

経済の時系列データ分析

1

1. 経済データとは
2. 移動平均
3. ボリンジャーバンド
4. ローソクチャート



経済・金融データとは

□ 経済データ

- 人口の推移、歳出・歳入、税収1
- 社会保障(医療、生活保護など)

□ 金融データ

- 株価、為替、預金、口座、資産など

□ 特徴

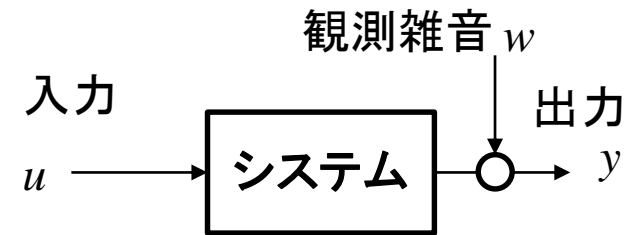
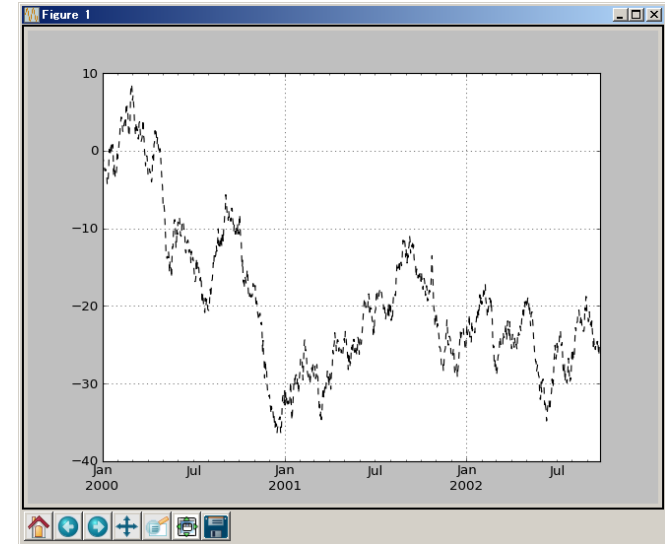
- 主に、整数である。観測雑音は混入しにくい
- 確率系の時系列データ(不規則過程)と見なされることが多い

□ 時系列 (Time Series)

- 時間的な変化を示すデータの系列
- 時間をインデックスに持つデータともいえる
- 系列は順序を変えてはだめ

□ 右は、代表的な時系列データ

- どのように評価, 分析する?
- 不規則過程に対し, 一致した推定・予測は望めない
- 観測値(出力 y)がある以上, 何らかの入力 u で駆動されていると考える。
- 右図において, 次で考え方が少し異なる。
 - u が確率変数, $w = 0$
 - u が確定信号, w が確率変数
 - u, w 共に確率変数



移動平均に基づく解析

□ 概要

- 移動平均には 幾つかの種類がある。
- いずれも，“位相が遅れ”，“なまる” ことは理論的に証明されている
- 平均操作個数が増えるほど，位相は より遅れる
- 単純移動平均を用い，，，，，**n**を変化させて 位相の遅れを見る
 - `pd.rolling_mean(ts, n).plot(style='k')`
- 株式の世界では，

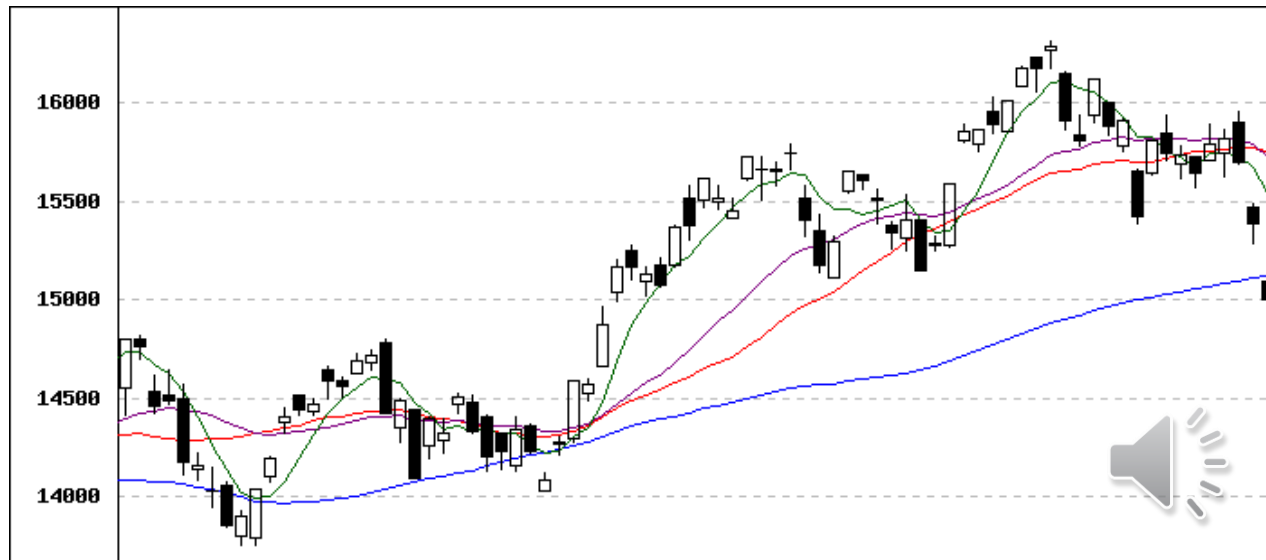
・ゴールドデックロス

・デッドクロス

を識別するのに用いている

眼で見て、上がりそう
か否かを予測する

ゴールドデックロス: 短期が
長期平均を上回る点
デッドクロス: 短期が長期
平均を下回る点



移動平均

□ 単純移動平均

- W は時刻における株価データ、は幾つかの名称があり、窓の幅、株価分析では25日平均と言ったときの日数、pandasではパラメータwindowに相当します。
- 例えば、 $W=5$ のとき、(1)式は次となります。

$$y_t = \frac{1}{W} \sum_{i=0}^{W-1} x_{t-i} \quad (1)$$

$$y_t = \frac{1}{5} (x_t + x_{t-1} + x_{t-2} + x_{t-3} + x_{t-4})$$

- 現時刻 t が進むにつれて、平均区間が移動することがわかるでしょう。

□ 指数移動平均

- 直近の x に強く影響を受けて、これより以前の x から受ける影響は少ない、という考え方

$$y_t = \alpha x_t + (1 - \alpha) y_{t-1} \quad (2)$$

- ここに、 α は平滑化定数で、 $0 \leq \alpha \leq 1$ の値を取ります。
- この式は指数的な重みを与える(天下りの説明)
- α の与え方の一つの指針

$$\alpha = \frac{2}{W+1}$$



(3)

移動平均

□ pandasを用いた移動平均と指数移動平均は次の関数を用いる

- `pandas.DataFrame.rolling(window=n).mean()`
- `pandas.DataFrame.ewm(span=n).mean()`
- ここに、ewmのパラメータspanは(3)式の α を指します。

□ これらの関数の使用法および使用例は次のようにして見ること

- 関数の使用法: <http://pandas.pydata.org/pandas-docs/stable/> のSearch窓から'rolling', 'ewm'を入れて検索してください。これらの関数のパラメータの説明があります。
- 使用例: <http://pandas.pydata.org/pandas-docs/stable/> の左欄にある“Computational tools”に使用の説明があります。



移動平均

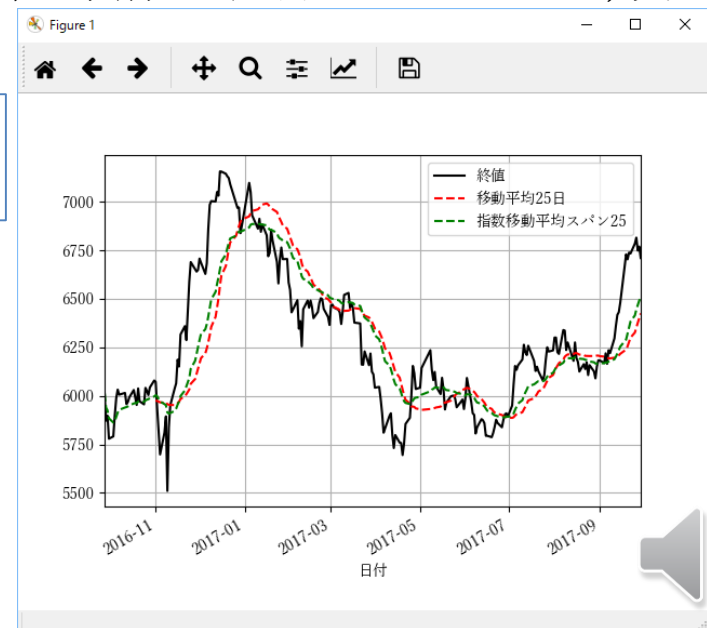
例

TSA_StockPrices

- windows = span = 25
- 平均操作の結果は、鋭い変化を鈍らせます。違う言葉で言うと、高周波成分を低減します。図2の実線と破線を比較して、実践波形の鋭い変化が鈍っていることが確認できます。
- 平均操作の結果、ゆっくりとした波形(低周波成分)の位相が遅れます。これは、元の黒い実線波形よりも破線が遅れて変化していることを意味します。
- 移動平均25日は、過去の24日分のデータと現在のデータの合計25点を用いて計算します。よって、開始日から24日は計算できませんから、移動平均25日は25日目から波形が始まります。指数移動平均は2日目から結果が現れます。
- 上記の2点は、デジタル信号処理論、システム制御工学論から証明されていることで、興味のある読者は、その分野の成書をお読みください。

次のデータは、銘柄コード7203(東証)のある時期の日毎の株価, utf-8 です。
 url = 'https://sites.google.com/site/datasciencehiro/datasets/Stock_7203.txt'
 この内容は次で、これを用います。

日付	始値	高値	安値	終値	出来高	売買代金
2017-09-29	8,703.00	8,729.00	8,698.00	8,710.00	5,879,200	39,438,999,100
2017-09-28	8,779.00	8,793.00	8,710.00	8,769.00	6,139,100	41,504,380,700
2017-09-27	8,752.00	8,778.00	8,735.00	8,748.00	5,465,900	36,916,080,800
2017-09-26	8,780.00	8,832.00	8,774.00	8,815.00	7,867,900	53,588,070,400
2017-09-25	8,781.00	8,794.00	8,762.00	8,784.00	5,874,600	39,827,130,000
2017-09-22	8,750.00	8,787.00	8,732.00	8,733.00	7,333,100	49,543,263,300
2017-09-21	8,774.00	8,788.00	8,737.00	8,738.00	7,048,000	47,667,280,600



ボリンジャーバンド

□ 相場におけるテクニカル指標の一つにボリンジャーバンド(J.A.Bollinger(米国, ファイナンシャルアナリスト)の考案)があります.

- ▶ 株価データが正規分布に従うランダム変数という仮定のもと, 次を考えます.

$$y_t = t \text{ 日の移動平均} \pm t \text{ 日の標準偏差} (\sigma_t) \times m \quad (4)$$

- ▶ この式は2本の時系列データを示し, この2本に挟まれた領域をボリンジャーバンドといいます.
- ▶ 次が一般に用いられる値であり, 正規分布の論では, 価格の変動がボリンジャーバンド内に収まる確率は

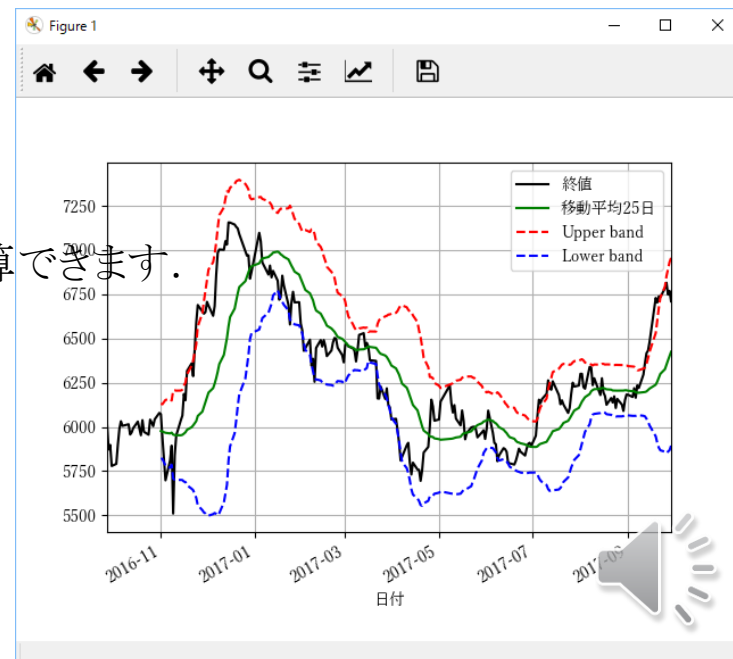
$\pm 1\sigma_t$ に収まる確率 = 68.26% ↓

$\pm 2\sigma_t$ に収まる確率 = 95.44% ↓

$\pm 3\sigma_t$ に収まる確率 = 99.73% ↓

- ▶ t 日の標準偏差は次のように, `std()`関数を用いて計算できます.

- `rolst = df[item].rolling(window=win).std()`



ローソク足チャート

□ 用語

- 陽線: 始値よりも終値の方が高い(上昇株)
- 陰線: 始値よりも終値の方が低い(下落株)
- その日の最安値・最高値は上下に突き出した細い線(通称:ヒゲ)で表します

□ 本チャートは始値・終値・最高値・最安値・上下落かの5つの情報を表現したものの

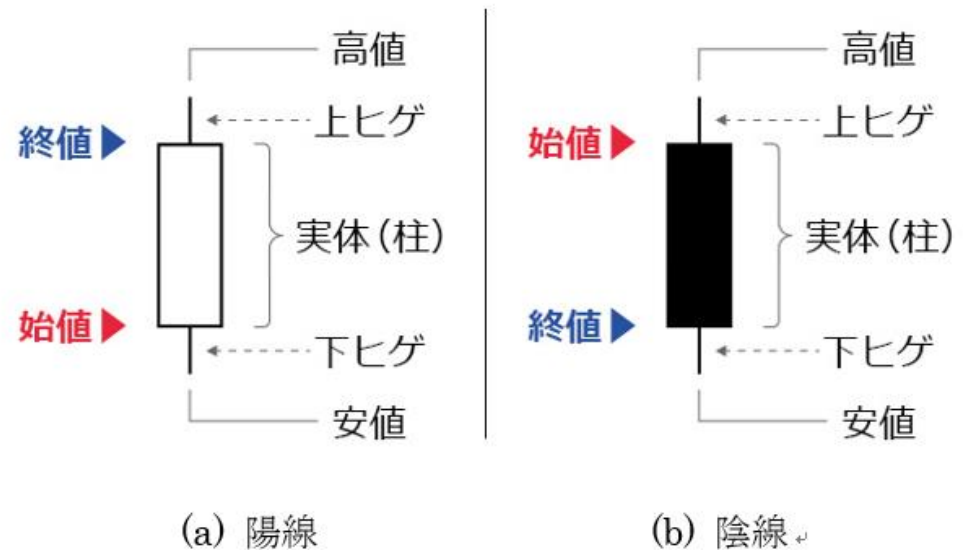


図4 ローソク足



ローソク足チャート

□ matplotlibは、ファイナンス用のツール

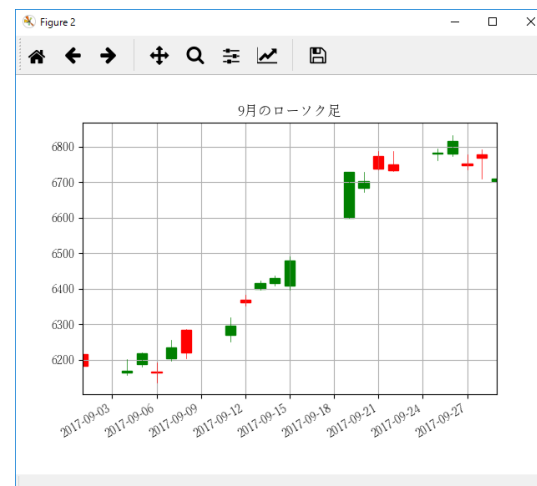
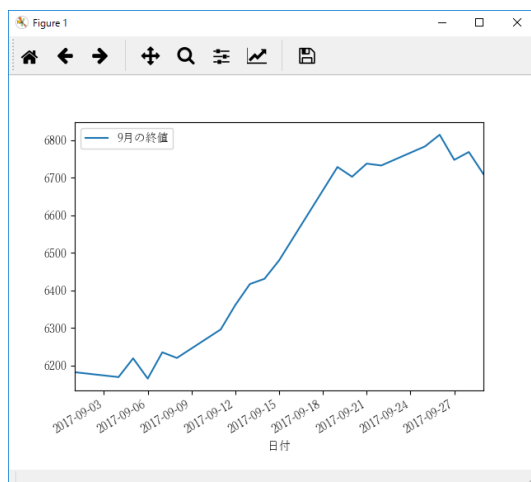
http://matplotlib.org/api/finance_api.html

を有しており、そのなかにローソク足チャートを表示させる次の関数があります。

- `candlestick2_ochl()`
- `candlestick2_ohlc()`
- `candlestick_ochl()`
- `candlestick_ohlc()`
- いずれも、始値 (`open`)、終値 (`close`)、最高値 (`high`)、最安値 (`low`) の4つを与えることは同じで、その与える順番 (`ochl`か`ohlc`)、または、4つ一括または別々に与えるかの違いのみです
- ここでは、`candlestick_ohlc()`を用いた例を示します。
 - `df0 = df.loc['20170901':'20170930', ['始値','高値','安値','終値']]`
- すなわち、2017年9月1日から2017年9月30日の1か月、かつ、4つのキーだけに対応する列を抽出します。
- 次に、このDataFrame構造をmatplotlibの`candlestick_ohlc()`に適合するようにデータ配列を設けます。
 - `xdate = [x.date() for x in df0.index]`
- これは、pandasのDataFrameのタイムスタンプをdatetimeに変換します。次に、
 - `ohlc = numpy.vstack((date2num(xdate), df0.values.T)).T`
 - `matplotlib.finance.candlestick_ohlc(ax, ohlc, width=0.7, colorup='g', colordown='r')`



ローソク足チャート



演習 株価

□ 日経平均, NYダウ, ナスダック(順)の株価ダウンロードサイト

- <http://finance.yahoo.com/q/hp?s=%5EN225>
- <http://finance.yahoo.com/q/hp?s=%5EDJI>
- <http://finance.yahoo.com/q/hp?s=%5EIXIC>

□ データの取得

- 次のページを参照
- CSV形式でデータをダウンロードできる。**エンコーディングが何かに注意!**

□ 課題

- 日経平均URLから、9か月～1.5年程度のデータを取得する。
- 対象は終値とし、この時系列と、移動平均のX日平均とY日平均を重ねてプロットする。XとYは各自で決める。
- レポートには、取得開始日と終り日 (**Close**)、グラフ、得られたグラフから、ゴールデンクロスとデッドクロスの日を明記。また、株の上下は、この2点に関して合っているか否かを評価してください。



Y ^N225 22,018.52 -339.91 X

← → ↻ 🏠 ⓘ https://finance.yahoo.com/quote/%5EN225/history?ltr=1 ☆ 📁 📧

Search

Sign in 📧

Finance Home Watchlists My Portfolio My Screeners Markets Industries Personal Finance Technology Originals

Time Period: Dec 01, 2017 - Mar 31, 2018 Show: Historical Prices Frequency: Daily Apply

↓ Download Data

Currency in JPY

Date	Open	High	Low	Close*	Adj Close**	Volume
May 30, 2018	0.00	0.00	0.00	22,018.52	22,018.52	-
May 29, 2018	22,431.95	22,439.10	22,240.39	22,358.43	22,358.43	59,800
May 28, 2018	22,488.95	22,547.67	22,410.91	22,481.09	22,481.09	52,900
May 25, 2018	22,380.22	22,509.36	22,318.15	22,450.79	22,450.79	70,400
May 24, 2018	22,621.29	22,644.69	22,366.60	22,437.01	22,437.01	85,600
May 23, 2018	22,868.79	22,949.73	22,649.85	22,689.74	22,689.74	75,000
May 22, 2018	23,025.95	23,031.67	22,952.87	22,960.34	22,960.34	59,400
May 21, 2018	22,937.58	23,050.39	22,935.31	23,002.37	23,002.37	59,100
May 18, 2018	22,907.20	22,954.19	22,867.30	22,930.36	22,930.36	65,300

↓ Download Data

至福のハワイは、空からはじまる。 HAWAIIAN AIRLINES

今すぐ予約

Yahoo Small Business

日付を指定

Historical Pricesを指定

Dailyを指定

全ての指定が終わったら、Applyをクリック

最後に、Download Dataをクリックすると、".csv"がダウンロードされる



データの形式

ダウンロードしたCSVファイルの形式

- ・UTF-8 (BOM無し)
- ・Dateはindexに指定, 他のラベル名は, 各自のスクリプトで引用できる

The screenshot shows an Excel spreadsheet titled '^N225.csv - Excel'. The ribbon includes 'ファイル', 'タッチ', 'ホーム', '挿入', 'ページレイアウト', '数式', 'データ', '校閲', '表示', 'アドイン', and 'サインイン'. The font settings are MS Pゴシック, size 11. The spreadsheet data is as follows:

	A	B	C	D	E	F	G	H
1	Date	Open	High	Low	Close	Adi Close	Volume	
2	2017/11/30	null	null	null	null	null	null	
3	2017/12/1	22916.93	22994.31	22675.07	22819.03	22819.03	89700	
4	2017/12/4	22843.53	22864.33	22693.99	22707.16	22707.16	68900	
5	2017/12/5	22595.33	22682.71	22522.28	22622.38	22622.38	75900	
6	2017/12/6	22525.38	22528.21	22119.21	22177.04	22177.04	97300	
7	2017/12/7	22317.15	22515.24	22317.15	22498.03	22498.03	79500	
8	2017/12/8	22627.95	22819.71	22625.41	22811.08	22811.08	106900	
9	2017/12/11	22894.3	22938.73	22787.4	22938.73	22938.73	69500	
10	2017/12/12	22926.41	22994.22	22824.89	22866.17	22866.17	86100	

Annotations in the image:

- A red box highlights the header row (row 1).
- A red box highlights row 2, with an arrow pointing to it from a text box that says "nullデータが存在する".
- An arrow points from a text box that says "ラベル名" to the header row.

UTF-8(BOM有)の方が互換性は高いが、上記のBOM無しとした理由は特にはない。



nullデータの扱い

先のCSVファイルには, nullデータが存在する

- nullデータ(データは無いという意味)
- pandasはnullをNan(Not a Number, 数字データは無いという意味)と解釈する。
- この対処法は, 提供している “TSA_StockPrices.ipynb” に記載している。



END

