Latent Variable Models
&
Expectation Maximization

Week 2

- What is a latent variable, why do we need it, and how to use it
- Common latent variable models (clustering and dimensionality reduction)
- How to train them with Expectation Maximization algorithm
- Extensions of Expectation Maximization such as handling missing data

Latent (hidden) variable is a variable that you never observe

Latent means 'hidden' in latin.

Phenomenon like height ,length, speed, etc. we can mesasure and observe directly.

But things like 'altruism' can't be quantified as such.
This variable is usually called 'latent'.

Lets demonstrate the pertinence of latent variables.

Suppose we want to hire some employees.

We have all their high school grades, some of them have uni grades, IQ scores. Also we conducted phone interviews.

	High school grade	University grade	IQ score	Phone Interview	Onsite interview
John	4.0	4.0	120	3/4	?
Helen	3.7	3.6	N/A	4/4	?
Jack	3.2	N/A	112	2/4	?
Emma	2.9	3.2	N/A	3/4	?

	High school grade	University grade	IQ score	Phone Interview	Onsite interview
Sophia	3.5	3.6	N/A	4/4	85/100
...					

Now, we can fly some of the people for an onsite. But flying EVERYONE is infeasible. We can't invite everyone - gotta pay for flight, hotel,etc.

So our idea is: let's predict their onsite interviews and get the ones that are the highest.

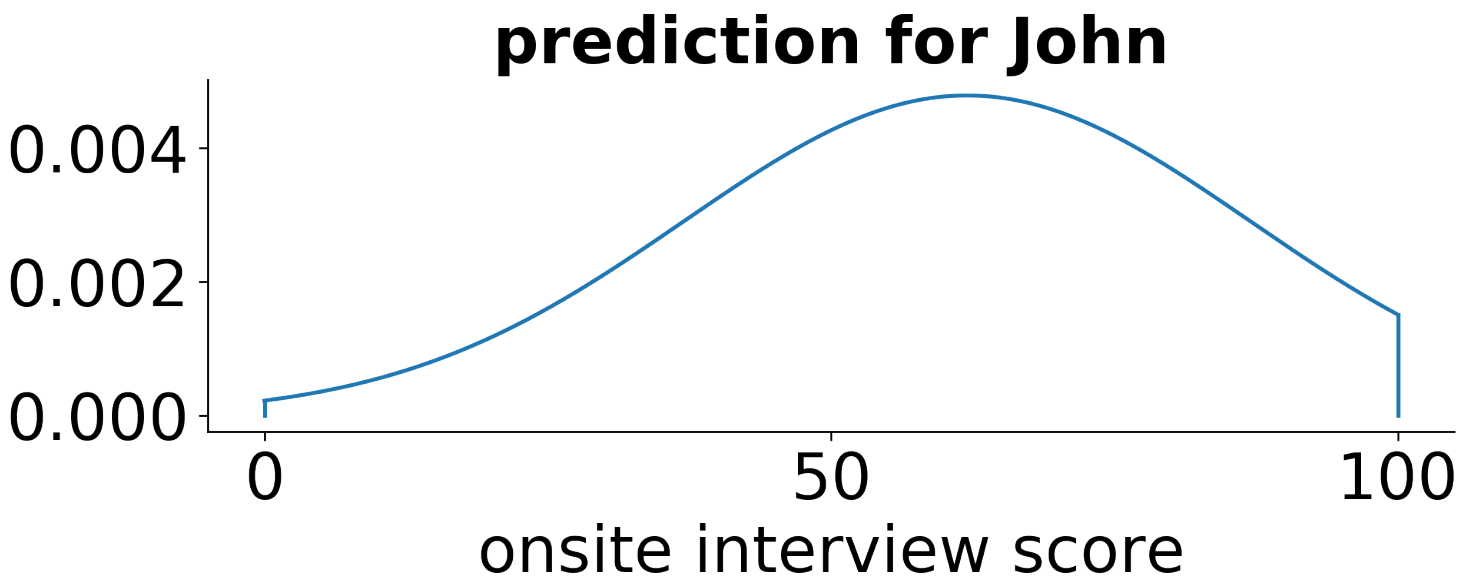
If we have historical data, we can use it to train a regression model. But two main problems arise:

1 . We have missing values and dont want this to negatively impact candidates- e.g. we don't know university grades for everyone, or not everyone could afford uni but nonetheless same intelligence

2. May want to quantify uncertainty in our predictions. Linear regression can't really do this.

This is the motivator to use some other method.

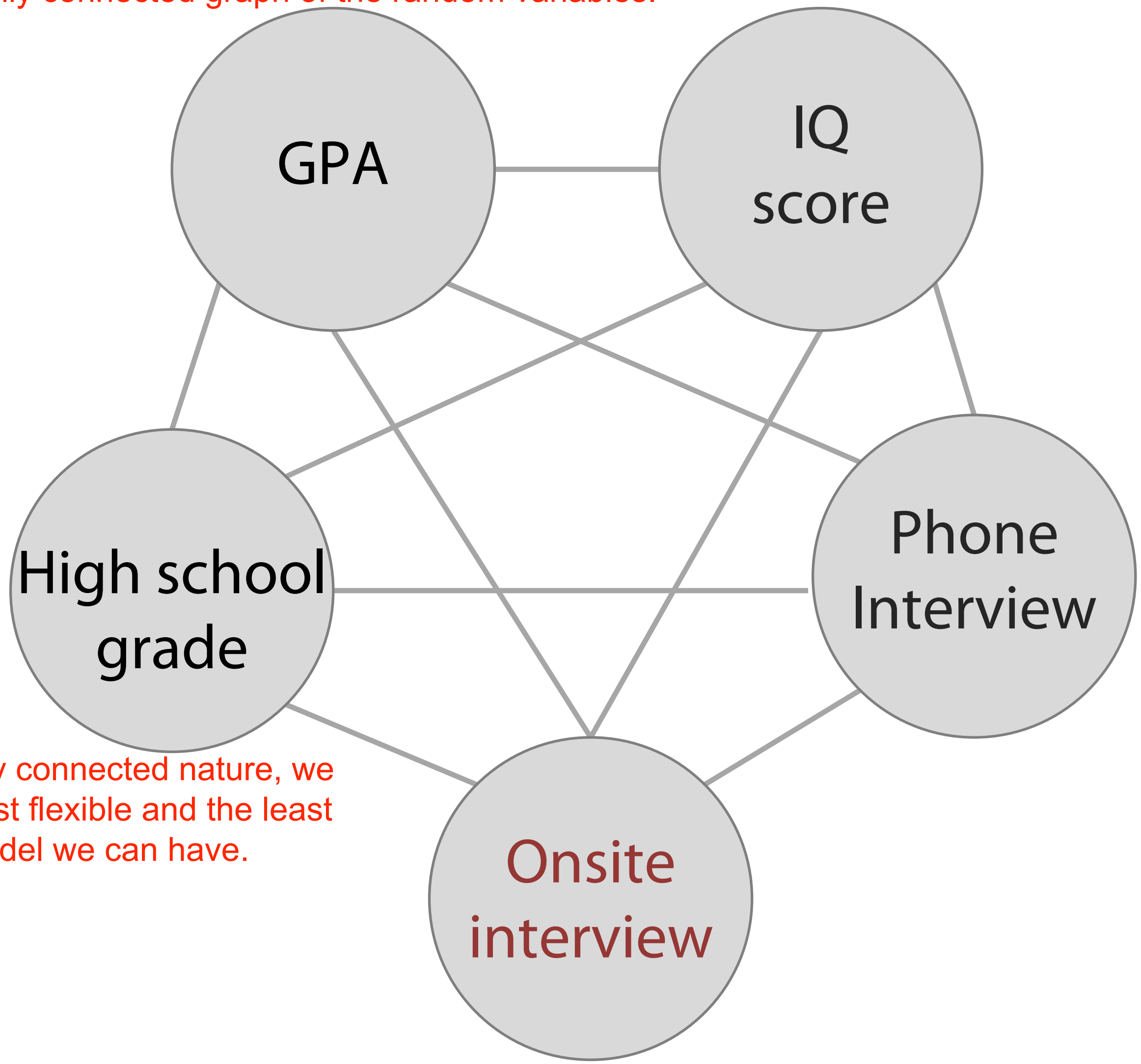
	High school grade	University grade	IQ score	Phone Interview	Onsite interview
John	4.0	4.0	120	3/4	?
Helen	3.7	3.6	N/A	4/4	?
Jack	3.2	N/A	112	2/4	?
Emma	2.9	3.2	N/A	3/4	?



^ This is what we want. Some probabilistic overview. We want to quantify the uncertainty in our prediction. If the predicted performance Is less than 50 and this is certain, then we don't bring him onsite. Otherwise, we bring him onsite. He may either be strong (Certainly), or we're unsure about it.

Probabilistic model

One way to build probabilistic model is to join random variables together.
Here, we have a fully connected graph of the random variables.



Because of this fully connected nature, we end up with the most flexible and the least structured model we can have.

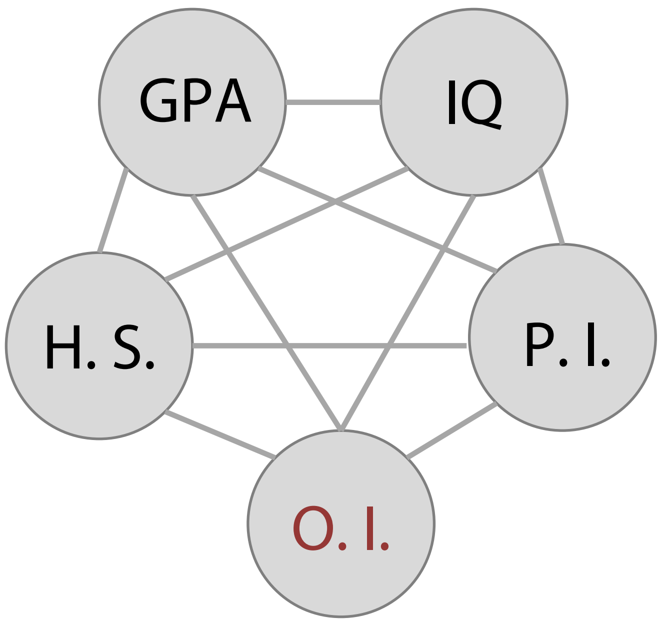
Probabilistic model

High school	GPA	IQ	Phone Interview	Onsite Interview	Probability
1.0	1.0	1	0/4	1/100	0.001
1.0	1.0	1	0/4	2/100	0.0023
…	…	…	…		
4.0	4.0	180	4/4	100	0.000001

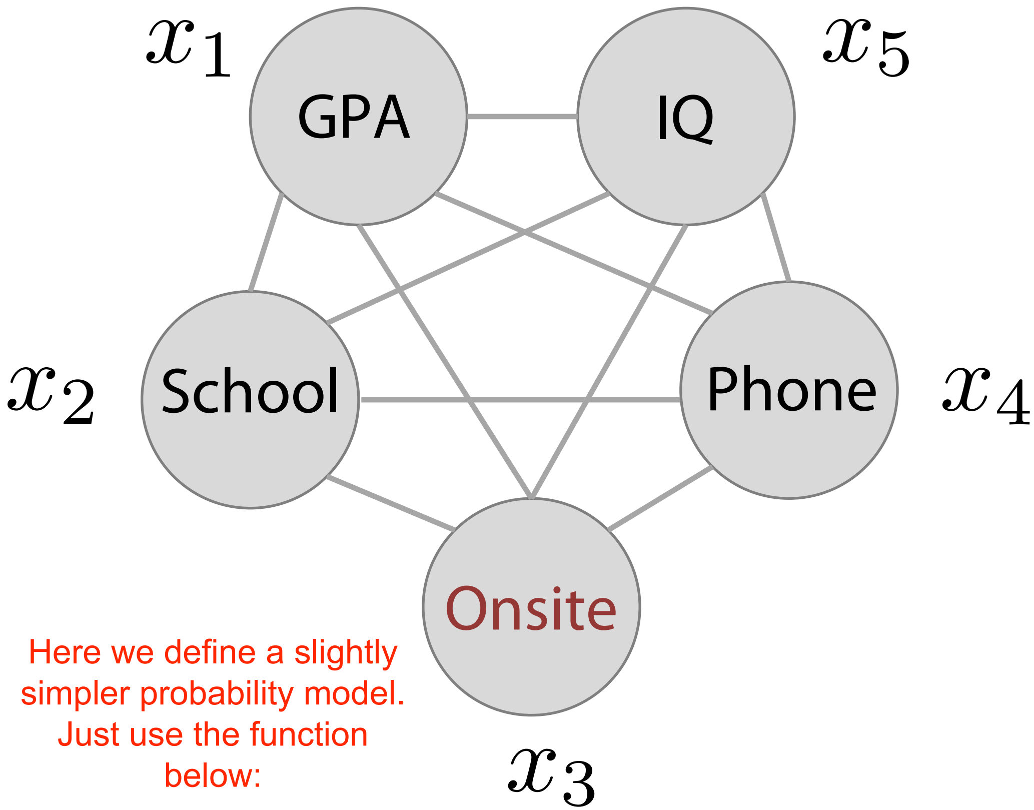
In this 'most flexible' method, there are exponentially many combinations that we have to account for.

For each of them , we have to assign
the probability.

As a result, it is impractical to create this model.



Probabilistic model



Here we define a slightly simpler probability model. Just use the function below:

$$p(x_1, x_2, x_3, x_4, x_5) = \frac{exp(-w^\intercal x)}{Z}$$

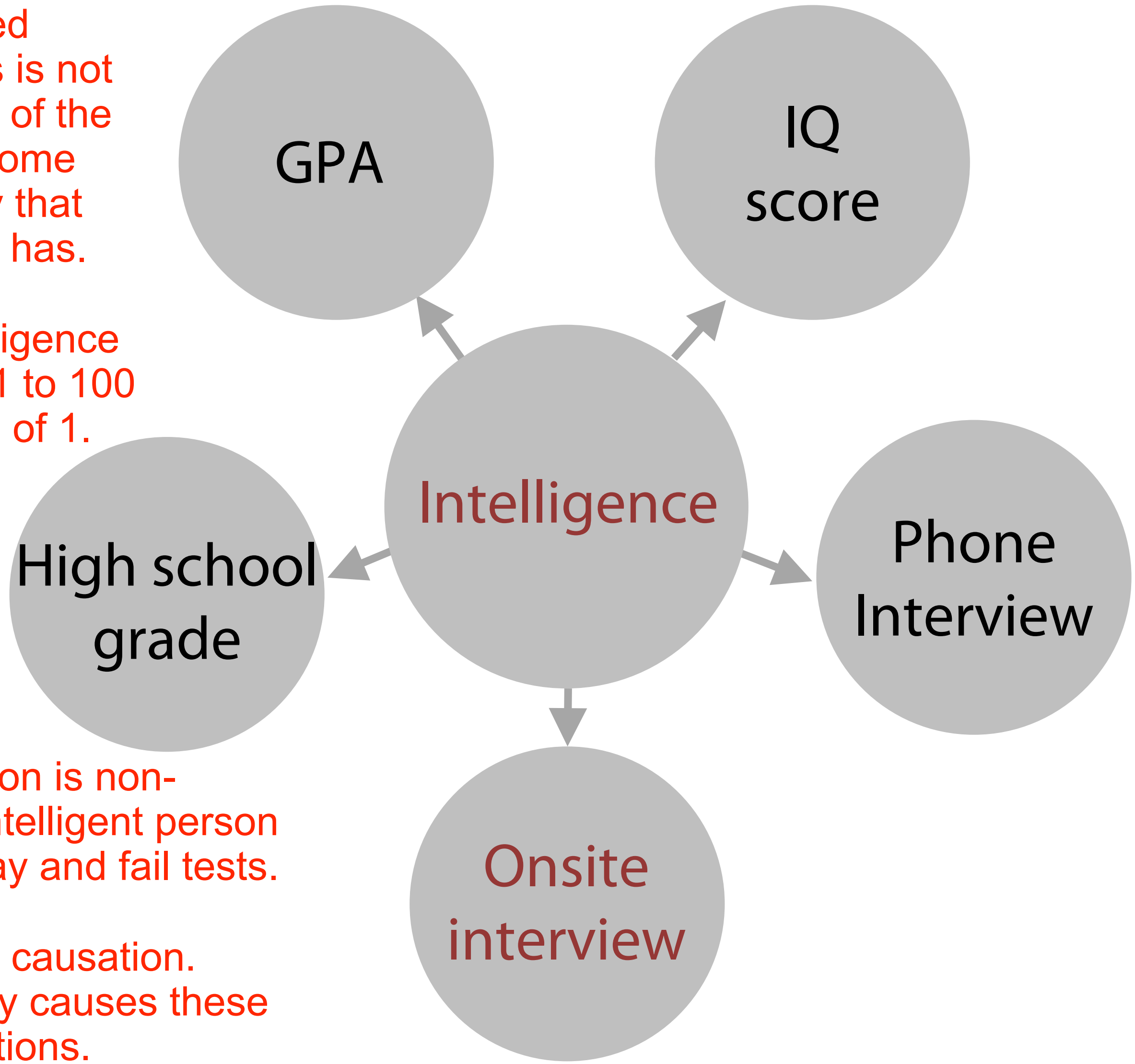
Z here is the normalization constant. Set z such that the sum of all probabilities is 1.

But the problem lies in the fact that we have to find Z by finding ALL possibilities. Same problem as last time. Exhaustive iteration.

Probabilistic model

Now, we insert some 'latent' / 'hidden' random variable called 'intelligence'. This is not stated in the data of the people, but is some 'innate' property that every candidate has.

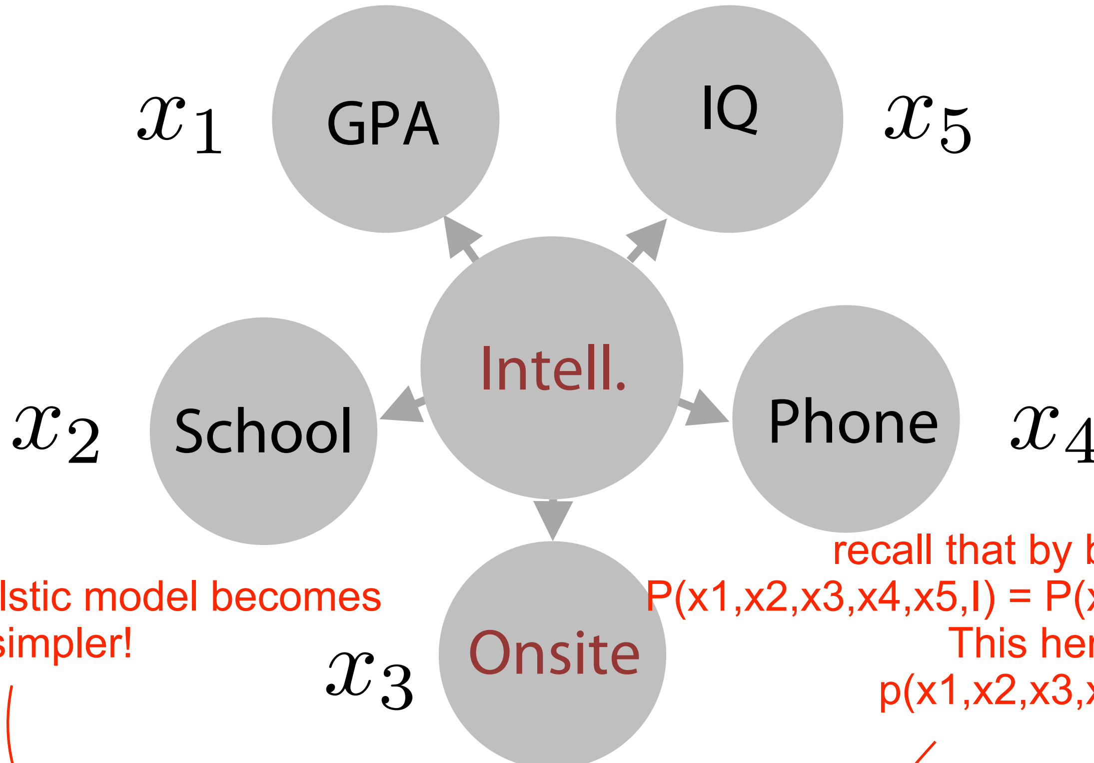
We suppose intelligence is measured from 1 to 100 in discrete steps of 1.



The connection is non-deterministic. An intelligent person may have a bad day and fail tests.

But, it is direct causation. Intelligence directly causes these observations.

Probabilistic model



So now, our probabilstic model becomes
much simpler!

recall that by bayes law,
P(x1,x2,x3,x4,x5,I) = P(x1,x2,x3,x4,x5|I) p(I)
This here is
p(x1,x2,x3,x4,x5,I).

$$p(x_1, x_2, x_3, x_4, x_5) = \sum_{I=1}^{100} p(x_1, x_2, x_3, x_4, x_5 \mid I)\, p(I) =$$

We loop from 1 to 100 for I
as those are the the values of I
(law of sum)

$$\sum_{I=1}^{100} p(x_1 \mid I) \ldots p(x_5 \mid I)\, p(I)$$

Latent variable models

Pros:

Yes

- Simpler models (less edges)
- Fewer parameters
- Latent variables are sometimes meaningful

autoencoders for
example!

Cons:

- Harder to work with