Something (somebody) *new* in machine learning

Mai 2014

Alain Tapp

Alain Tapp

- Interests
  - Privacy
  - Machine learning
  - Impact of technology
- June 16…
  - 6 month sabbatical

Learning machine learning

- Unsupervised learning!

- What is an *interesting* bit?
- What is an *interesting* function?

Supervised learning

- Data
  - All examples in binary
  - All examples have the same size
  - Each bit position has unique meaning
  - Order of the bits is unimportant
- Algorithm
  - Classification with 2 categories
  - Uses fast binary vector operation
  - Learn a binary circuit
  - Classifier implementable on FPGA

Binary functions: Gates

2 bits
16 fonctions

$y = F(x_1, x_2)$

Input	Output
00	F(0)
01	F(1)
10	F(2)
11	F(3)

3 bits
256 fonctions

$y = F(x_1, x_2, x_3)$

Input	Output
000	F(0)
001	F(1)
010	F(2)
011	F(3)
100	F(4)
101	F(5)
110	F(6)
111	F(7)

4 bits
65536 fonctions

$y = F(x_1, x_2, x_3, x_4)$

Input	Output
0000	F(0)
0001	F(1)
0010	F(2)
0011	F(3)
…	…
1101	F(13)
1110	F(14)
1111	F(15)

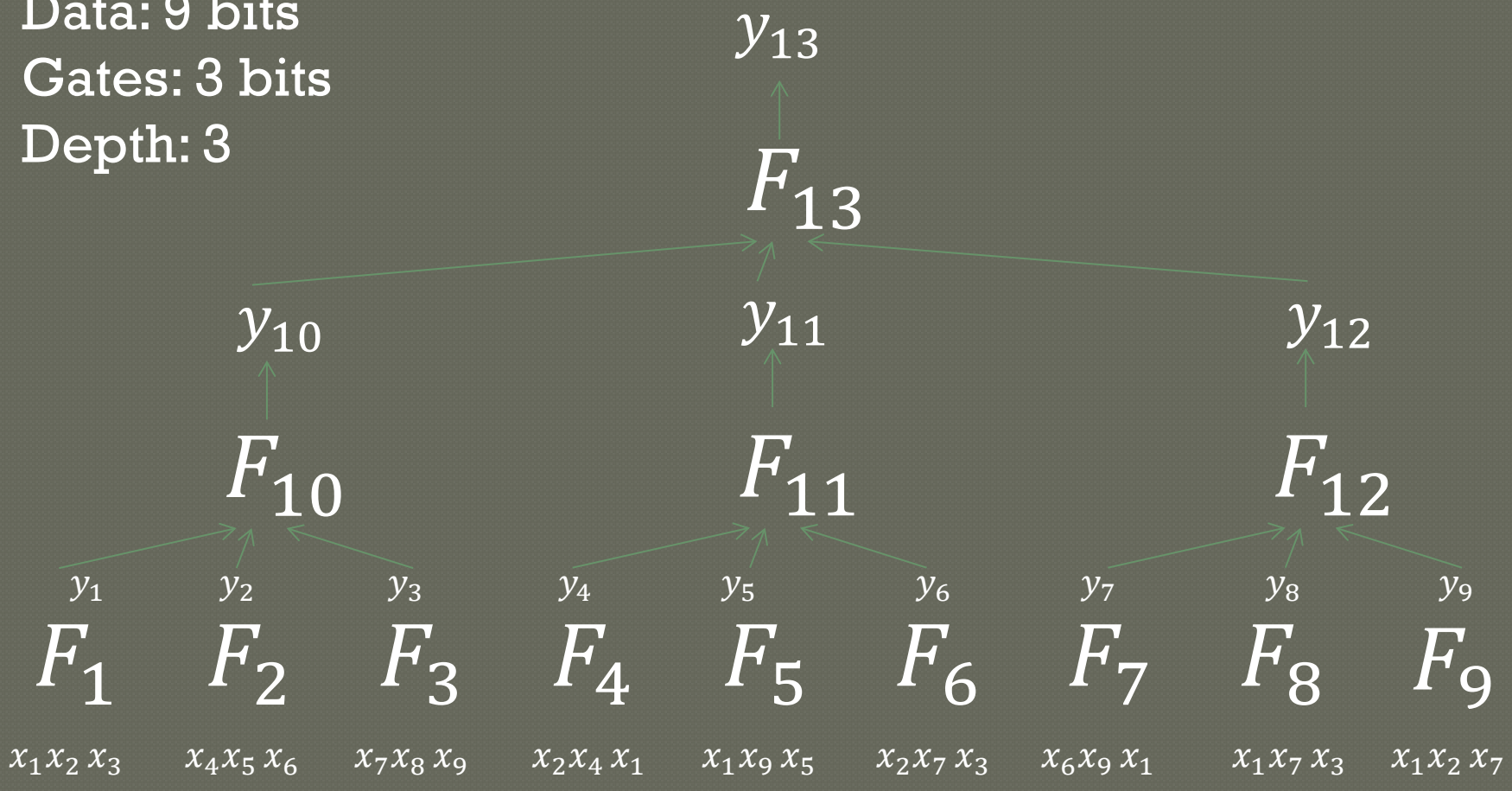
8 bits
115792089237316195423570985008687907853269
984665640564039457584007913129639936 fonctions

$y = F(x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8)$

Input	Output
00000000	F(0)
00000001	F(1)
00000010	F(2)
00000011	F(3)
…	…
11111101	F(253)
11111110	F(254)
11111111	F(255)

Function tree

Data: 9 bits
Gates: 3 bits
Depth: 3



Tensor

$k = 3$

$\{d_1, \overline{d_1}\} \quad \{d_2, \overline{d_2}\} \quad \{d_3, \overline{d_3}\}$

$$\{d_1, \overline{d_1}\} \otimes \{d_2, \overline{d_2}\} = \{d_1 d_2, d_1 \overline{d_2}, \overline{d_1} d_2, \overline{d_1 d_2}\}$$

$$\{d_1, \overline{d_1}\} \otimes \{d_2, \overline{d_2}\} \otimes \{d_3, \overline{d_3}\} = \{d_1 d_2, d_1 \overline{d_2}, \overline{d_1} d_2, \overline{d_1 d_2}\} \otimes \{d_3, \overline{d_3}\}$$
$$\{d_1 d_2 d_3, d_1 \overline{d_2} d_3, \overline{d_1} d_2 d_3, \overline{d_1 d_2} d_3, d_1 d_2 \overline{d_3}, d_1 \overline{d_2 d_3}, \overline{d_1} d_2 \overline{d_3}, \overline{d_1 d_2 d_3}\}$$

$k = 2$ requires 4 operations, and in general $2^{k+1} - 4$

$$d_1 \wedge d_2 \wedge d_3 = \{d_1 d_2 d_3, d_1 \overline{d_2} d_3 + \overline{d_1} d_2 d_3 + \overline{d_1 d_2} d_3 + d_1 d_2 \overline{d_3} + d_1 \overline{d_2 d_3} + \overline{d_1} d_2 \overline{d_3} + \overline{d_1 d_2 d_3}\}$$

$$d_1 \oplus d_2 \oplus d_3 = \{d_1 \overline{d_2} d_3 + \overline{d_1} d_2 d_3 + d_1 d_2 \overline{d_3} + \overline{d_1 d_2 d_3}, d_1 d_2 d_3 + \overline{d_1 d_2} d_3 + d_1 \overline{d_2 d_3} + \overline{d_1} d_2 \overline{d_3}\}$$

Any function is a subset of the tensor product terms.
We chose the fucntions that maximize some property.
There is $2^{(2^k)}$ possible functions

2 bit

$\{d_1, \overline{d_1}\} \otimes \{d_2, \overline{d_2}\} = \{d_1 d_2, d_1 \overline{d_2}, \overline{d_1} d_2, \overline{d_1 d_2}\}$

A total of 16 functions

- Zero bit (2): $\{1,0\}$
- One bit (4): $\{d_1, \overline{d_1}\}$
- And (8): $\{d_1 d_2, d_1 \overline{d_2} + \overline{d_1} d_2 + \overline{d_1 d_2}\}$
- Xor (2): $\{d_1 d_2 + \overline{d_1 d_2}, d_1 \overline{d_2} + \overline{d_1} d_2\}$

3 bit

$\{d_1, \overline{d_1}\} \otimes \{d_2, \overline{d_2}\} \otimes \{d_3, \overline{d_3}\} = \{d_1 d_2, d_1 \overline{d_2}, \overline{d_1} d_2, \overline{d_1 d_2}\} \otimes \{d_3, \overline{d_3}\}$
$\{d_1 d_2 d_3, d_1 \overline{d_2} d_3, \overline{d_1} d_2 d_3, \overline{d_1 d_2} d_3, d_1 d_2 \overline{d_3}, d_1 \overline{d_2 d_3}, \overline{d_1} d_2 \overline{d_3}, \overline{d_1 d_2 d_3}\}$

$T = \{t_1, t_2, t_3, t_4, t_5, t_6, t_7, t_8\}$
A total of 256 functions

- Zero bit (2): $\{1,0\}$
- One bit (3x2=6): $\{d_1, \overline{d_1}\}$
- Two bits (3x10=30):
- And (16): $\{d_1 d_2 d_3, \dots\}$
- Maj (8): $\{d_1 d_2 d_3 + d_1 \overline{d_2} d_3 + \overline{d_1} d_2 d_3 + d_1 d_2 \overline{d_3}, \dots\}$
- Maj (deux) (8x2x3) $\{d_1 d_2 d_3 + d_1 \overline{d_2} d_3 + \overline{d_1} d_2 d_3, \dots\}$
- Maj (un) (8x2x3) $\{d_1 d_2 d_3 + \overline{d_1} d_2 d_3, \dots\}$
- Maj (mixed) :
- Xor (2): $\{d_1 \overline{d_2} d_3, \overline{d_1} d_2 d_3, d_1 d_2 \overline{d_3}, \overline{d_1 d_2 d_3}, \dots\}$

Data vectors and solution vector

- n is the number of training element
- v is the solution vector
- $\{d_1, \overline{d_1}\}$

- $S_p(\{d_1, \overline{d_1}\}) = \frac{\#(d_1 v) + \#(\overline{d_1}\bar{v})}{n}$
  $=$ prob. of right guess

Maximize S

- $k = 3$
- $\{d_1, \overline{d_1}\} \otimes \{d_2, \overline{d_2}\} \otimes \{d_3, \overline{d_3}\} = \{d_1 d_2, d_1 \overline{d_2}, \overline{d_1} d_2, \overline{d_1 d_2}\} \otimes \{d_3, \overline{d_3}\}$
- $\{d_1 d_2 d_3, d_1 \overline{d_2} d_3, \overline{d_1} d_2 d_3, \overline{d_1 d_2} d_3, d_1 d_2 \overline{d_3}, d_1 \overline{d_2 d_3}, \overline{d_1} d_2 \overline{d_3}, \overline{d_1 d_2 d_3}\}$

- $T = \{t_1, t_2, t_3, t_4, t_5, t_6, t_7, t_8\}$

- $\max_{A \subseteq T} S_p(\{A, \bar{A}\}) = F$
- Output: $\{F, \bar{F}\}$

- $\{\sum_{t_i : \#(t_i v) > \#(t_i \bar{v})} t_i \, , \sum_{t_i : \#(t_i v) \leq \#(t_i \bar{v})} t_i\}$

Average information gain

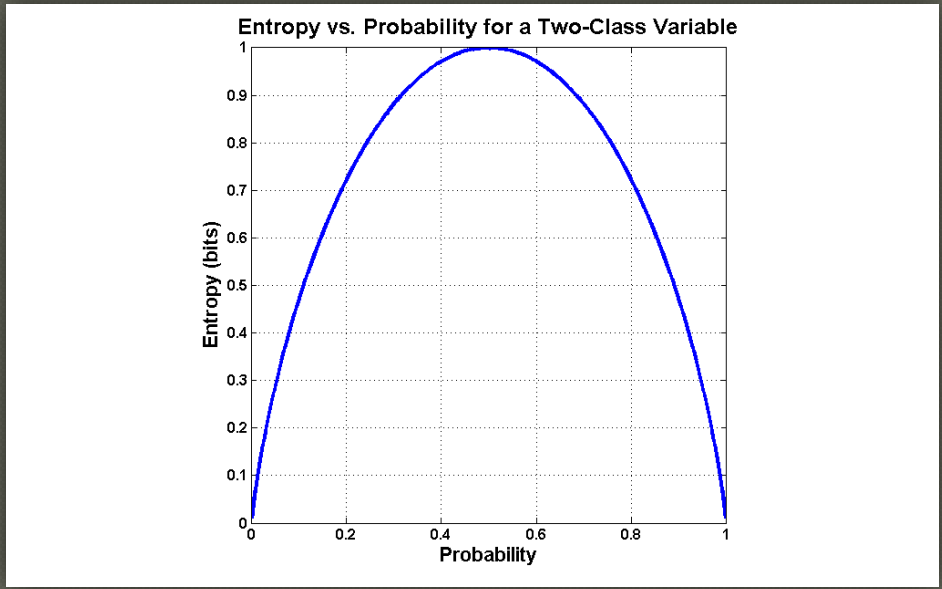
- $F(x) = 0 \rightarrow 75\%$
- $F(x) = 1 \rightarrow 75\%$

- $\mathrm{G}(x) = 0 \rightarrow 50\%$
- $G(x) = 1 \rightarrow 100\%$

- $F(x) \rightarrow 75\%$
- $F^3(x) \rightarrow 84\%$
- $F^5(x) \rightarrow 90\%$

- $G(x) \rightarrow 75\%$
- $G^3(x) \rightarrow 94\%$
- $G^5(x) \rightarrow 98\%$

Information



$$H(X) = -\sum_{x \in X} p_x \log(p_x)$$

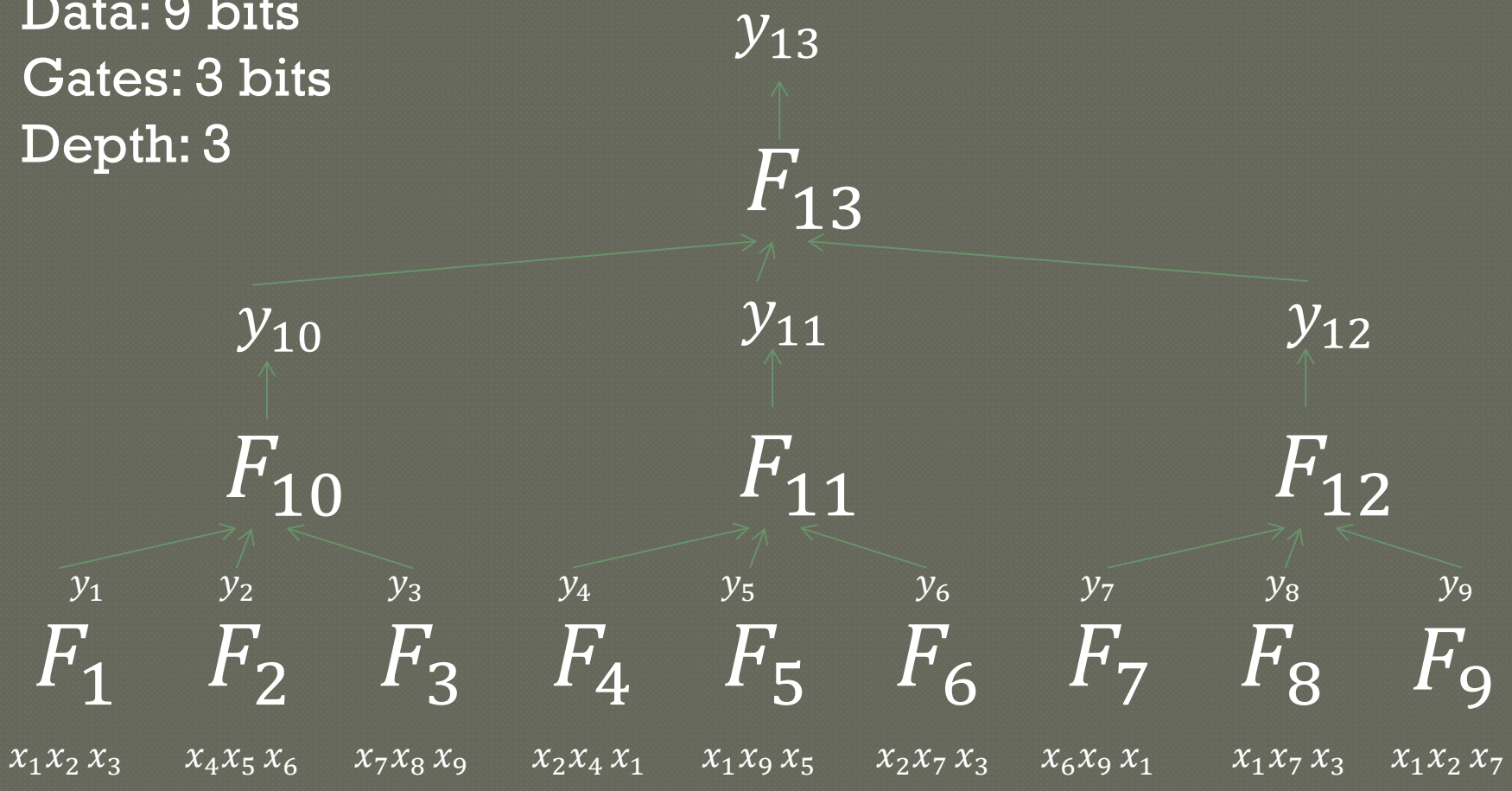
Data vectors and solution vector

- n is the number of training elements
- v is the solution vector
- $\{d_1, \overline{d_1}\}$

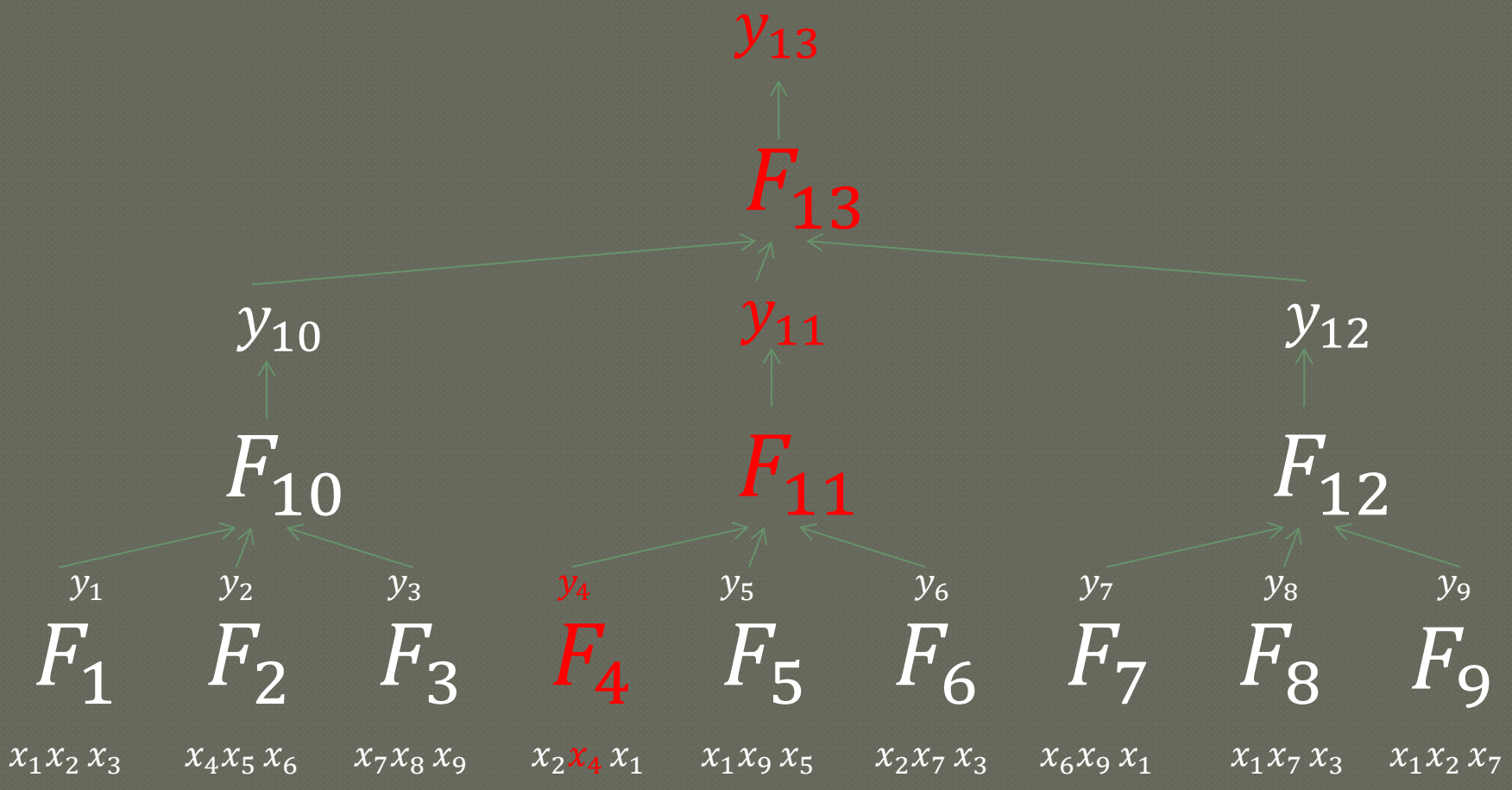
- $S_h(\{d_1, \overline{d_1}\}) = p\ H\left(\frac{\#(d_1 v)}{pn}\right) + (1-p)H\left(\frac{\#(\overline{d_1}\bar{v})}{(1-p)n}\right)$
  $=$ average information gain where
  $$p = \#(d_1 = 1)/\mathrm{n}$$

Function tree

Data: 9 bits
Gates: 3 bits
Depth: 3



Gradient decent



Benchmarks

An Empirical Evaluation of Deep Architectures on Problems with Many Factors of Variation

Larochelle, H., Erhan, D., Courville, A., Bergstra, J., Bengio, Y.

ICML 2007

WEB

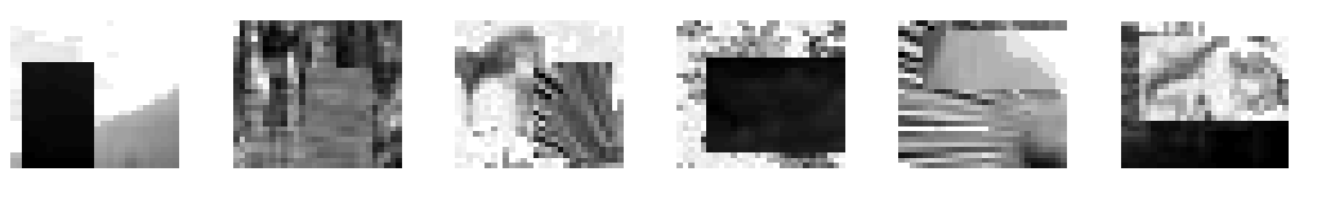
Convex

Tapp	SVM RBF	SVM Poly	NNet	DBN-3	SAA-3	DBN-1
17.2	19.1	19.8	32.3	18.6	18.4	19.9

Training size: 8000
Test size: 50000
Error ± 0.35

First level: Bit= 4, Depth= 4, Iterations= 204800
Second level Bit= 4, Depth= 5
Train: 6.48 Test: **17.22** ± 0.33
Time: 3h 10m

Rectangles images



Tapp	SVM RBF	SVM Poly	NNet	DBN-3	SAA-3	DBN-1
23.0	24.0	24.0	33.2	22.5	24.1	23.7

Training size: 12000
Test size: 50000
Bits per color: 2
Error ± 0.37

First level: Bit= 4, Depth= 4 Iterations= 25600
Second level: Bit= 4, Depth= 6
Train: 9.96 Test: **23.00** ± 0.37
Time: 2h 19m

Demo

- CIFAR 10 (2,1), 50k, 10k, (4,5)(4,5)
- CLOUD (Max prob, Max info)

Lots of possible improvements

- Only select input bits that convey information
- Explore multilevel
- Use topology of data
- Remove similar functions
- Test more appropriate encoding
- Explore multi class
- Preprocessing
- Incorporate unsupervised learning

Non supervised

- An interesting function is a function that conveys information on the data but that would not do so on random input.
- Similar to a compressed representation but fundamentally different.

Data

Zero One Random