Solving Forecasting Problems in R and Python

John Mount, Ph.D.
Win Vector LLC
jmount@win-vector.com
https://www.win-vector.com

All slides, code, and data here:
https://github.com/WinVector/Examples/tree/main/TimeSeries

John Mount

Win-Vector Principal Consultant and Trainer John Mount has a Ph.D. in computer science from Carnegie Mellon and over 15 years of applied experience in biotech research, online advertising, price optimization and finance. He is one of the authors of the popular book "Practical Data Science with R", Manning, 2020 (now in its second edition).

1

Hold on this slide until ready to start.

Motivation

"One of the biggest mistakes in my data science career was not paying enough attention to time series models.

Time series is everywhere—finance, sales, marketing—and it can drastically improve your decision-making process."

– Mark Eltsefon, Staff Scientist at Meta
https://www.linkedin.com/posts/mark-eltsefon_datascience-timeseries-activity-7253363004694491137-94D8?utm_source=share&utm_medium=member_desktop

2

Forecasting problems tend to be high value.

Outline

- Motivation
- What is time series forecasting?
- Families of methodologies
- Our example problem
- Solving the problem using Stan
- What is Stan?
- Results/Observations

Luca Cambiaso FightingFigures

3

We will fill in some details as we go.

4

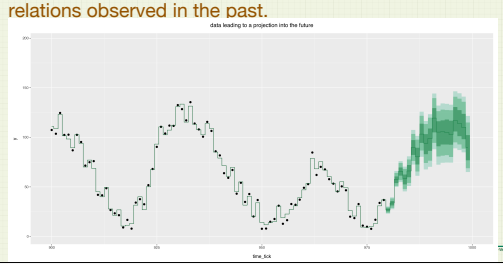
What is Time Series Forecasting?

(a slightly heterodox view)

5

Time Series Forecasting

- Produce an estimate of the future using trends and relations observed in the past.

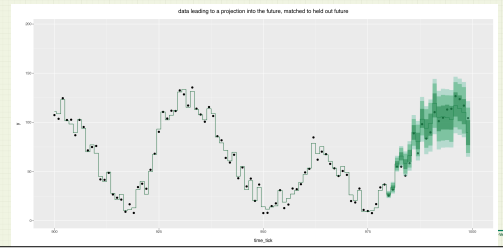


The dots are actual counts as a function of time, and the green line is the central prediction with uncertainty bands.

6

How to Evaluate Forecasts

- Hold out future, and compare prediction to eventual outcomes.

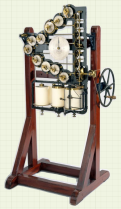


A prediction is good if it matches future data. Don't get bullied out of this.

Business Applications

7

- Predicting future demand
  - Online sales
  - Parking
  - Restaurant
- Predicting future price
  - Expenses
  - Stocks
- Predicting future revenue
  - Financial reporting/planning
- Physical systems affecting business
  - Tides
  - Weather



In 1876 A. Légé & Co., 20 Cross Street, Hatton Gardens, London completed the first "tide calculating machine" for William Thomson (later Lord Kelvin)

7

de-Motivation

8

"We have found that choosing the wrong model or parameters can often yield poor results, and it is unlikely that even experienced analysts can choose the correct model and parameters efficiently given this array of choices."

– Facebook research announcement of Prophet, 2017
https://research.facebook.com/blog/2017/2/prophet-forecasting-at-scale/

8

It is easy to get your hand caught in the gears.

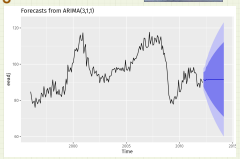
Example

9

From a manual on *how* to fit an ARIMA time series forecast model.

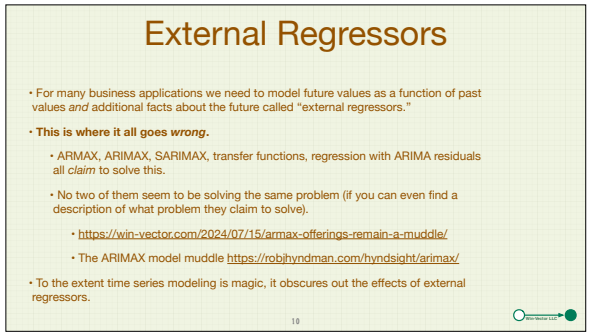
Two pages of deep technical analysis justifying the modeling choices and confirming technical modeling assumptions.

Followed by failing to call out that this is **obviously** a degenerate fit!



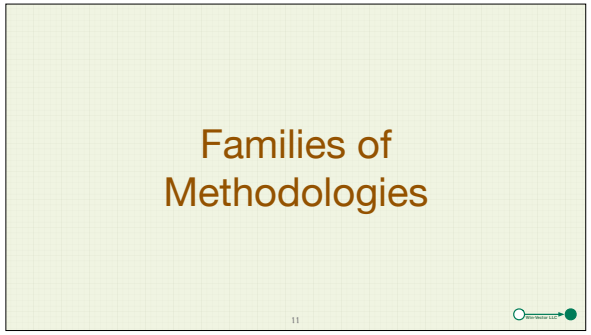
9

At no point in the training period was the curve flat. But that is precisely the predicted expected value. Yes, models of expected tend to have less variance than actuals https://win-vector.com/2024/06/15/good-models-do-not-match-variance/ . However, this is a degenerate fit. Most references avoid making the obvious by the simple expedient of never showing the final graph. Can't over emphasize the importance of insisting on method-agnostic hold out tests.



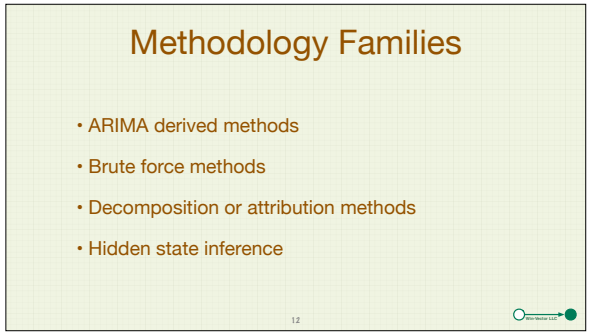
10

There is a fiction that ARMAX is how time series researches work. That appears to be wishful thinking. Time series research moved on to transfer functions and other terms of art.



11

One can really get lost in the terminology.



12

We can crudely put things like exponential smoothing under the "ARIMA" hat.

Packages Demonstrated

13

- R
  - rstan
  - fable
  - forecast
- Python
  - cmdstanpy
  - prophet
  - statsmodels
  - sklearn

We share worked solutions with all of these packages here:
https://github.com/WinVector/Examples/tree/main/TimeSeries

One advantage of R packages: Hyndman's writing is quite good.

ARIMA derived methods

14

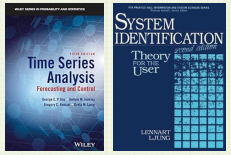
- "Auto regression integrated moving average"
- Fit for a conserved relation between observations
  - AR example $\hat{y}_t = \mathrm{E}[y_t] = 1.975 y_{t-1} - y_{t-2}$ for many t
  - Push that forward as our prediction

$$\hat{y}_t = 1.975 y_{t-1} - y_{t-2}$$
$$\hat{y}_{t+1} = 1.975 \hat{y}_t - y_{t-1}$$
$$\hat{y}_{t+2} = 1.975 \hat{y}_{t+1} - \hat{y}_t$$
$$\hat{y}_{t+k} = 1.975 \hat{y}_{t+k-1} - \hat{y}_{t+k-2}$$

  - Can run away!
- Regression with ARIMA residuals:

$$(1 - \phi_1 B - \phi_2 B^2)(y_t - \beta_0 - \beta_1 x_t - \beta_2 z_t) = c + (1 - \theta_1 B - \theta_2 B^2)\epsilon_t$$

- Very powerful
  - *Can* dominate other solution method when *only* measuring loss or fit quality.
- Wrongly thought of as "the only way to solve time series problems."



https://github.com/WinVector/Examples/blob/main/TimeSeries/ts_example.md
https://github.com/WinVector/Examples/blob/main/TimeSeries/sm_example.ipynb

Among the most powerful methods. The "get k steps out by running a 1 step out process k times" is very powerful, but also very sensitive. The epsilon terms are called shock or noise. It is a very bad idea to think of them as noise as they are often important external drivers (such as deposits to a bank account on in-flow to a reservoir).

Incompatible Definitions

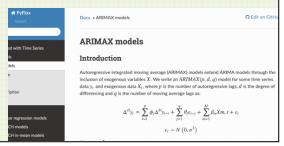
15

- Are external regressors transient? Just something sitting between your observations and a hidden underlying system, such as in regression with ARIMA residuals?

  R fable, R forecast: regression with ARIMA residuals

- Are external regressors durable? Something like a marketing effort or change in policy?

  PyFlux: durable
  https://win-vector.com/2024/07/15/armax-offerings-remain-a-muddle/



We don't have to fully read the equations to work out the implications, instead look at subscripts and difference operators. In our case the external regressors have only the current time-index and no integration or shifted time indices (such as the shock terms).

Brute force

- Re-write as a standard machine learning problem
  - Model $\hat{y}_{t+k} = \hat{E}[y_{t+k}] = f_k(y_t, \cdots y_{t-w})$ for $k = 1 \ldots u$
  - Fit for all the $f_k()$
    - Linear methods
      - https://win-vector.com/2023/05/07/a-time-series-apologia/
    - Neural net methods such as LSTM
    - Packages (not demonstrated)
      - sktime
      - skforecast

https://github.com/WinVector/Examples/blob/main/TimeSeries/nested_model_example.ipynb

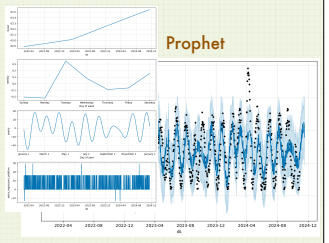
18

16

Not actually a bad idea. Misses a lot of the traps in forecasting by ignoring them. Kind of treated as “not allowed” until deep learning had great success with it. Low explainability even with linear methods, as we are a bundle of different models.

Decomposition or Attribution

- Fit past to things we know future values of:
  - Month of year …
- Use the combination of the future values of these things as our future prediction.



https://github.com/WinVector/Examples/blob/main/TimeSeries/Prophet_example.ipynb

17

17

Fit the past to things we know the future of: such as seasonal averages. Use this fit to project into the future. Very useful in attributing or explaining past effects. Tends to under-perform ARIMA on model quality evaluation- but returns more explainable results and useful data decompositions. The idea is: match past to things we know the future of.

Hidden State Inference

- Includes methods such as EM and particle filters
- Guess hidden state and parameters β, ζ

Estimate $\hat{\beta} = \hat{E}[\beta] = \frac{\int_{\beta,\zeta} d_{\beta,\zeta} P[\text{training data} \mid \beta, \zeta] \beta}{\int_{\beta,\zeta} d_{\beta,\zeta} P[\text{training data} \mid \beta, \zeta]}$

https://github.com/WinVector/Examples/blob/main/TimeSeries/Stan_soln.ipynb

18

18

Computationally expensive, but very versatile. This is the method we will pursue in this session.

19

Our Example Problem

20

Our Notional Example Problem

- We are modeling the number of diners visiting a restaurant each day.
- Improve prediction *and* utility of prediction by bringing in information
  - We have transient external regressors such as events being near our restaurant.
  - We have durable external regressors such as price changes and restaurant reviews.
  - Usually also add a few more easy external regressors to represent seasonality (day of week, month of year, and so on).
- Can also model this as two sub-populations of guests: loyal and transient
  - The issue is: these subpopulations are not labeled in our training data!

21

As Equations

$$\text{loyal\_demand}_{date} = \beta_{loyal} + \beta_1 \text{ loyal\_demand}_{date-1} + \beta_2 \text{ loyal\_demand}_{date-2} + \beta_{review}\, x_{review,date-1}$$

$$\text{transient\_demand}_{date} = \beta_{tmp} + \beta_{events}\, x_{events,date}$$

$$\text{total\_guests}_{date} = \max(0, \text{loyal\_demand}_{date} + \text{transient\_demand}_{date} + \text{noise}_{date})$$

- These equations are our specification of the problem. Under different assumptions we would write different equations.
  - Subtle point: we are modeling realized values, not the traditional expected values.
  - We only observe `total_guests` (not `loyal_demand` or `transient_demand`).
  - In practice we benefit from a *lot* more external regressors- such as reservations by date.
- Without controllable external regressors forecasting devolves into Cassandra's curse.

Cassandra by Evelyn De Morgan (1898, London)

This immediately requires bi-temporal notation: what was known about date A on date B. We will use “date” and “time” as synonyms in this talk.

Can Standard Packages Solve This?

22

$$\text{loyal\_demand}_{date} = \beta_{loyal} + \beta_1\ \text{loyal\_demand}_{date-1} + \beta_2\ \text{loyal\_demand}_{date-2} + \beta_{review}\ x_{review,date-1}$$

$$\text{transient\_demand}_{date} = \beta_{imp} + \beta_{events}\ x_{events,date}$$

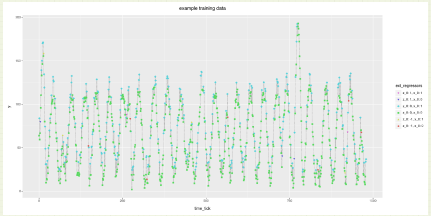
$$\text{total\_guests}_{date} = \max(0, \text{loyal\_demand}_{date} + \text{transient\_demand}_{date} + \text{noise}_{date})$$

- If we force $\beta_{review} = 0$, then this is "regression with ARIMA residuals" variation of ARIMAX.
- If we force $\beta_{events} = 0$, then this is the sort of system social scientists want to study policy changes (such as tariffs or seatbelt laws). Also called an ARMAX variation (may or may not in fact be).
- None of the ARIMA style packages we surveyed offered structural *choice or control*. The equations the package solves either match yours, or do not.

22

An Artificial Example

23



- Observed total count y
- Unobserved loyal customers y_{auto}
- External regressor z adds or removes loyal customers
  - modeling price changes, reviews, etc.
- External regressor x adds impermanent or transient customers
  - models people in the area for an event
- Problem more regular, and less noisy than most real world applications.

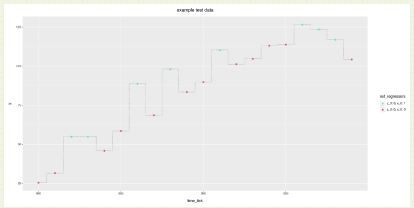
23

Time series methods tend to be really good and modeling sin(x). So if they are not good at noised up copies of sin(x), what good are they going to be on brutal real world data?

No real world data is ever this easy, so using the methodology does take some engineering (adding more lags, feature engineering and so on).

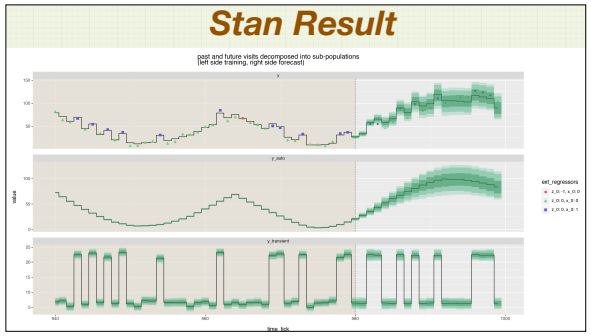
Held Out Test Data

24



- To cross-validate it is traditional (in finance) to design the system to be able to hold out arbitrary large contiguous blocks of time.
  - I.e. train the model in the future and see if it correctly works in the past.

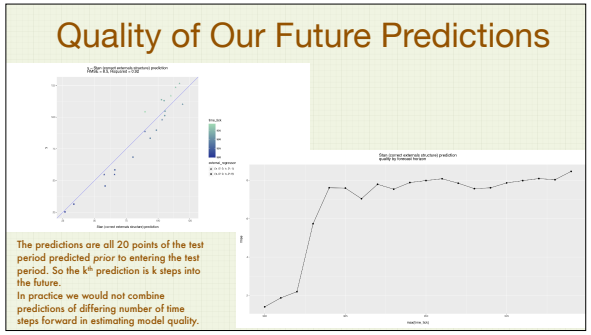
24



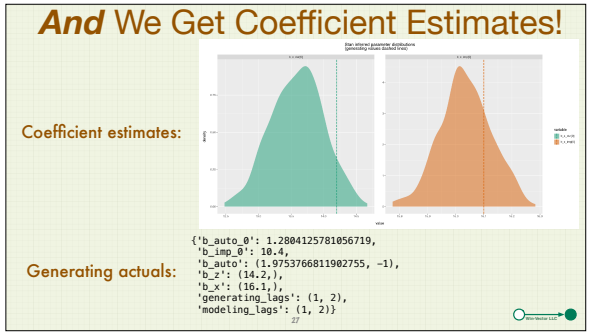
25

Even the historic factorization can be valuable to the business. y_auto is the loyal or durable customers, y_transient are the transient or impulse customers, and y is the sum of the y_loyal and y_transient. Only y is observed, and only in the training region.

In time series forecasting, really have to come to accept that the usual sklearn separation of .fit() and .predict() is not appropriate. Inferring parameters and the future are very related, and insisting on separating the two (while desirable) leads to problems.



26



27

Again even estimating b_z (the durable external regressor effects) and b_x (the transient external regressor effects) can be very valuable. The models that don't separate these concerns bias the estimate of at least one of these towards zero. These graphs are not evidence of bias as the distributions are the posterior estimates of the hidden parameters from a single data draw, and we reserve bias to describe a systematic defect seen across multiple data draws.

How We Solve

- Copy our equations into “Stan”
- And “turn the crank.”

```
y_auto[3:(N_y_observed + N_y_future)] ~ normal(
  b_auto_0
   + b_auto[1] * y_auto[2:(N_y_observed + N_y_future - 1)]
   + b_auto[2] * y_auto[1:(N_y_observed + N_y_future - 2)]
   + b_x_dur[1] * x_dur_1[3:(N_y_observed + N_y_future)],
  b_var_y_auto);

y ~ normal(
  y_auto
   + b_imp_0 + b_x_imp[1] * x_imp_1,
  b_var_y);
```

Complete workbook: https://github.com/WinVector/Examples/blob/main/TimeSeries/Stan_soln.ipynb

28

28

What we get from Stan

	lp__	accept_stat__	stepsize__	treedepth__	n_leapfrog__	divergent__	energy__	b_auto_0	b_imp_0	b_auto[0]	...	y[990]	y[991]	y[9
0	-3016.91	0.969951	0.007176	9.0	511.0	0.0	3558.90	0.827782	61.9535	-0.158711	...	47.5861	117.9720	73.2
1	-2993.15	0.942758	0.007176	8.0	255.0	0.0	3522.84	0.828451	61.3599	-0.180253	...	88.0751	82.6538	119.3
2	-2998.29	0.929456	0.007176	8.0	469.0	1.0	3525.98	0.831355	61.9564	-0.178463	...	92.3212	85.8090	31.7
3	-3042.34	0.550402	0.007176	9.0	511.0	0.0	3540.03	0.719813	62.3989	-0.118512	...	72.0224	82.1503	99.9
4	-2996.77	0.924559	0.007176	9.0	511.0	0.0	3538.41	0.736327	61.7225	-0.046399	...	41.1799	92.5996	77.61
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
3995	-2378.51	0.980328	0.001044	10.0	1023.0	0.0	2925.12	0.303021	58.8410	0.004656	...	49.0182	61.8958	34.61
3996	-2335.98	0.875501	0.001044	10.0	1023.0	0.0	2845.10	0.320605	58.1982	-0.002070	...	28.0628	60.1198	29.1
3997	-2326.53	0.952655	0.001044	10.0	1023.0	0.0	2864.31	0.332946	58.2261	-0.000332	...	23.1797	60.3515	25.0
3998	-2362.56	0.844985	0.001044	10.0	1023.0	0.0	2873.50	0.337636	57.8308	-0.030935	...	23.5624	76.7686	13.6
3999	-2344.09	0.990501	0.001044	10.0	1023.0	0.0	2838.88	0.309108	58.6106	-0.023015	...	40.2663	88.4004	11.6

4000 rows × 2035 columns

29

29

Each row is a sample. The columns we are interested in for a given row are the parameters (“b_*”) and hidden state (“y[*”, “y_auto[*”, “y_future[*”). One thing to keep in mind: the estimated state and parameters in a given row are mutually consistent as they were picked with the plausibility (exp(lp__) / Z , for some Z) high. It is a domain question if and when we can average these across rows.

Our Thesis

In business forecasting: ability to specify actionable structure beats technique in forecasting.

30

30

By actionable structure I mean properly structured external regressors that are under our control.

Thank you

All materials: https://github.com/WinVector/Examples/tree/main/TimeSeries#readme