

Machine learning algorithms: Practice 2.

Decision Tree Algorithm

Alexandr Gavrilko
MLA Course for CS-2227

Introduction to Decision Tree Algorithm

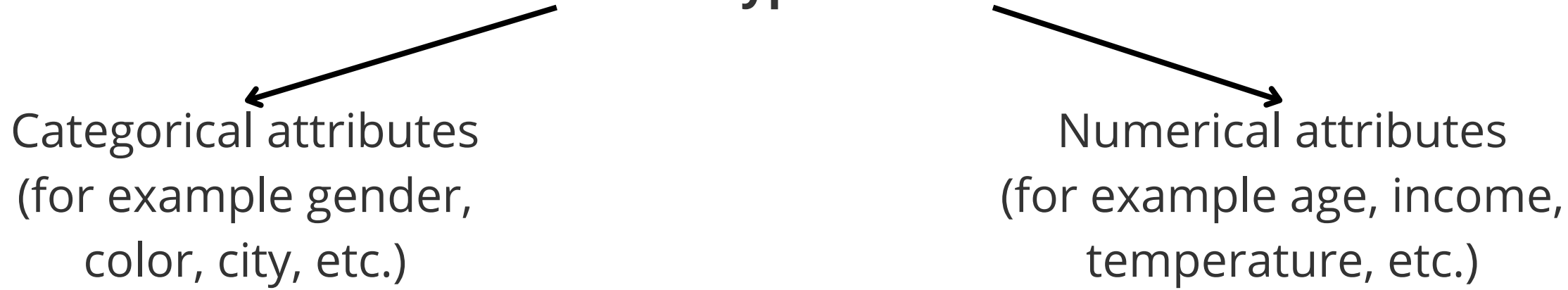
Decision Tree algorithm is a classification scheme which generates a tree and a set of rules from given data set. Algorithm can be used for both classification and regression tasks, but mostly for classification.

Main features of Decision Tree:

Eager learning algorithm - there is training phase, during which algorithm generate rules

Non-parametric algorithm - algorithm doesn't assume anything about training data

There are two types of attributes:



```
graph TD; A[There are two types of attributes:] --> B[Categorical attributes<br/>(for example gender,<br/>color, city, etc.)]; A --> C[Numerical attributes<br/>(for example age, income,<br/>temperature, etc.)];
```

Categorical attributes
(for example gender,
color, city, etc.)

Numerical attributes
(for example age, income,
temperature, etc.)

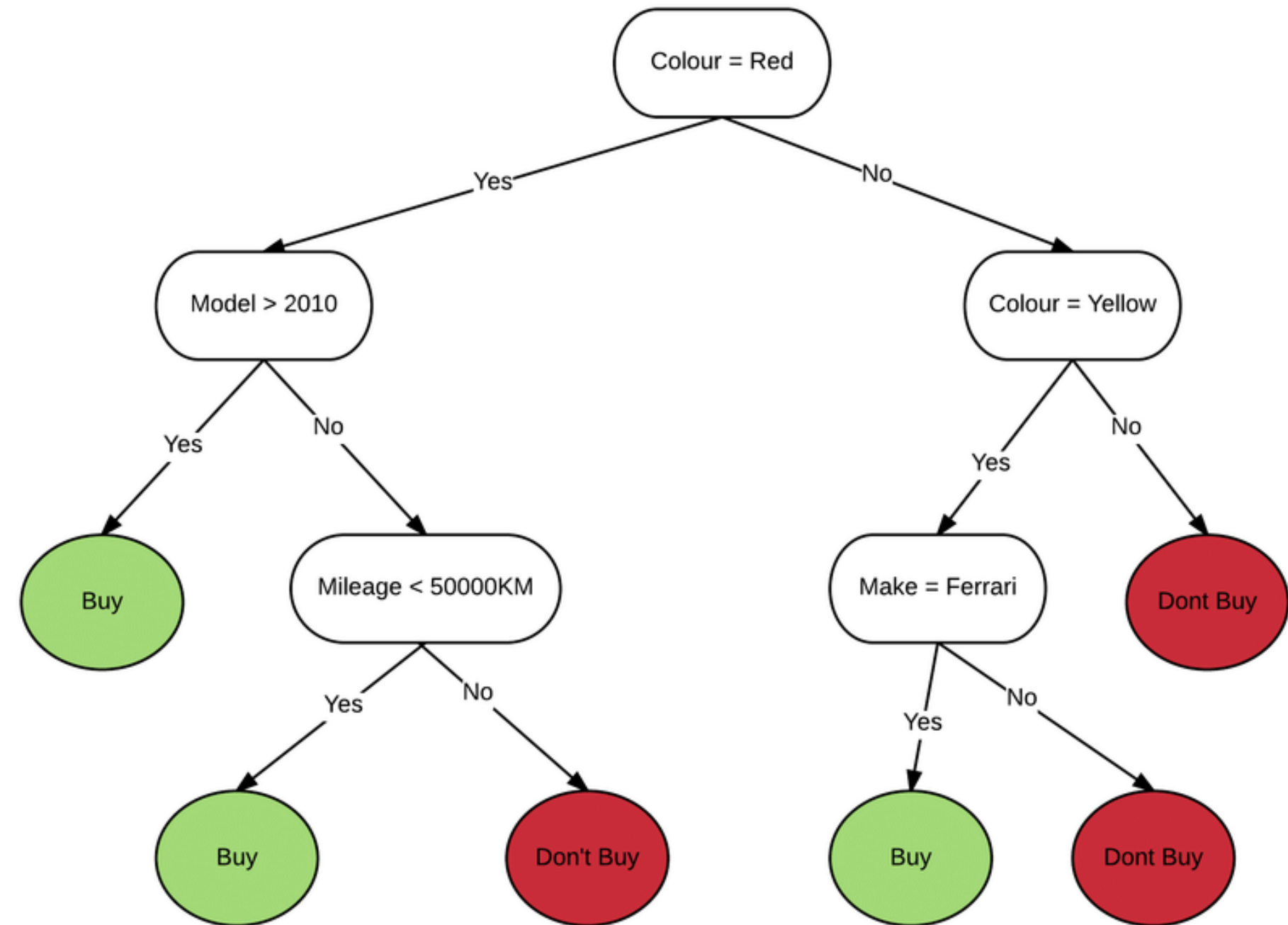
There are different methods for creating rules for different types of attributes

Decision tree has three following properties:

- An **inner node** representing **an attribute** and **a test** on the **attribute (rule)**.
Attribute with some condition is called **splitting attribute**.
- An **edge** representing a **flow** of decision-making process
- A **leaf** represents one of the classes

Construction of decision tree:

- is based on the train data
- is generated **from top to bottom**



Terminology for data set:

- There are **5 attributes**: outlook, temp, humidity, windy, class
- One attribute is **special**: class
- Temp and Humidity are **numerical attributes**
- Outlook and windy are **categorical attributes**

OUTLOOK	TEMP(F)	HUMIDITY(%)	WINDY	CLASS
sunny	79	90	true	no play
sunny	56	70	false	play
sunny	79	75	true	play
sunny	60	90	true	no play
overcast	88	88	false	no play
overcast	63	75	true	play
overcast	88	95	false	play
rain	78	60	false	play
rain	66	70	false	no play
rain	68	60	true	no play

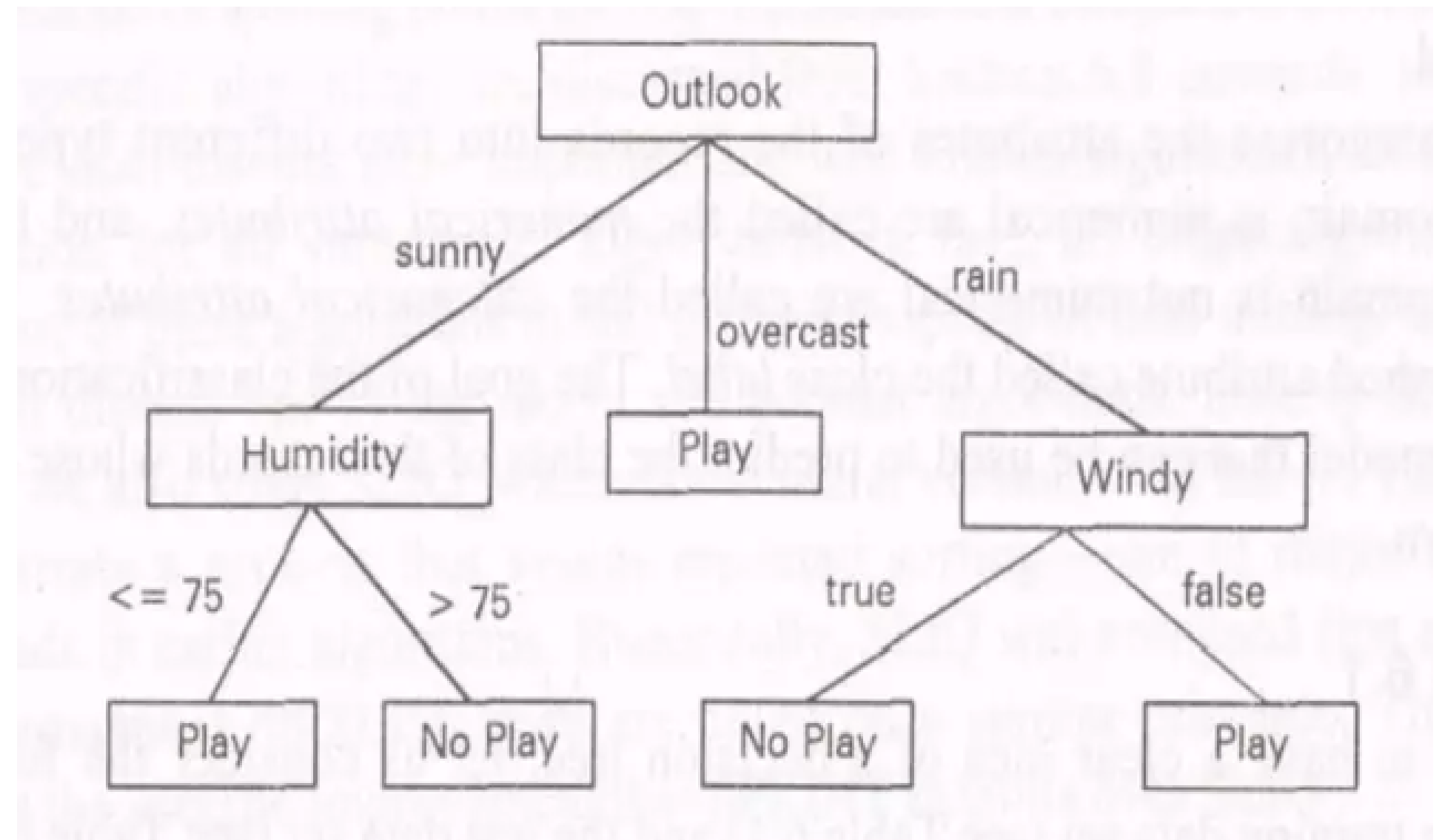
Based on the train data set we want to find a set of rules to know what values of attributes determine whether to play golf or not.

Description of rules for simple example of decision tree:

- **RULE 1:** If it is sunny and humidity ≤ 75 - play golf
- **RULE 2:** If it is sunny and humidity > 75 - do not play golf
- **RULE 3:** If it is overcast - play golf
- **RULE 4:** If it is rainy and windy - do not play golf
- **RULE 5:** If it is rainy and not windy - play golf

Try to predict class manually for the following vector:

- Outlook = rain
- Temp = 70
- Humidity = 65
- Windy = True

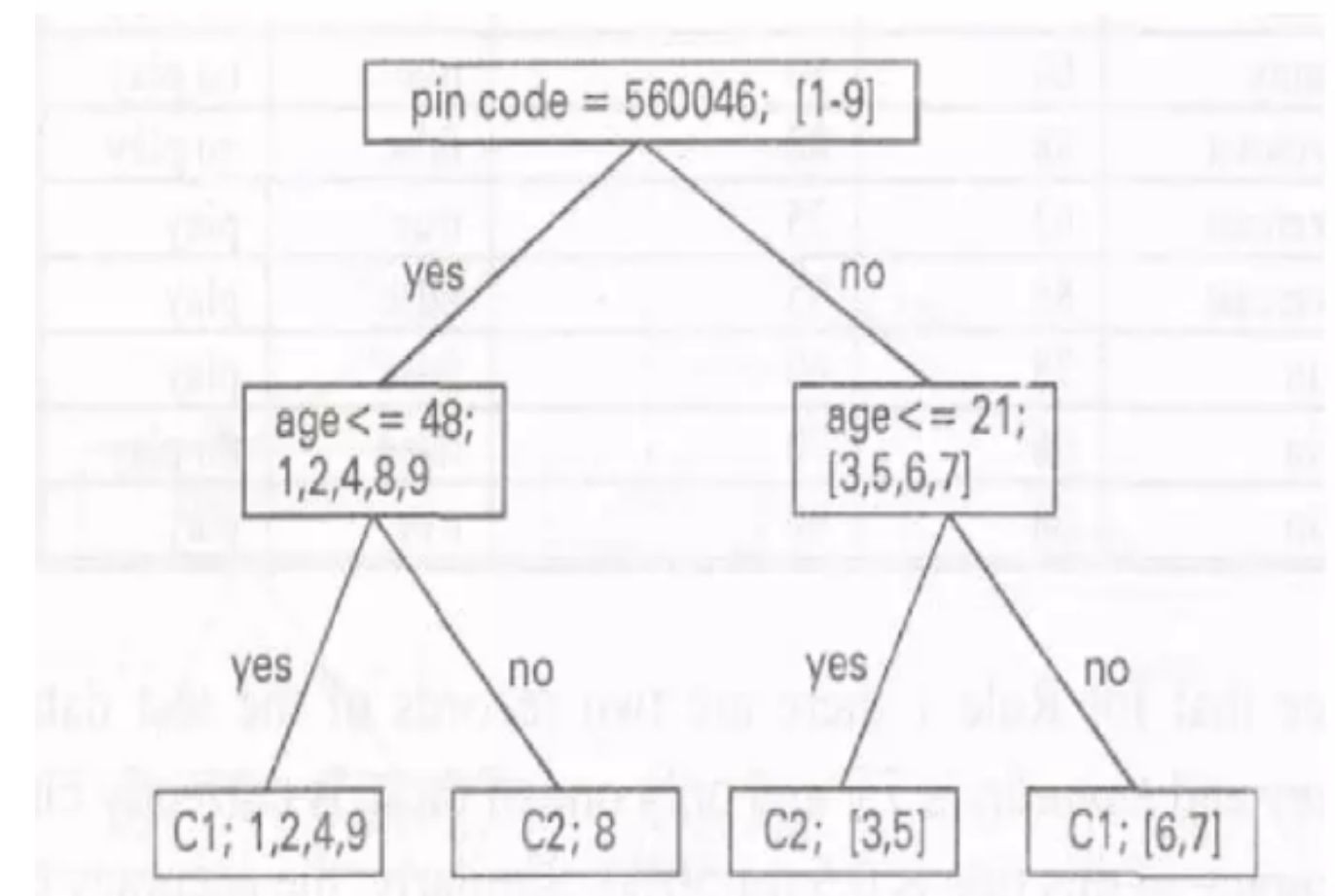


Possible misunderstanding about numerical and categorical attributes

pin code - is a **numeric attribute**, but there is no any useful information if one pincode is greater than another one

ID	AGE	PINCODE	CLASS
1	30	5600046	C1
2	25	5600046	C1
3	21	5600023	C2
4	43	5600046	C1
5	18	5600023	C2
6	33	5600023	C1
7	29	5600023	C1
8	55	5600046	C2
9	48	5600046	C1

So, we can accept **pin code** as **categorical attribute** to generate valid rules for classification



Advantages and disadvantages of Decision Tree Algorithm

Advantages:

- Decision tree is able to generate understandable and interpretable rules
- Decision tree is able to handle both categorical and numerical attributes
- Decision tree gives a clear understanding which attributes are more important for classification and which are less

Disadvantages:

- The process of “growing” a decision tree is computationally expensive. It requires examination for each node at each split.
- Some decision trees are not able to work with categorical attributes with 3 more unique values

Example on how to find rules for decision tree

$$Gini = 1 - \sum_j p_j^2$$

[Watch video on YouTube](#)

Build decision tree for the
following data set:

	Color	Shape	Class
1	Red	Square	0
2	Red	Circle	1
3	Blue	Square	1
4	Blue	Circle	0
5	Red	Square	0
6	Blue	Circle	1
7	Red	Square	0
8	Blue	Square	1
9	Blue	Circle	1
10	Red	Circle	0