Classification with multiple outputs

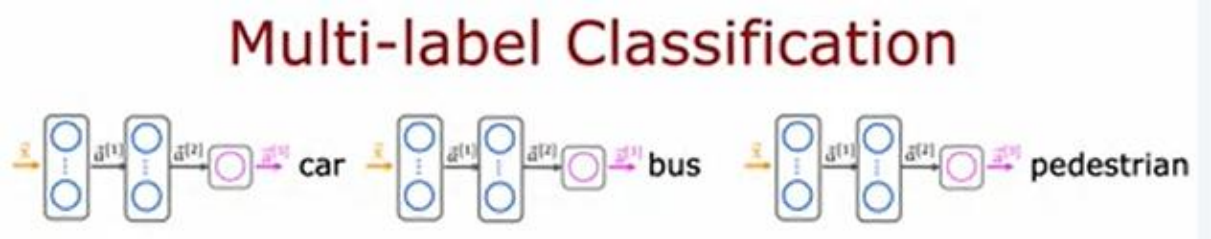
You've learned about multi-class classification, where the output label Y can be any one of two or potentially many more than two possible categories. There's a different type of classification problem called a multi-label classification problem, which is where associate of each image, they could be multiple labels. Let me show you what I mean by that.

If you're building a self-driving car or maybe a driver assistance system, then given a picture of what's in front of your car, you may want to ask a question like, is there a car or at least one car? Or is there a bus, or is there a pedestrian or are there any pedestrians? In this case, there is a car, there is no bus, and there is at least one pedestrian or in this second image, no cars, no buses and yes to pedestrians and yes car, yes bus and no pedestrians.

These are examples of multi-label classification problems because associated with a single input, image X are three different labels corresponding to whether or not there are any cars, buses, or pedestrians in the image.

In this case, the target of the Y is actually a vector of three numbers, and this is as distinct from multi-class classification, where for, say handwritten digit classification, Y was just a single number, even if that number could take on 10 different possible values. How do you build a neural network for multi-label classification?

One way to go about it is to just treat this as three completely separate machine learning problems. You could build one neural network to decide, are there any cars? The second one to detect buses and the third one to detect pedestrians. That's actually not an unreasonable approach. Here's the first neural network to detect cars, second one to detect buses, third one to detect pedestrians.

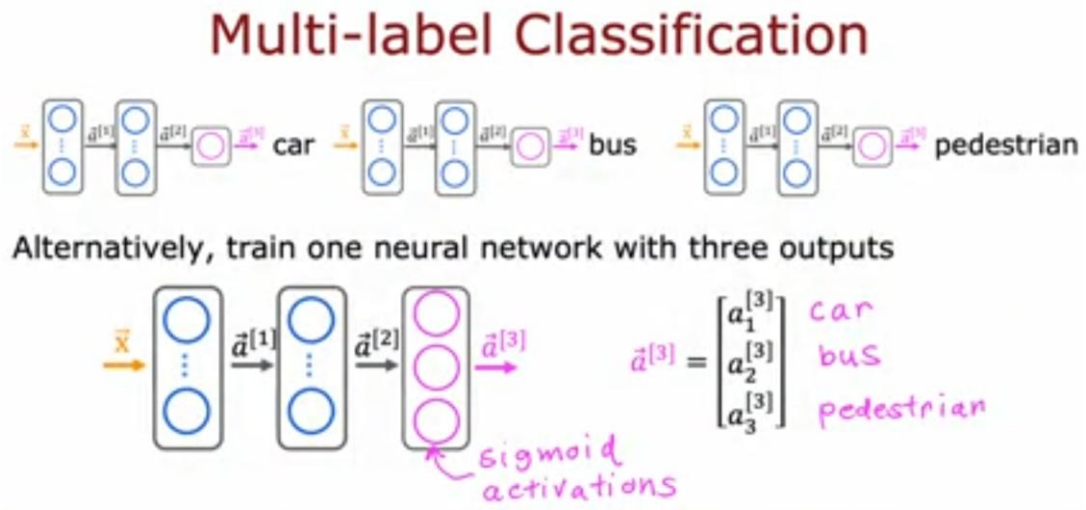


But there's another way to do this, which is to train a single neural network to simultaneously detect all three of cars, buses, and pedestrians, which is, if your neural network architecture, looks like this, there's input X.

First hidden layer offers a^1, second hidden layer offers a^2, and then the final output layer, in this case, we'll have three output neurals and we'll output a^3, which is going to be a vector of three numbers. Because we're solving three binary classification problems, so is there a car? Is there a bus? Is there a pedestrian?

You can use a sigmoid activation function for each of these three nodes in the output layer, and so a^3 in this case will be a_1^3, a_2^3, and a_3^3, corresponding to whether or not the learning [inaudible] as a car and no bus, and no pedestrians in the image.

Multi-class classification and multi-label classification are sometimes confused with each other, and that's why in this video I want to share with you just a definition of multi-label classification problems as well, so that depending on your application, you could choose the right one for the job you want to do.



So that's it for multi-label classification. I find that sometimes multi-class classification and multi-label classification are confused with other, which is why I wanted to expressively, in this video, share with you what is multi-label classification, so that depending on your application you can choose to write to for the job that you want to do.

And that wraps up the section on multi-class and multi-label classification. In the next video, we'll start to look at some more advanced neural network concepts, including an optimization algorithm that is even better than gradient descent. Let's take a look at that algorithm in the next video, because it'll help you to get your learning algorithms to learn much faster.