C-MEANS AND GUSTAFSON-KESSEL ALGORITHMES

تقدمة الطلاب:
عادل كبول – عبد الرحمن القطيفاني – أحمد السلطان

المشرفين:

د. ميساء أبو قاسم

م. عليا حموي

م. باسل المدني

Contents

خوارزمية الFuzzy c-Means (FCM):

هي عبارة عن خوارزمية عنقدة Clustering تختلف عن خوارزمية الk-means بوجود قيم انتماء Membership بين كل عينة و عنقود تعبر عن مدى انتماء كل عينة لكل عنقود.

المتغيرات الأساسية:

الرموز ضمن الخوارزمية	أسماء المتغيرات ضمن الكود	التوصيف
i	n_clusters	عدد العناقيد
z	X	العينات او المعطيات المدروسة
v	cluster_means	مراكز العناقيد
D	distances	المسافة بين العينات و العناقيد
μ	membership	مصفوفة انتماء كل عينة الى كل عنقود
L	max_iter	عدد مرات تكرار الخوارزمية
m	fuzziness	الأس الترجيحي الخاص بالمنطق الضبابي

خطوات الخوارزمية:

الخوارزمية تتألف من مجموعة من الخطوات, يتم تكرارها عدد من المرات max_iter, يتم تحديده من قبل المستخدم, بالإضافة الى تحديد المستخدم لعدد العناقيد n_clusters الذي سيتم استخدامه ضمن عملية العنقدة.

يتم بداية تنفيذ هذه الخطوات ضمن الكود ضمن التابع:

def fit(self, X)

الخطوة الأولى:

يتم حساب مراكز العناقيد اعتمادا على مصفوفة الانتماء, و لكن في بداية الخوارزمية لا يوجد مصفوفة انتماء, لذلك نفرض مراكز العناقيد بشكل عشوائي بحيث يتم اختيار هذه المراكز اعتمادا على قيم من العينات المدروسة, كما يمكن اختيار طرق أخرى بحيث يتم اختيار القيم العشوائية أبعد ما يمكن عن بعضها, و يتم حساب المراكز اعتمادا على العلاقة التالية :

$$\mathbf{v}_i^{(l)} = \frac{\sum_{k=1}^{N} (\mu_{ik}^{(l-1)})^m \mathbf{z}_k}{\sum_{k=1}^{N} (\mu_{ik}^{(l-1)})^m}, \quad 1 \le i \le c.$$

حيث من أجل كل عنقود k و اعتمادا على مصفوفة الانتماء لكل العينات يتم حساب المركز الجديد للعنقود.

و لكن برمجيا هذا الأمر مكلف, لذلك قمنا باستخدام الجداء السلمي بين مصفوفة جميع العينات و مصفوفة الانتماءات بحيث قمنا باختصار العملية السابقة بعملية جداء واحدة ضمن التابع:

def __compute_new_clusters(self, X, membership)

و يتم انشاء المراكز العشوائية ضمن التابع:

def __initialize_means(self, X, row_count)

الخطوة الثانية:

يتم حساب المسافات بين مراكز العناقيد و جميع العينات ضمن العلاقة:

$$D_{ik\mathbf{A}}^2 = (\mathbf{z}_k - \mathbf{v}_i^{(l)})^T \mathbf{A} (\mathbf{z}_k - \mathbf{v}_i^{(l)}), \quad 1 \le i \le c, \quad 1 \le k \le N.$$

حيث من أجل كل عنقود i و عينة k, يتم حساب المسافة الاقليدية, و لكن المعطيات عادة مكونة من أكثر من بعد, و لتوفير الوقت يتم الاستفادة من الجداء السلمي لمصفوفة أبعاد العنقود و مصفوفة أبعاد العينة.

و لكن برمجيا, المرور على كل عينة و عنقود مكلف جدا خاصة عند وجود عدد عينات كبير, لذلك قمنا باستخدام الجداء السلمي بين مصفوفة جميع العناقيد و مصفوفة جميع العينات, بحيث نحصل على مصفوفة تحوي المسافة الاقليدية بين كل عنقود و عنصر, و يتم ذلك ضمن التابع:

def __compute_distances(self, X, cluster, memberships)

(ملاحظة: هذا التابع أيضا يقوم بحساب المسافة الخاصة بخوارزمية Gustafson-Kessel, لذلك نلاحظ أنه يوجد عملية حساب خاصة حسب الخوارزمية المستخدمة)

الخطوة الثالثة:

حساب مصفوفة الانتماء بين العناقيد و العينات:

for $1 \leq k \leq N$

if $D_{ik\mathbf{A}} > 0$ for $1 \leq i \leq c$

$$\mu_{ik}^{(l)} = \frac{1}{\sum_{j=1}^{c} (D_{ik\mathbf{A}} / D_{jk\mathbf{A}})^{2/(m-1)}},$$

otherwise

$$\mu_{ik}^{(l)} = 0 \text{ if } D_{ik\mathbf{A}} > 0, \text{ and } \mu_{ik}^{(l)} \in [0,1] \text{ with } \sum_{i=1}^{c} \mu_{ik}^{(l)} = 1 .$$

حيث من أجل كل نقطة k و كل عنقود i اعتمادا على مصفوفة المسافات, يتم تعديل قيم الانتماء لكل عينة, و في حال كانت المسافة مع العنقود أصغرية أي تساوي الصفر, تكون قيمة انتماء العينة لذلك العنقود تساوي ال1.

و مجددا برمجيا الأمر مكلف, لذلك نقوم بالاستفادة من المصفوفات للقيام بالعملية السابقة على كل العينات دفعة واحدة و معالجة حالة القيمة المسافة الأصغرية لاحقا, و يتم ذلك ضمن التابع:

def __compute_membership(self, X, distances)

و يتم تكرار الخطوات بعدد الmax_iterations الذي قام المستخدم بتحديده, أو حتى يصبح تغير مراكز العناقيد أصغري أي أصغر من قيمة ε التي تكون قيمة صغيرة جدا أو ممكن تحديدها من قبل المستخدم ,و في نهاية الخوارزمية, نكون قد حصلنا على مصفوفة الانتماءات memberships و مراكز العناقيد clusters_centers, بحيث يتم الاستفادة منها في تمثيل عملية العنقدة.

خوارزمية Gustafson-Kessel (GK):

الى حد كبير تشبه هذه الخوارزمية خوارزمية ال c-Means, الفرق الأساسي أن عملية حساب المسافة لا يتم بمجرد حساب المسافة الاقليدية, و انما يدخل في عملية حساب المسافة ما يسمى بالتغاير الضبابي fuzzy covariance, حيث يكون شكل العنقود في حالة ال c-Means دائري, بينما في خوارزمية Gustafson يتناسب شكل العنقود مع شكل توزع العينات.

خطوات الخوارزمية:

اذا تتم الخوارزمية بنفس الخطوات السابقة و لكن بإضافة خطوة حساب الCovariance :

الخطوة الأولى:

حساب مراكز العناقيد بنفس الطريقة السابقة و بنفس الدالة ضمن الكود.

الخطوة الثانية:

حساب مصفوفة الcovariance لكل عنقود كما في العلاقة:

$$\mathbf{F}_i = \frac{\sum_{k=1}^{N} (\mu_{ik}^{(l-1)})^m (\mathbf{z}_k - \mathbf{v}_i^{(l)})(\mathbf{z}_k - \mathbf{v}_i^{(l)})^T}{\sum_{k=1}^{N} (\mu_{ik}^{(l-1)})^m}, \quad 1 \le i \le c\,.$$

حيث من أجل كل عنقود i, و بالاستفادة من مصفوفة الانتماء μ, و المسافة بين العنقود و الجميع العينات, يتم إيجاد التغاير لكل عنقود.

و كما في السابق تم تطبيق طريقة برمجية لتسريع عملية الحساب, و يتم ذلك ضمن التابع:

def _calculate_fuzzyCov(self, X, membership, new_cluster_centers)

الخطوة الثالثة:

حساب المسافات بين العناقيد و جميع العينات كما في السابق و لكن بوجود مصفوفة التغاير الضبابي كما في العلاقة التالية:

$$D^2_{ik\mathbf{A}_i} = (\mathbf{z}_k - \mathbf{v}_i^{(l)})^T \left[(\rho_i \det(\mathbf{F}_i)^{1/n} \mathbf{F}_i^{-1} \right] (\mathbf{z}_k - \mathbf{v}_i^{(l)}),$$
$$1 \le i \le c, \quad 1 \le k \le N\,.$$

حيث F هي مصفوفة التغاير الضبابي لكل العناقيد, و من أجل العنقود i و العينة k يتم حساب المسافة بتطبيق العلاقة, و تم تسريع العملية ضمن الكود عن طريق الاستفادة من المصفوفات, و يتم حساب المسافة ضمن التابع:

def –compute_distances(self, X, clusters, memberships)

نلاحظ ضمن التابع المتغير FconvInv يعبر عن مصفوفة التغاير الضبابي, و كما ذكرنا في السابق, ضمن هذا التابع تم مناقشة حالتي حساب المسافة من أجل الخوارزميتين السابقتين.

الخطوة الرابعة:

حساب مصفوفة الانتماء كما في السابق.

ضمن الكود, الصف الخاص بهذه الخوارزمية موجود ضمن ملف c_means.py.

ضمن ملف main.py, يتم انشاء الصف و تحديد المتغيرات المطلوبة (عدد العناقيد, عدد مرات التكرار, الخوارزمية المستخدمة ... الخ), ثم عن طرق التابع fit يتم ارسال العينات المطلوب معالجتها, و الحصول على النتائج التالية:

- مصفوفة تحوي كل عنقود و مركزه ضمن الفضاء حسب فضاء البعد الخاص بالعينات.
- مصفوفة تحوي انتماءات كل عينة الى العناقيد المطلوبة, علما أنه و حسب الخوارزمية يكون الانتماء بشكل ضبابي fuzzy, أي يوجد نسبة لانتماء كل عينة الى كل عنقود محدد.