



Taller: Enseñanza del álgebra lineal basada en el uso del paquete *VilGebra*

Máster Enrique **Vílchez** Quesada
Escuela de Informática, Universidad Nacional
Costa Rica
enrique.vilchez.quesada@una.ac.cr

Resumen

El taller pretende mostrar la funcionalidad de un paquete elaborado por el autor de esta propuesta, titulado “*VilGebra*”, con la intención de trabajar en el campo del álgebra lineal utilizando como apoyo el conocido software comercial *Mathematica*. *VilGebra* desarrolla una serie de funciones que por defecto no están integradas en *Mathematica* en las principales áreas de contenido de un curso clásico de álgebra lineal para ingeniería, es decir: ecuaciones lineales y matrices, determinantes, vectores, rectas y planos, espacios vectoriales, proyecciones ortogonales, transformaciones lineales, diagonalización de matrices y programación lineal. Desde un punto de vista didáctico, el paquete puede ser utilizado como una herramienta de verificación de resultados, o bien, como un medio para profundizar el ambiente de programación del software, implementando métodos de solución automatizados en un campo de conocimiento, muchas veces catalogado como abstracto.

Palabras clave: enseñanza, aprendizaje, álgebra, lineal, software.

Introducción

El paquete *VilGebra* ha sido desarrollado como un medio de resolución de problemas vinculados con el álgebra lineal a través del uso de software.

En la experiencia docente acumulada por el autor de este taller impartiendo cursos de álgebra lineal con una metodología asistida por computadora, surgió la necesidad de contar con una serie de funciones no incluidas por defecto en *Mathematica*, que implementaran una serie de procesos tradicionales como técnicas de resolución a problemas frecuentes en este campo.

Prerrequisitos

Manejo básico del software *Mathematica*.

Objetivos

- Mostrar el paquete *VilGebra* como medio de resolución de problemas vinculados con el área cognitiva del álgebra lineal.
- Comprender el contenido de cada una de las funciones integradas en *VilGebra*.
- Utilizar *VilGebra* como un recurso de resolución de problemas tradicionales a través del uso de software.
- Analizar las aplicaciones del paquete *VilGebra* desde un punto de vista didáctico.

Contenido

Se desarrollará en el taller ejemplos de uso de distintos comandos integrados en el paquete *VilGebra* en los ejes de contenido principales de un curso de álgebra lineal para ingeniería. Dentro de las funciones más importantes implementadas en *VilGebra* se encuentran:

1. IntercambiaFila: intercambia dos filas en una matriz.
2. MultiFila: multiplica un escalar a una fila de una matriz.
3. SumaFila: suma el múltiplo escalar de una fila en una matriz a otra fila dada.
4. RowReduceComplete: función que determina paso a paso la matriz escalonada de otra, recibida como parámetro.
5. InversaSP: encuentra paso a paso la matriz inversa de otra que no contiene ningún parámetro.
6. InversaCP: encuentra paso a paso la matriz inversa de otra que contiene parámetros.
7. Determinante: función que calcula el determinante de una matriz cuadrada desarrollando por cofactores de acuerdo a los elementos de la primera fila.
8. InversaAdjunta: calcula la inversa por el método de la adjunta donde cada uno de los cofactores son hallados a pie.
9. LinearSolveCompleteSP: resuelve un sistema de ecuaciones lineales sin parámetros paso a paso.
10. LinearSolveCompleteCCP: resuelve un sistema de ecuaciones lineales cuadrado con parámetros paso a paso.
11. Cramer: resuelve un sistema de ecuaciones lineales cuadrado a partir de la regla de *Cramer*.
12. CombiLinealQ: determina si un vector es combinación lineal de un conjunto de vectores dados, retornando True o False en cada caso.
13. LiLd: establece si un conjunto de vectores son linealmente independientes o linealmente dependientes.
14. GraficaVectores2D: grafica vectores en el plano.

15. *GraficaPuntos2D*: grafica puntos en el plano, los vectores se añaden como filas de una matriz $n \times 2$.
16. *GraficaVectores3D*: grafica vectores en el espacio tridimensional.
17. *GraficaPuntos3D*: grafica puntos en el espacio, los vectores se añaden como filas de una matriz $n \times 3$.
18. *VectoresParalelosQ*: determina si dos vectores son paralelos retornando True o False según corresponda.
19. *ProducMixto*: calcula el producto mixto entre tres vectores con tres componentes.
20. *AreaParalelogramo2*: calcula el área de un paralelogramo formado por dos vectores.
21. *AreaTriangulo2*: calcula el área de un triángulo formado por dos vectores.
22. *AreaParalelogramo3*: calcula el área de un paralelogramo formado por tres vectores, uno de ellos funciona como anclado.
23. *AreaTriangulo3*: calcula el área de un triángulo formado por tres vectores, uno de ellos funciona como anclado.
24. *VolumenParalelepipedo3*: calcula el volumen de un paralelepípedo formado por tres vectores.
25. *VolumenTetraedro3*: calcula el volumen de un tetraedro formado por tres vectores.
26. *VolumenParalelepipedo4*: calcula el volumen de un paralelepípedo formado por cuatro vectores.
27. *VolumenTetraedro4*: calcula el volumen de un tetraedro formado por cuatro vectores.
28. *PerteneceRectaQ*: determina si un punto Q pertenece a una recta, dado un punto P y su vector director.
29. *ColinealesQ*: retorna True si tres puntos recibidos como parámetros son colineales y False en caso contrario.
30. *EcuacRectaPuntos*: encuentra la ecuación vectorial de una recta dados dos puntos n -dimensionales.
31. *NSecar*: encuentra las coordenadas de los puntos que n -secan a un segmento de recta en cualquier dimensión.
32. *NSecar2D*: encuentra las coordenadas de los puntos que n -secan a un segmento de recta en el plano con extremos P y Q, y grafica el segmento y los puntos.
33. *NSecar3D*: encuentra las coordenadas de los puntos que n -secan a un segmento de recta en el espacio con extremos P y Q, y grafica el segmento y los puntos.
34. *EcuacVecToSime*: encuentra las ecuaciones simétricas o ecuación simétrica de una recta dada su ecuación vectorial.
35. *EcuacSimeToVec*: encuentra la ecuación vectorial de una recta dadas sus ecuaciones simétricas o ecuación simétrica.

36. *DistanciaP1Recta*: determina la distancia de un punto a una recta, dada la recta a través de un punto y su vector director.
37. *DistanciaP2Recta*: determina la distancia de un punto a una recta, dada la recta por medio de dos puntos.
38. *GraficaRectas2D*: grafica un conjunto de rectas en el plano dadas sus ecuaciones paramétricas.
39. *InterRectas2D*: encuentra el punto de intersección entre dos rectas en el plano cartesiano y grafica el punto y las rectas.
40. *GraficaRectas3D*: grafica un conjunto de rectas en el espacio dadas sus ecuaciones paramétricas.
41. *InterRectas3D*: encuentra el punto de intersección entre dos rectas en el espacio y grafica el punto y las rectas.
42. *GraficaRectasPuntos2D*: grafica un conjunto de rectas y puntos en el plano cartesiano, dadas sus ecuaciones paramétricas y la lista de vectores respectivamente.
43. *GraficaRectasPuntos3D*: grafica un conjunto de rectas y puntos en el espacio, dadas sus ecuaciones paramétricas y la lista de vectores respectivamente.
44. *PuntoRectaQ*: retorna True si un punto “ n ” dimensional pertenece a una recta L en dicha dimensión y False en caso contrario.
45. *RectaPuntos*: encuentra “ n ” puntos que pertenecen a una recta dada.
46. *GraficaPlanos1*: grafica un conjunto de planos dadas sus ecuaciones cartesianas.
47. *GraficaPlanos2*: grafica un conjunto de planos dado un sistema de ecuaciones lineales.
48. *PuntoPlanoQ*: retorna True si un punto “ n ” dimensional pertenece a un plano en dicha dimensión y False en caso contrario.
49. *PlanoPuntos*: encuentra “ n ” puntos que pertenecen a un plano dado.
50. *GraficaPlanosPuntos*: grafica un conjunto planos y puntos, dadas sus ecuaciones cartesianas y la lista de vectores respectivamente.
51. *GraficaRectasPlanos*: grafica un conjunto de rectas y planos dadas sus ecuaciones paramétricas y cartesianas respectivamente.
52. *GraficaRectasPlanosPuntos*: grafica un conjunto de rectas, planos y puntos, dadas sus ecuaciones paramétricas, cartesianas y la lista de vectores respectivamente.
53. *EcuacCartPuntoGene*: retorna la ecuación cartesiana de un plano en el espacio, conociendo un punto y sus vectores generadores.
54. *EcuacCartPuntos*: retorna la ecuación cartesiana de un plano en el espacio, conociendo tres puntos del plano no colineales.
55. *RectaPlanos*: determina la ecuación vectorial de una recta de intersección entre dos planos en la dimensión 3 y dibuja la recta y los planos.
56. *DistanciaPuntoPlano*: determina la distancia de un punto a un plano en la tercera dimensión.

57. DistanciaPlanos: encuentra la distancia entre dos planos paralelos.
58. CompletandoBase: completa una familia libre para formar una base de un espacio vectorial de dimensión finita.
59. VectorCoordenadas: determina el vector de coordenadas de uno dado, recibiendo la base correspondiente.
60. MGS: aplica paso a paso el método de *Gram Schmidt*.
61. PrOt: calcula la proyección ortogonal de un vector sobre un subespacio de \mathbb{R}^n .
62. ComplementOt: calcula el complemento de un vector ortogonal a un subespacio de \mathbb{R}^n .
63. TLQ: devuelve True si dada una función ella constituye una transformación lineal y False en caso contrario, si la función no es aplicación lineal brinda un contraejemplo.
64. Nucleo: determina como un conjunto generado el núcleo de una transformación lineal de \mathbb{R}^n a \mathbb{R}^m . Además, indica si la aplicación es inyectiva.
65. Imagen: determina como un conjunto generado la imagen de una transformación lineal de \mathbb{R}^n a \mathbb{R}^m . Además, indica si la aplicación es sobreyectiva.
66. TL: encuentra el criterio de una aplicación lineal dadas algunas imágenes cuyas preimágenes forman una base del dominio.
67. MatrizRepreTL: determina la matriz representativa de una transformación lineal.
68. TLMatrizRepre: encuentra el criterio de una aplicación lineal dada una matriz representativa.
69. MatrizPasaje: encuentra una matriz de pasaje o de cambio de base.
70. TLComposicion: determina el criterio de la composición entre dos transformaciones lineales en \mathbb{R}^n .
71. MatrizRepreTLCB: aplica el teorema de cambio de bases.
72. TLInversa: determina si una aplicación lineal es invertible y encuentra su criterio, siempre y cuando la dimensión del espacio vectorial dominio sea igual a la dimensión del espacio vectorial codominio.
73. DiagonalizacionQ: retorna True si una matriz cuadrada es diagonalizable y False en caso contrario.
74. Diagonalizacion: diagonaliza una matriz diagonalizable.
75. DiagonalizacionOrtogonal: diagonaliza ortogonalmente una matriz simétrica.
76. PotenciaNMatriz: calcula la potencia n -ésima de una matriz diagonalizable.
77. EigensystemOL: función Eigensystem para un operador lineal.
78. DiagonalizacionFC2D: diagonaliza una forma cuadrática en el plano.
79. DiagonalizacionFC3D: diagonaliza una forma cuadrática en el espacio.

80. PLMG1: resuelve un problema de programación lineal utilizando el método gráfico siempre y cuando la región factible sea acotada. Recibe la función objetivo y las restricciones de manera directa.
81. PLMG2: resuelve un problema de programación lineal utilizando el método gráfico. La función objetivo y las restricciones se reciben como matrices.
82. OptiFunc1: optimiza una función lineal recurriendo al comando LinearProgramming de *Mathematica*.
83. OptiFunc2: optimiza una función lineal recurriendo a los comandos Maximize y Minimize de *Mathematica*.
84. SimplexVH: aplica el método símplex para maximizar o minimizar una función lineal donde no existen variables artificiales (solo de holgura) y soluciones degeneradas. No resuelve problemas con soluciones múltiples.
85. SimplexVA: aplica el método símplex para maximizar o minimizar una función lineal donde existen variables artificiales, sin soluciones degeneradas. No resuelve problemas con soluciones múltiples.
86. Dual: retorna el dual del problema primal ingresado.

Metodología

La metodología se fundamentará en una explicación magistral de las principales características del paquete *VilGebra* y la resolución por parte de los participantes de distintos ejemplos de ejecución. Se entregará un CD con los contenidos necesarios para el desarrollo de cada una de las prácticas.

Conclusiones

El paquete *VilGebra* representa un esfuerzo docente con miras a sistematizar una metodología asistida por computadora en cursos de álgebra lineal para ingeniería. Su aporte principal reside en dotar al software *Mathematica* de una serie de nuevos comandos en un área específica de estudio, compartida por el currículo de distintas carreras universitarias en todo Latinoamérica.

Bibliografía y referencias

- Arce, C, Castillo, W., & González, J. (2004). *Álgebra lineal*. San José: Editorial de la Universidad de Costa Rica.
- Kolman, B. (1997). *Álgebra lineal con aplicaciones y Matlab*. México: Editorial Pearson.
- Shiskowski, K., & Frinkle, K. (2011). *Principles of Linear Algebra with Mathematica*. USA: Editorial Wiley.
- Vílchez, E. (2012). *Álgebra lineal apoyada con Mathematica*. Costa Rica: Editorial Tecnológica.