PROBABILISTIC INFERENCE, GENERATIVE MODELS AND HIDDEN VARIABLES

David Talbot, Yandex Translate

Autumn 2020

Yandex School of Data Analysis

PROBABILISTIC INFERENCE

- Mr. White has two children. What is the probability that both children are boys?

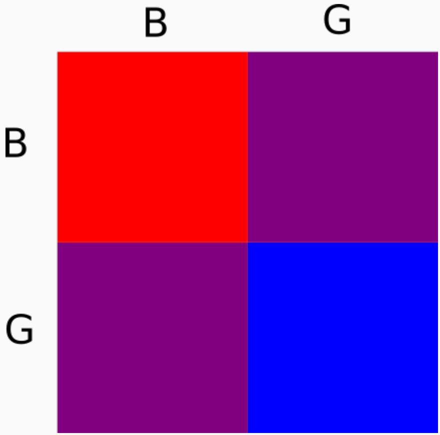
EQUAL OPPORTUNITIES

- Mr. White has two children. What is the probability that both children are boys?
- Mr. Jones has two children. The older child is a boy. What is the probability that both children are boys?

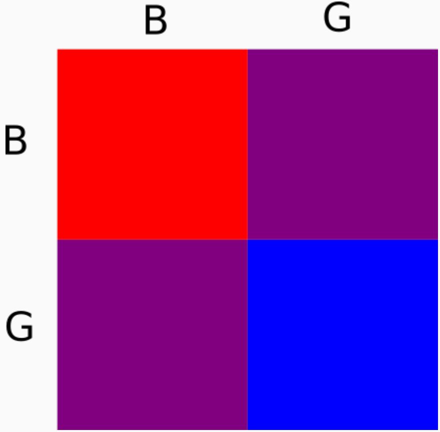
EQUAL OPPORTUNITIES

- Mr. White has two children. What is the probability that both children are boys?
- Mr. Jones has two children. The older child is a boy. What is the probability that both children are boys?
- Mr. Smith has two children. One of them is a boy. What is the probability that both children are boys?

PRIOR PROBABILITY



PRIOR PROBABILITY



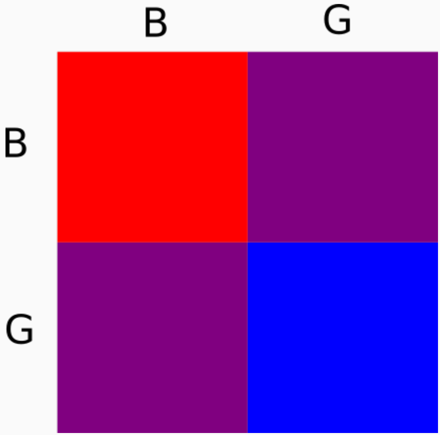
$\Pr(BB) = \frac{1}{4}$

CONDITION ON EVENT 'THE OLDER CHILD IS A BOY'

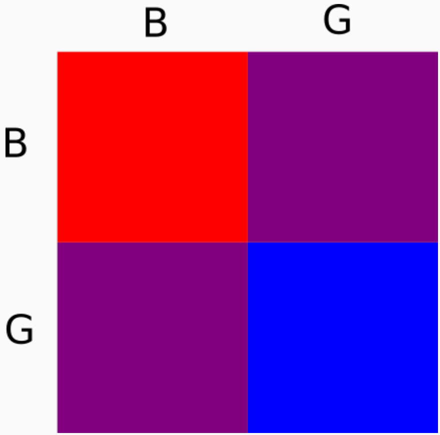
CONDITION ON EVENT 'THE OLDER CHILD IS A BOY'

$\Pr(BB|C_1 = B) = \frac{1}{2}$

PRIOR PROBABILITY

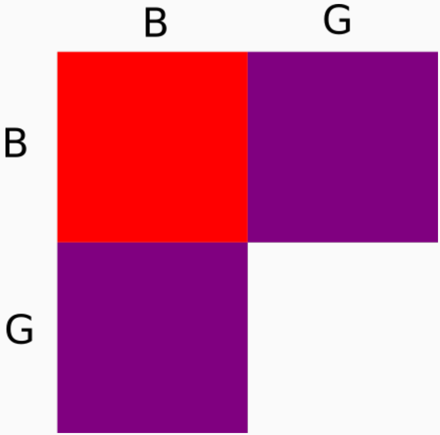


PRIOR PROBABILITY

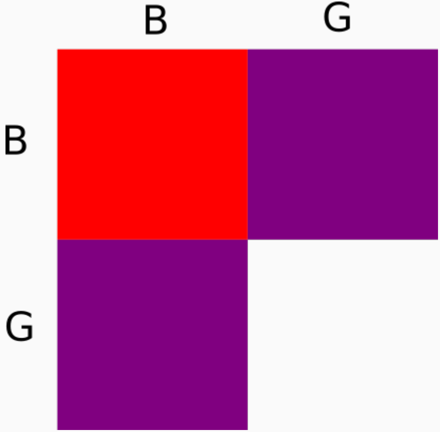


$\Pr(BB) = \frac{1}{4}$

CONDITIONED ON THE EVENT ‘ONE IS A BOY’



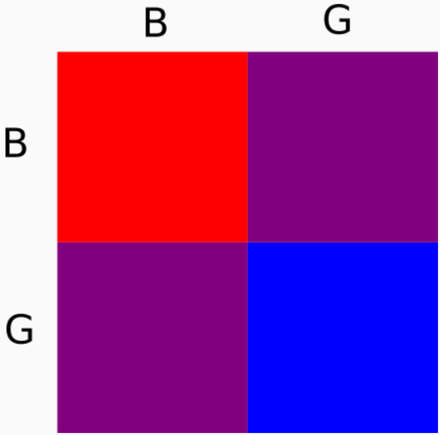
CONDITIONED ON THE EVENT ‘ONE IS A BOY’



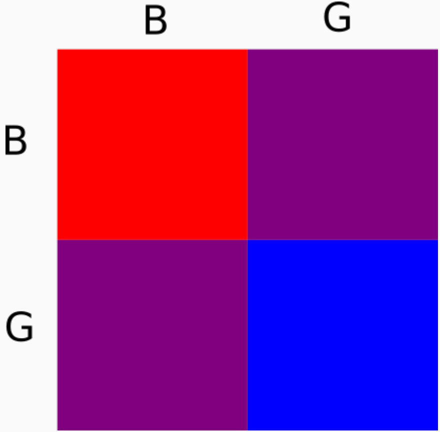
$\Pr(BB|C_1 = B \cup C_2 = B) = \frac{1}{3}$

Mr. Brown has two children. One of them is a boy born on a Tuesday. What is the probability that he has two boys?

PRIOR PROBABILITY

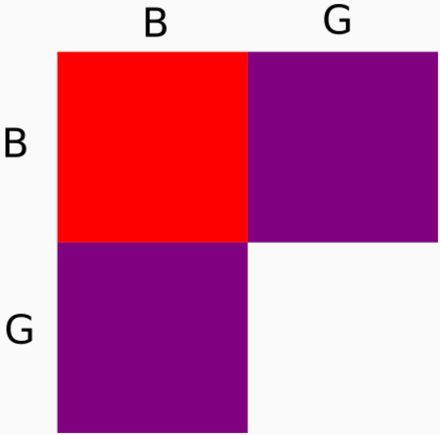


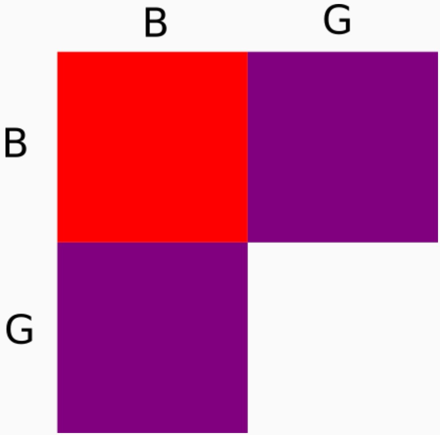
PRIOR PROBABILITY



$\Pr(BB) = \frac{1}{4}$

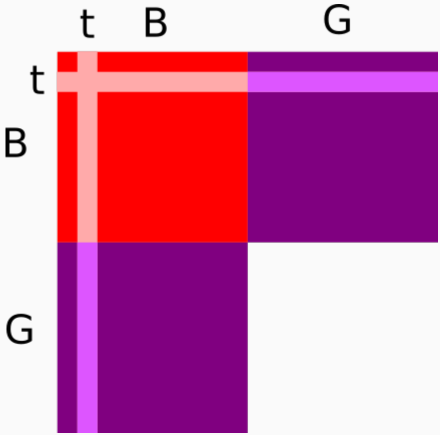
CONDITIONED ON ‘ONE IS A BOY’



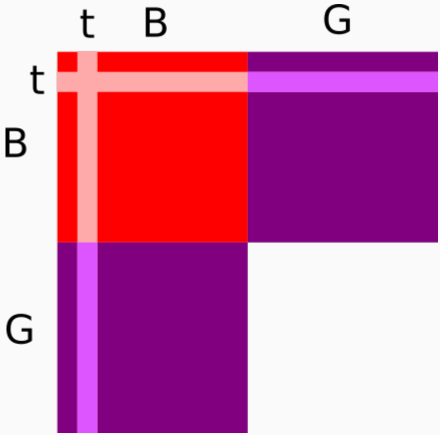


$\Pr(BB|C_1 = B \cup C_2 = B) = \frac{1}{3}$

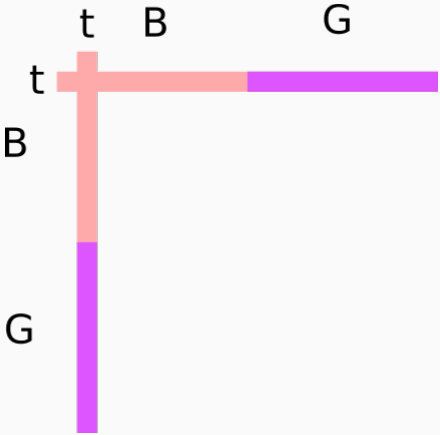
CONDITIONED ON 'ONE IS A BOY BORN ON TUESDAY'



CONDITIONED ON 'ONE IS A BOY BORN ON TUESDAY'

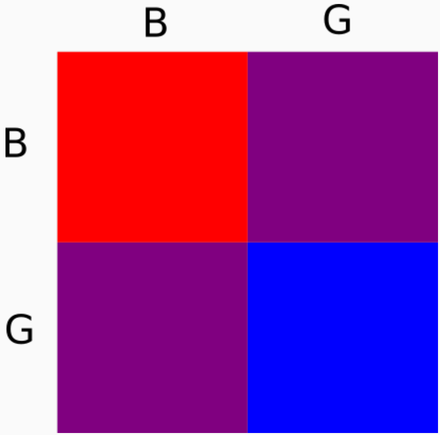


$$\Pr(BB|C_1 = B_{tue} \cup C_2 = B_{tue}) = \frac{13}{27} \approx \frac{1}{2}$$

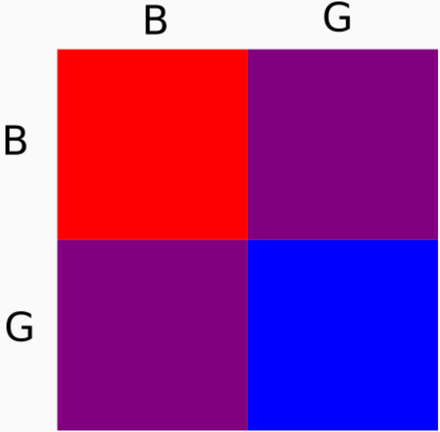


$$\Pr(BB|C_1 = B_{tue} \cup C_2 = B_{tue}) = \frac{13}{27} \approx \frac{1}{2}$$

PRIOR PROBABILITY



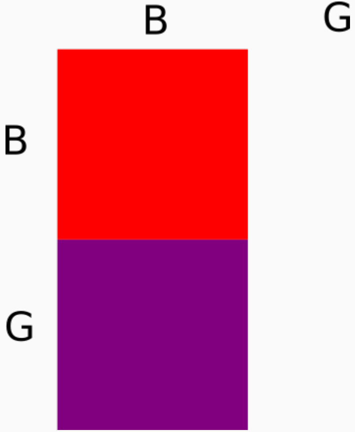
PRIOR PROBABILITY



$\Pr(BB) = \frac{1}{4}$

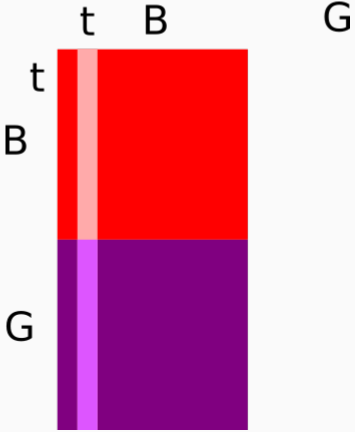
CONDITIONED ON ‘ONE RANDOMLY OBSERVED CHILD WAS A BOY’

CONDITIONED ON 'ONE RANDOMLY OBSERVED CHILD WAS A BOY'

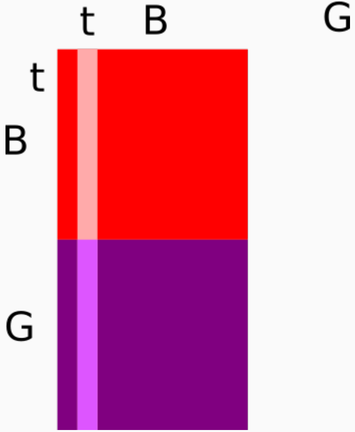


$$\Pr(BB|C_1 = B) = \frac{1}{2}$$

‘ONE RANDOMLY OBSERVED CHILD WAS A BOY BORN ON TUESDAY’



‘ONE RANDOMLY OBSERVED CHILD WAS A BOY BORN ON TUESDAY’



$\Pr(BB|C_1 = B_{tues}) = \frac{1}{2}$

‘ONE RANDOMLY OBSERVED CHILD WAS A BOY BORN ON TUESDAY’

$\Pr(BB|C_1 = B_{tues}) = \frac{1}{2}$

SOME LESSONS

- Assigning probabilities to data is easier if you clarify how the information was obtained.

- Assigning probabilities to data is easier if you clarify how the information was obtained.
- Describing a model of the data avoids such problems.

- Assigning probabilities to data is easier if you clarify how the information was obtained.
- Describing a model of the data avoids such problems.
- There are no paradoxes in probability, only badly posed questions :)

COINS

- You flip a coin 10 times and see 8 heads and 2 tails

COINS

- You flip a coin 10 times and see 8 heads and 2 tails
- What's the probability that on the next flip you see a head?

COINS

- You flip a coin 100 times and see 78 heads and 22 tails

COINS

- You flip a coin 100 times and see 78 heads and 22 tails
- What's the probability that on the next flip you see a head?

- You flip a coin 1,000 times and see 811 heads and 189 tails

- You flip a coin 1,000 times and see 811 heads and 189 tails
- What’s the probability that on the next flip you see a head?

- You're given a coin by someone you don't trust.

COINS

- You’re given a coin by someone you don’t trust.
- How would you determine the probability of heads?

COINS

- You’re given a coin by someone you don’t trust.
- How would you determine the probability of heads?
- You flip the coin n times. Describe the joint probability of the observed sequence.

- You’re given a coin by someone you don’t trust.
- How would you determine the probability of heads?
- You flip the coin n times. Describe the joint probability of the observed sequence.
- What assumptions are you making? Are they reasonable?

Two events E_1 and E_2 are *independent* if learning about one does not affect our belief about the other.

$$P(E_1|E_2) = P(E_1)$$

Two events E_1 and E_2 are *independent* if learning about one does not affect our belief about the other.

$$P(E_1|E_2) = P(E_1)$$

Two events are *conditionally independent* given a third event E_3, if they are indepdendent when E_3 is observed.

$$P(E_1, E_2|E_3) = P(E_1|E_3)P(E_2|E_3)$$

INDEPENDENCE AND CONDITIONAL INDEPENDENCE

Two events E_1 and E_2 are *independent* if learning about one does not affect our belief about the other.

$$P(E_1|E_2) = P(E_1)$$

Two events are *conditionally independent* given a third event E_3, if they are indepdendent when E_3 is observed.

$$P(E_1, E_2|E_3) = P(E_1|E_3)P(E_2|E_3)$$

In otherwords, E_3 explains E_1 and E_2 such that they no longer provide any additional information about each other.

What third event might make these pairs of events conditionally independent?

What third event might make these pairs of events conditionally independent?

- Flipping a single coin twice.

What third event might make these pairs of events conditionally independent?

- Flipping a single coin twice.
- Flipping two different coins once each.

What third event might make these pairs of events conditionally independent?

- Flipping a single coin twice.
- Flipping two different coins once each.
- The time at which two students arrive at class.

What third event might make these pairs of events conditionally independent?

- Flipping a single coin twice.
- Flipping two different coins once each.
- The time at which two students arrive at class.
- The amount of time you study and your results on the exam.

- You're given a bag with two coins in it C_1 and C_2. They look identical but ...

$$P(H|C_1) = 0.4$$

while

$$P(H|C_2) = 0.6.$$

- You're given a bag with two coins in it C_1 and C_2. They look identical but ...
$$P(H|C_1) = 0.4$$
while
$$P(H|C_2) = 0.6.$$

- You pick a coin at random from the bag. How would you determine which coin you had selected?

COINS

- You're given a bag with two coins in it C_1 and C_2. They look identical but ...

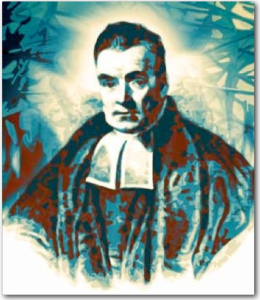
$$P(H|C_1) = 0.4$$

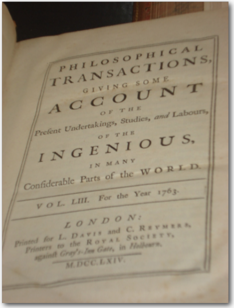
  while

$$P(H|C_2) = 0.6.$$

- You pick a coin at random from the bag. How would you determine which coin you had selected?
- How many times would you need to flip the coin?

REV. BAYES





How could this guy help you?

Given a prior $P(C_1)$ and an observation H_1 infer the posterior $P(C_1|H_1)$

Given a prior $P(C_1)$ and an observation H_1 infer the posterior $P(C_1|H_1)$

$$P(C_1|H_1) \propto P(C_1)P(H_1|C_1)$$

Given a prior $P(C_1)$ and an observation H_1 infer the posterior $P(C_1|H_1)$

$$P(C_1|H_1) \propto P(C_1)P(H_1|C_1)$$

Flip the coin again and update your belief taking the posterior $P(C_1|H_1)$ as your new prior.

$$P(C_1|H_1, H_2) \propto P(C_1|H_1)P(H_2|C_1)$$

Take a look at `bayesian_updating.ipynb`

GENERATIVE MODELS

Generative models model a joint distribution over inputs X and outputs Y

$$\Pr(X, Y|\theta).$$

Generative models model a joint distribution over inputs X and outputs Y

$$\Pr(X, Y|\theta).$$

Discriminative models model the conditional distribution over outputs Y given inputs.

$$\Pr(Y|X, \theta).$$

Generative models model a joint distribution over inputs X and outputs Y

$$\Pr(X, Y|\theta).$$

Discriminative models model the conditional distribution over outputs Y given inputs.

$$\Pr(Y|X, \theta).$$

Describing a generative model often involves stating the parametric form and independence assumptions it makes.

Describe a generative model of the experiment with 3 coins

Describe a generative model of the experiment with 3 coins
Let $X \in \{H, T\}$ be the observations and Y the coin's colour.

Describe a generative model of the experiment with 3 coins
Let $X \in \{H, T\}$ be the observations and Y the coin's colour.

$$P(X, Y) = P(Y) \prod_i P(X_i|Y) = \theta_Y \prod_i \theta_{X_i|Y}$$

Describe a generative model of the experiment with 3 coins
Let $X \in \{H, T\}$ be the observations and Y the coin's colour.

$$P(X, Y) = P(Y) \prod_i P(X_i|Y) = \theta_Y \prod_i \theta_{X_i|Y}$$

with parameters

$$\theta_{blue} := P(Y = blue) \qquad \text{etc.}$$

and

$$\theta_{head\_blue} := P(X_i = H|Y = blue) \qquad \text{etc.}$$

Describe a generative model of the experiment with 3 coins
Let $X \in \{H, T\}$ be the observations and Y the coin's colour.

$$P(X, Y) = P(Y) \prod_i P(X_i|Y) = \theta_Y \prod_i \theta_{X_i|Y}$$

with parameters

$$\theta_{blue} := P(Y = blue) \qquad \text{etc.}$$

and

$$\theta_{head\_blue} := P(X_i = H|Y = blue) \qquad \text{etc.}$$

Is this the *true* model for this process? How would you estimate these parameters from data?

MODELS IN NLP

- Unknown *underlying* generative process (unlike our coins)

- Unknown *underlying* generative process (unlike our coins)
- High dimensional output space

- Unknown *underlying* generative process (unlike our coins)
- High dimensional output space
- Sparse data

- Unknown *underlying* generative process (unlike our coins)
- High dimensional output space
- Sparse data
- Impossible to do inference exactly over all Y

EXAMPLES OF GENERATIVE MODELS IN NLP

- Naive Bayes text classifier

EXAMPLES OF GENERATIVE MODELS IN NLP

- Naive Bayes text classifier
- Hidden Markov models for tagging

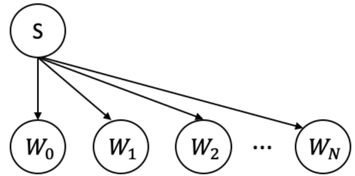
EXAMPLES OF GENERATIVE MODELS IN NLP

- Naive Bayes text classifier
- Hidden Markov models for tagging
- Noisy channel models for spelling correction

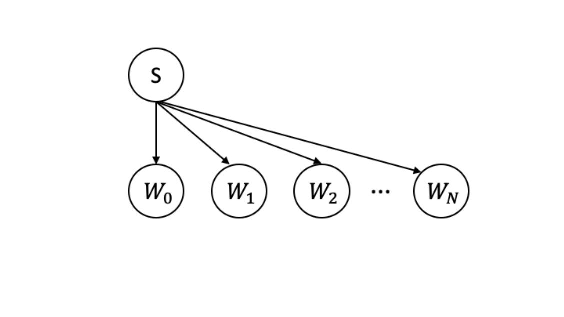
EXAMPLES OF GENERATIVE MODELS IN NLP

- Naive Bayes text classifier
- Hidden Markov models for tagging
- Noisy channel models for spelling correction
- Word alignment models in machine translation

$$P(W, S) = P(S) \prod_i P(W_i | S)$$

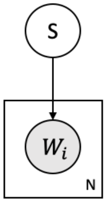


$$P(W, S) = P(S) \prod_i P(W_i | S)$$



How can this possibly work for spam classification?

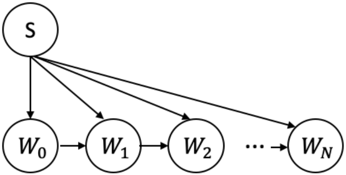
$$P(W, S) = P(S) \prod_i P(W_i|S)$$



How can this possibly work for spam classification?

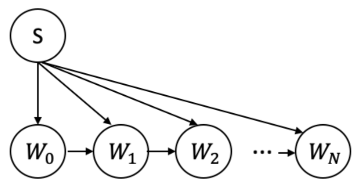
BIGRAM MODEL (SLIGHTLY LESS NAIVE BAYES)

$$P(W, S) = P(S)P(W_0|S)\prod_i P(W_i|W_{i-1}, S)$$



BIGRAM MODEL (SLIGHTLY LESS NAIVE BAYES)

$$P(W, S) = P(S)P(W_0|S) \prod_i P(W_i|W_{i-1}, S)$$



Why might this be even worse for spam classification?

Generative models often make *independence assumptions*:

Generative models often make *independence assumptions*:

- Naive Bayes text classifier

Generative models often make *independence assumptions*:

- Naive Bayes text classifier
  - conditional probability of each word given topic

Generative models often make *independence assumptions*:

- Naive Bayes text classifier
  - conditional probability of each word given topic
- *N*-gram language model

MAKING INDEPENDENCE ASSUMPTIONS

Generative models often make *independence assumptions*:

- Naive Bayes text classifier
  - conditional probability of each word given topic
- N-gram language model
  - conditional probability of each word given $n-1$ words

Generative models often make *independence assumptions*:

- Naive Bayes text classifier
    - conditional probability of each word given topic
- *N*-gram language model
    - conditional probability of each word given $n - 1$ words
- Variational autoencoder

Generative models often make *independence assumptions*:

- Naive Bayes text classifier
  - conditional probability of each word given topic
- *N*-gram language model
  - conditional probability of each word given $n - 1$ words
- Variational autoencoder
  - latent variables are independent a priori

Generative models often make *independence assumptions*:

- Naive Bayes text classifier
  - conditional probability of each word given topic
- *N*-gram language model
  - conditional probability of each word given $n - 1$ words
- Variational autoencoder
  - latent variables are independent a priori
- Sequence-to-sequence NMT model

Generative models often make *independence assumptions*:

- Naive Bayes text classifier
  - conditional probability of each word given topic
- N-gram language model
  - conditional probability of each word given $n - 1$ words
- Variational autoencoder
  - latent variables are independent a priori
- Sequence-to-sequence NMT model
  - probability of target sentence given source sentence

INDEPENDENCE ASSUMPTIONS

Why do we make these independence assumptions?

Why do we make these independence assumptions?

- They can reduce the parameter space

Why do we make these independence assumptions?

- They can reduce the parameter space
- They can make inference easier

INDEPENDENCE ASSUMPTIONS

Why do we make these independence assumptions?

- They can reduce the parameter space
- They can make inference easier

Sometimes the independence assumptions we make depends on the amount of data. Why?

Why do we make these independence assumptions?

- They can reduce the parameter space
- They can make inference easier

Sometimes the independence assumptions we make depends on the amount of data. Why?

Can you see any connection between *attention* and *independence assumptions*?

Many sequence problems can be modelled generatively as a *noisy channel* problem.

Many sequence problems can be modelled generatively as a *noisy channel* problem.

- Imagine observations $X = [X_0, X_1, \dots]$ are a corrupted version of $Y = [Y_0, Y_1, \dots]$

Many sequence problems can be modelled generatively as a *noisy channel* problem.

- Imagine observations $X = [X_0, X_1, \ldots]$ are a corrupted version of $Y = [Y_0, Y_1, \ldots]$
- Build a model of $P(Y)$ directly

Many sequence problems can be modelled generatively as a *noisy channel* problem.

- Imagine observations $X = [X_0, X_1, \dots]$ are a corrupted version of $Y = [Y_0, Y_1, \dots]$
- Build a model of $P(Y)$ directly
- Build a model $P(X|Y)$ that tries to explain the observations

Many sequence problems can be modelled generatively as a *noisy channel* problem.

- Imagine observations $X = [X_0, X_1, \dots]$ are a corrupted version of $Y = [Y_0, Y_1, \dots]$
- Build a model of $P(Y)$ directly
- Build a model $P(X|Y)$ that tries to explain the observations
- Search for $\hat{Y} = \mathrm{argmax}_Y P(Y)P(X|Y)$

Many sequence problems can be modelled generatively as a *noisy channel* problem.

- Imagine observations $X = [X_0, X_1, \dots]$ are a corrupted version of $Y = [Y_0, Y_1, \dots]$
- Build a model of $P(Y)$ directly
- Build a model $P(X|Y)$ that tries to explain the observations
- Search for $\hat{Y} = \text{argmax}_Y P(Y)P(X|Y)$

Speech recognition, spelling correction, machine translation, swipe etc.

NOISY CHANNEL EXAMPLE: SPELLING CORRECTION

TASK: Given some observed text X, generate corrected text Y

TASK: Given some observed text X, generate corrected text Y

$$X = \text{“htis is hw peopl right on the intrenrt”}$$

$$\hat{Y} = \text{“this is how people write on the internet”}$$

TASK: Given some observed text X, generate corrected text Y

$$X = \text{``htis is hw peopl right on the intrenrt''}$$

$$\hat{Y} = \text{``this is how people write on the internet''}$$

Score hypotheses using prior $P(Y)$ and inverse model $P(X|Y)$

$$P(Y|X) \propto P(Y = \text{this is how})P(X = \text{htis is hw}|Y = \text{this is how})$$

TASK: Given some observed text X, generate corrected text Y

$$P(Y|X) \propto P(Y = \text{this is how})P(X = \text{htis is hw}|Y = \text{this is how})$$

TASK: Given some observed text X, generate corrected text Y

$$P(Y|X) \propto P(Y = \text{this is how})P(X = \text{htis is hw}|Y = \text{this is how})$$

What independence assumptions would be helpful in $P(X|Y)$?

NOISY CHANNEL EXAMPLE: SPELLING CORRECTION

TASK: Given some observed text X, generate corrected text Y

$$P(Y|X) \propto P(Y = \text{this is how})P(X = \text{htis is hw}|Y = \text{this is how})$$

What independence assumptions would be helpful in $P(X|Y)$?

$$P(\text{htis is hw}|\text{this is how}) = P(\text{htis}|\text{this})P(\text{is}|\text{is})P(\text{hw}|\text{how})$$

TASK: Given some observed text X, generate corrected text Y

$$P(Y|X) \propto P(Y = \text{this is how})P(X = \text{htis is hw}|Y = \text{this is how})$$

What independence assumptions would be helpful in $P(X|Y)$?

$$P(\text{htis is hw}|\text{this is how}) = P(\text{htis}|\text{this})P(\text{is}|\text{is})P(\text{hw}|\text{how})$$

What could we train $P(Y)$ on? How about $P(X|Y)$?

How can it be easier to model $P(Y)P(X|Y)$ rather than $P(Y|X)$?

How can it be easier to model $P(Y)P(X|Y)$ rather than $P(Y|X)$?

- Abundance of data for Y (unlabeled examples)

How can it be easier to model $P(Y)P(X|Y)$ rather than $P(Y|X)$?

- Abundance of data for Y (unlabeled examples)
- $P(X|Y) \approx \prod_i P(X_i|Y_i)$ less harmful than $P(Y|X) \approx \prod_i P(Y_i|X_i)$

How can it be easier to model $P(Y)P(X|Y)$ rather than $P(Y|X)$?

- Abundance of data for Y (unlabeled examples)
- $P(X|Y) \approx \prod_i P(X_i|Y_i)$ less harmful than $P(Y|X) \approx \prod_i P(Y_i|X_i)$

Because

How can it be easier to model $P(Y)P(X|Y)$ rather than $P(Y|X)$?

- Abundance of data for Y (unlabeled examples)
- $P(X|Y) \approx \prod_i P(X_i|Y_i)$ less harmful than $P(Y|X) \approx \prod_i P(Y_i|X_i)$

Because

- $P(X|Y)$ is only evaluated on *reasonable* values of X

How can it be easier to model $P(Y)P(X|Y)$ rather than $P(Y|X)$?

- Abundance of data for Y (unlabeled examples)
- $P(X|Y) \approx \prod_i P(X_i|Y_i)$ less harmful than $P(Y|X) \approx \prod_i P(Y_i|X_i)$

Because

- $P(X|Y)$ is only evaluated on *reasonable* values of X
- When Y's are not observed, X's are no longer independent

How can it be easier to model $P(Y)P(X|Y)$ rather than $P(Y|X)$?

- Abundance of data for Y (unlabeled examples)
- $P(X|Y) \approx \prod_i P(X_i|Y_i)$ less harmful than $P(Y|X) \approx \prod_i P(Y_i|X_i)$

Because

- $P(X|Y)$ is only evaluated on *reasonable* values of X
- When Y's are not observed, X's are no longer independent
- Model of $P(Y)$ can compensate for these assumptions

HIDDEN VARIABLE MODELS

Why introduce hidden variables?

Why introduce hidden variables?

- Hidden variables can help simplify density estimation

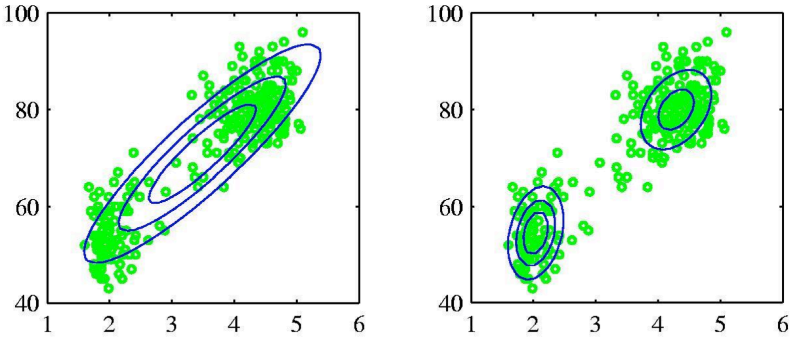
Why introduce hidden variables?

- Hidden variables can help simplify density estimation
- Hidden variables can allow more reasonable independence assumptions

Why introduce hidden variables?

- Hidden variables can help simplify density estimation
- Hidden variables can allow more reasonable independence assumptions
- Hidden variables can help cluster the data (e.g. share statistical strength)

GAUSSIAN MIXTURE MODELS



Single Gaussian

Mixture of two Gaussians

Generative model

Generative model

- Choose a cluster $i \in \{1, 2, \ldots, K\}$ from prior $\Pr(Y = i) = \lambda_i$

Generative model

- Choose a cluster $i \in \{1, 2, \ldots, K\}$ from prior $\Pr(Y = i) = \lambda_i$
- Generate an observation X from a Gaussian g_i with parameters μ_i, σ_i

Generative model

- Choose a cluster $i \in \{1, 2, \ldots, K\}$ from prior $\Pr(Y = i) = \lambda_i$
- Generate an observation X from a Gaussian g_i with parameters μ_i, σ_i

$$\Pr(X = x|\theta) = \sum_{i \in \{1,2,\ldots,K\}} \Pr(Y = i)\Pr(X = x|Y = i) = \sum_{i \in \{1,2,\ldots,K\}} \lambda_i g_i(x)$$

Generative model

- Choose a cluster $i \in \{1, 2, \ldots, K\}$ from prior $\Pr(Y = i) = \lambda_i$
- Generate an observation X from a Gaussian g_i with parameters μ_i, σ_i

$$\Pr(X = x|\theta) = \sum_{i \in \{1,2,\ldots,K\}} \Pr(Y = i)\Pr(X = x|Y = i) = \sum_{i \in \{1,2,\ldots,K\}} \lambda_i g_i(x)$$

How does a mixture model improve on a single Gaussian model?

Discrete generative models for grouped data

- Bag of words model - each word is independent

$$p(document) = \prod_{W \in D} P(W)$$

Discrete generative models for grouped data

- Bag of words model - each word is independent

$$p(document) = \prod_{W \in D} P(W)$$

- All documents generated from a single distribution $P(W)$

Discrete generative models for grouped data

Discrete generative models for grouped data

- Topic mixture model - words are conditionally independent given document topic

$$p(document) = \sum_{Z'} p(Z') \prod_{W \in D} p(W|Z')$$

Discrete generative models for grouped data

- Topic mixture model - words are conditionally independent given document topic

$$p(document) = \sum_{Z'} p(Z') \prod_{W \in D} p(W|Z')$$

- The assumption $p(W|D, Z) = p(W|Z)$ forces topics to explain word cooccurences

Discrete generative models for grouped data

Discrete generative models for grouped data

- Latent dirichlet allocation model

$$p(document) = \int_\theta P(\theta) \prod_i \sum_{Z_i} P(Z_i) P(W_i | Z_n)$$

Discrete generative models for grouped data

- Latent dirichlet allocation model

$$p(document) = \int_{\theta} P(\theta) \prod_{i} \sum_{Z_i} P(Z_i) P(W_i | Z_n)$$

- Each word W_i is conditionally independent of all others given its associated topic variable Z_i

Discrete generative models for grouped data

- Latent dirichlet allocation model

$$p(document) = \int_\theta P(\theta) \prod_i \sum_{Z_i} P(Z_i) P(W_i | Z_n)$$

- Each word W_i is conditionally independent of all others given its associated topic variable Z_i
- Each document is conditionally independent given its hidden distribution over topics θ

TOPIC MIXTURE MODEL

- Sample a topic $Z \in \{1, 2, \ldots, K\}$ for a document

- Sample a topic $Z \in \{1, 2, \ldots, K\}$ for a document
- Generate words independently given the topic

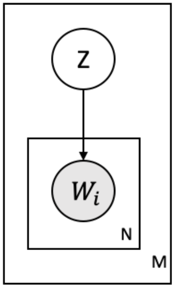
$$\Pr(W_1, W_2, \ldots, W_N | Z) = \prod_{i=1}^{N} \Pr(W_i | Z)$$

- Sample a topic $Z \in \{1, 2, \ldots, K\}$ for a document
- Generate words independently given the topic

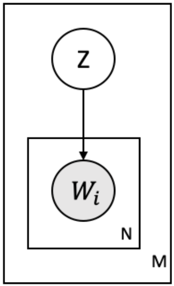
$$\Pr(W_1, W_2, \ldots, W_N | Z) = \prod_{i=1}^{N} \Pr(W_i | Z)$$

Each document has a single (hidden) topic

Model each *document* as having a single hidden topic Z



Each topic Z indexes a distribution over words.



LATENT DIRICHLET ALLOCATION

- Sample a distribution over topics for a document

$$\theta = (\theta_1, \theta_2, \ldots, \theta_K) \sim \text{Dirichlet}(\alpha)$$

- Sample a distribution over topics for a document

$$\theta = (\theta_1, \theta_2, \ldots, \theta_K) \sim \text{Dirichlet}(\alpha)$$

For each word in the document:

- Sample a distribution over topics for a document

$$\theta = (\theta_1, \theta_2, \ldots, \theta_K) \sim \text{Dirichlet}(\alpha)$$

For each word in the document:

- Sample a topic Z for each word from θ

$$Z_i = \Pr(Z_i = z) = \theta_z$$

- Sample a distribution over topics for a document

$$\theta = (\theta_1, \theta_2, \ldots, \theta_K) \sim \text{Dirichlet}(\alpha)$$

For each word in the document:

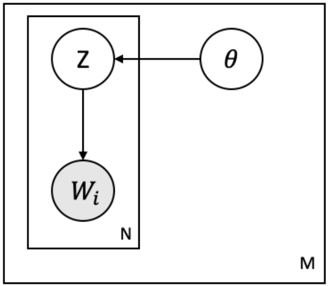
- Sample a topic Z for each word from θ

$$Z_i = \Pr(Z_i = z) = \theta_z$$

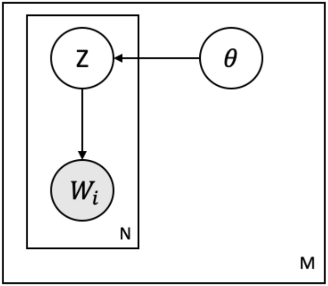
- Generate a word W according to the unigram distribution indexed by Z

$$W_i = \Pr(W_i = w|Z = z) = \beta_{z,w}$$

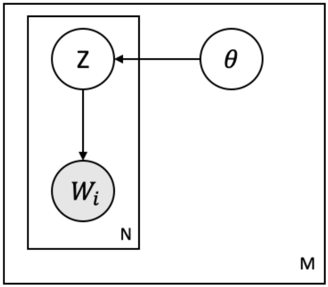
Model each *word* as having a single hidden topic.



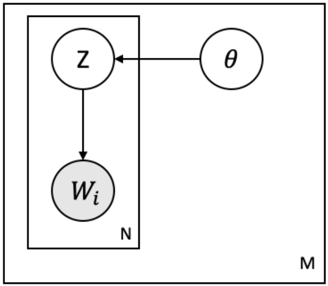
Each document has a hidden *distribution* over topics.



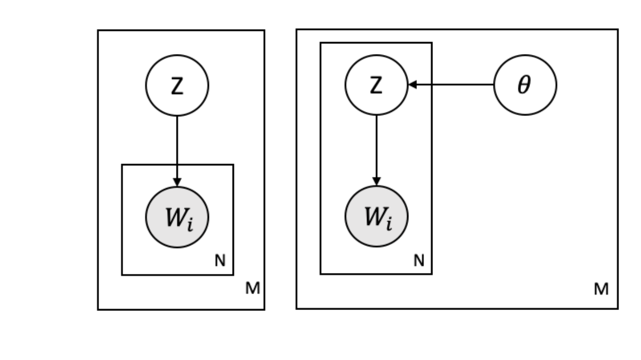
Topics can be shared across all documents



Documents are generated from multiple topics



MIXTURE VS. LDA MODEL



- Bigram language model (no hidden variables)

$$\Pr(w_t|w_{t-1},\ldots,w_0) \approx \Pr(w_t|w_{t-1})$$

- Class-based language model

$$\Pr(w_t|w_{t-1},\ldots,w_0) \approx \Pr(w_t|C(w_{t-1}))$$

Similar to a topic mixture model but trained on bigram data.

PARAMETER ESTIMATION

MAXIMUM LIKEHOOD PRINCIPLE

Choose parameters θ etc. s.t. *likelihood* of the data X, Y is maximized, i.e.

$$\theta^* = \operatorname*{argmax}_\theta \Pr(X, Y|\theta).$$

Choose parameters θ etc. s.t. *likelihood* of the data X, Y is maximized, i.e.

$$\theta^* = \underset{\theta}{\operatorname{argmax}} \Pr(X, Y|\theta).$$

Often easier to work with logarithm, e.g.

$$\log \Pr(Y)P(X|Y) = \log P(Y) + \log \Pr(X|Y).$$

So we can find the maximum of each parameter separately.

Choose parameters θ etc. s.t. *likelihood* of the data X, Y is maximized, i.e.

$$\theta^* = \operatorname*{argmax}_{\theta} \Pr(X, Y|\theta).$$

Often easier to work with logarithm, e.g.

$$\log \Pr(Y)P(X|Y) = \log P(Y) + \log \Pr(X|Y).$$

So we can find the maximum of each parameter separately.

How could you justify this method for choosing parameters?

We observed a sample D drawn from $(x, y) \in (X, Y)$ where $X \in \{H, T\}$, $Y = \{Red, Blue\}$. Each observation was labeled so

We observed a sample D drawn from $(x, y) \in (X, Y)$ where $X \in \{H, T\}$, $Y = \{Red, Blue\}$. Each observation was labeled so

$$\begin{aligned}\hat{\theta}_{mle} &= \underset{\theta}{\operatorname{argmax}} \sum_{(x,y)\in D} \log \Pr(X = x, Y = y|\theta) \\ &= \underset{\theta}{\operatorname{argmax}} \sum_{(x,y)\in(X,Y)} \#(X = x, Y = y) \log \Pr(X = x, Y = y|\theta)\end{aligned}$$

where we summarized the data using the *sufficient statistics*.

MAXIMIMUM LIKELIHOOD ESTIMATES FOR OUR MODEL

$$\hat{\theta}_{blue} = \frac{\#(B)}{\#(B) + \#(R)}$$

$$\hat{\theta}_{head\_blue} = \frac{\#(H, B)}{\#(B)}$$

If $T(X)$ are *sufficient statistics* for the sample X with respect to a model with parameters θ then

$$\Pr(\theta|T(X)) = \Pr(\theta|X).$$

If $T(X)$ are *sufficient statistics* for the sample X with respect to a model with parameters θ then

$$\Pr(\theta|T(X)) = \Pr(\theta|X).$$

Sufficient statistics summarize all the information about a sample that can influence our estimate of the parameters.

How would you approach this problem?

How would you approach this problem?

- You are given a bag with red and blue coins in it

How would you approach this problem?

- You are given a bag with red and blue coins in it
- But the colours have washed off

How would you approach this problem?

- You are given a bag with red and blue coins in it
- But the colours have washed off
- You don't know the proportions of red to blue

How would you approach this problem?

- You are given a bag with red and blue coins in it
- But the colours have washed off
- You don't know the proportions of red to blue
- Estimate the proportions of red and blue coins and the probability of heads for each coin

- Two unknowns: hidden variables and parameters (Z, θ)

- Two unknowns: hidden variables and parameters (Z, θ)
- If we knew Z, we could use MLE to estimate θ

MLE FOR MODELS WITH HIDDEN VARIABLES

- Two unknowns: hidden variables and parameters (Z, θ)
- If we knew Z, we could use MLE to estimate θ
- If we knew θ, we could use Bayes' rule to infer Z

- Initialize the parameters θ_0 somehow (randomly?)

- Initialize the parameters θ_0 somehow (randomly?)
- E-step: Compute $\Pr(Z|X, \theta_i)$ i.e. our best guess of the hidden data Z given our current parameters.

- Initialize the parameters θ_0 somehow (randomly?)
- E-step: Compute $\Pr(Z|X, \theta_i)$ i.e. our best guess of the hidden data Z given our current parameters.
- M-step: Update the parameters θ_{i+1} to maximize the expected log-likelihood.

- Initialize the parameters θ_0 somehow (randomly?)
- E-step: Compute $\Pr(Z|X, \theta_i)$ i.e. our best guess of the hidden data Z given our current parameters.
- M-step: Update the parameters θ_{i+1} to maximize the expected log-likelihood.
- Iterate until the expected log-likelihood stops increasing.

- Initialize the parameters θ_0 somehow (randomly?)
- E-step: Compute $\Pr(Z|X, \theta_i)$ i.e. our best guess of the hidden data Z given our current parameters.
- M-step: Update the parameters θ_{i+1} to maximize the expected log-likelihood.
- Iterate until the expected log-likelihood stops increasing.

Intuition: if we knew θ we could just infer Z, likewise if we knew Z we could just estimate θ. Since we don't know either, just guess and iteratively improve.

Let’s reformulate the expression for *mle* estimation.

Let's reformulate the expression for *mle* estimation.

$$\begin{aligned}\hat{\theta}_{mle} &= \underset{\theta}{\operatorname{argmax}} \sum_{(x,z)\in(X,Z)} \#(X = x, Z = z) \log \Pr(X = x, Z = z|\theta) \\ &= \underset{\theta}{\operatorname{argmax}} \sum_{(x,z)\in D} \sum_{y\in\{Red,Blue\}} \delta(z, y) \log \Pr(X = x, Z = z|\theta)\end{aligned}$$

where $\delta(x, y) = 1 \iff x = y$ otherwise 0.

We observed a sample D drawn from $(x, z) \in (X, Z)$ where $X \in \{H, T\}$, $Z = \{Red, Blue\}$. This time Z is hidden.

$$\hat{\theta}_{mle} = \underset{\theta}{\operatorname{argmax}} \sum_{(x,z)\in D} \sum_{y\in\{Red,Blue\}} \delta(z, y) \log \Pr(X = x, Z = z|\theta)$$

We observed a sample D drawn from $(x, z) \in (X, Z)$ where $X \in \{H, T\}$, $Z = \{Red, Blue\}$. This time Z is hidden.

$$\hat{\theta}_{mle} = \underset{\theta}{\operatorname{argmax}} \sum_{(x,z) \in D} \sum_{y \in \{Red, Blue\}} \delta(z, y) \log \Pr(X = x, Z = z | \theta)$$

Replace $\delta(z, y) \in \{0, 1\}$ by our best guess $\Pr(Z = z | X = x, \theta_i)$.

$$\hat{\theta}_{i+1} = \underset{\theta}{\operatorname{argmax}} \sum_{x \in D} \sum_{z \in \{Red, Blue\}} \Pr(Z = z | X = x, \theta_i) \log \Pr(X = x, Z = z | \theta_i)$$

We observed a sample D drawn from $(x, z) \in (X, Z)$ where $X \in \{H, T\}$, $Z = \{Red, Blue\}$. This time Z is hidden.

$$\hat{\theta}_{mle} = \underset{\theta}{\operatorname{argmax}} \sum_{(x,z) \in D} \sum_{y \in \{Red, Blue\}} \delta(z, y) \log \Pr(X = x, Z = z | \theta)$$

Replace $\delta(z, y) \in \{0, 1\}$ by our best guess $\Pr(Z = z | X = x, \theta_i)$.

$$\hat{\theta}_{i+1} = \underset{\theta}{\operatorname{argmax}} \sum_{x \in D} \sum_{z \in \{Red, Blue\}} \Pr(Z = z | X = x, \theta_i) \log \Pr(X = x, Z = z | \theta_i)$$

This term is known as the *expected log-likelihood*.

- Initialize the parameters θ_0 somehow (randomly?)

- Initialize the parameters θ_0 somehow (randomly?)
- E-step: Compute $\Pr(Z|X, \theta_i)$ i.e. our best guess of the hidden data Z given our current parameters.

- Initialize the parameters θ_0 somehow (randomly?)
- E-step: Compute $\Pr(Z|X, \theta_i)$ i.e. our best guess of the hidden data Z given our current parameters.
- M-step: Update the parameters θ_{i+1} to maximize the expected log-likelihood.

EXPECTATION MAXIMIZATION

- Initialize the parameters θ_0 somehow (randomly?)
- E-step: Compute $\Pr(Z|X, \theta_i)$ i.e. our best guess of the hidden data Z given our current parameters.
- M-step: Update the parameters θ_{i+1} to maximize the expected log-likelihood.
- Iterate until the expected log-likelihood stops increasing.

- Initialize the parameters θ_0 somehow (randomly?)
- E-step: Compute $\Pr(Z|X, \theta_i)$ i.e. our best guess of the hidden data Z given our current parameters.
- M-step: Update the parameters θ_{i+1} to maximize the expected log-likelihood.
- Iterate until the expected log-likelihood stops increasing.

Intuition: if we knew θ we could just infer Z, likewise if we knew Z we could just estimate θ. Since we don't know either, just guess and iteratively improve.

EM MAXIMIZES A BOUND ON THE OBSERVED LIKELIHOOD

$$
\begin{aligned}
\log \Pr(X|\theta) &= \log \sum_Z \Pr(X|\theta)\Pr(Z|X,\theta) \\
&= \log \sum_Z q(Z) \frac{\Pr(X|\theta)\Pr(Z|X,\theta)}{q(Z)} \\
&\geq \sum_Z q(Z) \log \frac{\Pr(X|\theta)\Pr(Z|X,\theta)}{q(Z)} \\
&= \sum_Z q(Z) \log \Pr(X|\theta) - \sum_Z q(Z) \log \frac{q(Z)}{\Pr(Z|,X,\theta)} \\
&= \log \Pr(X|\theta) - KL(q(Z)||\Pr(Z|,X,\theta))
\end{aligned}
$$

which implies that if $q(Z) = \Pr(Z|,X,\theta)$ the bound is tight.

Take a look at `mle_em_seminar.ipynb`