# Non-negative matrix factorization

- Lee & Seung (1999)
- like principal components (SVD), but data and components are assumed to be non-negative
- Model

$$\mathbf{X} \approx \mathbf{WH}$$

  where $X$ is $n \times p$, $W$ is $n \times r$, $H$ is $r \times p$, $r \leq p$.
- we assume $X_{ij}, W_{ij}, H_{ij} \geq 0$.
- criterion: minimize

$$L(\mathbf{W}, \mathbf{H}) = \sum_i \sum_u [X_{iu} \log(WH)_{iu} - (WH)_{iu}]$$

  This is the log-likelihood for the model $X_{iu} \sim \text{Poisson}(WH)_{iu}$.

The following alternating algorithm (Lee & Seung 2001) converges to a local maximum of $L(\mathbf{W}, \mathbf{H})$:

$$
\begin{aligned}
w_{ik} &\leftarrow w_{ik} \frac{\sum_{j=1}^{p} h_{kj} x_{ij} / (\mathbf{WH})_{ij}}{\sum_{j=1}^{p} h_{kj}} \\
h_{kj} &\leftarrow h_{kj} \frac{\sum_{i=1}^{N} w_{ik} x_{ij} / (\mathbf{WH})_{ij}}{\sum_{i=1}^{N} w_{ik}}
\end{aligned}
\tag{1}
$$

Ccan be viewed as an instance of the MM algorithm (see text) and iterative proportional scaling for log-linear models.

# Example

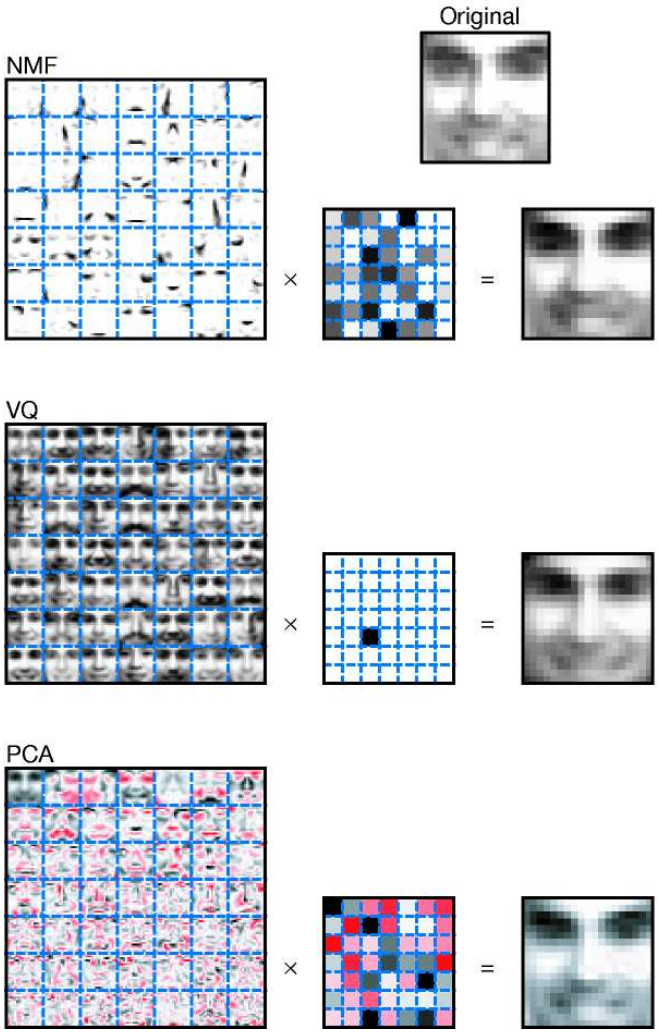


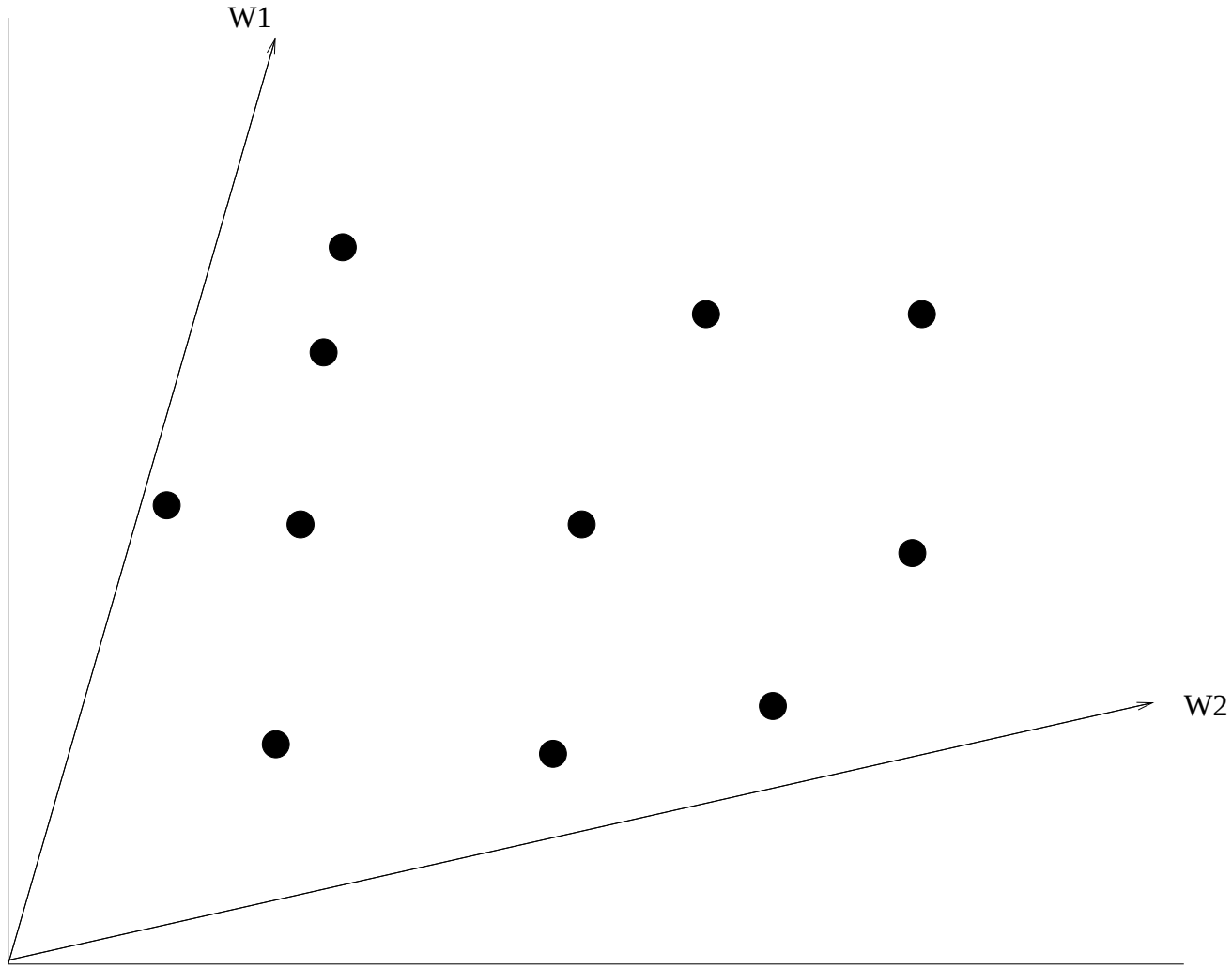
Figure 1 Non-negative matrix factorization (NMF) learns a parts-based representation of faces, whereas vector quantization (VQ) and principal components analysis (PCA) learn holistic representations. The three learning methods were applied to a database of $m = 2,429$ facial images, each consisting of $n = 19 \times 19$ pixels, and constituting an $n \times m$ matrix $V$. All three find approximate factorizations of the form $X = WH$, but with three different types of constraints on W and H, as described more fully in the main text and methods. As shown in the $7 \times 7$ montages, each method has learned a set of $r = 49$ basis images. Positive values are illustrated with black pixels and negative values with red pixels. A particular instance of a face, shown at top right, is approximately represented by a linear superposition of basis images. The coefficients of the linear superposition are shown next to each montage, in a $7 \times 7$ grid, and the resulting superpositions are shown on the other side of the equality sign. Unlike VQ and PCA, NMF learns to represent faces with a set of basis images resembling parts of faces.

**Big problem!**

See Donoho and Stodden (2004): "When does non-negative matrix factorization give a correct decomposition into parts?" Advances in Neural Information Processing Systems, 17, 2004

- columns of $W$ are not required to be orthogonal, as in principal components
- solution is not unique (even when $X = WH$ holds exactly): can choose for columns of $W$ any vectors in gap between axes and the data
- this limits its utility in practice

# Example



## Archetypal Analysis

- This method, due to Cutler & Breiman (1994), approximates data points by prototypes that are themselves linear combinations of data points. In this sense it has a similar flavor to $K$-means clustering.
- However, rather than approximating each data point by a single nearby prototype, archetypal analysis approximates each data point by a convex combination of a collection of prototypes. The use of a convex combination forces the prototypes to lie on the convex hull of the data cloud. In this sense, the prototypes are "pure,", or "archetypal."

# Archetypal Analysis- ctd

- The $N \times p$ data matrix $\mathbf{X}$ is modeled as

$$\mathbf{X} \approx \mathbf{WH} \tag{2}$$

  where $\mathbf{W}$ is $N \times r$ and $\mathbf{H}$ is $r \times p$.

- We assume that $w_{ik} \geq 0$ and $\sum_{k=1}^{r} w_{ik} = 1 \; \forall i$. Hence the $N$ data points (rows of $\mathbf{X}$) in $p$-dimensional space are represented by convex combinations of the $r$ archetypes (rows of $\mathbf{H}$).

- We also assume that

$$\mathbf{H} = \mathbf{BX} \tag{3}$$

  where $\mathbf{B}$ is $r \times N$ with $b_{ki} \geq 0$ and $\sum_{i=1}^{N} b_{ki} = 1 \; \forall k$.

- Thus the archetypes themselves are convex combinations of the data points.

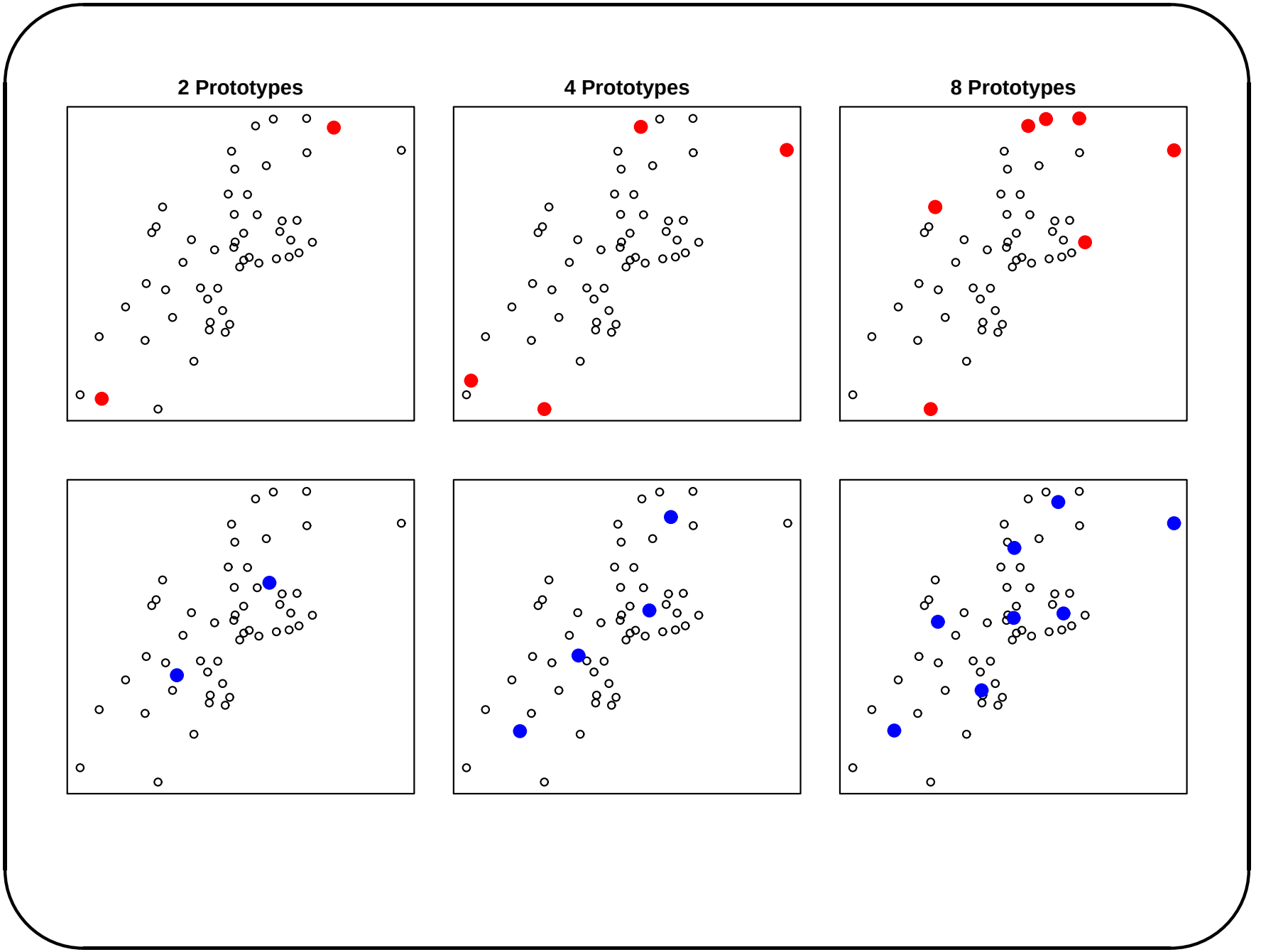
- Using both (2) and (3) we minimize

$$\begin{aligned} J(\mathbf{W}, \mathbf{B}) &= ||\mathbf{X} - \mathbf{WH}||^2 \\ &= ||\mathbf{X} - \mathbf{WBX}||^2 \end{aligned} \tag{4}$$

  over the weights $\mathbf{W}$ and $\mathbf{B}$.

- This function is minimized in an alternating fashion, with each separate minimization involving a convex optimization. The overall problem is not convex however, and so the algorithm converges to a local minimum of the criterion.

The next Figure shows an example with simulated data in two dimensions. The top panel displays the results of archetypal analysis, while the bottom panel shows the results from $K$-means clustering. In order to best reconstruct the data from convex combinations of the prototypes, it pays to locate the prototypes on the convex hull of the data. This is seen in the top panels of the Figure and is the case in general, as proven by Cutler & Breiman (1994). $K$-means clustering, shown in the bottom panels, chooses prototypes in the middle of the data cloud.


2 Prototypes
4 Prototypes
8 Prototypes

*Archetypal analysis (top panels) and $K$-means clustering (bottom panels) applied to* 50 *data points drawn from a bivariate Gaussian distribution. The colored points show the positions of the prototypes in each case.*

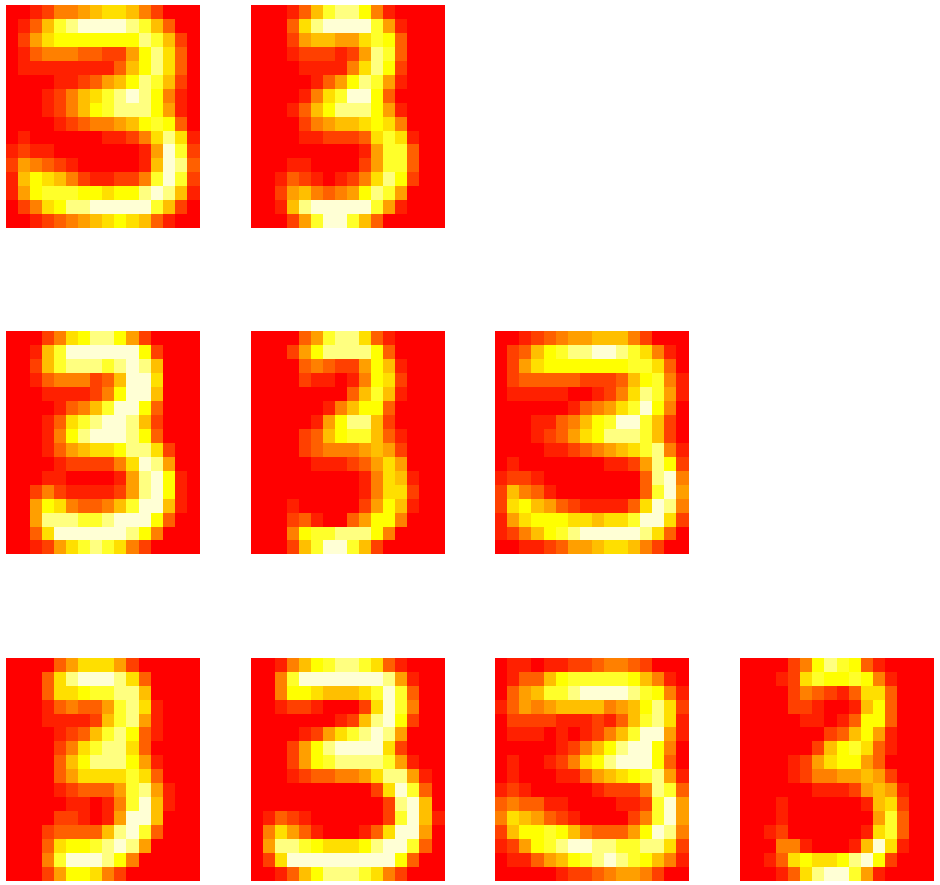
## Relation to K-means clustering and NNMF

- We can think of $K$-means clustering as a special case of the archetypal model, in which each row of $\mathbf{W}$ has a single one and and the rest of the entries are zero.
- Notice also that the archetypal model (2) has the same general form as the non-negative matrix factorization model (**??**). However, the two models are applied in different settings, and have somewhat different goals. Non-negative matrix factorization aims to approximate the columns of the data matrix $\mathbf{X}$, and the main output of interest are the columns of $\mathbf{W}$ representing the primary non-negative components in the data.

## Relation to K-means clustering and NNMF- ctd

- Archetypal analysis focuses instead on the approximation of the rows of $\mathbf{X}$ using the rows of $\mathbf{H}$, which represent the archetypal data points.
- Non-negative matrix factorization also assumes that $r \leq p$. With $r = p$, we can get an exact reconstruction simply choosing $\mathbf{W}$ to be the data $\mathbf{X}$ with columns scaled so that they sum to 1. In contrast, archetypal analysis requires $r \leq N$, but allows $r > p$.

The next Figure shows the results of archetypal analysis applied to the database of 3's discussed earlier. The three rows in the Figure are the resulting archetypes from three runs, specifying two, three and four archetypes, respectively. As expected, the algorithm has produced extreme 3's both in size and shape.



*Archetypal analysis applied to the database of digitized 3's. The rows in the figure show the resulting archetypes from three runs, specifying two, three and four archetypes, respectively.*

# References

Cutler, A. & Breiman, L. (1994), 'Archetypal analysis', Technometrics **36**(4), 338–347.

Lee, D. D. & Seung, H. S. (1999), 'Learning the parts of objects by non-negative matrix factorization', Nature **401**, 788.

Lee, D. D. & Seung, H. S. (2001), Algorithms for non-negative matrix factorization, in 'Advances in Neural Information Processing Systems, (NIPS*2001)'.