线性代数的五把钥匙 - 从几何直观到算法实践（通俗授课版）

幻灯片 1：封面页

标题：线性代数的五把钥匙 - 从几何直观到算法实践

副标题：5个经典问题串起整个知识图景

内容块：

- **教学理念**：先几何 → 后代数 → 再应用（比喻：先看风景全貌，再找路线图，最后亲自走一遍）
- **课时规划**：6次课 × 45分钟（比喻：每天解锁一把钥匙，最后用第六天整合开门）
- **核心工具**：Python + GeoGebra + B站视频（比喻：Python是计算器，GeoGebra是放大镜，B站视频是向导）
- **预期收获**：看见抽象概念背后的几何灵魂（比喻：给冰冷的公式，装上看得见的“骨架”）

幻灯片 2：课程导航图

标题：五把钥匙如何打开线性代数大门？

中心视觉：五角星结构图，每个顶点对应一个问题（比喻：线性代数大门是五角星锁，五把钥匙各开一个锁孔，缺一不可）

钥匙	核心问题	解锁知识点（通俗比喻）	现实应用（通俗场景）
#1	影子有多长？	投影、正交性、最小二乘（比喻：给向量“量身高”“找最近的路”）	3D渲染、信号处理（游戏画面渲染、手机信号降噪）
#2	照片如何压缩？	特征值、SVD、主成分（比喻：给照片“减肥”，只留核心轮廓）	图像压缩、推荐系统（微信发图省流量、抖音推荐喜欢的视频）
#3	谷歌如何排序？	马尔可夫链、谱收敛（比喻：网页“投票”，谁得票多排前面）	PageRank、网络分析（百度搜东西排顺序、朋友圈影响力分析）
#4	机器人如何定位？	四大子空间、伪逆（比喻：给机器人“画地图”，找精准位置）	GPS、传感器融合（手机导航定位、扫地机器人避障）

#5	形状如何变形?	行列式、矩阵分解（比喻：给图形“捏橡皮泥”，拉伸旋转不变形）	计算机图形学（动画人物变形、游戏场景缩放）

底部提示：每把钥匙对应一个B站精讲视频（10-15分钟）（比喻：每个钥匙都配了“使用说明书”视频）

幻灯片 3：第一讲 - 影子有多长？（投影与正交性）

3.1 问题引入（视觉优先）

左侧：动态GIF - 阳光下的旗杆影子变化（比喻：旗杆就是向量，影子就是向量在地面上的“投影分身”）

右侧：核心问题链

- 如何量化“一个向量在另一个方向上的分量”？（比喻：如何量出旗杆影子的长度？）
- 为什么正交基能让计算变得简单？（比喻：为什么用直角坐标系量东西，比歪坐标系方便？）
- 当方程无解时，如何找“最近似”的解？（比喻：找不到完美匹配的衣服，如何找最合身的那一件？）

3.2 几何本质（动画演示）

点积的真正含义：

$a \cdot b = |a| \times (|b|\cos\theta)$

↓　　↓

被投影长度 × 投影方向的长度

关键洞察：当$|b|=1$时，点积 = 投影长度（数值化影子长度）（比喻：把投影方向的“尺子”调成1米长，点积直接读出影子长度）

配图为：向量a在b上的投影动画，标注投影分量与垂直分量（比喻：影子是投影分量，从影子顶端到旗杆底部的线段，是垂直分量）

3.3 核心公式推导（分步显现）

正交投影矩阵：

1. 投影方向：$\text{proj}_u(v) = \frac{u \cdot v}{u \cdot u}u$（比喻：求v在u方向的影子，先算影子长度，再按u方向摆好）
2. 矩阵形式：$P = u(u^Tu)^{-1}u^T$（比喻：把“求影子”的操作，做成一个“一键生成器”矩阵）

3. 推广到子空间：$P = A(A^TA)^{-1}A^T$（比喻：从“一条线”上的影子，推广到“一个平面”上的影子）

教学提示：点击公式可跳转GeoGebra演示（链接已嵌入）（比喻：公式看不懂？点一下看动态演示，像看动画一样）

3.4 经典问题：最小二乘法

场景：测得三点(1,1), (2,2), (3,4)，求最佳拟合直线（比喻：在纸上画三个点，找一条线，让所有点到这条线的距离都最近）

几何解释：

- 无解方程组：$\begin{cases} k+b=1 \\ 2k+b=2 \\ 3k+b=4 \end{cases}$（比喻：找不到一条线同时穿过三个点，就像找不到一条路同时经过三个村子）
- **核心思想**：让残差向量$r = b - A\hat{x}$垂直于列空间（比喻：找一条线，让三个点到线的“距离向量”，都垂直于这条线的方向）
- **结论**：$A^TA\hat{x} = A^Tb$（比喻：这就是“找最合身直线”的数学配方）

视觉元素：三维空间中的三个平面，展示无解时的最小距离点（比喻：三个平面找不到交点，就找三个平面中间“最居中”的那个点）

3.5 配套视频与练习

B站视频：《线性代数的本质》- 第7集《点积与对偶性》

- **链接**：https://www.bilibili.com/video/BV1ys411472E
- **观看要求**：重点理解“对偶空间”视角（7:30-9:00段）（比喻：这段讲点积的本质，像看懂魔术背后的原理）

课堂互动：用激光测距仪数据现场做直线拟合（提供Python代码）（比喻：现场用工具实操，亲手“找最合身的直线”）

幻灯片 4：第二讲 - 照片如何压缩？（特征值与SVD）

4.1 问题震撼开场

左侧：显示一张500×500像素的灰度图（原始大小250KB）（比喻：这是一张“胖照片”，占内存大）

右侧：逐步显示使用前k个奇异值重建的效果

- k=10：模糊轮廓（5KB）（比喻：给照片“瘦身后”只剩骨架，能看出大概样子）
- k=50：可辨别人脸（25KB）（比喻：“瘦身”后保留五官，能认清是谁）
- k=100：接近原图（50KB）（比喻：“瘦身”后细节拉满，和原图几乎一样）

核心问题：为什么只用20%的数据就能保留90%的视觉信息？（比喻：为什么剪掉照片的“赘肉”，不影响看清主体？）

4.2 特征值的几何灵魂

矩阵作用下的不变方向：

- **动画演示**：单位圆 → 椭圆变换（比喻：矩阵就像一个“模具”，把圆形压成椭圆，特征向量就是椭圆的长轴和短轴方向）
- **标注**：椭圆长轴方向 = 最大特征值对应特征向量（比喻：长轴是最“显眼”的方向，对应最大的特征值）
- **数值**：长轴长度/短轴长度 = 特征值之比 = 图像拉伸程度（比喻：长轴比短轴长多少，就是特征值大多少，拉伸程度就多大）

关键定理：

- 对称矩阵 ⇒ 特征向量天然正交 ⇒ 可以找到“自然坐标系”（比喻：对称矩阵的“模具”，压出来的椭圆长轴短轴一定垂直，像直角坐标系一样规整）
- $A = PDP^{-1}$ ⇒ 换基 → 伸缩 → 换回来（比喻：先把图形转到“舒服的角度”，拉伸后再转回去，完成变形）

4.3 SVD：任意矩阵的“万能解剖”

分解式：$A = U\Sigma V^T$（比喻：把矩阵这个“大模具”，拆成三个“小模具”，分别负责旋转、拉伸、再旋转）

几何操作序列（动画分三阶段）：

1. **V^T**：输入空间旋转（找到最佳观察角度）（比喻：把要变形的图形，转个方向，方便后续操作）
2. **Σ**：各坐标轴独立伸缩（主要信息保留）（比喻：只拉伸长轴和短轴方向，保留核心轮廓，去掉细节噪声）
3. **U**：输出空间旋转（对齐到最终结果）（比喻：把拉伸好的图形，再转回到原来的方向）

低秩近似公式：

$$A_k = \sum_{i=1}^k \sigma_i u_i v_i^T$$（比喻：只保留前k个“核心拉伸模式”，后面的小模式都扔掉，实现压缩）

直观解释：只保留最大的k个“模式”，丢弃噪声（比喻：只保留照片的五官、轮廓这些大模式，丢掉像素级的小噪点）

4.4 实战演示环节

Python代码现场运行（代码块可直接复制）：

```python
import numpy as np
from PIL import Image

# 加载图像并转为灰度矩阵
A = np.array(Image.open('face.jpg').convert('L'))

# SVD分解
U, sigma, Vt = np.linalg.svd(A, full_matrices=False)

# 保留前k个奇异值
k = 50
A_compressed = U[:, :k] @ np.diag(sigma[:k]) @ Vt[:k, :]

# 显示压缩率
print(f"压缩率: {k * (U.shape[0] + Vt.shape[1] + 1) / A.size * 100:.2f}%")
```

可视化效果：实时显示不同k值下的重建图像与PSNR值（比喻：现场演示“给照片减肥”，看k值多大时，既省内存又不模糊）

4.5 配套视频与拓展

B站视频：《矩阵的【奇异值分解】(SVD)，图像压缩》

- **链接**：https://www.bilibili.com/video/BV1B44y1C7CX
- **重点片段**：4:00-7:00的奇异值衰减曲线分析（比喻：这段讲哪些是“核心模式”，哪些是“噪声模式”，像挑西瓜一样选关键的）

前沿应用：Eigenfaces人脸识别、Netflix推荐系统的矩阵补全（比喻：人脸识别靠SVD提取人脸特征，推荐系统靠SVD猜你喜欢什么）

幻灯片 5：第三讲 - 谷歌如何排序？（马尔可夫链与谱收敛）

5.1 从实际问题到数学模型

网页链接图示例（5个节点的小型网络）：

A → B, C（A网页有两个链接，指向B和C）

B → C, D（B网页有两个链接，指向C和D）

C → A（C网页有一个链接，指向A）

D → A, B, E（D网页有三个链接，指向A、B、E）

E → D（E网页有一个链接，指向D）

比喻：每个网页都是一个“投票人”，链接就是“投票”，指向谁就给谁投一票

马尔可夫矩阵构建：

- 列和为1（每个网页的投票权为100%）（比喻：每个投票人只能投100%的票，不能多投也不能少投）
- M_{ij} = 从j页面跳转到i页面的概率（比喻：Mij就是j网页给i网页投的“票数比例”）

5.2 幂迭代的几何魔术

核心问题：为什么不断点击“随机链接”最终会稳定？（比喻：为什么大家随机点网页链接，最后大部分人都会停留在几个热门网页上？）

动画演示：

- 初始概率分布：$p_0 = [0.2, 0.2, 0.2, 0.2, 0.2]^T$（比喻：一开始5个网页，每个网页都有20%的人停留）
- 第一次迭代：$p_1 = Mp_0$（各页面获得票数）（比喻：大家按网页链接投票，重新分配停留人数）
- 第k次迭代：$p_k = M^k p_0$（比喻：反复投票k次，不断重新分配人数）
- **收敛过程**：柱状图动态变化，最终稳定在特征向量方向（比喻：投票次数多了，每个网页的停留人数不再变，达到稳定状态）

数学本质：当$k\to\infty$时，M^k趋近于秩1矩阵，所有列都收敛到**主特征向量**（比喻：反复投票后，大家的选择趋于一致，都集中在“最受欢迎”的网页上）

5.3 Perron-Frobenius定理（可视化版）

定理要点：

- 正矩阵 ⇒ 存在唯一最大正特征值λ_1（比喻：所有网页都互相链接的矩阵，一定有一个“最受欢迎”的网页，对应最大特征值）
- λ_1对应正特征向量 ⇒ 可作为概率分布（比喻：最受欢迎的网页，有固定的“粉丝比例”，就是正特征向量）
- $|\lambda_i| < \lambda_1$ $(i\neq1)$ ⇒ 保证收敛（比喻：其他网页的受欢迎程度，都比不上最受欢迎的那个，所以最终会稳定）

教学技巧：用颜色深浅表示矩阵元素大小，展示“正性”如何保证稳定（比喻：颜色越深，链接越多，投票越集中，越容易稳定）

5.4 PageRank的改进模型

阻尼因子d=0.85：

$$M' = dM + (1-d)\frac{1}{n}J$$（比喻：大家85%的概率按网页链接点击，15%的概率随机乱点，避免卡在没链接的网页上）

现实意义：用户有15%概率随机跳转到任意页面，防止“死胡同”（比喻：就像走路遇到死胡同，会随机换条路走，不会一直卡在那）

收敛速度：第二大特征值$|\lambda_2|$决定，通常需要50-100次迭代（比喻：收敛速度就像投票多久能出结果，第二大特征值越小，结果出得越快）

5.5 配套视频与实验

B站视频：《PageRank算法详解：谷歌是如何给网页排名的》

- **链接**：https://www.bilibili.com/video/BV1Zt4y1D7sZ
- **关键片段**：8:00-10:00的阻尼因子解释（比喻：这段讲为什么要留15%的随机点击，像解释走路为什么要避开死胡同）

课堂实验：用NetworkX库模拟100个网页的PageRank收敛过程（比喻：现场模拟100个网页的投票过程，看最终哪些网页排前面）

幻灯片 6：第四讲 - 机器人如何定位？（四大子空间与最小二乘）

6.1 GPS定位的数学困境

场景设定：机器人接收到3个卫星的伪距测量数据（比喻：机器人靠3颗卫星“导航”，测量自己到每颗卫星的距离）

- 理想情况：3个方程，3个未知数 → 唯一解（比喻：3个距离能精准定位，像三角形三个顶点确定一个交点）
- **现实问题**：测量误差 + 时钟偏差 → 超定/欠定系统（比喻：测量距离有误差，就像三角形三个边长度不准，找不到精准交点）

核心矛盾：$Ax = b$ 通常无解或有无穷多解（比喻：找不到完美满足所有条件的位置，就像找不到同时满足所有人要求的方案）

6.2 四大子空间的宇宙观

正交分解全景图（中心大图）：

```
☒ⁿ                ☒☒

┌-----------------┐        ┌-----------------┐
│  行空间 Row(A)  │───┐     │  列空间 Col(A)  │
│            │  │    │           │
│  零空间 Null(A) │───┼───→ │  左零空间 Null(A☒) │
└-----------------┘   │    └-----------------┘
```

$\perp$ $\perp$

比喻：四大子空间就像四个“互不干扰的房间”，行空间和零空间在左边房间，列空间和左零空间在右边房间，同一侧的两个房间互相垂直（像卧室和客厅，互不打扰，门对门）

关键推论：

- $Ax=b$有解 ⇔ $b \in \text{Col}(A)$（比喻：机器人能定位，说明测量数据b在列空间这个“房间”里）
- 解唯一 ⇔ $\text{Null}(A) = \{0\}$（比喻：零空间只有一个“原点”，说明只有一个精准位置）
- 最小二乘解 ⇔ 投影到Col(A)上（比喻：测量数据不在列空间，就找列空间里“最近的点”，作为定位结果）

6.3 最小二乘的四种等价视角

切换视角理解同一问题（比喻：从四个不同角度看“机器人定位”，结论都一样）：

1. **几何视角**：找b到列空间的垂直投影点（比喻：在列空间房间里，找离b最近的点）
2. **代数视角**：解正规方程 $A^TA\hat{x} = A^Tb$（比喻：用公式算出这个最近的点）
3. **优化视角**：最小化 $\|Ax-b\|_2^2$（比喻：让测量误差的总长度最小）
4. **概率视角**：高斯-马尔可夫定理下的最优线性无偏估计（比喻：在所有可能的估计中，这个结果最靠谱）

教学提示：点击标签可展开每种视角的详细推导（比喻：哪个视角看不懂，点一下看详细解释，像查字典一样）

6.4 伪逆与稳定性

当A^TA病态时：

- 条件数$\kappa(A)$很大 ⇒ 数值解不稳定（比喻：测量数据稍微有点误差，定位结果就差很远，像天平两边重量差太小，一点风吹就歪）
- **岭回归**：$(A^TA + \lambda I)\hat{x} = A^Tb$（比喻：给天平加个“配重”，让它更稳定，减少误差影响）
- **几何解释**：在零空间方向加入“阻尼”，防止爆炸（比喻：给零空间这个“房间”加个门挡，不让误差随便扩散）

可视化：误差曲面从“峡谷”变为“碗状”（比喻：原来的误差曲面像峡谷，容易跑偏；加阻尼后像碗，球能稳定在碗底）

6.5 配套视频与实战

B站视频：《线性代数的本质》- 第14集《最小二乘法》

- **链接**：https://www.bilibili.com/video/BV1ys411472E?p=14

- **黄金片段**：3:00-5:00的残差正交性演示（比喻：这段讲误差向量和列空间垂直，像影子和旗杆垂直一样，直观好懂）

传感器融合实验：用 accelerometer + gyroscope 数据融合估计姿态角（比喻：结合加速度计和陀螺仪的数据，像用眼睛和耳朵一起判断位置，更精准）

幻灯片 7：第五讲 - 形状如何变形？（线性变换与几何）

7.1 矩阵 = 形状变形器

交互演示（嵌入GeoGebra小程序）：

- 拖动单位正方形的顶点（比喻：用手捏正方形橡皮泥）
- 实时观察矩阵A如何将其映射为平行四边形（比喻：矩阵就是捏橡皮泥的“手”，把正方形捏成平行四边形）
- **可调整参数**：shear, rotation, scale, reflection（比喻：调整参数，就是改变捏橡皮泥的方式：剪切、旋转、缩放、翻转）

核心性质：

- 原点不动（比喻：捏橡皮泥时，中心点固定不动）
- 直线保持直线（比喻：橡皮泥上的直线，捏完还是直线，不会变成曲线）
- 平行性保持（比喻：橡皮泥上平行的两条线，捏完还是平行）
- **不可逆 ⇔ 降维（压扁到更低维空间）**（比喻：把3D的橡皮泥压扁成2D的，再也变不回3D，就是不可逆）

7.2 行列式的几何意义

动画序列：

1. 单位正方形（面积=1）（比喻：一块边长1厘米的正方形橡皮泥，面积1平方厘米）
2. 经过矩阵A变换 → 平行四边形（面积=|det(A)|）（比喻：捏成平行四边形后，面积变成行列式的绝对值）
3. **定向翻转**：det(A) < 0 时，图形镜像翻转（比喻：捏橡皮泥时把它翻过来，行列式就变成负数）

三维推广：平行六面体的有向体积（比喻：3D空间中，矩阵变换把正方体橡皮泥捏成平行六面体，体积就是行列式的绝对值）

关键公式：$\det(AB) = \det(A)\det(B)$ ⇒ 连续变换的体积缩放相乘（比喻：先捏一次橡皮泥，体积缩放det(A)倍，再捏一次，缩放det(B)倍，总共缩放det(A)×det(B)倍）

7.3 矩阵分解 = 操作说明书

三种分解对比：

分解形式	几何操作（比喻）	应用场景（比喻）	数值稳定性（比喻）
LU	剪切序列（像用剪刀一步步剪橡皮泥）	解方程（像按步骤拆礼物）	需选主元（像剪东西要找对下手的地方）
QR	旋转+伸缩（像转转盘再放大缩小）	最小二乘（像找最圆的石头）	稳定（像转盘很稳，不容易歪）
SVD	旋转→伸缩→旋转（转两次转盘，中间放大缩小）	压缩/降维（像给橡皮泥减肥）	最稳定（像三个步骤都很稳，误差最小）

QR分解演示：Gram-Schmidt正交化过程的动画，每一步投影都可视化（比喻：把歪的坐标系，一步步调成直角坐标系，像把歪的桌子腿掰直）

7.4 仿射变换与齐次坐标

问题：如何用矩阵表示平移？（比喻：如何用“捏橡皮泥”的方式，把图形整体移走，不改变形状）

- **答案**：升维到齐次坐标（比喻：给2D图形加个“高度”维度，变成3D，平移就变成3D空间的旋转）
- **2D平移**：$\begin{pmatrix}1&0&t_x\\0&1&t_y\\0&0&1\end{pmatrix}\begin{pmatrix}x\\y\\1\end{pmatrix}$（比喻：t_x和t_y是x和y方向的平移距离，像把图形在纸上平移t_x厘米和t_y厘米）

三维图形学管线：model → view → projection变换的矩阵链（比喻：制作3D动画时，先摆好模型，再调整视角，最后投影到屏幕上，三步都用矩阵操作）

7.5 配套视频与创作

B站视频：《线性代数的本质》- 第2&3集

- **链接**：https://www.bilibili.com/video/BV1ib411t7YR
- **必看片段**：第3集5:00-7:00的负行列式动画（比喻：这段讲图形翻转，行列式变负，像把橡皮泥翻过来，直观好记）

学生项目：用NumPy实现2D/3D变换器，生成创意几何动画（比喻：自己做一个“捏橡皮泥”工具，做出属于自己的动画）

幻灯片 8：知识图景总览与串联

8.1 五把钥匙的相互关系

动态网络图（支持点击跳转）：

投影与正交性 (#1)

```
        │
        ├────────→ 最小二乘 (#4) ───→ 伪逆/正则化
        │
        └────────→ QR分解 ──────────────→ 矩阵分解家族
                          │
   特征值分解 (#2) ←──────────────────────┤
        │                 │
        └────────→ SVD分解 ──────────────┤
                          │
   马尔可夫链 (#3) ←──────────────────────┘
        │
行列式/变换 (#5)
```

比喻：五把钥匙不是孤立的，而是互相串联的“工具套装”，投影是基础，像螺丝刀；矩阵分解是核心，像扳手；其他钥匙都是基于这两个工具的延伸，能解决不同问题

交互功能：点击每个节点可回溯到对应章节（比喻：想复习哪把钥匙，点一下就回到对应的讲解，像查工具书目录）

8.2 线性代数的三大主题

主题一：空间结构

- 基、维数、子空间、正交性（比喻：空间结构像房子的框架，基是房梁，维数是房子的层数，子空间是房间，正交性是房间之间互不干扰）
- **统领概念**：内积空间的几何（比喻：内积空间是整个房子的设计图，决定了房子的结构和布局）

主题二：线性映射

- 矩阵表示、特征分析、不变子空间（比喻：线性映射像房子里的家具，矩阵是家具的形状，特征分析是找出家具的核心功能，不变子空间是家具放的固定位置）
- **统领概念**：谱分解揭示本质结构（比喻：谱分解是家具的拆解图，能看清家具的组成部分）

主题三：近似与优化

- 投影、最小二乘、低秩近似（比喻：近似与优化像整理房间，投影是把东西归位，最小二乘是找最整齐的摆放方式，低秩近似是扔掉没用的杂物）
- **统领概念**：正交性简化复杂问题（比喻：正交性是整理房间的技巧，用对了能让复杂的房间变得整齐有序）

8.3 从理论到算法的桥梁

每个概念对应的算法实现（比喻：每个理论概念，都有对应的“工具软件”，直接用就行）：

- 投影 → `numpy.linalg.qr`（比喻：投影工具，一键生成影子）
- 特征值 → `numpy.linalg.eig`（比喻：特征值工具，一键找出核心方向）
- SVD → `numpy.linalg.svd`（比喻：SVD工具，一键给照片减肥）
- 最小二乘 → `numpy.linalg.lstsq`（比喻：最小二乘工具，一键找最合身的线）
- 行列式 → `numpy.linalg.det`（比喻：行列式工具，一键计算图形面积/体积）

性能对比表：不同规模矩阵的算法复杂度与实践耗时（比喻：不同工具处理不同大小的任务，耗时不一样，像大锤子砸小钉子效率低，小锤子砸大钉子砸不动）

幻灯片 9：教学实施指南

9.1 课时分配建议

课时	内容	视频（比喻：看工具使用说明书）	实践（比喻：亲手用工具操作）	讨论（比喻：和同学交流使用心得）
1	投影与正交性	10分钟	15分钟	20分钟
2	SVD与压缩	15分钟	20分钟	10分钟
3	PageRank	12分钟	18分钟	15分钟
4	四大子空间	10分钟	20分钟	15分钟
5	线性变换几何	20分钟	15分钟	10分钟
6	综合项目	-	40分钟	5分钟

9.2 环境配置清单

学生端安装：

Code block

```
pip install numpy matplotlib pillow imageio
pip install geogebra-proxy  # 嵌入交互图形
```

比喻：学生端安装这些软件，像给电脑装“工具包”，里面有计算器、画图工具、交互图形工具，方便上课实操

教师端工具：

- **屏幕标注**：Epic Pen（视频讲解时圈重点）（比喻：像老师上课用的粉笔，能在屏幕上圈画重点）
- **实时投票**：Kahoot!（每节课3个概念检测题）（比喻：像上课提问，实时知道学生有没有听懂）
- **代码共享**：Google Colab（学生无需本地配置）（比喻：像老师提前准备好工具，学生直接用，不用自己找）

9.3 评估方式

过程性评估（60%）：

- 每讲完一个问题，提交1页“几何图解”（手绘或代码生成）（比喻：每学完一个工具，提交一份“使用成果”，证明自己会用）
- 小组项目：选一个问题，扩展成15分钟微课（比喻：小组合作，用学过的工具做一个“小讲座”，教别人怎么用）

总结性评估（40%）：

- 开卷考试：给出应用背景，推导相应矩阵公式（比喻：给一个实际问题，让你找出对应的“工具”，并说明怎么用）
- 禁止纯计算，必须包含几何解释（比喻：不光要算出结果，还要说明为什么这么算，像解释工具的使用原理）

幻灯片 10：资源汇总与拓展

10.1 B站视频播放列表

一键收藏链接：https://space.bilibili.com/88461692/channel/seriesdetail?sid=1234567

（包含全部5个推荐视频，按教学顺序排序）（比喻：这是“工具使用说明书”合集，按上课顺序排列，方便复习）

字幕设置建议：

- 中文为主，遇到关键词（rank, nullspace）切换英文（比喻：说明书主要是中文，关键零件名称保留英文，方便查资料）
- 0.75倍速播放数学推导密集段（比喻：遇到复杂的工具拆解步骤，放慢速度看，避免看不懂）

10.2 补充阅读材料

几何直观：

- 《线性代数的几何意义》（任广千著）- 第3、7章（比喻：工具的“原理图解书”，用图形解释为什么工具能这么用）

算法实践：

- 《Python数据科学手册》- 第2章NumPy（比喻：工具的“实操手册”，教你怎么用Python操作这些工具）

理论深化：

- MIT 18.06 Linear Algebra（Gilbert Strang）- Lecture 15, 21, 29（比喻：工具的“高级教程”，适合想深入研究工具原理的同学）

10.3 学生优秀作品展示

往届学生创作：

- 交互式投影计算器（网页版）（比喻：学生自己做的“简易工具”，能实现投影计算功能）
- PageRank模拟器（可导入真实网页链接）（比喻：学生自己做的“投票模拟器”，能模拟网页排名）
- 基于SVD的表情包压缩工具（比喻：学生自己做的“照片减肥工具”，专门压缩表情包）

激励标语："掌握这五把钥匙，你也能创造属于自己的线性代数工具！"（比喻：学会这五个核心工具，你不仅能解决问题，还能自己做新工具，成为“工具创造者”）

附录幻灯片：备用代码与工具

附录A：核心Python函数库

Code block

```python
# 投影矩阵计算
def projection_matrix(A):
    return A @ np.linalg.inv(A.T @ A) @ A.T

# SVD压缩
def svd_compress(A, k):
    U, sigma, Vt = np.linalg.svd(A, full_matrices=False)
    return U[:, :k] @ np.diag(sigma[:k]) @ Vt[:k, :]

# PageRank迭代
def pager
```