Math 410

—

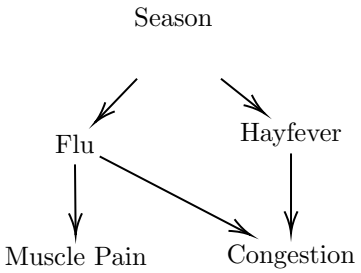
September 2019

Contents

1 Necessary Background

1.1 Brief Introduction to Bayesian Networks

The goal of Bayesian Networks is to represent a joint distribution over a set of random variables using a directed acyclic graph. Let $\mathcal{D} = (V, E)$ be such a graph. The set of nodes V will represent random variables and the set of edges E will correspond to the direct influence between two nodes. Intuitively, edge orientation can be interpreted at a causal relation between two rvs.
When working with Bayesian Networks, nodes (rvs) are independent from other nodes given their immediate parents. This is referred to as the **Markov property**.



By the Markov property, BN above represents the following joint probability:

$$P(s, f, h, m, c) = P(s)P(f|s)P(h|s)P(m|f)P(c|f, h)$$

1.2 Monoid

A set S with a binary operator $\bullet : S \times S \to S$ is a **monoid** if it satisfies the following properties:

- **Associativity:** $\forall a, b, c \in S, (a \bullet b) \bullet c = a \bullet (b \bullet c)$
- **Identity Element** $\exists e \in S \;\; s.t. \; \forall a \in S, e \bullet a = a \bullet e = a$

1.3 Semiring

A **semiring** $(R, \bigvee, \cdot)$ is defined by a set R and two of its operators with the following properties:

- $(R, \bigvee)$ is a commutative monoid with identity 0
- $(R, \cdot)$ is a monoid with identity 1

In the contex of max-linear models, we will be using $\bigvee$ as the max operator, and $\cdot$ as the matrix product.

2 Introduction to Max-Linear Models

2.1 Recursive Structural Equation Model

Given a DAG $\mathcal{D} = (V, E)$ with nodes $V = \{1, \ldots, d\}$ and edges $E = \{(k, i) : i \in V \text{ and } K \in pa(i)\}$, we define a **recursive structural equation model** to be any function

$$X_i = f_i(X_{pa(i)}, Z_i)$$

- with $pa(i)$ denoting the direct parents of node i in $\mathcal{D}$
- f_i as a real valued measurable function
- $\{Z_1, \ldots, Z_d\}$ as independent noise variables[1]

By analyzing the set of equations, $\mathbf{X} = (X_1, \ldots, X_d)$, we can see why this model has its name. $\mathbf{X}$ is recursively generated and, through parentage, the generation of each successive X_i is driven by the structure of the graph.
Let $nd(i)$ denote the set of non-descendants of i then, by construction,

$$X_i \perp\!\!\!\perp X_{nd(i)\setminus pa(i)} | X_{pa(i)} \quad i = 1, \ldots, d$$

which means that the distribution on $\mathbf{X}$ is Markov with respect to $\mathcal{D}$.

2.2 Recursive Max-Linear Model

This model is motivated by its applications to risk analysis and the tendency of risk to propagate through a network. The **recursive max-linear model** $\mathbf{X} = (X_1, \ldots, X_d)$ on a DAG $\mathcal{D}$ is defined as:

$$X_i := \bigvee_{k \in pa(i)} c_{ki} X_k \vee c_{ii} Z_i \quad i = 1, \ldots, d$$

- $\{c_{ki} \forall i \in V \text{and } k \in pa(i) \cup \{i\}\}$ as positive weights
- $\{Z_1, \ldots Z_d\}$ as independent, non negative rvs with positive and infinite support.

[1] The distribution $\mathbf{X}$ is uniquely determined by these noise variables

The weights represent relative quantities that a risk originates with a certain proportion in different ancestors.

2.2.1 Max-Linear Coefficient Matrix

This model can be re-written in the form

$$X_i := \bigvee_{j=1}^{d} b_{ji} Z_j \quad i = 1, \ldots, d$$

$B = (b_{ij})_{d \times d}$ is a matrix with non-negative entries called the **max linear coefficient matrix** of $\mathbf{X}$ and its entries are dubbed **max-linear coefficients**. The construction of the matrix B is fairly intuitive. First, for any two nodes (j, i) we assign a weight to every path $p = [j = k_0 \to k_1 \to \ldots \to k_n = i]$. This weight is equal to the edge product allong the path p multiplied by the noise variable (denoted Z_j) associated with the starting node j. Below is an equation that might assist in visualizing the edge product $d_{ji}(p)$:

$$d_{ji}(p) := c_{k_0,k_0} \cdot c_{k_0,k_1} \cdot \ldots \cdot c_{k_{n-2},k_{n-1}} \cdot c_{k_{n-1},k_n} = c_{k_0,k_0} \prod_{l=0}^{n-1} c_{k_l,k_{l+1}}$$

Let $an(i)$ denote the ancestors of a node, $An(i) = an(i) \cup \{i\}$, and P_{ji} denote all possible paths from node j to node i. Then, the ML coefficients for any $i \in V$ are given by:

$$\begin{aligned} b_{ii} &= c_{ii} \\ b_{ji} &= \bigvee_{p \in P_{ji}} d_{ji} \text{ for } j \in an(i) \\ b_{ji} &= 0 \text{ for } j \in V \setminus An(i) \end{aligned}$$

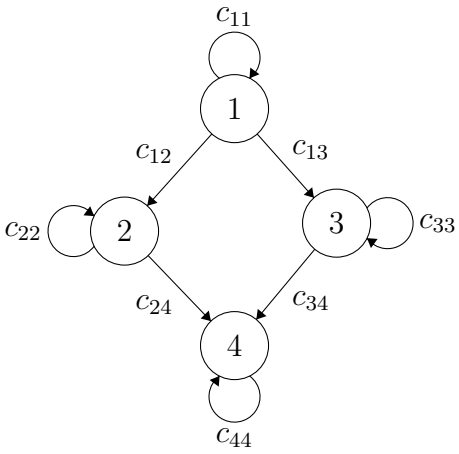
- The coefficient between a node i and itself is simply the weight c_{ii}
- The coefficient between any two nodes (j, i) is given by the path with the largest weight
- The coefficient between any two nodes with no path between them is 0

From our construction, our new model for X_i comes with a slightly new explanation. In *"pseudo stats"*, the model is given by :

$$X_i = \max_{\text{for all nodes } j \in V} \Big(\max(\text{path weight between nodes } i \text{ and } j) * \text{noise associated to } j \Big)$$

2.2.2 Example of Max-Linear Model

Let $\mathcal{D} = (V, E) = (\{1, 2, 3, 4\}, \{(1, 2), (1, 3), (2, 4), (3, 4)\})$, and $C = \{c_{ik} : i \in V, k \in pa(i)\}$. Now, let us build the recursive ML model $\mathbf{X} = (X_1, X_2, X_3, X_4)$



$$
\begin{aligned}
X_1 &= c_{11}Z_1 \\
X_2 &= c_{12}X_1 \vee c_{22}Z_2 = c_{12}c_{11}Z_1 \vee c_{22}Z_2 \\
X_3 &= c_{13}X_1 \vee c_{33}Z_3 = c_{13}c_{11}Z_1 \vee c_{33}Z_3 \\
X_4 &= c_{24}X_2 \vee c_{34}X_3 \vee c_{44}Z_4 \\
&= c_{24}(c_{12}c_{11}Z_1 \vee c_{22}Z_2) \vee c_{34}(c_{13}c_{11}Z_1 \vee c_{33}Z_3) \vee c_{44}Z_4 \\
&= (c_{24}c_{12}c_{11} \vee c_{34}c_{13}c_{11})Z_1 \vee (c_{24}c_{22})Z_2 \vee (c_{34}c_{33})Z_3 \vee c_{44}Z_4
\end{aligned}
$$

Our vector $\mathbf{X}$ is a ML model with ML coefficient matrix

$$
B = \begin{bmatrix}
c_{11} & c_{12}c_{11} & c_{13}c_{11} & c_{24}c_{12}c_{11} \vee c_{34}c_{13}c_{11} \\
0 & c_{22} & 0 & c_{22}c_{24} \\
0 & 0 & c_{33} & c_{33}c_{34} \\
0 & 0 & 0 & c_{44}
\end{bmatrix}
$$

To visualize the generation of this matrix given a graph $\mathcal{D}$ and weights C, please consult r code.

3 Model Distribution

Now that we have the model, our goal is to find the underlying distribution of each of the X_i. To do this, we need a stronger tool than the ones at our disposal.

3.1 Copulas

A **Copula** is a cumulative distribution function with univariate, uniform margins[2]. This term is derived from Latin and refers to a link. In this context, copulas help to determine the dependence between random variables.

Theorem 1 (Sklar's Theorem)**.** Given random variables $X_1, \ldots, X_d$ with cumulative distribution functions $F_1, \ldots, F_d$, there always exists a copula C such that

$$P(X_1 \leq x_1, \ldots, X_d \leq x_d) = C(F_1(x_1), \ldots, F_d(x_d)) \qquad \forall x_1 \ldots, x_d \in \mathbb{R}$$

This means that the joint cumulative distribution of random variables $\{X_i\}_{i=1}^d$ can be represented by a function of the marginal probability distributions.

Theorem 2 (Extension to Sklar's Theorem)**.** If the cumulative distribution functions are continuous, then the copula C is unique and can be retrieved by the joint cdf of the original random variable vector $(X_1, \ldots, X_d)$ after the **integral transform** has been applied to each of the components: $(F_1(X_1), \ldots, F_d(X_d))$.

Proof. In this paper, we show this result holds for the set of continuous and strictly increasing functions $\{F_i\}_{i=1}^d$. The result also holds for monotonic increasing functions but is slightly harder to show. From the restriction on the functions, we know that $\forall x, y \in \mathbb{R}, \forall i \; s.t. \; 1 \leq i \leq d$ we have $x \leq y \implies F_i(x) \leq F_i(y)$. We can now use this to prove our result.

$$\begin{aligned}
C(F_1(x_1), \ldots, F_d(x_d)) &= P(X_1 \leq x_1, \ldots, X_d \leq x_d) \\
&= P(F_1(X_1) \leq F_1(x_1), \ldots, F_d(X_d) \leq F_d(x_d)) \\
&= P(F_1(X_1) \leq u_1, \ldots, F_d(X_d) \leq u_d) \\
&= P(X_1 \leq F_1^{-1}(u_1), \ldots, X_d \leq F_d^{-1}(u_d)) \\
&= C(F_1(F_1^{-1}(u_1)), \ldots, F_d(F_d^{-1}(u_d))) \\
&= C(u_1, \ldots, u_d)
\end{aligned}$$

[2]ref Copula and copula models

From this we note that the copula can be set to the joint cdf of the random vector obtained by element wise integral transform. □

3.2 Copulas and ML-Models

Let us assume that we have data that consists of n observations from d sources denoted $X_1, \ldots, X_d$. If we want to use a ML model to learn more about the underlying structure and relationship between these sources, we make use of copulas. To calculate the copulas, we must calculate the following:

1. The marginal cdfs $\{F_1, \ldots, F_d\}$
2. The joint cdf $H(x_1, \ldots, x_d) = P(X_1 \leq x_1, \ldots, X_d \leq x_d)$
3. The copula $C(u_1, \ldots, c_d) = H(F_1^{-1}(u_1), \ldots, F_d^{-1}(u_d))$

3.2.1 Marginal cdfs (theory)

Assume the random variables $X_1, \ldots, X_d$ follow a ML model with i.i.d shock random variables $Z_1, \ldots, Z_d$ and ML coefficient matrix

$$B = \begin{bmatrix} b_{11} & b_{12} & \ldots & b_{1d-1} & b_{1d} \\ b_{21} & b_{22} & \ldots & b_{2d-1} & b_{2d} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \ldots & \ddots & \vdots \\ b_{d1} & b_{d2} & \ldots & b_{d-1} & b_d \end{bmatrix}$$

Where b_{ij} is the maximum weight associated with any path from i to j. We can calculate the marginal cdf for each of the X_k:

$$\begin{aligned} F_{X_k}(x_k) &= P(X_k \leq x_k) \\ &= P(\bigvee_{j=1}^{d} b_{jk} Z_j \leq x_k) \\ &= \prod_{j=1}^{d} P(Z_j \leq \frac{x_k}{b_{jk}}) \\ &= \prod_{j=1}^{d} F_{Z_j}(\frac{x_k}{b_{jk}}) \end{aligned}$$

Now, if we assume that $Z_i \overset{iid}{\sim}$ Fréchet (α) we get the following result

$$\begin{aligned} F_k(x_k) &= \prod_{j=1}^{d} \exp\left\{-\left(\frac{x_k}{b_{jk}}\right)^{-\alpha}\right\} \\ &= \exp\left\{-\sum_{j=1}^{d}\left(\frac{x_k}{b_{jk}}\right)^{-\alpha}\right\} \\ &= \exp\left\{-x_k^{-\alpha}\sum_{j=1}^{d} {b_{jk}}^{\alpha}\right\} \\ \implies F_k^{-1}(u) &= \left(\frac{\sum_{j=1}^{d} b_{jk}^{\alpha}}{-\log(u)}\right)^{1/\alpha} \end{aligned}$$

It is important to note that if $j \notin An(k)$ then $b_{jk} = 0$. This means that the sum $\sum_{j=1}^{d} b_{jk}$ is equivalent to $\sum_{j \in An(k)} b_{jk}$.

3.2.2 Marginal cdfs (from data)

From the data, we have

$$\{(X_{11}, \ldots, X_{1d}), \ldots, (X_{n1}, \ldots X_{nd})\}$$

To build a copula, we need the points

$$\{(F_1(X_{11}), \ldots, F_d(X_{1d})), \ldots, (F_1(X_{n1}), \ldots F_d(X_{nd}))\}$$

To represent a sample from C, because we do not have the analytic values for $\{F_1, \ldots, F_d\}$, we estimate them using the following formula:

$$\widehat{F_i}(x) = \frac{1}{n}\sum_{j=1}^{n} \mathbb{1}(X_{ij} \leq x)$$

This is an application of the **rank plot** method. Essentially, for each random variable X_i, we rank all observations from smallest to greatest. Then, the rank (position in observation vector) of each observation helps us build our cdf F_i. The marginal cdf of each random variable evaluated at x is calculated by dividing the number of observations greater than x

by the total number of observations. In practice, we use

$$\widehat{F}_i(x) = \frac{1}{n+1} \sum_{j=1}^{n} \mathbb{1}(X_{ij} \leq x)$$

because it leads to nicer results.

3.2.3 Joint cdf

In general, the joint cdf H of random variables following a max-linear model can be calculated in the following way:

$$\begin{aligned} H(x_1, \ldots, x_d) &= P(X_1 \leq x_1, \ldots, X_d \leq x_d) \\ &= P(\bigvee_{j=1}^{d} b_{j1} Z_j \leq x_1, \ldots, \bigvee_{j=1}^{d} b_{jd} Z_j \leq x_d) \\ &= P(Z_1 \leq \bigwedge_{k=1}^{d} \frac{x_k}{b_{1k}}, \ldots, Z_d \leq \bigwedge_{k=1}^{d} \frac{x_k}{b_{dk}}) \\ &= \prod_{i=1}^{d} F_i(\bigwedge_{k=1}^{d} \frac{x_k}{b_{ik}}) \end{aligned}$$

When we deal with Frechét shock, this simplifies.

$$\begin{aligned} F_i(\bigwedge_{k=1}^{d} \frac{x_k}{b_{ik}}) &= \exp\left\{ -\left(\bigwedge_{k=1}^{d} \frac{x_k}{b_{ik}} \right)^{-\alpha} \right\} \\ &= \exp\left\{ -\bigvee_{k=1}^{d} \left(\frac{b_{ik}}{x_k} \right)^{\alpha} \right\} \\ \implies H(x_1, \ldots, x_d) &= \prod_{i=1}^{d} \exp\left\{ -\bigvee_{k=1}^{d} \left(\frac{b_{ik}}{x_k} \right)^{\alpha} \right\} \\ &= \exp\left\{ -\sum_{i=1}^{d} \bigvee_{k=1}^{d} \left(\frac{b_{ik}}{x_k} \right)^{\alpha} \right\} \\ &= \exp\left\{ -\sum_{i=1}^{d} \bigvee_{k=1}^{d} \left(\frac{b_{ik}}{x_k} \right)^{\alpha} \right\} \end{aligned}$$

Again we note that the index of the max operator can be changed to be in terms of ancestors. This is because for any two random variables i, k, if $i \notin An(k)$, then $b_{ik} = 0$. So we can write

$$H(x_1, \ldots, x_d) = \exp\left\{ -\sum_{k=1}^{d} \bigvee_{i \in An(k)} \left(\frac{b_{ik}}{x_k}\right)^{\alpha} \right\}$$

3.2.4 Building the Copula

Now, we can put it all together to find the copula.

$$C(u_1, \ldots, u_d) = H(F_1^{-1}(u_1), \ldots, F_d^{-1}(u_d))$$

When dealing with Frechét shock, the calculation becomes:

$$\begin{aligned}
a_k &= \sum_{j=1}^{d} b_{jk}^{\alpha} \\
F_k^{-1}(u) &= \left(\frac{\sum_{j=1}^{d} b_{jk}^{\alpha}}{-\log(u)}\right)^{1/\alpha} = \left(\frac{a_k}{-\log(u)}\right)^{1/\alpha} \\
H(x_1, \ldots, x_d) &= \exp\left\{ -\sum_{i=1}^{d} \bigvee_{k=1}^{d} \left(\frac{b_{ik}}{x_k}\right)^{\alpha} \right\} \\
C(u_1, \ldots, u_d) &= \exp\left\{ -\sum_{i=1}^{d} \bigvee_{k=1}^{d} \frac{b_{ik}^{\alpha}}{a_k}(-\log(u_k)) \right\} \\
&= \prod_{i=1}^{d} \exp\left\{ -\bigvee_{k=1}^{d} \frac{b_{ik}^{\alpha}}{a_k}(-\log(u_k)) \right\} \\
&= \prod_{i=1}^{d} \exp\left\{ -\bigvee_{k=1}^{d} -\log(u_k^{\frac{b_{ik}^{\alpha}}{a_k}}) \right\} \\
&= \prod_{i=1}^{d} \exp\left\{ \bigwedge_{k=1}^{d} \log(u_k^{\frac{b_{ik}^{\alpha}}{a_k}}) \right\} \\
&= \prod_{i=1}^{d} \bigwedge_{k=1}^{d} u_k^{\frac{b_{ik}^{\alpha}}{a_k}}
\end{aligned}$$

This is the copula for a ML model with Frechét shock. Note that using the same indexing trick that we have applied before, we can rewrite this as

$$C(u_1, \ldots, u_d) = \prod_{k=1}^{d} 1 \wedge \left\{ \bigwedge_{i \in An(k)} u_k^{\frac{b_{ik}^{\alpha}}{a_k}} \right\}$$

4 Extreme Value Copulas

Definition 1 (Extreme Value Copula)**.** [3] A bivariate copula C is termed an **Extreme Value Copula** if, $\forall\ t > 0, u, v \in [0,1]$, $C(u^t, v^t) = C(u,v)^t$. A finite measure δ acting on the unit simplex (triangle/set of points that add up to one) uniquely characterizes such copulas. This measure also defines a **dependence function** A which (in the bivariate case) also uniquely characterizes an EVC. In fact, a copula C is an EVC iff

$$C(u,v) = (uv)^{A\left(\frac{\log u}{\log uv}\right)}, \qquad u, v \in [0,1] \tag{1}$$

For the rest of this section, unless specifically noted, we will be exclusively working with bivariate EVCs.

4.1 Spectral Measure

First, lets define the unit simplex: $S_2 = \{(u,v) \in [0,1]^2 : u + v = 1\}$. Then, the EVC's measure δ is a finite measure on S_2 called the **spectral measure** satisfies the condition:

$$\int_{S_2} u d\delta(u,v) = \int_{S_2} v d\delta(u,v) = 1$$

From this, it is simple to see that $\delta(S_2) = 2$.

$$\begin{aligned}
\delta(S_2) &= \int_{S_2} d\delta(u,v) \\
&= \int_{S_2} (u+v) d\delta(u,v) \\
&= \int_{S_2} u d\delta(u,v) + \int_{S_2} v d\delta(u,v) = 2
\end{aligned}$$

[3] def from Bivariate EVC paper

Now, let us consider another measure on this set δ^* such that $\delta^* = \frac{\delta}{2}$. We note that this transformation of δ is a probability measure with the unit simplex as support. In fact,

$$(X, Y) \sim \delta^* \implies \mathbb{E}[X] = \mathbb{E}[Y] = \frac{1}{2}$$

4.2 Pickands Dependence Function

From the copula measure, we define the **dependence function** (or Pickands dependence function) as

$$A(t) = 2 * \mathbb{E}[(tX \vee ((1-t)Y)]$$

Where $X, Y \sim \delta^*$. A also has the following properties:

- $A : [0, 1] \to [1/2, 1]$
- A is continuous and convex
- $A(0) = A(1) = 1$
- $\max\{t, 1-t\} \leq A(t) \leq 1$ for $t \in [0, 1]$

4.3 Reparameterization of EVCs

Since we are working on the unit simplex in two dimensions, we can use a mapping from the real number line to the unit simplex to simplify our calculations. This is done using the bijection ϕ (defined below)

$$\begin{aligned} \phi : [0, 1] &\to S_2 \\ x &\to (x, 1-x) \end{aligned}$$

With this bijection, we can project the unit simplex on the real number line. This means that we can now redefine our probability measure δ^* as a univariate distribution such that

$$X \sim \delta^* \implies \mathbb{E}[X] = \mathbb{E}[1-X] = \frac{1}{2}$$

Similarly, we can redefine our dependence function as

$$A(t) = 2 * \mathbb{E}[(tX \vee ((1-t)(1-X))]$$

4.4 Reparameterized Dependence Function

Lets say our finite measure δ^* defines a discrete random variable X with d atoms $\{a_1, \ldots, a_d\}$ each having a probability associated to it $\{p_1, \ldots, p_d\}$. Then, to calculate our copula, we must first calculate the dependence function:

$$\begin{aligned} A(t) &= 2 * \mathbb{E}[(tX \vee (1-t)(1-X))] \\ &= 2 * \sum_{i=1}^{d} (ta_i \vee (1-t)(1-a_i))p_i \end{aligned}$$

Now, we can use this to explicitly determine the form of the dependence function. Assume WLOG that the atoms are ordered such that $i < j \implies a_i < a_j \forall\ 1 < i, j < d$. Then, $A(t)$ can be written as

$$A(t) = \begin{cases} 2(1-t)\sum_{i=1}^{d}(1-a_i)p_i & 0 \le t \le 1 - a_d \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots & \\ 2\left\{t\sum_{j=k}^{d}(a_j p_j) + (1-t)\sum_{i=1}^{k-1}(1-a_i)p_i\right\} & 1 - a_k \le t \le 1 - a_{k-1} \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots & \\ 2t\sum_{i=1}^{d} a_i p_i & 1 - a_1 \le t \le 1 \end{cases}$$

It is important to note that we have continuity at each segment. We can generalize this formula. The above equation assumes that we do not have any atoms at 0 or 1. Let us maintain the ordering on the atoms but assume that we an atom at 0 and an atom at 1(i.e. $a_1 = 0$ and $a_d = 1$). Then, the general form of the dependence function becomes

$$A(t) = 2(1-t)p_1 + 2tp_d + 2\sum_{i=2}^{d-1}(ta_i \vee (1-t)(1-a_i))p_i$$

The piecewise description of the function now becomes

$$A(t) = \begin{cases} 2\left\{(1-t)p_1 + tp_d + (1-t)\sum_{i=2}^{d-1}(1-a_i)p_i\right\} & 0 \le t \le 1 - a_{d-1} \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots & \\ 2\left\{(1-t)p_1 + tp_d + t\sum_{j=k}^{d-1}(a_j p_j) + (1-t)\sum_{i=2}^{k-1}(1-a_i)p_i\right\} & 1 - a_k \le t \le 1 - a_{k-1} \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots & \\ 2\left\{(1-t)p_1 + tp_d + t\sum_{i=2}^{d-1} a_i p_i\right\} & 1 - a_2 \le t \le 1 \end{cases}$$

4.5 Analysis of $A(t)$

Given the piecewise dependence function $A(t)$, we may want to recover the atoms $\{a_1, \ldots, a_d\}$ and their associated probabilities $\{p_1, \ldots, p_d\}$. To do this, we use the continuity of the function at the "kinks" $\{1-a_i\}_{i=2}^{d-1}$. For any interval, $1-a_k \leq t \leq 1-a_{k-1}$, we can evaluate $A(t)$ at the endpoints and consider the difference $A(1-a_k) - A(1-a_{k-1})$.

$$A(1-a_k) = 2\Big\{a_k p_1 + (1-a_k)p_d + (1-a_k)\sum_{j=k}^{d-1}(a_j p_j) + a_k \sum_{i=2}^{k-1}(1-a_i)p_i\Big\} \qquad (*)$$

$$A(1-a_{k-1}) = 2\Big\{a_{k-1} p_1 + (1-a_{k-1})p_d + (1-a_{k-1})\sum_{j=k}^{d-1}(a_j p_j) + a_{k-1} \sum_{i=2}^{k-1}(1-a_i)p_i\Big\} \qquad (**)$$

$$\begin{aligned}
\frac{(*)-(**)}{(a_{k-1}-a_k)} &= 2\Big\{ -p_1 + p_d + \sum_{j=k}^{d-1} a_j p_j - \sum_{j=2}^{k-1} p_j + \sum_{j=2}^{k-1} a_j p_j \Big\} \\
&= 2\Big\{ \sum_{j=1}^{d} a_j p_j - \sum_{j=1}^{k-1} p_j \Big\} = 2\Big\{ \mathbb{E}[X] - \sum_{j=1}^{k-1} p_j \Big\} \\
&= 2\Big\{ \frac{1}{2} - \sum_{j=1}^{k-1} p_j \Big\}
\end{aligned}$$

From this, we can recover the probabilities recursively. It is important to note that the left hand side is proportional (up to a factor of -1) to the slope of $A(t)$ on the interval $[1-a_k, 1-a_{k-1}]$ and is therefore similarly proportional to the derivative at every point in this interval.

4.6 Representation of Bivariate EVCs

Now, the first step to using (1) to calculate the copula is to calculate A:

$$\begin{aligned}
\bar{a}_i &= 1 - a_i \qquad \forall 1 \leq i \leq d \\
A\Big(\frac{\log u}{\log uv}\Big) &= 2 * \sum_{i=1}^{d} \left(\frac{\log u}{\log uv} a_i \vee (1 - \frac{\log u}{\log uv})\bar{a}_i \right) p_i \\
&= 2 * \sum_{i=1}^{d} \left(\frac{\log u}{\log uv} a_i \vee \frac{\log v}{\log uv} \bar{a}_i \right) p_i \\
&= 2 * \frac{1}{\log(uv)} \sum_{i=1}^{d} (a_i \log u \wedge \bar{a}_i \log v) p_i
\end{aligned}$$

Now we use this to calculate the copula:

$$\begin{aligned}
C(u,v) &= (uv)^{A\left(\frac{\log u}{\log uv}\right)} \\
\implies C(u,v) &= (uv)^{2*\frac{1}{\log(uv)}\sum_{i=1}^{d}(a_i \log u \wedge \bar{a}_i \log v)p_i} \\
&= \exp\{2 * \sum_{i=1}^{d}(a_i \log u \wedge \bar{a}_i \log v)p_i\} \\
&= \prod_{i=1}^{d} \exp\{2 * a_i p_i \log u \wedge 2 * \bar{a}_i p_i \log v\} \\
&= \prod_{i=1}^{d} \left(u^{a_i} \wedge v^{\bar{a}_i}\right)^{2p_i}
\end{aligned}$$

From this, we see that any bivariate EVC copula can be retrieved from the product of other copulas. This follows **Theorem 1** from [4] that states that any copula C can be written as

$$C(u,v) = \prod_{i=1}^{d} C_{a_i,\bar{a}_i}^{p_i}$$

where $C_{a_i,\bar{a}_i}$ is a copula of the form $C_{s,t}(u,v) = \min(u^s, v^t)$ and $\sum_{i=1}^{d} p_i = 1$. Because the spectral measure used above differs from the paper's [1] measure by a factor of $\frac{1}{2}$ we find that the sum of our exponent is 2 rather than 1.

[4] Insert paper name here

5 Bivariate EVCs and Max-Linear Models

Our goal is to compare the copula that arises from the max-linear model to bivariate EVCs. For this, we will alter slightly alter the copula that arises from the ML model. We set the variables $u_1 = \ldots = u_d = 1$ except for two variables denoted u_{k_1}, u_{k_2}. The goal of this is to marginalise the copula so we only consider the joint distribution of two nodes at a time. The resulting copula is then

$$C(u_{k_1}, u_{k_2}) = \prod_{j=1}^{d} u_{k_1}^{\frac{b^{\alpha}_{jk_1}}{a_{k_1}}} \wedge u_{k_2}^{\frac{b^{\alpha}_{jk_2}}{a_{k_2}}}$$

We note that $j \notin An(k_1) \cup An(k_2)$ will contribute a factor of 1 to the product. So, the copula can the the following form:

$$C(u_{k_1}, u_{k_2}) = \prod_{j=1, j \in An(k_1) \cup An(k_2)}^{d} u_{k_1}^{\frac{b^{\alpha}_{jk_1}}{a_{k_1}}} \wedge u_{k_2}^{\frac{b^{\alpha}_{jk_2}}{a_{k_2}}}$$

Let $D_0 = \{l : l \in An(k_1) \cap An(k_2)\}$ with $|D_0| = n$, $D_1 = \{l : l \in An(k_1) \setminus An(k_2)\}$, and $D_2 = \{l : l \in An(k_2) \setminus An(k_1)\}$. The above product can then be written as:

$$C(u_{k_1}, u_{k_2}) = \prod_{j_l \in D_1} u_{k_1}^{\frac{b^{\alpha}_{j_l k_1}}{a_{k_1}}} \prod_{j_l \in D_2} u_{k_2}^{\frac{b^{\alpha}_{j_l k_2}}{a_{k_2}}} \prod_{l=1, j_l \in D_0}^{n} u_{k_1}^{\frac{b^{\alpha}_{j_l k_1}}{a_{k_1}}} \wedge u_{k_2}^{\frac{b^{\alpha}_{j_l k_2}}{a_{k_2}}}$$

Now, let us consider a bivariate EVC with atoms at points $\{q_0, \ldots, q_{n+1}\}$ where $q_0 = 0$ and $q_{n+1} = 1$:

$$C(u_{k_1}, u_{k_2}) = u_{k_1}^{2p_{n+1}} \times u_{k_2}^{2p_0} \times \prod_{l=1}^{n} u_{k_1}^{2q_l p_l} \wedge u_{k_2}^{2(1-q_l)p_l}$$

From this, we can calculate the location of the atoms as well as their associated probabilities. First we start with the set of common ancestors D_0:

$$\prod_{l=1, j_l \in D_0}^{n} u_{k_1}^{\frac{b^{\alpha}_{j_l k_1}}{a_{k_1}}} \wedge u_{k_2}^{\frac{b^{\alpha}_{j_l k_2}}{a_{k_2}}} = \prod_{l=1}^{n} u_{k_1}^{2q_l p_l} \wedge u_{k_2}^{2(1-q_l)p_l}$$

$$\implies p_l = \frac{b^{\alpha}_{j_l k_1}}{2a_{k_1}} + \frac{b^{\alpha}_{j_l k_2}}{2a_{k_2}}$$

$$\implies q_l = \frac{\frac{b^{\alpha}_{j_l k_1}}{a_{k_1}}}{\frac{b^{\alpha}_{j_l k_1}}{a_{k_1}} + \frac{b^{\alpha}_{j_l k_2}}{a_{k_2}}}$$

Second, we consider the ancestor set D_1:

$$\prod_{j_l \in D_1} u_{k_1}^{\frac{b^{\alpha}_{j_l k_1}}{a_{k_1}}} = u_{k_1}^{2p_{n+1}}$$
$$\implies p_{n+1} = \sum_{j_l \in D_1} \frac{b^{\alpha}_{j_l k_1}}{2a_{k_1}}$$

Note that if $p_{n+1} = 0$, then $An(k_1) \subseteq An(k_2)$. We have a similar result when we consider the set D_2:

$$\prod_{j_l \in D_2} u_{k_2}^{\frac{b^{\alpha}_{j_l k_2}}{a_{k_2}}} = u_{k_2}^{2p_0}$$
$$\implies p_0 = \sum_{j_l \in D_2} \frac{b^{\alpha}_{j_l k_2}}{2a_{k_2}}$$

Where we note that if $p_0 = 0$, then $An(k_2) \subseteq An(k_1)$.

5.1 Calculating $A(t)$ From B

Now, given a max linear coefficient matrix b, we can calculate each of the probabilities and the associated atoms. This method is explored at length in the code section of this report.

5.2 Calculating B from $A(t)$

Now, given the Pickands dependence function between two nodes of our ML model, we may want to generate a matrix C whose entries are proportional to the entries of B. We define the matrix

$$C = \begin{bmatrix} c_{11} & c_{12} & \dots & c_{1d-1} & c_{1d} \\ c_{21} & c_{22} & \dots & c_{2d-1} & c_{2d} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \dots & \ddots & \vdots \\ c_{d1} & c_{d2} & \dots & c_{dd-1} & c_{dd} \end{bmatrix}$$

Where

$$c_{ij} = \frac{b^{\alpha}_{ij}}{\sum_{k=1}^{d} b^{\alpha}_{kj}}$$

To retrieve this matrix from $A(t)$, we first define the Pickands dependence function between nodes X_{i_1} and X_{i_2} as $A_{i_1 i_2}(t)$ and assume it is given to us as the atom locations $\{q_0, ..., q_{n+1}\}$

$(q_0 = 0, q_{n+1} = 1)$ and their associated probabilities $\{p_0, ..., p_{n+1}\}$. From section 4.5, we have seen how to extract these given a particular dependence and its kinks. How can we use this to extract the C? In general, we can start by considering only the atoms in $(0, 1)$. Using these atoms, we can recover the values $c_{j_l i_1}$ and $c_{j_l i_2}$ for any j_l the set of common ancestors $(i.e. j_l \in An(i_1) \cap An(i_2))$. For any such j_l we can find the corresponding C entries using:

$$c_{j_l i_1} = 2q_l p_l$$
$$c_{j_l i_2} = p_l - q_l p_l$$

The issue is, these are recovered without preserving any particular ordering on the original ancestral set. First we consider a few edge cases:

- If $p_0 = 0$, then we know that $An(i_2) \subseteq An(i_1)$. This means that

Consider base case 2 lone nodes then, you are golden because p_0 and p_{n+1} will give us the result we want.