Factorización

Anexos de Factorización

Descomposición Matricial

Varias operaciones complejas matriciales no pueden ser realizadas de manera eficiente. Los métodos de descomposición matricial son métodos que sirven para para reducir una matriz en partes constituyentes que hacen que las operaciones sean mucho más simples. Se utilizan en gran cantidad a la hora de resolver sistemas de ecuaciones, calcular la inversa o la determinante.

Qué es una Descomposición Matricial

En términos generales, podemos pensar en la siguiente analogía, en lugar de pensar en 10, podemos pensar en 2×5. Es por esta precisa razón que también se le conoce a la descomposición matricial como factorización matricial.

Así como con los valores reales, existen varios métodos de descomposición matricial. Dos métodos ampliamente utilizados son los métodos *LU* y el método *QR* que veremos más adelante.

Descomposición LU

La descomposición LU se aplica en matrices cuadradas y descompone una matriz en las matrices L y U, es decir que $\mathrm{A} = \mathrm{L} \cdot \mathrm{U}$. En donde A es la matriz cuadrada que buscamos descomponer, L es la matriz triangular inferior y U es la matriz triangular superior.

La descomposición LU se encuentra utilizando un proceso numérico iterativo y puede llegar a fallar con matrices que son complejas de descomponer. Una variación de esta descomposición que es numéricamente más estable se llama descomposición LUP, o la descomposición LU con pivoteo parcial.

$$\mathrm{A} = \mathrm{L} \cdot \mathrm{U} \cdot \mathrm{P}$$

Los renglones de la matriz original se reordenan para simplificar la descomposición matricial. Aunque también existen otras variaciones del método LU. La descomposición LU se usa frecuentemente para simplificar la resolución de sistemas de ecuaciones lineales, así como encontrar los coeficientes en una regresión lineal, así como para calcular la inversa de la matriz. La muestra de cómo se realiza la descomposición en Python se encuentra en el *ANEXO 1*.

Descomposición QR

La descomposición QR no se ve limitada por la forma de las matrices, es decir, no necesariamente tienen que ser matrices cuadradas. En este caso, se va a descomponer a una matriz como $\mathrm{A} = \mathrm{Q} \cdot \mathrm{R}$. En donde A es la matriz que se busca descomponer de tamaño $\mathrm{n} \times \mathrm{m}$, Q es una matriz de tamaño $\mathrm{m} \times \mathrm{m}$ y R es una matriz de tamaño $\mathrm{m} \times \mathrm{n}$. La descomposición QR se realiza por medio de métodos numéricos iterativos que pueden llegar a fallar para ciertas matrices más complejas de descomponer. Así como con la descomposición LU, se puede utilizar para resolver sistemas de ecuaciones que no sean necesariamente cuadradas. La implementación de este método en Python puede verse en el *ANEXO 1.*

Descomposición Cholesky

La descomposición Cholesky se utiliza para matrices cuadradas simétricas en donde todos los valores son mayores a cero, también conocidas como matrices definidas positivas. Para nuestro interés en Machine Learning, nos concentraremos en la descomposición de Cholesky para matrices con valores reales y no complejos. La descomposición se define como:

$$\mathrm{A} = \mathrm{L} \cdot \mathrm{L}^{\mathrm{T}}$$

En donde A es la matriz que será descompuesta, L es la matriz triangular inferior y L^T es la matriz transpuesta de la matriz triangular inferior. La descomposición también se puede escribir como el producto con matrices triangulares superiores como:

$$\mathrm{A} = \mathrm{U}^{\mathrm{T}} \cdot \mathrm{U}$$

En donde U es la matriz triangular superior. La descomposición de Cholesky se realiza para resolver una regresión lineal con el método de mínimos cuadrados, así como para métodos de optimización y simulación. La descomposición Cholesky es el doble de eficiente que las descomposiciones LU y QR. La muestra de cómo se realiza esta descomposición en Python se encuentra en el *ANEXO 1*.

Eigendescomposición

Las descomposiciones matriciales son herramientas útiles para reducir una matriz a sus partes constituyentes para simplificar las operaciones complejas. El tipo de descomposición matricial más utilizado es el de eigendescomposición que descompone una matriz en sus eigenvectores y eigenvalores. Esta descomposición tiene su rol en métodos utilizados en Machine Learning como el método de PCA.

Eigendescomposición de una Matriz

La eigendescomposición de una matriz es un tipo de descomposición que involucra la descomposición de una matriz cuadrada a un set de eigenvectores y eigenvalores. Un vector es un eigenvector de una matriz si cumple la siguiente ecuación:

$$\mathrm{A} \cdot \mathrm{v} = \lambda \cdot \mathrm{v}$$

A esto se le conoce como la ecuación del eigenvalor, en donde A representa la matriz original que se busca descomponer, v es el eigenvector de la matriz, λ representa el eigenvalor. Una matriz puede tener un eigenvector y un eigenvalor por cada dimensión de la matriz original. No todas las matrices pueden ser descompuestas en eigenvalores y eigenvectores y algunas sólo pueden ser descompuestas con números complejos. La matriz original puede reconstruirse de la siguiente manera:

$$\mathrm{A} = \mathrm{Q} \cdot \Lambda \cdot \mathrm{Q}^{\mathrm{T}}$$

En donde Q es la matriz compuesta de los eigenvectores y Λ es la matriz diagonal que está compuesta por los eigenvalores.

Casi todos los vectores cambian de dirección cuando se multiplican por A, aquellas excepciones son precisamente los eigenvectores. La eigendescomposición se puede utilizar para calcular los componentes principales de una matriz en el método de PCA, que sirve para reducir dimensionalidad de los datos en Machine Learning.

Eigenvectores y Eigenvalores

Los eigenvectores son vectores unitarios y representados como un vector-columna. Por otro lado, los eigenvalores son los coeficientes que le son aplicados a los eigenvectores para darles su magnitud. Una matriz que tiene todos sus eigenvalores positivos, se le conoce como matriz definida positiva. Y en el caso contrario, se le conoce como matriz definida negativa.

Cálculo de la Eigendescomposición

La eigendescomposición se calcula en una matriz cuadrada utilizando un algoritmo iterativo eficiente. A veces, primero se encuentra el eigenvalor y luego el eigenvector asociado. El cálculo por este método en Python se puede ver en el *ANEXO 2*.

Descomposición en Valores Singulares

La descomposición matricial o factorial, involucra describir cierta matriz utilizando sus elementos constituyentes. Tal vez, uno de los métodos más utilizados es el de *Singular-Value Decomposition* o por sus siglas *SVD*. Todas las matrices tienen un *SVD* lo que la hace preferible a otros métodos como los de eigendescomposición. Por ello, se utiliza en gran parte para aplicaciones como para comprimir, eliminar ruido y reducción de datos.

Qué es la Descomposición en Valores Singulares

La descomposición SVD es una descomposición matricial para reducir una matriz a sus componentes constituyentes para realizar operaciones de manera más eficiente. Una descomposición SVD se ve de la manera:

$$\mathrm{A} = \mathrm{U} \cdot \Sigma \cdot \mathrm{V}^{\mathrm{T}}$$

En donde A es una matriz de $\mathrm{n} \times \mathrm{m}$ que buscamos descomponer, U es una matriz de $\mathrm{m} \times \mathrm{m}$, Σ es una matriz diagonal de $\mathrm{m} \times \mathrm{n}$ y T es una matriz de $\mathrm{n} \times \mathrm{n}$.

Los valores diagonales de Σ se conocen como los valores singulares de la matriz original, las columnas de U son los vectores singulares por la izquierda de A y las columnas de V son los vectores singulares por la derecha. Se calcula la *SVD* por medio de métodos numéricos iterativos. Cada matriz rectangular tiene una SVD, aunque en ocasiones, las matrices componentes resultan ser con números complejos. La SVD tiene grandes aplicaciones dentro de la Estadística, Machine Learning y Ciencias de la Computación. La aplicación de la descomposición se encontrará en el *ANEXO 3*.

Pseudoinversa

La seudo inversa es la generalización de las matrices inversas cuadradas a las matrices rectangulares en donde el número de columnas no coinciden con el número de renglones. También se le conoce con el nombre de la inversa de Moore-Penrose.

La pseudoinversa se denota como A^+, en donde A es la matriz que queremos invertir. La Pseudoinversa se calcula utilizando la descomposición en valores singulares de A:

$$A^+ = V \cdot D^+ \cdot U^T$$

En donde D^+ es la Pseudoinversa de la matriz Σ. La inversa de Σ se calcula fácilmente al tomar el recíproco de cada elemento que se encuentre en la diagonal y tomar la transpuesta si la matriz no es cuadrada. En el *ANEXO 3* se muestra un ejemplo de cómo se hace en Python.

Reducción de Dimensionalidad

Una aplicación popular de la SVD es la reducción de dimensionalidad. Si se cuenta con datos que contienen un gran número de características, como más columnas que renglones, se puede reducir el número de características a aquellas que sean más importantes para resolver el problema. El resultado es una matriz con un rango menor que se dice que se aproxima a la matriz original. Para hacer esto, se le aplica un SVD a los datos originales y se seleccionan los k valores singulares más altos de Σ. De manera que se seleccionan los renglones correspondientes de V^T, de manera que se puede construir una matriz aproximada de la siguiente manera:

$$B = U \cdot \Sigma_k \cdot V_k^T$$

En el procesamiento de lenguaje natural, se utiliza esta metodología y se le conoce como Análisis Latente de Semántica. En la práctica, podemos quedarnos y trabajar con un subconjunto de los datos T. Esto es un resumen o proyección de los datos:

$$T = U \cdot \Sigma_k$$

Incluso, se le puede realizar esta transformación a la matriz original:

$$T = A \cdot V_k^T$$

En el *ANEXO 4* se puede ver un ejemplo de esto.