Machine learning Apr 15, 2022

Lecture 7: Tree distribution

Lecturer: Haim Permuter *Scribe: Avital Yarden and Maman Gil*

This lecture discusses tree distribution and the method of types. We will introduce a method developed by Chow and Liu [1] to fit the best tree to the data, wherein the "Best tree" refers to the tree that have the minimum divergence. We will use some principles that originated in the field of the method of types, in order to show that minimum divergence is equivalent to maximum likelihood. The method of types introduced here was fully developed by Csiszar and Korner [2], who derived the main theorems of information theory from this perspective.

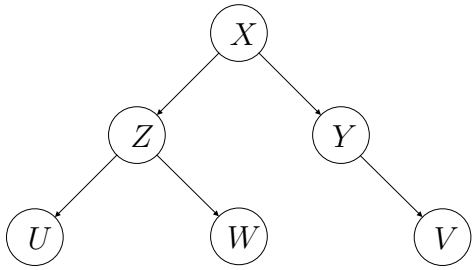

Fig. 1. A tree with distributions of the structure $P_t(x, y, z, u, v, w) = P(x)P(y|x)P(z|x)P(u|z)P(w|z)P(v|y)$.

I. Defining the problem

We will start by introducing the definition of a tree structure.

Definition 1 (Tree) A tree is an undirected graph with no cycles (loops).

A **tree** with nodes corresponding to random processes defines a conditional independence structure on the variables. Conditioned on any node, the subtrees on its edges are independent. For example, the tree in Figure 1 corresponds to

$$P_t(x, y, z, w) = P(x)P(y|x)P(z|x)P(u|z)P(w|z)P(v|y). \tag{1}$$

Chow and Liu used the tree distribution which have only one father for every node (excluding the root), they assumed that the criteria of the optimal tree is minimum divergence, and they proved that the minimum divergence achieved by maximum mutual information over the air of variables. Note that a tree structure has a much smaller number of parameters (linear in the number of nodes) when compared to the exponentially many parameters needed for a general distribution. We get less complex sparse approximator when we use the tree structure but we also get less accuracy compared to the general distribution.

Assuming we have information represented as follows:

$$\begin{matrix} x_1 & y_1 & z_1 & w_1 \\ x_2 & y_2 & z_2 & w_2 \\ \vdots & \vdots & \vdots & \vdots \\ x_n & y_n & z_n & w_n \end{matrix}$$

then X^n, Y^n, Z^n, W^n are n samples of processes. For example, n can be the number of students and X, Y, Z and W are the students' psychometric grade, first year average, the second year average, and the third year average, respectively. Of course, there are many possible trees that can describe the probability of the data presented, but we are interested in the tree that has the largest probability among all the possible trees. From equation (1):

$$\max_{All\ Trees} P_t(x^n, y^n, z^n, w^n) = \max_{All\ Trees} \prod_{i=1}^{n} P_t(x_i, y_i, z_i, w_i). \tag{2}$$

where the sequences of the samples are i.i.d.

II. Method of types

In order to show that minimum divergence is equivalent to maximum likelihood we will use some principles that originated in the field of the method of types. The method of types evolved from notions of strong typicality. Though some of its ideas were used by Wolfowitz [6] to prove channel capacity theorems, the method was fully developed

by Csiszar and Korner [2], who derived the main theorems of information theory from a method of types perspective.

Let $x^n = (x_1, x_2, ..., x_n)$ be a sequence from the alphabet $\mathcal{X} = (a_1, a_2, a_3, ...a_{|\mathcal{X}|})$. Let $N(a|x^n)$ be the number of times that a appears in sequence x^n.

Definition 2 (Type) The type P_{x^n} (or empirical probability distribution) of a sequence x^n is the relative proportion of occurrences of each symbol of $\mathcal{X}$, i.e., $P_{x^n}(a) = \frac{N(a|x^n)}{n}$ for all $a \in \mathcal{X}$.

Example 1 Let $\mathcal{X} = \{0, 1, 2\}$, let $n = 5$ and $x^5 = (1, 1, 2, 2, 0)$. Then $N(0|x^5) = 1$, $N(1|x^5) = 2$ and $N(2|x^5) = 2$. Hence, $P_{x^n} = \left(\frac{1}{5}, \frac{2}{5}, \frac{2}{5}\right)$.

Definition 3 (All possible types) Let $\mathcal{P}_n$ be the collection of all possible types of sequences of length n.

For example, if $\mathcal{X} = \{0, 1\}$, the set of possible types with denominator n is

$$\mathcal{P}_n = \left\{(P(0), P(1)) : \left(\frac{0}{n}, \frac{n}{n}\right), \left(\frac{1}{n}, \frac{n-1}{n}\right), ..., \left(\frac{n}{n}, \frac{0}{n}\right)\right\}. \tag{3}$$

Lemma 1 An upper bound for $|\mathcal{P}_n|$:

$$|\mathcal{P}_n| \leq (n+1)^{|\mathcal{X}|}. \tag{4}$$

Proof:

There are $|\mathcal{X}|$ components in the vector that specifies P_{x^n}. The numerator in each component can take on only $n + 1$ values. So there are at most $(n + 1)^{|\mathcal{X}|}$ choices for the type vector. ■

Definition 4 (Type class) Let $P \in \mathcal{P}_n$. The set of sequences of length n with type P is called type class of P, denoted $T(P)$:

$$T(P) = \{x^n : P_{x^n} = P\}. \tag{5}$$

Theorem 1 (Probability of a sequence in the type class) If $X \sim Q$ i.i.d., the probability of x^n depends only on the type of x^n, i.e., P_{x^n}

$$Q(x^n) = 2^{-n(H(P_{x^n}) + D(P_{x^n}||Q))}. \tag{6}$$

Proof:

Since $\{X_i\}_{i\geq 1}$ are i.i.d,

$$Q^n(x^n) = \prod_{i=1}^{n} Q(x_i). \tag{7}$$

Now consider

$$\log Q^n(x^n) = \sum_{i=1}^{n} \log Q(x_i) \tag{8}$$

$$\stackrel{(a)}{=} \sum_{a\in\mathcal{X}} N(a|x^n) \log Q(a) \tag{9}$$

$$\stackrel{(b)}{=} n \sum_{a\in\mathcal{X}} P_{x^n}(a) \log Q(a) \tag{10}$$

$$= n \sum_{a\in\mathcal{X}} P_{x^n}(a) \log \frac{Q(a)}{P_{x^n}(a)} \cdot P_{x^n}(a) \tag{11}$$

$$= n(-H(P_{x^n}) - D(P_{x^n}||Q)). \tag{12}$$

where

(a) follows because each $a \in \mathcal{X}$ contributes exactly $\log Q(a)$ times it's number of occurrences in x^n to the sum in (8).

(b) follows from the definition of $P_{x^n}(a)$.

Hence, we obtained

$$Q^n(x^n) = 2^{-n(H(P_{x^n})+D(P_{x^n}||Q))}. \tag{13}$$

In general, the equation can be constructed vectorially as follows:

$$Q^n(x_1^n, x_2^n, \ldots, x_m^n) = 2^{-n(H(P_{x_1^n,x_2^n,\ldots,x_m^n})+D(P_{x_1^n,x_2^n,\ldots,x_m^n}||Q_{X_1,X_2,\ldots,X_m}))}. \tag{14}$$

where $x_1^n, x_2^n, \ldots, x_m^n$ are the random processes. And $Q_{X_1,X_2,\ldots,X_m}$ is the joint distribution function. ■

For more information on method of types, see a whole lecture on the subject: `http://www.ee.bgu.ac.il/~haimp/multi2/lec1/lec1.pdf`

III. TREE DISTRIBUTION

We want to find the tree distribution that have the maximum probability (maximum likelihood principle). We saw in the last section about method of types, that is achieved by minimum divergence. From equation (13):

$$P_t(x_1^n, x_2^n, ..., x_m^n) = 2^{-n(H(P_{emp})+D(P_{emp}||P_t(x_1,x_2,...,x_m)))}, \tag{15}$$

where $P_{emp} = P_{x_1^n,x_2^n,...,x_m^n}(x_1, x_2, ..., x_m)$. Since the empirical entropy $H(P_{emp})$ does not depend on the selected tree but only on samples, then to obtain the maximum probability, we should look for the minimum of the divergence $D(P_{emp}||P_t(x_1, x_2, ..., x_m))$.

$$D\Big(P_{emp}||P_t(x_1, x_2, ..., x_m)\Big) = \sum_{x_1\in\mathcal{X}_1} ... \sum_{x_m\in\mathcal{X}_m} P_{emp} \log \frac{P_{emp}}{P_t(x_1, x_2, ..., x_m)}$$

$$= H(P_{emp}) - \sum_{x_1\in\mathcal{X}_1} ... \sum_{x_m\in\mathcal{X}_m} P_{emp} \log P_t(x_1, ..., x_m). \tag{16}$$

In a tree distribution, it can be said that each node in the tree depends only on its parent and not on any other node in the tree.

$$P_t(x_1, x_2, ..., x_m) = \prod_{i=1}^{m} P_t(x_i|x_{j(i)}), \tag{17}$$

where $x_{j(i)}$ is the parent of x_i, see Fig 1.

The Chow and Liu algorithm of creating a tree based on the assumption that minimum divergence is the criteria of the optimal tree. We will define the next lemma of minimum cross entropy for the following proof which show that the minimum divergence is equivalent to maximum sum of the mutual information.

Lemma 2 (Minimum cross entropy) The minimum by q of the cross entropy between (p, q) is equal to the entropy of p

$$\min_q H(p, q) = H(p), \tag{18}$$

and it is achieved by $q = p$.

Proof:

$$\min_q H(p, q) = \min_q - \sum_x p(x) \log q(x)$$

$$
\begin{aligned}
&= \min_q - \sum_x p(x) \log \frac{q(x)p(x)}{p(x)} \\
&= \min_q \sum_x p(x) \log \frac{p(x)}{q(x)} - \sum_x p(x) \log p(x) \\
&= \min_q D(p||q) + H(p) \\
&\stackrel{(a)}{=} H(p),
\end{aligned}
$$

where (a) follows from the fact that KL divergence $D(p||q)$ is always non-negative and it get zero if and only if $q = p$. ■

Theorem 2 (Chow and Liu results [1]) The minimum divergence achieved by maximum sum of mutual information, and it given by,

$$
\min_{P_t} \Big(D\Big(P_{emp}||P_t(x_1, x_2, ..., x_m)\Big)\Big) = const - \sum_{i=1}^{m} I(X_i, X_{j(i)}),
$$

where the mutual information induced by P_{emp}.

Proof:

$$
D\Big(P_{emp}||P_t(x_1, x_2, ..., x_m)\Big)
$$

$$
\stackrel{(a)}{=} H(P_{emp}) - \sum_{x_1 \in \mathcal{X}_1 .. x_m \in \mathcal{X}_m} P_{emp}(x^m) \log P_t(x_1, ..., x_m)
$$

$$
\stackrel{(b)}{=} H(P_{emp}) - \sum_{x^m} P_{emp}(x^m) \log \prod_{i=1}^{m} P_t(x_i|x_{j(i)})
$$

$$
= H(P_{emp}) - \sum_{x^m} P_{emp}(x^m) \log \prod_{i=1}^{m} \frac{P_t(x_i|x_{j(i)})P_{emp}(x_i)}{P_{emp}(x_i)}
$$

$$
= H(P_{emp}) - \sum_{x^m} P_{emp}(x^m) \sum_{i=1}^{m} \Big(\log \frac{P_t(x_i|x_{j(i)})}{P_{emp}(x_i)} + \log P_{emp}(x_i)\Big)
$$

$$
= H(P_{emp}) - \sum_{x^m} \sum_{i=1}^{m} P_{emp}(x^m) \Big(\log \frac{P_t(x_i|x_{j(i)})}{P_{emp}(x_i)} + \log P_{emp}(x_i)\Big)
$$

$$= H(P_{emp}) - \sum_{i=1}^{m} \sum_{x^m} P_{emp}(x^m) \log \frac{P_t(x_i|x_{j(i)})}{P_{emp}(x_i)} - \sum_{i=1}^{m} \sum_{x^m} P_{emp}(x^m) \log P_{emp}(x_i), \tag{19}$$

$$= H(P_{emp}) - \sum_{i=1}^{m} \sum_{x_i, x_{j(i)}} P_{emp}(x_i, x_{j(i)}) \log \frac{P_t(x_i|x_{j(i)})}{P_{emp}(x_i)} - \sum_{i=1}^{m} \sum_{x_i} P_{emp}(x_i) \log P_{emp}(x_i), \tag{20}$$

$$= H(P_{emp}) - \sum_{i=1}^{m} \sum_{x_{j(i)}} P_{emp}(x_j) \sum_{x_i} P_{emp}(x_i|x_{j(i)}) \log \frac{P_t(x_i|x_{j(i)})}{P_{emp}(x_i)} - \sum_{i=1}^{m} \sum_{x_i} P_{emp}(x_i) \log P_{emp}(x_i), \tag{21}$$

$$\stackrel{(d)}{=} H(P_{emp}) - \sum_{i=1}^{m} \sum_{x_{j(i)}} P_{emp}(x_j) \sum_{x_i} P_{emp}(x_i|x_j) \log \frac{P_{emp}(x_i|x_{j(i)})}{P_{emp}(x_i)} - \sum_{i=1}^{m} \sum_{x_i} P_{emp}(x_i) \log P_{emp}(x_i), \tag{22}$$

$$\stackrel{(c)}{=} Const - \sum_{i=1}^{m} I(X_i; X_{j(i)}), \tag{23}$$

where

- (a) - follows from equation (16)
- (b) - follows from equation (17)
- (c) - Note that $\sum_{x_i} P_{emp}(x_i|x_j) \log \frac{P_t(x_i|x_{j(i)})}{P_{emp}(x_i)}$ is a divergence and the minimum is achieved when $P_t(x_i|x_{j(i)}) = P_{emp}(x_i|x_{j(i)})$
- (d) we define $Const = H(P_{emp}) - \sum_{i=1}^{m} \sum_{x_i} P_{emp}(x_i) \log P_{emp}(x_i)$ which do not depends on the tree structure

■

IV. Maximum spanning tree algorithm

The problem is to create a tree that best describe the data, i.e. the tree that will take the divergence to minimum. In the previous section, we showed that minimizing the divergence is equivalent to maximizing the sum of all the mutual information between

each node and its parent in the tree. Our goal is to find an algorithm to create the tree. Kruskal [4] proposed the following algorithm called "Minimum spanning tree algorithm" to assemble the desired tree. This is a greedy algorithm which doesn't promise that an optimal tree will be found. The Greedy Choice is to pick the smallest weight edge that does not cause a cycle in the MST constructed so far. This is a generic pseudo-code for "Minimum spanning tree algorithm", when V is a set of all the vertices, E is a set of all the possible edges, A is a sub set of E which contain all the edges that included in the tree.

KRUSKAL(V):

$A = \emptyset$

for *each vertex v,u in V:* **do**
  E = calculate weight of edge(u,v);

end

E = SORT-INC(E);

for *each edge in E:* **do**
  if *cycle is not formed:* **then**
    A = A $\cup$ (u, v)
  end
  if *length(A)=length(V)-1:* **then**
    break;
  end

end

In our problem we need to use "Maximum spanning tree algorithm" which means instead of order the wights(mutual information) by increasing order, we need to order the wights by decreasing order. Before starting the routine we must compute all the mutual information and arrange the resulting process pairs in a list from the largest wight to the smallest wight.

Stage 1: Find the pair of nodes with the greatest mutual information in the list.

$$I_{(i,j)} = P(x_i, x_{j(i)}) \log \frac{P(x_i|x_{j(i)})P(x_{j(i)})}{P(x_{j(i)})P(x_i)}, i, j \in [1, 2 \ldots, N] \ , \ i \neq j. \tag{24}$$

Stage 2: Connect the pair of nodes found in stage 1, update the list of mutual

information and return to Stage 1 if the list still contains pair of processes. The update includes the following:

1. Delete the mutual information of the selected pair.

2. Delete all items in the list that represent prohibited connections, i.e., connections that create loops in the graph.

Stage 3: Decide which of the two processes will be at the head of the tree and determine the direction of the arrows. In fact, for either choice, we obtain the same sum of mutual information, and therefore, either choice is possible.

Example 2 Let us assume that we have the random processes $X\ Y\ Z\ W$. The mutual information between each possible pair was calculated. The results are shown in the table below and in Figure 2.

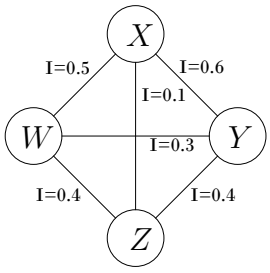

Fig. 2. Calculate the mutual information between each pair of random processes.

pairs	empirical mutual information
X, Y	0.6
X, Z	0.1
X, W	0.5
Y, Z	0.4
Y, W	0.3
Z, W	0.4

We perform step 1 of the routine and see that the greatest mutual information is

between X and Y. Thus, we connect them with a line. We then Proceed to step 2, delete X, Y from the table and return to step 1. See Figure 3.

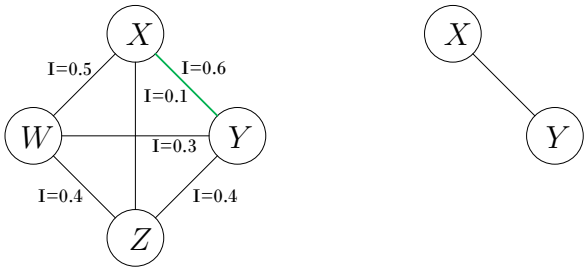

Fig. 3. Connect the two random processes with the largest mutual information, and remove that mutual information from the list.

pairs	empirical mutual information
X, Z	0.1
X, W	0.5
Y, Z	0.4
Y, W	0.3
Z, W	0.4

We again apply Step 1 followed by Step 2 again: in this instance, we delete X, W and we also delete Y, W, the latter pair because it might create a loop in the graph. See Figure 4.

pairs	empirical mutual information
X, Z	0.1
Y, Z	0.4
Z, W	0.4

We re-apply step 1 and find that we have two lines with the same mutual information, so we can arbitrarily choose between the two options.

After step 2, it appears that the table is empty, so we proceed to step 3 and select the

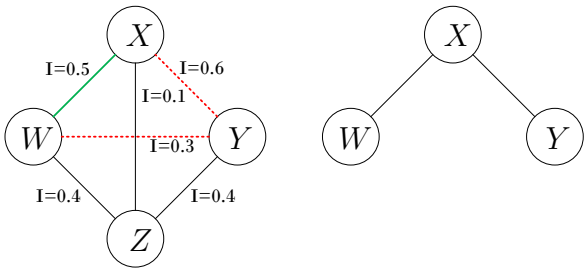

Fig. 4. Connect the next two random processes with the largest mutual information in the list, and remove it and all connections that might create loops on the graph.

tree head and the directions of the arrows. See Figure 5 and Figure 6.

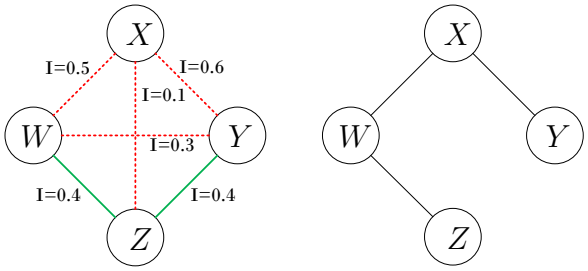

Fig. 5. Connect the next two random processes with the largest mutual information in the list, and remove it and all connections that might create loops on the graph.

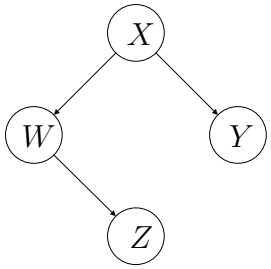

Fig. 6. Step 3: decide the direction of the arrows arbitrarily or by preference.

REFERENCES

[1] Chow, C. K., Liu, C.N. (1968), "Approximating discrete probability distributions with dependence trees", *IEEE Transactions on Information Theory*, IT-14 (3): 462-467

[2] I. Csiszar and J. Korner. Information Theory: "Coding Theorems for Discrete Memoryless Systems". Academic Press, New York, 1981.

[3] I Csiszar. "Sanov property, generalized I-projection and a conditional limit theorem". Ann. Prob., 12:768793, 1984.

[4] Kruskal, J. B. (1956). "On the shortest spanning subtree of a graph and the traveling salesman problem". Proceedings of the American Mathematical Society. 7: 4850. doi:10.1090/S0002-9939-1956-0078686-7. JSTOR 2033241.

[5] I. N. Sanov. "On the probability of large deviations of random variables". Mat. Sbornik, 42:1144, 1957. English translation in Sel. Transl. Math. Stat. Prob., Vol. 1, pp. 213-244, 1961.

[6] J. Wolfowitz. "Coding Theorems of Information Theory". Springer-Verlag, Berlin, and Prentice-Hall, Englewood Cliffs, NJ, 1978.

[7] T. M. Cover and J. A. "Thomas, *Elements of Information Theory*", 2nd ed. New-York: Wiley, 2006.

[8] T Weissman. Information Theory: "Conditional Differential Entropy, Info. Theory in ML", Lecture 20, EE376A/STATS376A, stanford university, 2018.
`https://web.stanford.edu/class/ee376a/files/lecture_20.pdf`