Why μ^p in Fuzzy Clustering?

Kehinde Akinola, Ahnaf Farhan, and
Vladik Kreinovich

University of Texas at El Paso
El Paso, TX 79968, USA

kaakinola@miners.utep.edu,
afarhan@miners.utep.edu,
vladik@utep.edu

1. Formulation of the Problem

- One of the main algorithms for clustering n given d-dimensional points:
  - selects K "typical" values c_k and
  - selects assignments $k(i)$ for each i from 1 to n
  - so as to minimize the sum

$$\sum_i (x_i - c_{k(i)})^2.$$

- This minimization is usually done iteratively.
- First, we pick c_k and assign each point x_i to the cluster k whose representative c_k is the closest to x_i.
- Then, we freeze $k(i)$ and select new typical representatives c_k by minimizing the objective function.
- This leads to c_k being an average of all the points x_i assigned to the k-th cluster.

2. Formulation of the Problem (cont-d)

- Then, the procedure repeats again and again – until the process converges.
- In practice, for some objects, we cannot definitely assign them to a single cluster.
- In such cases, it is reasonable to assign, to each object i,
  - degrees μ_{ik} of belongs to different clusters k,
  - so that $\sum\limits_k \mu_{ik} = 1$.
- In this case, it seems reasonable to take each term $(x_i - c_k)^2$ with the weight μ_{ik}.
- In other words, it seems reasonable to find the values μ_{ik} and c_k by minimizing the expression

$$\sum_{i,k} \mu_{ik} \cdot (x_i - c_k)^2.$$

3. Formulation of the Problem (cont-d)

- It seems reasonable to minimize
$$\sum_{i,k} \mu_{ik} \cdot (x_i - c_k)^2.$$
- However, this expression is linear in μ_{ik}.
- It is known that the minimum of a linear function under linear constraints is always at a vertex.
- Thus, the minimum is attained when one value μ_{ik} is 1 and the rest are 0s.
- We want to come up with truly *fuzzy* clustering, with $0 < \mu_{ik} < 1$ for some i and k.
- Thus, we need to replace the factor μ_{ik} with a non-linear expression $f(\mu_{ik})$.
- Then, we minimize the expression $\sum_{i,k} f(\mu_{ik}) \cdot (x_i - c_k)^2$.

4. Formulation of the Problem (cont-d)

- We minimize the expression

$$\sum_{i,k} f(\mu_{ik}) \cdot (x_i - c_k)^2.$$

- In practice, the functions $f(\mu) = \mu^p$ works the best.
- Why?

5. Our Explanation

- The weights μ_{ik} are normalized so that their sum is 1.
- So, if we delete some clusters or add more clusters, we need to re-normalize these values.
- A usual way to do it is to multiply them by a normalization constant c.
- It is therefore reasonable to require that:
  - the relative quality of different clustering ideas
  - not change is we simply re-scale.
- This implies, e.g., that:
  - if $f(\mu_1) \cdot v_1 = f(\mu_2) \cdot v_2$,
  - then after re-scaling $\mu_i \to c \cdot \mu_i$, we should have $f(c \cdot \mu_1) \cdot v_1 = f(c \cdot \mu_2) \cdot v_2$.
- We show that this condition implies that $f(\mu) = \mu^p$.

6. Our Explanation (cont-d)

- Indeed, $\dfrac{f(c\cdot\mu_2)}{f(c\cdot\mu_1)}=\dfrac{v_1}{v_2}=\dfrac{f(\mu_2)}{f(\mu_1)}$.
- Thus $r\stackrel{\text{def}}{=}\dfrac{f(c\cdot\mu_1)}{f(\mu_1)}=\dfrac{f(c\cdot\mu_2)}{f(\mu_2)}$ for all μ_1 and μ_2.
- So, the ratio r does not depend on μ: $r=r(c)$, and

$$f(c\cdot\mu)=r(c)\cdot f(\mu).$$

- It is known that the only continuous solutions of this functional equations are $f(\mu)=C\cdot\mu^p$.
- Minimization is not affected if we divide the objective function by C and get $f(\mu)=\mu^p$.

7. Proof

- We can solve the equation $f(c \cdot \mu) = r(c) \cdot f(\mu)$ when $f(\mu)$ is differentiable.
- Indeed, since $f(\mu)$ is differentiable, the ratio $r(c) = \dfrac{f(c \cdot \mu)}{f(\mu)}$ is also differentiable.
- If we differentiate both sides of the equation with respect to c, we get $\mu \cdot f'(c \cdot \mu) = r'(c) \cdot f(\mu)$.
- For $c = 1$, we get $\mu \cdot \dfrac{df}{d\mu} = p \cdot f$, where $p \stackrel{\text{def}}{=} r'(1)$.
- If we move all the terms containing f to one side and all others to another, we get $\dfrac{df}{f} = p \cdot \dfrac{d\mu}{\mu}$.
- Integrating, we get $\ln(f) = p \cdot \ln(\mu) + c_1$.
- If we apply exp to both sides, we get $f(\mu) = C \cdot \mu^p$, where $C = \exp(c_1)$.