

Gaussian Processes Part 2

Max Kochurov

Moscow State University

Lecture 5



① Introduction

② GP approach

- Introduction
- Non-periodic part
- Periodic part
- The Model
- Fourier Features



Time Series, Classical Approach

If data has seasonality, you usually use **STL** decomposition. However,

- Parameters are not interpretable, only decomposition is available
- No uncertainty estimates
- Quite strict on input values
- Significantly less flexible in modelling

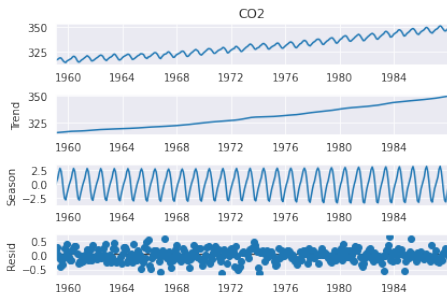


Figure: STL decomposition for CO2 data, Statsmodels



GP decomposition

A Gaussian process can handle a complicated set of assumptions in addition to what STL provides

- Granular seasonality (year + quarter + month + week)
- Changepoint models
- Flexible likelihood Function
- Panel regression models
- Missing values



Typical Model

Typical model is additive

$$x_t \sim \underbrace{g(t)}_{\text{non-periodic}} + \underbrace{s(t)}_{\text{periodic}} + \underbrace{h(t)}_{\text{holidays}}$$

Reference

See more in **Prophet** preprint [1]. Every time series model is unique



Reminder

$$x \in \mathbb{R}^n, y \in \mathbb{R}$$

$$Y \sim \mathcal{GP}(m(x), k(x, x'))$$



Reminder

$$x \in \mathbb{R}^n, y \in \mathbb{R}$$

$$Y \sim \mathcal{GP}(m(x), k(x, x'))$$

$$\begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix} \sim \mathcal{N} \left(\begin{bmatrix} m(x_1) \\ \vdots \\ m(x_N) \end{bmatrix}, \begin{bmatrix} k(x_1, x_1) & \dots & k(x_1, x_N) \\ \vdots & \ddots & \vdots \\ k(x_N, x_1) & \dots & k(x_N, x_N) \end{bmatrix} \right)$$

- ① \mathcal{GP} Gaussian Process - simply, a normal distribution with special mean $m(x)$ and covariance $k(x, x')$



Reminder

$$x \in \mathbb{R}^n, y \in \mathbb{R}$$

$$Y \sim \mathcal{GP}(m(x), k(x, x'))$$

$$\begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix} \sim \mathcal{N} \left(\begin{bmatrix} m(x_1) \\ \vdots \\ m(x_N) \end{bmatrix}, \begin{bmatrix} k(x_1, x_1) & \dots & k(x_1, x_N) \\ \vdots & \ddots & \vdots \\ k(x_N, x_1) & \dots & k(x_N, x_N) \end{bmatrix} \right)$$

- ① \mathcal{GP} Gaussian Process - simply, a normal distribution with special mean $m(x)$ and covariance $k(x, x')$
- ② $m(x)$ - mean function, e.g.
 - Linear regression $m(x) = x^\top \beta$
 - Simply Constant or Zero $m(x) = c$
 - Other custom functions $m(x) = \sin(x)$



Reminder

$$x \in \mathbb{R}^n, y \in \mathbb{R}$$

$$Y \sim \mathcal{GP}(m(x), k(x, x'))$$

$$\begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix} \sim \mathcal{N} \left(\begin{bmatrix} m(x_1) \\ \vdots \\ m(x_N) \end{bmatrix}, \begin{bmatrix} k(x_1, x_1) & \dots & k(x_1, x_N) \\ \vdots & \ddots & \vdots \\ k(x_N, x_1) & \dots & k(x_N, x_N) \end{bmatrix} \right)$$

- ① \mathcal{GP} Gaussian Process - simply, a normal distribution with special mean $m(x)$ and covariance $k(x, x')$
- ② $m(x)$ - mean function, e.g.
 - Linear regression $m(x) = x^\top \beta$
 - Simply Constant or Zero $m(x) = c$
 - Other custom functions $m(x) = \sin(x)$
- ③ $k(x, x')$ - kernel function, simply - measure of similarity for x and x'
 - $[K]_{ij} = k(x_i, x_j)$ is an SPD matrix



Non-periodic Part (mean function)

- Growth models
- Linear trend models
- Changepoint models



Non-periodic Part (mean function)

- Growth models
- Linear trend models
- Changepoint models

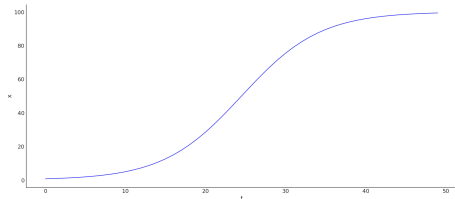


Figure: Growth Model

$$x = \frac{c}{1 + \exp(-k(t - m))}$$



Non-periodic Part (mean function)

- Growth models
- **Linear trend models**
- Changepoint models

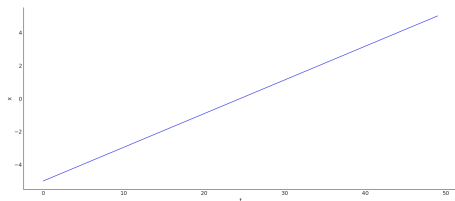


Figure: Linear Trend Model

$$x = \frac{c}{1 + \exp(-k(t - m))}$$



Non-periodic Part (mean function)

- Growth models
- Linear trend models
- **Changepoint models**

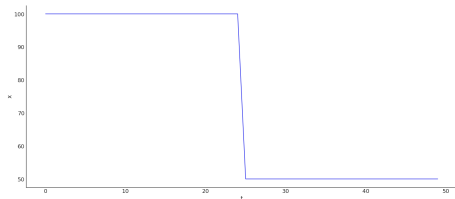


Figure: Changepoint Model

$$x = \begin{cases} c_1, & t < m \\ c_2, & t \geq m \end{cases}$$



Non-periodic Part (mean function)

- Growth models
- Linear trend models
- Changepoint models

Extentions

Extensions are possible, e.g. time dependent saturation in the growth model. See in [1]

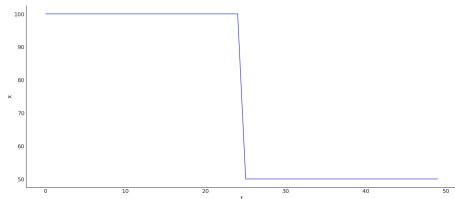


Figure: Changepoint Model

$$x = \begin{cases} c_1, & t < m \\ c_2, & t \geq m \end{cases}$$



Holidays

$$h(t) = \text{is-holiday}(t)$$

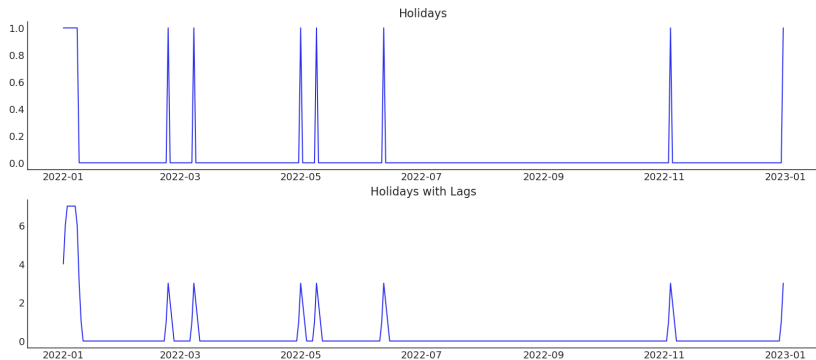


Figure: Holiday features



Periodic part (cov function)

Granularities are important here.
Multiple Periodic kernels can be used.

- Yearly
- Quarterly
- Monthly
- Weekly

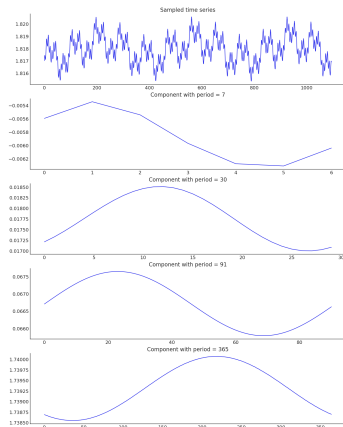


Figure: Seasonal Decomposition



Lengthscales for Periodic Part

Hyperparameters

Common sense driven lengthscale choice

- Week - couple of days make a change ($l_s \approx 3$)
- Month - week makes sense for a dramatic change ($l_s \approx 7$)
- Quarter - month makes sense for a dramatic change ($l_s \approx 30$)
- Year - quarter makes sense for a dramatic change ($l_s \approx 90$)



Lengthscales for Periodic Part

Hyperparameters

Common sense driven lenthscale choice

- Week - couple of days make a change ($l_s \approx 3$)
- Month - week makes sense for a dramatic change ($l_s \approx 7$)
- Quarter - month makes sense for a dramatic change ($l_s \approx 30$)
- Year - quarter makes sense for a dramatic change ($l_s \approx 90$)

In practice

Everybody is using Fourier features as a replacement for Periodic Kernel



Putting All Together

$$m(t) = \underbrace{g(t)}_{\text{non-periodic}} + \underbrace{h(t)}_{\text{holidays}}$$

$$\begin{aligned} k(*, *) = & \text{Periodic}(p = 365, l = 90) \\ & + \text{Periodic}(p = 90, l = 30) \\ & + \text{Periodic}(p = 30, l = 7) \\ & + \text{Periodic}(p = 7, l = 3) \end{aligned}$$



Putting All Together

$$m(t) = \underbrace{g(t)}_{\text{non-periodic}} + \underbrace{h(t)}_{\text{holidays}}$$

$$\begin{aligned} k(*, *) = & \alpha_{365} \text{Periodic}(p = 365, l = 90) \\ & + \alpha_{90} \text{Periodic}(p = 90, l = 30) \\ & + \alpha_{30} \text{Periodic}(p = 30, l = 7) \\ & + \alpha_7 \text{Periodic}(p = 7, l = 3) \end{aligned}$$

Missing Parts

- 1 Weights for periodic components



Putting All Together

$$m(t) = \underbrace{g(t)}_{\text{non-periodic}} + \underbrace{h(t)}_{\text{holidays}}$$

$$\begin{aligned} k(*, *) = & \alpha_{365} \text{Periodic}(p = 365, l = 90) \\ & + \alpha_{90} \text{Periodic}(p = 90, l = 30) \\ & + \alpha_{30} \text{Periodic}(p = 30, l = 7) \\ & + \alpha_7 \text{Periodic}(p = 7, l = 3) \\ & + \beta \text{ExpQuad}(l = 90) + \text{WhiteNoise}(\gamma) \end{aligned}$$

Missing Parts

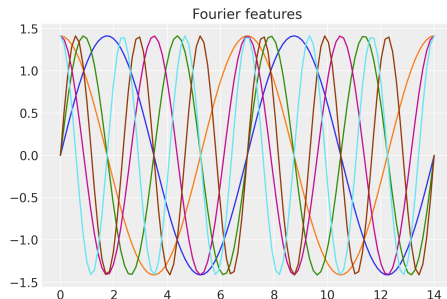
- 1 Weights for periodic components
- 2 Trend violations from $g(t)$



More Efficient Periodic

Having at least one Periodic kernel in a large time series prevents from optimizations.

- Fourier features can be added as regressors
- This allows reasonable periodicity to be present in the model





Choosing Features

Every regular period has optimal number of fourier components (aka order)

- Weekly: 3
- Monthly: 10
- Yearly: 5

```
from collections import namedtuple
from enum import Enum

Season = namedtuple("Season", "period,order")

class Daily(Season, Enum):
    Week = 7.0, 3
    Year = 365.25, 5
    Month = 365.25 / 12, 10

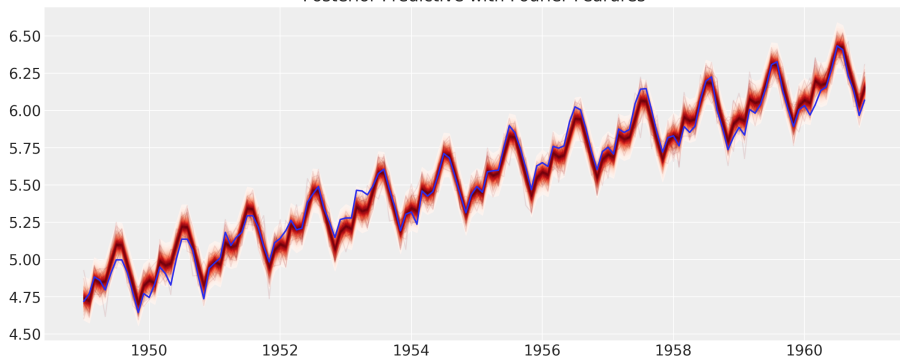
class Monthly(Season, Enum):
    Year = 12, 5
    Month = 1, 10
```



Applications of Fourier Features

- E.g. Prophet uses Fourier Features (FF)
- It is VERY Fast, drop-in replacement for Periodic GP
- Proven to be useful for Bayesian models

Posterior Predictive with Fourier Features





Integrating into the model

```
alpha = pm.Normal("alpha", 0, sigma, shape=Monthly.Year.order * 2)
features = fourier_series(months, season=Monthly.Year)
seasonality = features @ alpha

expectation = bias + trend + seasonality
```



References I



T. SJ and L. B.
Forecasting at scale.
2017.