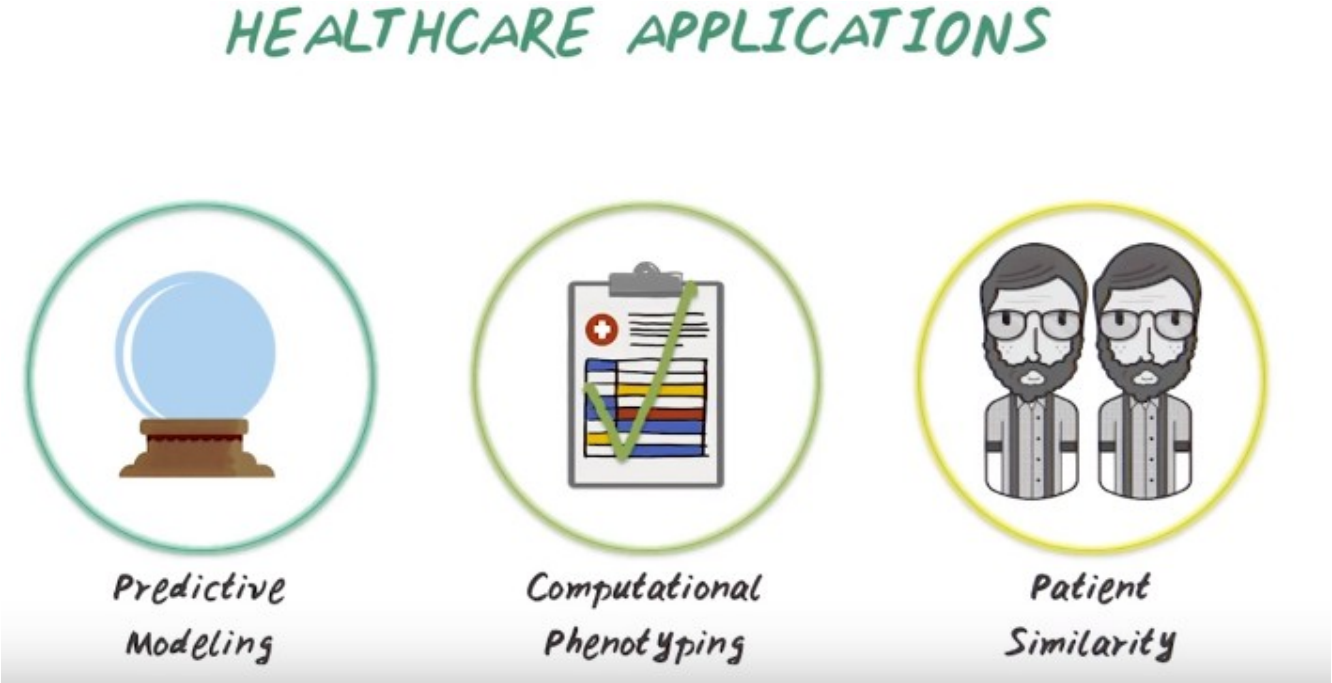
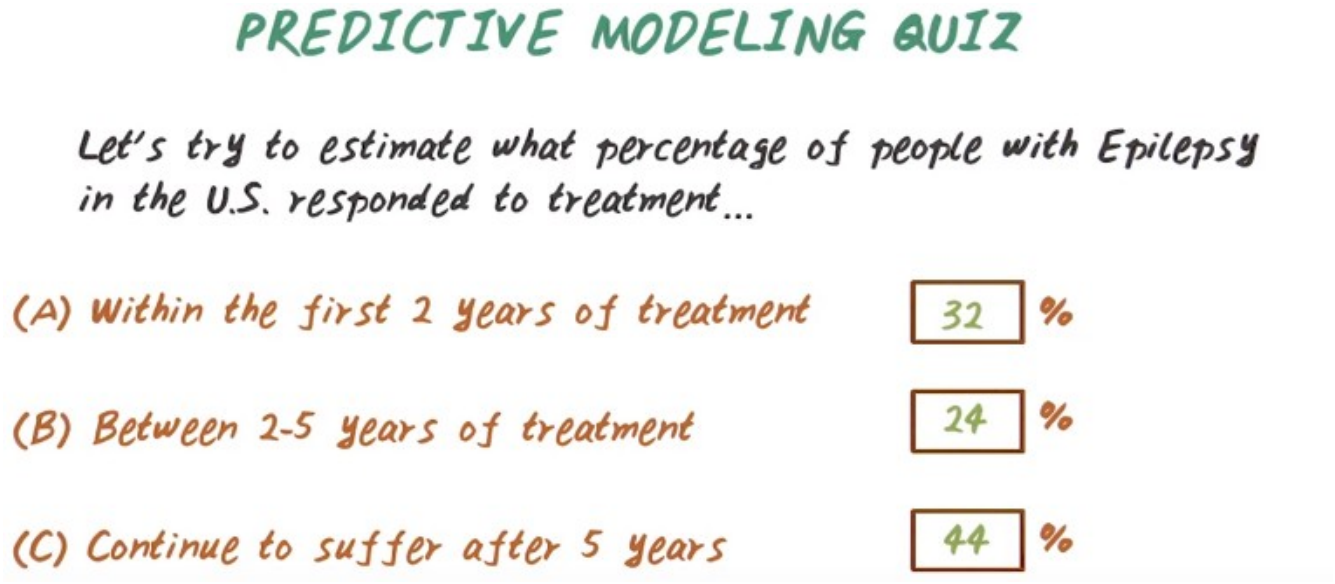
1. In the first lesson, we talked about big data analytics for healthcare as a whole and what we expect from this course overall. Now, let's break down the actual topics we will cover. We'll start by talking about some of the healthcare applications for big data. Then we'll talk about some of the algorithms we will use in those applications. And then we'll talk about some of the software systems that we will use to implement those algorithms and to support those applications. If you look at the calendar, you'll see that throughout the semester, we alternate among those general topics. So let's start by talking about healthcare applications.

2. To understand this course in general let's look at the big picture. So, this course we'll talk about three different things. We'll talk about big data systems. We also will introduce scalable machine learning algorithms. Then we'll talk about healthcare applications. And see how we can use those mission learning algorithms and big data system together to solve healthcare problems. So let's start by talking about healthcare applications.



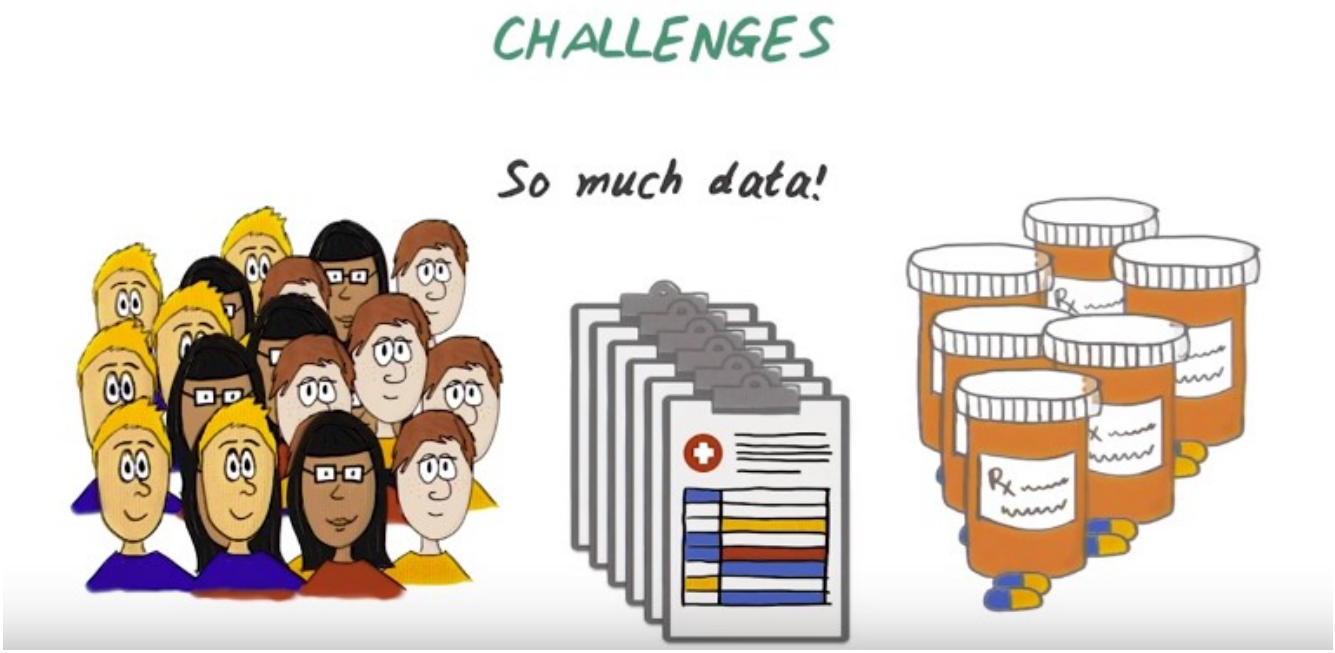
3. We'll talk about three types of healthcare applications in this course. Predictive modeling is about using historical data to view the model for predicting future outcome. Computational phenotyping is about turning messy electronic health records into meaningful clinical concepts. And patient similarity, it uses health data to identify groups of patients sharing similar characteristics. We'll begin with predictive modeling.

4. Predictive modeling is about using historical data to view the model for predicting future events. For example, we want to predict which treatment is likely to work for an epilepsy patient. Why do you we want to do predictive modelling? Let's motivate predictive modeling with. let's try to estimate which percentage patient was epilepsy in US responded to treatment Group A within first two years of treatment, Group B between two to five years of treatment, and Group C continued to suffer even after five years of treatment. So write your number in those boxes, and they should probably add up to a hundred.

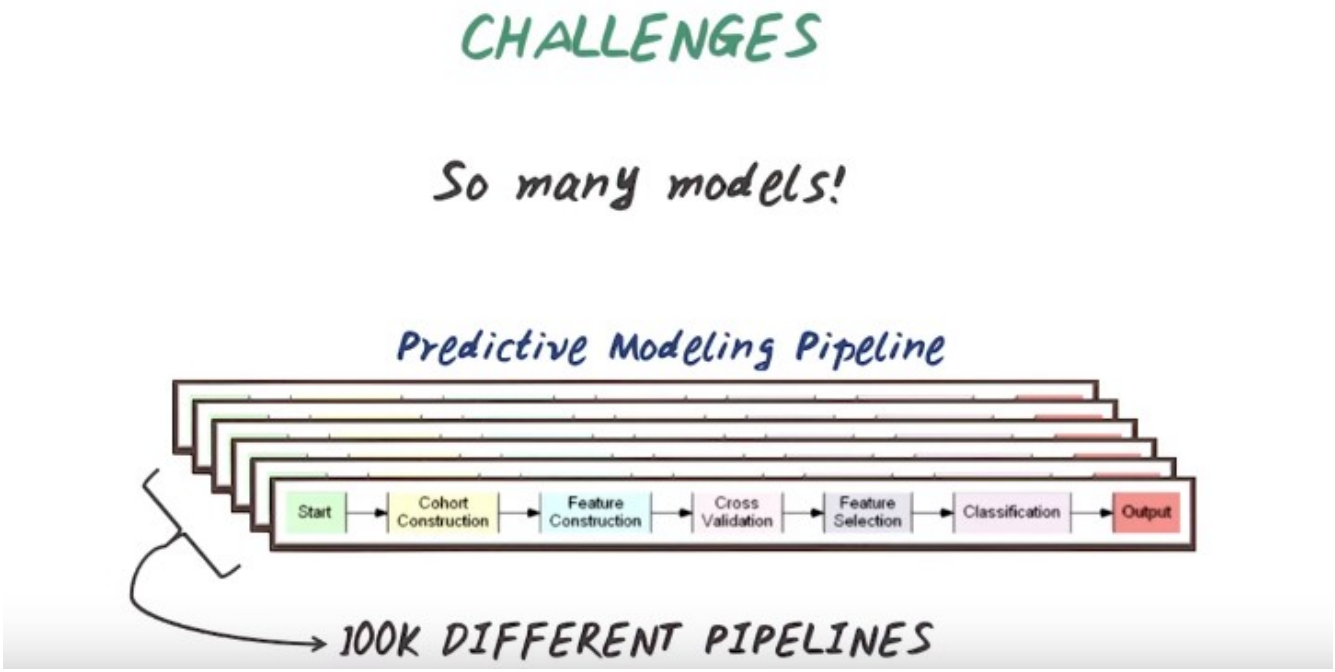


Epilepsy: 癲癇

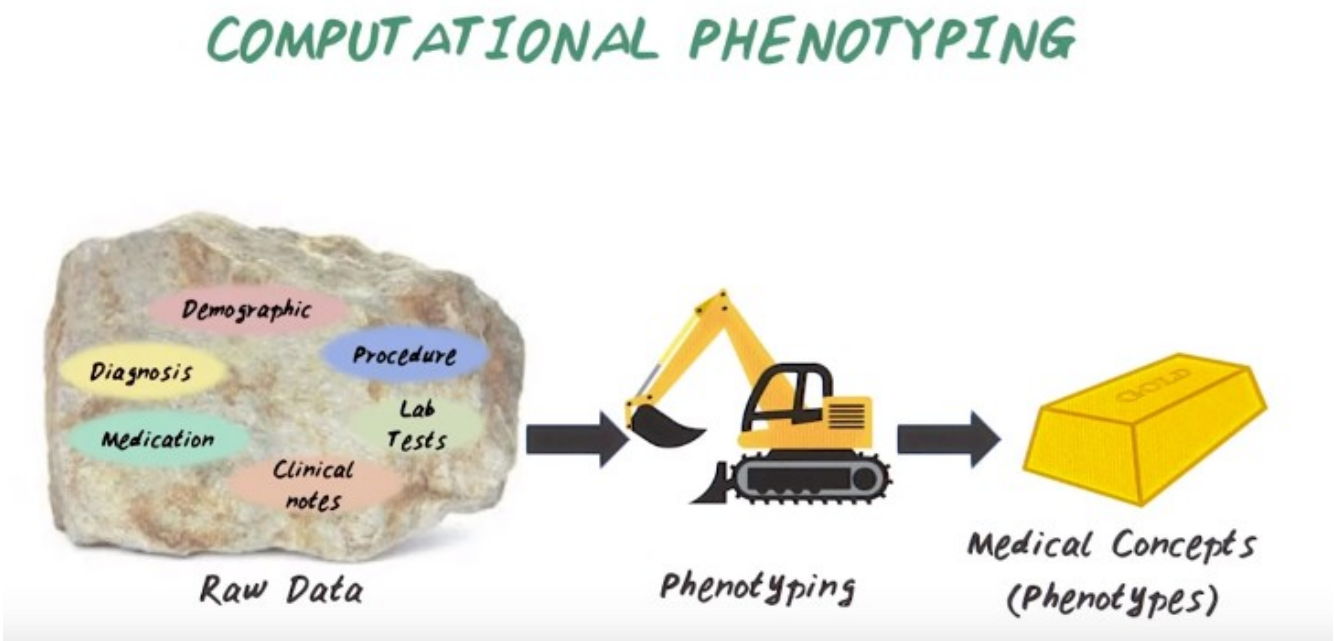
5. Here's the answer, group A, within the first two years of treatment, only 32% of patient in that group. Group B, between two and five years, there are 24% of patient, and group C continue to suffer after five years, there are 44% of patient. So clearly this is a problem, we like group A to be the majority. Predictive modeling would help improve matching patients to the right treatment quickly. So that late responders in group B will be come early responder in group A. Also this will help identify non-responder in group C quickly, so that new treatment can be developed for them.



6. So what makes predictive modeling difficult? We have millions of patients we want to analyze and their diagnosis information, medication information, and so on. So all this data combined together, create a big challenge.

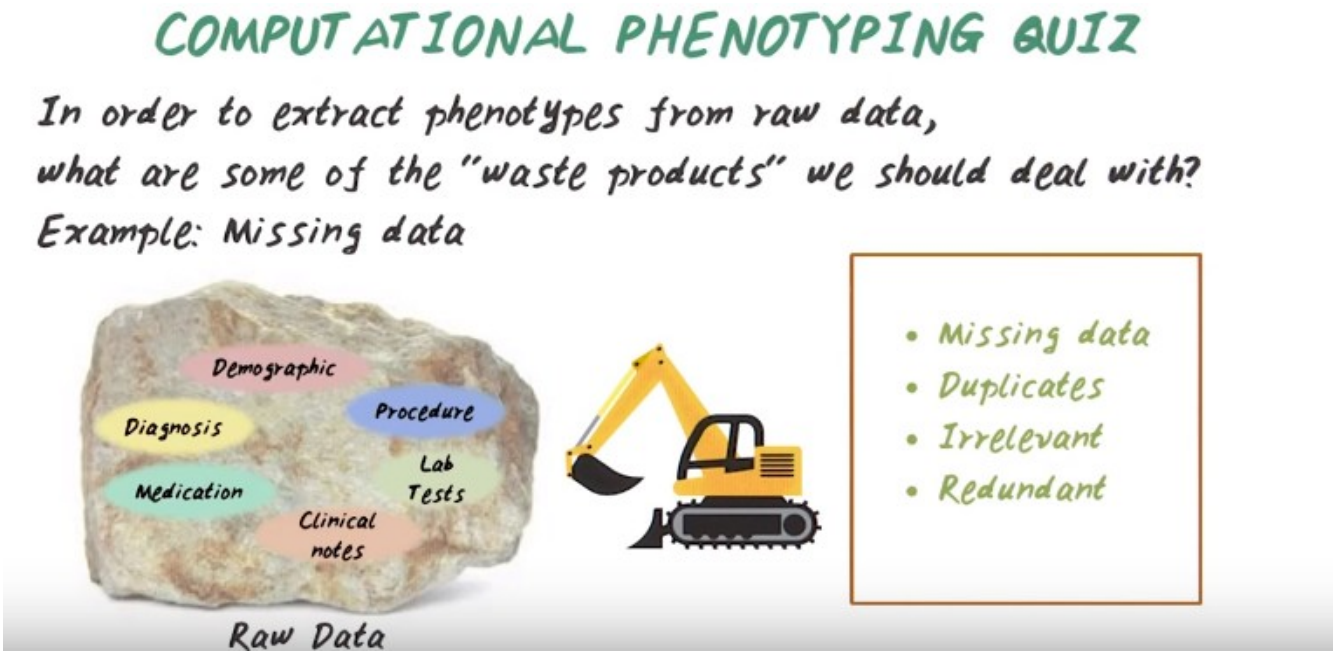


The second challenge in predictive modeling is there's so many models to be built. Predictive modeling is not a single algorithm, it's a sequence of computational tasks. We'll introduce predictive modeling pipeline in more details in a later lecture. But every steps in this pipeline has many different options. All of those combined give us many, many pipelines to be evaluated and compared.

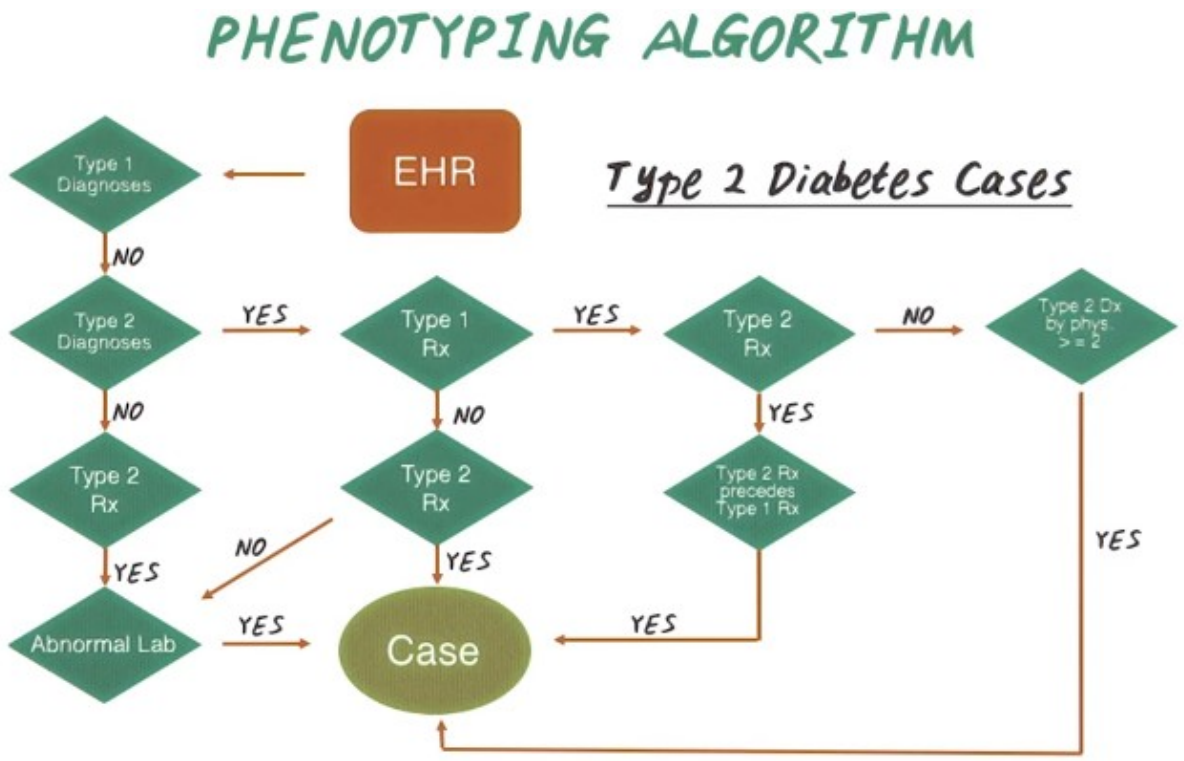


7. We just talked about predictive modeling. Next, let's talk about computational phenotyping. The input to computational phenotyping is the raw patient data. It consists of many different sources such as demographic information, diagnosis, medication, procedure, lab test, and clinical notes. And phenotyping is the process of turning the raw patient data into medical concepts or phenotypes.

8. To help us understand phenotyping better, let's do a quiz. Imagine you're trying to extract phenotypes from this raw data, so what are the waste products we should deal with? For example, missing data could be one. So write down some of those wast product in this box.



9. Okay, here are some possible answers. Missing value, some important data may be missing from the raw data. We have to deal with that. Duplicates, some patient record may show up multiple times due to recording errors. Irrelevant data, not all the raw information are relevant for a specific task. We want to get rid of those irrelevant information. Redundant information, different data records can indicate the same underlying problems. For example, both diagnosis and medication records from a patient indicates underlying condition of Type 2 diabetes. So we want to consolidate those redundant information.

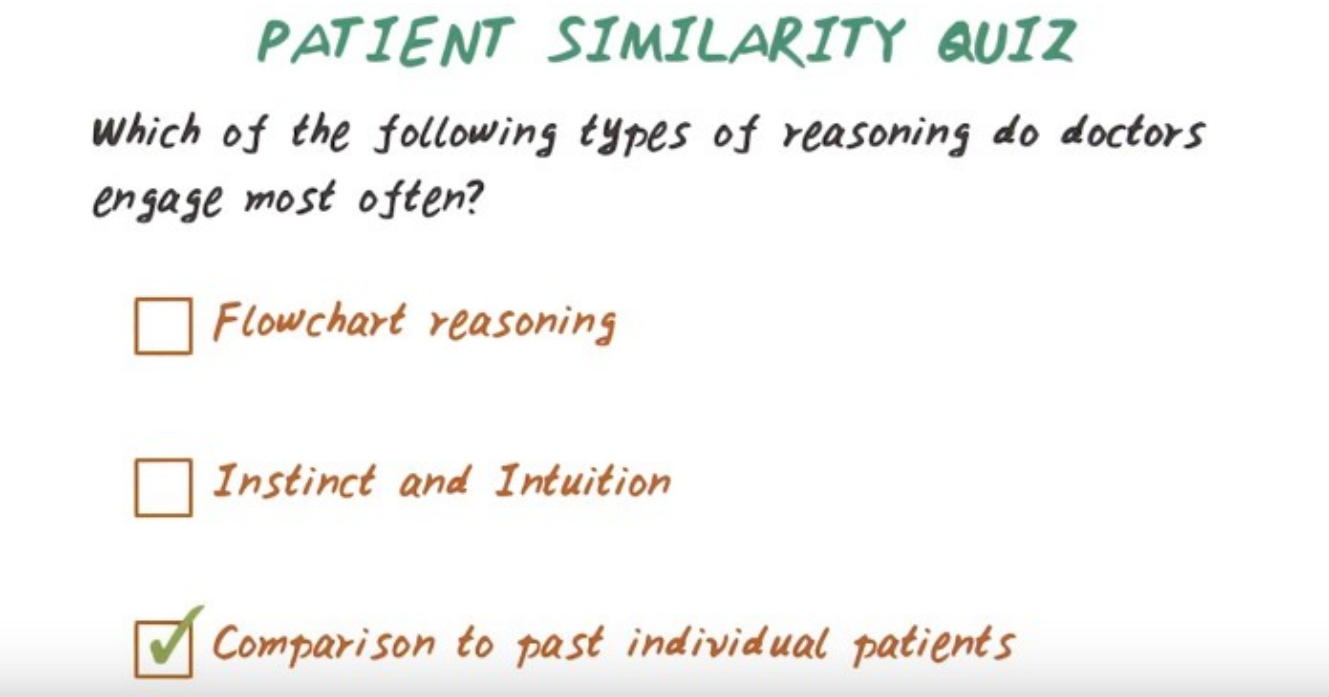


上圖的意思不是說 醫生如何確定一個人有沒有 type 2 diabetes, 而是說 data scientist 拿到一個人的病歷, 如何根据他病歷上記錄的活動來確定他有沒有 type 2 diabetes

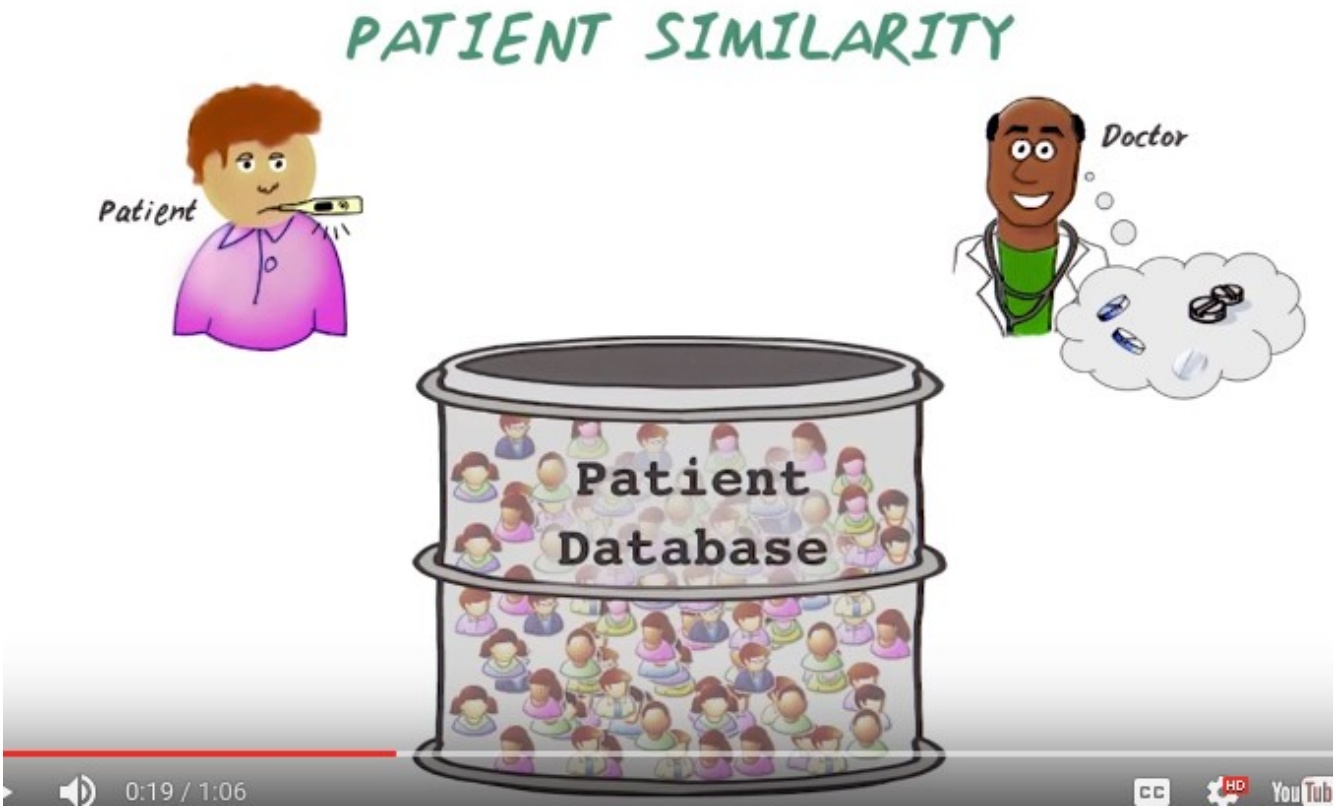
10. Next, let's see an example of phenotyping algorithm for type 2 diabetes. So the input to the algorithm is EHR, Electronic Health Record of a patient. Then we'll first check whether the patient record indicate type 1 diabetes diagnosis. If the answer is no, then we continue checking with a Type 2 Diabetes diagnosis is present. If again no, would check whether Type 2 medication (即 Type 2 Rx) is given. Then if the answer is yes, we check whether any abnormal lapse is present. If yes, then we confirm this patient record, indicate this patient has Type 2 Diabetes. And this is not the only way to identifying Type Two Diabetes cases. There's a different path. For example, from here we can check Type 2 Diabetes diagnosis, if the answer is yes, we'll check medication for Type 1 Diabetes. If the answer is no, we'll check medication for Type 2 Diabetes. And if this answer is no, then we go back to check for abnormal labs. If the answer is yes, again this record indicates Type 2 Diabetes. If at this stage the Type 2 Diabetes medication is confirmed, then immediately we know this record indicates Type 2 Diabetes patient. And this is still not the complete algorithm. There's two other paths can lead to type 2 diabetes. This entire flow chart give us one example of phenotyping algorithm for type 2 diabetes. So you may wonder why do we need such a complicated algorithm to determine whether patient have Type 2 Diabetes. Can we just ask whether patient have Type 2 Diabetes diagnoses present in the data? Shouldn't that be enough? The answer is no. The reason is because electronic health record data is very unreliable. There are missing data, redundant information, so sometimes for Type 2 Diabetes patient, the diagnosis is not present in the record. So we still have other way to check whether our Type 2 Diabetes patient, for example, their medication, lab tests. So that's why it's not sufficient just checking one source of information. At the same time, even the Type 2 Diabetes diagnosis is present, it's not necessarily confirm the patient has Type 2 Diabetes, because patient can come to the clinics for a check up, for screening purpose, then this diagnosis code can still be present in the data. So we have to check additional things, such as medication, and lab tests to really confirm this patient

has Type 2 Diabetes. In this class, we'll learn how to develop phenotyping algorithm look like this from data, and also how to implement such algorithm efficiently using big data systems.

11. Okay, so we talked about predictive modeling application phenotyping. Next, we'll introduce patient similarity. To motivate patient similarity let's do another quiz. So, which of the following type of reason do doctors engage most often during patient encounters. Is it based on flowchart reasoning like what we have seen it phenotyping algorithm? Or is it based on her instinct and intuition? Or is it comparison to past individual patients?

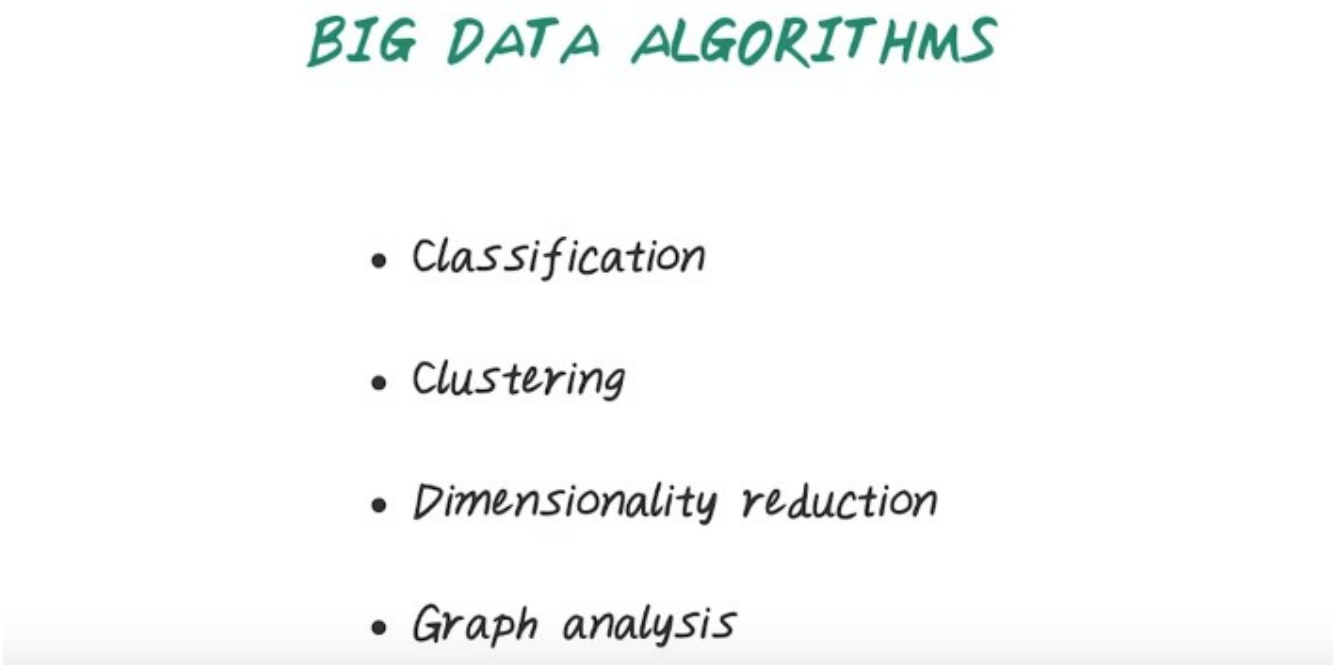


12. So the correct answer is comparison to the past individual patients or case based reasoning. Based on our anecdotal experiences, doctor often compared the current patient to the old patient they have seen.

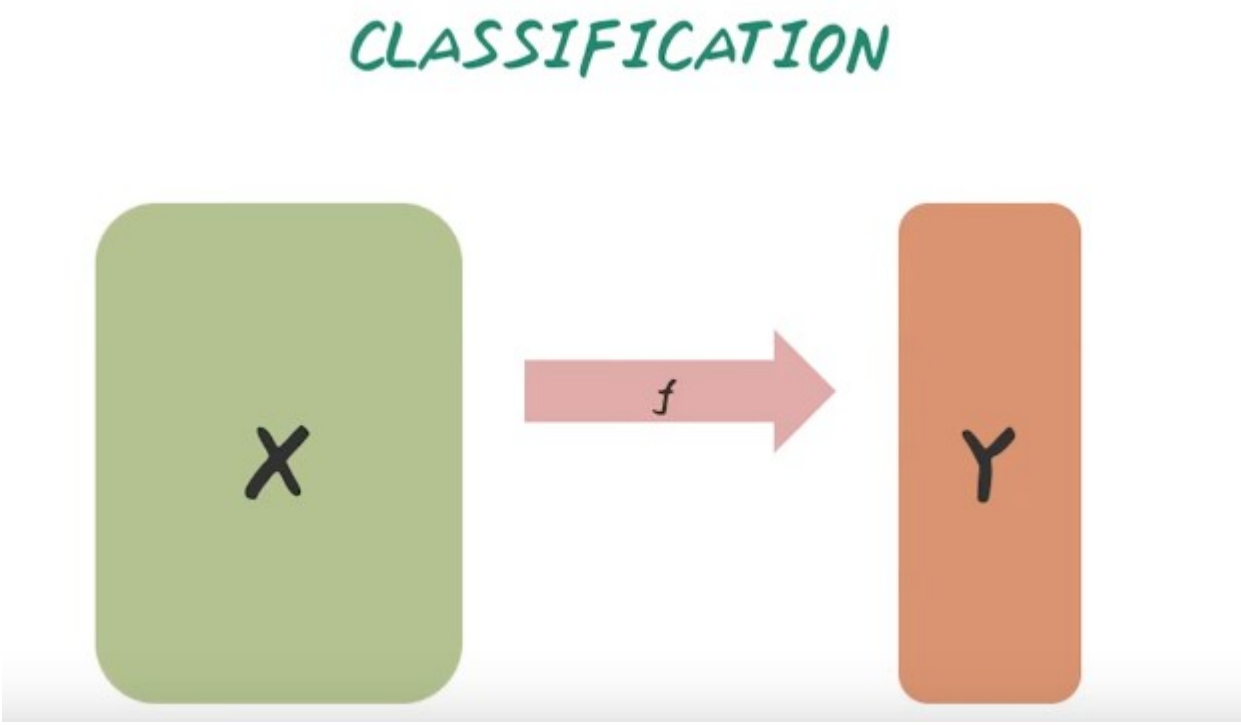


13. So patient similarity is about simulating the doctor's case based with the computer algorithms. Instead of depending on one doctor's memory, wouldn't it be nice if we can leverage all the patient data in the entire database? So the idea is when the patient comes in, the doctor does some examination on the patient. Then, based on that information, we can do a similarity search through the database. Find those potentially similar patients, then doctor can provide some supervision on that result to find those truly similar patients to the specific clinical context. Then we can group those patients, based on what treatment they are taking, and look at what outcome they're getting. Then recommend the treatment with the best outcome to the current patient. And that's what patient similarity does. So in this course, we'll learn about different patient similarity algorithms and how to implement that in an efficient manner using big data systems .

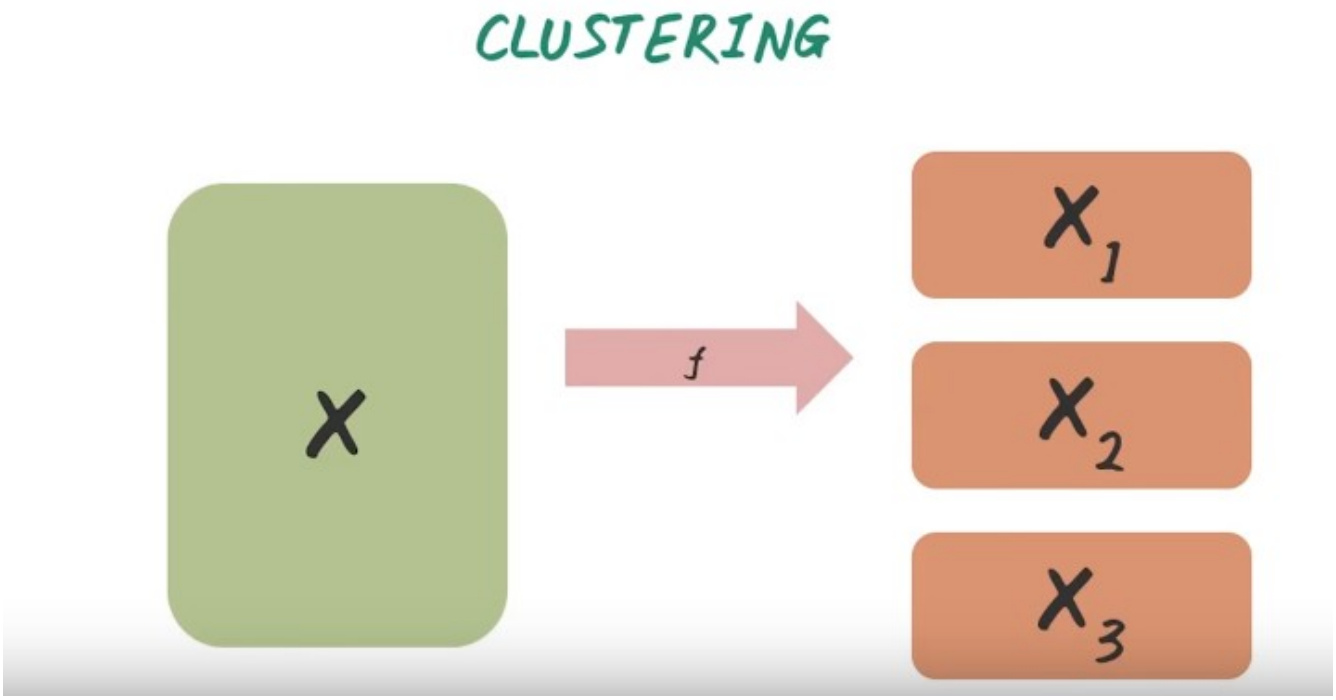
14. So far, we talked about health care applications. Next, we introduce what machine learning algorithms will be covered in this course.



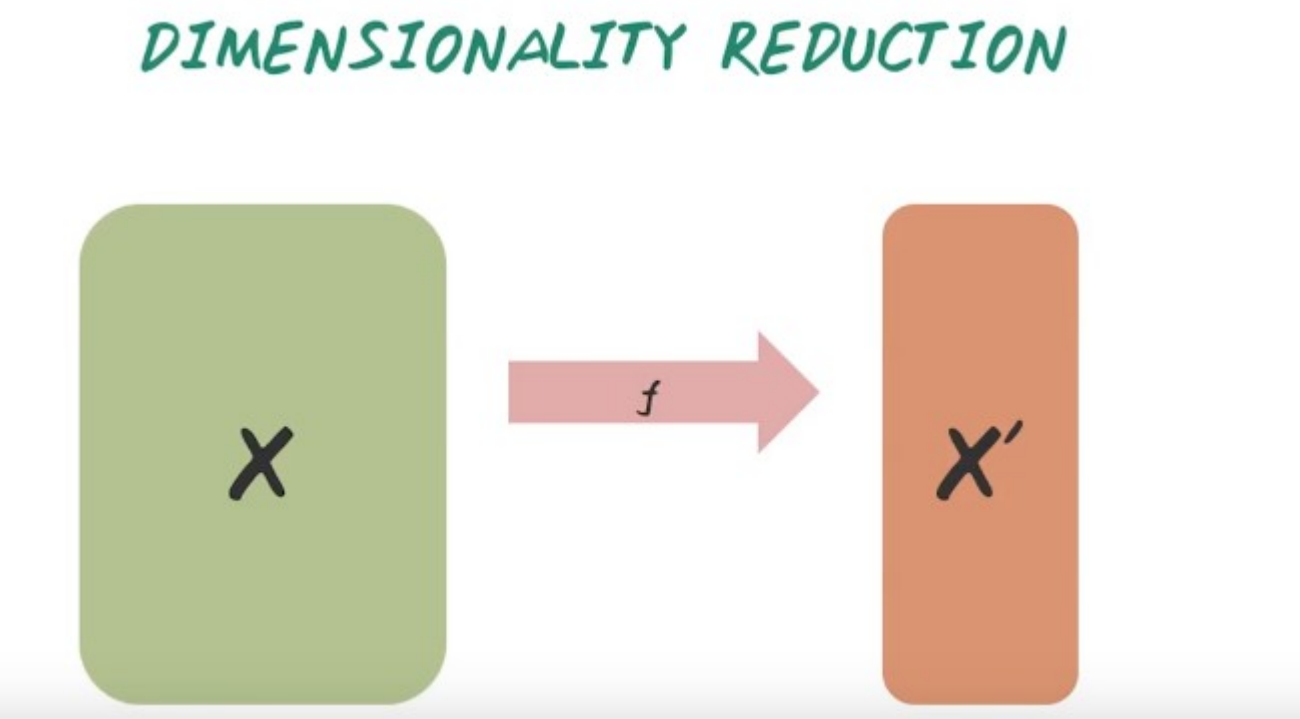
So in this course, we'll cover big data algorithms. We'll talk about classification algorithms, clustering algorithms, dimensionality reduction algorithms and graph analysis.



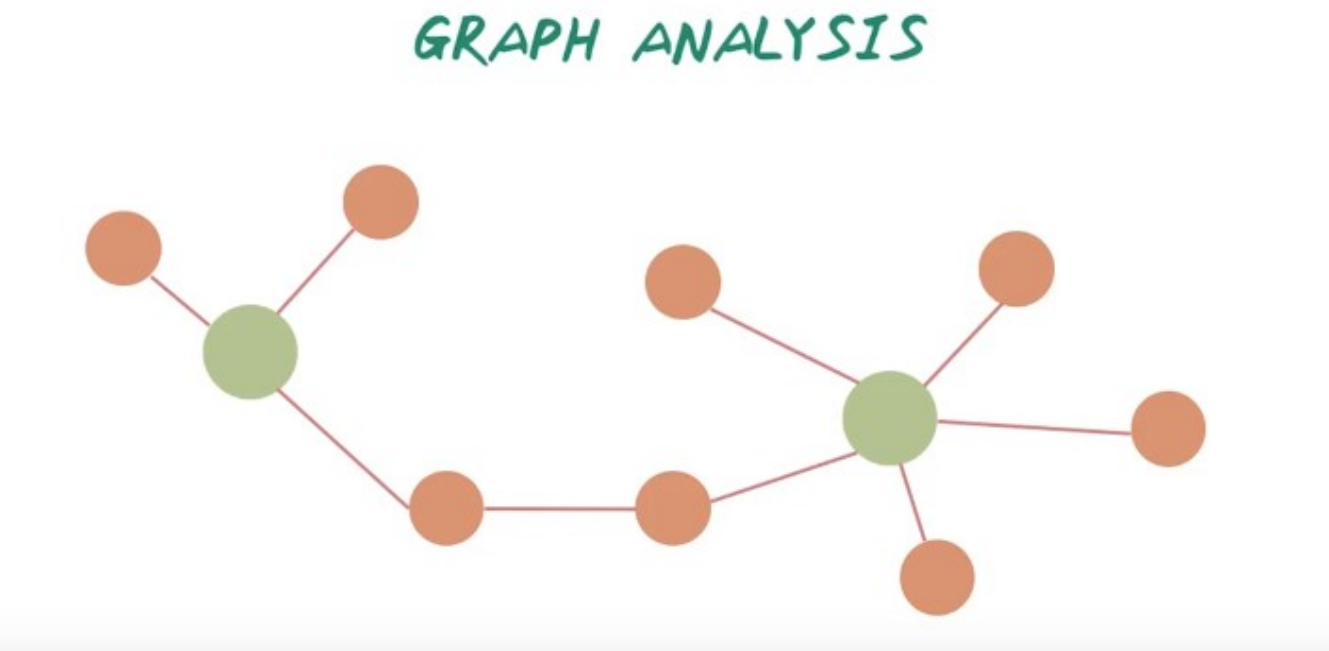
First, let's talk about classification algorithm. So given a matrix X, here every row represent a patient, every column represent a disease and every element here indicate whether a specific patient has a specific disease. Then, we learn a function f that map each patient to a target variable y. For example, here, the target could be whether a patient had a heart attack or not in next six months.



And we'll also talk about clustering algorithms. So here the input of clustering algorithm is similar to classification. So we have a matrix x, every row represent a patient, every column represent a disease, and we want to learn a function f that partition the set of patient into different clusters. For example, x1, x2, x3 represent different patient clusters. And the patient within a cluster are similar to each other and they're different from patient in different clusters.



We'll also talk about dimensionality reduction algorithm. Here the input is a large matrix of the set of patients with large number of features. And the output of dimensionality reduction is a smaller matrix, x prime, that consists of the same set of patients with smaller number of features. There are different ways to construct those features. Sometimes those features are good summary of all the feature in the original matrix. Sometimes those are the only features we care about in order to predict a specific target.



We also learned graph analysis. For example, we have two patient here, and we connect those patients to a set of diseases they have and also connect the diseases are related to each other. Given this network of patients and diseases, and we want to learn what are the important patients or disease in this network and also how they related to each other?

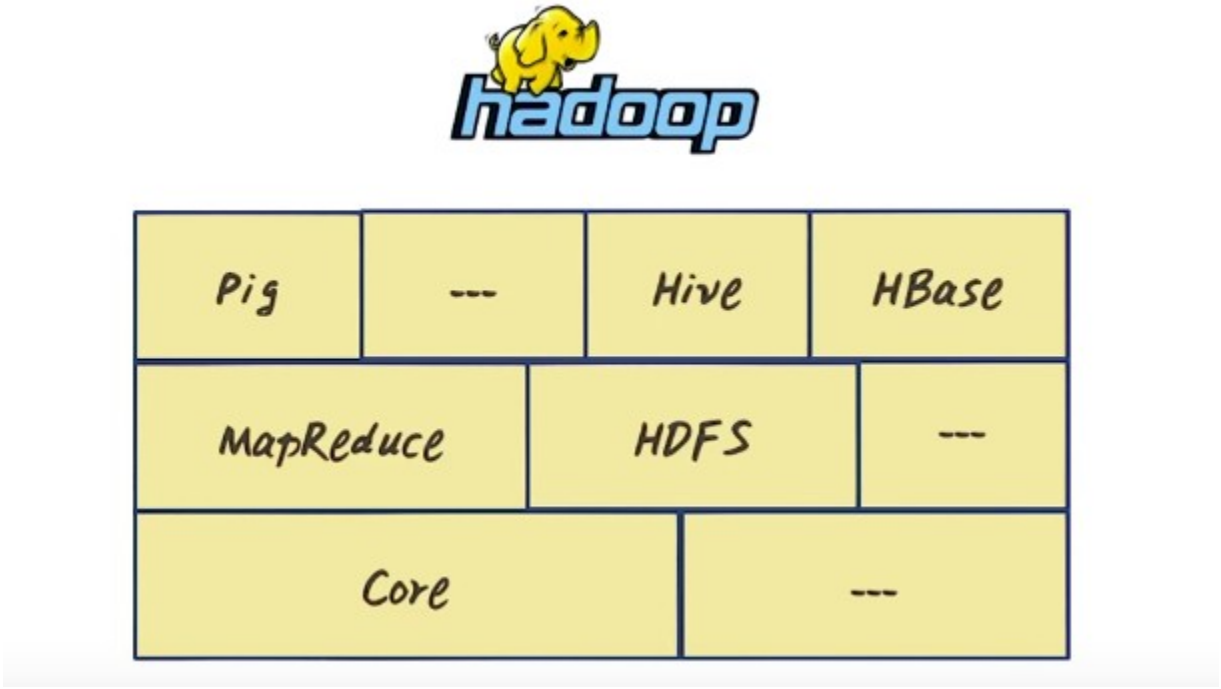
15. So far we're talking about healthcare application and machine learning algorithms. Next we'll talk about big data systems. In order to deal with big data set and implement your algorithm to process big data set, we need big data systems.

BIG DATA SYSTEMS

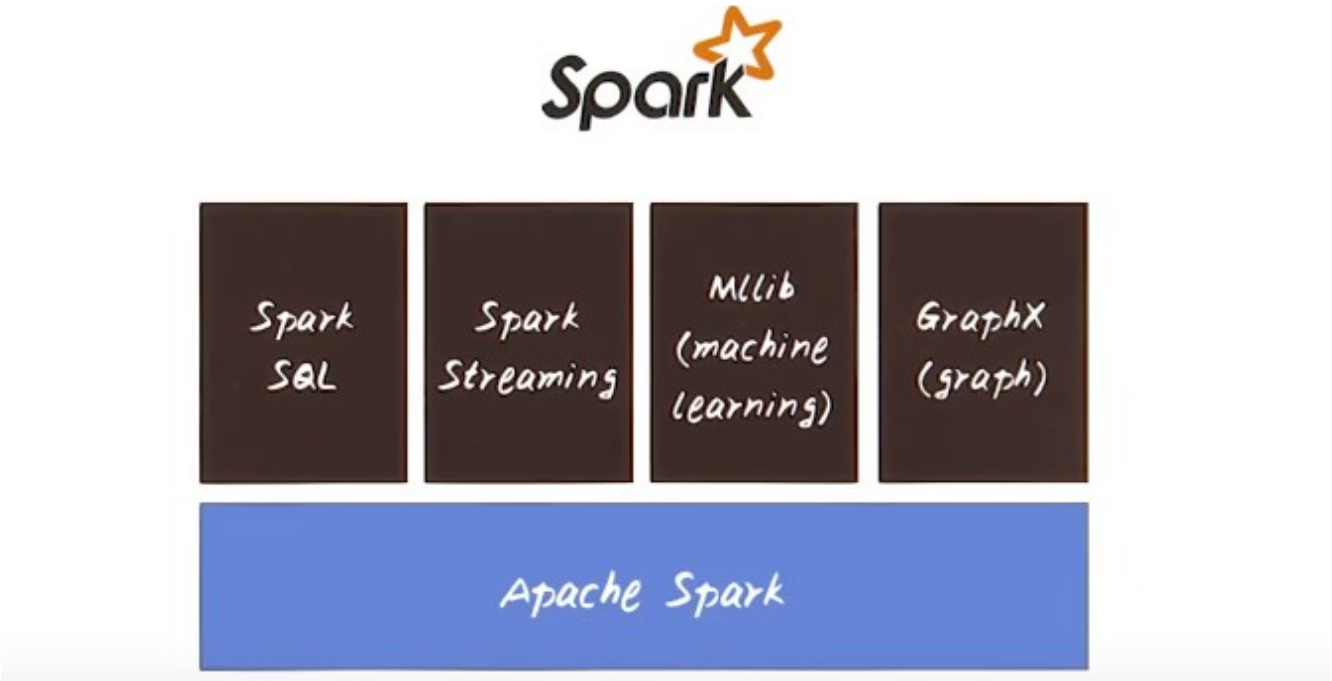
Distributed disk-based big data system

Distributed in-memory big data system

So in this course we'll introduce two popular big data systems. Hadoop (重音在 doo 上) and Spark. Hadoop is a distributed disk-based big data systems that all the data are stored in disks. Well, Spark is a distributed in-memory big data systems. That most data store in memory. So Spark in general is much faster than Hadoop, but both are popular big data system that people are using.



In this course, we'll talk about Hadoop and all the important building blocks in Hadoop. We'll talk about the core infrastructure of Hadoop, the MapReduce programming model and HDFS storage systems, and the high-level processing systems, such as Pig, Hive, and HBase.



So in this course, we will also talk about Spark, the core infrastructure of Spark, how do we store data and how do we process data. Using Spark and the high level abstractions such as Spark SQL and Spark streaming, MLLib for large scale machine learning library using Spark, and GraphX for processing graph data using Spark.

16. So today we talked about three big parts of this course. We talked about the healthcare applications, the machine learning algorithms, and the big data systems. We integrate all of this throughout the course. We'll move back and forth between application, algorithm, and systems. For example, we might build a scalable classifier using logistic regression on Hadoop for predicting heart failure. So let's get started.