

# SKFORECAST: TIME SERIES FORECASTING WITH MACHINE LEARNING

---

Joaquín Amat Rodrigo  
Javier Escobar Ortiz



# PEOPLE WHO KNOW SKFORECAST



## BEHIND THE SCENES



**Joaquín Amat Rodrigo**

Senior Data Scientist @Veeva Systems

LinkedIn 

[j.amatrodrigo@gmail.com](mailto:j.amatrodrigo@gmail.com)



**Javier Escobar Ortiz**

Senior Data Scientist @IKEA

 LinkedIn

[javier.escobar.ortiz@gmail.com](mailto:javier.escobar.ortiz@gmail.com)



# TABLE OF CONTENTS

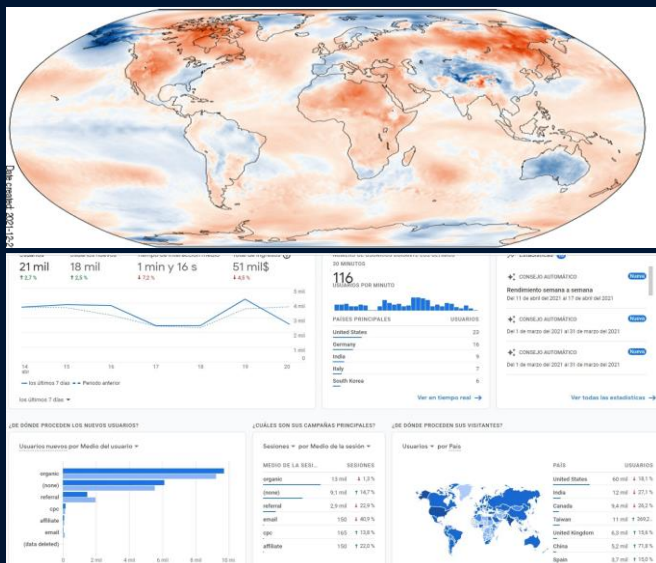
- Time series and forecasting
- Forecasting strategies
  - Recursive multi-step forecasting
  - Direct multi-step forecasting
- ¿Why skforecast?
- Model Evaluation and Tuning
- Feature selection
- Global Forecasters (multiple series)
- Probabilistic forecasting
- Model Explainability
- Additional material
- State of the art
- Get Involved - Share your feedback

# WHAT IS A TIME SERIES?



# TIME SERIES

A time series is a succession of data arranged chronologically and spaced at equal or irregular intervals.



# FORECASTING

The forecasting process consists of predicting the future value of a time series, either by **modeling** the series solely based on its past behavior (**autoregressive**) or by incorporating other **external variables**.



Historical data is used to obtain a mathematical representation capable of predicting future values to create a forecasting model. This idea is based on the premise that **the future behavior of a phenomenon can be explained by its past behavior**.



**This rarely happens in reality, or at least not in its entirety.**

---

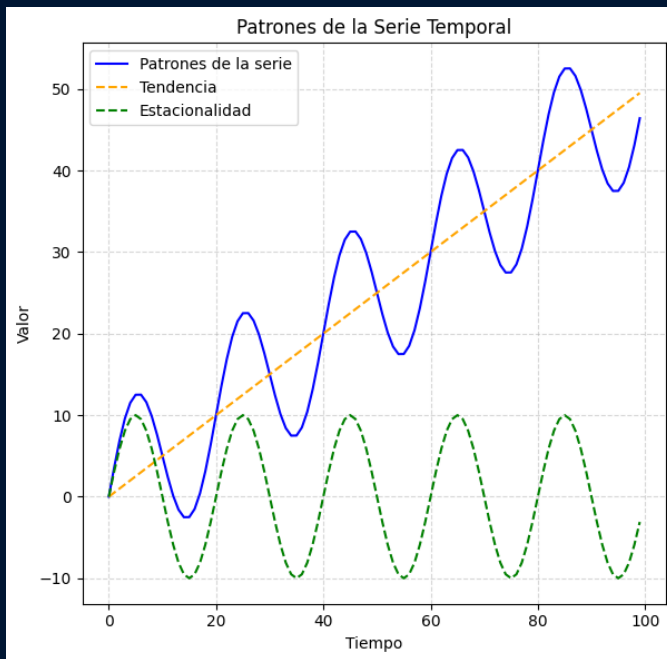
## COMPONENTES DE UN FORECAST

**Forecast** = **patterns of the series** + **unexplained variance**



# COMPONENTES DE UN FORECAST

Forecast = patterns of the series + unexplained variance

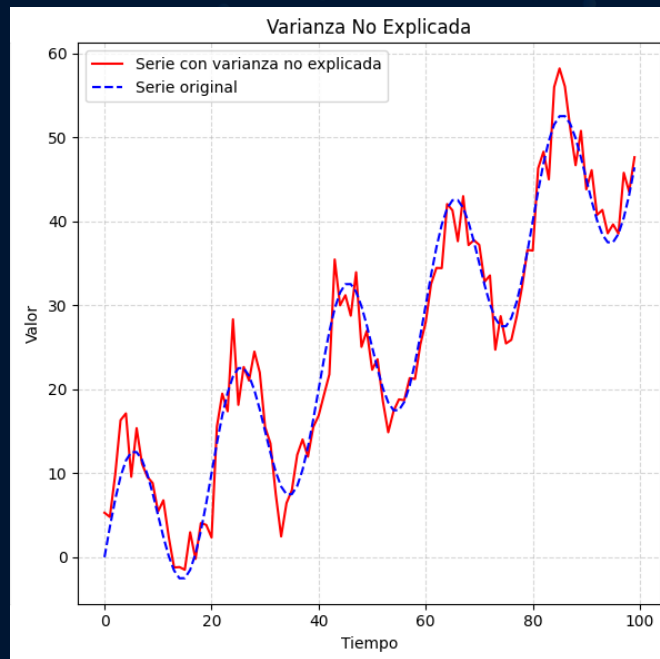


The first term in the equation refers to anything that **repeats itself over time** (trend, seasonality, cyclical factors, etc.). This is known as the **autoregressive component**.

# COMPONENTES DE UN FORECAST

**Forecast** = patterns of the series + **unexplained variance**

The second term represents everything that **influences** the series but is **not captured** (explained) **in its past** values.



# COMPONENTES DE UN FORECAST

**Forecast** = **patterns of the series** + **unexplained variance**

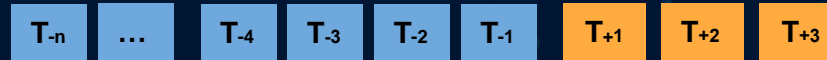
The greater the importance of the **first term** relative to the **second term**, the greater the likelihood of **success** in creating autoregressive predictive models.



As the **second term** becomes more important, it becomes necessary to include **additional variables** (if any) in the model **to help explain** the observed behavior.

A **good study** of the phenomenon to be modeled and the ability to recognize the extent to which its behavior can be explained by its past **can save a lot of unnecessary effort**.

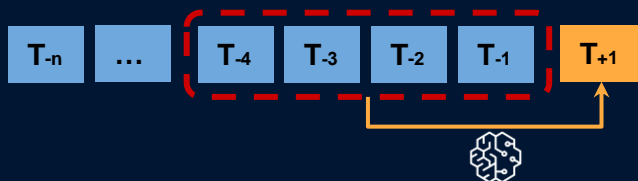
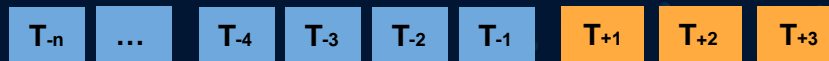
# FORECASTING STRATEGIES

# RECURSIVE MULTI-STEP FORECASTING







 Observed value  
 Predicted value

# RECURSIVE MULTI-STEP FORECASTING

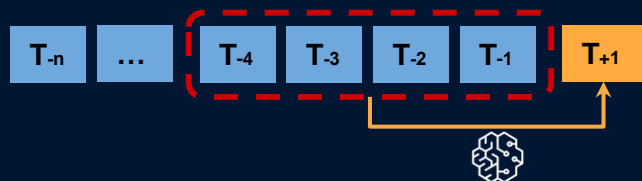
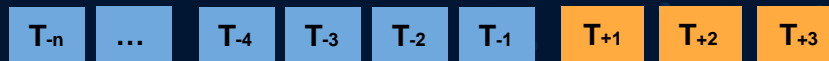


Step 1 Prediction

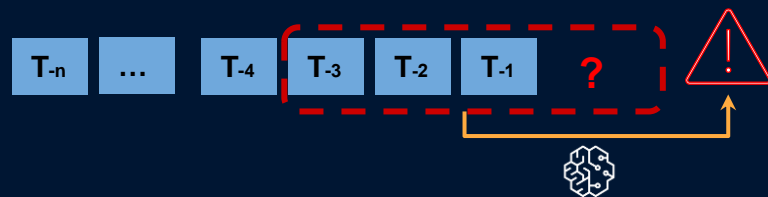
-  Observed value
-  Predicted value
-  Predictors (lags)
-  Model trained to predict step +1







# RECURSIVE MULTI-STEP FORECASTING



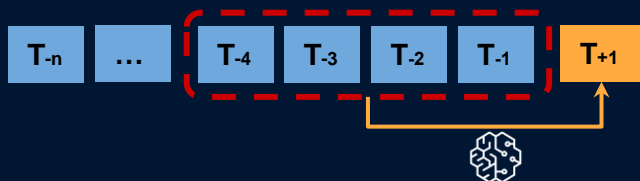
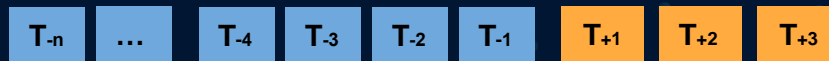
Step 1 Prediction



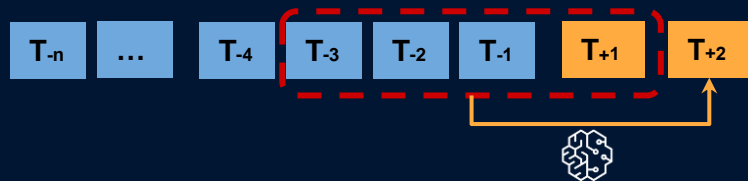
Step 2 Prediction

-  Observed value
-  Predicted value
-  Predictors (lags)
-  Model trained to predict step +1





# RECURSIVE MULTI-STEP FORECASTING



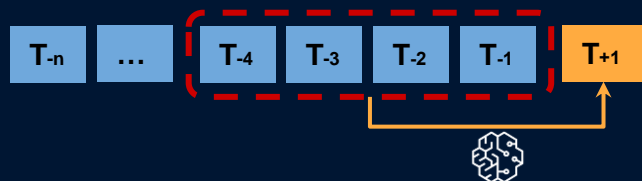
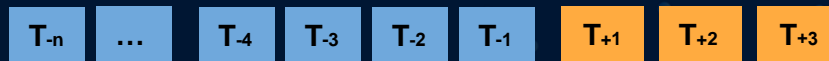
Step 1 Prediction



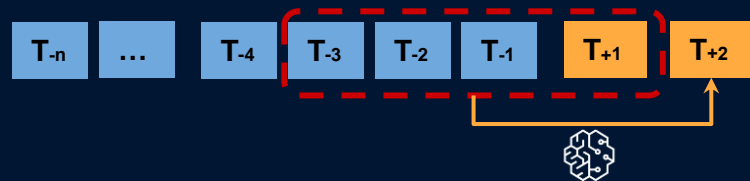
Step 2 Prediction

-  Observed value
-  Predicted value
-  Predictors (lags)
-  Model trained to predict step +1

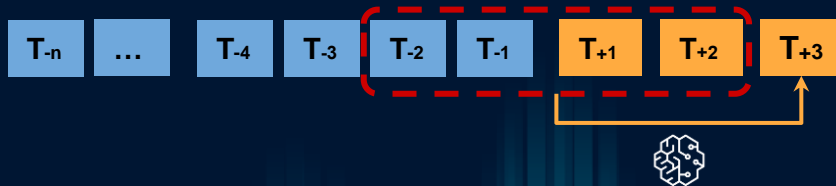
# RECURSIVE MULTI-STEP FORECASTING







Step 1 Prediction



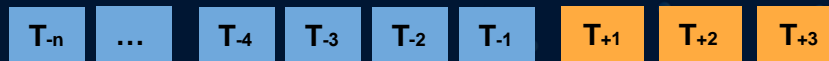
Step 2 Prediction





Step 3 Prediction

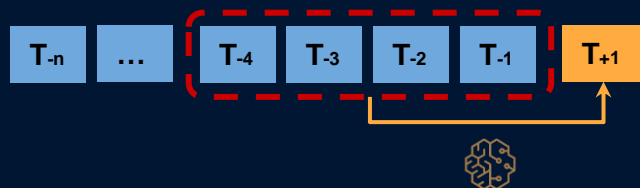
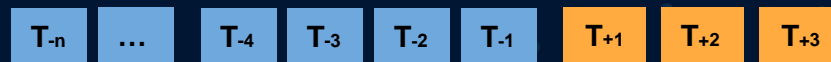
-  Observed value
-  Predicted value
-  Predictors (lags)
-  Model trained to predict step +1

# DIRECT MULTI-STEP FORECASTING



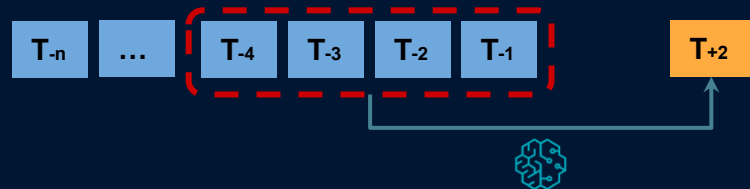
 Observed value  
 Predicted value

# DIRECT MULTI-STEP FORECASTING



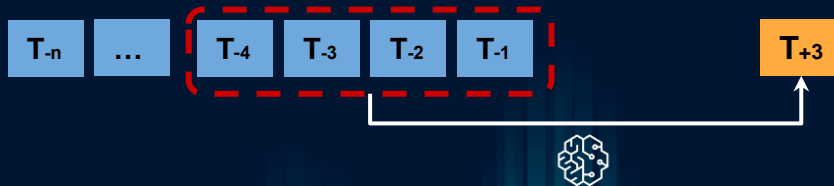
## Step 1 Prediction

 Model for step 1



## Step 2 Prediction

 Model for step 2



## Step 3 Prediction

 Model for step 3

 Observed value  
 Predicted value  
 Predictors (lags)

# WHY SKFORECAST?



# WHY SKFORECAST?

## Easily build forecasting models

Train models using the tools you already know: **scikit-learn**, **LightGBM**, or **XGBoost**.



## Answer key questions quickly

What impact would a decision have had on your business?



## Save time and effort

*¡Wow, the MVP that started 4.32 years ago has turned out well... Mmm, my DS won't get angry if I tell him that we have to get the models into production in 3 days, will he?*



# STRENGTHS OF MACHINE LEARNING

## Incorporation of Exogenous Variables

The past of a time series (autoregressive component) explains only part of the observed behavior. Exogenous variables (calendar variables, temperature, economic indicators, etc.) can **provide valuable information to the model**.

## Modeling Multiple Series Simultaneously (Global Models)

Time series that share common patterns can be modeled jointly, which allows for obtaining models that generalize better and are easier to maintain in production. **A single model capable of predicting n series vs. n individual models.**

## Handling Missing Values

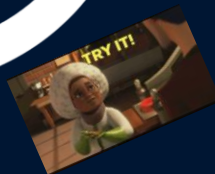
In practice, historical data often contains missing values.

# EVERYTHING ORBITS AROUND SKFORECAST

RECURSIVE/  
DIRECT  
METHODS



EXOGENOUS



What is going on...?



# FORECASTERS

A **Forecaster** object in the **skforecast** library is a **comprehensive container** that provides essential **functionality and methods** for training a forecasting model and generating predictions for future points in time.

Forecaster	Single series	Multiple series	Recursive strategy	Direct strategy	Probabilistic prediction	Time series differentiation	Exogenous features	Window features
ForecasterRecursive	✓		✓		✓	✓	✓	✓
ForecasterDirect	✓			✓	✓		✓	✓
ForecasterRecursiveMultiSeries		✓	✓		✓	✓	✓	✓
ForecasterDirectMultiVariate		✓		✓	✓		✓	✓
ForecasterRNN		✓		✓				
ForecasterSarimax	✓		✓		✓	✓	✓	

Complete table: <https://skforecast.org/latest/#forecasters>

# HOW TO CREATE A FORECASTER?

Let's imagine a time series of a city's electricity demand. The goal is to build a forecasting model that predicts the **demand for the entire city over the next 24 hours**.



## DATA

Electricity Demand

Exogenous variables:

- Temperature
- Calendar

## MODEL



LGBMRegressor

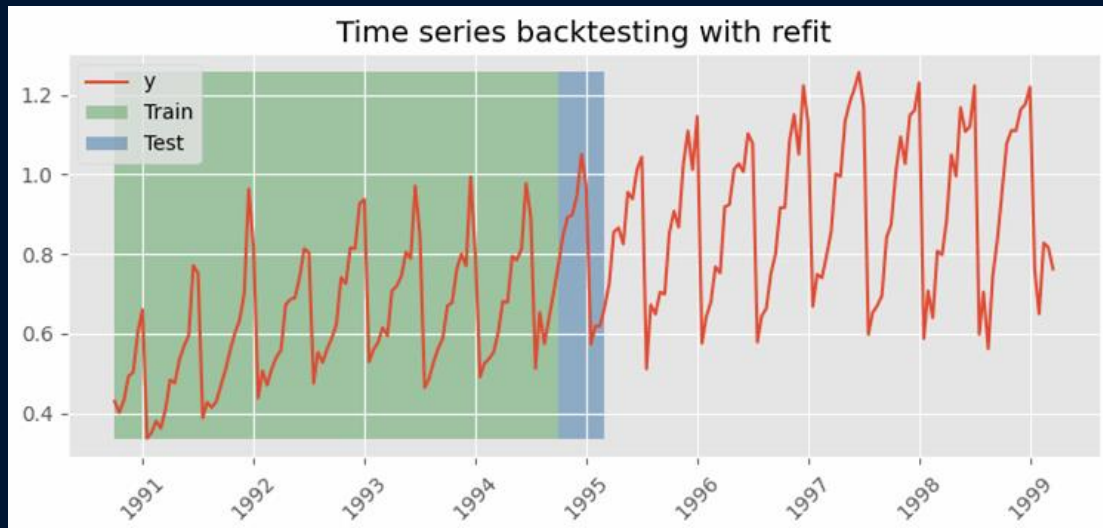
**BUT... HOW DO I  
KNOW THAT I AM  
MODELING WELL?**



# BACKTESTING (MODEL EVALUATION)

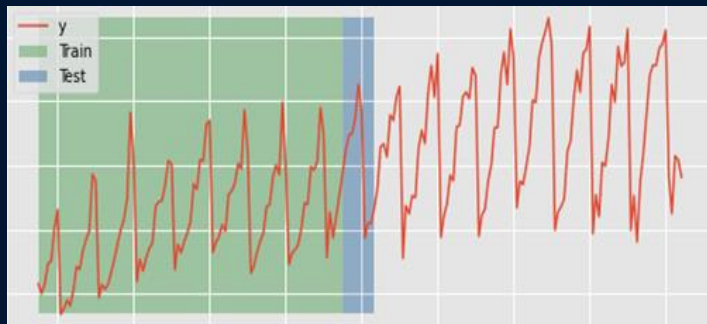
In time series forecasting, **backtesting** is the process of evaluating the performance of a predictive model by applying it retrospectively to **historical data**.

In other words, **How would my model have worked in the past?**

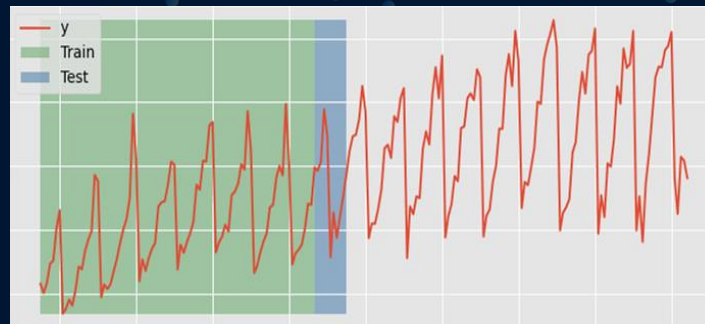


# BACKTESTING STRATEGIES

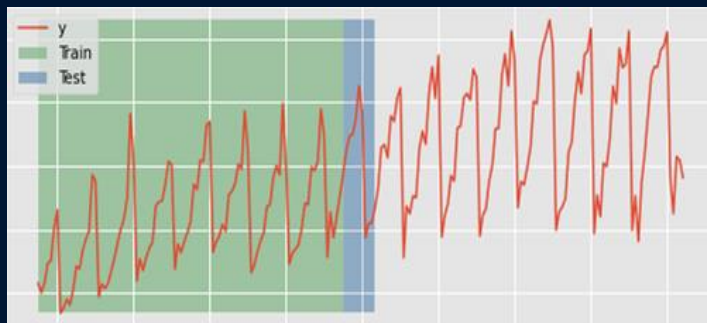
Backtesting without refit



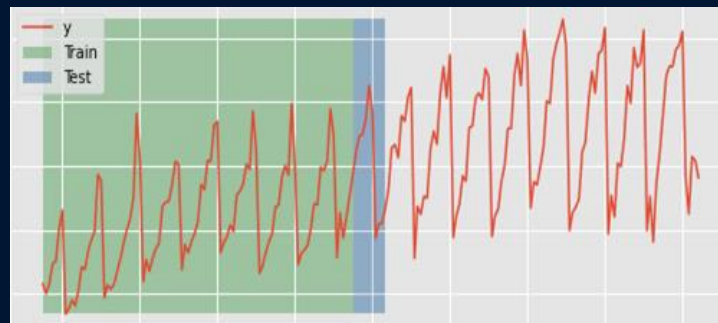
Backtesting with intermittent refit



Backtesting with refit (time series cross-validation, rolling origin)



Backtesting with refit (fixed origin)



**THIS SOUNDS GREAT,  
BUT... HOW DO I  
OPTIMIZE MY MODEL?**

# HYPERPARAMETER TURNING AND LAGS SELECTION

Hyperparameter tuning involves systematically testing **different values or combinations** of hyperparameters (including lags) to **find the optimal configuration** that produces the best results.

Skforecast combines the **common strategies** in machine learning and uses the **backtesting** or **one-step-ahead** technique as the **validation method** for this process.

- **Grid search**
- **Random search**
- **Bayesian search**

# FEATURE SELECTION

**Feature selection** is the process of selecting a subset of relevant features (variables, predictors) for use in model construction. This technique is used for several reasons:

- **Simplify models** to make them easier to interpret.
- To **reduce training time**.
- To **improve generalization** by reducing overfitting.

Skforecast is compatible with the feature selection methods implemented in the **scikit-learn** library. The most common are:

- **Recursive feature elimination (RFE)**
- **Sequential Feature Selection (Forward-SFS, Backward-SFS)**
- **Feature selection based on threshold (SelectFromModel)**

# FEATURE SELECTION

Recursive feature elimination (RFECV)

-----  
Total number of records available: 8712

Total number of records used for feature selection: 4356

Number of features available: 139

Lags (n=48)

Window features (n=3)

Exog (n=88)

Number of features selected: 52

Lags (n=31) : [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 14, 16, 19, 22, 23, 24, 25, 26, 28, 30, 32, 34, 35, 36, 40, 42, 44, 47, 48]

Window features (n=1) : ['roll\_mean\_24']

Exog (n=20) : ['hour\_day\_sin', 'hour\_day\_cos', 'poly\_month\_cos\_week\_of\_year\_sin', 'poly\_week\_of\_year\_sin\_week\_day\_sin', 'poly\_week\_of\_year\_sin\_week\_day\_cos', 'poly\_week\_of\_year\_sin\_hour\_day\_sin', 'poly\_week\_of\_year\_sin\_hour\_day\_cos', 'poly\_week\_of\_year\_sin\_\_sunset\_hour\_cos', 'poly\_week\_of\_year\_cos\_week\_day\_sin', 'poly\_week\_of\_year\_cos\_week\_day\_cos', 'poly\_week\_of\_year\_cos\_hour\_day\_sin', 'poly\_week\_of\_year\_cos\_hour\_day\_cos', 'poly\_week\_day\_sin\_hour\_day\_sin', 'poly\_week\_day\_sin\_hour\_day\_cos', 'poly\_week\_day\_sin\_sunset\_hour\_sin', 'poly\_week\_day\_cos\_hour\_day\_sin', 'poly\_week\_day\_cos\_hour\_day\_cos', 'poly\_hour\_day\_sin\_hour\_day\_cos', 'temp\_roll\_mean\_1\_day', 'temp']



# HANDS ON TIME

# DON'T TELL ME YOU ALSO MANAGE MULTIPLE TIME SERIES

---

# GLOBAL FORECASTING

## Independent multi-series forecasting

A **single model** is trained for all time series, but each time series remains **independent** of the others, meaning that past values of one series are not used as predictors of other series. However, modeling them together is useful because the series may follow the **same intrinsic pattern** regarding their past and future values.

## Dependent multi-series forecasting (Multivariate)

All series are modeled together in a **single model**, considering that each time series **depends** not only on its past values but also on the past values of the **other series**. The forecaster is expected not only to learn the information of each series separately but also to **relate them**.

# COMMON MULTI\_SERIES USE CASES

## Independent multi-series forecasting

Sales of **1000 products** in the **same store** may not be related, but they follow the **same dynamic**, that of the store.

How could we estimate the **energy consumption of hundreds of households** in an entire city?

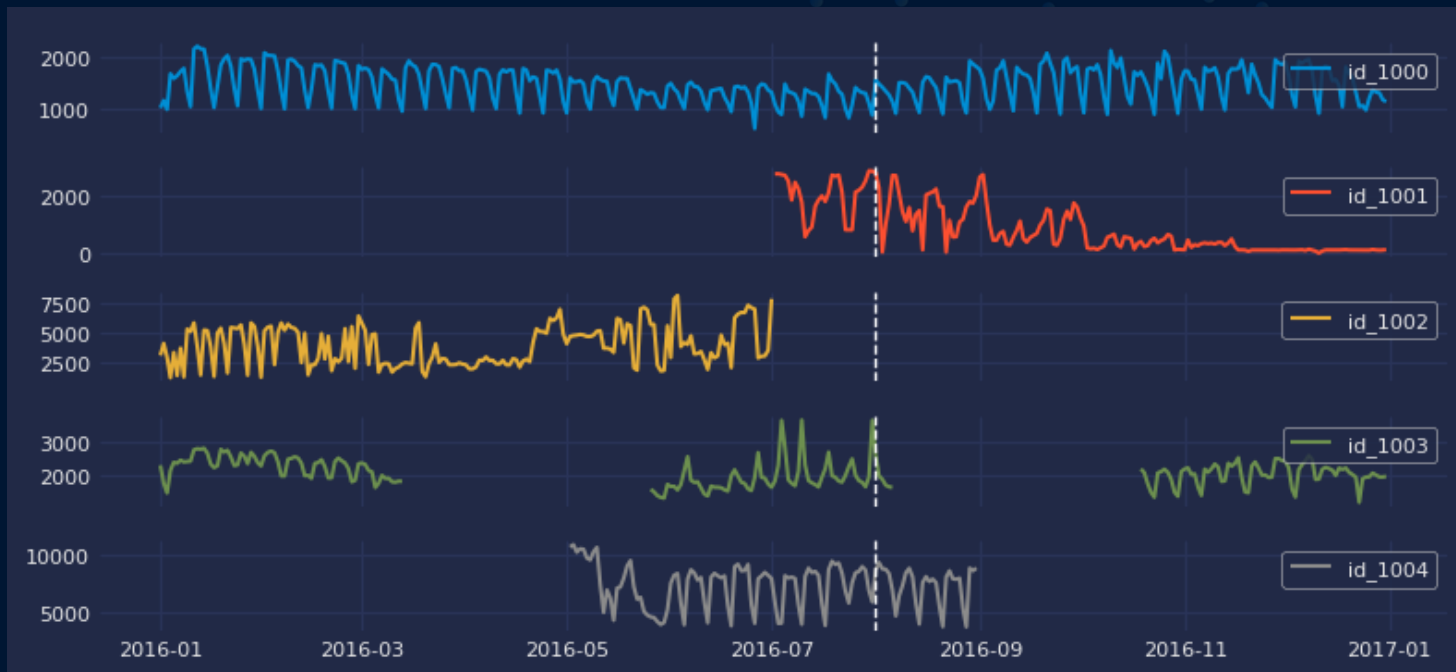
## Dependent multi-series forecasting (Multivariate)

What will be the level of **air pollutants** in the air in the next week?

Measurements made by **all sensors** (flow rate, temperature, pressure...) installed on an industrial air compressor.

# INDEPENDENT MULTI-SERIES FORECASTING

`recursive.ForecasterRecursiveMultiSeries`



# INDEPENDENT MULTI-SERIES FORECASTING

Series 1

1	2	3	4	5	6	7
---	---	---	---	---	---	---

Series 2

101	102	103	104	105	106	107
-----	-----	-----	-----	-----	-----	-----

Exogenous variable

a	b	c	d	e	f	g
---	---	---	---	---	---	---

value	id serie	exog
1	series 1	a
2	series 1	b
3	series 1	c
4	series 1	d
5	series 1	e
6	series 1	f
7	series 1	g
101	series 2	a
102	series 2	b
103	series 2	c
104	series 2	d
105	series 2	e
106	series 2	f
107	series 2	g

ForecasterAutoregMultiSeries Training Matrix

lag 3	lag 2	lag 1	series 1	series 2	exog	Y
1	2	3	1	0	d	4
2	3	4	1	0	e	5
3	4	5	1	0	f	6
4	5	6	1	0	g	7
101	102	103	0	1	d	104
102	103	104	0	1	e	105
103	104	105	0	1	f	106
104	105	106	0	1	g	107

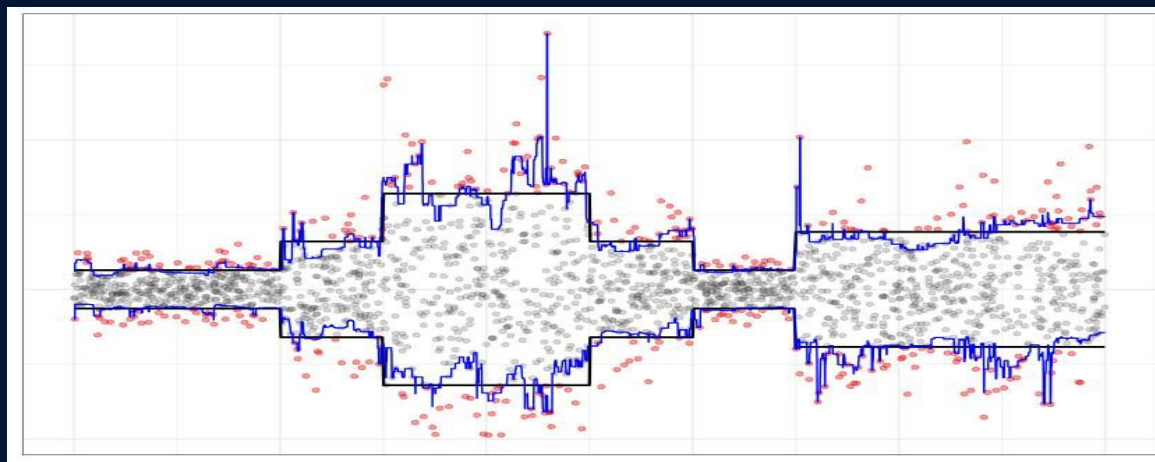
# HANDS ON TIME

# OKAY, BUT THEY ASKED ME FOR PREDICTION INTERVALS

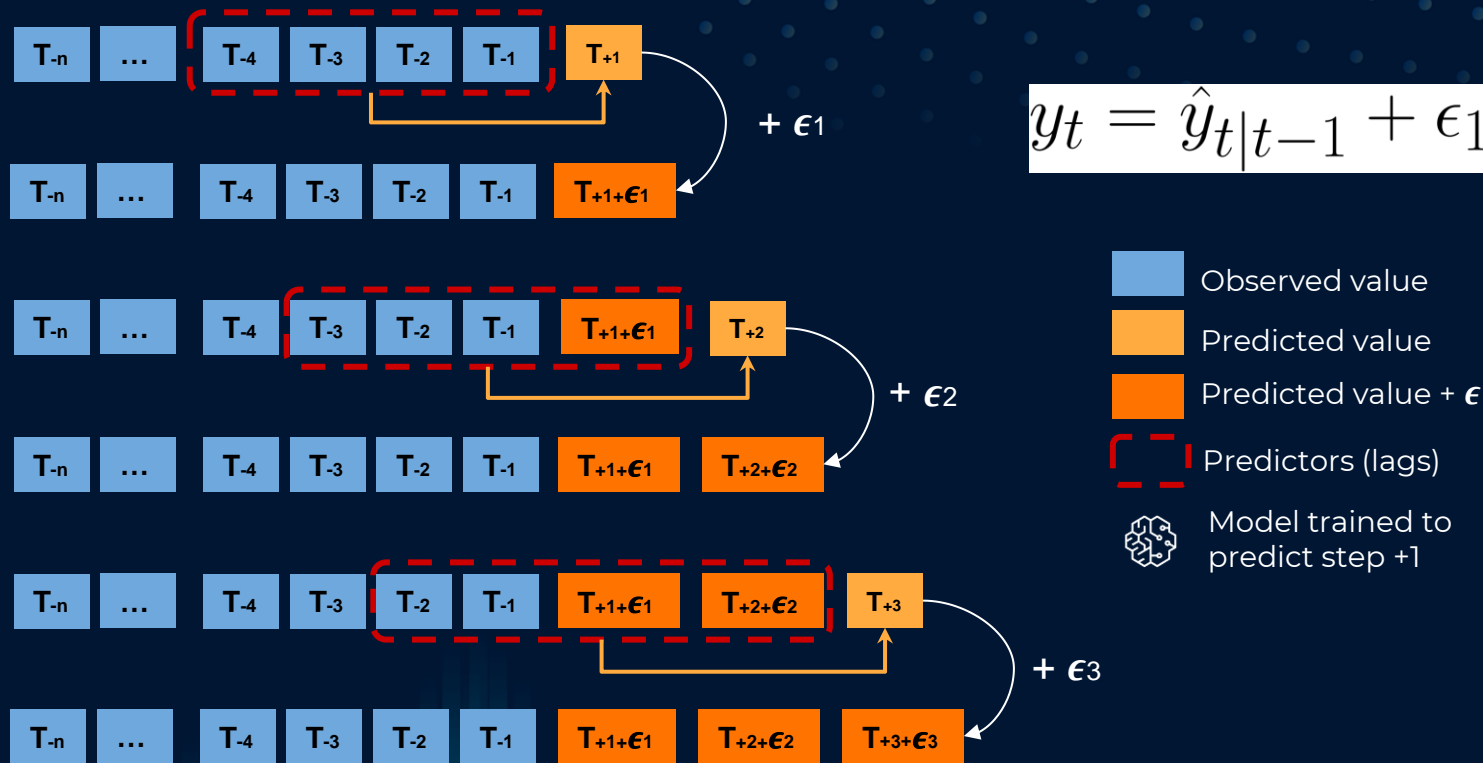


# PROBABILISTIC FORECASTING: PREDICTION INTERVALS

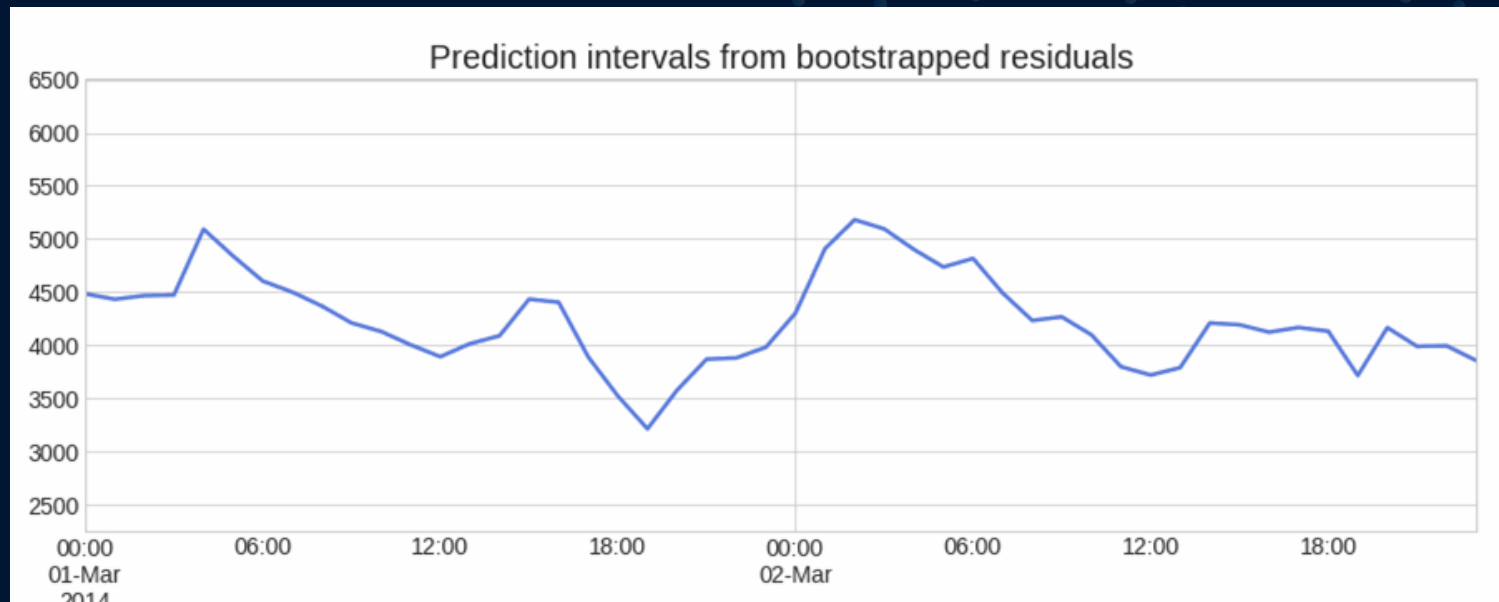
A prediction interval defines the interval within which the true value of  $y$  is expected to be found with a given probability. For example, the 80% prediction interval is expected to contain the true value of the prediction with 80% probability.



# PREDICTION INTERVALS WITH BOOTSTRAPPED RESIDUALS



# PREDICTION INTERVALS WITH BOOTSTRAPPED RESIDUALS



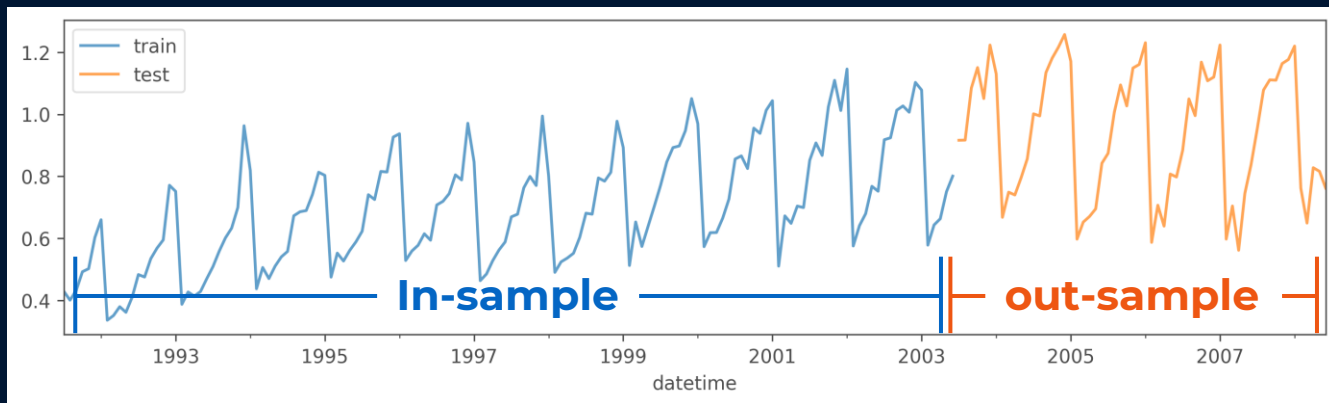
# PREDICTION INTERVALS: RESIDUALS ORIGIN

## In-sample residuals

Residuals calculated using the **training set**, ideal for evaluating the fit of the model to **historical data**.

## Out-sample residuals

Residuals generated in the **validation set**, useful for understanding how the model generalizes to **new data**.



# PREDICTION INTERVALS: BINNED RESIDUALS

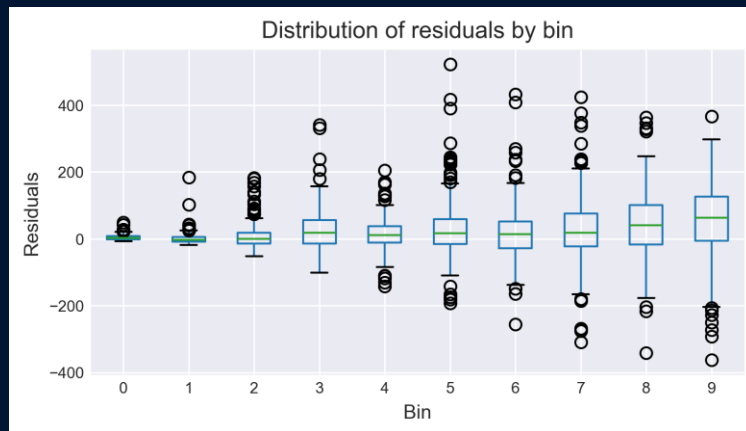
## Residuos condicionados con las predicciones (In or Out)

Residuals organized into **intervals or bins** according to the **value of the predictions**. This allows to reduce the bootstrapping uncertainty and **improve the coverage** of the intervals.

### Binner intervals

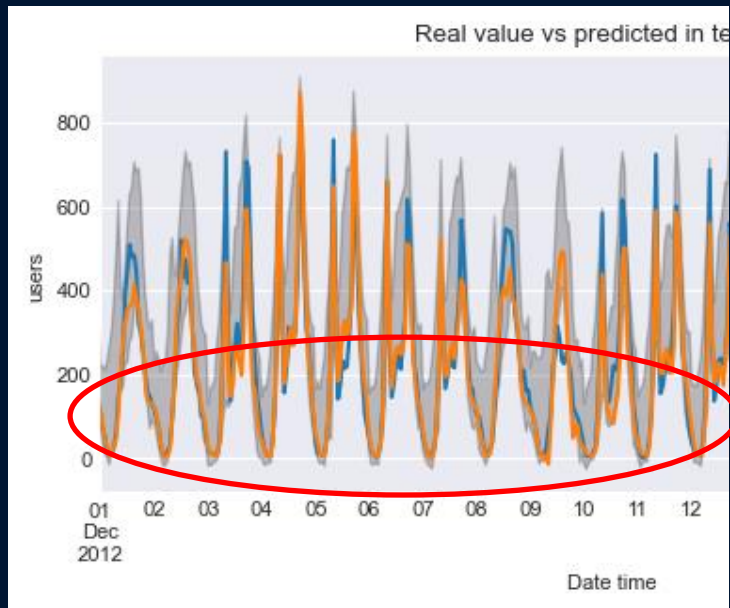
{bin: predictions Interval}

```
{0.0: (-8.229467171553717, 11.116037535200665),  
1.0: (11.116037535200665, 31.879155847370434),  
2.0: (31.879155847370434, 75.9019071402224),  
3.0: (75.9019071402224, 124.5691653220086),  
4.0: (124.5691653220086, 170.35484312260417),  
5.0: (170.35484312260417, 218.96823239624555),  
6.0: (218.96823239624555, 278.6496576655771),  
7.0: (278.6496576655771, 355.13229168292287),  
8.0: (355.13229168292287, 486.1660497574729),  
9.0: (486.1660497574729, 970.517259284916)}
```

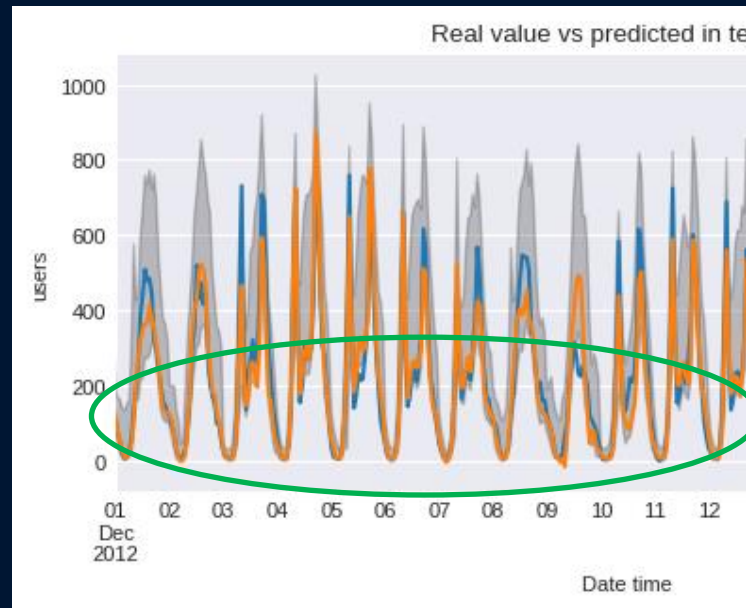


# PREDICTION INTERVALS: BINNED RESIDUALS

**Out-sample residuals no binned**  
Interval coverage: 75.4 %



**Out-sample residuals binned**  
Interval coverage: 84.5 %



# WHAT IF I HAVE TO EXPLAIN ALL THIS?

# FORECASTER EXPLAINABILITY: FEATURE IMPORTANCES

Feature importance is a technique used in machine learning to determine the relevance or importance of each feature (or variable) in a model's prediction. In other words, it measures **how much each feature contributes to the model's output**.

- **Linear regressors:** coefficients.
- **Tree-based models:** mean decrease impurity or permutation feature importance methods.

## Example LGBMRegressor

	feature	importance
7	Temperature	570
0	lag_1	470
2	lag_3	387
1	lag_2	362
6	lag_7	325
5	lag_6	313
4	lag_5	298
3	lag_4	275



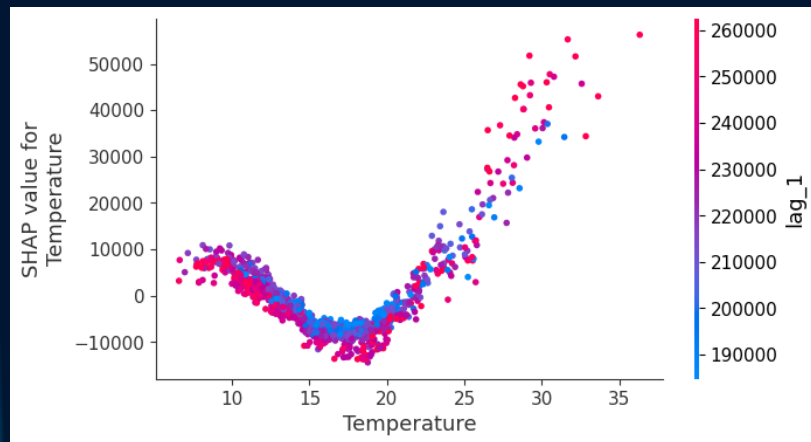
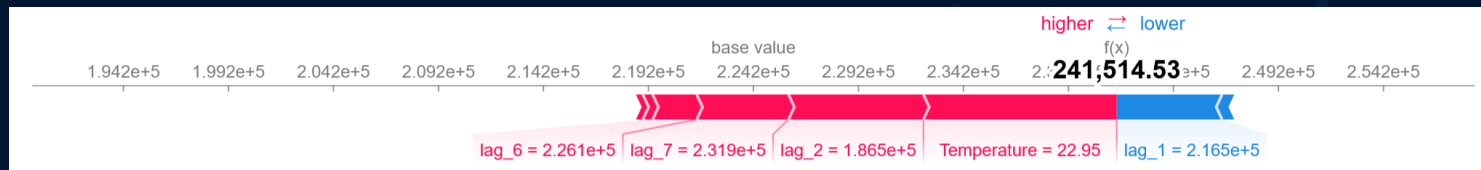
# FORECASTER EXPLAINABILITY: SHAP VALUES

**SHAP** (SHapley Additive exPlanations) values are a widely used approach for interpreting machine learning models, providing a clear understanding of how variables and their values impact predictions, both visually and numerically.

- The **internal regressor** used by the forecaster.
- The **input matrices** derived from the time series. These matrices can either be the ones used for **fitting** the forecaster or those required during the **prediction** phase.


# FORECASTER EXPLAINABILITY: SHAP VALUES

More information: [https://skforecast.org/latest/user\\_guides/explainability](https://skforecast.org/latest/user_guides/explainability)




# STILL MORE?

## ADDITIONAL MATERIAL

 **Introducción al forecasting** - Basics of forecasting concepts and methodologies

 **Quick start** - Get started quickly with skforecast

 **User guides** - Detailed guides on skforecast features and functionalities

 **Examples and tutorials** - Learn through practical examples and tutorials to master skforecast

 **FAQ and tips** - Find answers and tips about forecasting

 **API Reference** - Comprehensive reference for functions and classes

## STATE OF THE ART



## GET INVOLVED 😊

- Star on GitHub ★
- Share your feedback on LinkedIn 
- Report bugs and suggest new features
- Contribute with new code or add new features.



# ¿QUESTIONS?

Joaquín Amat Rodrigo  
Javier Escobar Ortiz