دوره آفلاین ریاضیات
در علوم‌داده و هوش‌مصنوعی

دوره آموزشی: ریاضیات در علوم‌داده و هوش‌مصنوعی
تاریخ گزارش: ۳۱ تیر تا ۶ مرداد ۱۴۰۵

مدرس: سرکار خانم مریم امیرمزلقانی

نگارنده: سیدعرفان مسعودی

@erfanmasoudiba erfanmasoudiba@gmail.com
linkedin.com/in/erfan-masoudi github.com/ErfanMasoudiBA

فهرست مطالب

۱ پروژه اول: خوشه‌بندی طیفی (Spectral Clustering) ۲
۲ پروژه دوم - تشخیص چهره با PCA ۶

۱ پروژه اول: خوشه‌بندی کاربران در شبکه‌های اجتماعی با Clustering Spectral

۱. هدف و معرفی مسئله

هدف از این پروژه، یافتن اجتماع‌ها (Communities) و گروه‌بندی رئوس یک گراف همبند است؛ به طوری که رئوس درون یک خوشه بیشترین ارتباط (یال) را با یکدیگر و کمترین ارتباط را با خوشه‌های دیگر داشته باشند. در شبکه‌های اجتماعی، این مسئله معادل یافتن گروه‌های دوستی متراکم است. برای حل این مسئله از الگوریتم خوشه‌بندی طیفی (Spectral Clustering) استفاده شده است که به جای استفاده از روش‌های سنتی گراف، مسئله را با استفاده از مقادیر ویژه و بردارهای ویژه‌ی ماتریس‌های گراف به یک مسئله‌ی هندسی و جبر خطی تبدیل می‌کند.

۲. تشکیل ماتریس‌های گراف و لاپلاسین (Graph Laplacian)

الگوریتم با دریافت ماتریس مجاورت (Adjacency Matrix) گراف (A) آغاز می‌شود. در این ماتریس متقارن، وجود یال بین دو راس i و j با مقدار ۱ و عدم وجود آن با صفر نشان داده می‌شود. در گام بعدی، ماتریس درجه (Degree Matrix) با نام D تشکیل می‌گردد که یک ماتریس قطری است و هر درایه روی قطر اصلی آن، برابر با مجموع درایه‌های سطر متناظر در ماتریس A (تعداد یال‌های متصل به آن راس) است.

سپس ماتریس لاپلاسین گراف (Laplacian Matrix) به شکل زیر تعریف می‌شود:

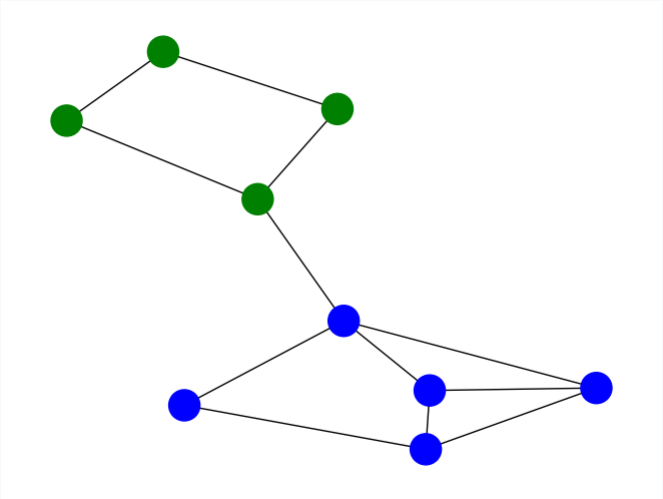
$$L = D - A$$

ویژگی مهم ماتریس لاپلاسین این است که مجموع درایه‌های هر سطر آن دقیقاً برابر با صفر است. این ویژگی باعث می‌شود که ماتریس L همواره تکین (Singular) بوده و حداقل یک مقدار ویژه‌ی صفر داشته باشد. همچنین این ماتریس مثبت نیمه‌معین (Positive Semi-Definite) است، یعنی تمامی مقادیر ویژه‌ی آن نامنفی هستند ($\lambda_i \geq 0$).

۳. استخراج طیف گراف و بردار فیدلر (Fiedler Vector)

پس از تشکیل ماتریس لاپلاسین، معادله‌ی مقادیر ویژه ($Lx = \lambda x$) حل شده و مقادیر ویژه به صورت صعودی مرتب می‌شوند. همان‌طور که اشاره شد، کوچک‌ترین مقدار ویژه همواره $\lambda_1 = 0$ است. دومین مقدار ویژه‌ی کوچک (λ_2) که به آن «همبندی جبری» (Algebraic Connectivity) می‌گویند، نقش کلیدی در خوشه‌بندی دارد. بردار ویژه‌ی متناظر با این مقدار ویژه، **بردار فیدلر** (Fiedler Vector) نامیده می‌شود.

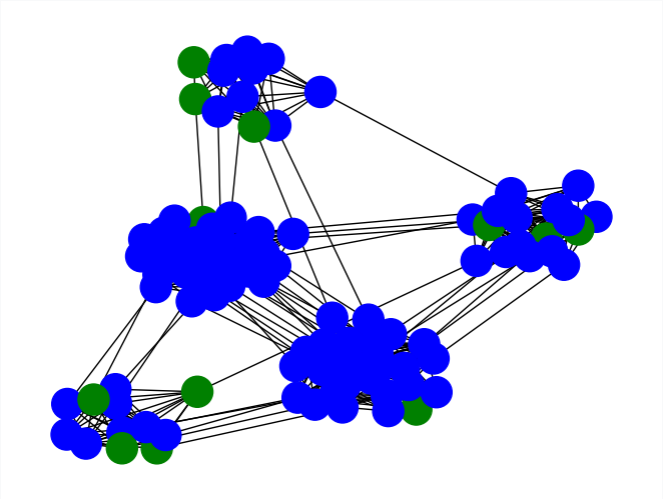
با بررسی علامت درایه‌های بردار فیدلر، می‌توان گراف را به دو بخش بهینه (برش کمینه یا Min-Cut) تقسیم کرد. رئوسی که درایه‌ی متناظرشان در بردار فیدلر مثبت است در یک خوشه، و رئوسی که درایه‌شان منفی یا صفر است در خوشه‌ی دیگر قرار می‌گیرند. این روش ابتدا روی یک گراف تمرینی ۹ راسی با موفقیت تست شد.



شکل ۱: خوشه‌بندی گراف ۹ راسی به ۲ خوشه با استفاده از علامت درایه‌های بردار فیدلر

۴. پیاده‌سازی بر روی گراف اصلی (Main Graph)

پس از تایید منطق روی گراف تمرینی، الگوریتم روی داده‌های گراف اصلی (شامل ۱۰۰ راس که از فایل متنی استخراج شده بود) اجرا گردید. ماتریس لاپلاسین ۱۰۰×۱۰۰ تشکیل و بردار فیدلر استخراج شد. با اعمال شرط علامت ($x > ۰$) روی بردار فیدلر، گراف پیچیده‌ی اصلی به دو اجتماع مجزا و متراکم تقسیم شد.

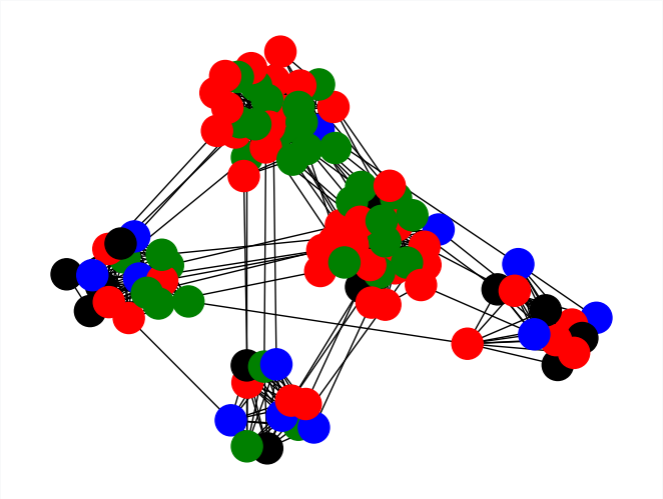


شکل ۲: تقسیم‌بندی گراف ۱۰۰ راسی اصلی به ۲ خوشه مجزا با استفاده از بردار فیدلر

۵. بهبود خوشه‌بندی با تعبیه در فضای جدید و الگوریتم K-Means

برای استخراج تعداد خوشه‌های بیشتر (مثلاً $k = ۴$)، استفاده‌ی دستی از علامتِ چند بردار ویژه (حالت‌های $+, -$ و ...) دقت بالایی ندارد. رویکرد اصولی در خوشه‌بندی طیفی این است که رئوس گراف در یک فضای هندسی جدید با ابعاد پایین‌تر تعبیه (Embed) شوند.

برای خوشه‌بندی به k دسته، k بردار ویژه‌ی اول ماتریس لاپلاسین انتخاب شده و به صورت ستونی کنار هم قرار می‌گیرند تا ماتریس تعبیه (Embedding Matrix) با ابعاد $n \times k$ ساخته شود. اکنون هر سطر این ماتریس، مختصات یک راس را در فضای $\mathbb{R}^k$ نشان می‌دهد. در نهایت، با اعمال الگوریتم استاندارد K-Means بر روی سطرهای این ماتریس، خوشه‌بندی نهایی و دقیق با هر تعداد دسته‌ی دلخواه استخراج می‌گردد.



شکل ۳: خوشه‌بندی نهایی گراف ۱۰۰ راسی به ۴ خوشه با ترکیب بردارهای ویژه و الگوریتم K-Means

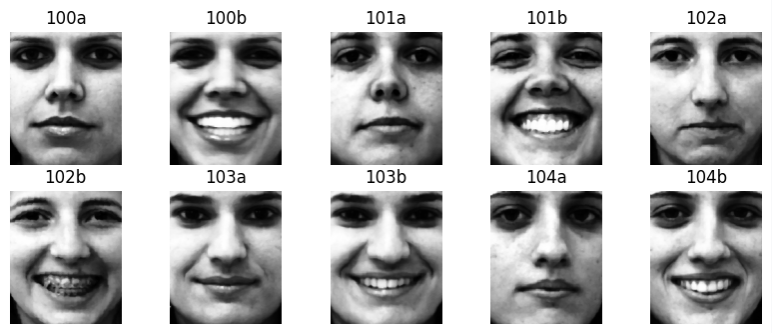
۶. جمع‌بندی

این پروژه نشان داد که چگونه می‌توان یک ساختار گسسته و پیچیده مانند گراف را با استفاده از ماتریس لاپلاسین و طیف آن (مقادیر و بردارهای ویژه) به یک فضای پیوسته‌ی هندسی نگاشت کرد. الگوریتم خوشه‌بندی طیفی، به ویژه زمانی که با K-Means ترکیب می‌شود، یکی از قدرتمندترین روش‌ها برای یافتن ساختارهای پنهان و اجتماع‌ها در شبکه‌های پیچیده است که نسبت به الگوریتم‌های سنتی گراف، مقاومت و دقت بسیار بالاتری در برابر نویز و یال‌های تصادفی نشان می‌دهد.

۲ پروژه دوم - تشخیص چهره با الگوریتم PCA

۱. آماده‌سازی و بارگذاری داده‌ها (Load Data)

در این پروژه قصد داریم از الگوریتم Principal Component Analysis (PCA) برای استخراج ویژگی‌های پنهان تصاویر چهره و بازسازی آن‌ها استفاده کنیم. دیتاست مورد استفاده شامل تصاویر چهره افراد در دو حالت طبیعی و خندان است. در گام نخست، تصاویر خوانده شده و برای بررسی صحت بارگذاری، تعدادی از آن‌ها رسم شدند.



شکل ۴: نمونه‌ای از تصاویر چهره‌ی بارگذاری شده از مجموعه داده

الگوریتم PCA برای انجام عملیات جبر خطی نیازمند داده‌های برداری است. بنابراین، هر تصویر دوبعدی با ابعاد ۱۶۲×۱۹۳ پیکسل، به یک بردار ستونی تک‌بعدی (Flatten) با طول $D = ۳۱۲۶۶$ تبدیل گردید. با قرار دادن ۱۹۰ تصویر اول (مربوط به چهره‌های طبیعی) در کنار یکدیگر، ماتریس کل دادگان با نام Γ تشکیل شد:

$$\Gamma = \begin{bmatrix} | & | & & | \\ \Gamma_۱ & \Gamma_۲ & \dots & \Gamma_{۱۹۰} \\ | & | & & | \end{bmatrix} \in \mathbb{R}^{۳۱۲۶۶ \times ۱۹۰}$$

بهینه‌سازی پیاده‌سازی: برای جلوگیری از افت عملکرد (Performance Drop) ناشی از تخصیص پویای حافظه در حلقه‌ها هنگام کار با کتابخانه NumPy، از روش پیش‌تخصیص حافظه (Pre-allocation) با استفاده از تابع `np.empty` استفاده گردید تا ماتریس دادگان با بالاترین سرعت ممکن تشکیل شود.

۲. محاسبه تصویر میانگین (Mean Image) و نرمال‌سازی

الگوریتم PCA به دنبال یافتن جهت‌های بیشترین واریانس است. بدین منظور، مبدأ مختصات فضای داده‌ها باید دقیقاً در مرکز آن‌ها قرار گیرد. لذا ابتدا بردار «چهره میانگین» (Ψ) با میانگین‌گیری روی تمامی ۱۹۰ تصویر محاسبه شد:

$$\Psi = \frac{1}{n}\sum_{i=1}^{n}\Gamma_i$$

شکل ۵: تصویر چهره میانگین (Ψ) محاسبه شده از ۱۹۰ چهره‌ی حالت طبیعی

سپس با کسر کردن این چهره میانگین از تک‌تک بردارهای تصاویر ($\Phi_i = \Gamma_i - \Psi$)، ماتریس داده‌های نرمال‌شده (A) تشکیل گردید که در مراحل بعدی به عنوان ورودی اصلی الگوریتم مورد استفاده قرار می‌گیرد.

۳. ماتریس کوواریانس و استخراج چهره‌های ویژه (Covariance Trick & Eigenfaces)

برای یافتن جهت‌های بیشترین تغییرات، باید مقادیر ویژه و بردارهای ویژه‌ی ماتریس کوواریانس $C = AA^T$ محاسبه می‌شد. اما با توجه به ابعاد ماتریس A (که برابر با ۱۹۰ × ۳۱۲۶۶ است)، ماتریس C ابعادی برابر با ۳۱۲۶۶ × ۳۱۲۶۶ خواهد داشت که محاسبه‌ی مقادیر ویژه‌ی آن از نظر محاسباتی و حافظه تقریباً غیرممکن است.

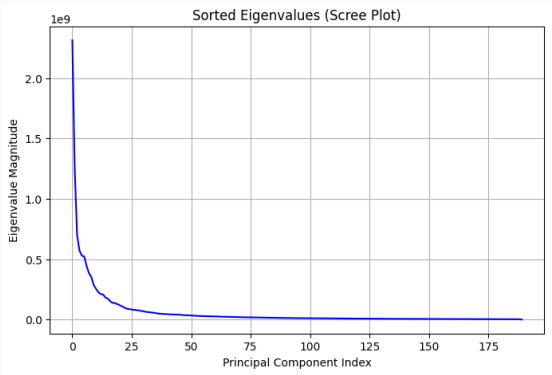
برای حل این چالش، از ترفند ریاضی ماتریس کوواریانس استفاده شد. به جای ماتریس بزرگ C، ماتریس بسیار کوچک‌تر $L = A^T A$ با ابعاد ۱۹۰ × ۱۹۰ تشکیل گردید.

از نظر جبری ثابت می‌شود که اگر v بردار ویژه‌ی ماتریس L با مقدار ویژه‌ی λ باشد:

$$\begin{aligned}(A^TA)v = \lambda v \implies & A(A^TA)v = A(\lambda v)\\ \implies & (AA^T)(Av) = \lambda(Av)\\ \implies & C(Av) = \lambda(Av)\end{aligned}$$

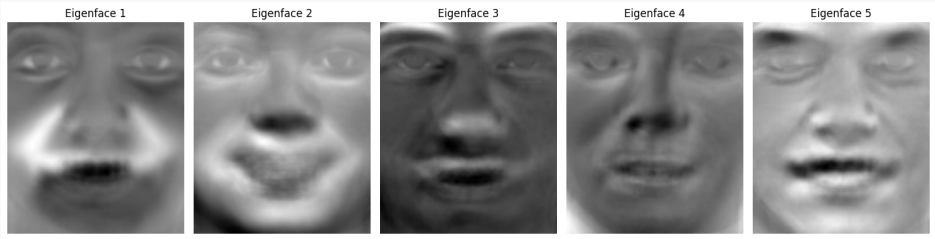
طبق این اثبات، با ضرب بردارهای ویژه‌ی ماتریس کوچک L در ماتریس داده‌ها (A)، بردارهای ویژه‌ی ماتریس اصلی ($U = Av$) به دست می‌آیند. در نهایت، برای ایجاد یک پایه‌ی یکّه و متعامد (Orthonormal Basis)، ستون‌های ماتریس U بر اندازه خود (L2-Norm) تقسیم شدند.

پس از استخراج، مقادیر ویژه به صورت نزولی مرتب شدند. نمودار زیر نشان می‌دهد که بخش عمده‌ی واریانس (اطلاعات) داده‌ها تنها در تعداد محدودی از مؤلفه‌های اصلی ابتدایی نهفته است.



شکل ۶: نمودار مقادیر ویژه مرتب‌شده (Scree Plot) که میزان اطلاعات هر مؤلفه را نشان می‌دهد

با تغییر شکل (Reshape) بردارهای ویژه‌ی استخراج‌شده به ابعاد تصویر اولیه، تصاویری موسوم به «چهره‌های ویژه» (Eigenfaces) به دست می‌آیند. در شکل زیر ۵ بردار ویژه‌ی برتر که بیشترین واریانس داده‌ها را در بر دارند، نمایش داده شده‌اند.



شکل ۷: پنج چهره‌ی پایه‌ی برتر (Top 5 Eigenfaces) استخراج‌شده از ماتریس کوواریانس

۴. بازسازی تصاویر (Image Reconstruction)

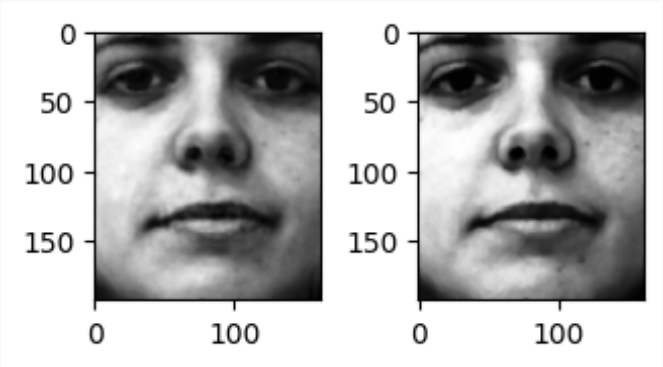
برای بازسازی یک تصویر، ابتدا باید آن را روی فضای جدیدِ ساخته‌شده توسط چهره‌های پایه تصویر (Project) کنیم تا وزن‌های ترکیب خطی به دست آیند. فرض کنید U_K ماتریسی شامل K بردار ویژه‌ی برتر باشد و Φ تصویر نرمال‌شده‌ی ورودی باشد:

$$W = U_K^T \Phi$$

ماتریس W نشان‌دهنده‌ی وزن‌هایی است که مشخص می‌کند تصویر ورودی چه مقدار از هر چهره‌ی پایه را در خود دارد. سپس با ضرب این وزن‌ها در چهره‌های پایه و افزودن چهره‌ی میانگین (Ψ)، تصویر بازسازی‌شده (Γ_{rec}) به فضای پیکسلی اولیه بازگردانده می‌شود:

$$\Gamma_{rec} = (U_K W) + \Psi$$

شکل زیر مقایسه‌ای بین چهره‌ی اصلی و چهره‌ی بازسازی‌شده را نشان می‌دهد. عملکرد موفقیت‌آمیز الگوریتم در حفظ ساختار کلی چهره (با وجود کاهش شدید ابعاد داده‌ها) کاملاً مشهود است.

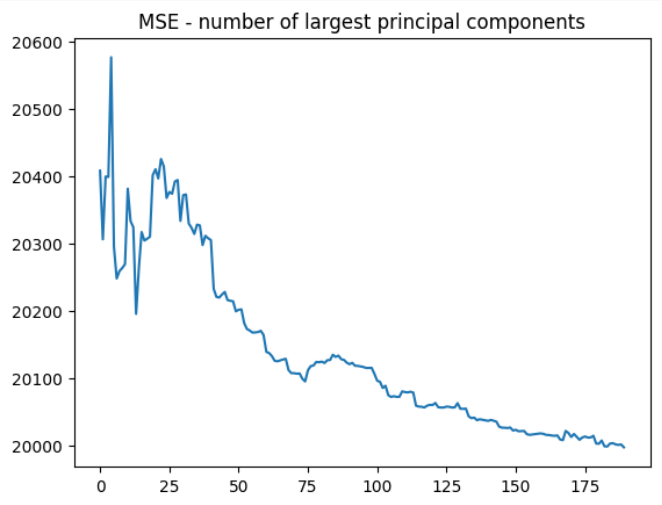


شکل ۸: مقایسه تصویر اصلی (راست) و تصویر بازسازی‌شده با استفاده از بردارهای ویژه (چپ)

۵. تحلیل اثر تعداد مؤلفه‌های اصلی بر کیفیت بازسازی (Trade-off Analysis)

در این بخش، هدف بررسی موازنه (Trade-off) میان میزان فشرده‌سازی اطلاعات و خطای بازسازی تصویر است. بدین منظور، یک تصویر به صورت تصادفی از مجموعه داده‌های آموزشی انتخاب گردید و عملیات بازسازی به ازای تمامی مقادیر ممکن برای تعداد چهره‌های پایه ($1 \leq K \leq 190$) روی آن انجام شد.

با محاسبه‌ی میانگین مربعات خطا (MSE) برای هر حالت، کمترین و بیشترین میزان خطا استخراج گردید. از نظر ریاضی، زمانی که از تنها یک بردار ویژه ($K = 1$) استفاده شود، بیشترین اطلاعات دور ریخته شده و خطا Max خواهد بود. در مقابل، زمانی که $K = 190$ باشد، از آنجا که تمام فضای ویژگی حفظ شده است، بازسازی باید بدون نقص باشد و خطای MSE به کمترین حد خود (نزدیک به صفر) می‌رسد. نمودار زیر روند نزولی MSE را با افزایش K نشان می‌دهد.



شکل ۹: نمودار کاهش خطای بازسازی (MSE) با افزایش تعداد مؤلفه‌های اصلی (K)

برای درک بصری این پدیده، تصویر اصلی به همراه ۵ تصویر بازسازی‌شده با مقادیر مختلف K (که کل محدوده‌ی چهره‌های پایه را پوشش می‌دهند) رسم شدند. همان‌طور که در شکل زیر پیداست، با $K = 5$ تصویر حالتی شبح‌گونه دارد، اما با رسیدن به $K = 50$، با وجود اینکه ابعاد داده از ۳۱۲۶۶ به تنها ۵۰ عدد کاهش یافته است، چهره به خوبی قابل تشخیص است. این موضوع بیانگر قدرت بالای PCA در فشرده‌سازی تصاویر (Image Compression) می‌باشد.

شکل ۱۰: روند بهبود کیفیت تصویر بازسازی‌شده به ازای مقادیر مختلف K

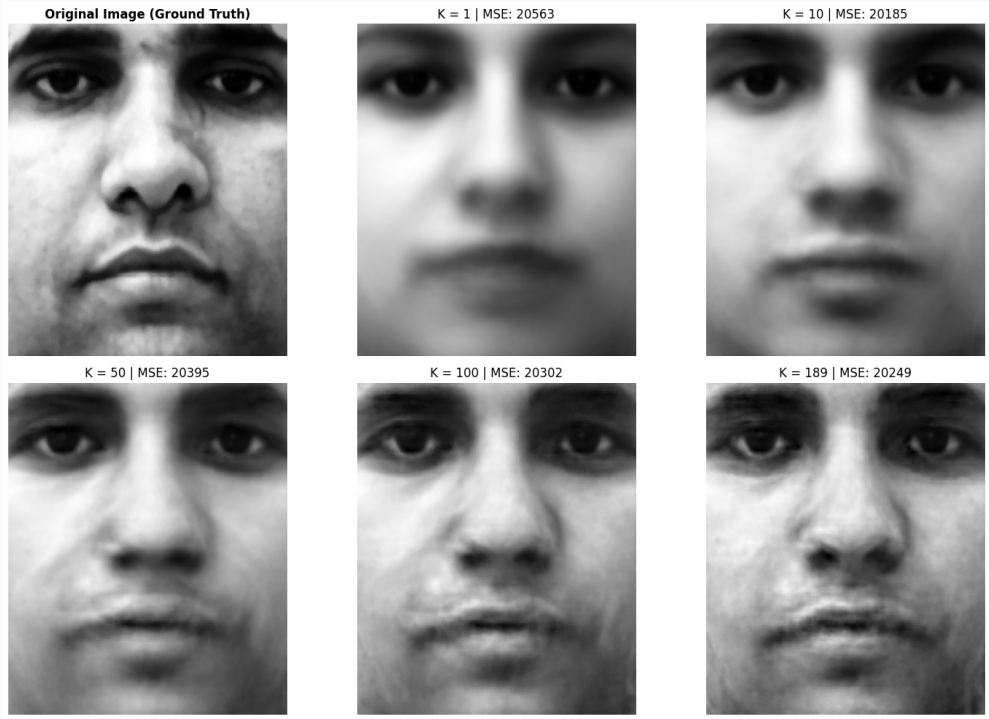
۶. ارزیابی مدل روی چهره‌های خندان (Smiling Images)

تا به اینجا، فضای ویژگی (Face Space) تنها با استفاده از چهره‌های با حالت طبیعی (Neutral) ساخته شده بود. در این گام، یک چهره‌ی خندان از مجموعه داده انتخاب و بازسازی شد. نتایج نشان داد که اگرچه الگوریتم توانست ساختار کلی چهره را بازسازی کند، اما خطای MSE نسبت به حالت چهره‌ی طبیعی کمی افزایش یافت. دلیل این امر آن است که تغییر حالت چهره به خندان، باعث جابجایی پیکسل‌های لب و چشم‌ها می‌شود و از آنجا که مدل PCA ما روی داده‌های خنثی آموزش دیده است، در بازسازی دقیق این تغییراتِ خارج از وزیع آموزش (Out-of-Distribution) کمی دچار خطا می‌گردد.

شکل ۱۱: روند بازسازی یک چهره‌ی خندان با استفاده از مدل آموزش‌دیده روی چهره‌های طبیعی

۷. ارزیابی روی داده‌های دیده نشده (Test Set)

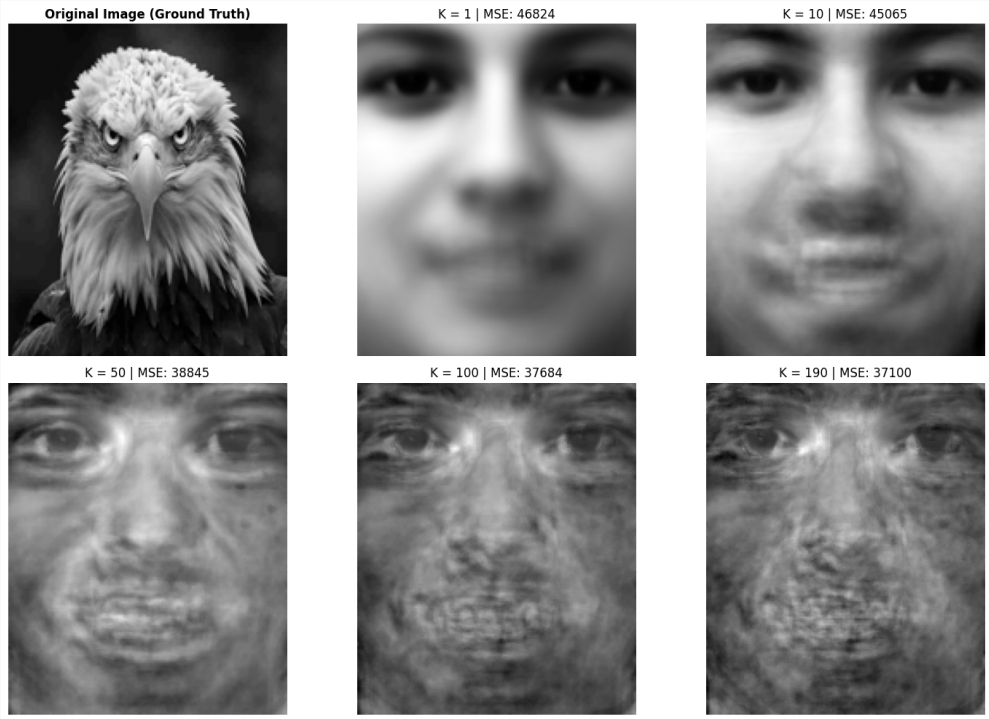
یکی از اصول اساسی یادگیری ماشین، ارزیابی مدل روی داده‌های جدید (Unseen Data) است. برای این کار، ۱۰ عکس جدید که در فاز آموزش حضور نداشتند انتخاب شدند. نکته‌ی بسیار مهم ریاضی در این بخش، عدم محاسبه‌ی مجدد میانگین و بردارهای ویژه برای داده‌های تست است. تصاویر تست باید با استفاده از چهره‌ی میانگینِ فاز آموزش (Ψ_{train}) نرمال‌سازی شده و روی همان بردارهای ویژه‌ی قبلی تصویر (Project) شوند. نتایج بازسازی نشان داد که الگوریتم Eigenfaces دارای قدرت تعمیم‌پذیری (Generalization) مناسبی برای چهره‌های جدید است.



شکل ۱۲: بازسازی موفقیت‌آمیز تصویر فردی که در مجموعه داده‌های آموزش حضور نداشته است

۸. چالش بازسازی تصاویر غیرانسانی (Non-human Images)

برای بررسی درک الگوریتم از مفهوم «چهره»، تصاویری از یک عقاب و یک ماشین به سیستم داده شد. از آنجا که پایه و اساس مدل ما (U_K) تنها از ویژگی‌های صورت انسان (چشم، بینی انسان و ...) تشکیل شده است، الگوریتم به صورت جبری تلاش می‌کند تا تصویر ماشین را با ترکیب اجزای صورت انسان بسازد! نتیجه‌ی این کار، تولید تصاویری شبح‌مانند و بسیار نامفهوم با خطای MSE به شدت بالا بود. این آزمایش به وضوح نشان داد که فضای PCA ساخته‌شده، کاملاً به دامنه داده‌های آموزش (Domain-Specific) محدود است.

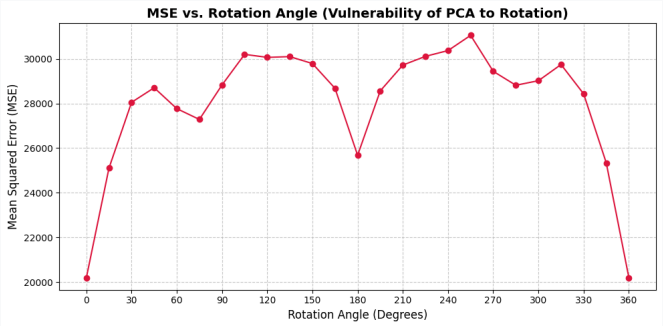


شکل ۱۳: تلاش الگوریتم برای بازسازی تصویر غیرانسانی با استفاده از چهره‌های پایه‌ی انسانی

۹. چالش مقاومت در برابر چرخش (Rotation Vulnerability)

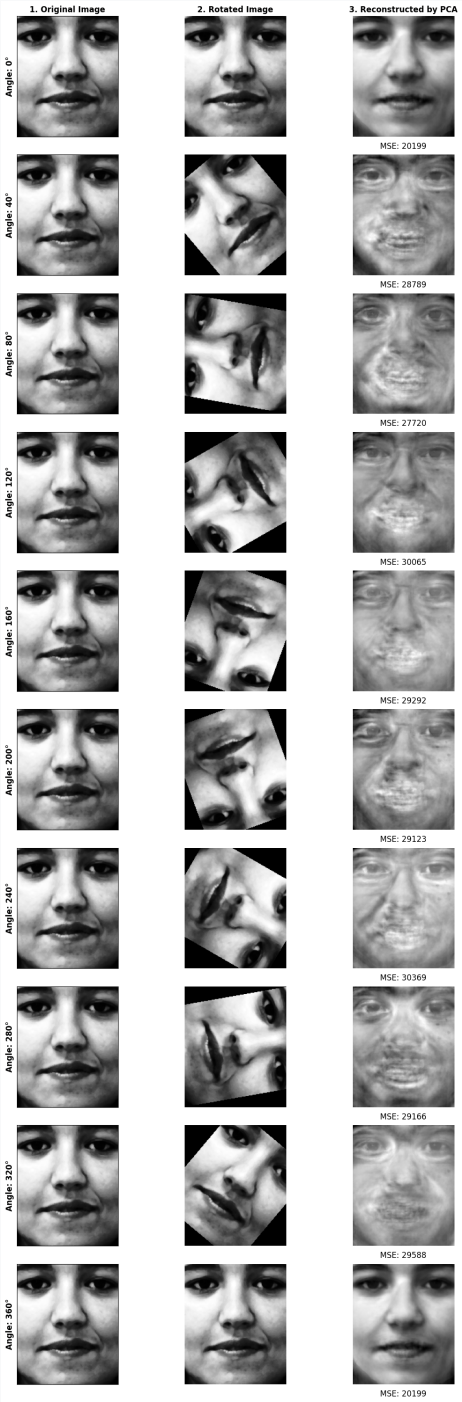
بزرگترین نقطه‌ضعف الگوریتم‌های مبتنی بر فلت کردن (Flattening) تصاویر، عدم مقاومت آن‌ها در برابر تغییرات هندسی مانند چرخش است. برای اثبات این موضوع، یک تصویر از ۰ تا ۳۶۰ درجه (با گام‌های مشخص) دوران داده شد و خطای بازسازی آن اندازه‌گیری گردید.

همان‌طور که در نمودار زیر مشاهده می‌شود، در زوایای ۰ و ۳۶۰ درجه خطا در کمترین حالت خود قرار دارد. اما با نزدیک شدن به زاویه ۱۸۰ درجه (تصویر سر و ته)، خطای MSE به یک قله‌ی بزرگ می‌رسد. از نظر منطقی، با چرخش ۱۸۰ درجه‌ای، پیکسل‌های مربوط به موی سر دقیقاً روی مختصاتی قرار می‌گیرند که الگوریتم انتظار دیدن فک و چانه را دارد.



شکل ۱۴: نمودار افزایش شدید خطای MSE در اثر چرخش تصویر (عدم مقاومت در برابر چرخش)

برای نمایش بصری این فاجعه‌ی محاسباتی، تصاویر اصلی، چرخش‌یافته و بازسازی‌شده در زوایای مختلف به صورت زیر رسم شدند. این آزمایش دلیل اصلی عبور جامعه‌ی هوش مصنوعی از الگوریتم PCA برای پردازش تصویر و روی آوردن به شبکه‌های عصبی پیچشی (CNN) را توجیه می‌کند.



شکل ۱۵: گرید بررسی تحلیلی روند شکست الگوریتم در بازسازی تصاویرِ دوران‌یافته