COGS 118A, Winter 2020

Supervised Machine Learning Algorithms

Lecture 15: Ensemble Methods

Zhuowen Tu

Decision Tree Classifier

http://www.r2d3.us/visual-intro-to-machine-learning-part-1/

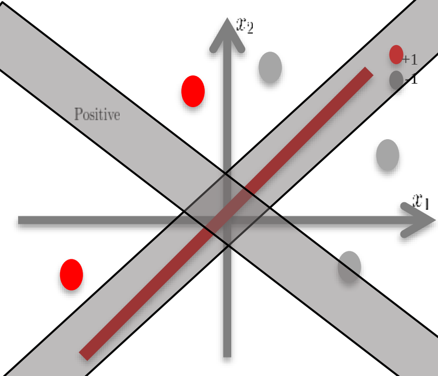
http://www.r2d3.us/visual-intro-to-machine-learning-part-2/

The main difference with the previous classifiers.

Training

Training: Minimize
$$\mathcal{L}(\mathbf{w}, b) = \sum_i max(0, -y_i(\mathbf{w}^T\mathbf{x}_i + b))$$

Testing

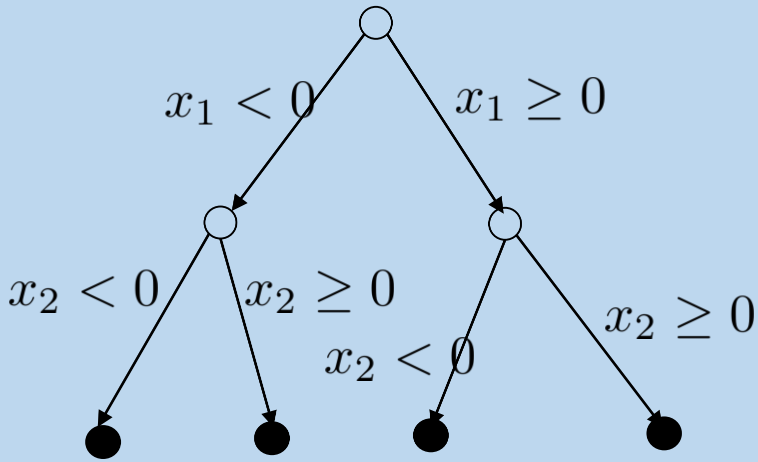


Test: $f(\mathbf{x}; \mathbf{w}, b) = \begin{cases} +1 & if\ \mathbf{w}^T\mathbf{x} + b \geq 0 \\ -1 & otherwise \end{cases}$

Training

Training: Minimize
an objective function that is recursively defined
for splitting.
No explicit error/loss is minimized here!

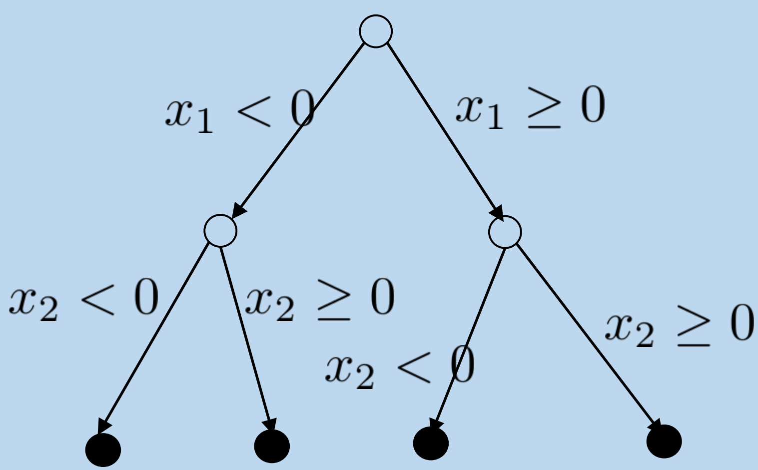
Decision Tree



Testing

The prediction is obtained by running a sequence decisions to go to the leaf node to obtain the classification.

Decision Tree



Why are we NOT able to define an explicit loss function to minimize like in Perceptron, Logisitic Regression, and SVM?

A. To complex to define.

B. It's a recursive function that has no intermediate loss.

C. The tree depths are not fixed.

D. This is a clustering task that is not suitable for classification.

E. None of the above.

Is the decision tree classifier a parametric model?

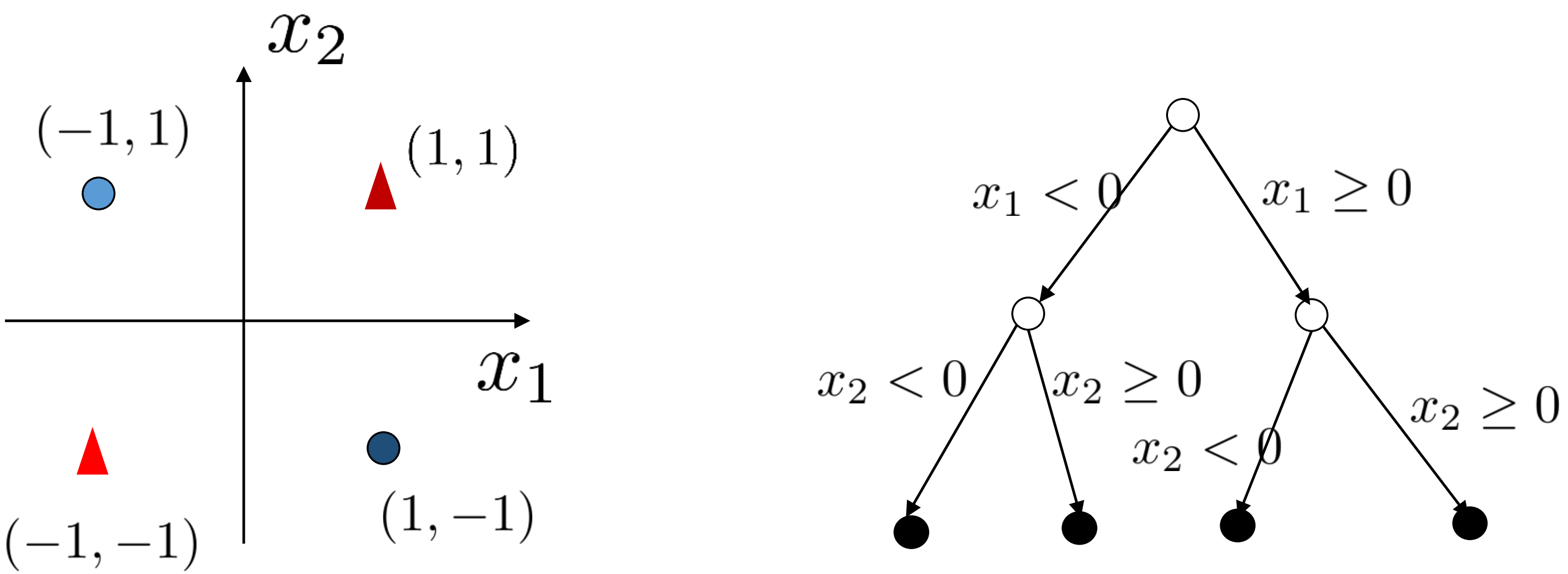
A. Yes

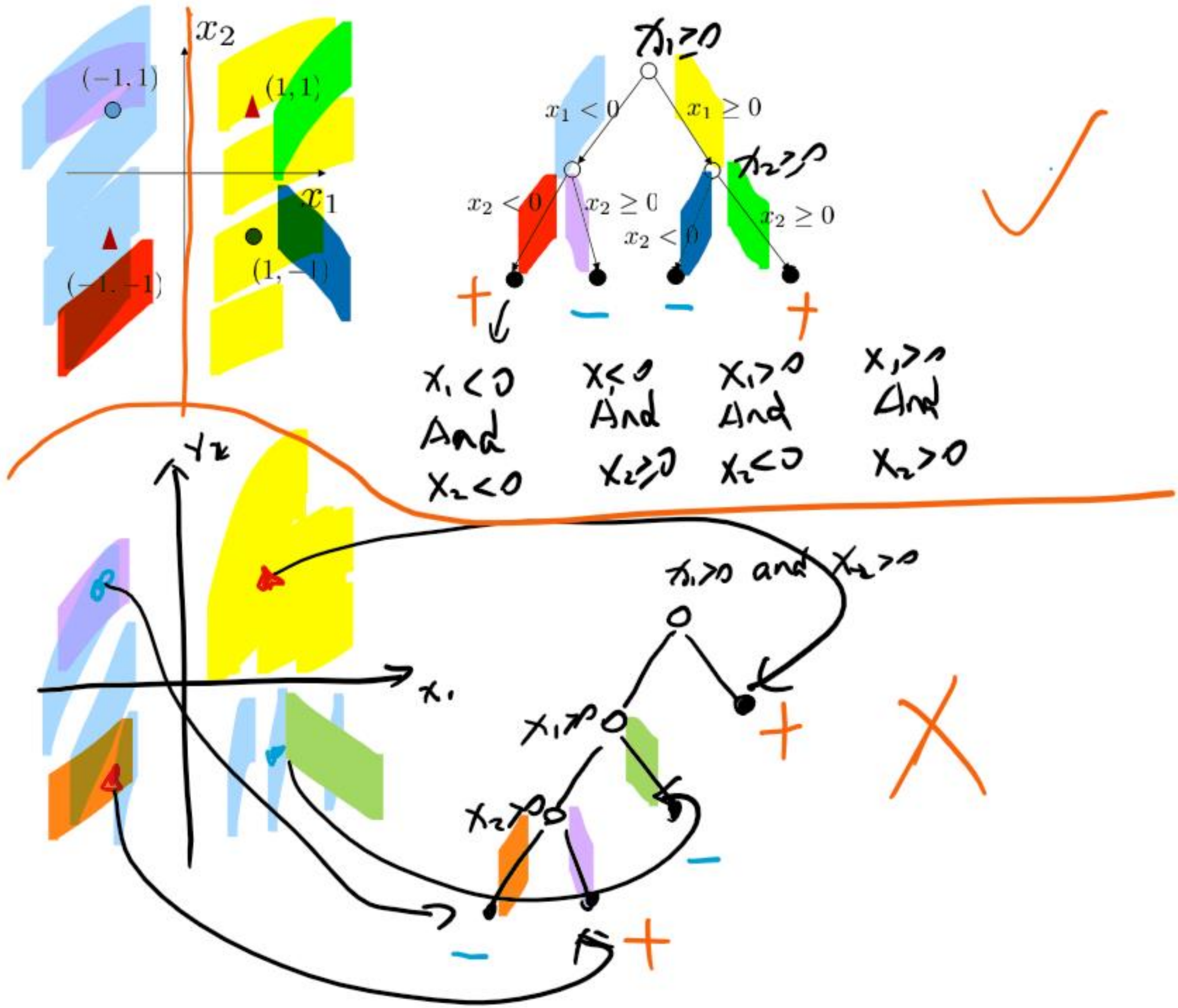
B. In general, no.

C. It depends.

The leaf node of the decision tree classifier typicaly stores the class-labels of the training samples. The depth and the number of the leaf nodes increase when having more training data.

Decision Tree for XOR





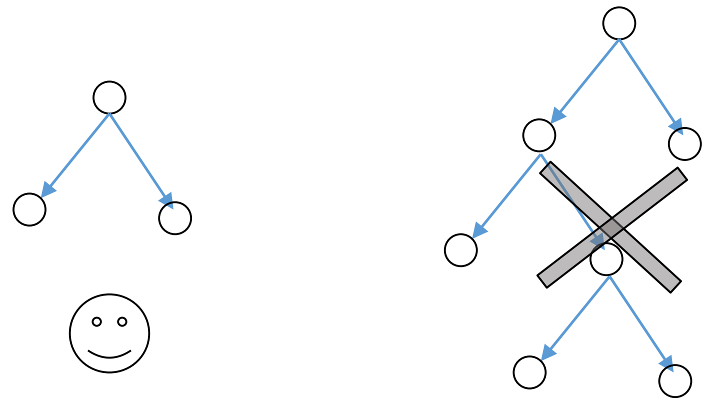
x_2
$(-1,1)$
$(1,1)$
x_1
$(-1,-1)$
$(1,-1)$
$x_1 \geq 0$
$x_1 < 0$
$x_1 \geq 0$
$x_2 \geq 0$
$x_2 < 0$
$x_2 \geq 0$
$x_2 < 0$
$x_2 \geq 0$
+
−
−
+
$x_1 < 0$ And $x_2 < 0$
$x_1 < 0$ And $x_2 \geq 0$
$x_1 > 0$ And $x_2 < 0$
$x_1 > 0$ And $x_2 > 0$
x_2
x_1
$x_1 > 0$ and $x_2 > 0$
+
$x_1 > 0$
−
$x_2 > 0$
−
+

A rule of thumb when constructing a decision tree classifier

What is a good decision tree classifier?

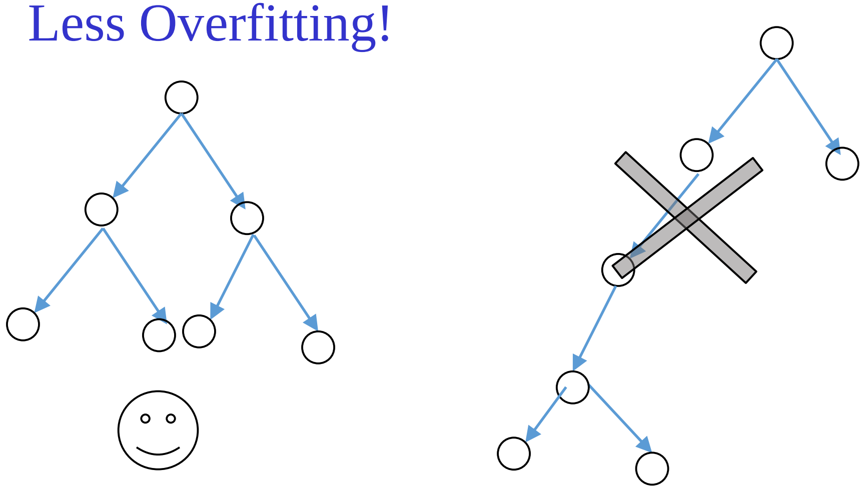
1. For the same training error, a shallow tree is more preferred than a deep tree.

Low Complexity!



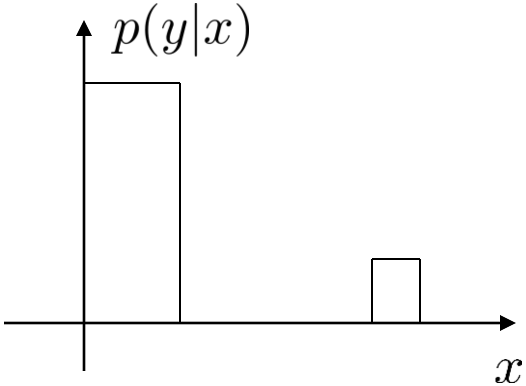
2. For the number of nodes, a balanced tree is more preferred than an unbalanced tree.

Less Overfitting!

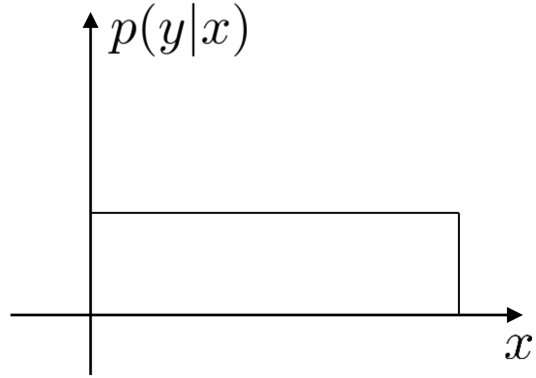


What is entropy?

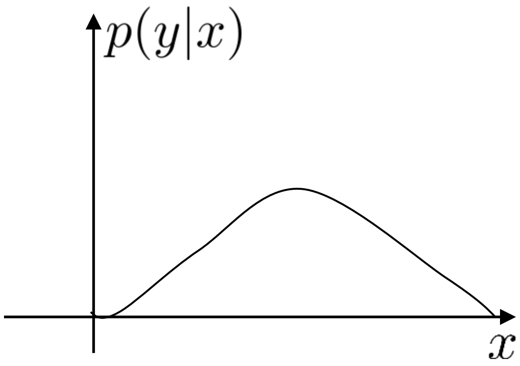
It is an uncertainty measure.



A.



B.



C.

Which one has the lowest entropy (uncertainty)?

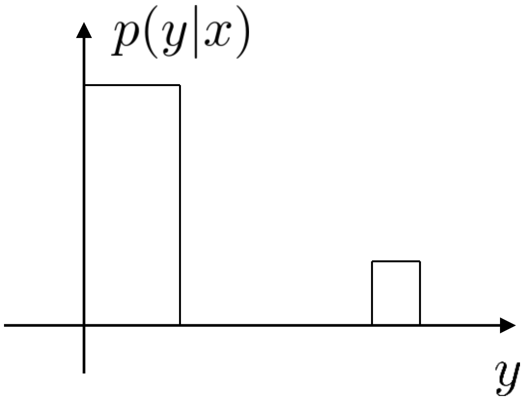
A.

B.

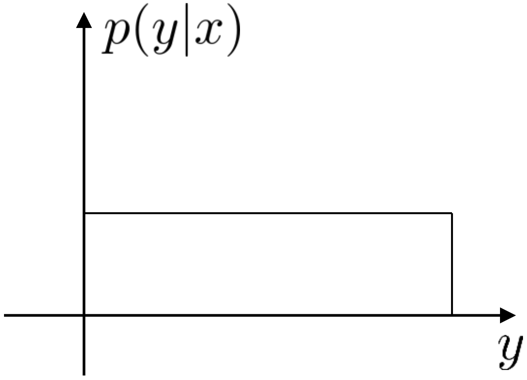
C.

What is entropy?

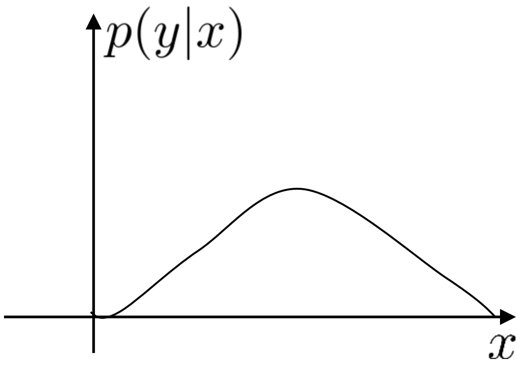
It is an uncertainty measure.



A.



B.



C.

Which one has the lowest entropy (uncertainty)?

A.

B.

C.

Entropy

General measure for knowing the underlying uncertainty of a random variable.

Discrete random variable:

$$H(X) = -\sum_i P(X = x_i) \log P(X = x_i)$$

Continuous random variable:

$$H(X) = -\int p(x) \log p(x) dx$$

0.5	**0.5**

$$H(X) = -(0.5 \times \log 0.5 + 0.5 \times \log 0.5) \approx 0.30$$

0.9	**0.1**

$$H(X) = -(0.9 \times \log 0.9 + 0.1 \times \log 0.1) \approx 0.14$$

0.1	**0.9**

$$H(X) = -(0.9 \times \log 0.9 + 0.1 \times \log 0.1) \approx 0.14$$

$\mathrm{Entropy}(P_1(Y|X))$

$||$

$\mathrm{Entropy}(P_2(Y|X))$

Joint entropy and mutual information

Joint entropy

Discrete random variable:

$$H(X,Y) = -\sum_{i,j} P(X=x_i, Y=y_j)\log P(X=x_i, Y=y_j)$$

Continuous random variable:

$$H(X,Y) = -\int\int p(x,y)\log p(x,y)dxdy$$

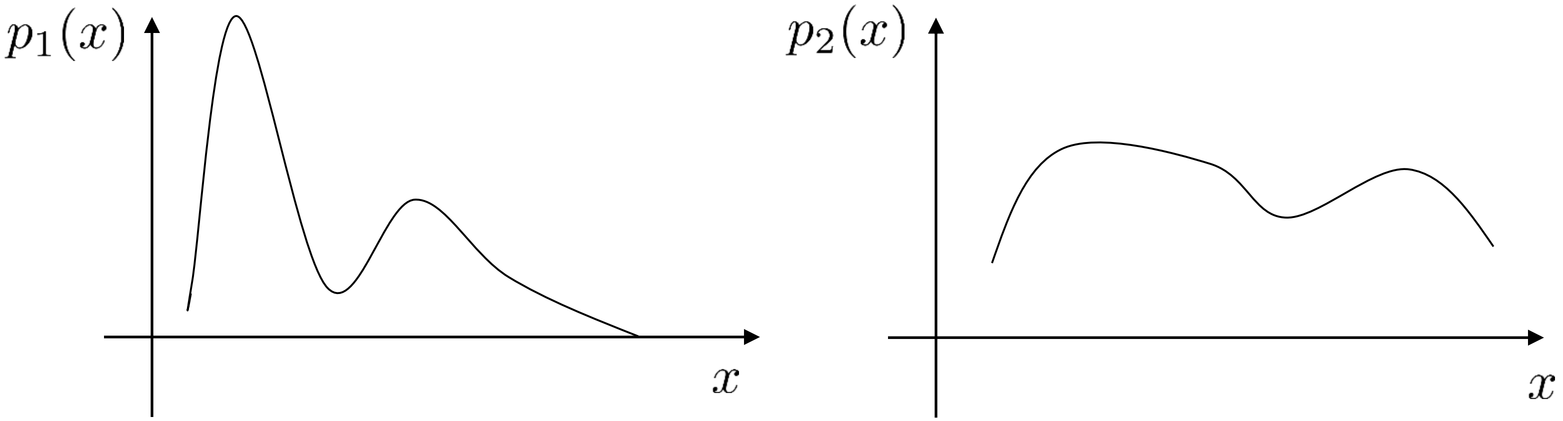
$X : 1 \quad 2$

$Y :$

1	0.2	0.4
2	0.35	0.05

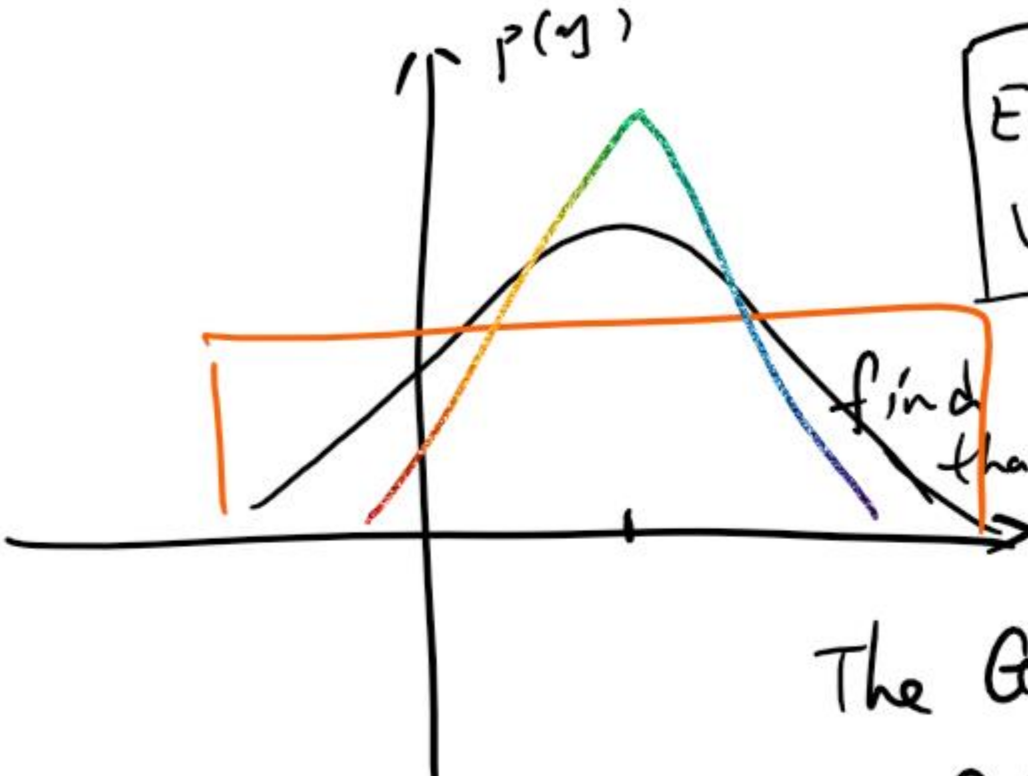
$$H(X,Y) = -[0.2\log 0.2 + 0.4\log 0.4 + 0.35\log 0.35 + 0.05\log 0.05]$$
$$= 1.2$$

Entropy



Entropy measures the amount of uncertainty for a probability distribution $p(x)$.

Or, how valuable it is if you would know x for under $p(x)$.



$E_Y(y) = \mu = 5.0$

$Var(y) = \sigma^2 = 3.0$

find a distribution that has a mean of μ variance of σ^2

The Gaussian Distribution

$G(\mu, \sigma^2)$

has the largest entropy

Joint entropy and mutual information

Conditional entropy:

$$H(Y|X) = \sum_i P(X = x_i) H(Y|X = x_i)$$

$$= -\sum_i P(X = x_i) [\sum_j P(Y = y_j|X = x_i) \log P(Y = y_j|X = x_i)]$$

Y : \ X :	1	2
1	0.2	0.4
2	0.35	0.05

$$P(Y = 1|X = 1) = 0.36, \; P(Y = 2|X = 1) = 0.64,$$

$$H(Y|X = 1) = -[0.36 \log 0.36 + 0.64 \log 0.64] = 0.66$$

$$P(Y = 1|X = 2) = 0.89, \; P(Y = 2|X = 2) = 0.11,$$

$$H(Y|X = 2) = -[0.89 \log 0.89 + 0.11 \log 0.11] = 0.35$$

$$H(Y|X) = 0.55 \times 0.66 + 0.45 \times 0.35 = 0.52$$

Carry averaged information between two random variables.

Relative entropy and mutual information

Relative entropy (Kullback-Leibler divergence), discrete random variable:

$$D(P||Q) = \sum_i P(X = x_i) \log \frac{P(X=x_i)}{Q(X=x_i)}$$

Relative entropy (Kullback-Leibler divergence), continues random variable:

$$D(p||q) = \int p(x) \log \frac{p(x)}{q(x)}$$

Mutual information, discrete random variable: a distance between joint and product

$$I(X;Y) = \sum_i \sum_j P(X = x_i, Y = y_j) \log \frac{P(X=x_i, Y=y_j)}{P(X=x_i)P(Y=y_j)}$$

Mutual information, continues random variable: a distance between joint and product

$$I(X;Y) = \int \int p(x,y) \log \frac{p(x,y)}{p(x)p(y)} dxdy$$

$Y:$	$X:1$	2
1	0.45	0.05
2	0.05	0.45

$Y:$	$X:1$	2
1	0.05	0.45
2	0.45	0.05

$Y:$	$X:1$	2
1	0.25	0.25
2	0.25	0.25

Why mutual information?

Mutual information, discrete random variable: a distance between joint and product

$$I(X;Y) = \sum_i \sum_j P(X = x_i, Y = y_j) \log \frac{P(X=x_i, Y=y_j)}{P(X=x_i)P(Y=y_j)}$$

Y : \ X :	1	2
1	0.45	0.05
2	0.05	0.45

$P(X = 1) = 0.5,\ P(X = 2) = 0.5$
$P(Y = 1) = 0.5,\ P(Y = 2) = 0.5$

0.37

Y : \ X :	1	2
1	0.05	0.45
2	0.45	0.05

$P(X = 1) = 0.5,\ P(X = 2) = 0.5$
$P(Y = 1) = 0.5,\ P(Y = 2) = 0.5$

0.37

Y : \ X :	1	2
1	0.25	0.25
2	0.25	0.25

$P(X = 1) = 0.5,\ P(X = 2) = 0.5$
$P(Y = 1) = 0.5,\ P(Y = 2) = 0.5$

0

We care about the information that exists in one random varilable that can be leveraged to predict the other random varilable.

Training C4.5 algorithm (J. Quinlan)

1. Tree construction (divide-and-conquer).

2. Tree pruning (in a way cross-validation to reduce generalization error).

Training C4.5 algorithm (J. Quinlan)

Hunt's method for constructing a decision tree from a set S of training samples. $\{C_1, C_2, ..., C_k\}$

There are three possibilities:

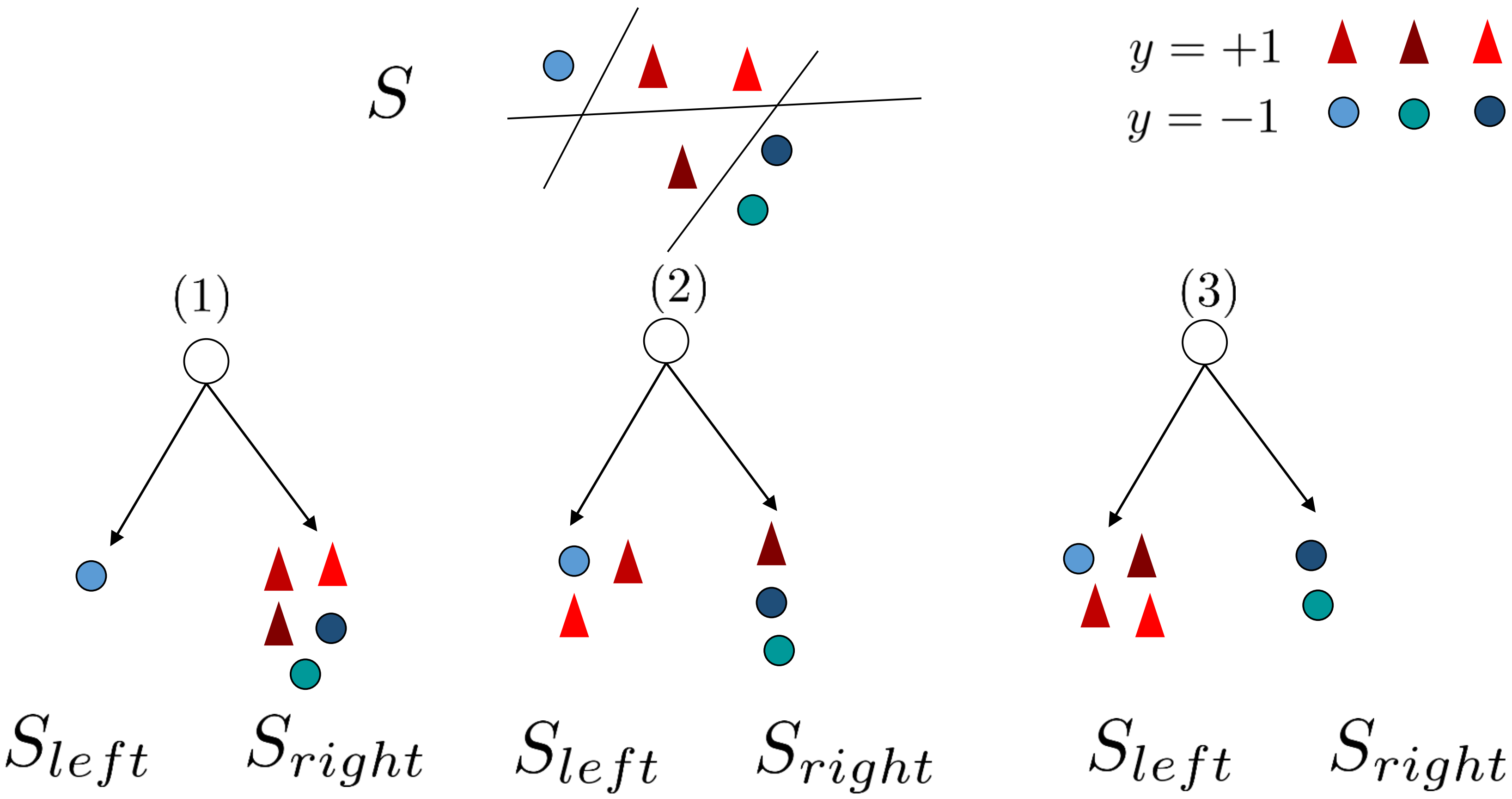
(1) S contains one or more samples that all belong to a single class. C_j

(2) S contains no samples.

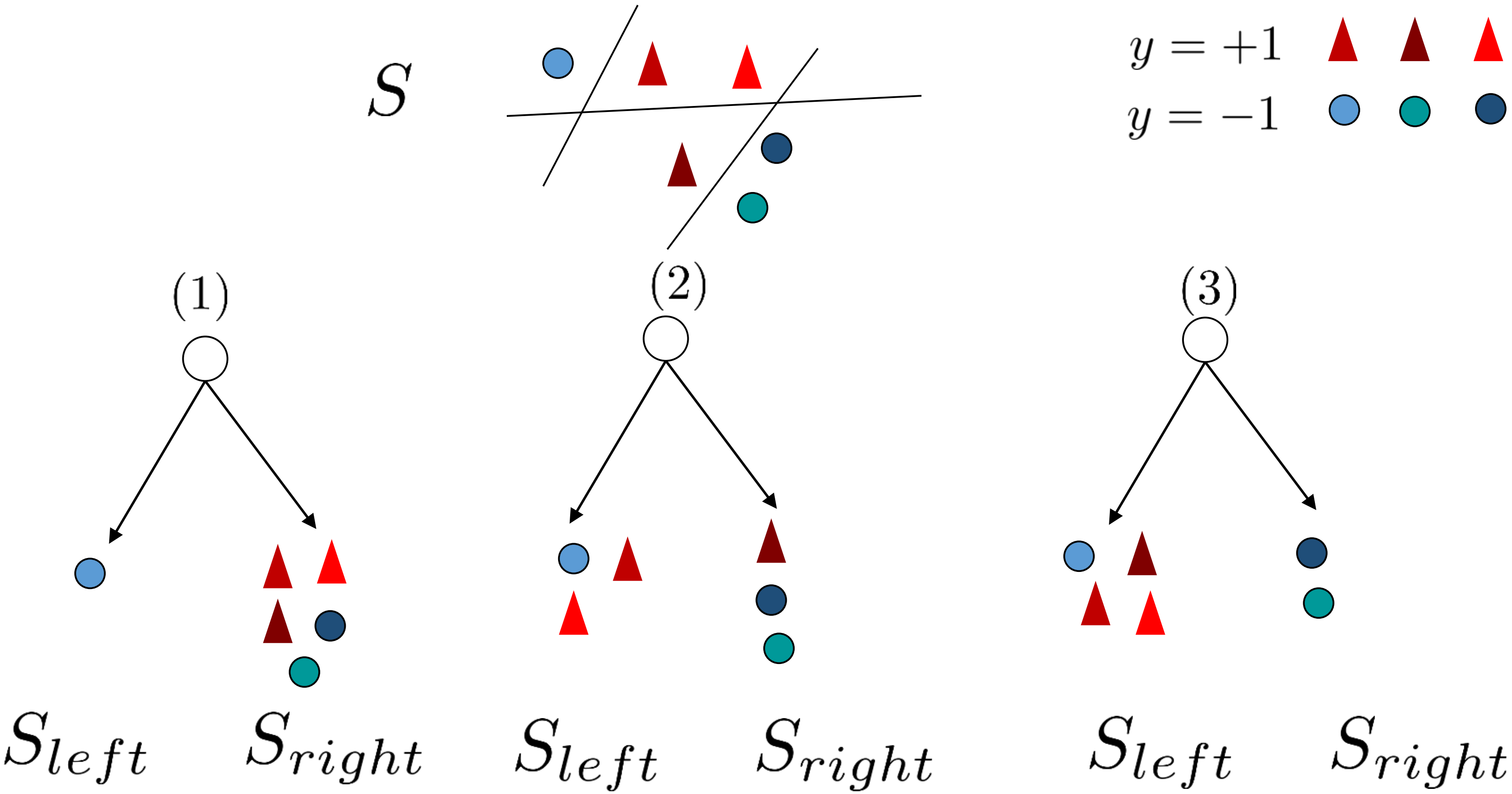
(3) S contains samples that belong to a mixture of classes.

Tree construction (J. Quinlan)

We recursively construct a tree each time to find the feature at a particular value to maximize the gain (minimize the cost).



Tree construction (J. Quinlan)



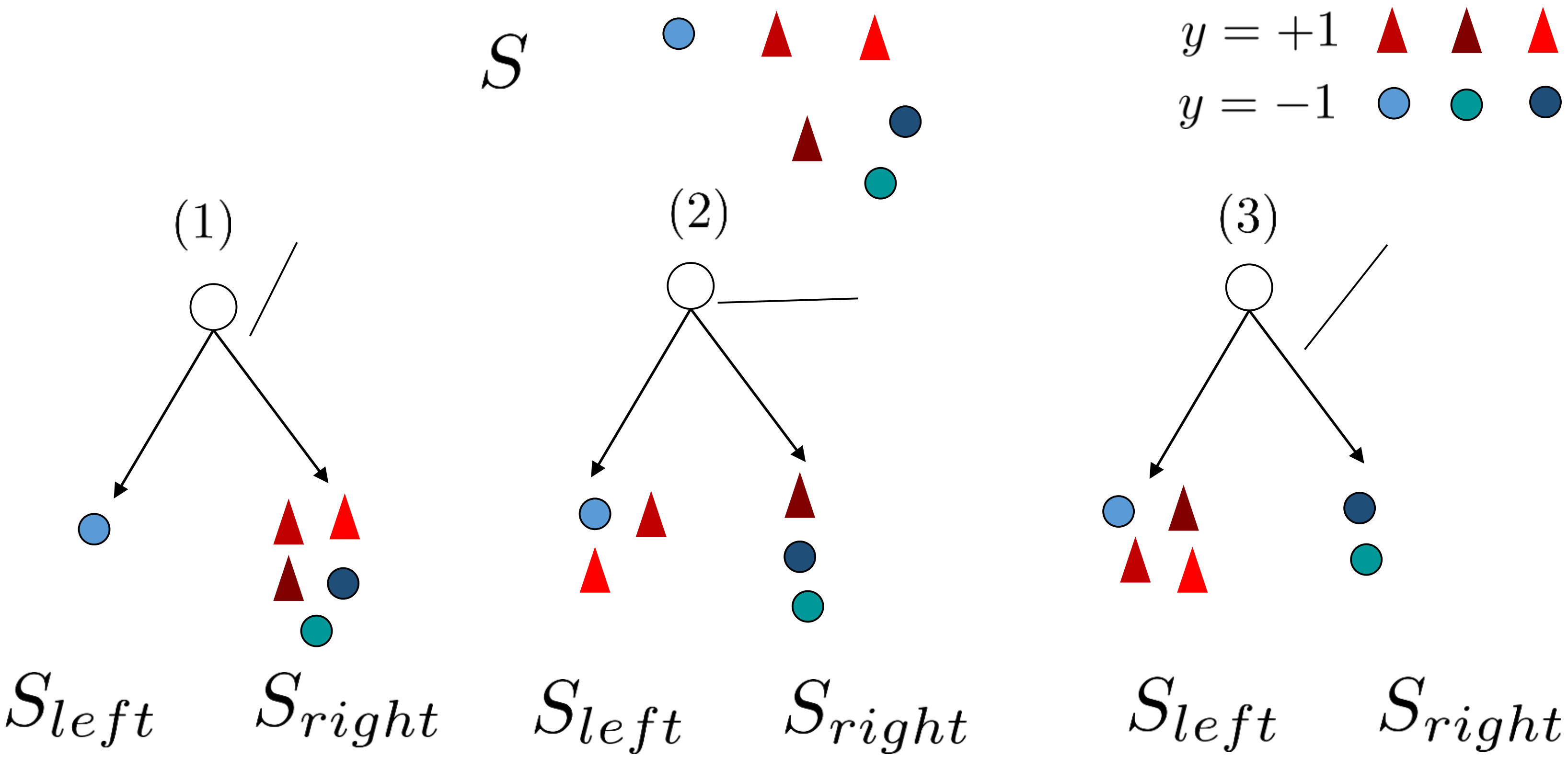
Which one to use:

A: (1)

B: (2)

C: (3)

Tree construction (J. Quinlan)

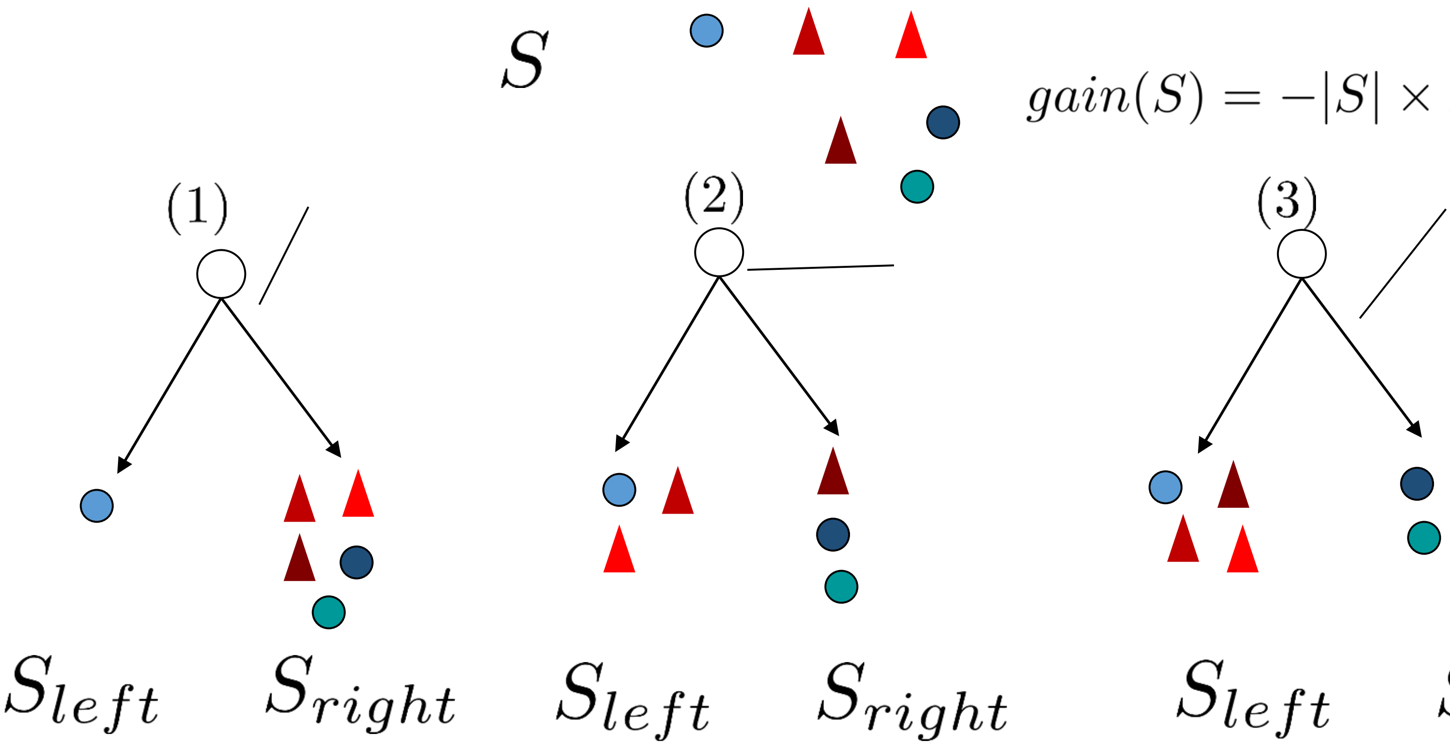


$$f^* = \arg max_f \quad gain(S_{left}^{(f)}) + gain(S_{right}^{(f)}) - gain(S)$$

$$gain(S) = -|S| \times Entropy(Y_S)$$

Tree construction (J. Quinlan)

$$gain(S) = -|S| \times Entropy(Y_S)$$



(1)
$$gain(S_{left}) = 1 \times (1 \times \log(1) + 0 \times \log(0) = 0$$
$$gain(S_{right}) = 5 \times (0.4 \times \log(0.4) + 0.6 \times \log(0.6) = -3.365$$
$$0 - 3.365 = -3.365$$

(2)
$$gain(S_{left}) = 3 \times (0.33 \times \log(0.33) + 0.67 \times \log(0.67) = -1.9095$$
$$gain(S_{right}) = 3 \times (0.67 \times \log(0.67) + 0.33 \times \log(0.33) = -1.9095$$
$$-1.9095 - 1.9095 = -3.819$$

(3)
$$gain(S_{left}) = 4 \times (0.25 \times \log(0.25) + 0.75 \times \log(0.75) = -2.2493$$
$$gain(S_{right}) = 2 \times (0 \times \log(0) + 1 \times \log(1) = 0$$
$$-2.2493 + 0 = -2.2493$$

Joint entropy and mutual information

Joint entropy

Discrete random variable:

$$H(X,Y) = -\sum_{i,j} P(X=x_i, Y=y_j) \log P(X=x_i, Y=y_j)$$

$$= E\left[\begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}\right] = 0$$

Continuous random variable:

$$H(X,Y) = -\int\int p(x,y) \log p(x,y) dx dy$$

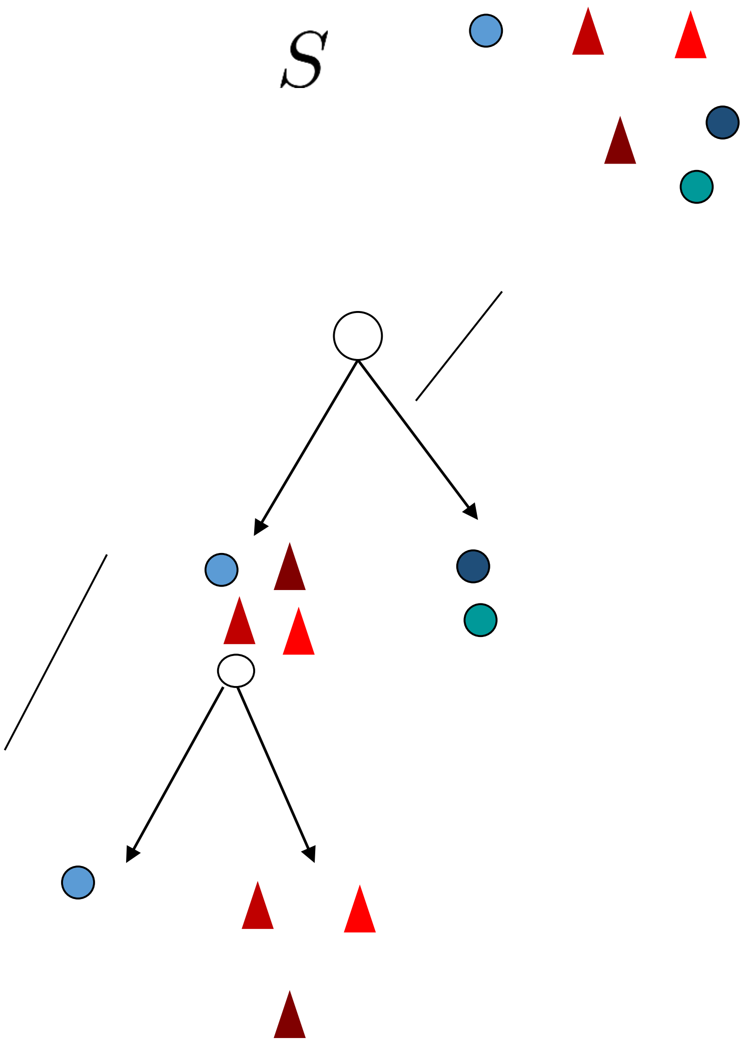
$$E\left[\begin{bmatrix} 0.2 & 0.05 \\ 0.05 & 0.7 \end{bmatrix}\right] = E\left[\begin{bmatrix} 0.05 & 0.2 \\ 0.05 & 0.7 \end{bmatrix}\right] = E\left[\begin{bmatrix} 0.7 & 0.2 \\ 0.05 & 0.05 \end{bmatrix}\right]$$

Y : \ X :	1	2
1	0.2	0.4
2	0.35	0.05

$$H(X,Y) = -[0.2 \log 0.2 + 0.4 \log 0.4 + 0.35 \log 0.35 + 0.05 \log 0.05]$$

$$= 1.2$$

Tree construction (J. Quinlan)



$$gain(S) = -|S| \times Entropy(Y_S)$$

Pruning decision trees

- Discarding one or more subtrees and replacing them with leaves simplify decision tree and that is the main task in decision tree pruning:
  - Prepruning
  - Postpruning
- C4.5 follows a postpruning approach (*pessimistic pruning)*.

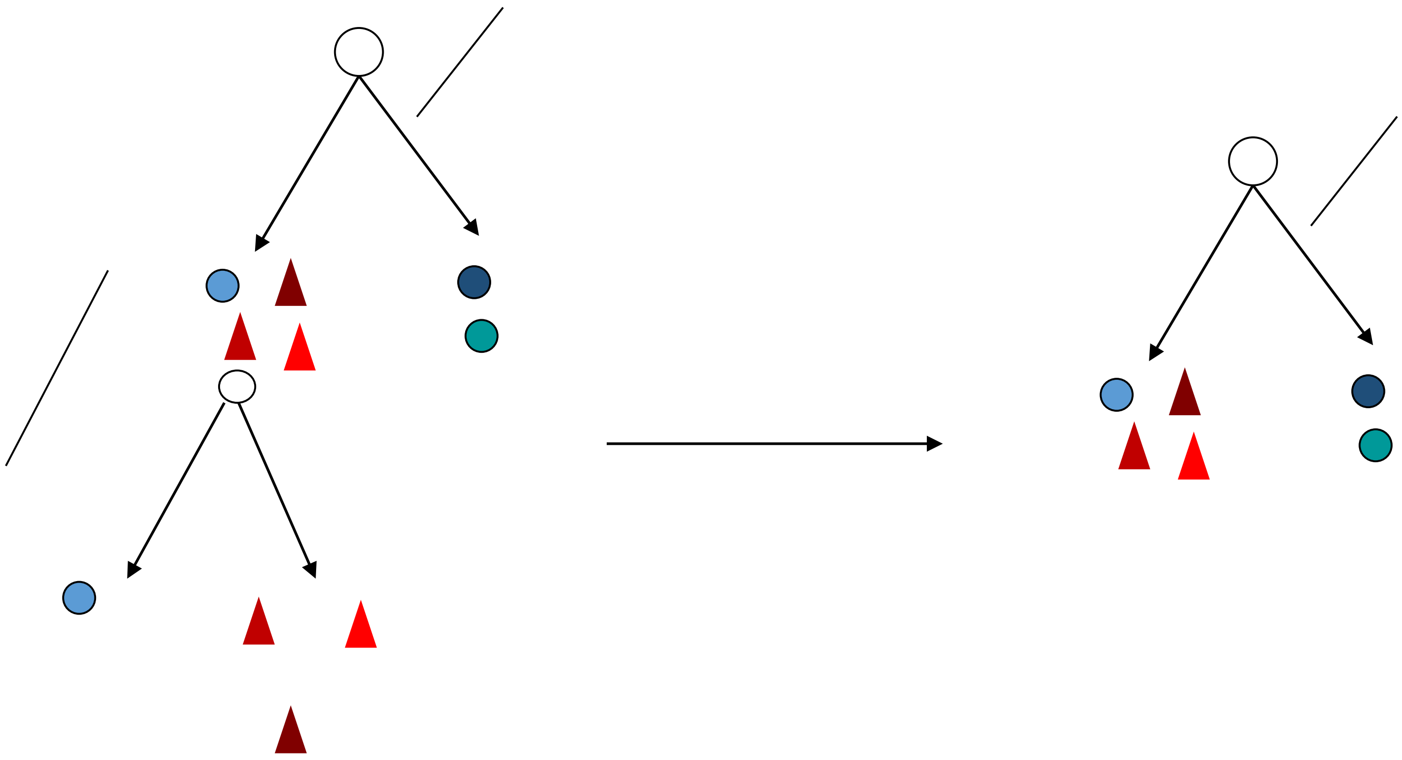
Prepruning

Deciding not to divide a set of samples any further under some conditions. The stopping criterion is usually based on some statistical test, such as the χ^2-test.

Postpruning

Removing retrospectively some of the tree structure using selected accuracy criteria.

Tree pruning (J. Quinlan)



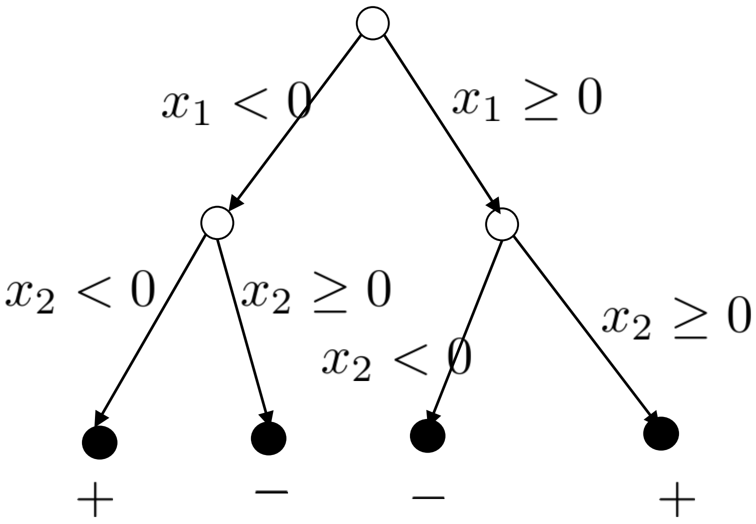
Decision Tree

The general rule is: **divide-and-conquer**

Decision node:○ decision to which path to pass the data.

Leaf (end) node:● which class (or class probability)

$$p(y|x)$$



C4.5 (J. Quinlan)

1.1 EXAMPLE: LABOR NEGOTIATION SETTLEMENTS

good, bad.	
duration:	continuous.
wage increase first year:	continuous.
wage increase second year:	continuous.
wage increase third year:	continuous.
cost of living adjustment:	none, tcf, tc.
working hours:	continuous.
pension:	none, ret_allw, empl_contr.
standby pay:	continuous.
shift differential:	continuous.
education allowance:	yes, no.
statutory holidays:	continuous.
vacation:	below average, average, generous.
longterm disability assistance:	yes, no.
contribution to dental plan:	none, half, full.
bereavement assistance:	yes, no.
contribution to health plan:	none, half, full.

Figure 1–1. File defining labor-neg classes and attributes

if wage increase first year $\leq$ 2.5 *then*
    if working hours $\leq$ 36 *then class good*
    else if working hours $>$ 36 *then*
        if contribution to health plan is none then class bad
        else if contribution to health plan is half then class good
        else if contribution to health plan is full then class bad
else if wage increase first year $>$ 2.5 *then*
    if statutory holidays $>$ 10 *then class good*
    else if statutory holidays $\leq$ 10 *then*
        if wage increase first year $\leq$ 4 *then class bad*
        else if wage increase first year $>$ 4 *then class good*

C4.5 (J. Quinlan)

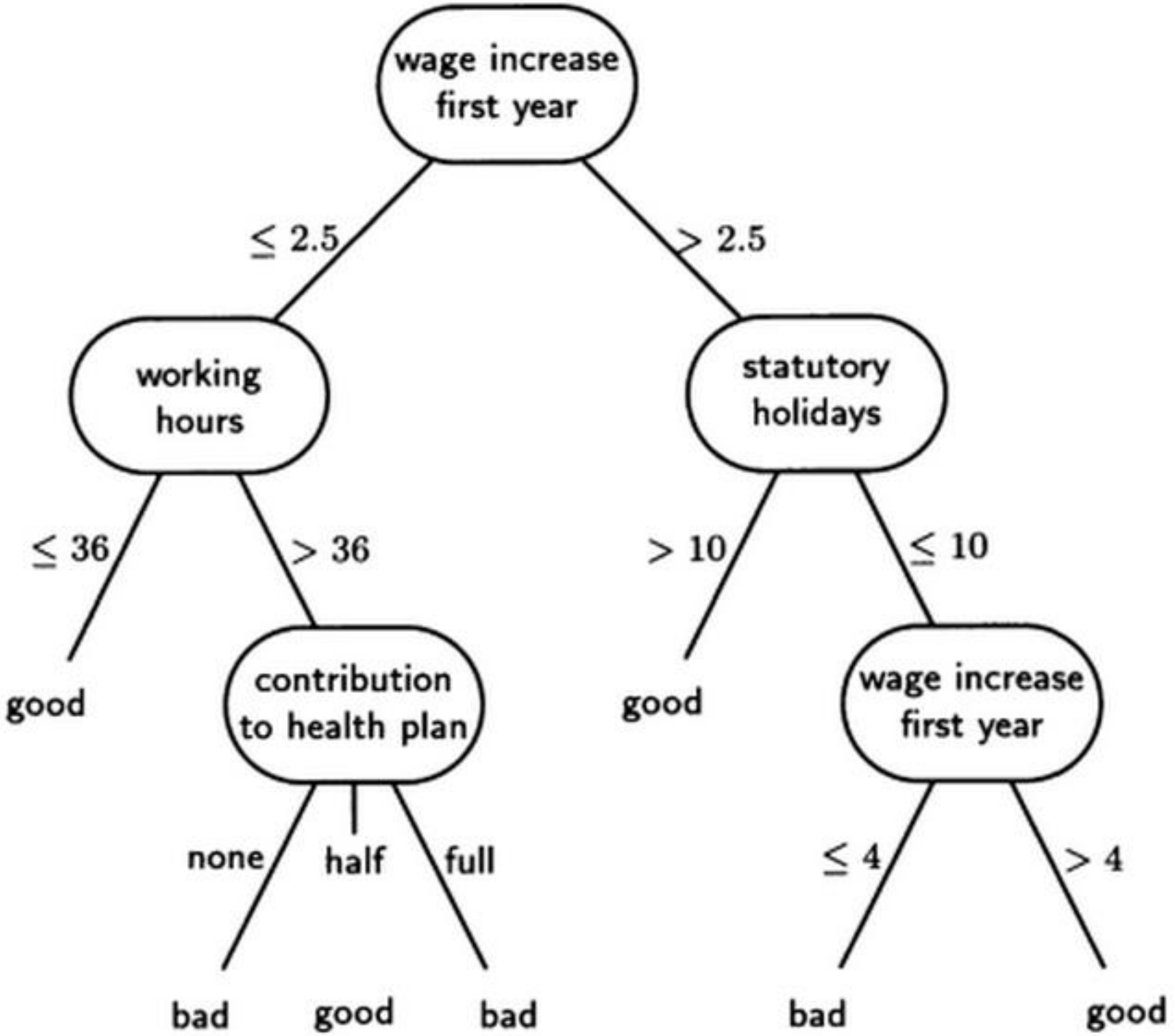


Figure 1–3. labor-neg decision tree in graph form

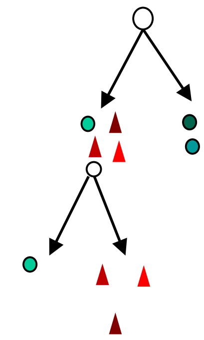
C4.5 (J. Quinlan)

1.1 EXAMPLE: LABOR NEGOTIATION SETTLEMENTS

```
C4.5 [release 5] decision tree generator        Fri Dec 6 13:33:54 1991

    Options:
        File stem <labor-neg>
        Trees evaluated on unseen cases

Read 40 cases (16 attributes) from labor-neg.data

Decision Tree:

wage increase first year ≤ 2.5 :
|   working hours ≤ 36 : good (2.0/1.0)
|   working hours > 36 :
|   |   contribution to health plan = none: bad (5.1)
|   |   contribution to health plan = half: good (0.4/0.0)
|   |   contribution to health plan = full: bad (3.8)
wage increase first year > 2.5 :
|   statutory holidays > 10 : good (21.2)
|   statutory holidays ≤ 10 :
|   |   wage increase first year ≤ 4 : bad (4.5/0.5)
|   |   wage increase first year > 4 : good (3.0)

Simplified Decision Tree:

wage increase first year ≤ 2.5 : bad (11.3/2.8)
wage increase first year > 2.5 :
|   statutory holidays > 10 : good (21.2/1.3)
|   statutory holidays ≤ 10 :
|   |   wage increase first year ≤ 4 : bad (4.5/1.7)
|   |   wage increase first year > 4 : good (3.0/1.1)

Tree saved

Evaluation on training data (40 items):

          Before Pruning          After Pruning
          --------------   -----------------------------
          Size   Errors     Size   Errors      Estimate

            12  1 ( 2.5%)      7  1 ( 2.5%)    (17.4%)   <<

Evaluation on test data (17 items):

          Before Pruning          After Pruning
          --------------   -----------------------------
          Size   Errors     Size   Errors      Estimate

            12  3 (17.6%)      7  3 (17.6%)    (17.4%)   <<

             (a)    (b)     <-classified as
            ----   ----
              10      1     (a): class good
               2      4     (b): class bad
```

Recap: Decision Tree

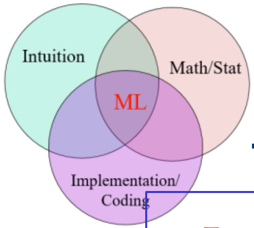
- Decision tree classifier is one of the most widely used classifiers in machine learning.
- It is a non-parametric model that can grow deep.
- Its key spirit is about divide-and-conquer.
- It has a nice balance between model complexity and classification power.
- It is often combined with other methods such as Boosting and Bagging to achieve enhanced performances.



Math:

$$f^* = \arg max_f \quad gain(S^{(f)}_{left}) + gain(S^{(f)}_{right}) - gain(S)$$

$$gain(S) = -|S| \times Entropy(Y_S)$$



Recap: Decision Tree

Implementation:

1.1 EXAMPLE: LABOR NEGOTIATION SETTLEMENTS

```
C4.5 [release 5] decision tree generator      Fri Dec 6 13:33:54 1991

    Options:
        File stem <labor-neg>
        Trees evaluated on unseen cases

Read 40 cases (16 attributes) from labor-neg.data

Decision Tree:

wage increase first year ≤ 2.5 :
|   working hours ≤ 36 : good (2.0/1.0)
|   working hours > 36 :
|   |   contribution to health plan = none: bad (5.1)
|   |   contribution to health plan = half: good (0.4/0.0)
|   |   contribution to health plan = full: bad (3.8)
wage increase first year > 2.5 :
|   statutory holidays > 10 : good (21.2)
|   statutory holidays ≤ 10 :
|   |   wage increase first year ≤ 4 : bad (4.5/0.5)
|   |   wage increase first year > 4 : good (3.0)

Simplified Decision Tree:

wage increase first year ≤ 2.5 : bad (11.3/2.8)
wage increase first year > 2.5 :
|   statutory holidays > 10 : good (21.2/1.3)
|   statutory holidays ≤ 10 :
|   |   wage increase first year ≤ 4 : bad (4.5/1.7)
|   |   wage increase first year > 4 : good (3.0/1.1)

Tree saved

Evaluation on training data (40 items):

           Before Pruning           After Pruning
          ----------------    ---------------------------
          Size    Errors      Size    Errors     Estimate

            12   1 ( 2.5%)       7   1 ( 2.5%)    (17.4%)   <<

Evaluation on test data (17 items):

           Before Pruning           After Pruning
          ----------------    ---------------------------
          Size    Errors      Size    Errors     Estimate

            12   3 (17.6%)       7   3 (17.6%)    (17.4%)   <<

             (a)   (b)        <-classified as
            ----  ----
              10     1        (a): class good
               2     4        (b): class bad
```

Ensemble Learning

Empirical Comparisons of Different Algorithms

Caruana and Niculesu-Mizil, ICML 2006

MODEL	1ST	2ND	3RD	4TH	5TH	6TH	7TH	8TH	9TH	10TH
BST-DT	0.580	0.228	0.160	0.023	0.009	0.000	0.000	0.000	0.000	0.000
RF	0.390	0.525	0.084	0.001	0.000	0.000	0.000	0.000	0.000	0.000
BAG-DT	0.030	0.232	0.571	0.150	0.017	0.000	0.000	0.000	0.000	0.000
SVM	0.000	0.008	0.148	0.574	0.240	0.029	0.001	0.000	0.000	0.000
ANN	0.000	0.007	0.035	0.230	0.606	0.122	0.000	0.000	0.000	0.000
KNN	0.000	0.000	0.000	0.009	0.114	0.592	0.245	0.038	0.002	0.000
BST-STMP	0.000	0.000	0.002	0.013	0.014	0.257	0.710	0.004	0.000	0.000
DT	0.000	0.000	0.000	0.000	0.000	0.000	0.004	0.616	0.291	0.089
LOGREG	0.000	0.000	0.000	0.000	0.000	0.000	0.040	0.312	0.423	0.225
NB	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.030	0.284	0.686

Overall rank by mean performance across problems and metrics (based on bootstrap analysis).

BST-DT: boosting with decision tree weak classifier

RF: random forest

BAG-DT: bagging with decision tree weak classifier

SVM: support vector machine

ANN: neural nets

KNN: k nearest neighboorhood

BST-STMP: boosting with decision stump weak classifier

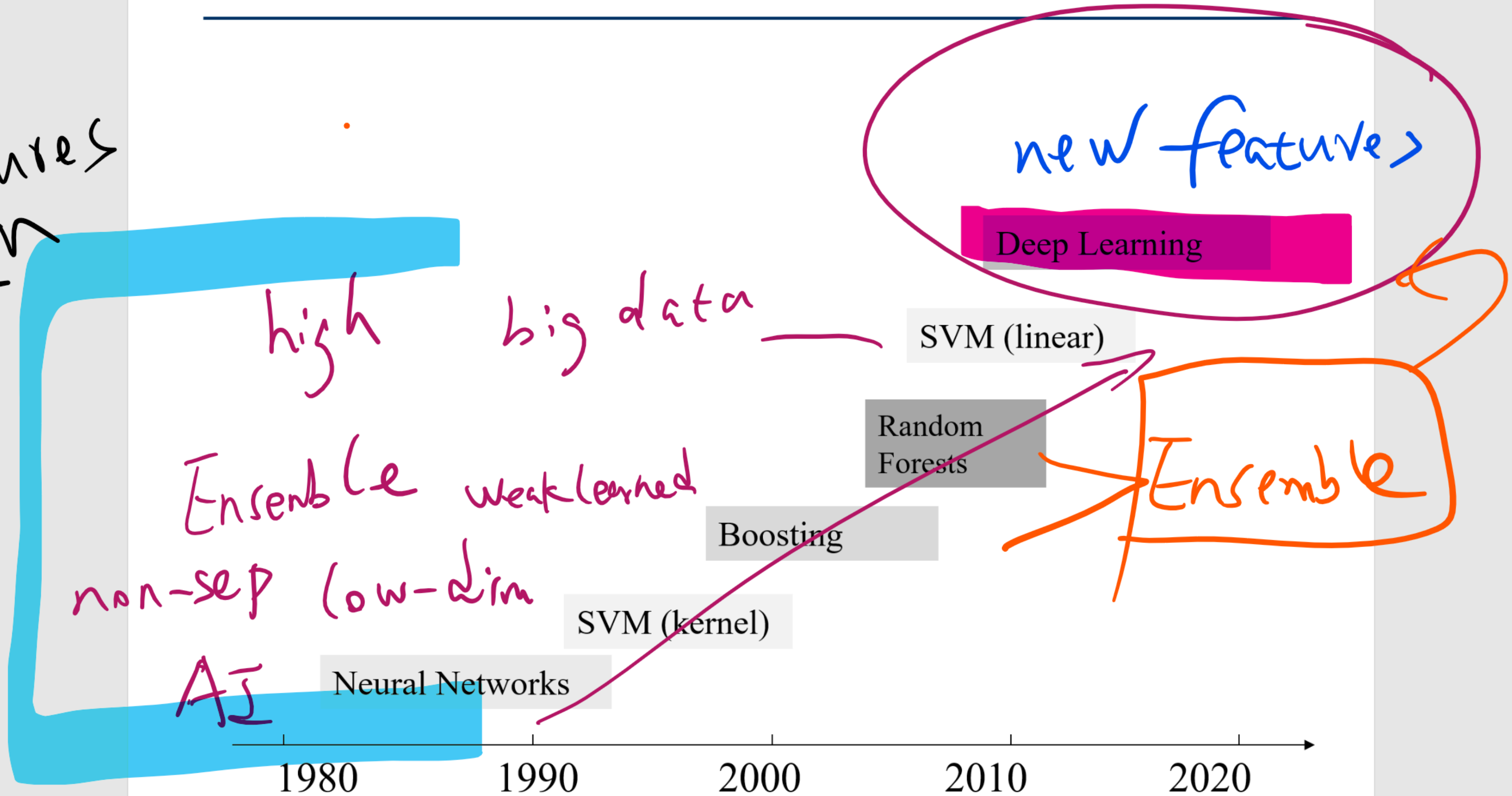
DT: decision tree

LOGREG: logistic regression

NB: naïve Bayesian

It is informative, but by no means final.

Trends of classification methods



features given

Trends of classification methods