Lecture 14 Quiz

Quiz, 5 questions

1 point

1.

Why is a Deep Belief Network not a Boltzmann Machine ?

- ( ) All edges in a DBN are directed.
- (•) Some edges in a DBN are directed.
- ( ) A DBN is not a probabilistic model of the data.
- ( ) A DBN does not have hidden units.

---

1 point

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.

- [ ] 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.
- [x] 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".
- [x] 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.
- [ ] 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.

1
point

Lecture 14 Quiz

Quiz, 5 questions

3.

Suppose you wanted to learn a neural net classifier. You have data and labels. All you care about is predicting the labels accurately for a test set. How can pretraining help in getting better accuracy, even though it **does not use any information** about the labels ?

- ( ) Pretraining will learn exactly the same features that a simple neural net would learn because after all, they are training on the same data set. But pretraining does not use the labels and hence it can prevent overfitting.
- ( ) The objective function used during pretraining is the same as the one used during fine-tuning. So pretraining provides more updates towards solving the same optimization problem.
- ( ) Pretraining will **always** learn features that will be useful for discrimination, no matter what the discriminative task is.
- (x) There is an **assumption** that pretraining will learn features that will be useful for discrimination and it would be difficult to learn these features using just the labels.

1
point

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.
- [x] 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.
- [x] Deeper nets have more parameters than shallow ones and they overfit easily. Therefore, initializing them sensibly is important.
- [ ] As nets get deeper, contrastive divergence objective used during pretraining gets closer to the classification objective.

1
point

Lecture 14 Quiz

Quiz, 5 questions

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 ?

- [ ] 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.
- [x] 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 $\mathbf{(v_1 + v_2)/2, h}$. This does not make sense for the kind of distributions we usually want to model.
- [x] If we use the old one, the real-valued vectors would end up being constrained to be binary.
- [x] 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.

- [ ] I, **Alan Wright**, understand that submitting work that isn't my own may result in permanent failure of this course or deactivation of my Coursera account.

Submit Quiz

Lecture 14 Quiz

Quiz, 5 questions