From Unsupervised Clustering Results to MLP Classification

- استفاده از نتایج Clustering برای آموزشMLP

مقاله مرجع (Clustering)

Performance Evaluation of K-Means and Hierarchical Clustering in Terms of Accuracy and Running Time

در این پروژه، هدف بررسی یک مسیر یادگیری ترکیبی است که از **Unsupervised Learning** آغاز می‌شود و در ادامه، خروجی آن برای آموزش یک **Multilayer Perceptron (MLP)** مورد استفاده قرار می‌گیرد.

در تمام مراحل پروژه، تحلیل‌ها و تصمیم‌گیری‌ها باید **صرفاً بر اساس مطالب، نتایج عددی و نتیجه‌گیری‌های مقاله مرجع Clustering (cluster.pdf)** انجام شود. پاسخ‌ها ماهیت **مفهومی** دارند و نیازی به استخراج یا محاسبه فرمول‌ها نیست.

در مقاله مرجع، عملکرد دو الگوریتم **K-Means** و **Hierarchical Clustering** روی دیتاست‌هایی مانند **Iris** و **Diabetes** مورد بررسی قرار گرفته است. اگرچه این دیتاست‌ها دارای Label واقعی هستند، اما در فرآیند Clustering مقاله، این Labelها **به‌صورت کامل نادیده گرفته شده‌اند** و تنها برای **ارزیابی نهایی Accuracy** مورد استفاده قرار گرفته‌اند. بنابراین، Clustering انجام‌شده در مقاله ماهیت **Unsupervised** دارد.

در ابتدای پروژه، دانشجو باید **یکی از دیتاست‌های معرفی‌شده در مقاله** (Iris یا Diabetes) را انتخاب کرده و همان دیتاست را در تمام مراحل پروژه دنبال نماید. سپس، باید **هر دو الگوریتم K-Means و**

Hierarchical Clustering را روی دیتاست انتخاب‌شده اجرا کند. هدف از اجرای هر دو الگوریتم، **مقایسه عملکرد آن‌ها** و تحلیل تفاوت نتایج است.

در این پروژه، دانشجو **ملزم به پیاده‌سازی دقیق، گام‌به‌گام یا ابزارمحور مقاله نیست**. با این حال، اجرای Clustering باید **با الهام از ایده، منطق کلی و چارچوب آزمایشی مقاله** انجام شود؛ به این معنا که:

- نوع دیتاست‌ها و مسئله مطابق مقاله انتخاب شود؛
- تعداد خوشه‌ها با ساختار مسئله هم‌خوان بوده و مشابه تنظیمات مقاله در نظر گرفته شود؛
- Labelهای واقعی کلاس در مرحله Clustering استفاده نشوند؛
- Accuracy صرفاً برای تحلیل و مقایسه با نتایج مقاله محاسبه شود.

هدف از این رویکرد آن است که نتایج به‌دست‌آمده توسط دانشجو از نظر **رفتار کلی الگوریتم‌ها** و **الگوی عملکرد**، قابل مقایسه با نتایج گزارش‌شده در **Table 2** مقاله باشد. تأکید می‌شود که یکسان بودن عددی Accuracy با مقاله مدنظر نیست؛ بلکه **تحلیل این موضوع اهمیت دارد که کدام الگوریتم، مطابق نتیجه‌گیری مقاله (Conclusion – Page 3)، برای دیتاست انتخاب‌شده مناسب‌تر است**.

پس از تحلیل نتایج Clustering و با تکیه بر نتیجه‌گیری مقاله، دانشجو باید **الگوریتم مناسب‌تر** را برای دیتاست انتخاب‌شده مشخص کند. خروجی این الگوریتم شامل شماره خوشه هر نمونه داده است. این شماره خوشه‌ها در ادامه پروژه به‌عنوان **Label جدید داده‌ها** در نظر گرفته می‌شوند که به آن‌ها **Pseudo-Label** گفته می‌شود.

در مرحله بعد، یک شبکه **Feedforward Multilayer Perceptron (MLP)** با استفاده از همین Pseudo-Labelها آموزش داده می‌شود. در این مرحله:

- ورودی شبکه، Featureهای اصلی دیتاست است؛
- خروجی شبکه، Pseudo-Labelهای تولیدشده توسط Clustering است.

در این پروژه، شبکه MLP وظیفه یادگیری Labelهای واقعی دیتاست را ندارد، بلکه هدف آن **یادگیری و مدل‌سازی ساختار خوشه‌بندی کشف‌شده در مرحله Unsupervised** است. بنابراین، آموزش MLP به‌عنوان **ادامه مفهومی فرآیند Clustering مقاله** تلقی می‌شود.

در نهایت، کل این فرآیند نمونه‌ای از **Clustering-based Label Propagation** محسوب می‌شود (نه Semi-Supervised به معنای دقیق کلمه، چون در هیچ مرحله‌ای از Label واقعی در آموزش استفاده نمی‌شود)؛ زیرا:

- کشف ساختار داده‌ها به‌صورت Unsupervised انجام شده است؛
- آموزش MLP به‌صورت Supervised صورت می‌گیرد، اما با Labelهایی که واقعی نیستند و به‌صورت Pseudo تولید شده‌اند.

هدف نهایی پروژه این است که دانشجو درک کند چگونه می‌توان، با تکیه بر تحلیل نتایج یک مقاله Unsupervised، از خروجی Clustering برای آموزش یک مدل Supervised مانند MLP استفاده کرد و این فرآیند را به‌صورت مفهومی تحلیل نمود.

مراحل پروژه

مرحله ۱ — انتخاب دیتاست و اجرای Clustering

۱-الف) یکی از دو دیتاست **Iris** یا **Diabetes** را که در مقاله مرجع بررسی شده‌اند انتخاب کنید. دیتاست انتخاب‌شده را در تمام مراحل پروژه ثابت نگه دارید.

۱-ب) هر دو الگوریتم **K-Means** و **Hierarchical Clustering** را روی دیتاست انتخاب‌شده اجرا کنید. در اجرای هر دو الگوریتم:

- تعداد خوشه‌ها باید با ساختار مسئله (تعداد کلاس‌های واقعی دیتاست) هم‌خوانی داشته باشد
- Label واقعی داده‌ها در مرحله Clustering استفاده **نشود**
- Accuracy نهایی Clustering را برای هر دو الگوریتم محاسبه و گزارش دهید

۱-ج) نتایج Accuracy به‌دست‌آمده را با **Table 2** مقاله مقایسه کنید. آیا الگوی رفتاری الگوریتم‌ها (کدام بهتر است) با نتایج مقاله همخوانی دارد؟ توضیح دهید.

۱-د) بر اساس **Conclusion** (**صفحه ۳**) مقاله، کدام الگوریتم برای دیتاست انتخاب‌شده مناسب‌تر است؟ دلیل بیاورید.

مرحله ۲ — تولید Pseudo Labels

۲-الف) الگوریتم برتری که در مرحله قبل مشخص کردید را انتخاب کنید. خروجی این الگوریتم (شماره خوشه هر نمونه) را به‌عنوان **Pseudo Label** در نظر بگیرید.

۲-ب) توضیح دهید که چرا به این Labelها «Pseudo» می‌گویند و چه تفاوتی با Label واقعی دیتاست دارند؟

مرحله ۳ — آموزش MLP با Pseudo Labels

در این مرحله یک شبکه Feedforward MLP (Multi-Layer Perceptron) **با استفاده از** Pseudo Label**های تولیدشده در مرحله قبل آموزش می‌دهید. ورودی شبکه** Featureهای اصلی دیتاست **و خروجی آن** Pseudo Label**ها هستند.**

۳-الف) آماده‌سازی داده‌ها (مقیاس‌بندی)

پیش از آموزش مدل MLP، لازم است که داده‌های ورودی (Featureها) را مقیاس‌بندی کنید. مدل‌های MLP به داده‌های مقیاس‌بندی شده حساس هستند و این کار به همگرایی بهتر و سریع‌تر مدل کمک می‌کند.

- **پیشنهاد:** از StandardScaler در کتابخانه scikit-learn برای استانداردسازی داده‌ها استفاده کنید. (یعنی میانگین Featureها صفر و واریانس آن‌ها یک شود).

۳-ب) طراحی و آموزش مدل MLP

یک مدل **MLP از نوع Feedforward** با استفاده از MLPClassifier در کتابخانه scikit-learn طراحی و آموزش دهید. در زمان تعریف مدل، به نکات زیر توجه کنید:

۱. **اندازه و تعداد لایه‌های پنهان (hidden_layer_sizes):**

- این پارامتر، تعداد لایه‌های میانی (پنهان) و تعداد نورون‌های هر لایه را مشخص می‌کند. این لایه‌ها وظیفه‌ی اصلی یادگیری الگوهای پیچیده در داده‌ها را بر عهده دارند.
- **پیشنهاد:** برای شروع، از ساختار hidden_layer_sizes=(۱۰, ۱۰) استفاده کنید. این یعنی دو لایه پنهان که هر کدام ۱۰ نورون دارند. این یک نقطه شروع مناسب برای این پروژه است.

۲. **تعداد Epochها (max_iter):**

- پارامتر max_iter (که همان تعداد Epochهاست) نشان می‌دهد که مدل چند بار کل مجموعه داده‌های آموزشی را ببیند و پارامترهای داخلی خود را به‌روزرسانی کند.
- **پیشنهاد:** مقدار max_iter=500 را تنظیم کنید. این تعداد معمولاً برای مدل‌های ساده MLP کافی است تا به یک همگرایی مناسب برسند و مدل بتواند Pseudo Labelها را یاد بگیرد.

۳. **تابع فعال‌ساز (Activation Function):**

- **لایه‌های داخلی (مخفی) شبکه:** مدل MLPClassifier به صورت پیش‌فرض از تابع فعال‌ساز relu برای لایه‌های داخلی استفاده می‌کند که یک انتخاب مناسب برای یادگیری الگوهاست. **نیازی به تغییر این تنظیم نیست.**
- **لایه خروجی شبکه:** در مورد لایه خروجی (که جواب نهایی را تولید می‌کند)، مدل MLPClassifier **به صورت هوشمند و خودکار** تابع فعال‌ساز مناسب (مثل Softmax برای مسائل چندکلاسه مانند Iris، یا Sigmoid برای مسائل دودویی مانند Diabetes) را **متناسب با تعداد دسته‌های دیتای شما** تنظیم می‌کند.
- **بنابراین، نیازی نیست شما هیچ تابعی برای لایه خروجی انتخاب کنید و تمرکزتان صرفاً بر استفاده از مدل باشد.**

مدل را با Pseudo Labelهای تولیدشده آموزش داده و عملکرد آن را روی داده‌های آزمایشی گزارش کنید.

۳-ج) تحلیل مفهومی خطای آموزش با افزایش Epoch

به صورت مفهومی توضیح دهید که به طور کلی با افزایش تعداد Epoch در آموزش یک شبکهMLP ، انتظار دارید چه اتفاقی برای خطای مدل بیفتد؟

۴) ارزیابی مدل

پس از آموزش، مدل را روی داده‌های Test ارزیابی کنید:

Accuracy

درصد نمونه‌هایی که MLP خوشه آن‌ها را درست پیش‌بینی کرده است را محاسبه و گزارش دهید. این عدد نشان می‌دهد مدل تا چه حد توانسته ساختار خوشه‌بندی را یاد بگیرد.

Confusion Matrix

دانشجو باید **Confusion Matrix مدل روی داده های تست** را محاسبه و نمایش دهد

سوالات تحلیلی

سوال ۱) عدد Accuracy به‌دست‌آمده از MLP را با Clustering Accuracy مرحله قبل مقایسه کنید. آیا این دو عدد باید به هم نزدیک باشند؟ چرا؟

سوال ۲) از روی Confusion Matrix تحلیل کنید که MLP در تشخیص کدام خوشه بیشترین خطا را داشته. این خطا با کیفیت Pseudo Labelهای آن خوشه ارتباط دارد؟