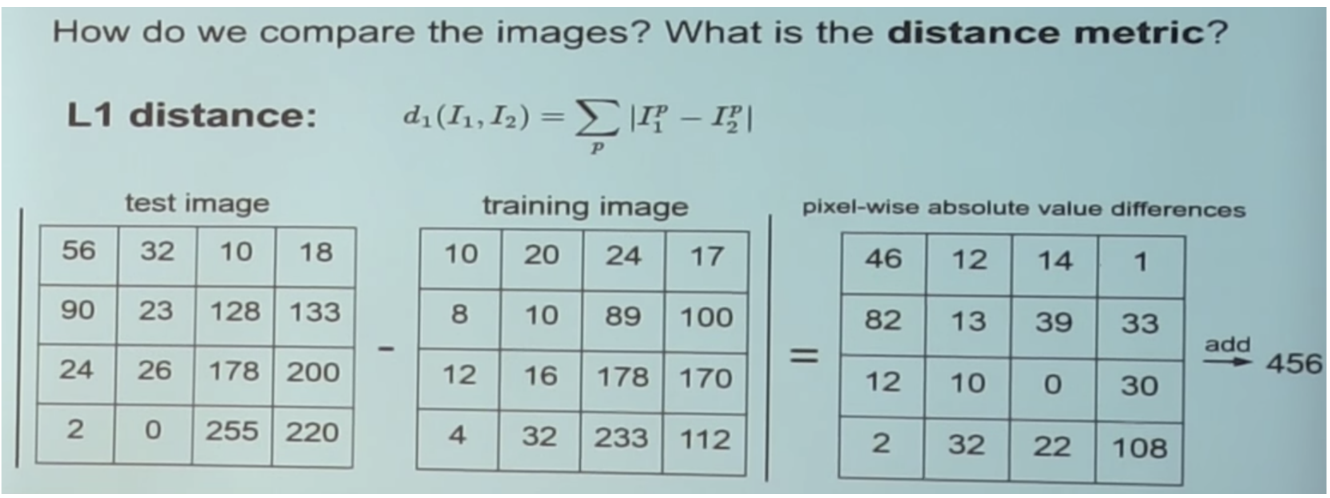
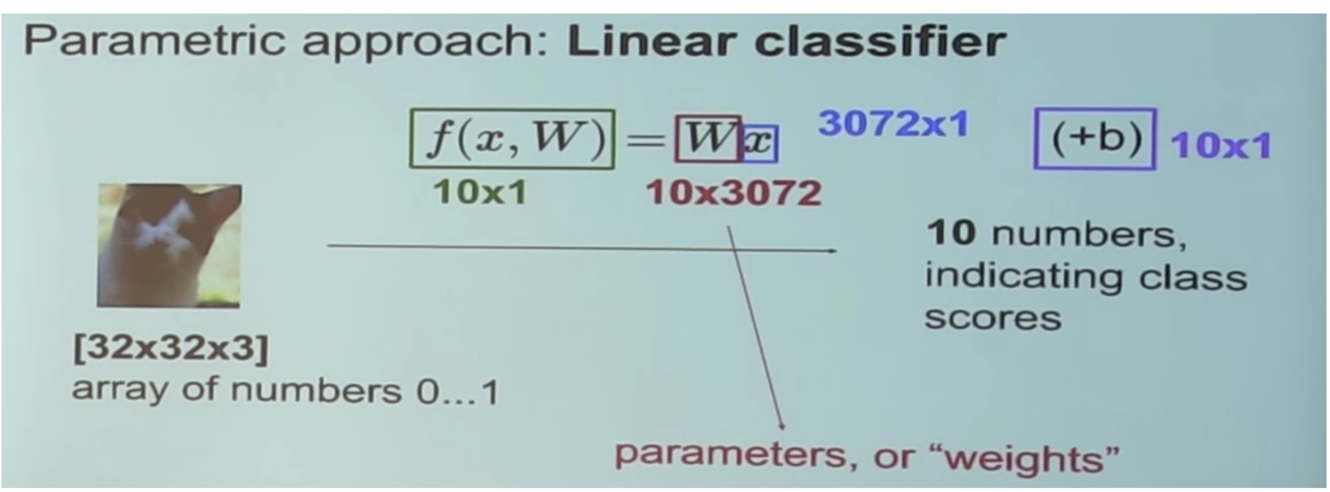
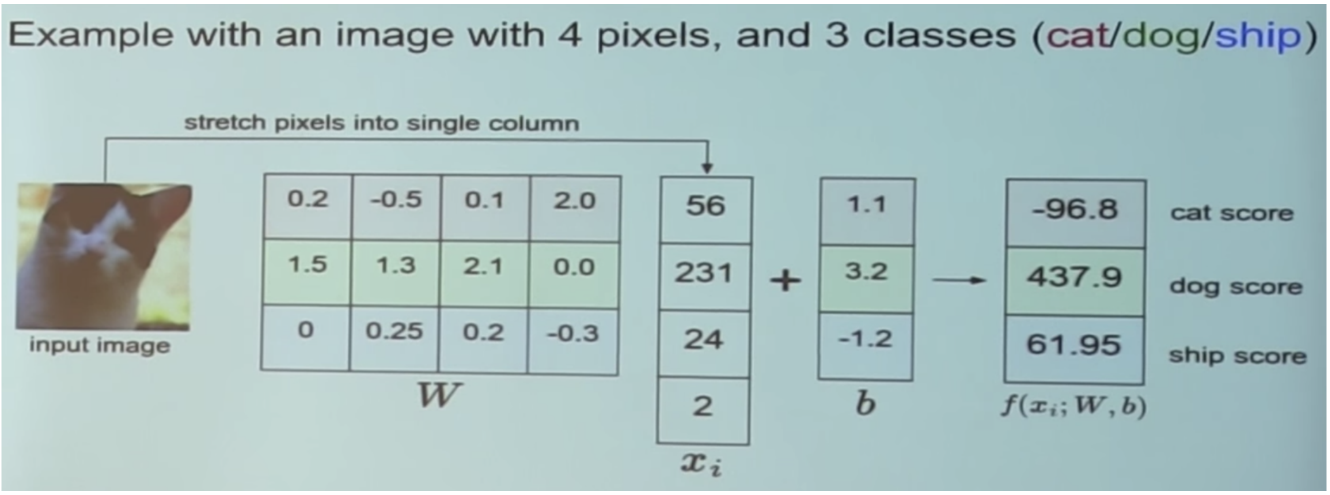
kNNs, linear classifiers with SVM loss, softmax/logistic classification

- image classification is the task of assigning an input image one label from a fixed set of categories.
- the object can be captured from various angles, at different dimesions, at different brightness, contrast, occlusion, background clutter/color. this makes it challenging. this makes the problem of classification hard
- in the past they used to extract a set of features and like edges, edge orientations, histogram of gradients etc and use this as features to a ML model like logistic regression, SVM etc.
- nearest neighbor classification: given an image and a set of training images with categories, see which one it is nearest to. classify it as that.

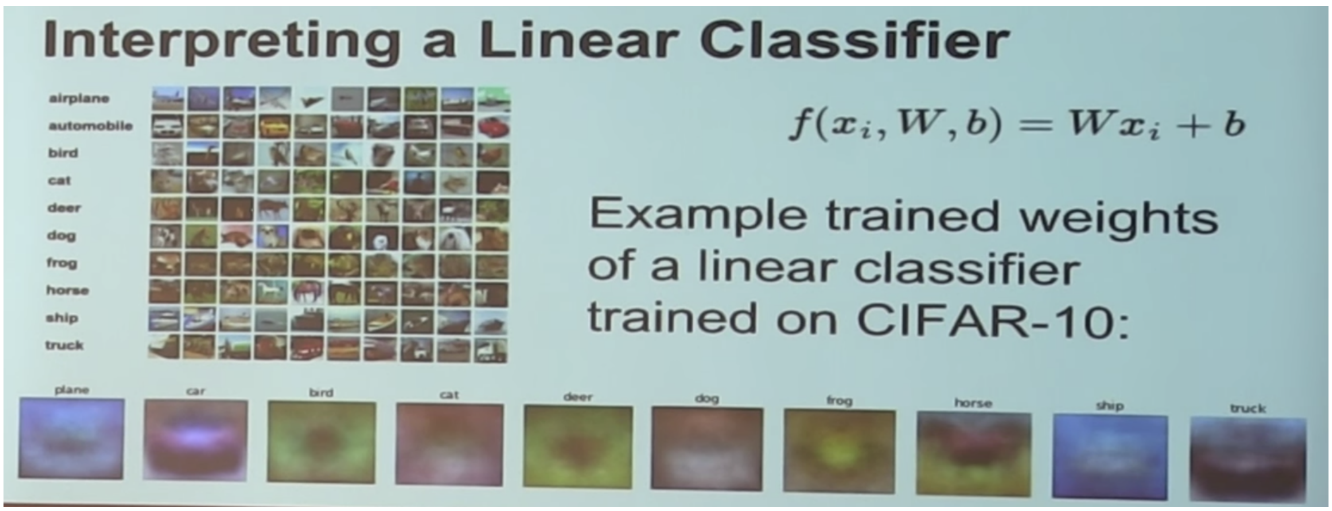


- note that the speed of classification is linear in the training dataset size!
- We could also use L2 instead of L1 distance
- The above is nearest neighbor algorithm. A kNN extension would be one in which we retrieve the k nearest neighbors and make them vote and take the one with highest vote.
- kNN is an example of a non-parametric model.
- Linear classifier



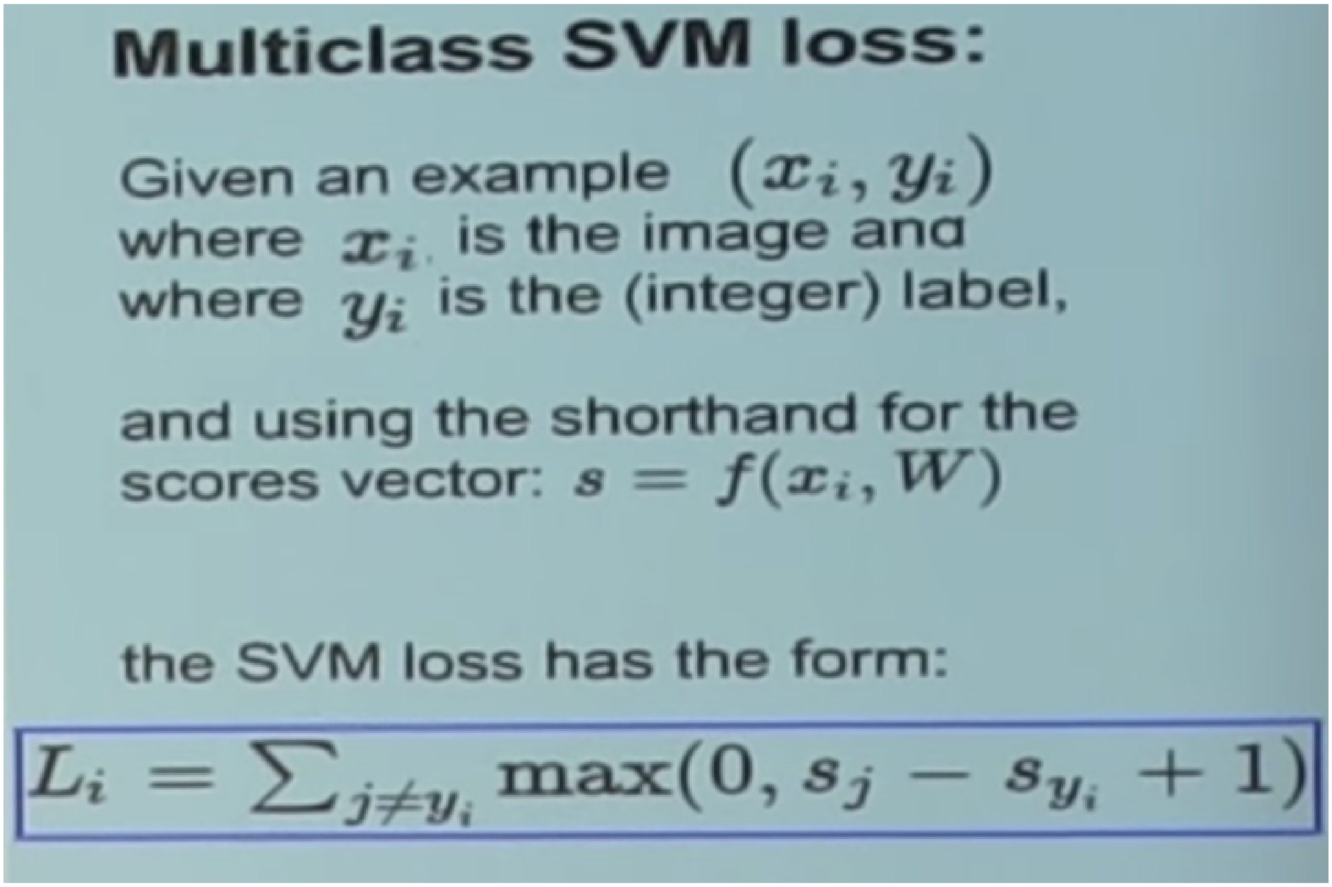


- linear classifier is basically a weighted sum. thats it
- look at the templates learnt for different image classes

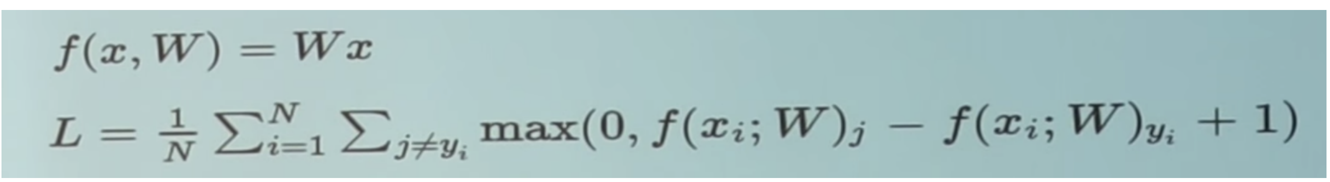


Look at the horse template. it looks like overlapping horses because it saw both left and right facing horses in the training set.

- SVM classification:
  - SVM loss function

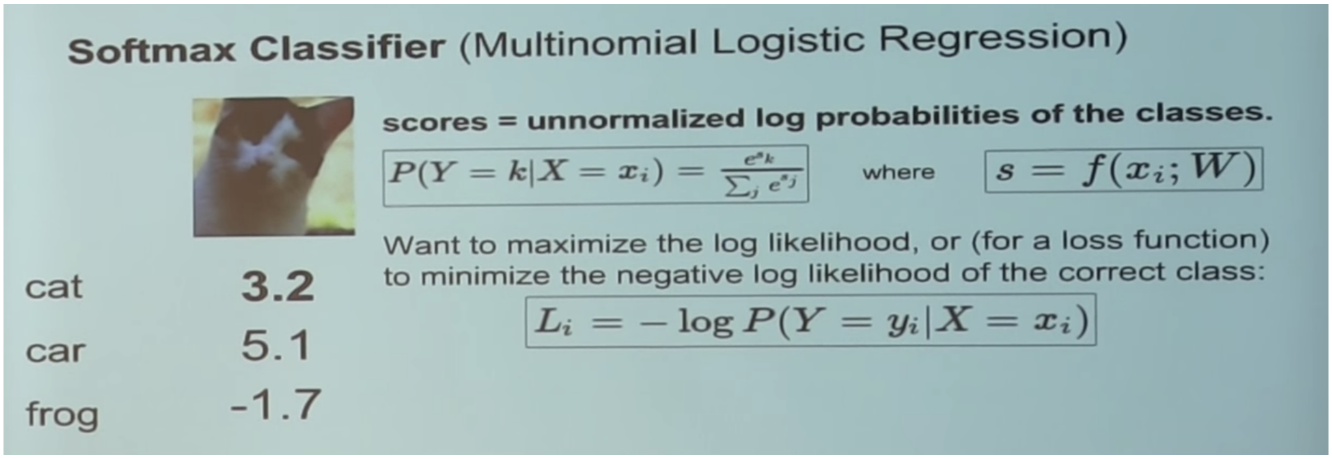


- the SVM loss function wants the score of the correct class to be larger than the incorrect class scores by at least by Δ (delta). If this is not the case, we will accumulate loss for that class.
- We could also square each term in the summation for losses. this more heavily optimizes to avoid large losses.
- The SVM loss function written out another way.



- During optimization, it could just cheat and make scale W by a constant factor to increase the max margin. To avoid this, we regularize by adding this to the loss function (L2 regularization): $\sum_k \sum_l W_{kl}^2$
- This regularization term makes the W spread out its magnitude across more parameters thus discouraging very big and very small parameters. Diffused W is better because it uses more parameters to make its classification, thus looking at more pixels and increasing robustness.
- The SVM is basically trying to **maximize its worst margin**

* softmax classification (multinomial logistic regression):



- SVM loss vs softmax loss: SVM has a hard boundary beyond which loss doesn't matter. in other words, loss is computed only for the badly performing examples that are either wrong or within the delta boundary of being wrong. Softmax on other hand has a smoother boundary, always encouraging the probability of correct class to go up, but dropping off the rate of this gradient as it assigns higher probability for the correct class.
- Using SVM and logistic regression, "features" were extracted from images, stacked into vectors and then fed into these SVM/softmax classification models. These features can be histogram of gradients, edges, etc etc. These are hand engineered. Whatever you think is relevant to the problem, write code to compute a feature and feed that into the model.