Universidad del Desarrollo
Facultad de Ingeniería

DETECCIÓN DE SECUENCIA DE CLÚSTERES GLOBULARES EN VIRGO

POR: JORGE ADRIÁN FERNÁNDEZ ZENTENO

Capstone project presentado a la Facultad de Ingeniería de la Universidad del Desarrollo para optar al grado académico de Magíster en Data Science

Profesor: Dra. Daniela Opitz
Profesor: Dr. Takeshi Asahi

Enero 2023
SANTIAGO DE CHILE

Dedicado a mi familia que son mi inspiración.

Agradecimiento:
A mi familia, quienes con amor incondicional, paciencia y apoyo constante, me han acompañado en cada paso de este viaje, siendo fuente de inspiración y aliento incluso en los momentos más desafiantes.

A mis profesores, que con su sabiduría, orientación y pasión por la ciencia, no solo han enriquecido este trabajo, sino que han moldeado mi formación y perspectiva.

Y dedicar un agradecimiento especial al Dr. Roberto Muñoz. Su pasión por la astronomía, y constante búsqueda de conocimiento no solo me inspiraron, sino que me impulsaron a seguir sus pasos y continuar con su labor.

1. Tabla de contenido

Resumen

En este trabajo estudiamos métodos de aprendizaje de máquina para identificar clústeres globulares del cúmulo de galaxias de Virgo, centrado en la Galaxia Messier 87 (también conocida como M87, Virgo A o NGC 4486), en el corazón de la región de Virgo, utilizando información magnitud en las bandas visible e infrarroja y la posición obtenida de los catálogos fotométricos denominado Next Generation Virgo Survey (NGVS) y Next Generation Virgo Survey- Infrarrojo (NGVS-IR). Estos catálogos contienen información de más de 300 mil objetos celestes entre estrellas, galaxias y cúmulos.

Los cúmulos globulares son grupos de estrellas muy densamente pobladas y están ubicados en el halo de las galaxias. Suelen tener colores y brillos muy similares entre sí, ya que están compuestos por estrellas de edad y tipo espectral similares. Los cúmulos globulares suelen representarse en “diagrama color-color gz-K”, los cuales permiten comparar el brillo de los objetos estelar en distintas longitudes de onda y permiten entender sus características y clasificarlos según su estructura y composición estelar.

En este trabajo proponemos combinar datos de fotometría en el espectro óptico e infrarrojo con algoritmos de machine learning no supervisados para distinguir cúmulos globulares.

Key words: galaxias: clúster de estrellas - galaxias: clústeres globulares - galaxias: gz-K - galaxias: cúmulos de estrellas – general: gaussian mixture model

1. Introducción

Los cúmulos globulares de galaxias son grupos de galaxias que se mantienen unidas por la fuerza de gravedad. Son consideradas estructuras poderosas en la escala del Universo (Lindholm et al. 2021) y algunas de sus propiedades tales como la densidad de objetos en función de la masa total y el agrupamiento espacial alimentan modelos cosmológicos (Durrell et al. 2014).

En este proyecto nos enfocaremos principalmente en el estudio del Cúmulo de Virgo, el cual constituye un cumulo de galaxias ubicado a una distancia media de 16,5 Mpc (Mei et al. 2007). Este cúmulo de Virgo es un importante objeto de estudio en astronomía ya que se caracteriza por una gran densidad de galaxias (Muñoz et al. 2013) y constituye un laboratorio natural para estudiar la formación y evolución de un cúmulo de galaxias típico (Durrell et al. 2014).

El cúmulo de Virgo contiene más de 2 mil galaxias identificadas a la fecha y cerca de 10 mil cúmulos globulares en su parte central y es 3 veces más grande que nuestra vía láctea. La distancia medida de la cD galaxy "central Dominant galaxy" M87 es consistente con la media del cúmulo y su densidad permite realizar múltiples estudios (D'Abrusco et al. 2016). *En* las últimas décadas, se iniciaron diversos estudios de cúmulos globulares extra galácticos (CGEG) en la zona de Virgo, por ello investigadores de todo el mundo analizan toda la información recolectada por los modernos telescopios, satélites de investigación con instrumentación especializada (Ferrarese et al. 2012), mejorando el proceso de clasificación e identificación.

Por ello la motivación científica de este documento es lograr encontrar otros métodos, distintos a los tradicionales en el dominio de la astro-informática, utilizando técnicas modernas del área de las ciencias de datos para lograr separar las estrellas de primer plano de las galaxias de fondo e identificar los cúmulos o clústeres globulares extra galácticos, lo cual es importante para muchas áreas de investigación astronómica, desde la ciencia galáctica hasta la cosmología (Ko, Youkyung et al. 2022a).

Actualmente algunos grupos han introducido nuevas técnicas de estudio debido a que las técnicas de clasificación morfológica convencionales separan las fuentes puntuales (principalmente estrellas) de las fuentes resueltas (galaxias) mediante selecciones en el espacio de magnitud-radio o variables similares (Ferrarese et al. 2012).

En este trabajo proponemos utilizar métodos distintos a los tradicionales para separar las estrellas de primer plano de las galaxias de fondo e identificar los cúmulos o clústeres globulares extra galácticos.

Por ello se propone el uso de técnicas de machine learning, donde se desarrollarán algoritmos no supervisados tales como Kmeans o Gaussian Mixture Modeling (GMM) y otras técnicas de ciencias de datos como la ingeniería de características entre otras para realizar este análisis.

1.1. Bandas electromagnéticas y diagrama color-color gz-K

1.1.1. Bandas Electromagnéticas

Una banda electromagnética es una sección del espectro de frecuencias o longitudes de onda del espectro electromagnético. Las bandas pueden ser clasificadas en categorías como: ultravioleta, visible e infrarrojo, según la longitud de onda o frecuencia de la radiación. La escala una banda electromagnética comúnmente se mide en unidades de longitud de onda (γ), como nanómetros (nm) o Ångstroms (Å). Un Ångstrom es una unidad de medida de longitud que equivale a 10^{-9} metros.

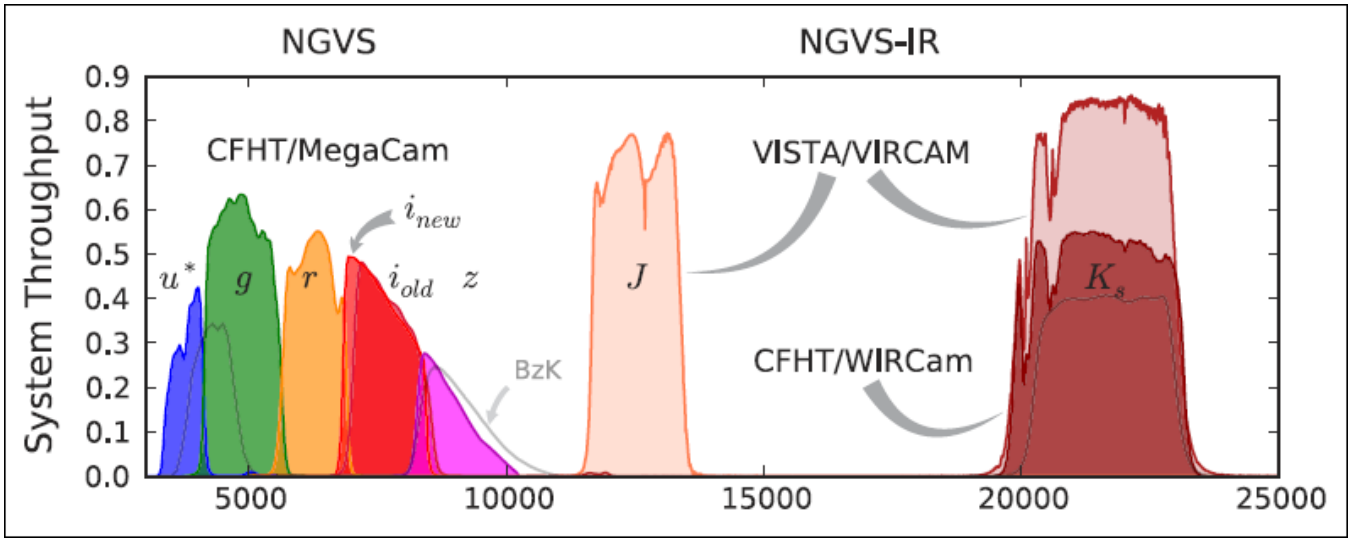


Figura 1. El eje **X** corresponde a longitudes de onda (γ) en Angstroms (Å) (5000-25000) y se observan las bandas [u,i,K] [g,z,K] entre otras y en el eje **Y** corresponde al parámetro *System Throughput* que es una medida de eficiencia de un telescopio o instrumento científico en la recolección de luz. Elaborado por *(Muñoz et al. 2013).*

La banda de ultravioleta (UV) tiene longitudes de onda de aproximadamente 10 a 400 nanómetros (nm), o 1000 a 4000 Å (Angstroms). La luz ultravioleta es emitida por estrellas calientes y nebulosas, tiene una mayor energía que la luz visible.

La banda visible tiene longitudes de onda de aproximadamente 400 a 750 nm, o 4000 a 7500 Å (Angstroms). Es la banda que nuestros ojos son capaces de detectar, y es la banda responsable de los colores que vemos en los objetos.

La banda de infrarroja (IR) tiene longitudes de onda de aproximadamente 750 nm a 1 millón de nm, o 7500 a 1000000 Å. La luz infrarroja es emitida por objetos calientes y tiene una menor energía que la luz visible; Respecto a la luz infrarroja cercana (NIR) sus longitudes de onda de radiación electromagnética se encuentran entre los 750 y 2500 nanómetros (nm) o 7500 a 25000 Å (Angstroms).

La figura 1 representa las longitudes de bandas incluidas en el catálogo fotométrico y su representación con las letras [u,i,K] o [g,z,K] *(Muñoz et al. 2013)* que se utilizarán en el análisis del presente documento.

El System Throughput es una medida de la eficiencia de un telescopio o instrumento científico en la recolección de luz. Es la relación entre la cantidad de luz que entra en el instrumento y la cantidad de luz que se detecta. Un alto System Throughput significa que el instrumento es capaz de recolectar y detectar una gran cantidad de luz, lo que permite estudiar objetos débiles.

1.1.2. Diagrama color-color gz-K

Los diagramas color-color corresponden a una herramienta tradicional en astronomía que permite comparar distintas bandas electromagnéticas para estudiar las propiedades de los objetos celestes. En este proyecto utilizamos diagramas color-color **gz-K** fotométricos, los cuales se basan en las bandas g, z, K,

estos diagramas proporcionan una herramienta valiosa para entender las características de las galaxias y los cúmulos globulares, e identificar características según su estructura y composición estelar.

La Figura 2, contiene un diagrama color-color **gz-K** fotométrico basado diferencias entre las bandas g, z, K elaborado por *Muñoz et al 2013.*

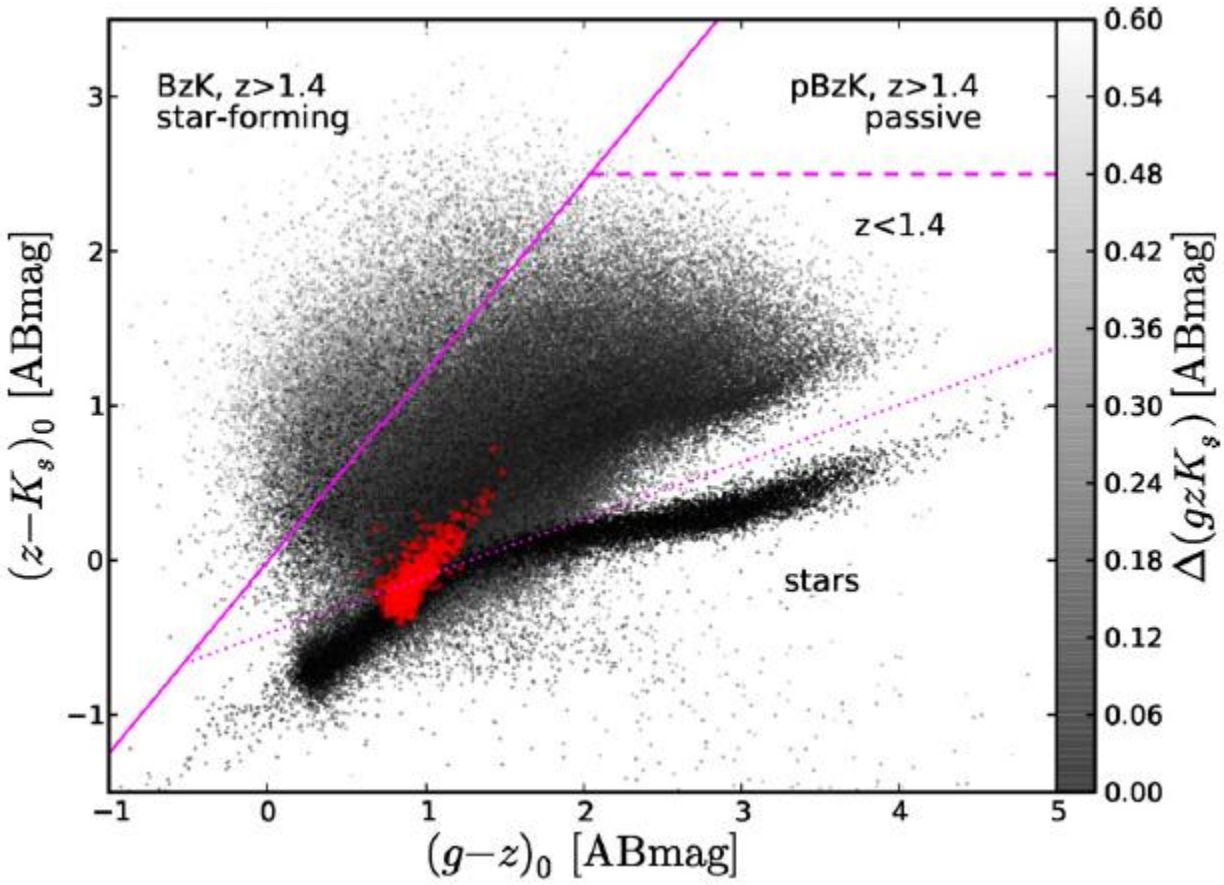


Figura 2. Diagrama color-color **gz-K**, el eje **X** corresponde a la diferencia entre bandas (g-z), mientras que el eje **Y** corresponde a la diferencia entre bandas (z-K), ambos medidos en aperturas circulares de diámetro 4″. Los puntos rojos muestran los cúmulos globulares de los cúmulos de Virgo confirmados mediante la medición de *velocidades radiales*.

2. Trabajo Relacionado

Los cúmulos globulares han sido tradicionalmente estudiados por técnicas astronómicas clásicas y más recientemente, con la ayuda de algoritmos de aprendizaje automático. A continuación, presentamos un resumen de los trabajos más relevantes distingüendo entre métodos tradicionales y no tradicionales.

2.1. Métodos tradicionales

Dentro de los estudios más importantes de cúmulos globulares destacan los trabajos de *(Ferrarese et al. 2012)* y otras investigaciones en el mismo campo *(Muñoz et al. 2013)*, quien desarrolló un catálogo de observaciones del cúmulo de galaxias de Virgo en el espectro ultravioleta, óptico e infrarrojo cercano y utilizando métodos tradiciones como los diagramas fotométricos gzK o gz-K ("g-z" vs "z-K"), también conocido como “diagrama color-color gz-K", como así otros análisis detallados en (Ko, Youkyung et al. 2022a) en combinación con algoritmos extreme deconvolution (xD) mencionado en el paper de referencia.

Trabajos similares de relevamiento astronómico del cúmulo de galaxias de Virgo en el espectro óptico e infrarrojo pueden encontrarse en (*Jin et al. 2019), (Durrell et al. 2014), (Dékány et al. 2013).*

2.2. Métodos no tradicionales

Trabajos más recientes comenzaron a utilizar métodos de agrupamientos (clustering), los cuales son una de las herramientas importantes de análisis de datos para identificar objetos no marcados (labels) y agruparlos por alto nivel de similitudes en los mismos grupos, los métodos tradicionales utilizan métodos K-Means *(Nie, Wang, and Li 2019),* desarrollados principalmente en lenguaje Fortran y otros más avanzados como K-Multiple-Means (KMM) quien calcula las estimaciones de máxima verosimilitud de sus medias y varianzas y evalúa la mejora de ese ajuste sobre una sola gaussiana para identificar Clúster globulares en la vía láctea *(Ashman, Bird, and Zepf 1994)* y otros métodos no supervisados más complejos que permiten mayor control de sus parámetros como Gaussian Mixture Models[4] *(Ko, Youkyung et al. 2022a.*), con el cual desarrollaremos en detalle este trabajo.

3. Objetivos

El objetivo de este trabajo es utilizar datos del catálogo fotométrico denominado Next Generation Virgo Survey (NGVS) y Next Generation Virgo Survey- Infrarrojo (NGVS-IR) que contiene información de más de 300 mil objetos celestes entre estrellas, galaxias y cúmulos, desarrollado por Muñoz *(Muñoz et al. 2013)* y aplicar algoritmos de aprendizaje automático no-supervisados para identificar clústeres globulares en la constelación de Virgo.

Como objetivo específico del trabajo es comparar dos métodos de aprendizaje automático no supervisado que puedan utilizarse en trabajos posteriores y puedan ser utilizando como referencia en la clasificación de los clústeres globulares.

4. Descripción de los Datos

4.1. Descripción General de los Datos

Los datos de este trabajo consisten en mediciones fotométricas (filtros g, z) y en el infrarrojo cercano NIR (filtro K) de objetos identificados en el cúmulo de galaxias de Virgo, el cual está formado por más de 2.000 galaxias; y son el resultado de 2 años de observación con el telescopio CFHT en Mauna Kea en Hawái[1].

4.1.1. Estructura del catálogo y formato

El catálogo se encuentra en formato FITS[3] (Flexible Image Transport System) es un formato de archivo utilizado en astronomía para almacenar, transportar y analizar datos científicos. Está diseñado principalmente para almacenar conjuntos de datos científicos que consisten en arreglos multidimensionales (imágenes y mediciones) en tablas bidimensionales organizadas en filas y columnas de información y ejes o slices. Los archivos FITS se componen de una serie de extensiones llamadas "headers" que contienen metadatos sobre el archivo, como la fecha de observación, la configuración del instrumento utilizado y una sección de datos que contiene los datos numéricos que representan la imagen o tabla.

Los datos en formato FITS se almacenan en formato binario, lo que permite una mayor precisión y eficiencia en comparación con los formatos de texto. FITS es ampliamente aceptado en la comunidad astronómica y es compatible con una amplia variedad de programas de análisis de datos.

4.1.2. Estructura del catálogo fotométrico

El catálogo fotométrico contiene 40 variables que incluyen información de fotométrica de los objetos y coordenadas celestes. A continuación, en la **Tabla 1**, presentamos una descripción de las variables del catálogo fotométrico. Los sufijos **1** y **2** permiten distinguir los datos fotométricos del espectro óptico e infrarrojo cercano respectivamente.

Nombre	Unidad	Banda	Descripcíon
ALPHA_J2000_1	degree	Optical	J2000 right ascension of the isophotal image centroid
DELTA_J2000_1	degree	Optical	J2000 declination of the isophotal image centroid
MAG_PSF_1		Optical	As an indicator of star–galaxy separation w e use the SPREAD_MODEL
MAGERR_PSF_1		Optical	Error estimate(s) as an indicator of star–galaxy separation w e use the SPREAD_MODEL
MAG_AP3_1	magnitude	Optical	Circular aperture magnitude(s)
MAG_AP4_1	magnitude	Optical	Circular aperture magnitude(s)
MAG_AP5_1	magnitude	Optical	Circular aperture magnitude(s)
MAG_AP6_1	magnitude	Optical	Circular aperture magnitude(s)
MAG_AP7_1	magnitude	Optical	Circular aperture magnitude(s)
MAG_AP8_1	magnitude	Optical	Circular aperture magnitude(s)
MAG_AP16_1	magnitude	Optical	Circular aperture magnitude(s)
MAG_AP32_1	magnitude	Optical	Circular aperture magnitude(s)
MAGERR_AP3_1	magnitude	Optical	RMS error estimate(s) for circular aperture magnitude(s)
MAGERR_AP4_1	magnitude	Optical	RMS error estimate(s) for circular aperture magnitude(s)
MAGERR_AP5_1	magnitude	Optical	RMS error estimate(s) for circular aperture magnitude(s)
MAGERR_AP6_1	magnitude	Optical	RMS error estimate(s) for circular aperture magnitude(s)
MAGERR_AP7_1	magnitude	Optical	RMS error estimate(s) for circular aperture magnitude(s)
MAGERR_AP8_1	magnitude	Optical	RMS error estimate(s) for circular aperture magnitude(s)
MAGERR_AP16_1	magnitude	Optical	RMS error estimate(s) for circular aperture magnitude(s)
MAGERR_AP32_1	magnitude	Optical	RMS error estimate(s) for circular aperture magnitude(s)
MAG_AUTO_1	magnitude	UltraViolet / Optical	Kron-like automated aperture magnitude (slice band u-g-z-i)
MAGERR_AUTO_1	magnitude	Optical	RMS error estimate for Kron-like automated aperture magnitude
FLAGS_1	…	Optical	Source extraction flags
CLASS_STAR_1	Binary	Binary	Star/galaxy classifier
FLUX_RADIUS_1		Optical	FLUX_RADIUS is the circular radius that encloses half of the light w ithin in the AUTO aperture.
ALPHA_J2000_2	degree	Near-Infrared	J2000 right ascension of the isophotal image centroid
DELTA_J2000_2	degree	Near-Infrared	J2000 declination of the isophotal image centroid
MAG_PSF_2		Near-Infrared	As an indicator of star–galaxy separation w e use the SPREAD_MODEL paramete provided by Sextractor
MAGERR_PSF_2		Near-Infrared	Error estimate(s) as an indicator of star–galaxy separation w e use the SPREAD_MODEL
MAG_APER_2	magnitude	Near-Infrared	Circular aperture magnitude(s)
MAGERR_APER_2	magnitude	Near-Infrared	RMS error estimate(s) for circular aperture magnitude(s)
MAG_AUTO_2	magnitude	Near-Infrared	Kron-like automated aperture magnitude
MAGERR_AUTO_2	magnitude	Near-Infrared	RMS error estimate for Kron-like automated aperture magnitude
FLAGS_2		Near-Infrared	Source extraction flags
IMAFLAGS_ISO_2	…	Near-Infrared	External flags combined w ithin the isophotal footprint
CLASS_STAR_2		Near-Infrared	Star/galaxy classifier
FLUX_RADIUS_2		Near-Infrared	FLUX_RADIUS is the circular radius that encloses half of the light w ithin in the AUTO aperture.
Separation	arcsec		Distance betw een matched objects along a great circle
stars_uiK	magnitude		
good_stars_uiK	magnitude		

Tabla 1 - Detalles de variables del Catalogo Next Generation Virgo Survey (NGVS, NGVS-IR)

Luego en la Tabla 2 se detallan todas las variables más significativas para el análisis del presente trabajo Adicionalmente el catálogo en formato FITS tiene cortes, slices o ejes (propio del formato FITS) en donde existe información relevante como por ejemplo dentro de la variable MAG_AUTO, slice **0** se encuentran la información de magnitud de la banda ultravioleta, identificada con la letra **u**, y así sucesivamente con las otras bandas visible (**g,i,z)** y la banda de infrarrojo cercano identificado con la letra **K**

Nombre	Unidad	Banda	Slice	Letra	Descripción
MAG_PSF_1	magnitud	Optical			Medición de la magnitud aparente de la fuente usando fotometría SPREAD_MODEL Function espectro óptico
MAG_APx_1	magnitud	Optical			Magnitud medida usando una apertura circular centrada en el objeto, el número que viene después de AP es el diámetro en pixels (3,4,5,6,7,8,16,32)
MAGERR_AP3_1	magnitud	Optical			Error de magnitud aparente RMS
MAG_AUTO_1	magnitud	**Ultraviolet**	**0**	**u**	Magnitud medida usando una apertura automática circular centrada en el objeto espectro ultravioleta
MAG_AUTO_1	magnitud	**Optical**	**1**	**g**	Magnitud medida usando una apertura automática circular centrada en el objeto espectro óptico
MAG_AUTO_1	magnitud	**Optical**	**3**	**i**	Magnitud medida usando una apertura automática circular centrada en el objeto espectro óptico
MAG_AUTO_1	magnitud	**Optical**	**4**	**z**	Magnitud medida usando una apertura automática circular centrada en el objeto espectro óptico / NIR
CLASS_STAR_1	Decimal	Optical			Star/galaxy Utilizada para distinguir si es estrella o galaxia, decimal que va entre 0 y 1
MAG_PSF_2	magnitud	Near-Infrared			Medición de la magnitud aparente de la fuente usando fotometría SPREAD_MODEL Function para el espectro infrarrojo.
MAG_AUTO_2	magnitud	**Near-Infrared**	0	**K**	Magnitud medida usando una apertura automática circular centrada en el objeto espectro infrarrojo
stars_uiK	Binario				
good_stars_uiK	Binario				True / False si es una estrella probable uiK gzK

Tabla 2: Variables significativas más relevantes del Catalogo NGVS NGVS-IR

4.2. Origen de la Fuente de Datos

El catálogo fotométrico es la unión de los catálogos Next Generation Virgo Survey (NGVS) y (Next Generation Virgo Survey- Infrarrojo (NGVS-IR) *(Muñoz et al. 2013).* NGVS contiene magnitudes de objetos medidos en los filtros U, G, I, Z mientras que el catálogo NGVS-IR contiene las magnitudes de objetos en el infrarrojo cercano (K).

Las medidas fotométricas de ambos catálogos son el resultado de procesar miles de imágenes provenientes del telescopio CFH de 3.6 metros, ubicado en Maunakea, Hawaii. Los datos crudos en el espectro óptico corresponden a aproximadamente a cientos de imágenes obtenidas con cerca de 100 noches de observaciones mientras que los datos crudos en para el caso del infrarrojo cercano corresponden a imágenes de cerca de 10 noches de observación. Los pipelines de las imágenes ópticas fueron diseñadas e implementadas por el equipo del Canadian Astronomy Data Centre (*Muñoz et al. 2013).*

5. Metodología

A continuación, describimos los pasos desarrollados para implementar los modelos de machine learning no supervisados:

- Selección e instalación de librerías y recursos

- Limpieza de datos.
- Selección y construcción de variables (Feature Engineering)
- Modelos de Machine Learning
- Selección de modelo y pruebas.
- Validación de resultados

5.1.Selección de Librerías y Recursos

Para el desarrollo de este proyecto utilizamos el lenguaje Python en su versión 3.8 y la herramienta Jupyter Notebooks que permite documentar en celdas y ejecutar código de manera ordenada y controlada. Las librerías más relevantes utilizadas en la ejecución y comúnmente utilizadas en proyectos de Data Science: Pandas, Sklearn, Numpy, Seaborn y Matplotlib.

Adicionalmente utilizamos la librería **AstroPy**[2] que es una biblioteca de software de código abierto para astronomía en Python. Proporciona una variedad de herramientas y funciones para la manipulación, análisis y visualización de datos científicos en astronomía, con un énfasis en el formato FITS. Algunas de las funciones clave de AstroPy incluyen:

- Lectura y escritura de archivos FITS: AstroPy proporciona una interfaz fácil de usar para leer y escribir archivos FITS.
- Manipulación de datos: AstroPy proporciona herramientas para manipular datos científicos, como el cálculo de las coordenadas celestiales, la corrección de las distorsiones cromáticas y la interpolación de datos.
- Análisis de datos: AstroPy brinda una variedad de herramientas para el análisis de datos, como el ajuste de modelos, la detección de objetos y el análisis espectral.

- Visualización de datos: AstroPy proporciona una variedad de herramientas para la visualización de datos, como la creación de mapas de color y la creación de gráficos de línea.
- Integración con otros paquetes: AstroPy está diseñado para ser fácilmente integrable con otros paquetes populares de Python para ciencia de datos y análisis, como NumPy, SciPy y Matplotlib.

5.2.Limpieza de datos

En esta etapa filtramos los datos de manera de incluir en los algoritmos solo objetos con magnitudes de valores adecuados. Por ejemplo, en los catálogos es posible encontrar magnitudes con el valor 99, debido a que durante la etapa de preprocesamiento de las imágenes y el cálculo de las magnitudes mediante el software (SExtractor) las fuentes luminosas muy débiles a las que fue difícil medir algunas señales se les asignó un valor para la magnitud de 99 *(Muñoz et al. 2013).* Luego, en nuestro análisis, sólo incluimos objetos con valores de magnitud menores < 99, reduciendo la cantidad de registros válido.

Luego se realiza la construcción del dataset necesario para las etapas posteriores de análisis y exploración de datos, validación de valores, distribución y balanceo.

Como también la corrección de nombres de columnas en DataFrame (g-z) y (z-K)

Corrección de nombres 2 de columnas en DataFrame (u-i) y (i-K)

```
df_gv_ui_iK_stars_uiK_all = df_gv_ui_iK_stars_uiK_all.rename(columns = {0:'u-i',1:'i-K',2:'stars_uiK'})
df_gv_ui_iK_all = df_gv_ui_iK_all.rename(columns = {0:'u-i',1:'i-K'})
```

```
display(df_gv_ui_iK_all.head(5))
```

	u-i	i-K
2	1.853043	1.756881
6	4.129240	1.542703
11	3.353079	2.088546
20	2.554876	1.676256
37	6.994230	1.185436

Se construyen varios tipos de dataset, para realizar la exploración de los datos y entendimiento, relación y distribución de los valores del dataset, por ello uno de los dataset que se desarrolló con en todas las diferencias de variables utilizadas en los diagramas “Color-color ui-K” y diagramas “Color-color gz-K”

```
display (df_gv_stars_ui_iK_gz_iz_zK_all.head())
```

	u-i	i-K	g-z	i-z	z-K
98	1.631947	-0.419604	0.620722	0.040039	-0.459643
149	4.912361	1.097898	3.272825	0.584925	0.512973
160	3.668308	0.541103	1.905350	0.267365	0.273739
172	1.032127	-0.714255	0.244297	-0.039270	-0.674984
225	4.554579	1.257158	3.562897	0.769684	0.487474

Luego se construyó el dataset final para realizar el análisis y descripto en el trabajo *(Muñoz et al. 2013)*, con las variables “g-z”, “z-K".

5.3. Selección y Construcción de Variable (Feature Engineering)

Luego de la etapa de selección de variable correspondientes a las magnitudes fotométricas “g”, “z”,”K”, se continua con la construcción de nuevas variables del resultado de la diferencia de las variables “g-z”, “z-K" para simular la información de los diagramas fotométricos “gz-K”. Estas nuevas variables calculadas son del resultado de la diferencia de las variables “g-z”, “z-K" y serán variables de entrada para el entrenamiento de los modelos no supervisados de machine learning en las etapas posteriores.

```
display (df_gv_stars_gz_zK_all.head())
```

	g-z	z-K
98	0.620722	-0.459643
149	3.272825	0.512973
160	1.905350	0.273739
172	0.244297	-0.674984
225	3.562897	0.487474

5.4. Análisis Preliminar de los Datos

Un punto importante en el análisis exploratorio inicial del dataset es la distribución de objetos o estrellas en la dimensión de análisis, para ello mediante la exploración visual utilizando varios tipos de gráficos exploramos el dataset con los valores identificados.

```
display (df_gv_stars_gz_zK_all.describe())
```

	g-z	z-K
count	10503.000000	10503.000000
mean	1.842180	0.124724
std	1.078318	0.452721
min	-5.639158	-10.272227
25%	0.782385	-0.229249
50%	1.933096	0.267059
75%	2.753682	0.430219
max	7.523455	4.833558

La figura 3 muestra la distribución de puntos (estrellas) usando diferencia de magnitudes.

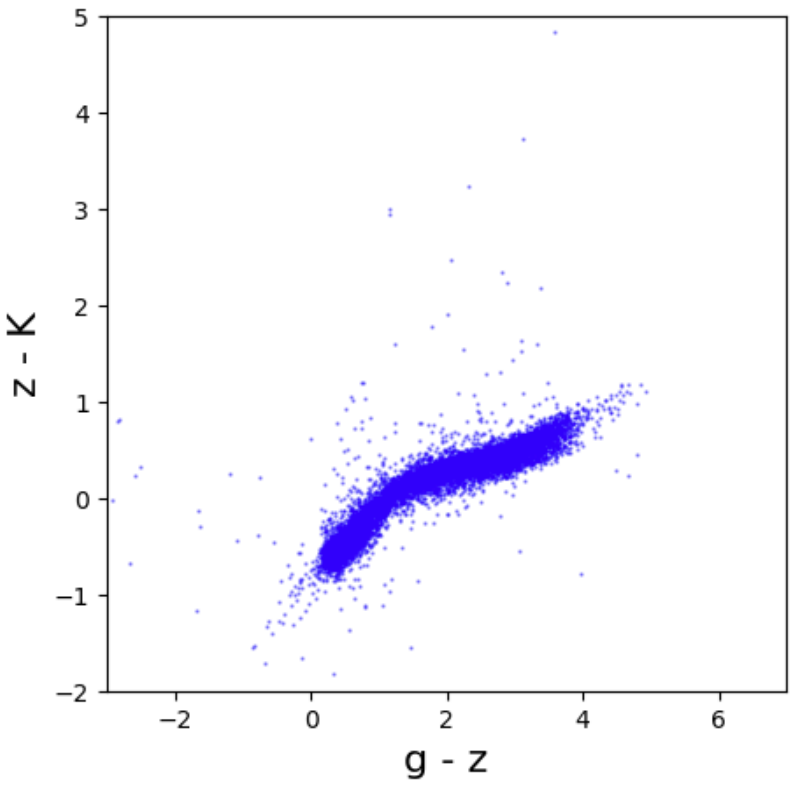


Figura 3. Corresponde a la distribución de los puntos del dataset. El eje coordenada x corresponde a la diferencia (g-z) mientras que el eje de coordenada Y corresponde a la diferencia (z-K)

La Figura 4 continua con el análisis exploratorio del dataset con la librería Matplotlib y muestra de la distribución completa de todos los objetos (estrellas) que contiene en loa eje de coordenadas x, versus las variables de diferencia de magnitudes (u-i),(ik), (g-i), (i-z), (i-K).

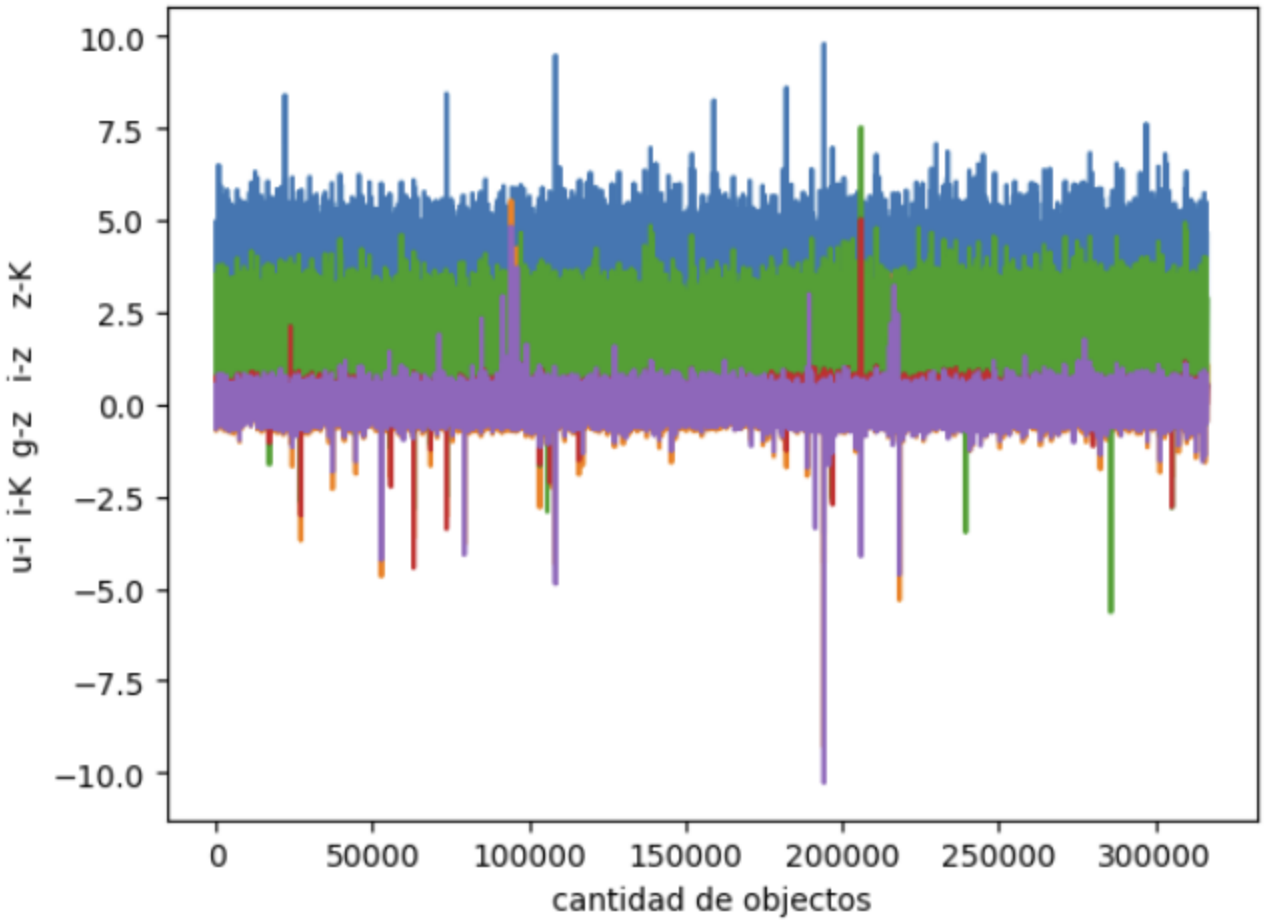


Figura 4. Corresponde a la distribución de los puntos del dataset. El eje de la coordenada X corresponde a todos los objetos o estrellas y el eje coordenada corresponde a las diferencias ('u-i i-K g-z i-z z-K')

La siguiente figura fue obtenida con la librería seaborn de matplotlib, con la visualización "**pairplot**" la cual traza las relaciones por pares en un conjunto de datos. Esta función crea una cuadrícula de ejes de modo que cada variable numérica en los datos se compartirá en los ejes "y" en este caso ((g-i) (i-z) (i-K)) en una sola fila "y" los ejes "x" en una sola columna ((g-i) (i-z) (i-K)).

Las gráficas diagonales se tratan de manera diferente: se dibuja una gráfica de distribución univariada para mostrar la distribución marginal de los datos en cada columna.

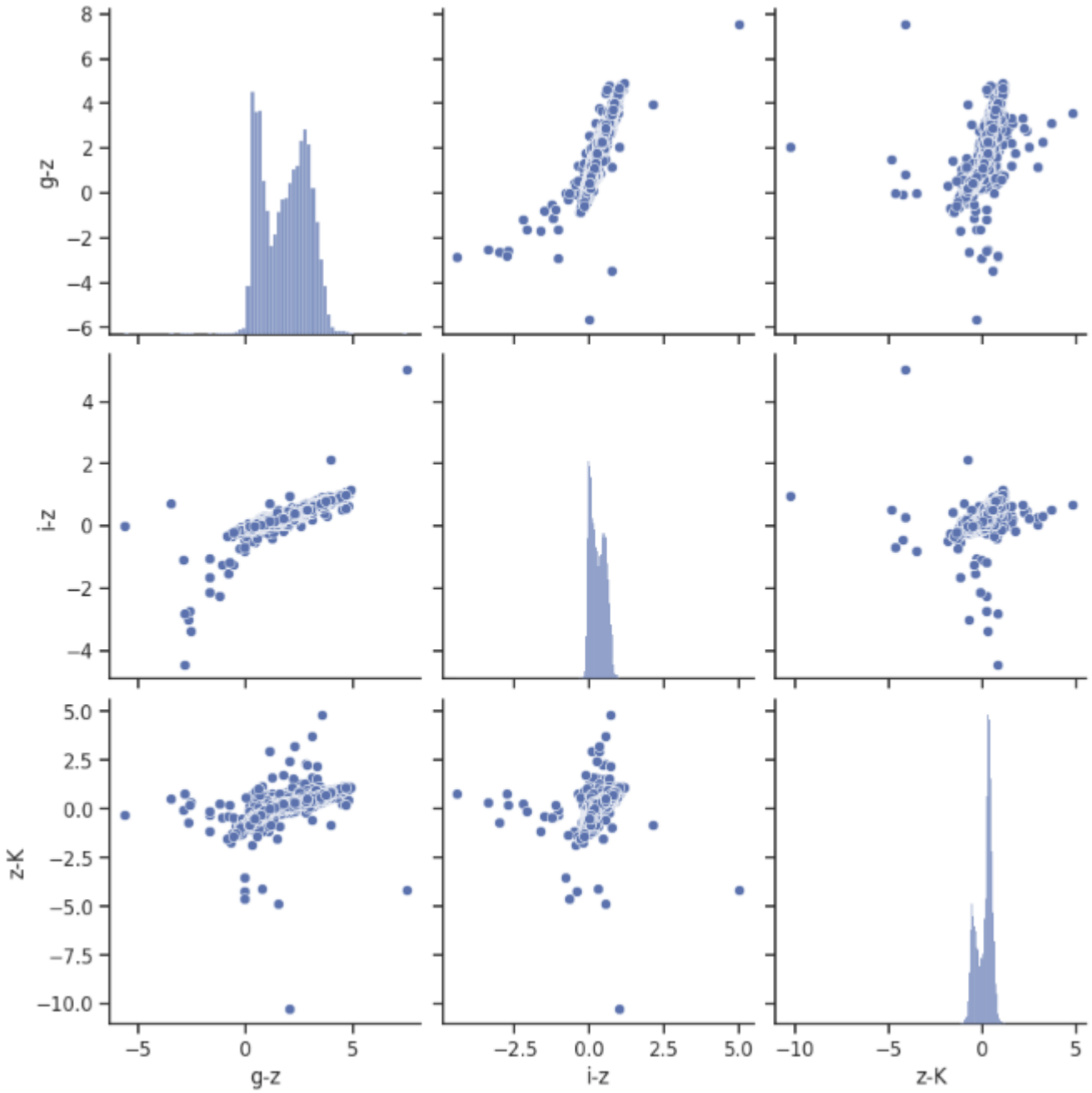


Figura 5. En eje coordenada las variables (g-z) (i-z) (z-K) juntas y relacionadas con el eje coordenada y con las mismas variables. La gráfica diagonal representa la distribución univariada marginal de los datos.

La siguiente figura representa la visualización "**heatmap**" o mapa de calor usa una función a nivel de ejes y dibujará el mapa de calor en los ejes actualmente activos si no se proporciona ninguno al argumento del eje. Parte de este espacio de ejes se tomará y se usará para trazar un mapa de colores.

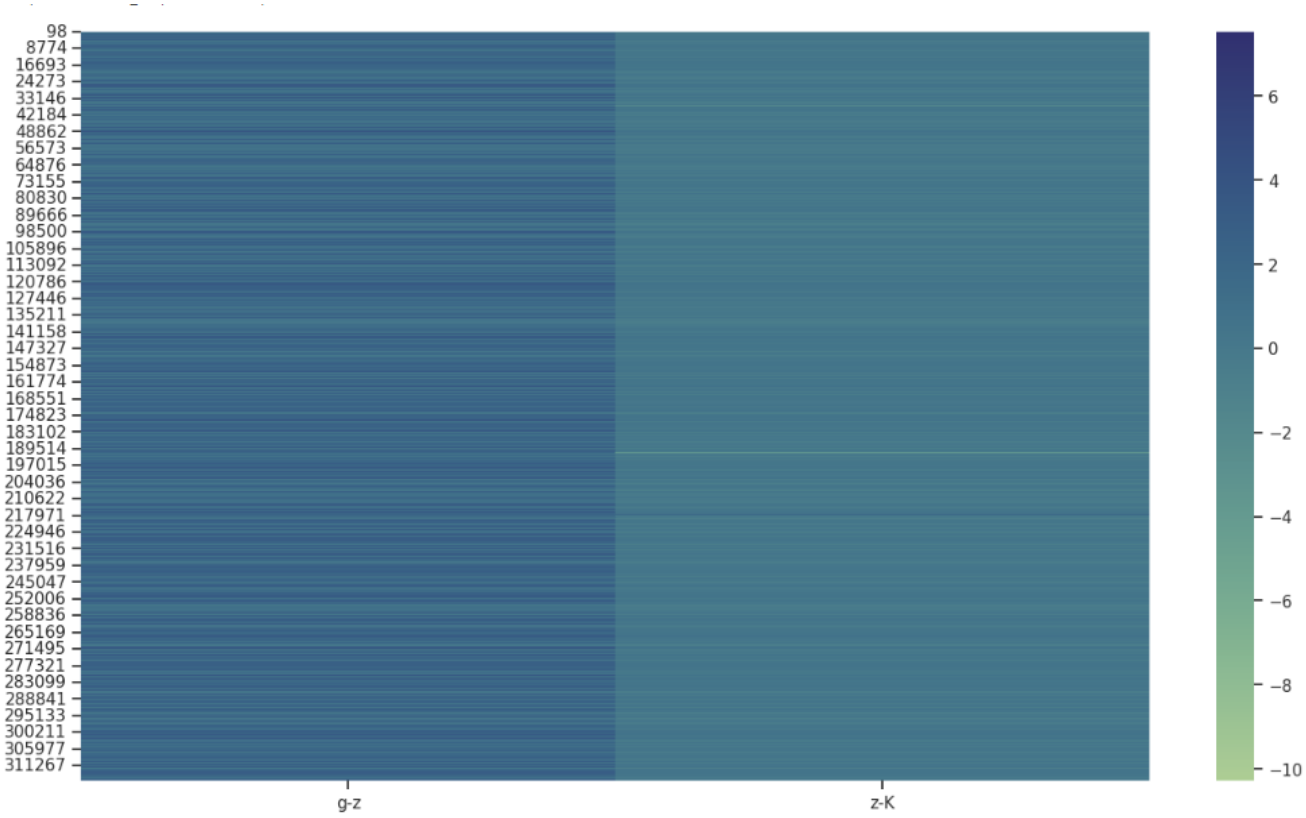


Figura 6. Mapa de calor usando las variables (g-z) (z-K) juntas y relacionadas con el eje coordenada y con las mismas variables.

5.5. Definición del Modelo Machine Learning (Modelos no supervisados)

En esta etapa seleccionamos dos algoritmos de machine learning no supervisados, para agrupamientos de elementos no etiquetados[5], como lo son: K-Means y Gaussian Mixtures Model (GMM).

5.5.1. K-Means

Inicialmente utilizamos el algoritmo de clusterización K-Means, el cual permite buscar un número predeterminado de clústeres dentro de un conjunto de datos multidimensionales sin etiquetar utilizando un centroide para agrupar y modelar cada clúster.

Para ello utiliza una concepción simple de cómo se ve la agrupación óptima:

- El "centroide" es la media aritmética de todos los puntos que pertenecen al grupo.
- Cada punto está más cerca de su propio centro de grupo que de otros centros de grupo.

Estos dos supuestos son la base del modelo de K-Means.

El modelo K-Means por su definición no probabilística y su uso de distancia simple desde el centroide del clúster para asignar la pertenencia, *(Morissette, L., & Chartier, S. 2013)*, puede ocasionar un deficiente desempeños en muchos casos.

Dos de los parámetros empleados en este tipo de análisis definen la varianza interna de un clúster (W(Ck)W(Ck)) como:

- La suma de las distancias euclídeas al cuadrado entre cada observación (xixi) y el centroide (μμ) de su clúster. Esto equivale a la suma de cuadrados internos del clúster.
- La suma de las distancias euclídeas al cuadrado entre todos los pares del clúster, dividida entre el número de observaciones del clúster.

5.5.2. Algoritmo de Mezcla Gaussiana (GMM)

Dado que el algoritmo K-Means solo utiliza una distancia simple como criterio de agrupación, lo que puede producir resultados deficientes, además de k-means, utilizamos el algoritmo de mezcla Gaussiana (GMM). este es un algoritmo de clusterización cuyo criterio de agrupamiento es probabilístico y se basa en la identificación de la mezcla de distribuciones de probabilidad gaussianas multidimensionales que modelen mejor cualquier conjunto de datos de entrada *(Muratov, Alexander L., and Oleg Y. Gnedin. 2010.*). Este algoritmo puede verse como una extensión de las ideas detrás de k-means, pero también pueden ser una gran herramienta para realizar una estimación más allá de la simple agrupación en clústeres.

En este trabajo, utilizamos el algoritmo GMM disponible en la librería Scikit-Learn y el método *predict_proba* para medir la probabilidad de que un objeto pertenezca a un grupo.

Uno de los hiperparámetros más importantes del algoritmo GMM es ***covariance_type***, que controla el tipo de covarianza. El valor predeterminado para ***covariance_type*** es ***diag***, lo que significa que el tamaño del clúster a lo largo de cada dimensión se puede establecer de forma independiente. Otro valore posibles de

este hiperparámetro es ***spherica***l y ***full***. El valor ***spherical*** permite restringir la forma del clúster de modo que todas las dimensiones sean iguales mientras que full permite modelar cada grupo como una elipse con orientación arbitraria. Como el método GMM tiene un modelo probabilístico interno, también es posible encontrar clústeres probabilísticos, por ello el paquete de Scikit-Learn hace esto usando el método ***predict_proba*.** Esto devuelve una matriz de tamaño *[n_samples, n_clústers]* que mide la probabilidad de que cualquier punto pertenezca al grupo.

5.5.3. Estimación del Número de Componentes en K-means. Método Elbow

Para el método de agrupamiento de k-means, el enfoque más común para responder a esta pregunta es el llamado método Elbow o del codo, *(Morissette, L., & Chartier, S. 2013),* implica ejecutar el algoritmo varias veces en un bucle, con un número creciente de opciones de grupos y luego trazar una puntuación de agrupación en función del número de grupos para encontrar las sumas de cuadrados dentro del clúster (WCSS) para cada número de clústeres.

El número óptimo de clústeres que deben formarse varía según la distribución de los datos.

5.5.4. Estimación del Número de Componentes en Modelo mezcla Gaussian (GMM)

El hecho de que GMM sea un modelo generativo nos brinda un medio natural para determinar el número óptimo de componentes para un conjunto de datos. Un modelo generativo es inherentemente una distribución de probabilidad para el conjunto de datos, por lo que simplemente podemos evaluar la probabilidad de los datos bajo el modelo, utilizando la validación cruzada para evitar un ajuste excesivo. Otro medio de corregir el sobreajuste es ajustar las probabilidades del modelo utilizando algún criterio analítico como el criterio de información de Akaike (AIC) o el criterio de información bayesiano (BIC). El estimador GMM de Scikit-Learn en realidad incluye métodos integrados que calculan ambas métricas.

5.1. Validación de Resultados

Los métodos no supervisados compuestos por métricas de evaluación que no se pueden comparar con valores predefinidos, pero se pueden comparar entre diferentes modelos de agrupamiento y, por lo tanto, podemos usar una técnica para elegir el mejor modelo. Algunos métodos que nos permiten comparar nuestros modelos son:

Modelos K-means:

- Medida de Silhouette
- SSE (Sum of squared errors)
- Medida de Davies-Bouldin
-

Modelos GMM

- Criterio de información de Akaike (AIC)
- Criterio de información bayesiano (BIC)

5.1.1. AIC y BIC

GMM sea un modelo generativo nos brinda un medio natural para determinar el número óptimo de componentes para un conjunto de datos. Un modelo generativo es inherentemente una distribución de probabilidad para el conjunto de datos, por lo que simplemente podemos evaluar la probabilidad de los datos bajo el modelo, utilizando la validación cruzada para evitar un ajuste excesivo.

Otro medio de corregir el sobreajuste es ajustar las probabilidades del modelo utilizando algún criterio analítico como el criterio de información de Akaike (AIC) o el criterio de información bayesiano (BIC). El estimador GMM de Scikit-Learn en realidad incluye métodos integrados que calculan ambos.

5.1.2. Medida de Silhouette

El método de Silhouette utiliza 2 medidas principales: cohesión y separación. La cohesión no es más que la compacidad o la estrechez de los puntos de datos dentro de un clúster, (*Rousseeuw, P. J. 1987).* Hay básicamente 2 formas de calcular la cohesión:

- Cohesión basada en gráficos
- Cohesión basada en prototipos

5.1.3. Medida de Davies-Bouldin.

La métrica Davies-Bouldin es una medida de la similitud entre los puntos de un clúster y el centroide del clúster. Es utilizada para evaluar la calidad de un agrupamiento (clustering) y para comparar diferentes agrupamientos y a su vez con otras métricas. La idea detrás de esta métrica es que un buen agrupamiento debería tener clústeres compactos y bien separados entre sí.

Esta métrica se calcula dividiendo cada clúster en dos partes: el primer término representa la distancia promedio entre cada punto del clúster y el centroide de ese clúster, y el segundo término representa la distancia mínima entre el centroide del clúster actual y el centroide de cualquier otro clúster. El valor final se obtiene sumando estos dos términos para cada clúster y luego tomando el promedio de estos valores.

La métrica Davies-Bouldin mide la similitud entre los puntos de un clúster y su centroide, y la separación entre los centroides de diferentes clústeres. Un valor bajo de Davies-Bouldin indica un buen agrupamiento, mientras que un valor alto indica un mal agrupamiento.

5.1.4. Medida de Calinski-Harabasz

La métrica Calinski-Harabasz es una métrica utilizada para evaluar la calidad de un agrupamiento (clustering) y para comparar diferentes agrupamientos. Se basa en la comparación entre la varianza dentro de los clústeres y la varianza entre los clústeres.
Esta métrica se calcula dividiendo la suma de las varianzas entre los clústeres entre la suma de las varianzas dentro de los clústeres. El valor final se obtiene dividiendo el número de puntos menos el número de clúster, entre el número de clúster menos uno.

La métrica Calinski-Harabasz mide la relación entre la varianza dentro de los clústeres y la varianza entre los clústeres. Un valor alto de Calinski-Harabasz indica un buen agrupamiento, mientras que un valor bajo indica un mal agrupamiento. A medida que aumenta la cantidad de clústeres, el valor de esta métrica aumenta.

5.2. Resultados

5.2.1. Resultados utilizando K-Means utilizando el método Elbow o Codo

En la figura 7 se identificar donde la curva se aplana y cambia de pendiente entre los valores 3 (nro Clúster = 3), después de ese punto, no se observa una gran diferencia en la formación de clúster adicionales.

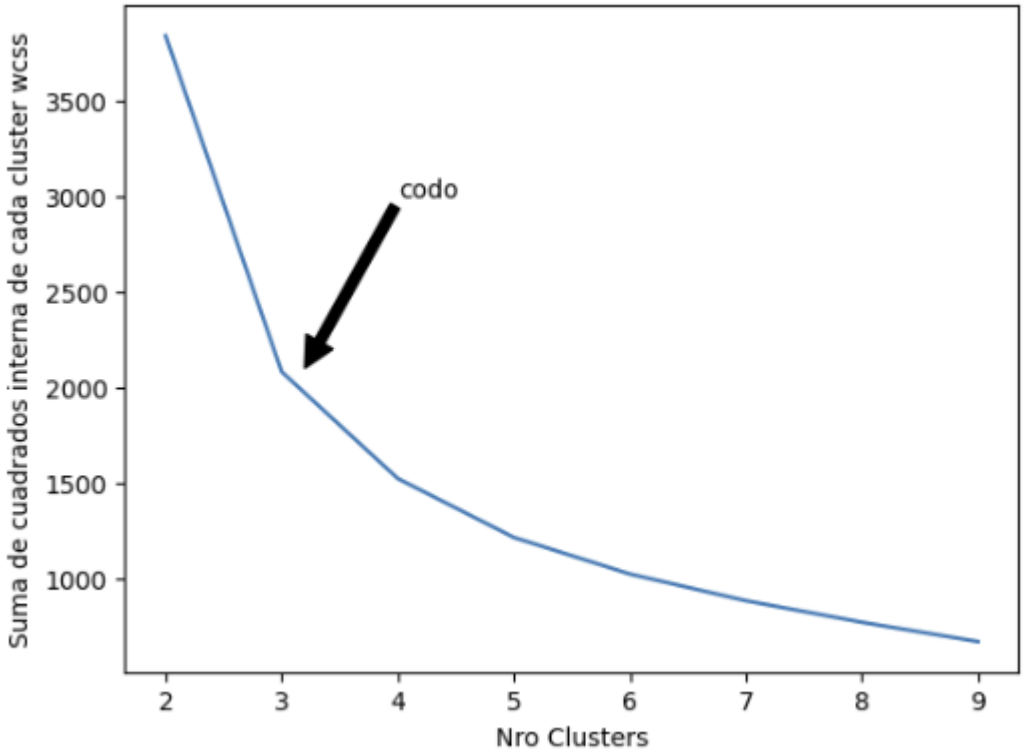


Figura 7. Método Elbow o Codo, en el eje x de la gráfica es el número de clúster, en el eje y es la suma de cuadrados dentro del clúster (WCSS) para cada número de clústeres.

5.2.2. Resultados utilizando K-Means utilizando el método Silhouette

En la figura 8 podemos ver la optimización del número de clústeres utilizando gráfico de Silhouette. Si usamos estos diagramas, podemos optimizar mediante una métrica el número de clústeres elegidos aquí, la línea verde representa el valor promedio (avg) y la línea roja el valor máximo (max), en la medida de Silhouette se utiliza en valor máximo

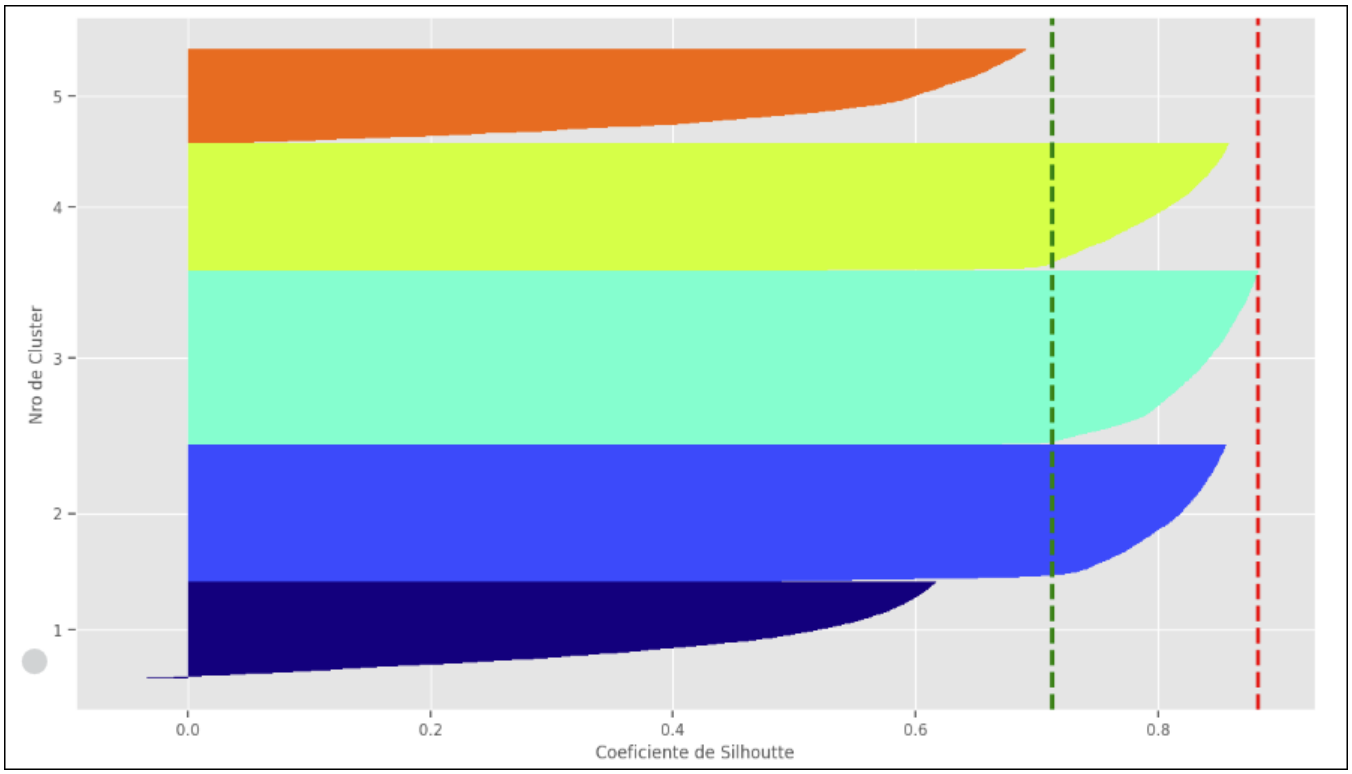


Figura 8. Método Silhouette basado en gráfico. El eje x de la gráfica corresponde al coeficiente de Silhouette, donde se considera el valor máximo (rojo) optimo, el eje y el número de clústeres estimados.

La siguiente tabla contiene las estimaciones de una secuencia de 10 valores de medida de Silhouette y puede comprobarse que el valor optimo según esta medida es “0.5658952190068853” equivalente a 3 clústeres, el valor más alto posible es mejor.

Descripción	k(clusters)	Medida de Silhouette
Silhouette score for k(clusters)	3	0.5658952190068853
Silhouette score for k(clusters)	4	0.528778994700878
Silhouette score for k(clusters)	5	0.467094708775932
Silhouette score for k(clusters)	6	0.46777857161218706
Silhouette score for k(clusters)	7	0.4680443128142951
Silhouette score for k(clusters)	8	0.4725602400801778
Silhouette score for k(clusters)	9	0.4550708464378773
Silhouette score for k(clusters)	10	0.42705620069289607
Silhouette score for k(clusters)	11	0.43443669379440986

Tabla 3: Valores de medida de Silhouette para k(clusters)

5.2.3. Dataset de ingreso a los modelos

Para el ingreso a los modelos se utilizará el dataset con los features (g-z) (z-K)

df_gv_stars_gz_zK_all

	g-z	z-K
98	0.620722	-0.459643
149	3.272825	0.512973
160	1.905350	0.273739
172	0.244297	-0.674984
225	3.562897	0.487474
...	...	...
316366	0.744181	-0.304682
316409	0.457062	-0.467043
316458	2.803841	0.417532
316459	0.436474	-0.496006
316494	2.889135	0.531803

5.2.4. Configuración del modelo KMeans

La configuración del modelo KMeans con parámetro de algoritmo k-means++, con 3 clúster, valor de inicialización 7, estado random 13 y máxima cantidad de iteraciones en 512, estos parámetros fueron el resultado de varios experimentos combinando los hiperparametros adecuados.

```
# Nro de Clusters o grupos
n_clusters=3

# Entrenamiento modelo K-means
from sklearn.cluster import KMeans
kmeans = KMeans(n_clusters , init = 'k-means++', n_init=7, random_state=13,max_iter=512)

kmeans.fit(data)

# Predicciones de K-means
pred = kmeans.predict(data)
frame = pd.DataFrame(data)
frame['cluster'] = pred
frame.columns = ['g-z','z-K','cluster']
```

El siguiente gráfico, figura 9 representa la distribución en 3 clústeres que sería el valor optimo según el método de elbow y el coeficiente de Silhouette.

Como así también la cantidad de elementos en cada uno de los 3 clúster y sus respectivos centroides.

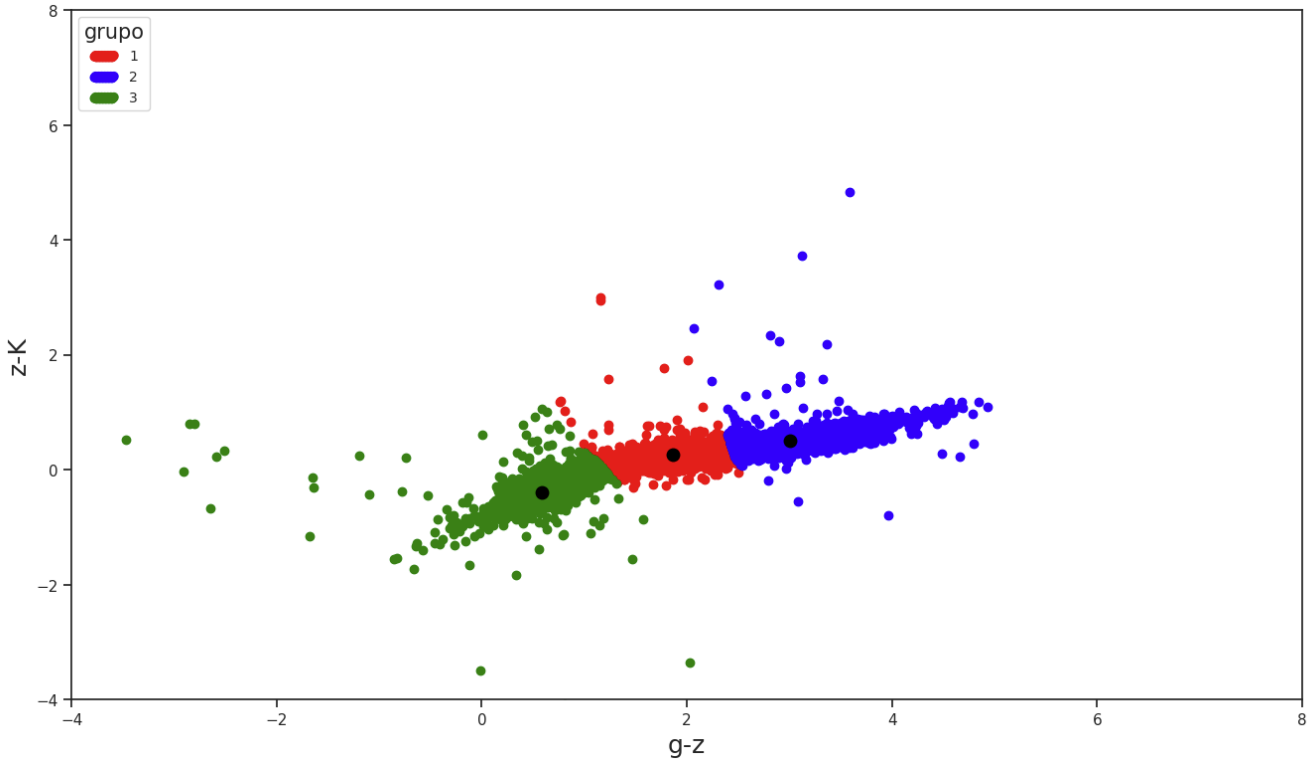


Figura 9. K-Means clustering con 3 grupos y centroides

Para los 3 grupos de un total de 28406 valores se detallan la cantidad de valores calculados:

```
Total Fig 9:  28406
grupo 1:  6206
grupo 2:  10440
grupo 3:  11097
```

5.2.1. Componentes principales PCA y K-Means

Es posible aplicar un análisis de componentes principales PCA antes de un algoritmo de agrupación como K-means el cual mejora los resultados de la agrupación. Un paper que profundiza el análisis entre PCA y K-Means. *(Ding, C., & He, X. 2004)*, presentando resultados que escenarios de agrupación usando K-Means usando análisis de componentes principales PCA mostraron beneficios y valor bajando el tiempo de entrenamiento, reduciendo la dimensionalidad y el conjunto de datos.

En esta estrategia que se evaluó fue la incorporación del análisis de componentes principales PCA, se diagrama el siguiente gráfico, figura 10 representa la distribución en 3 clústeres que sería el valor optimo según los métodos antes mencionados.

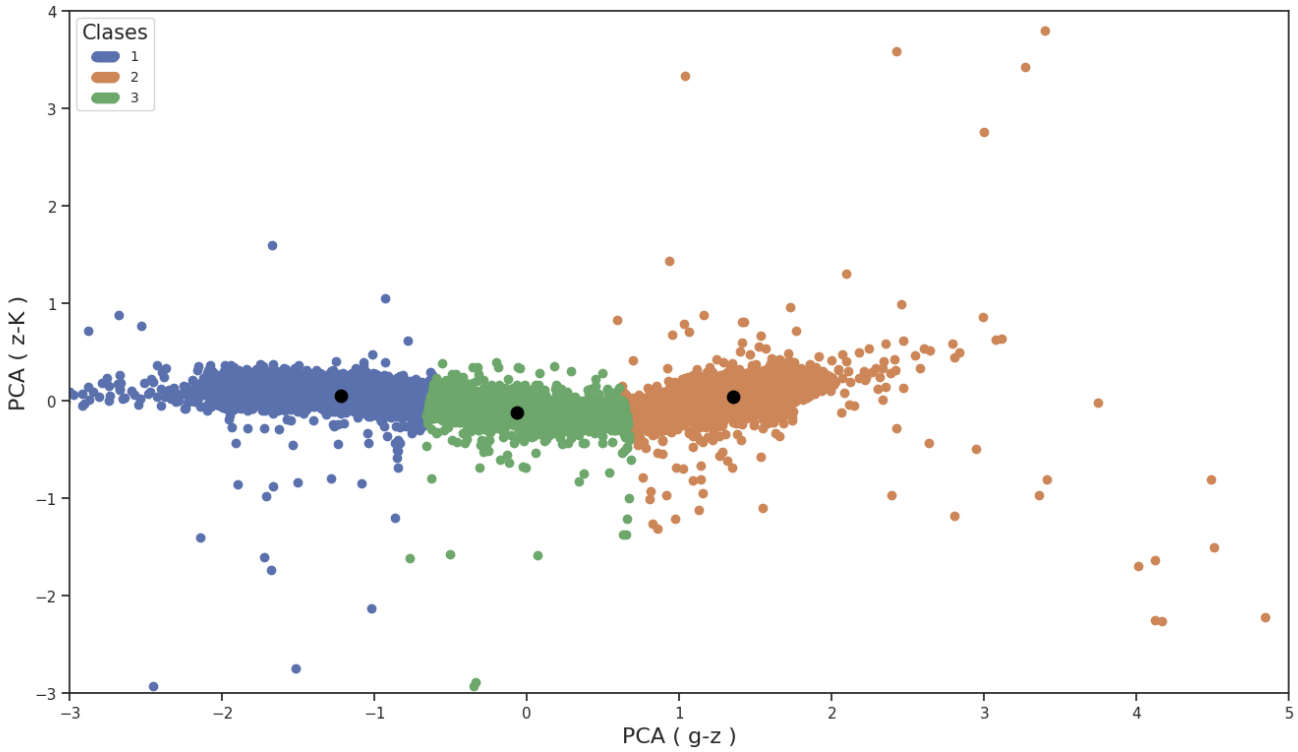


Figura 10. K-Means clustering con 3 grupos aplicando componentes principales PCA

Para los 3 grupos de un total de 21006 valores se detallan la cantidad de valores calculados.

```
Total Fig 10:  21006
Cluster 1:  7522
Cluster 2:  7146
Cluster 3:  6338
```

5.2.2. Resultados utilizando GMM, Modelo de Mezcla Gaussiana

Analizando los criterios AIC y el BIC en función del número de componentes GMM para el conjunto de datos del catálogo fotométrico analizado, el valor de clúster sugeridos para el entrenamiento de modelos GMM es AIC = 15 y BIC = 9

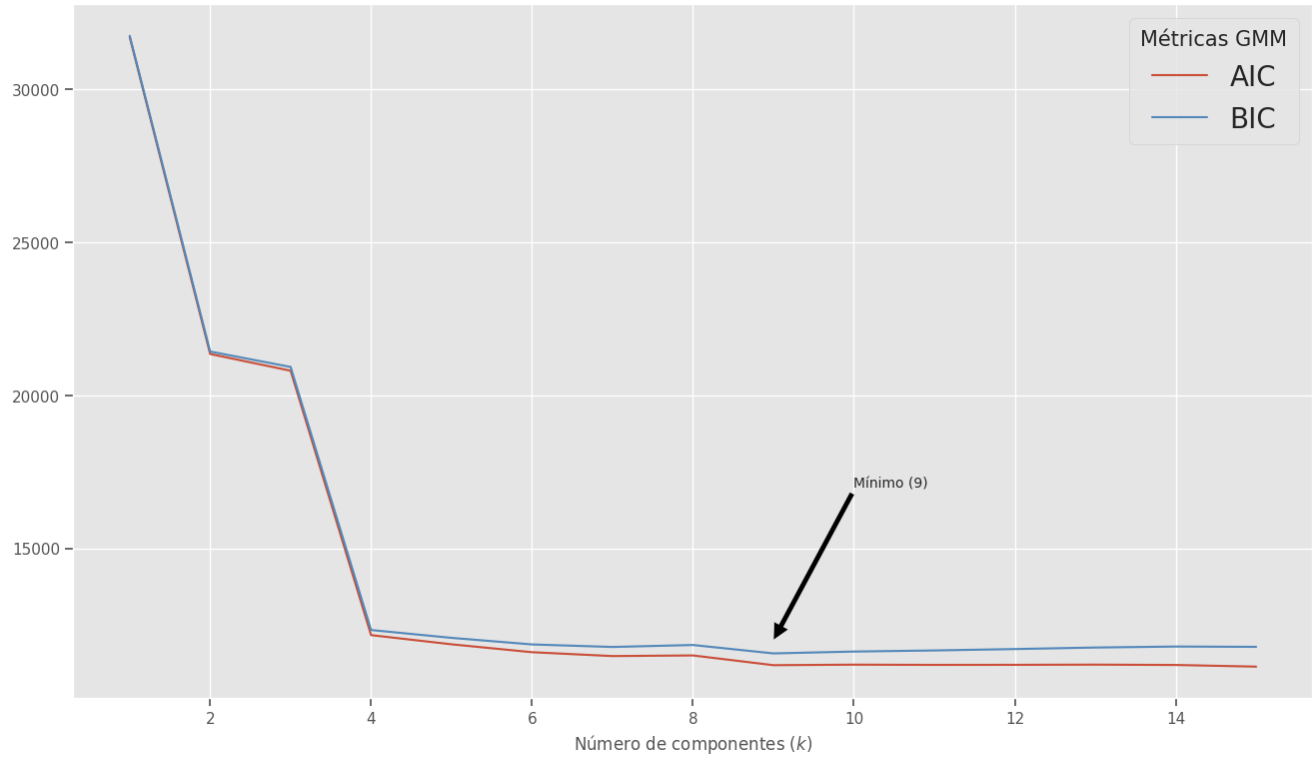


Figura 11. Métricas GMM usando criterios AIC y BIC

El siguiente gráfico combina todos los métodos de covarianza ('full' 'spherical', 'tied', 'diag') soportados por el modelo GMM de la librería **sklearn.mixture** para 10 valores distintos de k o grupos, realizando la medición de criterios BIC para encontrar el valor óptimo de k, es este caso el valor encontrado es 9 grupos.

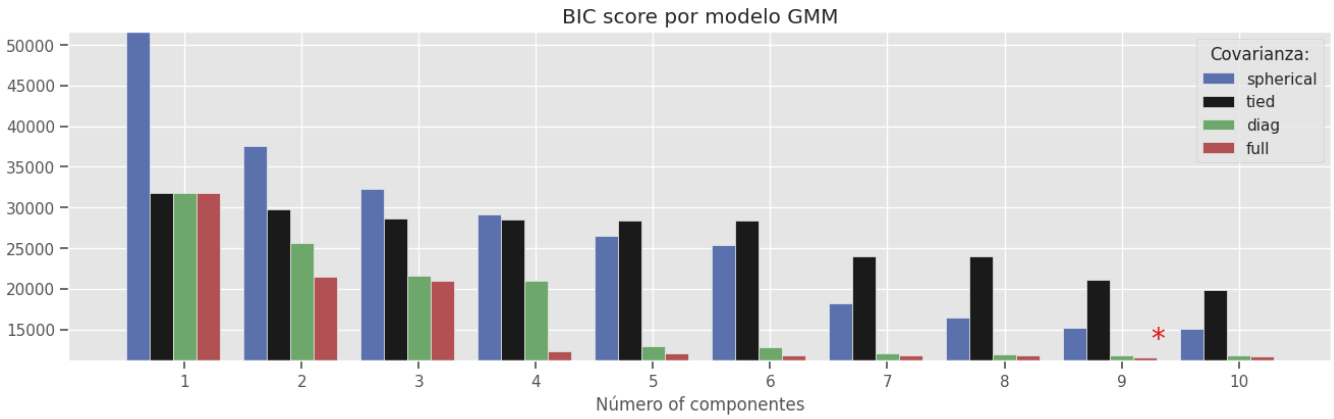


Figura 12. Diagrama que combina los cuatro modos de covarianza en 10 posibles tipos de grupos o clúster para identificar la mejor combinación usando los criterios AIC y BIC confirma el valor k = 9

5.2.1. Configuración del modelo GMM

La configuración del modelo GMM o Gaussian mixture con parámetro de covarianza full, con 9 clúster, valor de inicialización 7, estado random 13 y máxima cantidad de iteraciones en 512, estos parámetros fueron el resultado de varios experimentos principalmente de metricas como BIC y AIC combinando los hiperparametros adecuados.

```
# Nro de clusters o grupos
n_clusters=9

# Entrenamiento modelo Gaussian Mixture GMM
from sklearn.mixture import GaussianMixture
gmm = GaussianMixture(n_components=n_clusters, n_init=7, random_state=13,max_iter=512,covariance_type='full')
        # covariance_type='full' 'spherical', 'tied', 'diag'
gmm.fit(data)

# Predicciones de GMM
labels_gmm = gmm.predict(data)
frame_gmm = pd.DataFrame(data)
frame_gmm['cluster'] = labels_gmm

frame_gmm.columns = ['g-z','z-K','cluster']
```

El diagrama 13 representa el mejor modelo de todos los analizados usando un modelo GMM o Gaussian mixture con parámetro de covarianza full, con 9 clúster, sugeridos por las métricas

BIC y AIC, luego se realizaran otras comparaciones para corroborar los parámetros elegidos para este modelo.

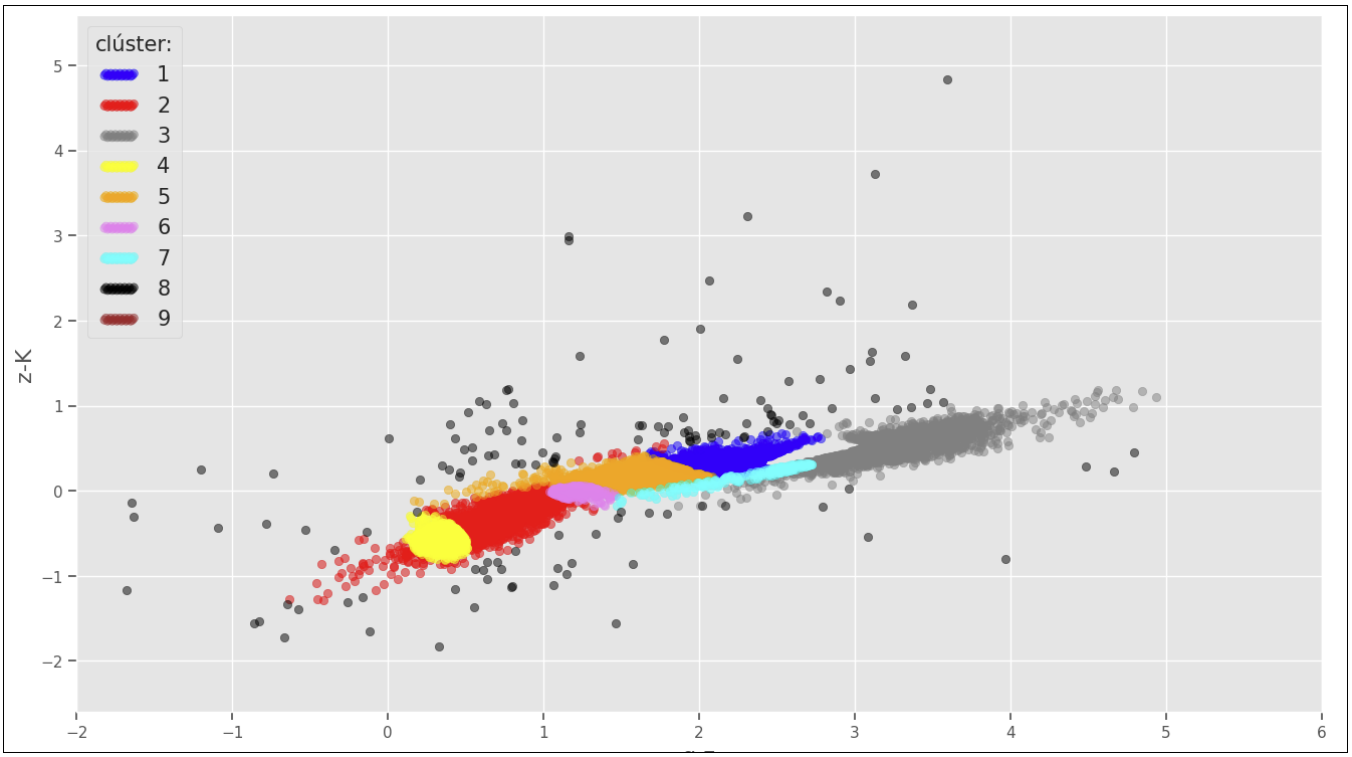


Figura 13. Optimización GMM con 9 clústeres usando criterios BIC

Para los 9 grupos sugeridos por AIC de un total de 29966 valores se detallan la cantidad de valores calcu lados.

```
Nro total Fig 13:  29966
Cluster 1:  3086
Cluster 2:  6507
Cluster 3:  4230
Cluster 4:  4608
Cluster 5:  6186
Cluster 6:  3441
Cluster 7:  657
Cluster 8:  765
Cluster 9:  486
```

5.2.2. Resultados comparación de modelos no supervisados usando métricas

La comparación entre el método de clusterización o agrupamiento mediante K-means versus GMM se puede realizar utilizando distintas métricas, algunas ya mencionadas en K-means como la medida de Silhouette y otras medidas mencionadas en el método GMM (Gaussian Mixture) como el criterio de información de Akaike (AIC) o el criterio de información bayesiano (BIC).

También se incluyeron métricas adicionales como Davies Bouldin y Calinski Harabas incluidas en la librería de métricas de **sklearn.metrics**

```
from sklearn.metrics import silhouette_score, davies_bouldin_score
from sklearn.metrics import calinski_harabasz_score
from sklearn.mixture import GaussianMixture
```

Los valores óptimos para las distintas métricas se detallan a continuación, en algunos casos como BIC, AIC, Davies un valor bajo es mejor, en otros casos como Silhouette y Calinski un valor alto es mejor.

Significado de métricas	
BIC	Bajo es mejor
AIC	Bajo es mejor
Silhouette	Alto es mejor
Davies	Bajo es mejor
Calinski	Alto es mejor

Tabla 4: Significado de métricas para evaluar K-means vs GMM

La siguiente tabla es el resultado de los cálculos de métricas usando la librería **sklearn.metrics**, se realizaron los cálculos para 10 valores de k o clúster.

La tabla 5 es fundamental para la realización de los gráficos comparativos de análisis y selección de que método es el óptimo para este caso de uso, K-means k=3 (Silhouette) y GMM k=9 (BIC)

Para ello se crearon funciones para el cálculo de todas las métricas para realizar la comparación de desempeño de los modelos K-means++ versus GMM covarianza full.

k	BIC	AIC	Silhouette	Davies	Calinski
3	20930.02792	20806.61784	0.572469	0.55096	31012.47654
4	12332.93366	12165.96709	0.518167	0.59889	29542.25695
5	12076.3449	11865.82183	0.48067	0.6488	28443.43228
6	11861.67464	11607.59507	0.466616	0.66007	27415.16055
7	11779.07558	11481.43951	0.469678	0.62474	26712.22502
8	11843.38263	11502.19007	0.446396	0.64491	26177.80209
9	11568.38565	11183.63659	0.451934	0.62382	26969.16358
10	11628.97542	11200.66986	0.441733	0.6386	27194.78496
11	11665.20795	11193.3459	0.444378	0.62266	27007.80407

Tabla 5: Valores calculados usando (g-z z-K) y la librería **sklearn.metrics**

La siguiente secuencia de gráficos (Fig 14 a Fig17) muestran los resultados de comparar distintas métricas para medir el desempeño de los dos modelos seleccionados K-means++ con GMM covarianza full

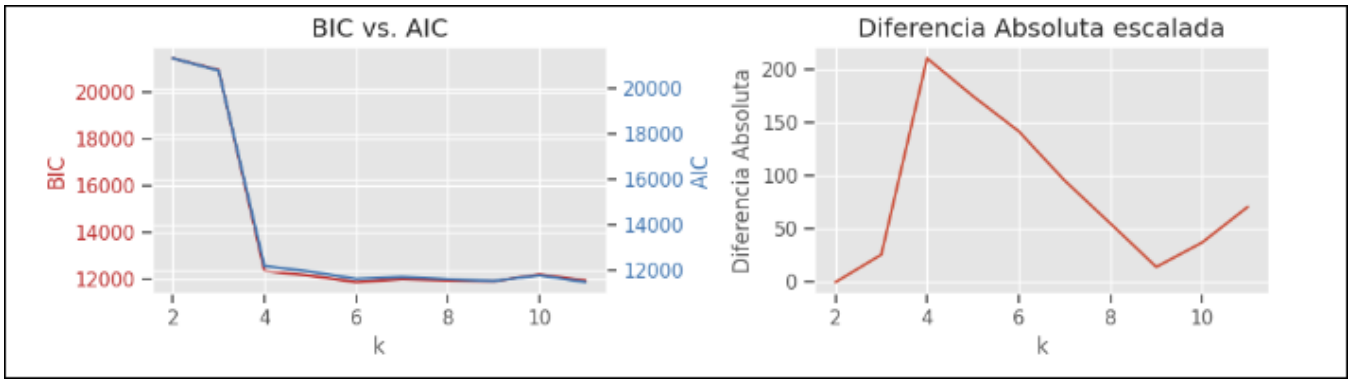


Figura 14. Comparación de métricas entre criterio de información bayesiano BIC y criterio de información de Akaike (AIC)

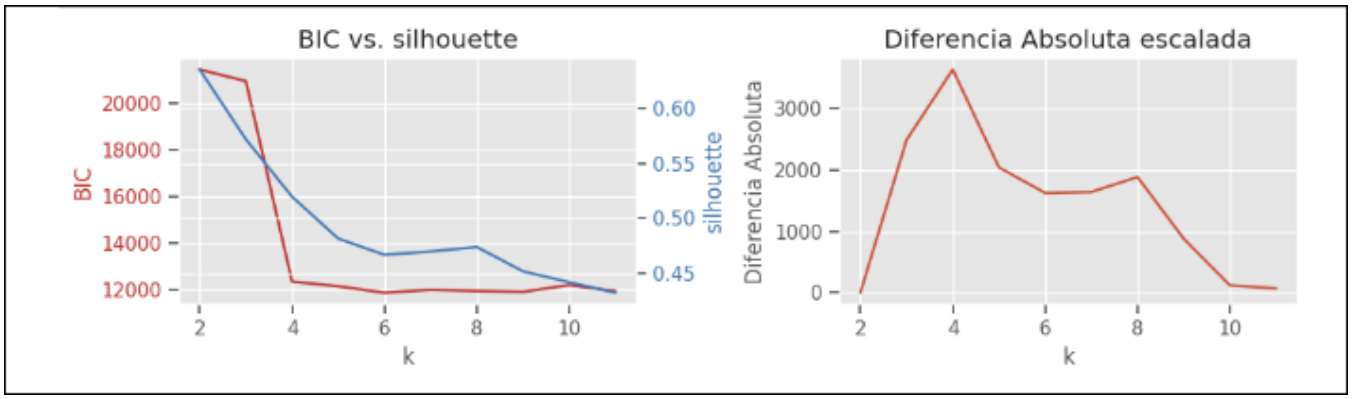


Figura 15. Comparación de métricas entre criterio de información bayesiano BIC y medida de Silhouette

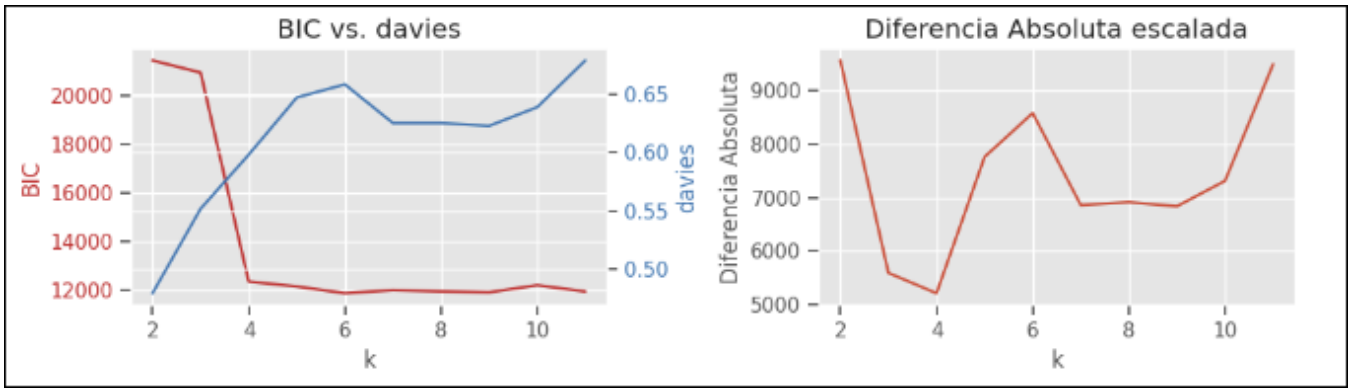


Figura 16. Comparación de métricas entre criterio de información bayesiano BIC y medida de Davies Bouldin

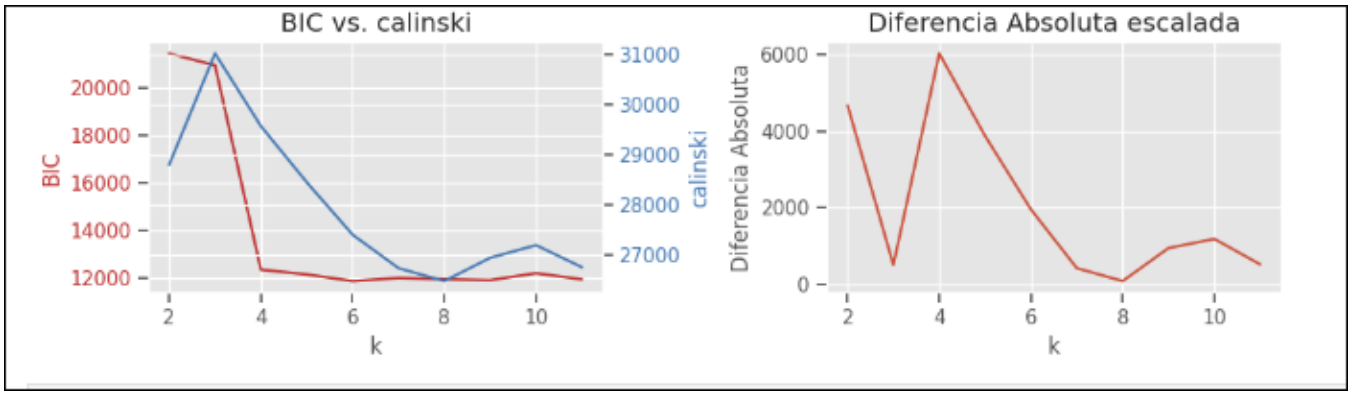


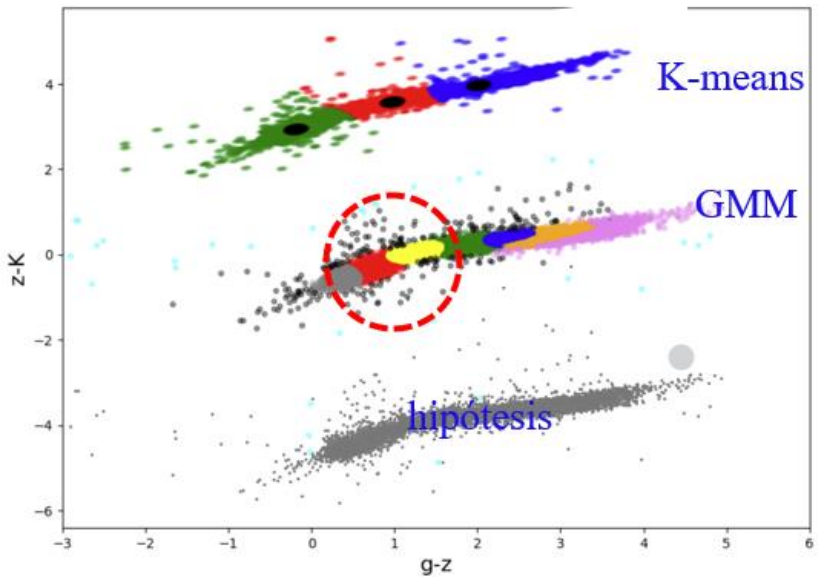
Figura 17. Comparación de métricas entre criterio de información bayesiano BIC y medida de Calinski Harabas

6. Conclusiones

El uso de métricas especializadas como las utilizadas permiten ponderar y calcular en forma efectiva los mejores parámetros para los modelos de entrenamiento de aprendizaje automático.

Uno de los objetivos del presente trabajo fue el encontrar un método reproducible de clasificación para este tipo de análisis en el ámbito de la comunidad de astronomía y que permita continuar con futuros estudios similares en este dominio.

Dentro de los métodos no supervisados analizados, el modelo Gaussian Mixture (GMM), con 9 grupos, con el hiper-parámetro de covarianza ‘full’ es el modelo de aprendizaje automático que mayor beneficio aporta para el análisis en la detección de Cúmulos Globulares en el catálogo fotométrico, en los diagrama “Color-Color gz-K”, ya que logra un etiquetado en una mayor cantidad de grupos ofreciendo más información para el análisis de detección buscado, esta conclusión está sustentado por el análisis de resultado de las medida de Silhouette, el criterio de información de Akaike (AIC), el criterio de información bayesiano (BIC) y métricas adicionales como Davies Bouldin y Calinski Harabas.



Bibliografía

Ashman, Keith M., Christina M. Bird, and Steven E. Zepf. 1994. “Detecting Bimodality in Astronomical Datasets.” *The Astronomical Journal* 108: 2348.

D’Abrusco, R. et al. 2016. “THE EXTENDED SPATIAL DISTRIBUTION OF GLOBULAR CLUSTERS IN THE CORE OF THE FORNAX CLUSTER.” *The Astrophysical Journal* 819(2): L31.

Dékány, I. et al. 2013. “VVV SURVEY NEAR-INFRARED PHOTOMETRY OF KNOWN BULGE RR LYRAE STARS: THE DISTANCE TO THE GALACTIC CENTER AND ABSENCE OF A BARRED DISTRIBUTION OF THE METAL-POOR POPULATION.” *The Astrophysical Journal* 776(2): L19.

Durrell, Patrick R. et al. 2014. “THE NEXT GENERATION VIRGO CLUSTER SURVEY. VIII. THE SPATIAL DISTRIBUTION OF GLOBULAR CLUSTERS IN THE VIRGO CLUSTER.” *The Astrophysical Journal* 794(2): 103.

Ferrarese, Laura et al. 2012. “THE NEXT GENERATION VIRGO CLUSTER SURVEY (NGVS). I. INTRODUCTION TO THE SURVEY*.” *The Astrophysical Journal Supplement Series* 200(1): 4.

Gnedin, Oleg Y. 2010. “Modeling Formation of Globular Clusters: Beacons of Galactic Star Formation.” *Proceedings of the International Astronomical Union* 6(S270): 381–84.

Iodice, E. et al. 2016. “THE FORNAX DEEP SURVEY WITH VST. I. THE EXTENDED AND DIFFUSE STELLAR HALO OF NGC 1399 OUT TO 192 Kpc.” *The Astrophysical Journal* 820(1): 42.

Jin, Xiangyu, Meicun Hou, Zhenlin Zhu, and Zhiyuan Li. 2019. “*Chandra* Detection of Intracluster X-Ray Sources in Fornax.” *The Astrophysical Journal* 876(1): 53.

Ko, Youkyung et al. 2022a. “The Next Generation Virgo Cluster Survey. XXXIII. Stellar Population Gradients in the Virgo Cluster Core Globular Cluster System.” *The Astrophysical Journal* 931(2): 120.

———. 2022b. “The Next Generation Virgo Cluster Survey. XXXIII. Stellar Population Gradients in the Virgo Cluster Core Globular Cluster System.” *The Astrophysical Journal* 931(2): 120.

Leveque, Agostino, Mirosław Giersz, and Maurizio Paolillo. 2021. “MOCCA Survey Database: Extra Galactic Globular Clusters. I. Method and First Results.” *Monthly Notices of the Royal Astronomical Society* 501(4): 5212–28.

Lindholm, Valtteri et al. 2021. “Clustering of CODEX Clusters.” *Astronomy & Astrophysics* 646: A8.

Liu, Chengze et al. 2020. “The Next Generation Virgo Cluster Survey. XXXIV. Ultra-Compact Dwarf (UCD) Galaxies in the Virgo Cluster.” *The Astrophysical Journal Supplement Series* 250(1): 17.

Liu, Michael C., Trent J. Dupuy, and Katelyn N. Allers. 2016. “THE HAWAII INFRARED PARALLAX PROGRAM. II. YOUNG ULTRACOOL FIELD DWARFS.” *The Astrophysical Journal* 833(1): 96.

Mei, Simona et al. 2007. “The ACS Virgo Cluster Survey. XIII. SBF Distance Catalog and the Three-dimensional Structure of the Virgo Cluster.” *The Astrophysical Journal* 655(1): 144–62.

Muñoz, Roberto P. et al. 2013. “THE NEXT GENERATION VIRGO CLUSTER SURVEY-INFRARED (NGVS-IR). I. A NEW NEAR-ULTRAVIOLET, OPTICAL, AND NEAR-INFRARED GLOBULAR CLUSTER SELECTION TOOL.” *The Astrophysical Journal Supplement Series* 210(1): 4.

Muratov, Alexander L., and Oleg Y. Gnedin. 2010. “MODELING THE METALLICITY DISTRIBUTION OF GLOBULAR CLUSTERS.” *The Astrophysical Journal* 718(2): 1266–88.

Nie, Feiping, Cheng-Long Wang, and Xuelong Li. 2019. “K-Multiple-Means: A Multiple-Means Clustering Method with Specified K Clusters.” In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, Anchorage AK USA: ACM, 959–67. https://dl.acm.org/doi/10.1145/3292500.3330846 (October 23, 2022).

Peng, Eric W. et al. 2006. “The ACS Virgo Cluster Survey. XI. The Nature of Diffuse Star Clusters in Early-Type Galaxies.” *The Astrophysical Journal* 639(2): 838–57.

Strader, Jay et al. 2011. “WIDE-FIELD PRECISION KINEMATICS OF THE M87 GLOBULAR CLUSTER SYSTEM.” *The Astrophysical Journal Supplement Series* 197(2): 33.

Morissette, L., & Chartier, S. (2013). The k-means clustering technique: General considerations and implementation in Mathematica. Tutorials in Quantitative Methods for Psychology, 9(1), 15–24. https://doi.org/10.20982/tqmp.09.1.p015

Rousseeuw, P. J. (1987). Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. Journal of Computational and Applied Mathematics, 20, 53–65. https://doi.org/10.1016/0377-0427(87)90125-7

Calinski, T., & Harabasz, J. (1974). A dendrite method for cluster analysis. Communications in Statistics - Theory and Methods, 3(1), 1–27. https://doi.org/10.1080/03610927408827101

Ding, C., & He, X. (2004). K -means clustering via principal component analysis. Twenty-First International Conference on Machine Learning - ICML '04, 29. https://doi.org/10.1145/1015330.1015408

Referencias

[1] Telescopio CFHT Hawaii: https://www.cfht.hawaii.edu/

[2] AstroPy: http://research.iac.es/sieinvens/python-course/astropy.html

[3] FITS o Flexible Image Transport System: https://www.inaoep.mx/~moises/Astrofisica/fits.html

[4] GMM: https://science.nu/amne/in-depth-gaussian-mixture-models/

[5] Métodos de clustering no etiquetados: https://scikit-learn.org/stable/modules/clustering.html

[6] Bayesian Gaussian Mixture: https://scikit-learn.org/stable/modules/generated/sklearn.mixture.BayesianGaussianMixture.html