

## A Limitations

**Indeterminate Probability Theory.** As we summarized in Section 4.5, we do not find any exceptions for our proposed three conditional mutual independency assumptions, see Assumption 2 Assumption 3 and Assumption 4. And our proposed Equation (12) is derived from these assumptions, in our opinion, this equation can be applied to any general random experiment.

**IPNN.** IPNN is one neural network framework based on indeterminate probability theory, it has three limitations: (1) The split shape need to be predefined, a proper sample space for an unknown dataset can only be found with try and error. The latent variables are continuous in CIPNN [10], therefore this issue does not exist in CIPNN. (2) It sometimes converges to local minimum, but we can avoid this problem with a proper model weights initialization, as discussed in Appendix D. (3) As joint sample space increases exponentially, the memory consumption and computation time also increase accordingly. This issue only exist during training, and can be avoided through monte carlo method for prediction task, as discussed in CIPNN [10], this paper will not further discuss it.

## B An Intuitive Explanation

Since our proposed indeterminate probability theory is quite new, we will explain this idea by comparing it with classical probability theory, see below table:

Table 5: An intuitive comparison between classical probability theory and our proposed theory.

Observation (Classical)	$P(Y = y_l   A^j = a_{i,j}^j) = \frac{\text{number of event } (Y=y_l, A^j=a_{i,j}^j) \text{ occurs}}{\text{number of event } (A^j=a_{i,j}^j) \text{ occurs}}$
Inference (Classical)	$X = x_{n+1} \xrightarrow[\text{Determinate}]{P(A^j=a_{i,j}^j   X=x_{n+1})=1} A^j = a_{i,j}^j \xrightarrow[\text{infer}]{P(Y=y_l   A^j=a_{i,j}^j)} Y = y_l$
Observation (Ours)	$P(Y = y_l   A^j = a_{i,j}^j) = \frac{\text{sum of event } (Y=y_l, A^j=a_{i,j}^j) \text{ occurs, in decimal}}{\text{sum of event } (A^j=a_{i,j}^j) \text{ occurs, in decimal}}$
Inference (Ours)	$X = x_{n+1} \left\{ \begin{array}{l} \xrightarrow{P(A^j=a_1^j   X=x_{n+1}) \in [0,1]} A^j = a_1^j \xrightarrow{P(Y=y_l   A^j=a_1^j)} \\ \xrightarrow{P(A^j=a_2^j   X=x_{n+1}) \in [0,1]} A^j = a_2^j \xrightarrow{P(Y=y_l   A^j=a_2^j)} \\ \vdots \xrightarrow{\quad \quad \quad} A^j = \dots \xrightarrow{\quad \quad \quad} \\ \xrightarrow{P(A^j=a_{M_j}^j   X=x_{n+1}) \in [0,1]} A^j = a_{M_j}^j \xrightarrow{P(Y=y_l   A^j=a_{M_j}^j)} \end{array} \right\} Y = y_l$ <div style="display: flex; justify-content: space-around; width: 100%;"> <span>Indeterminate</span> <span>infer</span> </div>

Note: Replacing  $A^j$  with joint random variable  $(A^1, A^2, \dots, A^N)$  is also valid for above explanation.

In other word, for classical probability theory, perform a random experiment  $X = x_k$ , the event state is Determinate (happened or not happened), the probability is calculated by counting the number of occurrences, we define this process here as observation phase. For inference, perform a new random experiment  $X = x_{n+1}$ , the state of  $A^j = a_{i,j}^j$  is Determinate again, so condition on  $X = x_{n+1}$  is equivalent to condition on  $A^j = a_{i,j}^j$ , that may be the reason why condition on  $X = x_{n+1}$  is not discussed explicitly in the past.

However, for our proposed indeterminate probability theory, perform a random experiment  $X = x_k$ , the event state is Indeterminate (understood as partly occurs), the probability is calculated by summing the decimal value of occurrences in observation phase. For inference, perform a new random experiment  $X = x_{n+1}$ , the state of  $A^j = a_{i,j}^j$  is Indeterminate again, each case contributes the inference of  $Y = y_l$ , so the inference shall be the summation of all cases. Therefore, condition on  $X = x_{n+1}$  is now different with condition on  $A^j = a_{i,j}^j$ , we need to explicitly formulate it, see Equation (12).

402 Once again, our proposed indeterminate probability theory does not have any conflict with classical  
 403 probability theory, the observation and inference phase of classical probability theory is one special  
 404 case to our theory.

## 405 C Global Minimum Analysis

406 *Proof of Proposition 1.* Equation (10) can be rewritten as:

$$P^{\mathbb{A}}(y_l | x_t) = \sum_{\mathbb{A}} \left( p_{\mathbb{A}} \cdot \prod_{j=1}^N \alpha_{i_j}^j(t) \right) \quad (21)$$

407 Where,

$$p_{\mathbb{A}} = P(y_l | a_{i_1}^1, a_{i_2}^2, \dots, a_{i_N}^N) \quad (22)$$

408 Theoretically, for  $P(y_l | x_k) = y_l(k) \in \{0, 1\}$  hard label case, model converges to global minimum  
 409 when the train and test loss is zero [33], and for the ground truth  $y_l(t) = 1$ , with Equation (18) we  
 410 have:

$$\sum_{\mathbb{A}} \left( p_{\mathbb{A}} \cdot \prod_{j=1}^N \alpha_{i_j}^j(t) \right) = 1 \quad (23)$$

411 Subtract the above equation from Equation (4) gives:

$$\sum_{\mathbb{A}} \left( (1 - p_{\mathbb{A}}) \cdot \prod_{j=1}^N \alpha_{i_j}^j(t) \right) = 0 \quad (24)$$

412 Because  $\prod_{j=1}^N \alpha_{i_j}^j(t) \in [0, 1]$  and  $(1 - p_{\mathbb{A}}) \in [0, 1]$ , The above equation is then equivalent to:

$$p_{\mathbb{A}} = 1, \text{ for } \prod_{j=1}^N \alpha_{i_j}^j(t) > 0, i_j = 1, 2, \dots, M_j. \quad (25)$$

413

□

## 414 D Local Minimum Analysis

415 Equation (21) can be further rewritten as:

$$P^{\mathbb{A}}(y_l | x_t) = \sum_{i_{\tau}=1}^{M_{\tau}} \left( \alpha_{i_{\tau}}^{\tau}(t) \cdot \sum_{\Lambda} \left( p_{\Lambda} \cdot \prod_{j=1, j \neq \tau}^N \alpha_{i_j}^j(t) \right) \right) = \sum_{i_{\tau}=1}^{M_{\tau}} (\alpha_{i_{\tau}}^{\tau}(t) \cdot p_{i_{\tau}}) \quad (26)$$

416 Where  $\Lambda = (A^1, \dots, A^j, \dots, A^N) \subset \mathbb{A}, j \neq \tau$  and,

$$p_{i_{\tau}} = \sum_{\Lambda} \left( p_{\Lambda} \cdot \prod_{j=1, j \neq \tau}^N \alpha_{i_j}^j(t) \right) \quad (27)$$

417 Substitute Equation (26) into Equation (18), and for the ground truth  $y_l(t) = 1$  the loss function can  
 418 be written as:

$$\mathcal{L} = -\log(\sum_{i_{\tau}=1}^{M_{\tau}} (\alpha_{i_{\tau}}^{\tau}(t) \cdot p_{i_{\tau}})) \quad (28)$$

Let the model output before softmax function be  $z_{i_j}$ , we have:

$$\alpha_{i_\tau}^\tau(t) = \frac{e^{z_{i_\tau}}}{\sum_{i_j=1}^{M_j} e^{z_{i_j}}} \quad (29)$$

In order to simplify the calculation, we assume  $p_{\mathbb{A}}$  defined in Equation (22) is constant during back-propagation. so the gradient is:

$$\frac{\partial \mathcal{L}}{\partial z_{i_\tau}} = - \frac{\alpha_{i_\tau}^\tau(t) \cdot \sum_{i_j=1, i_j \neq i_\tau}^{M_j} (e^{z_{i_j}} \cdot (p_{i_\tau} - p_{i_j}))}{\sum_{i_j=1}^{M_j} (e^{z_{i_j}} \cdot p_{i_j})} \quad (30)$$

Therefore, we have two kind of situations that the algorithm will go to local minimum:

$$\frac{\partial \mathcal{L}}{\partial z_{i_\tau}} = \begin{cases} \rightarrow 0, & \text{if } |z_{i_\tau} - z_{i_j}| \rightarrow \infty \\ 0, & \text{if } p_{i_\tau} = p_{i_j} \\ \text{Nonezero}, & o.w. \end{cases} \quad (31)$$

Where  $i_\tau = 1, 2, \dots, M_\tau$ .

The first local minimum usually happens when Corollary 1 is not satisfied, that is, the number of joint sample points is smaller than the classification classes, the results are shown in Figure 4a.

If the model weights are initialized to a very small value, the second local minimum may happen at the beginning of training. In such case, all the model output values are also small which will result in  $\alpha_1^j(t) \approx \alpha_2^j(t) \approx \dots \approx \alpha_{M_j}^j(t)$ , and it will further lead to all the  $p_{i_\tau}$  be similar among each other. Therefore, if the model loss reduces slowly at the beginning of training, the model weights is suggested to be initialized to an relative high value. But the model weights shall not be set to too high values, otherwise it will lead to first local minimum.

As shown in Figure 5, if model weights are initialized to uniform distribution of  $[-10^{-6}, 10^{-6}]$ , its convergence speed is slower than the model weights initialized to uniform distribution of  $[-0.3, 0.3]$ . Besides, model weights initialized to uniform distribution of  $[-3, 3]$  get almost stuck at local minimum and cannot go to global minimum. This result is consistent with our analysis.

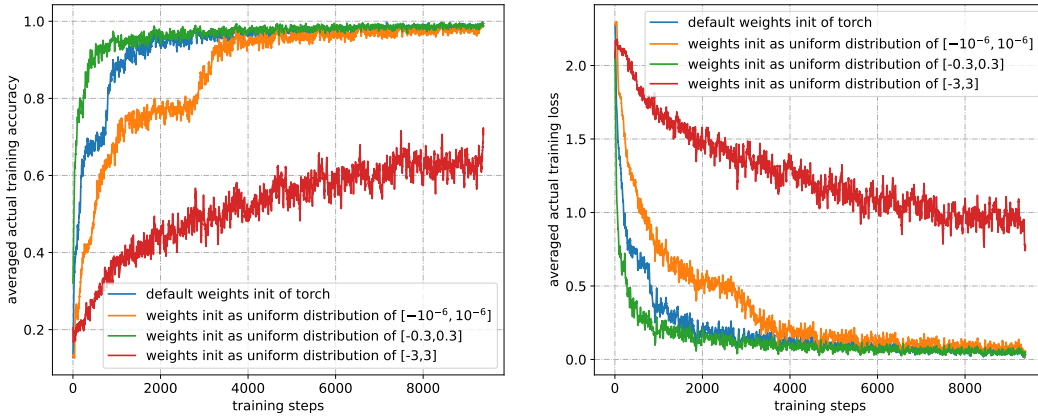


Figure 5: Model weights initialization impact analysis on MNIST. Split shape is  $\{2, 10\}$ , batch size is 64, forget number  $T = 5$ ,  $\epsilon = 10^{-6}$ .

## D.1 Avoiding Local Minimum with Multi-degree Classification

Another experiment is designed by us to check the performance of multi-degree classification (see Section 5.2): classification of binary vector into decimal value. The binary vector is the model inputs

439 from ‘000000000000’ to ‘111111111111’, which are labeled from 0 to 4095. The split shape is set  
 440 to  $\{M_1 = 2, M_2 = 2, \dots, M_{12} = 2\}$ , which is exactly able of making a full classification. Besides,  
 441 model weights are initialized as uniform distribution of  $[-0.3, 0.3]$ , as discussed in Appendix D.

442 The result is shown in Figure 6, IPNN without multi degree classification goes to local minimum  
 443 with only 69.5% train accuracy. We have only additionally labeled for 12 sub-joint spaces, and IPNN  
 444 goes to global minimum with 100% train accuracy.

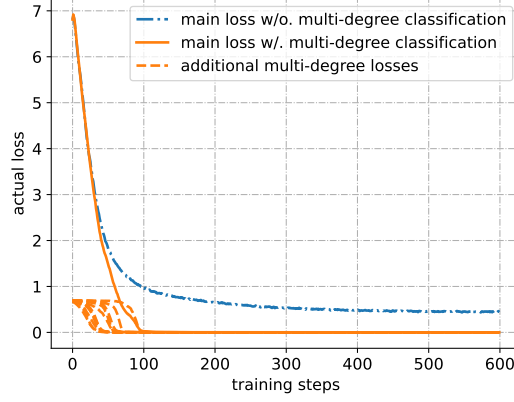


Figure 6: Loss of multi-degree classification of ‘binary to decimal’ on train dataset. Input samples are additionally labeled with  $Y^i \in \{0, 1\}$  for  $i^{th}$  bit is 0 or 1, respectively.  $Y^i$  corresponds to sub-joint sample space  $\Lambda^i$  with split shape  $\{M_i = 2\}, i = 1, 2, \dots, 12$ . Batch size is 4096, forget number  $T = 5$ ,  $\epsilon = 10^{-6}$ .

445 Therefore, with only  $\sum_{i=1}^{12} 2 = 24$  output nodes, IPNN can classify 4096 categories. Theoretically,  
 446 if model with 100 output nodes are split into 10 equal parts, it can classify 10 billion categories.  
 447 Hence, compared with the classification model with only one ‘softmax’ function, IPNN has no  
 448 computationally expensive problems (see Section 1).

## 449 E Mutual Independency

450 If we want the random variables  $A^1, A^2, \dots, A^N$  partly or fully mutually independent, we can use  
 451 their mutual information as loss function:

$$\begin{aligned} \mathcal{L}^* &= KL \left( P(A^1, A^2, \dots, A^N), \prod_{j=1}^N P(A^j) \right) = \sum_{\mathbb{A}} \left( P(a_{i_1}^1, \dots, a_{i_N}^N) \cdot \log \frac{P(a_{i_1}^1, \dots, a_{i_N}^N)}{\prod_{j=1}^N P(a_{i_j}^j)} \right) \\ &= \sum_{\mathbb{A}} \left( \frac{\sum_{k=1}^n \left( \prod_{j=1}^N \alpha_{i_j}^j(k) \right)}{n} \cdot \log \left( \frac{\sum_{k=1}^n \left( \prod_{j=1}^N \alpha_{i_j}^j(k) \right)}{n} \cdot \frac{n}{\prod_{j=1}^N \sum_{k=1}^n \alpha_{i_j}^j(k)} \right) \right) \end{aligned} \quad (32)$$

## 452 F Properties of Indeterminate Probability Theory

453 The indeterminate probability theory (see Equation (12)) may have the following properties, some  
 454 have not been proved mathematically due to our limited knowledge.

455 **Proposition 2.** *IF given  $A, B$  and  $Y$  is independent, we have  $P(Y | A, B) = P(Y | A)$ , THEN:*

$$P^{(A,B)}(Y | X = x_{n+1}) = P^A(Y | X = x_{n+1}) \quad (33)$$

456 *This property is understood as: Suppose given  $A, B$  and  $Y$  is independent, so  $B$  does not contribute*  
 457 *for the inference.*

*Proof.*

$$\begin{aligned}
& P^{(A,B)}(Y | X = x_{n+1}) \\
&= \sum_{A,B} (P(Y | A, B) \cdot P(A, B | X = x_{n+1})) \\
&= \sum_{A,B} (P(Y | A) \cdot P(A | X = x_{n+1}) \cdot P(B | X = x_{n+1})) \\
&= \sum_A (P(Y | A) \cdot P(A | X = x_{n+1})) \cdot \sum_B P(B | X = x_{n+1}) \\
&= \sum_A (P(Y | A) \cdot P(A | X = x_{n+1})) \\
&= P^A(Y | X = x_{n+1})
\end{aligned} \tag{34}$$

458

□

459 **Hypothesis 1.** Let  $Y, V$  be any two different random variables, Similarly, according to Assumption 2,  
460 we have  $P(Y, V | X = x_{n+1}) = P(Y | X = x_{n+1}) \cdot P(V | X = x_{n+1})$ . Our hypothesis is:

$$P^A(Y, V | X = x_{n+1}) = P^A(Y | X = x_{n+1}) \cdot P^A(V | X = x_{n+1}) \tag{35}$$

461 This property is understood as: Given  $X, Y$  and  $V$  is independent, so the inference outcome is also  
462 independent.

463 **Hypothesis 2.** Let  $P(A | X = x_{n+1}) \in [0, 1)$  and

$$\begin{aligned}
P(Y^0 = y_l | X = x_{n+1}) &= P^A(Y = y_l | X = x_{n+1}) \\
P(Y^1 = y_l | X = x_{n+1}) &= P^{Y^0}(Y = y_l | X = x_{n+1}) \\
P(Y^2 = y_l | X = x_{n+1}) &= P^{Y^1}(Y = y_l | X = x_{n+1}) \\
&\dots
\end{aligned} \tag{36}$$

464 Our hypothesis is:

$$P^{Y^\infty}(Y = y_l | X = x_{n+1}) = \frac{1}{m}, l = 1, 2, \dots, m. \tag{37}$$

465 This property is understood as: The inference accuracy will become poor as the information is  
466 transmitted one after another (from  $Y^{i-1}$  to  $Y^i$ ).

467 **Hypothesis 3.** Let  $P(Y = y_l | X = x_{n+1}) \in \{0, 1\}$  and  $P(A | X = x_{n+1}) \in [0, 1)$ . Our hypothe-  
468 sis is:

$$\max_{l=1,2,\dots,m} P^{(A,A)}(Y = y_l | X = x_{n+1}) > \max_{l=1,2,\dots,m} P^{(A)}(Y = y_l | X = x_{n+1}) \tag{38}$$

469 This property is understood as: The inference tendency will get more stronger with more same  
470 information  $(A, A)$ .

## 471 G Symbols

Table 6: Reading Symbols

Symbol	Meaning
$x_k$	input sample, $k = 1, 2, \dots, n$
$y_l$	output label, $l = 1, 2, \dots, m$
$A^j$	random variable, $j = 1, 2, \dots, N$
$a_{i_j}^j$	event of $A^j$ , $i_j = 1, 2, \dots, M_j$
$\mathbb{A}$	joint sample space
$\Lambda$	sub-joint sample space