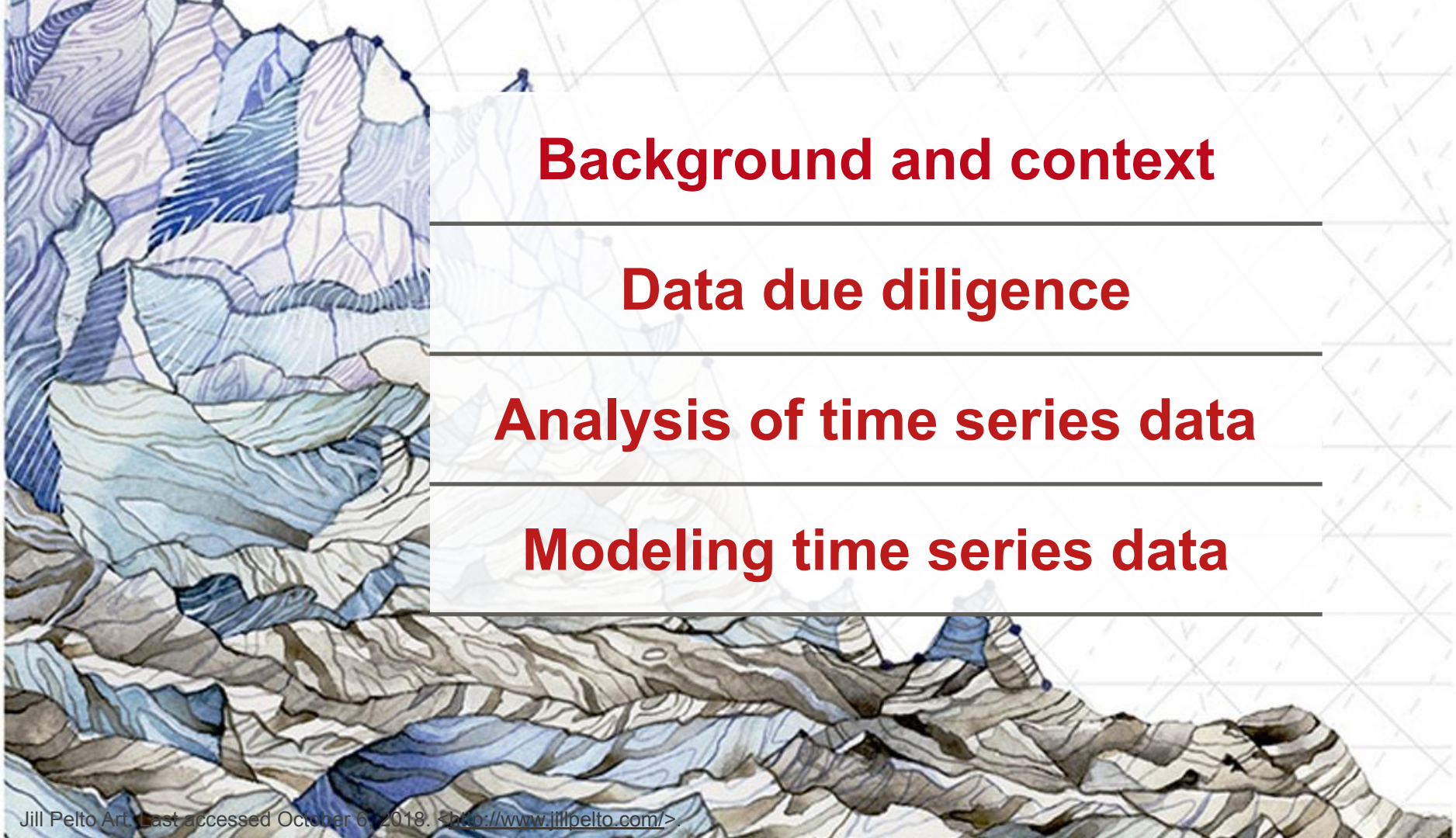
Applying statistical modeling and machine learning to perform time-series forecasting

Tamara Louie
October 21, 2018
PyData LA 2018

First things first.

- If you want to follow along with this tutorial presentation, the presentation is available here:

  https://goo.gl/xTgB7o



Background and context

Data due diligence

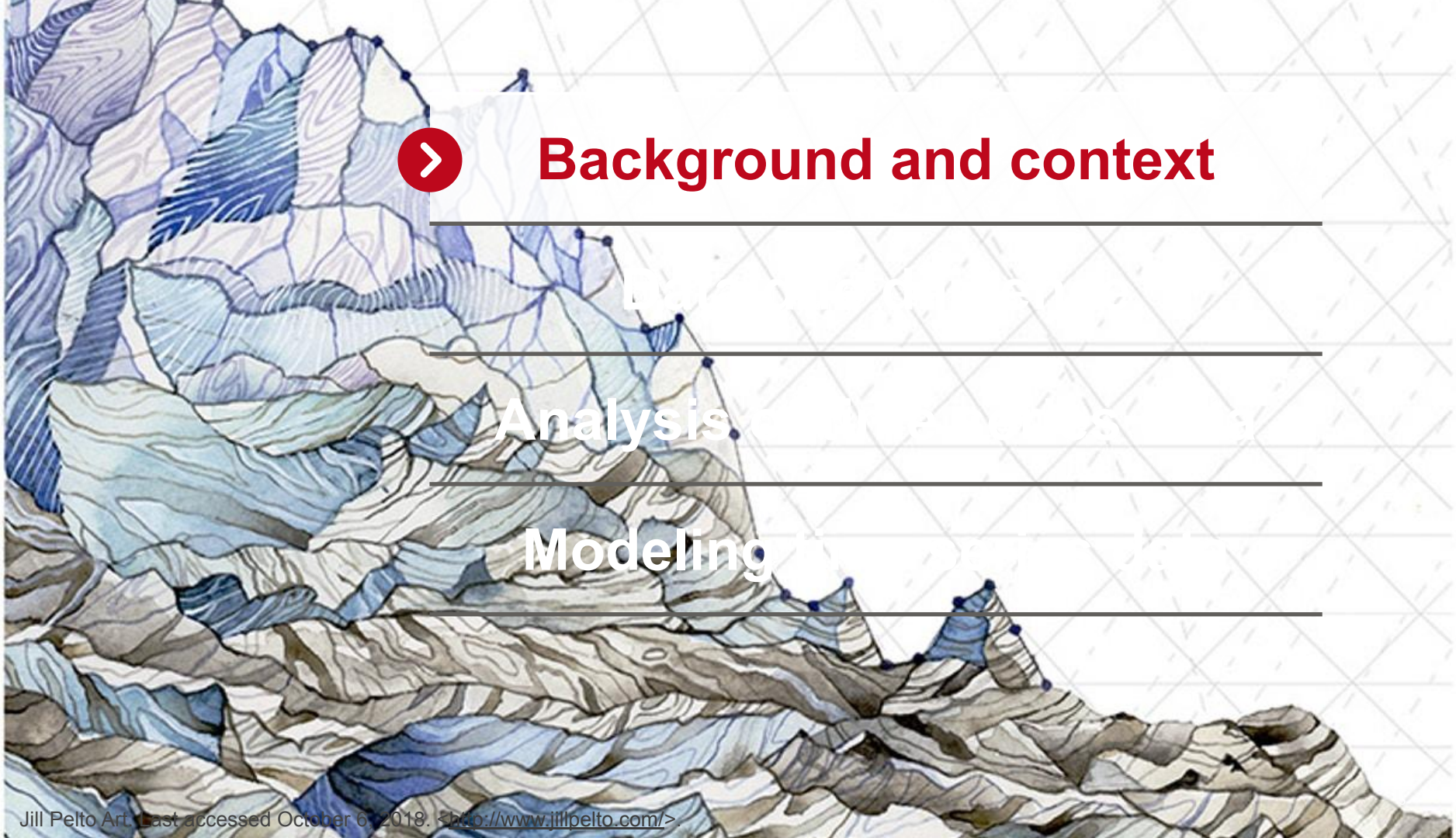
Analysis of time series data

Modeling time series data

Jill Pelto Art. Last accessed October 6, 2018. <http://www.jillpelto.com/>.

Some notes before we begin

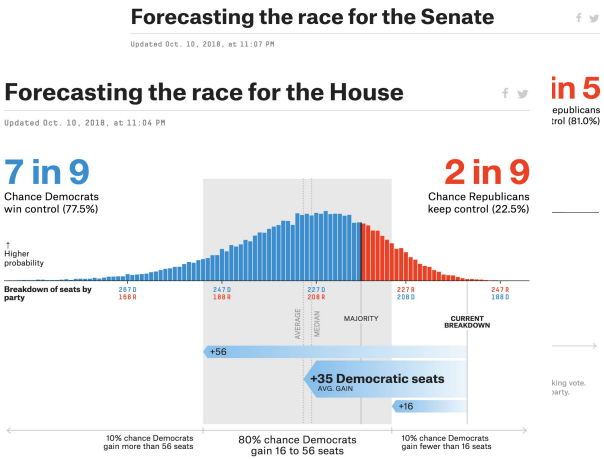
- Please let me know if you want to go more or less in depth into a particular subject.
- Please feel free to stop me and ask any clarifying questions.

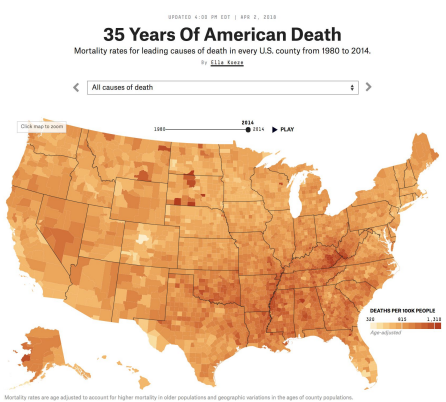


Jill Pelto Art. Last accessed October 6, 2018. <http://www.jillpelto.com/>.

Why is forecasting important (or at least, interesting)?

Applications in many fields, including politics, finance, health, etc.





Source of images, from left to right. 1. FiveThirtyEight. Last updated October 10, 2018. Last accessed October 10, 2018. Link. 2. Bloomberg. Last accessed October 10, 2018. Link. FiveThirtyEight. Last updated April 2, 2018. Last updated October 10, 2018. Link.

What are some things you may learn in this tutorial?

- Why time-series data is different.
- How to process time-series data.
- How to better understand time-series data.
- How to apply statistical and machine learning methods to time-series problems.
- Understand some strengths and weaknesses of these models.
- How to evaluate, interpret, and convey output from forecasting models.

What you will need for this tutorial

Toolbox:

- Colaboratory (Access to Chrome / Firefox, Google account) OR
- Access to a iPython notebook-like environment (e.g., conda, own jupyter environment)

Methodologies:

- Familiarity with data-related Python packages (e.g., numpy, pandas, matplotlib, datetime).
- Basic knowledge of statistical modeling and machine learning.

Who am I?

Why am I teaching this tutorial?

Cloud-based Electronic Health Records for Real-time, Region-specific Influenza Surveillance

Limits of Mechanistic versus Machine Learning Models for Influenza Forecasting in the United States

Sequence models

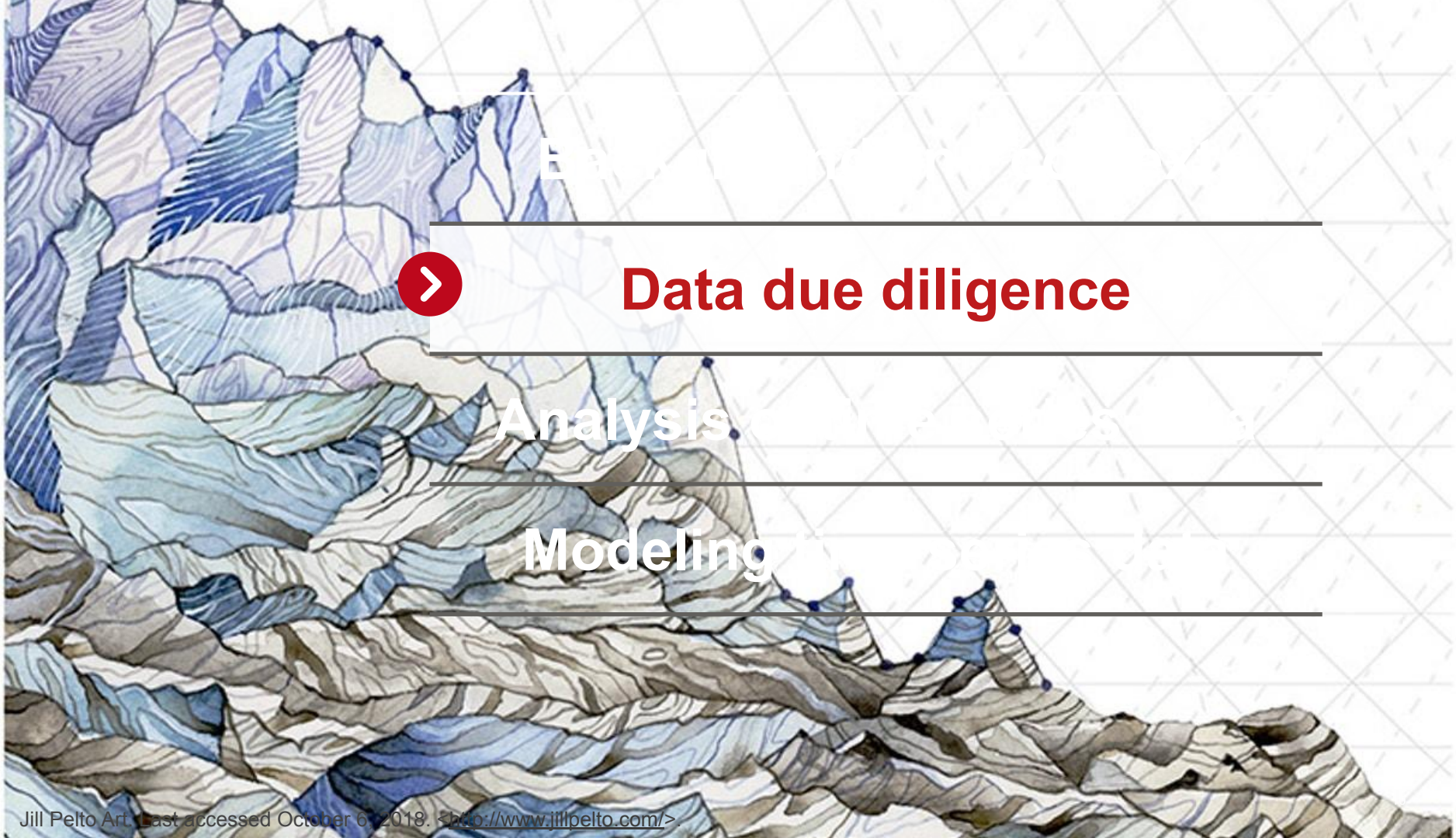
Time-series forecasting

Other things! Also some forecasting work

2014

2018

Source of images, from left to right. 1. Harvard T.H. Chan School of Public Health. Link Legendary Entertainment. Link Twitter. Link Pinterest. Link All last accessed October 10, 2018.



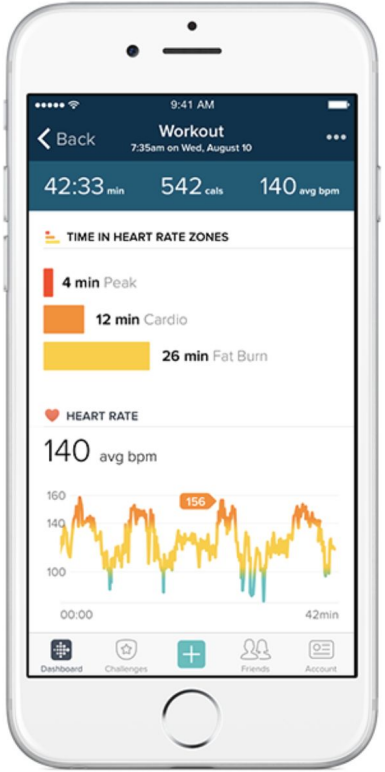
Jill Pelto Art. Last accessed October 6, 2018. <http://www.jillpelto.com/>.

What question do I want to answer?

- I think that it is important to start with a very concrete question that you think can be answered with data, specifically with time-series data.

- Some examples of questions that I might answer with forecasting:
  - What is the expected voter turnout for California for the 2018 midterm elections?
  - What is the future expected price of Apple stock over the next year?
  - What is the expected life expectancy of the average US female in 50 years?

What is time-series data? Do I need time-series data to answer this question?



- Time-series data can be defined in many ways. I tend to describe it as "data collected on the same metric or same object at regular or irregular time intervals."
- In terms of whether one needs time-series data depends on the question.
- I think about whether there is an inherent relationship or structure between data at various time points (e.g., is there a time-dependence), and whether we can leverage that time-ordered information.

Source of images. Alice Oglethorpe, Fitbit. Link. Last accessed October 10, 2018.

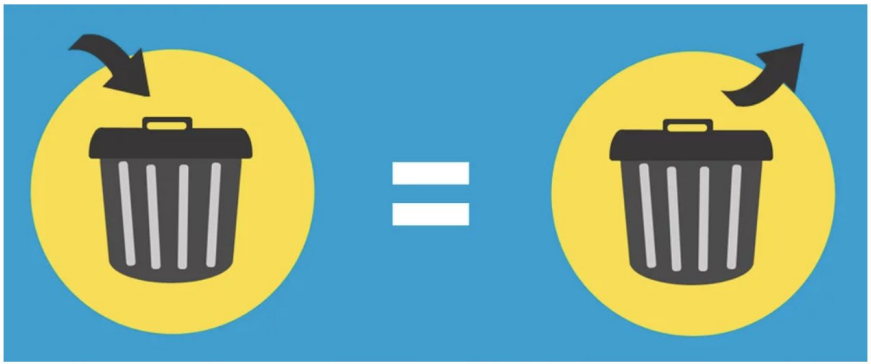
How do we get some time-series data?

- I think that this step is very important, as it will determine if I can even tackle this question.
- I tend to choose questions where there is available data, or the capability to create the data myself.
- I am planning on using data from this link, which appears to be from public data available about Airbnb.
- Before I start figuring out how to ingest this data, I first want to read about this data and understand if it is what I want, and if it fulfills some basic requirements about data quality.

How is the data is generated?

See here for some information. According to the site:

- Utilizes public information compiled from the Airbnb website, including the availability calendar for 365 days in the future, and the reviews for each listing
- Not associated with or endorsed by Airbnb or any of Airbnb's competitors.



Source of images. Abivin. Last updated December 25, 2017. Link. Last accessed October 10, 2018.

Is the data clean or dirty? Is additional processing necessary?

Is the data clean or dirty?

- Data was verified, cleansed, analyzed and aggregated by someone. I do not know how it was pre-processed.
- The author does not know if there has been additional processing or modification from Airbnb in releasing their public data.

Is additional processing necessary?

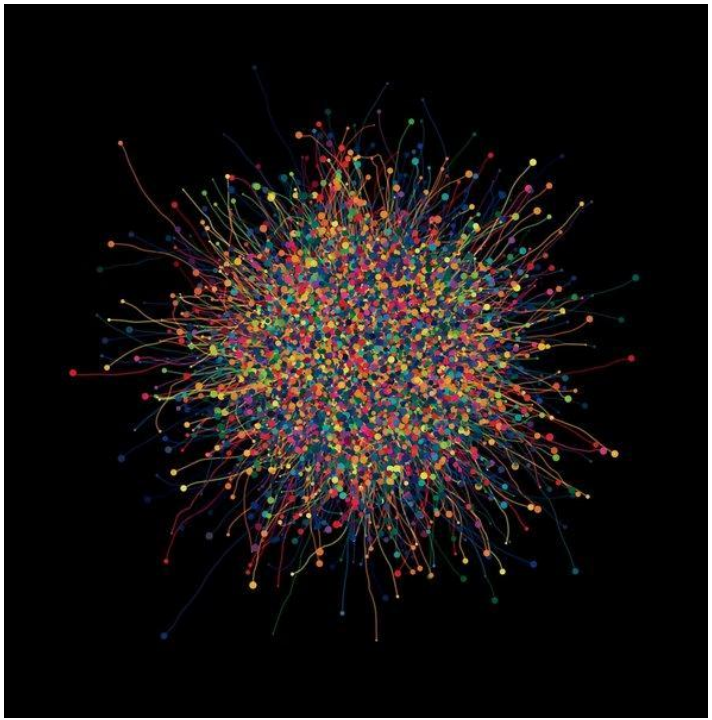
- **Are there nulls?** Yes, there are a lack of entries (i.e., rows) for some dates.
- **What are the data types? Units?** Need to change the dates to date type. Units appear okay.

Source of images. Abivin. Last updated December 25, 2017. Link. All last accessed October 10, 2018.

Where do I store my data? How do I access my data?

- This aspect is quite time intensive if your data is large and / or not in a state to be ingested by Python.

- In fact, this step may take the majority of your time when you are performing a modeling or machine learning task, between getting and reading in the data, cleaning data, and making sure the data is being ingested correctly, and verifying that the data is being logged correctly.



Source of images. Steve Lohr, New York Times. Last updated June 19, 2013. Link Last accessed October 10, 2018.

How do we load the data into a place where we can access it?

How do we access the data without crashing?

- It is just a 23MB CSV file. Most single clusters or machines can handle this size data.

Do I need to sample my data?

- Not in this case.

Can I load it all into memory on a single machine?

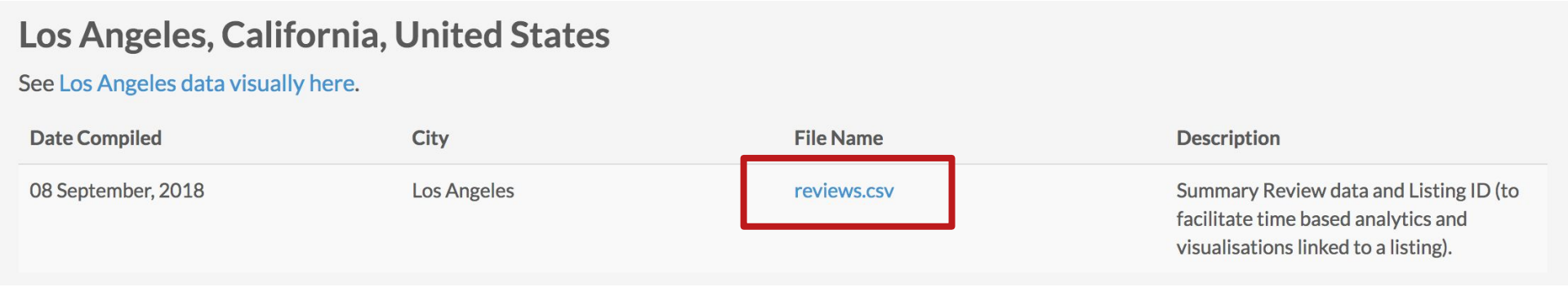
- Yes, we can load it all into memory on a single machine.

Now, let's download the data.

Download the data

Please use the link below to download the following CSV file (23 MB) from the "Los Angeles" section http://insideairbnb.com/get-the-data.html

reviews.csv



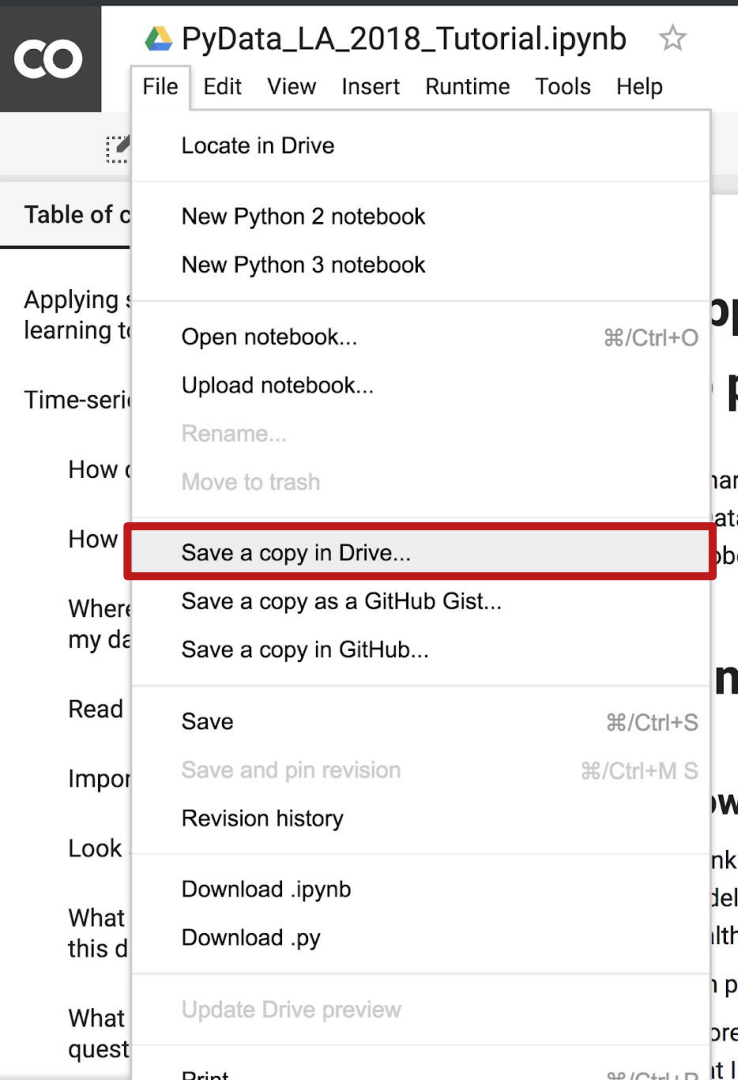
Los Angeles, California, United States

See Los Angeles data visually here.

Date Compiled	City	File Name	Description
08 September, 2018	Los Angeles	reviews.csv	Summary Review data and Listing ID (to facilitate time based analytics and visualisations linked to a listing).

Source of images. Inside Airbnb. Link. Last accessed October 10, 2018.

For the rest of this tutorial, let’s refer to the iPython notebook



Download the iPython notebook - option 1

[Colaboratory - preferred method]

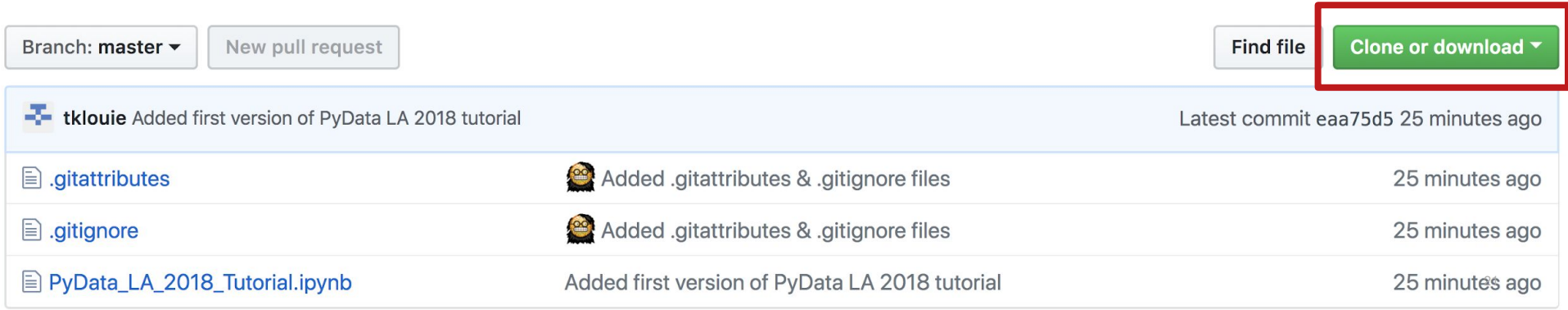
- Try opening the Google Drive link below https://goo.gl/r7CFcN
- Go to File → Save a copy in Drive…
- Save a copy of the iPython notebook: **PyData_LA_2018_Tutorial.ipynb**

Download the iPython notebook - option 2

[Alternative with Github]

- Download from my Github link below
  https://github.com/tklouie/PyData_LA_2018

- Upload the following iPython notebook to your preferred python / Jupyter environment **PyData_LA_2018_Tutorial.ipynb**



Let’s open the iPython notebook and proceed.

I will also add in some of the notes from the iPython notebook here in the relevant sections.

Import some relevant packages

- Ideally, one should set up their own virtual environment and determine the versions of each library that they are using.

- Here, we will assume that the Colaboratory environment has some shared environment with access to common Python libraries and the ability to install other libraries necessary.

Look at the data

- How many rows are in the dataset?
- How many columns are in this dataset?
- What data types are the columns?
- Is the data complete? Are there nulls? Do we have to infer values?
- What is the definition of these columns?
- What are some other caveats to the data?

What are some questions I can answer with this data?

- Understand the limitations of your data and what potential questions can be answered by data is important. These questions can reduce, expand, or modify the scope of your project.
- If you defined a scope or goal for your project before digging into the data, this might be a good time to revisit it.

Data. We have daily count of reviews for given listing ids for given dates.
Questions I could try to answer.

- Forecast future number of reviews for the Los Angeles area.
- Forecast the future number of reviews for specific listings in the Los Angeles area.

What techniques may help answer these questions?

Statistical models

- Ignore the time-series aspect completely and model using **traditional statistical modeling toolbox** (e.g., regression-based models).
- **Univariate statistical time-series modeling** (e.g., averaging and smoothing models, ARIMA models).
- **Slight modifications to univariate statistical time-series modeling** (e.g., external regressors, multi-variate models).
- **Additive or component models** (e.g., Facebook Prophet package).
- **Structural time series modeling** (e.g., Bayesian structural time series modeling, hierarchical time series modeling).

What techniques may help answer these questions?

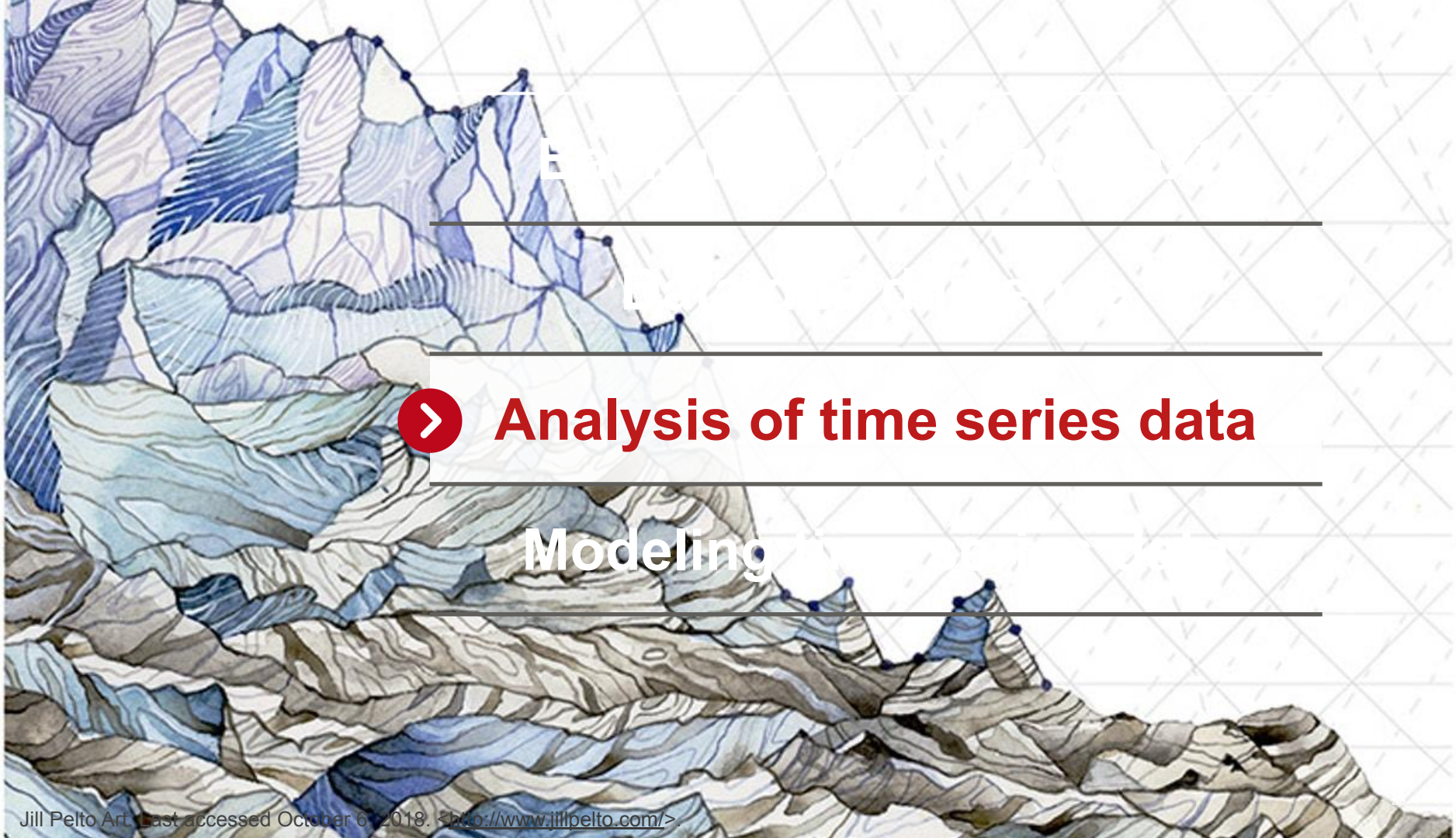
Machine learning models

- Ignore the time-series aspect completely and model using **traditional machine learning modeling** toolbox. (e.g., Support Vector Machines (SVMs), Random Forest Regression, Gradient-Boosted Decision Trees (GBDTs), Neural Networks (NNs).
- **Hidden markov models (HMMs)**.
- **Other sequence-based models**.
- **Gaussian processes (GPs)**.
- **Recurrent neural networks (RNNs).**

What techniques may help answer these questions?

Additional data considerations before choosing a model

- Whether or not to incorporate external data
- Whether or not to keep as univariate or multivariate (i.e., which features and number of features)
- Outlier detection and removal
- Missing value imputation



Analysis of time series data

Modeling

Jill Pelto Art. Last accessed October 6, 2018. <http://www.jillpelto.com/>.

Look at stationarity

Most time-series models assume that the underlying time-series data is **stationary**. This assumption gives us some nice statistical properties that allows us to use various models for forecasting.

If we are using past data to predict future data, we should assume that the data will follow the same general trends and patterns as in the past. This general statement holds for most training data and modeling tasks.

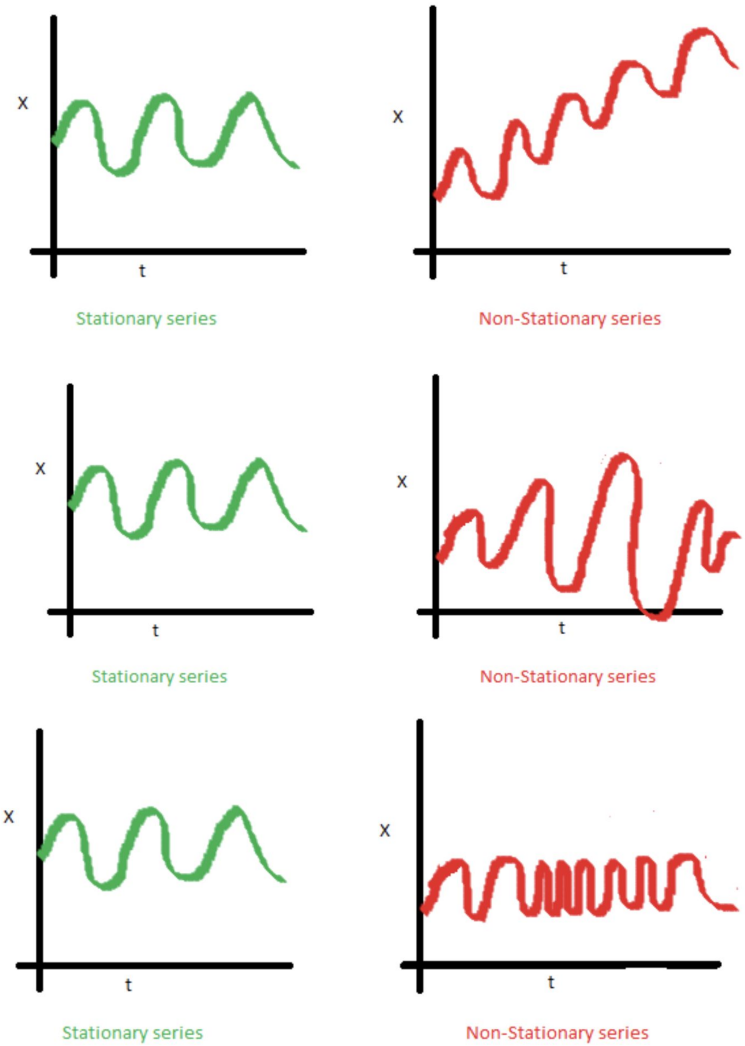
Look at stationarity

Stationarity is a statistical assumption that a time-series has:

- Constant mean
- Constant variance
- Autocovariance does not depend on time

There are some good diagrams and explanations on stationarity here and here.

Sometimes we need to transform the data in order to make it stationary. However, this transformation then calls into question if this data is truly stationary and is suited to be modeled using these techniques.



What happens if you do not correct for these things?

Many things can happen, including:

- Variance can be mis-specified
- Model fit can be worse.
- Not leveraging valuable time-dependent nature of the data.

Here are some resources on the **pitfalls of using traditional methods for time series analysis.**

- Quora link
- Quora link

Correct for stationarity

It is common for time series data to have to correct for non-stationarity.

2 common reasons behind non-stationarity are:

1. Trend – mean is not constant over time.
2. Seasonality – variance is not constant over time.

There are ways to correct for trend and seasonality, to make the time series stationary.

Eliminating trend and seasonality

Transformation

- Examples. Log, square root, etc.

Smoothing

- Examples. Weekly average, monthly average, rolling averages.

Differencing

- Examples. First-order differencing.

Polynomial Fitting

- Examples. Fit a regression model.

Decomposition

Modeling time series data

Jill Pelto Art. Last accessed October 6, 2018. <http://www.jillpelto.com/>.

Why is statistical forecasting important (or at least, interesting)?

"Forecasting can take many forms—staring into crystal balls or bowls of tea leaves, combining the opinions of experts, brainstorming, scenario generation, what-if analysis, Monte Carlo simulation, solving equations that are dictated by physical laws or economic theories—but statistical forecasting, which is the main topic to be discussed here, is the art and science of forecasting from data, with or without knowing in advance what equation you should use."

Robert Nau, Principles and Risks of Forecasting

Let us model some time-series data! Finally! ARIMA models.

We can use **ARIMA models** when we know there is dependence between values and we can leverage that information to forecast.

ARIMA = Auto-Regressive Integrated Moving Average.

Assumptions. The time-series is stationary.

Depends on:

1. Number of **AR** (Auto-Regressive) terms (p).
2. Number of **I** (Integrated or Difference) terms (d).
3. Number of **MA** (Moving Average) terms (q)

What do ARIMA models look like?

ARIMA Model

Y_t → AR filter → Integration filter → MA filter → ε_t

(long term) (stochastic trend) (short term) (white noise error)

ARIMA (2,0,1) $y_t = a_1 y_{t-1} + a_2 y_{t-2} + b_1 \varepsilon_{t-1}$

ARIMA (3,0,1) $y_t = a_1 y_{t-1} + a_2 y_{t-2} + a_3 y_{t-3} + b_1 \varepsilon_{t-1}$

ARIMA (1,1,0) $\Delta y_t = a_1 \Delta y_{t-1} + \varepsilon_t$, where $\Delta y_t = y_t - y_{t-1}$

ARIMA (2,1,0) $\Delta yt = a_1 \Delta y_{t-1} + a_2 \Delta y_{t-2} + \varepsilon_t$ where $\Delta yt = yt - yt\text{-}1$

To build a time series model issuing ARIMA, we need to study the time series and identify p,d,q

Venkat Reddy. Last accessed October 15, 2018. <https://www.slideshare.net/21_venkat/arima-26196965>.

ACF and PACF Plots

How do we determine p, d, and q? For p and q, we can use ACF and PACF plots (below).

- **Autocorrelation Function (ACF).** Correlation between the time series with a lagged version of itself (e.g., correlation of Y(t) with Y(t-1)).
- **Partial Autocorrelation Function (PACF).** Additional correlation explained by each successive lagged term.

How do we interpret ACF and PACF plots?

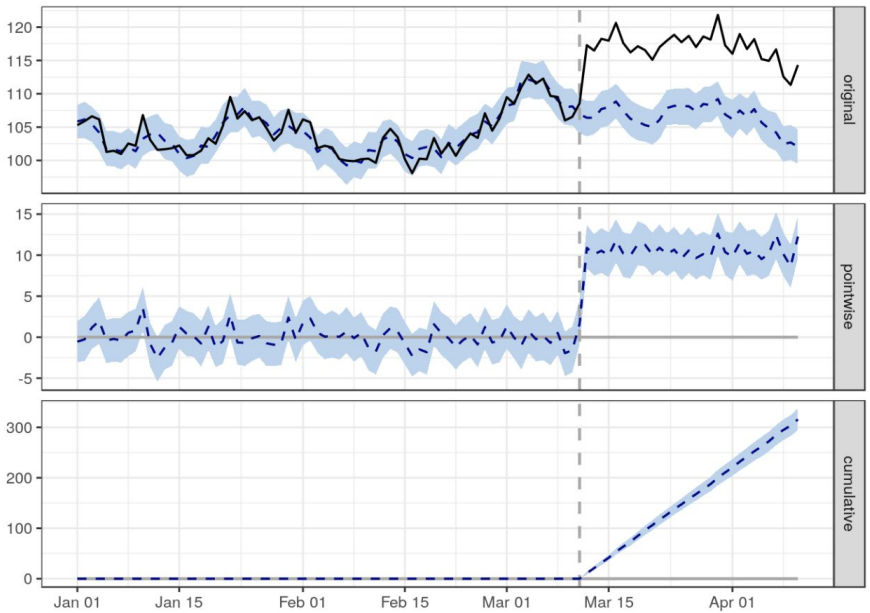
- **p** – Lag value where the PACF chart crosses the upper confidence interval for the first time.
- **q** – Lag value where the ACF chart crosses the upper confidence interval for the first time.

Let us model some time-series data! Finally! Facebook Prophet package.

Facebook Prophet is a tool that allows folks to forecast using additive or component models relatively easily. It can also include things like:

- Day of week effects
- Day of year effects
- Holiday effects
- Trend trajectory
- Can do MCMC sampling

Let us model some time-series data! Finally! Bayesian structural time series modeling.



Below are some resources available to learn more about **Bayesian structural time series modeling (BSTS)**:

- Causal Impact package from Google (available in R).
- Example implementation in python
- Example implementation in python in github

Google, Inc. Last accessed October 15, 2018. <https://google.github.io/CausalImpact/CausalImpact.html>.

Let us model some time-series data! Finally! LSTM for regression

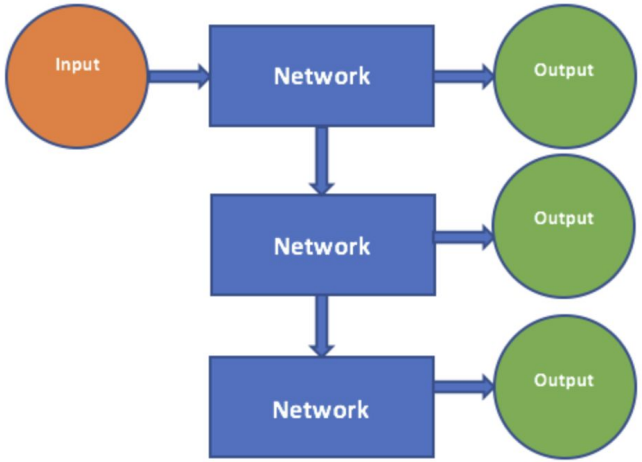
Here are some resources on **recurrent neural networks (RNN)** and **Long Short-Term Memory networks (LSTMs)**:

- Link 1
- Link 2
- Link 3

One to One. Classic Neural Network.

One to One

One to Many. Classic Neural Network.

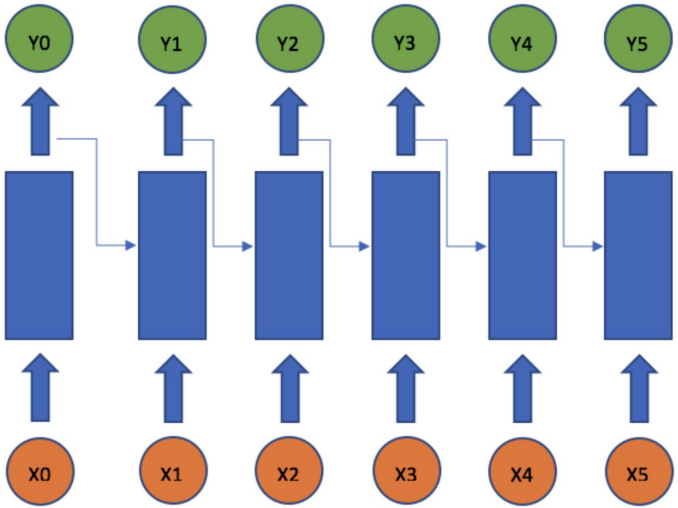


One to Many

Neelabh Pant. Last accessed October 15, 2018. <https://blog.statsbot.co/time-series-prediction-using-recurrent-neural-networks-lstms-807fa6ca7f>.

Let us model some time-series data! Finally! LSTM for regression

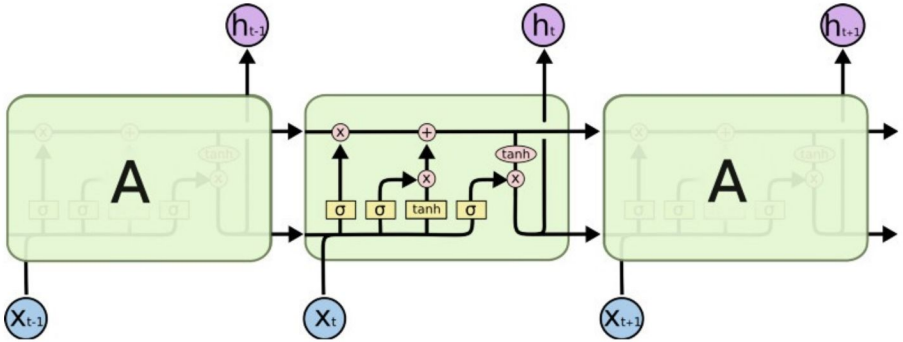
Recurrent Neural Network (RNN)



RNN Unrolled Time

$$Y_t = \tanh(wY_{t-1} + u\,x_t)$$

Long Short-Term Memory Network (LSTM)



LSTM Architecture

Able to capture longer-term dependencies in a sequence.

Neelabh Pant. Last accessed October 15, 2018. <https://blog.statsbot.co/time-series-prediction-using-recurrent-neural-networks-lstms-807fa6ca7f>.

How do we evaluate the success of our models? How does this evaluation differ from traditional modeling tasks?

One way is through traditional continuous evaluation modeling metrics, such as RMSE, MAPE, etc.

Here are some additional resources to learn about intricacies of time series model validation, specifically around cross-validation:

- Link 1
- Link 2
- Link 3
- Link 4

Thank you!

tamaralouie@pinterest.com

Appendix

Random Walk

Here are some resources on **random walks.**

- Link 1
- Link 2