Lecture 21:
Time Series Statistical Models

David Carlson

Today's Outline

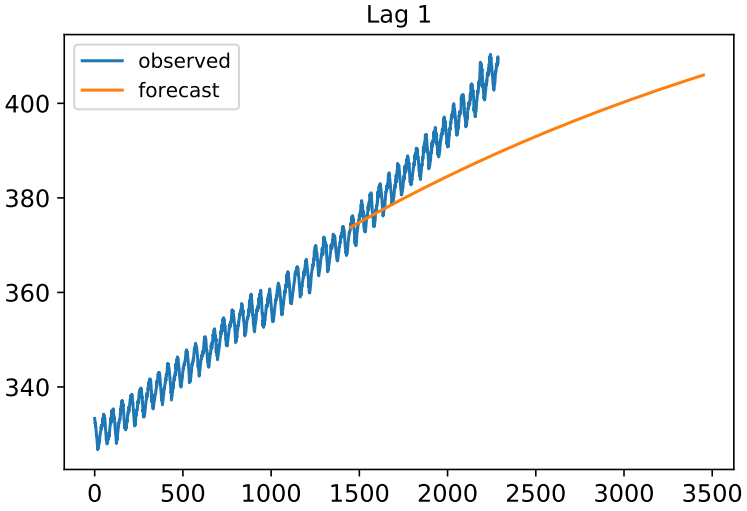
- Today is really a statistics lecture
- Traditional Linear Statistical Models for Time Series
  - Autoregressive Models
  - Moving Average Models
  - Autoregressive Moving Average Models
- Latent Space Models
  - Kalman Filtering
- Coming up, if time permits:
  - Gaussian Processes
  - Hidden Markov Models

Autoregressive Model

- An autoregressive model is a linear model that's based just on historical patterns.
- Let our time series be written as $\boldsymbol{x}_{1:T}$, then we have the statistical model:
  - $x_t = \beta_0 + \sum_{j=1}^{p} \beta_j x_{t-j} + \epsilon_t.$
- ϵ_t is additive iid Gaussian noise, which is modeled as $\epsilon_t \sim N(0, \sigma^2)$
- Given the "order" or number of lags *p* of the model, this is fit in exactly the same way as linear regression. See our first time series lecture for how to fit the data.

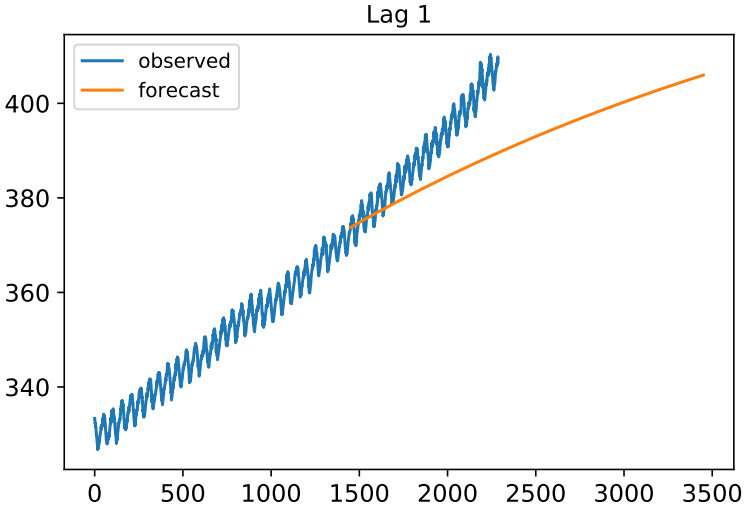
Prediction from an Autoregressive Model

- If we wanted to predict the future from an autoregressive model, the "optimal" (under the model) prediction is given by:
- $x_t = \beta_0 + \sum_{j=1}^{p} \beta_j x_{t-j}$
- On the right, we visualize the resu of the autoregressive model with p=1.
- Doesn't do great, why is that?

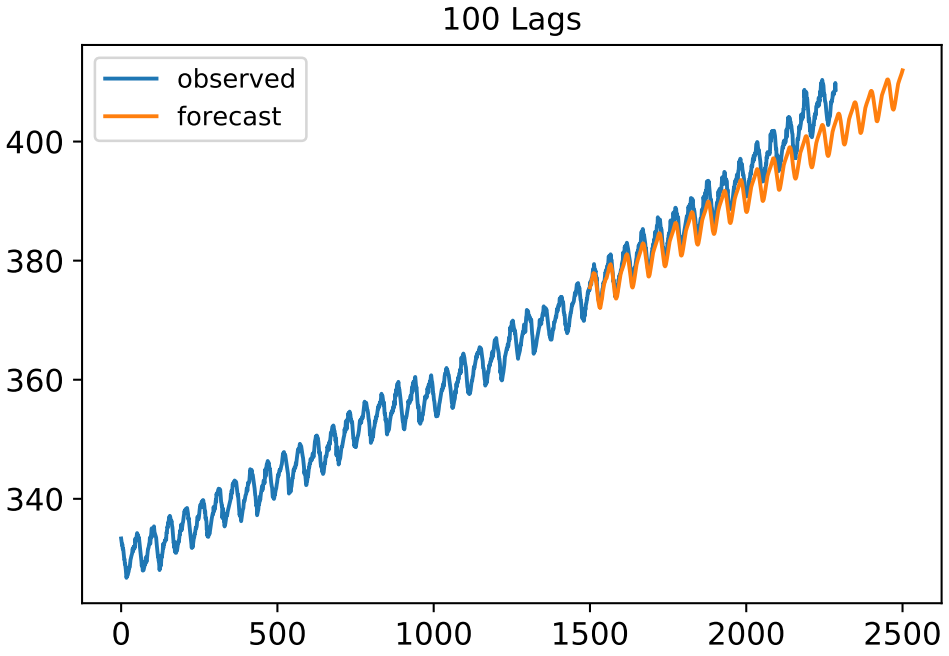


Prediction from an Autoregressive Model

- If we wanted to predict the future from an autoregressive model, the “optimal” (under the model) prediction is given by:
- $x_t = \beta_0 + \sum_{j=1}^{p} \beta_j x_{t-j}$
- On the right, we visualize the result of the autoregressive model with p=1.
- Doesn’t do great, why is that?



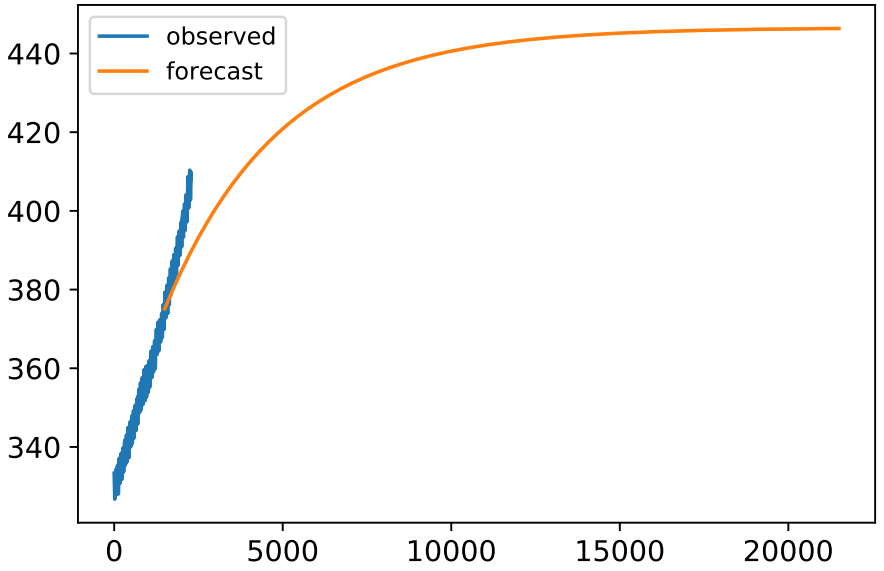
Eventually Captures the Periodicity



Implicit Assumption of Stationarity

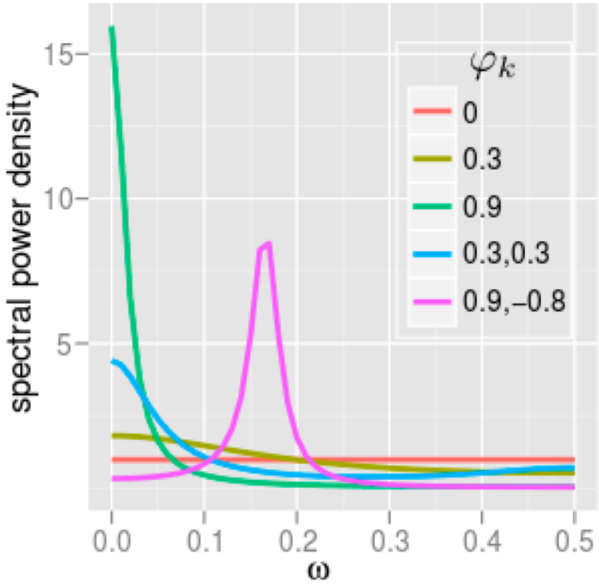
- An autoregressive model (typically) implicitly assumes that the distribution is *stationary.* Mathematically, this means that as the forecast distance goes further away from the observed data, the "optimal" prediction is the offset parameter.

- Think about the the model form
- $x_t = \beta_0 + \sum_{j=1}^{p} \beta_j X_{t-j} + \epsilon_t$



Frequency Response of an Autoregressive Model

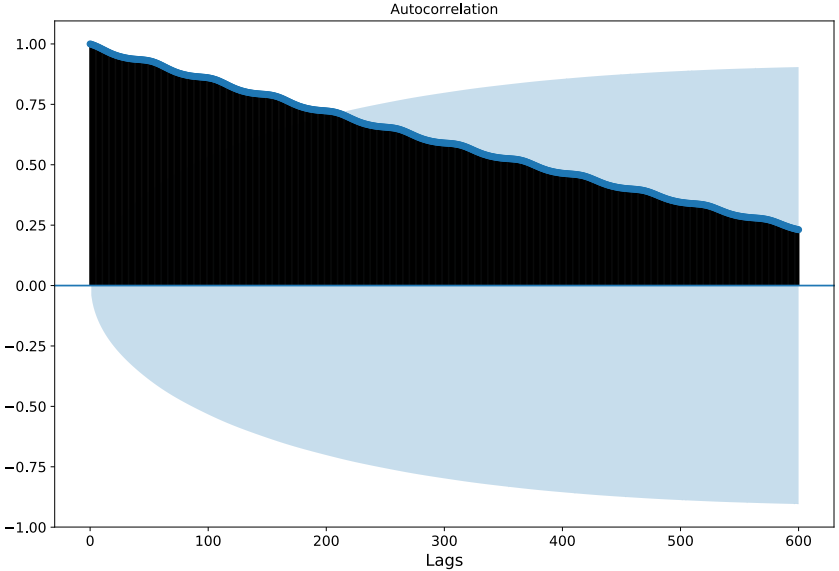
- When we're predicting with the autoregressive parameters, it has the formulation of a frequency filter.
- We can visualize the autoregressive model by viewing it's frequency spectrum, just as we looked at the Butterworth filter.
- Mathematically, the (stationary/asymptotic) frequency power is given by:
- $$S(\omega) = \frac{\sigma^2}{\left|1-\sum_{k=1}^{p} \beta_k e^{-i\omega k}\right|^2}$$
- Can start filtering for periodic signals once *p* is 2 or larger. Can filter for multiple periods.

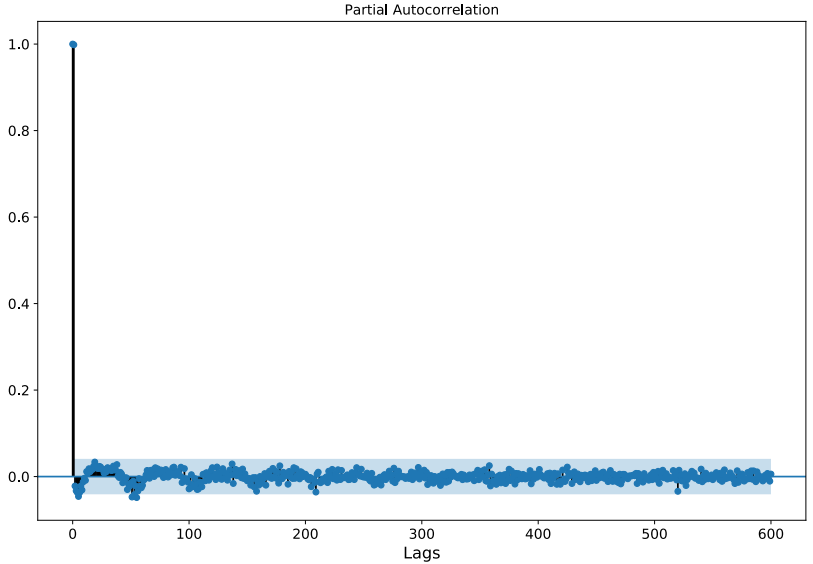


Choosing the complexity of the model

- The complexity of the model can be chosen by a variety of means:
  - Modification of Aikake Information Criterion (AIC) very close to the BIC model that we discussed previously
  - Partial Auto-Correlation (traditionally most commonly used)
  - Feed-Forward Cross-Validation (as discussed during the introduction to time series)
- Auto-correlation is given by:
- $\alpha(k) = E[(x_t - \mu)(x_{t+k} - \mu)]$
- The partial autocorrelation is defined as the amount of autocorrelation not explained by the intermediate lags. Let $\hat{x}_t^{k-1}$ be the best linear prediction from *k-1* lags (a *k-1*) autoregressive model, then
- $\alpha(k) = E\left[\left(x_t - \hat{x}_t^{k-1}\right)\left(x_{t+k} - \hat{x}_{t+k}^{k-1}\right)\right]$

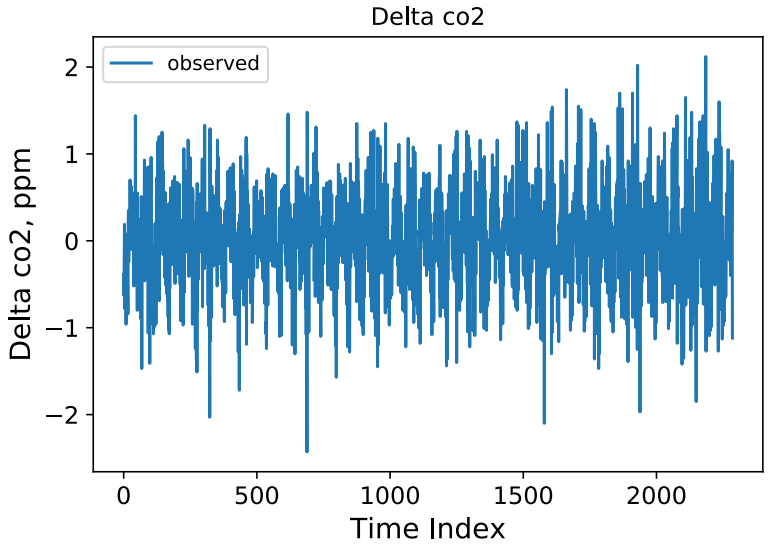
Visualization of the Partial Autocorrelation



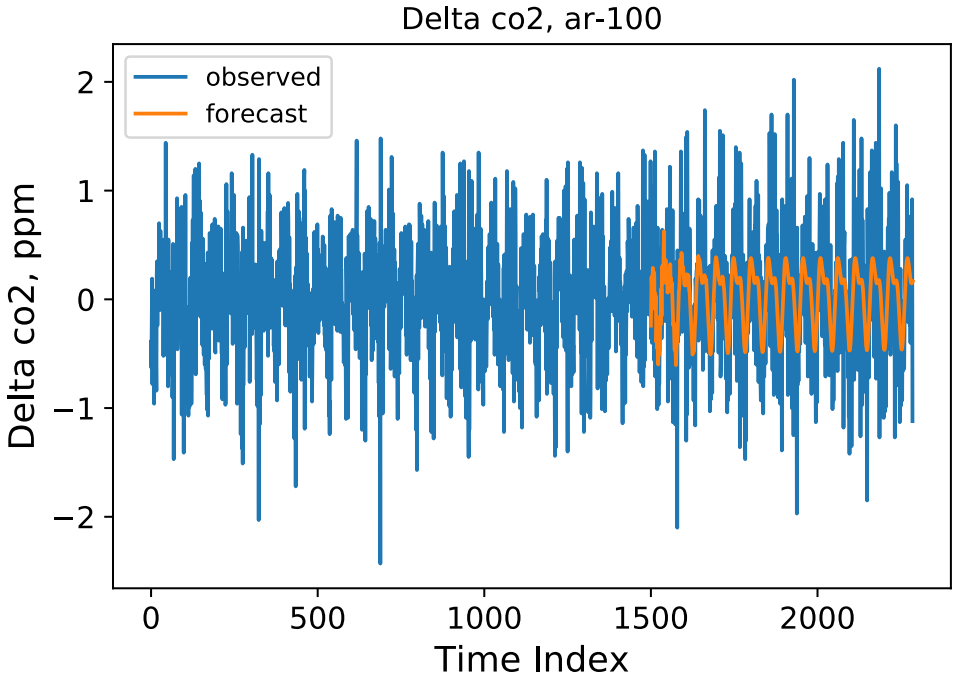


Differencing

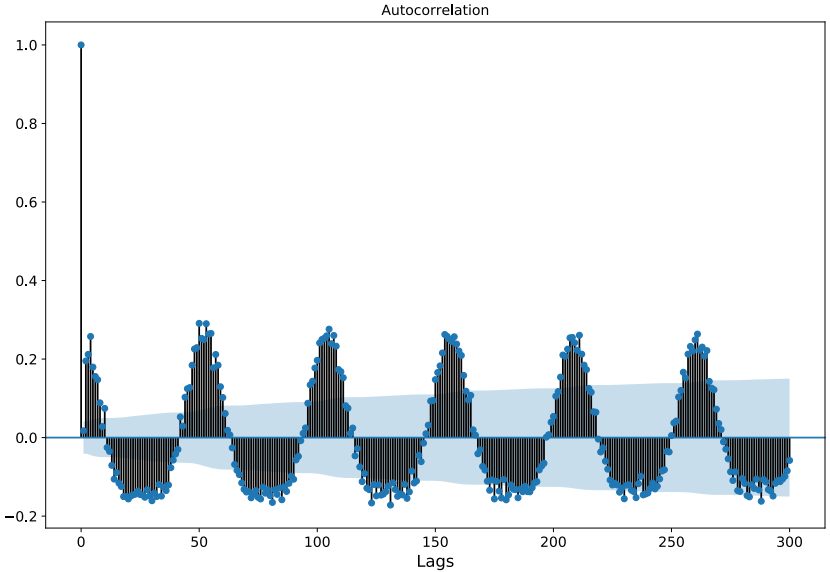
- Instead of using the original signal (which may have drift), a common technique is to use the *differences* of the signal. That is, to model:
- $x_t - x_{t-1}$ instead of x_t
- See the trend on the right, which looks *much* more stationary than the previous trend.
- To get final predictive result in the end, it has to be integrated (not an issue, but important to be aware of).

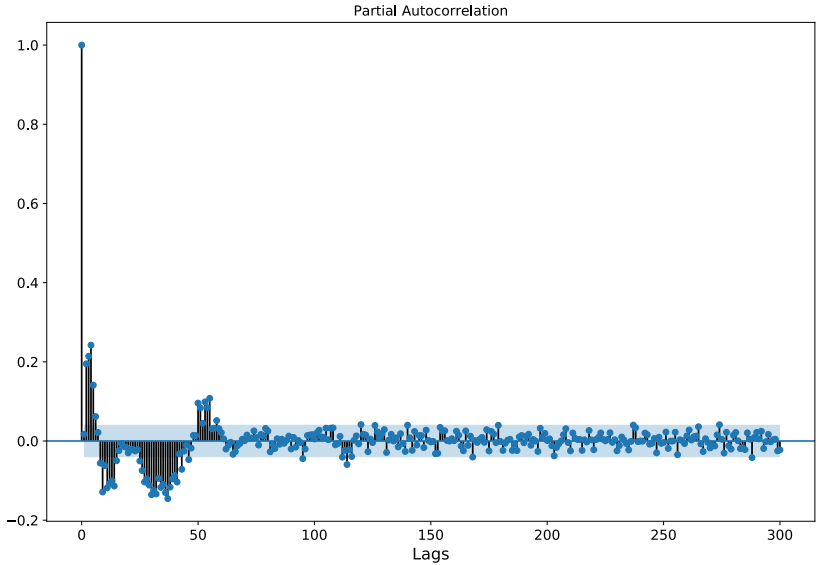


Prediction on the differences



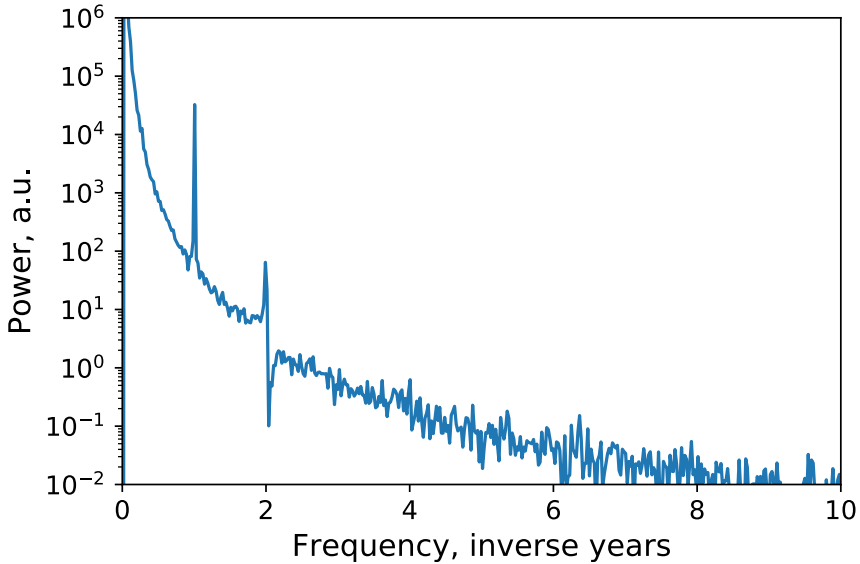
Visualization of the Partial Autocorrelation on the Differences





Differencing is a Filter

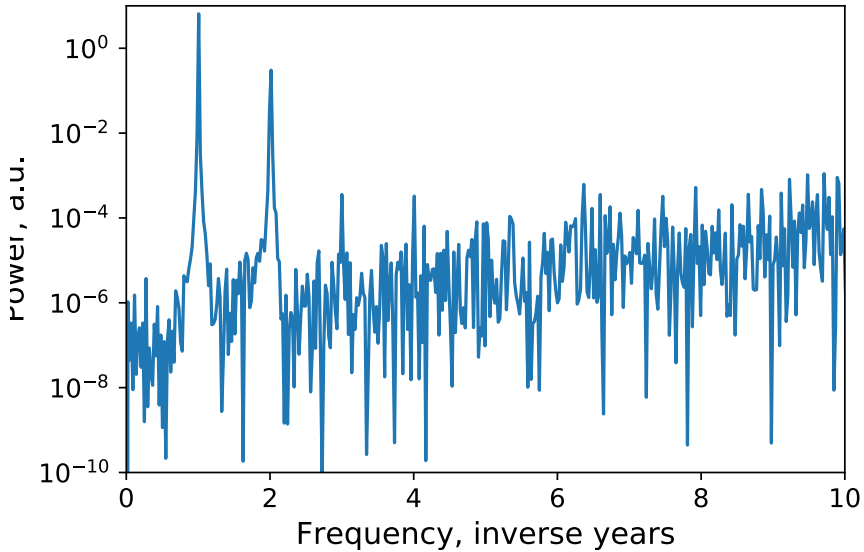
- The difference operator is given as
  - $\Delta_t = x_t - x_{t-1}$.
- This can be viewed as a *high-pass filter* that is removing low frequency information.
- Consider the original PSD on the right, what happens to it after the differencing?



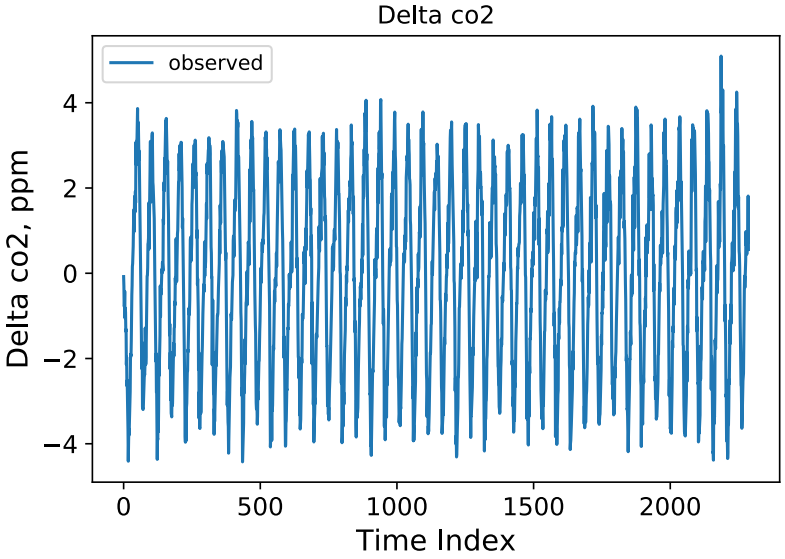
Differencing is a Filter

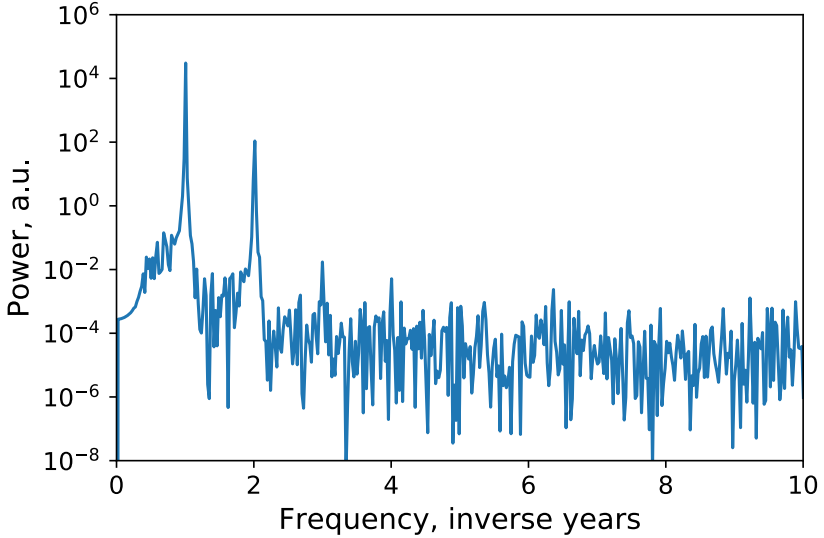
- The filter response is proportional to the frequency:
- $h(\omega) \leftarrow \mathcal{F}(f(t))$
- $\tilde{h}(\omega) \leftarrow \mathcal{F}(f'(t))$
- $|\tilde{h}(\omega)| \propto |\omega h(\omega)|$

- It intensifies high-frequencies, which can amplify the noise in the signal.



Filtering gives a Cleaner Signal





Why is Differencing Used?

- Differencing has several advantages:
  - Conceptually simple (easy to explain)
  - Reconstruction to the original signal is easy: $x_t = x_0 + \sum_{i=1}^{t} \Delta_i$
  - Easily handles linear types of non-stationarities (a non-zero mean gives a linear drift)
- Be careful with exploring high-frequency power
  - Can swamp the important information in the data
  - Can also add a low-pass filter to remove high-frequency information that isn't believed to be relevant

- Overall, has some nice advantages (and often we care about increments rather than the true value, e.g. stocks) and is commonly used.

Modeling the Differences Requires Integration

- If we model the differences, the reconstruction $x_t = x_0 + \sum_{i=1}^{t} \Delta_i$ is a discrete integration
- This type of model is a Autoregressive Integrated model, soon to be an Autoregressive Integrate Moving Average Model (ARIMA)

THE “MOVING AVERAGE” MODEL

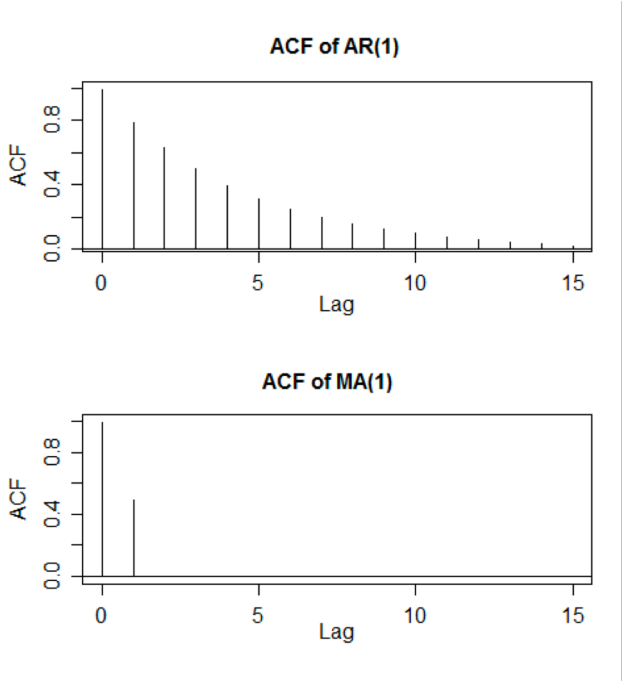
Moving Average Model

- A complement to the Autoregressive Model is the Moving Average model, which for an order q has a model formulation of:
  - $x_t = \beta_0 + \sum_{j=0}^{q} \theta_j \epsilon_{t-j}$.
- Note here that the correlation is only due to the correlation in the additive Gaussian noise. I.E. the error terms are correlated.

- This is a surprisingly common situation. We won't linger here because typically we'll want to combine this with a more complex framework.
- Inference is also tougher than the corresponding autoregressive model.

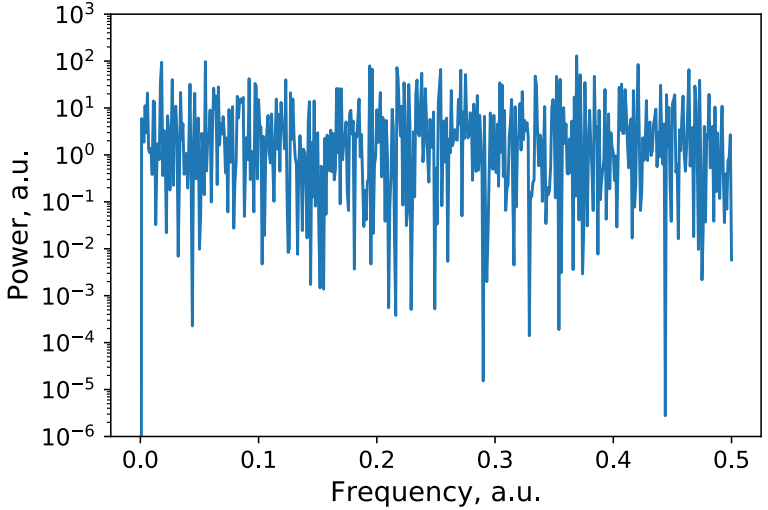
Key Difference Between Autoregressive and Moving Average

- The autoregressive (AR) model is given by:
- $x_t = \beta_0 + \sum_{j=1}^{p} \beta_j X_{t-j} + \epsilon_t$
- The moving average (MA) model is given by:
- $x_t = \beta_0 + \sum_{j=0}^{q} \theta_j \epsilon_{t-j}$.
- The difference is that the MA is dependent on the historical errors rather than the actual observed value.
  - Much shorter autoregressive history!
  - MA can be more realistic under sensor noise.



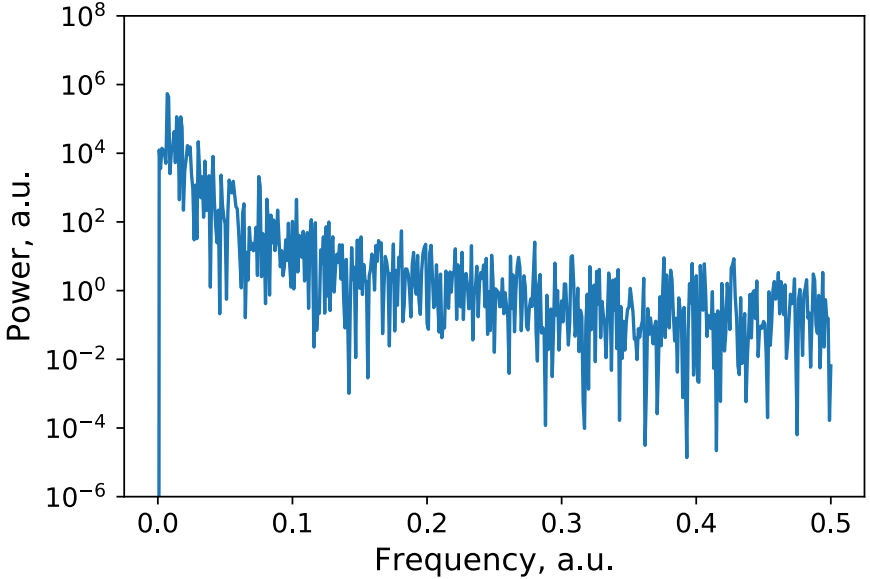
Are my errors correlated?

- Figuring out whether error (or unexplained variance) is correlated in time series can be facilitated by looking at the spectrum.
- This is facilitated by understanding what some of the spectrum of the time series look like.
- First, consider the case of independent random Gaussian errors, known as the ”white noise” spectrum.



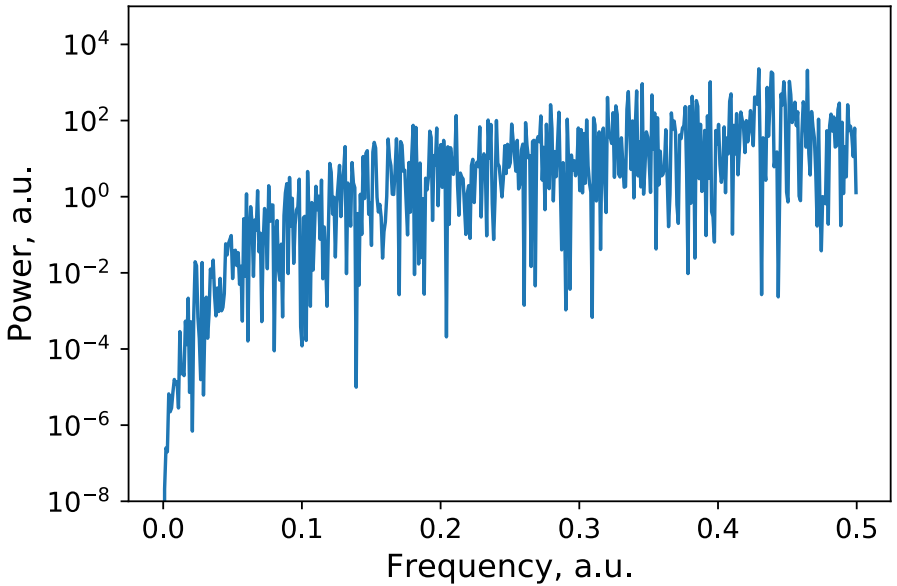
Autoregressive Noise

- An autoregressive noise model can be considered where
  - $\epsilon_t = \sum_{i=0}^{\infty} \alpha^i \gamma_t$.
- Where γ_t are all independent random Gaussians and α is between 0 and 1. This skews the power spectrum of the noise.
- Integration or Brownian Motion is a special case of this with $\alpha = 1$. Note that such a model isn't stationary.



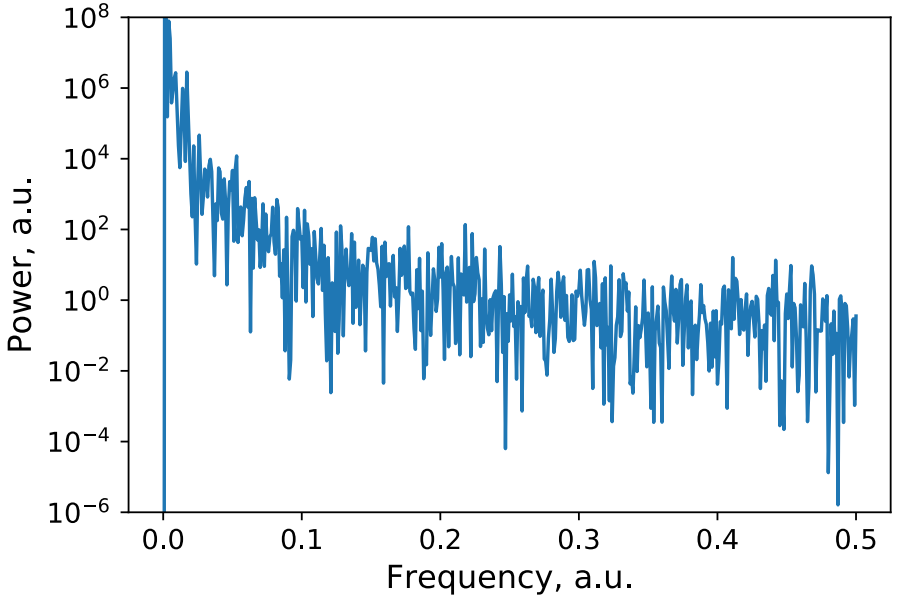
Derivatives of Errors

- The derivative induces a particular frequency spectrum. If we take the differences of errors, then we get increased strength in higher frequencies.



Brownian Motion

- Brownian motion covers the case where
  - $\epsilon_t = \sum_{i=0}^{\infty} \gamma_t$.
- This is a common physics situation (e.g. particle motion). The spectrum is shown on the right.
- Using differencing exactly results in a white noise spectrum!



Autoregressive Moving Average Model (ARMA)

- If we think some of both of the previous models capture some property of the data, we may want to combine it into an autoregressive moving average model.
- For an ARMA model of order *p*-*q*, this is given by:
  - $x_t = \beta_0 + \sum_{j=1}^{p} \beta_j x_{t-j} + \sum_{j=1}^{q} \theta_j \epsilon_{t-j}$.
- We have *many* more parameters now that we previously did, and have to determine both *p* and *q.* However, this formulation can often lead to simpler interpretations than either model separately.
- Can be included with (multiple) differencing to Autoregressive Integrated Moving Average (ARIMA) model.

BUILDING SEASONALITY INTO THE MODEL

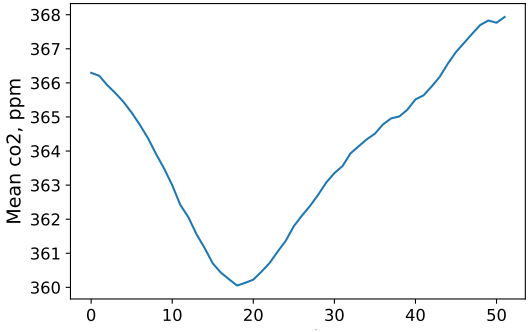
Accounting for Seasonality

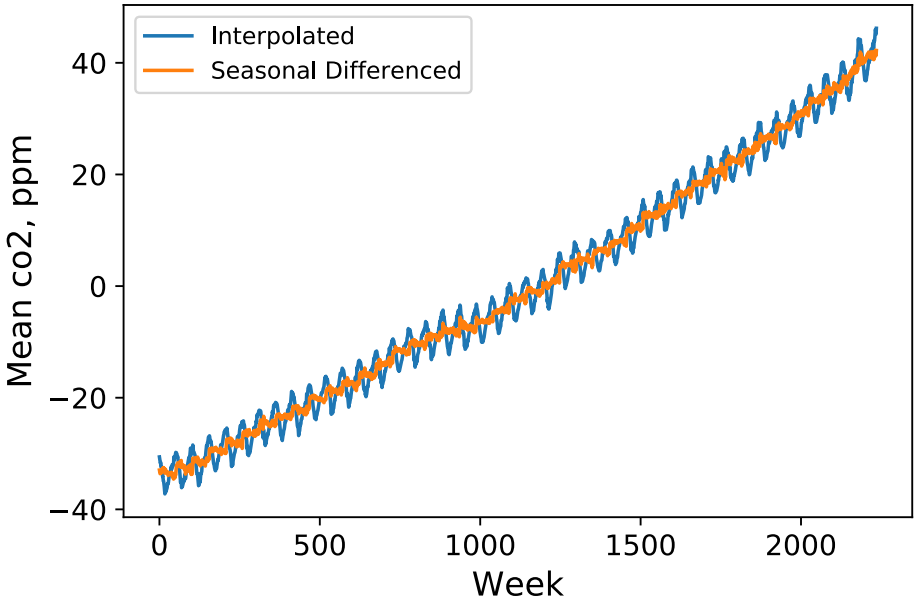
- Often we know the periodicity of the time series *a priori*, and we may want to build it in. In a lot of the examples so far, we've simply had yearly trends (or at least primarily).
- One approach to address this is to perform differencing with respect to the mean of all time points in the same index of each period (i.e. the value this January – the mean of all Januaries).
- This type of approach is usually denote as a Seasonal ARIMA (SARIMA) type model. What does this type of approach visually do?

- Note, we often don't have enough data to estimate yearly trends, so it is important to smooth/regularize the estimates of the seasonal trend.

Simple Seasonal Differencing

- This critically depends on the time series being sampled exactly on the same period. Then one can just define the seasonality period, and define the mean and subtract it.
- If the time series isn't on the exact same period, it can be interpolated.

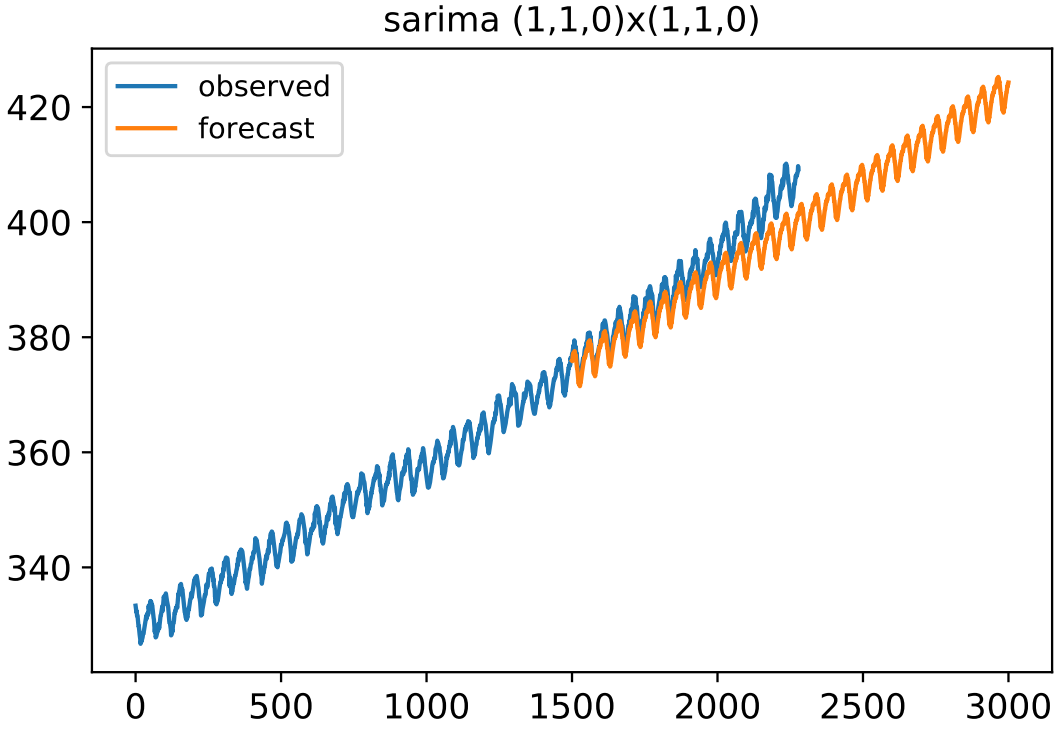




SARIMA

Explicitly using seasonality increases our ability to correctly forecast errors.

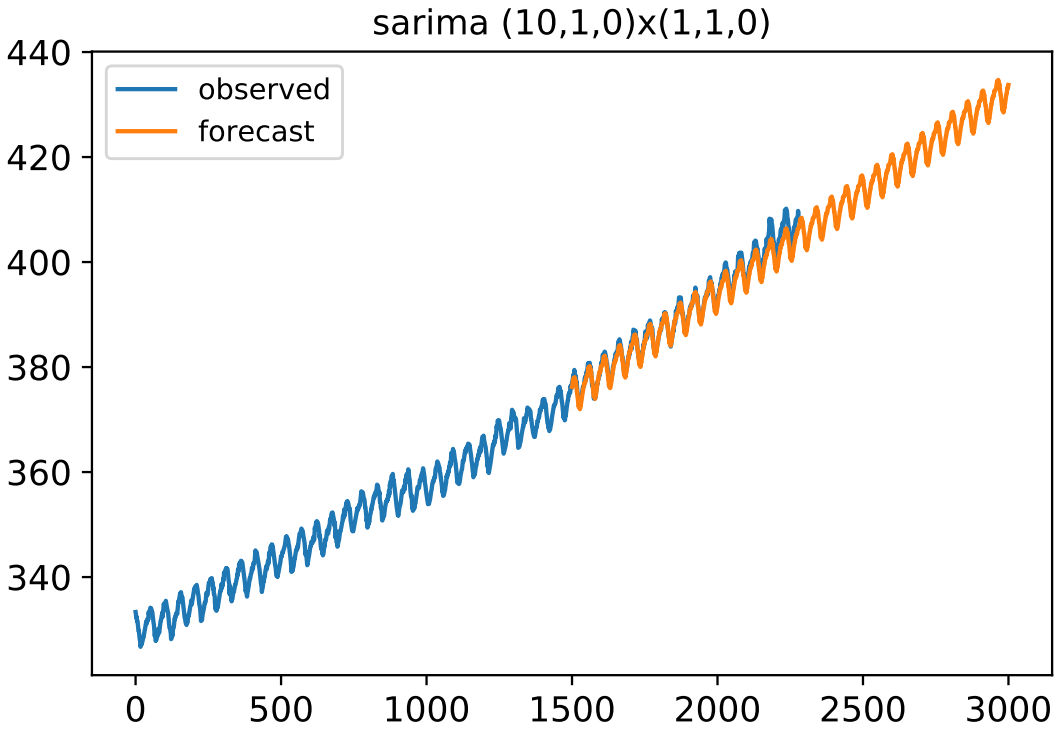
Note that the model is now a *very low* order/complexity and does a good job forecasting the data.



SARIMA

Explicitly using seasonality increases our ability to correctly forecast errors.

A higher order model does a little better

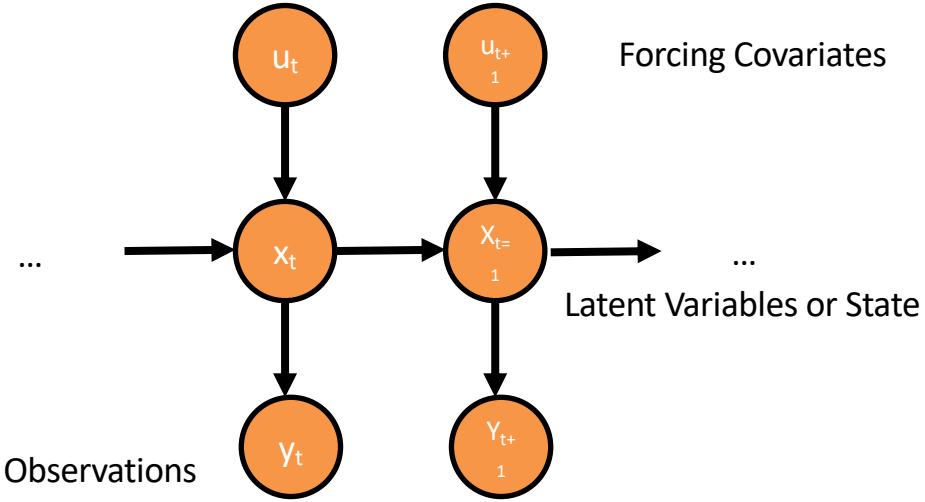


THE KALMAN FILTER

Kalman Filter:

- Many other names for the Kalman filter, such as linear quadratic estimation. We'll start with a one dimensional formulation, and then move to higher dimensions.
- Essentially, the Kalman filter begins by defining a *latent (unobserved)* set of dynamics on the latent space.
- Let $\boldsymbol{y}_{1:T}$ be defined as our observations, and $\boldsymbol{x}_{1:T}$ is our latent, unobserved process. The simplest evolution of the latent process is modeled as:
  - $x_t = a \cdot x_{t-1} + b \cdot u_t + \nu_t.$
- Where u_t is an input control vector and ν_t is a random noise (or an injected evolution term).
- The observation is modeled as:
  - $y_t = h \cdot x_t + \epsilon_t.$

Visualization of the Kalman Filter Graph



Why do we want a Kalman Filter?

- With the Kalman filter, we can track the underlying state.
- Easily handles missing data (assuming missing completely at random).
- Can forecast and handle missing data.
- Still assumes linearity.
- The ARIMA model class is subsumed by a Kalman Filter.

- The Kalman Filter explicitly handles *external inputs.*
  - Note that there are extensions to do this with the SARIMA model as well (in particular, the SARIMAX model)

Issue: Latent Variables are not Observed

- We don't know what $\boldsymbol{x}_{1:T}$ is because it was never observed.
- It must be observed. First, let's consider a case where everything is assumed Gaussian. Then assume that we have a distribution over
  - $x_{t-1} \sim N(\mu_{t-1}, \sigma_{t-1}^2)$.
- Using this formulation with the forward equation
  - $x_t = a \cdot x_{t-1} + b \cdot u_t + \nu_t$.
- We can state:
  - $p(x_t|x_{t-1}) \sim N(ax_{t-1} + bu_t, +\sigma_x^2)$
  - $p(x_t|\mu_{t-1}\sigma_{t-1}^2) \sim N(a\mu_{t-1} + bu_t, a^2\sigma_{t-1}^2 + \sigma_x^2)$

Included Observed Information

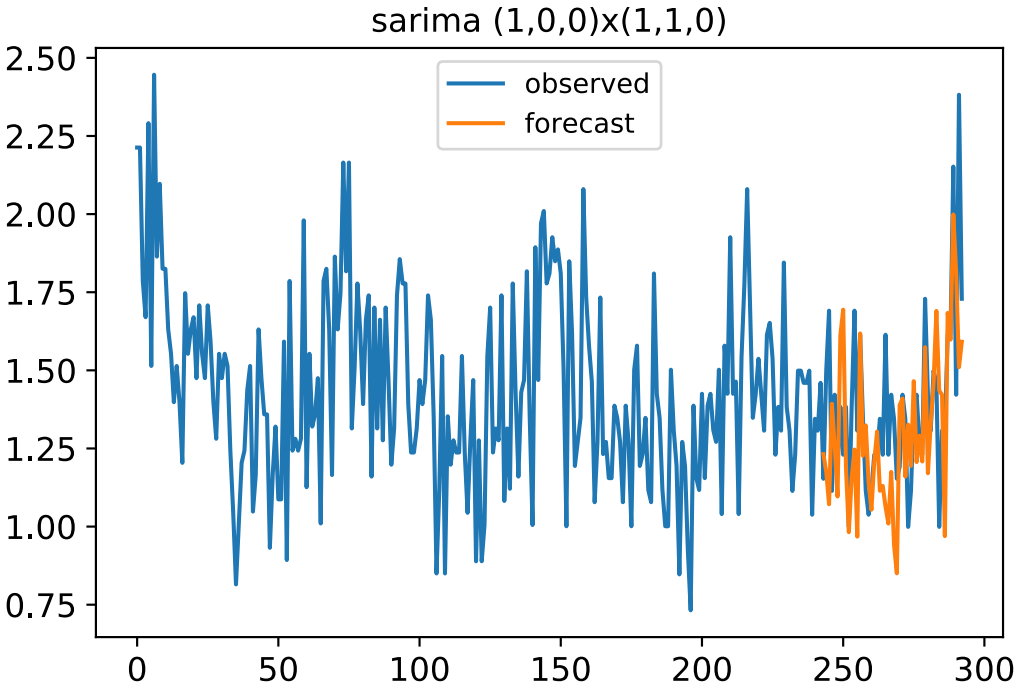
- We also have an observed information, where
  - $y_t = h \cdot x_t + \epsilon_t$.
- We can incorporate this information, too, via Bayes' law:
  - $p(x_t|\mu_{t-1}\sigma_{t-1}^2) \sim N(a\mu_{t-1} + bu_t, a^2\sigma_{t-1}^2 + \sigma_x^2)$

    $p(y_t|x_t) \sim N(hx_t, \sigma_y^2)$
- Which yields:
  - $p(x_t|y_t, \mu_{t-1}, \sigma_{t-1}^2) \sim N(\mu_{t|t}, \sigma_{t|t}^2)$
- Where $\mu_{t|t}$ and $\sigma_{t|t}^2$ are derived by posterior updates.
- Note: if there is no observation at time *t*, then the observation is not included and only $p(x_t|\mu_{t-1}\sigma_{t-1}^2)$ is used to propagate the model forward.

Extending to Higher-Dimensional Spaces

- Instead of the original formulation, we can extend the univariate case to high-dimensional cases, with:
  - $\boldsymbol{x}_t = \boldsymbol{A}\boldsymbol{x}_{t-1} + \boldsymbol{B}\boldsymbol{u}_t + \boldsymbol{\nu}_t.$
- Where u_t is an input control vector and ν_t is a random noise (or an injected evolution term).
- The observation is modeled as:
  - $\boldsymbol{y}_t = \boldsymbol{H}\boldsymbol{x}_t + \boldsymbol{\epsilon}_t.$
- In this more general formulation, capable of representing many things.
- Common scientific approach is to structure $\boldsymbol{A}$ to capture properties of interest (e.g. rotational dynamics)

Visualization of a Kalman Filter



INCLUDING “EXTERNAL” INPUTS

Does including external inputs help us?

- Often, we have extra information about what's happening outside the time series, and we think that there's some link between the external information and the time series.

- This can help us in two ways:
  - Can improve forecasting
  - Reveals statistical links between information sources

Conclusions

- Statistical modeling is very common in time series
- These types of models are very useful in modeling analysis
- There are *tradeoffs* with the machine learning models discussed before
  - Can't handle non-linearities (highly non-linear time series will work much better with machine learning methods)
  - Interpretable model <- can possibly relate to actual physical processes