دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)

دانشکده ریاضی و علوم کامپیوتر

استاد درس: دکتر مهدی قطعی

آبان ۱۴۰۴

تمرین دوم

درس داده‌کاوی محاسباتی

موضوع پروژه: استخراج ویژگی مبتنی بر استقلال خطی و تحلیل تأثیر آن بر فرآیند بهینه‌سازی توابع زیان

هدف از انجام پروژه

هدف از این پروژه، آشنایی عمیق دانشجو با دو مفهوم کلیدی در داده‌کاوی است:

- **استخراج ویژگی:** بررسی نقش استقلال خطی[1] در بهبود کیفیت نمایش داده‌ها و کاهش هم‌خطی[2] میان ویژگی‌ها.
- **تحلیل بهینه‌سازی:** درک این موضوع که چگونه کیفیت فضای ویژگی، مستقیماً بر پایداری و سرعت همگرایی الگوریتم‌های بهینه‌سازی (چه مبتنی بر مشتق و چه بدون مشتق) تأثیر می‌گذارد.

دانشجو می‌آموزد چگونه ویژگی‌هایی طراحی کند که پایه‌های مستقلی از فضای داده تشکیل دهند و سپس به صورت تجربی تحلیل کند که این کار چگونه فرآیند آموزش مدل (بهینه‌سازی تابع زیان) را آسان‌تر، سریع‌تر و پایدارتر می‌کند.

[1] Linear Independence
[2] Collinearity

مفاهیم و مهارت‌هایی که دانشجو خواهد آموخت

- درک مفهومی استقلال خطی و تحلیل ماتریس کوواریانس.
- آشنایی با روش‌های استخراج ویژگی نظیر PCA، ICA، SVD.
- شناخت روش‌های پایه‌ی انتخاب ویژگی مانند Variance Threshold، SelectKBest.
- **مفهوم توابع زیان** (مانند MSE در رگرسیون، Inertia در خوشه‌بندی).
- **درک فرآیند بهینه‌سازی مبتنی بر مشتق** (مانند Stochastic Gradient Descent - SGD).
- **درک فرآیند بهینه‌سازی بدون مشتق/تکرارشونده** (مانند راه‌حل تحلیلی رگرسیون یا الگوریتم E-M در KMeans).
- **درک مدل‌های مبتنی بر فاصله** (مانند KNN) و **مدل‌های مبتنی بر درخت** (مانند RandomForest) و تأثیرپذیری آن‌ها از فضای ویژگی.
- تحلیل تجربی ارتباط میان کیفیت ویژگی و سرعت همگرایی مدل.

تعاریف مفاهیم پایه

استخراج ویژگی[۳] فرآیندی است که طی آن از داده‌های اولیه، مجموعه‌ای از ویژگی‌های جدید، فشرده و معنادار استخراج می‌شود تا ساختار درونی داده‌ها به‌صورت بهینه‌تر نمایش داده شود. در بسیاری از مسائل داده‌کاوی، داده‌های خام شامل ویژگی‌های فراوان، هم‌پوشان یا نویزی هستند که مستقیماً قابل استفاده در مدل‌های یادگیری ماشین نیستند. بنابراین با استفاده از روش‌هایی مانند PCA، SVD، یا Autoencoder، ویژگی‌های جدیدی تولید می‌شوند که روابط پنهان میان متغیرها را آشکار کرده و منجر به بهبود دقت، کاهش ابعاد، و افزایش پایداری مدل می‌گردند. استخراج ویژگی برخلاف انتخاب ویژگی، داده‌های اصلی را به فضای جدیدی تبدیل می‌کند که ترکیب خطی یا غیرخطی از ویژگی‌های اولیه است.

استقلال خطی[۴] به حالتی گفته می‌شود که هیچ‌یک از ویژگی‌ها را نتوان با ترکیب خطی از سایر ویژگی‌ها تولید کرد. ویژگی‌های مستقل خطی، بردارهایی هستند که هرکدام اطلاعات منحصربه‌فردی از فضای داده را ارائه می‌کنند و در نتیجه از تکرار اطلاعات و افزونگی جلوگیری می‌نمایند. در مدل‌های یادگیری، وجود وابستگی خطی شدید میان متغیرها (هم‌خطی) باعث ناپایداری در تخمین ضرایب، افزایش واریانس خطا و کاهش قابلیت تعمیم مدل می‌شود. بنابراین حفظ استقلال خطی ویژگی‌ها از نظر عددی و آماری اهمیت فراوان دارد و یکی از معیارهای کلیدی در طراحی فضاهای ویژگی مؤثر است.

تحلیل مؤلفه‌های اصلی[۵] روشی آماری و هندسی برای یافتن جهت‌هایی در فضای داده است که بیشترین واریانس را توضیح می‌دهند. در این روش، محورهای جدید (مؤلفه‌های اصلی) به گونه‌ای انتخاب می‌شوند که متعامد (Orthogonal) و در نتیجه خطی مستقل باشند. هدف از PCA کاهش بعد داده‌ها، حذف هم‌خطی و تمرکز بر مؤلفه‌هایی است که بیشترین اطلاعات را در خود دارند. این الگوریتم بر پایه‌ی تجزیه‌ی مقادیر منفرد یا تحلیل ویژه‌مقدار (Eigenvalue Decomposition) عمل کرده و در بسیاری از کاربردها مانند فشرده‌سازی داده، حذف نویز، و مصورسازی داده‌ها مورد استفاده قرار می‌گیرد.

[۳] Feature Extraction
[۴] Linear Independence
[۵] Principal Component Analysis (PCA)

تحلیل مؤلفه‌های مستقل[۶] یک روش آماری پیشرفته برای استخراج ویژگی‌هایی است که نه‌تنها از نظر خطی بلکه از نظر آماری نیز مستقل از یکدیگر هستند. در حالی‌که PCA بر واریانس و همبستگی خطی تمرکز دارد، ICA به دنبال یافتن منابع آماری مستقل در داده‌هاست. خروجی ICA معمولاً ویژگی‌هایی با قدرت تفکیک بالا و نویز کمتر است.

تجزیه مقادیر منفرد[۷] یک ابزار ریاضی قدرتمند در جبر خطی است که هر ماتریس $\mathbf{A}$ را به صورت حاصل‌ضرب سه ماتریس $\mathbf{U\Sigma V^T}$ نمایش می‌دهد. در این تجزیه، ماتریس $\mathbf{U}$ و $\mathbf{V}$ شامل بردارهای متعامد و مستقل خطی هستند، و $\mathbf{\Sigma}$ شامل مقادیر منفردی است که میزان اهمیت هر مؤلفه را نشان می‌دهد. SVD پایه‌ی بسیاری از روش‌های کاهش بعد، فشرده‌سازی تصویر، حذف نویز و الگوریتم‌های تحلیل ساختار داده است. مزیت کلیدی آن در این است که حتی برای ماتریس‌های غیر مربعی نیز قابل استفاده است و بهترین تقریب کم‌رتبه‌ی داده‌ها را ارائه می‌دهد.

کاهش بعد[۸] فرآیندی است که طی آن داده‌های پرویژگی (با ابعاد بالا) به نمایش کم‌بعدتری تبدیل می‌شوند، به‌گونه‌ای که اطلاعات کلیدی و ساختار درونی آن‌ها حفظ گردد. این کار علاوه بر کاهش هزینه‌ی محاسباتی، موجب حذف ویژگی‌های غیرمؤثر و افزایش دقت مدل‌های یادگیری می‌شود. روش‌های کاهش بعد به دو دسته‌ی اصلی تقسیم می‌شوند: (۱) روش‌های مبتنی بر نگاشت مانند PCA و t-SNE، و (۲) روش‌های انتخاب ویژگی که زیرمجموعه‌ای از ویژگی‌های اصلی را نگه می‌دارند.

انتخاب ویژگی[۹] روشی است برای شناسایی و نگهداری آن دسته از ویژگی‌هایی که بیشترین تأثیر را بر خروجی مدل دارند. برخلاف استخراج ویژگی، در انتخاب ویژگی فضای داده تغییر نمی‌کند، بلکه برخی از ویژگی‌های غیرضروری یا همبسته حذف می‌شوند. هدف آن افزایش دقت، کاهش پیچیدگی مدل و جلوگیری از بیش‌برازش است. روش‌های انتخاب ویژگی معمولاً در سه گروه قرار می‌گیرند: روش‌های فیلتر (مانند Correlation یا Mutual Information)، روش‌های Wrapper (مانند RFE) و روش‌های Embedded (مانند

[۶] Independent Component Analysis (ICA)
[۷] Singular Value Decomposition (SVD)
[۸] Dimensionality Reduction
[۹] Feature Selection

.(LASSO

ماتریس کوواریانس[۱۰] ماتریسی مربعی است که میزان هم‌تغییری بین هر دو ویژگی را نشان می‌دهد. هر درایه از این ماتریس بیانگر این است که تغییر در یک ویژگی تا چه اندازه با تغییر در ویژگی دیگر همراه است. اگر مقدار کوواریانس نزدیک به صفر باشد، دو ویژگی تقریباً مستقل‌اند، و اگر مقدار بزرگ و مثبت یا منفی باشد، آن‌ها به‌ترتیب هم‌جهت یا خلاف‌جهت تغییر می‌کنند. تحلیل این ماتریس مبنای روش‌هایی مانند PCA است که هدف آن یافتن جهت‌هایی با بیشترین واریانس و کمترین وابستگی است.

هم‌خطی[۱۱] به وضعیتی گفته می‌شود که چند ویژگی رابطه‌ی خطی قوی با یکدیگر دارند. در حضور هم‌خطی، ماتریس کوواریانس یا ماتریس طراحی مدل ممکن است تکین (غیرقابل معکوس) شود و در نتیجه تخمین ضرایب در مدل‌هایی مانند رگرسیون خطی ناپایدار گردد. هم‌خطی موجب افزایش عدم قطعیت ضرایب و کاهش توانایی مدل در تفسیر اثر هر ویژگی می‌شود. برای کاهش این پدیده، از روش‌هایی مانند حذف ویژگی‌های وابسته، نرمال‌سازی داده، یا استفاده از الگوریتم‌هایی نظیر PCA و Ridge Regression بهره گرفته می‌شود.

[۱۰] Covariance Matrix
[۱۱] Collinearity

تعریف پروژه و ساختار اجرایی آن

پروژه باید به‌صورت گام‌به‌گام در محیط Google Colab و با زبان Python اجرا شود. دانشجویان موظف‌اند تمام مراحل زیر را اجرا کرده و نتایج را با نمودار، جدول و تحلیل عددی در گزارش خود ارائه کنند:

- **مرحله اول – انتخاب سه مسئله‌ی داده‌کاوی:** سه دیتاست عمومی زیر را بارگیری و سپس به صورت کلی و مختصرا بررسی نمایید:
  ۱. **طبقه‌بندی (Classification):** (مثال: Wisconsin Breast Cancer)
  ۲. **رگرسیون (Regression):** (مثال: Boston Housing)
  ۳. **خوشه‌بندی (Clustering):** (مثال: UCI Iris)
- **مرحله دوم – بررسی هم‌خطی اولیه داده‌ها:** ماتریس کوواریانس یا همبستگی ویژگی‌ها را محاسبه و با Heatmap نمایش دهید. مقدار وابستگی ویژگی‌ها را تحلیل و مستندسازی کنید.
- **مرحله سوم – استخراج ویژگی‌های مستقل:** با استفاده از الگوریتم‌های PCA، ICA و SVD ویژگی‌های جدیدی بسازید. تعداد مؤلفه‌ها را بر اساس نسبت واریانس توضیح‌داده‌شده (Explained Variance) انتخاب کنید.
- **مرحله چهارم – اعمال روش‌های پایه انتخاب ویژگی:** از روش‌های SelectKBest (مثلاً مبتنی بر f_regression) و RFE (Recursive Feature Elimination) برای انتخاب زیرمجموعه‌ای از ویژگی‌های اصلی استفاده کنید.

- **مرحله پنجم – آموزش مدل و تحلیل فرآیند بهینه‌سازی:** در این مرحله، تأثیر فضای ویژگی را بر فرآیند بهینه‌سازی بررسی می‌کنیم. شما باید هر مدل را روی **سه نسخه** از داده‌ها آموزش دهید: ۱. داده‌های اصلی، ۲. داده‌های استخراج‌شده (PCA)، ۳. داده‌های انتخاب‌شده (RFE/SelectKBest).

الف) مسئله رگرسیون: (ترکیب روش تحلیلی و مبتنی بر مشتق)

۱. **مدل ۱: بهینه‌سازی تحلیلی (بدون مشتق تکرارشونده):**

- از sklearn.linear_model.LinearRegression استفاده کنید.
- **تحلیل:** پایداری ضرایب (model.coef_) را در داده‌های اصلی (با هم‌خطی) و داده‌های PCA مقایسه کنید. آیا ضرایب در داده‌های اصلی به شدت بزرگ و ناپایدار هستند؟

۲. **مدل ۲: بهینه‌سازی مبتنی بر مشتق (Derivative-Based):**

- از sklearn.linear_model.SGDRegressor استفاده کنید (که از گرادیان کاهشی استفاده می‌کند).
- **تحلیل:** تعداد تکرارهای لازم برای همگرایی (n_iter_) و خطای نهایی مدل را روی داده‌های اصلی و داده‌های PCA مقایسه کنید.
- **خروجی الزامی:** نموداری رسم کنید که نشان دهد SGDRegressor روی داده‌های PCA **بسیار سریع‌تر** (در تکرارهای کمتر) به خطای مشابه یا بهتر می‌رسد.

ب) مسئله خوشه‌بندی: (بهینه‌سازی بدون مشتق - تکرارشونده)

۱. **مدل: KMeans (مبتنی بر E-M):**

- الگوریتم KMeans یک بهینه‌ساز تکرارشونده است که مستقیماً از مشتق استفاده نمی‌کند و به فاصله اقلیدسی حساس است.
- **تحلیل:** KMeans را یک بار روی داده‌های **اصلی** و یک بار روی داده‌های **کاهش‌بعدیافته با PCA** (مثلاً k=2 یا k=3 مؤلفه) اجرا کنید.
- مقدار نهایی تابع زیان (اینرسی یا inertia_)، شاخص سیلوئت (Silhouette Score) و زمان محاسبات را مقایسه کنید. (توجه: n_iter_ نیز قابل مقایسه است).

ج) مسئله طبقه‌بندی: (تحلیل مدل‌های مبتنی بر فاصله و مبتنی بر درخت)

۱. **مدل ۱: مبتنی بر فاصله (Instance-Based):**
   - از sklearn.neighbors.KNeighborsClassifier (KNN) استفاده کنید.
   - **تحلیل:** دقت (Accuracy) و **زمان پیش‌بینی** (predict_time) را روی داده‌های **اصلی** (با ابعاد بالا) و داده‌های **کاهش‌بعدیافته با PCA** مقایسه کنید.

۲. **مدل ۲: مبتنی بر درخت (Ensemble):**
   - از sklearn.ensemble.RandomForestClassifier استفاده کنید.
   - **تحلیل:** دقت (Accuracy) مدل را روی هر سه نسخه داده (اصلی، PCA، انتخابی) مقایسه کنید.

- **مرحله ششم – تحلیل نتایج و نمودارها:** تغییرات دقت، خطا، و شاخص‌های خوشه‌بندی را برای هر روش ترسیم و مقایسه کنید. به‌صورت تحلیلی به این سوالات پاسخ دهید:

۱. استقلال خطی (ناشی از PCA) چه تأثیری بر **پایداری** ضرایب LinearRegression داشت؟

۲. استقلال خطی چه تأثیری بر **سرعت همگرایی** الگوریتم SGDRegressor داشت؟ (به نمودار خود ارجاع دهید)

۳. **کاهش ابعاد** با PCA چه تأثیری بر عملکرد (سرعت و کیفیت) KMeans و KNeighborsClassifier داشت؟ چرا این دو مدل به ابعاد داده حساس هستند؟

۴. عملکرد RandomForestClassifier روی داده‌های **اصلی** (با هم‌خطی) چگونه بود؟ آیا می‌توانید نتیجه‌گیری کنید که مدل‌های مبتنی بر درخت نسبت به هم‌خطی مقاوم هستند؟

۵. در نهایت، آیا استخراج ویژگی (PCA) عملکرد بهتری از انتخاب ویژگی (RFE/SelectKBest) در این مسائل داشت؟ (پاسخ می‌تواند برای هر مدل متفاوت باشد).

- **مرحله هفتم – جمع‌بندی و نتیجه‌گیری:** نتایج سه مسأله را مقایسه کرده و توضیح دهید که کدام روش در ایجاد ویژگی‌های مؤثرتر و مستقل‌تر موفق‌تر بوده است و چرا ادغام این دو حوزه (استخراج ویژگی و تحلیل بهینه‌سازی) برای یک متخصص داده‌کاوی اهمیت دارد.

نکات پایانی و دستورالعمل ارسال

- نیاز است که گزارش ارسالی حتما در قالب خواسته شده باشد.
- در صورت هرگونه سوالی در گروه بپرسید.
- ذکر منابع اجباری است.
- نیازی به ریز شدن در مطالب نیست، بیان مسئله و مشکل، و چگونگی رفع مسئله با روش خوانده شده کافی است.
- فایل گزارش خود را به صورت PDF بفرستید.
- تمامی کدها، داده‌ها و فایل‌های پروژه را در یک مخزن GitHub Repository بارگذاری کنید، و لینک دسترسی به ریپازیتوری را در انتهای گزارش خود قرار دهید. مخزن باید به‌صورت Public تنظیم شود تا قابل مشاهده باشد.
- به یاد داشته باشید هدف این تمرین، درک اهمیت این درس، و آشنایی با برخی موضوعات است که در طول آینده بسیار با آن‌ها برخورد خواهید داشت، ملاک نمره حجم گزارش ارسالی نخواهد بود.

موفق باشید.