Features and classification

CSC401/2511 – Natural Language Computing –Spring 2024
Lecture 5 Gerald Penn, Sean Robertson, Raeid Saqur
University of Toronto

Lecture 5 overview

- Today:
- Quick introduction to Text Classification
- **Feature extraction** from text.
  - How to pick the right features?
  - Grammatical ‘parts-of-speech’.
    - (even when nobody is speaking)
- **Classification** overview

- Some slides *may* be based on content from Bob Carpenter, Dan Klein, Roger Levy, Josh Goodman, Dan Jurafsky, and Christopher Manning.

Quick Intro to Text Classification

From Technology Upskilling Machine Learning Software Foundations by En-Shiun Annie Lee

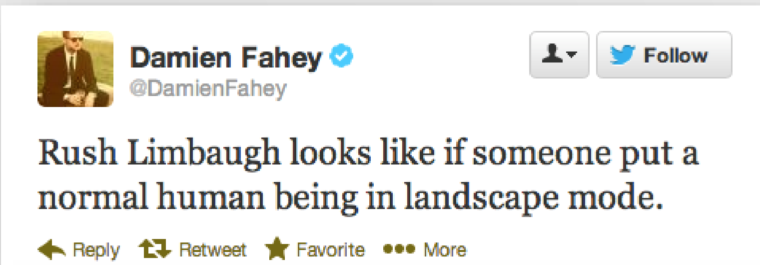
Features

- **Feature**: *n.* A measurable **variable** that is (or *should be*) **distinctive** of something we want to model.
- We often choose features to **classify** something.
  - e.g., an emotional, whiny **tone** is likely to indicate that the speaker is not professional, scientific, nor political.
  - Note that in neural networks, e.g., '**features**' often refer to something distinctive but not usually *nameable.*
- We often need **various, heterogeneous** features to adequately model something,

  e.g. tone plus aspects of their grammar.

Example: Feature vectors

- **Values** for **several** features of an **observation** can be put into a single **vector**.



# proper nouns	# 1st person pronouns	# commas
2	0	0
5	0	0
0	1	1

UNIVERSITY OF TORONTO

Feature vectors

- Features should be useful in **discriminating** between categories.

Table 3: Features to be computed for each text

- Counts:
  - First person pronouns
  - Second person pronouns
  - Third person pronouns
  - Coordinating conjunctions
  - Past-tense verbs
  - Future-tense verbs
  - Commas
  - Colons and semi-colons
  - Dashes
  - Parentheses
  - Ellipses
  - Common nouns
  - Proper nouns
  - Adverbs
  - *wh*-words
  - Modern slang acroynms
  - Words all in upper case (at least 2 letters long)
- Average length of sentences (in tokens)
- Average length of tokens, excluding punctuation tokens (in characters)
- Number of sentences

Higher values → this person is referring to themselves (to their opinion, too?)

Higher values → looking forward to (or dreading) some future event?

Lower values → this tweet is more formal. Perhaps not overly sentimental?

UNIVERSITY OF TORONTO

Different features for different tasks

- **Alzheimer's disease** involves atrophy in the brain.
  - Excessive **pauses** (acoustic disfluencies),
  - Excessive **word type repetition**, and
  - Simplistic or **short** sentences.
    - '**function words**' like *the* and *an* are often **dropped**.
- To **diagnose** Alzheimer's disease, one might measure:
  - **Proportion** of utterance spent in **silence**.
  - **Entropy** of **word type** usage.
  - **Number** of word **tokens** in a sentence.
    - **Number** of **prepositions** and **determiners** (explained shortly).

Explainability/Interpretability!

Features in Sentiment Analysis

- **Sentiment analysis** can involve detecting:
  - **Stress** or **frustration** in a conversation.
  - **Interest**, **confusion**, or **preferences**. Useful to marketers.
    - e.g., *'got socks for xmas wanted #ps5 fml'*
  - **Deceipt**. e.g., *'Let's watch Netflix and chill.'*
- Complicating factors include **sarcasm, implic** … a **subtle** spectrum from **negative** to **positive** opinions.
- **Useful features** for sentiment analyzers include:
  - Trigrams.
  - First-person pronouns.

What does this mean?

Pronouns? Prepositions? Determiners?

Pre-processing

- **Pre-processing** involves **preparing** your data to make feature extraction easier or more valid.
  - E.g., **punctuation** likes to press up against words. The sequence " *example,* " should be counted as **two** tokens – not one.
    - We separate the punctuation, as in " *example ,* ".
- ***<u>There is no perfect pre-processor</u>***.
  Mutually exclusive approaches can often **both** be justified.
  - E.g., Is *Newfoundland-Labrador* **one** word type or **two**?
    Each answer has a unique implication for splitting the dash.
  - Often, **noise-reduction** removes *some* information.
  - Being **consistent** is important.

Parts of Speech

Parts-of-speech (PoS)

- Linguists like to group words according to their **distribution** in building grammatical sentences.
  - This is similar to grouping Lego by their shapes.

- **Part-of-speech**: *n.* lexical category or morphological class.

Nouns collectively constitute a part-of-speech (called *Noun*)

UNIVERSITY OF TORONTO

Example parts-of-speech

Part of Speech	Description	Examples
Noun	is usually a **person**, **place**, **event**, or **entity**.	*chair, pacing, monkey, breath.*
Verb	is usually an **action** or **predicate**.	*run, debate, explicate.*
Adjective	modifies a **noun** to further describe it.	*orange, obscene, disgusting.*
Adverb	modifies a **verb** to further describe it.	*lovingly, horrifyingly, often*

Example parts of speech

Part of Speech	Description	Examples
Preposition	Often specifies aspects of **space**, **time**, or **means**.	*around, over, under, after, before, with*
Pronoun	Substitutes for nouns; referent typically understood in context.	*I, we, they*
Determiner	logically **quantify** words, usually nouns.	*the, an, both, either*
Conjunction	**combines** words or phrases.	*and, or, although*

Content categories

- Some PoSs convey content labels more than function or linguistic structure**.**
  - Usually nouns, verbs, adjectives, adverbs.
  - **Content** categories are usually multifarious.
    - e.g., there are more **nouns** than **prepositions**.
  - **New** content words are continually **added**

    e.g., an *app, to google, to misunderestimate*.
  - Some **archaic** content words go **extinct**.

    e.g., *fumificate, v., (1721-1792),*
    frenigerent, adj., (1656-1681),
    melanochalcographer, n., (c. 1697).

Function categories

- Some PoSs are ‘**glue**’ that holds others together.
  - E.g., prepositions, determiners, conjunctions.
  - **Functional** PoS usually cover a **small** and **fixed** number of word types (i.e., a ‘**closed class**’).

  - Their **semantics** depend on the contentful words with which they’re used.
    - E.g., *I’m* ***on*** *time* vs. *I’m* ***on*** *a boat*

UNIVERSITY OF TORONTO

Grammatical "features"

- There are several **grammatical features** that can be associated with words:
  - **Case**
  - **Person**
  - **Number**
  - **Gender**

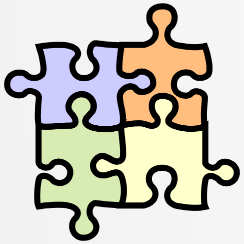
- These features can **restrict** other words in a sentence.

Other features of nouns

- **Proper noun**: **named** things (e.g., "*they've killed* ***Bill****!*")
- **Common noun**: **unnamed** things (e.g., "*they've killed the* ***bill****!*")

- **Mass noun**: **divisible** and **uncountable** (e.g., "***butter***" split in two gives two piles of butter – not two '*butters*')
- **Count noun**: **indivisible** and **countable**. (e.g., a "***pig***" split in two does not give two pigs)

Agreement



- Parts-of-speech ***should*** match (i.e., **agree**) in certain ways.

- **Articles** ‘have’ to **agree** with the **number** of their **noun**
  - e.g., “*these pretzels are making me thirsty*”
  - e.g., “*a winters are coming*”

- **Verbs** ‘have’ to **agree** (at least) with their **subject** (in English)
  - e.g., “*the dogs eats the gravy*” **no number** agreement
  - e.g., “*Yesterday, all my troubles seem so far away*”
    bad tense – should be past tense *seemed*
  - e.g., “*Can you handle me the way I are?*”

UNIVERSITY OF TORONTO

Tagging

PoS tagging

- **Tagging**: *v.g.* the process of **assigning** a **part-of-speech** to each word in a sequence.

- E.g., using the '**Penn treebank**' tag set (see appendix):

Word	The	nurse	put	the	sick	patient	to	sleep
Tag	DT	NN	VBD	DT	JJ	NN	IN	NN

Ambiguities in parts-of-speech

- Word types can have many parts-of-speech.
  - E.g., *back:*
    - *The back/JJ door* (adjective)
    - *On its back/NN* (noun)
    - *Win the voters back/RB* (adverb)
    - *Promise to back/VB you in a fight* (verb)

- We want to determine the **appropriate** tag for a given *token* in its context.

Why is tagging useful?

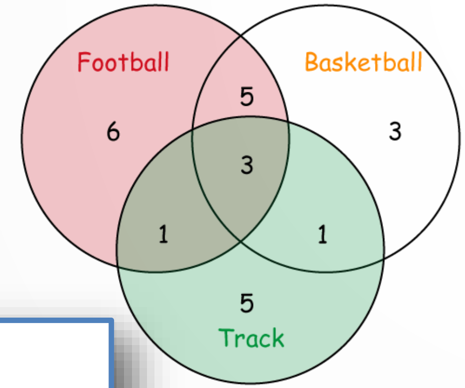
- First step towards many practical purposes.
  - **Speech synthesis**: how to pronounce text
    - *I'm* ***conTENT****/JJ* vs. *the* ***CONtent****/NN*
    - *I* ***obJECT****/VBP* vs. *the* ***OBJect****/NN*
    - *I* ***lead****/VBP ("l iy d")* vs. *it's* ***lead****/NN ("l eh d")*
  - **Information extraction**:
    - Help to find names and relations.
  - **Machine translation**:
    - Help to identify phrase boundaries
  - **Explainability?**

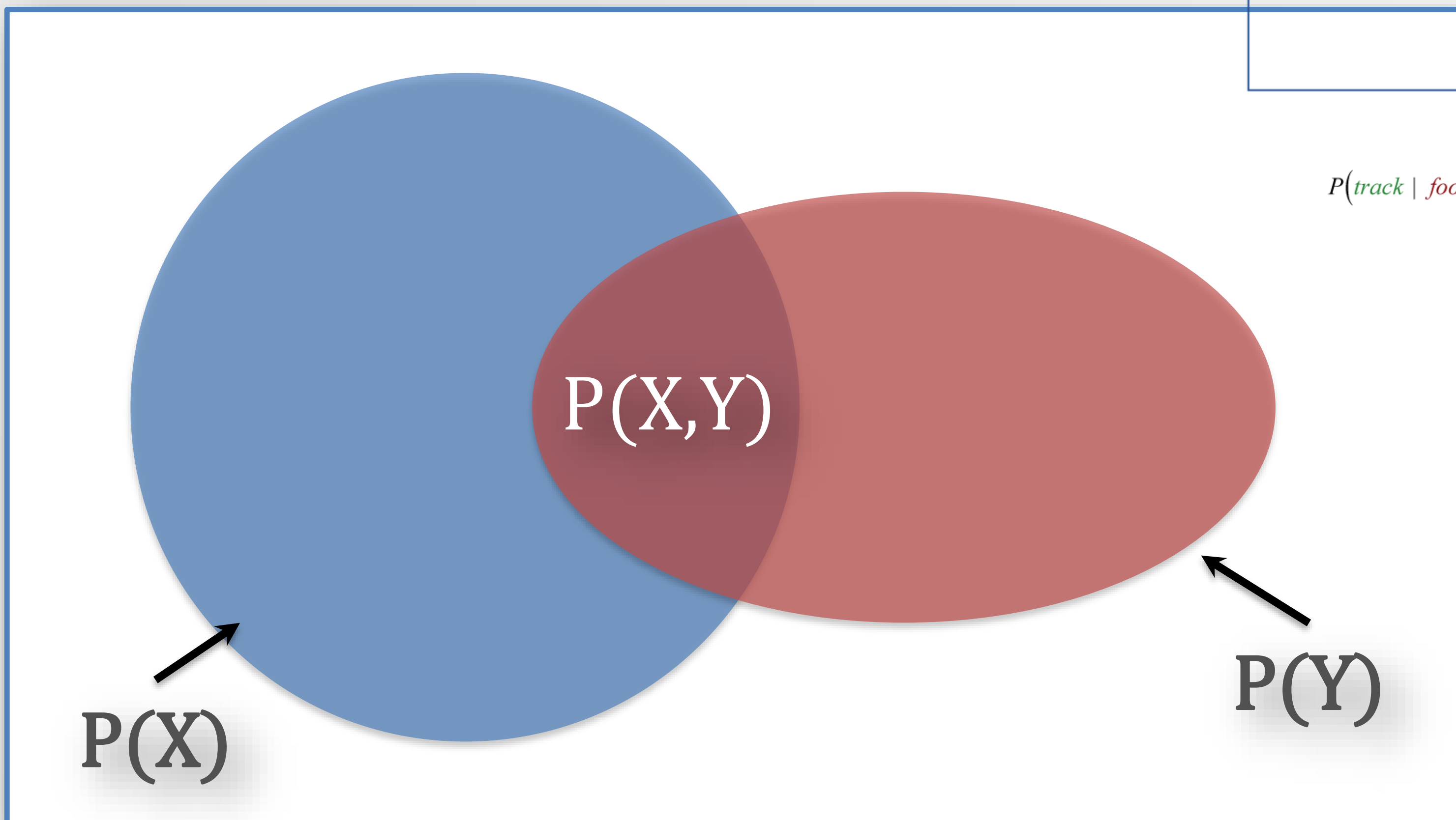
Tagging as classification

- We have access to a **sequence of observations** and are expected to decide on the best assignment of a **hidden variable**, i.e., the PoS

Hidden variable				NN		
				VB		
		VBN		JJ		**NN**
	PRP	**VBD**	**TO**	RB	**DT**	VB
Observation	**she**	**promised**	**to**	**back**	**the**	**bill**

Reminder: Bayes' Rule





$$P(track \mid football) = \frac{P(track \cap football)}{P(football)} = \frac{\left(\frac{4}{\cancel{24}}\right)}{\left(\frac{15}{\cancel{24}}\right)} = \frac{4}{15}$$

$$P(X,Y) = P(X)P(Y|X)$$
$$P(X,Y) = P(Y)P(X|Y)$$

$$\boxed{P(X|Y) = \frac{P(X)}{P(Y)} P(Y|X)}$$

Statistical PoS tagging

- Determine the **most likely** tag sequence $t_{1:n}$ by:

$$\operatorname*{argmax}_{t_{1:n}} P(t_{1:n}|w_{1:n}) = \operatorname*{argmax}_{t_{1:n}} \frac{P(w_{1:n}|t_{1:n})P(t_{1:n})}{P(w_{1:n})}$$

By Bayes' Rule

$$= \operatorname*{argmax}_{t_{1:n}} \frac{P(w_{1:n}|t_{1:n})P(t_{1:n})}{\cancel{P(w_{1:n})}}$$

Only maximize numerator

$$\boxed{\approx \operatorname*{argmax}_{t_{1:n}} \prod_{i}^{n} \underbrace{P(w_i|t_i)}_{\text{Assuming independence}} \underbrace{P(t_i|t_{i-1})}_{\text{Assuming Markov}}}$$

Assuming independence

Assuming Markov

UNIVERSITY OF TORONTO

Those are hidden Markov models!

- We'll see these soon…

Image sort of from *2001:A Space Odyssey*
by MGM pictures

Word likelihood probability $P(w_i|t_i)$

- **VBZ** (verb, 3rd person singular present) is likely ***is***.
- Compute $P(\boldsymbol{is}|VBZ)$ by **counting** in a corpus that has **already** been **tagged**:

$$P(w_i|t_i) = \frac{Count(w_i \text{ tagged as } t_i)}{Count(t_i)}$$

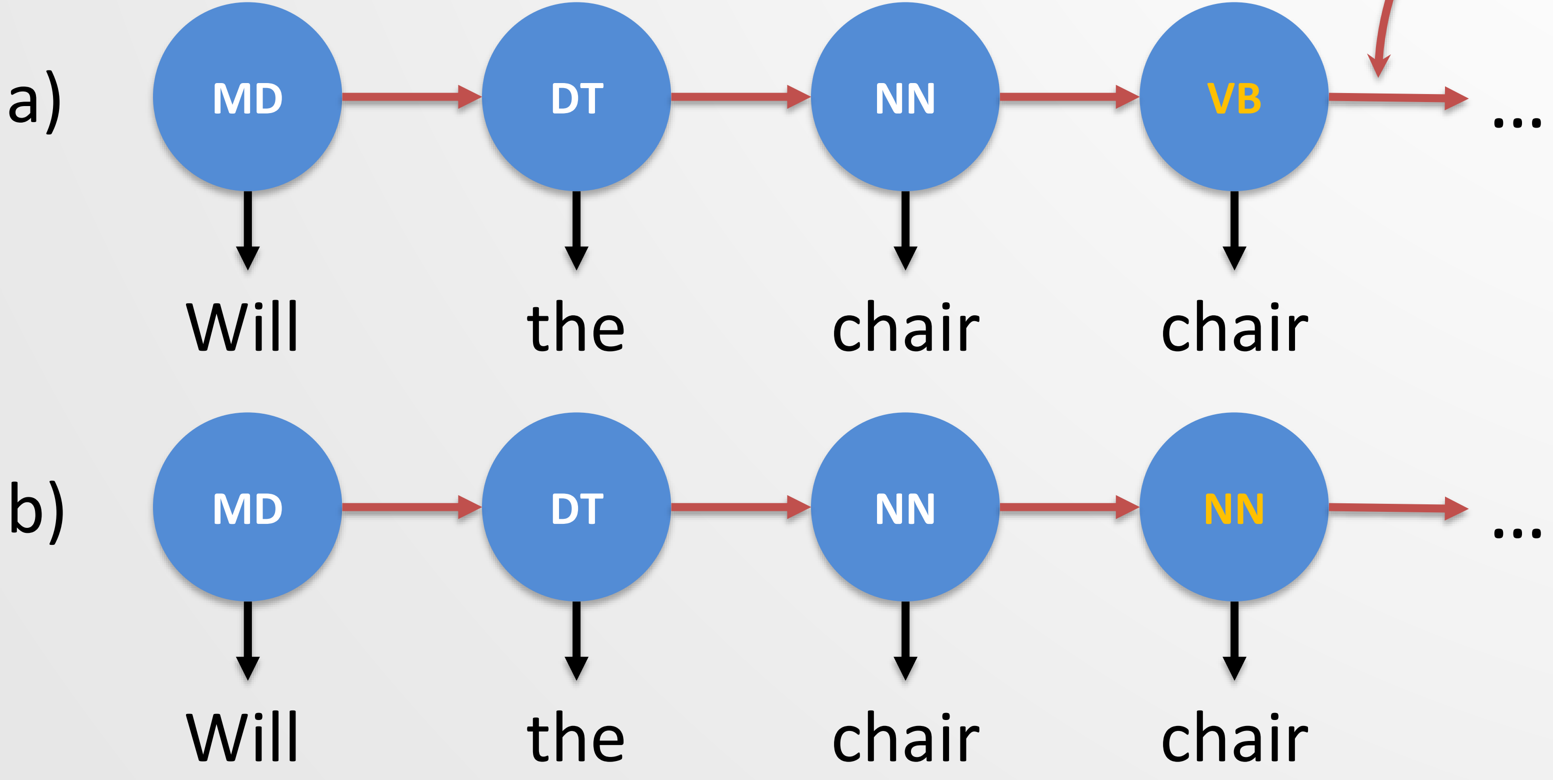
e.g.,

$$P(\boldsymbol{is}|\boldsymbol{VBZ}) = \frac{Count(\boldsymbol{is} \text{ tagged as } \boldsymbol{VBZ})}{Count(\boldsymbol{VBZ})} = \frac{10{,}073}{21{,}627} = 0.47$$

UNIVERSITY OF TORONTO

Tag-transition probability $P(t_i|t_{i-1})$

- *Will/MD the/DT **chair/NN** **chair/??** the/DT meeting/NN from/IN that/DT **chair/NN**?*



UNIVERSITY OF TORONTO

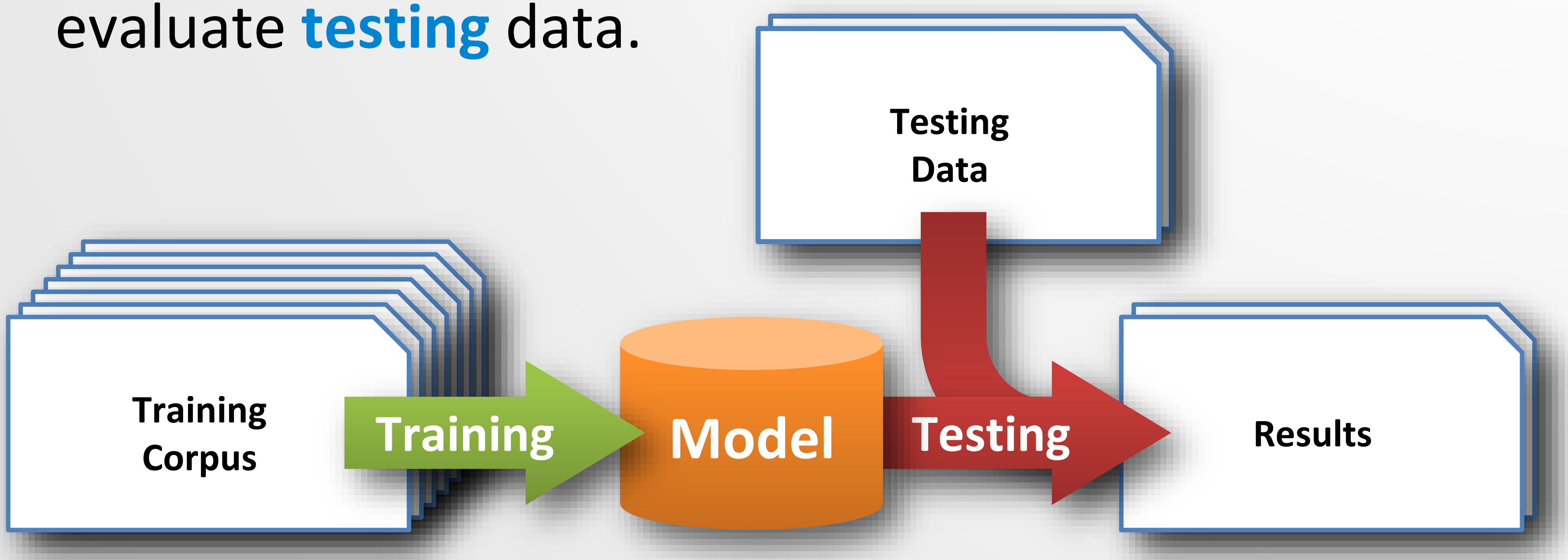
Lecture Review Slide

- What are some examples of Text Classification
- **What are features?**
  - What are unique features for the specific tasks of sentiment analysis versus spam detection?
  - What are some words with multiple POS tags?
  - Compute Baye’s rule for the POS tagging for an example.

Classification

General process

1. We gather a big and relevant **training** corpus.
2. We learn our **parameters** (e.g., probabilities) from that corpus to build our **model**.
3. Once that model is fixed, we use those probabilities to evaluate **testing** data.



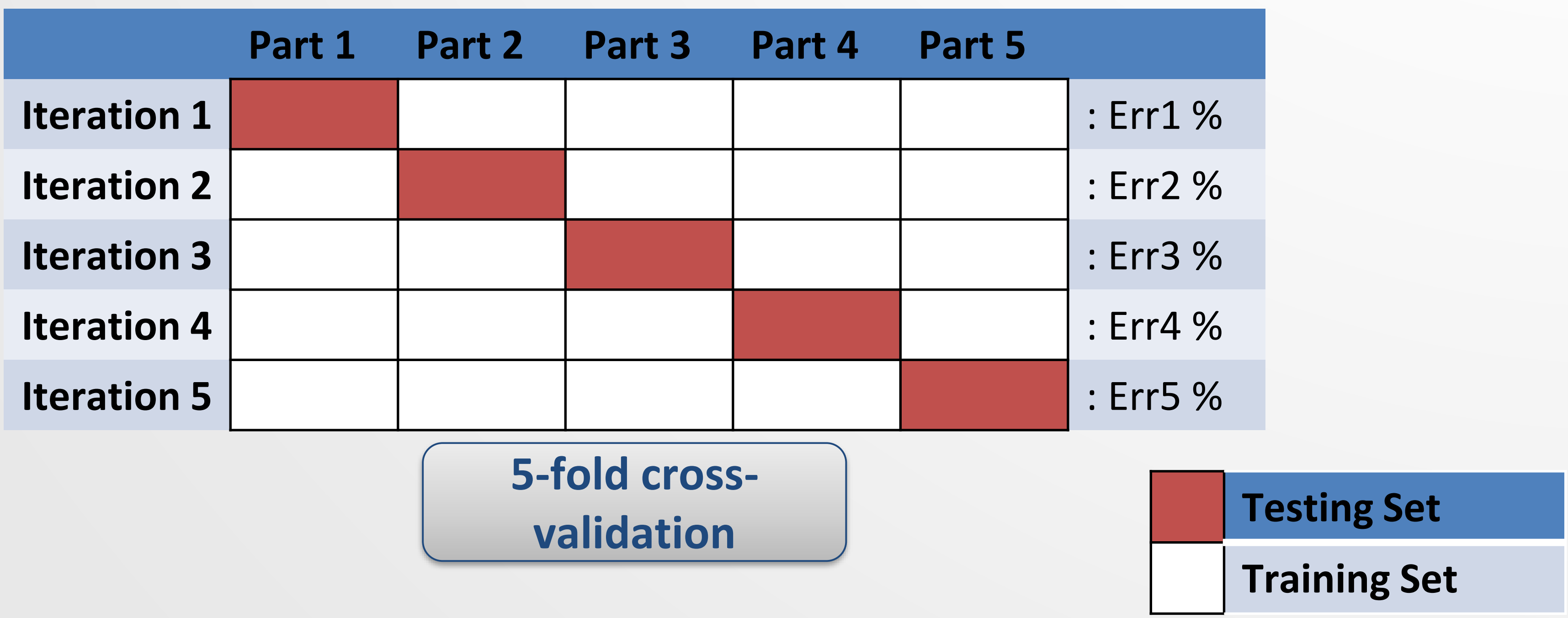
General process

- Often, **training data** consist of 80% to 90% of the available data.
  - Often, some subset of *this* is used as a **validation/development set**.

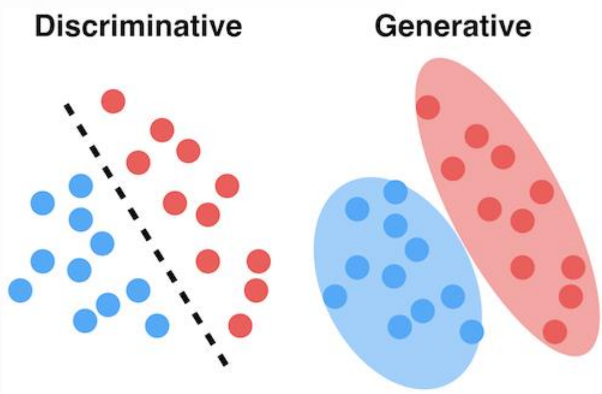
- **Testing data** are ***not*** used for training but often come from the same *corpus*.
  - It often consists of the remaining available data.
  - Sometimes, it's important to **partition** speakers/writers so they **don't** appear in both training and testing.
  - *But what if we just partitioned (un)luckily??*

Better process: *K*-fold cross-validation

- ***K*-fold cross validation**: *n.* splitting all data into *K* **partitions** and iteratively testing on each after training on the rest (report means and variances).



(Some) Types of classifiers



- **Generative** classifiers model the data.
  - Parameters set to maximize likelihood of training data.
  - We can *generate* new observations from these.
    - e.g., hidden Markov models

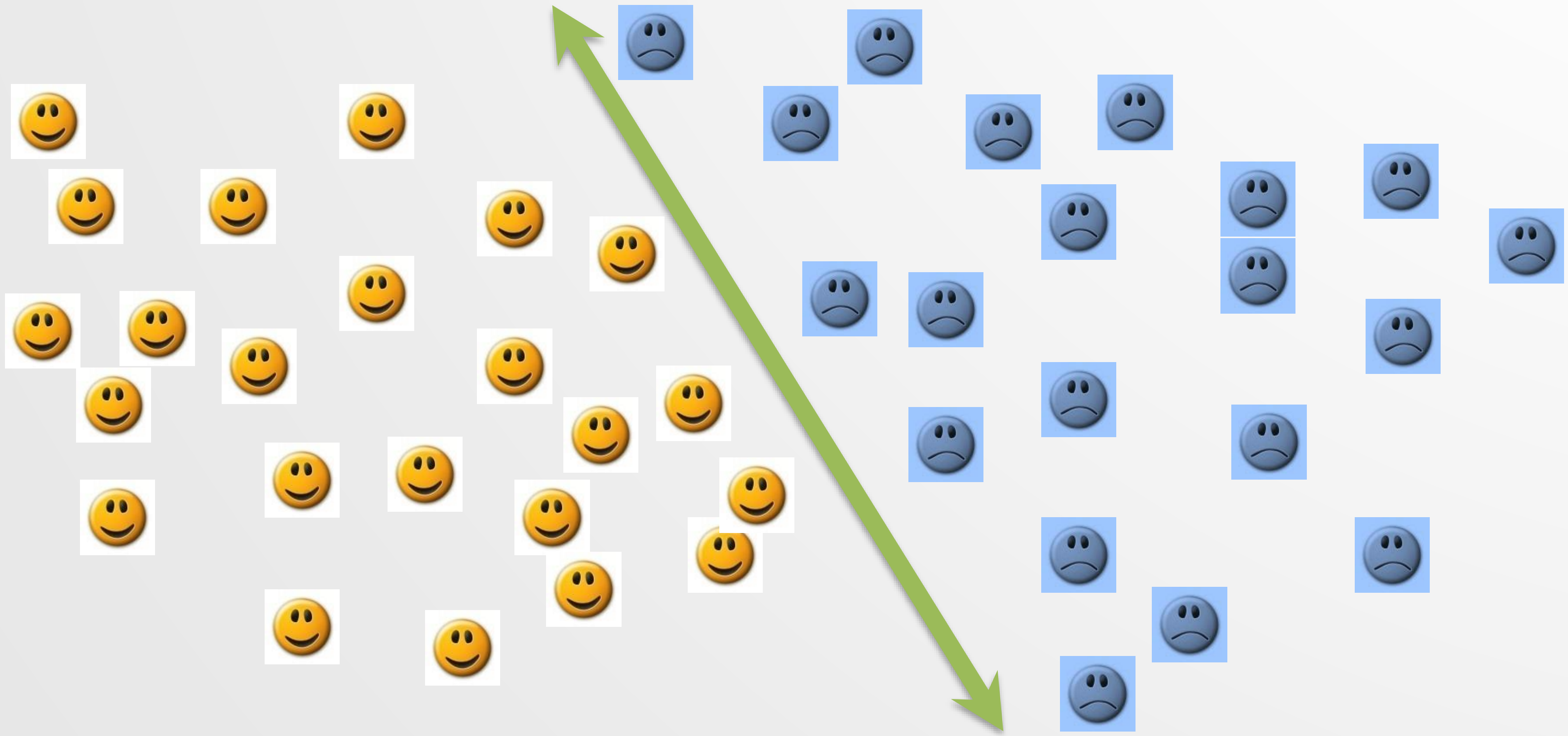
Vs.

- **Discriminative** classifiers emphasize **class boundaries**.
  - Parameters set to minimize error on training data.
    - e.g., support vector machines, decision trees.

- …*What do class boundaries look like in the data?*

Binary and linearly separable

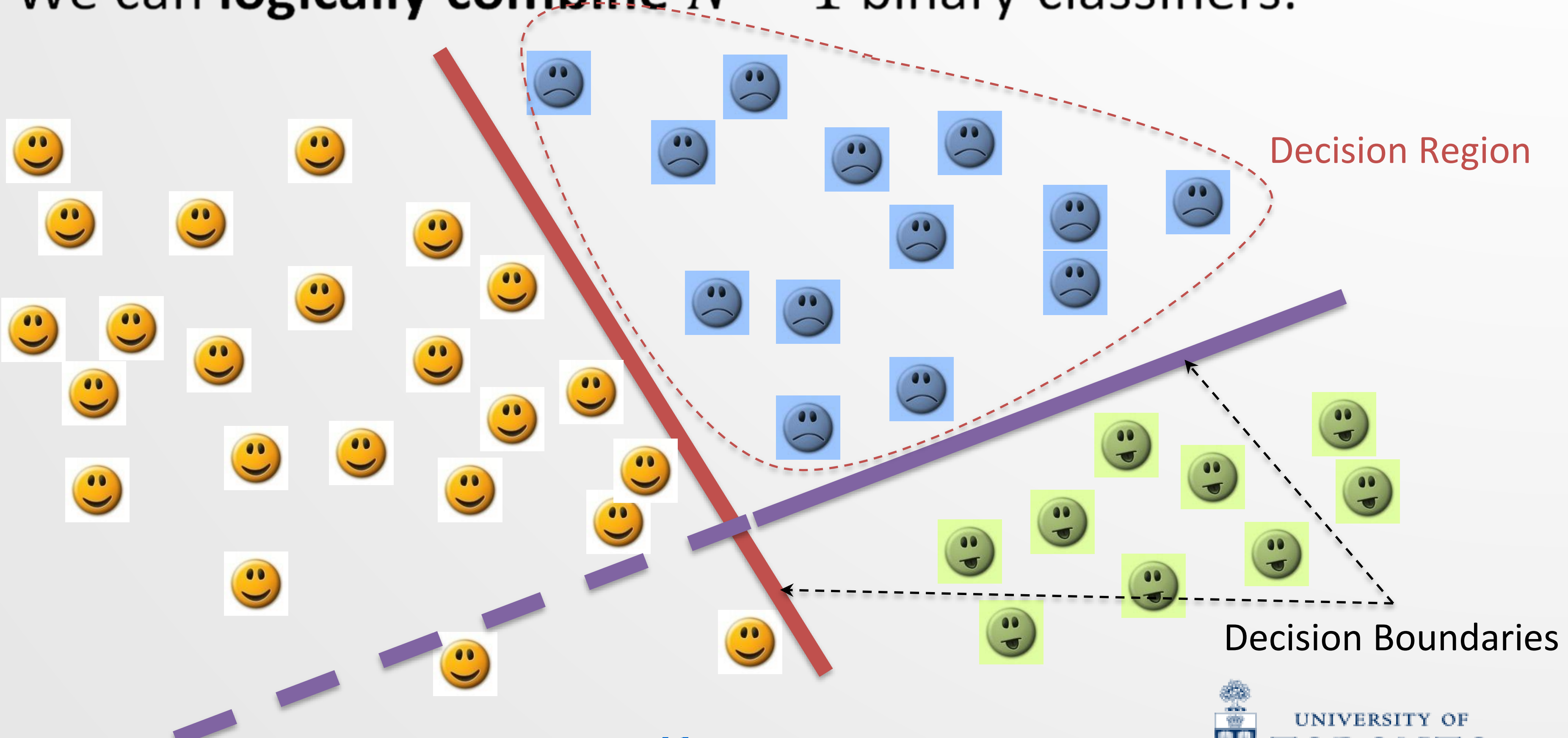
- Perhaps the easiest case.
  - Extends to dimensions $d \geq 3$, line becomes (hyper-)plane.



UNIVERSITY OF TORONTO

N-ary and linearly separable

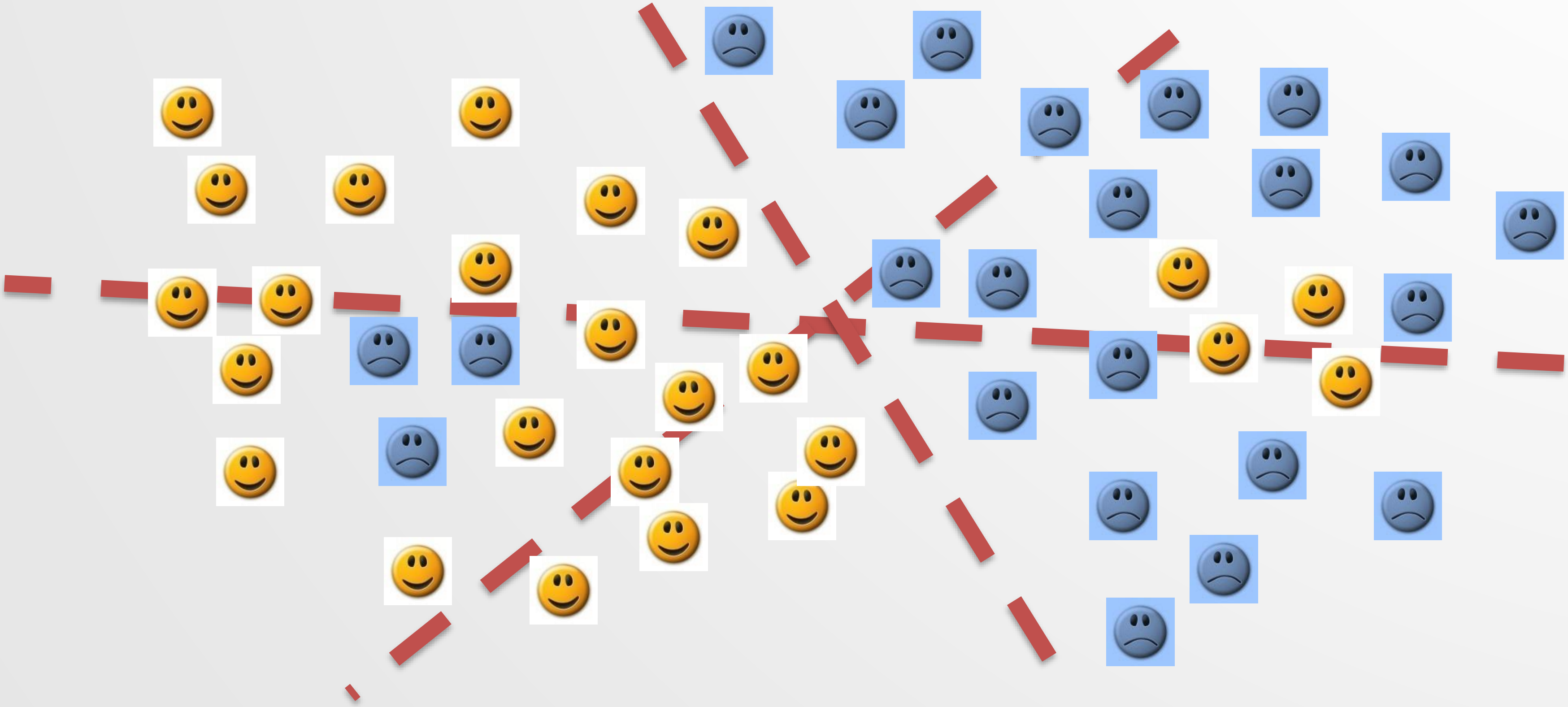
- A bit harder – random guessing gives $\frac{1}{N}$ accuracy (given equally likely classes).
  - We can **logically combine** $N-1$ binary classifiers.



UNIVERSITY OF TORONTO

Class holes

- Sometimes it can be impossible to draw *any* lines through the data to separate the classes.
  - *Are those troublesome points noise or real phenomena?*

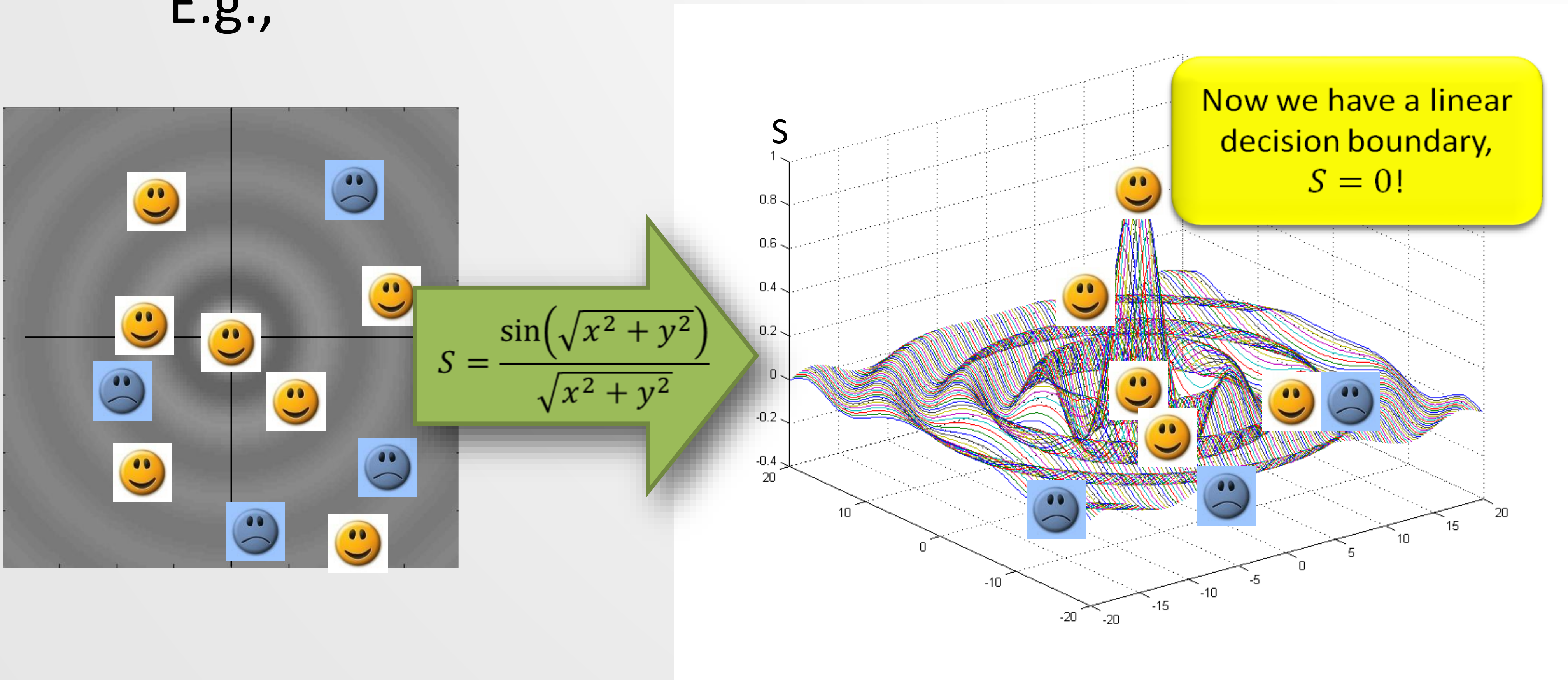


UNIVERSITY OF TORONTO

The kernel trick

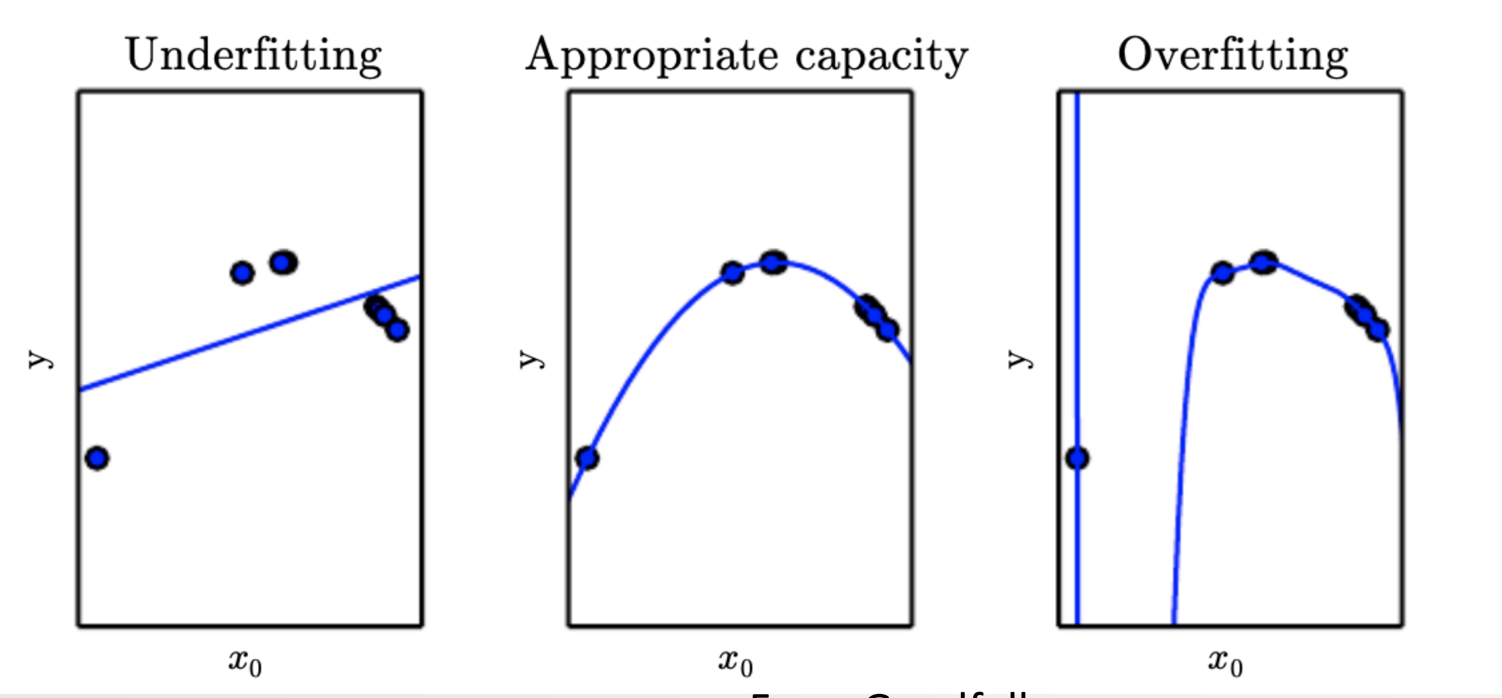
- We can sometimes linearize a non-linear case by moving the data into a higher dimension with a **kernel function**.

  E.g.,



Capacity and over/under-fitting

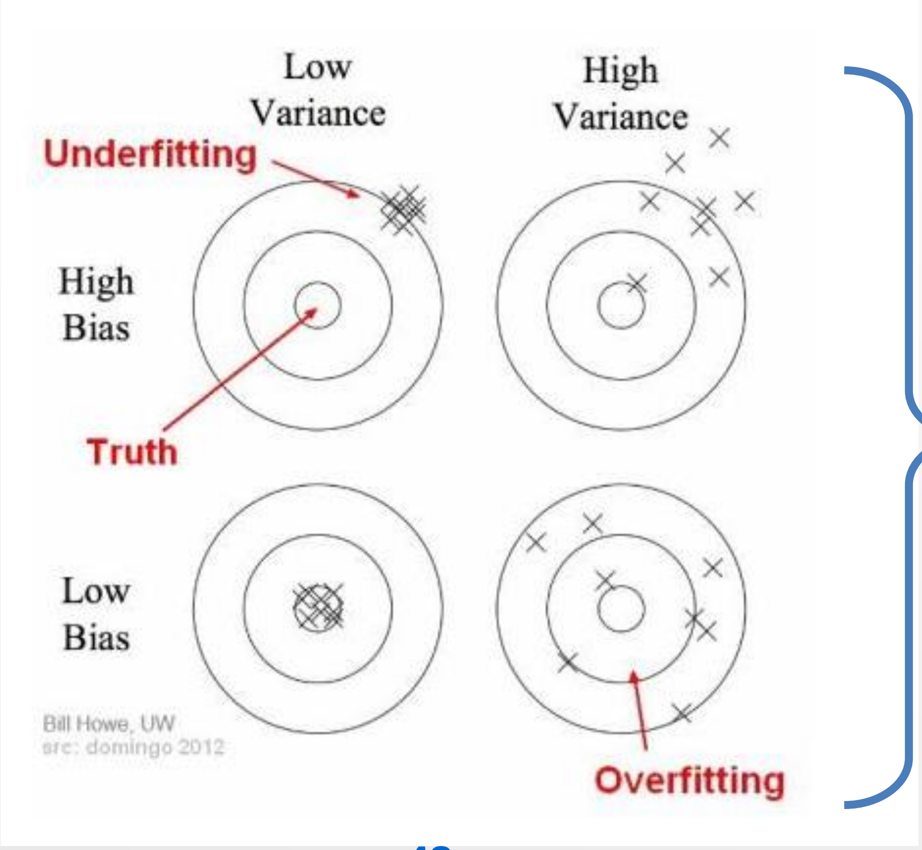
- A central challenge in machine learning is that our models should **generalize** to unseen data, so we need to set our (hyper-)parameters appropriately.



From Goodfellow

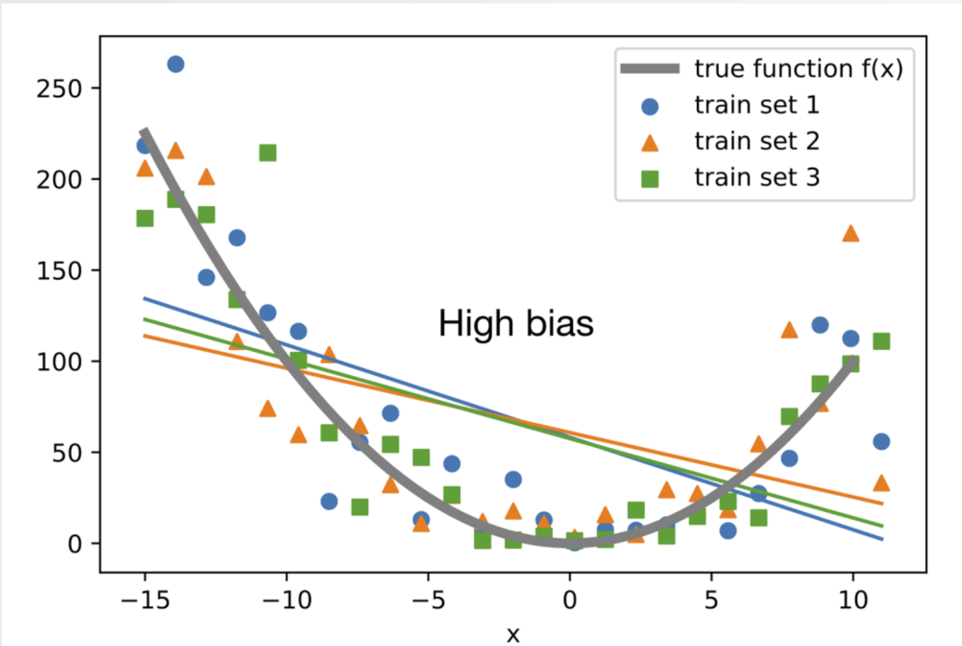
Capacity and over/under-fitting

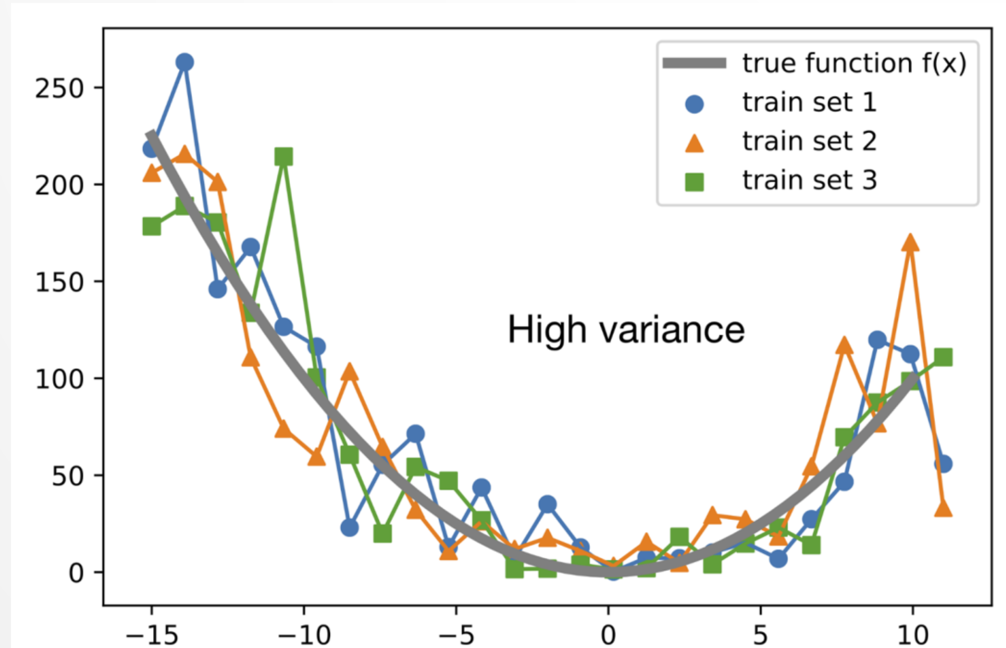
- A central challenge in machine learning is that our models should **generalize** to unseen data, so we need to set our (hyper-)parameters appropriately.

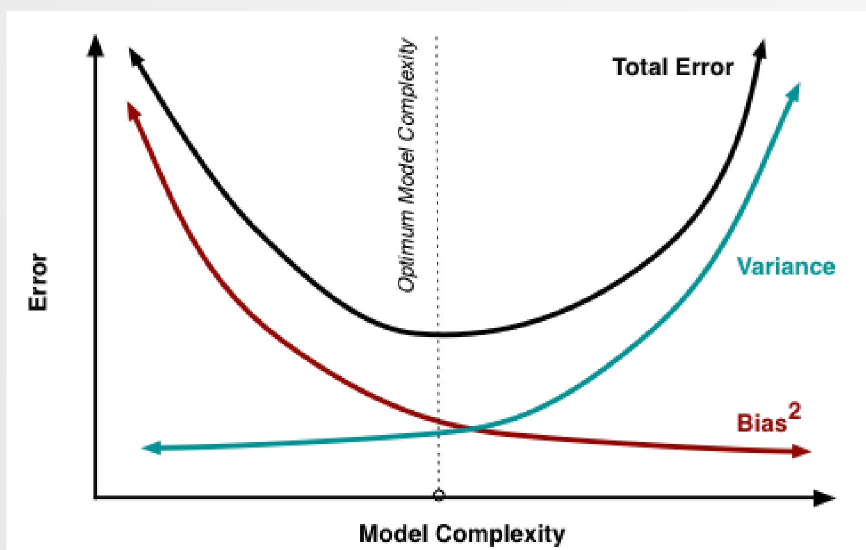


Symptoms of performance during inference of new data (×)

Bias and Variance







CSC 401/2511 – Spring 2024

Let’s summarize a few of the classifiers from Assignment 1

Decision Tree

From Technology Upskilling Machine Learning Software Foundations by En-Shiun Annie Lee

Trees!

(The … larch.)

Aspects of ID3

- **ID3** (iterative dichotomiser 3) is an algorithm invented by Ross Quinlan to produce decision trees from data.
- ID3 tends to build **short trees** since at each step we are removing the maximum amount of entropy possible.
- ID3 trains on the **whole training set** and does not succumb to issues related to **random initialization**.

- ID3 can **over-fit** to training data.
- Only **one attribute is used at a time** to make decisions
- It can be difficult to use **continuous** data, since many trees need to be generated to see where to break the continuum.

`from sklearn.ensemble import RandomForestClassifier`

Random Forests

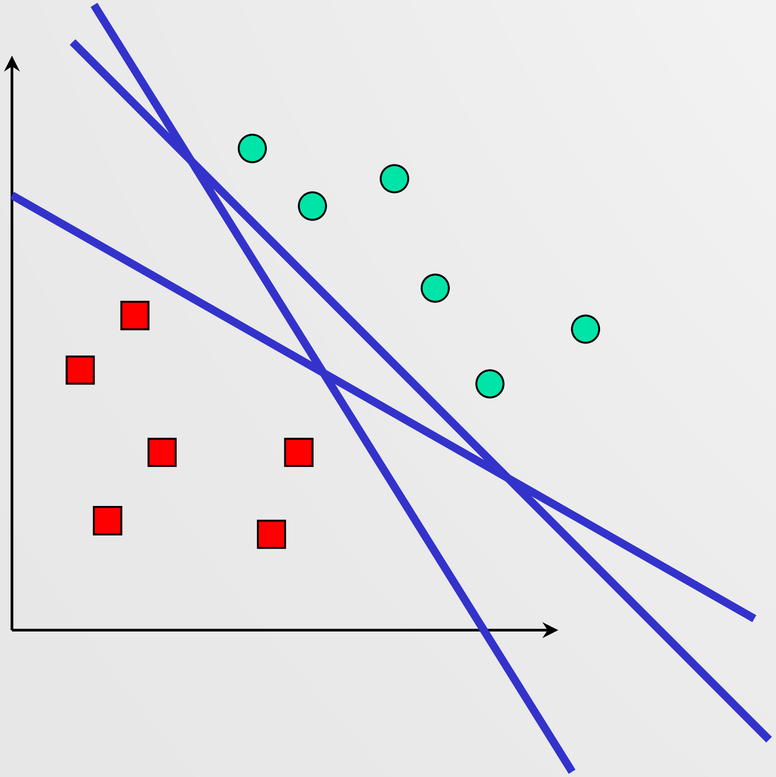
- **Random forests** *n.pl.* are **ensemble** classifiers that produce *K* decision trees, and output the **mode** class of those trees.
  - Can support continuous features.
  - Can support non-binary decisions.
  - Support cross-validation.

- The component trees in a random forest must differ.
  - Sometimes, decision trees are **pruned** randomly.
  - Usually, different trees accept different **subsets of features**.

That's a good idea – can we choose the best features in a reasonable way?

Support vector machines (SVMs)

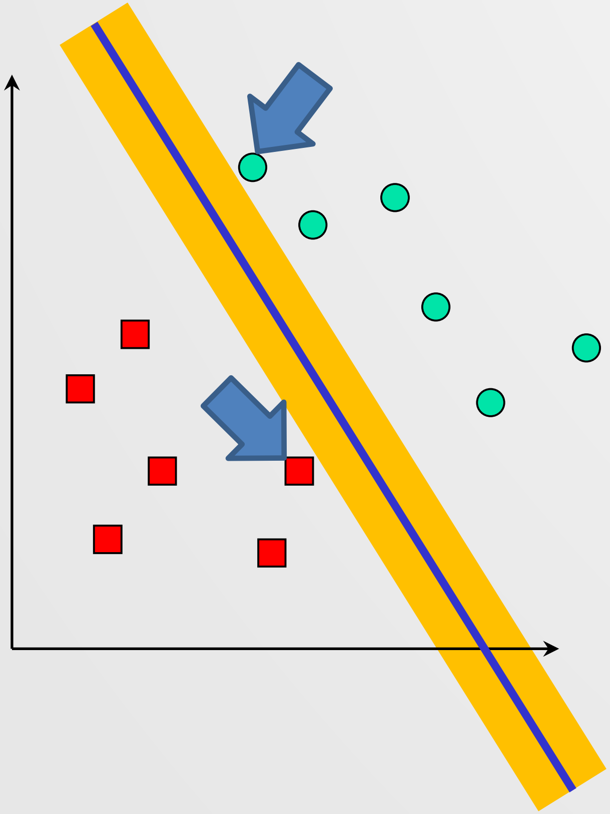
- In binary linear classification, two classes are assumed to be separable by a line (or plane). However, many possible separating planes might exist.



- Each of these blue lines separates the training data.
  - *Which line is the best?*

Support vector machines (SVMs)

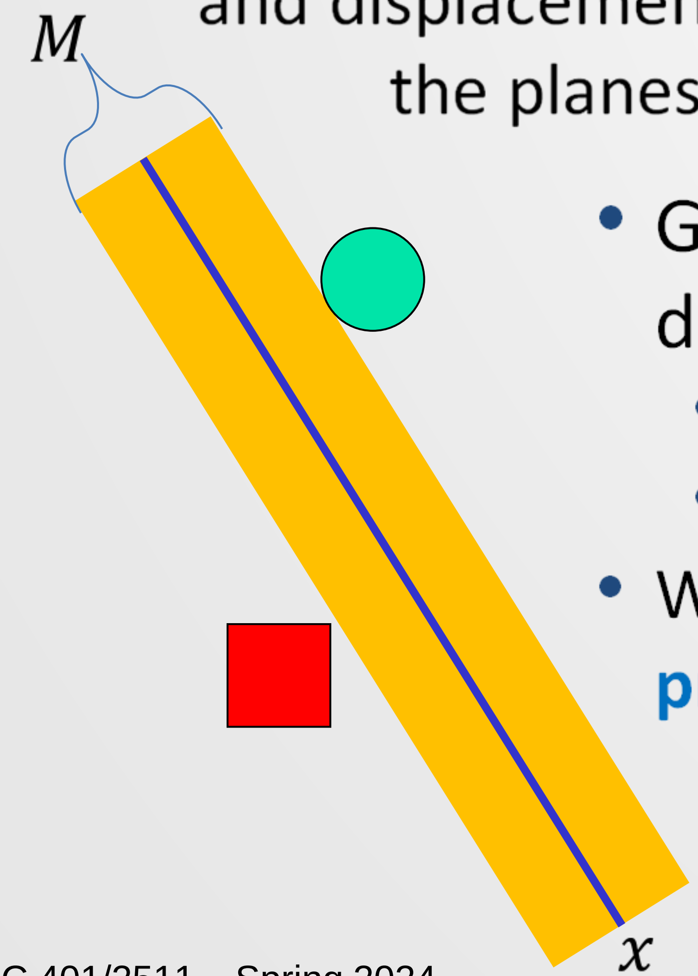
- The **margin** is the width by which the boundary could be **increased** before it hits a training datum.



- The **maximum margin linear classifier** is ∴ the linear classifier with the maximum margin.
- The **support vectors** (indicated) are those data points against which the margin is pressed.
- The bigger the margin – the less sensitive the boundary is to error.

Support vector machines (SVMs)

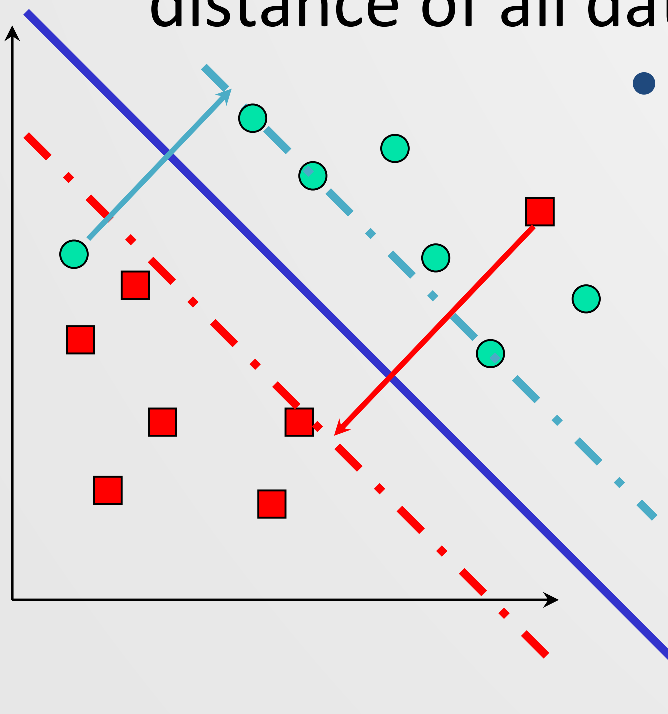
- The width of the margin, M, can be computed by the angle and displacement of the planar boundary, x, as well as the planes that touch data points.



- Given an initial guess of the angle and displacement of x we can compute:
  - whether all data is correctly classified,
  - The width of the margin, M.
- We update our guess by **quadratic programming**, which is semi-analytic.

Support vector machines (SVMs)

- The maximum margin helps SVMs **generalize** to situations when it's **impossible** to linearly separate the data.
  - We introduce a parameter that allows us to measure the distance of all data not in their correct 'zones'.



- We simultaneously **maximize the margin** while **minimizing the misclassification error**.
  - There is a straightforward approach to solving this system based on **quadratic programming**.

Support vector machines (SVMs)

- SVMs generalize to higher-dimensional data and to systems in which the data is non-linearly separable (e.g., by a circular decision boundary).
  - Using the **kernel trick** (from before) is common.

- Many binary SVM classifiers can also be combined to simulate a multi-category classifier.

- (Still) one of the most popular off-the-shelf classifiers.

Naïve Bayes and SoftMax

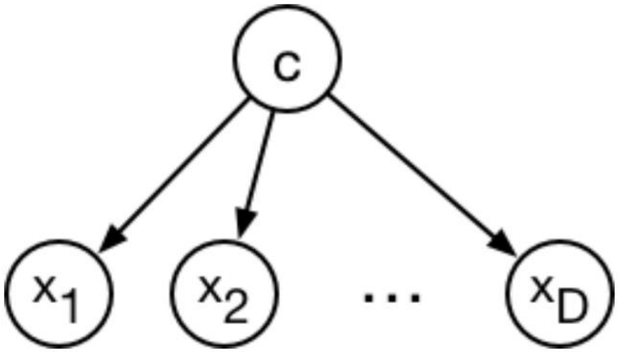
- Broadly, Bayesian probability conceives of probability *not* as frequency of some phenomenon occurring, but rather as an expectation related to our own certainty.
- Given an observation x, **Naïve Bayes** simply chooses the class $c \in C$ that maximizes $\boldsymbol{P(c \mid x)}$.
  - This can be done in many ways.

$$\underset{c}{\operatorname{argmax}} P(c|x) = \frac{P(c)}{\cancel{P(x)}} P(x|c)$$

Estimate the $P(\cdot)$ using Gaussians, or...

UNIVERSITY OF TORONTO

Bayesian Classifier



Given features $\mathbf{x} = [x_1, x_2, \cdots, x_D]^T$
want to compute class probabilities using Bayes Rule:

$$\underbrace{p(c|\mathbf{x})}_{\text{Pr. class given feature}} = \frac{\overbrace{p(\mathbf{x}|c)}^{\text{Pr. feature given class}} \quad p(c)}{p(\mathbf{x})}$$

In words,

$$\text{Posterior for class} = \frac{\text{Pr. of feature given class} \times \text{Prior for class}}{\text{Pr. of feature}}$$

To compute $p(c|\mathbf{x})$ we need: $p(\mathbf{x}|c)$ and $p(c)$.

CSC 401/2511 – Spring 2024

Independence Assumption

- Naive assumption: The features x_i are conditionally independent given the class c.
- Allows us to decompose the joint distribution:

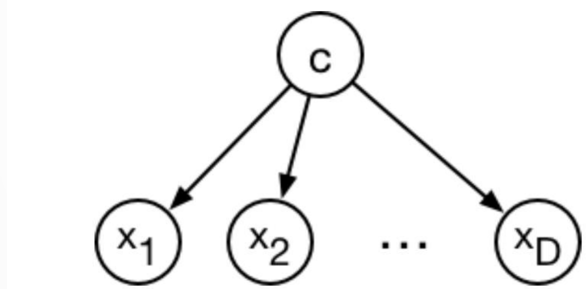
$$p(c, x_1, \ldots, x_D) = p(c)\, p(x_1|c) \cdots p(x_D|c).$$

- Compact representation of the joint distribution.
  - Prior probability of class: $p(c = 1) = \pi$
  - Conditional probability of feature given class:

$$p(x_j = 1|c) = \theta_{jc}$$

CSC 401/2511 – Spring 2024

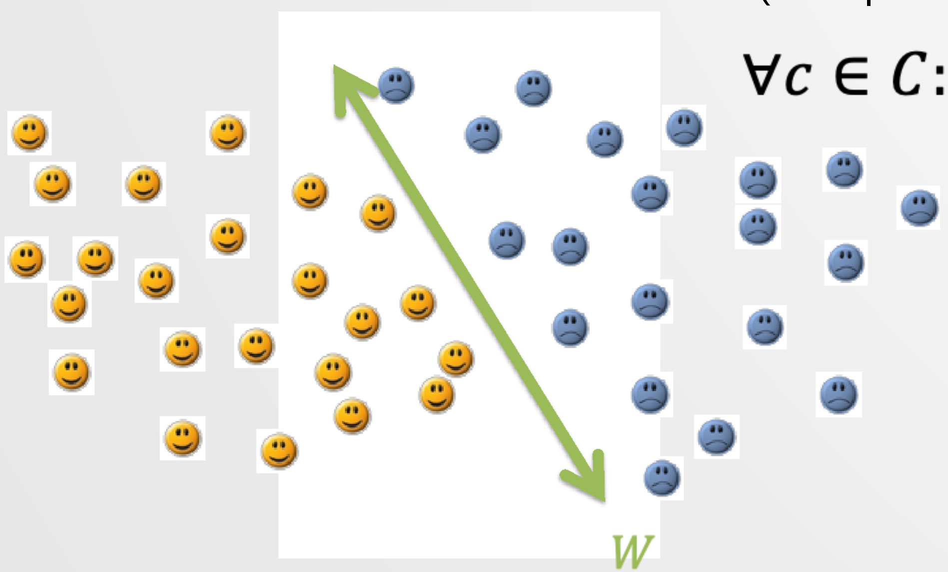
Naïve Bayes and SoftMax



- Assume $x \in \mathbb{R}^d$, learning a linear decision boundary is tantamount to learning $W \in \mathbb{R}^{C \times d}$.

P(Class|features) = P(features|Class)*P(Class)

$$\forall c \in C:\ \boldsymbol{f_c} = W[c, \cdots] \cdot x = \sum_{i=1}^{d} W[c, i] \cdot x[i]$$



Uh oh – $\boldsymbol{f_c}$ can be negative and we want something on $[0,1]$, to be a probability.

Solution: Just raise it with an exponent

Softmax:

$$P(y|x) = \frac{\exp(f_y)}{\sum_{c=1}^{C} \exp(f_c)}$$

Naive Bayes: https://www.youtube.com/watch?v=O2L2Uv9pdDA
SoftMax: https://www.youtube.com/watch?v=8ps_JEW42xs
Example on Text: https://www.youtube.com/watch?v=temQ8mHpe3k
Naive Bayes on Spam: https://youtu.be/M59h7CFUwPU
Why Naive Bayes are Cool: https://www.youtube.com/watch?v=8NEfN3JbINA

Naive Bayes Properties

- An amazingly cheap learning algorithm!
- **Training time**: Estimate parameters using maximum likelihood.
  - Compute co-occurrence counts of each feature with the labels. I Requires only one pass through the data!
- **Test time**: Apply Bayes' Rule.
  - Cheap because of the model structure. For more general models, Bayesian inference can be very expensive and/or complicated.
- Analysis easily extends to prob. distributions other than Bernoulli.
- Less accurate in practice compared to discriminative models due to its "naive" independence assumption.

CSC 401/2511 – Spring 2024

Features and classification

- We talked about:
  - How preprocessing can effect feature extraction.
  - What parts-of-speech are, and how to identify them.
  - How to prepare data for classification
  - SVMs
  - Decision trees (which are parts of random forests)

Lecture Review Slide

- Why do we use K-fold cross-validation and what should we watch out for?
- Explain Underfitting and Overfitting with respect to the Bias-Variance Tradeoff.
- What does SVM maximize and minimize simultaneously?
- What are some tricks that SVM uses?

Readings

- J&M: 5.1-5.5 (2nd edition)
- M&S: 16.1, 16.4

Appendix – prepositions from CELEX

of	540,085	through	14,964	worth	1,563	pace	12
in	331,235	after	13,670	toward	1,390	nigh	9
for	142,421	between	13,275	plus	750	re	4
to	125,691	under	9,525	till	686	mid	3
with	124,965	per	6,515	amongst	525	o'er	2
on	109,129	among	5,090	via	351	but	0
at	100,169	within	5,030	amid	222	ere	0
by	77,794	towards	4,700	underneath	164	less	0
from	74,843	above	3,056	versus	113	midst	0
about	38,428	near	2,026	amidst	67	o'	0
than	20,210	off	1,695	sans	20	thru	0
over	18,071	past	1,575	circa	14	vice	0

Appendix – particles

aboard	aside	besides	forward(s)	opposite	through
about	astray	between	home	out	throughout
above	away	beyond	in	outside	together
across	back	by	inside	over	under
ahead	before	close	instead	overhead	underneath
alongside	behind	down	near	past	up
apart	below	east, etc.	off	round	within
around	beneath	eastward(s),etc.	on	since	without

Appendix – conjunctions

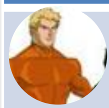
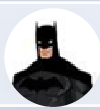
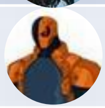
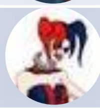
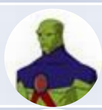
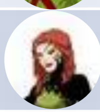
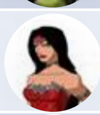
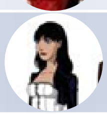
and	514,946	yet	5,040	considering	174	forasmuch as	0
that	134,773	since	4,843	lest	131	however	0
but	96,889	where	3,952	albeit	104	immediately	0
or	76,563	nor	3,078	providing	96	in as far as	0
as	54,608	once	2,826	whereupon	85	in so far as	0
if	53,917	unless	2,205	seeing	63	inasmuch as	0
when	37,975	why	1,333	directly	26	insomuch as	0
because	23,626	now	1,290	ere	12	insomuch that	0
so	12,933	neither	1,120	notwithstanding	3	like	0
before	10,720	whenever	913	according as	0	neither nor	0
though	10,329	whereas	867	as if	0	now that	0
than	9,511	except	864	as long as	0	only	0
while	8,144	till	686	as though	0	provided that	0
after	7,042	provided	594	both and	0	providing that	0
whether	5,978	whilst	351	but that	0	seeing as	0
for	5,935	suppose	281	but then	0	seeing as how	0
although	5,424	cos	188	but then again	0	seeing that	0
until	5,072	supposing	185	either or	0	without	0

Appendix – Penn TreeBank PoS tags

Tag	Description	Example	Tag	Description	Example
CC	coordin. conjunction	*and, but, or*	SYM	symbol	+,%, &
CD	cardinal number	*one, two, three*	TO	“to”	*to*
DT	determiner	*a, the*	UH	interjection	*ah, oops*
EX	existential ‘there’	*there*	VB	verb, base form	*eat*
FW	foreign word	*mea culpa*	VBD	verb, past tense	*ate*
IN	preposition/sub-conj	*of, in, by*	VBG	verb, gerund	*eating*
JJ	adjective	*yellow*	VBN	verb, past participle	*eaten*
JJR	adj., comparative	*bigger*	VBP	verb, non-3sg pres	*eat*
JJS	adj., superlative	*wildest*	VBZ	verb, 3sg pres	*eats*
LS	list item marker	*1, 2, One*	WDT	wh-determiner	*which, that*
MD	modal	*can, should*	WP	wh-pronoun	*what, who*
NN	noun, sing. or mass	*llama*	WP$	possessive wh-	*whose*
NNS	noun, plural	*llamas*	WRB	wh-adverb	*how, where*
NNP	proper noun, singular	*IBM*	$	dollar sign	$
NNPS	proper noun, plural	*Carolinas*	#	pound sign	#
PDT	predeterminer	*all, both*	“	left quote	‘ or “
POS	possessive ending	*’s*	”	right quote	’ or ”
PRP	personal pronoun	*I, you, he*	(	left parenthesis	[, (, {, <
PRP$	possessive pronoun	*your, one’s*	)	right parenthesis	],), }, >
RB	adverb	*quickly, never*	,	comma	,
RBR	adverb, comparative	*faster*	.	sentence-final punc	. ! ?
RBS	adverb, superlative	*fastest*	:	mid-sentence punc	: ; ... – -
RP	particle	*up, off*			

Example – Hero classification

Training data

Hero		Hair length	Height	Age	Hero Type
	Aquaman	2”	6’2”	35	Hero
	Batman	1”	5’11”	32	Hero
	Catwoman	7”	5’9”	29	Villain
	Deathstroke	0”	6’4”	28	Villain
	Harley Quinn	5”	5’0”	27	Villain
	Martian Manhunter	0”	8’2”	128	Hero
	Poison Ivy	6”	5’2”	24	Villain
	Wonder Woman	6”	6’1”	108	Hero
	Zatanna	10”	5’8”	26	Hero

Test data

	Red Hood	2”	6’0”	22	?

Example – Hero classification

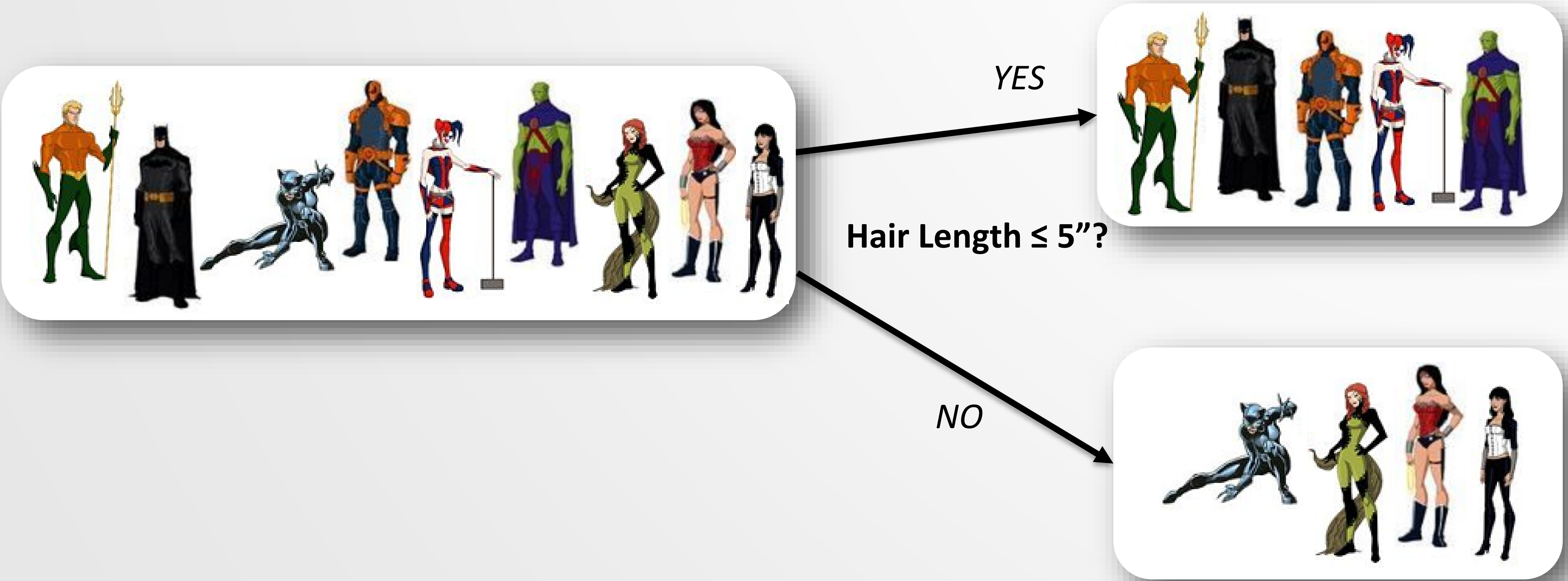
- How do we split?
  - Split on *hair length?*
  - Split on *height?*
  - Split on *age?*

- Let's compute the **information gain** for each:

$$Gain(Q) = H(S) - \sum_{child\ set} p(child\ set)H(child\ set)$$

Split on hair length?

$$Gain(Question) = H(S) - \sum_{child\ set} p(child\ set)H(child\ set)$$



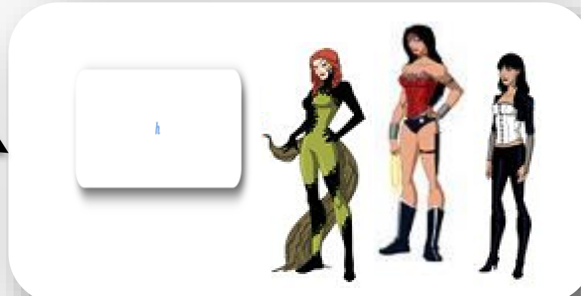
Split on hair length?

$$Gain(Question) = \boxed{H(S)} - \sum_{child\ set} p(child\ set)H(child\ set)$$

YES

Hair Length ≤ 5"?

NO



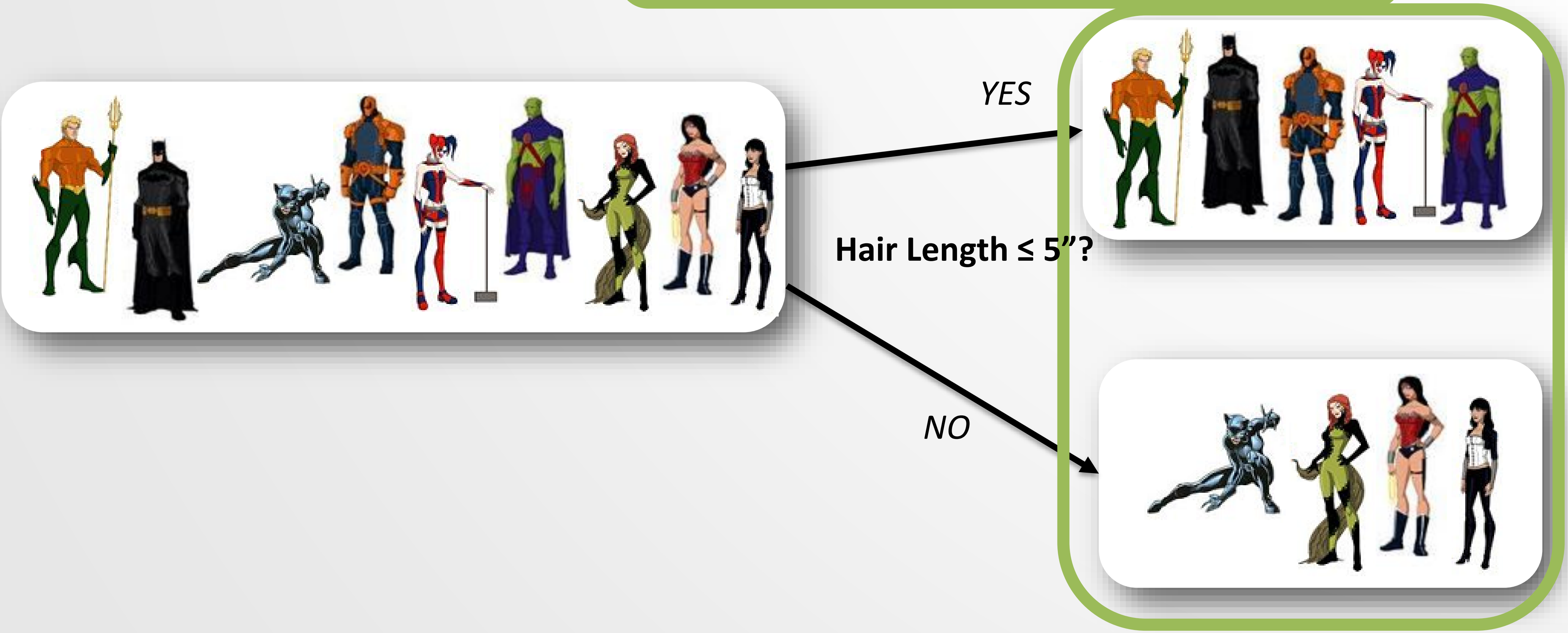
$$H(S) = \frac{h}{h+v}\log_2\left(\frac{h+v}{h}\right) + \frac{v}{h+v}\log_2\left(\frac{h+v}{v}\right)$$

$$H(5h, 4v) = \frac{5}{9}\log_2\left(\frac{9}{5}\right) + \frac{4}{9}\log_2\left(\frac{9}{4}\right) = \mathbf{0.9911}\text{ bits}$$

UNIVERSITY OF TORONTO

Split on hair length?

$$Gain(Question) = H(S) - \sum_{child\ set} p(child\ set)H(child\ set)$$



UNIVERSITY OF TORONTO

Split on hair length?

$$Gain(Question) = H(S) - \sum_{child\ set} p(child\ set)H(child\ set)$$

YES

Hair Length ≤ 5”?

NO

YES $H(\mathbf{4}\boldsymbol{h}, \mathbf{1}\boldsymbol{v}) = \frac{4}{5}\log_2\left(\frac{5}{4}\right) + \frac{1}{5}\log_2\left(\frac{5}{1}\right) = \mathbf{0.7219}$

UNIVERSITY OF TORONTO

Split on hair length?

$$Gain(Question) = H(S) - \sum_{child\ set} p(child\ set)H(child\ set)$$

YES

Hair Length ≤ 5”?

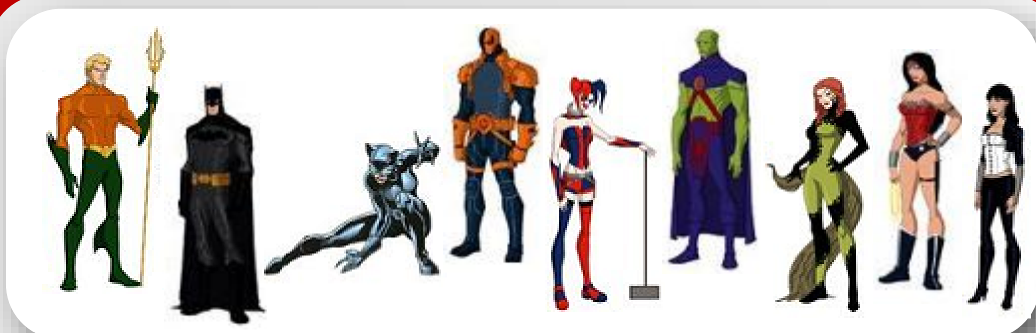
NO

YES $H(\mathbf{4}h, \mathbf{1}v) = \frac{4}{5}\log_2\left(\frac{5}{4}\right) + \frac{1}{5}\log_2\left(\frac{5}{1}\right) = \mathbf{0.7219}$

NO $H(\mathbf{2}h, \mathbf{2}v) = \frac{2}{4}\log_2\left(\frac{4}{2}\right) + \frac{2}{4}\log_2\left(\frac{4}{2}\right) = \mathbf{1}$

Split on hair length?

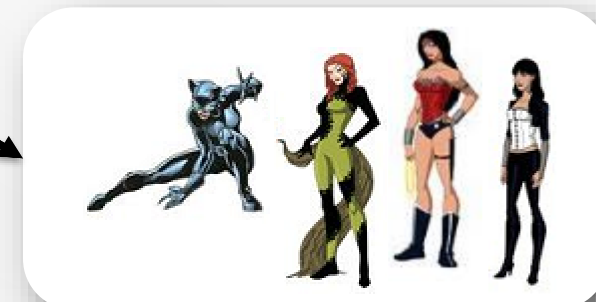
$$Gain(Question) = \boxed{H(S)} - \boxed{\sum_{child\ set} p(child\ set)H(child\ set)}$$



Hair Length ≤ 5”?

YES

NO



$$Gain(HairLength \leq 5") = \underbrace{0.9911} - \underbrace{\frac{5}{9}\mathbf{0.7219} - \frac{4}{9}\mathbf{1}} = \mathbf{0.00721}$$

UNIVERSITY OF TORONTO

Example – Hero classification

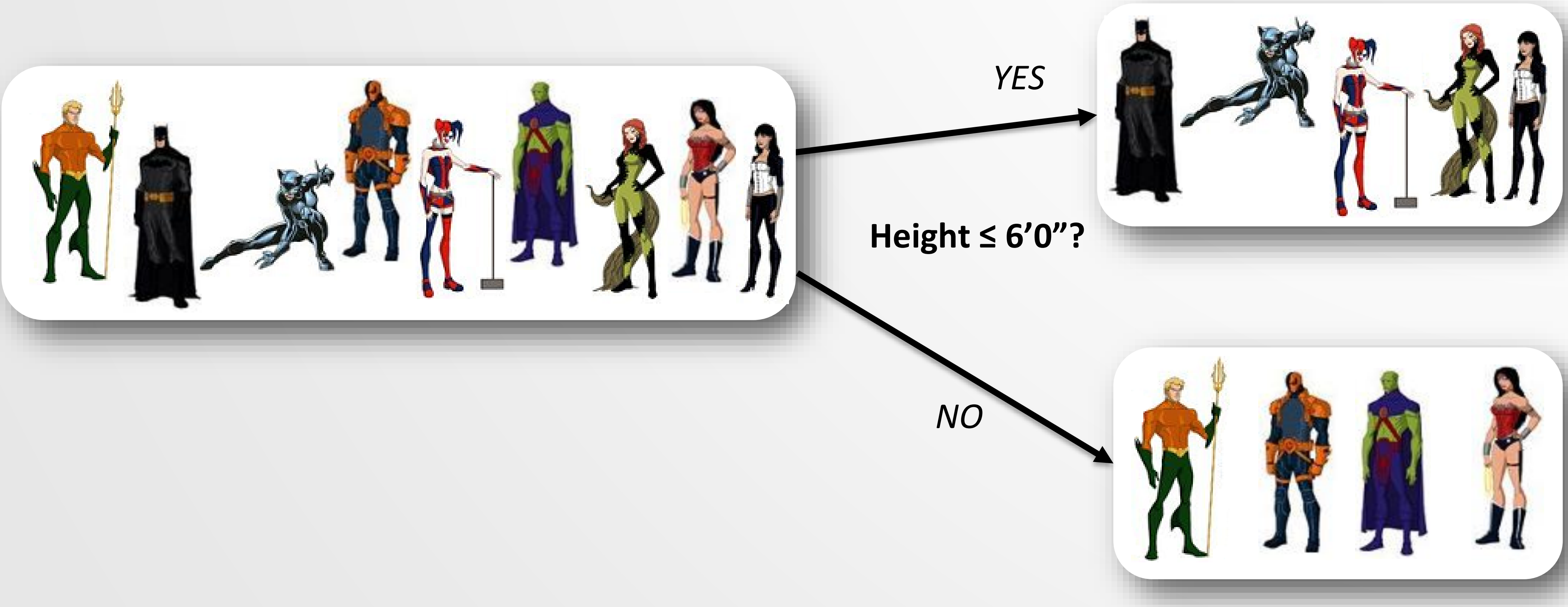
- How do we split?
  - Split on *hair length?* $Gain(HairLength \leq 5") = \mathbf{0.00721}$
  - Split on *height?*
  - Split on *age?*
- Let's compute the **information gain** for each:

$$Gain(Q) = H(S) - \sum_{child\ set} p(child\ set) H(child\ set)$$

UNIVERSITY OF TORONTO

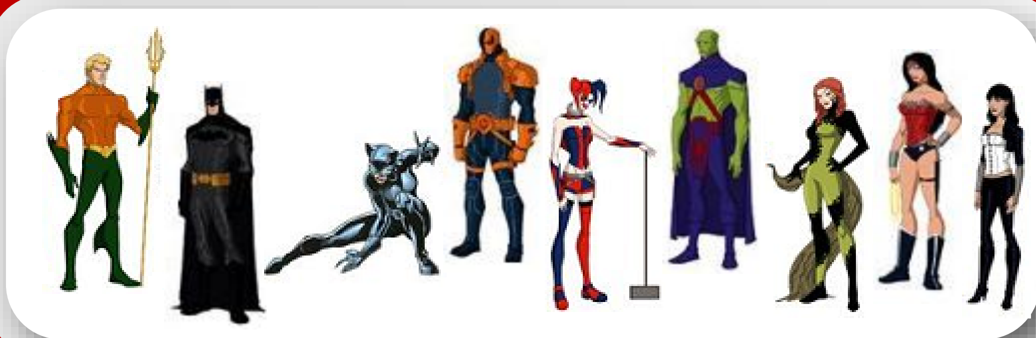
Split on height?

$$Gain(Question) = H(S) - \sum_{child\ set} p(child\ set)H(child\ set)$$



Split on height?

$$Gain(Question) = \boxed{H(S)} - \sum_{child\ set} p(child\ set)H(child\ set)$$



Height ≤ 6'0"?

YES

NO

$$H(S) = \frac{h}{h+v}\log_2\left(\frac{h+v}{h}\right) + \frac{v}{h+v}\log_2\left(\frac{h+v}{v}\right)$$

$$H(5h, 4v) = \frac{5}{9}\log_2\left(\frac{9}{5}\right) + \frac{4}{9}\log_2\left(\frac{9}{4}\right) = \mathbf{0.9911}\text{ bits}$$

UNIVERSITY OF TORONTO

Split on height?

$$Gain(Question) = H(S) - \sum_{child\ set} p(child\ set)H(child\ set)$$

YES

Height ≤ 6'0"

NO

YES $H(\mathbf{2}h, \mathbf{3}v) = \frac{2}{5}\log_2\left(\frac{5}{2}\right) + \frac{3}{5}\log_2\left(\frac{5}{3}\right) = \mathbf{0.971}$

NO $H(\mathbf{3}h, \mathbf{1}v) = \frac{3}{4}\log_2\left(\frac{4}{3}\right) + \frac{1}{4}\log_2\left(\frac{4}{1}\right) = \mathbf{0.813}$

UNIVERSITY OF TORONTO

Split on height?

$$Gain(Question) = \boxed{H(S)} - \boxed{\sum_{child\ set} p(child\ set)H(child\ set)}$$

YES

Height ≤ 6’0”

NO

$$Gain(Height \leq 6'0") = 0.9911 - \frac{5}{9}[0.971] - \frac{4}{9}[0.813] = 0.0903$$

Example – Hero classification

- How do we split?
  - Split on *hair length?* $Gain(HairLength \leq 5") = \mathbf{0.00721}$
  - Split on *height?*
  - Split on *age?*

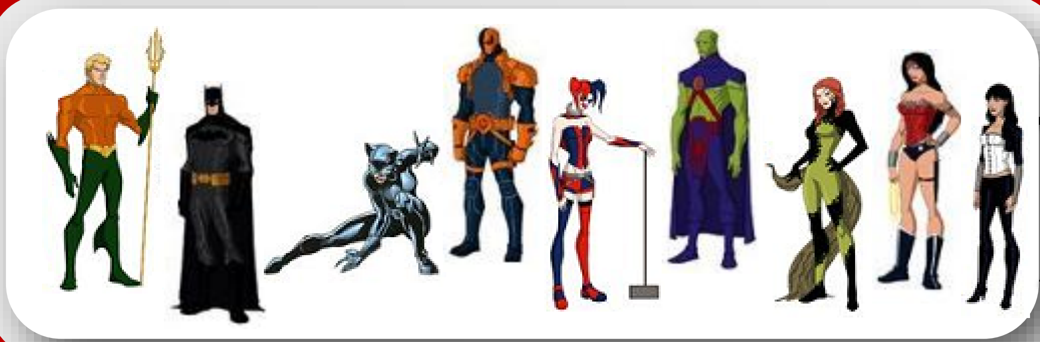
- Let's compute the **information gain** for each:

$$Gain(Q) = H(S) - \sum_{child\ set} p(child\ set)H(child\ set)$$

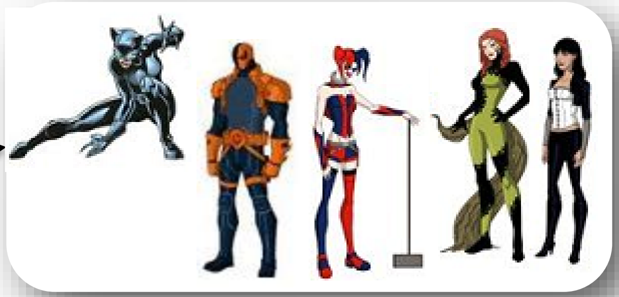
UNIVERSITY OF TORONTO

Split on age?

$$Gain(Question) = \boxed{H(S)} - \sum_{child\ set} p(child\ set)H(child\ set)$$



YES



Age ≤ 30?

NO

$$H(S) = \frac{h}{h+v}\log_2\left(\frac{h+v}{h}\right) + \frac{v}{h+v}\log_2\left(\frac{h+v}{v}\right)$$

$$H(5h, 4v) = \frac{5}{9}\log_2\left(\frac{9}{5}\right) + \frac{4}{9}\log_2\left(\frac{9}{4}\right) = \mathbf{0.9911}\text{ bits}$$

Split on age?

$$Gain(Question) = H(S) - \sum_{child\ set} p(child\ set)H(child\ set)$$

Age ≤ 30?

YES

NO

YES $H(\mathbf{1}h, \mathbf{4}v) = \frac{1}{5}\log_2\left(\frac{5}{1}\right) + \frac{4}{5}\log_2\left(\frac{5}{4}\right) = \mathbf{0.722}$

NO $H(\mathbf{4}h, \mathbf{0}v) = \frac{4}{4}\log_2\left(\frac{4}{4}\right) + \frac{0}{4}\log_2(\infty) = \mathbf{0}$

Split on age?

$$Gain(Question) = H(S) - \sum_{child\ set} p(child\ set)H(child\ set)$$

$$Gain(Age \le 30) = 0.9911 - \frac{5}{9}[0.722] - \frac{4}{9}[0] = 0.590$$

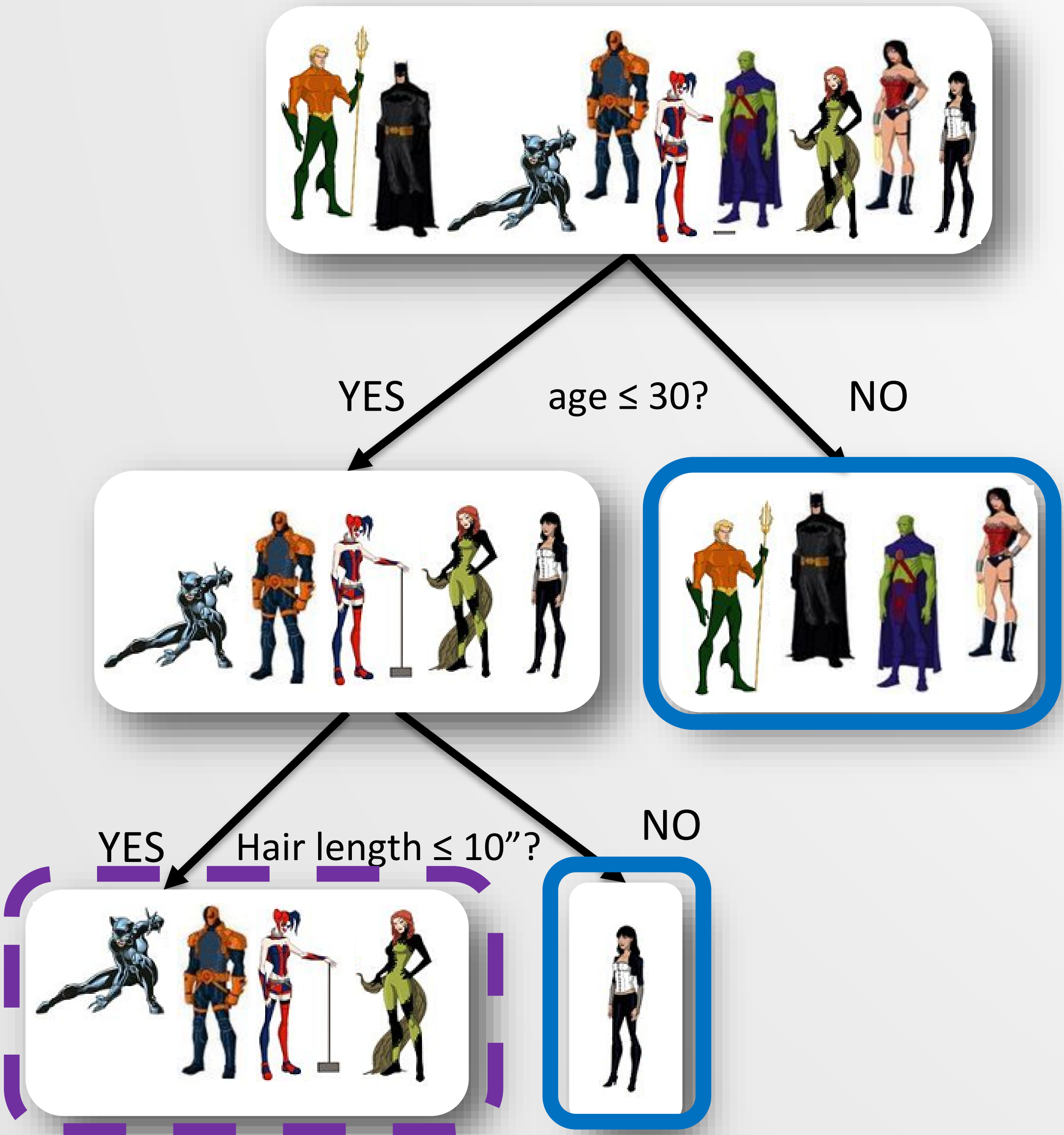
Example – Hero classification

- How do we split?
  - Split on *hair length?* $Gain(HairLength \leq 5") = \mathbf{0.00721}$
  - Split on *height?* $Gain(Height \leq 6'0") = \mathbf{0.0903}$
  - Split on *age?* $Gain(Age \leq 30) = \mathbf{0.590}$

- Let's compute the **information gain** for each:

$$Gain(Q) = H(S) - \sum_{child\ set} p(child\ set)H(child\ set)$$

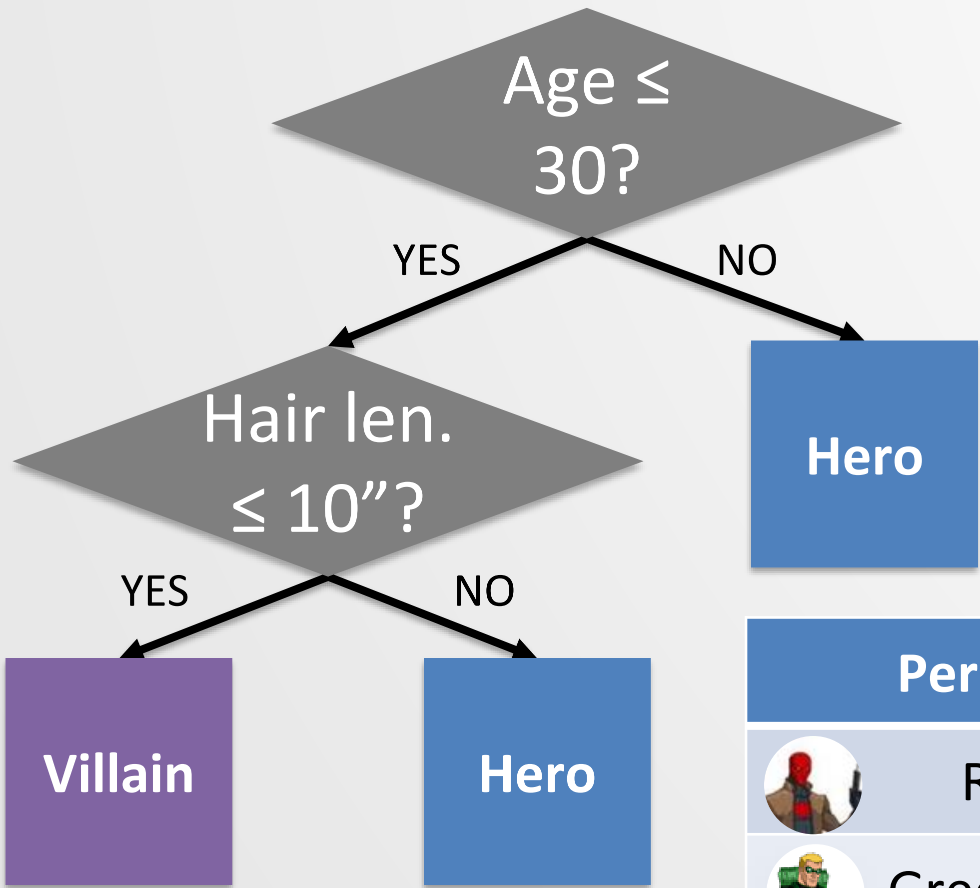
The resulting tree



- Splitting on *age* resulted in the greatest information gain.
- We’re left with one heterogeneous set, so we recurse and find that *hair length* results in a complete classification of the training data.

UNIVERSITY OF TORONTO

Testing



- We just need to keep track of the attribute questions – not the training data.
- How are the following characters classified?

Person	Hair length	Height	Age
Red Hood	2”	6’0”	22
Green Arrow	1”	6’2”	38
Bane	0”	5’8”	29

- Inspired from Allan Neymark’s (San Jose State University) Simpsons example.