✖

Lecture 14 Quiz

Back to Week 14

✔   5/5 points earned
(100%)

Quiz passed!

Be Recognized for Your Achievements.

"Course Certificates give you the recognition you need to get the job, the material gives you the skills to do the job. It makes you look more valuable because you are more valuable." - Peter B., USA, Software Developer

Showcase Your Accomplishment! Earn Your Course Certificate! CA$65 ❯

✔   1 / 1
points

1.
Why is a Deep Belief Network not a Boltzmann Machine ?

◯   All edges in a DBN are directed.

◯   A DBN is not a probabilistic model of the data.

◯   A DBN does not have hidden units.

◉   Some edges in a DBN are directed.

Correct
In a Boltzmann Machine, all edges must be undirected. A DBN has directed edges from the top-level RBM to each subsequent layer below.

✔   1 / 1
          points

2.

Brian looked at the direction of arrows in a DBN and was surprised to find that the data is at the "output". "Where is the input ?!", he exclaimed, "How will I give input to this model and get all those cool features?" In this context, which of the following statements are true? Check all that apply.

☑ A DBN is a generative model of the data, which means that, its arrows define a way of generating data from a probability distribution, so there is no "input".

Correct

Unlike a feed-forward neural net, a DBN is not a mapping from inputs to outputs. It is a probabilistic generative model of the data.

☐ A DBN is a generative model of the data and cannot be used to generate features for any given input. It can only be used to get features for data that was generated by the model.

Un-selected is correct

☐ In order to get features h given some data v, he must perform inference to find out $P(h|v)$. There is an easy **exact** way of doing this, just traverse the arrows in the opposite direction.

Un-selected is correct

☑ In order to get features h given some data v, he must perform inference to find out $P(h|v)$. There is an easy **approximate** way of doing this, just traverse the arrows in the opposite direction.
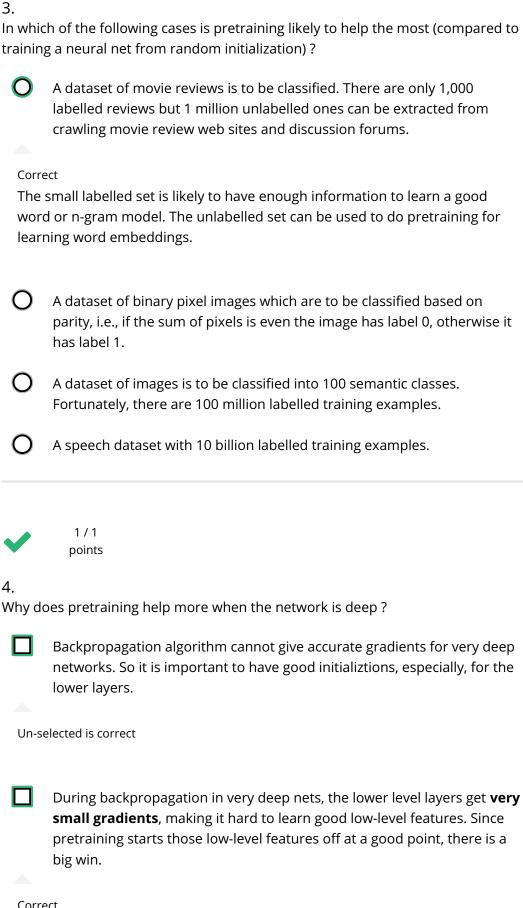
Correct

Traversing arrows in the opposite direction is an approximate inference procedure.

✔   1 / 1
          points

3.

In which of the following cases is pretraining likely to help the most (compared to training a neural net from random initialization) ?

⊙   A dataset of movie reviews is to be classified. There are only 1,000 labelled reviews but 1 million unlabelled ones can be extracted from crawling movie review web sites and discussion forums.

▲

Correct

The small labelled set is likely to have enough information to learn a good word or n-gram model. The unlabelled set can be used to do pretraining for learning word embeddings.

○   A dataset of binary pixel images which are to be classified based on parity, i.e., if the sum of pixels is even the image has label 0, otherwise it has label 1.

○   A dataset of images is to be classified into 100 semantic classes. Fortunately, there are 100 million labelled training examples.

○   A speech dataset with 10 billion labelled training examples.

✔   1 / 1
          points

4.

Why does pretraining help more when the network is deep ?

☑   Backpropagation algorithm cannot give accurate gradients for very deep networks. So it is important to have good initializtions, especially, for the lower layers.

▲

Un-selected is correct

☑   During backpropagation in very deep nets, the lower level layers get **very small gradients**, making it hard to learn good low-level features. Since pretraining starts those low-level features off at a good point, there is a big win.

▲

Correct

Lower level layers can get very small gradients, especially if saturating hidden units are used (such as logistic or tanh units). Pretraining can initialize the weights in a proper region of weight space so that the features

don't have to start learning from scratch.

☐ As nets get deeper, contrastive divergence objective used during
pretraining gets closer to the classification objective.

▲

Un-selected is correct

☐ Deeper nets have more parameters than shallow ones and they overfit
easily. Therefore, initializing them sensibly is important.

▲

Correct
More parameters means that the model can find ingenious ways of
overfitting by learning features that don't generalize well. Pretraining can
initialize the weights in a proper region of weight space so that the features
learned are not too bad.

✔ 1 / 1
points

5.
The energy function for binary RBMs goes by

$$E(\mathbf{v}, \mathbf{h}) = -\sum_i v_i b_i - \sum_j h_j a_j - \sum_{i,j} v_i W_{ij} h_j$$

When modeling real-valued data (i.e., when $\mathbf{v}$ is a real-valued vector not a binary
one) we change it to

$$E(\mathbf{v}, \mathbf{h}) = \sum_i \frac{(v_i - b_i)^2}{2\sigma_i^2} - \sum_j h_j a_j - \sum_{i,j} \frac{v_i}{\sigma_i} W_{ij} h_j$$

Why can't we still use the same old one ?

☐ If we use the old one, the real-valued vectors would end up being
constrained to be binary.

▲

Un-selected is correct

☐ If the model assigns an energy e_1 to state $\mathbf{v_1}, \mathbf{h}$, and e_2 to state $\mathbf{v_2}, \mathbf{h}$,
then it would assign energy $(e_1 + e_2)/2$ to state $(\mathbf{v_1} + \mathbf{v_2})/2, \mathbf{h}$. This
does not make sense for the kind of distributions we usually want to
model.

Correct

Suppose v_1 and v_2 represent two images. We would like e_1 and e_2 to be small. This makes the energy of the average image low, but the average of two images would not look like a natural image and should not have low energy.

☑ If we continue to use the same one, then in general, there will be infinitely many $\mathbf{v}$'s and $\mathbf{h}$'s such that, $E(\mathbf{v}, \mathbf{h})$ will be infinitely small (close to $-\infty$). The probability distribution resulting from such an energy function is not useful for modeling real data.

Correct

If some $b_i < 0$, then if $v_i \rightarrow -\infty$, then $E \rightarrow -\infty$. Similarly for $b_i > 0$ if $v_i \rightarrow \infty$, then $E \rightarrow -\infty$. So the Boltzmann distribution based on this energy function would behave in weird ways.

☑ Probability distributions over real-valued data can only be modeled by having a conditional Gaussian distribution over them. So we have to use a quadratic term.

Un-selected is correct

👍 👎 🚩