


Data collection

Profesor: Juan R. Rico 

Técnicas de Aprendizaje Automático

Dpto. Lenguajes y Sistemas Informáticos. Universidad de Alicante

Data
collection

Feature
Extraction

Typology of data
Example

Selection

Preprocessing

References

1. Feature Extraction

Feature extraction

Data
collection

Feature
Extraction

Typology of data
Example

Selection

Preprocessing

References

- It is a typical initial phase in machine learning, pattern recognition, or signal processing problems.
- It starts from an initial set of measured data, or this set is designed and built to extract **features** that are intended to be informative and non-redundant.
- These features facilitate subsequent learning and generalization steps.
- Feature extraction is also related to dimensionality reduction ([Sarangi et al., 2020](#)).
- Choosing a subset of the initial features is called feature selection ([Parsons, 2010](#)), which aims to allow learning while maintaining the properties of the initial set.

Classification of data according to their structure

Data
collection

Feature
Extraction

Typology of data

Example

Selection

Preprocessing

References

Structured

Vector (tabular)



Chain/String



Tree/Hierarchical



Graph



Unstructured

Image



Audio



Video



Text



Example of Feature Extraction I

Data
collection

Feature
Extraction

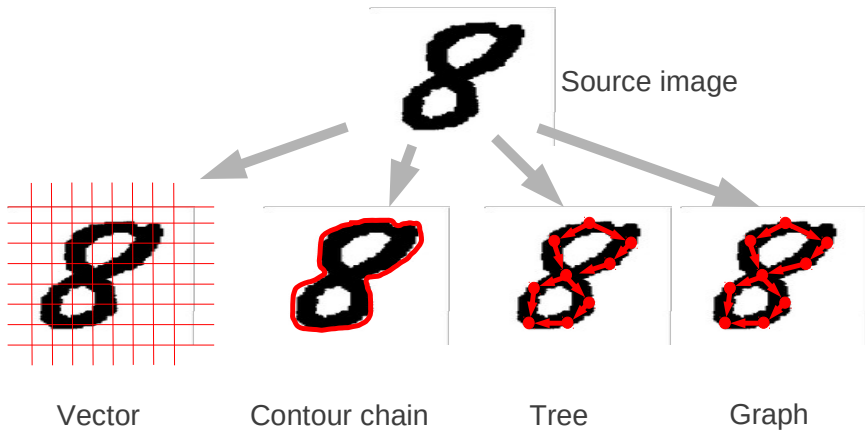
Typology of data

Example

Selection

Preprocessing

References



Example of feature extraction II

Data
collection

Feature
Extraction

Typology of data

Example

Selection

Preprocessing

References

- The structure of the aforementioned features can be a vector, a string, a tree, or a graph as needed.
- This **choice** will determine the **type of algorithms** we will use in classification or regression.
- Examples of representation and usage:
 - Vector** (numeric) general classifiers and regressors, L1/L2 distances, or neural networks.
 - Strings** classifiers and regressors based on neighborhood, string editing distance.
 - Trees** classifiers and regressors based on neighborhood, tree editing distance, or neural networks.
 - Graphs** classifiers and regressors based on neighborhood, approximate graph editing distance, now also with neural networks.

Considerations

Data
collection

Feature
Extraction

Typology of data

Example

Selection

Preprocessing

References

- The most common **data type** in machine learning problems is the **structured vector type**.
- It is now referred to as **tabular data** in contrast to problems with **unstructured data**, which have seen significant growth due to neural networks.
- **In this course**, we will focus on this **type of data**, the **tabular** one.

Data
collection

Feature
Extraction

Selection

Preprocessing

References

2. Selection

Feature Selection

Data
collection

Feature
Extraction

Selection

Preprocessing

References

- It consists of **selecting** the most suitable **type** of **features** or attributes to **describe** the **objects**, **samples**, or **processes** that we want to characterize.
- The features that have a decisive impact on solving the problem must be identified.
- There are feature selection algorithms that, based on variance, correlations, or specific predictive models, determine in advance which features are most relevant.

Algorithms for feature selection

Data
collection

Feature
Extraction

Selection

Preprocessing

References

Following the feature selection methods ([Jović et al., 2015](#)), we have:

- **Filter:** They are based on statistical measures such as correlation, chi-square test, and ANOVA (F-value).
- **Wrapper:** They select features by evaluating combinations of them using a predictive model. Examples include Recursive Feature Elimination (RFE), Backward Feature Elimination (BFE), or Forward Feature Selection (FFS).
- **Embedded:** They select features by learning their importance during the model training, for example, LASSO regression, Ridge regression or Random Forest.

In Python... I

Data
collection

Feature
Extraction

Selection

Preprocessing

References

Check out **Feature Selection** in scikit-learn.

```
from sklearn import feature_selection
```

Filter Methods

- Remove features with low variance:
`feature_selection.VarianceThreshold()`
- Select top K features based on a classification or regression score:
`feature_selection.SelectKBest()`

In Python... II

Data
collection

Feature
Extraction

Selection

Preprocessing

References

Wrapper Methods

- Recursively eliminate features based on an estimator:
`feature_selection.RFE()`
- Sequentially add features:
`feature_selection.SequentialFeatureSelector()`

Embedded Methods

- Select features based on predictive model importance:
`feature_selection.SelectFromModel()`

Data
collection

Feature
Extraction

Selection

Preprocessing

References

3. Preprocessing

Feature Preprocessing

Data
collection

Feature
Extraction

Selection

Preprocessing

References

- The aspects discussed below are related to **tabular data** (structured) as for unstructured data types like images, text, audio, or video, there are specific techniques to learn with neural networks.
- Once the most important features are selected, it is common to preprocess the data with some techniques to ensure their correct learning before applying a learning algorithm:
 - **Numeric values:** Use scaling or normalization techniques.
 - **Categorical values:** Apply binarization techniques (dummies or one-hot encoder), indicate that they are distinct or ordinal categories.
 - **Missing** or unknown values: Remove the variable or sample or use value imputation methods.

In Python... I

Data
collection

Feature
Extraction

Selection

Preprocessing

References

Consult **Preprocessing data** from scikit-learn or Pandas.

```
from sklearn import preprocessing
import pandas as pd
```

Numeric

- Remove the mean and scale to standard deviation:
preprocessing.StandardScaler()
- Scale to a specified range, usually between 0 and 1:
preprocessing.MinMaxScaler()

In Python... II

Data
collection

Feature
Extraction

Selection

Preprocessing

References

Categorical

- **Binarization:** Encode categories as a binary vector, using one for the specific category index:
`pd.get_dummies(dataframe)`
- **Distinct categories:** To indicate that variables are categorical, we can use:
`dataframe[column].astype('category')`
- **Ordinal categories:** Create an ordered list with the categories and apply it to the dataframe column:
`var_cat=pd.CategoricalDtype(list_ordered,ordered=True)` and
`dataframe[column].astype(var_cat)`

In Python... III

Data
collection

Feature
Extraction

Selection

Preprocessing

References

Missing Values

- Assign mean, median, or mode to missing values of the variable:
`impute.SimpleImputer()`
- Use the kNN algorithm on known values to assign missing values:
`impute.KNNImputer()`
- Multivariate imputation that estimates each feature from all the others with a rotation technique:
`impute.IterativeImputer()`