Principal component analysis applications

Oliver W. Layton

CS252: Mathematical Data Analysis and Visualization

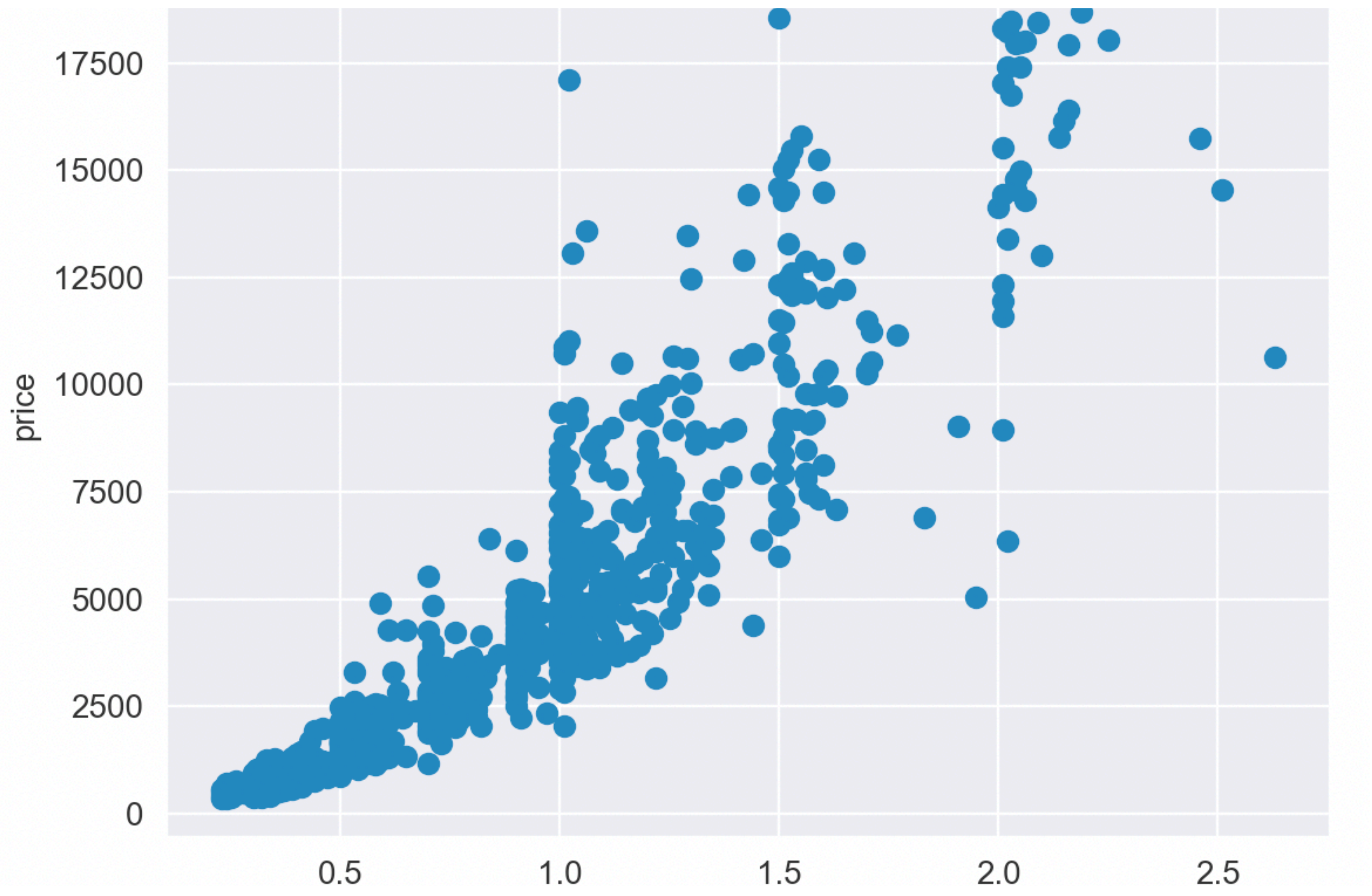
Lecture 20, Spring 2021

Monday March 29

Reconstruction of data from top principal components

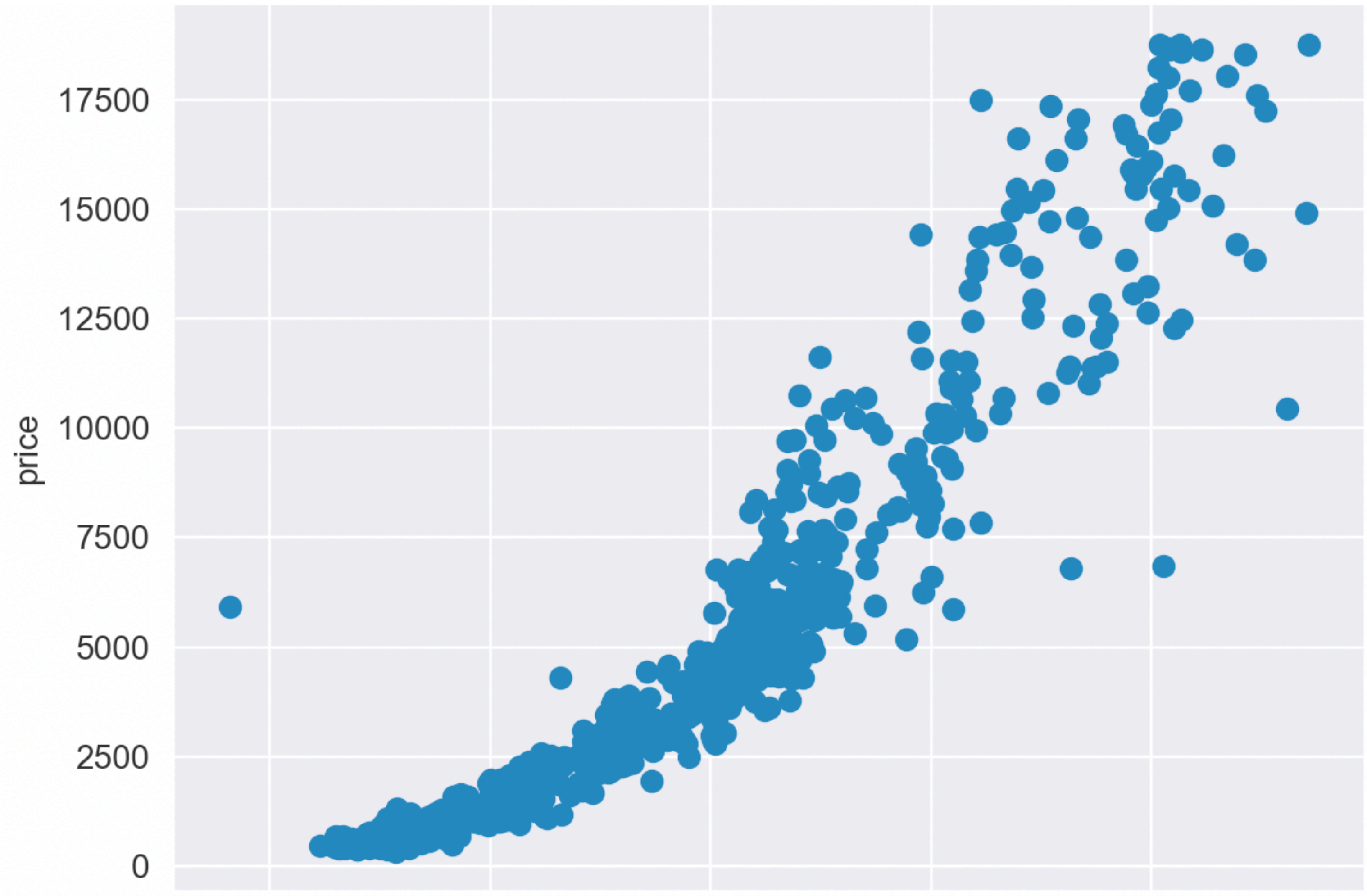
- PCA space useful for analysis, but sometimes we want to ask the question: "If I represent the data according to the top k eigenvectors, what would the data look like in their native space?"
- What might happen in the reconstruction process if we toss out eigenvectors?
  - This reconstruction process is **lossy** because we toss out information by dropping $M - k$ out of M eigenvectors.

Original diamond data



3 variables selected (2 shown): price, carat, Z (depth measurement)
(Not shown): Carat highly correlated with Z (doesn't account for much extra variance)

Reconstructed diamond data from top 2 PCs



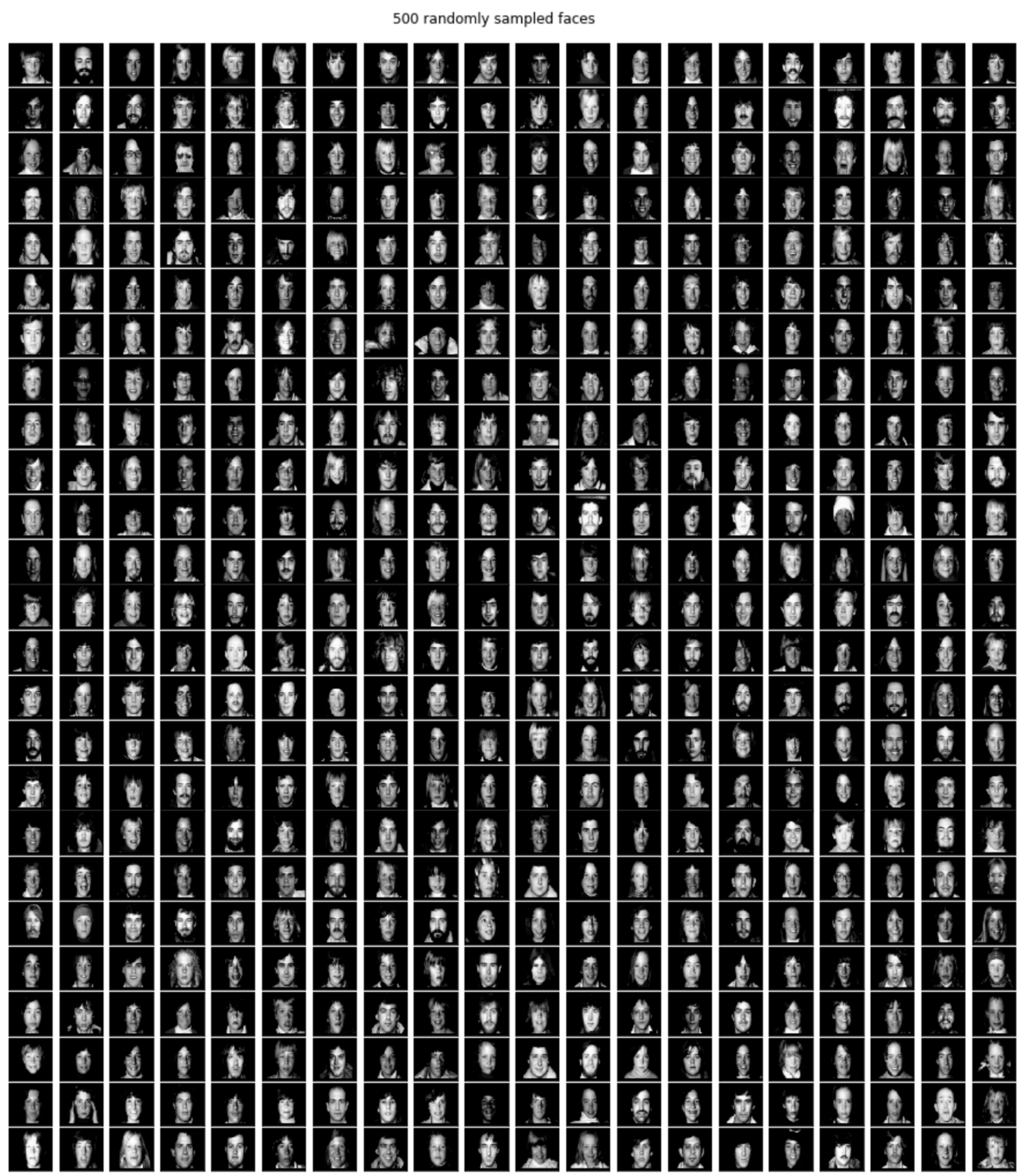
3 variables selected (2 shown): price, carat, Z (depth measurement

(Not shown): Carat highly correlated with Z (doesn't account for much extra variance)

Eigenfaces and facial recognition

- Project dataset: Celebrity Face images (LFWcrop)
- Paper on Eigenfaces: Turk & Pentland (1991).

Random sample of 500 faces



Eigenface algorithm: PCA on face images

1. Load in grayscale images, all the with same width and height: $I_1, I_2, \ldots, I_N$.
2. Collapse each 2D image into 1D vectors $\vec{x}_i$ (e.g. 16x16 2D image $\Rightarrow$ 256 1D vector). So, number of samples N = number of images. Variables are each of the pixels (e.g. if length($\vec{x}_i$) is 256. $M = 256$). Like usual, $A = [\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_M]$ (*rows: images/samples, cols: 1D pixel value variables*)
3. Center the images (subtract grand mean image): $A_c = A - \vec{\mu}$, where $\vec{\mu}$ is the column means of A (i.e. the mean pixel value at the same position across all images in the dataset).
4. Compute covariance matrix Σ then recover eigenvalues and eigenvectors.
5. Project images onto top k of principal components.

Grand mean of 500 faces ($\vec{\mu}$)

Because $\vec{\mu}$ is 1D vector, I had to **reshape** it into a 2D image format (e.g. 256 1D vector -> 16x16 2D image)

Variance explained by top eigenvalues/PCs