放電加工機の時系列解析（中間まとめ）

1.1 はじめに

放電加工機の時系列解析に（その１）から（その５）と取り組みました。

今回は、そのレビューです。以下の 2 点について、まとめます。

・データの定常性について、検討します。原系列（元データ）と原系列を変換したもの 3 点を、グラフと ADF 検定した結果とで比較します。

・今まで、自己回帰モデル、状態空間モデル、非線形非ガウスモデルに、取組みました。その比較をまとめます。

1.2 どのようなデータを扱ったか

放電加工機は、加工中に障害が発生すると自己診断して 3 段階のレベルに分け、レベルに応じたメッセージを表示します。内容は以下です。

・エラーメッセージ：続行不可能な障害が発生したときに表示し、動作を中断する。

・ハルトメッセージ：再開可能な障害が発生したときに表示し、一時停止する。

・コメントメッセージ：続行可能な障害で発生し、注意を促す。

また、メッセージの記録は、USB により CSV データとして取り出すことが可能です。

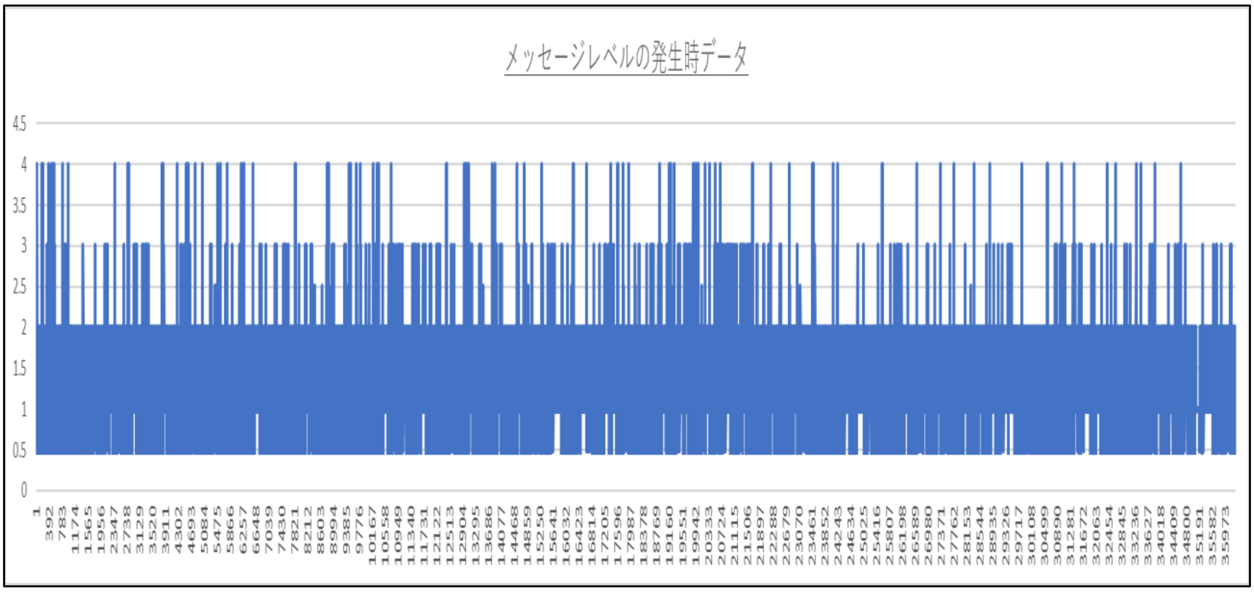
レベルに対応してデータ処理できるように、3 段階のレベルについて障害のレベルが大きいと、点数は大きくなるように点数を設けました。

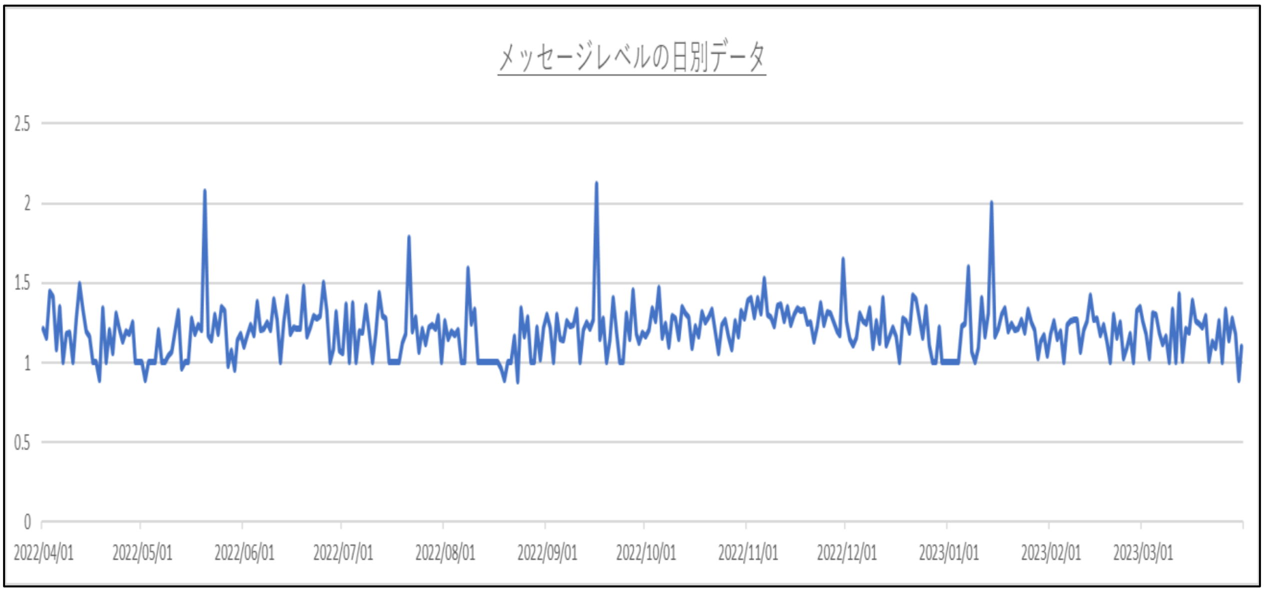
これらは、メッセージが発生した時の時間毎のデータで、点過程データと呼ばれるもので、時系列データではありません。故に、一日分の平均値を取って時系列データとして取り出すようしました。本データを以下に示します。2022 年 4 月　から　2023 年 3 月までのデータです。

上段が発生時毎のデータ（点過程データ）で、1 年で 37000 回発生してます。

1 点の発生は、一定時間の間隔ではありません。

下段は点過程データを、一日毎に平均して、平均したデータです。点の間隔は 1 日毎ですので、時系列データと言えます。


メッセージレベルの発生時データ
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
1 392 783 1174 1565 1956 2347 2738 3129 3520 3911 4302 4693 5084 5475 5866 6257 6648 7039 7430 7821 8212 8603 8994 9385 9776 10167 10558 10949 11340 11731 12122 12513 12904 13295 13686 14077 14468 14859 15250 15641 16032 16423 16814 17205 17596 17987 18378 18769 19160 19551 19942 20333 20724 21115 21506 21897 22288 22679 23070 23461 23852 24243 24634 25025 25416 25807 26198 26589 26980 27371 27762 28153 28544 28935 29326 29717 30108 30499 30890 31281 31672 32063 32454 32845 33236 33627 34018 34409 34800 35191 35582 35973


メッセージレベルの日別データ
2.5
2
1.5
1
0.5
0
2022/04/01 2022/05/01 2022/06/01 2022/07/01 2022/08/01 2022/09/01 2022/10/01 2022/11/01 2022/12/01 2023/01/01 2023/02/01 2023/03/01

1.3　プログラムについて

・ADF 検定　→　adfuller

Python の　statsmodels　というライブラリー内の以下の関数です。

詳細は以下で確認ください。

https://www.statsmodels.org/stable/api.html#statsmodels-api

・推定はカルマンフィルターを用いており、

Pykalman ライブラリで計算してます。

本報告書にはプログラムは記載していません。

～参考文献、WEB～

時系列解析　-自己回帰モデル･状態空間モデル･異常検知-　島田直希　著　共立出版（株）

時系列データ分析

https://qiita.com/tk-tatsuro/items/16ce74fc954b5a58df00

Python で時系列解析・超入門（その 1）

https://www.salesanalytics.co.jp/datascience/datascience085/

2　データの定常性　について

2.1 前処理（変換）の各種方法

　時系列データが定常性が無いと判断された場合、データの関係性を正しく解析ができない為、定常性を持たせるための変換を行う必要があります。

定常性とは、期待値（平均値）と自己共分散（バラツキ）が時間を通じて一定であるということで、トレンドを持たないということです。

　各種の変換方法を述べます。

　1.差分変換

差分変換とは、ある時点 t のときの値とその直前の時点 t−1 のときの値の差分 $\delta t=yt-yt-1$ を計算し、その差を新しいデータとする手法です。1 時点離れたデータとの差を取る手法です。その結果を差分系列、または階差系列と言います。

　2.対数変換

対数変換とは、変動が著しく大きな時系列データに対し、対数値変換を施すことで原系列の傾向を保持したまま、値を小さく変換する手法です。

変動の分散を一様にし、複雑な時系列でも変数変換で分析が簡単になることがあります。

対数変換された時系列データは対数系列と呼ばます。数式では $\log(yt)$ と表します。

　3.対数差分変換

対数差分変換とは、対数変換と差分変換の両方の変換を施す手法です。

時系列データが変化率や成長率などである場合、対数差分 $\Delta\log(yt)=\log(yt/yt-1)$ を計算して解析することがあります。

2.2　比較の結果

　原系列（元データ）と　3 種の変換データをグラフで比較し、

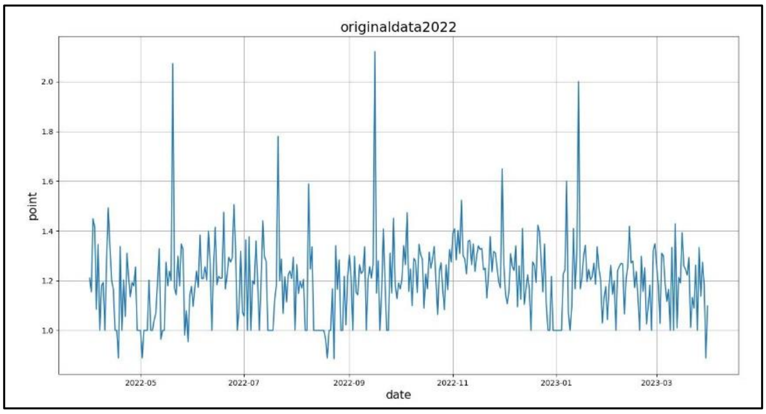
次に ADF 検定という、定常性を判定する関数があるので、結果を比較します。

まずグラフについての考察ですが

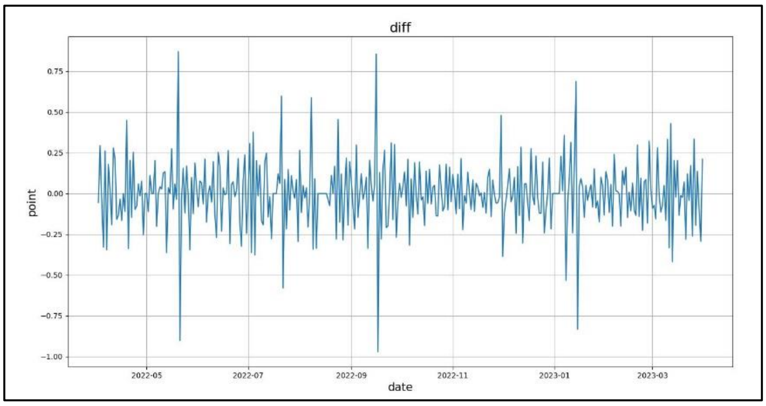
平均値が一定であるもの、ばらつきが少ないものが、適切ですが、

4 個のグラフ間では、あまり差がなく、差分変換したものが、平均値は一定のように見えます。

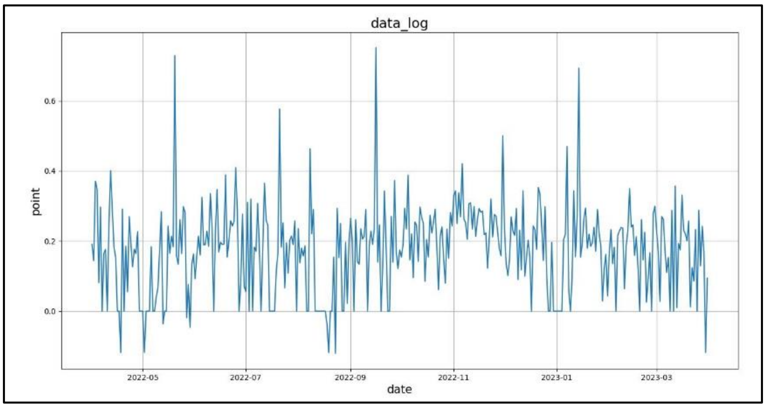
原系列



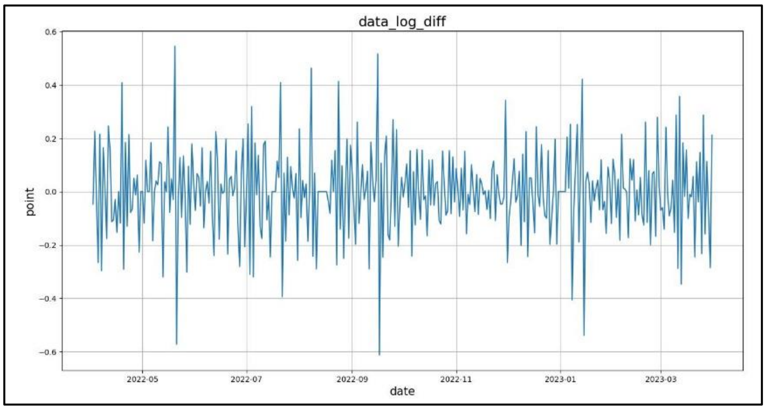
差分変換



対数変換



対数差分変換



次に、下の表が ADF 検定の結果です。

定常性があると判定できるのは

・検定統計量が、棄却値５%の値より小さいこと

・p 値が　0.05　以下であること

とされており、原系列も他の点の変換例も定常性ありと言えます。

	原系列	差分	log	log 差分
平均	1.200760631	-0.0003071	0.174702661	-0.000266268
分散	0.025515416	0.043376275	0.016170783	0.026269714
adf　検定統計量	-10.62998848	-8.840497998	-10.04156452	-8.169681629
p	5.24E-19	1.67E-14	1.49E-17	8.69E-13
棄却値　1%	3.448493651	3.449011444	3.448493651	-3.44911857
棄却値　5%	-2.869535228	-2.869762707	-2.869535228	-2.869809765
棄却値　10%	2.571029334	2.571150616	2.571029334	-2.571175706

先回の検討でも　原系列でも定常性は有りの判定でしたが、

同じ結果になりました。

3　取り組んだモデルの比較

自己回帰モデル、状態空間モデル、非線形非ガウスモデルについて、取り組んだので、比較してみます。

ざっくり言うと、時系列データには、以下の３回のアプローチで行います。

・統計モデルの「型」を決めます。　（人がやります）

・学習データを用いて、推定します。

（推定法には、最尤法、カルマンフィルタ等があります）

・予測します。（予測の評価として、テストデータと比較します）

3.1 自己回帰モデル

AR モデル、SARIMA モデル　に取り組みました。

・型について

AR モデルは以下で表します。

$y_n \;=\; \sum_{i=1}^{m} ai\, y_{n-1} \;+\; v_n$　　m は次数　ai は係数　です。

SARIMA モデルの型は説明が難しいです。

AR モデル　=　自己回帰モデル　　（時間の変化に規則的に値が変化）

MA モデル　=　移動平均モデル　（時間の変化に不規則的に値が変化）

ARIMA モデル　=　ARMA の差分モデル　（差分をデータとします）

SARIMA モデル　=　周期的変動（季節成分）を加えたモデル

という　モデルの構成で表します。

SARIMA モデルを数式で表すると、以下ですが、分かりにくいです。

ARIMA モデルまでは、加法ですが、ここは乗法となります。

Δ^d　は　ARIMA 過程の差分で、　Δ_s^D　は　季節階差の差分です。

$$\Phi(B)\Phi s(B)\Delta^d \Delta_s^D y_t \;=\; \theta(B)\theta s(B)\ \varepsilon_t$$

$$\Phi(B) = 1 - \sum_{i=1}^{p} \Phi i\, B^i \qquad \Phi s(B) = 1 - \sum_{i=1}^{P} \Phi si\, B^{si}$$

$$\theta(B) = 1 + \sum_{i=1}^{q} \theta i\, B^i \qquad \theta s(B) = 1 + \sum_{i=1}^{Q} \theta si\, B^{si}$$

$B^n \;=\; y_{t-n} \;/\; y_t$ (バックシフト演算子)

詳細は、参考文献で確認ください。

・推定について、

次数は予め設定し（AIC で判定することもできますが、経験的に差はそれほどありません）、係数は最尤法（最小二乗法）で求めます。

・予測について、

季節成分を加味した　SARIMA モデルが、優れます。

特徴的なこととして、自己回帰モデルにおいては、学習データの推定時において、実データと推定データの乖離があります。実データは、ある確率分布に従う真のデータの確率変数の実現値ですので、推定データは、確率分布の期待値（平均値）と考えれば、当然かもしれません、状態空間モデルのカルマンフィルタは逐次推定で、推定値は学習データから外れないので、両者間の差を感じます。

3.2 状態空間モデル

トレンド傾向、季節傾向、AR 傾向を加味したモデルが良好な結果となります。

・型について、

時系列データを、トレンド成分（時間とともに単調に増加/減少する変動）と季節成分（同じ周期で規則的に繰り返される変動）と不規則成分との合計と考え、不規則成分は、AR モデルに準じると考えます。

数式で表すと以下です。y_n　は　観測データです。

$$y_n = t_n + s_n + r_n + w_n \qquad w_n \sim \mathrm{N}(0,\ \delta^2)$$

$$t_n = \sum_{i=1}^{k} c_i^{(k)}\ t_{n-i} + \boldsymbol{v_{n1}} \qquad v_{n1} \sim \mathrm{N}(0,\ \tau_1^2) \qquad \text{トレンド成分}$$

$$s_n = -\sum_{i=1}^{p-1} s_{n-i} + \boldsymbol{v_{n2}} \qquad v_{n2} \sim \mathrm{N}(0,\ \tau_2^2) \qquad \text{季節成分}$$

$$r_n = \sum_{i=1}^{q} \Phi_i\, r_{n-1} + \boldsymbol{v_{n3}} \qquad v_{n3} \sim \mathrm{N}(0,\ \tau_3^2) \qquad \text{AR 成分}$$

トレンド成分は、傾きは無いとして、次数を 1 としました。係数は 1 です。

季節成分は、月単位の変化を想定して、次数を 30 としていたのですが、

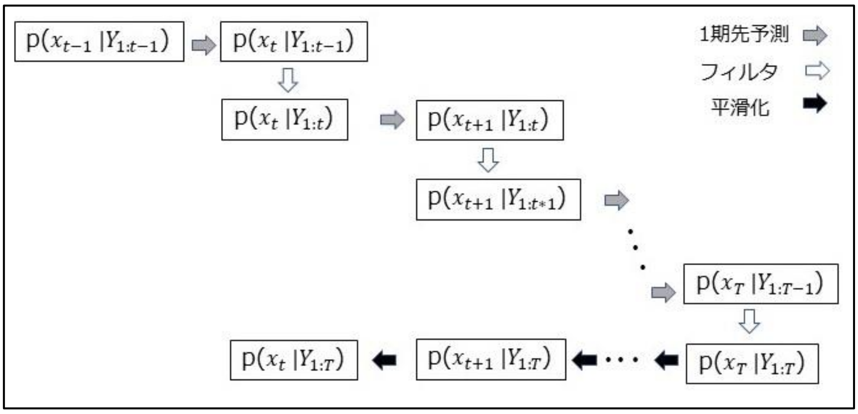
現状は月単位、週単位でもないので、グラフの周期から、次数は、 4 としました。

AR 成分は、次数を 2 としました。

・推定について、

カルマンフィルタで計算しますが、ライブラリは Pykalman　を用いています。

詳細は、先回に記載したので、説明図のみ、表示します。



状態空間モデルのカルマンフィルタは逐次推定で、学習データについては、推定値は、はずれません。

3.3 非ガウス非線形モデル

非ガウス型の近似には、色々な手法がありますが、今回は、粒子フィルターで、行います。

・型について

確率分布を点の集まりとして近似します。密度関数の高いところに多くの粒子が集中します。粒子の集中度合いを、実測値と比較して、フィルタリングとサンプリングを繰り返します。式は以下のような　簡単な数式です。

$x_t = F_t \ (x_{t-1}, v_t)$　　状態関数

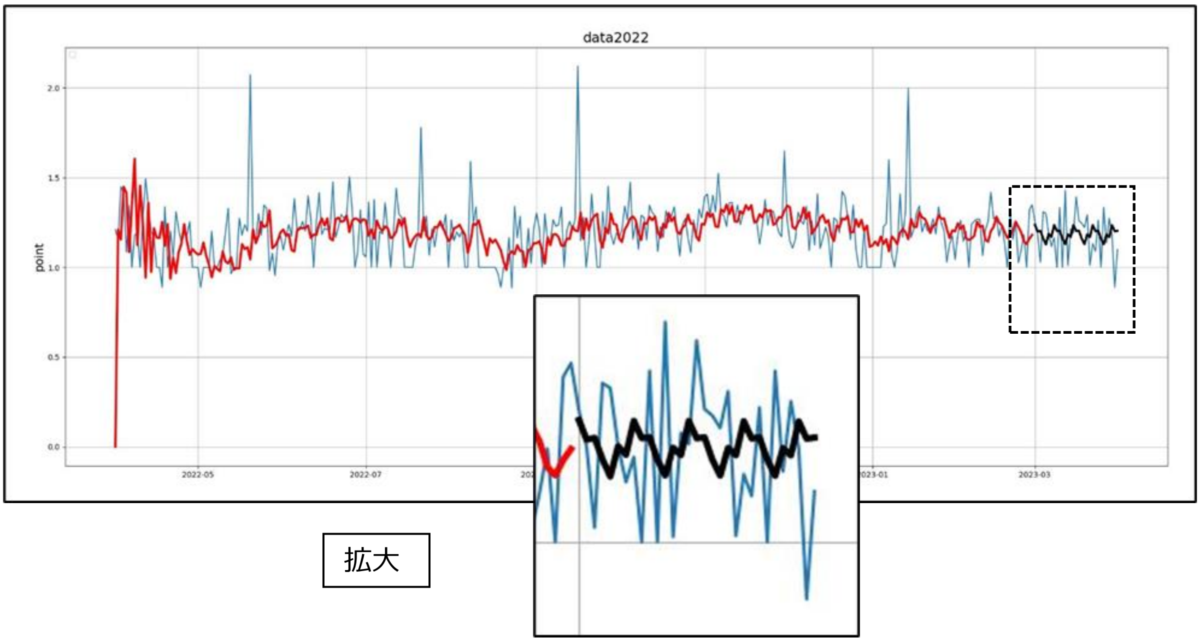
$y_t = H_t \ (x_t, w_t)$　　観測関数

・推定について

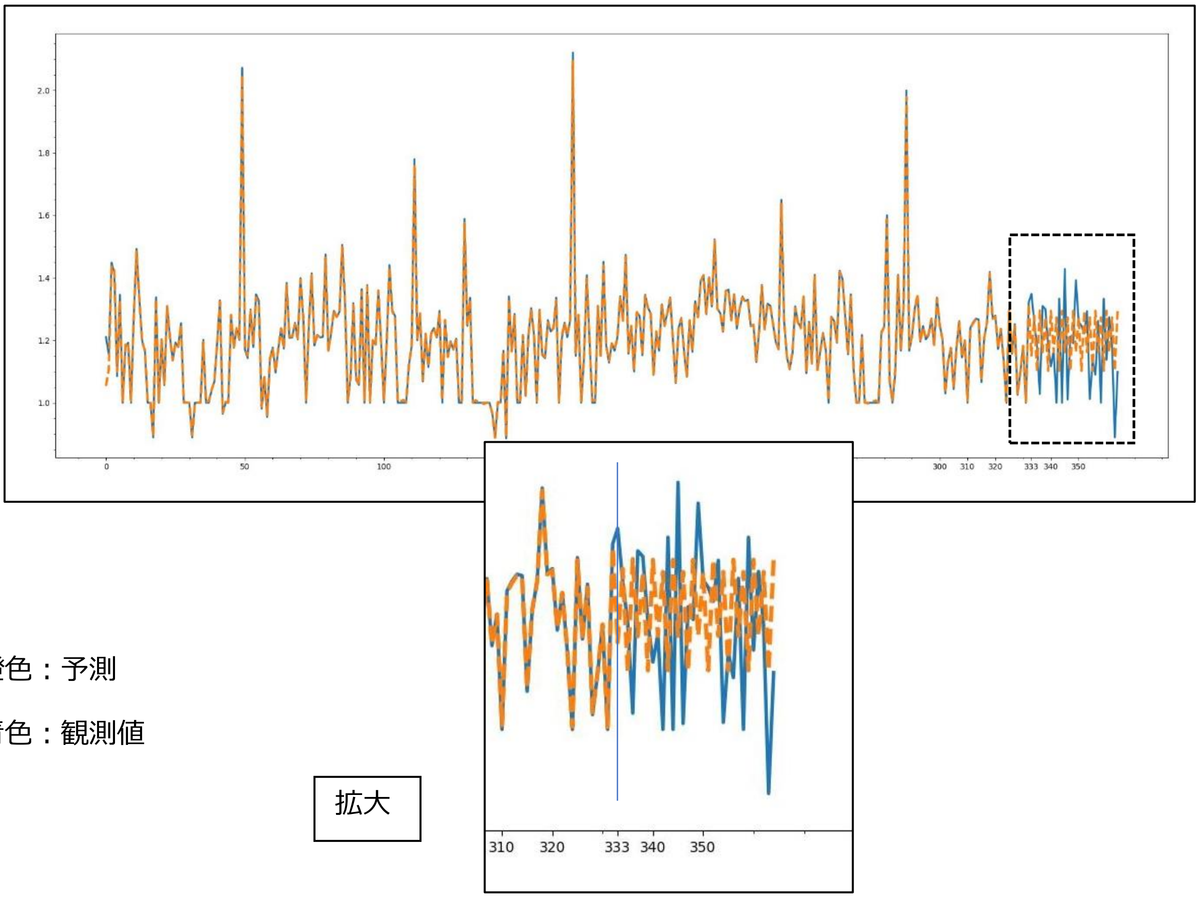
粒子フィルタにより行います。（カルマンフィルタに似てます）

4 結果

4.1　SARIMA モデルの　推定値　と　予測（黒線）　です。



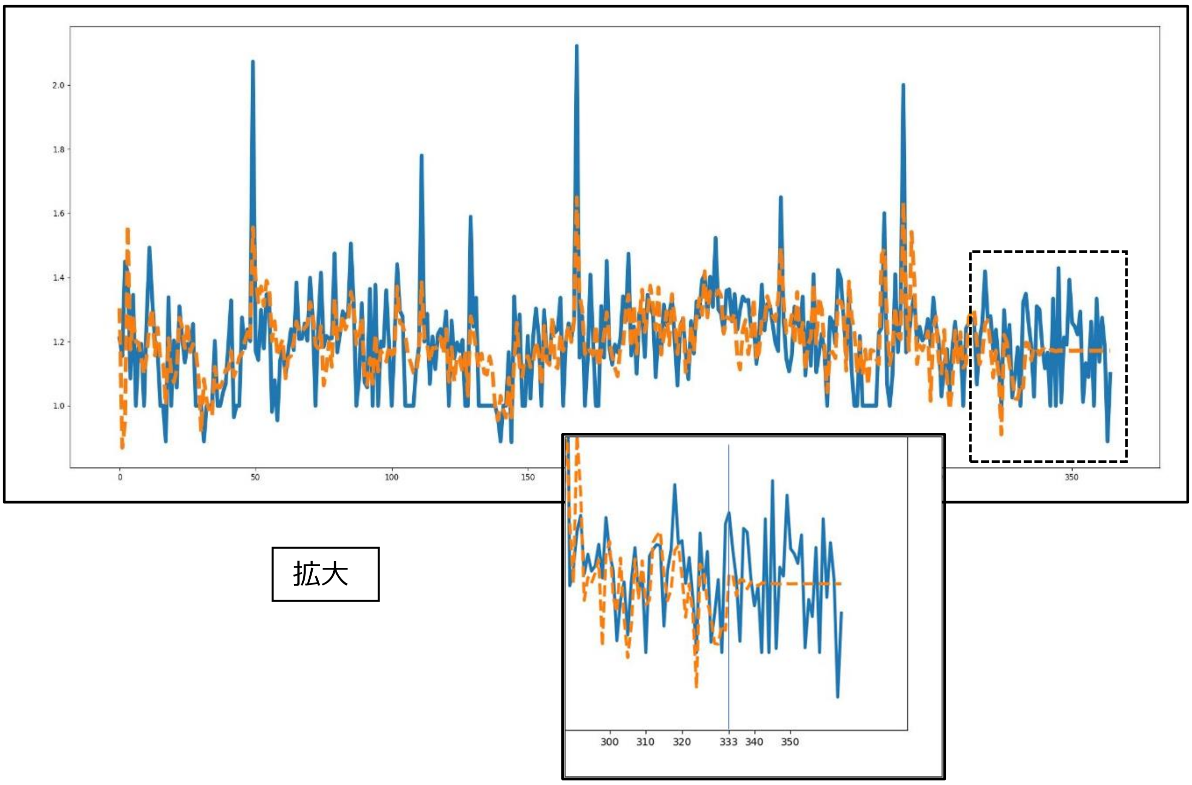
4.2　状態空間モデルの　推定値と予測です。



橙色：予測

青色：観測値

4.3　非ガウス非線形モデルの　推定値と予測です。



5 まとめ

・推定、予測について、状態空間モデルが優位でした。

・モデルは、季節成分（循環成分）を加えて、計算する方が優位でした。

・今後も、継続検討します。