Module 19.2: The concept of a latent variable

- We now introduce the concept of a latent variable

- We now introduce the concept of a latent variable
- Recall that earlier we mentioned that the neighboring pixels in an image are dependent on each other

- We now introduce the concept of a latent variable
- Recall that earlier we mentioned that the neighboring pixels in an image are dependent on each other
- Why is it so?

- We now introduce the concept of a latent variable
- Recall that earlier we mentioned that the neighboring pixels in an image are dependent on each other
- Why is it so? (intuitively, because we expect them to have the same color, texture, etc.?)

- We now introduce the concept of a latent variable
- Recall that earlier we mentioned that the neighboring pixels in an image are dependent on each other
- Why is it so? (intuitively, because we expect them to have the same color, texture, etc.?)
- Let us probe this intuition a bit more and try to formalize it

- Suppose we asked a friend to send us a good wallpaper and he/she thinks a bit about it and sends us this image

- Suppose we asked a friend to send us a good wallpaper and he/she thinks a bit about it and sends us this image
- Why are all the pixels in the top portion of the image blue?

- Suppose we asked a friend to send us a good wallpaper and he/she thinks a bit about it and sends us this image
- Why are all the pixels in the top portion of the image blue? (because our friend decided to show us an image of the sky as opposed to mountains or green fields)

- Suppose we asked a friend to send us a good wallpaper and he/she thinks a bit about it and sends us this image
- Why are all the pixels in the top portion of the image blue? (because our friend decided to show us an image of the sky as opposed to mountains or green fields)
- But then why blue why not black?

- Suppose we asked a friend to send us a good wallpaper and he/she thinks a bit about it and sends us this image
- Why are all the pixels in the top portion of the image blue? (because our friend decided to show us an image of the sky as opposed to mountains or green fields)
- But then why blue why not black? (because our friend decided to show us an image which depicts daytime as opposed to night time)

- Suppose we asked a friend to send us a good wallpaper and he/she thinks a bit about it and sends us this image
- Why are all the pixels in the top portion of the image blue? (because our friend decided to show us an image of the sky as opposed to mountains or green fields)
- But then why blue why not black? (because our friend decided to show us an image which depicts daytime as opposed to night time)
- Okay, But why is it not cloudy (gray)?

- Suppose we asked a friend to send us a good wallpaper and he/she thinks a bit about it and sends us this image
- Why are all the pixels in the top portion of the image blue? (because our friend decided to show us an image of the sky as opposed to mountains or green fields)
- But then why blue why not black? (because our friend decided to show us an image which depicts daytime as opposed to night time)
- Okay, But why is it not cloudy (gray)?(because our friend decided to show us an image which depicts a sunny day)

- Suppose we asked a friend to send us a good wallpaper and he/she thinks a bit about it and sends us this image
- Why are all the pixels in the top portion of the image blue? (because our friend decided to show us an image of the sky as opposed to mountains or green fields)
- But then why blue why not black? (because our friend decided to show us an image which depicts daytime as opposed to night time)
- Okay, But why is it not cloudy (gray)?(because our friend decided to show us an image which depicts a sunny day)
- These decisions made by our friend (sky, sunny, daytime, etc) are not explicitly known to us (they are hidden from us)

- Suppose we asked a friend to send us a good wallpaper and he/she thinks a bit about it and sends us this image
- Why are all the pixels in the top portion of the image blue? (because our friend decided to show us an image of the sky as opposed to mountains or green fields)
- But then why blue why not black? (because our friend decided to show us an image which depicts daytime as opposed to night time)
- Okay, But why is it not cloudy (gray)?(because our friend decided to show us an image which depicts a sunny day)
- These decisions made by our friend (sky, sunny, daytime, etc) are not explicitly known to us (they are hidden from us)
- We only observe the images but what we observe depends on these latent (hidden) decisions

Latent Variable = daytime

- So what exactly are we trying to say here?

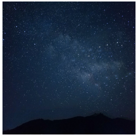
Latent Variable = daytime

- So what exactly are we trying to say here?
- We are saying that there are certain underlying hidden (latent) characteristics which are determining the pixels and their interactions

Latent Variable = daytime

- So what exactly are we trying to say here?
- We are saying that there are certain underlying hidden (latent) characteristics which are determining the pixels and their interactions
- We could think of these as additional (latent) random variables in our distribution

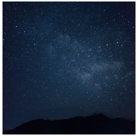
Latent Variable = daytime



Latent Variable = night

- So what exactly are we trying to say here?
- We are saying that there are certain underlying hidden (latent) characteristics which are determining the pixels and their interactions
- We could think of these as additional (latent) random variables in our distribution

Latent Variable = daytime

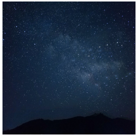


Latent Variable = night

Latent Variable = cloudy

- So what exactly are we trying to say here?
- We are saying that there are certain underlying hidden (latent) characteristics which are determining the pixels and their interactions
- We could think of these as additional (latent) random variables in our distribution

Latent Variable = daytime

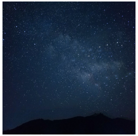


Latent Variable = night

Latent Variable = cloudy

- So what exactly are we trying to say here?
- We are saying that there are certain underlying hidden (latent) characteristics which are determining the pixels and their interactions
- We could think of these as additional (latent) random variables in our distribution
- These are latent because we do not observe them unlike the pixels which are observable random variables

Latent Variable = daytime



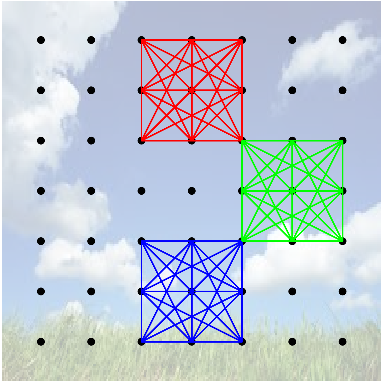
Latent Variable = night

Latent Variable = cloudy

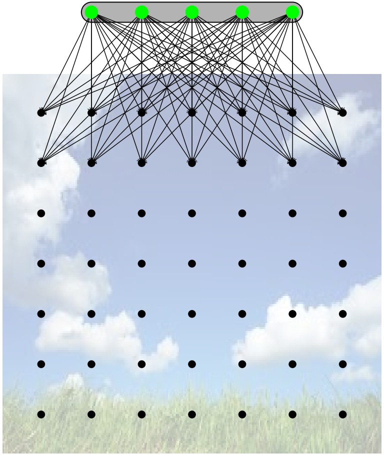
- So what exactly are we trying to say here?
- We are saying that there are certain underlying hidden (latent) characteristics which are determining the pixels and their interactions
- We could think of these as additional (latent) random variables in our distribution
- These are latent because we do not observe them unlike the pixels which are observable random variables
- The pixels depend on the choice of these latent variables

- More formally we now have visible (observed) variables or pixels ($V = \{V_1, V_2, V_3, \ldots, V_{1024}\}$) and hidden variables ($H = \{H_1, H_2, ..., H_n\}$)

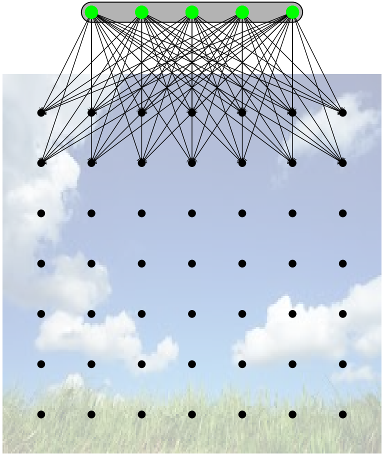
- More formally we now have visible (observed) variables or pixels ($V = \{V_1, V_2, V_3, \ldots, V_{1024}\}$) and hidden variables ($H = \{H_1, H_2, ..., H_n\}$)
- Can you now think of a Markov network to represent the joint distribution $P(V, H)$?



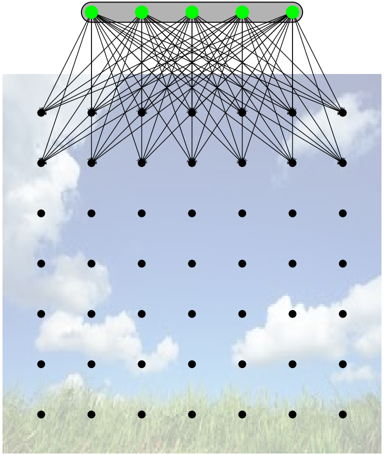
- More formally we now have visible (observed) variables or pixels ($V = \{V_1, V_2, V_3, \ldots, V_{1024}\}$) and hidden variables ($H = \{H_1, H_2, ..., H_n\}$)
- Can you now think of a Markov network to represent the joint distribution $P(V, H)$?
- Our original Markov Network suggested that the pixels were dependent on neighboring pixels (forming a clique)



- More formally we now have visible (observed) variables or pixels ($V = \{V_1, V_2, V_3, \ldots, V_{1024}\}$) and hidden variables ($H = \{H_1, H_2, ..., H_n\}$)
- Can you now think of a Markov network to represent the joint distribution $P(V, H)$?
- Our original Markov Network suggested that the pixels were dependent on neighboring pixels (forming a clique)
- But now we could have a better Markov Network involving these latent variables



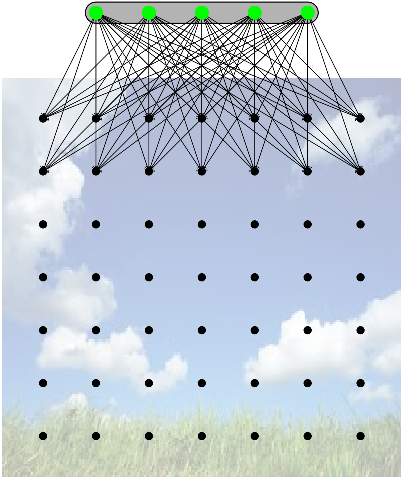
- More formally we now have visible (observed) variables or pixels ($V = \{V_1, V_2, V_3, \ldots, V_{1024}\}$) and hidden variables ($H = \{H_1, H_2, ..., H_n\}$)
- Can you now think of a Markov network to represent the joint distribution $P(V, H)$?
- Our original Markov Network suggested that the pixels were dependent on neighboring pixels (forming a clique)
- But now we could have a better Markov Network involving these latent variables
- This Markov Network suggests that the pixels (observed variables) are dependent on the latent variables (which is exactly the intuition that we were trying to build in the previous slides)



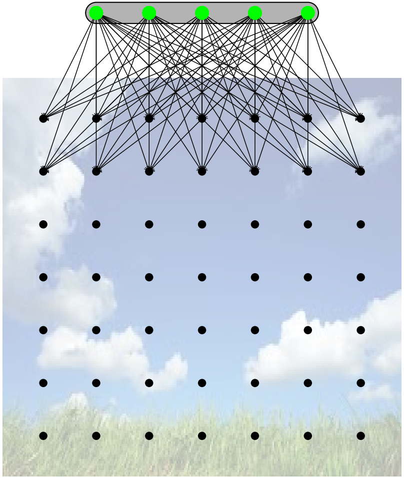
- More formally we now have visible (observed) variables or pixels ($V = \{V_1, V_2, V_3, \ldots, V_{1024}\}$) and hidden variables ($H = \{H_1, H_2, ..., H_n\}$)
- Can you now think of a Markov network to represent the joint distribution $P(V, H)$?
- Our original Markov Network suggested that the pixels were dependent on neighboring pixels (forming a clique)
- But now we could have a better Markov Network involving these latent variables
- This Markov Network suggests that the pixels (observed variables) are dependent on the latent variables (which is exactly the intuition that we were trying to build in the previous slides)
- The interactions between the pixels are captured through the latent variables

- Before we move on to more formal definitions and equations, let us probe the idea of using latent variables a bit more

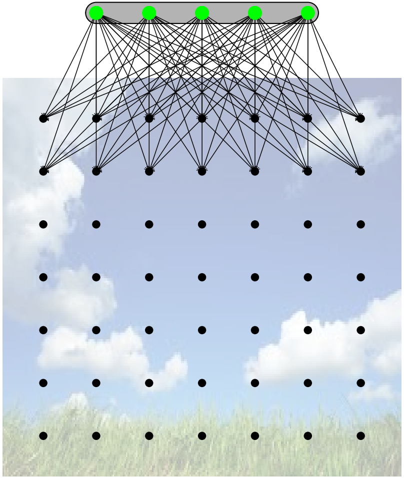
- Before we move on to more formal definitions and equations, let us probe the idea of using latent variables a bit more
- We will talk about two concepts: *abstraction* and *generation*



- First let us talk about *abstraction*

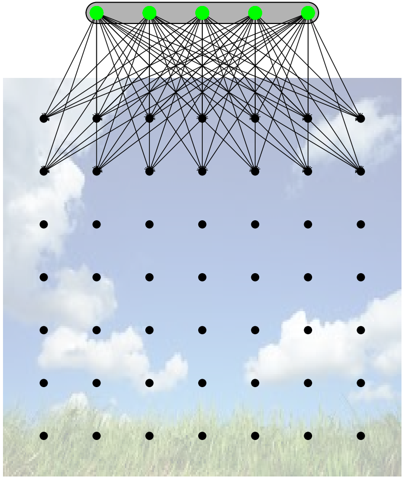


- First let us talk about *abstraction*
- Suppose, we are able to learn the joint distribution $P(V, H)$



- First let us talk about *abstraction*
- Suppose, we are able to learn the joint distribution $P(V, H)$
- Using this distribution we can find

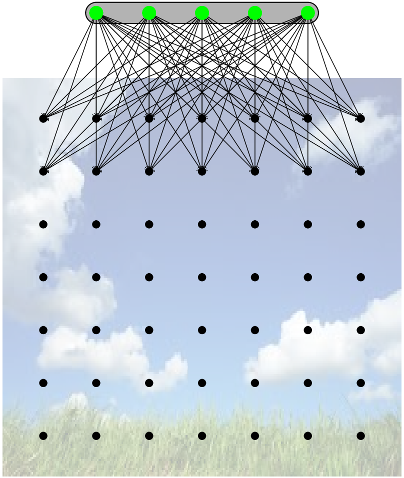
$$P(H|V) = \frac{P(V, H)}{\sum_H P(V, H)}$$



- First let us talk about *abstraction*
- Suppose, we are able to learn the joint distribution $P(V, H)$
- Using this distribution we can find

$$P(H|V) = \frac{P(V, H)}{\sum_H P(V, H)}$$

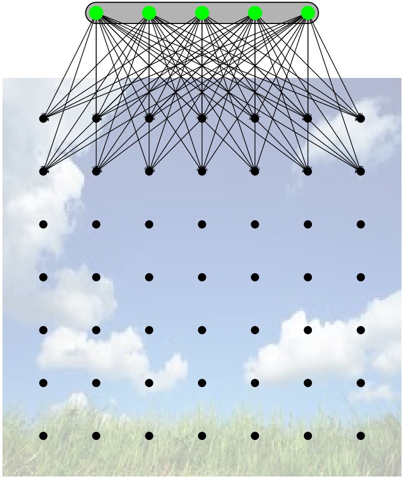
- In other words, given an image, we can find the most likely latent configuration ($H = h$) that generated this image (of course, keeping the computational cost aside for now)



- First let us talk about *abstraction*
- Suppose, we are able to learn the joint distribution $P(V, H)$
- Using this distribution we can find

$$P(H|V) = \frac{P(V, H)}{\sum_H P(V, H)}$$

- In other words, given an image, we can find the most likely latent configuration $(H = h)$ that generated this image (of course, keeping the computational cost aside for now)
- What does this h capture?



- First let us talk about *abstraction*
- Suppose, we are able to learn the joint distribution $P(V, H)$
- Using this distribution we can find

$$P(H|V) = \frac{P(V, H)}{\sum_H P(V, H)}$$

- In other words, given an image, we can find the most likely latent configuration ($H = h$) that generated this image (of course, keeping the computational cost aside for now)
- What does this h capture? It captures a latent representation or abstraction of the image!

- In other words, it captures the most important properties of the image

- In other words, it captures the most important properties of the image
- For example, if you were to describe the adjacent image you wouldn't say "I am looking at an image where pixel 1 is blue, pixel 2 is blue, ..., pixel 1024 is beige"

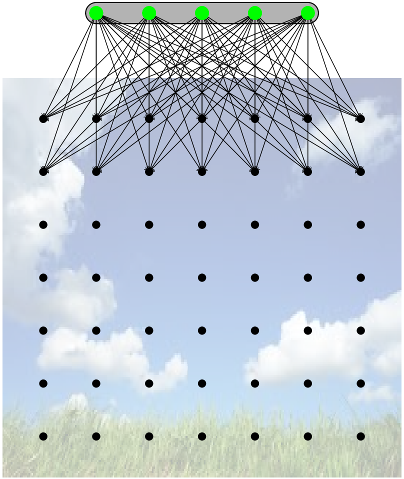
- In other words, it captures the most important properties of the image
- For example, if you were to describe the adjacent image you wouldn't say "I am looking at an image where pixel 1 is blue, pixel 2 is blue, ..., pixel 1024 is beige"
- Instead you would just say "I am looking at an image of a sunny beach with an ocean in the background and beige sand"

- In other words, it captures the most important properties of the image
- For example, if you were to describe the adjacent image you wouldn't say "I am looking at an image where pixel 1 is blue, pixel 2 is blue, ..., pixel 1024 is beige"
- Instead you would just say "I am looking at an image of a sunny beach with an ocean in the background and beige sand"
- This is exactly the abstraction captured by the vector h

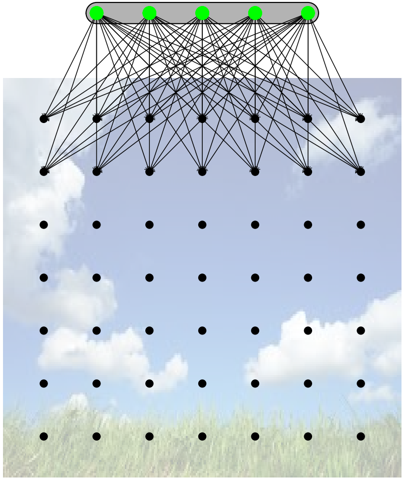
- Under this abstraction all these images would look very similar (i.e., they would have very similar latent configurations h)

- Under this abstraction all these images would look very similar (i.e., they would have very similar latent configurations h)
- Even though in the original feature space (pixels) there is a significant difference between these images, in the latent space they would be very close to each other

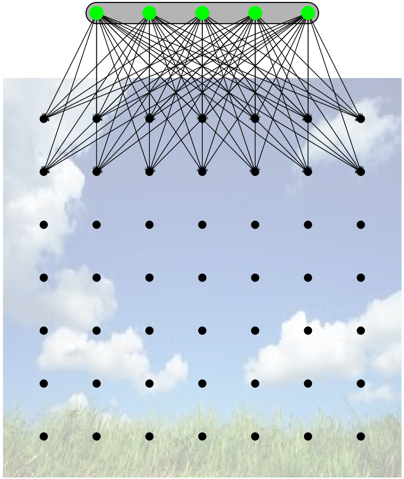
- Under this abstraction all these images would look very similar (i.e., they would have very similar latent configurations h)
- Even though in the original feature space (pixels) there is a significant difference between these images, in the latent space they would be very close to each other
- This is very similar to the idea behind PCA and autoencoders



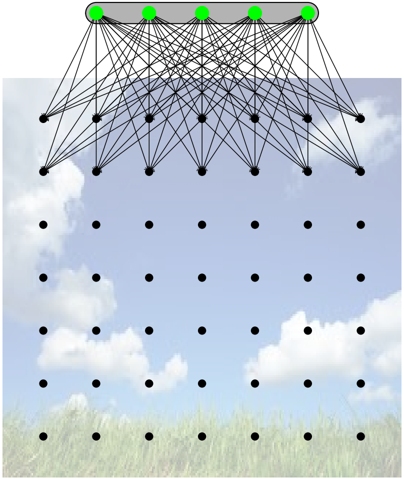
- Of course, we still need to figure out a way of computing $P(H|V)$



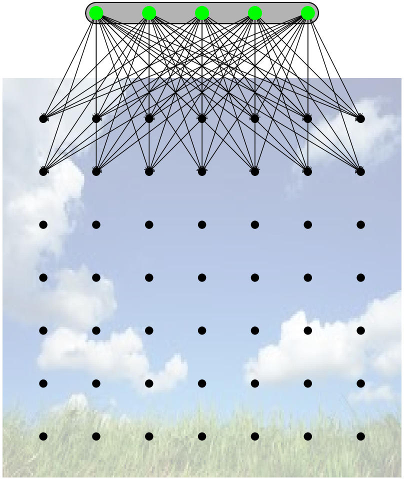
- Of course, we still need to figure out a way of computing $P(H|V)$
- In the case of PCA, learning such latent representations boiled down to learning the eigen vectors of $X^{\top}X$ (using linear algebra)



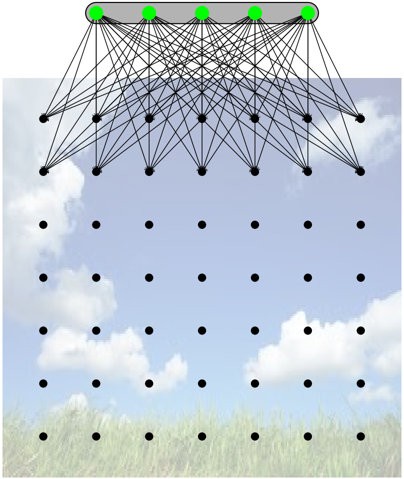
- Of course, we still need to figure out a way of computing $P(H|V)$
- In the case of PCA, learning such latent representations boiled down to learning the eigen vectors of $X^\top X$ (using linear algebra)
- In the case of Autoencoders, this boiled down to learning the parameters of the feedforward network (W_{end}, W_{dec}) (using gradient descent)



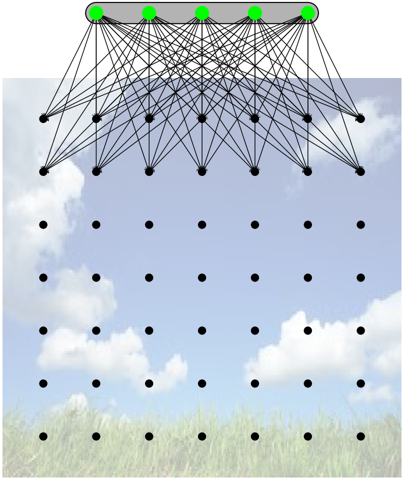
- Of course, we still need to figure out a way of computing $P(H|V)$
- In the case of PCA, learning such latent representations boiled down to learning the eigen vectors of $X^\top X$ (using linear algebra)
- In the case of Autoencoders, this boiled down to learning the parameters of the feedforward network (W_{end}, W_{dec}) (using gradient descent)
- We still haven't seen how to learn the parameters of $P(H, V)$ (we are far from it but we will get there soon!)



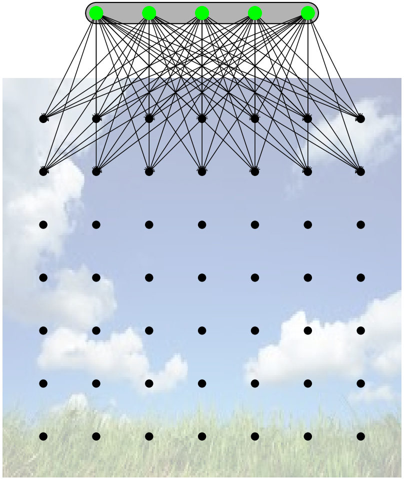
- Ok, I am just going to drag this a bit more! (bear with me)



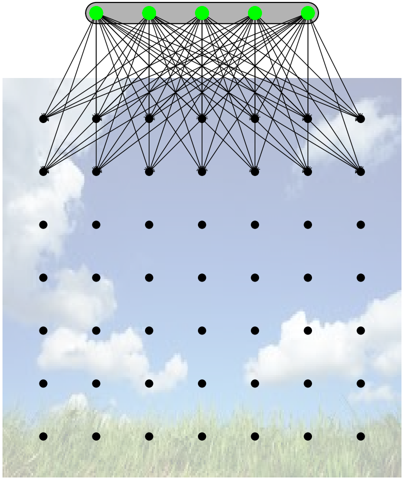
- Ok, I am just going to drag this a bit more! (bear with me)
- Remember that in practice we have no clue what these hidden variables are!



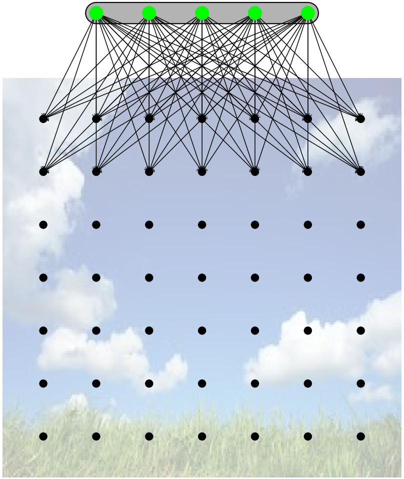
- Ok, I am just going to drag this a bit more! (bear with me)
- Remember that in practice we have no clue what these hidden variables are!
- Even in PCA, once we are given the new dimensions we have no clue what these dimensions actually mean



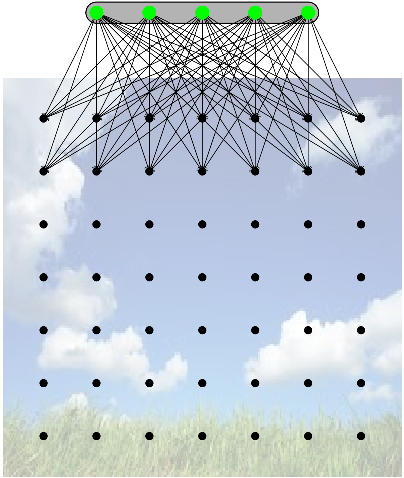
- Ok, I am just going to drag this a bit more! (bear with me)
- Remember that in practice we have no clue what these hidden variables are!
- Even in PCA, once we are given the new dimensions we have no clue what these dimensions actually mean
- We cannot interpret them (for example, we cannot say dimension 1 corresponds to weight, dimension 2 corresponds to height and so on!)



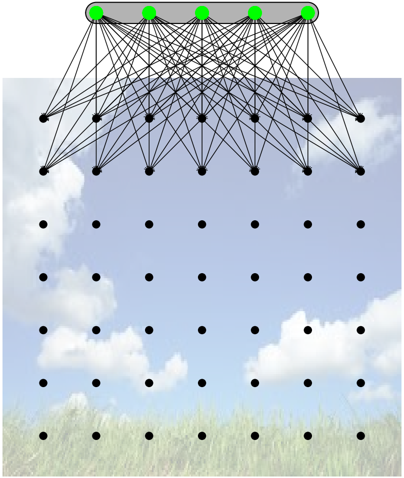
- Ok, I am just going to drag this a bit more! (bear with me)
- Remember that in practice we have no clue what these hidden variables are!
- Even in PCA, once we are given the new dimensions we have no clue what these dimensions actually mean
- We cannot interpret them (for example, we cannot say dimension 1 corresponds to weight, dimension 2 corresponds to height and so on!)
- Even here, we just assume there are some latent variables which capture the essence of the data but we do not really know what these are (because no one ever tells us what these are)



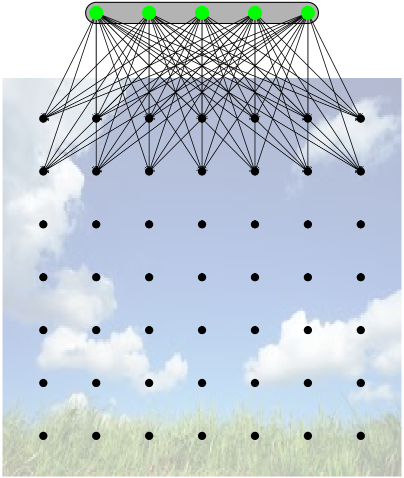
- Ok, I am just going to drag this a bit more! (bear with me)
- Remember that in practice we have no clue what these hidden variables are!
- Even in PCA, once we are given the new dimensions we have no clue what these dimensions actually mean
- We cannot interpret them (for example, we cannot say dimension 1 corresponds to weight, dimension 2 corresponds to height and so on!)
- Even here, we just assume there are some latent variables which capture the essence of the data but we do not really know what these are (because no one ever tells us what these are)
- Only for illustration purpose we assumed that h_1 corresponds to sunny/cloudy, h_2 corresponds to beach and so on



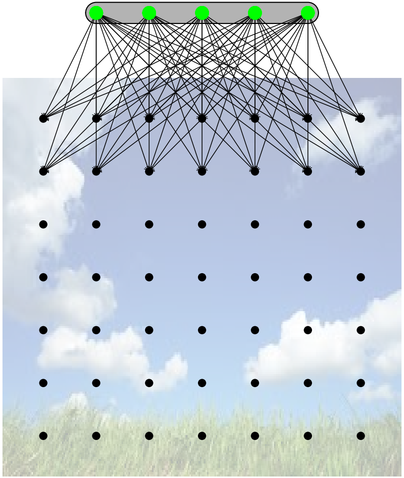
- Just to reiterate, remember that while sending us the wallpaper images our friend never told us what latent variables he/she considered



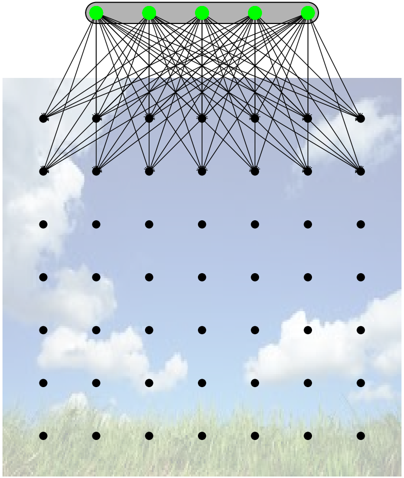
- Just to reiterate, remember that while sending us the wallpaper images our friend never told us what latent variables he/she considered
- Maybe our friend had the following latent variables in mind: $h_1 = cheerful$, $h_2 = romantic$, and so on



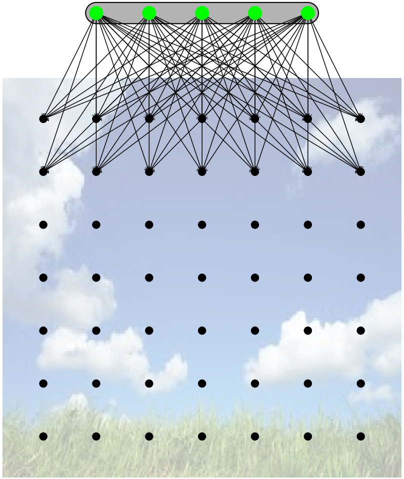
- Just to reiterate, remember that while sending us the wallpaper images our friend never told us what latent variables he/she considered
- Maybe our friend had the following latent variables in mind: $h_1 = cheerful$, $h_2 = romantic$, and so on
- In fact, it doesn't really matter what the interpretation of these latent variable is



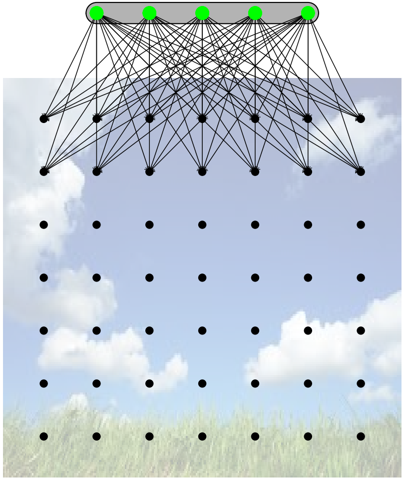
- Just to reiterate, remember that while sending us the wallpaper images our friend never told us what latent variables he/she considered
- Maybe our friend had the following latent variables in mind: $h_1 = cheerful$, $h_2 = romantic$, and so on
- In fact, it doesn't really matter what the interpretation of these latent variable is
- All we care about is that they should help us learn a good abstraction of the data



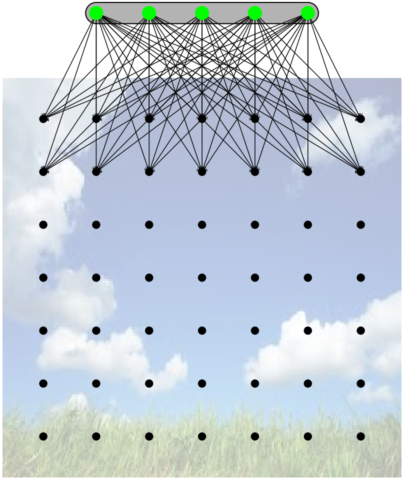
- Just to reiterate, remember that while sending us the wallpaper images our friend never told us what latent variables he/she considered
- Maybe our friend had the following latent variables in mind: $h_1 = cheerful$, $h_2 = romantic$, and so on
- In fact, it doesn't really matter what the interpretation of these latent variable is
- All we care about is that they should help us learn a good abstraction of the data
- How? (we will get there eventually)



- We will now talk about another interesting concept related to latent variables: *generation*

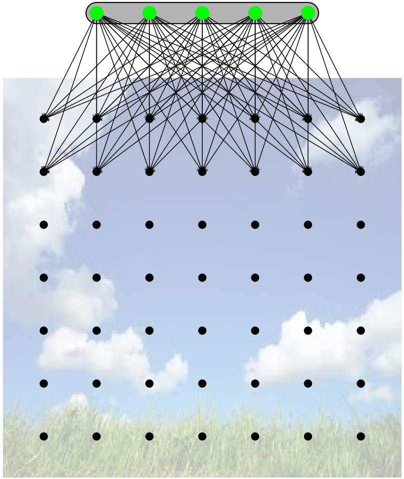


- We will now talk about another interesting concept related to latent variables: *generation*
- Once again, assume that we are able to learn the joint distribution $P(V, H)$



- We will now talk about another interesting concept related to latent variables: *generation*
- Once again, assume that we are able to learn the joint distribution $P(V, H)$
- Using this distribution we can find

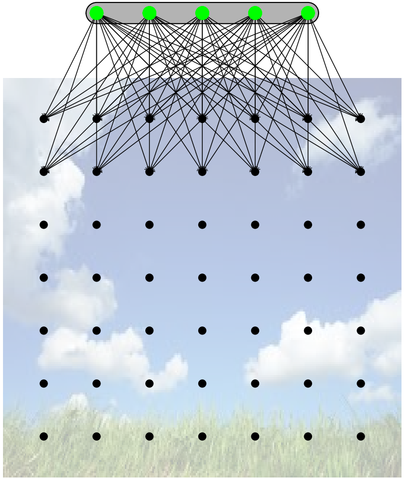
$$P(V|H) = \frac{P(V, H)}{\sum_V P(V, H)}$$



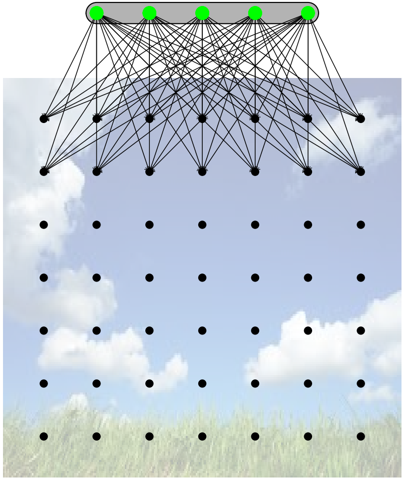
- We will now talk about another interesting concept related to latent variables: *generation*
- Once again, assume that we are able to learn the joint distribution $P(V, H)$
- Using this distribution we can find

$$P(V|H) = \frac{P(V, H)}{\sum_V P(V, H)}$$

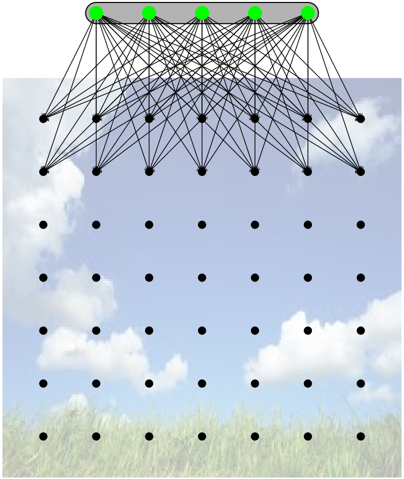
- Why is this interesting?



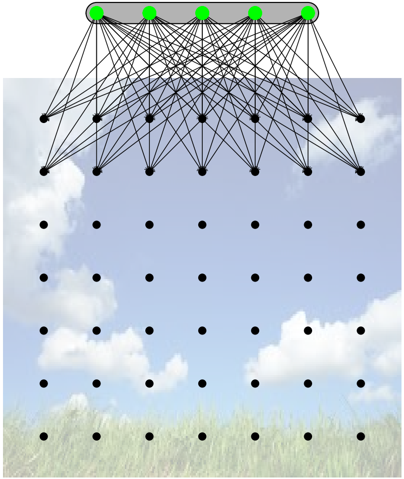
- Well, I can now say "Create an image which is cloudy, has a beach and depicts daytime"



- Well, I can now say "Create an image which is cloudy, has a beach and depicts daytime"
- Or given $h = [....]$ find the corresponding V which maximizes $P(V|H)$



- Well, I can now say "Create an image which is cloudy, has a beach and depicts daytime"
- Or given $h = [....]$ find the corresponding V which maximizes $P(V|H)$
- In other words, I can now generate images given certain latent variables



- Well, I can now say "Create an image which is cloudy, has a beach and depicts daytime"
- Or given $h = [....]$ find the corresponding V which maximizes $P(V|H)$
- In other words, I can now generate images given certain latent variables
- The hope is that I should be able to ask the model to generate very creative images given some latent configuration (we will come back to this later)

The story ahead...

- We have tried to understand the intuition behind latent variables and how they could potenatially allow us to do abstraction and generation

The story ahead...

- We have tried to understand the intuition behind latent variables and how they could potenatially allow us to do abstraction and generation
- We will now concretize these intuitions by developings equations (models) and learning algoritms

The story ahead...

- We have tried to understand the intuition behind latent variables and how they could potenatially allow us to do abstraction and generation
- We will now concretize these intuitions by developings equations (models) and learning algoritms
- And of course, we will tie all this back to neural networks!

- For the remainder of this discussion we will assume that all our variables take only boolean values

- For the remainder of this discussion we will assume that all our variables take only boolean values
- Thus, the vector V will be a boolean vector $\in \{0, 1\}^m$ (there are a total of 2^m values that V can take)

- For the remainder of this discussion we will assume that all our variables take only boolean values
- Thus, the vector V will be a boolean vector $\in \{0, 1\}^m$ (there are a total of 2^m values that V can take)
- And the vector H will be a boolean vector $\in \{0, 1\}^n$ (there are a total of 2^n values that H can take)