

Proyecto Final SIC

**Red neuronal para clasificación de literatura
científica**

José Miguel López Zuluaga
Julián David Pulgarín Pirazan
Profesor: Néstor Darío Duque
Fecha: 01 de junio de 2023

Manizales, Caldas
Universidad Nacional De Colombia

Resumen

La inteligencia artificial (IA) ha transformado la manera en que analizamos y procesamos grandes volúmenes de datos en diferentes áreas. En este proyecto, se propone el uso de una red neuronal de IA para abordar el desafío de clasificar la literatura científica. La clasificación de artículos científicos es fundamental en el ámbito académico e investigativo, ya que permite identificar y seleccionar los estudios relevantes para un tema específico.

El objetivo principal del proyecto es desarrollar una red neuronal capaz de clasificar automáticamente la literatura científica en categorías específicas, como raíz, tallo o hojas. La red neuronal utilizará técnicas de aprendizaje automático supervisado y se entrenará con un conjunto de datos etiquetados, donde cada artículo científico está asociado a una categoría particular. Una vez entrenada, la red neuronal podrá clasificar nuevos artículos científicos en las categorías correspondientes.

Para lograr este objetivo, se utilizará una red neuronal multicapa (MLP) con una estructura "n:n:3", donde "n" representa el número de características de los artículos científicos. La MLP consta de una capa de entrada, una capa oculta y una capa de salida con tres nodos, representando las categorías raíz, tallo y hojas. Se empleará la biblioteca de aprendizaje automático en Python para implementar la red neuronal y entrenarla con los datos disponibles.

Palabras clave

inteligencia artificial, IA, red neuronal, clasificación, literatura científica, aprendizaje automático supervisado, conjunto de datos etiquetados, categorías, raíz, tallo, hojas, MLP, estructura "n:n:3", biblioteca de aprendizaje automático, Python.

índice

1. Introducción

2. Objetivo general

3. Marco teórico

4. Pantallazos

5. Conclusiones

6. Referencias Bibliográficas

Introducción

La inteligencia artificial (IA) ha revolucionado la forma en que analizamos y procesamos grandes cantidades de datos en diversos campos. Su aplicación se extiende a sectores como la medicina, la industria, el transporte y la educación, entre otros. En particular, la IA ha mostrado un potencial significativo en el ámbito académico y de investigación al abordar el desafío de clasificar la literatura científica.

La clasificación de artículos científicos es un proceso crucial para los profesionales y académicos, ya que les permite identificar y seleccionar los estudios relevantes para un tema específico de investigación. Sin embargo, debido al creciente volumen de publicaciones científicas, se ha vuelto cada vez más difícil y laborioso llevar a cabo esta tarea de manera manual.

Con el objetivo de superar este desafío, en este proyecto se propone utilizar técnicas de IA, específicamente una red neuronal, para desarrollar un sistema automatizado de clasificación de literatura científica. La red neuronal se entrenará utilizando técnicas de aprendizaje automático supervisado, utilizando un conjunto de datos etiquetados donde cada artículo científico esté asociado con una categoría o tema específico.

El propósito principal de este proyecto es diseñar una red neuronal capaz de clasificar automáticamente los artículos científicos en categorías como raíz, tallo o hojas. Esta clasificación permitirá a los investigadores y académicos acceder de manera más eficiente a los estudios relevantes para sus investigaciones, ahorrando tiempo y esfuerzo en el proceso.

Para lograr este objetivo, se utilizará una red neuronal multicapa (MLP) con una estructura "n:n:3", donde "n" representa el número de características de los artículos científicos. La biblioteca de aprendizaje automático en Python se empleará para implementar la red neuronal y entrenarla con los datos disponibles.

En resumen, este proyecto busca aprovechar los avances en inteligencia artificial para simplificar y agilizar la clasificación de la literatura científica. La aplicación de técnicas de aprendizaje automático y el uso de una red neuronal permitirán a los investigadores y académicos acceder de manera más rápida y precisa a los estudios relevantes, fomentando así el avance del conocimiento en diversas áreas científicas.

Objetivo general

El objetivo principal de este proyecto es desarrollar una red neuronal que pueda clasificar automáticamente la literatura científica dando como salida si el artículo analizado está ubicado en la raíz, el tallo o las hojas. La red neuronal utilizará técnicas de aprendizaje automático supervisado para entrenarse con un conjunto de datos etiquetados, donde cada artículo científico está asociado con una categoría o tema específico. Una vez entrenada, la red neuronal será capaz de clasificar nuevos artículos científicos en las categorías adecuadas.

Marco Teórico

Para lograr este objetivo, se utilizará una red neuronal multicapa (MLP, por sus siglas en inglés). La MLP es un tipo de red neuronal artificial compuesta por múltiples capas de nodos interconectados. En este caso, la estructura de la red neuronal será de "n:n:3", lo que significa que tendrá una capa de entrada con "n" nodos (correspondientes a las características de los artículos científicos), una capa oculta con "n" nodos y finalmente una capa de salida con 3 nodos (representando si es raíz, tallo, o hojas). dando paso a la clusterización, Se utilizará la biblioteca de aprendizaje automático en Python para implementar la red neuronal y entrenarla con los datos disponibles.

SITUACIÓN A RESOLVER

La aplicación de técnicas de IA en analítica de datos, genera un impacto disruptivo en orden de establecer nuevas metodologías para el desarrollo de múltiples actividades.

En este caso se pretende desarrollar un modelo que permita la selección de artículos científicos y de literatura por medio de una red neuronal

TÉCNICA DE IA QUE SE UTILIZARA:

-Neural network for Clustering in Python

-MLP

-Estructura de red: → n:n:3

Por lo general, un perceptrón multicapa (MLP) no se usa para agrupar porque MLP es un algoritmo de aprendizaje supervisado, lo que significa que está diseñado para aprender de un conjunto de datos etiquetados.

Por el contrario, la agrupación en clústeres es un problema de aprendizaje no supervisado, en el que se intenta agrupar datos sin etiquetar.

Sin embargo, un enfoque posible para agrupar usando MLP es usar un codificador Aporte:

Una vez que se ha entrenado un codificador automático, la idea es usar la parte del codificador para mapear sus datos en un espacio dimensional más bajo donde puede usar un algoritmo de agrupamiento estándar (como K-means) o usar alguna regla para definir los clústeres.

Aquí hay un ejemplo simple de cómo puede hacer esto con Keras (deberá tener instalado Tensor Flow y Keras) automático. Los codificadores automáticos son un tipo de red neuronal que están capacitados para copiar su entrada a su salida. Tienen una capa interna (oculta) que describe un código utilizado para representar la entrada y constan de dos partes principales: un codificador, que asigna la entrada al código, y un decodificador, que asigna el código a una reconstrucción del original.

RESULTADOS (Herramienta)

Los resultados esperados de este proyecto se basan en la implementación de una red neuronal multicapa (MLP) para clasificar automáticamente los artículos científicos en las categorías de raíz, tallo o hojas. La red neuronal se entrenará utilizando técnicas de aprendizaje automático supervisado y se utilizará un conjunto de datos etiquetados para este propósito.

Después de completar el entrenamiento de la red neuronal con el conjunto de datos, se espera que pueda realizar predicciones precisas sobre la categoría a la que pertenece un artículo científico dado. Esto se logrará proporcionando el artículo como entrada a la red neuronal, y la red generará una salida que indica si el artículo se clasifica como raíz, tallo u hoja.

Estructura del Dataset trabajado

Artículo	PY	CR	TC	SO	DT	Clase
Antiguo						RAÍZ
Medio						TRONC O
Nuevo						HOJAS

- **normalizar datos!!!**

PY (año) → { 3 [Min, 2000], → 2 [2000, 2013], → 1 [2013, 2018], → 0 [2018,2023] }

Sum → CR (citaciones salidas): → CITACIONES SALIDA { }

TC → (Número de citaciones): ← CITACIONES ENTRADA { }

Journal SO (Q) → Impact: {3-[Q1],2-[Q2],1-[Q3],1-[Q4], 0-[NA]} → “scimago2020.csv”--> Ranking SJR

DT → : { 3[Article], 2[review], 1[otros] }

Conclusiones

Referencias Bibliográficas

- 1. Bayat, F. M., Prezioso, M., Chakrabarti, B., Nili, H., Kataeva, I., & Strukov, D. (2018). Implementation of multilayer perceptron network with highly uniform passive memristive crossbar circuits. *Nature Communications*, 9(1), 2331. doi: 10.1038/s41467-018-04482-4**
- 2. Finne, P., Finne, R., Auvinen, A., Juusela, H., Aro, J., Määttänen, L., Hakama, M., Rannikko, S., Tammela, T. L. J., & Stenman, U.-H. (2000). Predicting the outcome of prostate biopsy in screen-positive men by a multilayer perceptron network. *Urology*, 56(6), 1018-1022. doi: 10.1016/S0090-4295(00)00672-5**
- 3. Parlos, A. G., Fernandez, B., Atiya, A. F., Muthusami, J., & Tsai, W. K. (1994). An accelerated learning algorithm for multilayer perceptron networks. *IEEE Transactions on Neural Networks*, 5(4), 563-578. doi: 10.1109/72.286921**
- 4. Bello, M. G. (1992). Enhanced training algorithms, and integrated training/architecture selection for multilayer perceptron networks. *IEEE Transactions on Neural Networks*, 3(5), 784-793. doi: 10.1109/72.165589**
- 5. Mat Isa, N. A., & Mamat, W. M. F. W. (2011). Clustered-hybrid multilayer perceptron network for pattern recognition application. *Applied Soft Computing*, 11(8), 5634-5643. doi: 10.1016/j.asoc.2010.04.017**

6. Yun, J., Park, J. E., Lee, H., Ham, S., Kim, N., & Kim, H. S. (2019). Radiomic features and multilayer perceptron network classifier: A robust MRI classification strategy for distinguishing glioblastoma from primary central nervous system lymphoma. Scientific Reports, 9(1), 4826. doi: 10.1038/s41598-019-42276-w

7. Fukumizu, K. (1996). A regularity condition of the information matrix of a multilayer perceptron network. Neural Networks, 9(1), 19-26. doi: 10.1016/0893-6080(95)00119-0

8. Su, H.-T., & McAvoy, T. J. (1993). Integration of multilayer perceptron networks and linear dynamic models: A Hammerstein modeling approach. Industrial & Engineering Chemistry Research, 32(12), 3154-3162. doi: 10.1021/ie00021a017

9. Ahmad, F., Mat Isa, N. A., Hussain, Z., & Osman, M. K. (2013). Intelligent medical disease diagnosis using improved hybrid genetic algorithm - Multilayer perceptron network. Journal of Medical Systems, 37(3), 9920. doi: 10.1007/s10916-013-9934-7

10. Xu, Y.G., Liu, G.R., Wu, Z.P., & Huang, X.M. (2001). Adaptive multilayer perceptron networks for detection of cracks in anisotropic laminated plates. Composite Structures, 54(3), 345-353. doi: 10.1016/S0020-7683(00)03681-1

11. Soares, F.A.A.M.N., Flôres, E.L., Cabacinha, C.D., Carrijo, G.A., & Veiga, A.C.P. (2011). Recursive diameter prediction and volume calculation of eucalyptus trees using multilayer perceptron networks. Computers and Electronics in Agriculture, 75(2), 287-294. doi: 10.1016/j.compag.2011.05.008

12. García Nieto, P.J., Martínez Torres, J., De Cos Juez, F.J., & Sánchez Lasheras, F. (2012). Using multivariate adaptive regression splines and multilayer perceptron networks to evaluate paper manufactured using Eucalyptus globulus. Applied Mathematical Modelling, 36(11), 5220-5228. doi: 10.1016/j.amc.2012.07.001

13. Abderrahim, H., Chellali, M.R., & Hamou, A. (2016). Forecasting PM10 in Algiers: Efficacy of multilayer perceptron networks. Journal of Environmental Engineering and Landscape Management, 24(4), 265-276. doi: 10.3846/16486897.2016.1175049

14. Vilán Vilán, J.A., Alonso Fernández, J.R., García Nieto, P.J., Sánchez Lasheras, F., de Cos Juez, F.J., & Díaz Muñoz, C. (2013). Support Vector Machines and Multilayer Perceptron Networks Used to Evaluate the Cyanotoxins Presence from Experimental Cyanobacteria Concentrations in the Trasona Reservoir (Northern Spain). Water, Air, & Soil Pollution, 224(5), 1545. doi: 10.1007/s11270-013-1545-4

15. Gallagher, M., & Downs, T. (2003). Visualization of learning in multilayer perceptron networks using principal component analysis. IEEE Transactions on Neural Networks, 14(3), 669-674. doi: 10.1109/TSMCB.2003.808183

16. Polikar, R., Udpa, L., Udpa, S.S., & Honavar, V. (2000). LEARN++: An incremental learning algorithm for multilayer perceptron networks. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 30(4), 497-508. doi: 10.1109/5326.877215

17. Hametner, C., Stadlbauer, M., Deregnaucourt, M., Jakubek, S., & Winsel, T. (2013). Optimal experiment design based on local model networks and multilayer perceptron networks. Industrial & Engineering Chemistry Research, 52(47), 16735-16742. doi: 10.1021/ie101513z

18. Raji, C.G., & Vinod Chandra, S.S. (2017). Long-term forecasting the survival in liver transplantation using multilayer perceptron networks. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 47(6), 1577-1586. doi: 10.1109/TSMC.2017.2661996

19. Neto, N.F.S., Stefenon, S.F., Meyer, L.H., Bruns, R., Nied, A., Seman, L.O., Gonzalez, G.V., & Leithardt, V.R.Q. (2021). A study of multilayer perceptron networks applied to classification of ceramic insulators using ultrasound. *Applied Acoustics*, 179, 107997. doi: 10.1016/j.apacoust.2021.107997

20. Eslamloueyan, R., Khademi, M.H., & Mazinani, S. (2011). Using a multilayer perceptron network for thermal conductivity prediction of aqueous electrolyte solutions. *Engineering Applications of Artificial Intelligence*, 24(8), 1407-1414. doi: 10.1016/j.engappai.2012.05.016

Proyecto Final SIC

Red neuronal para clasificación de literatura científica

José Miguel López Zuluaga

Julián David Pulgarín Pirazan

Profesor: Néstor Darío Duque

Fecha: 01 de junio de 2023

Manizales, Caldas

Universidad Nacional De Colombia

Resumen

La inteligencia artificial (IA) ha transformado la manera en que analizamos y procesamos grandes volúmenes de datos en diferentes áreas. En este proyecto, se propone el uso de una red neuronal de IA para abordar el desafío de clasificar la literatura científica. La clasificación de artículos científicos es fundamental en el ámbito académico e investigativo, ya que permite identificar y seleccionar los estudios relevantes para un tema específico.

El objetivo principal del proyecto es desarrollar una red neuronal capaz de clasificar automáticamente la literatura científica en categorías específicas, como raíz, tallo o hojas. La red neuronal utilizará técnicas de aprendizaje automático supervisado y se entrenará con un conjunto de datos etiquetados, donde cada artículo científico está asociado a una categoría particular. Una vez entrenada, la red neuronal podrá clasificar nuevos artículos científicos en las categorías correspondientes.

Para lograr este objetivo, se utilizará una red neuronal multicapa (MLP) con una estructura "n:n:3", donde "n" representa el número de características de los artículos científicos. La MLP constará de una capa de entrada, una capa oculta y una capa de salida con tres nodos, representando las categorías raíz, tallo y hojas. Se empleará la biblioteca de aprendizaje automático en Python para implementar la red neuronal y entrenarla con los datos disponibles.

Palabras clave

inteligencia artificial, IA, red neuronal, clasificación, literatura científica, aprendizaje automático supervisado, conjunto de datos etiquetados, categorías, raíz, tallo, hojas, MLP, estructura "n:n:3", biblioteca de aprendizaje automático, Python.

índice

- 1. Introducción**
- 2. Objetivo general**
- 3. Marco teórico**
- 5. Diseños y fotografías**
- 6. Conclusiones**
- 7. Referencias Bibliográficas**

Introducción

La inteligencia artificial (IA) ha revolucionado la forma en que analizamos y procesamos grandes cantidades de datos en diversos campos. Su aplicación se extiende a sectores como la medicina, la industria, el transporte y la educación, entre otros. En particular, la IA ha mostrado un potencial significativo en el ámbito académico y de investigación al abordar el desafío de clasificar la literatura científica.

La clasificación de artículos científicos es un proceso crucial para los profesionales y académicos, ya que les permite identificar y seleccionar los estudios relevantes para un tema específico de investigación. Sin embargo, debido al creciente volumen de publicaciones científicas, se ha vuelto cada vez más difícil y laborioso llevar a cabo esta tarea de manera manual.

Con el objetivo de superar este desafío, en este proyecto se propone utilizar técnicas de IA, específicamente una red neuronal, para desarrollar un sistema automatizado de clasificación de literatura científica. La red neuronal se entrenará utilizando técnicas de aprendizaje automático supervisado, utilizando un conjunto de datos etiquetados donde cada artículo científico esté asociado con una categoría o tema específico.

El propósito principal de este proyecto es diseñar una red neuronal capaz de clasificar automáticamente los artículos científicos en categorías como raíz, tallo o hojas. Esta clasificación permitirá a los investigadores y académicos acceder de manera más eficiente a los estudios relevantes para sus investigaciones, ahorrando tiempo y esfuerzo en el proceso.

Para lograr este objetivo, se utilizará una red neuronal multicapa (MLP) con una estructura "n:n:3", donde "n" representa el número de características de los artículos científicos. La biblioteca de aprendizaje automático en Python se empleará para implementar la red neuronal y entrenarla con los datos disponibles.

En resumen, este proyecto busca aprovechar los avances en inteligencia artificial para simplificar y agilizar la clasificación de la literatura científica. La aplicación de técnicas de aprendizaje automático y el uso de una red neuronal permitirán a los investigadores y académicos acceder de manera más rápida y precisa a los estudios relevantes, fomentando así el avance del conocimiento en diversas áreas científicas.

Objetivo general

El objetivo principal de este proyecto es desarrollar una red neuronal que pueda clasificar automáticamente la literatura científica dando como salida si el artículo analizado está ubicado en la raíz, el tallo o las hojas. La red neuronal utilizará técnicas de aprendizaje automático supervisado para entrenarse con un conjunto de datos etiquetados, donde cada artículo científico está asociado con una categoría o tema específico. Una vez entrenada, la red neuronal será capaz de clasificar nuevos artículos científicos en las categorías adecuadas.

Marco Teórico

Para lograr este objetivo, se utilizará una red neuronal multicapa (MLP, por sus siglas en inglés). La MLP es un tipo de red neuronal artificial compuesta por múltiples capas de nodos interconectados. En este caso, la estructura de la red neuronal será de "n:n:3", lo que significa que tendrá una capa de entrada con "n" nodos (correspondientes a las características de los artículos científicos), una capa oculta con "n" nodos y finalmente una capa de salida con 3 nodos (representando si es raíz, tallo, o hojas). dando paso a la clusterización, Se utilizará la biblioteca de aprendizaje automático en Python para implementar la red neuronal y entrenarla con los datos disponibles.

SITUACIÓN A RESOLVER:

La aplicación de técnicas de IA en analítica de datos, genera un impacto disruptivo en orden de establecer nuevas metodologías para el desarrollo de múltiples actividades.

En este caso se pretende desarrollar un modelo que permita la selección de artículos científicos y de literatura por medio de una red neuronal

TÉCNICA DE IA QUE SE UTILIZARA:

-Neural network for Clustering in Python

-MLP

-Estructura de red: → n:n:3

Por lo general, un perceptrón multicapa (MLP) no se usa para agrupar porque MLP es un algoritmo de aprendizaje supervisado, lo que significa que está diseñado para aprender de un conjunto de datos etiquetados.

Por el contrario, la agrupación en clústeres es un problema de aprendizaje no supervisado, en el que se intenta agrupar datos sin etiquetar.

Sin embargo, un enfoque posible para agrupar usando MLP es usar un codificador Aporte:

Una vez que se ha entrenado un codificador automático, la idea es usar la parte del codificador para mapear sus datos en un espacio dimensional más bajo donde puede usar un algoritmo de agrupamiento estándar (como K-means) o usar alguna regla para definir los clústeres.

Aquí hay un ejemplo simple de cómo puede hacer esto con Keras (deberá tener instalado Tensor Flow y Keras) automático. Los codificadores automáticos son un tipo de red neuronal que están capacitados para copiar su entrada a su salida. Tienen una capa interna (oculta) que describe un código utilizado para representar la entrada y constan de dos partes principales: un codificador, que asigna la entrada al código, y un decodificador, que asigna el código a una reconstrucción del original.

RESULTADOS (Herramienta)

Los resultados esperados de este proyecto se basan en la implementación de una red neuronal multicapa (MLP) para clasificar automáticamente los artículos científicos en las categorías de raíz, tallo o hojas. La red neuronal se entrenará utilizando técnicas de aprendizaje automático supervisado y se utilizará un conjunto de datos etiquetados para este propósito.

Después de completar el entrenamiento de la red neuronal con el conjunto de datos, se espera que pueda realizar predicciones precisas sobre la categoría a la que pertenece un artículo científico dado. Esto se logrará proporcionando el artículo como entrada a la red neuronal, y la red generará una salida que indica si el artículo se clasifica como raíz, tallo o hoja.

Estructura del Data set trabajado

Artículo	PY	CR	TC	SO	DT	Clase
Antiguo						RAIZ
Medio						TRONC O
Nuevo						HOJAS

- **normalizar datos!!!**

**PY (año) → { 3 [Min, 2000], → 2
[2000, 2013], → 1 [2013, 2018], → 0
[2018,2023] }**

**Sum → CR (citaciones salidas): →
CITACIONES SALIDA { }**

**TC → (Número de citaciones): ←
CITACIONES ENTRADA { }**

**Journal SO (Q) → Impact:
{3-[Q1],2-[Q2],1-[Q3],1-[Q4], 0-[NA]}
→ “scimago2020.csv”--> Ranking
SJR**

**DT → : { 3[Article], 2[review],
1[otros] }**

Conclusiones

Referencias Bibliográficas

A continuación, se presenta la cita en formato APA para cada fila de la tabla:

- 1. Bayat, F. M., Prezioso, M., Chakrabarti, B., Nili, H., Kataeva, I., & Strukov, D. (2018). Implementation of multilayer perceptron network with highly uniform passive memristive crossbar circuits. *Nature Communications*, 9(1), 2331. doi: 10.1038/s41467-018-04482-4**
- 2. Finne, P., Finne, R., Auvinen, A., Juusela, H., Aro, J., Määttänen, L., Hakama, M., Rannikko, S., Tammela, T. L. J., & Stenman, U.-H. (2000). Predicting the outcome of prostate biopsy in screen-positive men by a multilayer perceptron network. *Urology*, 56(6), 1018-1022. doi: 10.1016/S0090-4295(00)00672-5**
- 3. Parlos, A. G., Fernandez, B., Atiya, A. F., Muthusami, J., & Tsai, W. K. (1994). An accelerated learning algorithm for multilayer perceptron networks. *IEEE Transactions on Neural Networks*, 5(4), 563-578. doi: 10.1109/72.286921**

- 4. Bello, M. G. (1992). Enhanced training algorithms, and integrated training/architecture selection for multilayer perceptron networks. IEEE Transactions on Neural Networks, 3(5), 784-793. doi: 10.1109/72.165589**
- 5. Mat Isa, N. A., & Mamat, W. M. F. W. (2011). Clustered-hybrid multilayer perceptron network for pattern recognition application. Applied Soft Computing, 11(8), 5634-5643. doi: 10.1016/j.asoc.2010.04.017**
- 6. Yun, J., Park, J. E., Lee, H., Ham, S., Kim, N., & Kim, H. S. (2019). Radiomic features and multilayer perceptron network classifier: A robust MRI classification strategy for distinguishing glioblastoma from primary central nervous system lymphoma. Scientific Reports, 9(1), 4826. doi: 10.1038/s41598-019-42276-w**
- 7. Fukumizu, K. (1996). A regularity condition of the information matrix of a multilayer perceptron network. Neural Networks, 9(1), 19-26. doi: 10.1016/0893-6080(95)00119-0**
- 8. Su, H.-T., & McAvoy, T. J. (1993). Integration of multilayer perceptron networks and linear dynamic models: A Hammerstein modeling approach. Industrial & Engineering Chemistry Research, 32(12), 3154-3162. doi: 10.1021/ie00021a017**
- 9. Ahmad, F., Mat Isa, N. A., Hussain, Z., & Osman, M. K. (2013). Intelligent medical disease diagnosis using improved hybrid genetic algorithm - Multilayer perceptron network. Journal of Medical Systems, 37(3), 9920. doi: 10.1007/s10916-013-9934-7**
- 10. Xu, Y.G., Liu, G.R., Wu, Z.P., & Huang, X.M. (2001). Adaptive multilayer perceptron networks for detection of cracks in anisotropic laminated plates. Composite Structures, 54(3), 345-353. doi: 10.1016/S0020-7683(00)03681-1**

11. Soares, F.A.A.M.N., Flôres, E.L., Cabacinha, C.D., Carrijo, G.A., & Veiga, A.C.P. (2011). Recursive diameter prediction and volume calculation of eucalyptus trees using multilayer perceptron networks. Computers and Electronics in Agriculture, 75(2), 287-294. doi: 10.1016/j.compag.2011.05.008

12. García Nieto, P.J., Martínez Torres, J., De Cos Juez, F.J., & Sánchez Lasheras, F. (2012). Using multivariate adaptive regression splines and multilayer perceptron networks to evaluate paper manufactured using Eucalyptus globulus. Applied Mathematical Modelling, 36(11), 5220-5228. doi: 10.1016/j.amc.2012.07.001

13. Abderrahim, H., Chellali, M.R., & Hamou, A. (2016). Forecasting PM10 in Algiers: Efficacy of multilayer perceptron networks. Journal of Environmental Engineering and Landscape Management, 24(4), 265-276. doi: 10.3846/16486897.2016.1175049

14. Vilán Vilán, J.A., Alonso Fernández, J.R., García Nieto, P.J., Sánchez Lasheras, F., de Cos Juez, F.J., & Díaz Muñiz, C. (2013). Support Vector Machines and Multilayer Perceptron Networks Used to Evaluate the Cyanotoxins Presence from Experimental Cyanobacteria Concentrations in the Trasona Reservoir (Northern Spain). Water, Air, & Soil Pollution, 224(5), 1545. doi: 10.1007/s11270-013-1545-4

15. Gallagher, M., & Downs, T. (2003). Visualization of learning in multilayer perceptron networks using principal component analysis. IEEE Transactions on Neural Networks, 14(3), 669-674. doi: 10.1109/TSMCB.2003.808183

16. Polikar, R., Udpa, L., Udpa, S.S., & Honavar, V. (2000). LEARN++: An incremental learning algorithm for multilayer perceptron networks. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 30(4), 497-508. doi: 10.1109/5326.877215

17. Hametner, C., Stadlbauer, M., Deregnaucourt, M., Jakubek, S., & Winsel, T. (2013). Optimal experiment design based on local model networks and multilayer perceptron networks. *Industrial & Engineering Chemistry Research*, 52(47), 16735-16742. doi: 10.1021/ie101513z

18. Raji, C.G., & Vinod Chandra, S.S. (2017). Long-term forecasting the survival in liver transplantation using multilayer perceptron networks. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 47(6), 1577-1586. doi: 10.1109/TSMC.2017.2661996

19. Neto, N.F.S., Stefenon, S.F., Meyer, L.H., Bruns, R., Nied, A., Seman, L.O., Gonzalez, G.V., & Leithardt, V.R.Q. (2021). A study of multilayer perceptron networks applied to classification of ceramic insulators using ultrasound. *Applied Acoustics*, 179, 107997. doi: 10.1016/j.apacoust.2021.107997

20. Eslamloueyan, R., Khademi, M.H., & Mazinani, S. (2011). Using a multilayer perceptron network for thermal conductivity prediction of aqueous electrolyte solutions. *Engineering Applications of Artificial Intelligence*, 24(8), 1407-1414. doi: 10.1016/j.engappai.2012.05.016

Proyecto Final

“Red neuronal para clasificación de literatura científica”.

Introducción:

La inteligencia artificial (IA) ha revolucionado la forma en que analizamos y procesamos grandes cantidades de datos en diversos campos. En este proyecto, se propone utilizar técnicas de IA, específicamente una red neuronal, para abordar el desafío de clasificar literatura científica. La clasificación de artículos científicos es un proceso crucial en el

ámbito académico y de investigación, ya que permite identificar y seleccionar los estudios relevantes para un tema específico.

Objetivo general:

El objetivo principal de este proyecto es desarrollar una red neuronal que pueda clasificar automáticamente la literatura científica dando como salida si el artículo analizado está ubicado en la raíz, el tallo o las hojas. La red neuronal utilizará técnicas de aprendizaje automático supervisado para entrenarse con un conjunto de datos etiquetados, donde cada artículo científico está asociado con una categoría o tema específico. Una vez entrenada, la red neuronal será capaz de clasificar nuevos artículos científicos en las categorías adecuadas.

Marco teórico:

Para lograr este objetivo, se utilizará una red neuronal multicapa (MLP, por sus siglas en inglés). La MLP es un tipo de red neuronal artificial compuesta por múltiples capas de nodos interconectados. En este caso, la estructura de la red neuronal será de "n:n:3", lo que significa que tendrá una capa de entrada con "n" nodos (correspondientes a las características de los artículos científicos), una capa oculta con "n" nodos y finalmente una capa de salida con 3 nodos (representando si es raíz, tallo, o hojas). dando paso a la clusterización, Se utilizará la biblioteca de aprendizaje automático en Python para implementar la red neuronal y entrenarla con los datos disponibles.

CLARIDAD EN SITUACIÓN A RESOLVER

La aplicación de técnicas de IA en analítica de datos, genera un impacto disruptivo en orden de establecer nuevas metodologías para el desarrollo de múltiples actividades.

En este caso se pretende desarrollar un modelo que permita la selección de artículos científicos y de literatura por medio de una red neuronal

CLARO USO DE LA TÉCNICA DE IA

- + Neural network for Clustering in Python
- + MLP

Estructura de red: → n:n:3

Por lo general, un perceptrón multicapa (MLP) no se usa para agrupar porque MLP es un algoritmo de aprendizaje supervisado, lo que significa que está diseñado para aprender de un conjunto de datos etiquetados.

Por el contrario, la agrupación en clústeres es un problema de aprendizaje no supervisado, en el que se intenta agrupar datos sin etiquetar.

Sin embargo, un enfoque posible para agrupar usando MLP es usar un codificador
Aporte:

Una vez que se ha entrenado un codificador automático, la idea es usar la parte del codificador para mapear sus datos en un espacio dimensional más bajo donde puede usar un algoritmo de agrupamiento estándar (como K-means) o usar alguna regla para definir los clústeres.

Aquí hay un ejemplo simple de cómo puede hacer esto con Keras (deberá tener instalado Tensor Flow y Keras) automático. Los codificadores automáticos son un tipo de red neuronal que están capacitados para copiar su entrada a su salida. Tienen una capa interna (oculta) que describe un código utilizado para representar la entrada y constan de dos partes principales: un codificador, que asigna la entrada al código, y un decodificador, que asigna el código a una reconstrucción del original.

DESARROLLO

<https://matthew-parker.rbind.io/post/2021-01-16-pytorch-keras-clustering/>

<https://matthew-parker.rbind.io/post/2021-01-16-pytorch-keras-clustering/#>

<https://www.kaggle.com/datasets/grikomsn/amazon-cell-phones-reviews>

https://github.com/jmlopezzu/clust_neural_networks_SIC.git

You must install pydot (`pip install pydot`) and install graphviz (see instructions at <https://graphviz.gitlab.io/download/>) for plot_model to work.

<https://www.graphviz.org>

Optimal number of clusters with k means

RESULTADOS (Herramienta)

Dataset

Estandarización de indicadores. 0,1,2,3 → [0,3] escala de importancia

→ 0 = MIN, 3 = MAX

→ EJ Cesarea

Artículo	“edad”		“tiempo”	“presion”	“cardiaco”	ID “cesaria” Clasificador SAlida Raíz ¿?, Tronco ¿?, Hoja ¿?
→ Entrada						
Nuevo						
Viejo						
Medio						

CAMPOS

- + Artículo: X entrada
- + Año: → 3 [Min, 2000], → 2 [2000, 2013], → 1 [2013, 2018], → 0 [2018,2023]
- + Impacto: → 3 [Max,]

→ EJ: Lirios

Artículo	Sepal Length	Sepal Width	Petal length	Petal width	Cluster
Antiguo					
Medio					
Nuevo					

→ EJ: Literatura

Artículo	PY	CR	TC	SO	DT	Clase
Antiguo						RAIZ
Medio						TRONCO
Nuevo						HOJAS

- normalizar datos!!!

PY (año) → { 3 [Min, 2000], → 2 [2000, 2013], → 1 [2013, 2018], → 0 [2018,2023] }

Sum → **CR** (citaciones salida): → CITACIONES SALIDA {}

TC → (Número de citaciones): ← CITACIONES ENTRADA {}

Journal SO (Q) → Impacto: {3-[Q1],2-[Q2],1-[Q3],1-[Q4], 0-[NA]} →
“scimago2020.csv”--> Ranking SJR

DT → : { 3[Article], 2[review], 1[otros] }

Abstract	An overview or summary of research submitted to or presented at a meeting.
Art and Literature	An original work of fiction, music, poetry, or a script.
Article	A published research paper.
Bibliography	A listing of publications typically devoted to a particular topic, author, or time period
Biography	An article or publication, usually containing a bibliography, on the life and works of an individual.
Book	A monograph or publication written on a specific topic.
Case Report	A description of the presentation, diagnosis and treatment of a patient.
Clinical Trial	A study performed on patients to test the safety and efficacy of medical procedures.
Correction	A second citation of an article or publication for the purpose of correcting an error.

Data Paper	A scholarly publication describing a particular dataset or collection of datasets and usually published in the form of a peer-reviewed article in a scholarly journal. The main purpose of a data paper is to provide facts about the data (metadata, such as data collection, access, features etc) rather than analysis and research in support of the data, as found in a conventional research article.
Data Set	A single or coherent set of data (or a data file) provided by the repository as part of a collection, data study, or experiment. These can be of multiple file formats such as spreadsheet, video, and audio.
Data Study	A description of studies or experiments held in repositories with the associated data used in the data study. These are linked to a repository and may optionally link to Data Set records relating to the more granular data files.
Editorial	An authored viewpoint on a topic. It includes discussions, commentary and opinion pieces.
Government Publication	A publication produced by national, state, or local government.
Legislation	A proposed or enacted law.
Letter	Usually an informal, unsolicited, brief notation submitted to a journal concerning a specific finding or commenting on a previously published item in that journal.
Meeting	A collection of papers or abstracts from a meeting or symposium.
News	An item that reports recent events or developments.
Patent	A uniquely numbered legal, technical description of an invention which secures exclusive rights to its assignees.

Reference Material	Source of collected information to aid in research.
Report	A publication describing the process, progress, or results of research.
Repository	A database or collection of records that stores and provides access to the data from Data Studies and Data Sets.
Retracted Publication	An article that has been withdrawn by an author, institution, editor, or a publisher because of errors or unsubstantiated data.
Retraction	A public notice that an article should be withdrawn because of errors or unsubstantiated data.
Review	A renewed study or survey of previously published literature providing new analysis or summarization of the research topic.
Standard	A publication of established guidelines for a given biomedical subject area created by an accepted authority such as a committee, society, or government agency.
Thesis/Dissertation	Formal research document written to qualify for an academic degree.
Other	Used when none of the above document types apply.

PRESENTACIÓN

INFORME ENTREGADO

OBSERVACIONES

CÓDIGO MLP

```

# Importando las librerías necesarias
from sklearn.datasets import load_iris
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.neural_network import MLPClassifier
from sklearn.metrics import classification_report, confusion_matrix

# Cargando el dataset Iris
iris = load_iris()

# Dividiendo los datos en conjuntos de entrenamiento y prueba
X_train, X_test, y_train, y_test = train_test_split(iris.data, iris.target, random_state=1)

# Estandarizando los datos (es importante para la red neuronal)
scaler = StandardScaler()
scaler.fit(X_train)

# Aplicando la transformación a los datos
X_train = scaler.transform(X_train)
X_test = scaler.transform(X_test)

# Creando la red neuronal MLP
mlp = MLPClassifier(hidden_layer_sizes=(10, 10, 10), max_iter=1000)

# Entrenando la red neuronal
mlp.fit(X_train, y_train)

# Haciendo las predicciones
predictions = mlp.predict(X_test)

# Evaluando el modelo
print(confusion_matrix(y_test, predictions))
print(classification_report(y_test, predictions))

```

✓ 0.1s

```

[[13  0  0]
 [ 0 15  1]
 [ 0  0  9]]

```

	precision	recall	f1-score	support
0	1.00	1.00	1.00	13
1	1.00	0.94	0.97	16
2	0.90	1.00	0.95	9
accuracy			0.97	38
macro avg	0.97	0.98	0.97	38
weighted avg	0.98	0.97	0.97	38

3D PLOT



```
from sklearn.datasets import load_iris

iris = load_iris()
df = pd.DataFrame(data=iris.data, columns=iris.feature_names)
df['class'] = iris.target_names[iris.target]

df
```

[26] ✓ 0.7s

...	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)	class
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa
...
145	6.7	3.0	5.2	2.3	virginica
146	6.3	2.5	5.0	1.9	virginica
147	6.5	3.0	5.2	2.0	virginica
148	6.2	3.4	5.4	2.3	virginica
149	5.9	3.0	5.1	1.8	virginica

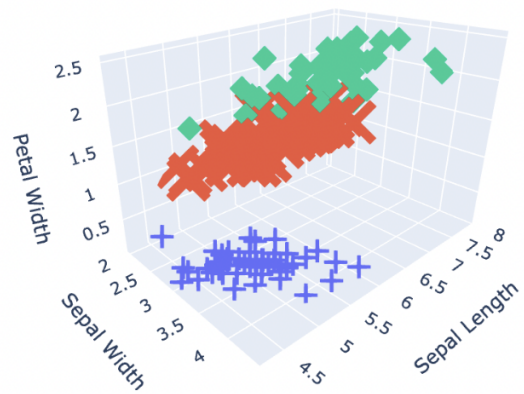
150 rows × 5 columns

este art habla de (Bayat et al., 2018)

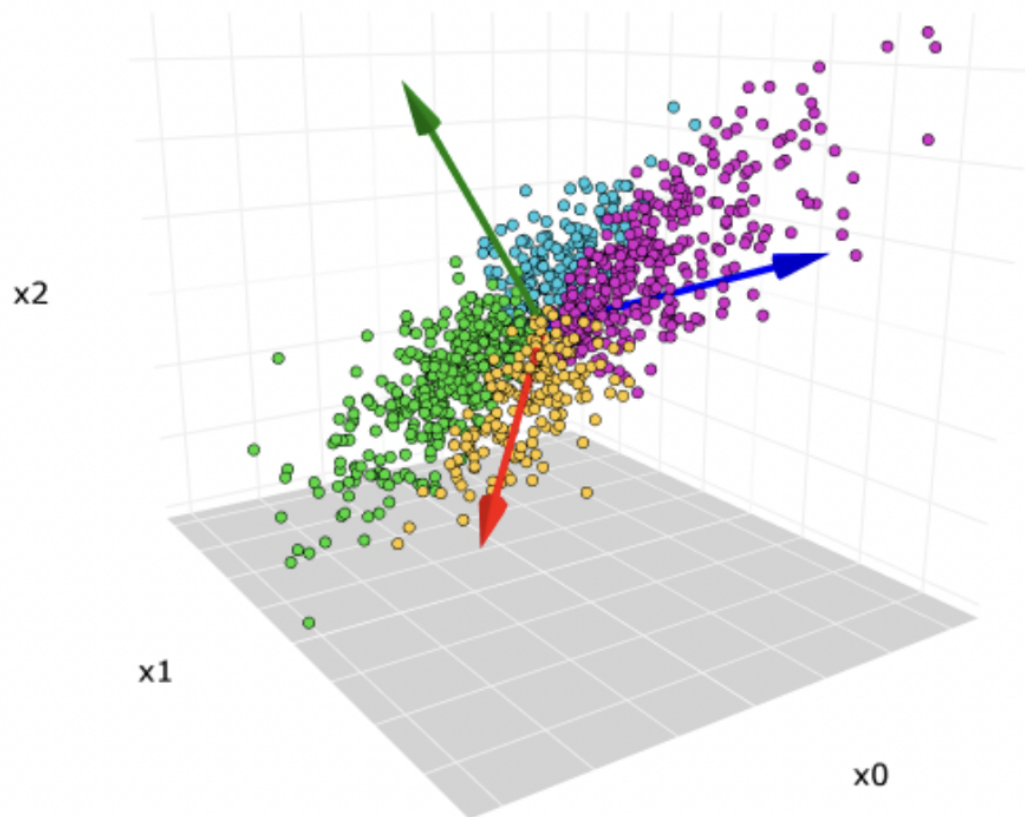
```
fig = px.scatter_3d(df, x='sepal width (cm)', y='sepal length (cm)', z='petal width (cm)',
                    color='class', symbol='class', size_max=10,
                    symbol_map={'setosa': 'cross', 'versicolor': 'x', 'virginica': 'diamond'})
fig.update_layout(title='Iris Dataset 3D Scatter Plot', scene=dict(xaxis_title='Sepal Width',
                                                                    yaxis_title='Sepal Length',
                                                                    zaxis_title='Petal Width'))
fig.show()
```

✓ 0.7s

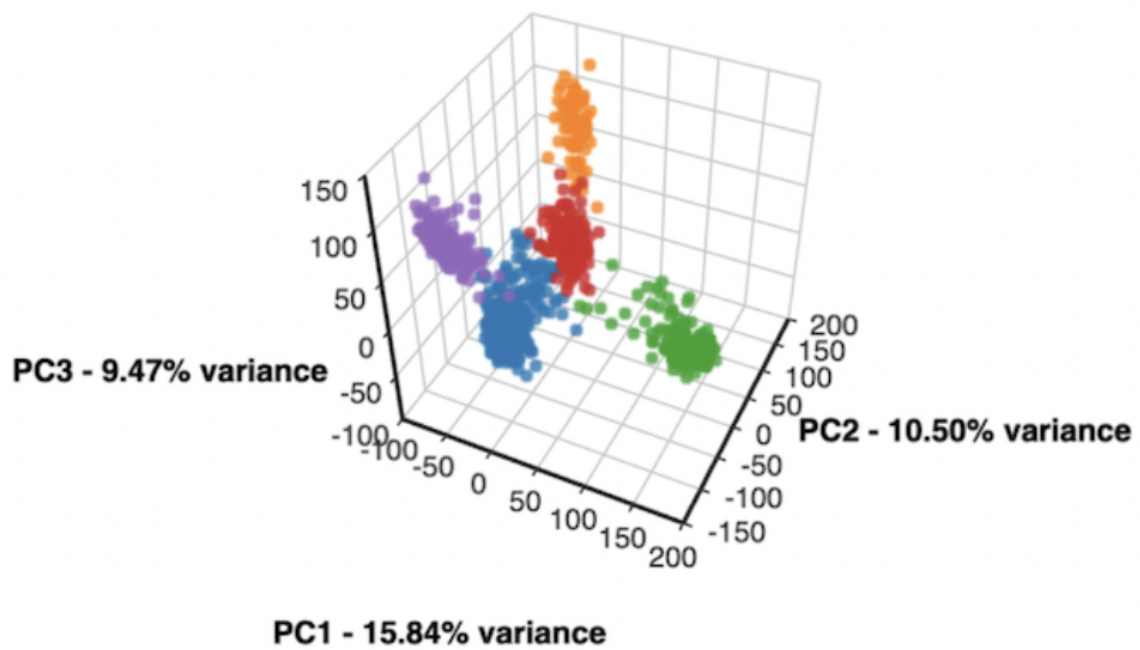
Iris Dataset 3D Scatter Plot



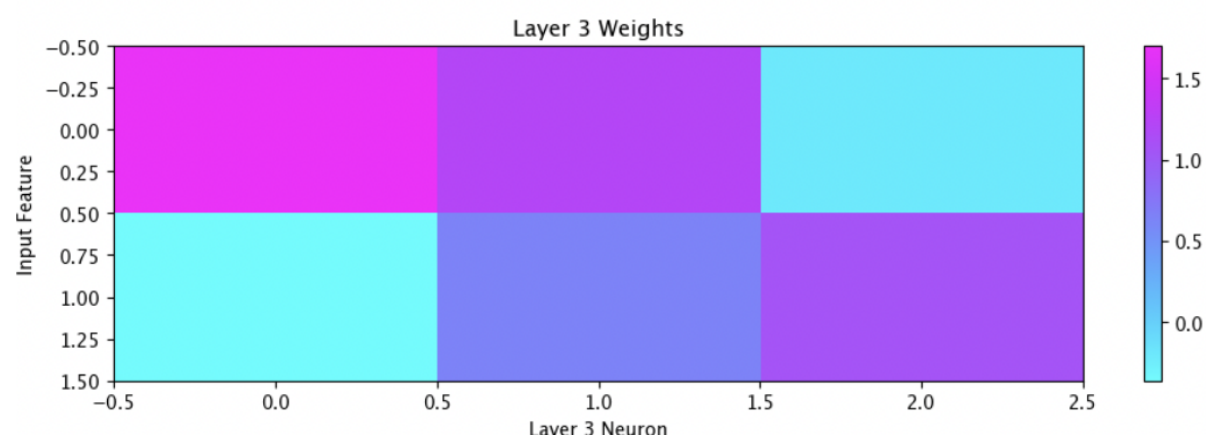
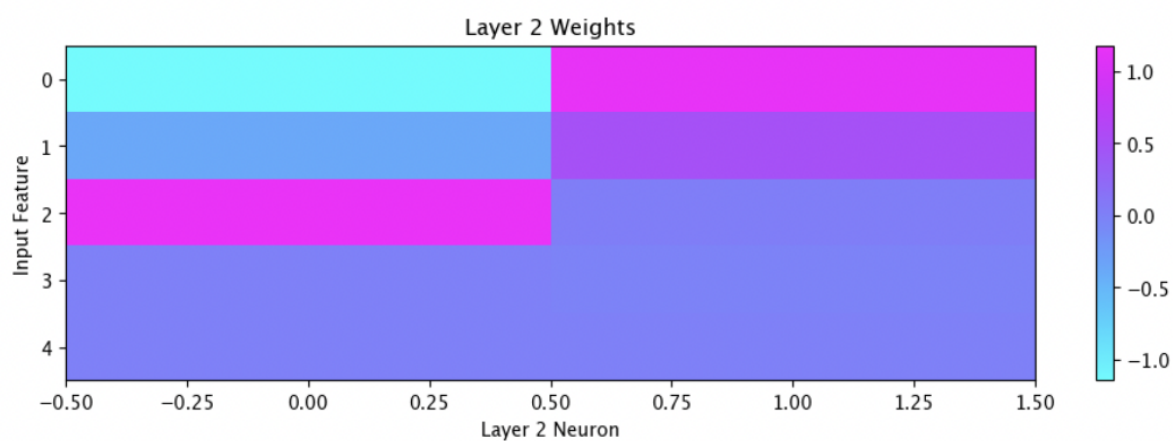
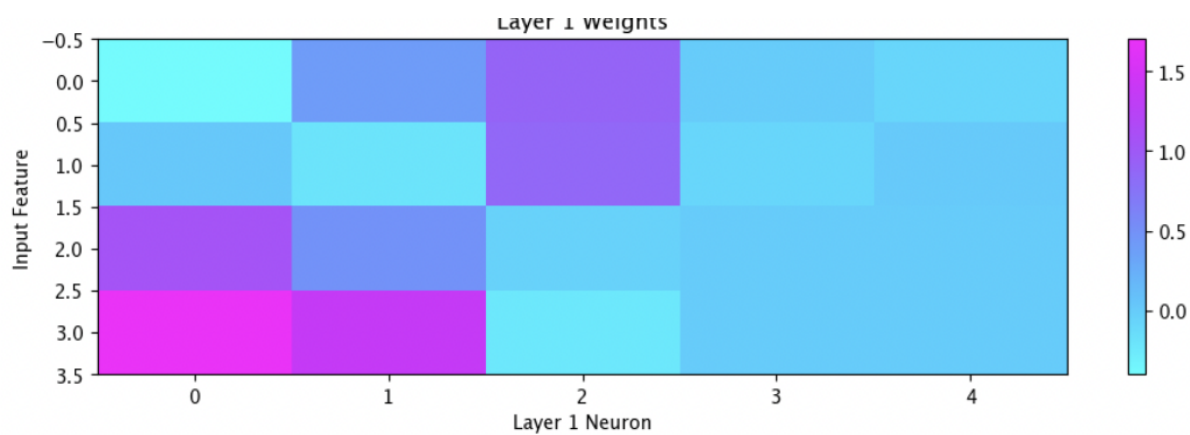
PCA

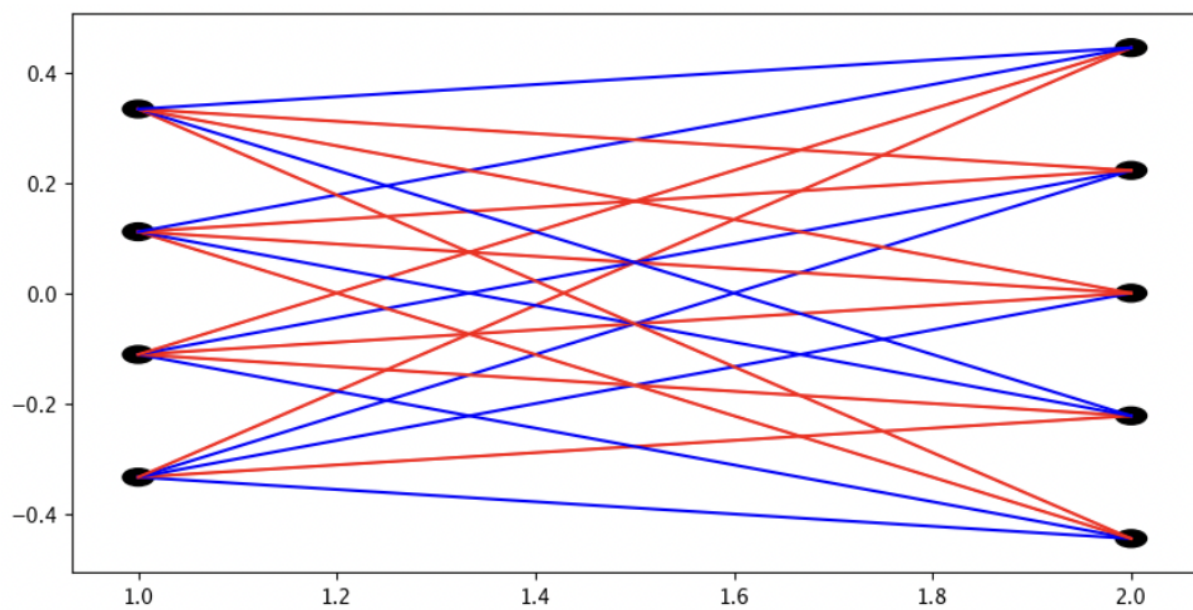


● BRCA ● COAD ● KIRC ● LUAD ● PRAD



NEURAL NETWORK PLOT





```
import numpy as np
import matplotlib.pyplot as plt

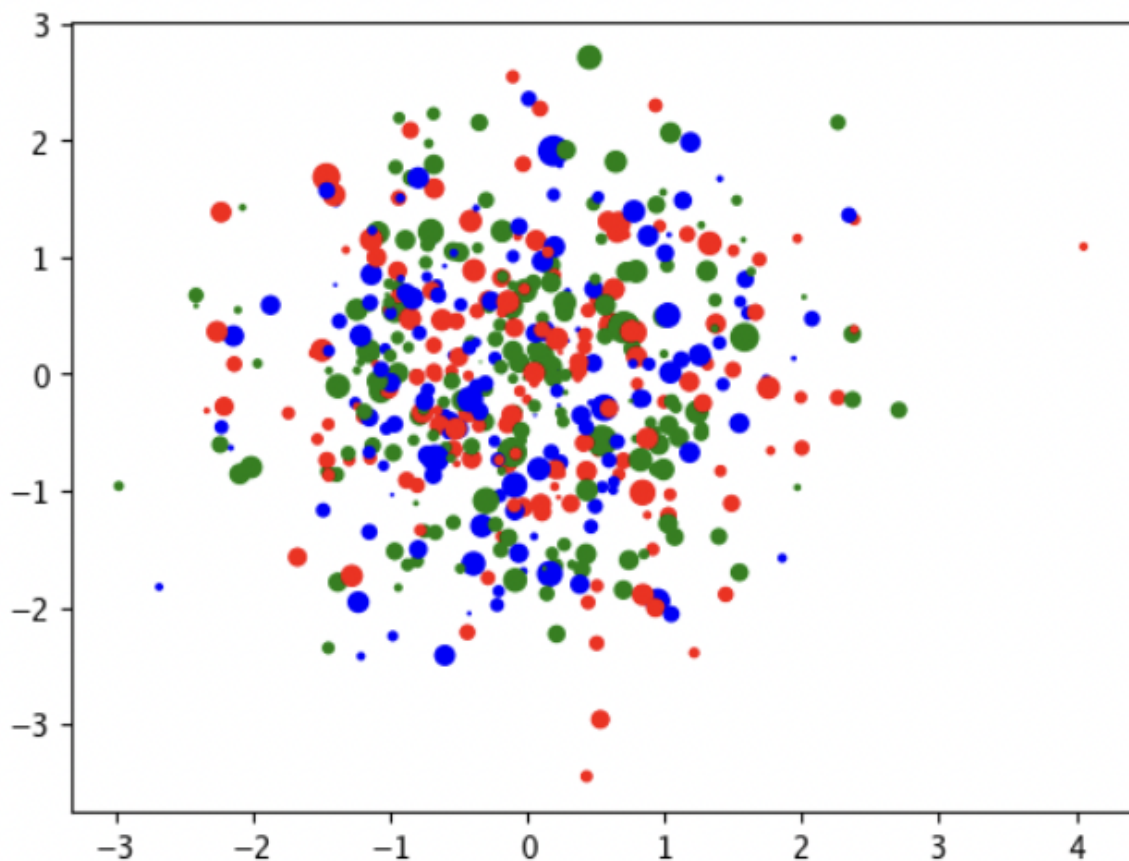
# Generate some sample data
x = np.random.normal(size=500)
y = np.random.normal(size=500)
z = np.abs(np.random.normal(size=500))
colors = np.random.choice(['r', 'g', 'b'], size=500)

# Define the size of each point based on z
sizes = 100 * z / np.max(z) # scale sizes to have max size of 100

# Create a scatter plot with colors and sizes
plt.scatter(x, y, c=colors, s=sizes)

# Show the plot
plt.show()
```

✓ 0.1s



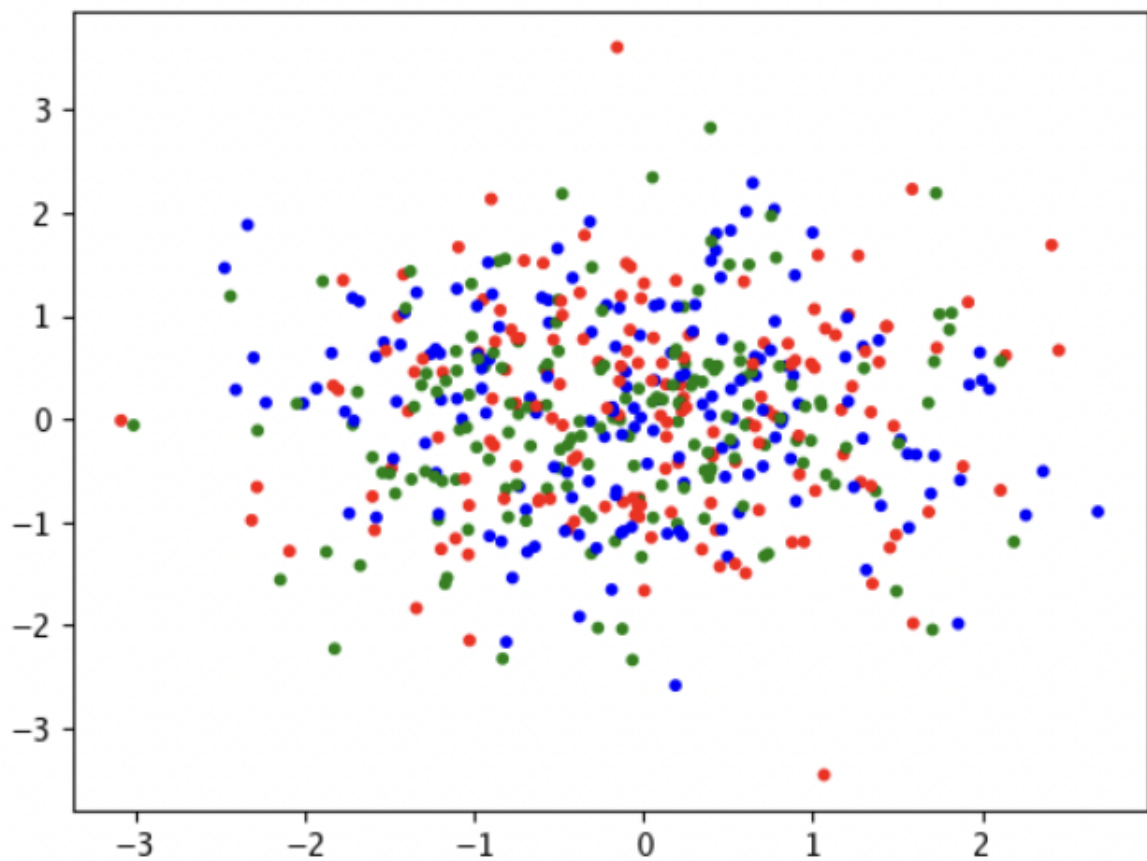

```
import numpy as np
import matplotlib.pyplot as plt

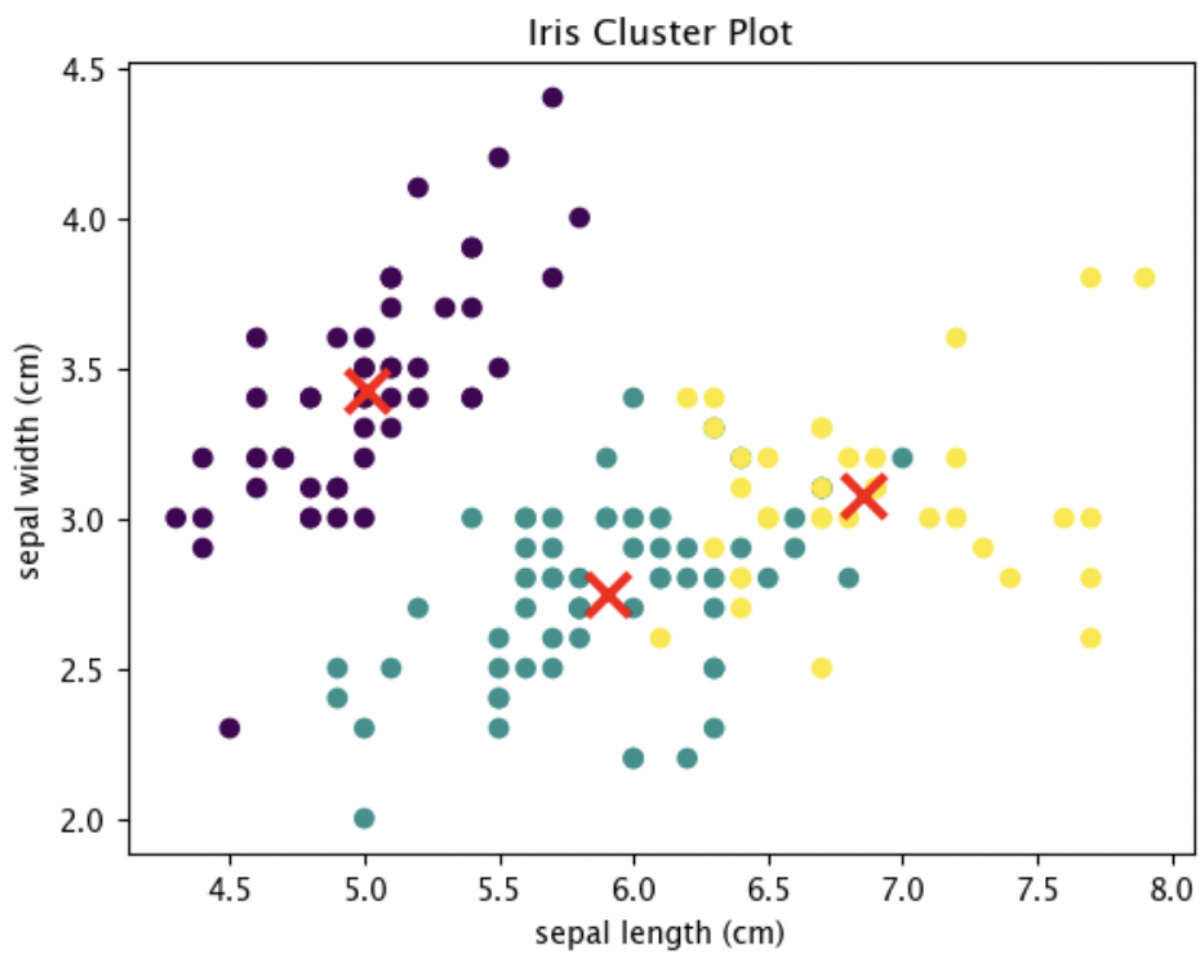
# Generate some sample data
x = np.random.normal(size=500)
y = np.random.normal(size=500)
colors = np.random.choice(['r', 'g', 'b'], size=500)

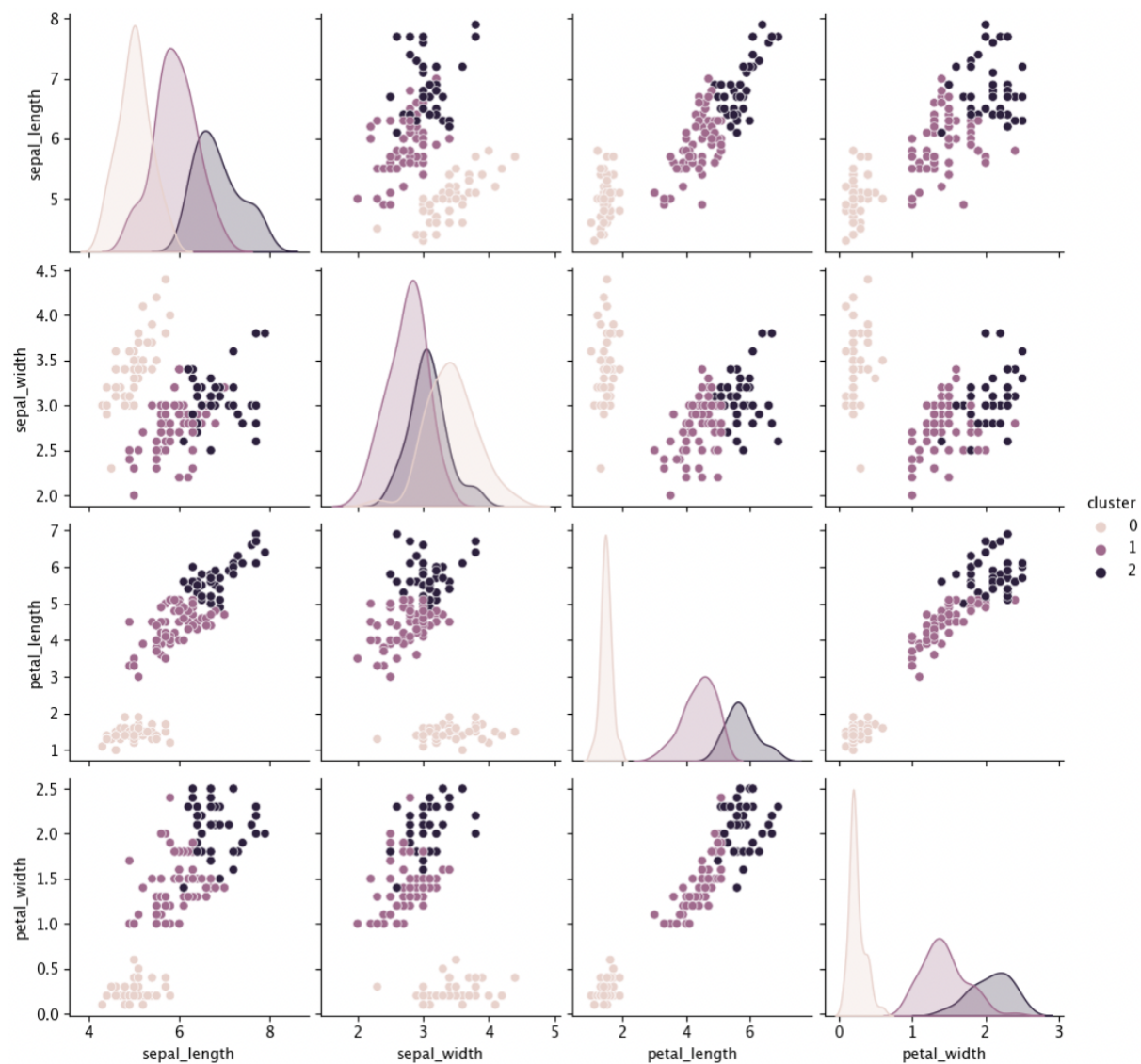
# Create a scatter plot with colors and same size for all points
plt.scatter(x, y, c=colors, s=10)

# Show the plot
plt.show()
```

✓ 0.9s







```

import pandas as pd
from sklearn.cluster import KMeans
from pandas.plotting import parallel_coordinates

# Load the iris dataset
iris = pd.read_csv('https://raw.githubusercontent.com/mwaskom/seaborn-data/master/iris.csv')

# Drop the "species" column
iris_numeric = iris.drop('species', axis=1)

# Cluster the data using k-means algorithm
kmeans = KMeans(n_clusters=3, random_state=42).fit(iris_numeric)

# Add the cluster labels to the iris dataframe
iris['cluster'] = kmeans.labels_

# Create a new dataframe with only the numeric features and cluster labels
numeric_cluster_df = pd.concat([iris_numeric, iris['cluster']], axis=1)

# Create dictionary to map colors to each cluster label
colors = {0: 'r', 1: 'g', 2: 'b'}

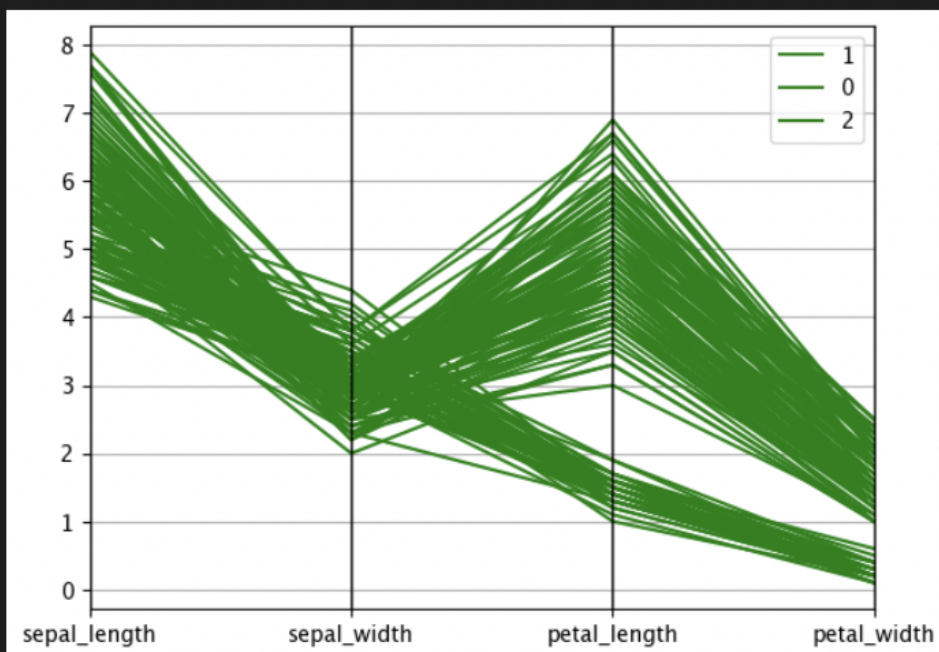
# Create parallel coordinates plot with color-coded lines based on cluster label
parallel_coordinates(numeric_cluster_df, 'cluster', color=[colors[label] for label in numeric_cluster_df['cluster'].unique()])

```

✓ 0.2s

/opt/homebrew/lib/python3.10/site-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The default v
warnings.warn(

<AxesSubplot: >



Bayat, F. M., Prezioso, M., Chakrabarti, B., Nili, H., Kataeva, I., & Strukov, D. (2018).

Implementation of multilayer perceptron network with highly uniform passive memristive crossbar circuits. *Nature Communications*, 9(1), 2331.