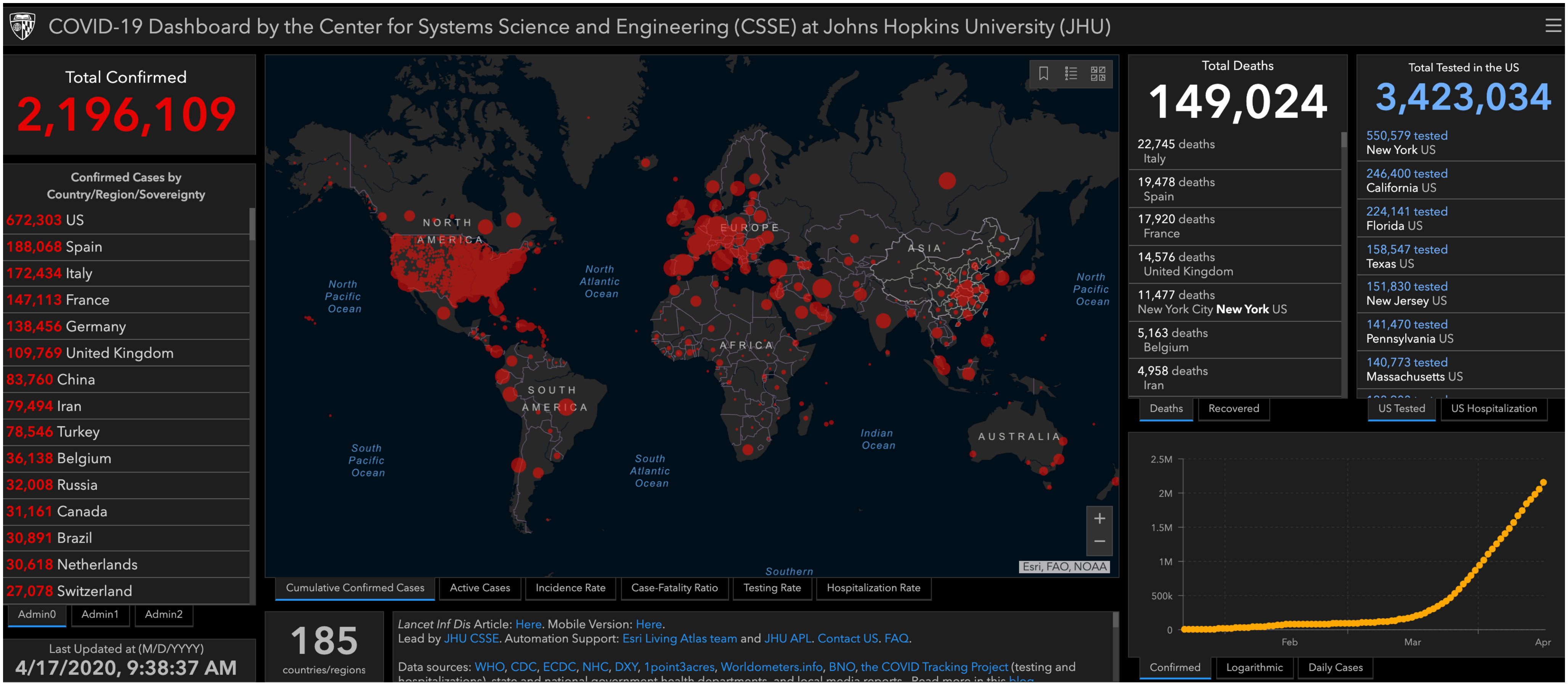
Math 189Z

Lecture 3: Time series data, Markov Chains, and HMM

 https://math189covid19.github.io/



- https://coronavirus.jhu.edu/map.html

Overview

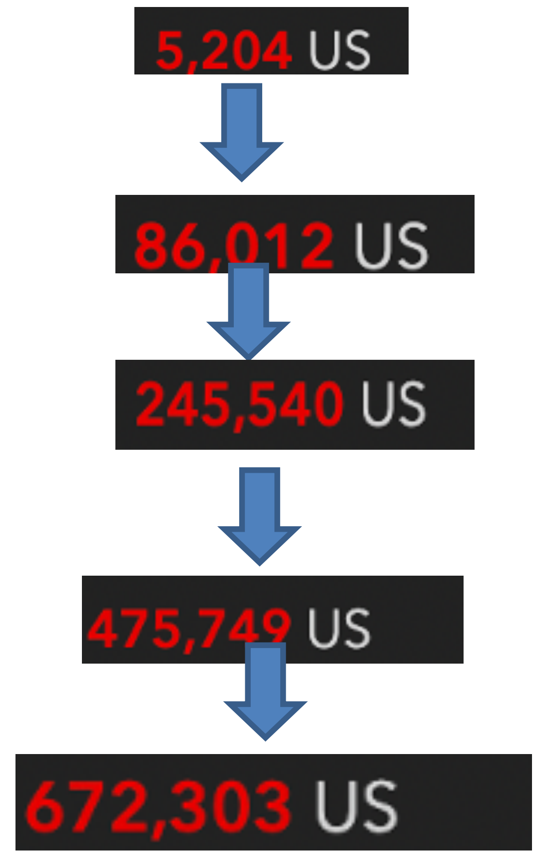
Total Confirmed 190,124 → Total Confirmed 553,244 → Total Confirmed 1,016,128 → Total Confirmed 1,650,210 → Total Confirmed 2,196,109

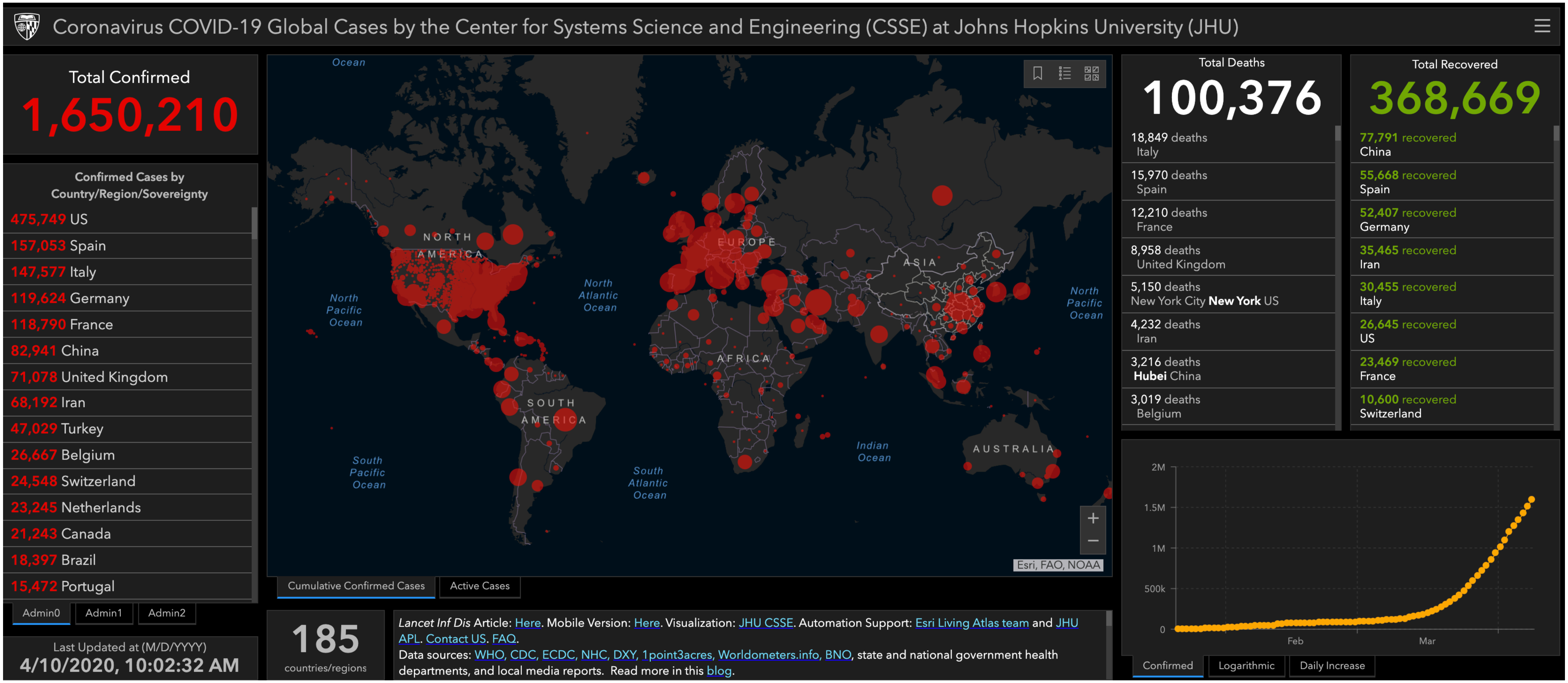
- COVID-19 confirmed cases have been increased but not doubled since our last meeting

In the case of Italy:

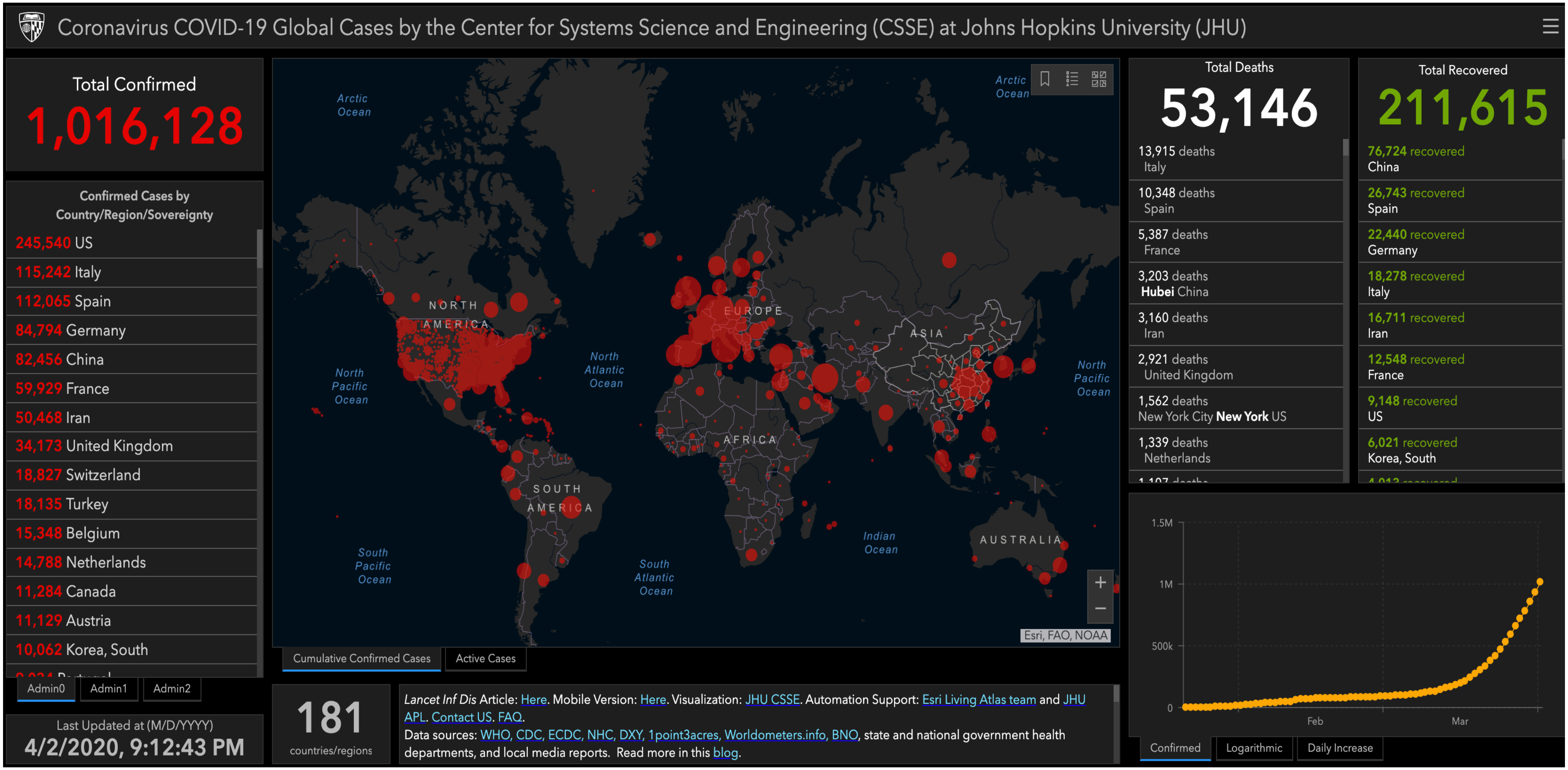
R1 = (147-115)/(115-80) = 0.91 < 1
R2 = (172-147)/(147-115)=
.78 < 0.91 < 1

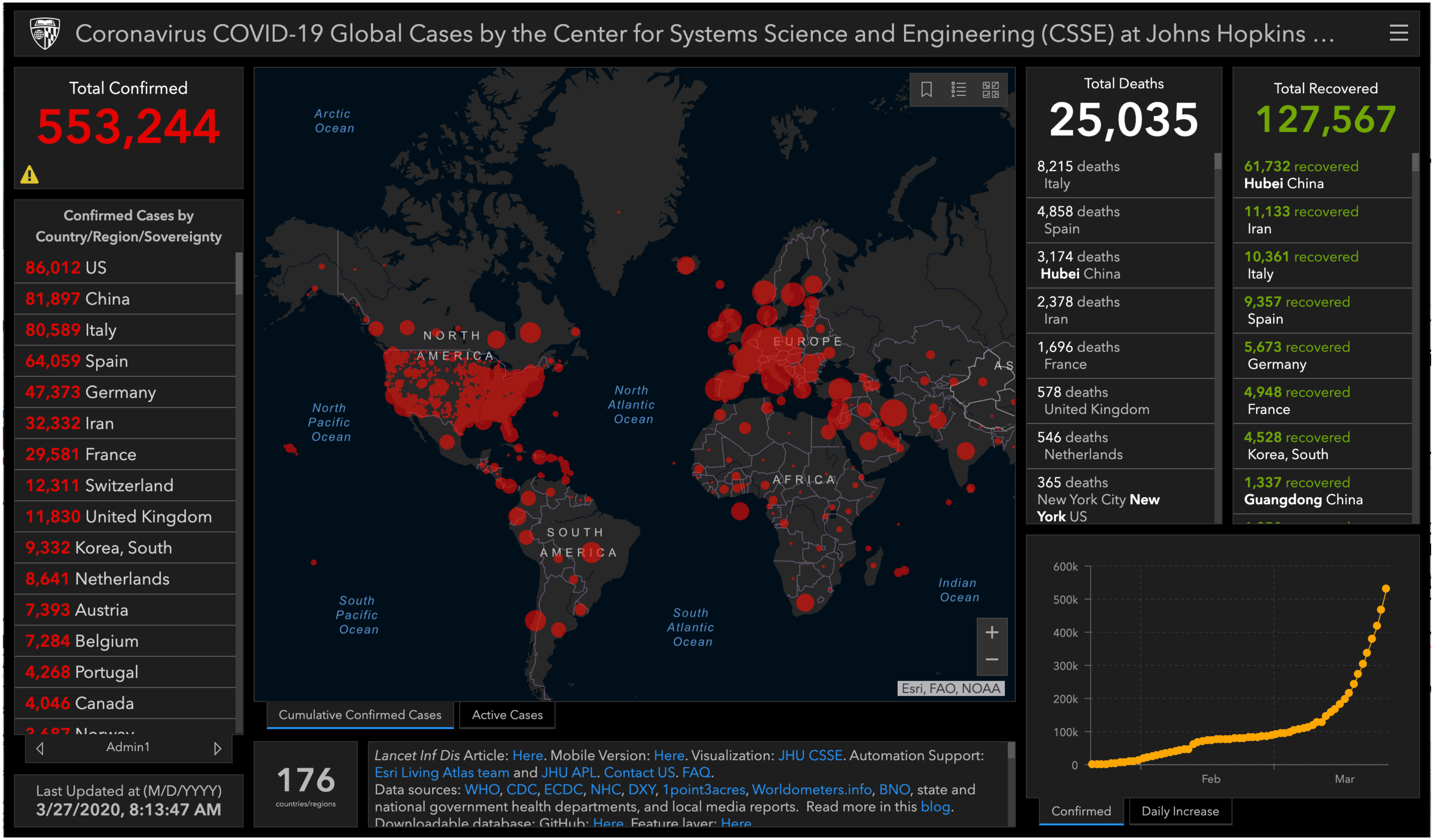
USA:
US_R0 = (245-86)/(86-0.5) =
1.97
US_R1 = (475-245)/(245-86)
=1.44
US_R2 = (672-457)/ (475-245)
0.85 <1

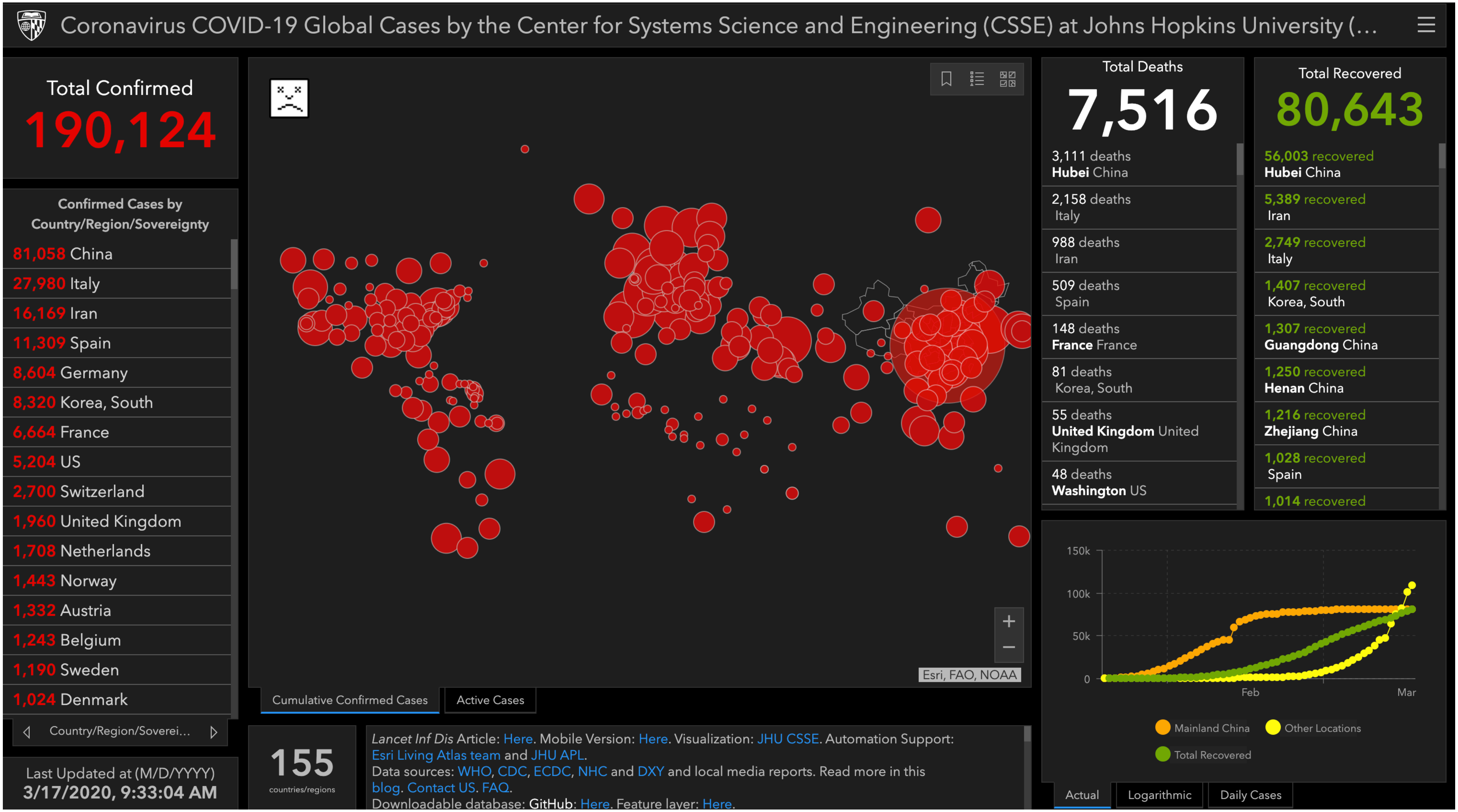




Coronavirus COVID-19 Global Cases by the Center for Systems Science and Engineering (CSSE) at Johns Hopkins University (JHU)
Total Confirmed
1,650,210
Confirmed Cases by Country/Region/Sovereignty
475,749 US
157,053 Spain
147,577 Italy
119,624 Germany
118,790 France
82,941 China
71,078 United Kingdom
68,192 Iran
47,029 Turkey
26,667 Belgium
24,548 Switzerland
23,245 Netherlands
21,243 Canada
18,397 Brazil
15,472 Portugal
Admin0
Admin1
Admin2
Last Updated at (M/D/YYYY)
4/10/2020, 10:02:32 AM
Ocean
NORTH AMERICA
North Pacific Ocean
North Atlantic Ocean
EUROPE
ASIA
North Pacific Ocean
AFRICA
SOUTH AMERICA
South Pacific Ocean
South Atlantic Ocean
Indian Ocean
AUSTRALIA
Esri, FAO, NOAA
Cumulative Confirmed Cases
Active Cases
185
countries/regions
Lancet Inf Dis Article: Here. Mobile Version: Here. Visualization: JHU CSSE. Automation Support: Esri Living Atlas team and JHU APL. Contact US. FAQ.
Data sources: WHO, CDC, ECDC, NHC, DXY, 1point3acres, Worldometers.info, BNO, state and national government health departments, and local media reports. Read more in this blog.
Total Deaths
100,376
18,849 deaths Italy
15,970 deaths Spain
12,210 deaths France
8,958 deaths United Kingdom
5,150 deaths New York City New York US
4,232 deaths Iran
3,216 deaths Hubei China
3,019 deaths Belgium
Total Recovered
368,669
77,791 recovered China
55,668 recovered Spain
52,407 recovered Germany
35,465 recovered Iran
30,455 recovered Italy
26,645 recovered US
23,469 recovered France
10,600 recovered Switzerland
2M
1.5M
1M
500k
0
Feb
Mar
Confirmed
Logarithmic
Daily Increase


Coronavirus COVID-19 Global Cases by the Center for Systems Science and Engineering (CSSE) at Johns Hopkins University (JHU)
Total Confirmed
1,016,128
Confirmed Cases by Country/Region/Sovereignty
245,540 US
115,242 Italy
112,065 Spain
84,794 Germany
82,456 China
59,929 France
50,468 Iran
34,173 United Kingdom
18,827 Switzerland
18,135 Turkey
15,348 Belgium
14,788 Netherlands
11,284 Canada
11,129 Austria
10,062 Korea, South
Admin0
Admin1
Admin2
Last Updated at (M/D/YYYY)
4/2/2020, 9:12:43 PM
Arctic Ocean
NORTH AMERICA
North Pacific Ocean
North Atlantic Ocean
EUROPE
ASIA
AFRICA
SOUTH AMERICA
South Pacific Ocean
South Atlantic Ocean
Indian Ocean
AUSTRALIA
Esri, FAO, NOAA
Cumulative Confirmed Cases
Active Cases
181
countries/regions
Lancet Inf Dis Article: Here. Mobile Version: Here. Visualization: JHU CSSE. Automation Support: Esri Living Atlas team and JHU APL. Contact US. FAQ.
Data sources: WHO, CDC, ECDC, NHC, DXY, 1point3acres, Worldometers.info, BNO, state and national government health departments, and local media reports. Read more in this blog.
Total Deaths
53,146
13,915 deaths Italy
10,348 deaths Spain
5,387 deaths France
3,203 deaths Hubei China
3,160 deaths Iran
2,921 deaths United Kingdom
1,562 deaths New York City New York US
1,339 deaths Netherlands
Total Recovered
211,615
76,724 recovered China
26,743 recovered Spain
22,440 recovered Germany
18,278 recovered Italy
16,711 recovered Iran
12,548 recovered France
9,148 recovered US
6,021 recovered Korea, South
1.5M
1M
500k
0
Feb
Mar
Confirmed
Logarithmic
Daily Increase


Coronavirus COVID-19 Global Cases by the Center for Systems Science and Engineering (CSSE) at Johns Hopkins ...
Total Confirmed
553,244
Confirmed Cases by Country/Region/Sovereignty
86,012 US
81,897 China
80,589 Italy
64,059 Spain
47,373 Germany
32,332 Iran
29,581 France
12,311 Switzerland
11,830 United Kingdom
9,332 Korea, South
8,641 Netherlands
7,393 Austria
7,284 Belgium
4,268 Portugal
4,046 Canada
Admin1
Last Updated at (M/D/YYYY)
3/27/2020, 8:13:47 AM
Arctic Ocean
NORTH AMERICA
North Pacific Ocean
North Atlantic Ocean
EUROPE
AFRICA
SOUTH AMERICA
South Pacific Ocean
South Atlantic Ocean
Indian Ocean
Esri, FAO, NOAA
Cumulative Confirmed Cases
Active Cases
176
countries/regions
Lancet Inf Dis Article: Here. Mobile Version: Here. Visualization: JHU CSSE. Automation Support: Esri Living Atlas team and JHU APL. Contact US. FAQ.
Data sources: WHO, CDC, ECDC, NHC, DXY, 1point3acres, Worldometers.info, BNO, state and national government health departments, and local media reports. Read more in this blog.
Total Deaths
25,035
8,215 deaths Italy
4,858 deaths Spain
3,174 deaths Hubei China
2,378 deaths Iran
1,696 deaths France
578 deaths United Kingdom
546 deaths Netherlands
365 deaths New York City New York US
Total Recovered
127,567
61,732 recovered Hubei China
11,133 recovered Iran
10,361 recovered Italy
9,357 recovered Spain
5,673 recovered Germany
4,948 recovered France
4,528 recovered Korea, South
1,337 recovered Guangdong China
600k
500k
400k
300k
200k
100k
0
Feb
Mar
Confirmed
Daily Increase


Coronavirus COVID-19 Global Cases by the Center for Systems Science and Engineering (CSSE) at Johns Hopkins University (…
Total Confirmed
190,124
Confirmed Cases by Country/Region/Sovereignty
81,058 China
27,980 Italy
16,169 Iran
11,309 Spain
8,604 Germany
8,320 Korea, South
6,664 France
5,204 US
2,700 Switzerland
1,960 United Kingdom
1,708 Netherlands
1,443 Norway
1,332 Austria
1,243 Belgium
1,190 Sweden
1,024 Denmark
Country/Region/Soverei…
Last Updated at (M/D/YYYY)
3/17/2020, 9:33:04 AM
Esri, FAO, NOAA
Cumulative Confirmed Cases
Active Cases
155
countries/regions
Lancet Inf Dis Article: Here. Mobile Version: Here. Visualization: JHU CSSE. Automation Support: Esri Living Atlas team and JHU APL.
Data sources: WHO, CDC, ECDC, NHC and DXY and local media reports. Read more in this blog. Contact US. FAQ.
Downloadable database: GitHub: Here. Feature layer: Here.
Total Deaths
7,516
3,111 deaths Hubei China
2,158 deaths Italy
988 deaths Iran
509 deaths Spain
148 deaths France France
81 deaths Korea, South
55 deaths United Kingdom United Kingdom
48 deaths Washington US
Total Recovered
80,643
56,003 recovered Hubei China
5,389 recovered Iran
2,749 recovered Italy
1,407 recovered Korea, South
1,307 recovered Guangdong China
1,250 recovered Henan China
1,216 recovered Zhejiang China
1,028 recovered Spain
1,014 recovered
150k
100k
50k
0
Feb
Mar
Mainland China
Other Locations
Total Recovered
Actual
Logarithmic
Daily Cases

Today: Overview

Today:

1. **Time series data**
2. **Markov Chains**
3. **HMM**

Recall: Last time, we covered

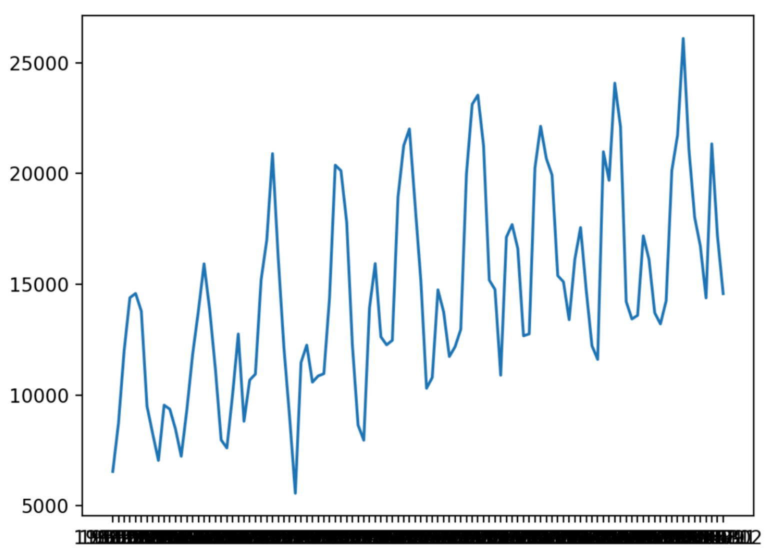
1. NMF (Non-negative Matrix Factorization)
2. LSA (Latent Semantic Analysis)

- https://math189covid19.github.io/

What is a Time series (data)?

- A **time series** is a **series** of data points indexed (or listed or graphed) in **time** order.
- Most commonly, a **time series** is a sequence taken at successive equally spaced points in **time**.
- Thus it is a sequence of discrete-**time** data.

```
"Month","Sales"
"1960-01",6550
"1960-02",8728
"1960-03",12026
"1960-04",14395
"1960-05",14587
"1960-06",13791
"1960-07",9498
"1960-08",8251
"1960-09",7049
"1960-10",9545
"1960-11",9364
"1960-12",8456
"1961-01",7237
"1961-02",9374
"1961-03",11837
"1961-04",13784
"1961-05",15926
"1961-06",13821
"1961-07",11143
"1961-08",7975
"1961-09",7610
"1961-10",10015
"1961-11",12759
```



This dataset is monthly and has nine years, or 108 observations. In our testing, will use the last year, or 12 observations, as the test set.

https://raw.githubusercontent.com/jbrownlee/Datasets/master/monthly car-sales.csv

```
# load
series = read_csv('monthly-car-sales.csv', header=0, index_col=0)
```

Once loaded, we can summarize the shape of the dataset in order to determine the number of observations.

```
# summarize shape
print(series.shape)
```

We can then create a line plot of the series to get an idea of the structure of the series.

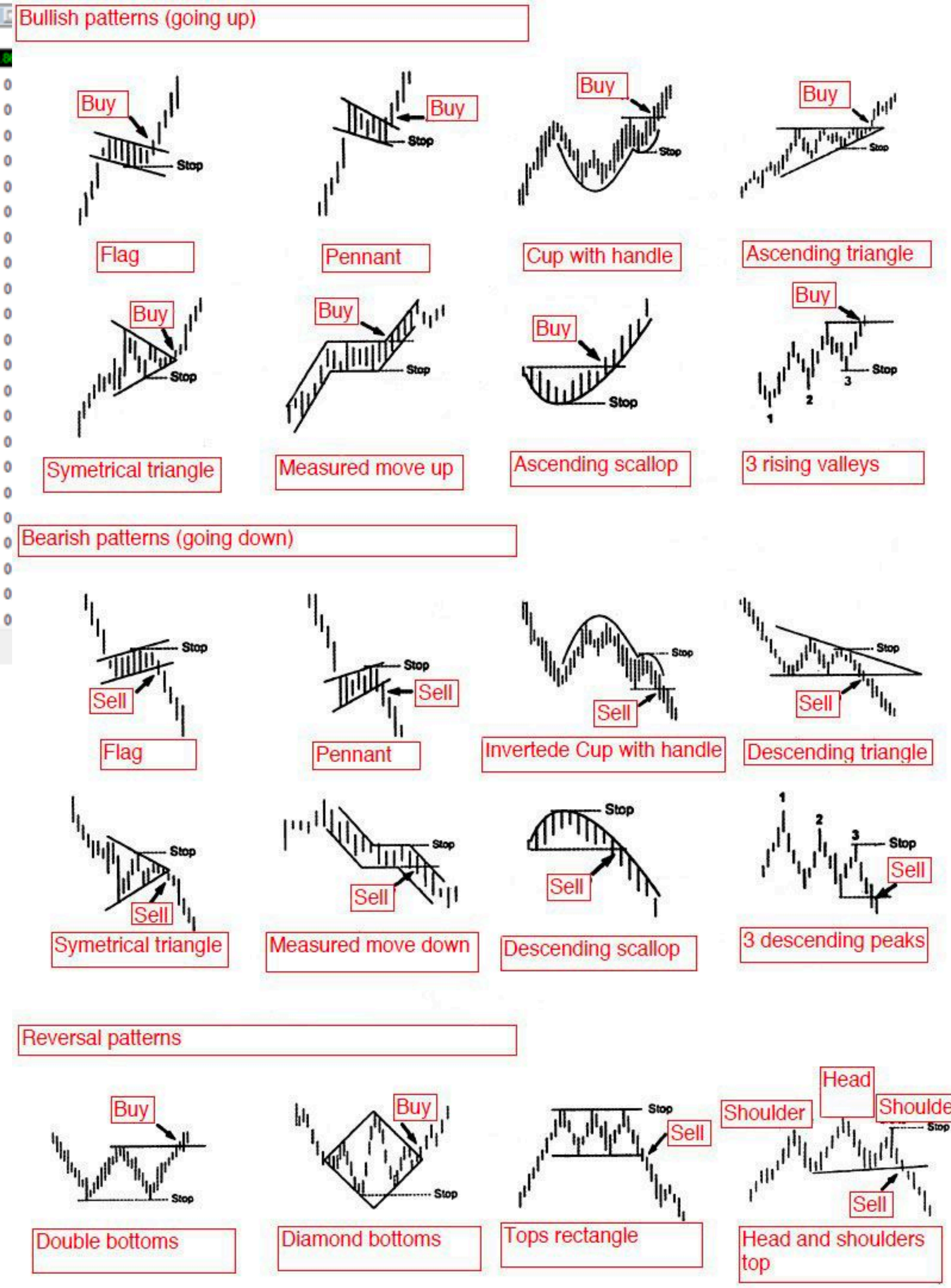
```
# plot
pyplot.plot(series)
pyplot.show()
```

We can tie all of this together; the complete example is listed below.

```
# load and plot dataset
from pandas import read_csv
from matplotlib import pyplot
# load
series = read_csv('monthly-car-sales.csv', header=0, index_col=0)
# summarize shape
print(series.shape)
# plot
pyplot.plot(series)
pyplot.show()
```

Stock data is time series data

Task example: Find patterns in stock time series data



Markov Chains

- Well known example: predict weather

In our simplified universe, the weather can only be in one of 2 possible states, “sunny” or “rainy”.

- The catch (in the context of Markov chains) is that the probability of it being sunny or rainy **tomorrow**, depends on whether it is sunny or rainy **today**.
- We’ll derive these probabilities from past data, and construct a **transition matrix**.

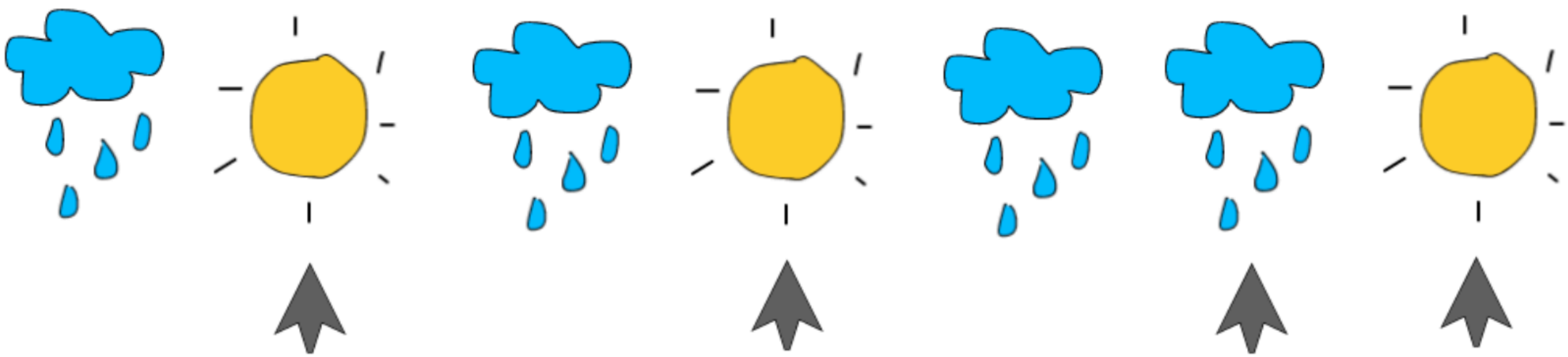
Using the historic data to build a transition matrix

- Here we use 7 days of historical data on which to “train” our Markov chain. The days are: [rain, sun, rain, sun, rain, rain, sun]

R S R S R R S

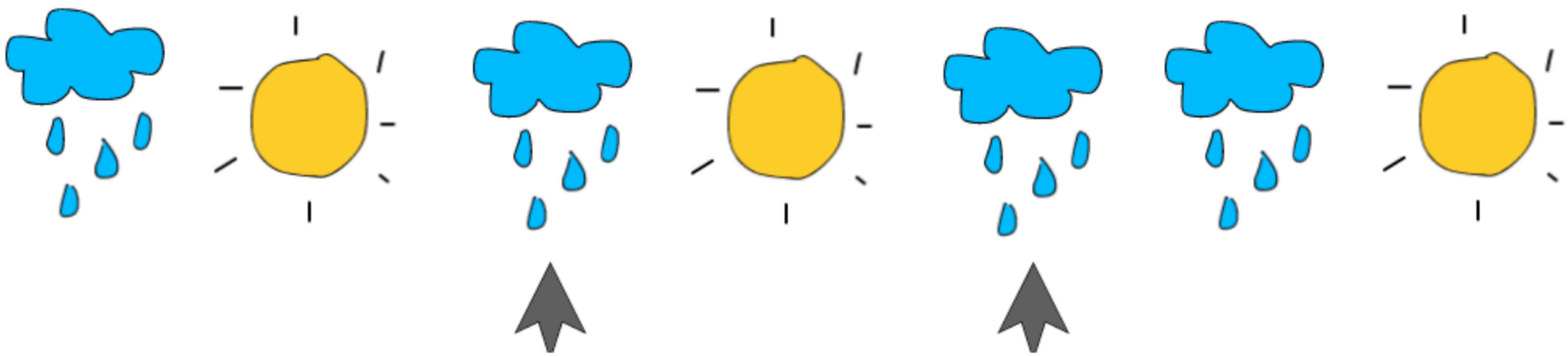
https://towardsdatascience.com/predicting-the-weather-with-markov-chains-a34735f0c4df

Now calculate the percentage of instances its sunny on days directly following rainy days.



3/4 so 75%.

Now calculate the percentage of instances its rainy on days directly following sunny days.

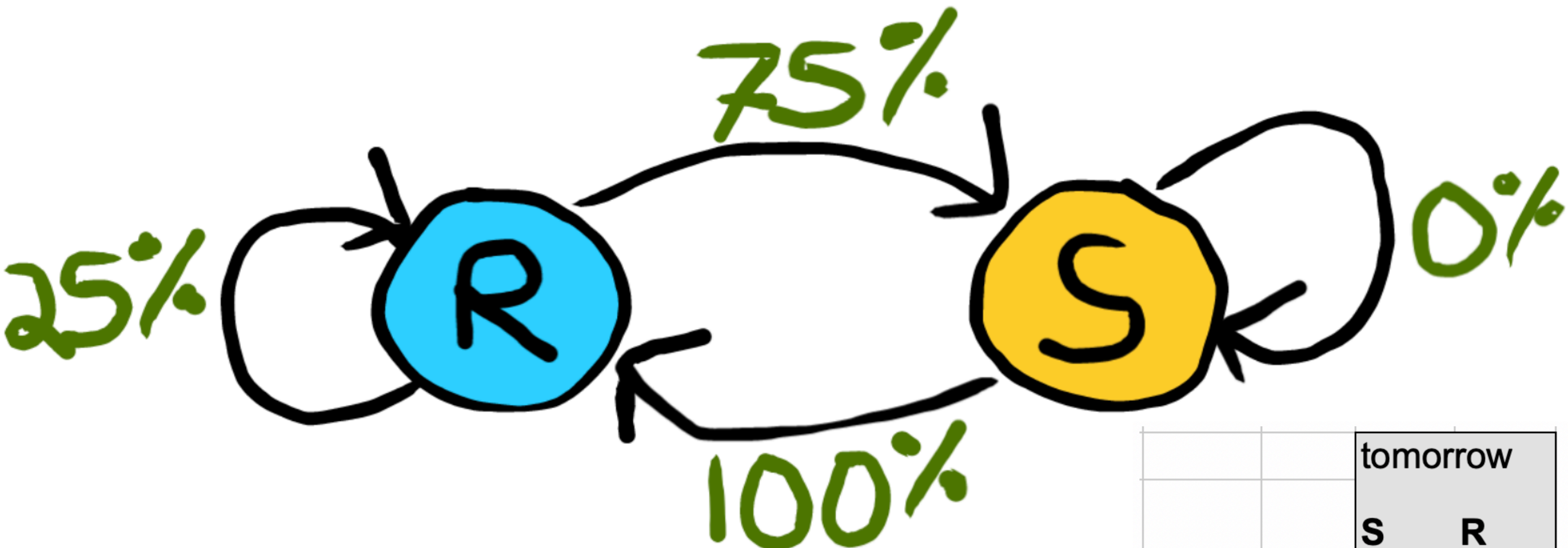


We'll build our transition matrix with that information, inferring missing percentages from the information we've already derived (rain-after-rain = 25% and sun-after-sun = 0%).

Transition Matrix

		tomorrow	
		S	**R**
today	**S**	0	1
	R	0.75	0.25

The Markov Chain



R = rainy, S = sunny

		tomorrow	
		S	**R**
today	**S**	0	1
	R	0.75	0.25

This diagram hits home the fact that probabilities are completely dependent on the current state, not the weather yesterday or the day before that.

Example 1:

The previous 3 days are [rainy, sunny, rainy].

What's the probability of rainy weather tomorrow?

R, S, R

Based on our previously trained model. Tomorrow has a 75% chance of sun and 25% chance of rain.

Example 2:

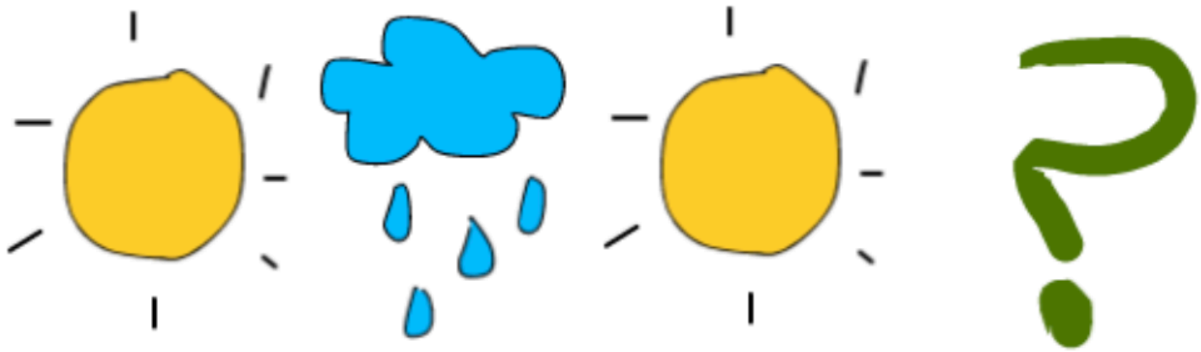
The previous 2 days are [rainy, rainy].

R R

Again, tomorrow has a 75% chance of sun and 25% chance of rain.

Example 3:

The previous 3 days are [sunny, rainy, sunny].



S R S

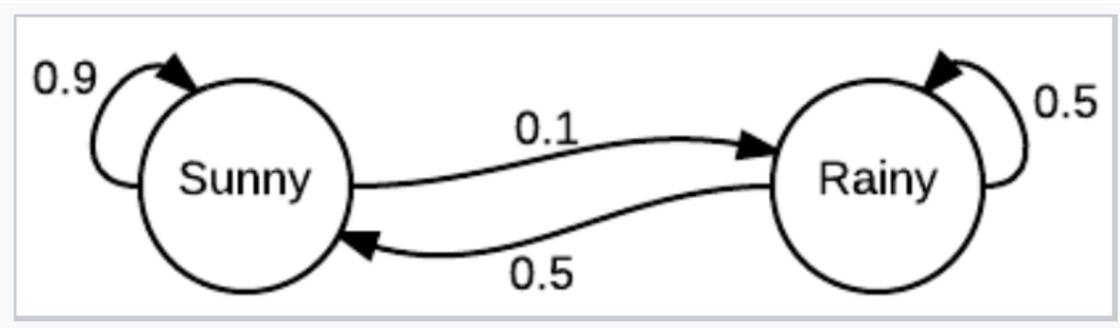
There is a 100% chance of rain tomorrow. It always rains on days after sun… sad I know…

Suppose we get the transition matrix with lots lots of data

- Say

$$P = \begin{bmatrix} 0.9 & 0.1 \\ 0.5 & 0.5 \end{bmatrix}$$

The matrix *P* represents the weather model in which a sunny day is 90% likely to be followed by another sunny day, and a rainy day is 50% likely to be followed by another rainy day. The columns can be labelled "sunny" and "rainy", and the rows can be labelled in the same order.

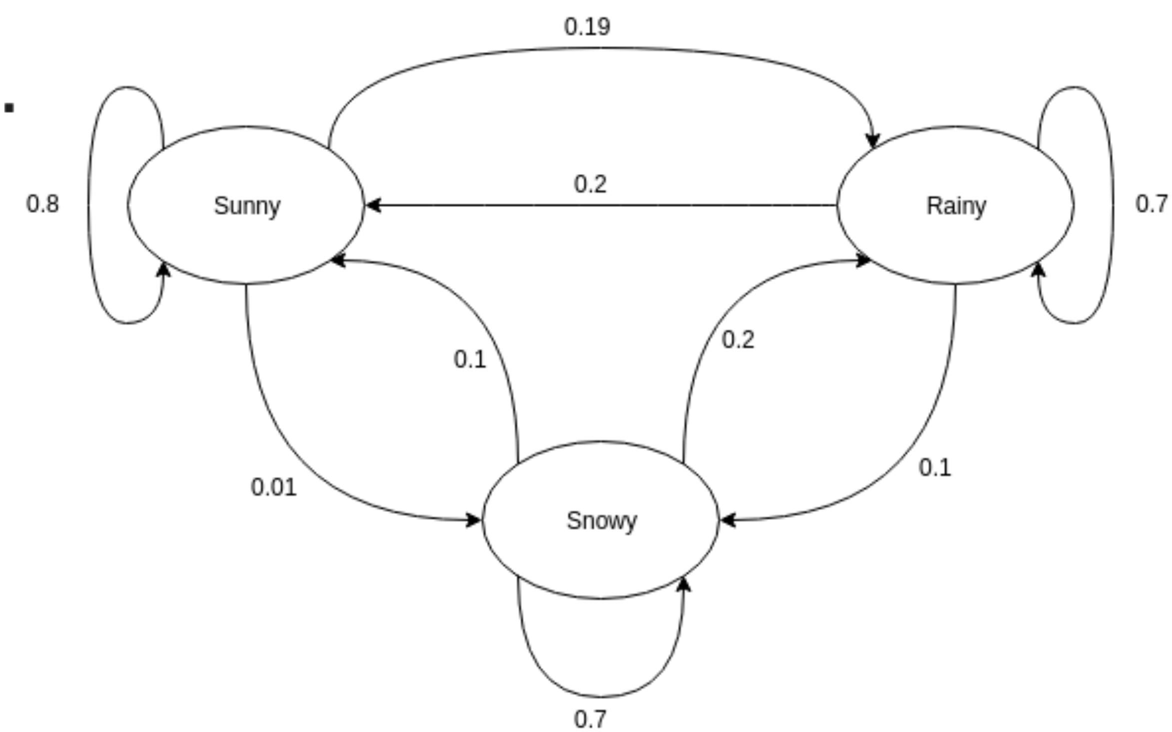


The above matrix as a graph.

Definition: stochastic matrix

$(P)_{ij}$ is the probability that, if a given day is of type i, it will be followed by a day of type j.

Notice that the rows of P sum to 1: this is because P is a stochastic matrix.



Predicting the weather

The weather on day 0 (today) is known to be sunny. This is represented by a vector in which the "sunny" entry is 100%, and the "rainy" entry is 0%:

$$\mathbf{x}^{(0)} = \begin{bmatrix} 1 & 0 \end{bmatrix}$$

The weather on day 1 (tomorrow) can be predicted by:

$$\mathbf{x}^{(1)} = \mathbf{x}^{(0)} P = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} 0.9 & 0.1 \\ 0.5 & 0.5 \end{bmatrix} = \begin{bmatrix} 0.9 & 0.1 \end{bmatrix}$$

Thus, there is a 90% chance that day 1 will also be sunny.

The weather on day 2 (the day after tomorrow) can be predicted in the same way:

$$\mathbf{x}^{(2)} = \mathbf{x}^{(1)} P = \mathbf{x}^{(0)} P^2 = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} 0.9 & 0.1 \\ 0.5 & 0.5 \end{bmatrix}^2 = \begin{bmatrix} 0.86 & 0.14 \end{bmatrix}$$

Iterative process

or

$$\mathbf{x}^{(2)} = \mathbf{x}^{(1)} P = \begin{bmatrix} 0.9 & 0.1 \end{bmatrix} \begin{bmatrix} 0.9 & 0.1 \\ 0.5 & 0.5 \end{bmatrix} = \begin{bmatrix} 0.86 & 0.14 \end{bmatrix}$$

General rules for day *n* are:

$$\mathbf{x}^{(n)} = \mathbf{x}^{(n-1)} P$$

$$\mathbf{x}^{(n)} = \mathbf{x}^{(0)} P^n$$

The steady state vector is defined as:

$$\mathbf{q} = \lim_{n \to \infty} \mathbf{x}^{(n)}$$

but converges to a strictly positive vector only if P is a regular transition matrix (that is, there is at least one P^n with all non-zero entries).

Since the **q** is independent from initial conditions, it must be unchanged when transformed by P.[4] This makes it an eigenvector (with eigenvalue 1), and means it can be derived from P.[4] For the weather example:

$$\begin{aligned} P &= \begin{bmatrix} 0.9 & 0.1 \\ 0.5 & 0.5 \end{bmatrix} \\ \mathbf{q}P &= \mathbf{q} && \text{(}\mathbf{q}\text{ is unchanged by }P\text{.)} \\ &= \mathbf{q}I \\ \mathbf{q}(P - I) &= \mathbf{0} \end{aligned}$$

$$\mathbf{q}\left(\begin{bmatrix} 0.9 & 0.1 \\ 0.5 & 0.5 \end{bmatrix} - \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\right) = \mathbf{0}$$

$$\mathbf{q}\begin{bmatrix} -0.1 & 0.1 \\ 0.5 & -0.5 \end{bmatrix} = \mathbf{0}$$

$$\begin{bmatrix} q_1 & q_2 \end{bmatrix}\begin{bmatrix} -0.1 & 0.1 \\ 0.5 & -0.5 \end{bmatrix} = \begin{bmatrix} 0 & 0 \end{bmatrix}$$

$$-0.1q_1 + 0.5q_2 = 0$$

and since they are a probability vector we know that

$$q_1 + q_2 = 1.$$

Solving this pair of simultaneous equations gives the steady state distribution:

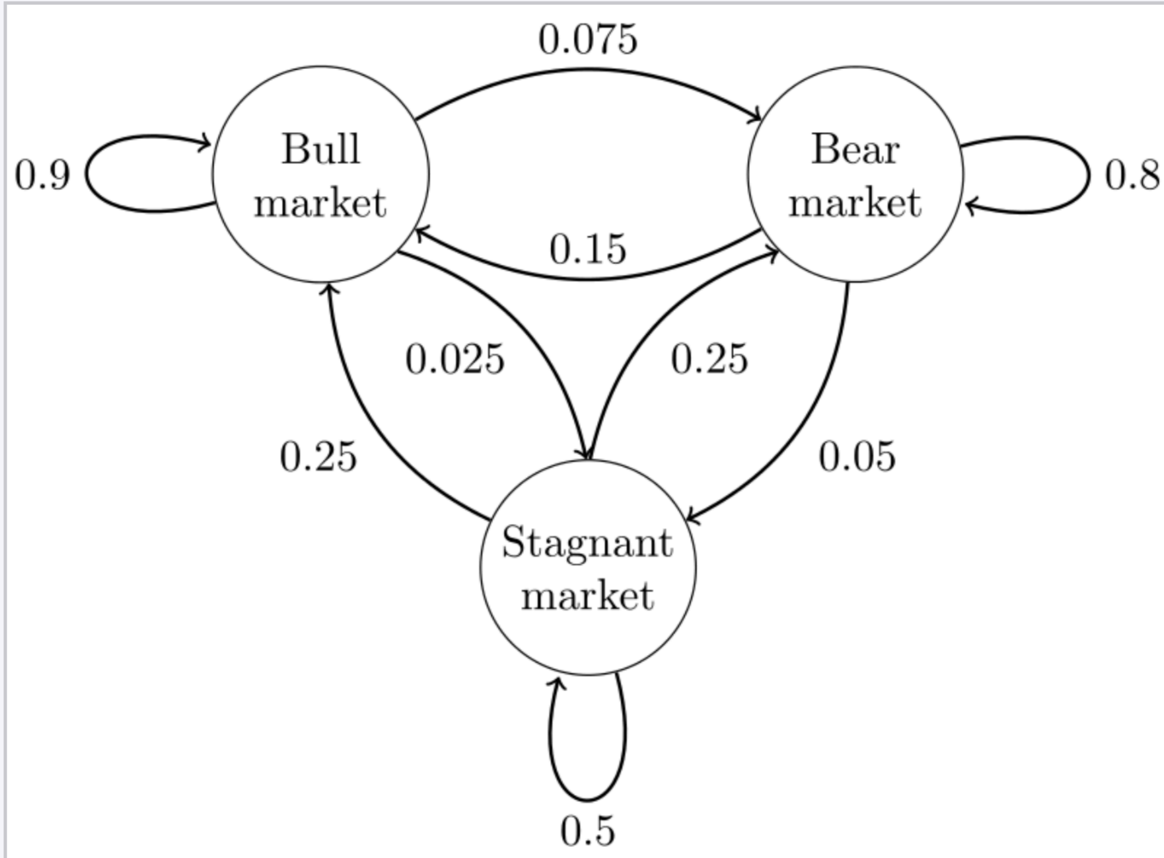
$$\begin{bmatrix} q_1 & q_2 \end{bmatrix} = \begin{bmatrix} 0.833 & 0.167 \end{bmatrix}$$

In conclusion, in the long term, about 83.3% of days are sunny.

Similarly you can use Markov Chains to predict stock trends

Stock market [edit]

A state diagram for a simple example is shown in the figure on the right, using a directed graph to picture the state transitions. The states represent whether a hypothetical stock market is exhibiting a bull market, bear market, or stagnant market trend during a given week.



Using a directed graph, the probabilities of the possible states a hypothetical stock market can exhibit is represented. The matrix on the left shows how probabilities corresponding to different states can be arranged in matrix form.

- https://en.wikipedia.org/wiki/Examples_of_Markov_chains

Exercise

- Please write out the stochastic matrix using the above graph (**called a probability graphic model**).

In real life modeling, often the situation is much more complicated

- We need to consider global economic environment.
- There are a lot of **hidden things which are not directly observable**.

Study two examples on Wikipedia

https://en.wikipedia.org/wiki/Hidden_Markov_model

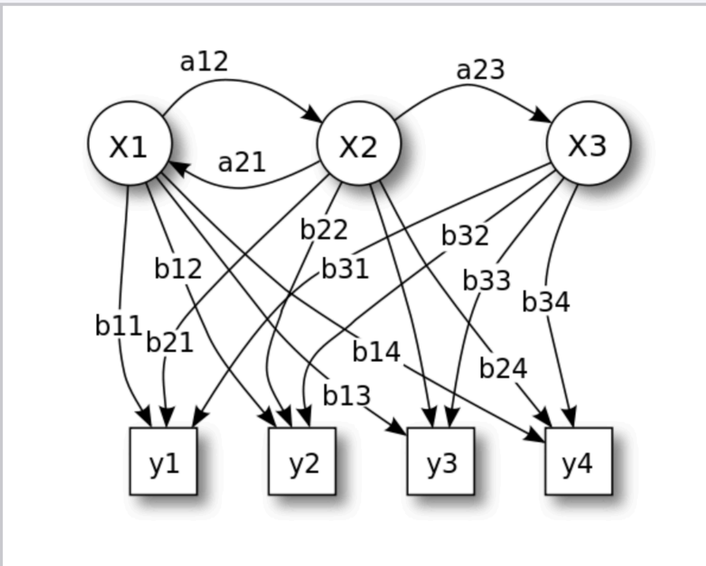
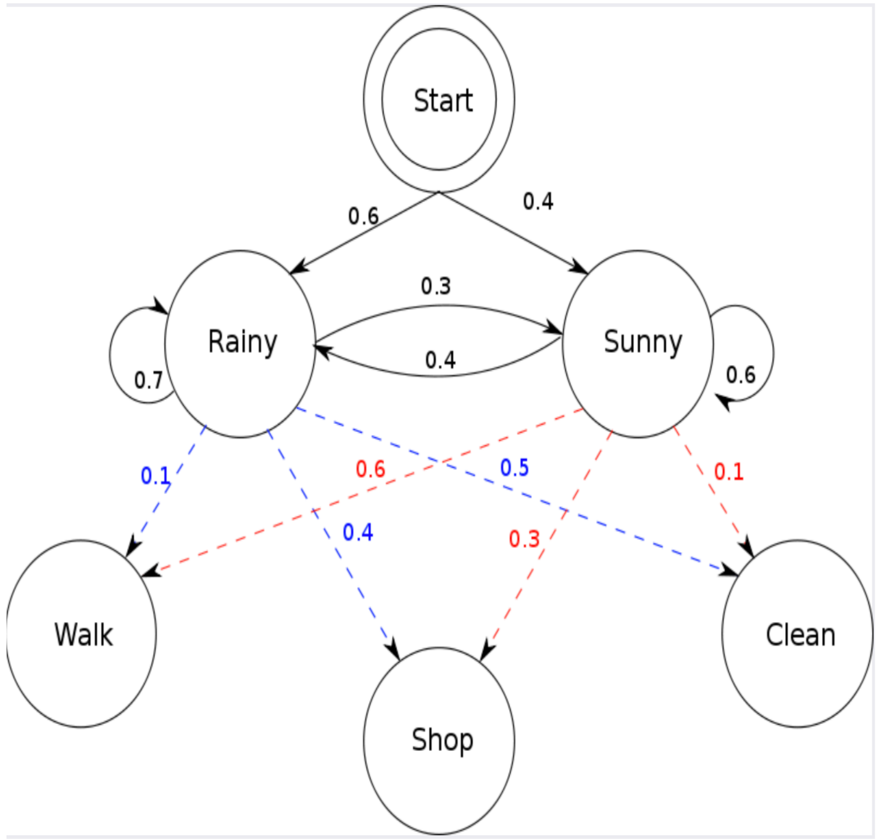


Figure 1. Probabilistic parameters of a hidden Markov model (example)
X — states
y — possible observations
a — state transition probabilities
b — output probabilities

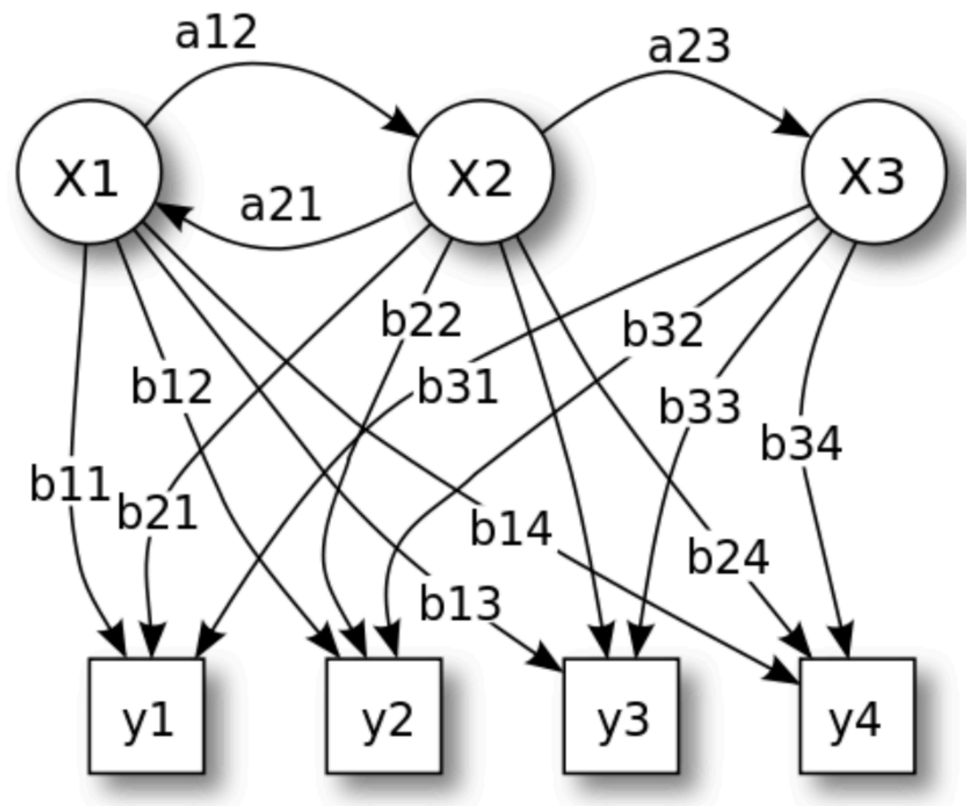


Hidden Markov Models

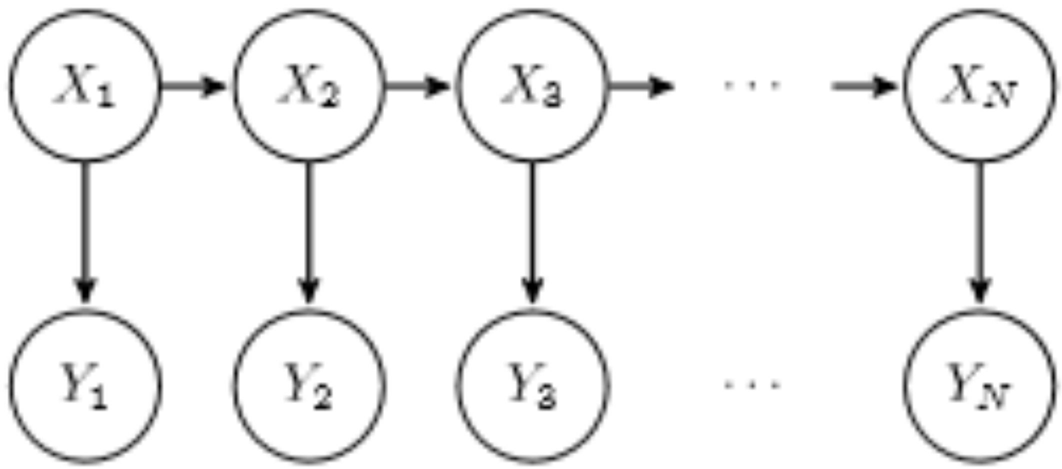
- Look an example on Wikipedia: https://en.wikipedia.org/wiki/Hidden_Markov_model

A = (a_{ij})
= Transition Matrix

B = (b_{kl})
= Emission Matrix.



Hidden Markov Models

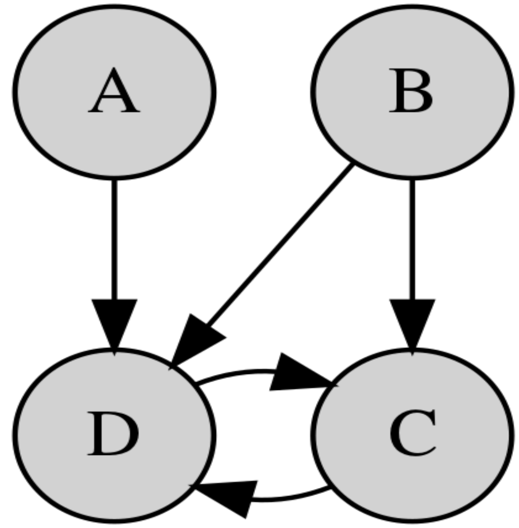


- Work out details with students on iPAD.
- Please see the detailed notes of HMM that I sent to you in email.

HMM is a typical example of a Probabilistic Graphical Model

- **What is a probabilistic Graphical Model?**
- **A probabilistic graphical model (PGM)** is a probabilistic model for which a graph expresses the conditional dependence structure between random variables. They are commonly used in probability theory, statistics—particularly Bayesian statistics—and machine learning.

Recall: F: X → Y.
We say Y is a function of X, i.e. Y depends on X.
Note: the arrow starts from X and ends on Y.



An example of a graphical model. Each arrow indicates a dependency. In this example: D depends on A, B, and C; and C depends on B and D; whereas A and B are each independent.

Note: there are 3 arrows starts from A, B, C and ends on D. This means D depends on A, B, and C.

Why using Probabilistic Graphical Models

- Generally, probabilistic graphical models use a graph-based representation as the foundation for encoding a distribution over a multi-dimensional space and a graph that is a compact or factorized representation of a set of independences that hold in the specific distribution.

Recall: Chain rule for random variables

Two random variables [edit]

For two random variables X, Y, to find the joint distribution, we can apply the definition of conditional probability to obtain:

$$\mathrm{P}(X,Y) = \mathrm{P}(X|Y) \cdot P(Y)$$

More than two random variables [edit]

Consider an indexed collection of random variables $X_1, \ldots, X_n$. To find the value of this member of the joint distribution, we can apply the definition of conditional probability to obtain:

$$\mathrm{P}(X_n, \ldots, X_1) = \mathrm{P}(X_n | X_{n-1}, \ldots, X_1) \cdot \mathrm{P}(X_{n-1}, \ldots, X_1)$$

Repeating this process with each final term creates the product:

$$\mathrm{P}\left(\bigcap_{k=1}^{n} X_k\right) = \prod_{k=1}^{n} \mathrm{P}\left(X_k \,\middle|\, \bigcap_{j=1}^{k-1} X_j\right)$$

Recall: Probability Chain Rule for Events

The chain rule for two random events A and B says

$$P(A \cap B) = P(B \mid A) \cdot P(A).$$

For more than two events $A_1, \ldots, A_n$ the chain rule extends to the formula

$$\mathrm{P}(A_n \cap \ldots \cap A_1) = \mathrm{P}(A_n | A_{n-1} \cap \ldots \cap A_1) \cdot \mathrm{P}(A_{n-1} \cap \ldots \cap A_1)$$

which by induction may be turned into

$$\mathrm{P}(A_n \cap \ldots \cap A_1) = \prod_{k=1}^{n} \mathrm{P}\left(A_k \,\middle|\, \bigcap_{j=1}^{k-1} A_j\right).$$

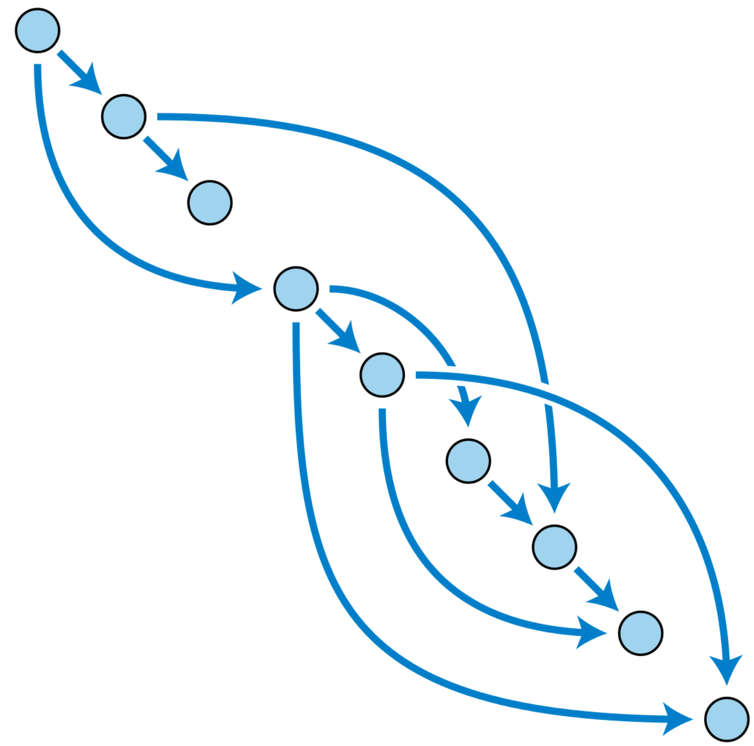
Example [edit]

With four events ($n = 4$), the chain rule is

$$\begin{aligned}
\mathrm{P}(A_4 \cap A_3 \cap A_2 \cap A_1) &= \mathrm{P}(A_4 \mid A_3 \cap A_2 \cap A_1) \cdot \mathrm{P}(A_3 \cap A_2 \cap A_1) \\
&= \mathrm{P}(A_4 \mid A_3 \cap A_2 \cap A_1) \cdot \mathrm{P}(A_3 \mid A_2 \cap A_1) \cdot \mathrm{P}(A_2 \cap A_1) \\
&= \mathrm{P}(A_4 \mid A_3 \cap A_2 \cap A_1) \cdot \mathrm{P}(A_3 \mid A_2 \cap A_1) \cdot \mathrm{P}(A_2 \mid A_1) \cdot \mathrm{P}(A_1)
\end{aligned}$$

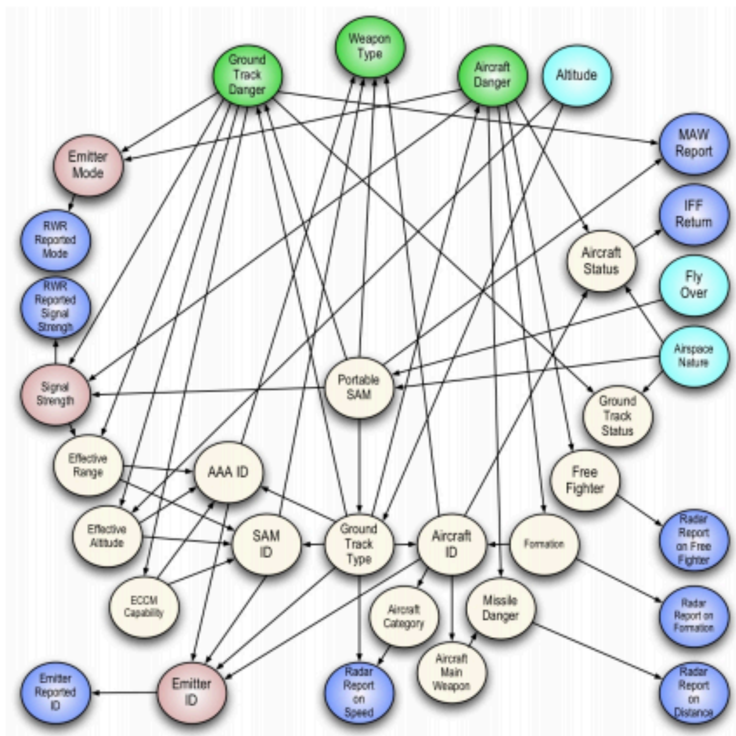
HMM is a typical example of Directed Acyclic Graph (DAG)

- A DAG is a finite directed graph with no directed cycles. That is, it consists of finitely many vertices and edges (also called *arcs*), with each edge directed from one vertex to another, such that there is no way to start at any vertex *v* and follow a consistently-directed sequence of edges that eventually loops back to *v* again. Equivalently, a DAG is a directed graph that has a topological ordering, a sequence of the vertices such that every edge is directed from earlier to later in the sequence.



There are many applications of DAG: Radar and Aircraft Control

- Modeling multiple planes and radar signals:



https://pr-owl.org/basics/bn.php

HMM and Directed Acyclic Graphical (DAG) Prob. Models

DAG models use a factorization of the joint distribution,

$$p(x_1, x_2, \ldots, x_d) = \prod_{j=1}^{d} p(x_j | x_{\text{pa}(j)}),$$

where pa(j) are the "parents" of node j.

This assumes a Markov property (generalizing Markov property in chains),

$$p(x_j | x_{1:j-1}) = p(x_j | x_{\text{pa}(j)}),$$

Note: Also can factor into blocks

Instead of factorizing by variables j, could factor into blocks b:

$$p(x) = \prod_b p(x_b \mid x_{\mathsf{pa}(b)}),$$

and have the nodes be blocks.

- Usually assuming full connectivity within the block.

We will work out an example on HMM using iPAD on how to factor into blocks after the slides.

Review of Independence

- Let A and B are random variables taking values $a \in \mathcal{A}$ and $b \in \mathcal{B}$.
- We say that A and B are independent if we have

$$p(a, b) = p(a)p(b),$$

  for all a and b.
- To denote independence of x_i and x_j we use the notation

$$x_i \perp x_j.$$

- In a product of Bernoullis, we assume $x_i \perp x_j$ for all i and j.

Review of Independence

- For independent a and b we have

$$p(a \mid b) = \frac{p(a,b)}{p(b)} = \frac{p(a)p(b)}{p(b)} = p(a).$$

- This gives us a more intuitive definition: A and B are independent if

$$p(a \mid b) = p(a)$$

  for all a and $b \neq 0$.
  - In words: knowing b tells us nothing about a (and vice versa).
    - This will tend to simplify calculations involving a.

- Useful fact: $a \perp b$ iff $p(a,b) = f(a)g(b)$ for some functions f and g.

Conditional Independence

- We say that A is conditionally independent of B given C if

$$p(a, b \mid c) = p(a \mid c)p(b \mid c),$$

  for all a, b, and $c \neq 0$.
- Equivalently, we have

$$p(a \mid b, c) = p(a \mid c).$$

- "If you know C, then *also* knowing B would tell you nothing about A"'.
  - In mixture of Bernoullis, given cluster there is no dependence between variables.

- We often write this as

$$A \perp B \mid C.$$

- In a mixture of Bernoullis, we assume $x_i \perp x_j \mid z$ for all i and j.
  - This simplifies calculations involving x_i and x_j, provided that we know z.

Extra Conditional Independences in Markov Chains

- In Markov chains, the Markov assumption is $x_j \perp x_1, x_2, \ldots, x_{j-2} \mid x_{j-1}$,

$$p(x_j \mid x_{j-1}, x_{j-2}, \ldots, x_1) = p(x_j \mid x_{j-1}).$$

- But note that this also implies the additional conditional independence that

$$p(x_j \mid x_{j-2}, x_{j-3}, \ldots, x_1) = p(x_j \mid x_{j-2}).$$

- We can use this property to easily compute $p(x_j \mid x_{j-2}, x_{j-3}, \ldots, x_1)$:

$$\begin{aligned}
p(x_j \mid x_{j-2}, x_{j-3}, \ldots x_1) &= p(x_j \mid x_{j-2}) \\
&= \sum_{x_{j-1}} p(x_j, x_{j-1} \mid x_{j-2}) \\
&= \sum_{x_{j-1}} p(x_j \mid x_{j-1}, x_{j-2}) p(x_{j-1} \mid x_{j-2}) \\
&= \sum_{x_{j-1}} \underbrace{p(x_j \mid x_{j-1})}_{\text{tran prob}} \underbrace{p(x_{j-1} \mid x_{j-2})}_{\text{tran prob}}.
\end{aligned}$$

DAGs and Conditional Independence

- Conditional independences can substantiall simplify inference.
- But it's tedious to formally show that the above are true.
  - See the last slide, and the EM notes.

- In DAGs we make the conditional independence assumption that

$$p(x_j \mid x_{j-1}, x_{j-2}, \ldots, x_1) = p(x_j \mid x_{\text{pa}}(j)).$$

- Is there an easy way to find out what other independences are ture?
  - If so, we could quickly find out which calculations are easy to do in a given DAG.

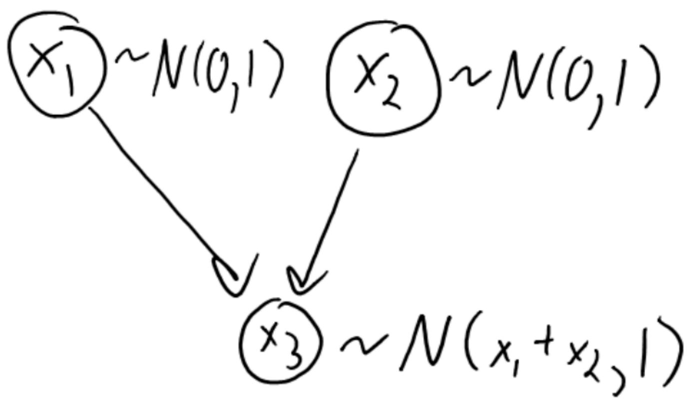
D-Separation: From Graphs to Conditional Independence

- All conditional independences implied by a DAG can be read from the graph.
- In particular: variables A and B are conditionally independent given C if:
  - "D-separation blocks all undirected paths in the graph from any variable in A to any variable in B."
- In the special case of product of independent models our graph is:

- Here there are no paths to block, which implies the variables are independent.
- Checking paths in a graph tends to be faster than tedious calculations.
  - We can start connecting properties of graphs to computational complexity.

D-Separation as Genetic Inheritance

- The rules of d-separation are intuitive in a simple model of gene inheritance:
  - Each person has single number, which we'll call a "gene".
  - If you have no parents, your gene is a random number.
  - If you have parents, your gene is a sum of your parents plus noise.
- For example, think of something like this:



- Graph corresponds to the factorization $p(x_1, x_2, x_3) = p(x_1)p(x_2)p(x_3 \mid x_1, x_2)$.
  - In this model, does $p(x_1, x_2) = p(x_1)p(x_2)$? (Are x_1 and x_2 independent ?)

D-Separation as Genetic Inheritance

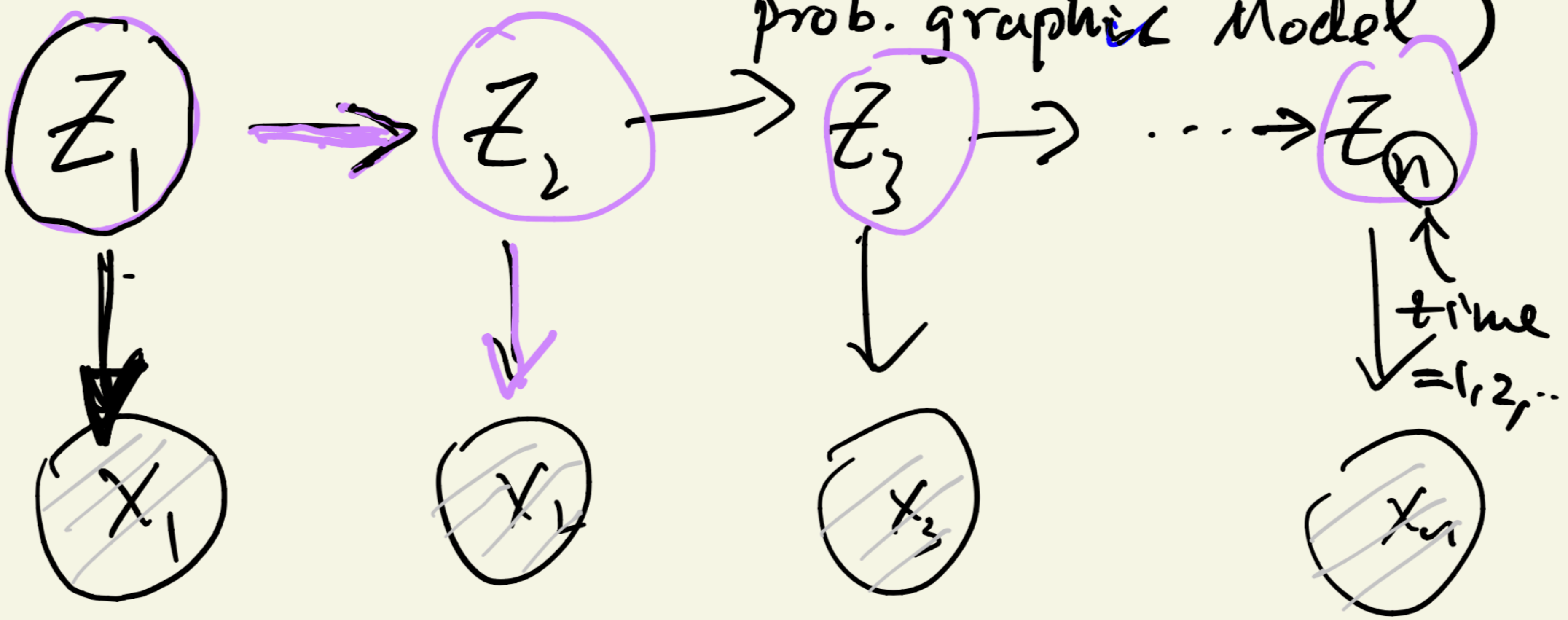
- Genes of people are independent if knowing one says nothing about the other.
- Your gene is dependent on your parents:
  - If I know you your parent's gene, I know something about yours.
- Your gene is independent of your (unrelated) friends:
  - If know you your friend's gene, it doesn't tell me anything about you.
- Genes of people can be conditionally independent given a third person:
  - Knowing your grandparent's gene tells you something about your gene.
  - But grandparent's gene isn't useful if you know parent's gene.

Hidden Markov Models

- Work out details with students on iPAD.
- Please see the detailed notes of HMM that I sent to you in email.

Hidden Markov Model (HMM)

Trellis diagram (a special kind of prob. graphic Model)



Hidden variables

$$\overbrace{z_1, \cdots, z_n}^{\text{Hidden variables}} \in \{1, 2, \cdots, m\}$$

$$\underbrace{x_1, \cdots\cdots x_n}_{\text{"observable" data}} \in \mathcal{X} \left(\begin{array}{l} \text{discrete}, \\ \mathbb{R}, \mathbb{R}^d, \cdots \end{array} \right)$$

"observable" data

Key: The Joint prob factorized in following way:

Initial prob, transition prob, Emission prob.

$$P(z_1, z_2, \cdots, z_n, x_1, \cdots, x_n) = \underbrace{P(z_1)}_{\text{Initial prob}} P(x_1 | z_1) \prod_{k=2}^{n} \underbrace{P(z_k | z_{k-1})}_{\text{transition prob.}} \underbrace{P(x_k | z_k)}_{\text{Emission prob.}}$$