მონაცემთა ანალიტიკა Python

ლექცია 14: კლასიფიკაცია. ერთ-ცვლადიანი ლოჯისტიკური რეგრესია. მრავალ-ცვლადიანი ლოჯისტიკური რეგრესია

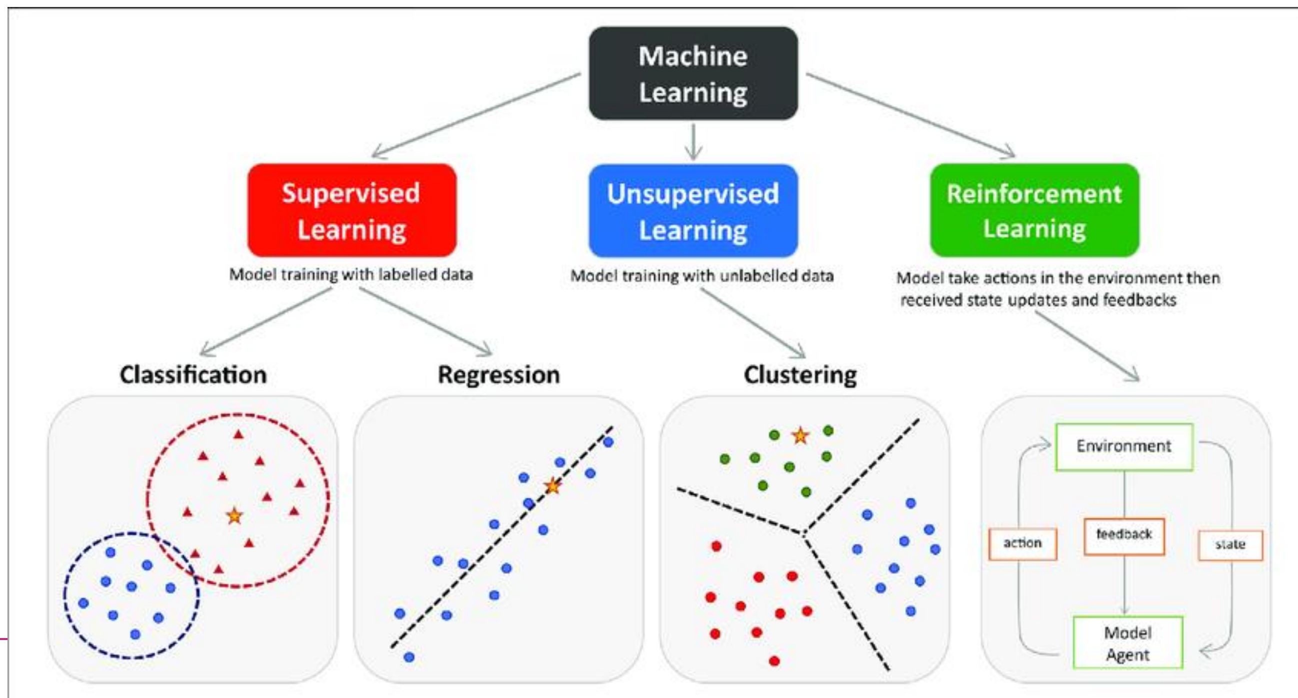
ლექცია 15: კლასიფიკაცია. გადაწყვეტილების ხე

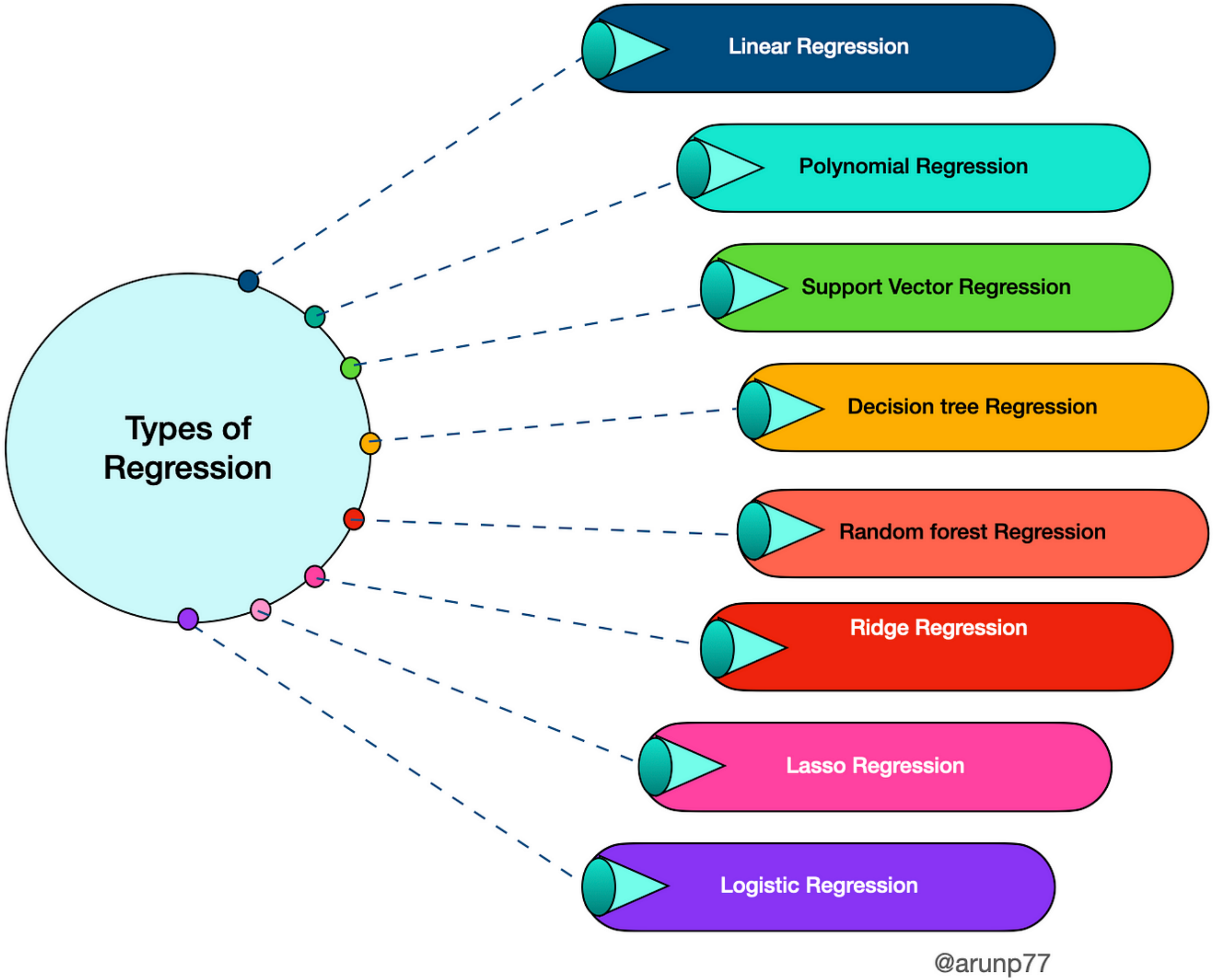
ლიკა სვანაძე
lika.svanadze@btu.edu.ge

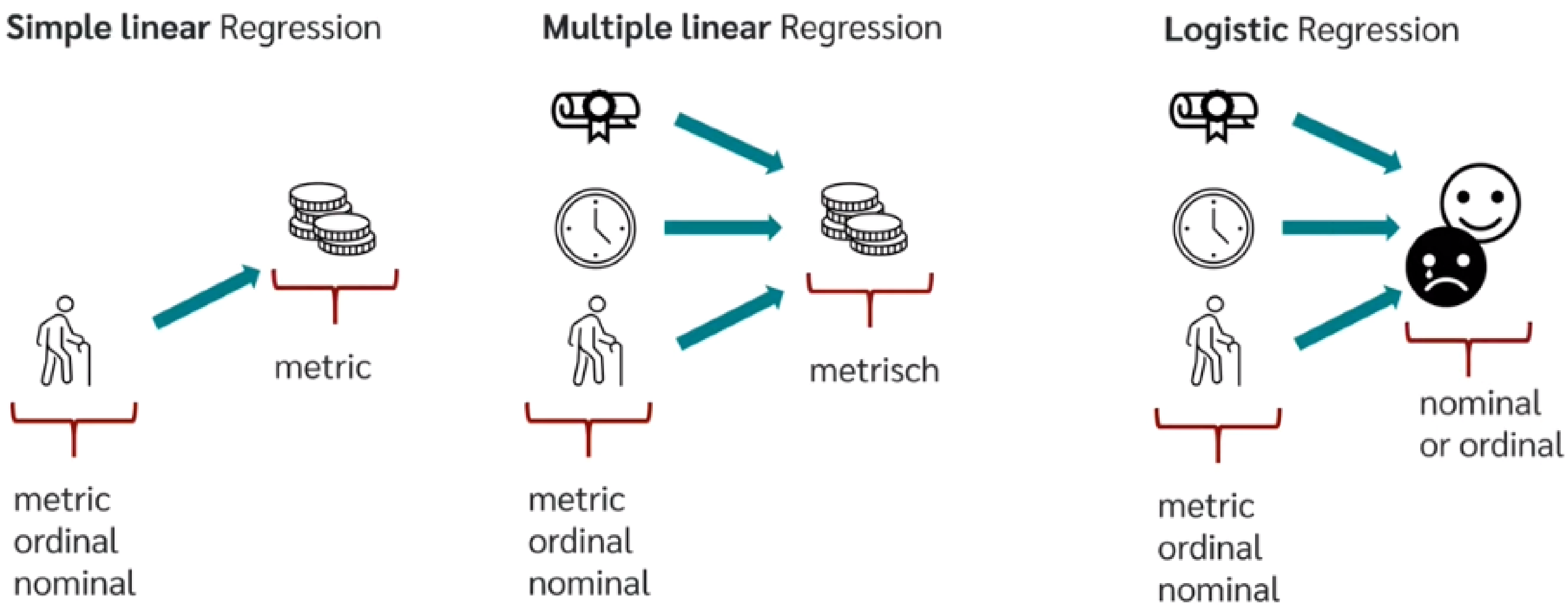
Machine learning Models

Based on the tasks performed and the nature of the output, you can **classify** machine learning models into three types:

1. **Regression:** where the output variable to be predicted is a continuous variable
2. **Classification:** where the output variable to be predicted is a categorical variable
3. **Clustering:** where there is no pre-defined notion of a label allocated to the groups/clusters formed.

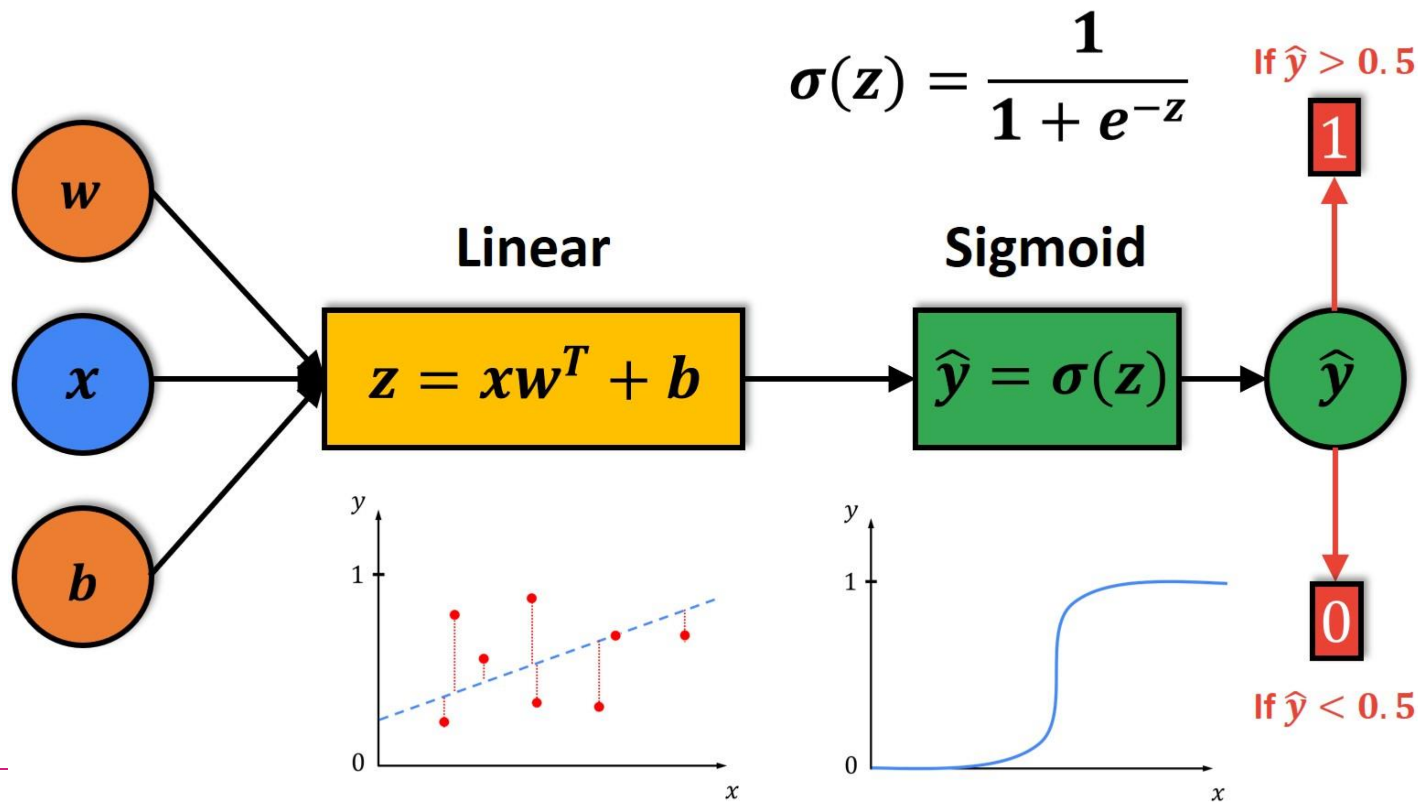



Types of Regression
Linear Regression
Polynomial Regression
Support Vector Regression
Decision tree Regression
Random forest Regression
Ridge Regression
Lasso Regression
Logistic Regression
@arunp77


Simple linear Regression
metric
metric
ordinal
nominal
Multiple linear Regression
metrisch
metric
ordinal
nominal
Logistic Regression
nominal
or ordinal
metric
ordinal
nominal

Logistic Regression

- Unlike linear regression which outputs continuous number values, **logistic regression** uses the logistic sigmoid function to transform its output to return a probability value which can then be mapped to two or more discrete classes.

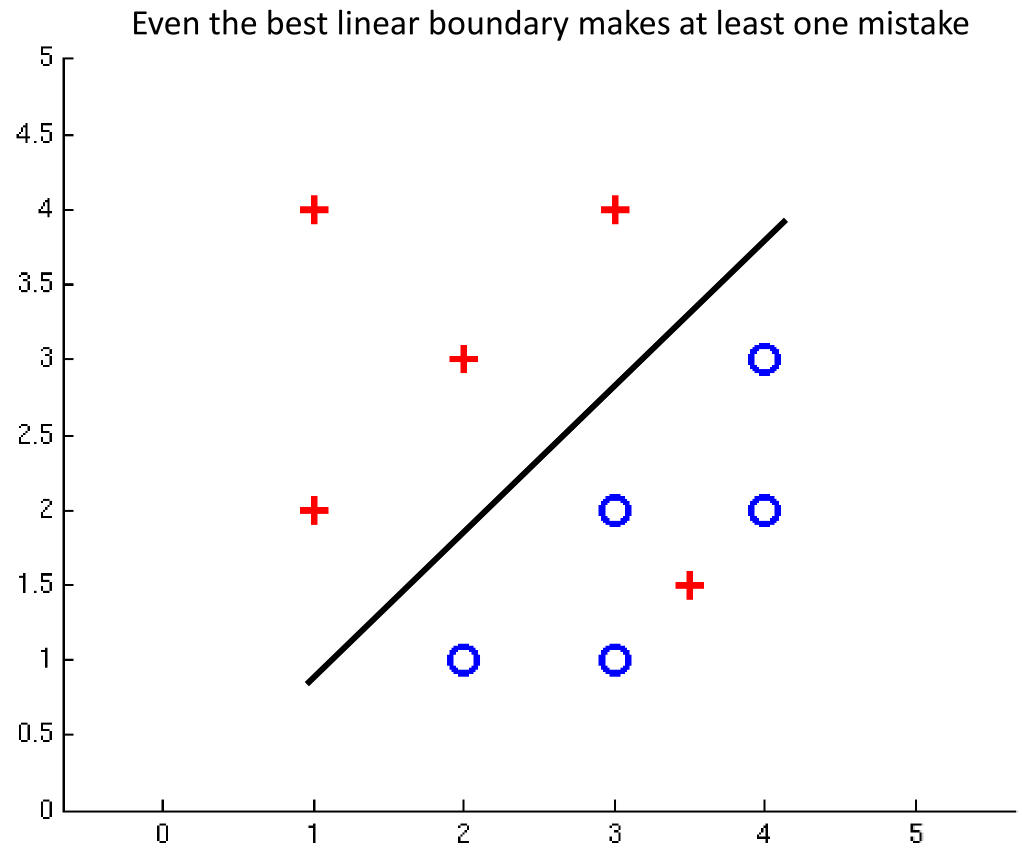


ilka.svanadze@btu.edu.ge

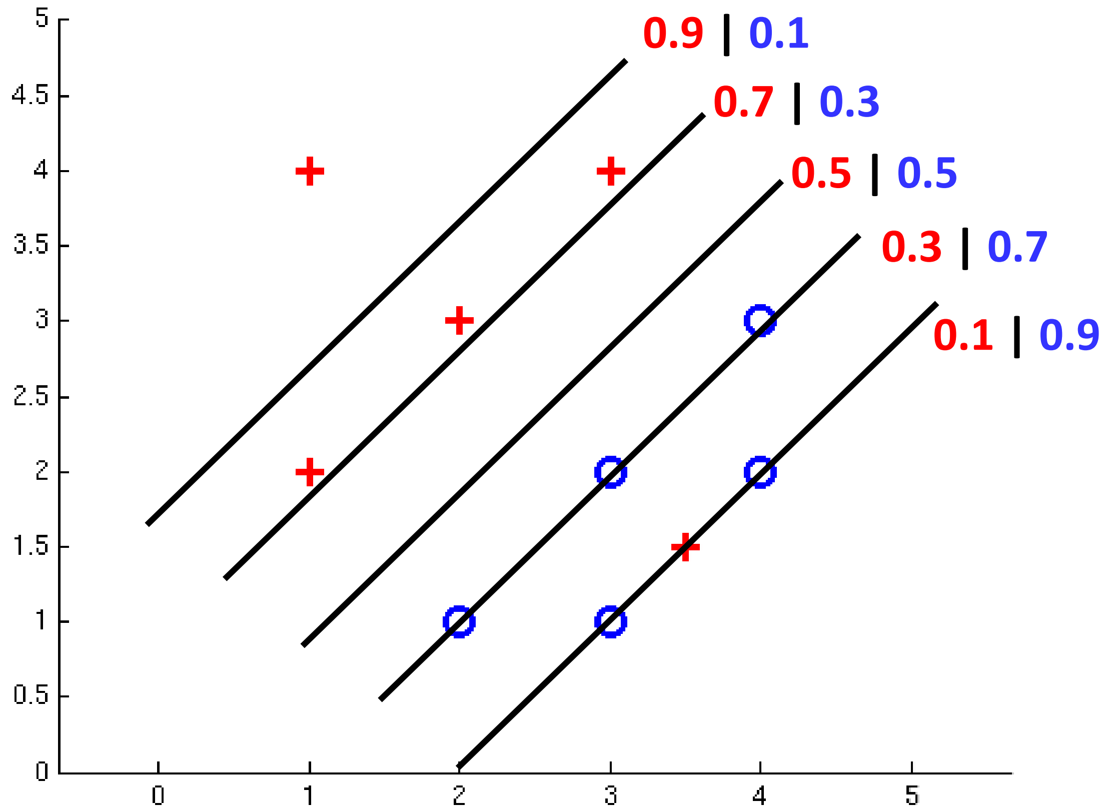
Types of Logistic Regression

1. Binary (true/false, yes/no)
2. Multi-class (sheep, cats, dogs)
3. Ordinal (Job satisfaction level — dissatisfied, satisfied, highly satisfied)

Non-Separable Case: Deterministic Decision



Non-Separable Case: Probabilistic Decision



How to get probabilistic decisions?

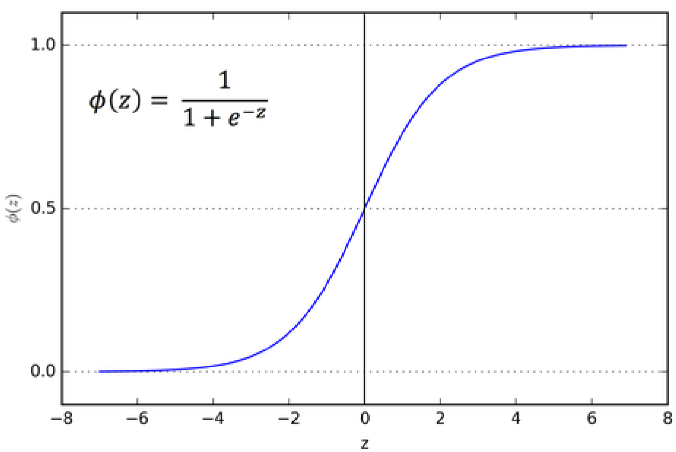
Perceptron scoring: $z = w \cdot f(x)$

If $z = w \cdot f(x)$ very positive → want probability going to 1

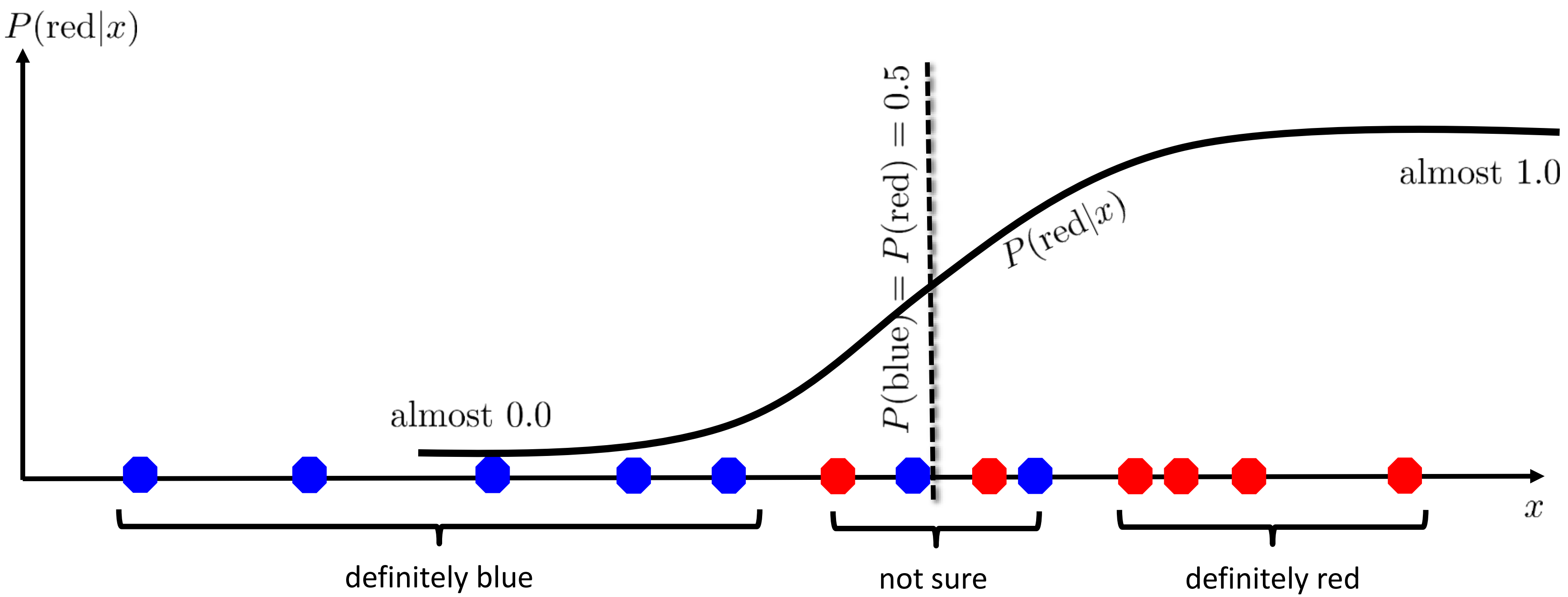
If $z = w \cdot f(x)$ very negative → want probability going to 0

Sigmoid function

$$\phi(z) = \frac{1}{1 + e^{-z}}$$

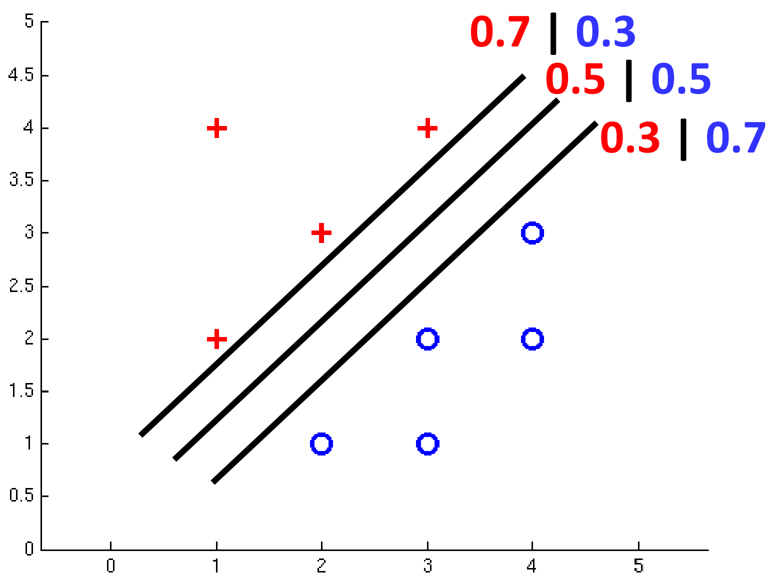


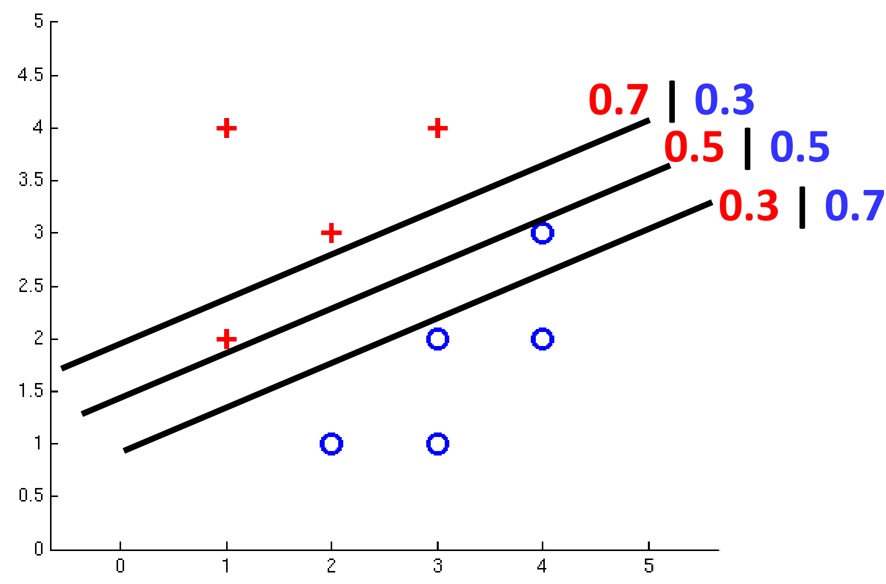
A 1D Example



$P(\text{red}|x)$
$P(\text{blue}) = P(\text{red}) = 0.5$
almost 1.0
$P(\text{red}|x)$
almost 0.0
x
definitely blue
not sure
definitely red

Separable Case: Probabilistic Decision – Clear Preference





Best w?

Maximum likelihood estimation:

$$\max_{w} \quad ll(w) = \max_{w} \quad \sum_{i} \log P(y^{(i)} | x^{(i)}; w)$$

with:

$$P(y^{(i)} = +1 | x^{(i)}; w) = \frac{1}{1 + e^{-w \cdot f(x^{(i)})}}$$

$$P(y^{(i)} = -1 | x^{(i)}; w) = 1 - \frac{1}{1 + e^{-w \cdot f(x^{(i)})}}$$

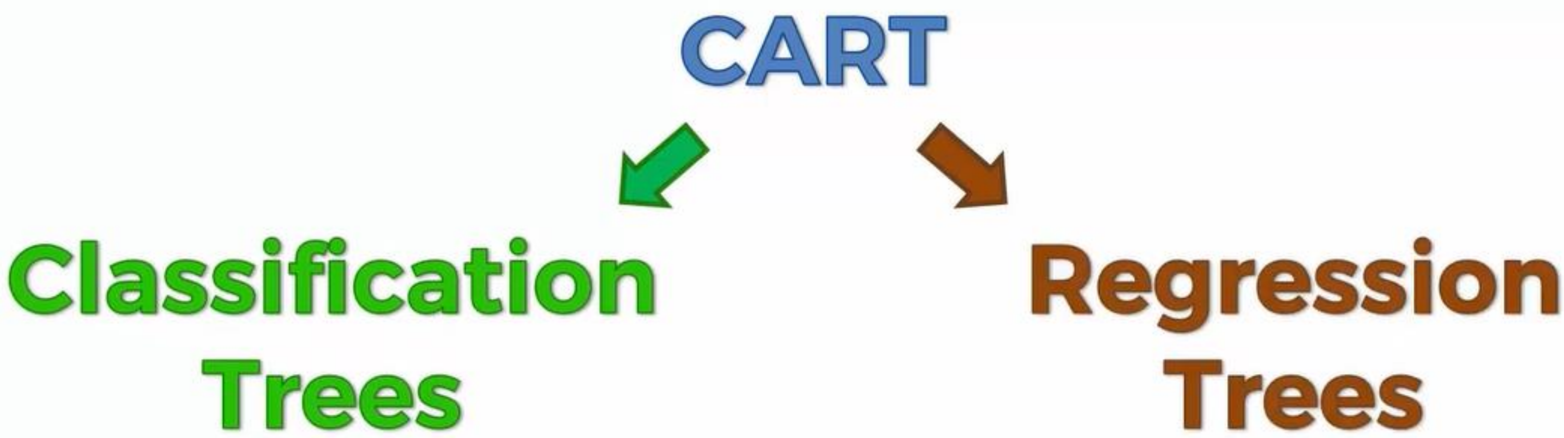
= Logistic Regression

მონაცემთა ანალიტიკა Python

ლექცია 15: კლასიფიკაცია. გადაწყვეტილების ხე. მრავალი გადაწყვეტილების ხე.

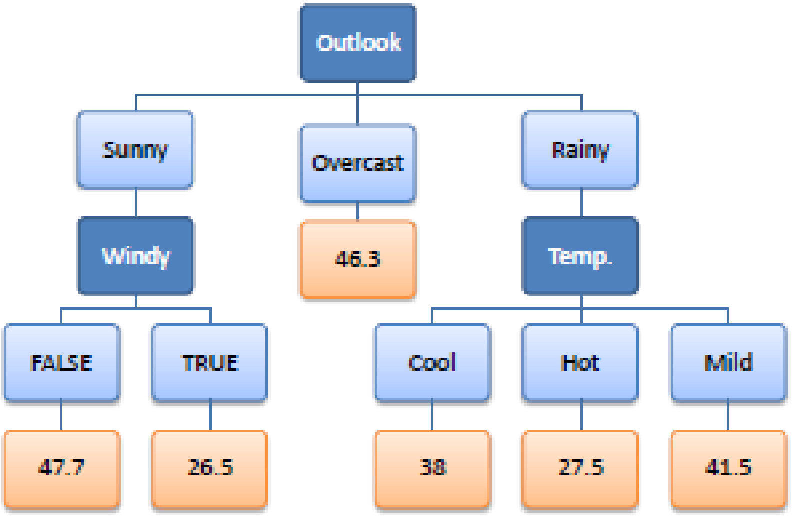
ლიკა სვანაძე
lika.svanadze@btu.edu.ge

Decision Trees

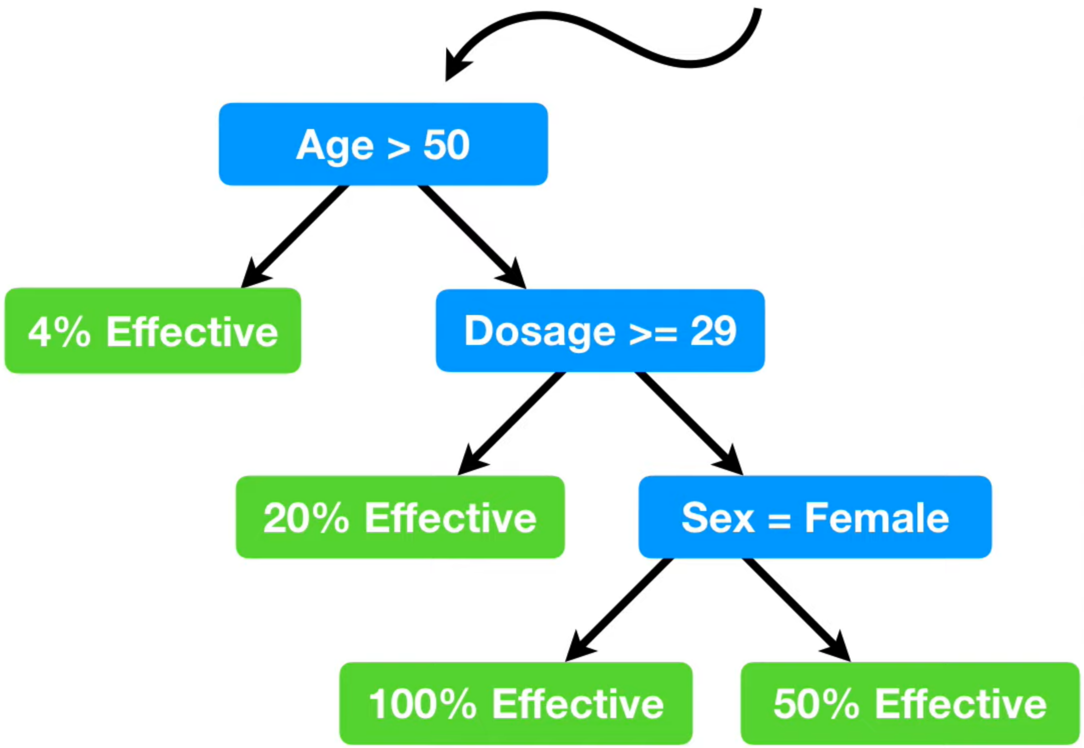


Decision Tree - Regression

Outlook	Temp.	Humidity	Windy	Hours Played
Rainy	Hot	High	False	25
Rainy	Hot	High	True	30
Overcast	Hot	High	False	46
Sunny	Mild	High	False	45
Sunny	Cool	Normal	False	52
Sunny	Cool	Normal	True	23
Overcast	Cool	Normal	True	43
Rainy	Mild	High	False	35
Rainy	Cool	Normal	False	38
Sunny	Mild	Normal	False	46
Rainy	Mild	Normal	True	48
Overcast	Mild	High	True	52
Overcast	Hot	Normal	False	44
Sunny	Mild	High	True	30



In contrast, a **Regression Tree** easily accommodates the additional predictors.



Dosage	Age	Sex	Etc.	Drug Effect.
10	25	Female	…	98
20	73	Male	…	0
35	54	Female	…	100
5	12	Male	…	44
etc…	etc…	etc…	etc…	etc…

Decision Trees

Reminder: Features

Features, aka attributes

- Sometimes: TYPE=French
- Sometimes: $f_{\text{TYPE=French}}(x) = 1$

Example	Attributes										Target
	Alt	*Bar*	*Fri*	*Hun*	*Pat*	*Price*	*Rain*	*Res*	*Type*	*Est*	*WillWait*
X_1	T	F	F	T	Some	$$$	F	T	French	0–10	T
X_2	T	F	F	T	Full	$	F	F	Thai	30–60	F
X_3	F	T	F	F	Some	$	F	F	Burger	0–10	T
X_4	T	F	T	T	Full	$	F	F	Thai	10–30	T
X_5	T	F	T	F	Full	$$$	F	T	French	>60	F
X_6	F	T	F	T	Some	$$	T	T	Italian	0–10	T
X_7	F	T	F	F	None	$	T	F	Burger	0–10	F
X_8	F	F	F	T	Some	$$	T	T	Thai	0–10	T
X_9	F	T	T	F	Full	$	T	F	Burger	>60	F
X_{10}	T	T	T	T	Full	$$$	F	T	Italian	10–30	F
X_{11}	F	F	F	F	None	$	F	F	Thai	0–10	F
X_{12}	T	T	T	T	Full	$	F	F	Burger	30–60	T

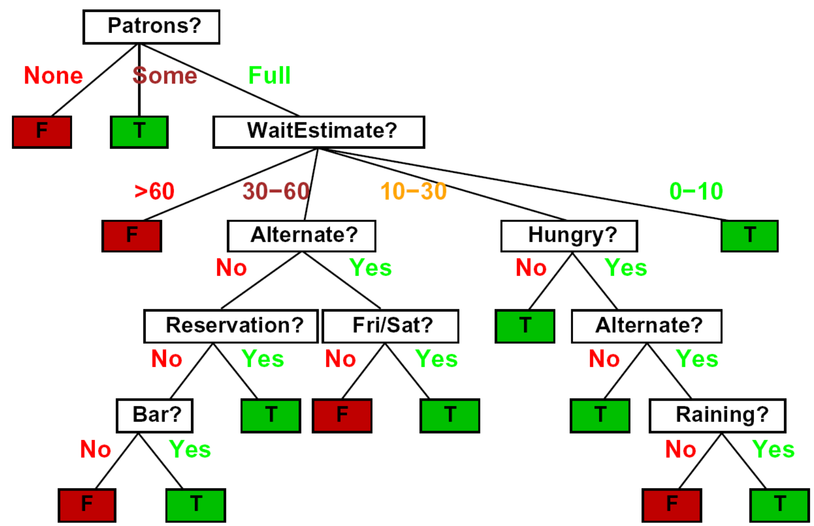
Decision Trees

Compact representation of a function:

- Truth table
- Conditional probability table
- Regression values

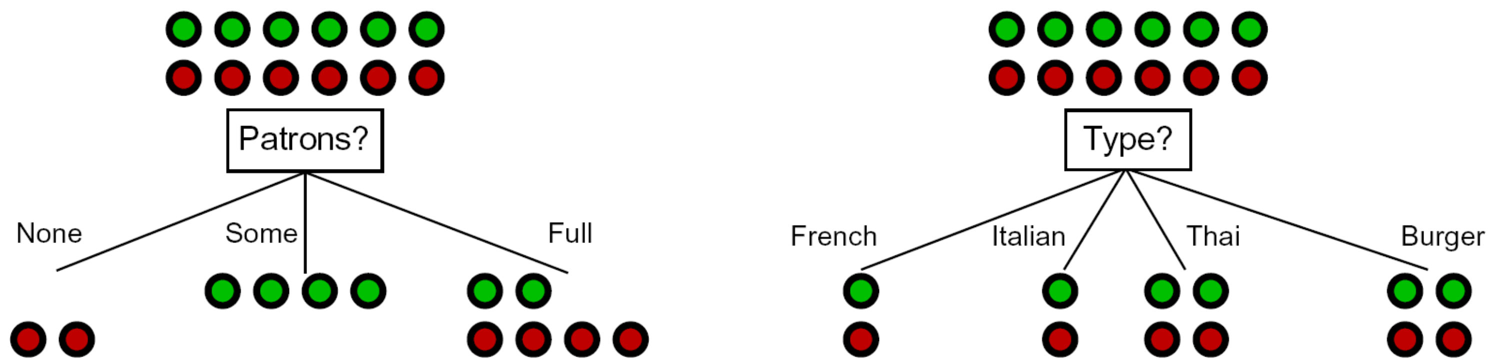
True function

- Realizable: in H



Choosing an Attribute

Idea: a good attribute splits the examples into subsets that are (ideally) “all positive” or “all negative”



So: we need a measure of how “good” a split is, even if the results aren’t perfectly separated out

Entropy and Information

Information answers questions

- The more uncertain about the answer initially, the more information in the answer
- Scale: bits
  - Answer to Boolean question with prior <1/2, 1/2>?
  - Answer to 4-way question with prior <1/4, 1/4, 1/4, 1/4>?
  - Answer to 4-way question with prior <0, 0, 0, 1>?
  - Answer to 3-way question with prior <1/2, 1/4, 1/4>?

A probability p is typical of:

- A uniform distribution of size 1/p
- A code of length log 1/p

Entropy

General answer: if prior is <$p_1, \ldots, p_n$>:

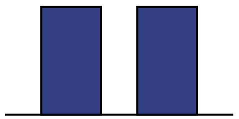
- Information is the expected code length

$$H(\langle p_1, \ldots, p_n \rangle) = E_p \log_2 1/p_i$$

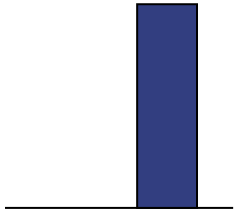
$$= \sum_{i=1}^{n} -p_i \log_2 p_i$$

Also called the entropy of the distribution

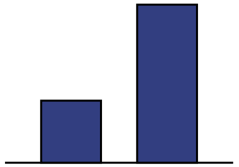
- More uniform = higher entropy
- More values = higher entropy
- More peaked = lower entropy
- Rare values almost “don’t count”



1 bit



0 bits

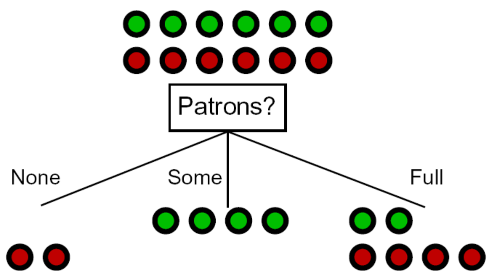


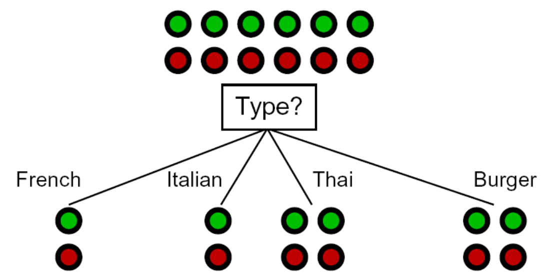
0.5 bit

Information Gain

Back to decision trees!

For each split, compare entropy before and after
- Difference is the information gain
- Problem: there's more than one distribution after split!





- Solution: use expected entropy, weighted by the number of examples

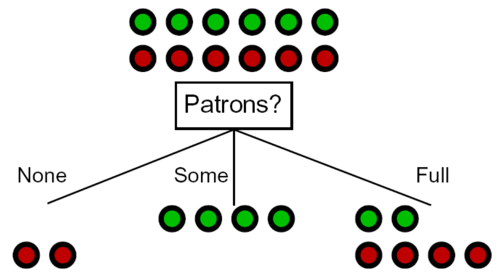
Next Step: Recurse

Now we need to keep growing the tree!

Two branches are done (why?)

What to do under “full”?

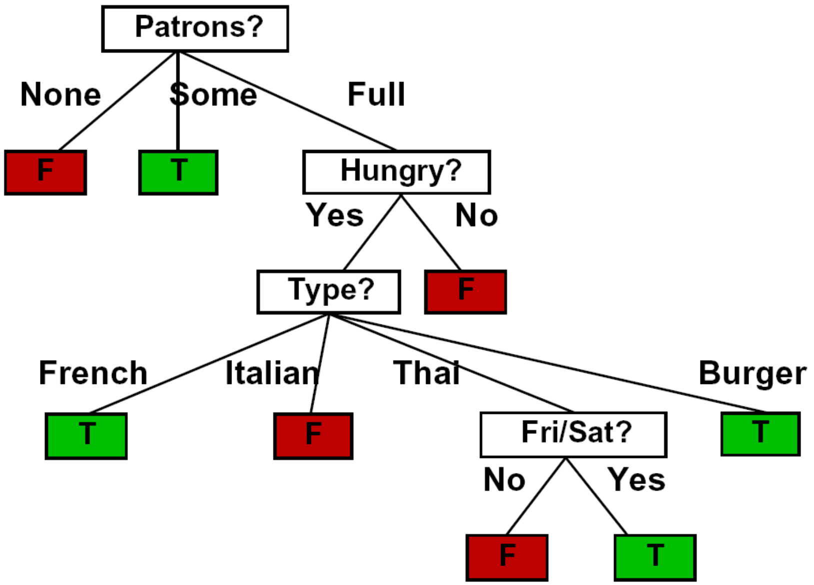
- See what examples are there…



Example	Attributes										Target
	Alt	*Bar*	*Fri*	*Hun*	*Pat*	*Price*	*Rain*	*Res*	*Type*	*Est*	*WillWait*
X_1	T	F	F	T	Some	\$\$\$	F	T	French	0–10	T
X_2	T	F	F	T	Full	\$	F	F	Thai	30–60	F
X_3	F	T	F	F	Some	\$	F	F	Burger	0–10	T
X_4	T	F	T	T	Full	\$	F	F	Thai	10–30	T
X_5	T	F	T	F	Full	\$\$\$	F	T	French	>60	F
X_6	F	T	F	T	Some	\$\$	T	T	Italian	0–10	T
X_7	F	T	F	F	None	\$	T	F	Burger	0–10	F
X_8	F	F	F	T	Some	\$\$	T	T	Thai	0–10	T
X_9	F	T	T	F	Full	\$	T	F	Burger	>60	F
X_{10}	T	T	T	T	Full	\$\$\$	F	T	Italian	10–30	F
X_{11}	F	F	F	F	None	\$	F	F	Thai	0–10	F
X_{12}	T	T	T	T	Full	\$	F	F	Burger	30–60	T

Example: Learned Tree

Decision tree learned from these 12 examples:



Substantially simpler than “true” tree

- A more complex hypothesis isn't justified by data

Also: it’s reasonable, but wrong

Python implementation (See python files)

lika.svanadze@btu.edu.ge