سوال ۱: تشخیص نوع گل با استفاده از الگوریتم KNN

شما در حال کار بر روی یک پروژه دسته‌بندی گل‌ها هستید و باید نوع گل را با استفاده از داده‌های مربوط به ویژگی‌های آن پیش‌بینی کنید. داده‌های شما مربوط به سه نوع گل است:

۱- Setosa ۲- Versicolor ۳- Virginica

هر گل دارای ۴ ویژگی است:

- طول کاسبرگ (Sepal Length)
- عرض کاسبرگ (Sepal Width)
- طول گلبرگ (Petal Length)
- عرض گلبرگ (Petal Width)

این داده‌ها در مجموعه داده معروف Iris موجود هستند که شامل ۱۵۰ نمونه است. هدف شما این است که با استفاده از الگوریتم K-Nearest Neighbors (KNN):

مرحله ۱: تقسیم داده‌ها

- ابتدا داده‌ها را به دو بخش آموزشی (Training) و آزمایشی (Testing) تقسیم کنید.
- از داده‌های آموزشی برای ساخت مدل استفاده کنید و داده‌های آزمایشی را برای ارزیابی مدل به کار ببرید.

مرحله ۲: ساخت مدل KNN

- یک مدل KNN ایجاد کنید و مقدار k (تعداد همسایه‌ها) را برابر با ۳ قرار دهید.

مرحله ۳: ارزیابی مدل

- دقت مدل را روی داده‌های آزمایشی محاسبه کنید و مشخص کنید که مدل چقدر دقیق عمل کرده است.

مرحله ۴: پیش‌بینی برای یک نمونه جدید

- با استفاده از مدل ساخته شده، نوع گلی که ویژگی‌های زیر دارد را پیش‌بینی کنید:
  - طول کاسبرگ = ۵.۵
  - عرض کاسبرگ = ۲.۴
  - طول گلبرگ = ۳.۸
  - عرض گلبرگ = ۱.۱

نکات کمکی:

- برای بارگذاری دیتاست Iris، می‌توانید از کتابخانه sklearn استفاده کنید.
- برای تقسیم‌بندی داده‌ها به بخش‌های آموزشی و آزمایشی از train_test_split استفاده کنید.
- برای ساخت مدل KNN از KNeighborsClassifier استفاده کنید.
- با استفاده از accuracy_score دقت مدل را محاسبه کنید.

سؤال دوم: استفاده از PCA برای پیش‌پردازش داده‌ها و دسته‌بندی نوع گل

در ادامه‌ی سؤال اول، هدف این است که پیش از استفاده از الگوریتم KNN برای دسته‌بندی نوع گل، از تکنیک **تحلیل مؤلفه‌های اصلی** (**PCA**) برای کاهش ابعاد داده‌ها استفاده کنید و بررسی کنید که آیا این کاهش ابعاد تأثیری بر دقت مدل دارد یا خیر.

مرحله اول: کاهش ابعاد با PCA

۱. داده‌های مربوط به ویژگی‌های گل‌ها را با استفاده از تکنیک **PCA** از ۴ ویژگی به ۲ مؤلفه اصلی کاهش دهید.

۲. مقدار **واریانس تجمعی توضیح داده شده** توسط این دو مؤلفه اصلی را محاسبه کنید و گزارش دهید.

مرحله دوم: دسته‌بندی با KNN

۱. از داده‌های کاهش‌یافته (۲ مؤلفه اصلی) برای آموزش مدل **KNN** استفاده کنید.

۲. داده‌ها را به دو بخش **آموزشی** (Training) و **آزمایشی** (Testing) تقسیم کنید.

۳. دقت مدل را روی داده‌های آزمایشی محاسبه کنید.

٤. نوع گلی که ویژگی‌های زیر را دارد پیش‌بینی کنید (پس از اعمال PCA):

- طول کاسبرگ = **۵.۵**
- عرض کاسبرگ = **۲.۴**
- طول گلبرگ = **۳.۸**
- عرض گلبرگ = **۱.۱**

سوال سوم: ساخت و مقایسه سیستم Rule-Based با KNN

۱. یک سیستم Rule-Based برای دسته‌بندی داده‌های **Iris** طراحی کنید. به عنوان مثال:
   - اگر طول گلبرگ کمتر از ۲ باشد، نوع گل **Setosa** است.
   - اگر طول گلبرگ بین ۲ تا ۵ باشد، نوع گل **Versicolor** است.
   - اگر طول گلبرگ بیشتر از ۵ باشد، نوع گل **Virginica** است.
۲. داده‌های آزمایشی را با این سیستم Rule-Based دسته‌بندی کنید و دقت آن را محاسبه کنید.
۳. مدل KNN را نیز با مقدار k=۳ اجرا کنید و دقت آن را محاسبه کنید.
٤. دقت دو روش را مقایسه کنید و تحلیل کنید:
   - کدام روش بهتر عمل کرده است؟
   - مزایا و معایب هر روش چیست؟
   - آیا Rule-Based در شرایط خاصی می‌تواند بهتر از KNN عمل کند؟

سؤال چهارم : استانداردسازی داده‌ها و تاثیر آن بر KNN

هدف: بررسی تأثیر استانداردسازی ویژگی‌ها روی دقت مدل KNN و نحوه رفتار داده‌ها قبل و بعد از استانداردسازی.

مرحله ۱: آماده‌سازی داده‌ها

۱. دیتاست Iris را بارگذاری کنید و آن را به دو بخش ویژگی‌ها (X) و برچسب‌ها (y) تقسیم کنید.
۲. داده‌ها را به دو بخش **آموزشی (Training)** و **آزمایشی (Testing)** تقسیم کنید (مثلاً ٪۷۰ آموزش و ٪۳۰ تست).

مرحله ۲: ساخت مدل KNN بدون استانداردسازی

۱. یک مدل KNN با k=3 بسازید و روی داده‌های اصلی (غیر استاندارد) آموزش دهید.
۲. دقت مدل را روی داده‌های تست محاسبه کنید.

مرحله ۳: استانداردسازی داده‌ها

۱. با استفاده از StandardScaler داده‌های آموزشی را استاندارد کنید (میانگین برابر صفر و واریانس برابر یک).
۲. داده‌های تست را نیز با همان مقیاس داده‌های آموزشی استاندارد کنید.
۳. دوباره مدل KNN با k=3 را روی داده‌های استاندارد شده آموزش دهید و دقت مدل را روی داده‌های تست محاسبه کنید.

مرحله ۴: مقایسه و تحلیل

۱. دقت مدل قبل و بعد از استانداردسازی را با هم مقایسه کنید.

۲. تفاوت در رفتار مدل KNN را توضیح دهید.

۳. تحلیل کنید چرا استانداردسازی روی الگوریتم KNN اهمیت دارد.

نکته کمکی:

- KNNاز فاصله بین نقاط داده برای پیش‌بینی استفاده می‌کند، بنابراین اگر مقیاس ویژگی‌ها متفاوت باشد، ویژگی‌های با مقیاس بزرگ‌تر وزن بیشتری پیدا می‌کنند.

سوال پنجم: بررسی تأثیر معیارهای مختلف فاصله (Distance Metrics) بر عملکرد الگوریتم KNN همراه با رسم نواحی تصمیم(Decision Boundary)

در این سؤال قصد داریم بررسی کنیم که الگوریتم KNN هنگام استفاده از **معیارهای مختلف فاصله (Distance Metrics)** چگونه عملکرد متفاوتی دارد. همچنین مشاهده خواهیم کرد که هر معیار فاصله چگونه **مرزبندی تصمیم (Decision Boundary)** متفاوتی ایجاد می‌کند.

فاصله‌هایی که باید تست شوند:

Euclidean Distance - Manhattan Distance- Chebyshev Distance - Cosine Distance - Mahalanobis Distance

مرحله ۱: بارگذاری و آماده‌سازی داده‌ها

دیتاست Iris را با استفاده از کتابخانه sklearn بارگذاری کنید.

داده‌ها را به دو بخش ویژگی‌ها (X) و برچسب‌ها (y) تقسیم کنید.

ویژگی‌ها را استانداردسازی کنید.

برای رسم نمودار دو بعدی، داده‌ها را با استفاده از PCA از ۴ بعد به ۲ مؤلفه اصلی کاهش دهید.

توجه:
داده‌های ۲بعدی فقط برای visualize استفاده می‌شود، اما مدل‌ها روی داده استانداردشده ۴بعدی آموزش داده شوند.

مرحله ۲: ساخت مدل‌هاKNN با فاصله‌های مختلف

یک مدل KNN با مقدار **k = 3** بسازید.
این کار را ۵ بار انجام دهید و هر بار از یکی از معیارهای فاصله را استفاده کنید:

مرحله ۳: ارزیابی و مقایسه مدل‌ها

برای هر یک از ۵ مدل:KNN

۱. دقت (Accuracy) آن را روی داده‌های تست محاسبه کنید

مرحله ۴: رسم نواحی تصمیم(Decision Boundary)

برای داده‌های **۲بعدی حاصل از PCA**:

- برای هر یک از فاصله‌ها یک نمودار مجزا رسم کنید.
- نواحی تصمیم (Decision Boundary) را با رنگ‌بندی نمایش دهید.
- نقاط داده واقعی را نیز روی نمودار رسم کنید.

در نهایت باید **۵ نمودار** تولید شود.

مرحله ۵: تحلیل نهایی

در یک بخش تحلیلی به پرسش‌های زیر پاسخ دهید:

۱. کدام معیار فاصله بهترین عملکرد را داشت؟

۲. چرا بعضی فاصله‌ها مرزبندی‌های نامناسب تولید کردند؟

۳. فاصله مالاهانوبیس در چه شرایطی برتری دارد؟

۴. فاصله کسینوسی چه زمانی بهتر است؟

۵. اگر داده‌ها Scale نشوند، کدام فاصله‌ها بیشترین آسیب را می‌بینند؟