Time Series Forecasting Models in Python

Time Series Forecasting Models in Python

Introduction to Forecasting

Predictions that were just wrong

Thomas Watson, chairman of IBM
When: 1943
I think there is a world market for maybe five computers.

Jonh Maynard Keynes
Three hour shifts or a fifteen-hour work week

Steve Ballmer
There's no chance that the iPhone is going to get any significant market share.

Einstein
There is not the slightest indication that nuclear energy will ever be obtainable. That would mean that the atom would have to be shattered at will.

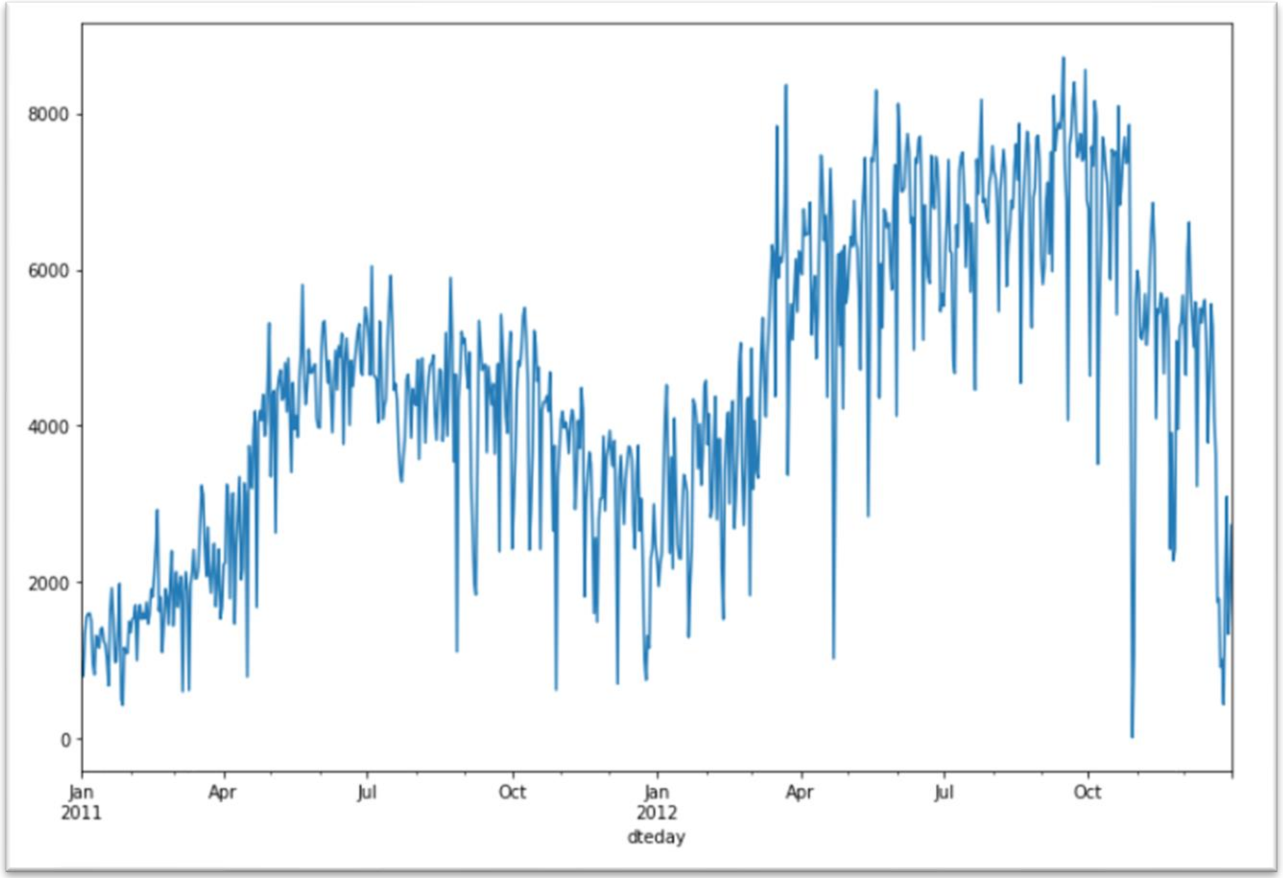
Analytics is key to drive Forecasting

Description

1. Bringing Science to a sometimes gut-feeling job
2. Barometer for the company -> Quantifies direction
3. Understanding turning points
4. Can uncover opportunities

What is Time Series Data?

Visualization



Diogo Resende | Time Series Forecasting Models in Python

Key ideas

- Sequence of data points in time order (oldest to newest)
- Most commonly, it is data recorded in equally distanced time periods
- Type of Panel Data (multidimensional dataset)

Case Study Briefing – Demand Forecasting

Bike Sharing

How many rides are done per day?

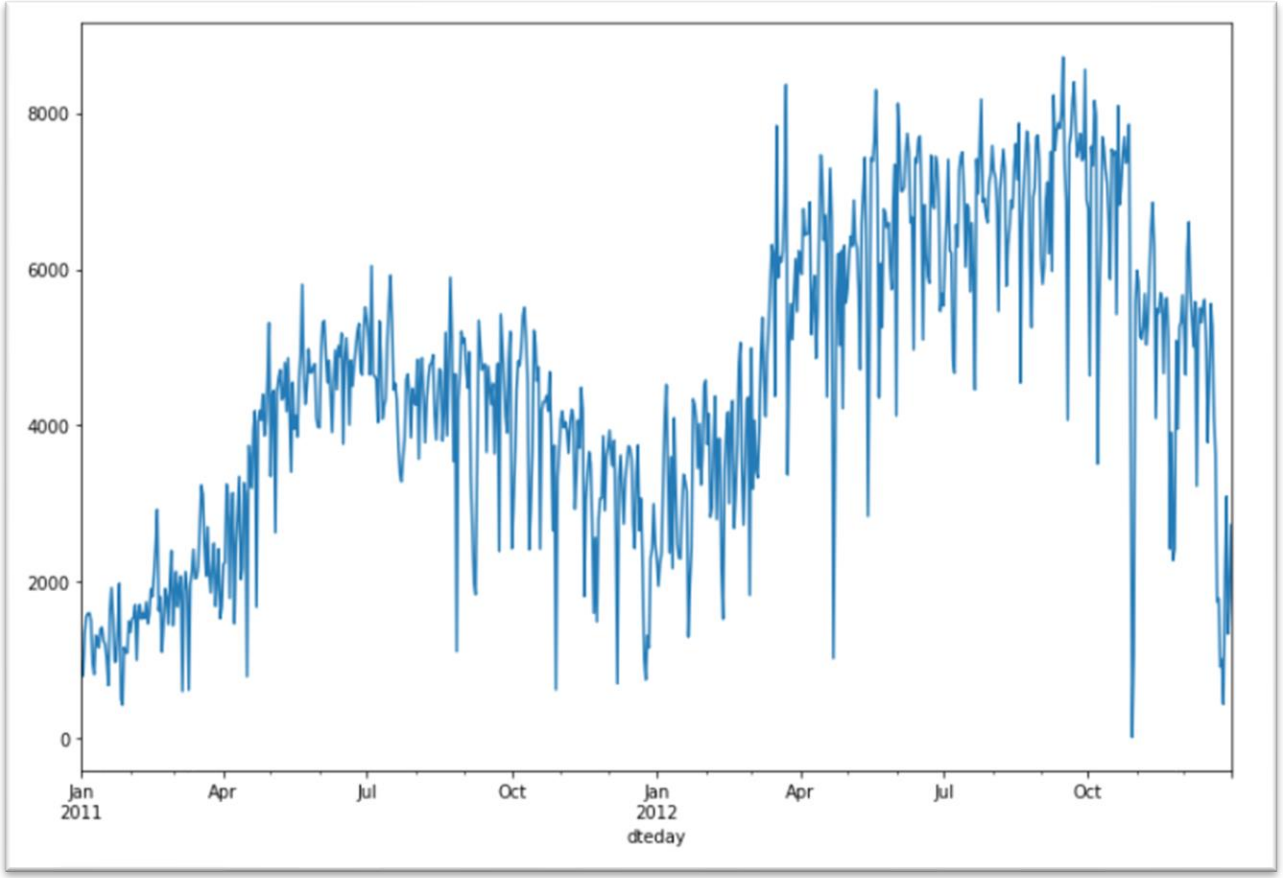
1. Holidays and weather KPIs included
2. Time periods: 2011 and 2013
3. Forecast December 2012 to assess each forecasting model

[1] Fanaee-T, Hadi, and Gama, Joao, "Event labeling combining ensemble detectors and background knowledge", Progress in Artificial Intelligence (2013): pp. 1-15, Springer Berlin Heidelberg, doi:10.1007/s13748-013-0040-3.

Seasonal Decomposition

Seasonal Decomposition: the actuals values to be decomposed

Visualization



Key ideas

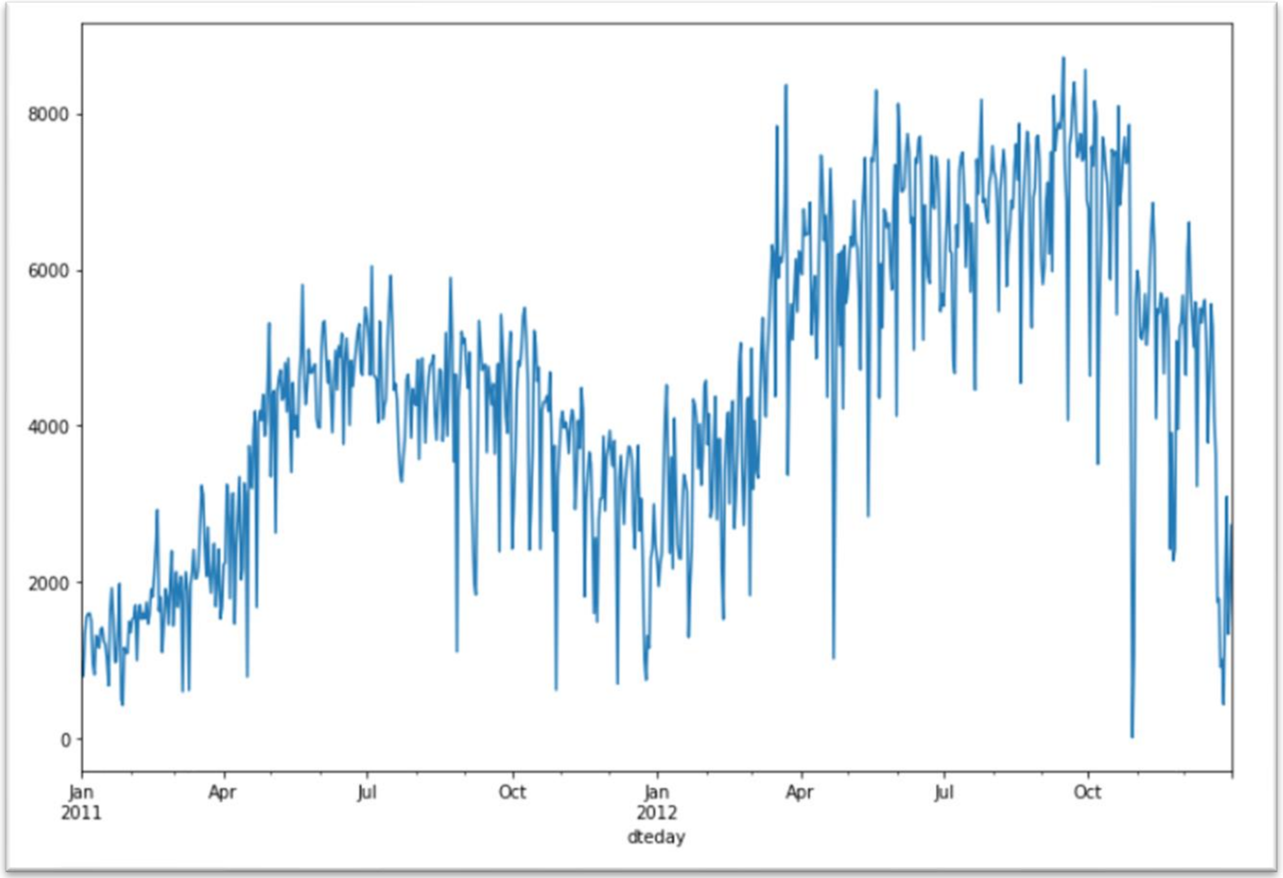
A seasonal Time Series can be decomposed into:

- **Trend**
- **Seasonality**
- **Error**

We try to use external regressors to model the remaining error term.

Seasonal Decomposition: Trend

Visualization

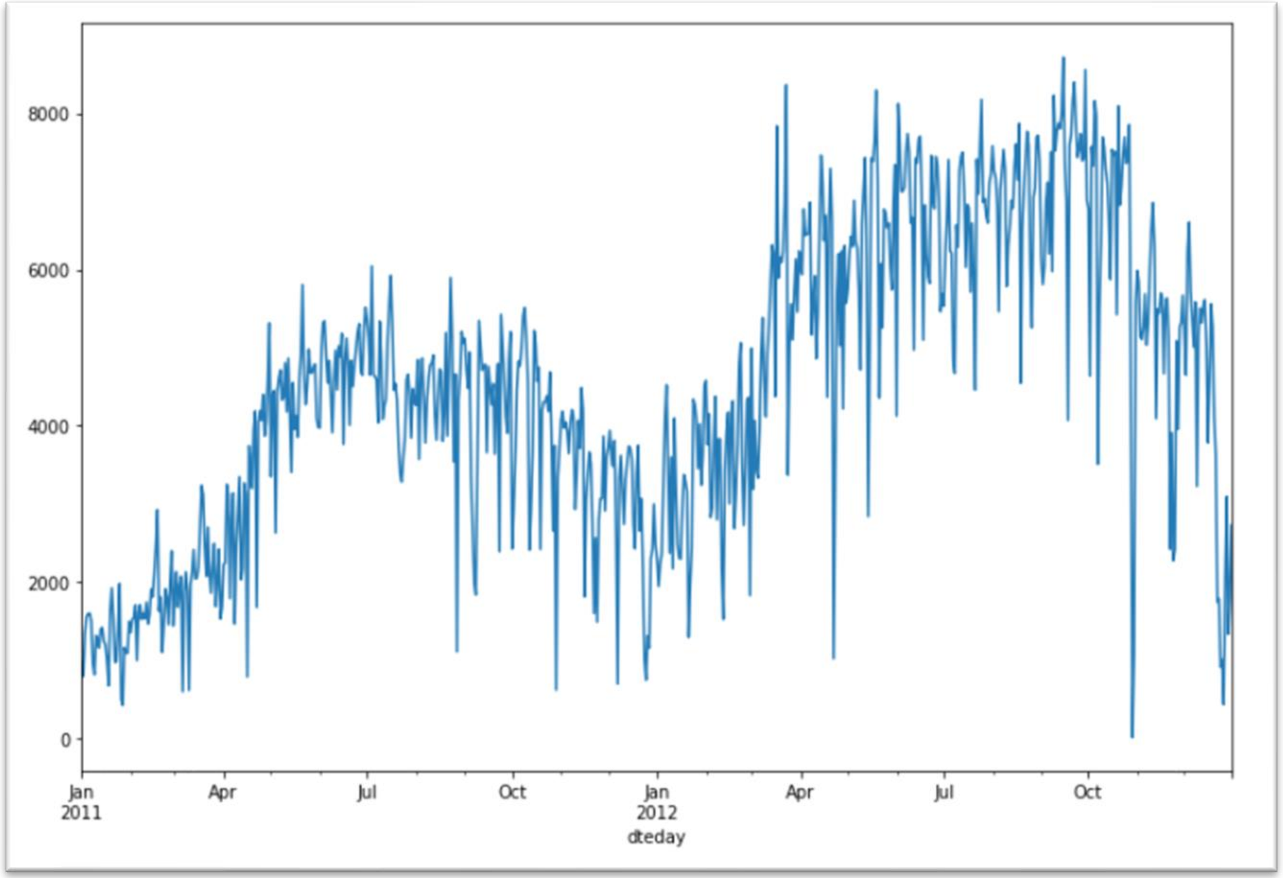


Diogo Resende | Time Series Forecasting Models in Python

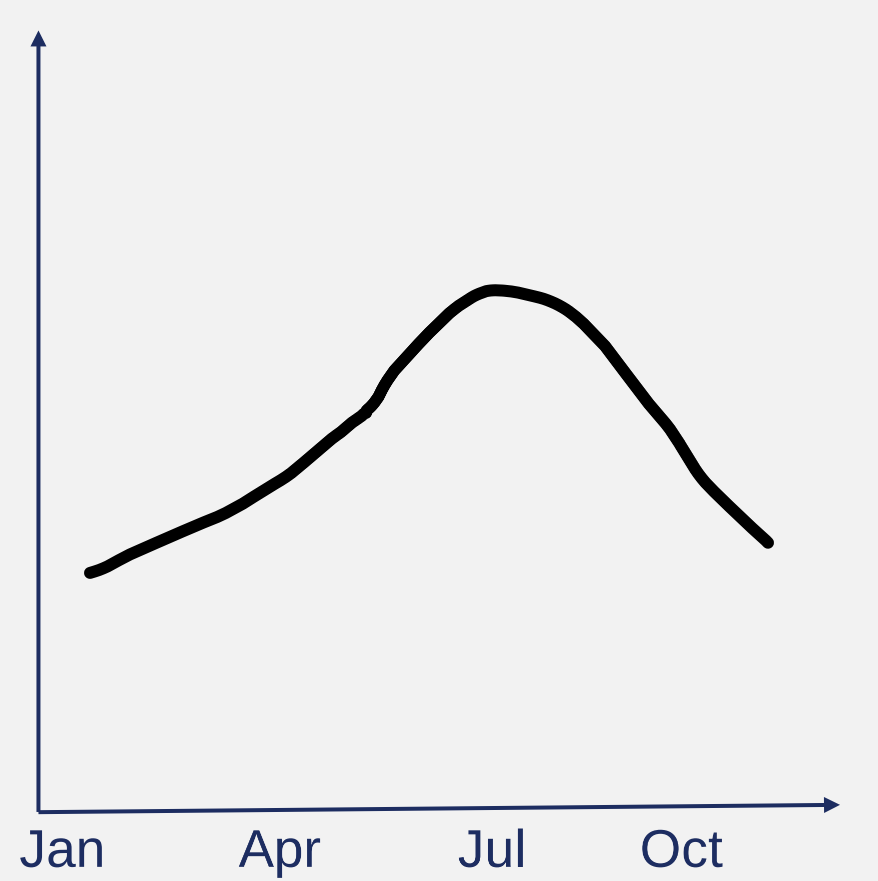
Trend

Seasonal Decomposition: Seasonality

Visualization

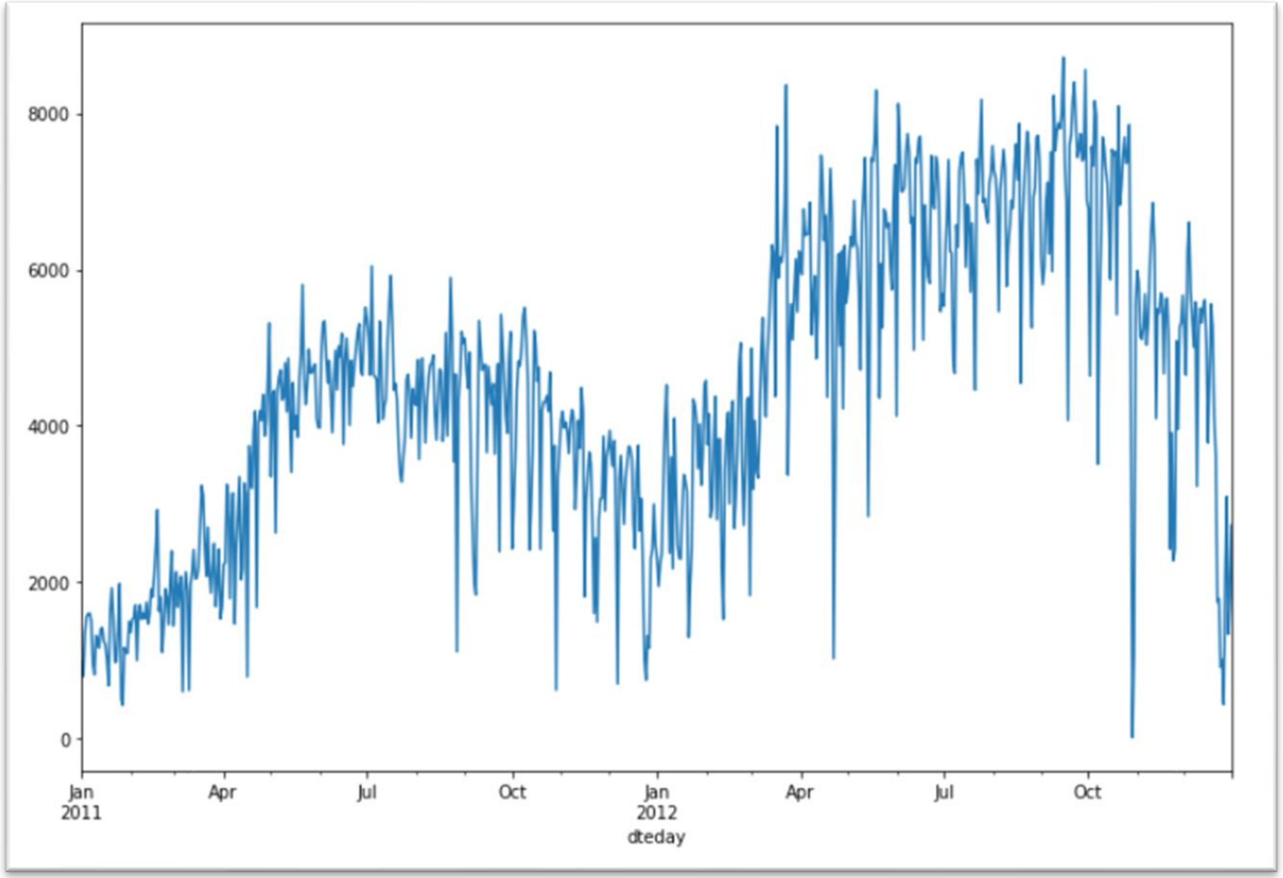


Seasonality



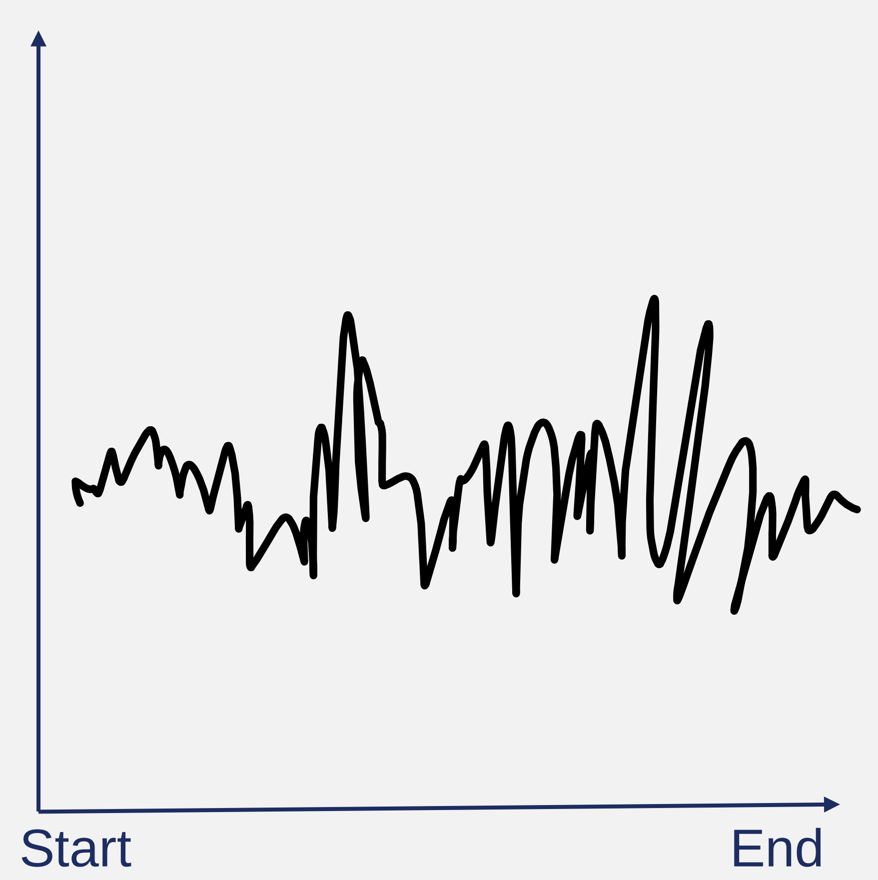
Seasonal Decomposition: Error

Visualization



Diogo Resende | Time Series Forecasting Models in Python

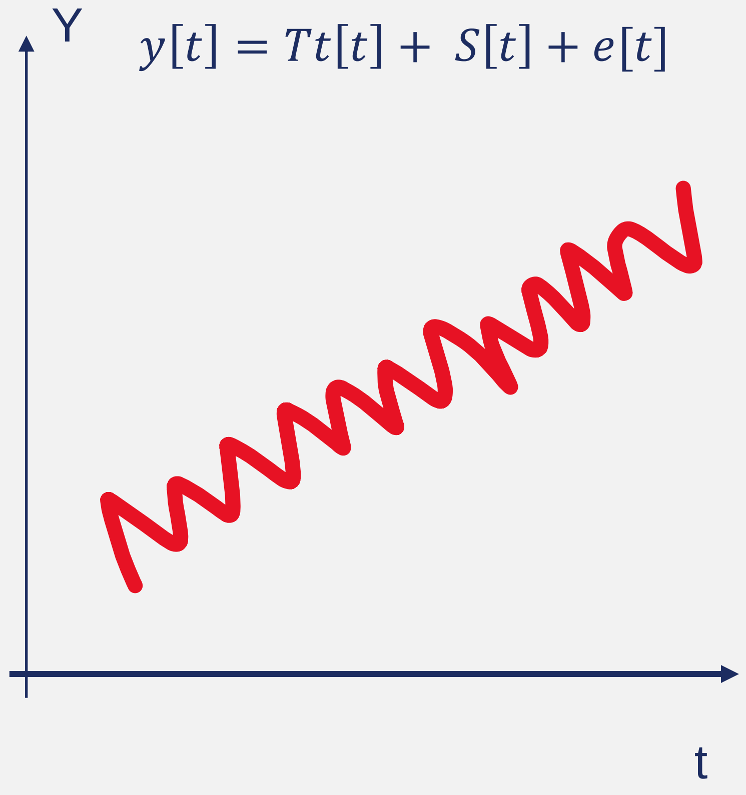
Error



Additive vs. Multiplicative

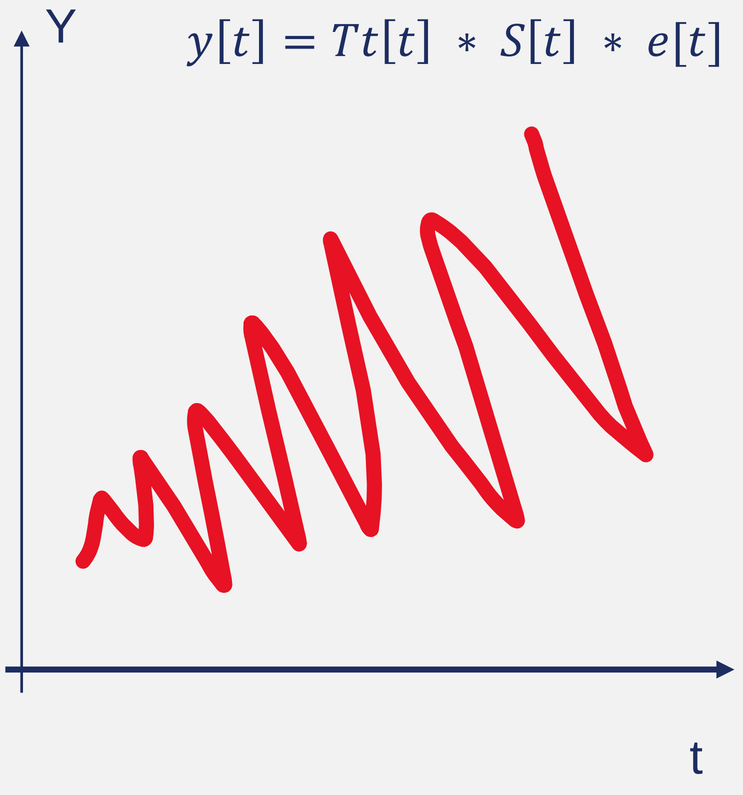
Additive

$$y[t] = Tt[t] + S[t] + e[t]$$



Multiplicative

$$y[t] = Tt[t] * S[t] * e[t]$$



Key ideas

If we talk about seasonality in terms of percentage, then we should consider a multiplicative seasonality.

If it is in adding absolute values, then it is additive.

If trend is exponential, then it is multiplicative

Forecasting is all about error modelling

Descrition

- The essential part of forecasting
- Understanding what else can explain the Error
- How? Usually in the form of external regressors
- High errors in the beginning of dataset? Consider discarding that part of the data.

Data without patterns : Stocks

Key Idea

- If there is no pattern, you should not use forecasting models
- Forecasting models work best with consistent seasonality and trends

Trend

- Heavily dependent on the company

Seasonality

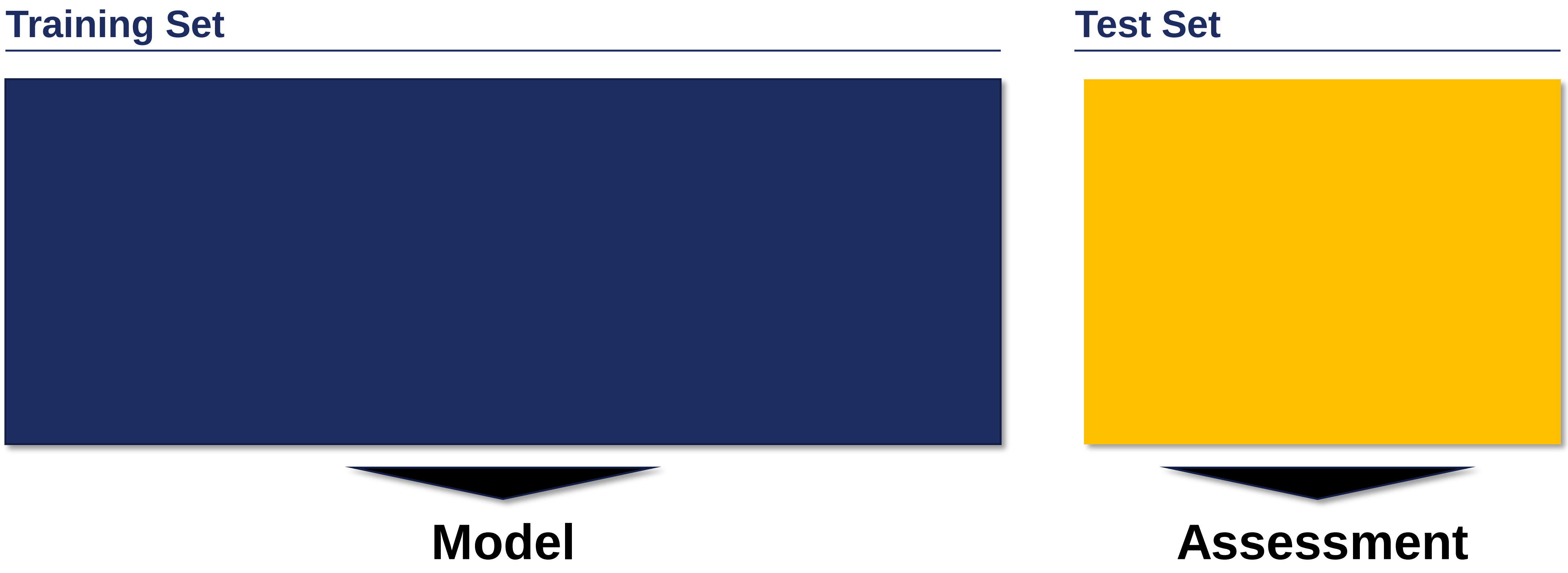
- Depends more on the industry, thus it is more predictable.

Exponential Smoothing & Holt-Winters

Let‘s imagine this is our full data set

Description

Splitting between training and test enables an unbiased model assessment



Training and Test set in Time series

Key Ideas

Forecasting Models are usually split into a pre and post period from a time perspective

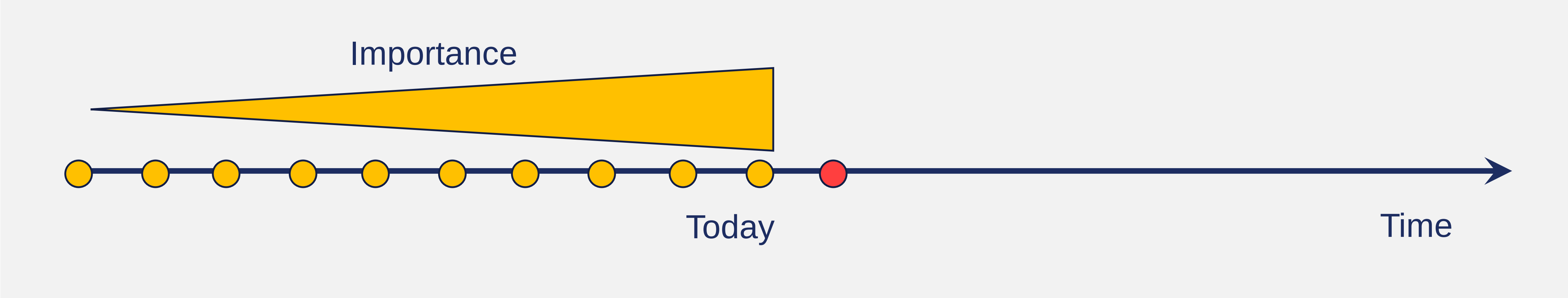
The Test Set should be of the size of a real-world forecast

What is Exponential Smoothing?

Key Ideas

Weighted averages of past observations, with the weights decaying exponentially as the observations get older

Visualization



Holt-Winters is a Triple split Exponential Smoothing

Splits the time series into 3:

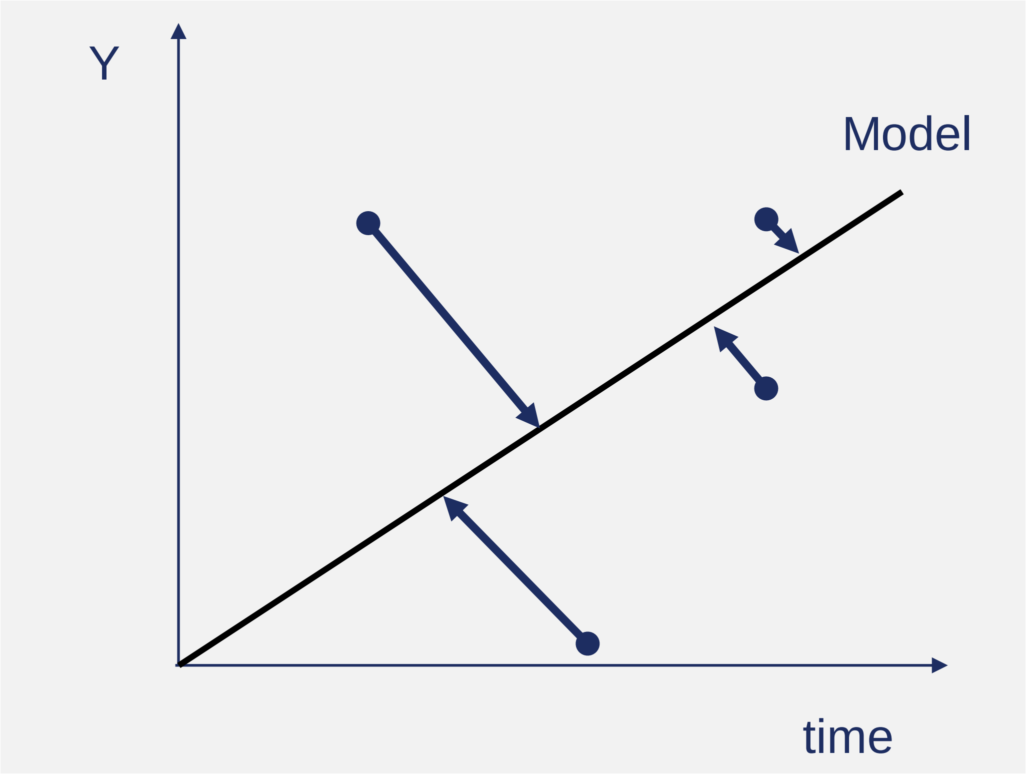
- Level
- Trend
- Seasonality

Key Ideas

- Performs Exponential Smoothing in each of the 3 levels
- Holt-Winters is also called Triple Exponential Smoothing
- There are 2 variants: Additive and Multiplicative

Mean Absolut Error (MAE) vs Root Squared Mean Error (RSME)

Visualization



Key ideas

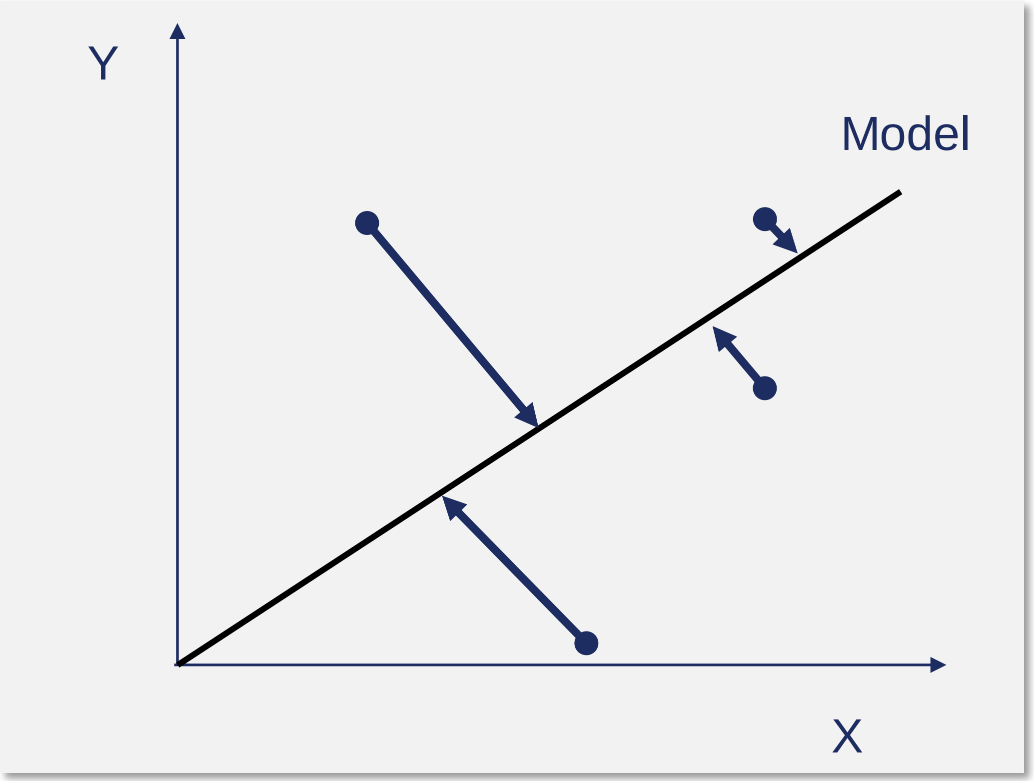
- MAE and RSME are performance indicators for Regression models with continuous dependent variables

$$MAE = \frac{\sum|y - \hat{y}|}{n} \quad \text{x} \quad RSME = \sqrt{\frac{\sum(\hat{y} - y)^2}{n}}$$

- RSME is quite useful for models with extremes / outliers
- MAE is more interpretable.

Mean Absolut Percent Error (MAPE)

Visualization



Key ideas

- MAPE represents a very interpretable way of measuring errors

$$MAPE = \frac{\sum \frac{|y - \hat{y}|}{y}}{n}$$

- Clear downside is that all error has the same relevance, regardless of the magnitude, if the percent error is the same

- There is no universal good accuracy measure. It will depend on your problem and business need!

Pros and Cons

+

Pros		Cons
1	Easy to Apply	Does not allow external regressors
2	Easy to understand	Low Flexibility
3		Better with low amount of time periods or frequency

−

Challenge

Description

Use Holt-Winters to predict the amount of airmiles

1. Set Index frequency to Monthly. Use „MS“
2. Visualize data
3. Create Training and Test Set. Test Set should be 12 months
4. Create Holt-Winters Model
5. Predict 12 months and visualize them, together with the training and test set
6. Assess Model based on MAE

Dataset: TSA package

TBATS

Meaning of TBATS

Description

1. Trigonometrics seasonality
2. Box-Cox transformation
3. AutoRegressive Moving Average
4. Trend
5. Seasonality

Origin

Created in 2011
Similar to Exponential Smoothing

Why

The math behind has several similarities

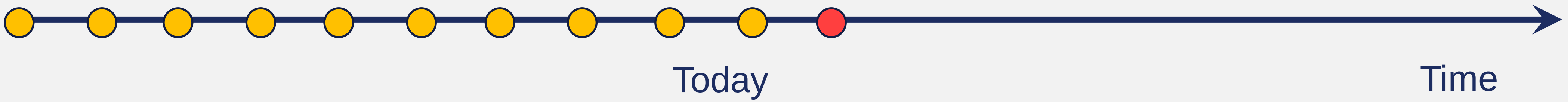
AutoRegressive components

Key Idea

Past values, the lags, contain information that help predict future values

Visualization

$$Y_t = c + \alpha_1{}^{*}\, Y_{t-1} + \alpha_2 *\, Y_{t-2} + \ldots + \alpha_n *\, Y_{t-n}$$

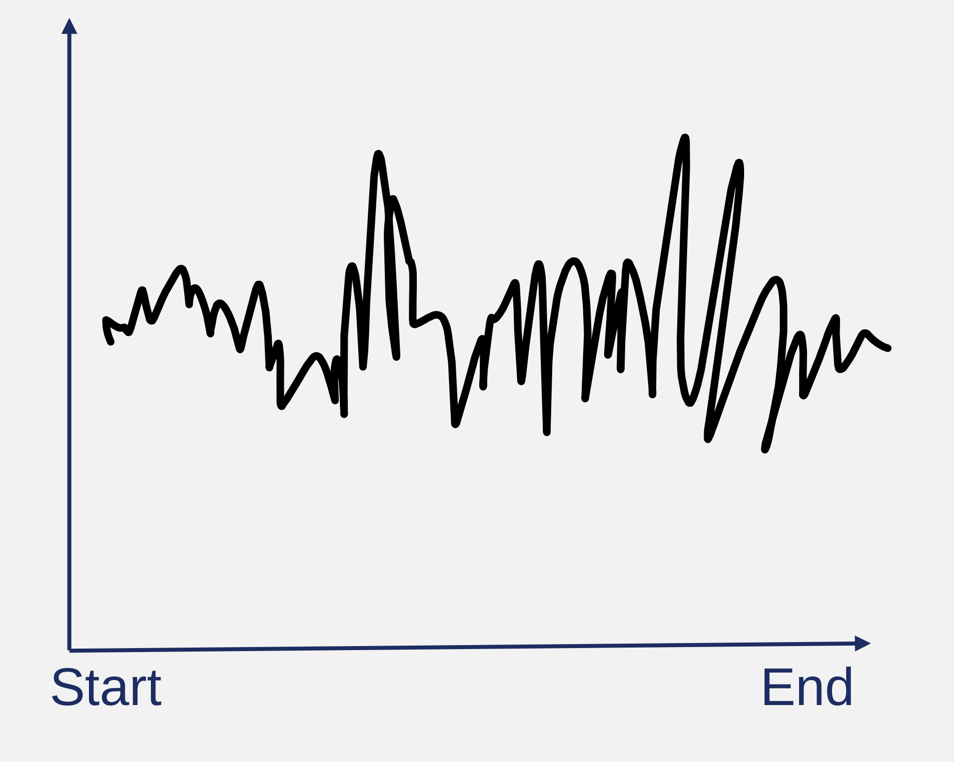


How to determine how many lags

We will do it **automatically** in the practice tutorials

Moving Average components

Visualization of the errors



Methodological Framework

$$y_t = c + \alpha_1 * \varepsilon_{t-1} + \ldots + \alpha_n * \varepsilon_{t-n}$$

What it is?

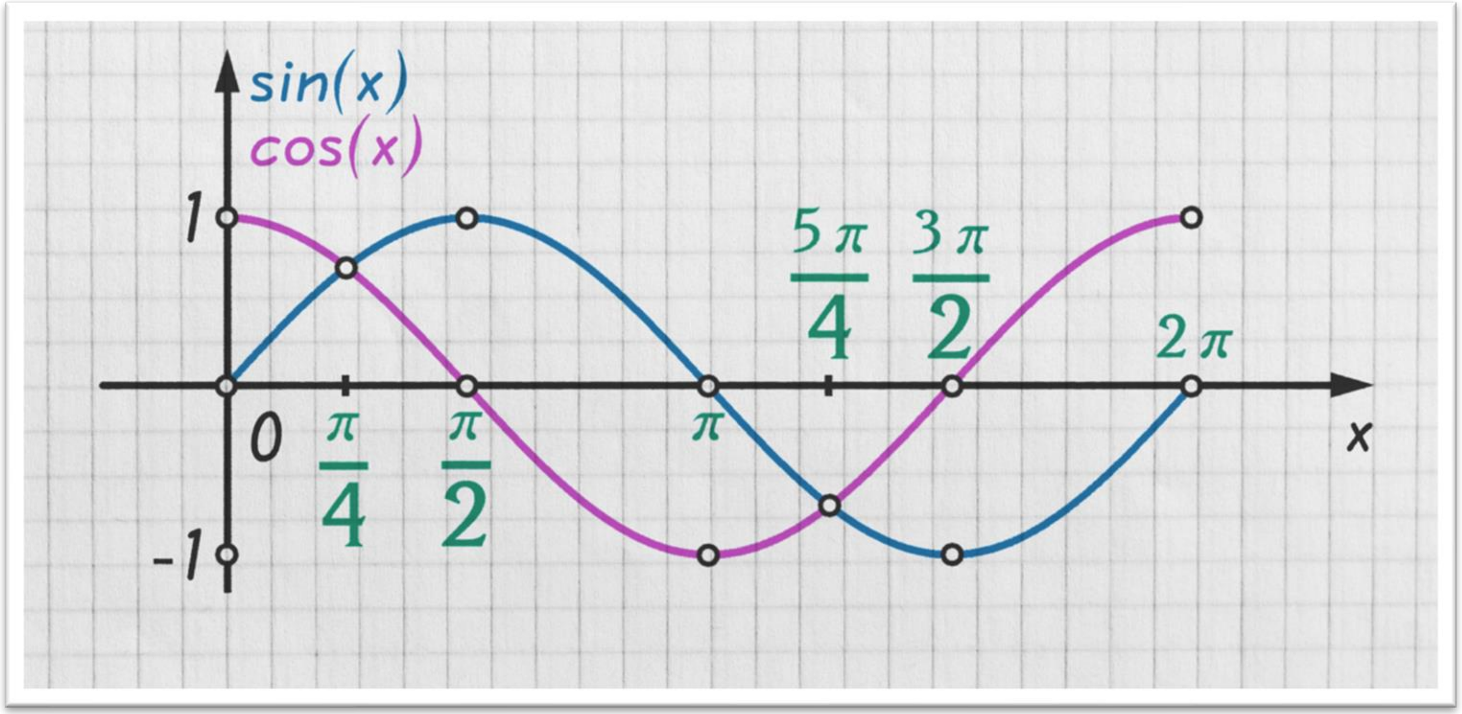
Past error lags, contain information that help predict future values

How to do it?

We will do it **automatically** in the practice tutorials

Trigonometric seasonality

Visualization

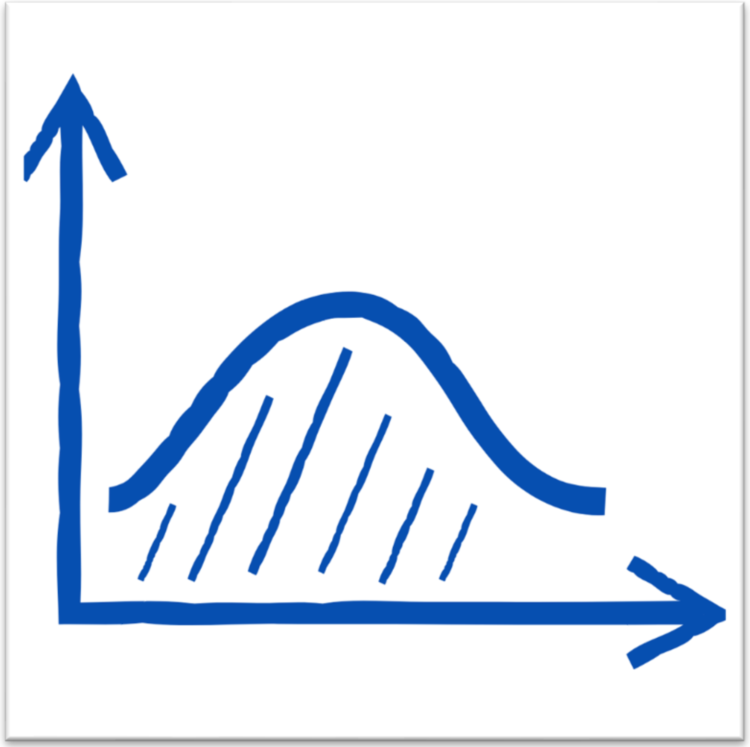


Description

- Trigonometry is part of the modelling.
- Seasonality equation contain the Sine and Cosine
- In practical terms, we do not need to do anything

BOX-COX

Visualization



What is it?

Transforming the dependent variable into a normal distribution

Why do we care?

Normal distribution is a requirement or assumption of many statistical techniques

Key Idea

Box Cox is part of the modelling.
In practical terms, we do not need to do anything

Pros and Cons

+

Pros		Cons	
1	Seasonality is allowed to change overtime	1	Prediction intervals often wide
2	Automated Optimization	2	Does not allow external regressors
3	Easy implementation	3	Slow

Challenge

Description

Use TBATS to predict weekly store footfall

1. Transform Index to have weekly frequency. Use „W“
2. Visualize data. Something will be off ;)
3. Create Training and Test Set. Test Set should be 5 weeks
4. Create TBATS Model
5. Predict 5 weeks and visualize them, together with the training and test set
6. Assess Model based on RMSE

Source: UK Government

ARIMA, SARIMA & SARIMAX

What does it all mean?

Acronym	Description
ARIMA	AutoRegregressive Integrated Moving Average
SARIMA	Seasonal + ARIMA
SARIMAX	SARIMA + Exogenous variables

What is ARIMA?

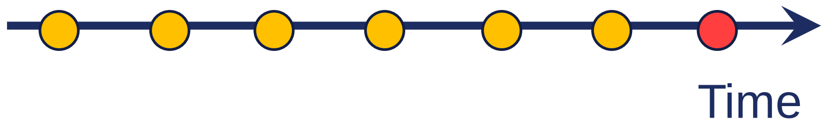
Component	Description
AutoRegressive	The output is regressed on its own lagged values
Integrated	Number of times we need to do differencing to make our time series stationary
Moving Average	Instead of using the past values, the MA model uses past forecast errors.

ARMA recap

AutoRegressive

Past values, the lags, contain information that help predict future values

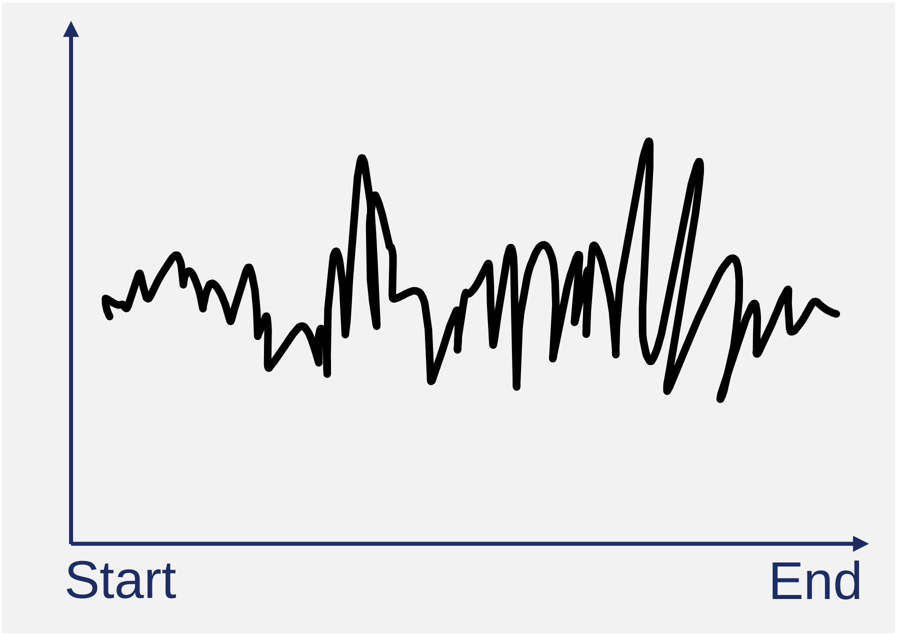
Visualization



Moving Average

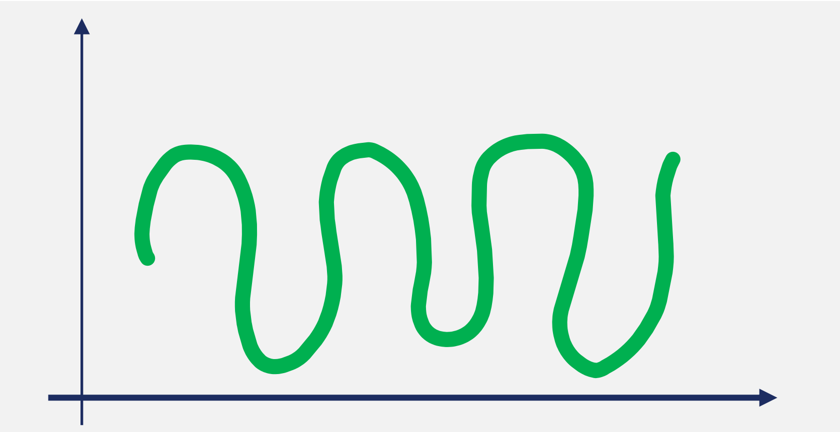
Past error lags, contain information that help predict future values

Visualization

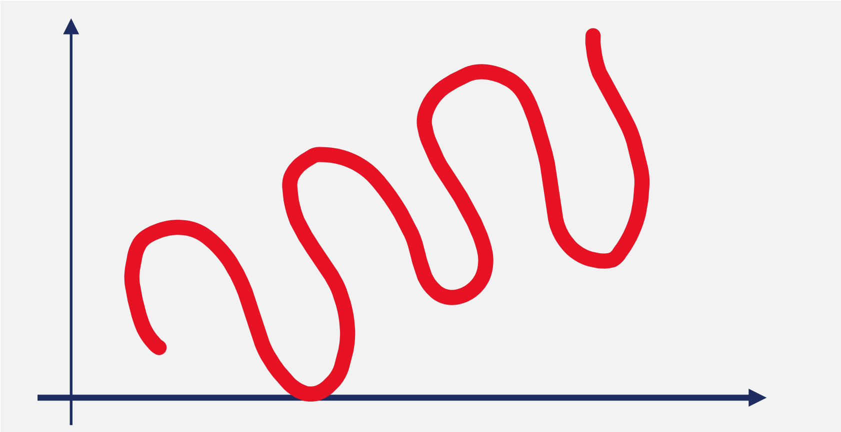


Stationarity

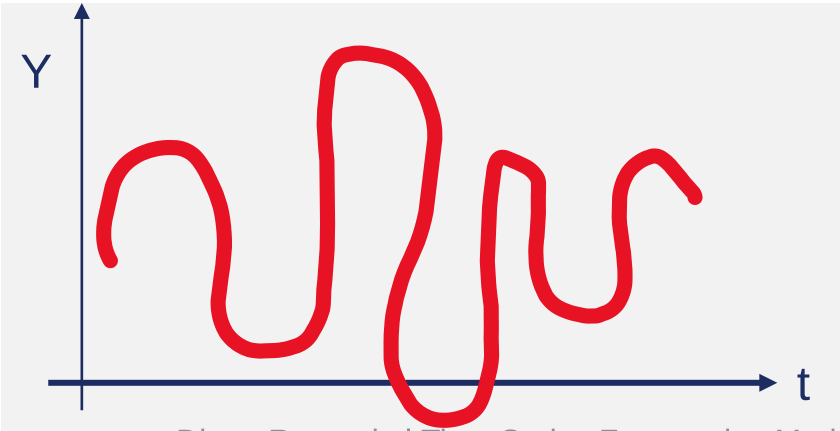
Stationary Time Series



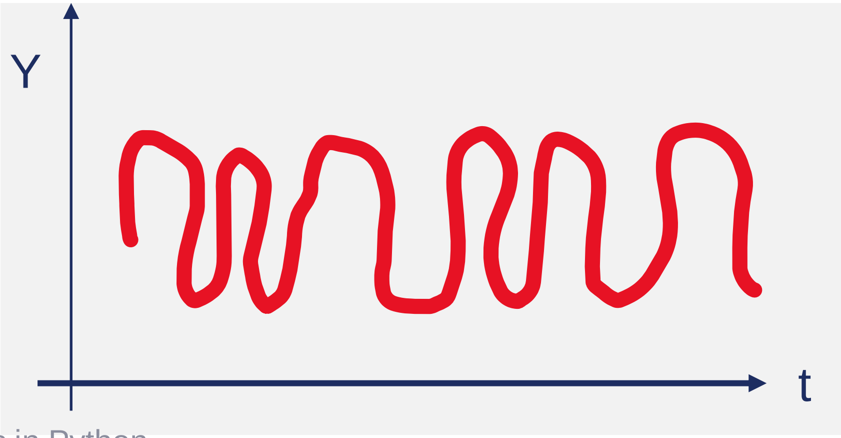
Time dependent mean



Time dependent variance



Time dependent covariance



Key idea

Mean, variance and covariance are not time dependent

Stationary Time Series have a clearly defined pattern

Statistical test:

Dickey-Fuller test. If p-value is less than 0.05, time series is considered stationary

Making Data Stationary

Time Series	1st differencing	2nd differencing
5	NA	NA
9	4	NA
1	-8	-12
7	6	14
3	-4	-10
7	4	8
4	-3	-7

Key idea

Making data stationary is simple, yet the concept is confusing.

From a practical perspective, it is a check that we need to do

The Auto.arima function does it automatically for us!

SARIMAX

External Regressors

- The goal of the regressors is to model the remaining error.
- Information that is not recurrent over time or modifies itself.

Examples

- **Moving seasonality**

Events like Black Friday or seasonal holidays like Easter or Diwali are not in the same dates every year.

- **Events outside the company control**

Factors like weather or corona interfere with the usual seasonality or trend, thus you need to model them in your forecast to decrease errors

- **Events caused by the company**

Major investment or strategy shifts affect the normal development of a KPI. You need to try to find a metric that represents any of these factors

3 factors to optimize in ARIMA(p,d,q)

Order	Description	Explanation
p	Order of the Autoregressive	Number of unknown terms that multiply your signal at past times
d	Degree of first Differencing involved	Number of differences to make time series stationary
q	Order of the Moving Average part	Number of unknown terms that multiply your forecast errors at past times

Key Idea

P, d, and q are non-negative integers.

No extra work, there are functions to optimize the factors automatically

6 factors to optimize in SARIMA

	Data Type	Acronym	Factors
Time Series	Seasonal Data →	S →	P, D, Q
	Non-seasonal Data →	ARIMA →	p, d, q

Key Idea

Despite having 3 more factors to optimize, they mirror the classic ARIMA (p, d, q)

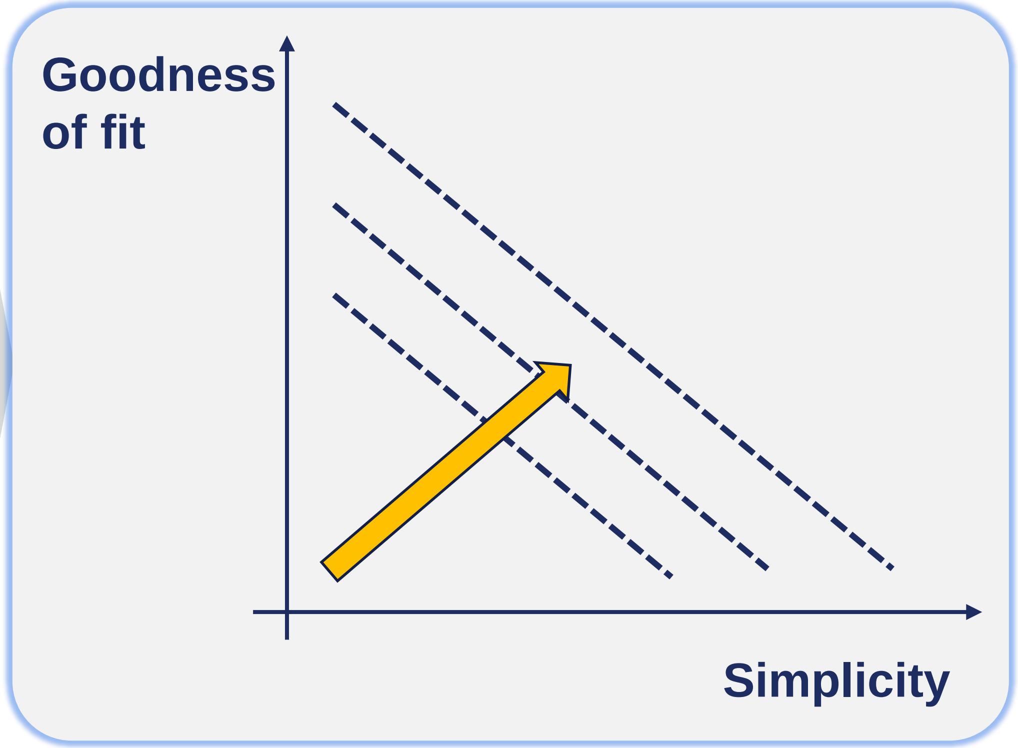
No extra work, there are functions to optimize the factors automatically

Akaike's Information Criterion (AIC) and Bayesian Information Criterion (BIC)

Key Ideas

- AIC and BIC provide a means to select a model
- Trade-off between simplicity and goodness of fit
- Deal with overfitting and underfitting

Pseudo-visualization



Pros and Cons

+

1. Easy Implementation
2. Automated Optimization
3. Easy to Understand

−

1. Better with low amount of time periods or frequency
2. Low Flexibility

Challenge

Description

Use SARIMAX to predict interest in Churrasco

1. Transform Index to have weekly frequency. Use „W“
2. Visualize data.
3. Create Training and Test Set. Test Set should be 10 weeks
4. Extract Exogenous Variables and Create SARIMAX model
5. Predict 10 weeks and visualize them, together with the training and test set
6. Assess Model based on MAPE

Source: Google Trends

Tensorflow Probabilities Structural Time Series

Structural Time Series

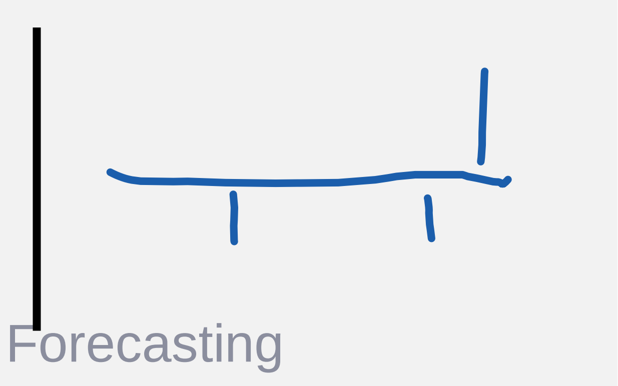
Visualization

Data

Seasonality

Trend

Exogenous impacts



Description

- Structural Time Series is the decomposition of the data in at least:
  - Trend
  - Seasonality
  - Exogenous impacts
  - Leftovers: noise

Methodological framework

$$y(t) = c(t) + s(t) + x(t) + \epsilon$$

Tensorflow Structural Time series

Decomposition

- Trend
- **Seasonality - multiple**
- Exogenous impacts
- **AutoRegressive**
- Noise

Seasonality

- Weekly
- Monthly
- Yearly

Autoregressive

- Focus on giving weight to recent information

Hamiltonian Monte Carlo

Description

Simulation used for Bayesian Inference

Causal inference problem statement

We know what happenened, but we do not know what led to it

Bayes Theorem

$$P(buy|impression) = \frac{P(impression|buy) * P(buy)}{P(impression)}$$

$$= \frac{P(impression|buy) * P(buy)}{\int P(impression|buy) * P(buy) d(buy)}$$

Problem statement

It is not possible to solve the equation and thus we simulate outcomes

Tensorflow Structural Time Series Pros and Cons

+

Pros		Cons	
1	Flexible	1	Complex programming
2	Continuous Regressors	2	Very slow
3	Good with short-term dynamics		
4	Intuitive		

Challenge

Description

Udemy wikipedia page visits

1. Set as regressors Easter and Christmas variables
2. Split into training and test set and isolate Y
3. Create weekly and monthly seasonality objects
4. Create Trend and Autoregressive components
5. Create Tensorflow model and fit it with Hamiltonion Monte Carlo
6. Predict 30 days and add index to the predictions.
7. Visualize forecast, trainining and test data

Dataset: TSA package

Facebook Prophet

Facebook Prophet quick facts

Which?

Description

1. Built by facebook
2. Stan background - probabilistic programming language for statistical inference
3. Dynamic Holidays
4. Prophet forecasts are customizable in ways that are intuitive to non-experts
5. Built-in Cross Validation & Hyperparameter Tuning

Prophet Mechanics

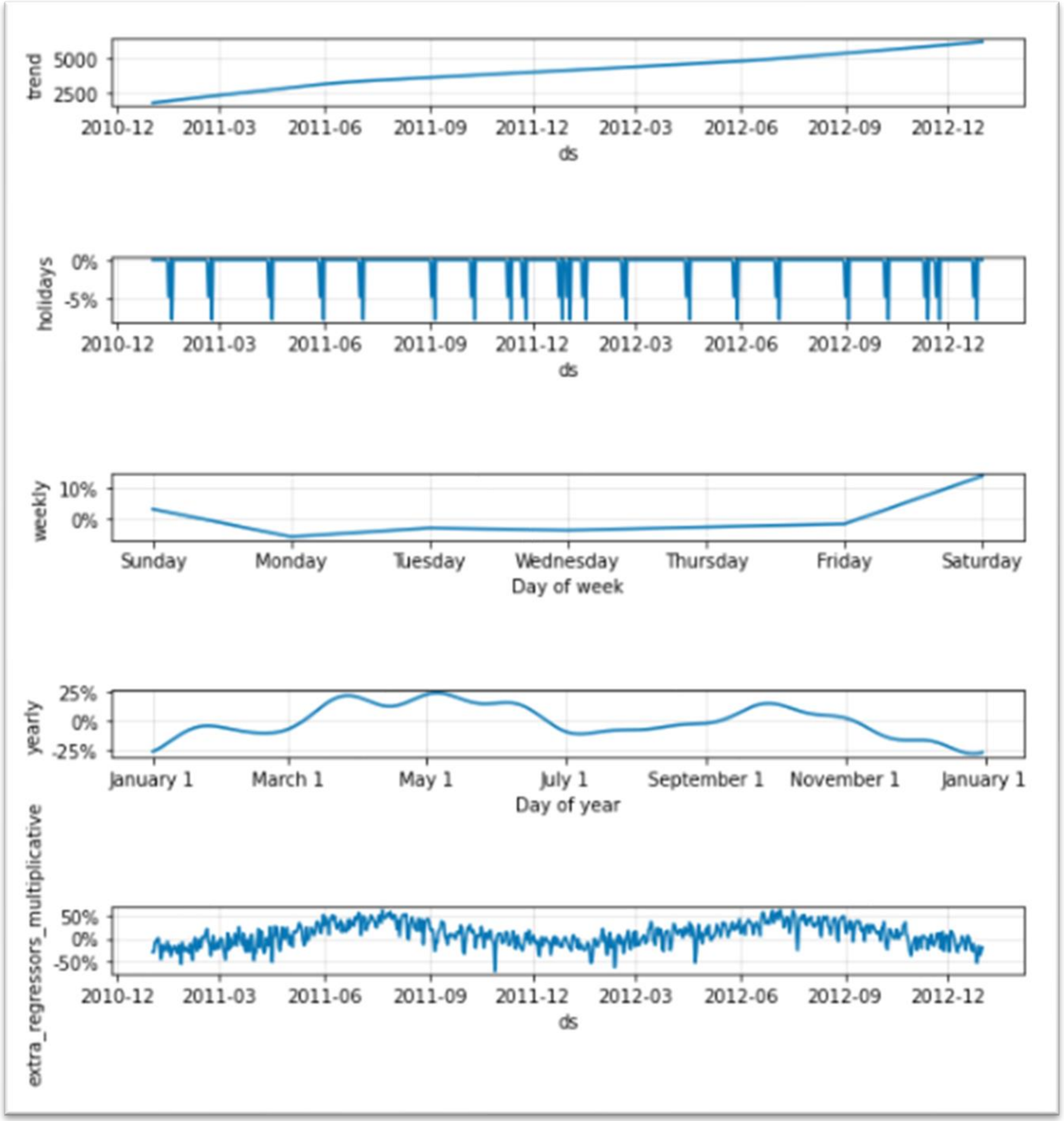
Methodological framework

$$y(t) = c(t) + s(t) + h(t) + x(t) + \epsilon$$

Where:

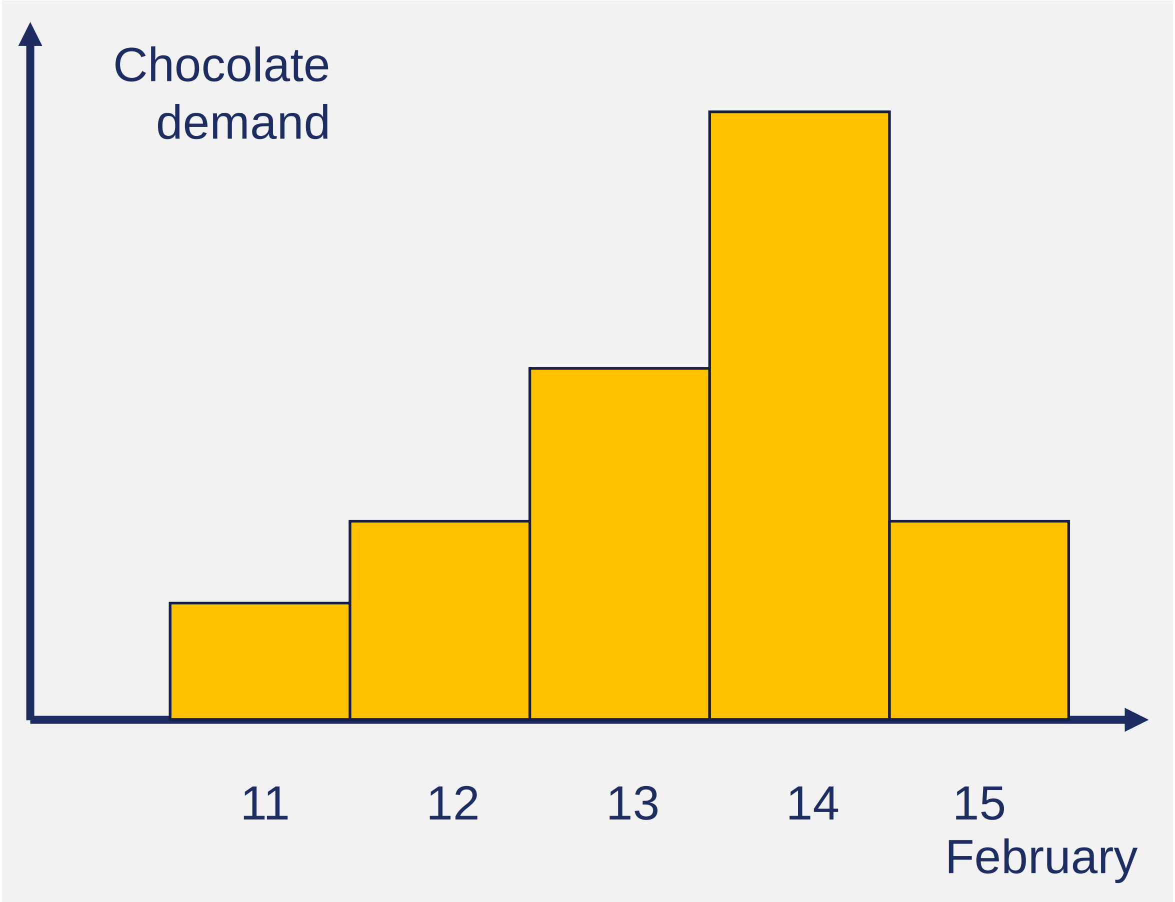
c(t)	Trend +
s(t)	Seasonality +
h(t)	Holiday effects +
x(t)	External regressors +
e	error

Visualization



Dynamic Holidays – Valentine‘s example

Visualization



Facebook Prophet

You state Valentine‘s as a key event and specify how many days before/after to quantify

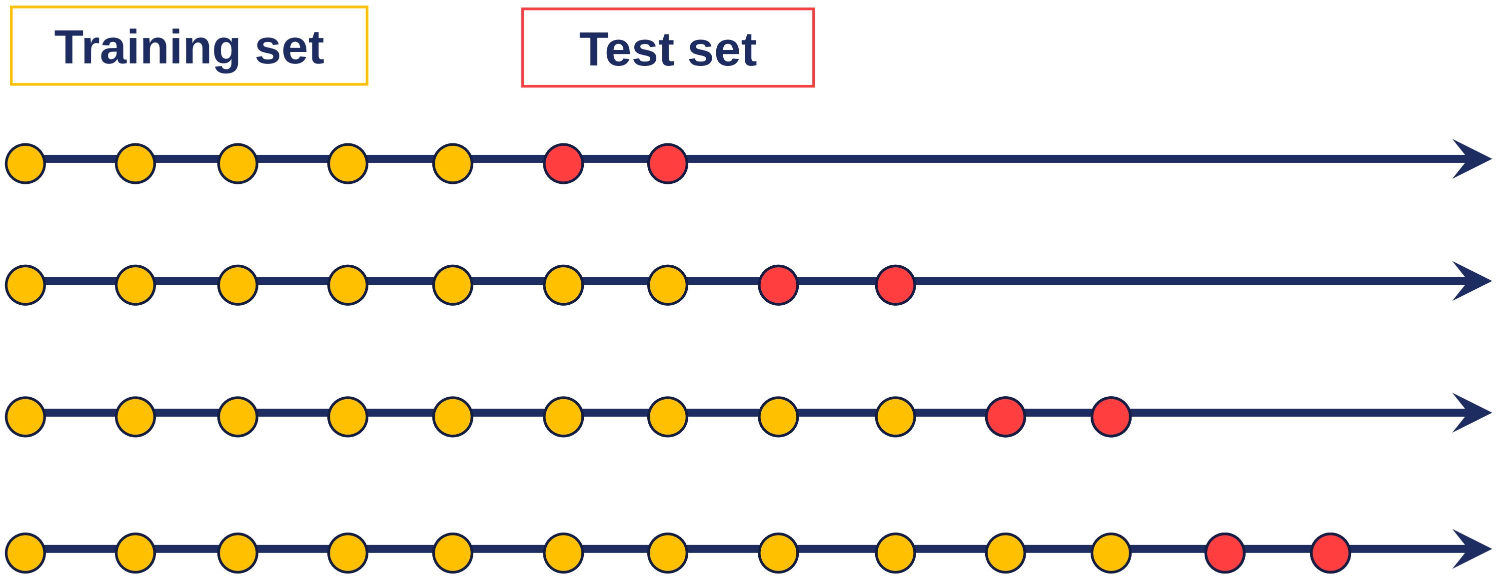
Other models:

You must create dummy variables for each day, if you believe they have different impacts

Facebook Prophet Model

Component	Description
Growth	Linear or Logistic
Holidays	Dataframe that we prepared
Seasonality	Yearly, weekly or daily. True or False
Seasonality_mode	Multiplicative or additive
Seasonality_prior_scale	Strength of the seasonality
Holiday_prior_scale	Larger values allow the model to fit larger seasonal fluctuations
Changepoint_prior_scale	flexibility of the automatic changepoint selection

Cross Validation



Key Idea

Repeating the assessment of our model reinforces its evaluation

Parameters to tune

Component	Description
Seasonality_prior_scale	Strength of the seasonality
Holiday_prior_scale	Larger values allow the model to fit larger seasonal fluctuations
Changepoint_prior_scale	flexibility of the automatic changepoint selection

Pros and Cons

+ Pros

1. Flexible
2. Built-in Cross Validation
3. Dynamics Events
4. Allows regressors

− Cons

1. Complex programming
2. Can need intense optimization
3. Not good with short-term dynamics
4. Not good with non-linear regressors

Challenge

Description

Demand for Shelter in New York City

1. Rename Dependent and Time Variable to y and ds
2. Declare Easter and Thanksgiving as holidays. Combine them. Use pd.concat
3. Create Prophet model. Christmas is a regressor
4. Cross Validation. Horizon = 31, initial = 2400. Assess via MAE
5. Create Parameter Grid for Tuning
6. Perform Hyperparameter Tuning. Use MAE as the KPI to optimize. Gather Results

Dataset: Open Data NYC initiative

Facebook Prophet
+
XGBoost

Prophet and XGBoost step by step

Tuned Prophet Model

Borrow Seasonality, Trend and other Variables

Prepare XGBoost Matrices

Set Parameters

Run XGBoost

Assess Model

XGBoost is a state-of-art Machine Learning Algorithm

Description

1. Stands for Extreme Gradient Boosting
2. Can be contructed with a tree based algorithm or linear (worse results)
3. It is an emsemble algorithm
4. Each new model is built upon the precedent one -> continuous improvement
5. Can be used for both Regression and Classification

XGBoost gives different weights depending on how difficult it is to predict

First Tree

	Outcome		Predictor	Weight
✓	1	←	X	25%
✓	0	←	X	25%
✗	0	←	X	25%
✗	1	←	X	25%

Second Tree

	Outcome		Predictor	Weight
✗	1	←	X	20%
✓	0	←	X	20%
✗	0	←	X	30%
✓	1	←	X	30%

Third Tree

	Outcome		Predictor	Weight
✗	1	←	X	23%
✓	0	←	X	15%
✓	0	←	X	35%
✓	1	←	X	27%

XGBoost looks at parts of the observations at a time

First Tree

	Outcome		Predictor	Weight
✓	1	←	X1	25%
✓	0	←	X2	25%
✗	1	←	X4	25%

Second Tree

	Outcome		Predictor	Weight
✗	1	←	X1	20%
✓	0	←	X2	20%
✗	0	←	X3	30%

Third Tree

	Outcome		Predictor	Weight
✗	1	←	X1	23%
✓	0	←	X3	35%
✓	1	←	X4	27%

Key Idea

XGBoost only looks at a fraction of the observation at the time

Observations that are more difficult to predict are given a bigger weight

The logic is similar for Regression-based tasks

First Tree

Error	Outcome		Predictor	Weight
- 5	15	←	X1	33%
2	22	←	X2	33%
4	34	←	X4	33%

Second tree

Error	Outcome		Predictor	Weight
- 1	19	←	X1	40%
-1	25	←	X2	30%
3	35	←	X4	35%

XGBoost also gives different weights to different predictors

First Tree

Error	Outcome	X1	X2	X3	Weight
-5	15	50%	50%	[redacted]	33%
2	22				33%
4	34				33%

Second Tree

Error	Outcome	X1	X2	X3	Weight
-1	19	50%	[redacted]	50%	40%
-1	25				30%
3	35				35%

Third Tree

Error	Outcome	X1	X2	X3	Weight
1	21	[redacted]	40%	60%	35%
0	24				30%
2	36				40%

Key Idea

Predictors also have different weights if they yield different model results

XGBoost quirks

Description

Which?

NA:
Unlike other regression models, XGBoost treats NA's as information

Non-linearity:
XGBoost is excellent dealing with non-linearity relationship between the dependent and the independent variables.

Which parameters are there?

Parameter	Description
Minimum Child weight	Relates to the sum of the weights of each observation. Low values can mean that maybe not a lot of observations are in the round
ETA	Learning Rate. How fast do you want the model to learn?
Max depth	How big should the tree be? Bigger trees go into more detail
Gamma	How fast should the tree be split?
Subsample	Share of observations in each tree?
Colsample by tree	How much of the tree should be analysed per round?
Number of rounds	How many times do we want the analysis to be run?

Prophet + XGBoost Pros and Cons

Pros (+)		Cons (−)	
Flexible	1	1	Complex programming
Great with Regressors	2	2	Can need intense optimization
Decent with short-term dynamics	3		

Challenge

Description

Demand for Shelter in New York City

1. Create future DF with test set length. Add regressor
2. Forecast and create a DF with: trend, weekly, yearly, holidays, multiplicative_terms
3. Concatenate with df. Drop Easter and Thanksgiving
4. Generate Training and Test Set. Isolate X and Y and form XGBoost Matrices
5. Set Parameters and Create XGBoost model
6. Predict. Visualize Test Set and Predictions. Assess model using MAPE

Source: UK Government

Ensemble

Ensemble mechanism

Example

Date	Y	Holt-Winters	SARIMAX	TBATS	TFP	Prophet	XGBoost	Ensemble
t	50	48	49	51	50.5	53	51	50.5

Key Idea

Ensemble is an average of models. The goal models have flaws, but if you group all of them, then some models will average out the error

To consider:

- Dynamic average. You give more weight to models that have less errors, punish the ones that are not performing as well.

Why Ensemble

Deep dives

The research on combining forecasts to achieve better accuracy is extensive, persuasive, and consistent.

Essam Mahmoud,
“Accuracy in Forecasting: A Survey,” *Journal of Forecasting*, April–June 1984, p. 139;

Spyros Makridakis and Robert L. Winkler,
“Averages of Forecasts: Some Empirical Results,” Management Science, September 1983, p. 987

Victor Zarnowitz,
“The Accuracy of Individual and Group Forecasts from Business Outlook Surveys,” Journal of Forecasting, January–March 1984, p. 10.

Pros and Cons

+

−

+	−
1 Accuracy	1 Lack of visibility
	2 Preparation