Time series Principles

Mainly based on https://otexts.com/fpp2/

Quick Notes:

Contents:

Autocorrelation: measures the linear relationship between lagged values of a time series

- If trend exists: smaller lags(lag1,2,…) will have high positive correlations (Autocorrelations)
- If Seasonal: large for seasonal lags (lag12, 24…)
- If Both trend and seasonal: slow decrease in lag correlations due to trend and peaks in respective lags due to seasonality
- White noise: No autocorrelation, not exactly 0 as there will be random variations, ACF: 95% acf within +/- 2/sqrt(length of series)

Forecasting methods: Simple Methods used as a benchmark for evaluating various advanced methods

- **Average, Naïve**: value of last observation, optimal when data follows random walk
- **Seasonal Naïve**: last observed same season of the year

- **Drift**: variation of naïve, forecast increase/decrease over time; Amount of change over time is based on the average change seen in the historical data, Drawing line b/w first and last and extrapolating for future, captures the trend

Transformations and adjustments:

Simplification of pattern in the historical data; by removing known sources of variations, to make past more consistent.

Calendar:

Variations due to different month lengths, Variations due to different trading days in month

Mathematical:

If data shows variations that increase/decrease with the level of series.

- Log: change in log values are relative/% change on the original scale
- Log base 10: increase in 1 on log scale means increase of multiplication of 10 on original scale, log base 10: Stay positive
- Power: square, root

Box-Cox: Test for Transformation (Log/Power) based on lambda

Lambda: 0: log; otherwise, power

Bias adjustment:

Residual Diagnostics (Actual vs. Fitted values)

- Residuals: uncorrelated, zero mean (if not then forecast is biased)
- If residuals have mean 'm', then simply add mean m to all forecast, to overcome bias issue
- Constant variance (timeplot), normally distributed (histogram- Tail, skewness), (Box-Cox) to overcome
- ACF to check autocorrelation on residuals
- If residuals violated constant variance and normal distribution, forecast may be correct but prediction interval may go wrong
- Box-Pierce test: Formal Test for autocorrelation apart from checking individual r values within limits
- Ljung-Box test: more accurate test

Evaluating accuracy:

- Train Test split, window/subset function; no future observations can be used in constructing the forecast
- Residuals: errors calculated on train set
- Forecast errors: errors calculated on test set
- Scale dependent: MAE/RMSE
- MAPE: unit free, compare bet diff data sets
- Prediction Interval:
    - Interval within which prediction lies with certain prob. uncertainty associated with each forecast.
    - Assuming forecast errors are normally distributed (1.96stdoferrors), if one step forecast, stddev of residuals will be used
- RMSE: same scale like std.dev, : Eg. Y=20; RMSE=2; std.dev around 2 units;
- 68% sure, yt=between 18-22; 95% sure yt=between 16-26 (X+1.96SE)

Time series Regression models:

- Assumptions about errors: mean zero, not autocorrelated, unrelated to predictors, constant variance, normally distributed
- Least square: choose coeffs, by minimizing sum of squares of errors
- std. error of coeffs: uncertainity in coeff estimate
- t value: estimated coeff/std.errcoeff
- pvalue:
- Fitted values:
- Goodness of fit: Coeff of determination: prop of variation in y explained by the model
- Test data performance: validating model performance on test data is more important than r2
- Standard deviation of error: residual std. error; can compare this with sample mean of y or std dev of y

Evaluating Regression models:

ACF plot of residuals: Lagrange Multiplier - test for autocorrelation for Timeseries Regression model

Histogram of residuals:

Residual plot against predictors: residual vs each predictor scatter plot should show random behaviour. If systematic, then non-linear can be tried

Residual plot against fitted: Hetroscedacity if pattern exists, may try transformation of y

Outliers/influential obs:

Useful predictors: trend, dummy variable (seasonal), Intervention variable: Handling spike, level shift, slope change, Trading day: bizdays(), effects of different # of days of the week, Distributed lags, Easter, Fourier series: harmonic regression

Selecting Predictors: Leave one out sample CV average, AIC, BIC, Adjusted R2, if we have less predictors, we can model all possibilities (2^n) and check various measures, Backward step regression, Use lagged predictors

Non Linear Regression:

Log-Log form: Slope is interpreted as elasticity, avg% change in y resulting from 1% increase in X

X, Y will be transformed to log/sqrt but model coeff will be in linear form:

Log (1+X): this is being used if x has any 0 values

Piecewise regression/spline regression:

Piecewise cubic spline: smooth

Nonlinear trend: Nonlinear trend constructed of linear pieces, natural cubic smoothing splines

Multicollinearity:

Similar info provided by 2/more predictors, not a problem except when there is a perfect correlation

Decomposition:

MA, Trend: captures main movement in the series without the seasonal/minor fluctuations, symmetric/non MA: 3x3; 2x4

Classical: End point estimation problem, : trend get over smoothened: same seasonal pattern repeats each year: not robust to unusual pattern/observation

X11: Handling Trading day, holiday, effect of known predictors: robust to outliers and level shifts

STL: t.windows/s.windows-small values capture rapid changes, #of consecutive observations to estimate trend, #of consecutive periods to estimate seasonality

Irregular: Happened in the past, will not repeat in future; statistically insignificant, we can ignore

How stl is used as multiplicative:

Log- Number of zeros, log10=1, log100=2…..represents multiplicative,

Logy=t+s+e

Y=10^(t+s+e)=10^t X 10^s X 10^e (becomes multiplicative)

Exponential Smoothing: Weighted avg. of past observations, weights decaying exponentially as observations get older

Simple exponential smoothing (ses), used for data with no clear trend/seasonality

naïve is also ses with last obs taking entire weight, average smooth- takes equal weights to all the obs

Forecast at T+1 is equal to weighted avg. between most recent obs yt and forecast of previous value that is yt fitted

Forecast yT+1 = WA(yT, yTfitted), fitted values are one step forecast

y2fitted|1 = WA(y1, y1fitted)

y3fitted|2 = WA(y2, y2fitted|1)

y4fitted|3 = WA(y3, y3fitted|2)

Optimization to get the smoothing parameters and initial values through minimizing SSE

SES Eqn:

$$\hat{y}_{T+1|T} = \alpha y_T + \alpha(1-\alpha)y_{T-1} + \alpha(1-\alpha)^2 y_{T-2} + \cdots, \tag{7.1}$$

Component form:

$$\begin{aligned} \text{Forecast equation} \quad & \hat{y}_{t+h|t} = \ell_t \\ \text{Smoothing equation} \quad & \ell_t = \alpha y_t + (1-\alpha)\ell_{t-1}, \end{aligned}$$

If we replace ℓ_t with $\hat{y}_{t+1|t}$ and ℓ_{t-1} with $\hat{y}_{t|t-1}$ in the smoothing equation, we will recover the weighted average form of simple exponential smoothing.

Holts linear Trend:

$$\begin{aligned} \text{Forecast equation} \quad & \hat{y}_{t+h|t} = \ell_t + hb_t \\ \text{Level equation} \quad & \ell_t = \alpha y_t + (1-\alpha)(\ell_{t-1} + b_{t-1}) \\ \text{Trend equation} \quad & b_t = \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)b_{t-1}, \end{aligned}$$

Forecasting with trend: Has 2 eqns: 1 Forecast eqn and 2 smoothing eqn (level and Trend)

lt (estimate of level) with smoothing parameter (alpha), Bt (estimate of Trend) with smoothing parameter (beta) weighted average of current value and previous level+Trend)

weighted average of level change and previous trend

Weighted average of error (current -prev level, trnd) and previous season

Holt Trend

Previous Level + Trend

WA(current value + (previous level +Trend))

WA(current level-previous level + previous Trend)

HoltWinter - Removing seasonality

Previous Level + Trend + Seasonality

WA((current value-seasonality)+ (previous level +Trend))

WA(current level-previous level-Previous Trend, previous Trend)

WA(current value-previous level-previous trend, last period seasonality)

level is seasonaly adjusted

Additional Notes on SES:

Exp. Smoothing:

- Combination of previous values
- Yt+1 = (forecast of today + how wrong I was today)
- Yt+1 = yt_{cap} + (1-a)[yt-yt_{cap}]
- Yt+1 = (actual of today + forecast of today)
- Yt+1 = ayt+ (1-a)yt_{cap}
- If a=1, becomes naïve forecast, tomorrows forecast is equal to todays actual
- If a=0, becomes constant forecast, tomorrows forecast is equal to todays forecast

SES=half of what is today, ¼ of yesterday, 1/8 of daybefore yesterday

$$\hat{y}_{T+1|T} = \alpha y_T + \alpha(1-\alpha) y_{T-1} + \alpha(1-\alpha)^2 y_{T-2} + \cdots, \qquad (7.1)$$

Double Exp. Smoothing:

- Forecast = Level + Trend
- Same as SES, but it captures trend additionally.
- Yt+1=lt + bt
- Lt = SES= Yt+1 = ayt+ (1-a)yt_{cap}
- bt = Trend = change in level + Previous Trend
- bt = b(lt-lt-1) + (1-b)bt-1
- if b=1, bt = change in level
- if b=0, bt= bt-1; constant trend….
- Triple (Holts Winter) = Level + Trend +Seasonal
- Seasonal = what is left over after removing previous level and trend adding previous season

- SES: will give same forecast if we use h=>1, so it is useful only when we need onetime ahead forecast, fits only level.
- Holts: allow trend change, recover the trend
- Holts winter: allow trend and seasonal, recover seasonal

Statespace models

Building forecast model through max likelihood instead of minimizing sse; Generate Prediction intervals, Forecast distribution

ARIMA:

Stationary: Properties doesn't depend on time at which series is observed; time series with trends, or with seasonality, are not stationary — the trend and seasonality will affect the value of the time series at different times

White noise series: stationary — it does not matter when you observe it, it should look much the same at any point in time.

- No predictable pattern in the long run:
- google stock price data is non stationary as there is some trend visible:
- but when we calculate the daily stock change, series becomes stationary (Differencing:
- Differencing: stabilise mean, reduce trend and seasonality
- ACF: Non stationary-Decreases very slowly, large r1
- ACF: stationary - drop to 0 quickly
- Random walk model: forecast equal to last observation
- Random walk model with drift:
- Log : will stabilizes the variance
- seasonal difference : will remove trend, seasonality
- unit root test: to test if differencing is required
- Null Hypothesis that data is stationary, look for evidence that null hypothesis is false, if p<< small, rejecting null hyp, differencing required:
- kpss test: Measure of seasonal strength; if data has higher seasonal strength, then seasonal diff is required

Backshift operator

- timeseries lags
- Byt=yt-1
- B(Byt) = yt-2
- B12yt=yt-12
- y'=yt-yt-1
- yt-Byt
- yt(1-B)
- y'=(1-B)yt
- y''=(1-B)^2yt

AR Models

Linear combinations of past values of variables (series); auto- regression of variable against itself; Multiple regression with lagged values of yt as predictors

AR(p) models; random walk is ar(1); random walk with drift is ar(1) with constant term

MA models: Past forecast errors in regression like models

Non seasonal arima models

- Lagged values of Y, lagged errors
- ACF and PACF plots:to find values of p and q
- PACF
- if yt and yt-1 are correlated, then yt-1 and yt-2 are also correlated,
- then yt and yt-2 are also correlated simply bcos they both are connected to yt-1, rather than bcos yt-2 containing new information which is not there in yt

PACF

- measures relationship b/w yt and yt-k after removing the effects of lags of 1,2,3,…k-1
- It helps to find AR terms
- auto arima with this option helps us to find better estimates
- stepwise=FALSE and approximation=FALSE

Max likelihood estimation

- similar to least squares
- Information criteria
- AIC BIC
- determining parameters of arima

Autoarima: (Internal working)

a. Four initial models are fitted:

- ARIMA$(0, d, 0)$,
- ARIMA$(2, d, 2)$,
- ARIMA$(1, d, 0)$,
- ARIMA$(0, d, 1)$.

- 1: number of differences using kpss test
- 2: value of p, q are chosen based on low AIC values
- A: 4 initial models
- A: constant is included for d=0,1
- A: if d<=1,
- B: best model with low aic is termed as current model
- C: Variations in current model , best is choosen, becomes new current model
- D: repeat step 2C
- Outliers, Transformation: stabilizes the variance, differencing- Stationary, ACF, PACF
- ACF of residuals, portmanteau test of the residuals, check for whitenoise, if not try different model
- calculate forecast
- First Random walk model will be built, if errors are autocorrelated, try adding lag of y difference
- positive autocorrelation is usually best treated by adding an AR term to the model and negative autocorrelation is usually best treated by adding an MA term.
- ARIMA(0,1,1) without constant = simple exponential smoothing:
- ARIMA(0,1,1) with constant = simple exponential smoothing with growth:

- ARIMA(0,2,1) or (0,2,2) without constant = linear exponential smoothing:
- ARIMA(1,1,2) without constant = damped-trend linear exponential smoothing
- Forecast with Arima errors
- to forecast using regression with arima errors,
- 1st regression part of the model, 2nd arima part of the model, combine results

Stochastic and deterministic trend

https://www.econometrics-with-r.org/14-7-nit.html

Deterministic: Adding time to a timeseries model to capture trend/ non random function of time

Stochastic: driven by random shocks, difference the data before modelling/ random function of time

Trending mean is a common violation of stationarity: 2 ways to handle

Trend stationary: mean trend is deterministic, function of time, remove trend, residual is stochastic stationary

Difference stationary: mean trend is stochastic, not a function of time, difference the series, residual is stochastic stationary

Check the residuals for autocorrelation, model residuals as stationary stochastic process. Arima process

Forecasting Model selection guidelines:

- Longer seasonal periods, dynamic regression with fourier terms will be better
- Shorter seasonal periods, monthly data (12), quarterly (4) Arima, ets can be used
- Large m, estimation becomes difficult and many (e.g, need to estimate 365 parameters if it is daily)
- Arima, Ets- memory issues
- Seasonal differencing of high order doesn't make sense, e.g daily data(today and 1 yr back)
- Harmonic Regression is used in such cases
- Adv:
- any length of seasonality
- more than one seasonal pattern
- smoothness in seasonality
- k- number of fourier sin and cos
- fourier transform
- https://betterexplained.com/articles/an-interactive-guide-to-the-fourier-transform/
- circular path can approximate any signal, smoothie to ingredients/reciepe eg. fourier term for capturing seasonality
- Weekly data: STL with nonseasonal method applied to seasonally adjusted data
- STL with ETS(A,N,N) model applied to seasonally adjusted data

Number of fourier term is based on minimizing aic, Arima order is also based on minimizing AIC

STL/Tbat- when seasonality changes over time; dynamic- if we need to add useful covariates

Daily/sub daily data

- has multiple seasonal pattern, complex seasonality
- Use ETS/Seasonal Arima, if series is short, so that one seasonality is present
- Use STL/Dynamic/Tbat, if series is long, so that longer seasonality period apparent
- Use dynamic regression, if we have moving holidays denoted as dummy variable along with fourier terms

Lagged Predictors

Impact of the predictor may not be immediate always

Adv. Campaign may impact the sales after the end of campaign

Adv. Expense will have an impact after few months

Change in safety policy may reduce accidents immediately, but will have a diminishing effect as people become familiar and careful with the new policies

Lagged predictor with arima error; value of lags selected using AICs

Insurance quotations provided and expenditure on adv.

Optimal lags will be decided based on AIC value of the fit

Regression with Arima Error

Error of Arima Error model will be white noise

number of quotations will be dependent on current and previous month exp values

Tbats model: Different from dynamic regression by seasonality is allowed to change slowly over time; Takes time

Complex seasonality with covariates

OLS: underlying linear relationship and normally distributed error

#General

random variable - continuous/descrete

prob.mass function -discrete variable

prob.density function - continuous

independent and identically distributed

Parametric and Non Parametric

If data is not normal and unlikely to classify as specific distribution, then assume non parametric

Uses fewer assumption about the data, increased complexity, but better flexibility, applicability

Confidence intervals and run statistical tests to assess the probability of obtaining the observed results.

Confidence intervals describe a range of values that would contain the true parameter (1− alpha)% of the time.

Collect infinite number of sets of randomized data from known distribution, make 95% CI for each set

95% of interval generated will contain true parameter.

How PACF/Partial Correlation works?

https://www.rdocumentation.org/packages/ppcor/versions/1.1/topics/pcor.test

We have data for Y, Lag1 of Y, Lag2 of Y…….

PACF YLag2: Correlation between Y and YLag2, removing the effect of YLag1

- Step1: Regression Model between Y and YLag1, Get the Error (Residuals)-Model1Error
- Step2: Regression Model between YLag2 and YLag1, Get the Error (Residuals)-Model2Error
- Step3: Correlation Between Model1Error and Model2Error will give PACF YLag2

PACF YLag4: Correlation between Y and YLag4, removing the effect of YLag1, YLag2, YLag3

- Step1: Regression Model between Y and YLag1, YLag2, YLag3, Get the Error (Residuals)-Model1Error
- Step2: Regression Model between YLag4and YLag1, YLag2, YLag3, Get the Error (Residuals)-Model2Error
- Step3: Correlation Between Model1Error and Model2Error will give PACF YLag4