Data Visualization

Encodings and Dash

CentraleDigitalLab@Nice

Solutions - Practical Part

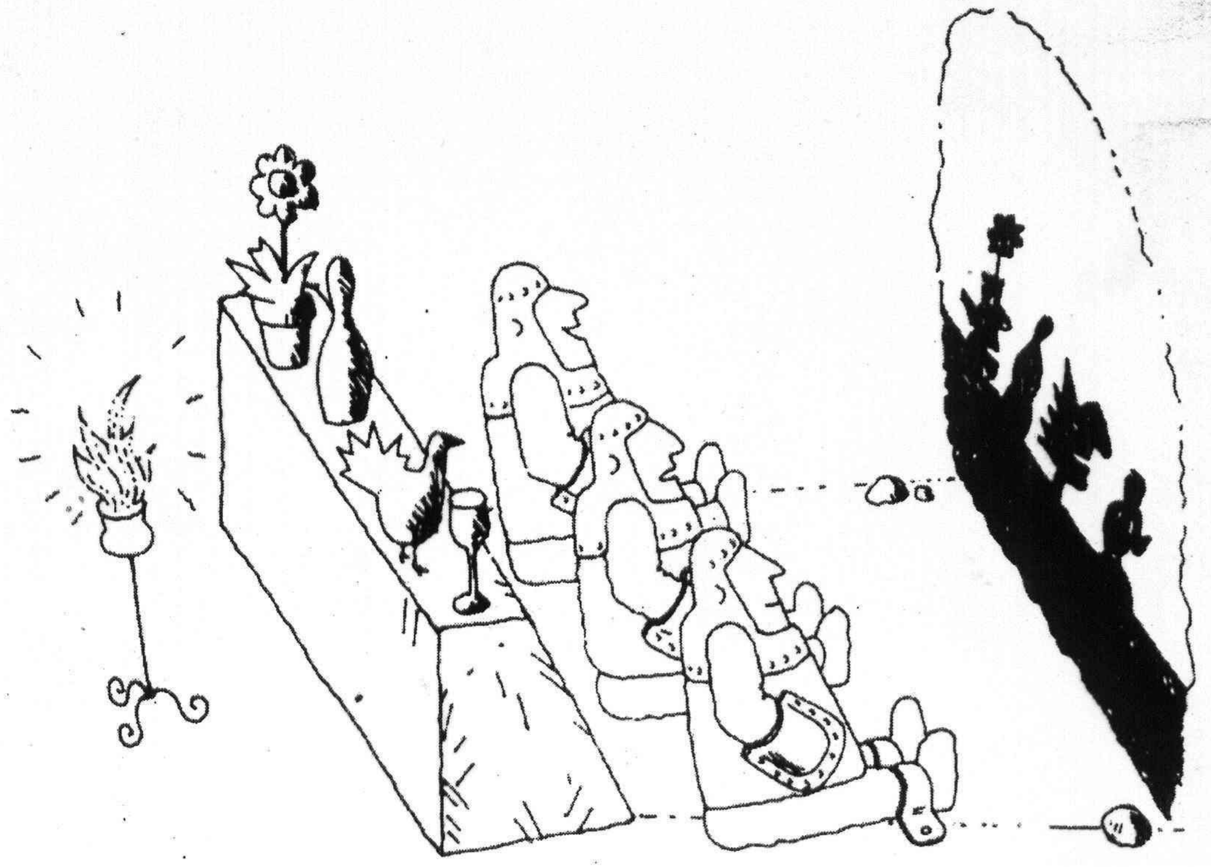
02__practical_exercises_solutions.ipynb

We want to bring out the **important features** for a given task

Data science is like

Plato's cave allegory

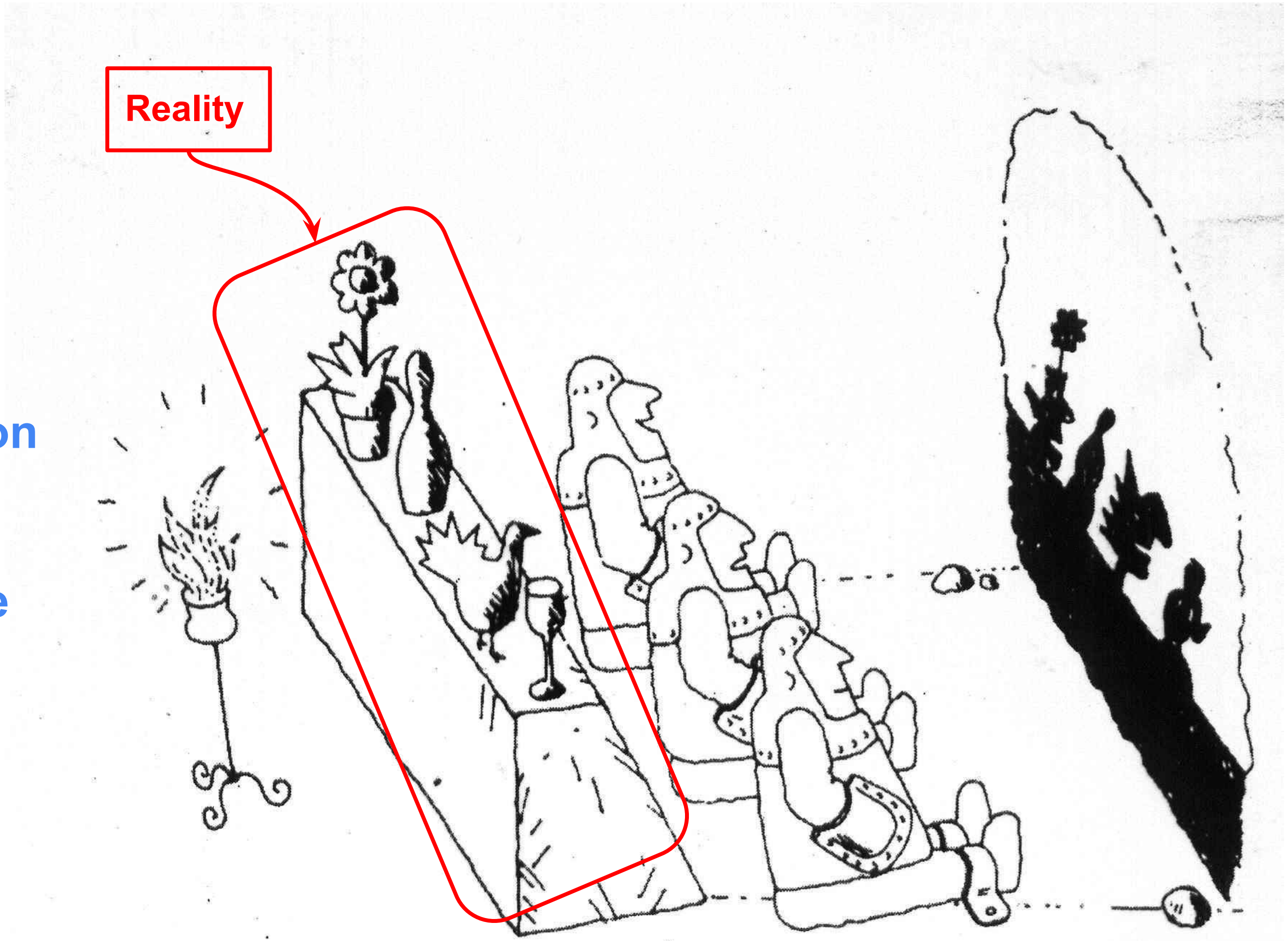
The data is a **projection** that shows us only **certain aspects of the phenomenon** we are studying.



Data science is like

Plato's cave allegory

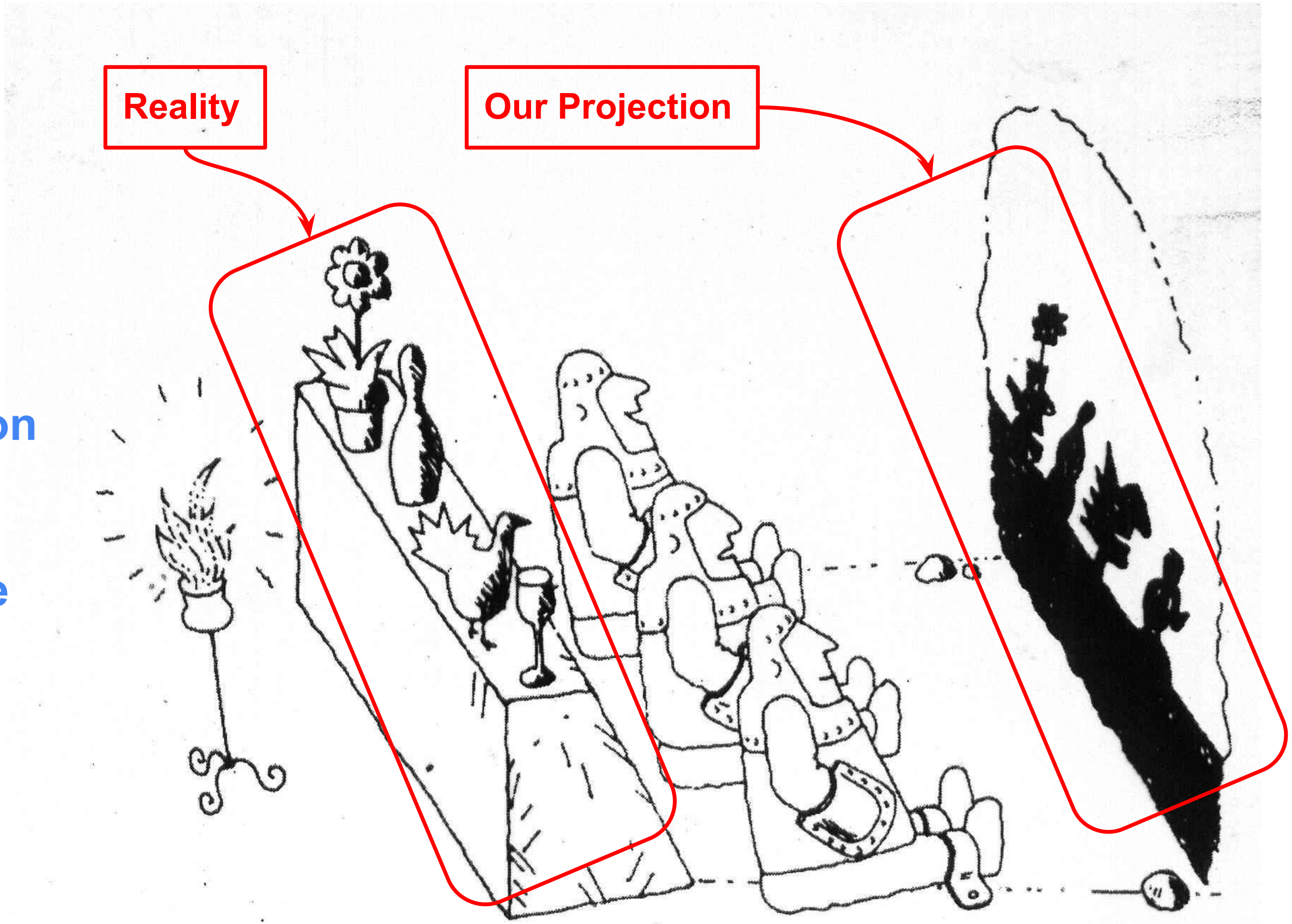
The data is a **projection** that shows us only **certain aspects of the phenomenon** we are studying.



Data science is like

Plato's cave allegory

The data is a **projection** that shows us only **certain aspects of the phenomenon** we are studying.



Filtering, Projecting and Curating

To decide on the manipulation processes, we have to **understand our data** as a whole. This includes:

Filtering, Projecting and Curating

To decide on the manipulation processes, we have to **understand our data** as a whole. This includes:

- All the analytics tools we've seen in **data visualization**.
- More complex techniques for data analysis that allow **multiple variables to be related**.
- Tradeoff: **filtering/curating** our dataset VS **limiting our dataset** too much.

Some Examples

Problematic situation	**Data**	**Curation decisions**
Predict programmers salaries in Argentina in 2020		

Some Examples

Problematic situation	Data	Curation decisions
Predict programmers salaries in Argentina in 2020	Voluntary survey with age, gender, years of experience and salary columns	

Some Examples

Problematic situation	**Data**	**Curation decisions**
Predict programmers salaries in Argentina in 2020	Voluntary survey with age, gender, years of experience and salary columns	➔ **Delete ages** less than 18 and greater than 99

Some Examples

Problematic situation	**Data**	**Curation decisions**
Predict programmers salaries in Argentina in 2020	Voluntary survey with age, gender, years of experience and salary columns	➔ Delete ages less than 18 and greater than 99 ➔ **Eliminate salaries** greater than 1 million pesos

Some Examples

Problematic situation	Data	Curation decisions
Predict programmers salaries in Argentina in 2020	Voluntary survey with age, gender, years of experience and salary columns	➔ Delete ages less than 18 and greater than 99 ➔ Eliminate salaries greater than 1 million pesos ➔ **Standardize** the years of experience so that the mean is 0.

Some Examples

Problematic situation	**Data**	**Curation decisions**
Predict programmers salaries in Argentina in 2020	Voluntary survey with age, gender, years of experience and salary columns	➔ Delete ages less than 18 and greater than 99 ➔ Eliminate salaries greater than 1 million pesos ➔ Standardize the years of experience so that the mean is 0. ➔ **Rescale** the ages in a range from 0 to 1, such that 18 years or less corresponds to 0 and 70 years or more corresponds to 1.

Some Examples

Problematic situation	**Data**	**Curation decisions**
Predict programmers salaries in Argentina in 2020	Voluntary survey with age, gender, years of experience and salary columns	➔ Delete ages less than 18 and greater than 99 ➔ Eliminate salaries greater than 1 million pesos ➔ Standardize the years of experience so that the mean is 0. ➔ Rescale the ages in a range from 1 to 0, such that 18 years or less corresponds to 0 and 70 years or more corresponds to 1. ➔ **Delete** the gender **column**.

Some Examples

Problematic situation	Data	Curation decisions
Predict programmers salaries in Argentina in 2020	Voluntary survey with age, gender, years of experience and salary columns	➔ Delete ages less than 18 and greater than 99 ➔ Eliminate salaries greater than 1 million pesos ➔ Standardize the years of experience so that the mean is 0. ➔ Rescale the ages in a range from 1 to 0, such that 18 years or less corresponds to 0 and 70 years or more corresponds to 1. ➔ Delete the gender column.
Predict the price of a property		

Some Examples

Problematic situation	Data	Curation decisions
Predict programmers salaries in Argentina in 2020	Voluntary survey with age, gender, years of experience and salary columns	➔ Delete ages less than 18 and greater than 99 ➔ Eliminate salaries greater than 1 million pesos ➔ Standardize the years of experience so that the mean is 0. ➔ Rescale the ages in a range from 1 to 0, such that 18 years or less corresponds to 0 and 70 years or more corresponds to 1. ➔ Delete the gender column.
Predict the price of a property	Government database with records of real house transactions. It has price, date and location.	

Some Examples

Problematic situation	**Data**	**Curation decisions**
Predict programmers salaries in Argentina in 2020	Voluntary survey with age, gender, years of experience and salary columns	➔ Delete ages less than 18 and greater than 99 ➔ Eliminate salaries greater than 1 million pesos ➔ Standardize the years of experience so that the mean is 0. ➔ Rescale the ages in a range from 1 to 0, such that 18 years or less corresponds to 0 and 70 years or more corresponds to 1. ➔ Delete the gender column.
Predict the price of a property	Government database with records of real house transactions. It has price, date and location.	➔ **Delete day and month** of the transaction.

Some Examples

Problematic situation	Data	Curation decisions
Predict programmers salaries in Argentina in 2020	Voluntary survey with age, gender, years of experience and salary columns	➔ Delete ages less than 18 and greater than 99 ➔ Eliminate salaries greater than 1 million pesos ➔ Standardize the years of experience so that the mean is 0. ➔ Rescale the ages in a range from 1 to 0, such that 18 years or less corresponds to 0 and 70 years or more corresponds to 1. ➔ Delete the gender column.
Predict the price of a property	Government database with records of real house transactions. It has price, date and location.	➔ Delete day and month of the transaction. ➔ ***Scrape* buying/selling sites** to extract additional information about each property.

Some Examples

Problematic situation	**Data**	**Curation decisions**
Predict programmers salaries in Argentina in 2020	Voluntary survey with age, gender, years of experience and salary columns	➔ Delete ages less than 18 and greater than 99 ➔ Eliminate salaries greater than 1 million pesos ➔ Standardize the years of experience so that the mean is 0. ➔ Rescale the ages in a range from 1 to 0, such that 18 years or less corresponds to 0 and 70 years or more corresponds to 1. ➔ Delete the gender column.
Predict the price of a property	Government database with records of real house transactions. It has price, date and location.	➔ Delete day and month of the transaction. ➔ *Scrape* buying/selling sites to extract additional information about each property. ➔ **Impute missing values** using estimates based on similar examples.

The curse of the categories

What information does the **address of a property** give me?

The curse of the categories

What information does the **address of a property** give me?

The address of a property for sale is a **categorical variable that cannot be used without transforming it**.

The curse of the categories

What information does the **address of a property** give me?

The address of a property for sale is a **categorical variable that cannot be used without transforming it**.

Intuitively, we infer the neighborhood of a property based on its address.

The curse of the categories

What information does the **address of a property** give me?

The address of a property for sale is a **categorical variable that cannot be used without transforming it**.

Intuitively, we infer the neighborhood of a property based on its address.

The **categories** give me information because they **group different examples**.

The curse of the categories

What information does the **address of a property** give me?

The address of a property for sale is a **categorical variable that cannot be used without transforming it**.

Intuitively, we infer the neighborhood of a property based on its address.

The **categories** give me information because they **group different examples**.

The **fewer examples** they group together, the **less informative** they are.

The curse of the categories

Possible approaches with categories with fewer instances:

The curse of the categories

Possible approaches with categories with fewer instances:

- **Delete the variable**.

The curse of the categories

Possible approaches with categories with fewer instances:

- **Delete the variable**.
- **Combine it** with another variable.
    - Ex: We only use the zipcode for neighborhoods that have more than one postal code.

The curse of the categories

Possible approaches with categories with fewer instances:

- **Delete the variable**.
- **Combine it** with another variable.
    - Ex: We only use the zipcode for neighborhoods that have more than one postal code.
- **Create new categories**:
    - Group similar categories.
    - Create an “other” category for categories that don't have many examples.

Data Enrichment

Combining different datasets

Data Enrichment

Another common preprocessing strategy is **scrapping new information from other sources** and **merging it** with your current dataset. This helps to:

Data Enrichment

Another common preprocessing strategy is **scrapping new information from other sources** and **merging it** with your current dataset. This helps to:

- Add new random variables that might help to improve get better performance in your task.

Data Enrichment

Another common preprocessing strategy is **scrapping new information from other sources** and **merging it** with your current dataset. This helps to:

- Add new random variables that might help to improve get better performance in your task.
- Curate missing values.

Data Enrichment

Another common preprocessing strategy is **scrapping new information from other sources** and **merging it** with your current dataset. This helps to:

- Add new random variables that might help to improve get better performance in your task.
- Curate missing values.
- The data structure is not the same as the type of database.

Demo with notebook

06__combining_datasets.ipynb

Encodings

Machine learning algorithms require exclusively numerical data

We need to transform our categorical variables to some numerical format

One-hot encoding

Id	neighbourhood
1	Saint Vincent
2	Hill of the Roses
3	Maipú
4	Saint Vincent
5	Ituzaingó

Id	neighbourhood =Saint Vincent	neighbourhood =Hill of the Roses	neighbourhood =Maipú	neighbourhood =Ituzaingó
1				
2				
3				
4				
5				

One-hot encoding

Id	neighbourhood
1	Saint Vincent
2	Hill of the Roses
3	Maipú
4	Saint Vincent
5	Ituzaingó

Id	neighbourhood =Saint Vincent	neighbourhood =Hill of the Roses	neighbourhood =Maipú	neighbourhood =Ituzaingó
1	1	0	0	0
2				
3				
4				
5				

One-hot encoding

Id	neighbourhood
1	Saint Vincent
2	Hill of the Roses
3	Maipú
4	Saint Vincent
5	Ituzaingó

Id	neighbourhood =Saint Vincent	neighbourhood =Hill of the Roses	neighbourhood =Maipú	neighbourhood =Ituzaingó
1	1	0	0	0
2	0	1	0	0
3	0	0	1	0
4	1	0	0	0
5	0	0	0	1

The curse of dimensionality

By encoding the data in this way, we generate **high-dimensional sparse vectors**

The curse of dimensionality

By encoding the data in this way, we generate **high-dimensional sparse vectors**

- Takes up a **lot of memory space**

The curse of dimensionality

By encoding the data in this way, we generate **high-dimensional sparse vectors**

- Takes up a **lot of memory space**
- The resulting **vectors are orthogonal**.

The curse of dimensionality

By encoding the data in this way, we generate **high-dimensional sparse vectors**

- Takes up a **lot of memory space**
- The resulting **vectors are orthogonal**.
  - All vectors are the **same distance from each other** (if they have norm 1)

The curse of dimensionality

By encoding the data in this way, we generate **high-dimensional sparse vectors**

- Takes up a **lot of memory space**
- The resulting **vectors are orthogonal**.
    - All vectors are the **same distance from each other** (if they have norm 1)

Text encoding in bags of words

Id	comment
1	No traffic no
2	Near the airport
3	airport traffic
4	Near the beach

Id	no	traffic	near	the	airport	beach
1	2	1	0	0	0	0
2	0	0	1	1	1	0
3	0	1	0	0	1	0
4	0	0	1	1	0	1

Free text analysis

Suburb	closest_airbnb_neighborhood_overview
Melton South	Close to the CBD, 30-60 minutes from top Victorian beaches and suitable for day trips out to the beautiful Victoria countryside...
Oakleigh	Close to Chadstone Shopping centre, Oakleigh Centro, Walking distance approx 500m to Oakleigh and Huntingdale train station .Bus stops are easily available a couple of streets away...
Balwyn	Filled with gorgeous parks, award winning restaurants and shops and leading Deli's across Melbourne. It's close to the city- 15 minute tram ride into the city or 12 minutes into Richmond...

Scaling

- **Standardization**: Common requirement for many ML estimators in scikit-learn; they might behave badly if the individual features do not look like standard normally distributed data.

```
z = (x - u) / s
```

Scaling

- **Standardization**: Common requirement for many ML estimators in scikit-learn; they might behave badly if the individual features do not look like standard normally distributed data.

```
z = (x - u) / s
```

- **MinMaxScaler**: Scales features between a given minimum and maximum value, often between zero and one,

```
x_s = (x - min) / (max- min)
x_s (R - L) + L
```

Scaling

- **Standardization**: Common requirement for many ML estimators in scikit-learn; they might behave badly if the individual features do not look like standard normally distributed data.

```
z = (x - u) / s
```

- **MinMaxScaler**: Scales features between a given minimum and maximum value, often between zero and one,

```
x_s = (x - min) / (max- min)
x_s (R - L) + L
```

- **MaxAbsScaler**: Special case of MinMaxScaler but for `[-1, 1]`.

Ordinal Encoding

Given an ordinal categorical r.v X with categories $C_1 < C_2 < ... < C_n$ we enumerate them with integers $0 < ... < n-1$. This encoding preserves the order.

Ordinal Encoding

Given an ordinal categorical r.v X with categories $C_1 < C_2 < \ldots < C_n$ we enumerate them with integers $0 < \ldots < n-1$. This encoding preserves the order.

Enumeration

Primary	0
Secondary	1
University	2
Doctorate	3
Postdoc	4

DataFrame to Encode

Index	Studies Level
0	Primary
1	Postdoc
2	University
3	Doctorate
4	Secondary
5	Primary

Ordinal Encoding

Given an ordinal categorical r.v X with categories $C_1 < C_2 < ... < C_n$ we enumerate them with integers $0 < ... < n-1$. This encoding preserves the order.

Enumeration

Primary	0
Secondary	1
University	2
Doctorate	3
Postdoc	4

DataFrame to Encode

Index	Studies Level
0	Primary
1	Postdoc
2	University
3	Doctorate
4	Secondary
5	Primary

Encoded dataframe

Index	Studies Level
0	Primary
1	Postdoc
2	University
3	Doctorate
4	Secondary
5	Primary

Ordinal Encoding

Given an ordinal categorical r.v X with categories $C_1 < C_2 < ... < C_n$ we enumerate them with integers $0 < ... < n-1$. This encoding preserves the order.

Enumeration

Primary	**0**
Secondary	1
University	2
Doctorate	3
Postdoc	4

DataFrame to Encode

Index	Studies Level
0	**Primary**
1	Postdoc
2	University
3	Doctorate
4	Secondary
5	Primary

Encoded dataframe

Index	Studies Level
0	**0**
1	Postdoc
2	University
3	Doctorate
4	Secondary
5	Primary

Ordinal Encoding

Given an ordinal categorical r.v X with categories $C_1 < C_2 < \ldots < C_n$ we enumerate them with integers $0 < \ldots < n-1$. This encoding preserves the order.

Enumeration

Primary	0
Secondary	1
University	2
Doctorate	3
Postdoc	**4**

DataFrame to Encode

Index	**Studies Level**
0	Primary
1	**Postdoc**
2	University
3	Doctorate
4	Secondary
5	Primary

Encoded dataframe

Index	**Studies Level**
0	0
1	**4**
2	University
3	Doctorate
4	Secondary
5	Primary

Ordinal Encoding

Given an ordinal categorical r.v X with categories $C_1 < C_2 < ... < C_n$ we enumerate them with integers $0 < ... < n-1$. This encoding preserves the order.

Enumeration

Primary	0
Secondary	1
University	**2**
Doctorate	3
Postdoc	4

DataFrame to Encode

Index	**Studies Level**
0	Primary
1	Postdoc
2	**University**
3	Doctorate
4	Secondary
5	Primary

Encoded dataframe

Index	**Studies Level**
0	0
1	4
2	**2**
3	Doctorate
4	Secondary
5	Primary

Ordinal Encoding

Given an ordinal categorical r.v X with categories $C_1 < C_2 < \ldots < C_n$ we enumerate them with integers $0 < \ldots < n-1$. This encoding preserves the order.

Enumeration

Primary	0
Secondary	1
University	2
Doctorate	**3**
Postdoc	4

DataFrame to Encode

Index	Studies Level
0	Primary
1	Postdoc
2	University
3	**Doctorate**
4	Secondary
5	Primary

Encoded dataframe

Index	Studies Level
0	0
1	4
2	2
3	**3**
4	Secondary
5	Primary

Ordinal Encoding

Given an ordinal categorical r.v X with categories $C_1 < C_2 < ... < C_n$ we enumerate them with integers $0 < ... < n-1$. This encoding preserves the order.

Enumeration

Primary	0
Secondary	**1**
University	2
Doctorate	3
Postdoc	4

DataFrame to Encode

Index	**Studies Level**
0	Primary
1	Postdoc
2	University
3	Doctorate
4	**Secondary**
5	Primary

Encoded dataframe

Index	**Studies Level**
0	0
1	4
2	2
3	3
4	**1**
5	Primary

Ordinal Encoding

Given an ordinal categorical r.v X with categories $C_1 < C_2 < ... < C_n$ we enumerate them with integers $0 < ... < n-1$. This encoding preserves the order.

Enumeration

Primary	**0**
Secondary	1
University	2
Doctorate	3
Postdoc	4

DataFrame to Encode

Index	**Studies Level**
0	Primary
1	Postdoc
2	University
3	Doctorate
4	Secondary
5	**Primary**

Encoded dataframe

Index	**Studies Level**
0	0
1	4
2	2
3	3
4	1
5	**0**

Discretizers

We can take a **numerical variable** and **segment** it **equally** in categories.

For example, if we are dealing with the salary of developers, we can discretize it in three groups, in such a way these groups have more or less the same number of instances.

Polynomial Features

Often it's useful to add complexity to a model by considering nonlinear features of the input data. One possibility is to use polynomial features.

For example, if we have the features of `x1` and `x2`, we can create six features from them by **combining through multiplications** obtaining:

```
(1, X1, X2, X1.X1, X1.X2, X2.X2)
```

Demo with notebook

07__encodings.ipynb

What is Dash?

Dash is a Python framework for creating web-based analytics applications that integrates Plotly plots seamlessly.

We can create dashboards in order to report results to clients.

Demo with notebook

08__dash.ipynb