1. **What is the key difference between Deep Learning (DL) and Shallow Learning?**
   1. Follow-up: Can a shallow network approximate any function given enough neurons?

Both Deep learning (DL) and Shallow learning involves neural networks but the differences the amount of layers each one have. DL has multiple hidden layers which allows them to learn hierarchical representation of data. Shallow Learning only has one or a very few hidden layers

**When we add more dimensions to the data we go to DL and**

1. **Why do Deep Learning models often outperform traditional machine learning models?**
   1. Follow-up: In what scenarios might traditional ML still be preferable?

**DL outperforms traditional machine learning because of the multiple dimensions we can use travel learning. Traditional learning we need to extract features and DL requirements is a lot amount of data. To reduce cost we would do machine learning.**

1. **How does increasing the number of hidden layers help in feature extraction?**
   1. Follow-up: Can too many layers ever be a bad thing? Why?

**Adding one layer for each extraction but the bad thing makes it too complicated and leads to overfeeds. SO feature extraction is important**

1. **What is feature abstraction in deep networks? Why is it important?**
   1. Follow-up: How does a CNN abstract features from images in a hierarchical manner?
2. **Why does Deep Learning require large amounts of data compared to traditional ML models?**
   1. Follow-up: Can we train deep networks effectively on small datasets? How?

**Yes but we would select a technique like**

1. **Deep Learning models are often described as "black boxes." What does this mean?**
   1. Follow-up: How can we improve interpretability in deep learning?

**In the 2 slide of the chapter**

**We don’t know what is going on which neural**

1. **What is the vanishing gradient problem, and how does it affect deep networks?**
   1. Follow-up: Why does this problem occur with sigmoid activation?
2. **What is the difference between overfitting and underfitting in Deep Learning?**
   1. Follow-up: How can regularization techniques help prevent overfitting?
3. **What are some common solutions to the vanishing gradient problem?**
   1. Follow-up: Why is ReLU a preferred activation function over sigmoid in hidden layers?
4. **How does dropout regularization help in preventing overfitting?**
   1. Follow-up: Why do we only apply dropout during training, not inference?
5. **Why is cross-entropy loss preferred over squared loss in deep networks?**
   1. Follow-up: How does cross-entropy loss improve gradient updates?

Some of the activaztion requires square function and we can replace it with cross-entropy and it

1. **What is batch normalization, and why is it beneficial in training deep networks?**
   1. Follow-up: Does batch normalization help with the vanishing gradient problem?
2. **What role does the Adam optimizer play in training deep networks?**
   1. Follow-up: How does Adam differ from standard stochastic gradient descent (SGD)?
3. **How do CNNs differ from fully connected neural networks?**
   1. Follow-up: Why are CNNs particularly effective for image recognition?
4. **What are recurrent neural networks (RNNs), and why are they useful for sequential data?**
   1. Follow-up: What problem does LSTM solve in standard RNNs?
5. **What is transfer learning, and why is it useful for deep learning applications?**
   1. Follow-up: How does transfer learning allow training deep models with less data?

AT each layer we prevent the model to realie on the nodes and learn the pattern