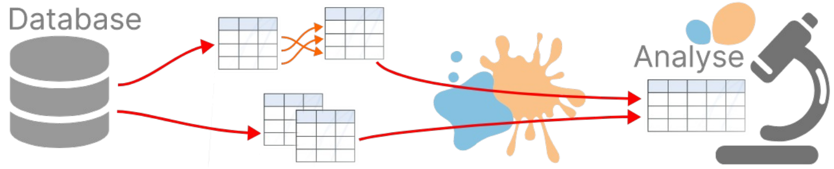
skrub: prepping tables for machine learning

*Jovan Stojanovic and Lilian Boulard**

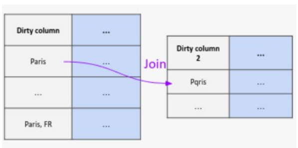
Inria Saclay, Soda team, France

**Presenting authors* | *E-mail:* `{first name}.{last name}@inria.fr`



Assembling

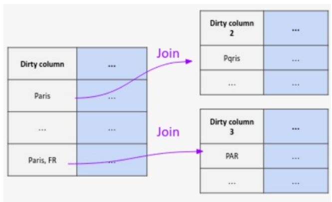
`Joining two tables`



- Fuzzy joining of dirty tables.
- Works on numerical, datetime, string or mixed types.
- Works on multiple join keys.

```
> fuzzy_join(df1, df2, left_on, right_on)
```

`Joining a pool of tables`

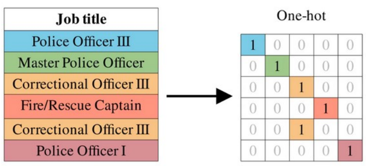


- Multiple tables transformer.
- Scikit-learn compatible.

```
> joiner = Joiner([(df2, “col2”),
(df3, ”col3”)...], main_key=“col1”)
> joiner.fit_transform(df1)
```

Encoding

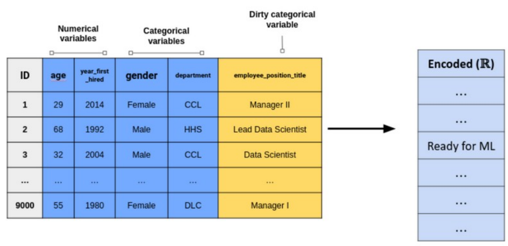
Encoding: transform **dataframes** into **numerical arrays** ready for machine learning.
Classical methods (**One-hot encoding**…) need **clean data**, as they encode each unique value **independently**.
They fail to capture **morphological similarities** (variations or typos), which often have meaning.
- skrub encoders propose **new, adapted** methods. **[1]**



`TableVectorizer`

- One line of code - automatic encoding.
- Encoders chosen based on **heuristics**, but can also be **customized**.
- Replacement for the **ColumnTransformer**.

```
> X = TableVectorizer().fit_transform(df)
```



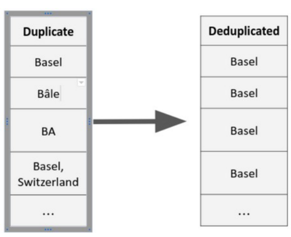
`GapEncoder`

Describes each sample as a linear combination of **latent categories** (topics). **Interpretable** output. **[2]**

`MinHashEncoder`

Very fast, stateless (for parallel encoding). Based on **hashing functions** (min-hash of sub-strings). **[2]**

Deduplicating



Based on **hierarchical clustering**.

```
> deduplicate()
```

References

[1] Patricio Cerda, Gaël Varoquaux, Balázs Kégl. Similarity encoding for learning with dirty categorical variables. Machine Learning, Springer Verlag, 2018, 10.1007/s10994-018-5724-2. hal-01806175

[2] Patricio Cerda, Gaël Varoquaux. Encoding high-cardinality string categorical variables. 2019. hal-02171256v4