K-means

总体步骤

1. 初始化 cluster centroids
2. 循环：

   Step 1 Cluster assignment（根据“距离”染色）

   Step 2 Move centroid（根据所分配样本的均值移动簇中心）

- What if no points assigned to a cluster centroid?

  Eliminate it. (Most common case)

  or: Reinitialize it.

优化目标（代价函数）

$$J\left(c^{(1)},\dots,c^{(m)},\mu_1,\dots,\mu_K\right)=\frac{1}{m}\sum_{i=1}^{m}\left\|x^{(i)}-\mu_{c^{(i)}}\right\|^2$$

其中$c^{(1)},\dots,c^{(m)}$为 m 个样本各自的对应的簇的 index；$\mu_1,\dots,\mu_K$为 K 个簇的 positon.

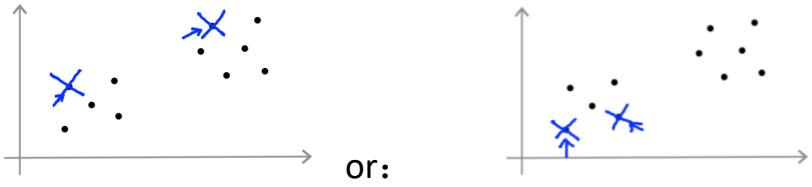
- Also called “Distortion cost function”（“损失代价函数”）

随机初始化 Random Initialization

步骤：

Step 1 根据簇的个数 K，从 m 个样本中随机抽取 K 个

Step 2 将该 K 个样本分配给初始的 K 个 cluster centroid：$\mu_1, \ldots, \mu_K$



- 如何避免坏的局部最优解？

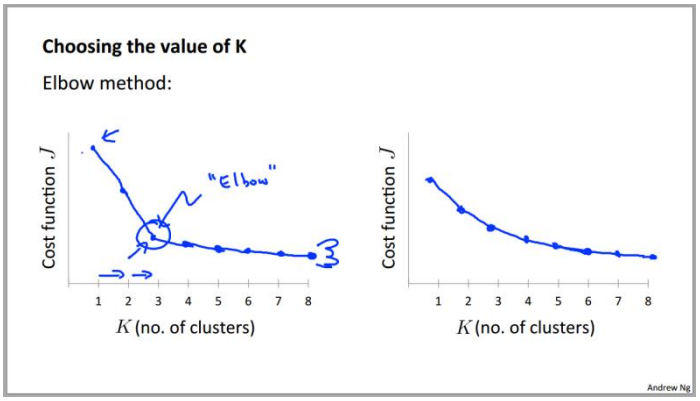
  多次初始化（即多次 K-means），取其最小。

  具体地，运行 i 次 K-means，i 一般取 50-1000。

  注：K=2~10 时有效。K 太大时不需要多次初始化。

簇的数量(K)的选取

1. 一般地：手动选取
2. Elbow method 肘部法则



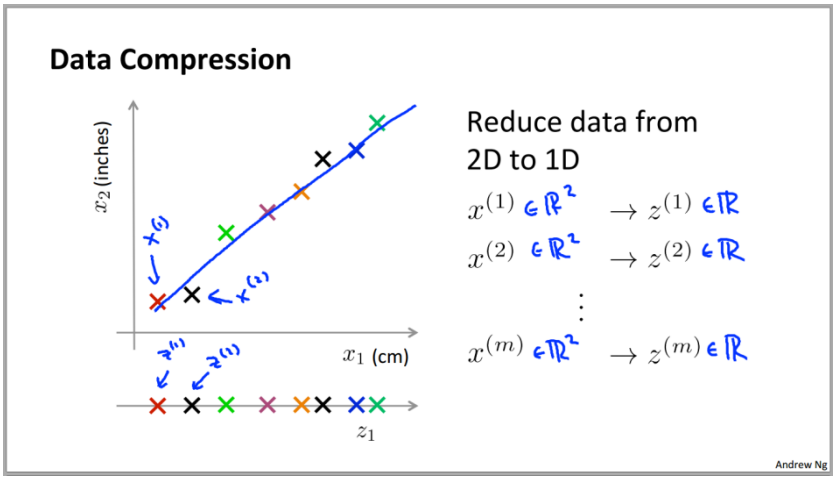
* 一般 K 越大 J 越小，如果反而增大，是由于卡在了坏的局部最优解上。

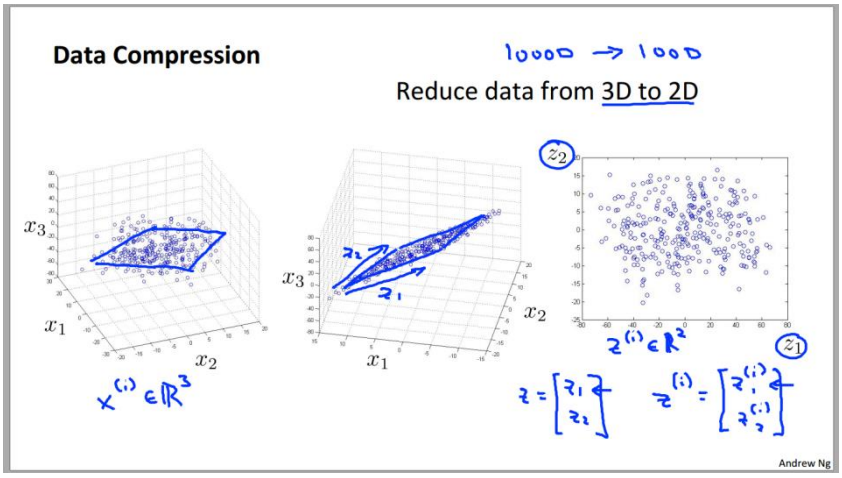
3. 根据 later purpose 的表现而定

Dimensionality Reduction 维数约简

Motivation 1：数据压缩

作用：节省内存、存储空间；提高算法运行速度





Motivation 2：数据可视化

Data Visualization

Country	x_1 GDP (trillions of US$)	x_2 Per capita GDP (thousands of intl. $)	x_3 Human Develop-ment Index	x_4 Life expectancy	x_5 Poverty Index (Gini as percentage)	x_6 Mean household income (thousands of US$)	...
Canada	1.577	39.17	0.908	80.7	32.6	67.293	...
China	5.878	7.54	0.687	73	46.9	10.22	...
India	1.632	3.41	0.547	64.7	36.8	0.735	...
Russia	1.48	19.84	0.755	65.5	39.9	0.72	...
Singapore	0.223	56.69	0.866	80	42.5	67.1	...
USA	14.527	46.86	0.91	78.3	40.8	84.3	...
...	...	...	...	...	...	...	...

[resources from en.wikipedia.org]

Andrew Ng

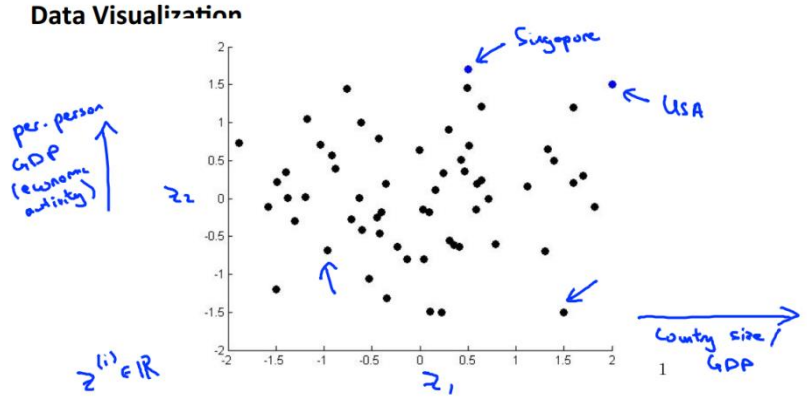
Data Visualization

Country	z_1	z_2
Canada	1.6	1.2
China	1.7	0.3
India	1.6	0.2
Russia	1.4	0.5
Singapore	0.5	1.7
USA	2	1.5
...	...	...

Reduce data from 50D to 2D

Andrew Ng

Data Visualization



Andrew Ng

Principle Component Analysis (PCA)
主成分分析

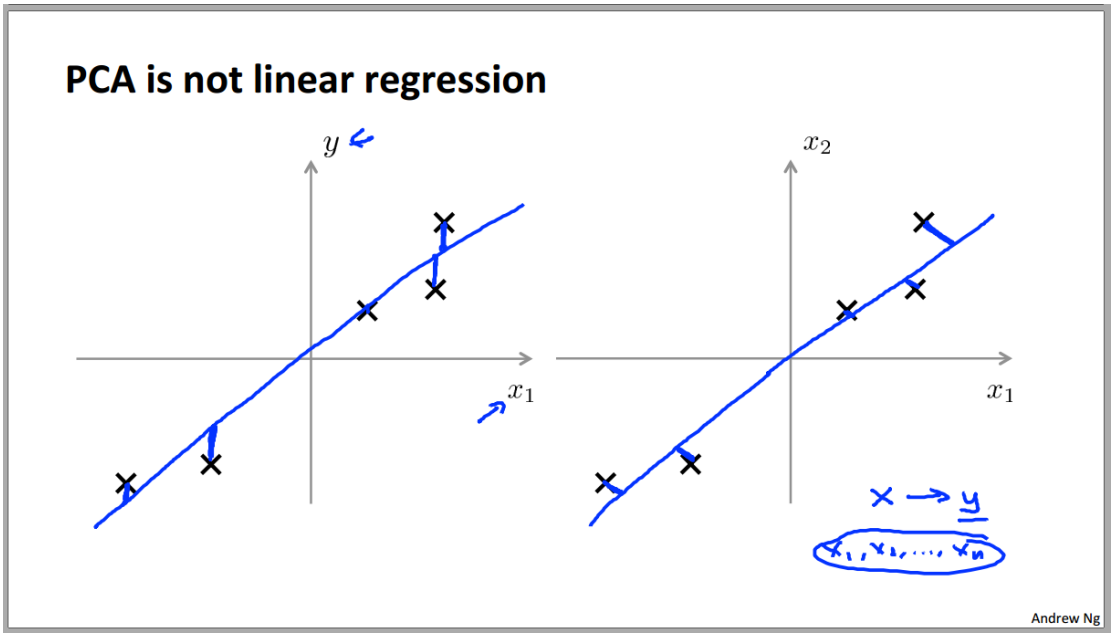
原理

寻找一个低维“平面”来对数据进行投影，使得投影误差（平均平方映射误差 Average Squared Projection Error）最小。

也即：

从 n 维降至 k 维：寻找一组向量$u^{(1)}, u^{(2)}, \ldots, u^{(k)}$（其中各个 u 都是 n 维的），将数据投影到这 k 个向量展开的低维线性子空间上。

- 主成分分析不是线性回归！



步骤

Step 1 数据预处理：mean normalization; feature scaling (optinal)

Step 2 计算 feature 的协方差矩阵 sigma

Step 3 计算协方差矩阵的特征向量（构成的矩阵 U），取前 k 个得到约简后的矩阵U_{reduce}，即各个投影方向向量（方法：svd/eig）

Step 4 与原 feature 相乘

$$z = {U_{reduce}}^{T} * x$$

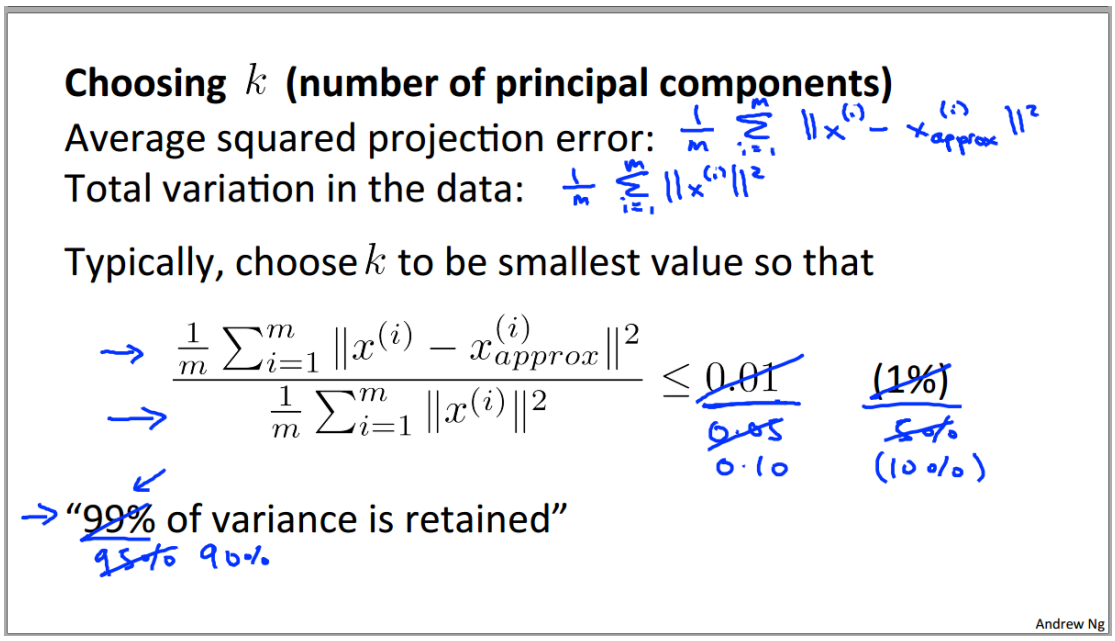
Reconstruction 原始数据重构

$$x_{approx} = U_{reduce} * z$$

Choosing "k"

从 1 开始尝试不同 k 值，使得"平均平方映射误差"与"总变差"之商小于某值（0.01, 0.05 等）

(Choose k by % of variance retained)



实际应用时，借助 svd 函数中返回的 S 矩阵来求。

应用建议

1. 用于加快监督学习时，仅仅在 training set 上做，不在 cross valiation set 或 test set 上做。
2. 不应该用于：避免过拟合。过拟合就只用正则化来解决，PCA 仅用于节省空间、加快速度。