

# Deep Neural Decision Trees

読み会@2020/12/03

楊明哲

- 著者
  - Yongxin Yang, Irene Garcia Morillo, Timothy M. Hospedales
  - The University of Edinburg
- 出典: WHI 2018

# 概要

どんな論文？

- 決定木とニューラルネットワークを組み合わせた！
- 決定木の解釈性，ニューラルネットワークの表現力（精度）のいいところ取りを目指している.

# 概要

## 研究背景

- 近年，ニューラルネットワークによるモデルの精度が高いためよく使われているが，出力に対しての解釈性が乏しい.
- 従来使われていた，決定木はニューラルネットワークよりも精度が出せないこともある一方で，解釈性に優れている.

# 概要

## 貢献

- 既存のNNライブラリに数行追加するだけで，実装が可能な手法を提案
- 学習方法は全て逆誤差電波で行うことが可能である．
- 本手法では，特に表形式の分類問題に有効である．

# 提案手法

## Soft binning function

- Soft binning function (Dougherty et al., 1995)のテクニックを利用する.
- 実数の連続値  $x$  をbinで分けるテクニック.
  - → 微分可能な関数をもちいて行うため, 逆誤差伝播が可能である!!

# 提案手法

## Soft binning function

increasing manner<sup>2</sup>, i.e.,  $\beta_1 < \beta_2 < \dots < \beta_n$ .

Now we construct a one-layer neural network with softmax as its activation function.

$$\pi = f_{w,b,\tau}(x) = \text{softmax}((wx + b)/\tau) \quad (1)$$

Here  $w$  is a constant rather than a trainable variable, and its value is set as  $w = [1, 2, \dots, n + 1]$ .  $b$  is constructed as,

$$b = [0, -\beta_1, -\beta_1 - \beta_2, \dots, -\beta_1 - \beta_2 - \dots - \beta_n]. \quad (2)$$

and  $\tau > 0$  is a temperature factor. As  $\tau \rightarrow 0$  the output tends to a one-hot vector.

温度付ソフトマックス関数

$\tau \rightarrow 0$  の時

One-hotベクトルと同等になる

$b$ のベクトルのなかに

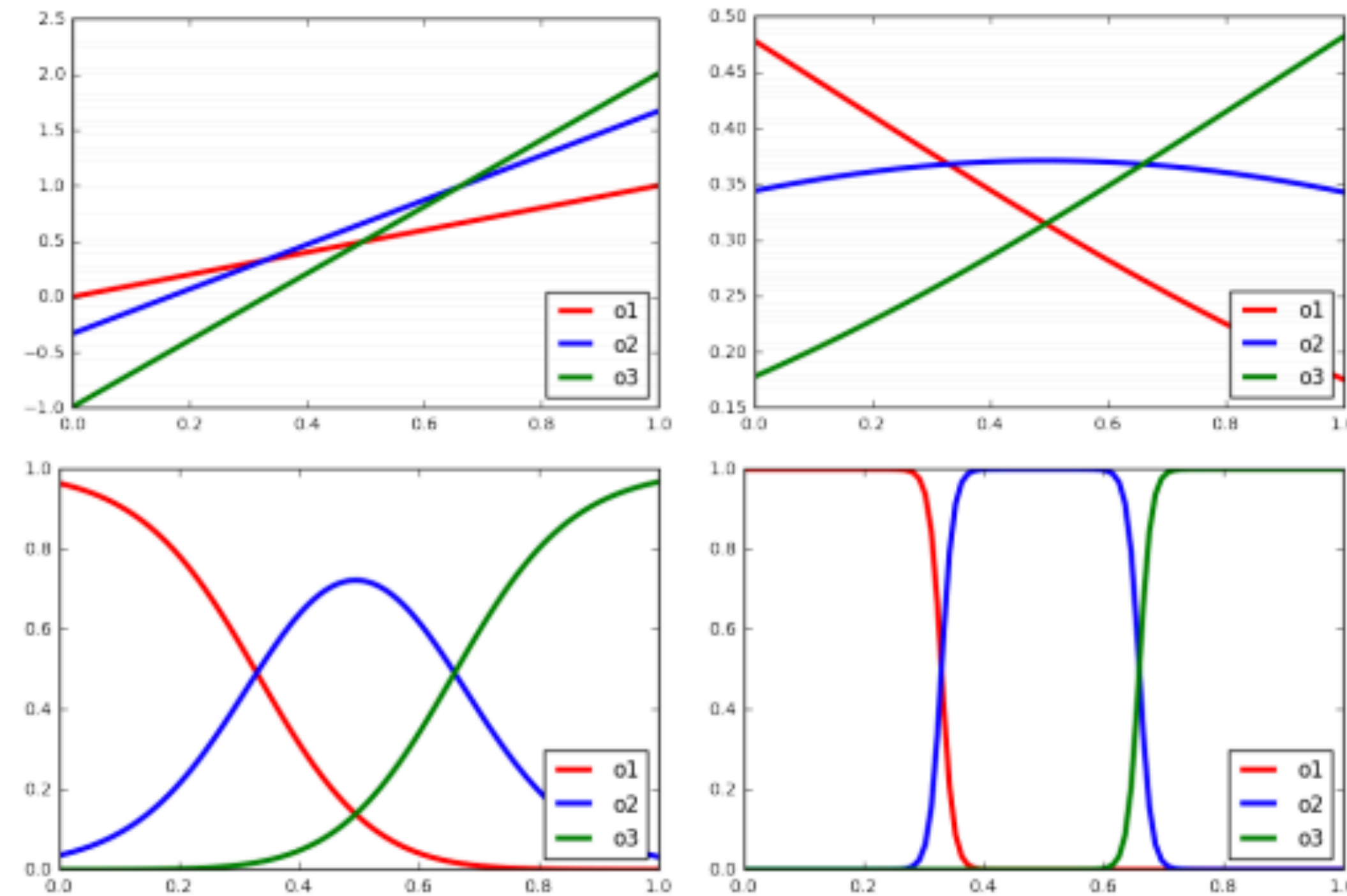
binningを行う

閾値が入っている



# 提案手法

## Soft binning function



温度付ソフトマックス関数  
 $\tau \rightarrow 0$  の時  
One-hotベクトルと同等になる

*Figure 1.* A concrete example of our soft binning function using cut-points at 0.33 and 0.66. x-axis is the value of a continuous input variable  $x \in [0, 1]$ . Top-left: the original values of logits; Top-right: values after applying softmax function with  $\tau = 1$ ; Bottom-left:  $\tau = 0.1$ ; Bottom-right:  $\tau = 0.01$ .



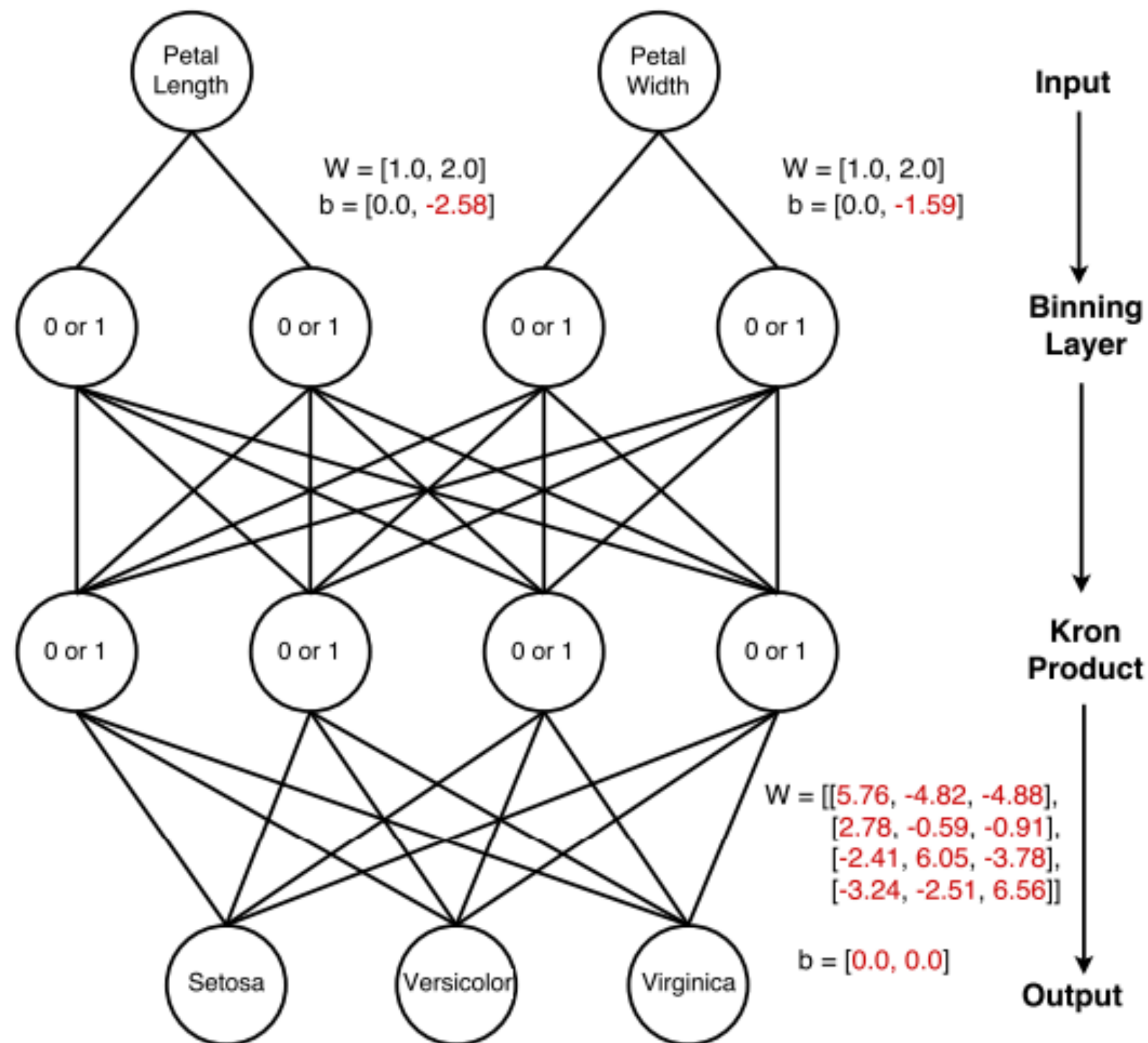
# 提案手法

## 予測パート

- Binning functionによって、分けられた特徴量を用いて、ニューラルネットワークモデルで順電波を行う。
- NNの最終出力に対して、クロネッカー積を行う（葉ができる）
  - → ほとんどone-hotと同じ状態になる。
- 葉を全結合した結果が、分類した結果になる。

# 提案手法

## 予測パート



特徴量に対して  
Binningを行う

Biningした特徴を用いて  
ニューラルネットワークを行う

クロネッカー積をとり  
全結合する

# 提案手法

## 学習時

- Soft binning functionとNN部分は微分可能であるから学習は誤差逆伝播を用いることができる！

# 実験

## 実験で確認すること

- 既存手法（NNモデルと決定木モデル）と提案手法を複数の表形式データセットで精度の比較を行う.



# 実験

## 実験結果

Dataset	#inst.	#feat.	#cl.
Iris	150	4	3
Haberman's Survival	306	3	2
Car Evaluation	1728	6	4
Titanic (K)	714	10	2
Breast Cancer Wisconsin	683	9	2
Pima Indian Diabetes (K)	768	8	2
Gime-Me-Some-Credit (K)	201669	10	2
Poker Hand	1025010	11	9
Flight Delay	1100000	9	2
HR Evaluation (K)	14999	9	2
German Credit Data	1000	20	2
Connect-4	67557	42	2
Image Segmentation	2310	19	7
Covertime	581012	54	7

Table 1. Collection of 14 datasets from Kaggle (indicated with (K)) and UCI: number of instances (#inst.), number of features (#feat.), and number of classes (#cl.)

Dataset	DNDT	DT	NN
Iris	<b>100.0</b>	<b>100.0</b>	<b>100.0</b>
Haberman's Survival	<b>70.9</b>	66.1	<b>70.9</b>
Car Evaluation	95.1	<b>96.5</b>	91.6
Titanic	<b>80.4</b>	79.0	76.9
Breast Cancer Wisconsin	94.9	91.9	<b>95.6</b>
Pima Indian Diabetes	66.9	<b>74.7</b>	64.9
Gime-Me-Some-Credit	98.6	92.2	<b>100.0</b>
Poker Hand	50.0	<b>65.1</b>	50.0
Flight Delay	<b>78.4</b>	67.1	78.3
HR Evaluation	92.1	<b>97.9</b>	76.1
German Credit Data (*)	<b>70.5</b>	66.5	<b>70.5</b>
Connect-4 (*)	66.9	<b>77.7</b>	75.7
Image Segmentation (*)	70.6	<b>96.1</b>	48.05
Covertime (*)	49.0	<b>93.9</b>	49.0
# of wins	5	<b>7</b>	5
Mean Reciprocal Rank	0.65	<b>0.73</b>	0.61

Table 2. Test set accuracy of each model: DT: Decision tree. NN: neural network. DNDT: Our deep neural decision tree, where (\*) indicates that the ensemble version is used.

# 実験

## 実験結果

- 既存手法に優ったり劣っていたり，データセットによって一長一短であった．
- 既存手法と同等の性能は持っている．

# 実験

## 実験結果

Feat. Idx \ Dataset	0	1	2	3	4	5	6	7	8	9
Haberman's	<b>100</b>	<b>100</b>	0	-	-	-	-	-	-	-
Iris	<b>100</b>	90	50	10	-	-	-	-	-	-
Pima	10	0	0	0	20	0	0	<b>100</b>	-	-
Titanic	0	0	0	0	0	10	20	10	20	40

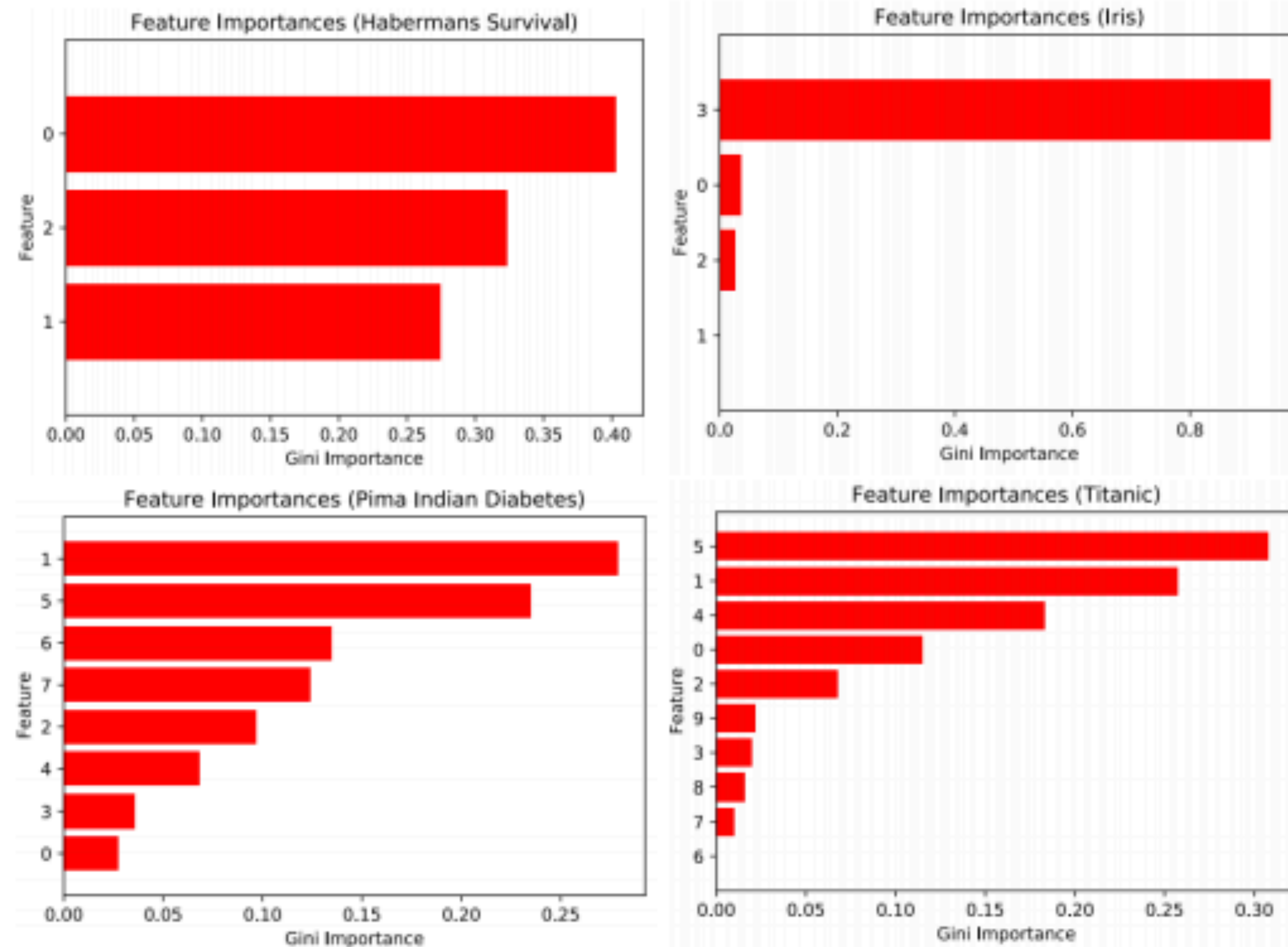
Table 3. Percentage (%) of times that DNDT ignores each feature.

- 提案手法では使われない特徴があることが確認できた.
- →予測に必要ないと判断できる



# 実験

## モデルの解釈性

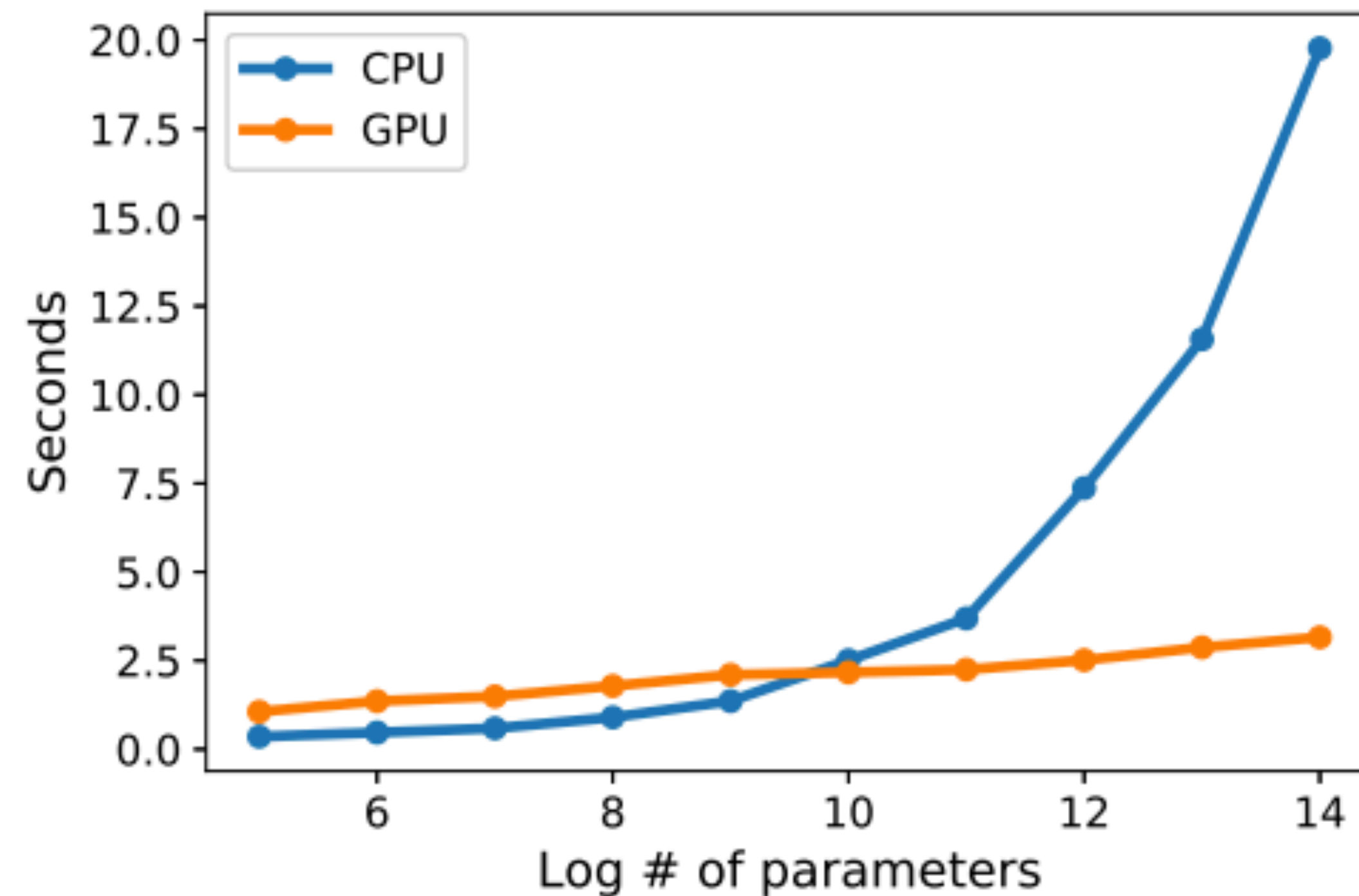


- 葉のクロネッカー積をもとにどの特徴量が使われているかを抽出することができる。

Figure 5. Feature importance ranking produced by DT (Gini).

# 実験

## 計算効率



- 特徴量が増えた時でも、GPUによる計算効率は良かった。

Figure 6. GPU Acceleration illustration: DNDT training time on 3.6GHz CPU vs GTX Titian GPU. Average over 5 runs.

# まとめ

- 本研究では、決定木とNNモデルを組み合わせた手法を提案した.
- 本手法では、NNモデルと同等の性能を持ち、決定木みたいな解釈性の高いモデルになった.