

# Copulas and Machine Learning

## UAI 2012 Tutorial

*for anyone interested in real-valued modeling*

Gal Elidan  
Department of Statistics  
Hebrew University

# WIRED

## Recipe for Disaster: The Formula That Killed Wall Street

By Felix Salmon  02.23.09



In the mid-'80s, Wall Street turned to the quants—brainy financial engineers—to invent new ways to boost profits. Their methods for minting money worked brilliantly... until one of them devastated the global economy.  
*Photo: Jim Krantz/Gallery Stock*

# Was David Li the guy who 'blew up Wall Street?'

| Last Updated: Thursday, April 9, 2009 | 10:16 AM ET by Mike Hombrook CBC News



## Canadian scholar scapegoat for global meltdown

...

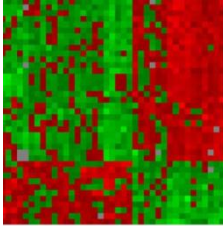
In a scholarly paper published in 2000, Li proposed the theorem be applied to credit risks, encompassing everything from bonds to mortgages. This particular copula was not new, but the financial application Li proposed for it was.

**Disastrously, it was just simple enough for untrained financial analysts to use, but too complex for them to properly understand.**

It appeared to allow them to definitively determine risk, effectively eliminating it. The result was an orgy of misspending that sent the U.S. banking system over a cliff.

# Motivation

The world around us is continuous



Gene expression



Chemical content



Stock market

Many of these domains

- have a complex structure
- are highly non-linear
- are high-dimensional

**Our goal:** to learn realistic joint distributions  
(and use them for prediction, explanation, discovery)

# Motivation

Density estimation is easy in one dimension:

- ✓ Many convenient families (Gaussian, Gamma,  $\text{Chi}^2$ ,...)
- ✓ Non-parametric approach is efficient and accurate

In contrast, for two (or more) variables:

- ✗ Few explicit non-Gaussian families
- ✗ Non-parametric estimation is demanding
- ✗ Sensitive to noise

⇒ most of multivariate ML is discrete!

# Why should we care about copulas?

## Graphical Models

a framework for modeling multivariate distributions

- ✓ Intuitive representation
- ✓ Tools for large-scale estimation and computation
- ✗ limited to few workable forms (for continuous domains)

## Copulas

a framework for modeling multivariate distributions

- ✓ Highly flexible representation
- ✓ Separate univariates from the true nature of dependence (we will see this shortly)
- ✗ typically limited to a small number of dimensions

Can our community take advantage of both?

# Outline

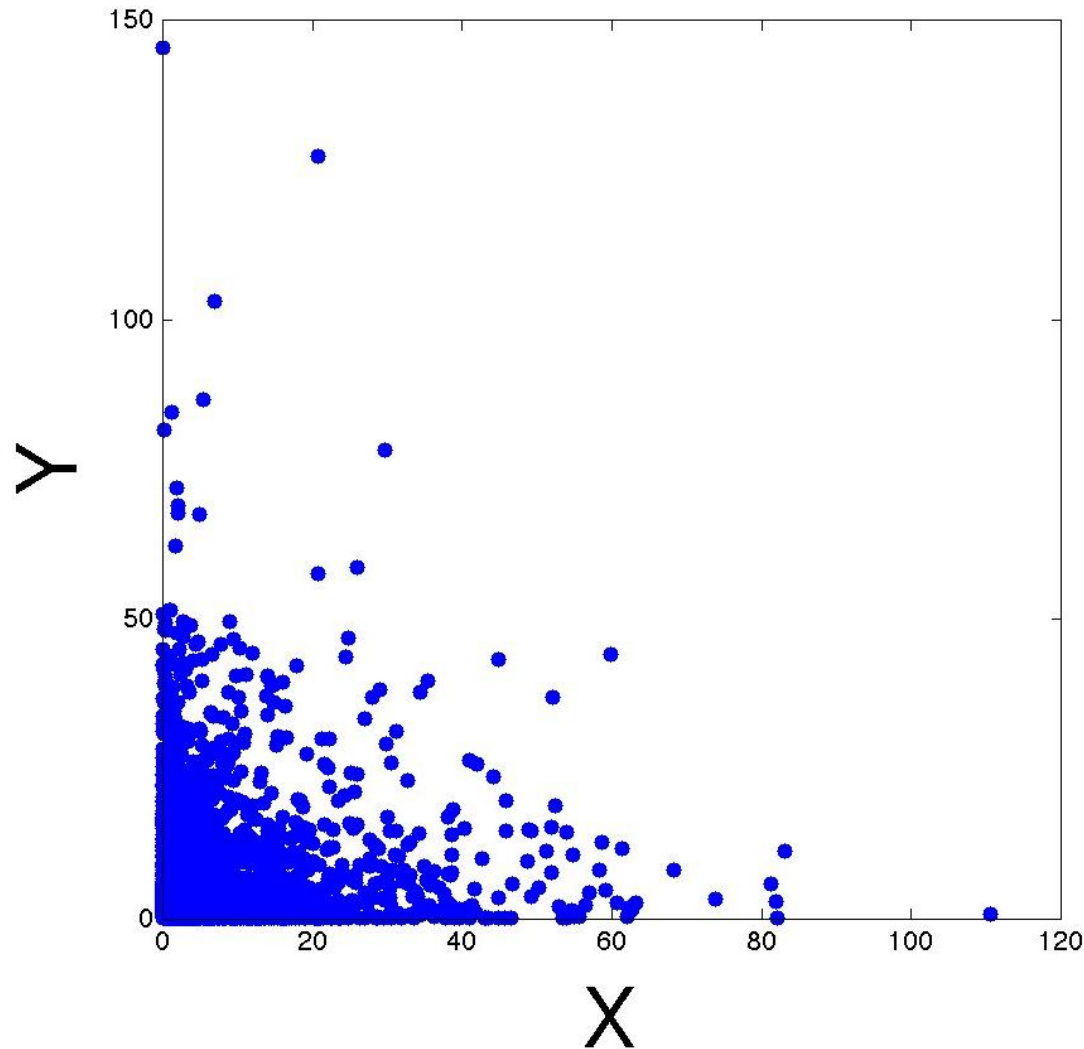
- Part I: Introduction to Copulas
- Part II: Graphical Copula Models
- Part III: Other Copula-based works in ML

# Part I: Introduction to Copulas

- A dependence prelude
- What are copulas
- Copula models
- Copulas and dependence
- Multivariate copulas



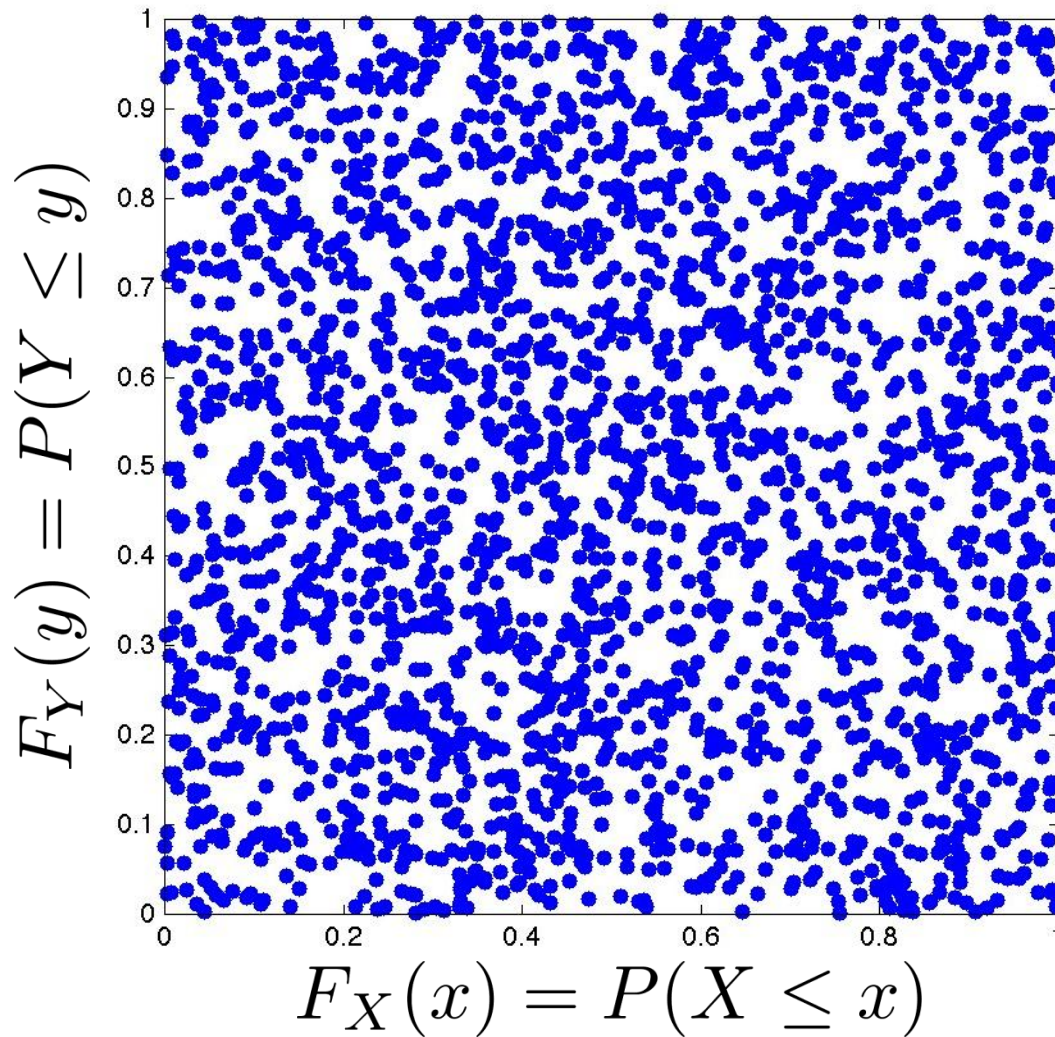
# What is the dependency structure?



When X is large Y is low and vice-versa

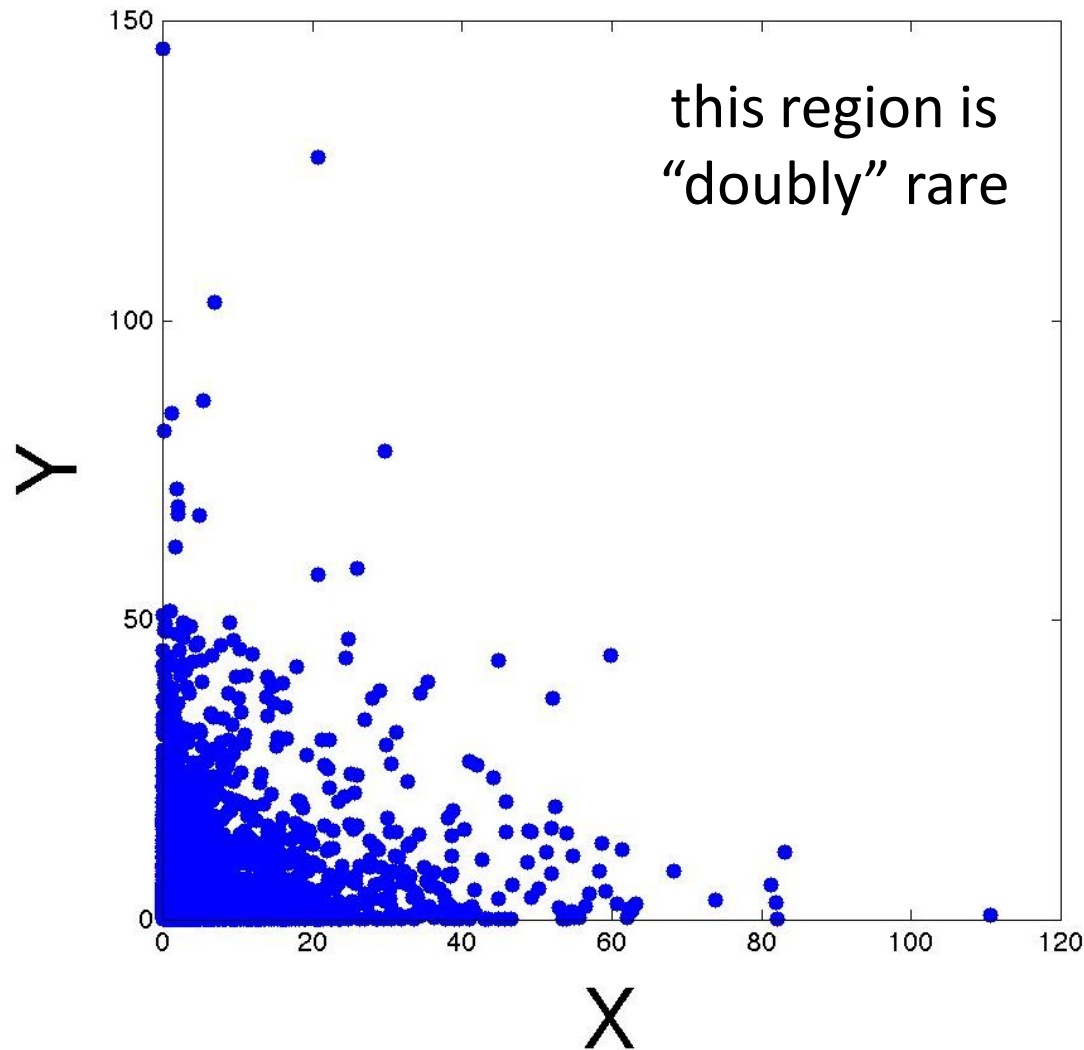
Example thanks to Christian Genest

# What is the dependency structure?



Example thanks to  
Christian Genest

# What is the dependency structure?



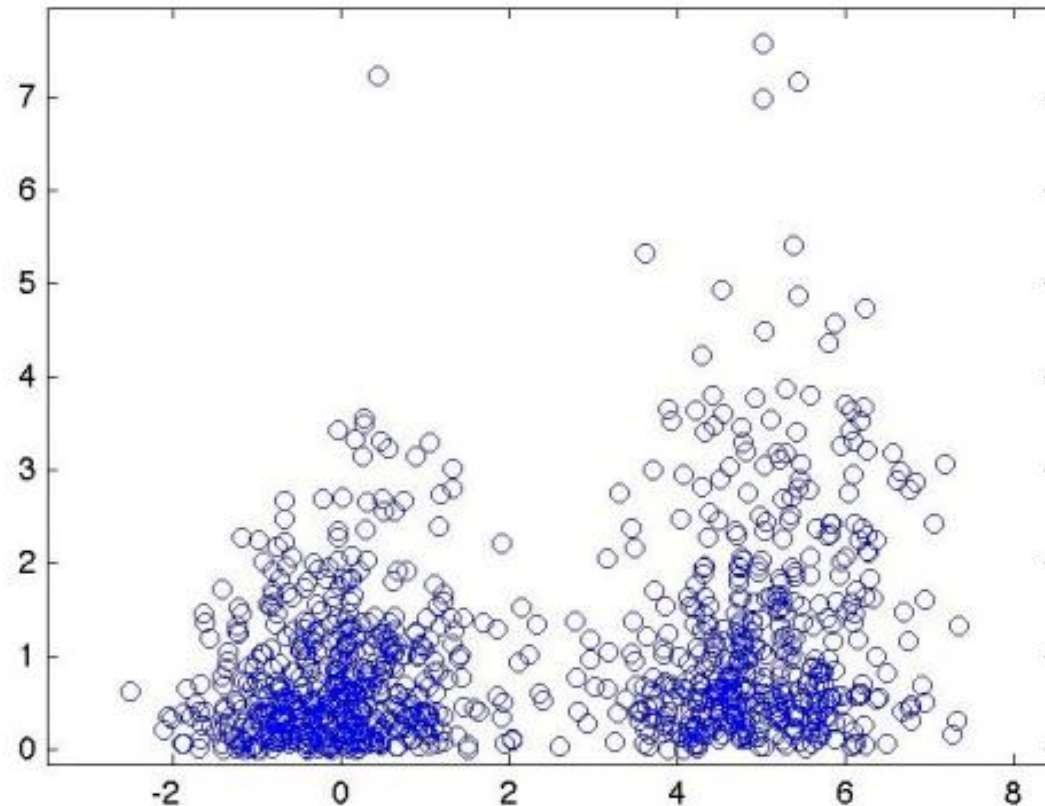
$$X \sim \text{EXP}(\lambda_X)$$

$$Y \sim \text{EXP}(\lambda_Y)$$

$$F_{X,Y}(x,y) = F_X(x)F_Y(y)$$

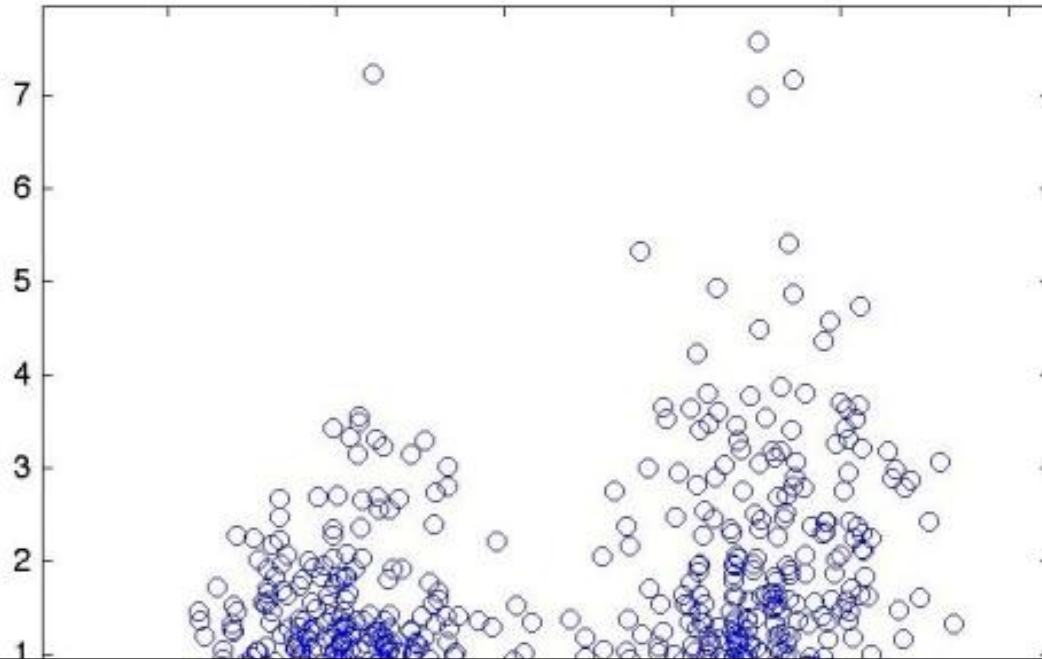
Example thanks to  
Christian Genest

# What is the dependency structure?



Is the dependency multi-modal? Heavy tailed?

# What is the dependency structure?



Humans are inapt at “seeing”  
the dependency structure

# Probability 101 Example

$X_1$  = minimum of the two numbers

$X_2$  = maximum of the two numbers



The variables are obviously dependent ( $X_2 \geq X_1$ )

It is easy to show that:

$$P(X_1 \leq x_1, X_2 \leq x_2) = 2F(\min\{x_1, x_2\})F(x_2) - F(\min\{x_1, x_2\})^2$$

What if we change the numbers of each die to 7, ..., 12?

Obviously, the joint distribution changes.

**But, intuitively, the dependence structure does not!**

# Probability 101 Example

$X_1$  = minimum of the two numbers

$X_2$  = maximum of the two numbers



The variables are obviously dependent ( $X_2 \geq X_1$ )

It is easy to show that:

$$P(X_1 \leq x_1, X_2 \leq x_2) = 2F(\min\{x_1, x_2\})F(x_2) - F(\min\{x_1, x_2\})^2$$

Copulas are all about separating the univariate marginals from all other (dependence) factors

# Part I: Introduction to Copulas

- A dependence prelude
- **What are copulas**
- Copula models
- Copulas and dependence
- Multivariate copulas



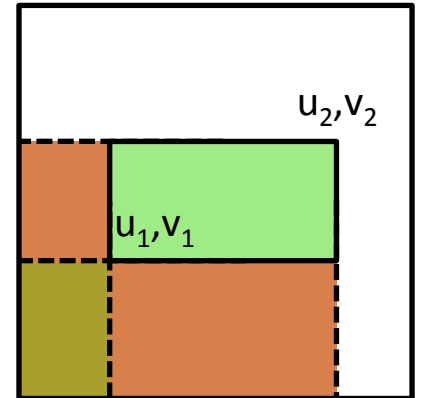
# Copulas

A (bivariate) copula is a function  $C:[0,1]^2 \rightarrow [0,1]$  such that

- for all  $u,v$ :  $C(u, 0) = C(0, v) = 0$
- for all  $u,v$ :  $C(u, 1) = u$  ,  $C(1, v) = v$
- for all  $u_1 \leq u_2, v_1 \leq v_2$ :

$$C(u_2, v_2) - C(u_2, v_1) - C(u_1, v_2) + C(u_1, v_1) \geq 0$$

this is just the positive mass (2-increasing) property



## Equivalent probabilistic definition:

Let  $U_1 \dots U_N$  be real random variables  $\sim U([0,1])$

A copula function  $C:[0,1]^N \rightarrow [0,1]$  is a joint distribution

$$C_\theta(u_1, \dots, u_n) = P(U_1 \leq u_1, \dots, U_n \leq u_n)$$

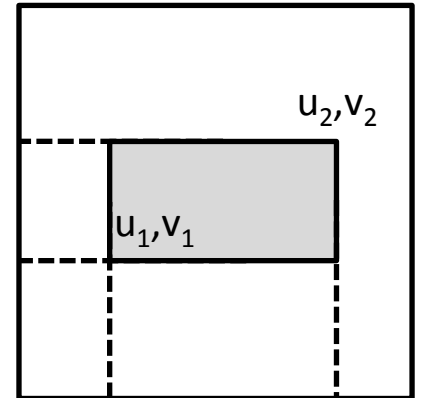
# Copulas

A (bivariate) copula is a function  $C:[0,1]^2 \rightarrow [0,1]$  such that

- for all  $u,v$ :  $C(u, 0) = C(0, v) = 0$
- for all  $u,v$ :  $C(u, 1) = u$  ,  $C(1, v) = v$
- for all  $u_1 \leq u_2, v_1 \leq v_2$ :

$$C(u_2, v_2) - C(u_2, v_1) - C(u_1, v_2) + C(u_1, v_1) \geq 0$$

this is just the positive mass (2-increasing) property



**Equivalent probabilistic definition:**

But uniform random variables are  
uninteresting...

# The Copula Trick

Let  $X \sim F$  be (almost) any continuous RV

What is the distribution of  $F(X)=P(X \leq x)$ ?

$$P(F(X) \leq u) = P(F^{-1}(F(X)) \leq F^{-1}(u))$$

# The Copula Trick

Let  $X \sim F$  be (almost) any continuous RV


What is the distribution of  $F(x)=P(X \leq x)$ ?

$$\begin{aligned} P(F(X) \leq u) &= P(F^{-1}(F(X)) \leq F^{-1}(u)) \\ &= P(X \leq F^{-1}(u)) \\ &= F(F^{-1}(u)) = u \end{aligned}$$

**Constructively:**

1) Choose **any**  $F_i(x_i)$

2)  $F_i(x_i) \sim U([0,1])$  so plug into **any** copula function

  $C_\theta(F_1(x_1), \dots, F_n(x_n))$  is a valid joint distribution!

# How powerful is this framework

Sklar's Theorem (1959): For any joint distribution over  $X_1, \dots, X_N$ , there exists a copula function  $C$  such

$$F_{\mathbf{X}}(x_1, \dots, x_N) = C_{\theta}(F_1(x_1), \dots, F_N(x_N))$$

and if the marginals are continuous, the copula is unique (if discontinuous, see Genest & Neslehova 2007)

## **A word of warning:**

Finding the “right” copula may be as hard as finding  $F_{\mathbf{X}}$ !

## **A word of encouragement:**

We now have significant constructive flexibility!

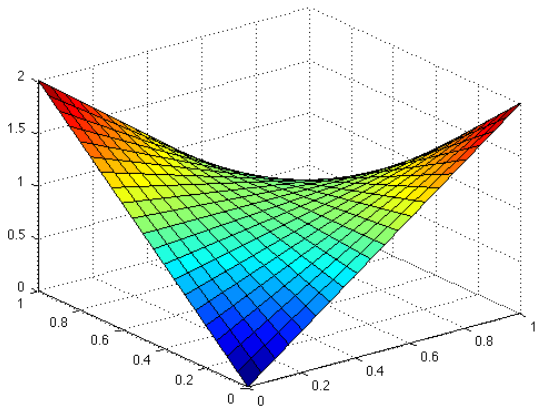
# Part I: Introduction to Copulas

- A dependence prelude
- What are copulas
- Copula models
- Copulas and dependence
- Multivariate copulas

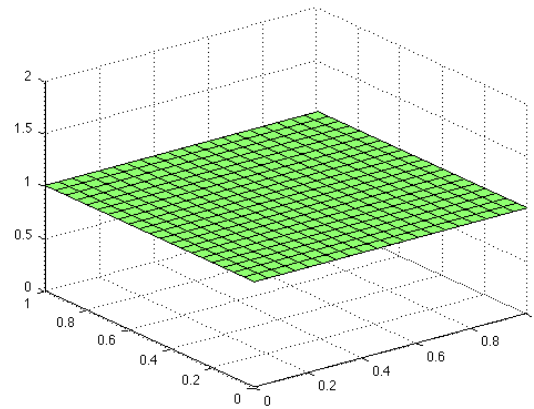
# Example 1: The FGM Copula

An analytically simple copula:

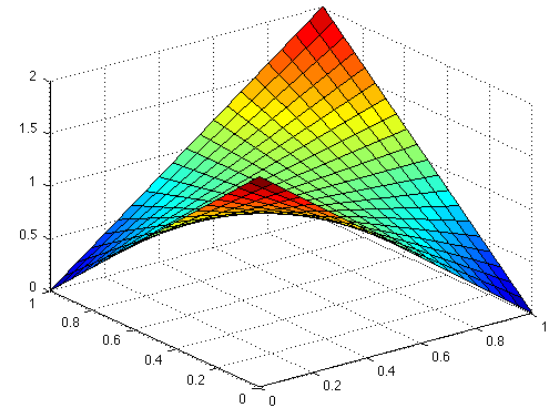
$$C_{\theta}(u, v) = uv + \theta uv(1 - u)(1 - v)$$



$\theta = -1$



$\theta = 0$  (independence)



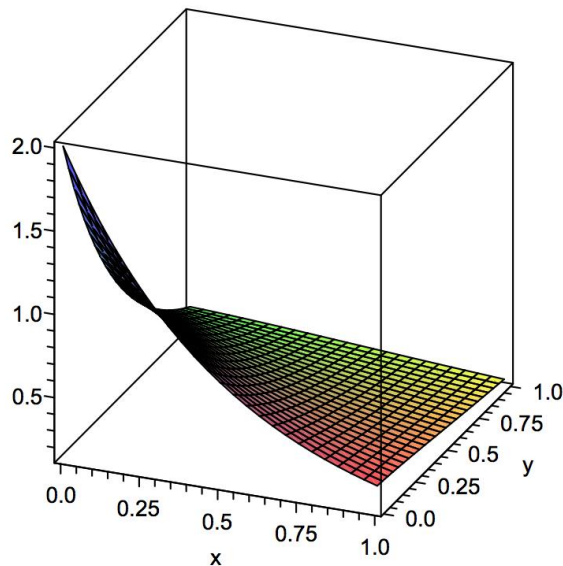
$\theta = +1$

$\theta$  sets “distance” from independence copula  $uv$

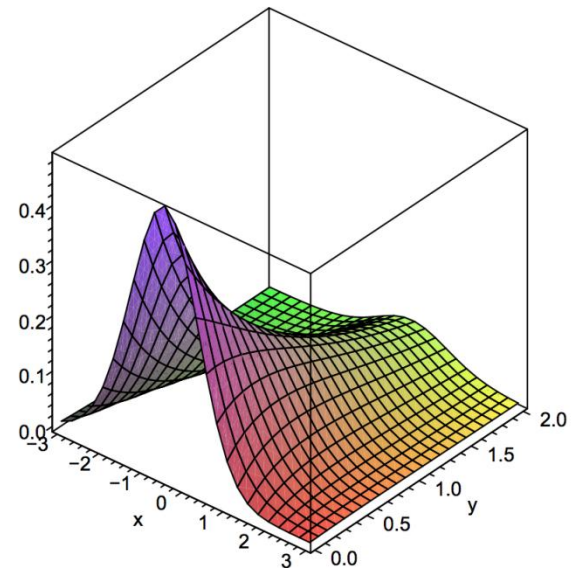
# Example 1: The FGM Copula

An analytically simple copula:

$$C_{\theta}(u, v) = uv + \theta uv(1 - u)(1 - v)$$



with Exp(1) marginals



with Exp(1) and N(0,1) marginals



# Example 2: Inversion of Sklar's

1. Start with a multivariate distribution

$$F_{\mathbf{X}}(\mathbf{X}) = P(X_1 \leq x_1, \dots, X_n \leq x_n) = \Phi_{\Sigma}(x_1, \dots, x_n)$$

in bivariate case copula is specified by  $\rho$

2. Extract its (Gaussian) copula

$$F_{\mathbf{X}}(\mathbf{x}) = \Phi_{\Sigma}(F_1^{-1}(F_1(x_1)), \dots, F_n^{-1}(F_n(x_n)))$$

# Example 2: Inversion of Sklar's

1. Start with a multivariate distribution

$$F_{\mathbf{X}}(\mathbf{X}) = P(X_1 \leq x_1, \dots, X_n \leq x_n) = \Phi_{\Sigma}(x_1, \dots, x_n)$$

in bivariate case copula is specified by  $\rho$

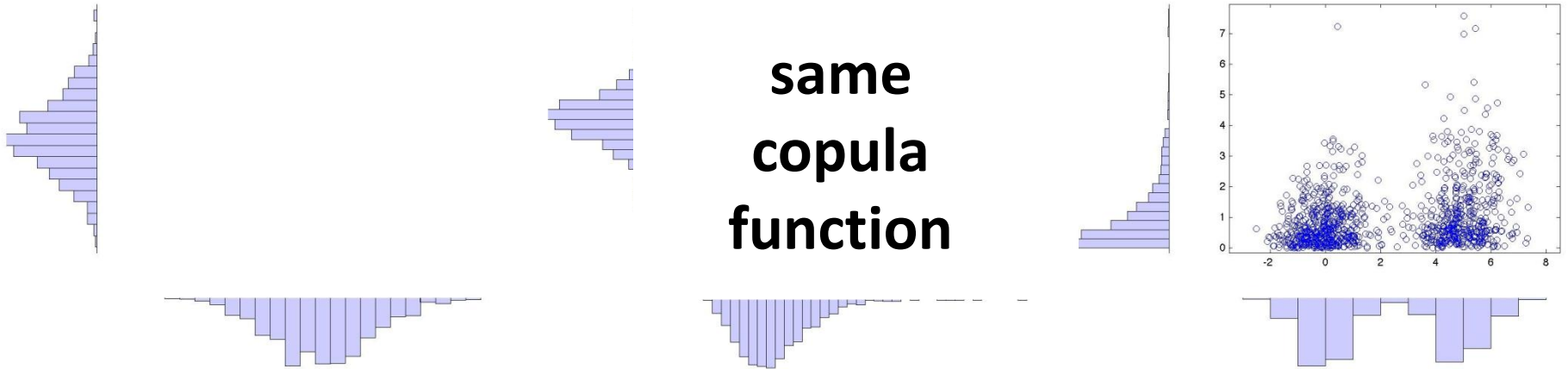
2. Extract its (Gaussian) copula

$$\begin{aligned} F_{\mathbf{X}}(\mathbf{x}) &= \Phi_{\Sigma}(F_1^{-1}(F_1(x_1)), \dots, F_n^{-1}(F_n(x_n))) \\ &= \Phi_{\Sigma}(F_1^{-1}(u_1), \dots, F_n^{-1}(u_n)) \\ &= \Phi_{\Sigma}(\Phi^{-1}(u_1), \dots, \Phi^{-1}(u_n)) \\ &= C_{\Sigma}(\Phi^{-1}(u_1), \dots, \Phi^{-1}(u_n)) \end{aligned}$$

3. Plug in any marginal into our copula function

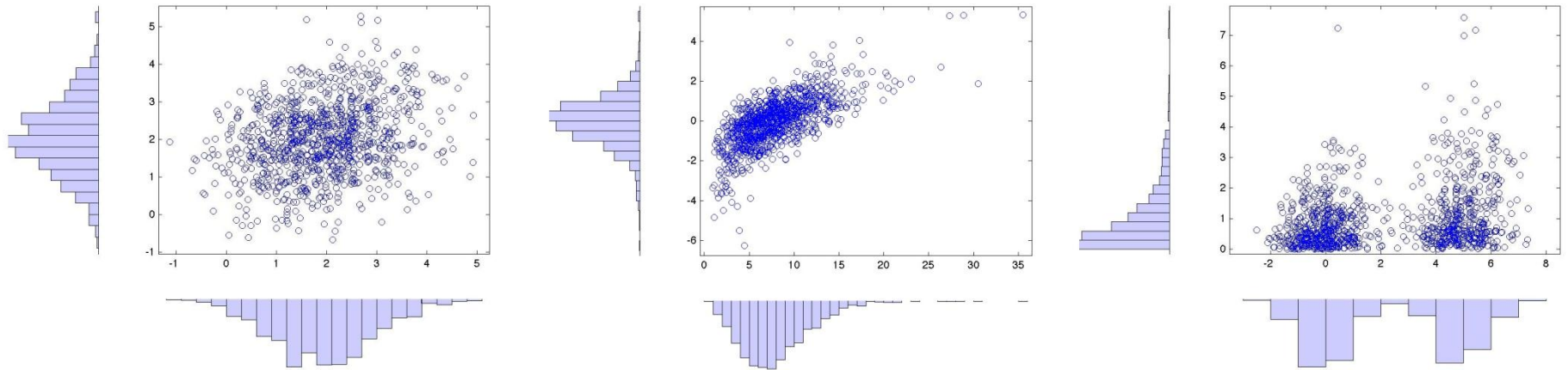
# Example 2: The Gaussian Copula

$$C(\{F_i(x_i)\}) = \Phi_{\Sigma}(\Phi^{-1}(F_1(x_1)), \dots, \Phi^{-1}(F_N(x_N)))$$



# Example 2: The Gaussian Copula

$$C(\{F_i(x_i)\}) = \Phi_{\Sigma}(\Phi^{-1}(F_1(x_1)), \dots, \Phi^{-1}(F_N(x_N)))$$

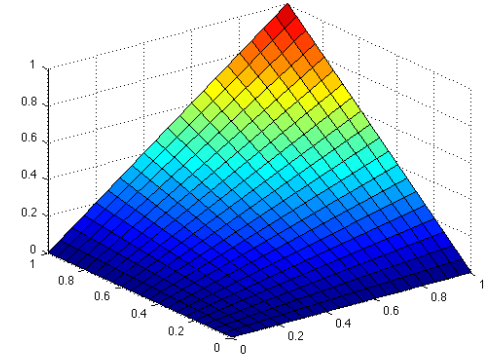


More generally, we can mix any univariate marginals with one of the many copula functions!

# Some Copula Examples

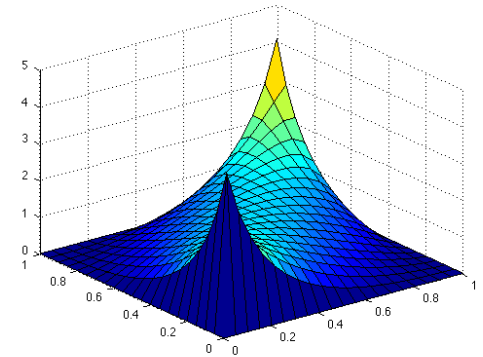
- Independence copula:

$$C(u, v) = uv$$



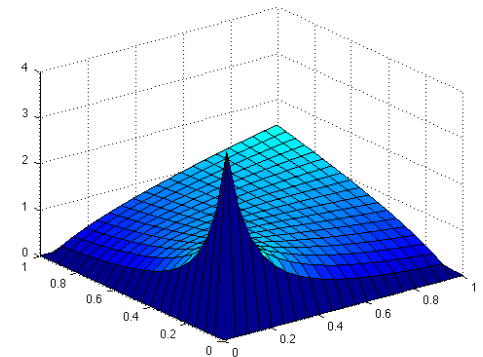
- Gaussian copula (Inversion):

$$C_\rho(u, v) = \Phi_\rho(\Phi^{-1}(u), \Phi^{-1}(v))$$



- Clayton copula (Archimedean):

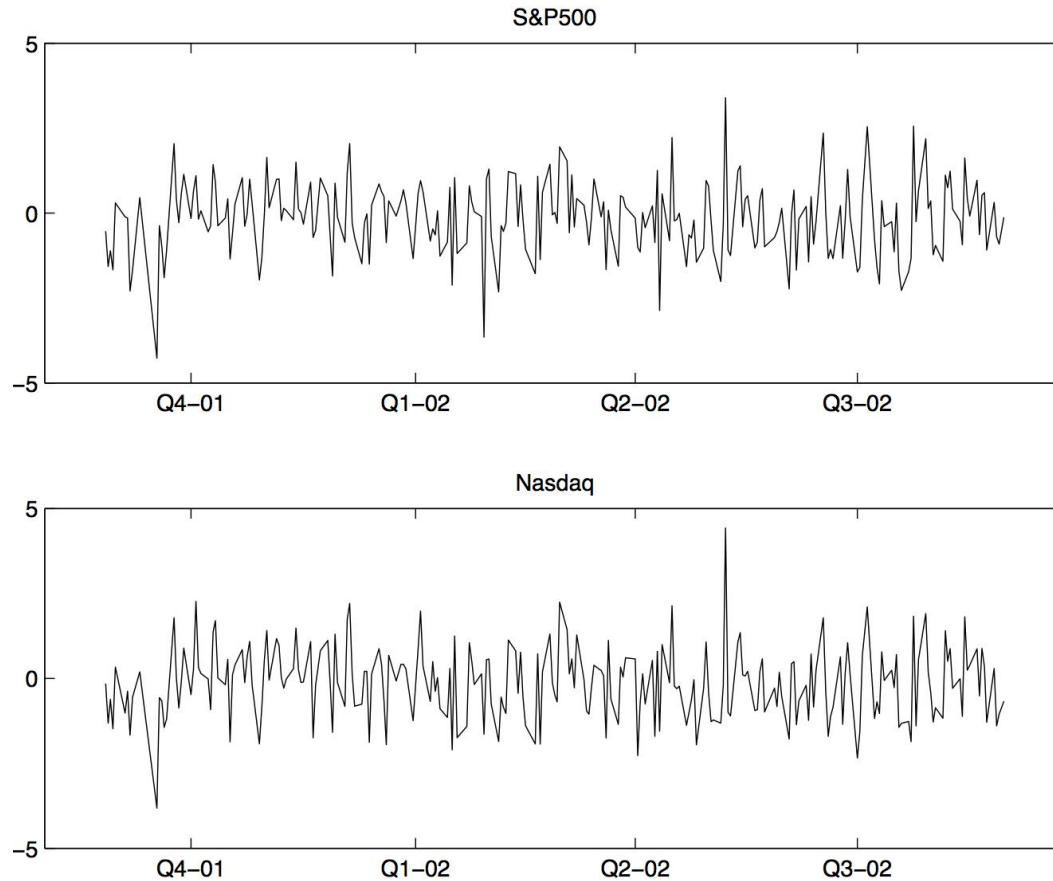
$$C_\theta(u, v) = (u^{-\theta} + v^{-\theta} - 1)^{-1/\theta}$$



- ...

# A Real Example

Consider the Nazdaq and S&P 500 GARCH innovations:



Example from van den Goorbergh et al. 2005 and thanks to Christian Genest

# A Real Example

**How can we view the underlying copula?**

Step 1: Estimate the margins in the most conservative way:

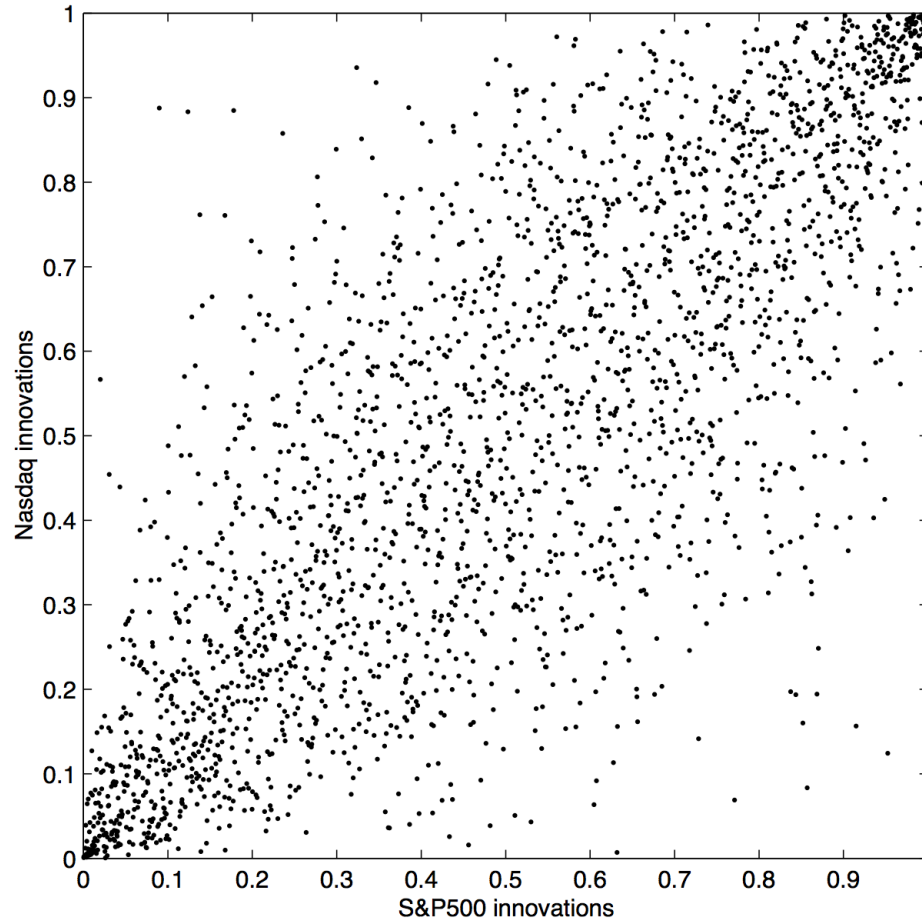
$$F^M(x) = \frac{1}{M+1} \sum_{m=1}^M \mathbf{1}(X[m] \leq x) \quad G^M(y) = \frac{1}{M+1} \sum_{m=1}^M \mathbf{1}(Y[m] \leq y)$$

Step 2: Plot the pairs

$$(\hat{U}[m], \hat{V}[m]) = (F^M(X[m]), G^M(Y[m])) = \left( \frac{R[m]}{M+1}, \frac{S[m]}{M+1} \right)$$

where  $R[m]$  and  $S[m]$  are the ranks of the samples

# A Real Example

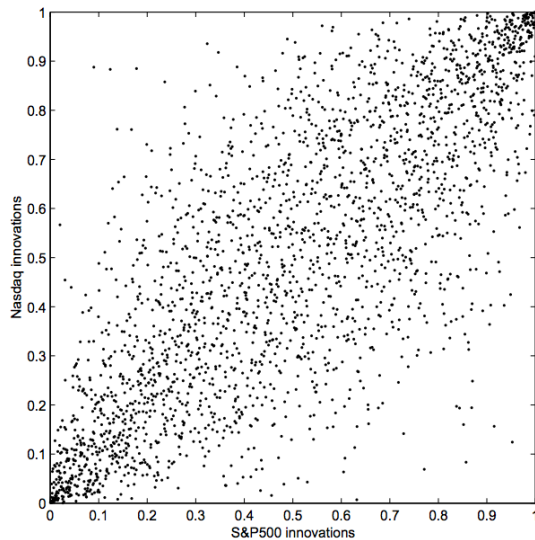


This empirical copula was studied by many starting with Ruschendorf (1976) and has many appealing properties

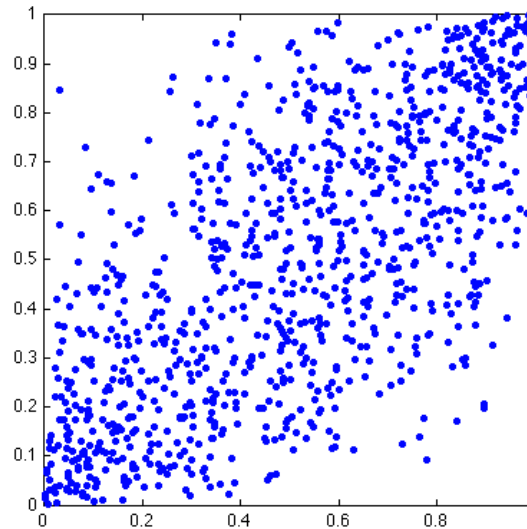


# A Real Example

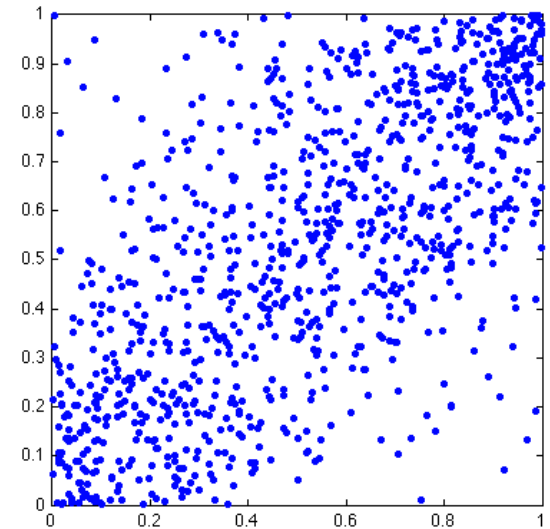
Step 3 (optional?): find a copula with a similar structure



Gaussian copula



Frank copula



Can now perform model selection, followed by estimation followed by model validation (no time for this today)

# Copula Densities

Assuming  $F(x_1, \dots, x_n)$  has  $n$ -order partial derivatives (true almost everywhere for continuous distributions)

$$f(x_1, \dots, x_n) = \frac{F(x_1, \dots, x_n)}{\partial X_1 \dots \partial X_n}$$

# Copula Densities

Assuming  $F(x_1, \dots, x_n)$  has  $n$ -order partial derivatives (true almost everywhere for continuous distributions)

$$\begin{aligned} f(x_1, \dots, x_n) &= \frac{F(x_1, \dots, x_n)}{\partial X_1 \dots \partial X_n} \\ &= \frac{C(F_1(x_1), \dots, F_n(x_n))}{\partial X_1 \dots \partial X_n} \\ &= \frac{C(F_1(x_1), \dots, F_n(x_n))}{\partial F_1(X_1) \dots \partial F_n(X_n)} \prod_i \frac{\partial F_i(X_i)}{\partial X_i} \\ &\equiv \underbrace{c(F_1(x_1), \dots, F_n(x_n))}_{\text{copula density}} \prod_i f_i(x_i) \end{aligned}$$

**And decomposition is always an opportunity...**

# A quick word on estimation

**Caution:** likelihood decomposition is misleading since the copula function depends on univariate marginals

**However, the following procedure:**

1. Estimate marginals first
  2. Estimate dependence parameter second
- is an unbiased, asymptotically Gaussian efficient estimate!

**Caution:** can fail miserably if marginals are misspecified (Kim et al., 2007)

**Solution:** estimate marginals conservatively (as before) (see Genest 1995 for properties)

# Part I: Introduction to Copulas

- A dependence prelude
- What are copulas
- Copula models
- Copulas and dependence
- Multivariate copulas

# Measuring Association

We are interested in measuring association in a way that is invariant to monotone transformations (why?)

What is the simplest measure for interaction between the ranks  $R[m]$ ,  $S[m]$  of the samples  $X[m]$ ,  $Y[m]$ ?

$$\rho(M) = \frac{\sum_m (R[m] - \bar{R})(S[m] - \bar{S})}{\sqrt{\sum_m (R[m] - \bar{R})^2 \sum_m (S[m] - \bar{S})^2}}$$

Or asymptotically (using  $F(X)$  and  $G(Y)$  to denote marginals)

$$\rho_S = \text{corr}\{F(X), G(Y)\}$$

This is **Spearman's Rho** measure of association

# Copulas and Spearman's Rho

$$\rho_s = 12 \int \int C(u, v) du dv - 3$$

Proof: use  $U=F(X)$  and  $V=G(Y)$  to denote marginals

$$\rho_s = \frac{E[F(X)G(Y)] - E[F(X)]E[G(Y)]}{STD(F(X))STD(G(Y))}$$

# Copulas and Spearman's Rho

$$\rho_s = 12 \int \int C(u, v) du dv - 3$$

Proof: use  $U=F(X)$  and  $V=G(Y)$  to denote marginals

$$\begin{aligned} \rho_s &= \frac{E[F(X)G(Y)] - E[F(X)]E[G(Y)]}{STD(F(X))STD(G(Y))} \\ &= \frac{E[F(X)G(Y)] - \frac{1}{2}^2}{\frac{1}{12}} \\ &= 12E[F(X)G(Y)] - 3 \\ &= 12 \int \int uv dC(u, v) - 3 \\ &= 12 \int \int C(u, v) du dv - 3 \end{aligned}$$



# Copulas and Spearman's Rho

$$\rho_s = 12 \int \int C(u, v) du dv - 3$$

Proof: use  $U=F(X)$  and  $V=G(Y)$  to denote marginals

$$\begin{aligned} \rho_s &= \frac{E[F(X)G(Y)] - E[F(X)]E[G(Y)]}{STD(F(X))STD(G(Y))} \\ &= \frac{E[F(X)G(Y)] - \frac{1}{2}^2}{\frac{1}{12}} \end{aligned}$$

Nice, but why is this interesting?

# Copulas and Spearman's Rho

$$\rho_s = 12 \int \int C(u, v) du dv - 3$$

**Fact:** for essentially all copula families, by construction

$$\theta_2 > \theta_1 \rightarrow C_{\theta_2}(u, v) \geq C_{\theta_1}(u, v) \quad \forall u, v$$

This is also called concordance or PQD ordering

Example:  $C_{\theta}(u, v) = uv + \theta uv(1 - u)(1 - v)$

**In this case, Spearman's is a dependence measure (i.e. =0 only if X and Y are independent)**

 copula families define a dependence ordering!

# Copulas and Spearman's Rho

$$\rho_s = 12 \int \int C(u, v) du dv - 3$$

Appealing properties of copulas and Spearman's Rho:

1. Both are non-parametric measures of association
2. Both are invariant to monotone transformations (substantially strengthening Pearson's correlation)
3. Both do not depend on the univariate marginals (by now it should be obvious that we require this)

- Similar relationship with other dependence measures
- Copulas can be viewed as a tool to gauge dependence

# Copulas and Mutual Information

$$I(X, Y) = \int \int f(x, y) \log \frac{f(x, y)}{f_X(x) f_Y(y)} dx dy$$

Probably **THE** dependence measure in ML

**But:** seems like it heavily depends on the marginals...

**Recall:**  $f(x, y) = c(F_X(x), F_Y(y)) f_X(x) f_Y(y)$

It follows that MI is simply the negative copula entropy!

$$I(X, Y) = \int \int c(F_X(x), F_Y(y)) f(x) f(y) \log c(F_X(x), F_Y(y)) dx dy$$

# Copulas and Mutual Information

$$I(X, Y) = \int \int f(x, y) \log \frac{f(x, y)}{f_X(x) f_Y(y)} dx dy$$

Probably **THE** dependence measure in ML

**But:** seems like it heavily depends on the marginals...

**Recall:**  $f(x, y) = c(F_X(x), F_Y(y)) f_X(x) f_Y(y)$

It follows that MI is simply the negative copula entropy!

$$\begin{aligned} I(X, Y) &= \int \int c(F_X(x), F_Y(y)) f(x) f(y) \log c(F_X(x), F_Y(y)) dx dy \\ &= \int \int c(u, v) \log c(u, v) du dv \equiv -H(c(U, V)) \end{aligned}$$

# Part I: Introduction to Copulas

- A dependence prelude
- What are copulas
- Copula models
- Copulas and dependence
- **Multivariate copulas**

# Attempts at multivariate Copulas

Explicit constructions:

- Some of the families we have seen have a multivariate form
- Koehler & Symanowski (1995)
- Morillas (2005)
- Liebscher (2006)
- Fischer & Kock (2007)
- ...

Compositions of bivariate copulas:

Rarely used for more than 10 dimensions  
(will mention an exception later)

# Vines

For **two** variables we have

$$f(x_1 | x_2) f_2(x_2) = c_{12}(F_1(x_1), F_2(x_2)) f_1(x_1) f_2(x_2)$$

$\Rightarrow$

$$f(x_1 | x_2) = c_{12}(F_1(x_1), F_2(x_2)) f_1(x_1)$$

For **three** variables

$$f(x_1 | x_2, x_3) = c_{12|3}(F_{1|3}(x_1|x_3), F_{2|3}(x_2|x_3)) f_{1|3}(x_1|x_3)$$

Or

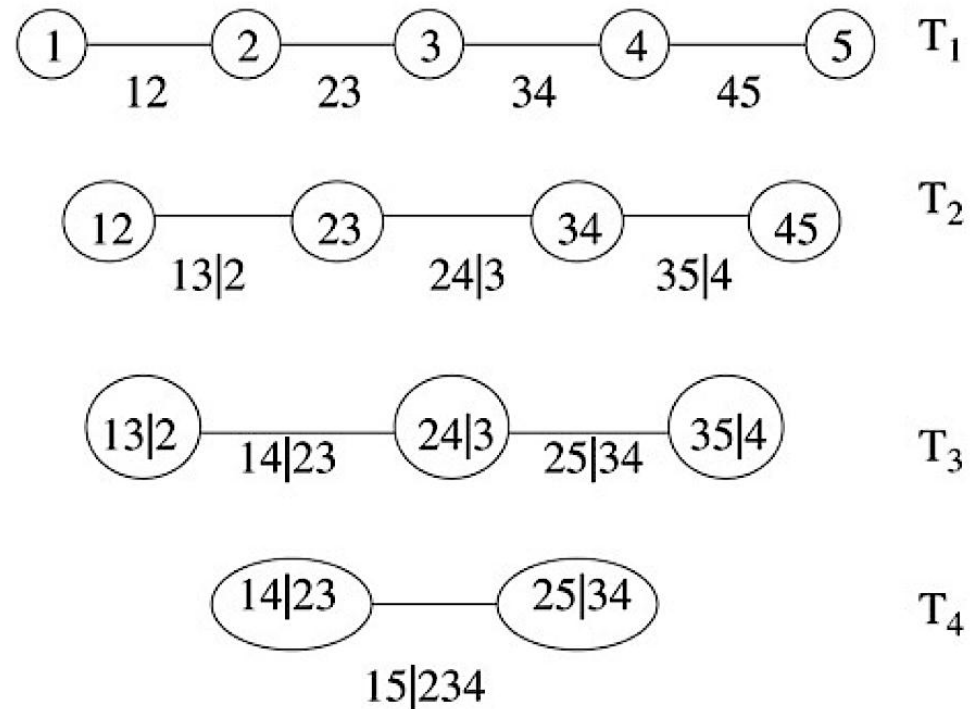
$$f(x_1 | x_2, x_3) = c_{13|2}(F_{1|2}(x_1|x_2), F_{3|2}(x_3|x_2)) f_{1|2}(x_1|x_2)$$

And so on...



# Graphical Representation of a D-Vine

- A bivariate copula is associated with each edge
- Density is defined by product over edge copulas and univariates



➔ a very general and flexible representation that is well understood and uses only bivariate copulas

**So what are we doing here?**

# Limitations of Vines

- High-dimensional conditional terms are hard to estimate!
- Cumbersome construction does not take advantage of independencies
- In practice, only first “levels” have any effect

## 2.4. Five variables

The general expression for the five-dimensional canonical vine structure is

$$\begin{aligned}
 f(x_1, x_2, x_3, x_4, x_5) = & f_1(x_1) \cdot f_2(x_2) \cdot f_3(x_3) \cdot f_4(x_4) \\
 & \cdot f_5(x_5) \cdot c_{12} \{F_1(x_1), F_2(x_2)\} \cdot c_{13} \{F_1(x_1), F_3(x_3)\} \\
 & \cdot c_{14} \{F_1(x_1), F_4(x_4)\} \\
 & \cdot c_{15} \{F_1(x_1), F_5(x_5)\} \cdot c_{23|1} \{F(x_2|x_1), F(x_3|x_1)\} \\
 & \cdot c_{24|1} \{F(x_2|x_1), F(x_4|x_1)\} \cdot c_{25|1} \{F(x_2|x_1), F(x_5|x_1)\} \\
 & \cdot c_{34|12} \{F(x_3|x_1, x_2), F(x_4|x_1, x_2)\} \\
 & \cdot c_{35|12} \{F(x_3|x_1, x_2), F(x_5|x_1, x_2)\} \\
 & \cdot c_{45|123} \{F(x_4|x_1, x_2, x_3), F(x_5|x_1, x_2, x_3)\},
 \end{aligned}$$

and the general expression for the D-vine structure is

$$\begin{aligned}
 f(x_1, x_2, x_3, x_4, x_5) = & f_1(x_1) \cdot f_2(x_2) \cdot f_3(x_3) \\
 & \cdot f_4(x_4) \cdot f_5(x_5) \\
 & \cdot c_{12} \{F_1(x_1), F_2(x_2)\} \cdot c_{23} \{F_2(x_2), F_3(x_3)\} \\
 & \cdot c_{34} \{F_3(x_3), F_4(x_4)\} \\
 & \cdot c_{45} \{F_4(x_4), F_5(x_5)\} \cdot c_{13|2} \{F(x_1|x_2), F(x_3|x_2)\} \\
 & \cdot c_{24|3} \{F(x_2|x_3), F(x_4|x_3)\} \cdot c_{35|4} \{F(x_3|x_4), F(x_5|x_4)\} \\
 & \cdot c_{14|23} \{F(x_1|x_2, x_3), F(x_4|x_2, x_3)\} \\
 & \cdot c_{25|34} \{F(x_2|x_3, x_4), F(x_5|x_3, x_4)\} \\
 & \cdot c_{15|234} \{F(x_1|x_2, x_3, x_4), F(x_5|x_2, x_3, x_4)\}.
 \end{aligned}$$

In the five-dimensional case there are regular vines that are neither canonical nor D-vines. One example is the following:

$$\begin{aligned}
 f(x_1, x_2, x_3, x_4, x_5) = & f_1(x_1) \cdot f_2(x_2) \cdot f_3(x_3) \\
 & \cdot f_4(x_4) \cdot f_5(x_5) \\
 & \cdot c_{12} \{F_1(x_1), F_2(x_2)\} \cdot c_{23} \{F_2(x_2), F_3(x_3)\} \\
 & \cdot c_{13} \{F_1(x_1), F_3(x_3)\} \cdot c_{14} \{F_1(x_1), F_4(x_4)\} \\
 & \cdot c_{15} \{F_1(x_1), F_5(x_5)\} \cdot c_{24|1} \{F(x_2|x_1), F(x_4|x_1)\} \\
 & \cdot c_{25|1} \{F(x_2|x_1), F(x_5|x_1)\} \\
 & \cdot c_{34|12} \{F(x_3|x_1, x_2), F(x_4|x_1, x_2)\} \\
 & \cdot c_{35|12} \{F(x_3|x_1, x_2), F(x_5|x_1, x_2)\} \\
 & \cdot c_{45|123} \{F(x_4|x_1, x_2, x_3), F(x_5|x_1, x_2, x_3)\}.
 \end{aligned}$$

# Limitations of Vines

- High-dimensional conditional terms are hard to estimate!
- Cumbersome construction does not take advantage of independencies
- In practice, only first

## 2.4. Five variables

The general expression for the five-dimensional canonical vine structure is

$$\begin{aligned}
 f(x_1, x_2, x_3, x_4, x_5) = & f_1(x_1) \cdot f_2(x_2) \cdot f_3(x_3) \cdot f_4(x_4) \\
 & \cdot f_5(x_5) \cdot c_{12} \{F_1(x_1), F_2(x_2)\} \cdot c_{13} \{F_1(x_1), F_3(x_3)\} \\
 & \cdot c_{14} \{F_1(x_1), F_4(x_4)\} \\
 & \cdot c_{15} \{F_1(x_1), F_5(x_5)\} \cdot c_{23|1} \{F(x_2|x_1), F(x_3|x_1)\} \\
 & \cdot c_{24|1} \{F(x_2|x_1), F(x_4|x_1)\} \cdot c_{25|1} \{F(x_2|x_1), F(x_5|x_1)\} \\
 & \cdot c_{34|12} \{F(x_3|x_1, x_2), F(x_4|x_1, x_2)\} \\
 & \cdot c_{35|12} \{F(x_3|x_1, x_2), F(x_5|x_1, x_2)\} \\
 & \cdot c_{45|123} \{F(x_4|x_1, x_2, x_3), F(x_5|x_1, x_2, x_3)\},
 \end{aligned}$$

and the general expression for the D-vine structure is

$$\begin{aligned}
 f(x_1, x_2, x_3, x_4, x_5) = & f_1(x_1) \cdot f_2(x_2) \cdot f_3(x_3) \\
 & \cdot f_4(x_4) \cdot f_5(x_5) \\
 & \cdot c_{12} \{F_1(x_1), F_2(x_2)\} \cdot c_{23} \{F_2(x_2), F_3(x_3)\} \\
 & \cdot c_{34} \{F_3(x_3), F_4(x_4)\} \\
 & \cdot c_{45} \{F_4(x_4), F_5(x_5)\} \cdot c_{13|2} \{F(x_1|x_2), F(x_3|x_2)\} \\
 & \cdot c_{24|3} \{F(x_2|x_3), F(x_4|x_3)\} \cdot c_{35|4} \{F(x_3|x_4), F(x_5|x_4)\}
 \end{aligned}$$

Rarely used for more than 10 dimensions  
(will mention an exception next)

t are  
g:

# Distribution free BBNs

**Basic idea:** use vines-like construction to parameterize conditional distributions and combine as in BNs

## Pros:

- Compact and flexible
- To-date only copula model that has been applied to high-dimensions (hundreds of variables)

## Cons:

- Requires conditional correlations – in practice assumes these are specified and limited to Gaussian copula

Most similar to development in ML that we will soon see

# Recommended Reading

- Everything you always wanted to know about copula modeling but were afraid to ask [Genest, 2007]
- Modeling dependence with copulas [Embrechts, 2001]
- Understanding relationships using copulas [Frees & Valdes, 1998]
- The Joy of Copulas [Genest, 1986]
- Coping with Copulas [Schmidt, 2006]

## Book references:

- An Introduction to Copulas [Nelsen, 2006]
- Multivariate Models & Dependence Concepts [Joe, 1997]
- Vine Copula Handbook [Kurwicka & Joe, 2012]

# Part II: Graphical Copula Models

# Scope

- Learning with tree-averaged distributions [Kirshner, 2008], MCMC for Bayes Mix of Copulas [Silva and Gramacy, 2009]
- The Nonparanormal [Liu, Laffery, Wasserman, 2009]
- Copula Bayesian Networks [Elidan, 2010]
- Copula Processes [Wilson and Ghahramani, 2010]

## What will not be covered:

- Kernel-based copula processes [Jaimungal and Ng, 2009]
- Mixed cumulative distribution networks [Silva et al., 2011]
- Inference-less inference, copula network classifiers, lightning-speed structure learning [Elidan, 2010, 2012]

# Markov Networks

U is an undirected graph that encodes independencies:

$$X_i \perp \mathcal{X} - \{X_i\} - N(X_i) \mid N(X_i)$$

where  $N(X_i)$  are the neighbors of  $X_i$  in U

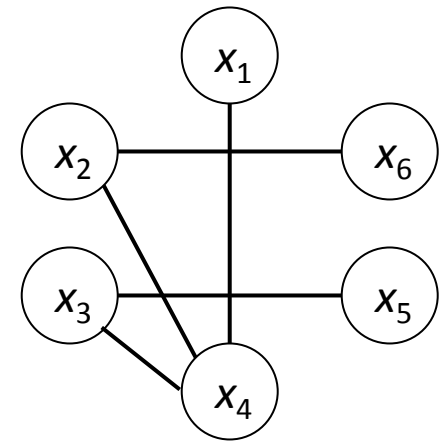
**Theorem** (Hammersley-Clifford):

If  $f$  is positive and the independencies hold then it factorizes according to U



**For trees:**

$$f_{\mathcal{X}}(\mathbf{x}) = \left[ \prod_i f_i(x_i) \right] \left[ \prod_{(i,j) \in E} \frac{f_{i,j}(x_i, x_j)}{f_i(x_i) f_j(x_j)} \right]$$





# From Bivariate Copulas to Copula Trees

It follows that the joint copula also decomposes:

$$c_{\mathcal{X}}(\{F_i(x_i)\}) = \frac{f_{\mathcal{X}}(\mathbf{x})}{\prod_i f_i(x_i)} =$$

# From Bivariate Copulas to Copula Trees

It follows that the joint copula also decomposes:

$$\begin{aligned} c_{\mathcal{X}}(\{F_i(x_i)\}) &= \frac{f_{\mathcal{X}}(\mathbf{x})}{\prod_i f_i(x_i)} = \prod_{(i,j) \in E} \frac{f_{i,j}(x_i, x_j)}{f_i(x_i) f_j(x_j)} \\ &= \prod_{(i,j) \in E} c_{i,j}(F_i(x_i), F_j(x_j)) \end{aligned}$$



Given marginals, we can find the optimal tree efficiently using a maximum spanning tree algorithms

**Upside:** only bivariate estimation (different than Vines!)

**Downside:** assumptions are too simplistic

# Mixture of All Trees

**Challenge:** there are  $N^{(N-2)}$  trees

**Idea:** use edge weight matrix  $\beta$  to define a prior over trees

$$P(T | \beta) = \frac{1}{Z} \prod_{(i,j) \in T} \beta_{i,j} \quad \text{with} \quad Z = \sum_T \prod_{(i,j) \in T} \beta_{i,j}$$

**Theorem** (Meila and Jaakkla 2006):

1. Easy to compute  $Z$  (via generalized Laplacian matrix)
2. Decomposability of the prior allows us to compute average over all trees efficiently



Average density over copula trees (still a copula!) can be computed via ratio of matrix determinants

# Estimation using EM

**Parameters:** 1) the edge weight matrix  $\beta$   
2) the bivariate copula parameters  $\theta_{ij}$

**E-Step:** need to compute posterior over  $N^{(N-2)}$  trees!

**Decomposability**  $\Rightarrow$  need only compute  $N(N-1)/2$  edge probabilities and reuse computations.

**M-Step:** standard optimization of bivariate copulas that depends only on pairs of variables



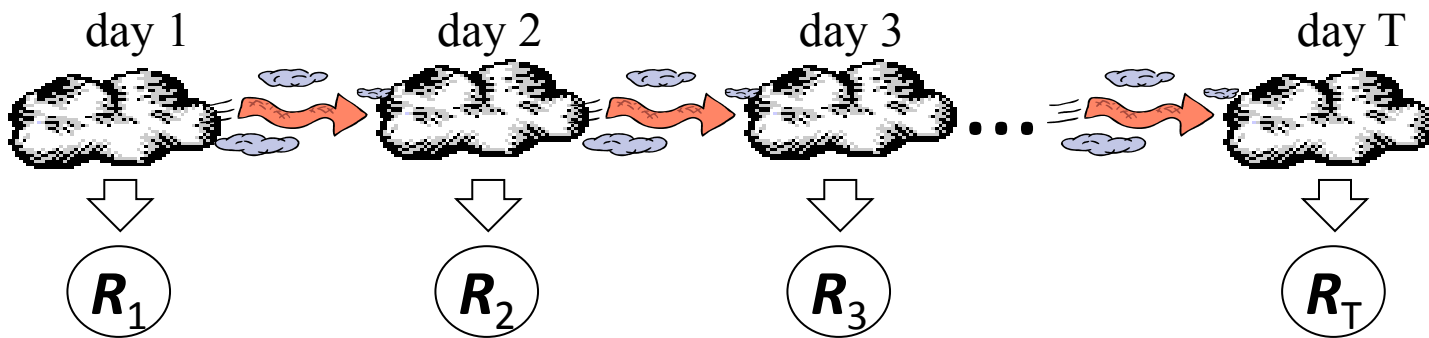
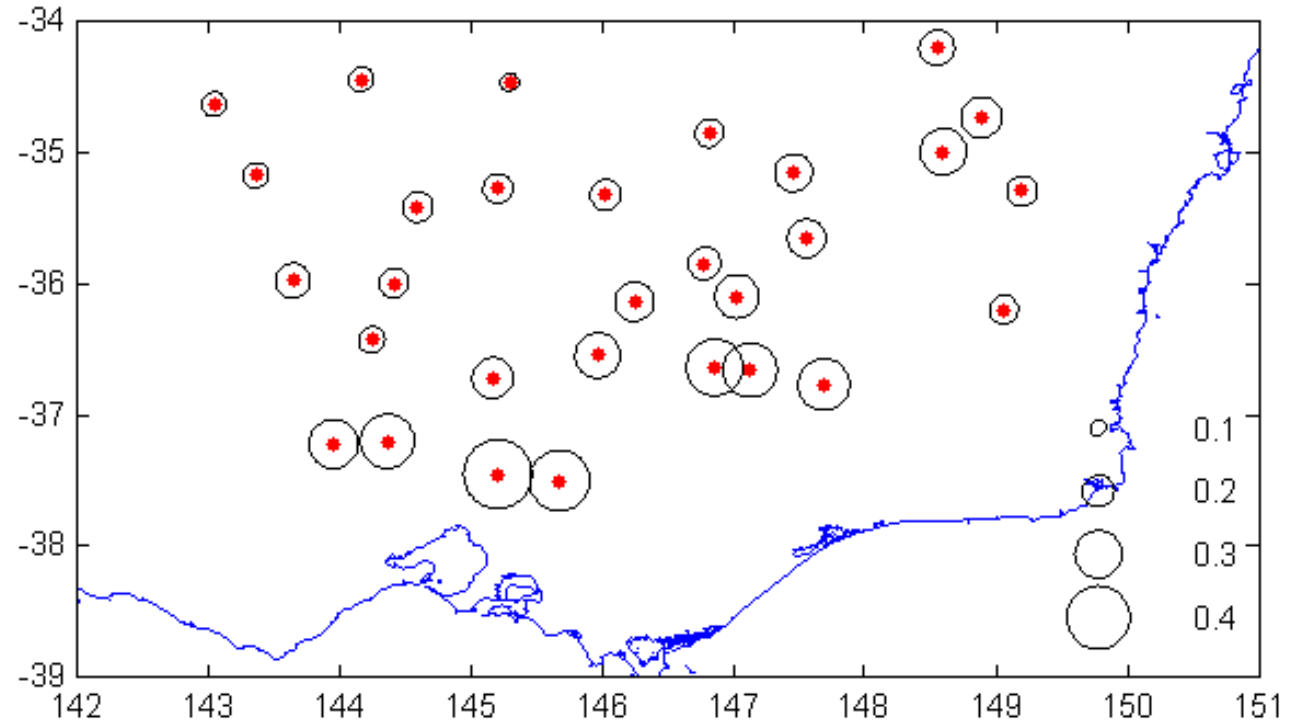
Assuming copula estimation complexity of  $O(M)$ :  
complexity of learning the model is  $O(MN^3)$

**Practical for tens of variables!**

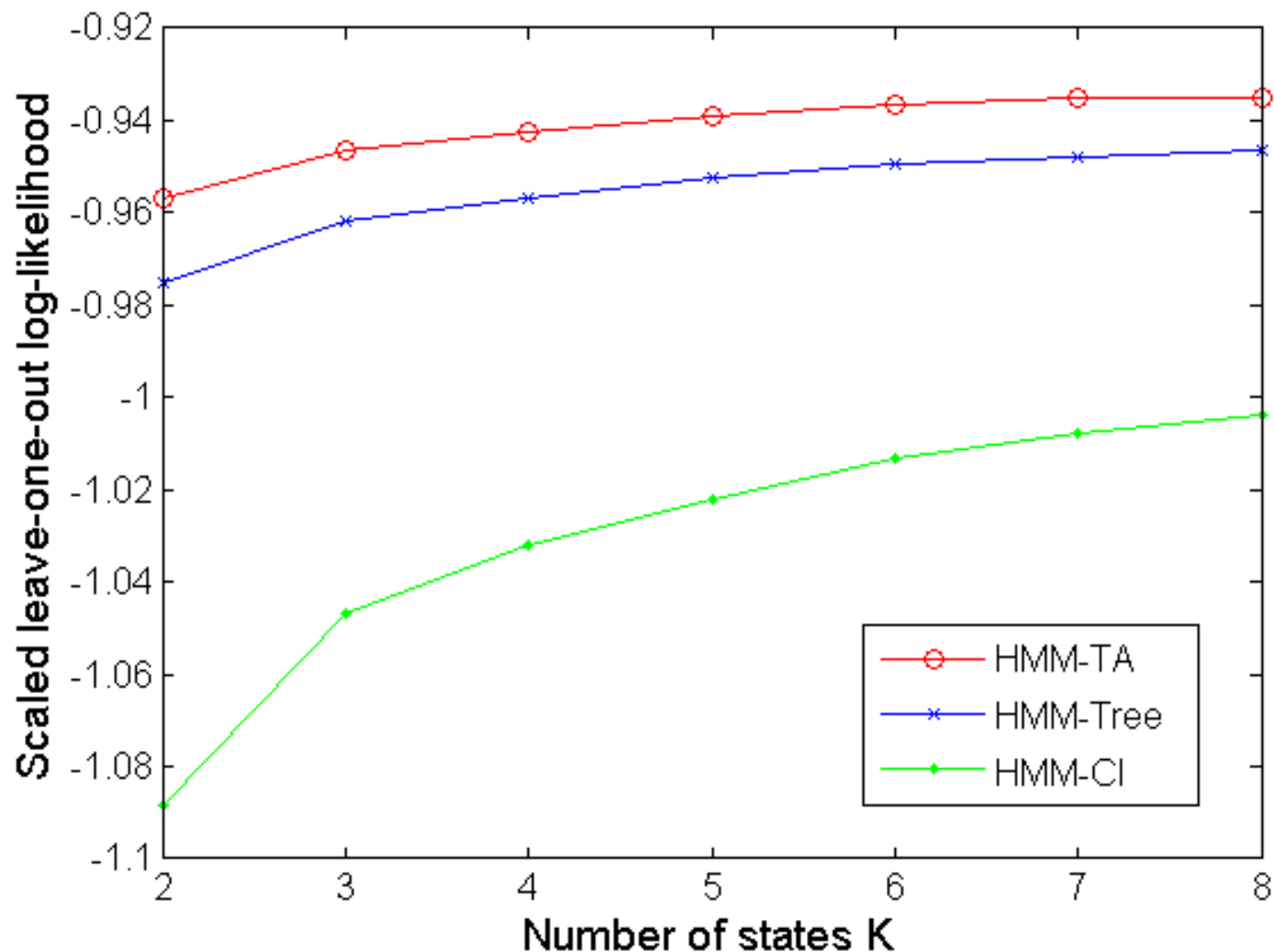
# Modeling Daily Multi-Site Rainfall

N stations with  
unique marginals  
(10-40)

M observed days  
(3000-8000)



# Selecting Number of States



# More Bayesian, More Flexibility

**Advantage of Kirshner:** the set of all trees is parameterized by a matrix with  $O(N^2)$  parameters

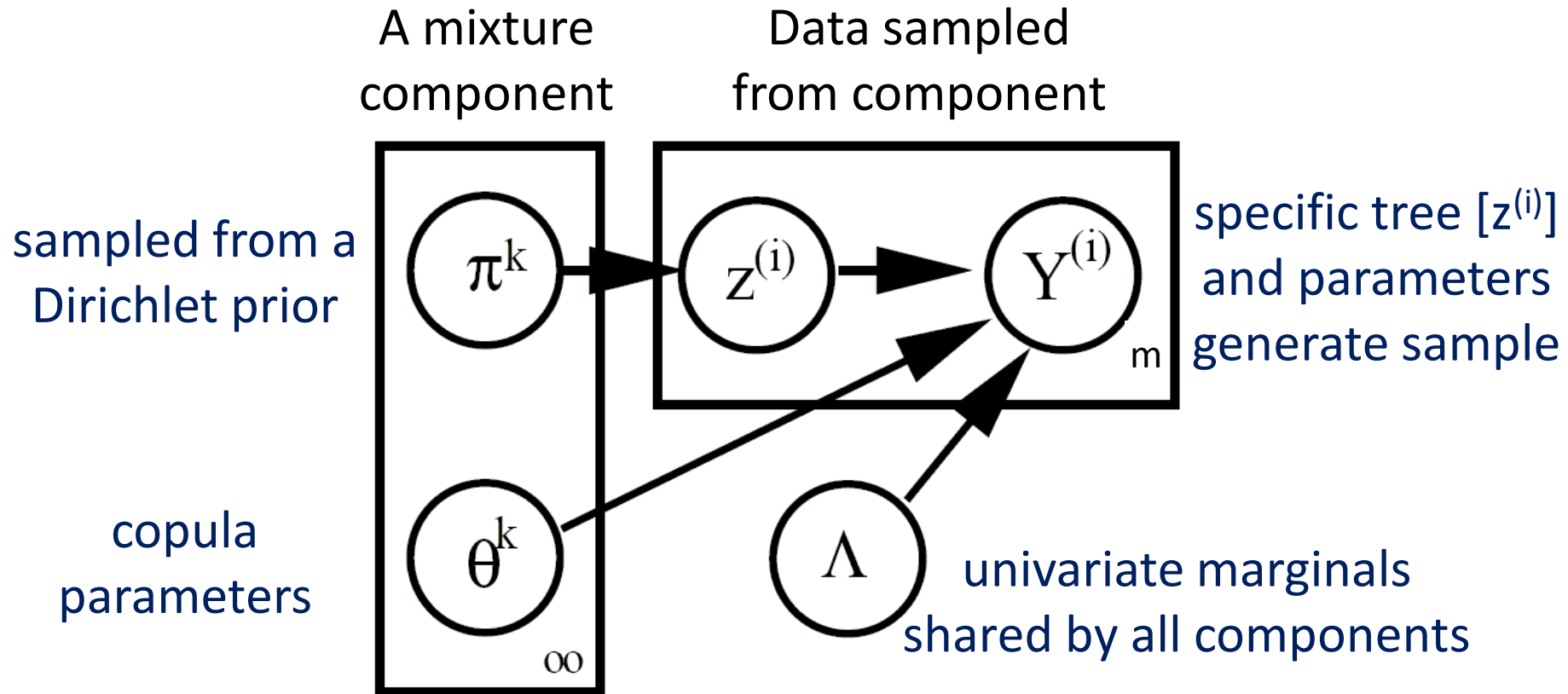
**Limitations of Kirshner:** the set of all trees is parameterized by  $O(N^2)$  parameters

- Heavy parameter sharing
- Matrix and mixture proportions are learned using MLE
- No possibility of using some trees

**Goal:** use Bayesian paradigm to allow for more flexibility

# More Bayesian, More Flexibility

**Idea:** a Dirichlet Process with shared univariate marginals





# Markov Chain Monte Carlo

As usual, the devil is in the computations:

- Given a set of trees and cluster assignments, propose parameters in the standard way
- Given a set of tree and parameters, proposed cluster assignments in the standard way
- Given fixed parameters and cluster assignments, proposing trees is a potentially problematic combinatorial problem

# Caution

**Recall:** need to maintain parameters  $\theta_{i,j}$  for all  $i,j$

**But:** given one tree  $T$ , with  $e_{i,j}$  edge indicators

$$c_{\mathcal{X}}(\{F_i(x_i)\}) = \prod_{(i < j)} c_{i,j}(F_i(x_i), F_j(x_j))^{e_{i,j}}$$

➡ some  $\theta_{i,j}$  are independent of data!  
(and will be useless later if sampled from prior)

➡ requires innovative sampling of trees with parameters (Silva and Gramacy, 2009)

# Consistent estimation in high-dimension

Assumptions	Dimension	Regression	Graphical Models
Parametric	Low	Linear model	Multivariate normal
	High	LASSO	Graphical LASSO
Nonparametric	Low	Additive model	?
	High	Sparse additive model	

**Goal:** theoretically founded estimation for nonparametric high-dimensional undirected graphs

# The Nonparanormal Distribution

$X = (X_1, \dots, X_p)^\top \sim \mathbf{NPN}(\mu, \Sigma, \mathbf{f})$  if there exists univariate functions  $\{f_j(X_i)\}$  such that

$$(f_1(X_1), \dots, f_p(X_p)) \sim N(\mu, \Sigma)$$

**Isn't this is just a Gaussian copula?**

**Yes**, if  $f_i(X_i)$  are monotone and differentiable

**So what is the problem?**

- High-dimensionality leads to estimation issues ( $p > n$ )
- Plugging in the empirical distribution does not work in the semiparametric case...

# Density-less Structure Estimation

Let  $h_j(x) = \Phi^{-1}(F_j(x))$  and  $\Lambda$  be the covariance of  $h(x)$

**Key insight:**  $(X_j \perp X_i | \text{rest})$  if and only if  $\Lambda_{ij}^{-1} = 0$



can estimate structure solely from ranks

1. Replace observation with normal score

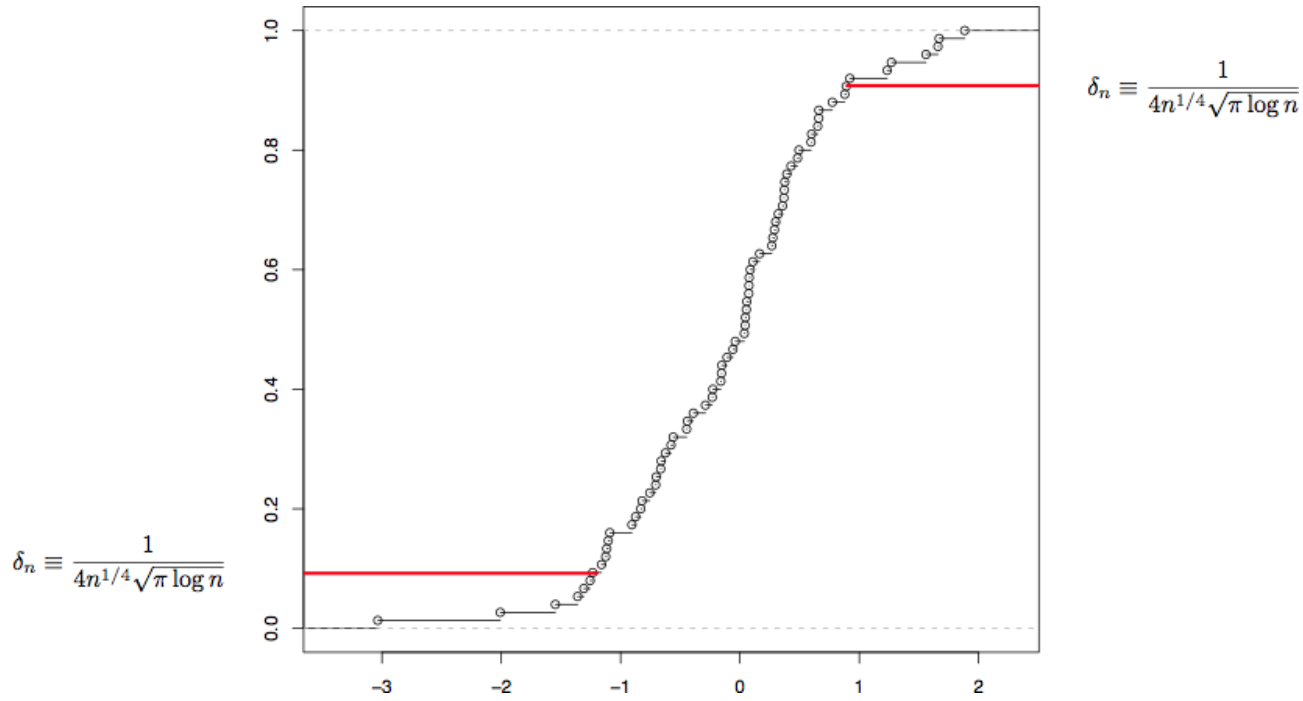
$$\tilde{f}_j(x) = \Phi^{-1}(\tilde{F}_j(x))$$

2. Compute functional sample covariance

$$S_m(\tilde{f}) = \frac{1}{m} \sum_{i=1}^m \tilde{f}(X[i]) \tilde{f}(X[i])^T$$

3. Estimate structure from  $S_m(\tilde{f})$  (e.g. using glasso)

# Winsorized Estimator $\tilde{F}_j$



**Main result:**  $\max_{i,j} \left| S_m(\tilde{f})_{ij} - S_m(f)_{ij} \right| = o_P(m^{-1/4})$



risk, norm (of  $\Sigma$ ) and model selection consistency  
(using analysis of Rothman et al, 2008, and Ravikumar, 2009)

# Synthetic Structure Recovery

- 40 nodes
- 2 different transforms
- several training sample sizes

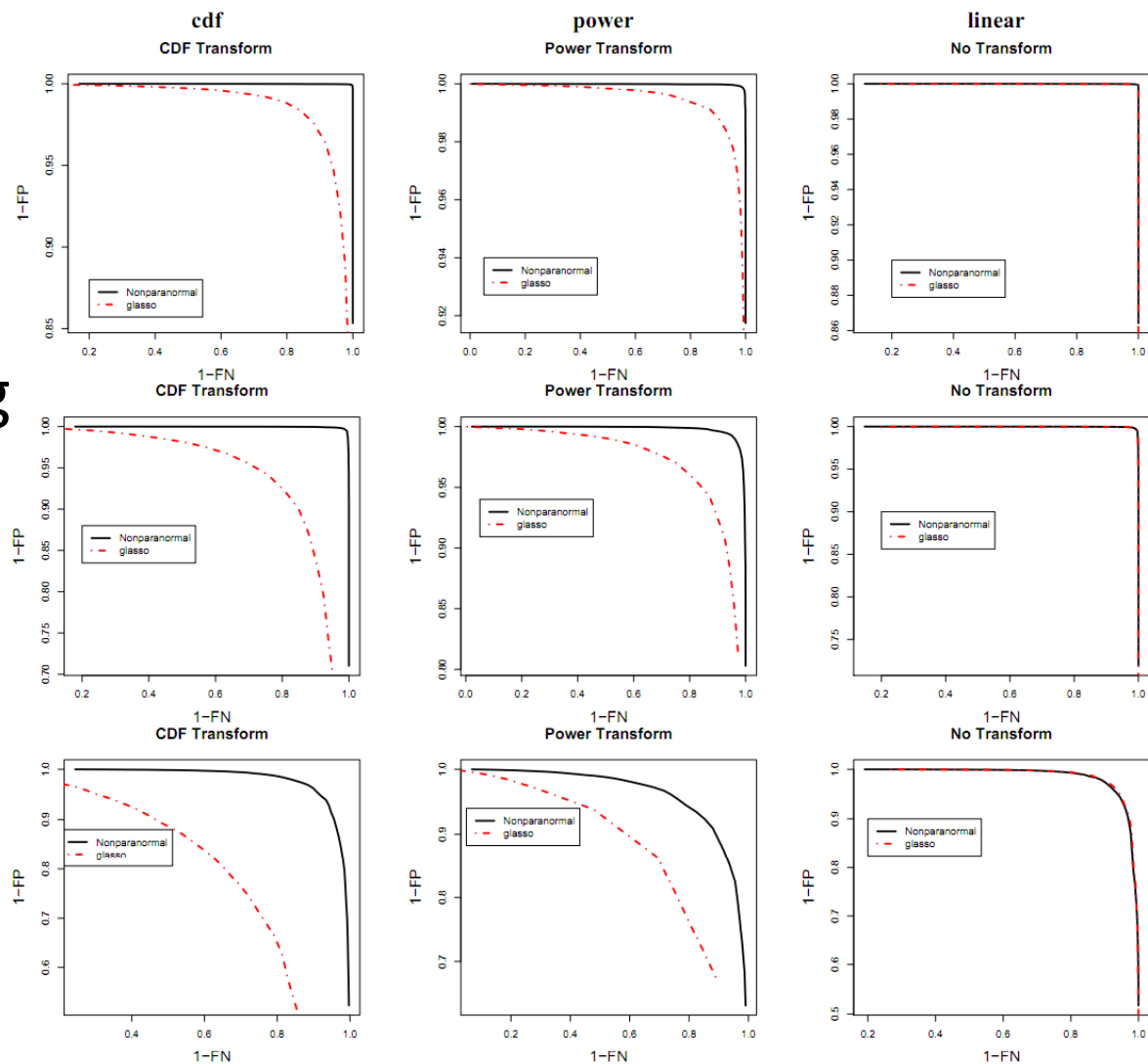
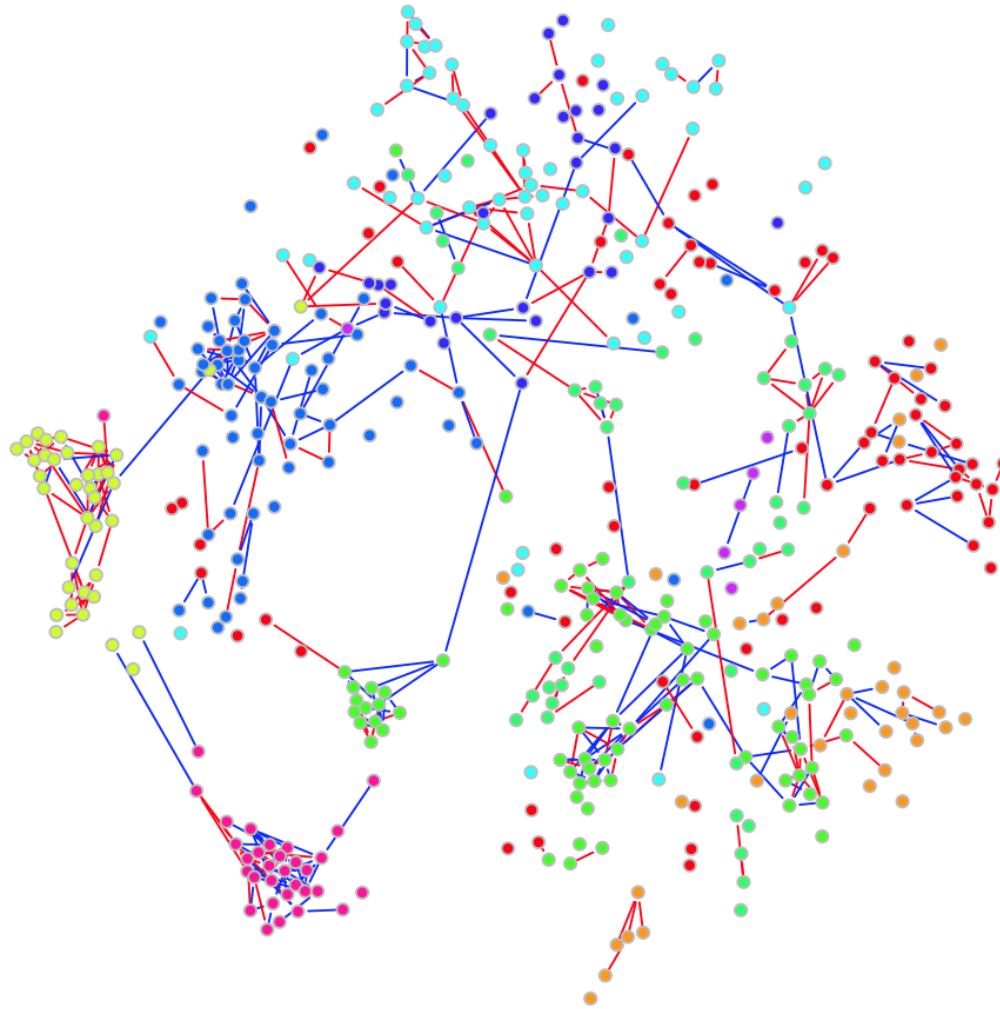


Figure 7: ROC curves for sample sizes  $n = 1000, 500, 200$  (top, middle, bottom).

# S&P 500: differences from glasso



Non-Gaussian case possibly reveals new useful information



# Bayesian Networks

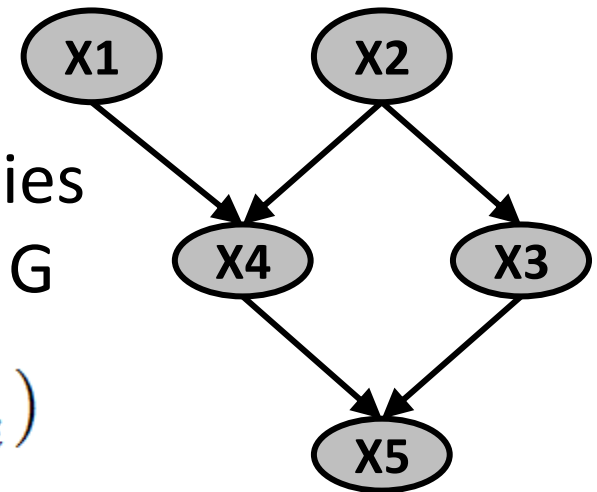
G is a directed graph that encodes independencies:

$$X_i \perp \text{Non-descendants}_i \mid \text{Parents}_i$$

## Theorem:

If  $f$  is positive and the independencies hold then it factorizes according to G

$$f_{\mathcal{X}}(\mathbf{x}) = \prod_i f_{i|\text{par}_i}(x_i \mid x_{\text{par}_i})$$



- ✓ Intuitive representation of uncertainty
- ✓ Easy to construct using local  $f_{i|\text{par}_i}(x_i \mid x_{\text{par}_i})$

# Conditional Densities Using Copulas

Simple bivariate case:

$$f(x|y) = \frac{f(x,y)}{f(y)}$$

# Conditional Densities Using Copulas

Simple bivariate case:

$$f(x|y) = \frac{f(x,y)}{f(y)} = \frac{c(F(x),F(y))f(x)f(y)}{f(y)} = c(F(x),F(y))f(x)$$

**Theorem:** For any  $f(x|\mathbf{y})$ , there **exists** a copula such that

$$f(x|\mathbf{y}) = R_c(F(x), F(y_1), \dots, F(y_K))f(x)$$

# Conditional Densities Using Copulas

Simple bivariate case:

$$f(x|y) = \frac{f(x,y)}{f(y)} = \frac{c(F(x),F(y))f(x)f(y)}{f(y)} = c(F(x),F(y))f(x)$$

**Theorem:** For any  $f(x|\mathbf{y})$ , there **exists** a copula such that

$$\begin{aligned} f(x|\mathbf{y}) &= R_c(F(x), F(y_1), \dots, F(y_K)) f(x) \\ &\equiv \frac{c(F(x), F(y_1), \dots, F(y_K))}{\frac{\partial^K C(1, F(y_1), \dots, F(y_K))}{\partial F(y_1) \dots \partial F(y_K)}} f(x) \end{aligned}$$

**And constructive converse also holds!**

simpler than the  
copula density!

# From local to global Copulas

**Theorem:** if the independencies in  $G$  hold then

$$c(F_1(x_1), \dots, F_n(x_n)) = \prod_i R_{c_i}(F_i(x_i), \{F_{ik}(\mathbf{pa}_{ik})\})$$

and (partially) vice-versa



A **Copula Network** defines a valid joint density

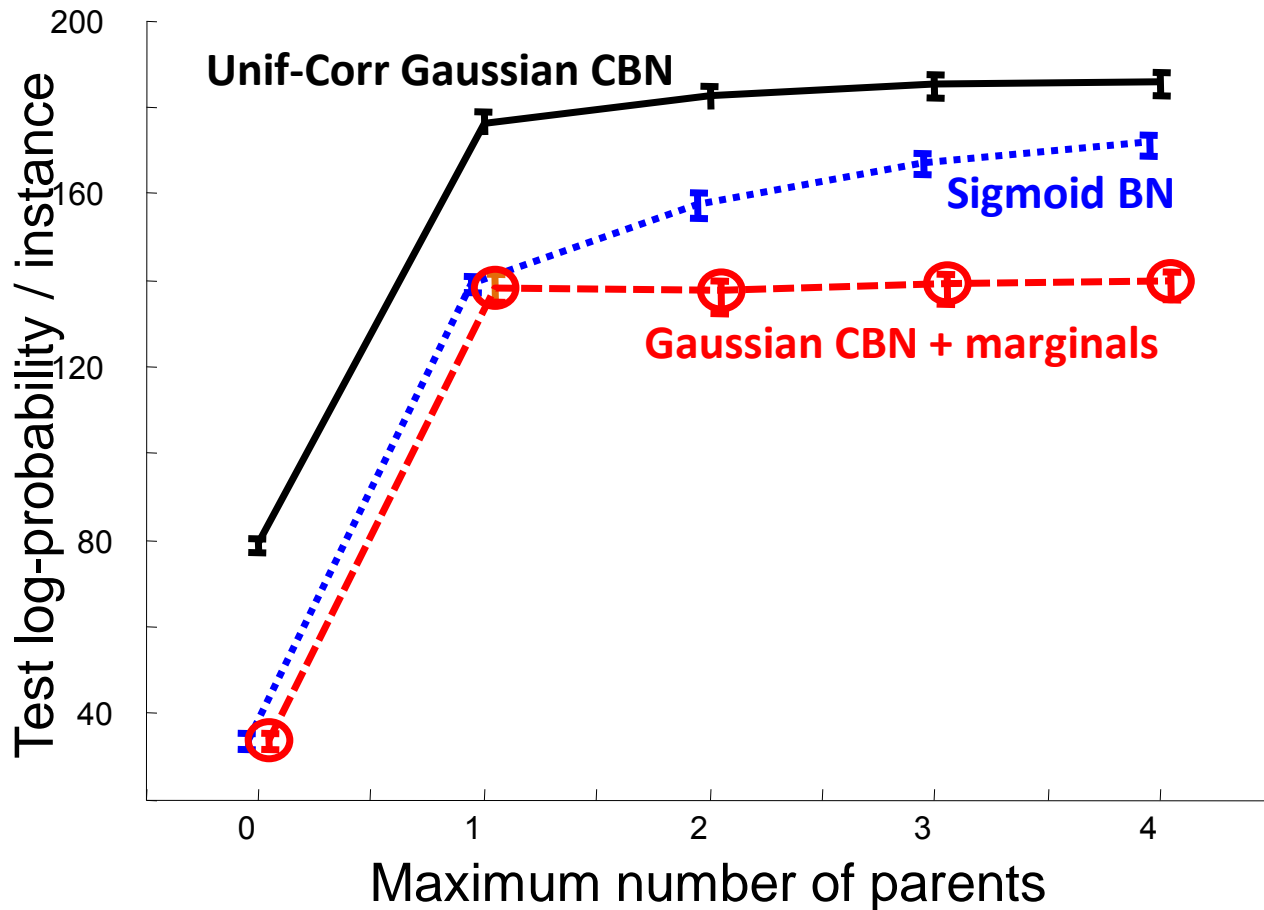
$$f(\mathbf{x}) = \prod_i R_{c_i}(F_i(x_i), \{F_{ik}(\mathbf{pa}_{ik})\}) f_i(x_i)$$



we can now use graphical model tools!

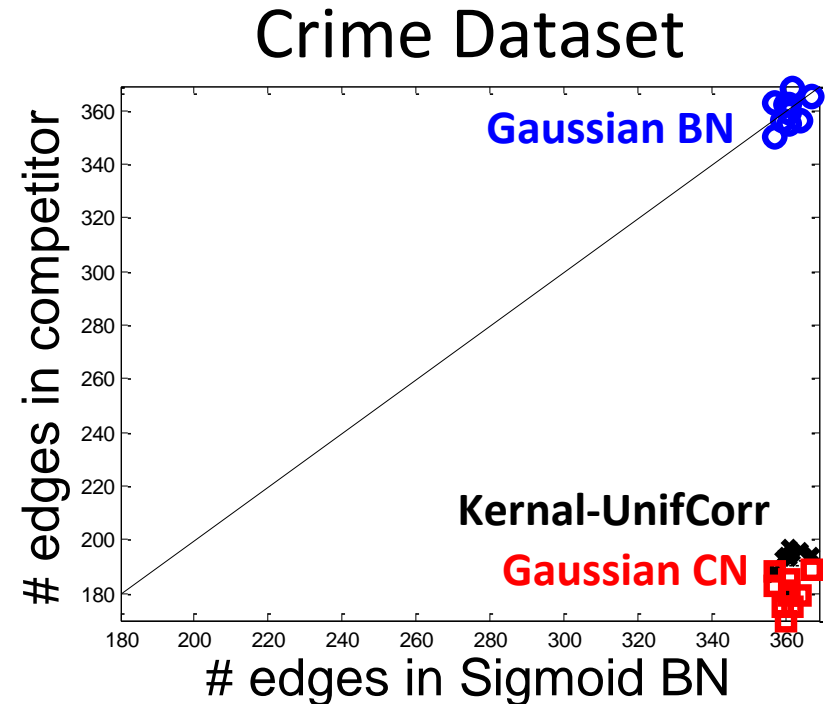
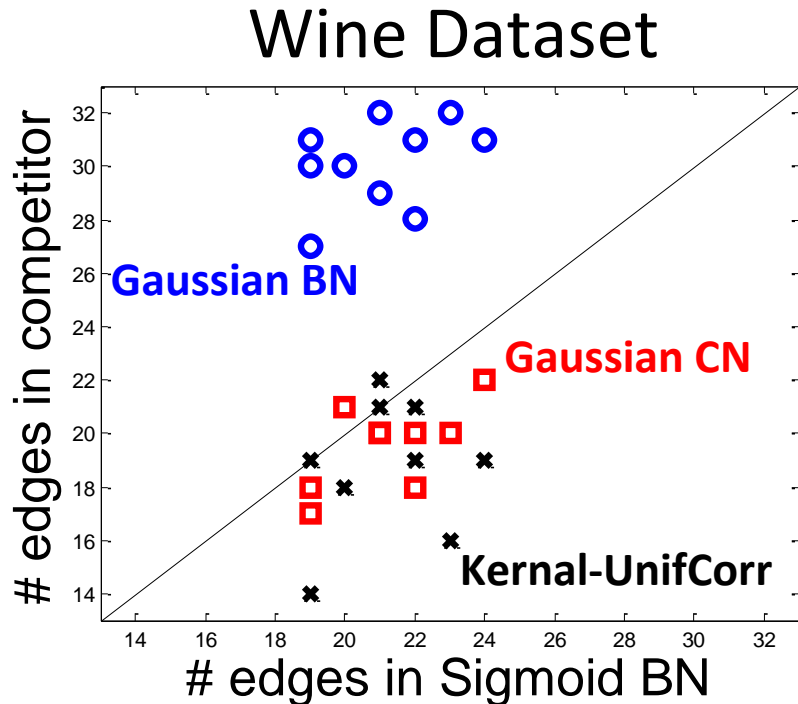
Note: this is similar to non-parameteric BBNs (Hanea 2009) without relying on conditional rank correlations

# Crime (100 variables)



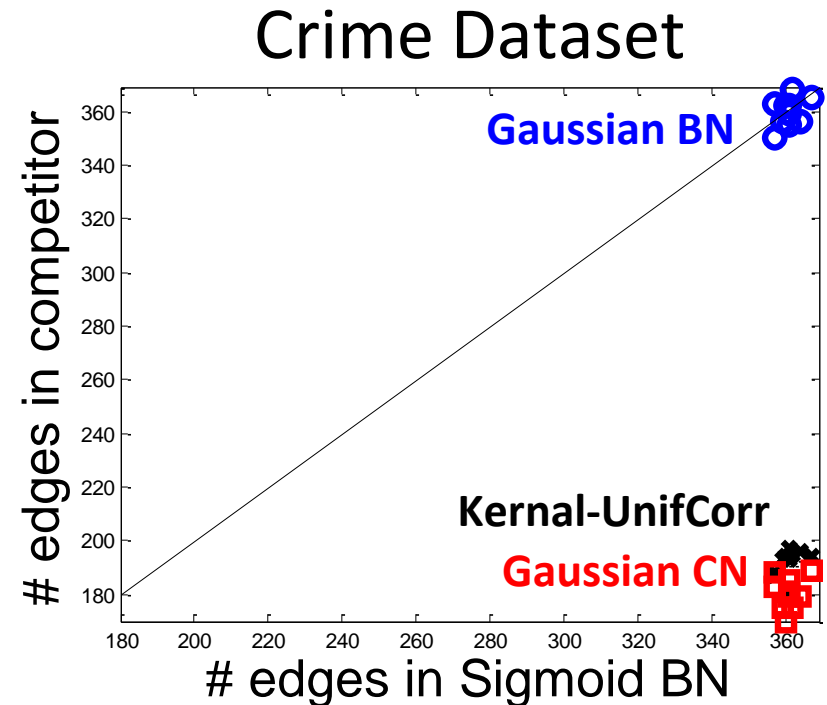
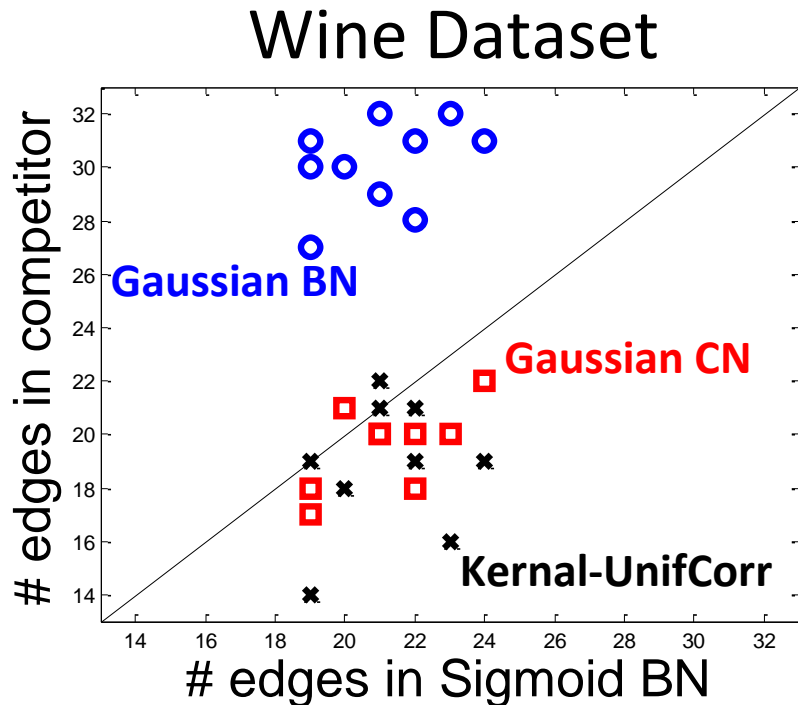
- ✓ Copula networks dominate BN models
- ✓ Learn structure in less than ½ hour!

# Complexity of Dependency Structure



- ✓ Better generalization with sparser structures
- ✓ Simple (one parameter) copula resists over-fitting

# Complexity of Dependency Structure



- ✓ Better generalization with sparser structures
- ✓ Simple (one parameter) copula resists over-fitting



# Control Over Marginals

**Caveat:** the valid density defined via

$$\prod_i R_{c_i} (F_i(x_i), \{F_{ik}(\mathbf{pa}_{ik})\})$$

is only a copula for tree structures

➡ generally, the univariate marginals are skewed

**If you are copula person:** this is a disaster  
(easily fix for Gaussian copula as is done for NPBBNs)

**From the UAI perspective:**



and the marginals in practice are quite accurate!

# Expressiveness vs Efficiency

**Common sense in ML:** there is a computational price for additional expressiveness / flexibility

**However:** separation of univariates from dependence can “magically” avoid this:

- Because local copula functions are simple (i.e. one parameter), estimation is efficient despite flexibility
- Perform mean-field like inference faster than standard mean field [Elidan, 2010]
- Significantly faster structure learning using new relationship of  $\rho_S$  and expected likelihood [Elidan, 2012]

# Expressiveness vs Efficiency

**Common sense in ML:** there is a computational price for additional expressiveness / flexibility

**However:** separation of univariates from dependence can “magically” avoid this:

- Because local copula functions are simple (i.e. one parameter), estimation is efficient despite flexibility
- Perform mean-field like inference faster than standard mean field [Elidan, 2010]

■

Can we take further advantage  
of the representation?

# Real-life Processes

## Motivation:

- Relationship between distance and velocity of rocket
- Relationship between volatilities of RVs, e.g. the returns on equity indices (hetero-scedastic sequence)

## Challenges:

- Infinitely many interacting variables  $Z_t$
- Non-Gaussian interaction
- Varied marginal distributions

Wilson and Ghahramani, 2010

See also related work by Jaimungal and Ng, 2009

# Gaussian Processes

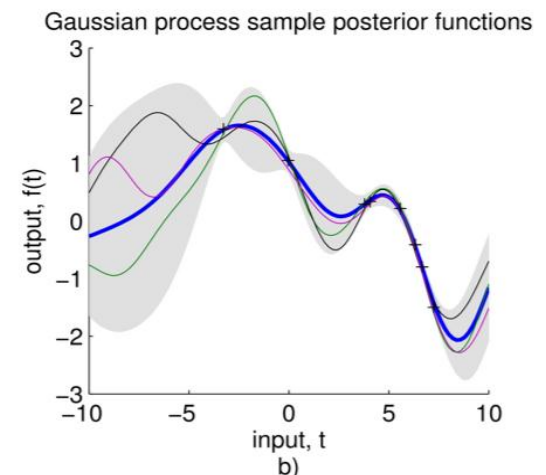
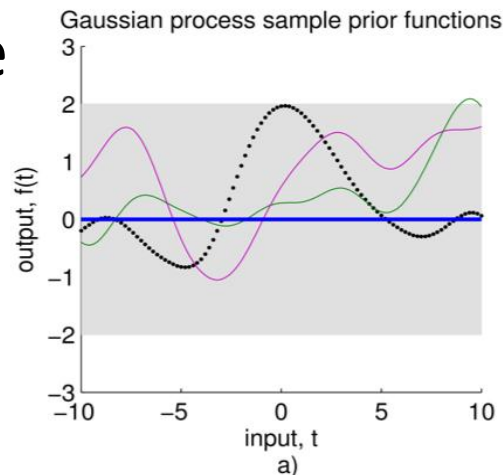
A collection of random variables  $Z_t$ , any finite number of which have a joint Gaussian distribution

**Used to define distribution over functions:**

$$f(z) \sim \mathcal{GP}(m(z), k(z, z'))$$

1. any finite set  $\{f(z_i)\}$  have a joint Gaussian distribution
2.  $m(z_i)$  is the expectation of  $f(Z_i)$
3.  $\Sigma_{ij}=k(z_i, z_j)$  defines the functions properties

Rasmussen and Williams 2006  
for (many) more details



# Copula Processes

Let  $\mu$  be a process measure with marginals  $G_t$  and joint  $H$ .  $Z_t$  is a **copulas process** distributed with base measure  $\mu$  if

$$P\left(\bigcap_{i=1}^n \{G_{t_i}^{-1}(F_{t_i}(Z_{t_i})) \leq a_i\}\right) = H_{t_1, \dots, t_n}(a_1, \dots, a_n)$$

**Example:** Gaussian Copula Process =  $\mu$  is a standard GP

**Another way to think about this:**

There is a mapping  $\Psi$  that transform  $Z_t$  into a GP

$$\Psi(Z_t) \sim \mathcal{GP}(m(t), k(t, t'))$$

# Gaussian Copula Process Volatility

Let  $y_1, \dots, y_n$  be a heteroscedastic sequence (varying  $\sigma_t$ )

**Goal:** model joint of  $\sigma_1, \dots, \sigma_n$  and predict unrealized  $\sigma_t$

1. Observations:  $y(t) \sim \mathcal{N}(0, \sigma^2(t))$  [this can be relaxed]
2. Volatility modeled as a Gaussian Copula Process

$$f(t) = \Psi^{-1}(\sigma(t)) \quad [\text{warping function}]$$

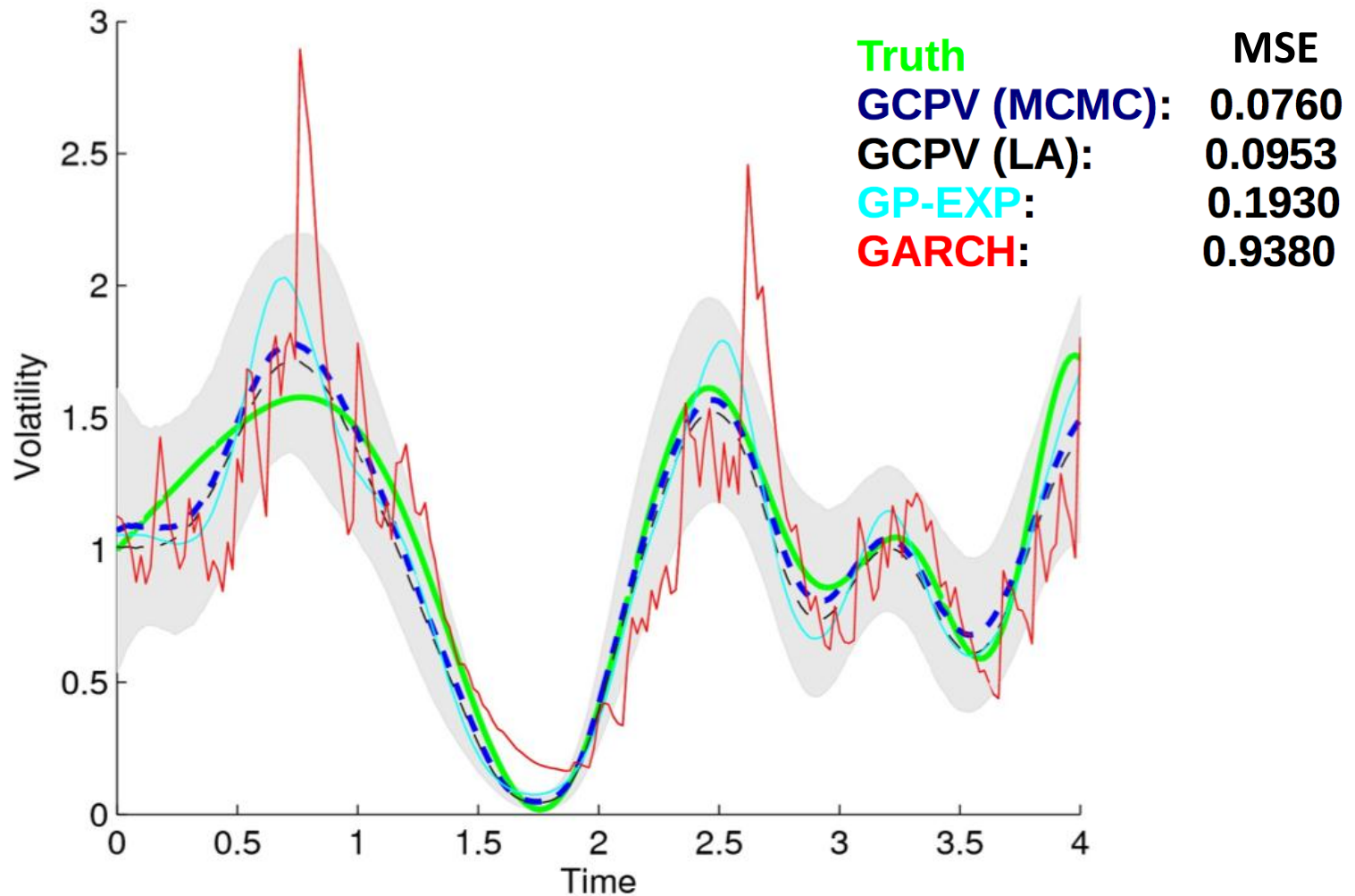
$$f(t) \sim \mathcal{GP}(m(t) = 0, k(t, t'))$$

## Challenges:

- Learn a flexible warping function
- Need to do inference over many latent RVs

Interesting technical solutions in the paper! (no time 😞)

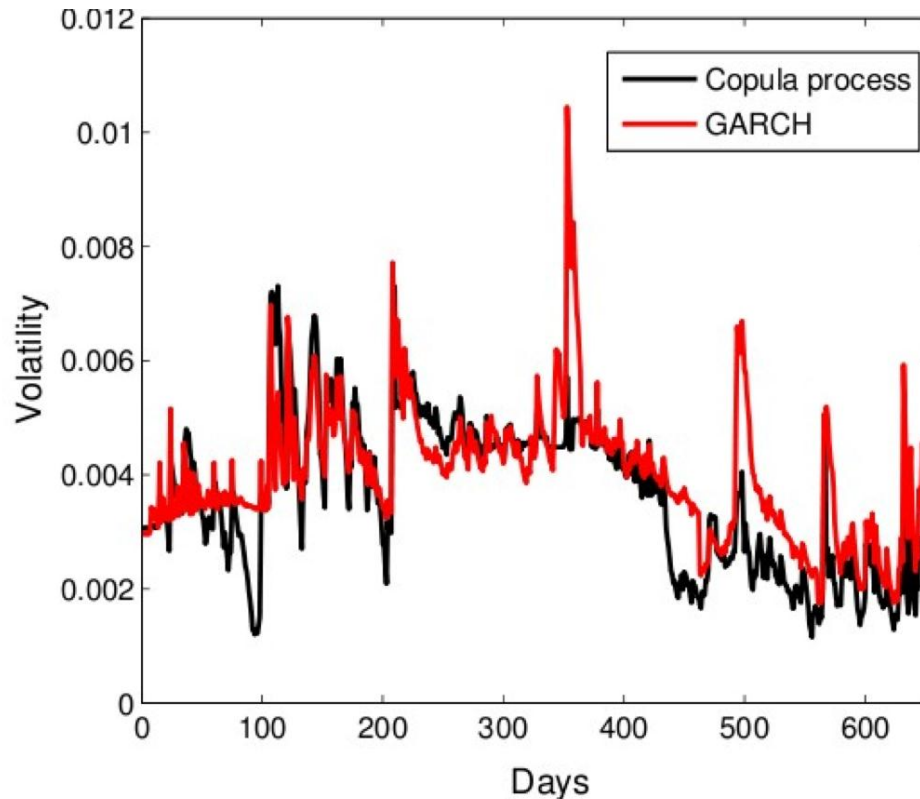
# Simulation Results



Very promising results also for “JUMP” (spike like) sequence

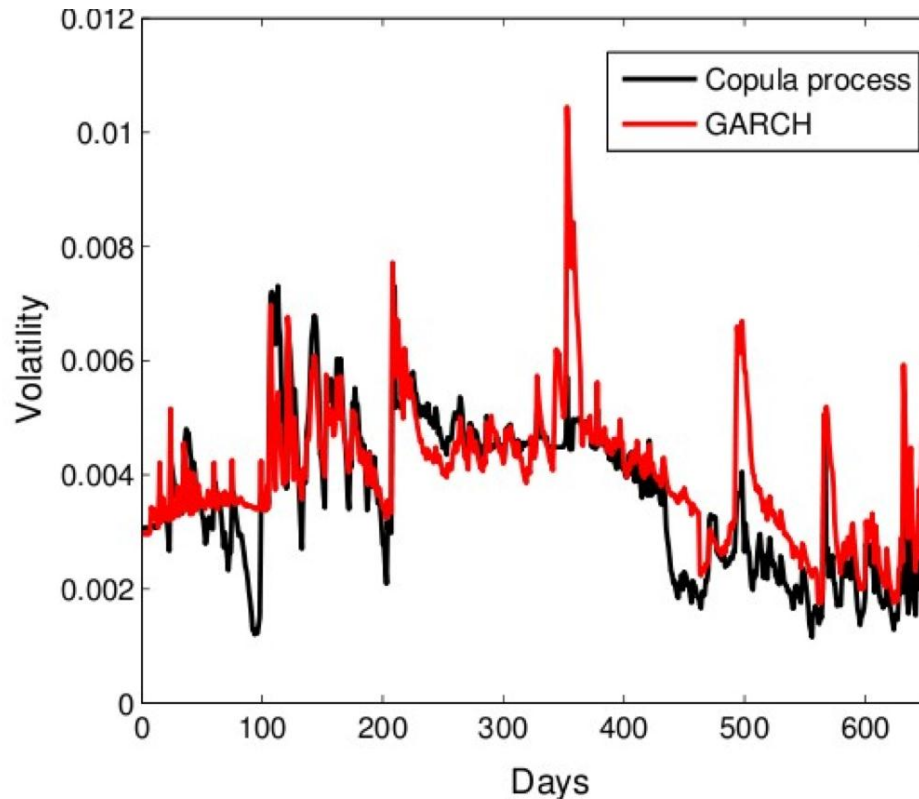


# DM-GBP exchange rate returns



	Model	Historical	1 step	7 step	30 step
$\times 10^{-9}$	GCPV (LA)	2.43	3.00	3.08	3.17
	GCPV (MCMC)	2.39	3.00	3.08	3.17
	GP-EXP	2.52	3.20	3.46	5.14
	GARCH	2.83	3.03	3.12	3.32

# DM-GBP exchange rate returns



Next step: multivariate stochastic predictions  
“Generalised Wishart Processes”, Wilson and Ghahramani 2011

# Summary

Model	Base Copula	# RVs	Structure	Central merit
Vines	any bivariate	<10s	conditional dependence	Well understood general purpose framework
NPBBN	Gaussian in practice	100s	BN+Vines	Mature application to large hybrid domains
Tree-averaged	any bivariate	10s	Markov	Bayesian averaging over structures
Non-paranormal	Gaussian	100-1000s	Markov	Large scale undirected estimation with guarantees
Copula Networks	any multivariate	100s	BN	very flexible at the cost of partial control over marginals
Copula Processes	any multivariate	$\infty$ of few dimensions	-	Arbitrarily many variables

# Part III: Other copula-based works in ML and applications

# Scope

- REGO: Rank-based Estimation of Renyi Information Using Euclidean Graph Optimization [Poczos et al., 2010]
- Copula Mix Model for Dependency-seeking Clustering [Rey and Roth, 2012]

## What will not be covered:

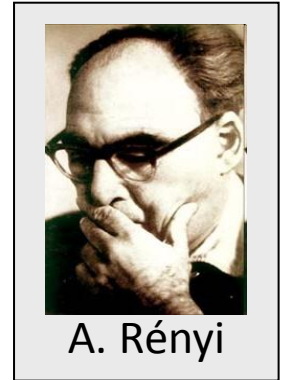
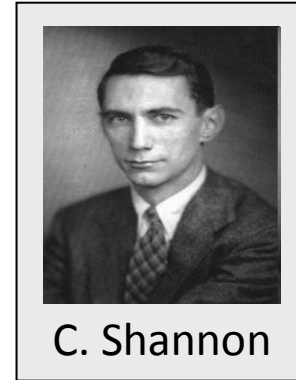
- ICA & ISA Using Schweizer-Wolff [Kirshner and Poczos, 2008]
- Estimation of Renyi Entropy and Mutual Info. Based on Generalized Nearest-Neighbor Graphs [Pal et al., 2010]
- Copula-based Kernel Depend. Measures [Poczos et al. 2012]
- Copula-based applications
- Other related works in computational statistics venues

# Mutual Information

**Goal:** estimate entropy/information

$$I_S(f_{\mathbf{X}}) = - \int f(\mathbf{x}) \log \frac{f(\mathbf{x})}{\prod f_i(\mathbf{x}_i)} d\mathbf{x}$$

$$R_\alpha(f_{\mathbf{X}}) = \frac{1}{1-\alpha} \log \int f^\alpha(\mathbf{x}) \left( \prod_i f_i(\mathbf{x}_i) \right)^{1-\alpha} d\mathbf{x}$$



**Plug-in approach:**

1. Estimate  $f_{\mathbf{X}}(\mathbf{x})$
2. Plug into divergence equation

**Problem:** density estimation is difficult

**Hope:** the density is a nuisance parameter, do we need it?

# Euclidean Entropy Estimation

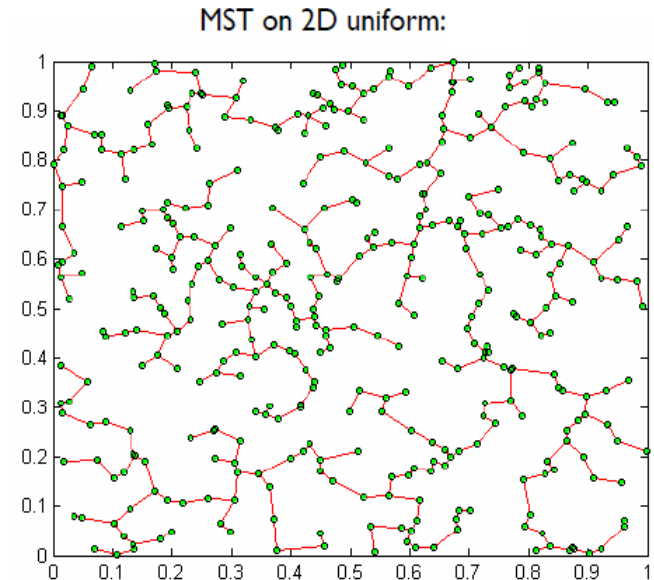
1. A 2D (uniform) graph with samples as nodes
2. Compute length  $L_n(\mathbf{X}^1, \dots, \mathbf{X}^n)$  of MST (TSP, k-NN, ...)

Theorem (Steel 1988 for MST)

$$\frac{1}{1 - \alpha} \log \frac{L_n(\mathbf{X}^1, \dots, \mathbf{X}^n)}{\text{const} \times n^\alpha} \rightarrow H_\alpha(f_{\mathbf{X}})$$

Hero and Michel (1998):

➡ use graph optimization algorithms to estimate entropy



# From Entropy to Mutual Information

Recall from the first part of the tutorial:

$$I(X, Y) = \int \int c(u, v) \log c(u, v) du dv \equiv -H(c(U, V))$$

**Problem:** we don't know  $U=F_X(x), V=F_Y(y)$

**REGO (Poczos et al., 2010):**

1. Transform data into empirical ranks
2. Use Euclidean graph optimization to estimate entropy

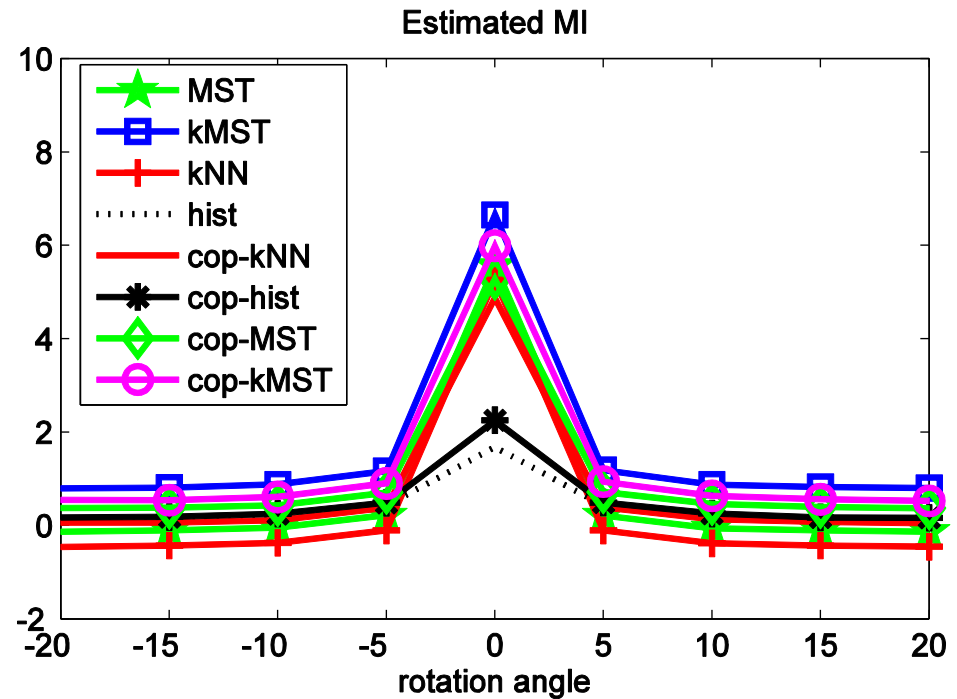
 non-parametric estimator for Renyi information that is provably strongly consistent and robust

See also Pal et al. (2010), Poczos et al. (2012) for follow-ups



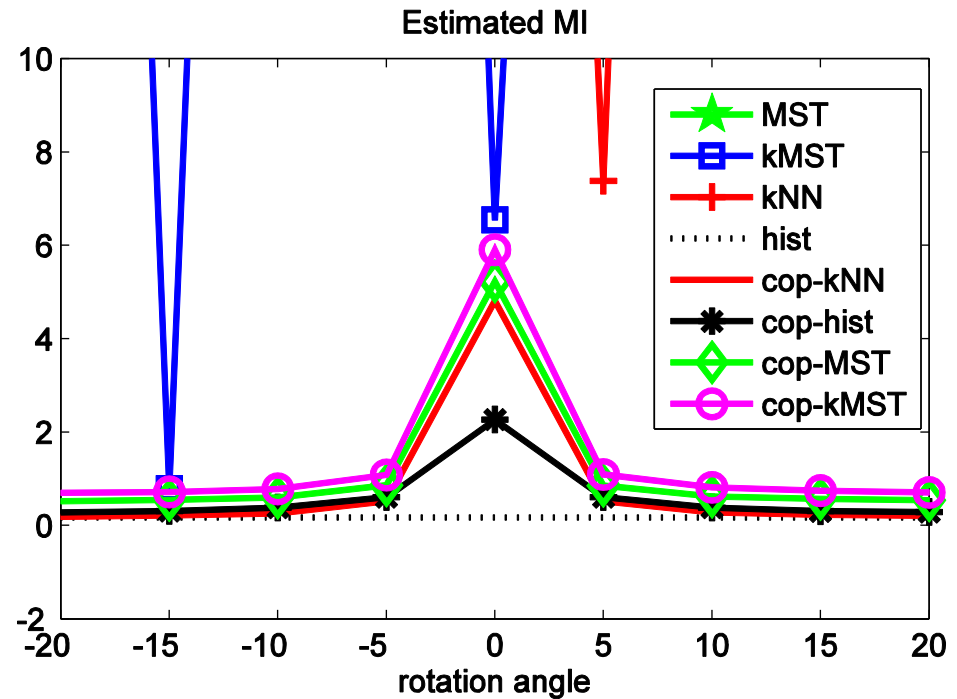
# Example: Image Registration

Task: register image rotated at different angles



# Example: Image Registration

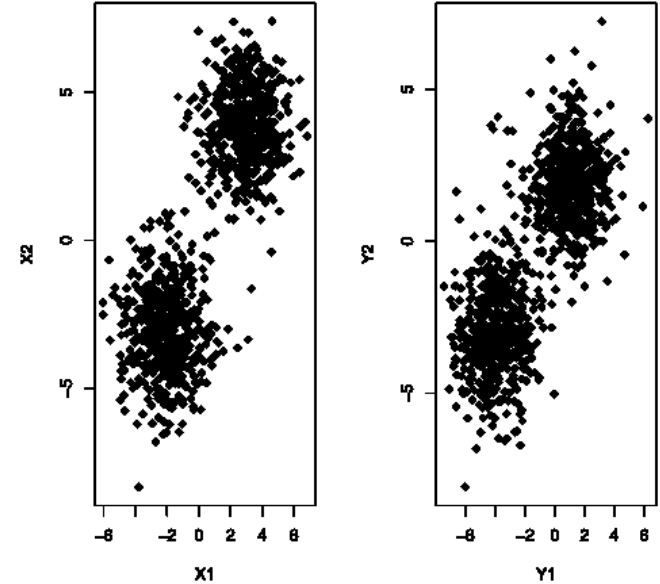
Task: register image rotated at different angles  
(with  $<5\%$  of the pixels corrupted)



# Multiview Dependency Learning

We are given two paired datasets

How are these “views” related?



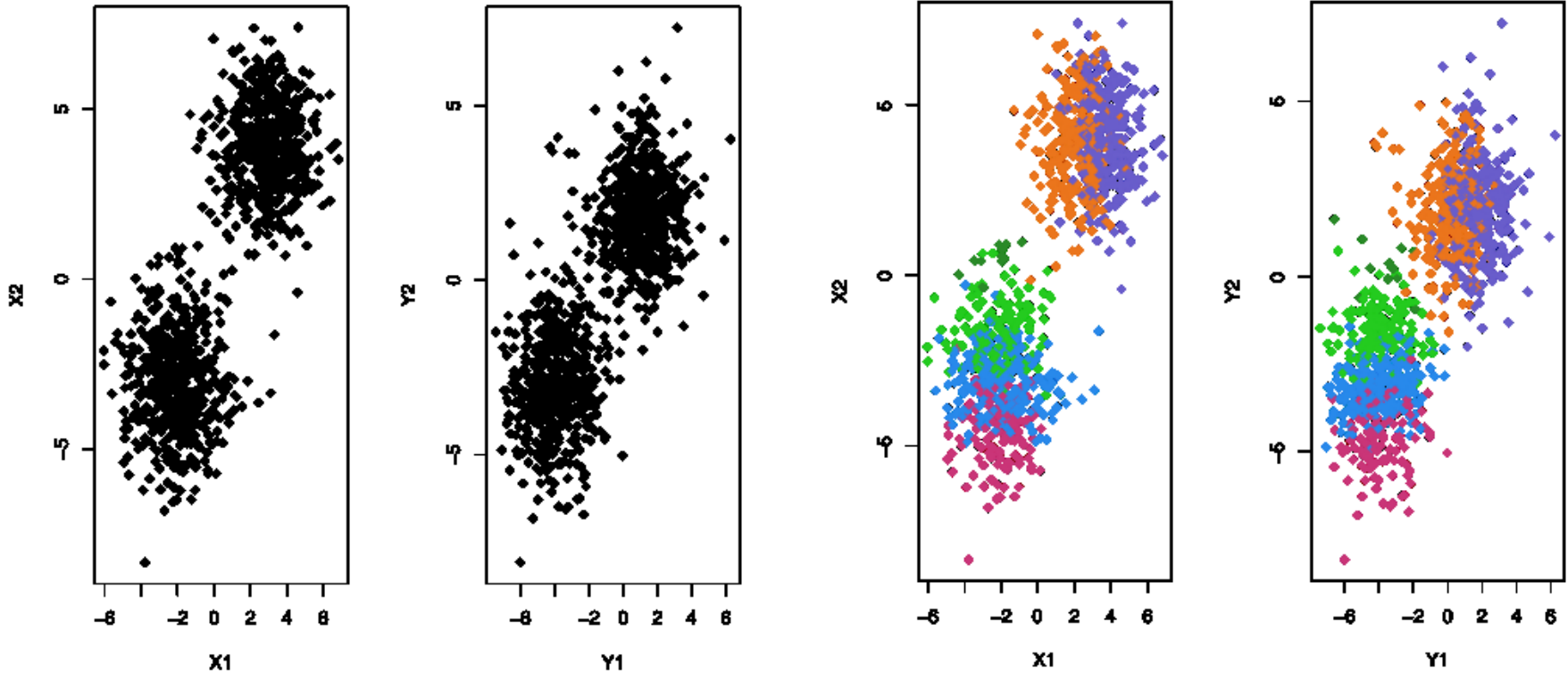
**Canonical Correlation Analysis (CCA):**

linearly project views so as to maximize correlation

➔ weights of the projection are indicative of dimensions that underlie the dependence

# Dependency Seeking Clustering

Idea: dependence may be evident only locally



➔ seek for clusters where dependency “manifests”

# From CCA to Clustering

**Probabilistic interpretation of CCA** (Bach & Jordan, 2005)

$$\begin{aligned}Z &\sim \mathcal{N}_d(0, I_d) \\(X|Z) &\sim \mathcal{N}_p(W_X Z + \mu_X, \Psi_X) \\(Y|Z) &\sim \mathcal{N}_p(W_Y Z + \mu_Y, \Psi_Y)\end{aligned}$$

**Dependency seeking clustering** (Klami & Kaski, 2008):

$$\begin{aligned}Z &\sim CRP(\lambda) \\(X|Z) &\sim \mathcal{N}_p(\mu_X(Z), \Psi_X) \\(Y|Z) &\sim \mathcal{N}_p(\mu_Y(Z), \Psi_Y)\end{aligned}$$

**Problem:** still assumes Gaussian structure within X and Y

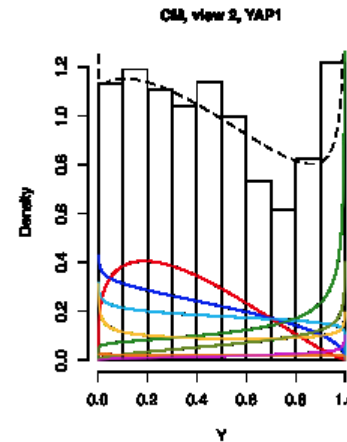
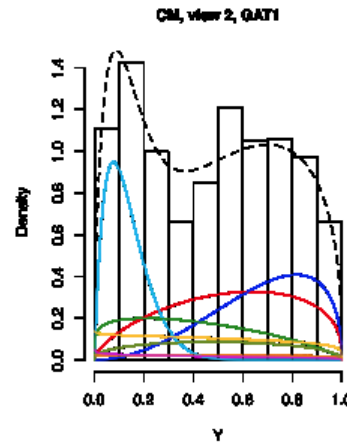
**Idea:** replace Gaussian distribution with a copula

# Yeast under Heat-shock

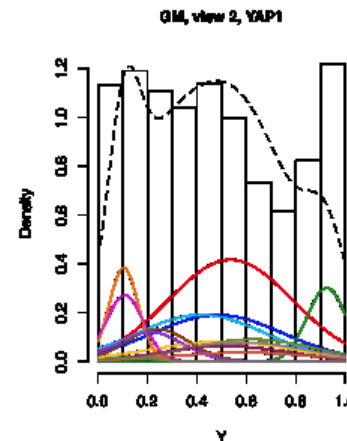
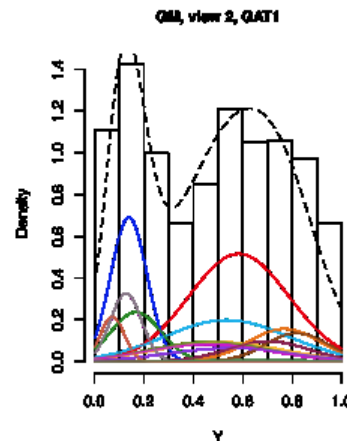
View 1: **Gene expression**

View 2: **Binding affinities**

Copula Mixture:  
8 clusters



Gaussian Mixture:  
14 clusters



# Take Home Messages

- Copulas (like graphical models) are a general framework for multivariate modeling
- Separation between univariate marginals and dependence function provides great flexibility
- Copulas are closely related to dependence concepts
- High-dimensional copula models are in their infancy

**MAKE ML LESS GAUSSIAN**

# Challenges for Grabs

- Effective inference and learning for large-scale copula-based models (we talked about some of these)
- Copula-like constructions for discrete data (see Mayor 2005, 2007)
- Large-scale hybrid (discrete/continuous) models
- What if we wanted to control more than univariate distributions (the compatibility problem)

**MAKE ML LESS GAUSSIAN**