# Lag Features

1. Motivation for lag features

* We want to predict future values of the target
* Past values of the target are likely to be predictive
* Past values of a feature could also be predictive (e.g., the sales on a day is related to ad spend on prior days)

1. Lag features

* A lag feature is the value of the target or feature k period(s) in the past
* k is the lag, set by user

A table with numbers and arrows

Description automatically generated with medium confidence

A table with numbers and a few words

Description automatically generated with medium confidence

* We can create multiple lag features with different lags from the target and features
* Problem: which lags to use? How many lag features to create?

1. Lag feature implementation in Pandas

* Pandas.DataFrame.shift

A screenshot of a computer code

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer program

Description automatically generated

1. Lag feature implementation in Feature-engine

* Feature\_engine.timeseries.forecasting.LagFeatures

A screenshot of a computer program

Description automatically generated

A screenshot of a computer

Description automatically generated

1. Summary

* Lag features are a way of using the past to predict the future
* Can lag the target or other features

# How to choose the lags

1. How to choose the lags

* Domain knowledge
* Feature selection and modeling
* Time-series correlation methods

1. Domain knowledge

* If seasonality is known use a lag of the same seasonal order (seasonal lag)
* E.g. retail sales: yearly seasonality -> use lag of 1 year
* E.g. electricity demand: multiple seasonalities such as yearly, weekly, daily -> use lag of 1 year, 1 week, and 1 day
* Most recent values tend to be predictive -> use small lag
* When lagging the features
* Use the subset of features you think are most important in affecting the target
* Only use the value of the feature that is known at predict time to avoid data leakage
* Most recent values tend to be predictive -> use small lags
* Pros:
* Likely to result in fewer additional features as we will pick a small number of lags and features known to be important
* Cons:
* We may not know all the seasonal patterns or which ones are most important
* We may not know which features are important to lag
* Not scalable

1. Feature selection and modeling

* Create a bunch of different lags which are reasonable given the feature and use case
* Use feature selection and/or modeling to best utilize the feature and determine a subset which minimizes forecast error
* Pros:
* Automatic – less hands-on decision making
* May find useful features which you may not have been used otherwise
* Cons:
* Will create very large number of features
* The lags of the same features will be highly correlated to each other
* More complex model than necessary
* Computationally expensive

1. Time series correlation methods

* The main idea
* Measure how corelated the lag features are with the target
* If the lag feature is highly correlated to the target then it might be helpful
* 3 main methods
* Autocorrelation function (ACF)
* Partial autocorrelation function (PACF)
* Cross-correlation function (CCF)
* Pros
* More robust way to find relevant lags
* Can indicate whether there is any predictive info in the historic time series at all
* Can help identify important seasonalities
* Cons:
* Can be difficult to interpret correlation plots
* Time consuming to interpret and read correlation plots -> not scalable to large number of features
* Even if one feature is not highly correlated with the target it could still be predictive in the presence of other features -> not captured in these methods
* These methods only measure linear relationships between variables

1. Summary

* Lags of the target and other features can create predictive features for forecasting
* User must decide which lag features to create
* Domain knowledge, feature selection & modeling, time series correlation methods can help in selection of lag

# Autoregressive (AR) processes

1. Motivation

* We will want to show how tools we introduce later (e.g. lag plots, correlation functions) behave for time series with various properties
* Three properties already covered in the course so far: trend, seasonality, white noise
* New property of a time series
* Autoregressive (AR) property

1. Scope

* AR processes are a large topic covered in the theory of time series analysis and is broader topic than just forecasting.
* In scope:
* Definitions of an AR process
* Intuition about the behavior of AR processes
* Out of scope:
* Mathematical proofs, derivations, theorems about AR processes

1. White noise

* White noise has no predictive info in past values – no correlation at any 2 points in time

1. Autoregressive (AR) processes

* Generate a time series , which is determined only from the previous values , a constant , and some white noise :
* An AR(1) process as it depends only on a lag of 1

A computer screen shot of a program

Description automatically generated

* **If**

A graph showing the time of a graph

Description automatically generated with medium confidence

A graph showing a graph of time

Description automatically generated with medium confidence

* Lag component coef is now smaller -> noise component makes it look noisier
* **If ,** the time series settle at a new baseline. This follows from the fact that the mean of an AR(1) process when is given by

A graph showing the time and time

Description automatically generated with medium confidence

* **If** , the time series grow exponentially

A graph with numbers and a line

Description automatically generated

* **If** , the time series grow exponentially but oscillates from a negative to positive value over each iteration

A graph with numbers and a line

Description automatically generated

* **If ,** the time series grow linearly – all we are doing is adding a factor and some noise to the previous time step

A graph with a line

Description automatically generated

1. AR(1) summary

* Correlated to the most recent lag (i.e., lag of 1)
* : exponential growth, time series is not stationary
* : linear grow, time series is not stationary
* : varies around a mean value, time series is stationary
* : oscillates between positive and negative values, stationary if
* We can use an AR(1) process to generate time series where future values are correlated to past values. This allows us to test methods which identify whether a lag of 1 is helpful or not

1. AR(p) process

* Time series which depend on more lags
* An AR(p) process depends on p lags and is defined as:
* Recall that an AR(1) process requires to be stationary
* For an AR(p) process there are much more complex requirements on all the coefficients to ensure the process is stationary
* These time series are interesting because by design they depend on multiple lagged values
* Hence, an AR(p) process is a good test case for methods that select which set of lags are important or not.

1. Summary

* An autoregressive (AR) process is a class of time series where future values depend on past values and white noise
* An AR process is determined by prev values and therefore will be correlated to lag values of itself. Hence, lag features should help predict an AR process
* AR processes provide time series which we can use to test methods which identify helpful lags

# Lag plots

1. Scatterplots

* Scatterplots can help identify if 2 variables are related

1. Lag plot

* A scatterplot of a time series against a lagged version of itself

1. What can we learn from a lag plot

* A lag plot is a visual tool which can help show if show a non-random relationship with
* If it does then a lag of k could be a useful feature for forecasting
* We shall look at time series with properties
* White noise
* AR(1) process
* Completely periodic (just seasonality)
* Trend
* Trend and seasonality

1. White noise

* Random time series with no correlation between points
* No predictive info in the historic data
* No strong relationship in the lag plot, as expected from white noise
* This shows us what to look for when determining when a lag may not be useful

A group of squares with blue dots

Description automatically generated

1. AR(1) process

* The time series is determined by the previous lag
* We expect this time series to be correlated to lagged values
* Strong linear correlation between and
* We see the correlation between and its lagged values decay as we look at bigger lags
* a time series which is determined only by a small number of previous lags can generate correlations at multiple lags
* When we discuss PACF, we will see how we can identify that lag 1 is the most important

A group of blue dots in squares

Description automatically generated

1. Seasonality

* A time series which repeats exactly every 12 months
* Any multiple lag of 12 should be the most predictive of future values as the time series is exactly periodic every 12 months

A graph showing the number of seasons

Description automatically generated with medium confidence

* Lag plot:

A graph with blue dots

Description automatically generated

* Strong seasonality at lag 12 indicated by the strong linear correlation
* A lag of 12 could be a helpful feature
* Additional lags:

A group of graphs with numbers and symbols

Description automatically generated with medium confidence

* is exactly periodic with a seasonal period of 12
* There are only 12 unique values that can take
* This causes every lag plot to only have 12 data points in different configurations which repeat every 12 lags
* Important when we discuss the autocorrelation function
* Any seasonal patterns will appear as a strong linear correlation on the lag plot. This occurs at multiplies of the seasonal period (e.g. every 12 lags for yearly seasonality)

1. Trend

* A lot of predictive info in historic data
* When is small so is
* Large positive relationship

A graph with blue and orange lines

Description automatically generated

* Lag plot

A group of blue lines with numbers and letters

Description automatically generated with medium confidence

* Strong linear relationship in the lag plot, as expected
* This is seen across multiple lags because of the overall shape of the original time series means that when is relatively large so is
* The trend causes correlations at many lags – this can make it difficult to identify patterns (e.g. seasonality) which appear as strong correlations only at specific lags
* The lag plot does not provide much info about whether a specific lag will be helpful

1. Trend and seasonality

* A lot of predictive info in historic data
* Will get a combination of effects from the trend, seasonality, and noise in the lag plots
* Lag plot
* The strong trend results in linear relationships across many lags
* The seasonal component is seen in the seasonal lag as a much stronger relationship (lag 12 in this example)

A screenshot of a graph

Description automatically generated

1. Lag plot implementation in Pandas

A screenshot of a computer program

Description automatically generated

1. Limitations

* Lag plots are a visual tool which can help identify useful lags but are not scalable
* If we quantify when is highly correlated with – easier to identify useful lags
* Autocorrelation is a method to quantify the correlation of a time series with itself and can be used to understand properties of a time series including useful lags

1. Summary

* Scatterplots can help identify if 2 variables are related
* Lag plot is a scatterplot of a time series against a lagged version of itself
* Lag plots can identify lags which are strongly related to the original time series
* Trend, seasonality, autocorrelation, and noise leave their own signatures on a lag plot