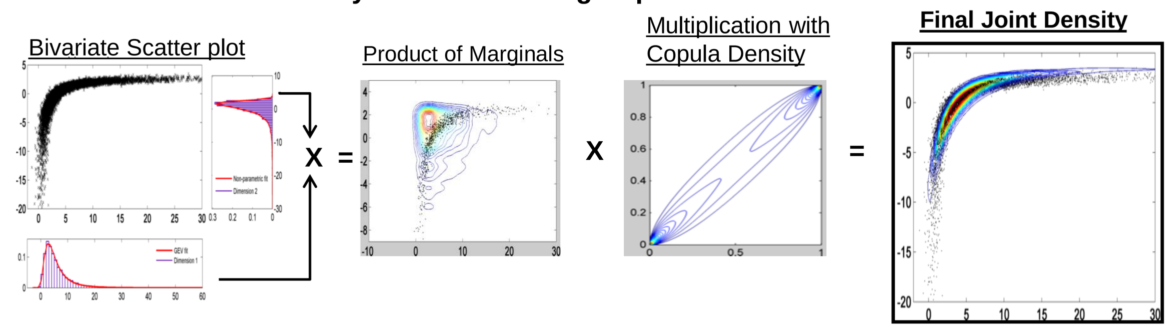
Copula functions allow factorization of joint densities as a product of *marginal* densities and a *Copula* density.

$u_j = F_j(x_j)$ is the j^{th} marginal CDF

$$f(x_1, x_2, \ldots \ldots x_d) = f_1(x_1) \times f_2(x_2) \cdots \times f_d(x_d) \times c(u_1, u_2, \ldots \ldots u_d)$$

Product of Marginal PDFs — Copula PDFs

Illustration of Joint Density Estimation using Copula functions:



Choosing the Best Copula Density.

Unknown Copula density

$$f(x_1, x_2, \ldots \ldots x_d) = f_1(x_1) \times f_2(x_2) \cdots \times f_d(x_d) \times c(u_1, u_2, \ldots \ldots u_d)$$

OPTION 1: Use copula functions from know parametric families:

Gumbel Copula $C(u_1,u_2;\theta) = \exp\left(-(u_1^{-\theta} + u_2^{-\theta})^{1/\theta}\right)$

Frank Copula $C(u_1,u_2;\theta) = \frac{-1}{\theta}\log\left(1+\frac{(e^{-\theta u_1}-1)(e^{-\theta u_2}-1)}{e^{-\theta}-1}\right)$

etc.

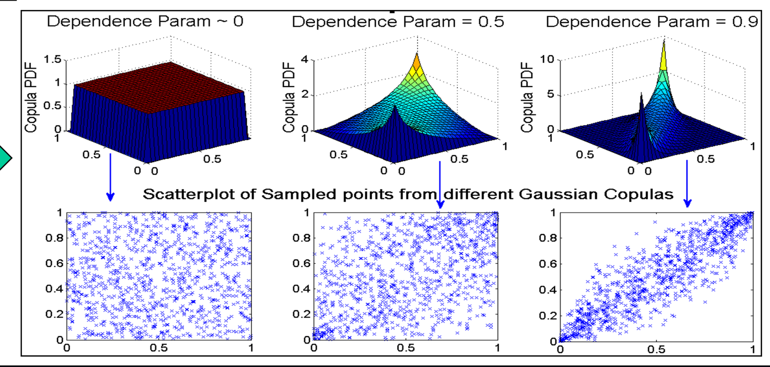
OPTION 2: Derive Copula Functions from known joint densities.

$$c(u_1, u_2 \ldots u_d) = \frac{f(x_1, x_2, \ldots \ldots x_d)}{f_1(x_1) \times f_2(x_2) \cdots \times f_d(x_d)}$$

Substitute for x_i: $F_i(x_i) = u_i \rightarrow x_i = F_i^{-1}(u_i)$

$$c(u_1, u_2 \ldots u_d; \Theta) = \frac{f(F_1^{-1}(u_1), F_2^{-1}(u_2) \cdots F_d^{-1}(u_d); \Theta)}{f_1(F_1^{-1}(u_1)) \times f_2(F_2^{-1}(u_2)) \cdots f_d(F_d^{-1}(u_d))}$$

Gaussian Copula (Derived from multivariate *Gaussian* density)



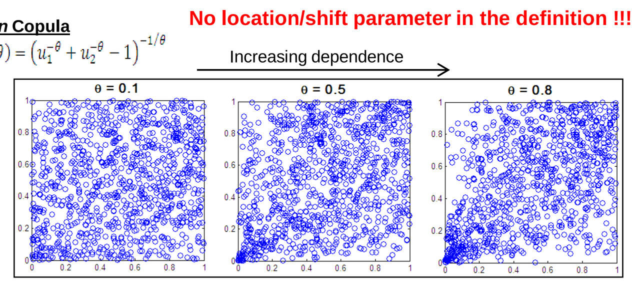
Motivation

Why the existing Copulas can't be used for Multimodal Dataset ?

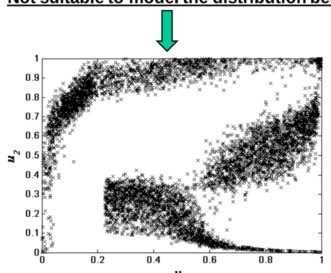
Known Copula families are not designed to capture dependencies in multimodal distributions (*absence of location parameter*)!!!

Clayton Copula $C(u_1,u_2;\theta) = (u_1^{-\theta} + u_2^{-\theta} - 1)^{-1/\theta}$

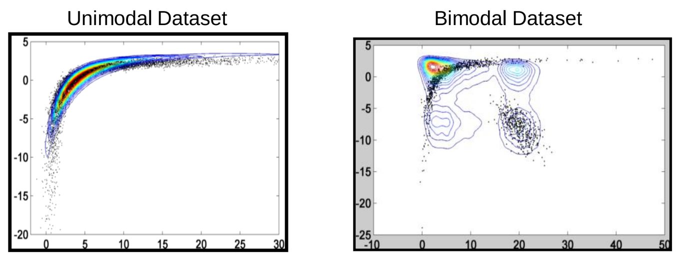
No location/shift parameter in the definition !!!



Not suitable to model the distribution below



***Gaussian* Copula Based Joint Density Fit on:**



Gaussian copula could not capture the dependence in a bimodal dataset. !!!

Our Contribution (*Gaussian Mixture Copula (GMC) Function**)

GMC function is derived from a density defined by a finite Mixture of Gaussians!!!

$$c(u_1, u_2 \cdots u_n) = \frac{f(F_1^{-1}(u_1), \cdots\cdots F_n^{-1}(u_n))}{f_1(F_1^{-1}(u_1)) \times f_2(F_2^{-1}(u_2)) \cdots f_n(F_n^{-1}(u_n))}$$

Substitute

Sum of Gaussians — Multivariate Gaussian

$$f = \psi = \sum_k \alpha^{(k)} \phi^{(k)}(x_1, x_2 \ldots x_n; \theta^{(k)})$$

$f_j = \psi_j$ — Marginal Density GMM

$F_j^{-1} = \Psi_j^{-1}$ — Inverse function of GMM marginal CDF

Gaussian Mixture Copula (GMC) Function

$$c_{gmc}(u_1, u_2 \ldots u_n; \Theta) = \frac{\psi(\Psi_1^{-1}(u_1), \Psi_2^{-1}(u_2) \cdots \Psi_n^{-1}(u_n); \Theta)}{\psi_1(\Psi_1^{-1}(u_1)) \times \psi_2(\Psi_2^{-1}(u_2)) \cdots \psi_n(\Psi_n^{-1}(u_n))}$$

The parameter set of GMC function consists of the *mixing proportions*, *mean vector* and *covariance matrix* of all the components

$$\Theta = \{\alpha^{(k)}, \theta^{(k)}\} \text{ for all } k$$

*A. Tewari, A. Raghunathan, M. Giering, "Parametric Characterization of Multimodal Dataset with Non-Gaussian Modes" OEDM workshop, ICDM 2011

Algorithms for Parameter Estimation

Estimate parameters such that *observed data likelihood is maximized*

Define the objective function:

Given N i.i.d. samples, $\{u^{(i)}\}_{i=1}^N$, define observed data log-likelihood as.

$$\ell(\Theta|\{u^{(i)}\}_{i=1}^N) = \sum_{i=1}^N \log\left(c_{gmc}(u_1^{(i)}, u_2^{(i)} \ldots u_n^{(i)}; \Theta)\right)$$

Obtain the Solution:

$$\hat{\Theta}(MLE) = \underset{\Theta}{argmax}\,[\ell(\Theta|\{u^{(i)}\}_{i=1}^N)]$$

s.t.
Mixing weights sum to unity
Covariance matrices are positive definite

***Expectation-Maximization* Algorithm:**

Pros:
1. Fast, because does not involve gradient / optimal step length computations.
2. Constraints are implicitly satisfied.

Cons:
1. Does not converge to the local maximum (for the above objective function).
2. 1st order optimization method **.

***Gradient-Based* Algorithm:**

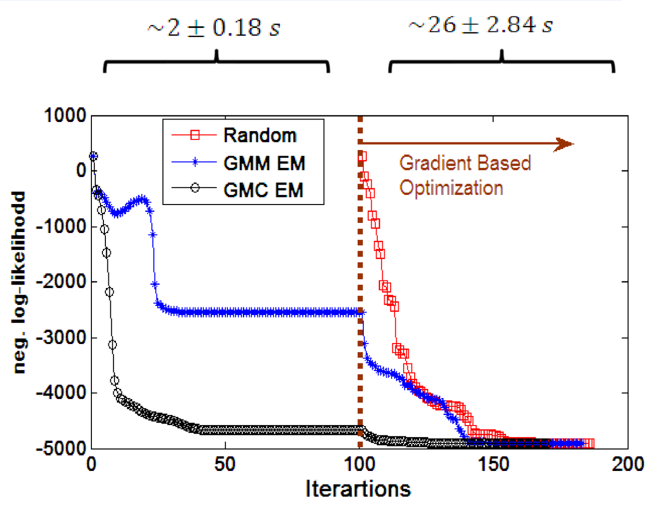
Pros:
1. Guarantees convergence to the local maximum.
2. 2nd order derivative information can be used to speed up the optimization.

Cons:
1. Unavailability of analytical order derivatives can add significant computational overhead.

**L. Xu and M. I. Jordan, "On convergence properties of the em algorithm for gaussian mixtures," Neural Computation, vol. 8, pp. 129–151, 1995.

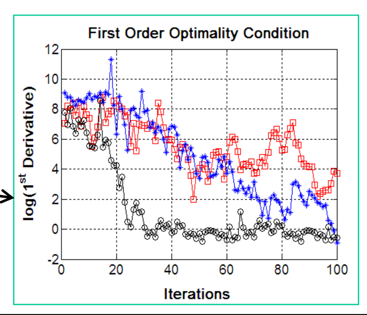
Convergence and Initialization

1. Gradient-based algorithm guarantees convergence to a locally optimal solution.
2. EM updates are significantly faster than the gradient-based updates.
3. Both GMM-EM and GMC-EM improve the quality of initial guess.
4. The GMM-EM does not guarantee monotonically decreasing objective function



EM Algorithm generates a good initial guess for the derivative based optimization !!

First order optimality condition indicates convergence to a local minimum.!!!



Results (Comparing GMCMs with GMMs)

- Gaussian Mixture Copula Model (GMCM) results in a better generative model than a Gaussian Mixture Model (GMM)

Experiments with Synthetic Datasets:

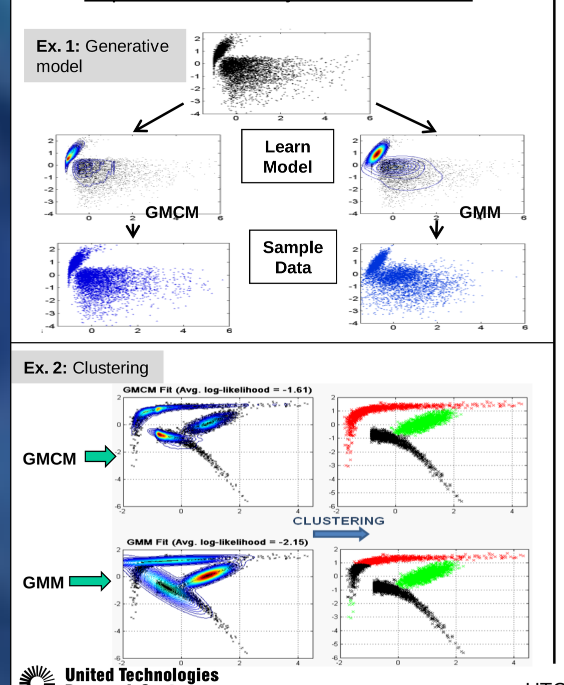
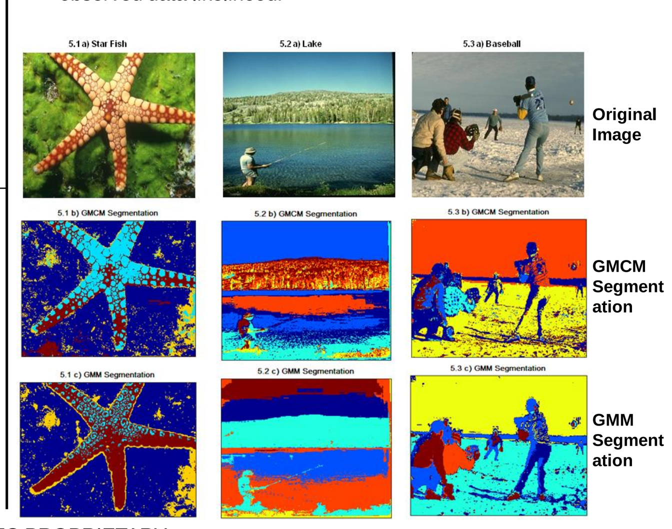


Image segmentation experiment based on pixel clustering:

1. Only RGB values were used for the segmentation.
2. The number of segments kept the same for both methods.
3. One experiment consist of *20 runs* of the learning algorithm (with different initializations) and choosing the model with highest observed data likelihood.



Conclusion

- Proposed a *Copula* function to model dependencies in multi-modal distributions.
- Resulting *Gaussian Mixture Copula* models can learn non-Gaussian components with non-linear dependencies.
- Proposed an expectation-maximization (EM) and a derivative-based algorithm for parameter estimation.
- Results on synthetic and real-life datasets corroborate the benefits of *GMCM* over *GMM*.

Future Work

Aimed at speeding up the parameter estimation by:

- Providing analytical approximations for Gradient and Hessian.
- Explore other optimization schemes, Cross-Entropy-based, Swarm etc.