Course: Intelligent Systems

Unit 4: Language Technologies

Language technologies
Part 3

Mariano Rico
2022
Technical University of Madrid

NLP at a glance

- Session 1 (**29th Nov**)
  - Encodings
  - Corpus
  - Normalization
  - Hands-on 1
- Session 2 (**13th Dec**)
  - Part of Speech
  - Sparse Vector models
  - TF-IDF
  - Sentiment analysis
  - Hands-on 2
- Session 3 (**Today 20 Dec**)
  - Document classification
  - Information extraction
  - Hands-on 3
- Session 4 (after Xmas, **Tue 10 Jan**)
  - The neural revolution
  - Language Models 4 NLP tasks
  - Hands-on 4

Table of Contents

DOCUMENT CLASSIFICATION

Dataset

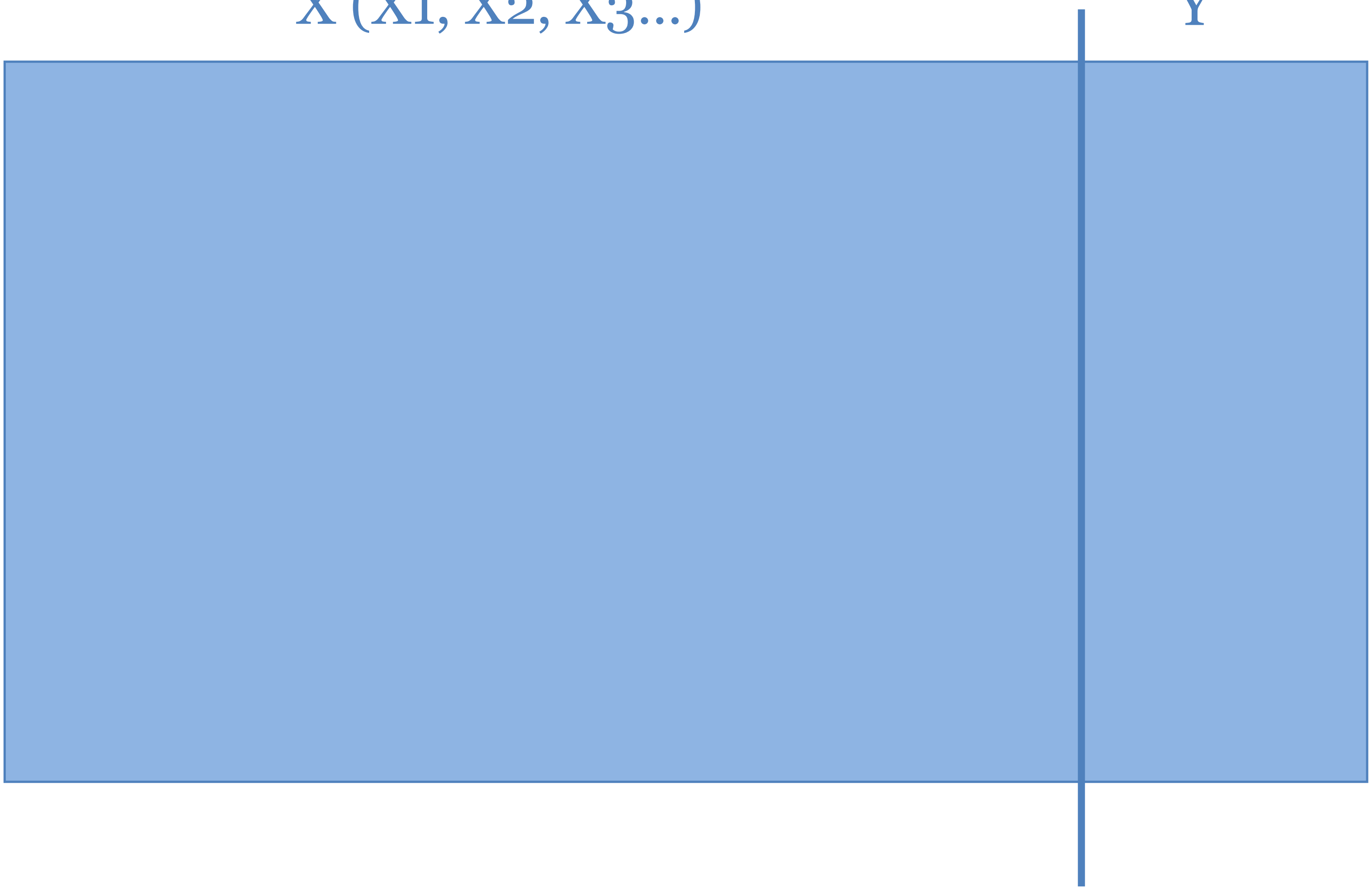
Data
Frame

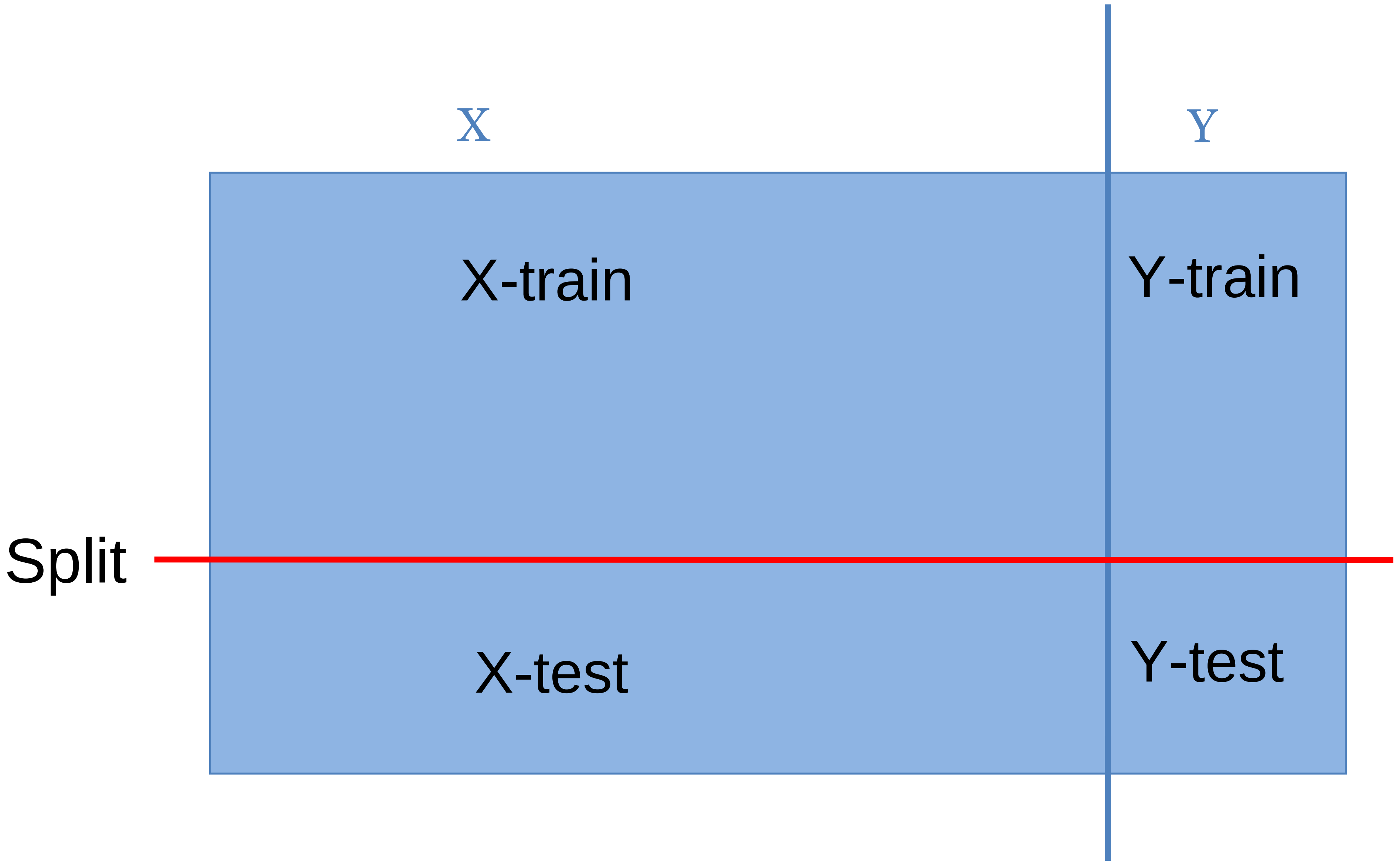
Independent variables,
features, characteristics…

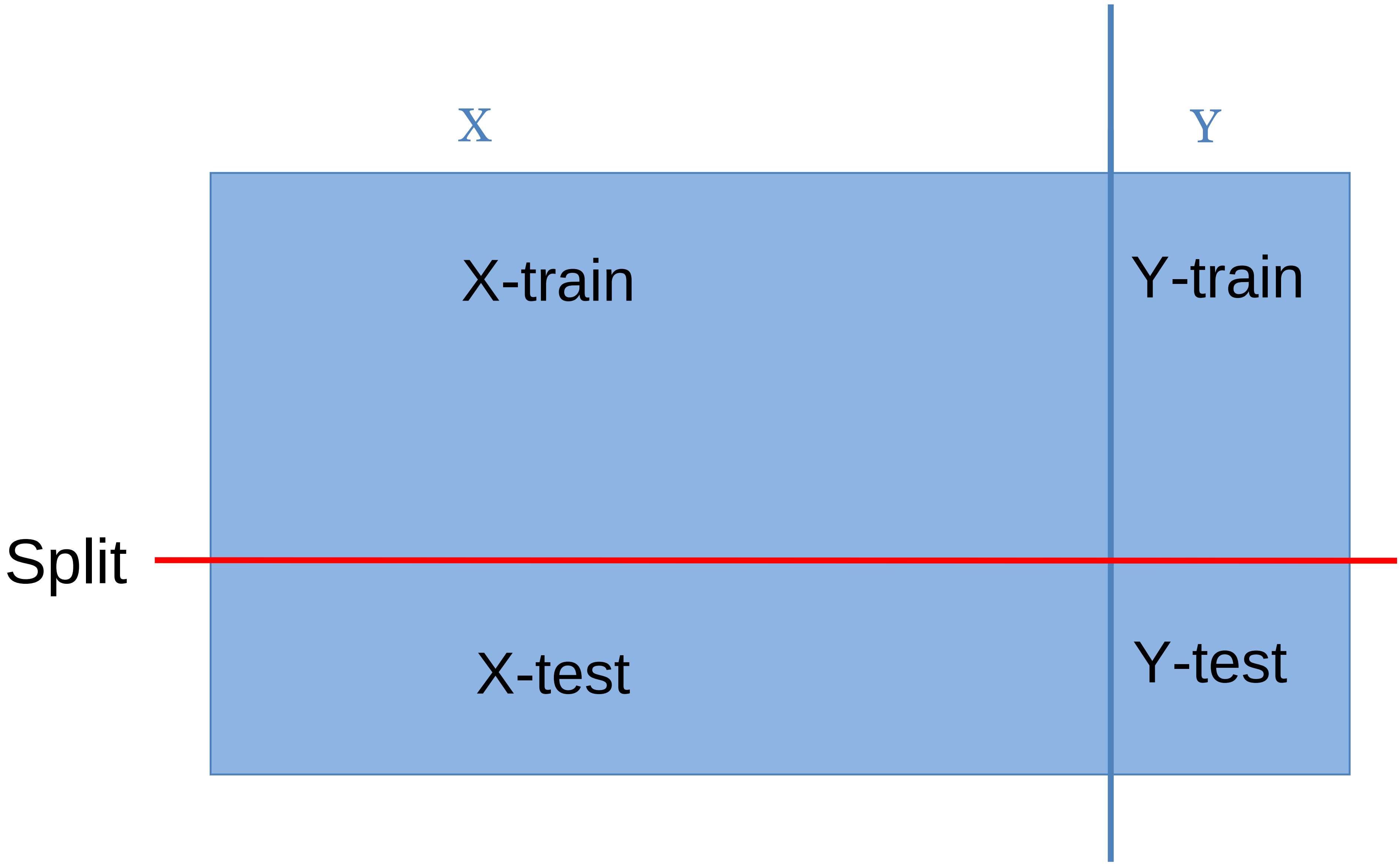
Dependent
variable,
class…

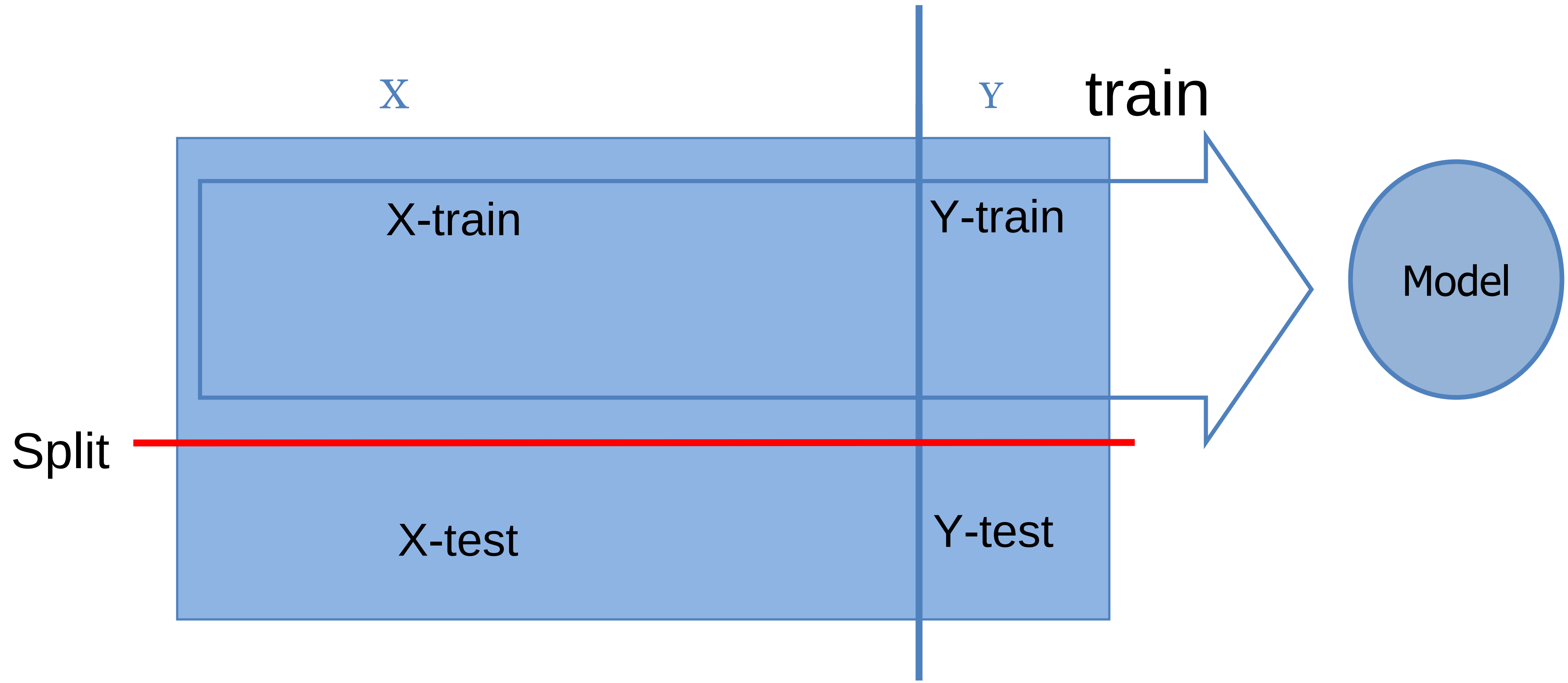
X (X_1, X_2, X_3…)

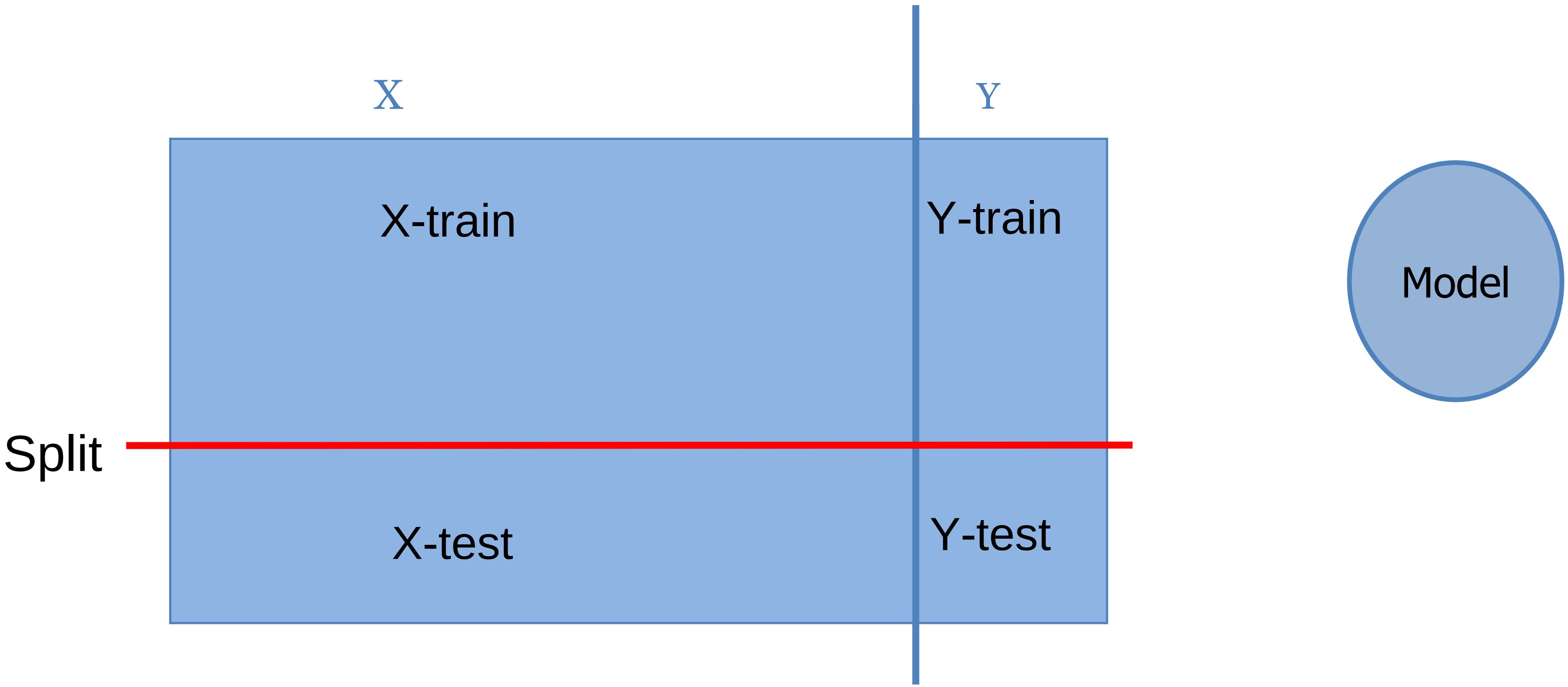
Y

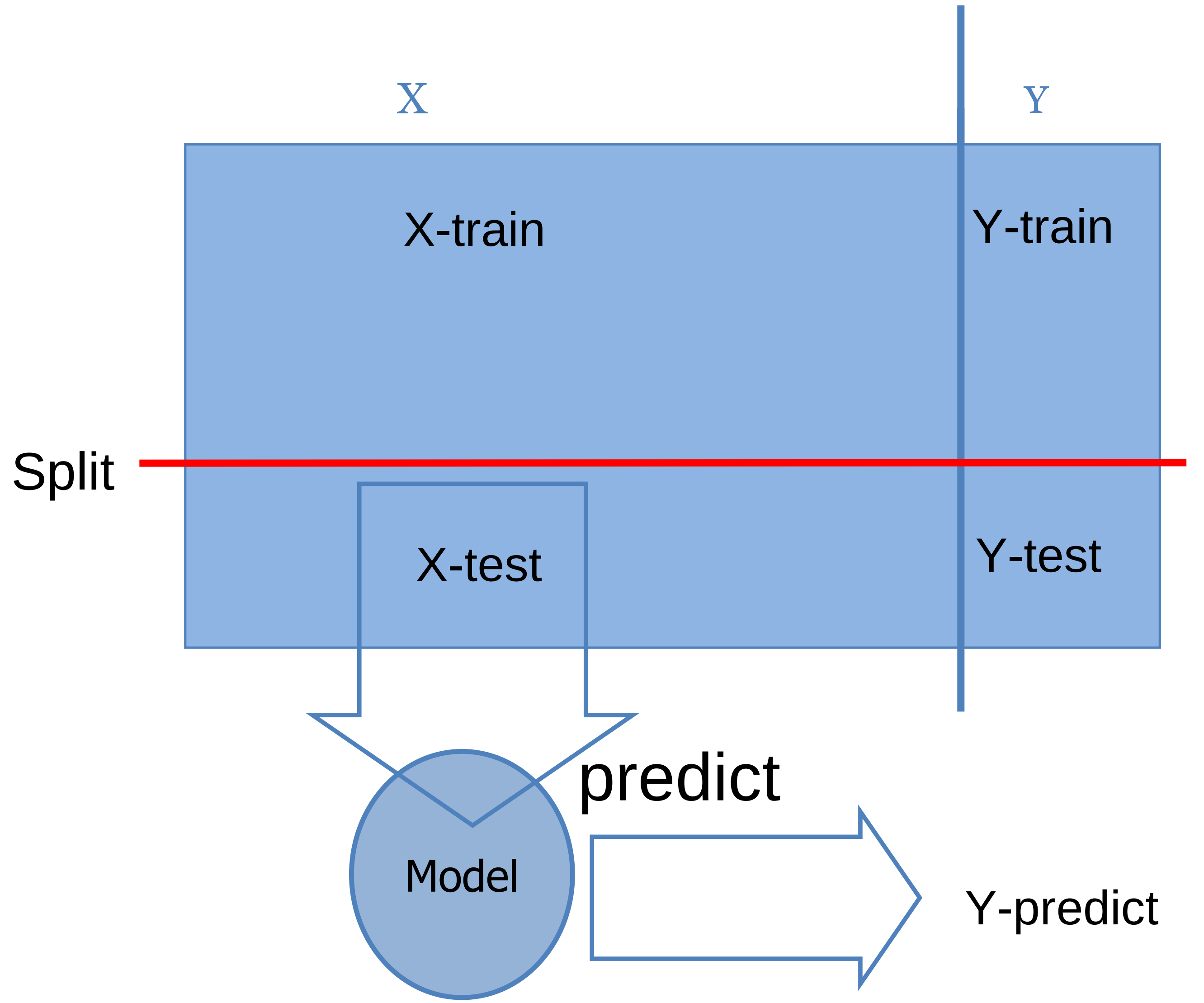


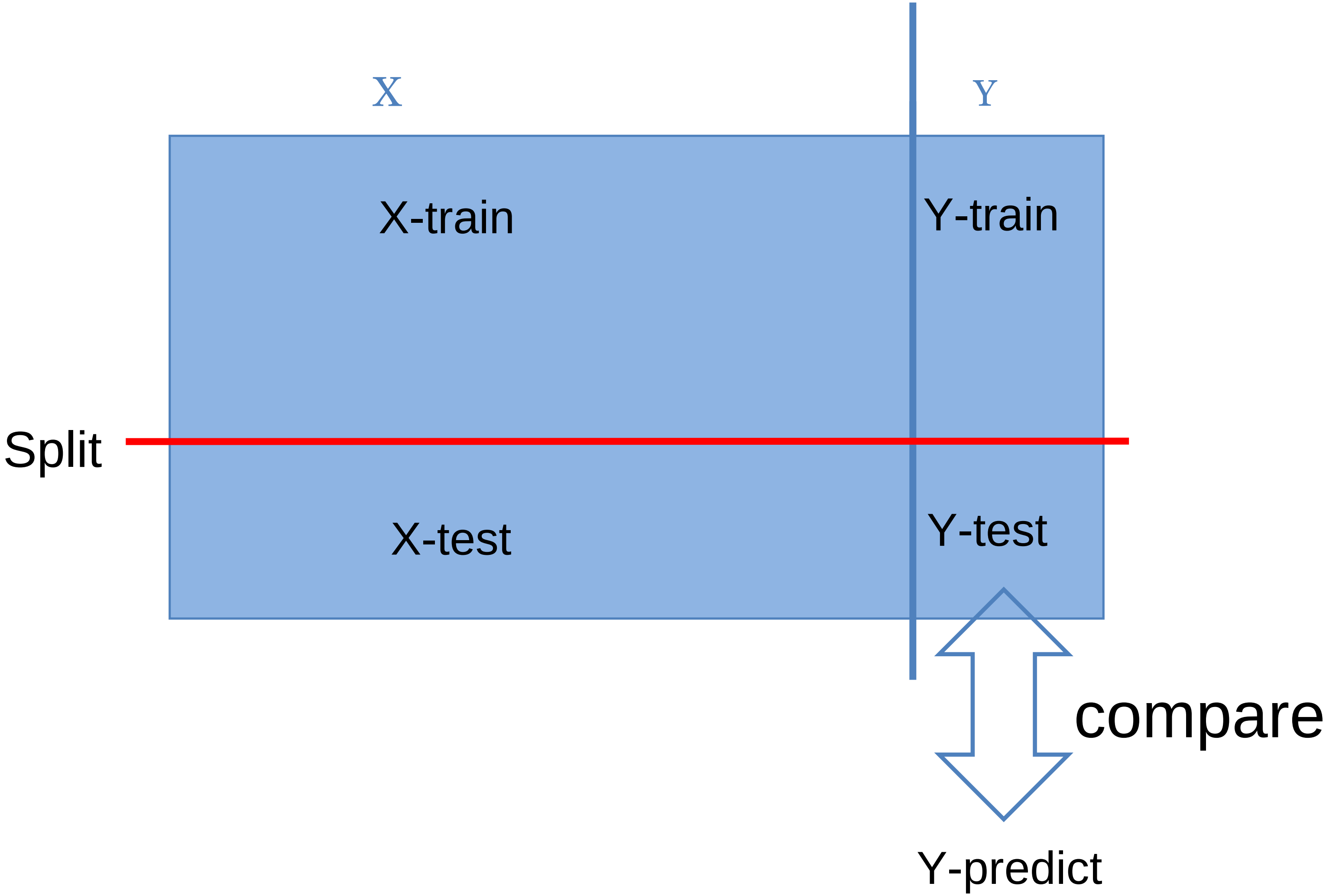

X
Y
X-train
Y-train
Split
X-test
Y-test


X
Y
X-train
Y-train
Split
X-test
Y-test


X
Y
train
X-train
Y-train
Model
Split
X-test
Y-test

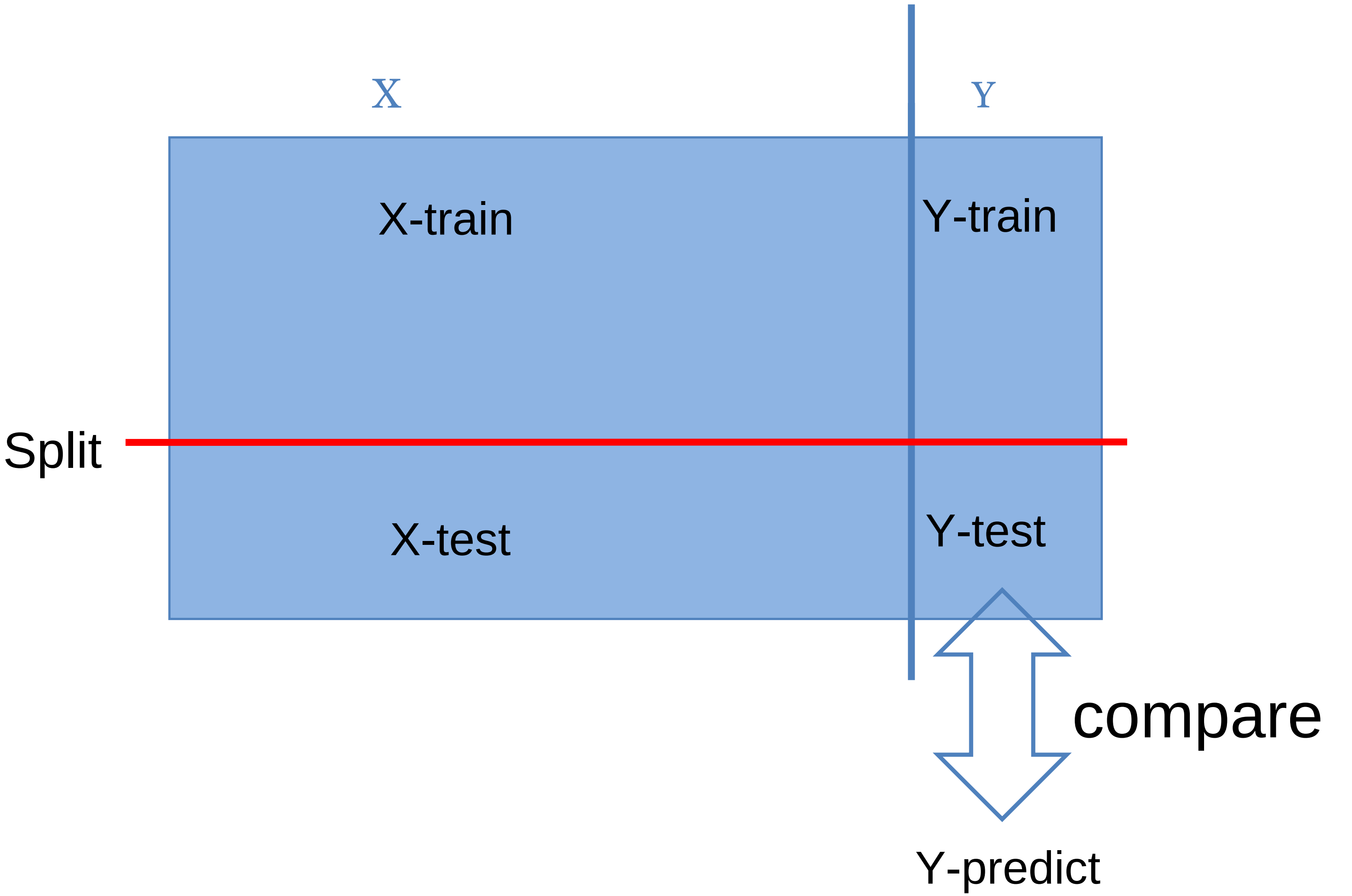

X
Y
X-train
Y-train
Model
Split
X-test
Y-test


X
Y
X-train
Y-train
Split
X-test
Y-test
predict
Model
Y-predict


X
Y
X-train
Y-train
Split
X-test
Y-test
compare
Y-predict

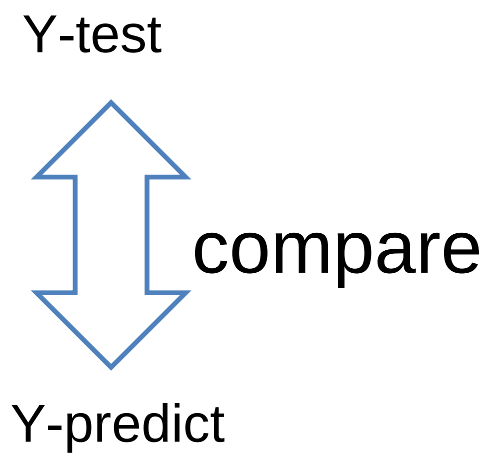
Classification

MODELS: EVALUATION


X
Y
X-train
Y-train
Split
X-test
Y-test
compare
Y-predict

How do we measure if prediction (**binary**) is good?

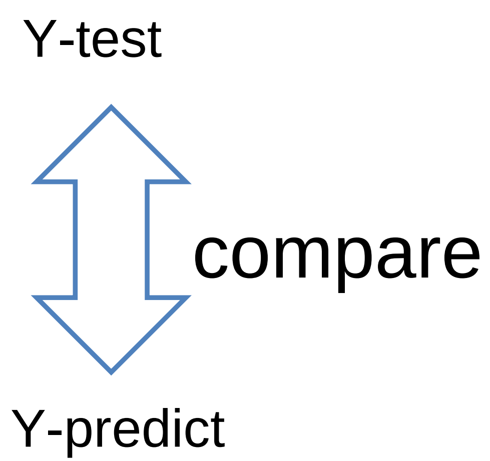
		Y-test	
		pos	neg
Y-predict	pos		
	neg		



How do we measure if prediction (**binary**) is good?

We use the **confusión matrix**

		Y-test	
		pos	neg
Y-predict	pos	True positives (*TP*)	False positives (*FP*)
	neg	False negativos (*FN*)	True negatives (*TN*)



How do we measure if prediction (**binary**) is good?

We use the **confusión matrix**,
and **calcule *p* (precision) and *r* (*recall*)**

		Y-test	
		pos	neg
Y-predict	pos	True positives (*TP*)	False positives (*FP*)
	neg	False negatives (*FN*)	True negatives (*TN*)

$$p = \frac{TP}{TP + FP}$$

$$r = \frac{TP}{TP + FN}$$

An example with spam detection:

Each email is classified as ***spam*** or ***normal***
The confusion matrix is this:

Total emails: 60+50+30+200 = 340

We know (Y-test) than 60+30=90 are normal

50+200=250 are spam

Our predictor (Y-predict) says that 60+50 = 110 are normal

30+200 = 230 are spam

		Y-test	
		pos (normal)	neg (spam)
Y-predict	pos (normal)	True positives (*TP*) = 60	False positives (*FP*) = 50
	neg (spam)	False negatives (*FN*) = 30	True negatives (*VN*)=200

$$p = \frac{TP}{TP + FP}$$

$$= \frac{60}{60 + 50}$$

$$= 0.54 \text{ (54\%)}$$

$$r = \frac{TP}{TP + FN} = \frac{60}{60 + 30} = 0.66 = 66\%$$

If we have more than two classes (is not binary):

Example: each email is classified as ***spam***, ***normal***, or as ***urgent***.

		Y-test		
		urgent	normal	spam
Y-predict	urgent	8	10	1
	normal	5	60	50
	spam	3	30	200

$$p_{urgent} = \frac{8}{8+10+1}$$

$$p_{normal} = \frac{60}{5+60+50}$$

$$p_{spam} = \frac{200}{3+30+200}$$

$$r_u = \frac{8}{8+5+3} \quad r_n = \frac{60}{10+60+30} \quad r_s = \frac{200}{1+50+200}$$

Example from Jurafsky 2021

INFORMATION EXTRACTION

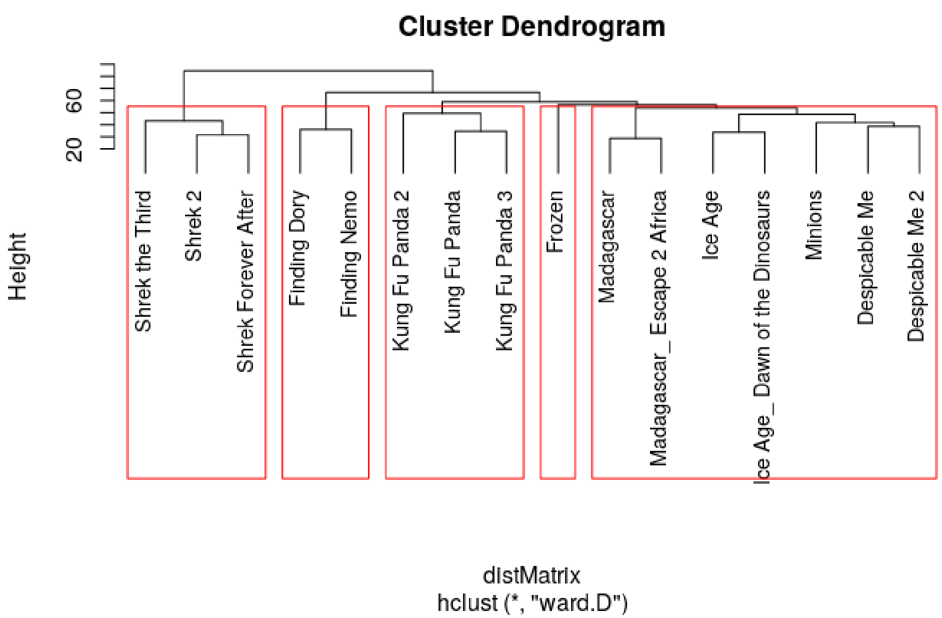
Text classification

From R

- Dendrograms with `hclust()`

```
m  <- as.matrix(dtm)
distMatrix <- dist(m, method="euclidean")

groups <- hclust(distMatrix, method="ward.D")
plot(groups, cex=0.9, hang=-1)
rect.hclust(groups, k=5)
```



Source: [here]

Text classification

From R

- Package quanteda.textmodels (in CRAN). Has 8 basic models for quanteda corpora
  - The simplest is the Naive Bayes classifier
    - Function `textmodel_nb()`. With 2 types of distributions:
      - » Multinomial
      - » Bernoulli
  - A more advanced (SVM)
    - Function `textmodel_svm()`
- Package quanteda.classifiers (**no** in CRAN). Advanced models for quanteda corpora
  - Two classifiers (using neuronal networks)
    - Multilevel perceptron network
    - Convolutional neural network + LSTM model fitted to word embeddings

Named entities (NEs)

- The process is *NER = NE Recognition*
- 4 basic types
  - **PER** (*Person*). Example: “Madam Curie”, “Marie Curie”
  - **LOC** (*Location*). Example: “Nueva York”, “New York”
  - **ORG** (*Organization*). Example: “Universidad de Stanford”
  - **GPE** (*Geo-political entity*). Example: “Teruel, España”, “Comunidad de Madrid”
- Extended types (thangs that, *a priori,* are not entities)
  - Date
  - Hours
  - Prices

Named entities (NEs)

- An example of NER Annotation

  Citing high fuel prices, [$_{\text{ORG}}$ **United Airlines**] said [$_{\text{TIME}}$ **Friday**] it has increased fares by [$_{\text{MONEY}}$ **$6**] per round trip on flights to some cities also served by lower-cost carriers. [$_{\text{ORG}}$ **American Airlines**], a unit of [$_{\text{ORG}}$ **AMR Corp.**], immediately matched the move, spokesman [$_{\text{PER}}$ **Tim Wagner**] said. [$_{\text{ORG}}$ **United**], a unit of [$_{\text{ORG}}$ **UAL Corp.**], said the increase took effect [$_{\text{TIME}}$ **Thursday**] and applies to most routes where it competes against discount carriers, such as [$_{\text{LOC}}$ **Chicago**] to [$_{\text{LOC}}$ **Dallas**] and [$_{\text{LOC}}$ **Denver**] to [$_{\text{LOC}}$ **San Francisco**].

  – There are 13 NEs (5 **org**anizations, 4 **loc**ations, 2 dates (**TIME**), 1 **per**son, and one price (**MONEY**))

Named entities

Tagging formats

- NEs use to be formed by several words (e.g. “Don Quijote de la Mancha”)
  - What labels do we add to each of these words?
- There are several formats:
  - BIO labelling
    - B for *Begin*
    - I for *Inside*
    - O for *Outside*

Words	BIO Label
Jane	B-PER
Villanueva	I-PER
of	O
United	B-ORG
Airlines	I-ORG
Holding	I-ORG
discussed	O
the	O
Chicago	B-LOC
route	O
.	O

–

Words	IO Label	BIO Label	BIOES Label
Jane	I-PER	B-PER	B-PER
Villanueva	I-PER	I-PER	E-PER
of	O	O	O
United	I-ORG	B-ORG	B-ORG
Airlines	I-ORG	I-ORG	I-ORG
Holding	I-ORG	I-ORG	E-ORG
discussed	O	O	O
the	O	O	O
Chicago	I-LOC	B-LOC	S-LOC
route	O	O	O
.	O	O	O

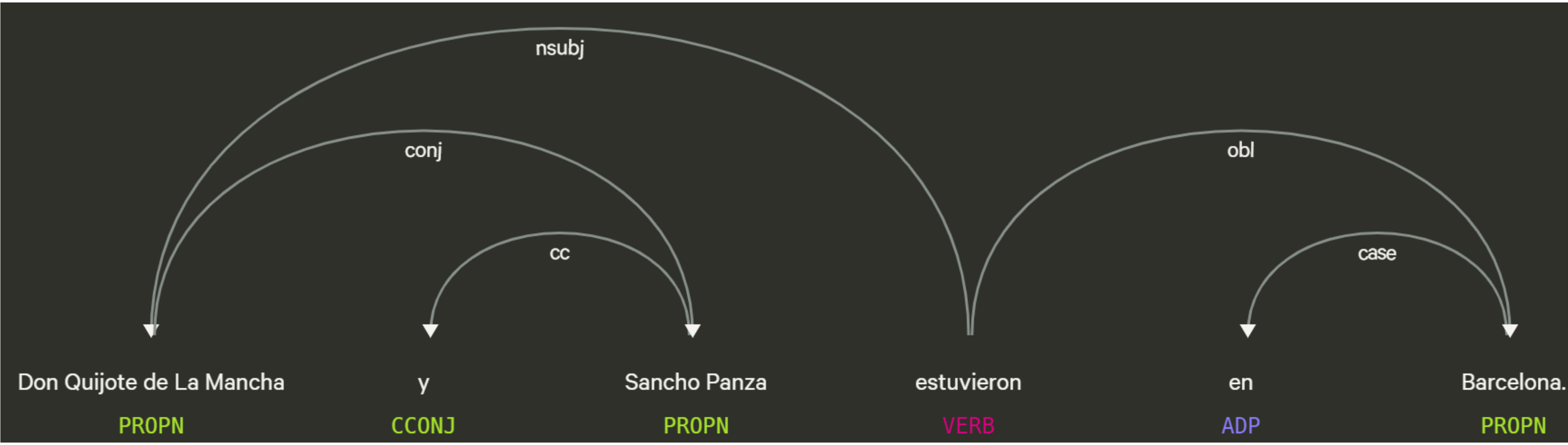
Named Entities

using R

- In the `spacyr` package there is *NER* for several languages (Spanish among them)
  - Doesn't follow any of the shown tagging formats ☹
  - But it is quite similar ☺

Dependencies

- Relations between the elements of a sentence
  - The root is the verb (principal) of the sentence
    - The non principal verbs are the aux
  - The subjet (nsubj, nominal subject)
    - The arrow *head* is the subject. The tail is the verb



Dependencies

- The current standard is UD2.0
  - [Dependency types]

	Nominals	Clauses	Modifier words	Function Words
Core arguments	nsubj obj iobj	csubj ccomp xcomp		
Non-core dependents	obl vocative expl dislocated	advcl	advmod* discourse	aux cop mark
Nominal dependents	nmod appos nummod	acl	amod	det clf case
Coordination	**MWE**	**Loose**	**Special**	**Other**
conj cc	fixed flat compound	list parataxis	orphan goeswith reparandum	punct root dep

Dependencies

using R

- In the `spacyr` package there is dependency extraction
- Also the `udpipes` package

Relation extraction

lexical patterns

- Hearst patterns (Martha Alice Hearst, 1992)
  - She proposed 5 patterns to identify **hyponyms**
  - For English
  - Easily extensible to any other language

"*Word whose meaning includes that of another*". Sparrow *is* hyponym *of* bird. "Subclass of", "is-a".

NP {, NP}* {,} (and|or) other NP_H	temples, treasuries, and other important civic buildings
NP_H such as {NP,}* {(or|and)} NP	red algae such as Gelidium
such NP_H as {NP,}* {(or|and)} NP	such authors as Herrick, Goldsmith, and Shakespeare
NP_H {,} including {NP,}* {(or|and)} NP	common-law countries, including Canada and England
NP_H {,} especially {NP}* {(or|and)} NP	European countries, especially France, England, and Spain

- NP is *Noun Phrase* (in Spanish, sintagma nominal)
- NP_H is the parent (upper class, most generic)
- {A} indicates that A is optional
- {A}* indicates that can be repeated

Course: Intelligent Systems

Unit 4: Language Technologies

Language technologies
Part 3

Mariano Rico
2022
Technical University of Madrid