Time Series Modeling

GA DAT3

Agenda

- Time Series Modeling
- What Are The Time Series Models?
- Conclusion

Time Series Modeling

TIME SERIES MODELING

- In the last class, we focused on exploring time series data and common statistics for time series analysis.
- In this class, we will advance those techniques to show how to predict or forecast forward from time series data.
- With a sequence of values (a time series), we will use the techniques in this class to predict a future value.

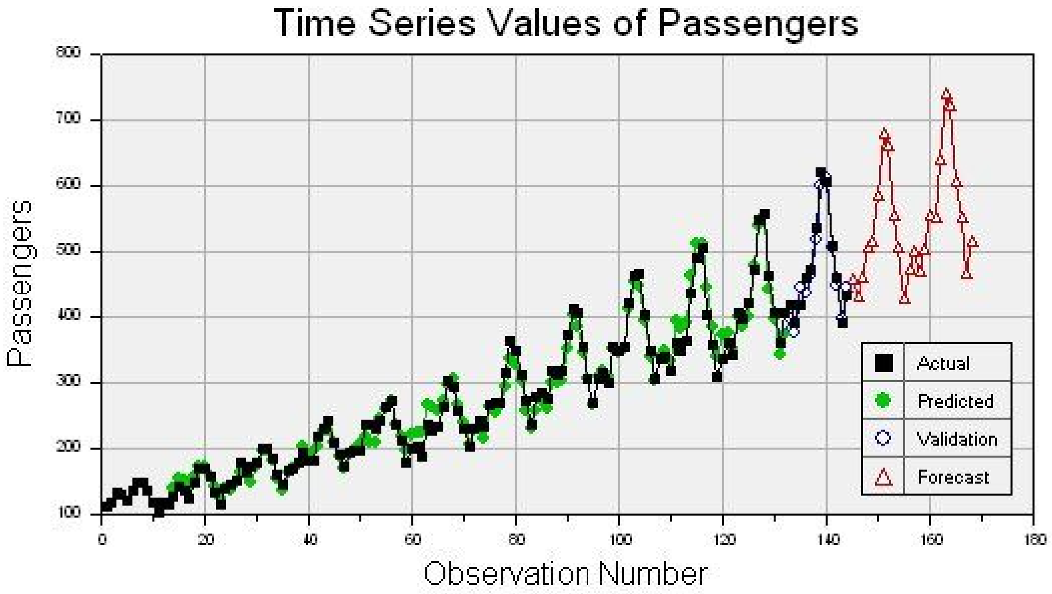
TIME SERIES MODELING

- There are many times when you may want to use a series of values to predict a future value.
- The number of sales in a future month
- Anticipated website traffic when buying a server
- Financial forecasting
- The number of visitors to your store during the holidays

What Are The Time Series Models

WHAT ARE TIME SERIES MODELS?

- Time series models are models that will be used to predict a future value in the time series.
- **Like** other predictive models, we will use prior history to predict the future.
- **Unlike** previous models, we will use the earlier in time *outcome* variables as *inputs* for predictions.



WHAT ARE TIME SERIES MODELS?

- **Like** previous modeling exercises, we will have to evaluate the different types of models to ensure we have chosen the best one.
- We will want to evaluate on a held-out set or test data to ensure our model performs well on unseen data.

WHAT ARE TIME SERIES MODELS?

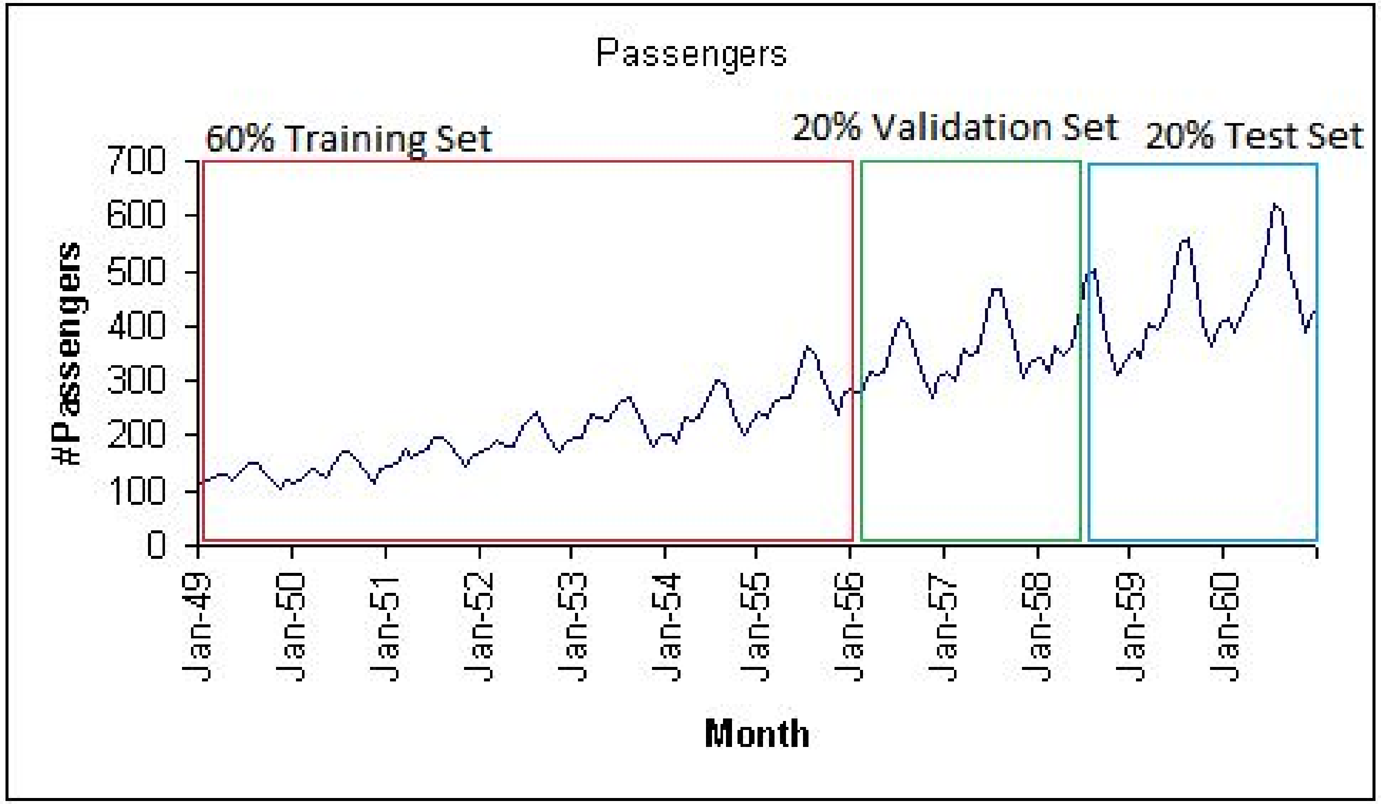
- **Unlike** previous modeling exercises, we won’t be able to use standard cross-validation for evaluation.
- Since there is a time component to our data, we cannot choose training and test examples at random.
- Suppose we did select a random 80% sample of data points for training and a random 20% for testing. What could go wrong?

WHAT ARE TIME SERIES MODELS?

- The training dataset would likely contain data from *before* AND *after* a test dataset.
- This would not be possible in real life (you can't use future, unseen data points when building your model). Therefore, it's not a valid test of how our model would perform in practice.

WHAT ARE TIME SERIES MODELS?

- Instead, we will exclusively train on values earlier (in time) in our data and test our model on values at the end of the data period.



PROPERTIES FOR TIME-SERIES PREDICTION

- A *moving average* is an average of *p* surrounding data points in time.

... to *t - p + 1*. This includes t, t + 1, t + 2, ..., t-p, t-p+1.

$$F_t = \frac{1}{p} \sum_{k=t}^{t-p+1} Y_k$$

Divide by the *p* surrounding data points

Get the *p* points from *t* ...

PROPERTIES FOR TIME-SERIES PREDICTION

- *Autocorrelation* is how correlated a variable is with itself. Specifically, how related are variables earlier in time with variables later in time.

$$r_k = \frac{\sum_{t=k+1}^{n} (y_t - \bar{y})(y_{t-k} - \bar{y})}{\sum_{t=1}^{n} (y_t - \bar{y})^2}$$

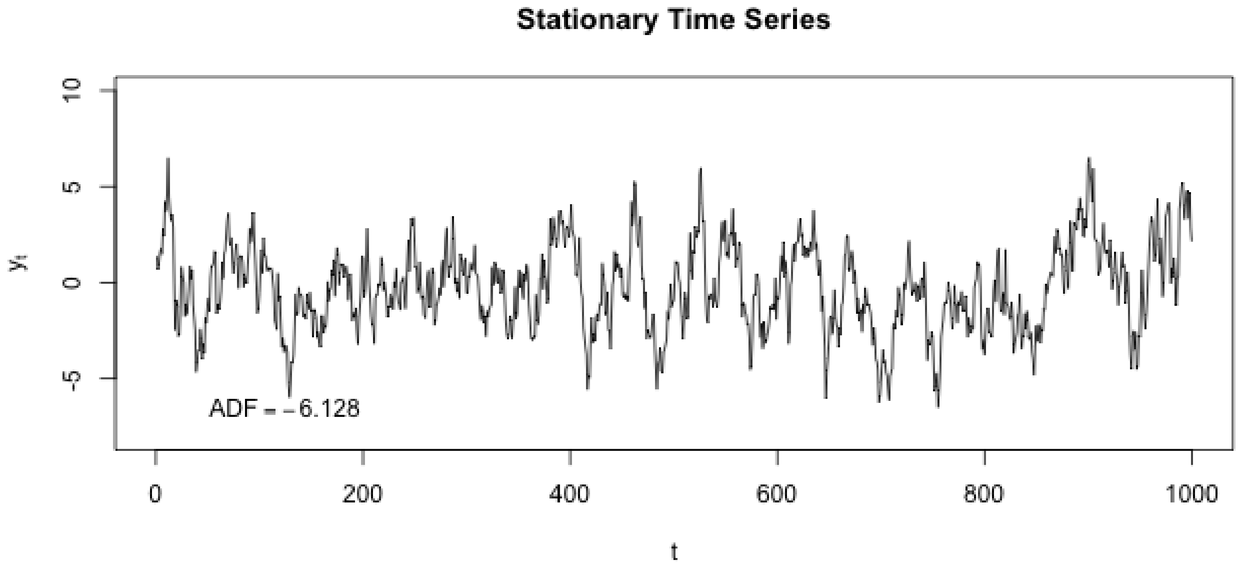
PROPERTIES FOR TIME-SERIES PREDICTION

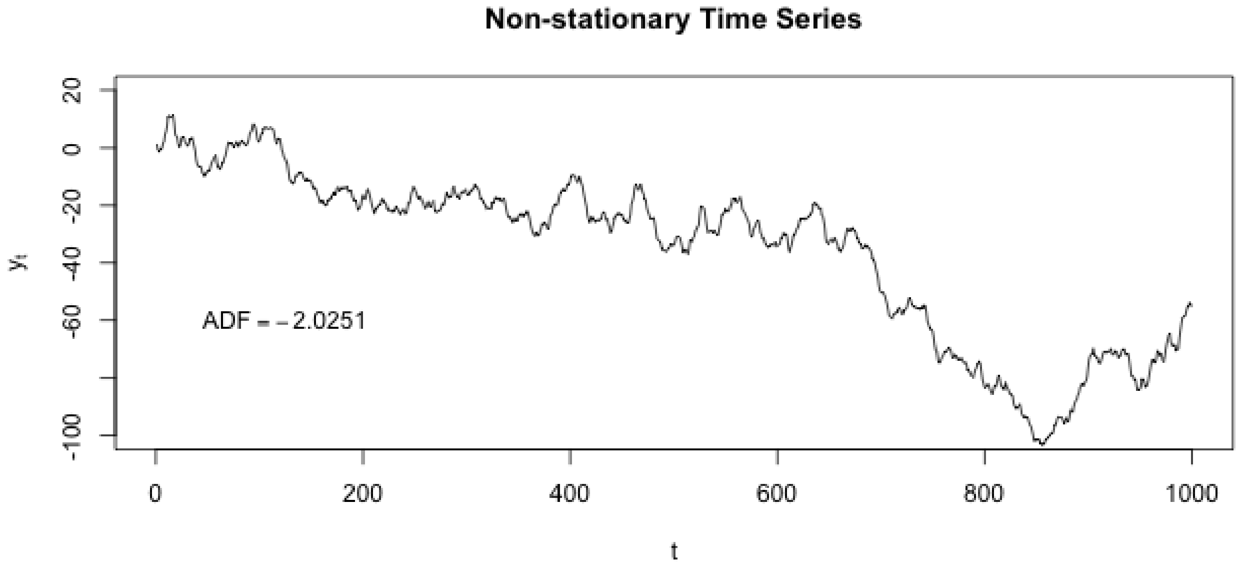
- We can use these values to assess how we plan to model our time series.
- Typically, for a high quality model, we require some autocorrelation in our data.
- We can compute autocorrelation at various lag values to determine how far back in time we need to go.

PROPERTIES FOR TIME-SERIES PREDICTION

- Many models make an assumption of *stationarity*, assuming the mean and variance of our values is the *same* throughout.
- While the values (e.g. of sales) may shift up or down over time, the mean and variance of sales is constant (i.e. there aren't many dramatic swings up or down).
- These assumptions may not represent real world data; we must be aware of that when we are breaking the assumptions of our model.

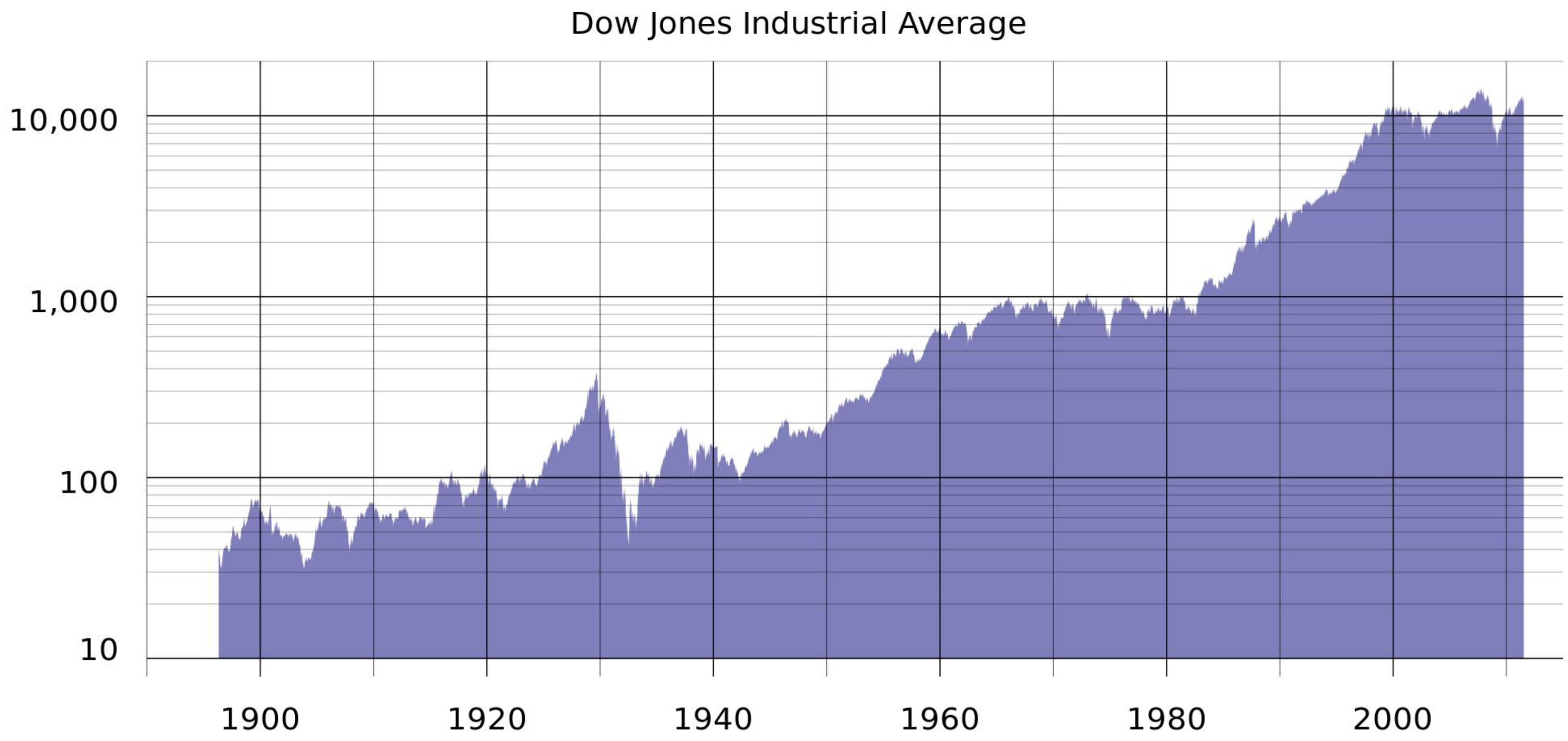
PROPERTIES FOR TIME-SERIES PREDICTION





PROPERTIES FOR TIME-SERIES PREDICTION

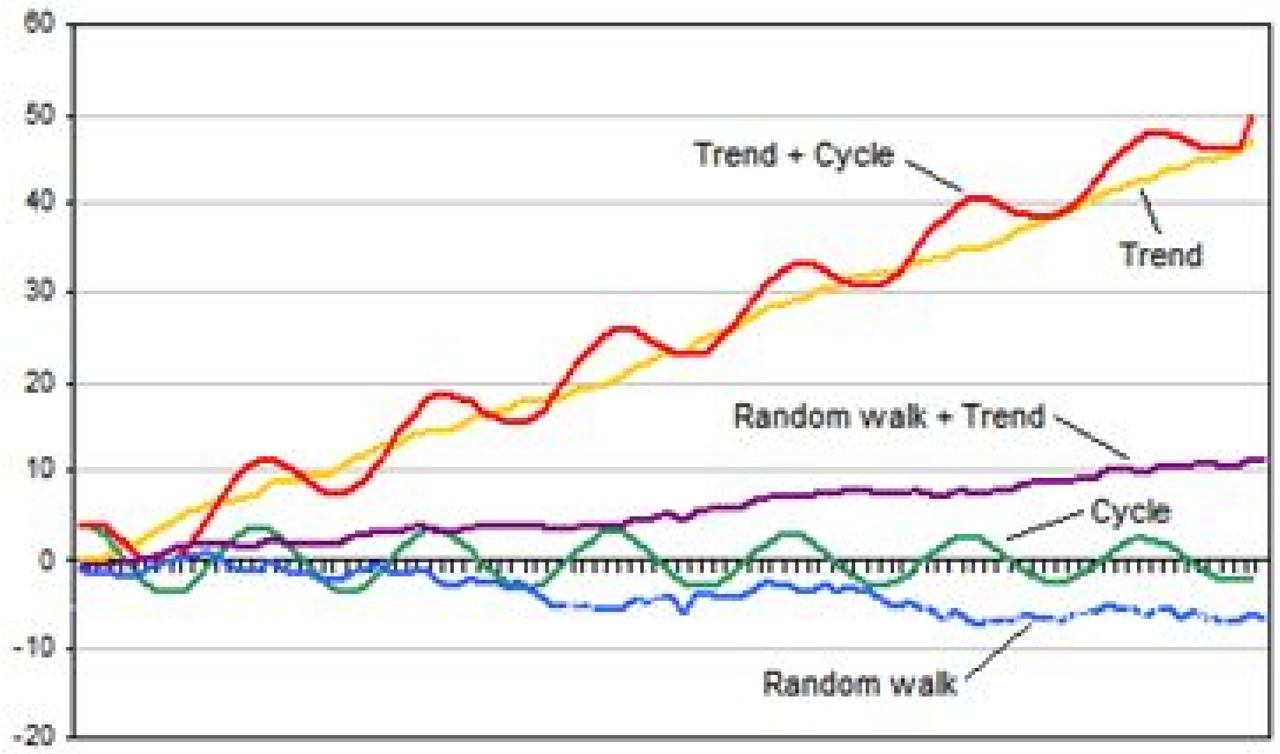
- For example, typical stock or market performance is not stationary. In this plot of Dow Jones performance since 1986, the mean is clearly increasing over time.



PROPERTIES FOR TIME-SERIES PREDICTION

- Below are simulated examples of non-stationary time series and why they might occur.

Table 1 Non-stationary behavior

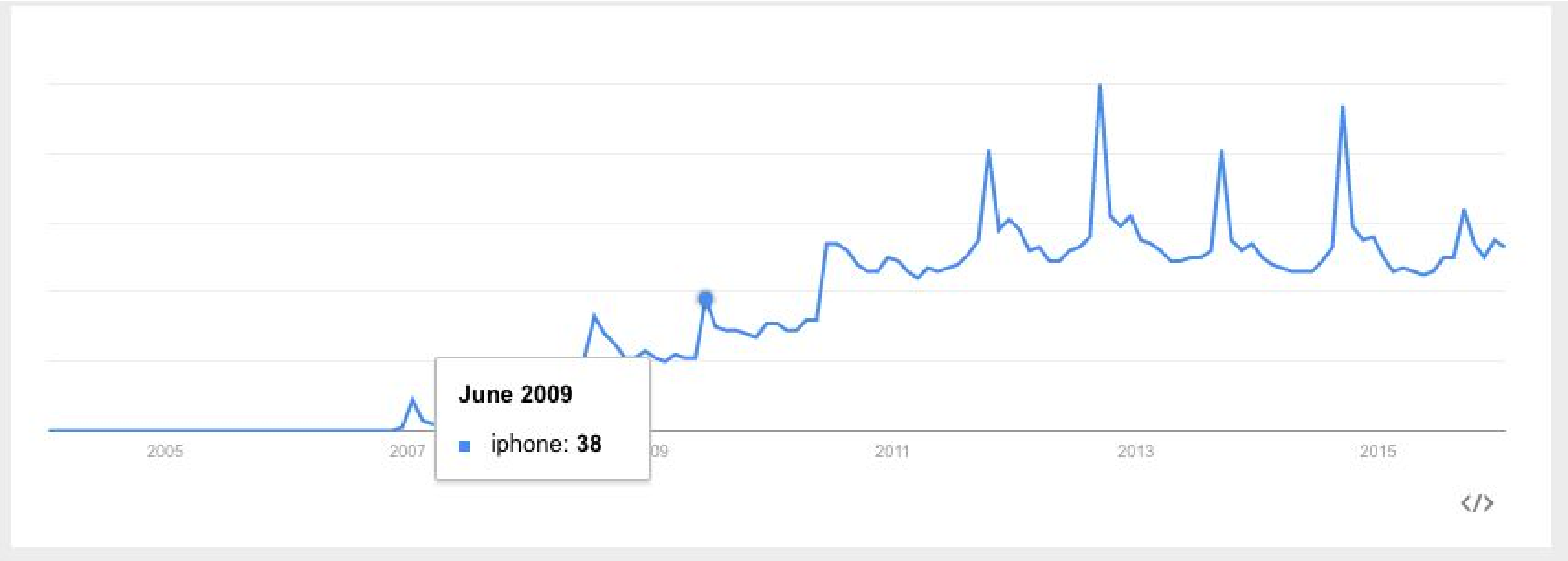


PROPERTIES FOR TIME-SERIES PREDICTION

- Often, if these assumptions don't hold, we can alter our data to make them true. Two common methods are *detrending* and *differencing*.
- *Detrending* would mean to remove any major trends in our data.
- We could do this is many ways, but the simplest is to fit a line to the trend and make a new series that is the difference between the line and the true series.

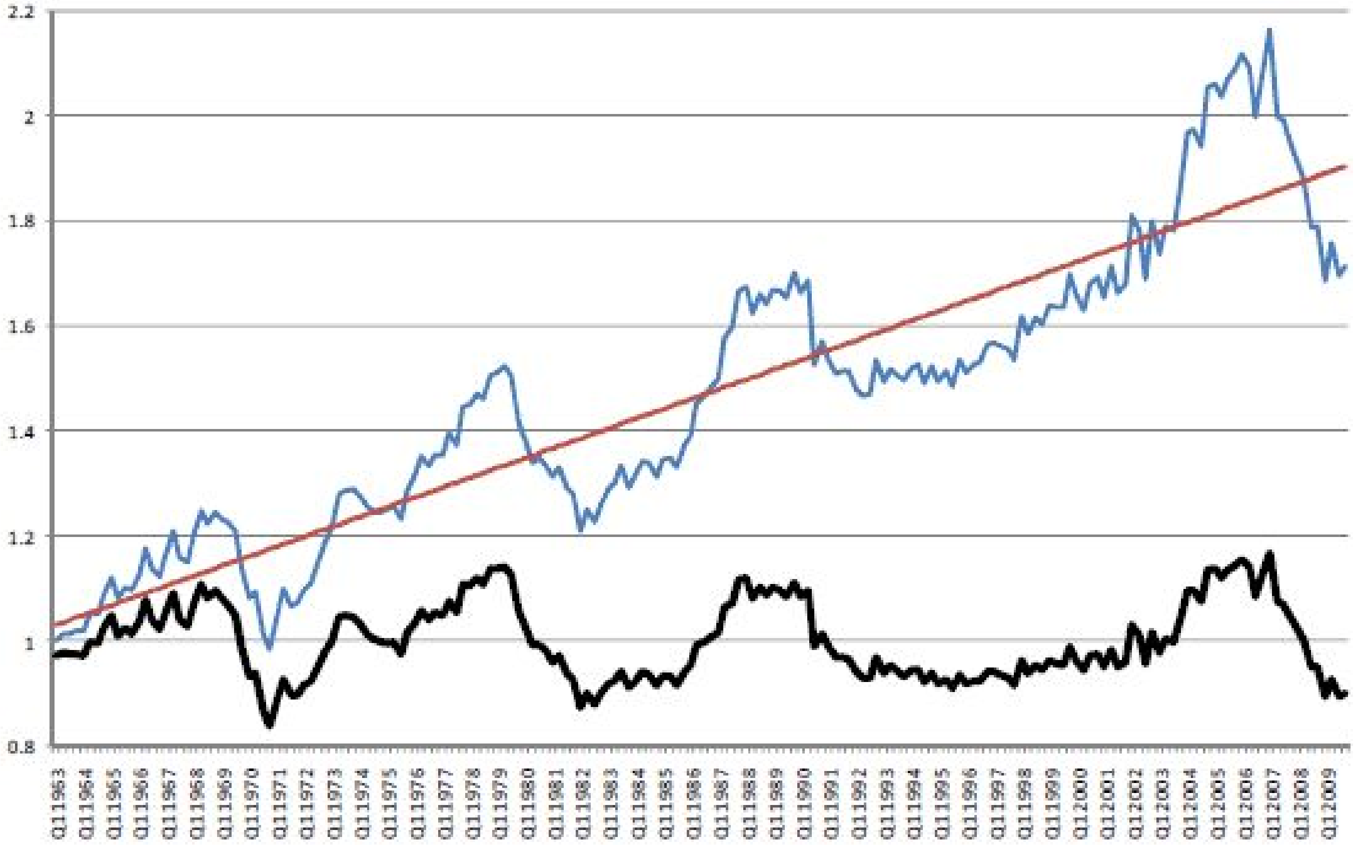
PROPERTIES FOR TIME-SERIES PREDICTION

- For example, there is a clear upward (non-stationary) trend in google searches for “iphone”.



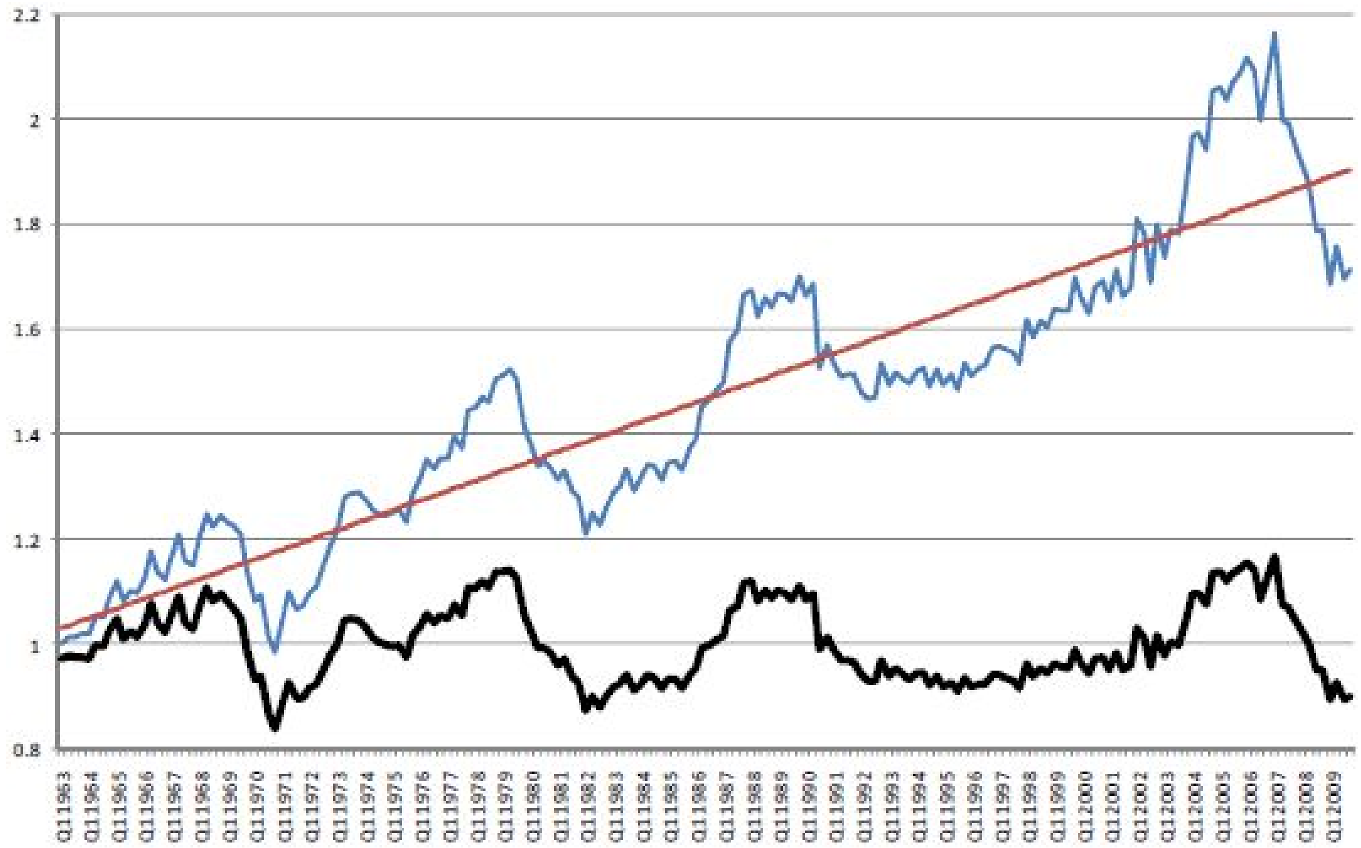
PROPERTIES FOR TIME-SERIES PREDICTION

- Below is an example where we look at US housing prices over time. Clearly, there is an upward trend, making the time series non-stationary (ie: the mean house price is increasing).



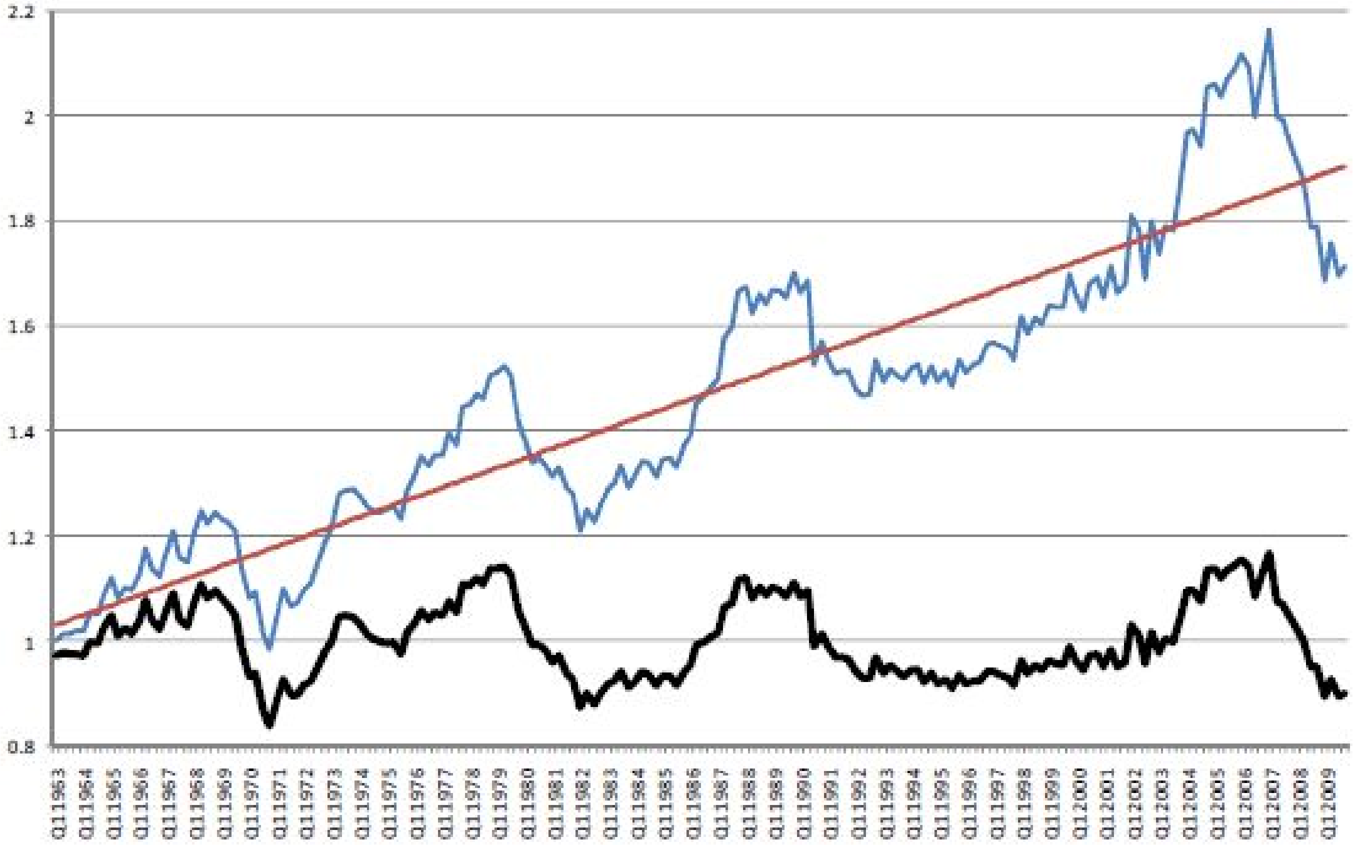
PROPERTIES FOR TIME-SERIES PREDICTION

- We can fit a line that represents the trend. With our trend line, we can subtract the trend line value from the original value to get the bottom figure.



PROPERTIES FOR TIME-SERIES PREDICTION

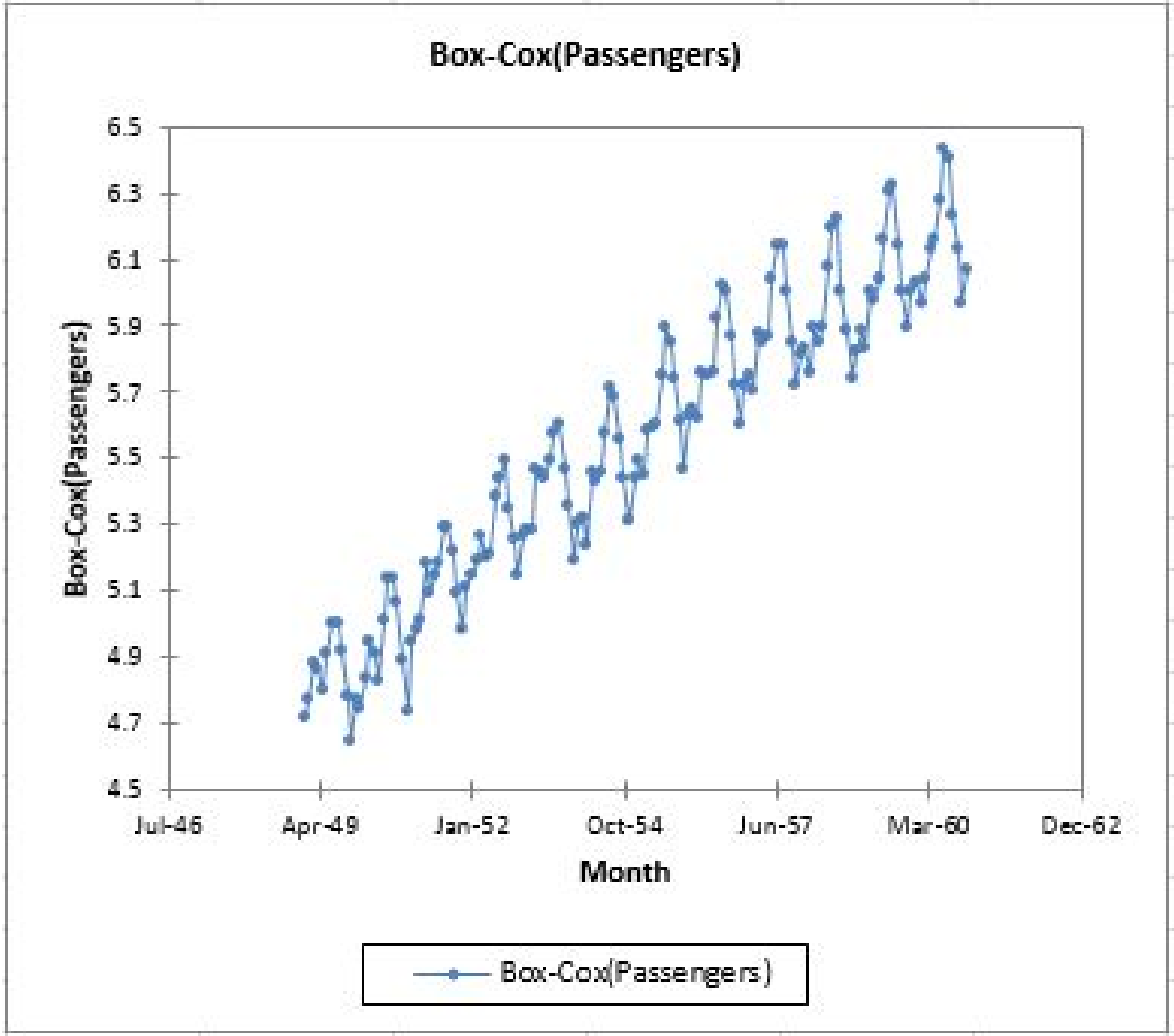
- The data now has a fixed mean and will be easier to model. This pattern is similar to mean-scaling our features in earlier models with `StandardScaler`.

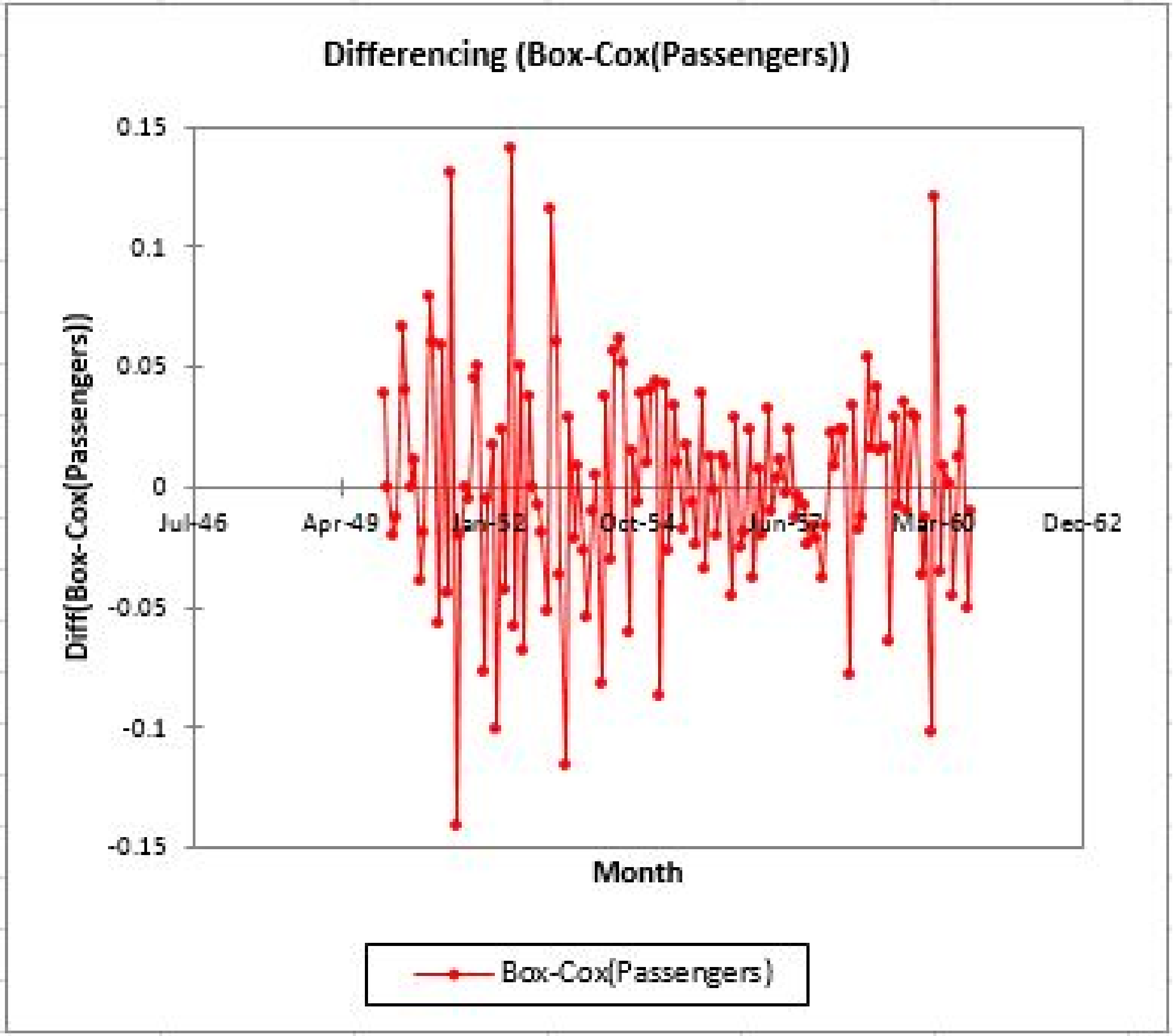


PROPERTIES FOR TIME-SERIES PREDICTION

- A simpler method is *differencing*. This is very closely related to the `diff` function we saw in the last class.
- Instead of predicting the series (again our non-stationary series), we can predict the difference between two consecutive values.

PROPERTIES FOR TIME-SERIES PREDICTION





ACTIVITY: KNOWLEDGE CHECK

EXERCISE

ANSWER THE FOLLOWING QUESTIONS

Non-stationary data is the most common; almost any interesting dataset is non-stationary.

1. Can you think of some interesting datasets that might be stationary?

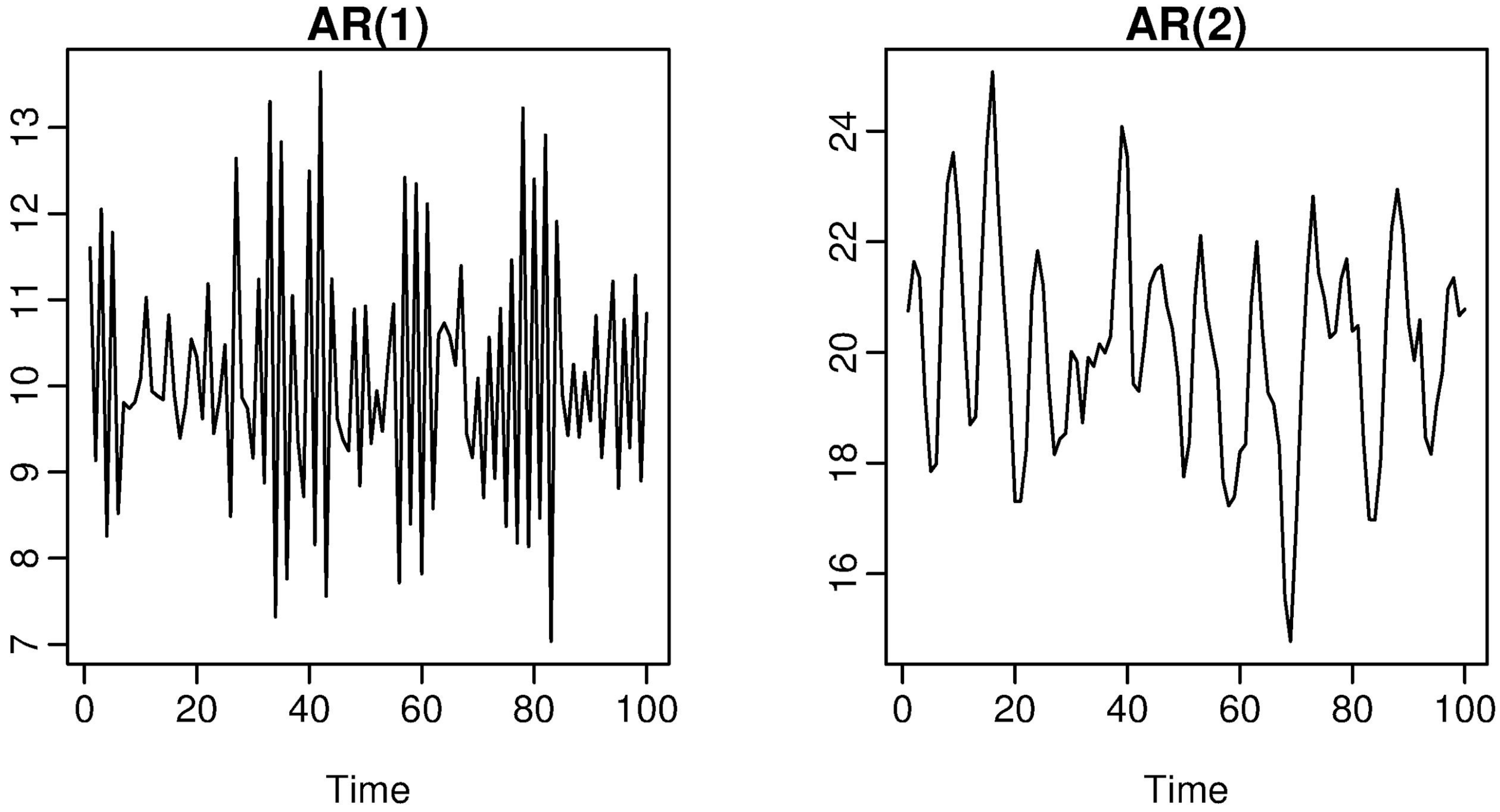
DELIVERABLE

Answers to the above questions

AR MODELS

- Autoregressive (AR) models are those that use data from previous time points to predict the next.
- This is very similar to previous regression models, except as input, we take the previous outcome.
- If we are attempting to predict weekly sales, we use the sales from a previous week as input.
- Typically, AR models are notes AR(p) where *p* indicates the number of previous time points to incorporate, with AR(1) being the most common.

AR MODELS



AR(1)
7
8
9
10
11
12
13
0
20
40
60
80
100
Time
AR(2)
16
18
20
22
24
0
20
40
60
80
100
Time

AR MODELS

- In an autoregressive model, similar to standard regression, we are learning regression coefficients for each of the *p* previous values. Therefore, we will learn *p* coefficients or **β** values.

- If we have a time series of sales per week, y_i, we can regress each yi from the last *p* values.

$$y_i = \boldsymbol{\beta}_0 + \boldsymbol{\beta}_1 y_{i-1} + \boldsymbol{\beta}_2 y_{i-2} + \ldots + \boldsymbol{\beta}_p y_{i-p} + \varepsilon$$

- As with standard regression, our model assumes that each outcome variable is a linear combination of the inputs and a random error term.

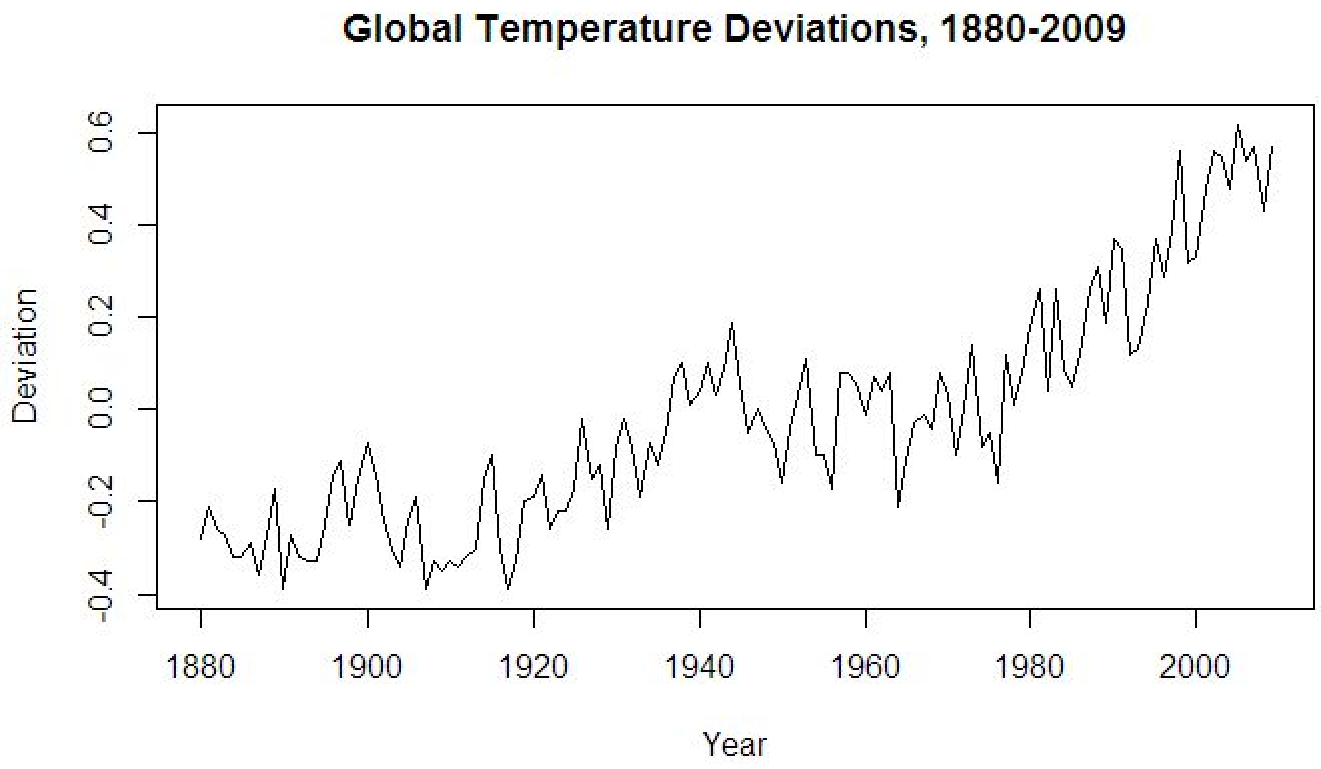
AR MODELS

- For an AR(1) model, we will learn a single coefficient.
- This coefficient, $\boldsymbol{\beta}$, will tell us the relationship between the previous value, $Y_{t - 1}$, and the next value, Y_t.

$$Y_t = \boldsymbol{\beta} \cdot Y_{t - 1}$$

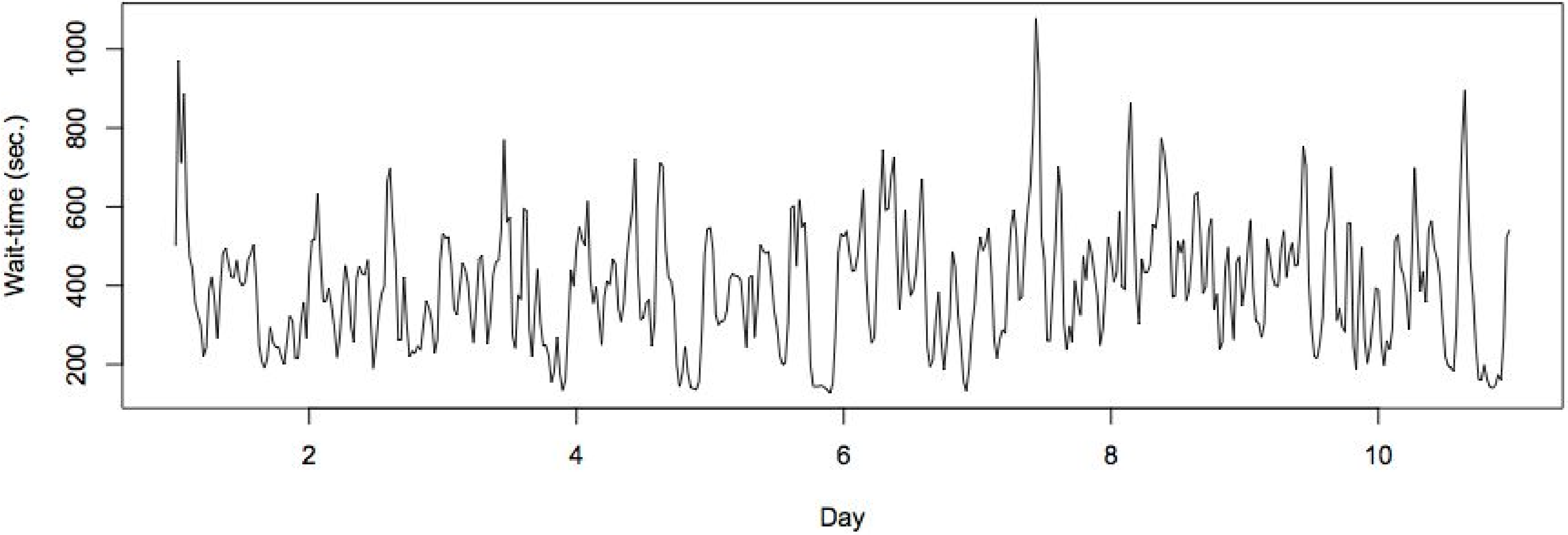
AR MODELS

- A value > 1 would indicate a growth over previous values. This would typically represent non-stationary data, since if we compound the increases, the values are continually increasing.



AR MODELS

- Values between 1 and -1 represent increasing and decreasing patterns from previous patterns.



AR MODELS

- As with other models, interpretation of the model becomes more complex as we add more factors.

- Going from AR(1) to AR(2) can add significant *multi-collinearity*.

AR MODELS

- Recall that *autocorrelation* is the correlation of a value with its series *lagged* behind.
- A model with high correlation implies that the data is highly dependent on previous values and an autoregressive model would perform well.

AR MODELS

- Autoregressive models are useful for learning falls or rises in our series.
- This will weight together the last few values to make a future prediction.
- Typically, this model type is useful for small-scale trends such as an increase in demand or change in tastes that will gradually increase or decrease the series.

MA MODELS

- **Moving average (MA) models**, as opposed to AR models, do not take the previous outputs (or values) as inputs. They take the previous error terms.

- We will attempt to predict the next value based on the overall average and how off our previous predictions were.

MA MODELS

- This model is useful for handling specific or abrupt changes in a system.
- AR models slowly incorporate changes in the system by combining previous values; MA models use prior errors to quickly incorporate changes.
- This is useful for modeling a sudden occurrence - something going out of stock or a sudden rise in popularity affecting sales.

MA MODELS

- As in AR models, we have an order term, *q*, and we refer to our model as MA(q). The moving average model is dependent on the last *q* errors.

- If we have a time series of sales per week, y_i, we can regress each y_i from the last *q* error terms.

$$y_i = \text{mean} + \boldsymbol{\beta}_1 \boldsymbol{\varepsilon}_{i-1} + \boldsymbol{\beta}_2 \boldsymbol{\varepsilon}_{i-2} + \ldots + \boldsymbol{\beta}_q \boldsymbol{\varepsilon}_{i-q}$$

- We include the mean of the time series (that’s why it’s called a moving average) as we assume the model takes the mean value of the series and randomly jumps around it.

MA MODELS

- Of course, we don’t have error terms when we start - where do they come from?
- This requires a more complex fitting procedure than we have seen previously.
- We need to iteratively fit a model (perhaps with random error terms), compute the errors and then refit, again and again.

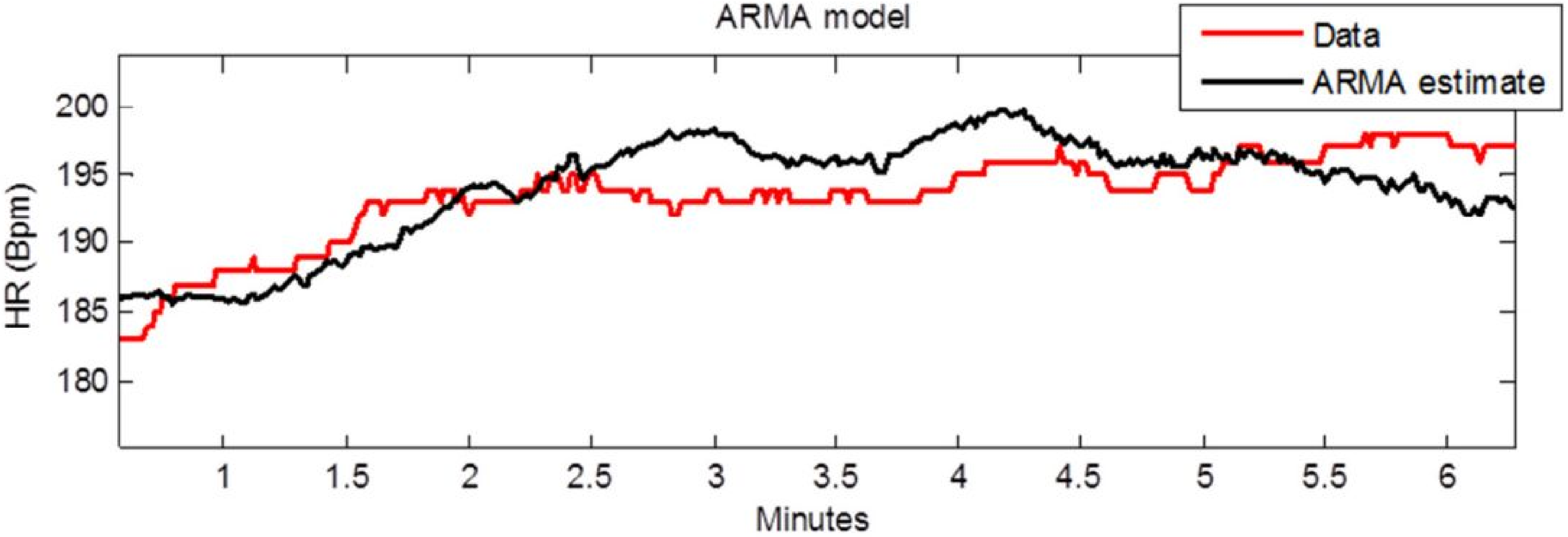
MA MODELS

- In this model, we learn q coefficients.
- In an MA(1) model, we learn one coefficient.
- This value indicates the impact of how our previous error term on the next prediction.

ARMA MODELS

- **ARMA** (pronounced ‘R-mah’) models combine the autoregressive and moving average models.

- An ARMA(p,q) model is simply a combination (sum) of an AR(p) model and MA(q) model.



ARMA MODELS

- We specify two model settings, *p* and *q*, which correspond to combining an AR(p) model with an MA(q) model.

- Incorporating both models allows us to mix two types of effects.

- AR models slowly incorporate changes in preferences, tastes, and patterns.

- Moving average models base their prediction on the prior error, allowing to correct sudden changes based on random events - supply, popularity spikes, etc.

ARIMA MODELS

- **ARIMA** (pronounced ‘uh-ri-mah’) is an **A**uto**R**egressive **I**ntegrated **M**oving **A**verage model.

- In this model, we learn an ARMA(p,q) model to predict *the difference* of the series (as opposed to the value of the series).

ARIMA MODELS

- Recall the pandas `diff` function. This computes the difference between two consecutive values.

- In an ARIMA model, we attempt to predict this difference instead of the actual values.

$$\mathbf{y_t - y_{t-1} = ARIMA(p,q)}$$

- This handles the stationarity assumption we wanted for our data. Instead of detrending or differencing manually, the model does this.

ARIMA MODELS

- An ARIMA model has three parameters and is specified ARIMA(p, d, q).
- *p* is the order of the autoregressive component
- *q* is the order of the moving average component
- *d* is the degree of differencing.
  - *d* was 1 in our prior example. For d=2, our model would be

$$\text{diff}(\text{diff}(y)) = (y_t - y_{t-1}) - (y_{t-1} - y_{t-2}) = \text{ARIMA}(p,q)$$

ARIMA MODELS

- Compared to an ARMA model, ARIMA models do **not** rely on the underlying series being stationary.
- The differencing operation can *convert* the series to one that is stationary.
- Instead of attempting to predict values over time, our new series is the difference in values over time.
- Since ARIMA models include differencing, they can be used on a broader set of data without the assumption of a constant mean.

Conclusion

CONCLUSION

- Time-series models use previous values to predict future values, also known as forecasting.
- AR and MA model are simple models on previous values or previous errors respectively.
- ARMA combines these two types of models to account for both gradual shifts (due to AR models) and abrupt changes (MA models).

CONCLUSION

- ARIMA models train ARMA models on differenced data to account for non-stationary data.
- Note that none of these models may perform well for data that has more random variation.
- For example, for something like iphone sales (or searches) which may be sporadic, with short periods of increases, these models may not work well.

Q??