

سوال ۱:

مراحل عملکرد کد به شرح زیر است:

1. ابتدا، کتابخانه‌های مورد نیاز را وارد می‌کنیم، از جمله pandas برای کار با داده‌های جدولی، sklearn برای استفاده از المانت‌های مربوط به ماشین بردار پشتیبان و معیارهای ارزیابی، و matplotlib برای رسم نمودارها.
2. سپس، داده‌ها از یک فایل CSV با استفاده از تابع 'read\_csv' در pandas خوانده می‌شوند. فایل CSV دو ستون دارد "label": که برچسب هم‌گونی یا هرزنامه را نشان می‌دهد و "message" که متن پیام‌ها را حاوی است.
3. ستون "message" به عنوان متغیر ورودی (X) و ستون "label" به عنوان متغیر خروجی (y) در نظر گرفته می‌شوند. برچسب‌های "ham" به مقدار 0 و برچسب‌های "spam" به مقدار 1 نگاشت می‌شوند.
4. سپس، بردار ویژگی TF-IDF بر روی متن پیام‌ها اعمال می‌شود. بردار ویژگی‌های TF-IDF اطلاعات مربوط به وزن و اهمیت کلمات در هر پیام را نشان می‌دهند.
5. یک مدل SVM با کرنل خطی ایجاد می‌شود SVM. یک الگوریتم یادگیری ماشینی است که برای دسته‌بندی داده‌ها استفاده می‌شود و با استفاده از بردارهای ویژگی، مرزهای تصمیم‌گیری بین دسته‌ها را تعیین می‌کند.
6. مدل روی بردار ویژگی‌های TF-IDF آموزش داده می‌شود تا بتواند پیش‌بینی‌های هم‌گونی/هرزنامه صحیحی را ارائه دهد.
7. با استفاده از مدل آموزش دیده، برچسب‌های پیش‌بینی شده برای همه نمونه‌ها محاسبه می‌شود.
8. سپس، دقت مدل با استفاده از معیار دقت (accuracy) محاسبه می‌شود. دقت نسبت تعداد پیش‌بینی‌های صحیح به کل نمونه‌ها را نشان می‌دهد.
9. در ادامه، منحنی مشخصه عملکرد گیرنده (Receiver Operating Characteristic - ROC) برای ارزیابی کارایی مدل با استفاده از شاخص‌های تشخیص هم‌گونی و هرزنامه محاسبه می‌شود. این منحنی توزیع برچسب‌ها در تمام حدود تصمیم‌گیری ممکن را نشان می‌دهد.
10. در انتها، نمودار ROC با استفاده از کتابخانه 'matplotlib' رسم می‌شود. این نمودار نشان می‌دهد که با تغییر آستانه تصمیم‌گیری، نرخ اشتباه مثبت و نرخ صحیح مثبت چگونه تغییر می‌کنند.

سوال ۲ :

مراحل عملکرد کد به شرح زیر است:

1. ابتدا، کتابخانه‌های مورد نیاز را وارد می‌کنیم، از جمله pandas برای کار با داده‌های جدولی، sklearn برای استفاده از المانت‌های مربوط به ماشین بردار پشتیبان و اسکیلر استاندارد، و matplotlib برای رسم نمودارها.
2. سپس، داده‌ها از یک فایل CSV با استفاده از تابع 'read\_csv' در pandas خوانده می‌شوند. فایل CSV شامل ویژگی‌های مربوط به بیماران دیابتی مانند سن، ضربان قلب، فشار خون و غیره است، و ستون "Outcome" برچسب دیابت را نشان می‌دهد.
3. ستون "Outcome" به عنوان متغیر خروجی (y) در نظر گرفته می‌شود و سایر ستون‌ها به عنوان متغیرهای ورودی (X).
4. داده‌های ورودی و خروجی به دو بخش آموزشی و آزمون تقسیم می‌شوند با استفاده از تابع 'train\_test\_split' از sklearn. در اینجا، 80٪ داده‌ها برای آموزش و 20٪ داده‌ها برای آزمون استفاده می‌شود.
5. اسکیلر استاندارد بر روی داده‌های آموزشی و آزمون اعمال می‌شود تا ویژگی‌ها را به مقادیر استاندارد مبدل کند. این مرحله باعث می‌شود ویژگی‌ها بین داده‌ها قابل قیاس و مقایسه باشند.
6. یک مدل SVM با کرنل شعاعی (RBF) ایجاد می‌شود. SVM یک الگوریتم یادگیری ماشینی است که برای دسته‌بندی داده‌ها استفاده می‌شود و با استفاده از مرزهای تصمیم‌گیری بین دسته‌ها، داده‌ها را دسته‌بندی می‌کند.
7. مدل روی داده‌های استاندارد آموزش داده می‌شود تا بتواند پیش‌بینی دیابت صحیحی را ارائه دهد.
8. با استفاده از مدل آموزش دیده، برچسب‌های پیش‌بینی شده برای داده‌های آزمون محاسبه می‌شود.
9. دقت، بازیابی و امتیاز F1 مدل با استفاده از معیارهای مربوطه محاسبه می‌شوند. دقت (accuracy) نسبت تعداد نمونه‌های درست پیش‌بینی شده به کل نمونه‌ها است، بازیابی (recall) نسبت تعداد نمونه‌های درست پیش‌بینی شده از یک کلاس به کل نمونه‌های واقعی این کلاس است و امتیاز F1 (F1-score) معیاری است که ترکیبی از دقت و بازیابی است.
10. ماتریس درهم‌ریختگی (confusion matrix) بر اساس برچسب‌های واقعی و برچسب‌های پیش‌بینی شده برای داده‌های آزمون رسم می‌شود. در اینجا، ماتریس درهم‌ریختگی نشان می‌دهد که چند داده به درستی به هر دسته تعلق می‌گیرد و چند داده به اشتباه به هر دسته تعلق می‌گیرد.

11. نمودار منحنی مشخصه‌ی عملکرد متقاطع (ROC curve) رسم می‌شود. این نمودار نحوه تغییر نرخ اشتباه مثبت و نرخ درست مثبت مدل با تغییر آستانه تصمیم‌گیری را نشان می‌دهد. مساحت زیر منحنی (AUC) نیز یک معیار از عملکرد مدل است.

سوال ۳ :

این برنامه یک مدل پیش‌بینی قیمت مسکن با استفاده از روش پشتیبان وکتور رگرسیون (Support Vector Regression) ایجاد می‌کند. ابتدا داده‌های مسکن از منبع مشخص شده در آدرس `url` دریافت می‌شوند و در یک شیء DataFrame در Pandas ذخیره می‌شوند.

سپس داده‌های ورودی (X) و خروجی (y) برای مدل ساخته می‌شود. داده‌های ورودی شامل تمامی ستون‌های داده به جز ستون "MEDV" (قیمت مسکن) است و داده‌های خروجی شامل ستون "MEDV" هستند.

سپس داده‌ها به دو مجموعه آموزش و آزمون تقسیم می‌شوند (80 درصد برای آموزش و 20 درصد برای آزمون).

سپس داده‌های آموزش مقیاس‌بندی می‌شوند با استفاده از مقیاس‌بندی استاندارد (StandardScaler) تا تأثیر متغیرهای ورودی را به یک مقیاس مشابه برساند. داده‌های آزمون نیز با استفاده از همان مقیاس‌بندی استاندارد مقیاس‌بندی می‌شوند.

سپس یک مدل SVR با هسته RBF ایجاد می‌شود و روی داده‌های آموزش آموزش داده می‌شود.

مقادیر پیش‌بینی شده برای داده‌های آزمون به دست می‌آیند و خطای میانگین مربعات (MSE) بین مقادیر پیش‌بینی شده و واقعی محاسبه می‌شود و چاپ می‌شود.

در نهایت، نموداری از قیمت‌های واقعی و پیش‌بینی شده رسم می‌شود تا بتوان تطابق مدل با داده‌های واقعی را مشاهده کرد.

سوال ۴ :

**\*\* آماده‌سازی داده‌ها: \*\***

- این کد از مجموعه داده‌های 20 خبرگروه (20 Newsgroups) استفاده می‌کند که یک مجموعه داده رایج برای طبقه‌بندی متن است. این مجموعه داده به دو بخش آموزش و آزمایش تقسیم می‌شود.

**\*\* استخراج ویژگی‌ها: \*\***

- کد از TfidfVectorizer استفاده می‌کند تا متن را به یک فرمت عددی تبدیل کند. این فرمت عددی می‌تواند به عنوان ورودی به مدل یادگیری ماشین داده شود.

**\*\* آموزش مدل: \*\***

- کد از یک طبقه‌بند SVM خطی (LinearSVC) استفاده می‌کند و آن را با استفاده از داده‌های آموزشی آموزش می‌دهد.

**\*\* ارزیابی مدل: \*\***

- کد از مدل آموزش داده شده برای پیش‌بینی برچسب‌های کلاس داده‌های آزمایشی استفاده می‌کند.  
- دقت مدل با استفاده از تابع accuracy\_score محاسبه می‌شود.  
- ماتریس سردرگمی (confusion matrix) برای نشان دادن عملکرد مدل ترسیم می‌شود.  
- گزارش طبقه‌بندی (classification report) که شامل صحت، یادآوری، F1-score و پشتیبانی برای هر کلاس است، چاپ می‌شود.

**\*\* نمایش گزارش طبقه‌بندی: \*\***

- کد یک نمودار جدید ایجاد می‌کند و گزارش طبقه‌بندی را به صورت متن روی آن نمایش می‌دهد. این نمایش خواناتر و بصری‌تر از چاپ ساده گزارش است.

سوال ۵ :

**\*\*1. پیش‌پردازش داده‌ها:\*\***

- اسکرپیت فایل CSV را از URL ارائه شده می‌خواند و در یک DataFrame پاندا به نام `df` ذخیره می‌کند.
- سپس 5 سطر اول DataFrame را چاپ می‌کند و همچنین ابعاد دیتاست (تعداد ردیف‌ها و ستون‌ها) را نمایش می‌دهد.
- متغیر هدف به عنوان `SMQ020` انتخاب می‌شود، که احتمالاً یک ستون دسته‌ای است.
- اسکرپیت سپس هر ردیف با مقدار گمشده را حذف می‌کند.
- متغیر هدف را با استفاده از `LabelEncoder` از scikit-learn رمزگذاری می‌کند.

**\*\*2. تقسیم داده‌ها به آموزش و آزمایش:\*\***

- اسکرپیت داده‌ها را به مجموعه‌های آموزش و آزمایش با استفاده از تابع `train\_test\_split` از scikit-learn تقسیم می‌کند.
- اندازه آزمایش به 20٪ از کل داده‌ها تنظیم شده است.

**\*\*3. مقیاس‌سازی ویژگی‌ها:\*\***

- اسکرپیت از `StandardScaler` از scikit-learn برای استاندارد کردن ویژگی‌های ورودی (X) در مجموعه‌های آموزش و آزمایش استفاده می‌کند.

**\*\*4. آموزش مدل:\*\***

- اسکرپیت یک نمونه از طبقه‌بند ماشین بردار پشتیبان (SVM) از scikit-learn با هسته 'rbf' ایجاد می‌کند.
- مدل SVM سپس بر روی داده‌های آموزش برازش داده می‌شود.

**\*\*5. ارزیابی مدل:\*\***

- اسکرپیت با استفاده از مدل SVM آموزش دیده، پیش‌بینی‌هایی بر روی داده‌های آزمایش انجام می‌دهد.
- دقت، بازیابی و نمره F1 عملکرد مدل را محاسبه می‌کند.

**\*\*6. ترسیم تصویر:\*\***

- اسکرپیت یک نمودار ماتریس ابهام ایجاد می‌کند تا عملکرد طبقه‌بندی را نمایش دهد.
- همچنین یک منحنی عملکرد دریافت-عملیاتی (ROC) ایجاد می‌کند و معیار سطح زیر منحنی (AUC) را محاسبه می‌کند.