Proyectos para Análisis Numérico II - 2025

Luis Biedma y Claudio Armas

FAMAF Facultad de Matemática, Astronomía, Física y Computación

UNC Universidad Nacional de Córdoba

Presentación

1. Deberán formar grupos de a dos.
2. Para cada proyecto deberán desarrollar:
   - parte teórica,
   - modelado y,
   - programación.
3. Escribir un informe (sin restricción de tamaño) y explicarlo en una exposición de 15 minutos.

   Para la redacción del informe se recomienda utilizar **LaTeX, Markdown, Jupyter Notebook o Google Docs**, mientras que para la presentación pueden emplearse **Canva, PowerPoint**, **Google Slides** o también **LaTex**. En cualquier caso, se deberá entregar el informe y la presentación en formato **PDF**.

.

Índice de proyectos

1. Ordenando Aeropuertos
2. Reconocimiento Facial
3. Correspondencia entre imágenes
4. Support Vector Machine
5. Dinámica de poblaciones
6. Reconocimiento de rostros
7. Cadena Colgante
8. Regiones de Convergencia
9. Matrices simétricas indefinidas
10. Propagación del calor
11. Sist. de Recomendación
12. Análisis Discriminante Lineal
13. Implementación de la descomposición QR para la detección de MIMO
14. Factorización QR de Householder por bloques
15. Optimización Convexa para Invertir en la Bolsa
16. La Factorización LU con pivoteo parcial puede fallar en la práctica
17. La Factorización de Cholesky Incompleta
18. Metódo de Lanzcos

Importancia de Aeropuertos por Vuelos

En este artículo se muestra cómo funciona el algoritmo PageRank que creó Google para ordenar páginas web por importancia a partir de cuáles son más citadas por otras, todas ellas relacionadas con la búsqueda de un autovalor dominante cuya existencia es garantizada por el Teorema de Perron-Frobenius. Los requerimientos del proyecto son:

- Entender y demostrar el Teorema de Perrón-Frobenius.
- Utilizar los datos de vuelos de airports.dat y routes.dat, para estimar un ranking de la importancia de los aeropuertos usando el algoritmo de PageRank.

Reconocimiento de Características Faciales

En este artículo se muestra las aplicaciones y varias formas de obtener la descomposición no negativa de una matriz (Nonnegative Matrix Factorization). Los requerimientos del proyecto son:

- Explicar cómo funciona la NMF y alguno de los algoritmos propuestos para obtenerla.
- Implementar uno de los métodos propuestos en el artículo y usarlo sobre parte del dataset Align&Cropped del proyecto de reconocimiento de caras de celebridades.

Correspondencia entre imágenes

El método de Scott and Longuet-Higgins puede utilizarse para crear una correspondencia entre pixeles de dos imágenes. Puede encontrar todos los detalles en este artículo. Los requerimientos del proyecto son:

- Entender y explicar el algoritmo de Scott and Longuet-Higgins.
- Implementar en `Python` este algoritmo y mostrar su funcionamiento con algunas imágenes de retinas del dataset FIRE en Kaggle.

Support Vector Machine

Las máquinas de soporte vectorial son una herramienta muy útil para el aprendizaje automático, permitiendo discriminar datos y clasificarlos. En esta Tesis encontrará la deducción de este método y una aplicación para determinar usuarios fraudulentos. Los requerimientos del proyecto son:

- Entender y explicar la formulación del problema matemático linealmente separable y su problema dual. Mencionar su extensión a SVM multiclases.
- Implementar el algoritmo de SVM usando **scipy.optimize** y probarlo con el conjunto de datos de cáncer de mama de SKLearn.

Dinámica de poblaciones

Cuando se estudia la dinámica en la población de una especie, las tasas de natalidad, mortalidad y de transición de etapas (v.g., de joven a adulto) suelen representarse en una matriz de proyección (o matriz de Leslie-Lewis). Los requerimientos del proyecto son:

- Explicar cómo se construye una matriz de proyección y que información brinda. Detallar que información se obtiene a partir del autovalor y autovector dominante de la matriz de proyección y de su transpuesta (Cap. 2 de este libro).
- Implementar en `Python` un algoritmo que, dada una matriz de proyección, calcule la tasa de crecimiento de población, el valor reproductivo, la sensibilidad y elasticidad de cada etapa.

Reconocimiento de rostros

Armaremos una base de datos con fotografías de rostros y usaremos la descomposición SVD para identificarlos. Los requerimientos del proyecto son:

- Siguiendo estas notas, explicar el procedimiento para identificar rostros mediante la descomposición SVD. Mostrar como puede ser implementado.
- Implementar en `Python` los algoritmos necesarios para que un usuario pueda armar su propia base de datos de fotos y usarla para reconocer rostros.

Posición de una cadena colgante

En el archivo1 encontrarán el modelo de una cadena colgante y el método de Newton para problemas con restricciones de igualdad. En el archivo2 tienen detalles algorítmicos y datos para experimentos numéricos. Los requerimientos del proyecto son:

- Entender y explicar el modelo junto con la adaptación del esquema de simulación a `Python`.
- Implementar el algoritmo y verificar su funcionamiento reproduciendo los resultados mostrados.

Regiones de Convergencia en Sistemas No Lineales

En este artículo muestra cómo la solución de un sistema de ecuaciones no lineales mediante métodos numéricos puede depender del punto inicial que se utilice. Esto genera regiones de convergencia en el caso en el que exista más de una solución y se relaciona con la teoría de fractales. El posible proyecto requiere:

- Completar los detalles de ésta demostración del método de Newton.
- Implementar en `Python` el método de Newton para resolver sistemas no lineales en más de una dimensión y graficar las regiones de convergencia para muchos puntos en un plano, como las mostradas aquí.

Matrices simétricas indefinidas

Para resolver un sistema lineal $Ax = b$ donde A es simétrica pero indefinida, una forma de proceder es mediante la descomposición LBL^T de A, explicada en la subsección 4.9.3 de este libro. Los requerimientos del proyecto son:

- Entender y explicar la factorización LBL^T.
- Implementar en `Python` el Algoritmo 4.1 del libro y mostrar su funcionamiento con algunos ejemplos.

Propagación del calor

Se desea modelar cómo se distribuye el calor a lo largo del tiempo en una placa metálica Ω. La temperatura en cada punto de la placa es dada por la función $u : \mathbb{R}^2 \times [0, \infty) \to \mathbb{R}$ tal que

$$
\begin{gathered}
\frac{\partial u}{\partial t} - c^2 \frac{\partial^2 u}{\partial x^2} - c^2 \frac{\partial^2 u}{\partial y^2} = f(x, y, t), \quad (x, y) \in \Omega, \quad t > 0 \\
u(x, y, 0) = g(x, y), \quad (x, y) \in \Omega.
\end{gathered}
$$

Los requerimientos del proyecto son:

- Formular el esquema implícito en diferencias finitas tomando $\mathrm{u}_{ij}^k = u(x_i, y_j, t_k)$. Escribir el problema en forma matricial y mostrar como resolverlo numericamente.
- Implementar en `Python` un algoritmo que dadas f y g muestre como se distribuye la temperatura en la placa a lo largo del tiempo.

Sistemas de Recomendación

En esta presentación pueden encontrar una explicación de cómo las descomposiciones matriciales pueden ayudar a recomendar películas o cualquier otro producto a personas en una plataforma. Un algoritmo con buenos resultados es una aproximación de la descomposición SVD. Los requerimientos del proyecto son:

- Estudiar y hacer una presentación sobre el algoritmo de aproximación de SVD propuesto por Simon Funk.
- Implementar esta aproximación de la descomposición SVD para recomendar películas a partir de una matriz de usuarios y calificaciones, con los datos de Movielens.

Análisis Discriminante Lineal (LDA)

El análisis discriminante lineal es una de las herramientas básicas del Machine Learning, utilizado para discriminar datos y clasificarlos. En este artículo encontrará la deducción de este método y una forma de implementarlo. Los requerimientos del proyecto son:

- Entender y explicar la formulación del problema matemático y el algoritmo.
- Implementar en Python el algoritmo de LDA y probarlo con el conjunto de datos de cáncer de mama de SKLearn.

Implementación de la descomposición QR para la detección MIMO (Multiple Input Multiple Output)

Los dispositivos de comunicación se han convertido en una parte crucial de nuestra vida cotidiana. La variedad de aplicaciones que pueden llevar a cabo exige una alta velocidad de los datos y garantizar una alta calidad de los mismos. La tecnología MIMO busca aumentar la capacidad de transmisión de datos y mejorar la calidad de la señal. En este artículo se presenta un método modificado de descomposción QR basado en la rotación de Givens para la detección MIMO. Los requerimientos del proyecto son:

- Explicar cómo funciona la descomposicón QR para la detección MIMO.
- Implementar el método modificado de descomposición QR y comparar su eficiencia con el método original.

Factorización QR de Householder por bloques

Los procesadores son rápidos pero las memorias lentas, por lo que se usan memorias cachés para maximizar la reutilización de datos. Operaciones simples tienen poca reutilización, mientras que las operaciones entre matrices permite aprovechar mejor la caché. Es por ello que los algoritmos que operan sobre bloques reorganizan los cálculos para usar esta estrategia de manera eficiente. En la subsección 3.4.1 del Libro se explica un algoritmo de la factorización QR de Householder por bloques. Los requerimientos del proyecto son:

- Entender y explicar cómo funciona la factorización QR de Householder por bloques.
- Implementar el algoritmo y mostrar su funcionamiento con algunos ejemplos.

Optimización Convexa para Invertir en la Bolsa

La optimización convexa es una técnica matemática que puede utilizarse para encontrar la mejor solución para un problema de inversión en la bolsa. En esta presentación se explica el uso de la optimización convexa y los conceptos básicos. Los requerimientos del proyecto son:

- Explicar cómo funciona la optimización convexa y cómo se puede aplicar para seleccionar un portfolio de inversión.
- Implementar una estrategia de inversión utilizando la optimización convexa y crear portfolios de inversión en acciones, obteniendo los datos con la librería yfinance.

La Factorización LU con pivoteo parcial puede fallar en la práctica

Aunque la factorización LU con pivoteo parcial se utiliza ampliamente, existen matrices para las cuales el error puede crecer de manera exponencial, lo que resulta problemático, especialmente en matrices de gran dimensión. En este artículo se presentan ejemplos de ecuaciones integrales y diferenciales que ilustran este problema. Tomando como referencia los ejemplos del artículo, los requerimientos del proyecto son:

- Entender y explicar el fallo de la factorización LU con pivoteo parcial.
- Evaluar y mostrar (gráficamente) el crecimiento exponencical del error para matrices de menor a mayor dimensión y las soluciones exactas y aproximadas.

Factorización de Cholesky Incompleta

La factorización de Cholesky incompleta (IC) es un método para descomponer una matriz dispersa en matrices triangulares inferior y superior, considerando solo ciertos elementos de la matriz durante el proceso de factorización, lo que resulta en un cálculo más eficiente. Este tipo de factorización se utiliza con frecuencia como precondicionador en sistemas lineales dispersos de gran escala. Los requerimientos del proyecto son:

- Explicar la factorización de Cholesky incompleta y su importancia para la resolución en sistemas lineales dispersos.
- Mostrar el funcionamiento de la factorización incompleta aplicada a un sistema lineal disperso $Ax = b$.

Metódo de Lanczos

Supongamos que tenemos una matriz $A \in \mathbb{R}^{n\times n}$ que es simétrica, dispersa y de gran tamaño, y que necesitamos aproximar algunos de sus autovalores extremos. El algoritmo de Lanczos busca resolver este problema, y genera para ello una secuencia de matrices tridiagonales $\{T_k\}$ con la propiedad de que los autovalores extremos de dichas matrices son estimaciones progresivamente mejores de los autovalores extremos de la matriz A. Los requerimientos del proyecto son:

- Deducir el método de Lanczos y explicar la convergencia del mismo.
- Implementar el algoritmo de Lanczos y probarlo con matrices aleatorias.