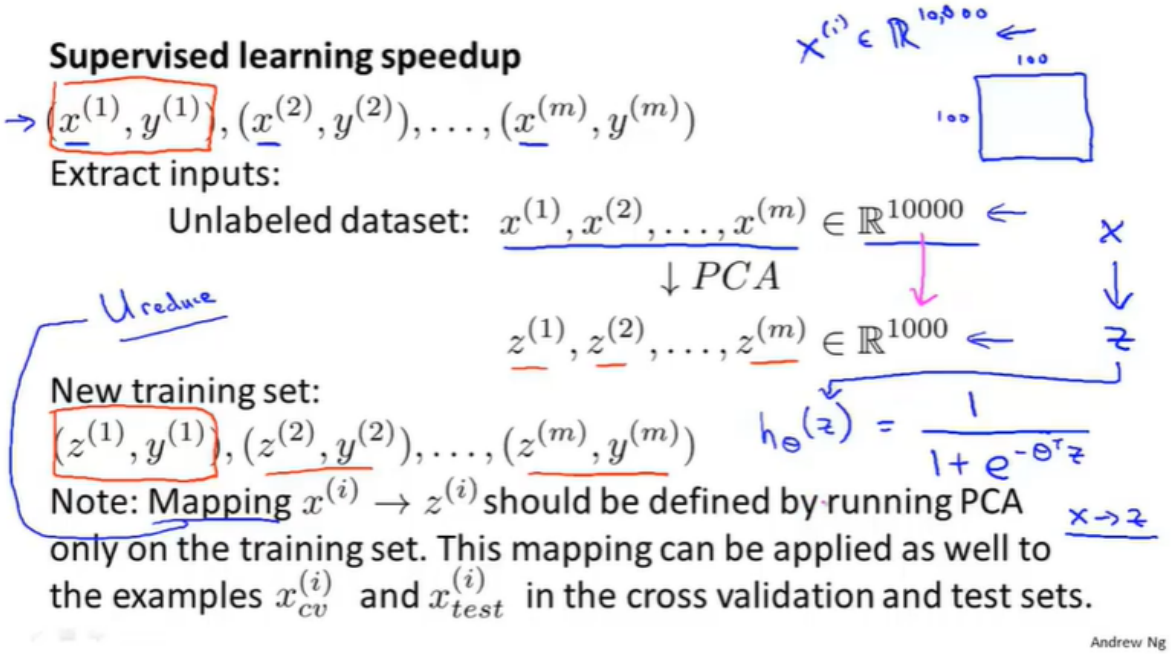
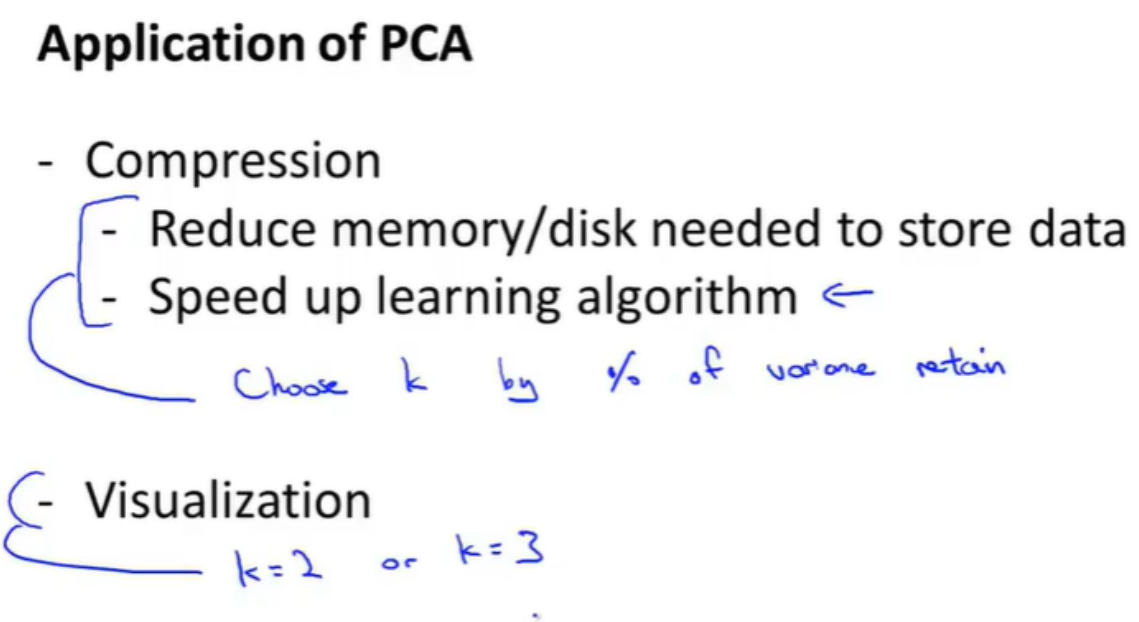
应用PCA的建议

加速有监督学习



当有100*100维度的(x，y)训练样本的时候，x(1)..x(m)维度时10000，这样运行线性回归或者让神经网络算法速度会很慢，这时我们可以在**训练集**上运行PCA，将x降维到1000，原来的0.1倍，这样可以在不影响性能的情况下加速算法，在训练集上完成后，再在验证集和测试集上运行。



PCA的误用

Bad use of PCA: To prevent overfitting

→ Use $z^{(i)}$ instead of $x^{(i)}$ to reduce the number of features to $k < n$.
Thus, fewer features, less likely to overfit.

Bad!

This might work OK, but isn't a good way to address overfitting. Use regularization instead.

$$\min_\theta \frac{1}{2m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2 + \frac{\lambda}{2m}\sum_{j=1}^{n}\theta_j^2$$

使用PCA去防止过拟合，是对于PCA的误用，因为PCA虽然会有保留99%，95%方差的原则在，但在对（x,y）进行降维的过程中，PCA并不关注y标签，尽管它可以达到防止过拟合的效果，但仍然会导致一些有用信息的损失，但**正则化**的方法不会导致有用信息的损失，它的最小化代价函数公式中仍会考虑标签的值，所以**PCA用于加速算法，而不是防止过拟合。**

在计划使用PCA之前，先考虑不用PCA的话是否可行。

PCA is sometimes used where it shouldn't be

Design of ML system:

- Get training set $\{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \ldots, (x^{(m)}, y^{(m)})\}$
- ~~Run PCA to reduce $x^{(i)}$ in dimension to get $z^{(i)}$~~
- Train logistic regression on $\{(z^{(1)}, y^{(1)}), \ldots, (z^{(m)}, y^{(m)})\}$
- Test on test set: Map $x_{test}^{(i)}$ to $z_{test}^{(i)}$. Run $h_\theta(z)$ on $\{(z_{test}^{(1)}, y_{test}^{(1)}), \ldots, (z_{test}^{(m)}, y_{test}^{(m)})\}$

→ How about doing the whole thing without using PCA?

→ Before implementing PCA, first try running whatever you want to do with the original/raw data $x^{(i)}$. Only if that doesn't do what you want, then implement PCA and consider using $z^{(i)}$.