Universidad Nacional Autónoma
de México

Instituto de Ciencias Físicas

Algoritmos de clasificación aplicados a simulaciones de formación de estructura cosmológica

TESIS

que para optar por el grado de

Maestro en Ciencias
(Física)

presenta

Jazhiel Chacón Lavanderos

Director de Tesis
Dr. José Alberto Vázquez
González

Cuernavaca, Morelos, México. Abril 2021

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
PROGRAMA DE POSGRADO EN CIENCIAS FÍSICAS

Acta de Grado

No. de cuenta
519020483

En la Universidad Nacional Autónoma de México, a través de un **AULA VIRTUAL UNIVERSITARIA**, a las **12:00** horas del día **22** de **abril** del año **2021**, el alumno de nacionalidad **mexicana**

JAZHIEL CHACÓN LAVANDEROS

cuya fotografía aparece al margen, se presentó con el fin de sustentar el examen para obtener el grado de

MAESTRO EN CIENCIAS (FÍSICA)

en su modalidad de graduación por **TESIS**, con el trabajo titulado: **Algoritmos de clasificación aplicados a simulaciones de formación de estructura cosmológica**, del cual fue tutor principal el **DR. JOSÉ ALBERTO VÁZQUEZ GONZÁLEZ**.

El alumno cursó sus estudios en el período **2019-2** a **2021-1**, obtuvo un promedio de **9.90** y cumplió con los requisitos académicos señalados en el plan de estudios **4204** aprobado por el H. Consejo Universitario.

El jurado presente asentó la calificación de

APROBADO CON MENCION HONORIFICA
(APROBADO, SUSPENDIDO, APROBADO CON MENCIÓN HONORÍFICA, NO APROBADO)

le hizo saber el resultado de su examen y le tomó la Protesta Universitaria
El jurado estuvo integrado por:

Nombre	Firma
DR. JUAN CARLOS DEGOLLADO DAZA Presidente	
DR. SEBASTIEN MICKAEL MARC FROMENTEAU Vocal	
DR. OCTAVIO VALENZUELA TIJERINO Vocal	
DR. JOSÉ ALBERTO VÁZQUEZ GONZÁLEZ Secretario	

"POR MI RAZA HABLARÁ EL ESPÍRITU"

El suscrito Coordinador del Programa constató que las firmas corresponden al jurado designado

Doctor Alberto Güijosa Hidalgo

No. Folio del Programa
4204092012123

Nombre y firma del Coordinador del Programa

Miembros del Comité Tutoral:
Dr. José Alberto Vázquez González
Instituto de Ciencias Físicas, UNAM
Dr. Juan Carlos Hidalgo Cuéllar
Instituto de Ciencias Físicas, UNAM
Dr. Roberto Allan Sussman Livovsky
Instituto de Ciencias Nucleares, UNAM

Miembros del Jurado:
Dr. José Alberto Vázquez González
Instituto de Ciencias Físicas, UNAM
Dr. Sébastien Mickael Marc Fromenteau
Instituto de Ciencias Físicas, UNAM
Dra. Celia del Carmen Escamilla Rivera
Instituto de Ciencias Nucleares, UNAM
Dr. Octavio Valenzuela Tijerino
Instituto de Astronomía, UNAM
Dr. Juan Carlos Degollado Daza
Instituto de Ciencias Físicas, UNAM

Dedicado a quienes piensan que hago astronomía.
No, claro que no.

Agradecimientos

Primero, le debo un gran reconocimiento a mi asesor, el Dr. José Alberto Vázquez, quien ha brindado su apoyo desde hace ya más de 3 años, mismos que han sido de una estrecha relación profesional y personal. Gracias por recibirme en tu oficina en el CINVESTAV y escuchar mis ideas, y ahora siendo parte del Instituto de Ciencias Físicas de la UNAM, brindarme un gran apoyo para participar en cantidad de eventos, mismos que han forjado este trabajo, también en momentos difíciles mostraste tu preocupación y ayuda. Ser tu estudiante y colaborador ha sido un privilegio.

Gracias también a los compañeros que conocí a lo largo de mis estudios de maestría, sobre todo al grupo de Luises (Luis A., Luis O., Luis P.) y en especial a todos los que conforman el grupo de colaboración y asesoría del Dr. Vázquez, el compañerismo del grupo siempre ha mostrado ser su gran fortaleza, gracias también por los (repentinos) viajes a lugares eh horarios nocturnos, "para recolectar ideas". Debo agradecer también al Dr. Erick Almaraz, ya que sin su ayuda y gran capacidad, este trabajo no hubiera sido posible. Gracias por resolver tantas dudas y mostrar tu interés en el desarrollo del tema.

Mi reconocimiento a todo el personal académico, técnico y administrativo del Instituto de Ciencias Físicas, desde los cursos de inducción al programa de maestría hasta el final de la misma, el apoyo para el proceso de registro y admisión al programa y por permitirme ser parte del Instituto. En especial agradezco a los miembros del grupo de gravitación y cosmología: Al Dr. Juan Carlos Degollado, por el apoyo e interés brindado desde el inicio y gran sentido del humor, al Dr. Juan Carlos Hidalgo, miembro del comité tutoral y responsable del proyecto CONACYT 282569, con duración de febrero a julio de 2019. Al Dr. Sébastien Fromentau y sus comentarios hacia mi trabajo de Licenciatura así como el curso de métodos numéricos, acompañado de la Dra. Mariana Vargas del Instituto de Física de la UNAM y que, sin dudarlo, ha sido uno de los mejores cursos que haya tenido la oportunidad de tomar. Agradezco también al posgrado de la UNAM por el otorgamiento de la beca de maestría por parte de CONACYT (000306 - MAESTRÍA

EN CIENCIAS FÍSICA) en el transcurso de agosto 2019 a enero 2021 y JAV agradece a FOSEC SEP-CONACYT Investigación Básica A1-S-21925 y UNAM-DGAPA-PAPIIT IA102219.

Mi agradecimiento hacia el Dr. Tonatiuh Matos del CINVESTAV, que a pesar de no haber tomado participación directa del proyecto, sin duda fue y seguirá siendo un gran motivador para alcanzar las metas que me proponga, además de permitirme usar el clúster computacional "EKBEK", al cuál le tengo mucho cariño, ya que en mi servicio social me encargué de ponerlo en funcionamiento nuevamente y fue responsable de la mayoría de resultados de las simulaciones presentadas en este trabajo. Le debo también un especial reconocimiento y mi completa gratitud a Malú, quien desde mi entrada a su cubículo en el CINVESTAV, mostró una gran alegría y, que al día de hoy y como siempre lo digo, me ha dado "zapes" emocionales para continuar mi camino. Gracias infinitas, Malú, por preocuparte tanto por mí y por mis seres queridos. A mis amigos de toda una vida, más de 10 años y aunque la distancia nos separa a muchos, saben que los llevo en mi corazón.

En especial, se dice que los matemáticos deben encontrar la variable dependiente Y, yo encontré a mi Y (doble Y, de hecho). Agradezco que el Universo nos haya puesto en un camino, aunque con altibajos, que pueda pavimentarse de los logros de ambos.

Finalmente, le debo todo a mi familia, absolutamente todo mi amor, mi gratitud, mi respeto, mi admiración y mi orgullo. A mi madre, que superó una de las etapas más difíciles de su vida como si no hubiese tenido un rasguño, sabes lo mucho que te quiero y la gran inspiración que eres para mí. A mi padre, que pasó por momentos tan fuertes y complicados y que es un gran ejemplo de vida, que expresó su apoyo para completar mis estudios. Mi admiración es tan grande que no puedo hacer más que demostrarte el aprecio de la mejor manera que he podido hasta ahora, espero devolverte todo ese apoyo algún día. A mi hermana, quien ha crecido tan rápido que en el lapso de dos trabajos similares, ahora se encuentra a punto de terminar sus estudios universitarios, pareciera que fuese ayer cuando recién entraba a la universidad. Sigue adelante, que el camino es largo y gratificante sin duda. Los quiero y los amo con toda mi alma. Este trabajo es por y para ustedes.

Resumen

El presente trabajo tiene como finalidad explorar la cosmología numérica, su estrecha relación con la cosmología observacional y su potencial al correlacionarlo con la rama de la inteligencia artificial: Machine Learning.

La formación de estructura cosmológica ha sido objeto de estudio desde antes del descubrimiento de la expansión del Universo. El proceso de evolución conlleva a proponer la existencia de elementos hasta ahora desconocidos, la materia oscura y la energía oscura, ambos descritos en el modelo cosmológico estándar: Lambda Cold Dark Matter (ΛCDM).

La distribución de materia en el Universo es observable mediante el espectro de potencias de masa, el cual se obtiene mediante ajustes a observaciones de diferentes sondeos galácticos, cuyo comportamiento es lineal en grandes escalas, y no-lineal en escalas más pequeñas. Es en el régimen no-lineal donde las simulaciones numéricas de N-cuerpos juegan un papel importante, ya que su uso ha sido necesario para ajustar el espectro de potencias.

El modelo de colapso esférico es la piedra angular teórica para explicar la abundancia de halos de materia oscura en simulaciones y su perfil de densidad obtenido en observaciones. Las propiedades de este modelo teórico se ven reflejadas también en las condiciones iniciales de una simulación. Por tanto, existe una forma de enlazar las propiedades del campo de densidad inicial de materia oscura con la formación de estructura final en halos de materia oscura.

El método descrito utiliza la habilidad de algoritmos de machine learning para aprender la relación entre las propiedades no-lineales del campo de densidad y eventualmente describir qué aspectos del campo son importantes para la formación de halos de materia oscura.

El capítulo 1 está dedicado a una breve introducción a la cosmología, a los procesos de evolución, el modelo estándar cosmológico (ΛCDM) y los diferentes métodos teóricos para explicar la formación de estructura observada en el Universo.

En el capítulo 2 se destacan los diferentes métodos de simulación numérica usados en cosmología, así como detalles técnicos del código GADGET y una

descripción del análisis de la simulación usada en el trabajo.

El capítulo 3 explora Machine Learning, una rama de la inteligencia artificial. Se describen los diferentes algoritmos (supervisados y no supervisados) y se explican los métodos de clasificación y las métricas utilizadas para evaluarlos.

Posteriormente, el capítulo 4 utiliza los métodos descritos en los capítulos 2 y 3, donde se describe a la cosmología numérica como un proyecto de clasificación binaria. Los algoritmos son entrenados con propiedades extraídas de las condiciones iniciales de un campo de densidad de materia oscura en una simulación de N-cuerpos, estas propiedades funcionan como variables independientes para los algoritmos de clasificación. La variable dependiente es el resultado final de la simulación, ya que, al seguir la trayectoria de las partículas que al final forman halos de materia oscura en un rango de masas de $10^{11} - 10^{14}$ $M_{\odot}$, el método descrito es capaz de clasificar si una partícula de las condiciones iniciales terminará en un halo de materia oscura de un cierto valor umbral.

Finalmente, el capítulo 5 explica que este es solo un acercamiento y uso de algoritmos de clasificación para la cosmología numérica, además de que se buscan nuevos métodos para no solo clasificar un solo tipo de simulación, si no extenderlo a otros modelos que utilicen diferentes características para explicar la evolución del Universo, y que son alternativas al modelo ΛCDM.

Abstract

This thesis has the objective of exploring numerical cosmology, the relation with observations and the potential when the use of Machine Learning is inside the equation.

Cosmological structure formation has been the subject of study even before the discovery of the expansion of the Universe.The evolution process leads to propose the existance of elements that upon until now, remain unknown, dark matter and dark energy, both described in the cosmological standard model: Λ Cold Dark Matter (ΛCDM).

Matter distribution in the Universe is described via the matter power spectrum, which is obtained using different galactic surveys and with a linear behaviour in larger scales, whereas at smaller scales, the behaviour of the power spectrum is non-linear. It is in the non-linear regime where numerical N-body simulations have an important part, since its use has been necessary to make adjustments to the power spectrum.

The spherical collapse model is the theoretical cornerstone to explain the number and abundance of dark matter haloes in simulations and their observed density profile. Properties of this theoretical model are also described inside a simulation initial conditions. Therefore, there is a way to link the initial dark matter density field properties with the final cosmological structure formation and dark matter haloes.

The method uses the ability of machine learning algorithms to learn the relationship between the non-linear properties of the density field and eventually describe what aspects of the field are important for the formation of dark matter haloes.

Chapter 1 gives a brief introduction to cosmology, the evolution proceses, the standard cosmological model (ΛCDM) and the diverse theoretical approaches that explain cosmological structure formation in the Universe.

In chapter 2, distinct numerical simulation methods used in cosmology are described, as well as some technical details about GADGET code and a simulation used for this thesis.

Chapter 3 explores Machine Learning, a sufbield of artificial intelligence.

Supervised and unsupervised algorithms are described as well as an explanation of classification metrics for evaluation.

Later, in chapter 4, the methods described in chapters 2 and 3 are used in a numerical cosmology framework performing as a binary classification problem. The algorithms are trained with characterisitics extracted from the initial conditions of a dark matter density field in an N-body simulation. These properties have the functionality of being the independent variables of the algorithm, and the dependent variable is the final result of the simulation run, since, by following the trajectory of the particles which eventually form dark matter haloes in a mass range of $10^{11} - 10^{14}$ $M_{\odot}$, the method is capable of classifying wether a particle would end up in a halo of a certain threshold, given the properties of the initial conditions.

Finally, chapter 5 tells that this is only a first approach for classification algorithms in numerical cosmology, furthermore, it explains that new methods are in the hunt not only for classification tasks, but an extension for different cosmological evolution models, which use a variety of characteristics and act as an alternative to the ΛCDM model.

Índice general

Índice de figuras

Índice de tablas

Capítulo 1

Introducción a la Cosmología

La cosmología es la ciencia que estudia al Universo como un todo, busca explicar su origen, su evolución y la base fundamental que ha llevado a desarrollar las leyes de la física que gobiernan a todo el Universo. Esta suposición se ha hecho a partir del principio cosmológico: **El Universo es esencialmente homogéneo e isótropo en un espacio tridimensional.** Al tomar en cuenta una hipótesis de ergodicidad, el Universo se considera homogéneo e isótropo a gran escala. Esta escala debe estar en un rango mayor que 150 Megapársecs (Mpc). La homogeneidad se refiere a que la materia está uniformemente distribuida en el espacio que es posible observar. La isotropía hace referencia a que las propiedades del Universo son las mismas en cualquier dirección a la que se observe. En un Universo homogéneo e isótropo, la distribución de materia debe ser la misma para cualquier observador ubicado en cualquier punto espacial. No existen observadores preferenciales.

El principio cosmológico puede suponer la existencia de un Universo infinito, pero la teoría de la relatividad dicta que el espacio puede ser curvo, y que por tanto, puede ser que el Universo sea finito, pero que no tenga bordes. Este tipo de Universo puede captarse si se imagina un espacio bidimensional formando una superficie esférica, sobre la cual un observador bidimensional puede recorrerla y nunca encontrar un borde. Al extrapolar esta imagen a un espacio tridimensional permite comprender un Universo finito sin bordes. Aunque este concepto puede llevar a un problema de coalescencia, ya que la gravedad, pasado cierto tiempo, haría que todo objeto o cuerpo dentro del Universo se atrajera, juntando toda la materia hasta un solo punto. La manera de evitar la coalescencia es mediante la misma relatividad, ya que todo movimiento es siempre en referencia a algo, dado que no existe el movimiento absoluto. Evidentemente, un Universo en expansión evita la coalescencia.

De esta manera, las galaxias no se acercarían unas a otras, y el espacio en expansión pudo haber sido provocado por una explosión. La autogravitación

puede entonces detener la expansión y en ciertos casos, revertirla, dependiendo del contenido de materia y de la violenta explosión inicial. Este concepto es lo que se conoce como el Big Bang.

El principio cosmológico fue utilizado por conveniencia matemática. Si un observador observa un espacio isótropo, otro observador que estuviese en movimiento relativo a gran velocidad respecto al primero seguramente ya no observaría un Universo isótropo. El segundo observador vería un tipo de desplazamiento Doppler de los objetos y galaxias a su alrededor hacia el espectro del azul en la dirección de su movimiento, mientras que las que se desplazan lejos de él se desplazan al espectro rojo. Para este segundo observador, el principio cosmológico no se cumple. La teoria de la relatividad general permite resolver el problema. Esta teoria establece que en todo punto del espacio, en un sistema cualquiera existe un observador que ve su microentorno plano, en el sentido de que para este no hay gravedad. Uno de los experimentos mentales de Einstein estipula que un observador dentro de un ascensor que está sometido a un efecto de caída libre no nota ningún efecto gravitatorio. Para este tipo de observadores se cumple el principio cosmológico, y son conocidos como observadores fundamentales. A su vez, estos deben tener un tiempo común entre ellos, conocido como tiempo cósmico. Todos los observadores fundamentales ven lo mismo cuando utilizan ese tiempo cósmico.

1.1. Expansión del Universo

En la década de 1920 Edwin Hubble realizó una serie de observaciones y mediciones de distancias entre galaxias a partir del corrimiento al rojo relativo entre ellas, descubrió que cuanto más alejada esté una galaxia, mayor es su desplazamiento al rojo, probando así que el Universo se encuentra en expansión. La Ley de Hubble establece que la velocidad aparente $\mathbf{v}$ de alejamiento entre galaxias es proporcional a su distancia $\mathbf{r}$ (Hubble, E., 1929 [1])

$$\mathbf{v(t)} = H(t)\mathbf{r(t)}, \tag{1.1}$$

donde $\mathbf{r}(t)$ es la distancia propia, es decir, la distancia medida entre una galaxia y un observador en un tiempo t. La velocidad de recesión $\mathbf{v}$ es la tasa a la que la distancia $\mathbf{r}$ aumenta y $H(t)$ el parámetro de Hubble. Al día de hoy (t_0) se conoce que el parámetro ronda los valores $H(t_0) = H_0 = 67 - 70$ kms^{-1}Mpc^{-1} es la constante de Hubble. En cosmología es de gran utilidad la deficinión de las coordenadas comóviles. En este sistema, una partícula de prueba se mueve junto con el mismo, de manera que una partícula dentro de un sistema de referencia comóvil siempre estará en reposo. Las coordenadas

comóviles se expanden junto con el Universo. Si se utiliza la Vía Láctea con una coordenada comóvil $\mathbf{x} = 0$ y una galaxia con coordenada comóvil $\mathbf{x}$, entonces la distancia propia hasta ella será

$$\mathbf{r} = a(t)\mathbf{x}, \tag{1.2}$$

donde $a(t)$ se define como el factor de escala. De esta manera, la velocidad $\mathbf{v}$ obtiene una nueva forma

$$\mathbf{v} = \dot{\mathbf{r}} = \frac{d}{dt}(a(t)\mathbf{x}), \tag{1.3}$$

notando que

$$\frac{d}{dt}(a\mathbf{x}) = Ha\mathbf{x}. \tag{1.4}$$

Por definición, la coordenada comóvil $\mathbf{x}$ no depende del tiempo, el cambio el término $(a\mathbf{x})$ se debe al factor de escala a y el parámetro de Hubble H puede entonces escribirse como

$$H = \frac{\dot{a}(t)}{a(t)}. \tag{1.5}$$

Por otro lado, la luz proveniente de galaxias y estrellas distantes no es monocromática, sino que tiene caracteríısticas espectrales propias de los átomos de los gases alrededor de las estrellas. Cuando se examinan las líneas de emision, se encuentra que están desplazadas hacia el extremo rojo del espectro. Este cambio se debe a la misma expansión del Universo y puede entenderse como un tipo de efecto Doppler, ya que indica que las galaxias se están alejando del observador. De esta deducción se define el corrimiento al rojo o redshift z como

$$z \equiv \frac{\lambda - \lambda_0}{\lambda_0}, \tag{1.6}$$

siendo λ_0 la longitud de onda emitida por una galaxia o estrella u objeto luminoso y λ la longitud de la línea espectral medida por un observador. Las velocidades se infieren con una fórmula aproximada tal que

$$z \approx \frac{v}{c}, \tag{1.7}$$

con v el valor absoluto de la velocidad relativa entre la galaxia y el observador y c la velocidad de la luz.

1.2. Relatividad General

La piedra angular de la cosmología moderna es la teoría general de la relatividad. La teoría Newtoniana argumentaba que la gravedad era una fuerza externa ejercida a un objeto, en relatividad general la gravedad es una propiedad geométrica del espacio-tiempo. La relatividad general está compuesta por 10 ecuaciones diferenciales parciales no lineales y acopladas, que rigen la física de cualquier sistema gravitacional. Para trabajar en el marco de la relatividad general, se inicia introduciendo una variedad 4-dimensional dotada de una métrica de la forma

$$ds^2 = g_{\alpha\beta} dx^\alpha dx^\beta, \tag{1.8}$$

donde $\alpha, \beta = 0, 1, 2, 3$ y ds es el elemento de línea de este espacio o variedad. Luego de una serie de cálculos que involucran a los símbolos de Christoffel $\Gamma^\alpha_{\mu\nu}$, de donde se deduce el tensor de Riemman $R^\alpha_{\beta\mu\nu}$, el tensor de Ricci $R_{\mu\nu}$ y el escalar de curvatura R, se llega a las ecuaciones de Einstein (Schutz, B., 2009 [2])

$$G_{\alpha\beta} = R_{\alpha\beta} - \frac{1}{2} R g_{\alpha\beta} + \Lambda g_{\alpha\beta} = \frac{8\pi G}{c^4} T_{\alpha\beta}. \tag{1.9}$$

donde $T_{\alpha\beta}$ es el tensor de energía-momento, G es la constante de gravitación universal de Newton y Λ la constante cosmológica. Ignorando este útimo término, las ecuaciones de Einstein dictan que la geometría del Universo está determinada por su contenido de materia y energía. La constante cosmológica, introducida originalmente por Einstein permite un Universo estático y estable ante un colapso gravitacional. Luego del descubrimiento de Hubble, Einstein dejó de lado esta idea, llamándola el mayor error de su vida. Actualmente, la constante cosmológica se utiliza para representar a la *energía del vacío* del espacio-tiempo, también llamada energía oscura.

1.3. Las ecuaciones de movimiento del Universo

Asumiendo las condiciones de homogeneidad e isotropía del Universo, las ecuaciones de movimiento pueden deducirse directamente de las ecuaciones de Einstein para un Universo en expansión, la métrica que se utiliza para resolver este conjunto de ecuaciones es la métrica de Friedmann-Lemaître-Robertson-Walker (FLRW), que tiene la forma

$$ds^2 = -c^2dt^2 + a^2(t)\left[\frac{dr^2}{1-kr^2} + r^2d\Omega^2\right], \tag{1.10}$$

donde

$$d\Omega^2 = d\theta^2 \sin^2\theta d\phi^2. \tag{1.11}$$

El término k es la curvatura constante espacial. Si $k < 0$ se obtiene un espacio con curvatura abierta, si $k = 0$ la curvatura es plana, y si $k > 0$ la curvatura es cerrada. En un sentido cosmológico, las ecuaciones de Einstein pueden usarse para encontrar una relación entre el factor de escala $a(t)$, la curvatura k, y la cantidad de densidad de materia $\rho(t)$ del Universo. Recordando la ley de Hubble (1.1) las ecuaciones de Friedmann se escriben de la siguiente manera

$$H^2(t) = \frac{8\pi G}{3}\rho(t) - \frac{kc^2}{a^2(t)}. \tag{1.12}$$

La suposición de un Universo homogéneo e isótropo requiere que el tensor de energía-momento en las ecuaciones de campo de Einstein también sea homogéneo e isótropo. El tensor de un fluido perfecto cumple con este requisito, y viene dado por la siguiente expresión general (Schutz, B., 2009 [3]; Grøn, Ø. Hervik, S., 2004 [4])

$$T_{\alpha\beta} = (\rho + p)u_\alpha u_\beta + pg_{\alpha\beta}, \tag{1.13}$$

donde ρ es la densidad de energía del fluido, u_γ su cuadrivelocidad, y p la presión ejercida por el fluído. La homogeneidad implica que la presión y la densidad deben ser independientes de la posición y sólo dependen del tiempo. Para la métrica descrita en la ecuación (1.10), este tensor es diagonal

$$T_{\alpha\beta} = \text{diag}(-\rho, p, p, p), \tag{1.14}$$

que por definición, se caracteriza por su densidad de masa ρ y su presión isotrópica p, relacionadas mediante su ecuación de estado $p = w\rho$. Insertando la ecuación (1.14) en las ecuaciones de campo de Einstein (1.9) se obtienen las siguientes ecuaciones

$$\frac{\dot{a}^2 + kc^2}{a^2} = \frac{8\pi G\rho}{3}, \tag{1.15}$$

$$\frac{\ddot{a}}{a} = -\frac{4\pi G}{3}\left(\rho + \frac{3p}{c^2}\right) + \frac{\Lambda c^2}{3}, \tag{1.16}$$

las cuales describen la tasa de expansión y la aceleración del Universo como una función de su densidad, presión y curvatura espacial.

En resumen, las ecuaciones de Friedmann describen la cantidad de materia contenida en el Universo y determinan su geometría. En secciones posteriores se describirá el modelo estándar de la cosmología: ΛCDM y su impacto en el estudio moderno de la cosmología.

1.4. Evolución cosmológica

El primer intento de aplicar la relatividad al Universo se debió al propio Einstein. Aunque con este acercamiento se llegaba a una solución de un Universo inestable, ya sea en expansión o en colapso. Para llegar a un Universo estático, Einstein admitió el uso de la constante cosmológica, lo que hoy se interpretaría como una forma de energía oscura. Este término le da al Universo una facultad expansiva que contrarresta la autogravitación debida a la densidad de masa, evitando la coalescencia y obteniendo una entorno estático. Lo más novedoso de esta deducción es que, al contrario de la gravitación Newtoniana generada por la masa, la fuerza expansiva del término cosmológico es generada por el vacío.

Junto con el modelo de Universo estático de Einstein, exisitieron también otros modelos que intentaron describir el entorno, ejemplo de esto es el modelo FLRW, que es una solución exacta de las ecuaciones de campo de Einstein de la relatividad general (ecuación 1.10). Sin embargo, distintos datos observacionales de galaxias distantes mostraban que existía, (o existe) un tipo de materia "faltante" en el Universo. En los últimos años de 1930, se estudió la dinámica del cúmulo de galaxias Coma Berenice. En dicho trabajo se proporciona evidencia sobre la masa luminosa en el cúmulo, mencionando finalmente que cantidad necesaria para mantener a las galaxias unidas gravitacionalmente no coincide con lo observado (Zwicky, F., 1937 [5]). Como consecuencia, se llega a la conclusión de que debe existir más materia bariónica que no fuese visible, la cual compensara la cantidad de materia faltante necesaria para mantener este cúmulo de galaxias unido. Este fué uno de los primeros indicios de la hipotética existencia de la ***materia oscura*** sin llegar a tener una fuerte evidencia de ella.

Para la década de 1970, el tema de materia oscura resurgió, se mencionó que la estabilidad gravitacional de las galaxias es debido a una cantidad de masa mayor a la observada (Rubin, V., Ford, W. K., 1970 [6]). En particular, se menciona que las curvas de rotación de galaxias espirales, que se encargan de medir la velocidad radial de las estrellas en función de su distancia hacia el centro de las galaxias no obedecen un comportamiento del tipo Kepleriano,

es decir, que la velocidad de las estrellas disminuye al aumentar la distancia desde el centro de las galaxias, como se describe en la siguiente ecuación

$$\mathbf{v}(r) = \sqrt{\frac{GM(r)}{r}}, \tag{1.17}$$

donde G es la constante de gravitación universal de Newton, r la posición radial (en valor absoluto) de la estrella y $M(r)$ la cantidad de masa observada dentro del radio r. Se esperaría que la velocidad decaiga rápidamente al aumentar la distancia, sin embargo se observa que en muchos casos, la velocidad no disminuye e incluso, en ocasiones, llegaba a aumentar. Suponiendo que la teoría de Newton es correcta, entonces la única forma de explicar este fenómeno es mediante el término $M(r)$. Este gran descubrimiento ha impactado hasta épocas recientes y sigue siendo materia de discusión.

1.4.1. Radiación del fondo cósmico de microondas

Por sus siglas en inglés, *Cosmic Microwave Background Radiation*, de aquí en adelante, CMB, es un tipo de radiación que data de alrededor de 380,000 años después del comienzo del Universo. Antes de esta época, el Universo era tan caliente y denso que era opaco para todo tipo de radiación. Ni siquiera los átomos simples podían formarse sin ser instantáneamente desgarrados en sus protones y electrones constituyentes por la radiación intensa. Los fotones del CMB se emitieron cuando el Universo tenía una temperatura de aproximadamente 3.000 Kelvin. Por la propia expansión han sido desplazados hacia el rojo a longitudes de onda más largas, razón por la cual se detectan en la región de microondas del espectro electromagnético a una temperatura promedio de 2.725 K. La uniformidad de la imagen del CMB apoya las hipótesis del principio cosmológico. El patrón impreso en el CMB (figura 1.1) indica dos fuerzas actuando sobre la materia: la gravedad, causando que la materia colapse, y la presión de radiación que previene el colapso gravitacional. Esta rivalidad causa que los fotones y la materia oscilen en regiones densas, formando patrones que se modifican debido a la cantidad de materia oscura presente en esa época. Es decir, la existencia de materia oscura deja una marca característica en el espectro del CMB, ya que se agrupa en regiones densas y contribuye al colapso gravitacional de la materia, pero no se ve afectada por la presión de radiación. El espectro del CMB muestra la fuerza de estas oscilaciones a diferentes escalas. Por ejemplo, la sonda WMAP (Spergel, D. N. et al., 2007 [7]) o más reciente, las medidas del satélite Planck (Ade, P. et al 2014 [8]) han medido con bastante exactitud el espectro de potencias del CMB y favorecen la existencia de materia oscura.

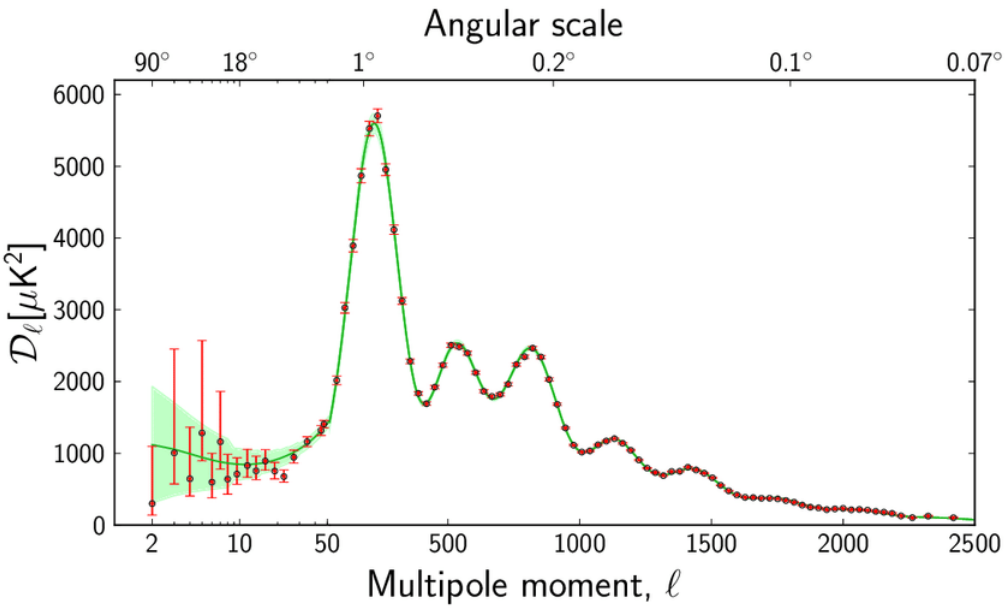


Figura 1.1: Espectro de potencias del CMB. Los puntos rojos con barras de error son puntos de datos recabados observacionalmente por Planck y la línea verde es un ajuste teórico del modelo. La región sombreada es un error teórico debido a la varianza cósmica, una incertidumbre presente en las mediciones. Imagen tomada de Ade, P. 2014 y Durrer, R. 2015.

La materia oscura también se ha visto altamente favorecida cuando se estudia la formación estructura a gran escala. Las oscilaciones del CMB evolucionan en estructuras más grandes, dada la cantidad de tiempo disponible para que se efectuara el colapso gravitacional, se formaron oscilaciones, llamadas Oscilaciones Acústicas de Bariones (BAO por sus siglas en inglés). En la época del CMB, la materia oscura no estuvo bajo las mismas oscilaciones que con la materia y la radiación, si no que fue libre de colapsar por su propia cuenta, creando regiones densas que fortalecieron la formación de estructura. Este mecanismo permitió que la distribución de galaxias y cúmulos de galaxias se formara como se observa hoy en día (Blanton, M. R. et al., 2017 [9]).

1.5. Lambda Cold Dark Matter (ΛCDM)

Como se mencionó anteriormente, la existencia de la materia oscura está basada en resultados más que nada observacionales, respaldados por los registros del cúmulo de Coma y el espectro de potencias del CMB. Fue hasta la década de 1980 cuando la materia oscura se introduce en las ecuaciones de

movimiento de Friedmann (1.12). La introducción de la materia oscura así como la constante cosmológica Λ componen el modelo de Lambda Cold Dark Matter. Este modelo es una parametrización de las ecuaciones de movimiento y muchas veces referido como el modelo estándar de la cosmología, pues su descripción envuelve gran cantidad de hechos teóricos y observacionales que a continuación se describen:

- Un marco teórico con base en la Teoría General de la Relatividad. Gracias a esto se obtiene una teoría de campo para la gravedad a escalas cosmológicas.
- El principio cosmológico: El Universo es en esencia homogéneo e isótropo a grandes escalas (Ryden, B., 2003 [10]).
- El modelo de fluido perfecto, las galaxias y estructuras observables en el Universo se obtienen dentro de la relatividad general junto con la ecuación de continuidad (Schutz, B., 2009 [2]).
- La Ley de Hubble. La expasión del Universo establece que la velocidad de recesión de las galaxias es proporcional a su velocidad (Hubble, E., 1929 [1]).
- La radiación del CMB. Las mediciones que obtienen el espectro de potencias del CMB evidencia el principio cosmológico a grandes escalas (Planck Collaboration, 2014 [8]).
- La determinación de la abundancia relativa de elementos primordiales como ^{1}H, ^{2}D, ^{3}He, ^{4}He y ^{7}Li, provenientes de reacciones nucleares durante la época de Big Bang Nucleosíntesis (BBN) (Peebles, J. 1993 [11]).
- El análisis de estructura a gran escala del Universo, proveniente de observaciones de distintos satélites y surveys como Planck y SDSS (Blanton, M. R., Bershady, M. A., Abolfathi, B., Albareti, F. D. and et al. 2017 [9]), apoya a la determinación de parámetros del modelo cosmológico estándar.

Además, se agregan otras características que permiten explicar la evolución del Universo

- La evolución de las perturbaciones de la materia provenientes de fluctuaciones cuánticas de la densidad. Explican la estructura a gran escala (Sasaki, M. 1986 [12]).

- La Inflación cósmica, postula una expansión extremadamente acelerada en épocas muy tempranas y explica la homogeneidad y planitud observada en el Universo (Guth, A. 1981 [13]).
- La constante cosmológica Λ, originada de las ecuaciones de Einstein para forzar un Universo estático. Debido a que se sabe que el Universo se está expandiendo de forma acelerada, se refiere a Λ como la energía del vacío o ***energía oscura*** (Peebles, J. and Ratra, B. 2003 [14]).
- Materia Oscura Fría (CDM). Un tipo de materia que interactúa solo de manera gravitacional con los bariones. No tiene ningún tipo de interacción con la radiación (oscura) y no se mueve a velocidades relativistas (Fría).

Todo lo listado anteriormente conforma lo que se conoce como el modelo estándar de la cosmología, ΛCDM. Durante los últimos 30 años ha resistido a cualquier cantidad de pruebas, pero aún está lejos de ser el modelo definitivo de evolución cosmológica.

1.6. Formación de estructura

Las galaxias y estructuras complejas se forman de inestabilidades gravitacionales, en un campo de densidad existen regiones sobredensas que aglomeran masa al pasar el tiempo para después producir la estructura que se observa el día de hoy. Este crecimiento es jerárquico, es decir, las estructuras más pequeñas se forman primero para después combinarse en estructuras más grandes. Las inestabilidades son principalmente generadas por perturbaciones al campo de densidad de materia oscura. Después de la evolución se llega a un momento llamado dominación de materia, las perturbaciones del campo de densidad tienen impacto sobre el potencial gravitacional y son las que empiezan a generar el crecimiento en la estructura. Normalmente se define el contraste de densidad δ como

$$\delta = \frac{\rho - \bar{\rho}}{\bar{\rho}}, \tag{1.18}$$

donde $\bar{\rho}$ es la densidad media del Universo a un redshift z dado. Aunque la evolución de estructura es predominantemente no lineal, las ecuaciones que desriben la evolución se pueden linearizar si las fluctuaciones de densidad son pequeñas. Esto último es cierto, ya que las anisitropías del CMB fueron pequeñas en un principio, de manera que la teoría lineal respalda la formación de estructura.

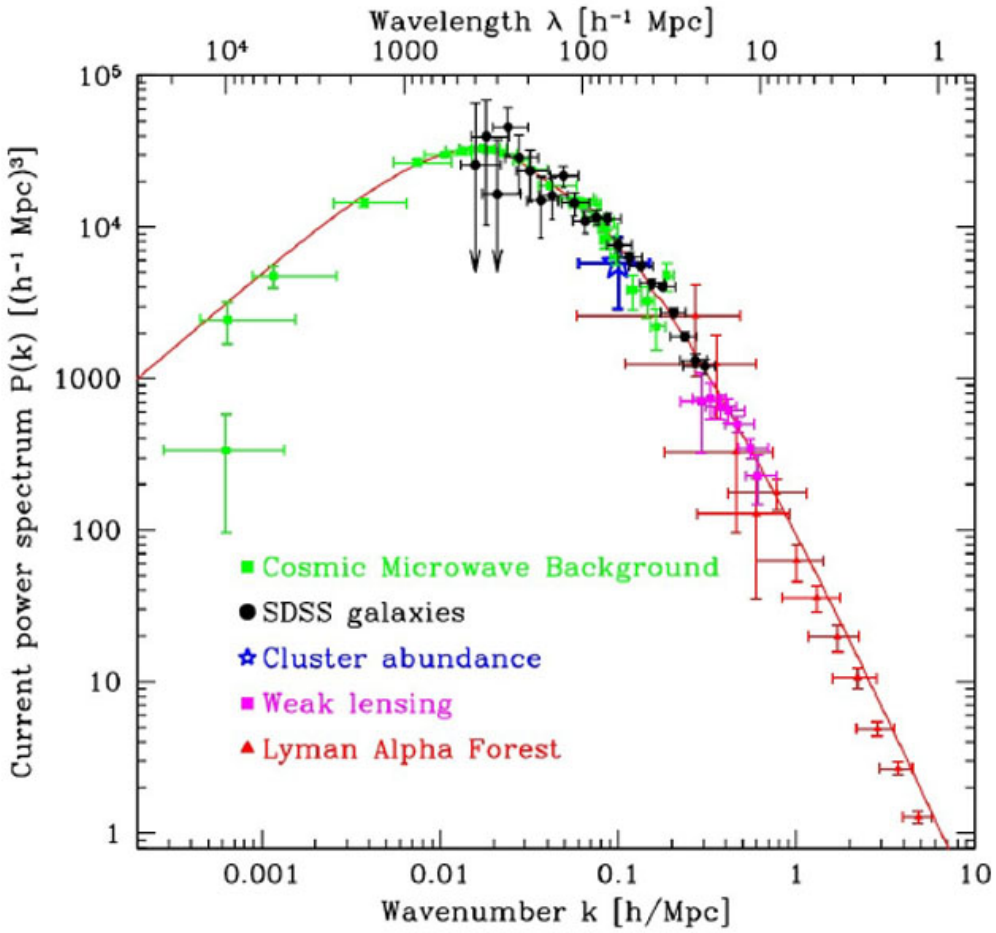


Figura 1.2: Espectro de potencias de masa $P(k)$ en la época $z = 0$. Se compilan diversas medidas de evolución de estructura, ajustadas con el modelo ΛCDM en la línea sólida. Imagen tomada de Tegmark, M. et al., 2004.

1.6.1. Régimen lineal

La estructura cósmica es el resultado de la amplificación de las fluctuaciones a la densidad primordial debido a inestabilidades gravitacionales. La mayor cantidad de información de las fluctuaciones de densidad se obtiene de medidas estadísticas, lo que se conoce como el *espectro de potencias de masa*. Este espectro es una observable cosmológica, dado que se obtiene de sondeos de galaxias y puede compararse con marcos teóricos.

El espectro de potencias de masa se ha medido con gran precisión durante los últimos años, se define por

$$P(k) \propto T(k)^2 P_0(k), \tag{1.19}$$

donde $T(k)$ es una función de transferencia la cual contiene la dependencia temporal y la dependencia en el espacio de Fourier de la evolución de las pertubaciones a la densidad, luego de un tiempo de evolución y $P_0(k)$ es el espectro primordial de fluctuaciones de materia (Dodelson, S., 2003 [15]). La figura 1.2 es una muestra de la evolución del espectro de potencias de masa en el régimen lineal ajustado con medidas de las anisotropías del CMB, estructura galáctica a gran escala, weak lensing y Lyman Alpha forest (Tegmark, M., Strauss, M. A., Blanton, M. R., and et al., 2004 [16]). Se observa que existe una línea sólida ajustada a los datos, la cual corresponde al modelo ΛCDM. A continuación se describe brevemente la obtención de $P(k)$ (Norman, M. L., 2010 [17]).

En cualquier época, z por ejemplo, la densidad de materia del Universo se expresa en términos de la densidad media y una sobredensidad local (fluctuación)

$$\rho(\mathbf{x}) = \bar{\rho}(1 + \delta(\mathbf{x})). \tag{1.20}$$

El contraste de densidad $\delta(\mathbf{x})$ se expande en modos de Fourier

$$\delta(\mathbf{x}) = \frac{\rho(\mathbf{x}) - \bar{\rho}}{\bar{\rho}} = \int \delta(\mathbf{k}) \exp\left(-i\mathbf{k}\cdot\mathbf{x}\right) d^3k, \tag{1.21}$$

mientras que la función de autocorrelación de $\delta(\mathbf{x})$ define el espectro de potencias mediante las relaciones

$$\langle \delta(\mathbf{x})\delta(\mathbf{x})\rangle = \int_0^\infty \frac{dk}{k}\frac{k^3\langle|\delta^2(\mathbf{k})|\rangle}{2\pi^2} = \int_0^\infty \frac{dk}{k}\frac{k^3 P(k)}{2\pi^2} = \int_0^\infty \frac{dk}{k}\Delta^2(k), \tag{1.22}$$

donde se ha definido $P(k) \equiv \langle|\delta^2(\mathbf{k})|\rangle$ y $\Delta^2(k) \equiv \frac{k^3P(k)}{2\pi^2}$. La cantidad $\Delta^2(k)$ es denominada el espectro de potencias adimensional y es una cantidad importante en la teoría de formación de estructura, dado que mide la contribución de las perturbaciones por intervalo logarítmico en el número de onda k a la varianza en las fluctuaciones de densidad de materia. Se observa que el espectro de potencias de ΛCDM tiene una forma similar a $P(k) \sim k$ para k pequeño (escalas grandes) y $P(k) \sim k^{-3}$ (escalas pequeñas). Tiene un pico en $k^* \sim 2 \times 10^{-2}h$ Mpc^{-1} correspondiente a una longitud de onda $\lambda^* \sim 350h^{-1}$ Mpc.

La forma del espectro se explica bien mediante las fluctuaciones cuánticas en la época de inflación, dado que deberián ser aleatorias, las perturbaciones primordiales de la densidad tendrían que ser semejantes a un campo Gaussiano aleatorio, lo cual está confirmado por la forma y el espectro del CMB.

El origen del espectro está ligado a la amplitud de una fluctuación de una longitud de onda λ comóvil y su crecimiento temporal.

El crecimiento de las fluctuaciones se estudia en dos regiones, super-horizonte, donde se debe utilizar teoría de perturbaciones en relatividad general, y sub-horizonte, donde se usa un análisis de Newton-Jeans (Ma, C.-P., Bertschinger, E., 1995 [18]). La distribución de materia se determina a partir de fuentes de luz, como quasares, galaxias, supernovas, etc. Los sondeos de galaxias y grandes experimentos como el Sloan Digital Sky Survey (SDSS)[1] (Tegmark, M. et al., 2004 [16]), Dark Energy Survey(DES)[2] (Abbot, T.M.C., et al., 2019 [19]), o el Large Synoptic Sky Survey (LSST)[3], que aún se encuentra en construcción, han permitido y permitirán constreñir los parámetros de densidad de materia y energía con datos observacionales. Los sondeos mencionados son una forma de trazar las observaciones de luz emitida de diversos objetos astronómicos para obtener información de la distribución de materia en el Universo.

Es interesante notar que la línea superpuesta de las predicciones teóricas de ΛCDM sobre el espectro de potencias se ajuste tan bien a observaciones en el régimen lineal, mostrando la capacidad de la teoría y mostrando por qué es considerado como el modelo cosmológico estándar. Sin embargo, a escalas pequeñas el espectro se define por colapso gravitacional no-lineal y física bariónica, en esta parte el espectro debe ajustarse con perturbaciones de mayor orden o simulaciones numéricas.

1.6.2. Régimen no-lineal

Las perturbaciones a la densidad de materia crecen de manera $t^{2/3}$ en la época de dominación de materia. Después de recombinación, el espectro de potencias lineal mantiene su forma hasta el punto donde $\Delta^2(k)$ para un número de onda k llega al orden de ~ 1. Después de eso, la teoría lineal no es suficiente para explicar la distribución de materia y se debe recurrir a otros métodos para determinar el crecimiento de las fluctuaciones.

Uno de los métodos más usados son las simulaciones numéricas, ya que son usadas para modelar el crecimiento dado que en el régimen no-lineal, los modos no crecen de manera independiente. El acoplamiento de modos cambia la forma y amplitud del espectro de potencias en el rango no-lineal (Dodelson, S., 2003, [15]).

Existe un número de onda k_{nl} crítico que determina la parte del espectro que ha evolucionado hacia el régimen no lineal. Los modos de onda con

[1] https://www.sdss.org/
[2] https://www.darkenergysurvey.org/es/
[3] https://www.lsst.org/

$k < k_{nl}$ son no lineales, y los modos lineales son $k > k_{nl}$. Se define una escala de masa no-lineal evaluando la amplitud de las fluctuaciones de masa dentro de esferas de radio R a redshift z. La masa encerrada es $M = (4\pi/3)\bar{\rho}(z)R^3$. La varianza de las fluctuaciones de masa definida de esta manera es entonces

$$\langle(\delta(M)/M)^2\rangle \equiv \sigma^2(M) = \int d^3k^2 W_T^2(kR)P(k,z), \tag{1.23}$$

donde $W_T(kR)$ es la transformada de Fourier de la función ventana del tipo sombrero de copa (top-hat)

$$W(\mathbf{x}) = \begin{cases} \frac{3}{4\pi R^3} & \text{si } |\mathbf{x}| \leq R \\ 0 & \text{si } |\mathbf{x}| > R. \end{cases} \tag{1.24}$$

$P(k)$ puede aproximarse de manera local con una ley de potencias

$$P(k,z) = D^2(z)k^m, \tag{1.25}$$

donde D es el factor de crecimiento lineal, entonces la ecuación (1.23) lleva a

$$\sigma^2(M) \propto D^2 R^{(-3+m)} \propto D^2 M^{-(3+m)/3}. \tag{1.26}$$

A escalas muy pequeñas, se tiene el comportamiento $m \to -3$, de manera que las fluctuaciones tienden a un valor constante. Definiendo $\sigma(M_{nl}) = 1$ para la escala de masa no-lineal

$$M_{nl}(z) \propto D(z)^{6/(3+m)}. \tag{1.27}$$

Para $m > -3$, las escalas menores de masa se vuelven no-lineales primero, llevando a la formación de estructura jerárquica mencionada (White, S. D. M., 1994 [20]). Un elemento importante a notar es que para $k >> k_{nl}$ las estructuras de menor masa son formadas antes que las estructuras cuya cantidad de masa es mayor.

1.6.3. Modelo de colapso esférico

El modelo de colapso esférico genera una solución explícita de la evolución no lineal. Hay que considerar el caso en que un volumen esférico de masa M y radio R excede la escala de masa no-lineal. Considere una perturbación esférica de radio R para un contraste de densidad constante $\bar{\delta} = (3M/4\pi\bar{\rho}R^3) - 1$ en un Universo Einstein-de-Sitter (EdS). La ecuación de movimiento es

$$\frac{d^2R}{dt^2} = -\frac{GM}{R^2} = -\frac{4\pi G}{3}\bar{\rho}R. \tag{1.28}$$

Resolviendo las ecuación de movimiento (1.16), el Universo se expande a una tasa

$$\frac{d^2a}{dt^2} = -\frac{4\pi G}{3}\bar{\rho}a. \tag{1.29}$$

Al comparar las ecuaciones (1.28) y (1.29), es claro que las perturbaciones crecen como en un Universo de diferente densidad media, con la misma tasa de expansión inicial. Al integrar la ecuación (1.28) respecto al tiempo se obtiene

$$\frac{1}{2}\left(\frac{dR}{dt}\right)^2 - \frac{GM}{R} = E, \tag{1.30}$$

donde E es la energía total de la perturbación. Tomando la solución parametrizada $t = (\theta - \sin\theta)$ y $R = (1 - \cos\theta)$, en la región $E < 0$, las perturbaciones están ligadas

$$\frac{R}{R_m} = \frac{1-\cos\theta}{2}, \quad \frac{t}{t_m} = \frac{\theta - \sin\theta}{\pi}. \tag{1.31}$$

En el momento $t = 0$, o $\theta = 0$, este "cascarón" esférico se empieza a expandir hasta alcanzar un radio máximo o de turnaround $R_m$en $\theta = \pi$. Después, el cascarón vuelve a colapsar a un punto $\theta = 2\pi$. A medida que $t \to t_m$, $R \to 0$ y se dice que la fluctuación ha colapsado. El modelo de colapso esférico puede ser escalado a perturbaciones con masa arbitraria, usando propiedades físicas del objeto virializado. El radio virial r_{vir} se define como el radio de un volumen esférico dentro del cual la densidad media es $Delta_c$ multiplicada por el corrimiento al rojo ($M = 4\pi r_{vir}^3 \rho_{crit}\Delta_c/3$). Este radio es llamado el radio virial, formando un halo de materia oscura virializado. El radio virial define una región que describe bien el halo de materia oscura colapsado (Mo, H., van den Bosch, F. C., White, S., 2010 [21]). A pesar de su gran poder predictivo, el modelo de colapso esférico no es lo que pasa en la realidad, ya que los halos no solamente se expanden y colapsan de regiones densas para formar objetos virializados, en cambio, la estructura se forma de manera jerárquica, resultado de acreción de materia. Resulta interesante observar que la estadística obtenida mediante el uso de este modelo está en correspondencia con simulaciones de N-cuerpos.

1.6.4. Teoría de Press-Schechter

Obteniendo ya un modelo simple para describir la evolución de las fluctuaciones esféricas y propiedades observables del radio virial, el modelo de colapso esférico genera soluciones analíticas a las ecuaciones de evolución de

estructura. Se puede estimar el número de objetos virializados de masa M como función del redshift, relacionando la masa y el tiempo de colapso del halo con el campo de densidad lineal. Las simulaciones numéricas, ofrecen una idea y pueden usarse para este propósito, sin embargo, existe una aproximación analítica, introducida por primera vez por Press y Schechter (Press, W. H., Schechter, P., 1974 [22]), que resulta estar muy cercano a las simulaciones numéricas, incluso Press y Schechter fueron de los primeros en desarrollar simulaciones numéricas de formación de estructura con N-cuerpos.

La idea básica es imaginar la densidad cosmológica del campo suavizado en cualquier época z y a escala R tal que la escala de masa de objetos virializados satisfaga la relación $M = (4\pi/3)\bar{\rho}(z)R^3$. El campo de densidad, (suavizado o sin suavizar) es un campo aleatorio Gaussiano, la probabilidad de que la sobredensidad media en esferas de radio R exceda una sobredensidad crítica δ_c es

$$p(R,z) = \frac{2}{\sqrt{2\pi}\sigma(R,z)} \int_{\delta_c}^{\infty} d\delta \exp\left(-\frac{\delta^2}{2\sigma^2(R,z)}\right), \tag{1.32}$$

donde $\sigma(R,z)$ es la varianza de la densidad en esferas de radio R, dada por

$$\sigma^2(R,z) = \int \frac{d^3k}{(2\pi)^3} |\hat{W}_R(k)|^2 P(k,z), \tag{1.33}$$

donde $\hat{W}_R(k)$ es la transformada de Fourier de la función ventana del modelo top-hat esférico. Press y Schechter sugieren que la probabilidad definida en la ecuación (1.32) sea identificada con la fracción de partículas que son parte de un bulto de masa M, tomando $\delta_c = 1.686$, que es la sobredensidad lineal en la escala de virialización.

La fracción del volumen colapsado en objetos con masa entre M y $M+dM$ se da por $(dp/dM)dM$. Al multiplicar por el promedio de densidad de número de esos objetos ρ_m/M, la densidad de número de objetos colapsados entre M y $M + dM$ es

$$dn(M,z) = -\frac{\rho}{M}\frac{dp(M(R),z)}{dM}dM. \tag{1.34}$$

El signo negativo significa que p es una función decreciente de M. Usando el hecho de que $dM/dR = 3M/R$, se obtiene

$$\frac{dn(M,z)}{dM} = \sqrt{\frac{2}{\pi}}\frac{\bar{\rho}\delta_c}{3M^2\sigma} e^{-\delta_c^2/2\sigma^2} \left[-\frac{d\ln\sigma}{d\ln R}\right], \tag{1.35}$$

donde el término en los paréntesis cuadrados está relacionado con la forma del espectro de potencias, más objetivamente, su pendiente, cuyo valor en

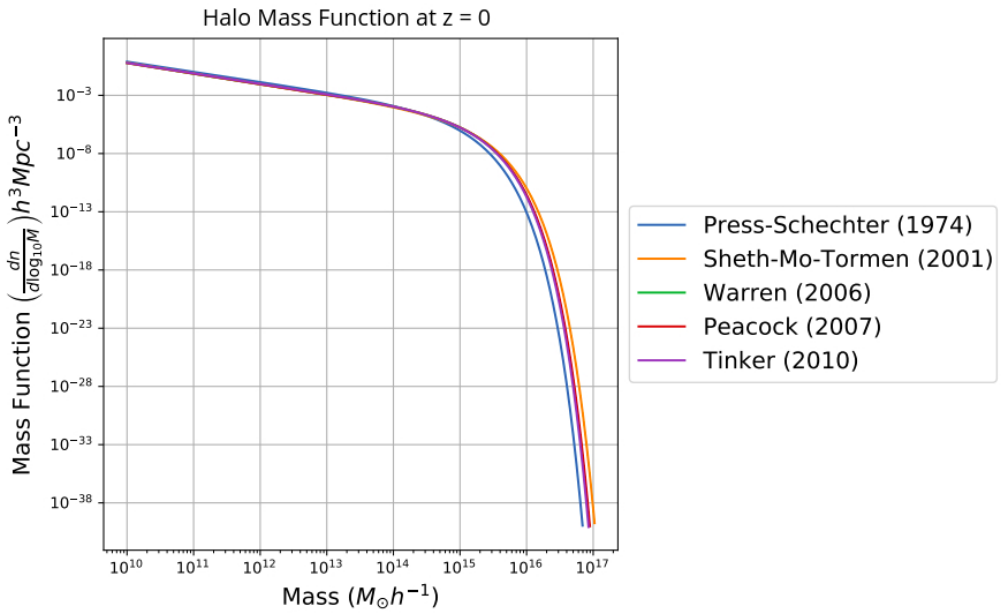


Figura 1.3: Halo mass function en el redshift $z = 0$ ajustado con diferentes parametrizaciones en escala logarítmica. Imagen generada con HMFcalc (Murray, S. G., 2013.)

escala de cúmulos de galaxias es cerca de 1. La ecuación (1.35) es la función de masa de halo, (*halo mass function*, en adelante, HMF), definida como la densidad de número de halos colapsados en función de su masa. Esta función tiene una forma de una ley de potencias multiplicado por una exponencial.

Existen otros ajustes paramétricos a la HMF para obtener de mejor manera la abundancia de halos de materia oscura predicha por simulaciones de N-cuerpos, (Sheth, R. K., Mo, H. J.; Tormen, G., 2001 [23]; Warren, M. S., Abazajian, K., Holz, D. E., Teodoro, L., 2006 [24]; Peacock, J. A., 2007 [25]; Tinker, J. L. and et al. 2010 [26]). Los ajustes involucran parámetros de simulaciones que pueden variar en volumen, resolución o definición de halos o del redshift. La figura 1.3 muestra la HMF para los ajustes paramétricos de Press-Schechter (1974), Warren (2006), Sheth-Mo-Tormen (2001), Peacock (2007) y Tinker (2010) en $z = 0$, (Figura creada con HMFcalc: Murray, S. G.; Power, C.; Robotham, A. S. G., 2013 [27]).

1.7. Big Data y la importancia de los datos en cosmología

Finalmente, como parte del proceso introductorio a este trabajo, se hará un breve resumen sobre la importancia de los datos en la astrofísica y la cosmología en los últimos años. Las observaciones y sondeos de hace un par de décadas no son comparables con los datos que pueden obtenerse hoy en día, ya que hace tiempo el volumen de datos que se obtenía de un sondeo completo, hoy puede realizarse en una sola noche de observación, y a menudo se necesita el análisis de datos en tiempo real. La astronomía moderna y la cosmología requieren un acercamiento al Big Data, (John Mashey (1998) fue uno de los primeros en utilizar el término) utilizar algoritmos de Machine Learning y análisis de imágenes y datos. La escalabilidad es un reto, además de que los algoritmos de machine learning son una buena herramienta para aplicar a la cosmología, dado que tienen que aprender de los errores, de datos con sesgo (bias) y ruido (Kremer, J., Stensbo-Smidt, K., Gieseke, F., Steenstrup Pedersen, K., Igel, C., 2017 [28]).

1.7.1. Volumen y escalabilidad de surveys

El Sloan Digital Sky Survey (SDSS) es uno de los surveys más importantes, dado que cada noche, recolecta alrededor de 200 GB de datos. Sin embargo, éste dió inicio hace aproximadamente dos décadas, es de esperarse que para épocas recientes, la cantidad de datos haya crecido en gran manera, lo cual se puede corroborar. Se estima que el LSST, próximo a iniciar operaciones, recolecte alrededor de 30 TB de información por noche. Evidentemente, hacer un análisis en tiempo real de tal cantidad de información se vuelve una tarea titánica para cosmólogos y científicos en general. En la figura 1.4 se observa la gran cantidad de datos de distintos surveys recolectados cada noche y la cantidad estimada de arreglos a futuro.

Una vez obtenidas las observaciones, se necesita una serie de algoritmos para extraer la información, como recién se obtuvo con la primera imagen de un agujero negro, recopilada por el Event Horizon Telescope (EHT), donde la cantidad de datos ronda los 5 petabytes.

1.7.2. Volumen en simulaciones numéricas

Las simulaciones no se quedan atrás, dado que también son un problema de escalabilidad de datos. La simulación del Milenio (Springel, V. and et al., 2005 [29]) tiene la característica de ser un problema cercano al Big

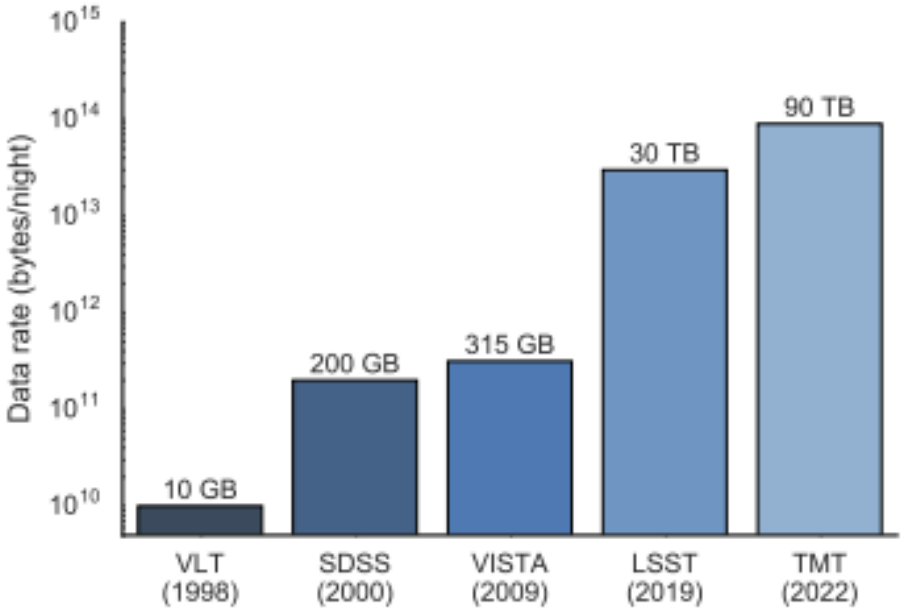


Figura 1.4: Incremento de la capacidad de observación y cantidad de datos de surveys en los últimos 20 años. Se lista: Very Large Telescope (VLT), Sloan Digital Sky Survey (SDSS), Visible and Infrared Telescope for Astronomy (VISTA) y Large Synoptic Survey Telescope (LSST) and Thirty Meter Telescope (TMT). Imagen tomada de Kremer, J. and et al., 2017.

Data. Utilizó más de 10 mil millones de partículas, a modo de observar la evolución de la distribución de materia en una región cúbica de 500 Mpc. La simulación mantuvo ocupado el superordenador principal del Centro de Supercomputación de la Sociedad Max Planck en Garching, Alemania. Las computadoras estuvieron funcionando de manera paralela alrededor de un mes, y la cantidad de información obtenida en esta particular simulación ronda los 25 Terabytes. La figura 1.5 es un acercamiento de un corte frontal de la simulación del milenio cuyo tamaño original es de 500 Mpc. Otras simulaciones cosmológicas más recientes y sofisticadas son Millenium-XXL (Angulo, R. E.; Springel, V.; White, S. D. M.; Jenkins, A.; Baugh, C. M.; Frenk, C. S., 2012, [30]), Horizon Run 3 (Kim, J.; Park, C.; Rossi, G.; Lee, S. M.; Gott, J. R., III, 2011, [31]), DEUS FUR (Alimi, J.-M.; Bouillot, V.; Rasera, Y.; Reverdy, V.; Corasaniti, P.-S.; Balmes, I.; Requena, S.; Delaruelle, X.; Richet, J.-N., 2012 [32]), ILLUSTRIS (Vogelsberger, M.; Genel, S.; Springel, V.; et al., 2014, [33]) y OUTER RIM (Heitmann, K.; et al., 2019, [19]).

El volumen de datos no es bajo en cosmología, como se ha demostrado, la cantidad de datos es inmensa, recolectada cada noche en observaciones al firmamento nocturno. Es preciso contar con la capacidad suficiente de almacenamiento así como equipos competentes para realizar un análisis profundo y en tiempo real. Sin duda, es un reto escalable que requiere de conocimiento en la ciencia de datos y en las áreas afines, como computación, software engineering e incluso, astrofísica y cosmología. En capítulos y secciones posteriores, se hablará un poco más a fondo de las herramientas para el análisis de datos, y del problema particular que se intentará resolver.

Figura 1.5: Acercamiento en la imagen de la simulación Illustris en $z = 0$ centrada en el clúster más masivo, la cual muestra la densidad de materia oscura superpuesta a la densidad de materia bariónica.

Capítulo 2

Cosmología Numérica

Durante los últimos años, la cosmología ha probado ser una de las ramas científicas con mayor provecho ya que utiliza diferentes esquemas y puntos de vista que al final sirven al propósito general de conocer cómo se comporta el Universo. Las observaciones del fondo cósmico de microondas han sido la base para llevar a cabo un sin fin de teorías, que a lo largo del tiempo han sido probadas o descartadas. Las pequeñas fluctuaciones alrededor de la homogeneidad de ese estado, junto con inestabilidades gravitacionales han resultado en los cúmulos de galaxias y vacíos que hoy se observan. Conocer y entender el comportamiento de las estructuras a gran escala en el Universo es un problema bien definido al día de hoy.

Estudiar la composición a gran escala del Universo es un problema complicado, pero puede ser solucionado. En este punto se proponen emplear simulaciones que involucren toda la fisica desarrollada en la teoría y que a su vez permitan tener resultados que den una percepción comparable con las observaciones. Así mismo, estas observaciones han llegado al punto de ser tan desafiantes que se requiere crear métodos mucho más sofisticados que puedan dar información relevante sobre la materia y energía oscura.

2.1. Simulaciones de N-Cuerpos

Las simulaciones de N-cuerpos proveen de una herramienta muy amplia para comprender la formación a gran escala del Universo, la inestabilidad gravitacional en escalas cosmológicas, y la formación y evolución de las galaxias. El poder computacional ha permitido crear simulaciones de alta resolución que muestran la evolución del Universo desde la época del CMB, iniciando de un z $\sim$ 1100. La formación de estructura se origina por pequeñas perturbaciones en la densidad de materia, en la época de inflación es debido a

fluctuaciones cuanticas que se expanden a escalas cosmologicas.

Para poder deducir las ecuaciones que rigen a la materia a grandes escalas, es necesario escoger un modelo cosmológico de base que describa la expansión del Universo. En un marco de Relatividad General, se debe escoger un modelo Friedmann-Lemaître-Robertson-Walker (FLRW) con sus respectivos parámetros cosmológicos, el parámetro de Hubble H_0 y las contribuciones debido a los bariones, materia oscura, energía oscura o constante cosmológica al parámetro de densidad total Ω_0.

Debido a que los sistemas autogravitacionales tales como galaxias están compuestos mayoritariamente de materia oscura, esta debe modelarse como un sistema no colisional, descrito por la ecuación de Boltzmann no colisional. Esta ecuación describe el comportamiento del espacio fase de la densidad de las partículas de materia oscura.

Para un sistema de este estilo, su descripción está dada por su masa m, coordenada comóvil $\mathbf{x}$ acoplada junto con la ecuación de Poisson. Este comportamiento y evolución del sistema bajo fuerzas externas tiene la siguiente forma

$$\frac{\partial f}{\partial t} + \mathbf{v} \cdot \nabla_r f + \frac{\mathbf{F}}{m_i} \cdot \nabla_v f = 0, \tag{2.1}$$

donde $f = f(\mathbf{r}, \mathbf{v}, t)$ es una función de distribución de la densidad, $\mathbf{v}$ es la velocidad, $\mathbf{r}$ es la posición, $\mathbf{F}$ es la fuerza y m_i la masa de cada que describen completamente al sistema de partículas (Reif, F., Scott, H. L., 1998 [34]).

Si esta fuerza $\mathbf{F}$ se deriva de un potencial Φ se tiene que

$$\mathbf{F} = -m_i \nabla \Phi. \tag{2.2}$$

Sustituyendo la ecuación (2.2) en (2.1) se encuentra

$$\frac{\partial f}{\partial t} + \mathbf{v} \cdot \nabla_r f - \nabla \Phi \cdot \nabla_v f = 0. \tag{2.3}$$

De esta manera se observa que la evolución del sistema no depende directamente de su masa, sino de una función f que depende más de su densidad y posición en el espacio fase. El potencial Φ debe satisfacer la ecuación de Poisson

$$\nabla^2 \Phi(\mathbf{r}, t) = 4\pi \int_S f(\mathbf{r}, \mathbf{v}, t) d^3 v, \tag{2.4}$$

donde S es el espacio comprendido en el volumen definido por la posición $\mathbf{r}$ y velocidad $\mathbf{v}$ y f se define mediante la siguiente expresión

$$f = f(\mathbf{r}, \mathbf{v}, t) d^3 v d^3 r, \tag{2.5}$$

que viene dada por la masa total de las partículas que se encuentran en un cubo de volumen d^3r centrado en $\mathbf{r}$ y velocidad ubicada en un cubo de volumen d^3v centrado en $\mathbf{v}$. Al integrar en el espacio, lo que se obtiene es la densidad de masa que puede depender del tiempo $\rho(t)$. Así la ecuación de Poisson descrita por la ecuación (2.4) se reduce a la conocida. Esencialmente, la ecuación de Boltzmann no colisional establece que el flujo de puntos a través del espacio fase es incompresible, o que la densidad del espacio fase alrededor del punto estudiado permanece constante.

Para conocer la evolución del fluido, es necesario utilizar un gran número de partículas, así que se debe resolver numéricamente de la siguiente manera:

- Dadas las coordenadas $\mathbf{r}_0$ y velocidad $\mathbf{v}_0$ iniciales de N partículas con masa m_i en el instante $t = t_0$, encontrar su posición $\mathbf{r}$ y velocidad $\mathbf{v}$ en un instante siguiente $t = t_{next}$, suponiendo que las partículas interaccionan solo gravitacionalmente. En esta aproximación la interacción se considera Newtoniana.

Si $\mathbf{r}_i$ y m_i es la coordenada y masa para cada partícula, entonces las ecuaciones de movimiento son

$$\frac{d^2\mathbf{r}_i}{dt^2} = -G \sum_{j=1, i\neq j}^{N} \frac{m_i(\mathbf{r}_i - \mathbf{r}_j)}{|\mathbf{r}_i - \mathbf{r}_j|^3}, \tag{2.6}$$

donde G es la constante de gravitación universal. Antes de resolver este conjunto de ecuaciones se debe considerar que las simulaciones gravitacionales de N-cuerpos son solo una aproximación al comportamiento verdadero de un sistema de esta dimensión. Para evitar aceleraciones grandes (ecuación 2.6) se debe introducir un *softening* gravitacional, denotado por ϵ, que actúa como un parámetro umbral para la distancia entre las partículas de materia oscura.

Este ϵ se agrega en el denominador de la ecuación (2.6), de manera que

$$\frac{d^2\mathbf{r}_i}{dt^2} = -G \sum_{j=1, i\neq j}^{N} \frac{m_j(\mathbf{r}_i - \mathbf{r}_j)}{(\Delta\mathbf{r}_{ij}^2 + \epsilon^2)^{3/2}}, \tag{2.7}$$

donde se ha definido $\Delta\mathbf{r}_{ij}^2 = |\mathbf{r}_i - \mathbf{r}_j|^2$ y ϵ es el parámetro de softening o "*softening length*" (Bodenheimer et al. 2007 [35]). Físicamente, se puede interpretar a este parámetro ϵ como la distancia entre los centros de dos partículas que están "unidas". Existe una explicación más detallada en el trabajo previo a este (Chacón, J., Vázquez-González, J. A., Gabbasov, R., 2020 [36]), se refiere al lector a esta publicación si desea conocer más sobre la implementación de N-cuerpos en cosmología.

La mayor limitación de este método es la escala, ya que lo hace como N^2, por lo que resolver el problema de N-cuerpos resulta más un problema de eficiencia computacional. Existen diversas aproximaciones para reducir el tiempo de cálculo sin arriesgar que el resultado no tenga sentido físico.

2.1.1. Particle-Mesh

Este método utiliza una malla para describir el campo de densidad de materia y otras de sus cantidades, además de calcular sus derivadas. Debido a la estructura de la ecuación de Poisson y de que se escoge un volumen que represente bastante bien al Universo, es conveniente escribir nuevamente las ecuaciones relevantes en el espacio de Fourier. Las transformaciones rápidas de Fourier (FFT) permiten que se reduzca el tiempo de CPU utilizada para cada paso de tiempo, aunque esta reducción en el tiempo de cómputo se ve afectada en el resultado debido a la pérdida de resolución y exactitud. El método Particle-Mesh (en adelante PM) no puede seguir de cerca las interacciones entre partículas a pequeña escala. De hecho, usar una cuadrícula para describir las cantidades de campo (como densidad, potencial y fuerza) no permite una representación justa en escalas más pequeñas que la distancia entre cuadrículas. Dado que la introducción de una malla computacional es equivalente a un suavizado local del campo, para el método PM no es necesario adoptar el parámetro de suavizado en la expresión de la fuerza y/o potencial gravitacional.

PM resuelve de la siguiente manera en cada paso temporal:

1. Calcula la densidad en cada punto de la malla a partir de la distribución espacial de las partículas.

2. Se soluciona la ecuación de Poisson para el potencial.

3. Calcula la fuerza en cada punto de la malla.

4. Estima la fuerza en la posición de cada partícula utilizando un esquema de interpolación adecuado.

5. Integra las ecuaciones de movimiento.

En un volumen computacional de lado L y número total de partículas N, cada una con masa m_p una malla tridimensional con M nodos tendría un espaciamiento $\Delta \equiv L/M$, cada uno indentificándose por su coordenada espacial $\mathbf{x}_{i,j,k}$ con $i, j, k = 1, ..., \mathrm{M}$.

La densidad de masa ρ en el espacio de Fourier se describe como

$$\rho(\mathbf{x}_{i,j,k}) = m_p M^3 \sum_{l=1}^{N} W(\delta\mathbf{x}_l), \tag{2.8}$$

donde W es una función de interpolación ajustada para la cantidad $\delta\mathbf{x}_l = \mathbf{x}_l - \mathbf{x}_{i,j,k}$, l indica la posición de la partícula e i, j, k indican la posición de la malla considerada en ese punto. La elección de W depende de la exactitud que requiera tener el resultado. Las interpolaciones más comunes son:

- "Nearest Grid Point" (NGP): en este caso, la masa de cada partícula se asigna por completo al punto más cercano a la malla. La consecuencia de esto es que la densidad muestra una descontinuidad cada vez que una partícula cruza el borde de la malla. la interpolación NGP se escribe como
$$w_i = 1, \ M|\delta x_i| \leq 1/2. \tag{2.9}$$
- "Cloud In Cell" (CIC): la masa de las partículas se asigna a los dos puntos cercanos ($2^3 = 8$ en el caso 3D) de una manera inversamente proporcional con respecto a la distancia del punto de la malla; de esta manera la densidad varía de manera continua si una partícula cruza el borde de la cuadrícula, aunque su gradiente sigue siendo discontinuo. La interpolación CIC se describe de la siguiente manera
$$w_i = 1 - M|\delta(x_i)|, \ M|\delta x_i| \leq 1. \tag{2.10}$$
- "Triangular Shaped Cell" (TSC): la descomposición de la masa incluye tres puntos más cercanos ($3^3 = 27$ para el caso 3D). Así, el gradiente de la densidad varía de manera suave cuando se cruza la frontera. Su segunda derivada es discontinua. La interpolación TSC se escribe como
$$w_i = \begin{cases} 3/4 - M^2|\delta x_i|^2, & M|\delta x_i| \leq 1/2 \\ (1/2)(3/2 - M|\delta x_i|)^2, & 1/2 \leq M|\delta x_i| \leq 3/2. \end{cases} \tag{2.11}$$

La ecuación de Poisson en el espacio de Fourier es más sencilla de tratar, ya que tiene la siguiente forma

$$\Phi_k = G_k \delta_k, \tag{2.12}$$

donde Φ_k y δ_k son las transformadas de Fourier del potencial gravitacional y del contraste de densidad. En la ecuación (2.12) G_k representa una función de Green adecuada del Laplaciano, para el cual, en el caso de un sistema

continuo se da por $G_k \propto k^{-2}$; aunque existen otras funciones dependiendo del sistema.

Para obtener la fuerza en cada punto de la malla se requiere diferenciar el potencial Φ para derivar el i-ésimo componente de la fuerza

$$F(\mathbf{x}) = -m_p \frac{d\Phi}{dx_i}. \tag{2.13}$$

Esto puede realizarse utilizando diferencias finitas para resolver ecuaciones diferenciales. La aproximación depende del número de puntos de la malla dentro del cálculo. Por ejemplo la componente x de la fuerza para dos puntos, que es el de menor orden, se tiene que

$$\frac{F_x(\mathbf{x}_{i,j,k})}{m_p} = \frac{\Phi(\mathbf{x}_{i-1.j,k}) - \Phi(\mathbf{x}_{i+1,k,k})}{2\Delta}. \tag{2.14}$$

Es bien sabido que los esquemas de diferencias finitas introducen errores de truncamiento en las soluciones. Una forma de evitar este problema y obtener una mayor precisión es obtener las fuerzas directamente del potencial gravitacional en el espacio de Fourier: $F_k = -i\mathbf{k}\Phi_k$. Se necesita hacer la transformación inversa de manera individual para cada componente de la fuerza, usando rutinas de FFT más frecuentes.

La gran ventaja del método PM es la reducción de tiempo de cálculo, de hecho, el número de operaciones se escala como $N_p + N_g \log(N_g)$, donde N_p es el número de partículas y $N_g = M^3$ el número de puntos de la cuadrícula. Esta ventaja se paga con el rango dinámico del método, ya que está muy limitado al número de puntos de la cuadrícula y por la ocupación de memoria. La manera de alcanzar una resolución óptima para simulaciones cosmológicas es utilizar métodos híbridos.

2.1.2. Códigos tipo árbol

La exactitud y el desempeño de este código es lo que lo hace el más popular para realizar simulaciones cosmológicas. La idea de resolver el problema de N-cuerpos se basa en la expansión multipolar jerárquica, el denominado algoritmo de árbol. Esta forma rápida de cálculo se obtiene, para partículas suficientemente distanciadas, usando una sola fuerza multipolar, a pesar de calcular todas las distancias como se requiere para métodos de suma directa. De esta manera, la suma reduce el orden de operaciones a $N_p \log(N_p)$. La expansión multipolar se basa en un agrupamiento jerárquico que se obtiene subdividiendo el volumen de la simulación de manera recursiva. Esto es debido a que se considera un cubo mínimo que reúne a todas las partículas. Al calcular la expansión multipolar del potencial de las partículas, cosiderando

el suavizamiento y el centro de masa, se ubica una partícula y se hace la pregunta : ¿Es la distancia del centro de masa del conjunto agrupado mayor que el tamaño del cubo inicial dividido por algún parámetro a escoger?

Es decir, se pregunta si se cumple la relación

$$r > \frac{l}{\theta}, \tag{2.15}$$

donde r es la distancia de la partícula al centro de masa del agrupamiento, l es el largo del cubo inicial y θ es un parámetro de precisión. Si la expresión (2.15) resulta ser cierta para todas las partículas de la simulación, ésta sigue su evolución, si una o más de ellas no satisface esa condición, el cubo inicial se divide en un cubo más pequeño de lado $l/2$ y se repite el proceso. El cálculo de la fuerza se realiza descendiendo el árbol. Empezando desde el nodo raíz, el código evalúa aplicando el criterio mencionado y decide si la expansion multipolar del nodo resulta en una fuerza parcial suficientemente exacta. Se calculan las expansiones multipolares y los centros de masa para cada cubo y la pregunta se vuelve a repetir para cada proceso. Mientras los grupos de partículas sean más pequeños y distantes, la expansión multipolar tendrá mayor exactitud.

2.1.3. Métodos Híbridos: Tree-PM

Es posible combinar los métodos mencionados en las subsecciones 2.1.1 y 2.1.2, y usar las ventajas de ambos. Los códigos híbridos se construyen reemplazando los métodos de suma directa por algoritmos de árbol, denomidados códigos TreePM. En este caso, el potencial gravitacional en el espacio de Fourier se divide en dos términos, la parte de largo alcance y la parte de corto alcance

$$\Phi_k = \Phi_k^{long} + \Phi_k^{short}, \tag{2.16}$$

$$\Phi_k^{long} = \Phi_k \exp\left(-k^2 r_s^2\right), \tag{2.17}$$

donde r_s corresponde a la escala espacial donde se realiza la división de la fuerza. El potencial de largo alcance se calcula de manera eficiente con métodos de malla en el espacio de Fourier, como el método PM. La parte de corto alcance del potencial se resuelve en el espacio real notando que para $r_s \ll L$, la solución de la ecuación de Poisson en este rango está dada por

$$\Phi^{short}(\mathbf{x}) = -G \sum_i \frac{m_i}{\mathbf{r}_i} \mathrm{erfc}\left(\frac{\mathbf{r}_i}{2r_s}\right). \tag{2.18}$$

En la ecuación (2.18), $\mathbf{r}_i$ es la distancia de cualquier partícula al punto $\mathbf{x}$. La fuerza de corto alcance se calcula entonces usando el algoritmo de árbol, excepto que se ve modificada por un término de corte para largo alcance. Esta aproximación permite mejorar el desempeño computacional manteniendo las ventajas de los métodos: el rango dinámico a gran escala, la insensibilidad de agrupamiento y el control preciso de la escala de suavizado de la fuerza gravitacional (Moscardini, L. & Dolag, K. 2011 [37]). En la sección posterior, se describirá uno de los códigos más populares para realizar simulaciones cosmológicas, además de algunos ejemplos realizados para este trabajo.

2.2. El código GADGET

Por sus siglas, GAlaxies with Dark matter and Gas IntEracT (GADGET), es un código de uso libre hasta su segunda versión, GADGET-2 (Springel, V. 2005 [38]). Hace uso del método de N-cuerpos para simular partículas de materia oscura, así como Smoothed Particle Hidrodynamics (SPH) para simular materia bariónica y física gaseosa, la cual permite poblar, por ejemplo, los halos de materia oscura creados, de esta manera es posible observar galaxias y cúmulos de galaxias masivos en la simulación.

El código está escrito en lenguaje C y utiliza dos recursos computacionales principales: Paralelización y el algoritmo Tree-Particle-Mesh (TreePM). Si se usara un método tradicional para propósitos de cálculo, se requerirían $N(N-1)$ operaciones de las N partículas. Esto llevaría un tiempo muy grande, dado que se requiere conocer la fuerza entre partículas. El método TreePM reduce el tiempo a un orden de $N \ln N$ mediante la selección de partículas en un cubo de tamaño determinado mínimo. Este proceso, junto con la paralelización, que permite que para un sistema de millones de partículas se efectúen cálculos sin la pérdida de mucha resolución.

2.2.1. Ejemplo: Simulación de formación de estructura a gran escala

Es un ejemplo que puede ejecutarse fácilmente. Está compuesto de 32^3 partículas de materia oscura y bariónica respectivamente dentro de una caja periódica de $50h^{-1}$ Mpc de lado en un Universo tipo ΛCDM. La simulación distribuye a las partículas en una malla cúbica. Cada centro de la malla tiene una población de materia oscura que rodea a partículas bariónicas. Una perturbación en las condiciones iniciales hace que las partículas inicien su movimiento y eventualmente formen estructuras pobladas de halos de materia oscura y galaxias. El código inicia desde un redshitf inicial de $z = 10$

y termina en la época actual $z = 0$. Los parámetros de esta simulación se indican en la tabla 2.1.

Tabla 2.1: Parámetros de una simulación de ΛCDM con GADGET.

Descripción	Símbolo	Valor
Densidad de materia oscura	Ω_0	0.3
Densidad de energía oscura	Ω_Λ	0.7
Densidad de materia bariónica	Ω_b	0.04
Parámetro de Hubble ($h = H_0/100$ Mpc $\cdot$ km $\cdot$ s^{-1})	h	0.7

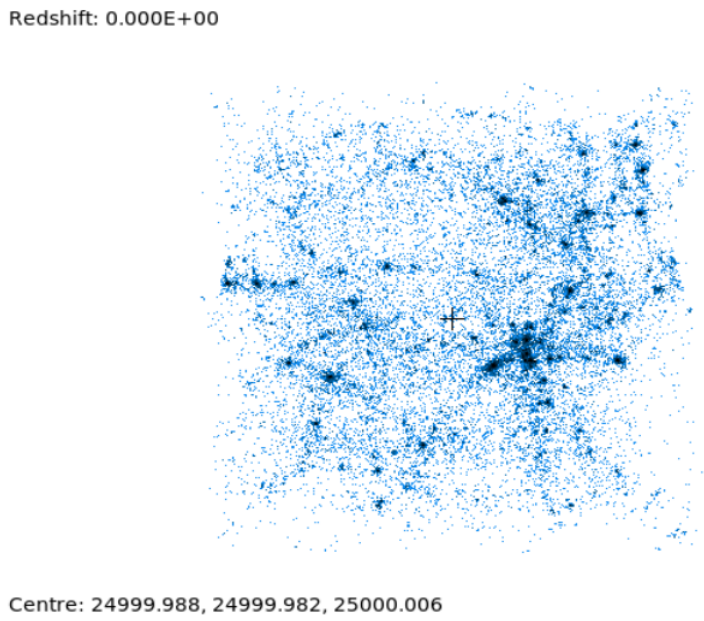


Figura 2.1: Corte frontal de la distribución de materia oscura a redshift $z = 0$ en la simulación de ejemplo del código GADGET. En este snapshot existen 32^3 partículas en total esparcidas en una caja de longitud comóvil de $L = 50h^{-1}$ Mpc. Imagen ilustrativa obtenida en la realización de este trabajo.

2.3. Simulaciones cosmológicas con GADGET-2

En general, para efectuar una simulación numérica se debe contar con un archivo de condiciones iniciales. En este trabajo se utilizo el código N-GenIC[1]. Se puede utilizar otro generador de condiciones iniciales como 2LPTiC[2] o MUSIC[3]. Para el código GADGET-2, las condiciones iniciales deben tener la siguiente información:

1. Número de partículas en la simulación.
2. Archivo de campo de densidad. Puede ser tipo glass (White, S. D. M., 1994 [20]) o malla cartesiana (Sprinkel, V. & et al., 2012 [30]). Este viene dentro de los archivos de condiciones iniciales.
3. Redshift inicial z_i.
4. Densidad de materia oscura Ω_m.
5. Densidad de energía oscura Ω_Λ.
6. Parámetro de Hubble ($h = H_0/100$).
7. Tamaño de la caja de simulación, con o sin condiciones peródicas a la frontera.
8. Normalización del espectro de potencias de masa σ_8.

Las condiciones iniciales se generan mediante la *aproximación Zeldovich*, la cual describe la evolución no lineal del estado de un campo de densidad de materia generada por una perturbación gravitacional. El campo de densidad se considera homogéneo y no-colisional, justamente para reproducir las propiedades de la materia oscura.

Durante el procedimiento de este trabajo, se generaron campos de densidad de las condiciones iniciales indicadas en la tabla 2.2 y se hizo evolucionar el sistema. Cabe destacar que no se utilizaron partículas de materia bariónica, solo partículas de materia oscura, con el fin de obtener una distribución de materia como se observa en la figura 2.2.

[1] https://www.h-its.org/2014/11/05/ngenic-code/
[2] https://cosmo.nyu.edu/roman/2LPT/
[3] https://www-n.oca.eu/ohahn/MUSIC/

Tabla 2.2: Condiciones iniciales de la simulación cosmológica

Descripción	Símbolo	Valor
Densidad de materia oscura	Ω_m	0.268
Densidad de energía oscura	Ω_Λ	0.683
Densidad de materia bariónica	Ω_b	0.049
Tamaño de caja	L	50 Mpc
No. de partículas	N	192^3
Redshift inicial	z_{init}	23
Redshift final	z_f	0
Parámetro de Hubble	h	0.7
Normalización del espectro de potencias de masa	σ_8	0.8
Cantidades técnicas de la simulación		
ErrorTolIntAccuracy		0.025
MaxRMSDisplacementFact		0.2
CourantFact		0.15
MaxSizeTimestep		0.03
ErrorTolTheta		0.5
TypeOfOpeningCriterion		1
ErrTolForceAcc		0.005

2.3.1. Suavizado gravitacional

Como se mencionó en la sección 2.1 de este capítulo, el suavizado gravitacional es uno de los parámetros importantes para ejecutar una simulación, dado que este evita el cálculo de divergencias de fuerza ejercida entre partículas de materia oscura. Este parámetro no puede ser cero ni aleatoriamente grande, dado que la distribución de densidades es definida mediante un kernel del tipo Gaussiano.

Para una distribución de densidad de masa del tipo Gaussiana, la aceleración gravitacional de una partícula de prueba está definida por la ecuación

$$\ddot{\mathbf{r}} = \frac{GM(<r)}{r^3}\mathbf{r}, \tag{2.19}$$

donde $M(<r)$ es la cantidad de masa dentro de un radio r del centro del kernel Gaussiano. Esta masa puede parametrizarse nuevamente como

$$\frac{M(<r)}{M(r=\infty)} = \frac{\int_0^r \exp\left(\frac{-2r^2}{\lambda^2}\right)4\pi r^2 dr}{\int_0^\infty \exp\left(\frac{-2r^2}{\lambda^2}\right)4\pi r^2 dr}$$

$$= \text{erf}\left(\frac{\sqrt{2}r}{\lambda}\right) - 1.13\exp\left(-\frac{2r^2}{\lambda^2}\right)\frac{\sqrt{2}r}{\lambda} \tag{2.20}$$

donde $\text{erf}\left(\frac{\sqrt{2}r}{\lambda}\right)$ es una función de error del kernel Gaussiano, definida por

$$\text{erf}(z) \equiv \frac{2}{\sqrt{\pi}}\int_0^z \exp(-t^2)dt. \tag{2.21}$$

Estudios recientes (Zhang, J., Tsai, Y.-L. S., Kuo, J. L., Cheung, K., Chu, M.-C., 2018 [39]) han demostrado que el mejor ajuste del valor de suavizado gravitacional para una distribución de densidad Gaussiana tiene un valor de $\epsilon = 0.89$ kpc, como se observa en la figura 2.3. Este fue el valor de suavizamiento utilizado en las simulación de ΛCDM.

2.3.2. Dinámica no colisional

La ecuación de Boltzmann no colosional (2.1) es el límite continuo de la materia oscura no interactuante. La alta dimensionalidad requiere que

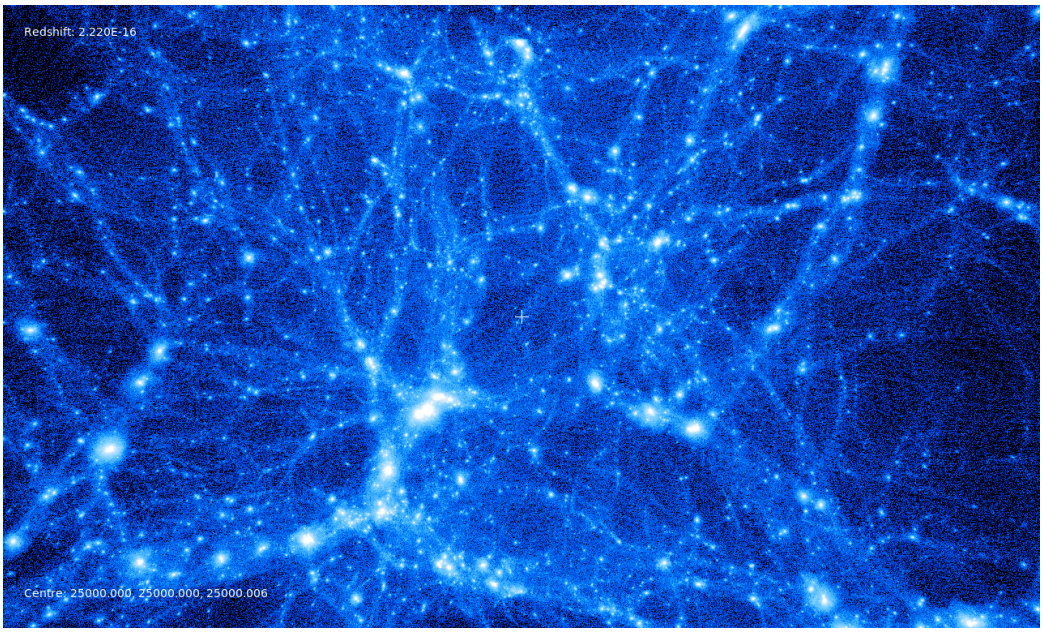


Figura 2.2: Corte frontal de la simulación de 192^3 partículas de materia oscura evolucionada desde $z = 23$ hasta $z = 0$. Los halos de materia oscura son los puntos más brillantes. En esta simulación se contabilizaron cerca de 400 halos de materia oscura cuya masa es mayor que $10^{12}M_{\odot}$.

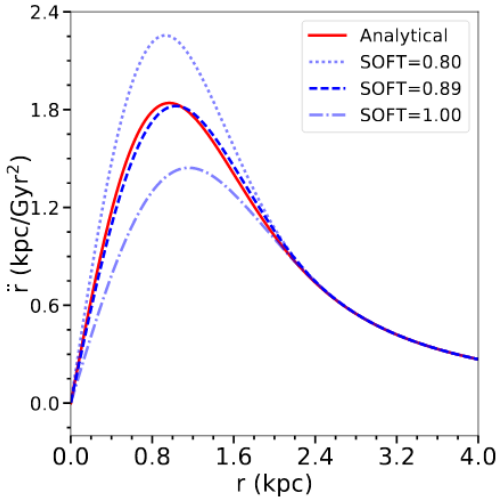


Figura 2.3: Comparación de la curva de aceleracióon analıtica de una distribucion de una densidad de masa gaussiana con diferentes longitudes de suavizado gravitacional. La curva de mejor ajuste al cálculo analítico es la de valor $\epsilon = 0.89$ kpc. Figura tomada de Zhang, J. et al., 2018.

las soluciones del problema se obtengan con el método de N-cuerpos. La dinámica de estas partículas se describe con el Hamiltioniano

$$H = \sum_i \frac{\mathbf{p}_i^2}{2m_i a(t)^2} + \frac{1}{2}\sum_i \frac{m_i m_j \varphi(\mathbf{x}_i - \mathbf{x}_j)}{a(t)}, \tag{2.22}$$

donde $H = H(\mathbf{p}_1, ..., \mathbf{p}_N, \mathbf{x}_1, ..., \mathbf{x}_N, t)$ y $\mathbf{x}_i$ son los vectores de las coordenadas comóviles, y el momento canónico es $\mathbf{p}_i = a^2 m_i \dot{\mathbf{x}}_i$. El hamiltoniano tiene una dependencia temporal debido a la evolución del factor de escala proveniente de que el modelo es tipo FLRW.

Suponiendo condiciones periódicas en la frontera, para una caja de lado L^3 el potencial de interacción $\varphi(\mathbf{x})$ es solución de la ecuación

$$\nabla^(2)\varphi(\mathbf{x}) = 4\pi G\left[-\frac{1}{L^3} + \sum_i \delta(\mathbf{x} - \mathbf{n}L)\right], \tag{2.23}$$

donde la suma sobre $\mathbf{n} = (n_1, n_2, n_3)$ se extiende sobre una triada de números enteros. Esta corresponde a un potencial peculiar donde la dinámica de sistema se rige por $\nabla^2\phi(\mathbf{x}) = 4\pi G[\rho(\mathbf{x}) - \bar{\rho}]$. El potencial peculiar para un sistema discreto está definido por

$$\phi(\mathbf{x}) = \sum_i m_i \varphi(\mathbf{x} - \mathbf{x}_i). \tag{2.24}$$

La función de distribución de densidad se define por la $\delta(\mathbf{x})$ de Dirac convolucionada con un kernel de suavizado de escala comóvil ϵ. Para esto se utiliza una función de spline cúbico definido por $\delta(\mathbf{x}) = W(|\mathbf{x}|, 2.8\epsilon)$ (Monaghan, J. J.; Lattanzio, J. C. 1985 [40]), donde W cumple

$$W(r,h) = \frac{8}{\pi h^3} \begin{cases} 1 - 6\left(\frac{r}{h}\right)^2 + 6\left(\frac{r}{h}\right)^3, & 0 \leq \frac{r}{h} \leq \frac{1}{2}, \\ 2\left(1 - \frac{r}{h}\right)^3, & \frac{1}{2} < \frac{r}{h} \leq 1, \\ 0, & \frac{r}{h} > 1. \end{cases} \tag{2.25}$$

esta función es de soporte compacto, donde las interacciones entre partículas se anulan en $r > 2h$, la continuidad de la función permite además que el desorden de partículas no influye en resultado de la integración. El núcleo de interpolación permite reducir la dimensionalidad de las ecuaciones y convierte la interacción de una partícula con su entorno en algo similar a centrarse en esa misma partícula y promediar su entorno en un campo de densidad.

2.3.3. Espectro de potencias de masa

Como se mencionó en la sección 1.6.1 del capítulo 1, el espectro de potencias de masa es la cantidad que permite medir las fluctuaciones de densidad del CMB para explicar la distribución de materia en el Universo. La manera de obtenerlo es mediante el valor de la normalización del espectro σ_8. Una manera de medir estas fluctuaciones es mediante las anisotropías de la temperatura del CMB. Observaciones del espectro mediante la sonda COBE (White, S. D. M.; Efstathiou, G.; Frenk, C. S., 1993 [41]) constriñen la amplitud del espectro en grandes escalas, es decir a $k \sim 0.001h$ Mpc^{-1}. Otro método utilizado es el conteo de número de cúmulos emisores de rayos-X en el Universo local (Bahcall, N. A.; Ostriker, J. P.; Perlmutter, S.; Steinhardt, P. J., 1999 [42]). La abundancia de estos objetos es sensible a la amplitud de las fluctuaciones de densidad en escalas alrededor de $8h$ Mpc^{-1}, que corresponden a una masa de $10^{15}h^{-1}M_\odot$.

La forma y amplitud del espectro de potencias de masa de las fluctuaciones de densidad contiene información acerca de la cantidad de materia y su comportamiento. Las mediciones directas del espectro de potencias se obtienen de observaciones y conteo de galaxias.

Para este fin, se ha modificado el código CAMB (Lewis, A., Challinor, A., Lasenby, A., 2000 [43]), el cual resuelve una serie de ecuaciones analíticas tipo Boltzmann para obtener el espectro de potencias de masa de una distribución de densidad, mediante perturbaciones al contenido de materia y energía en el Universo. Al efectuar la modificación se utilizaron valores iguales a los de la tabla 2.2, adjunto a esto se hace una reconstrucción del espectro de

potencias de la simulación numérica con el código POWMES (Colombi, S., Jaffe, A., Novikov, D. Pichon, C., 2009 [44]), el cual efectúa transformaciones de Fourier inversas para este fin. El resultado final fue la reconstrucción del espectro de potencias de masa de la simulación de N-cuerpos en $z = 0$, tal como se observa en la figura 2.4. En el régimen no lineal del espectro ($k \gg 0$) se observa que los datos de CAMB y de las simulaciones concuerdan muy bien, incluso las simulación se asemeja bastante en el inicio de la misma que naturalmente se desacopla al avanzar temporalmente. La parte lineal, por otra parte, no coincide, pero esto es debido al tamaño de la caja de simulación, ya que, como se ha mencionado, el espectro de potencias debe crecer de manera lineal a grandes escalas.

2.4. ROCKSTAR Halo Finder

Al ejecutar una de estas simulaciones es necesario extraer la mayor cantidad de información posible. En particular, se está interesado en los catálogos de halos de materia oscura, dado que contienen información sobre las partículas dentro del halo, como velocidad, posición, masa y radio. Existen una gran variedad de halo finders, como se les suele llamar, en este trabajo se utilizó ROCKSTAR (Robust Overdensity Calculation using K-Space Topologically Adaptive Refinement). En resumen, el código identifica halos de materia oscura y subestructuras, identificadas como Parent halos & halos. Usa un refinamiento de distancia entre partículas en un espacio fase de seis dimensiones además de una dimensión temporal, lo cual permite que la identificación de subestructura sea más efectiva e idependiente de la malla y la forma del halo (Behroozi, P. S., Wechsler, R. H.& Wu, H.-Y., 2013 [45]).

En la figura 2.5 se observa el corte frontal de la simulación numérica usada en este trabajo, destacando la abundancia de halos masivos ($M \sim 10^{12}\mathrm{M}_{\odot}$) y la respectiva identificación con el código ROCKSTAR y el paquete de uso libre de python YT (Turk, Matthew J.; Smith, Britton D.; Oishi, Jeffrey S.; Skory, Stephen; Skillman, Samuel W.; Abel, Tom; Norman, Michael L., 2011 [46]), una herramienta útil que permite analizar sistemas astrofísicos simulados.

2.4.1. Halo Mass Function

Como se ha mencionado, la cantidad de información obtenida de una sola simulación (de hecho, de un solo snapshot de la simulación) puede ser bastante grande, todo depende de qué es lo que se quiera analizar. Uno de estos resultados, muy importante por supuesto es la Halo Mass Function de

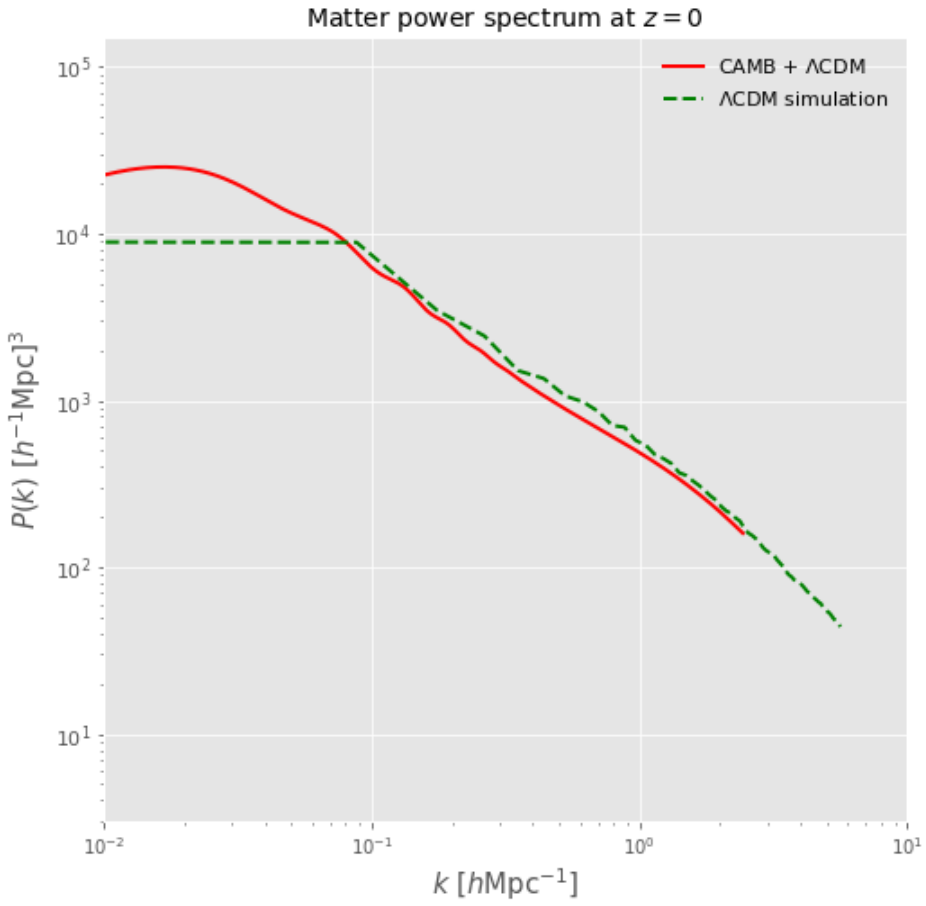


Figura 2.4: Espectro de potencias reconstruido para la simulación de ΛCDM comparado con el espectro obtenido con el Código CAMB en $z = 0$. Las curvas representan la distribución de materia en la época actual. La línea constante en escalas grandes ($k \ll 0$) de la simulación es debido al tamaño de la caja. A escalas pequeñas ($k > 0$ se observa la gran similitud entre ambas gráficas. Gráfica generada de los resultados de la simulación de la tabla 2.2

la simulación en el redshift $z = 0$. La densidad de número de halos en función de su masa es una de las cantidades más importantes al momento de querer evaluar observaciones y simulaciones numéricas, tal como se menciona en la sección 1.6.4. La Teoría de Press-Schechter otorga una idea general de la cantidad de halos de materia oscura presentes en el Universo dependiendo de su masa. Vale la pena volver a decir que Press y Schechter fueron de los

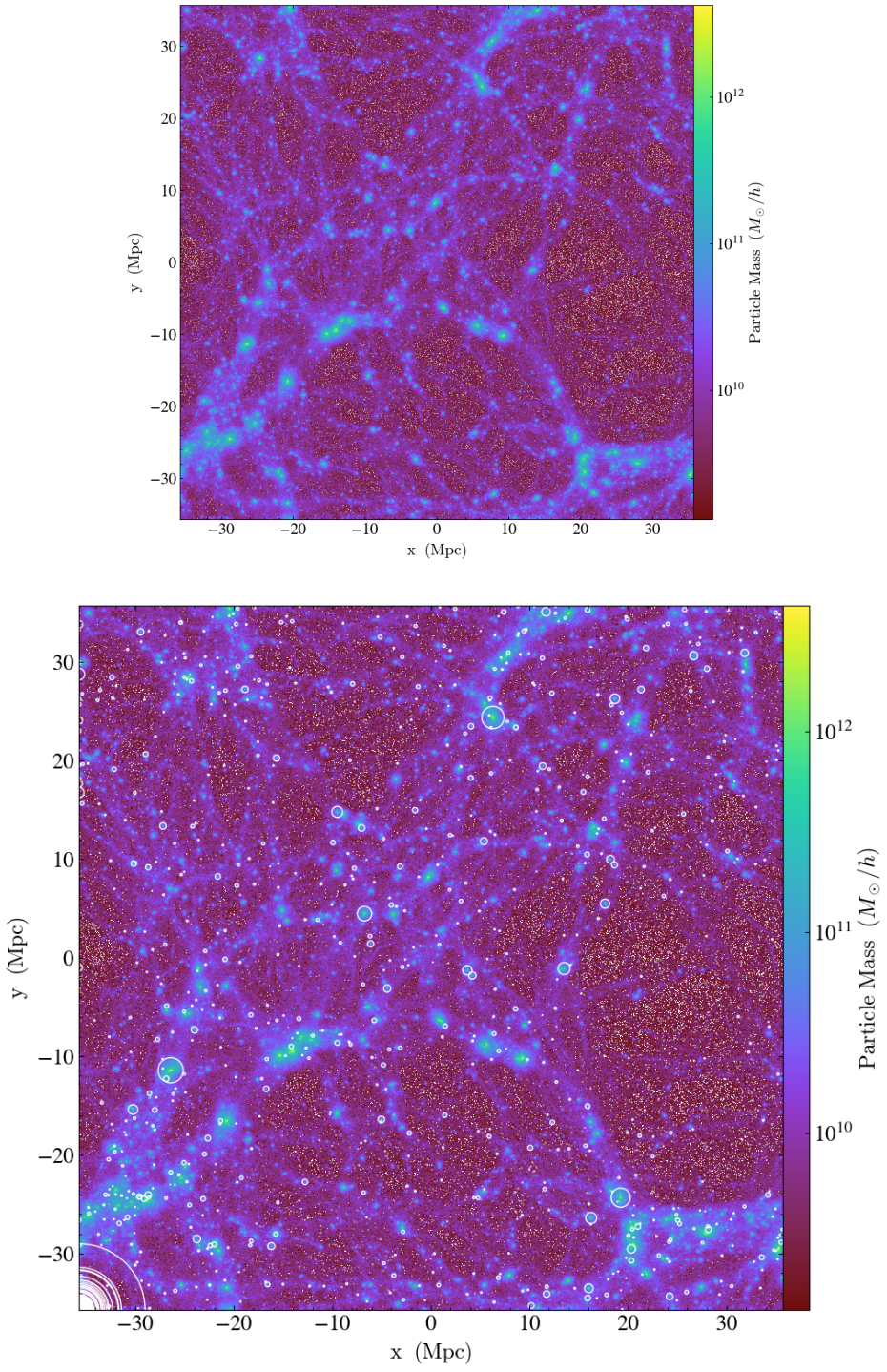


Figura 2.5: Corte del resultado de una simulación de N-cuerpos antes y después de ejecutar el buscador de halos. En la imagen se detallan los halos que son parte de estructuras llamadas parents. Obtenida del análisis de los datos de la simulación de la tabla 2.2 y YT.

primeros en desarrollar un código de N-cuerpos, con la finalidad de obtener la HMF.

Nuevamente, utilizando el paquete de análisis astrofísico y cosmológico de python: , así como la identificación de estructura y subestructura de halos de materia oscura en la simulación con ROCKSTAR, fue posible reconstruir la HMF, tal como se presenta en la figura 2.6. En ella, se grafica la densidad de número de halos de materia oscura (en escala logarítmica) en función de su masa (cuyo rango está entre 10^{11} $M_\odot$ y 10^{14} $M_\odot$). Además, se incluyen los ajustes a la función hechas por Press-Schechter (1974) así como Tinker (2008) a manera de comparación. La simulación de ΛCDM tiene un comportamiento muy parecido a los ajustes mencionados, excepto en el rango de $\sim 10^{11}$ $M_\odot$. Esto es principalmente por el tiempo de ejecución, ya que inicia en $z = 23$, por lo que no exisitió un tiempo suficiente para formar halos de materia oscura menos masivos.

Ahora bien, ya se ha dado a entender que la formación de estructura es jerárquica, los pequeños cúmulos se aglomeran para formar halos más y más masivos. El espectro de potencias reconstruido para la simulación se asemeja bastante a la parte no lineal de la teoría de ΛCDM. Más aún, la densidad de número de halos concuerda bastante bien con los ajustes teóricos mencionados, exceptuando los halos menos masivos (pero eso ya se ha explicado). De manera que, la ejecución computacional concuerda bien con datos teóricos, es momento de ir más a fondo.

La simulación será utilizada como modelo de entrenamiento de algoritmos de machine learning, con el objetivo de obtener información de la relación entre los halos formados al final de la ejecución y de las condiciones iniciales.

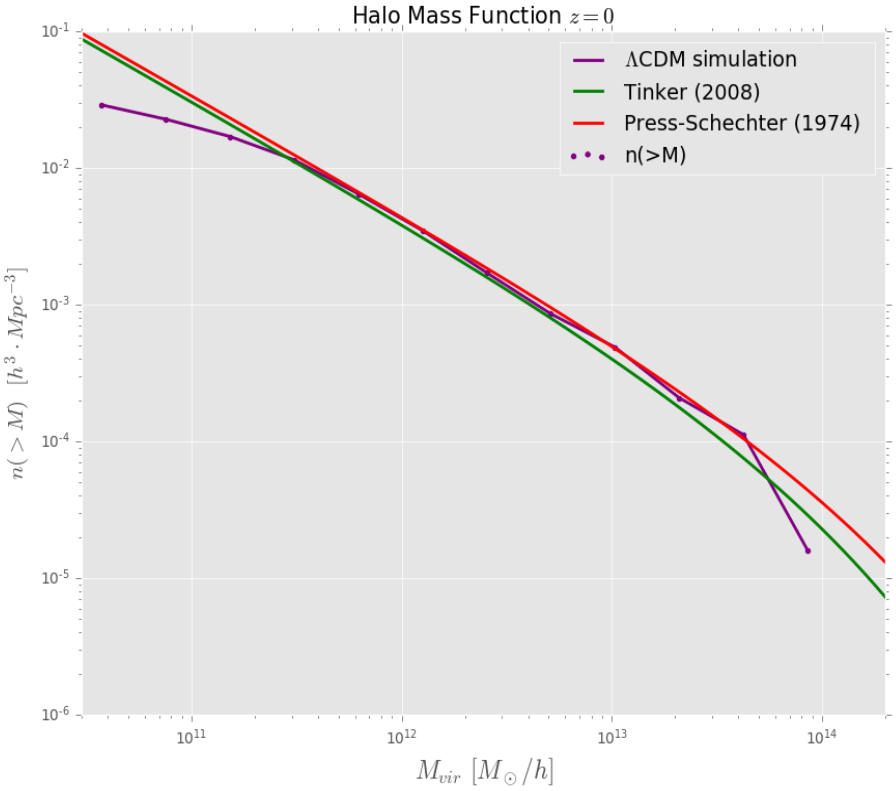


Figura 2.6: Halo Mass Function de Press-Schechter (1974), Tinker (2008) y el reconstruido para la simulación de ΛCDM en $z = 0$. Entre el rango de masa de 10^{11} $M\odot$ y 10^{14} $M_\odot$, la semejanza con la teoría es notable. En la figura se grafica la masa virial del halo versus la densidad de número de halos de materia oscura. El pequeõ corte en la parte izquierda de la figura es debido al volumen de la simulación y la sensibilidad de la resolución de ROCKSTAR para poder encontrar halos dado cierto umbral de partículas de materia oscura.

Capítulo 3

Machine Learning

El término machine learning hace referencia a la manera de aprendizaje automático que puede llevar a cabo una "máquina" mediante la adaptación de ciertos algoritmos respecto a una cantidad de datos de entrada al sistema. Se dice que esta máquina está aprendiendo si mejora su rendimiento en diferentes tareas después de llevar a cabo observaciones o mediciones de su entorno (Russel, S., Norvig P., 2009 [47]). Todos los modelos de machine learning están afectados por una compensación en su capacidad para minimizar el sesgo y varianza en las predicciones. El sesgo o bias es la diferencia entre el promedio de la predicción del modelo y el valor correcto que se intenta predecir. La varianza es la variabilidad de la predicción del modelo para un valor correcto dado.

Una de las principales características de machine learning es la capacidad de aprendizaje de los distintos algoritmos usados en esta rama de la inteligencia artificial. Los componentes de una máquina pueden mejorar mediante el aprendizaje previo de datos, sin embargo, la mejoría depende de 3 factores principales

- Cual componente debe mejorarse.
- Que representación se usa para los datos y para el componente.
- El feedback o retroalimentación que se puede obtener mediante el proceso de aprendizaje.

Un modelo que ajuste pobremente los datos tiene alto sesgo y poca varianza, es decir que no determina bien su predicción. Por otro lado, un modelo que sobre ajuste los datos tiene alta varianza y un valor de sesgo muy bajo. Se debe encontrar la manera de nivelar estos valores para obtener una predicción correcta y óptima del modelo. Entender estos errores es clave al

momento de escoger un algoritmo de machine learning que sea apropiado para un problema dado y obtener algoritmos eficientes. Existen dos clases de algoritmos, los supervisados y no supervisados, que son utilizados en gran cantidad de proyectos y ramas dentro de la ciencia y de la industria. Se describen a continuación.

Algoritmos supervisados: Requieren datos de entrada que contengan variables independientes (predictores) y variables dependientes (objetivo). En el proceso, el componente "aprende" como predecir el valor de la variable independiente basada en los predictores. En el aprendizaje supervisado, se ejerce la aplicación de un par de datos de entrada-salida y la máquina aprende o ejecuta una función que mapea de los datos de entrada a los datos de salida.

Ejemplos de estos algoritmos son: Regresión Logística, Árboles de Decisión, Redes Neuronales, Support Vector Machines, etc (Gron, A. 2017, [48]).

Algoritmos no supervisados: Estos no identifican una variable objetivo y tratan a todas las variables de igual manera. El objetivo de estos algoritmos no es predecir un valor o un resultado, sino observar e identificar patrones, agrupamientos u otras maneras de caracterizar los datos. En el aprendizaje no supervisado, el algoritmo aprende patrones en la entrada, aunque no se suministre reatroalimentación de manera explícita.

Ejemplos de estos algoritmos son: Cluster Analysis, Correlación y Principal Component Analysis (PCA) (Goodfellow, I., Bengio, Y., Courville, A., 2016, [49]).

3.1. Aprendizaje Supervisado

La tarea del aprendizaje supervisado es la siguiente:

Dado un conjunto de entrenamiento o **training set** de N pares de entrada-salida

$$(x_1, y_1), (x_2, y_2), ..., (x_N, y_N), \tag{3.1}$$

donde cada y_j se calculó mediante una función tipo $y = f(x)$, encontrar una función g que aproxime a la verdadera función f. Las variables x e y pueden tomar cualquier valor y no necesariamente tiene que leer un valor numérico, puede ser un atributo. La función g es una hipótesis.

El aprendizaje se efectúa a través de una búsqueda dentro del espacio de posibles hipótesis por una función que tenga un buen rendimiento, incluso al alimentarla de nuevos ejemplos más allá del conjunto de entrenamiento.

Para medir la exactitud de la hipótesis se proporciona de un conjunto de de prueba o **test set** distinto del conjunto de entrenamiento. Se dice que la hipótesis g generaliza bien a la función f si predice correctamente el valor de y para otros valores.

La variable dependiente y puede resultar ser categórica, también llamada cualitativa o de atributos. Los valores de una variable categórica son mutuamente excluyentes y en ese caso el problema de aprendizaje se denomina de **clasificación**, el cual a su vez se denomina como Booleano o de clasificación binaria si solo dos valores son posibles.

Cuando la variable dependiente y resulta en un valor numérico, el problema de aprendizaje será de **regresión**.

Ejemplos de variables categóricas	
Tipos de datos	Ejemplos
Numérico	Sexo (1 = Mujer, 0 = Hombre, clasificación binaria) Resultados de una encuesta (0 = De acuerdo, 1 = Neutral, 2 = En desacuerdo)
Texto	Formas de pago (Efectivo o crédito) Tipos de producto (Madera, plástico, metal)
Fecha/Hora	Días de la semana (lunes, miércoles, etc.) Meses del año (enero, mayo, septiembre, etc.)

Tabla 3.1: Variables categóricas en aprendizaje supervisado

En muchos algoritmos de machine learning el tamaño del conjunto de entrenamiento y la precisión de clasificación están estrechamente relacionados. En particular, el desempeño de un algoritmo es pobre si el conjunto de datos de entrenamiento es muy pequeño.

3.1.1. Regresión Logística

La regresión logística es un algoritmo de clasificación utilizado para asignar observaciones a un conjunto discreto de clases. A diferencia de la regresión lineal que genera valores numéricos continuos, la regresión logística transforma su salida utilizando la función sigmoide logística para devolver un valor de probabilidad que luego puede asignarse a dos o más clases discretas.

La regresión logística utiliza la función sigmoide para predecir una variable dependiente binaria representado por indicadores entre “0” y “1”. En este algoritmo se utilizan probabilidades logarítmicas para el valor etiquetado como “1”, que a su vez se determinan a partir de una combinación lineal

de los valores de las variables independientes. La manera de efectuar la predicción de la variable independiente se efectúa mediante una probabilidad de ocurrencia.

De esta manera se debe aproximar de obtener "0", es decir, que no ocurre cierto suceso o en caso contrario, de obtener "1", si ocurre el suceso. La probabilidad aproximada del suceso se aproxima mediante la función logística o sigmoide (Hosmer, D. W., Lemeshow, S. 2000 [50])

$$\sigma(x) = \frac{e^{\theta_0+\theta_1 x}}{1 + e^{\theta_0+\theta_1 x}}, \tag{3.2}$$

donde θ_0 y θ_1 son pesos o parámetros determinados a partir de los datos de entrenamiento y x la variable independiente, que en un conjunto de datos son los atributos o características. La manera de calcular se efectúa mediante la estimación de máxima verosimilitud. Los mejores coeficientes resultarían en un modelo que predice un valor muy cercano a 1, o un valor muy cercano a 0 para la clase contraria. La intuición de máxima verosimilitud es un procedimiento que busca valores para los coeficientes que minimicen el error de las probabilidades predichas por un modelo a partir de los datos. La gráfica de la función sigmoide se muestra en la figura 3.1.

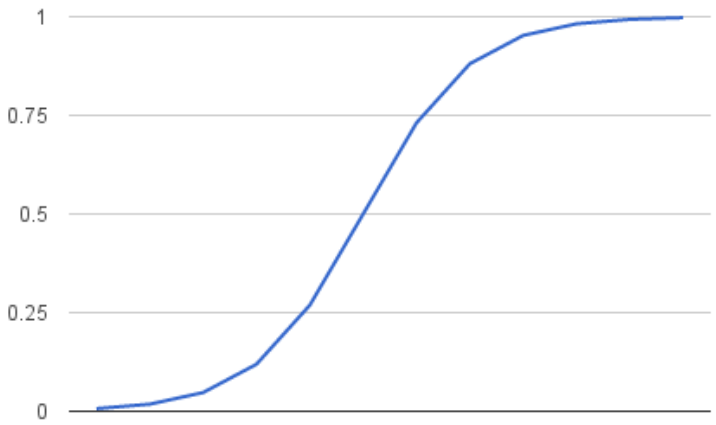


Figura 3.1: Función Logística o sigmoide. Los valores tienden a un límite ya sea 0 o 1. El eje horizontal indica los valores que puede tomar la función $f(x)$ y el eje vertical los valores que toma la función de activación $\sigma(x)$. El típico valor umbral para llevar a cabo una decisión está en 0.5.

La ecuación (3.2) puede generalizarse de la forma

$$\sigma(x) = \frac{e^{f(X)}}{1 + e^{f(X)}}. \tag{3.3}$$

Aquí $f(X)$ es una función generalizada del modelo de regresión lineal donde X es un vector de variables independientes, abtributos o características. La función sigmoide puede tomar cualquier valor real y aproximarlo al valor 0 o 1, aunque nunca puede tomar exactamente ninguno de esos límites. Para predecir a qué clase pertenece un conjunto de datos, se debe establecer un valor umbral. Por ejemplo, si el valor predicho es ≥ 0.5 entonces se clasifica como 1, de otra manera se clasifica como 0. Existen otro tipo de funciones logísticas como $\tanh \mathbf{x}$, RELU, o paso binario, que son extensiones a la función sigmoide descrita, la diferencia básica entre ellas es el dominio en el que cae la variable dependiente. En el caso de $\tanh \mathbf{x}$ el dominio va entre -1 y 1, para RELU, el dominio va de 0 hasta ∞, mientras que en el paso binario el dominio está también entre 0 y 1, a deferencia de la sigmoide no tiene un comportamiento suave por lo que su diferenciabilidad complicaría la activación de la función (Karlic, B.; Olgac, A.; 2011 [51]).

La manera de evaluar este tipo de algoritmos es mediante el uso de una función de costo llamada Cross-Entropy, o entropía cruzada, también conocida como pérdida logarítmica. La función de costo puede separarse para valores de ciertas clases, en especial para una clasificación binaria, existe una función de costo para el valor 1 y otra función de costo para el valor cero.

La función de costo se obtiene de una función de error, $J(\theta)$ que promedia el error de los valores calculados respecto a los predichos. En la regresión lineal, esta función de error es el Mean Squared Error o Error Cuadrático Promedio, y que tiene la forma

$$J(\theta) = \frac{1}{m} \sum_{i=1}^{m} \left(y^{(i)} - \hat{y}^{(i)} \right)^2, \tag{3.4}$$

donde $y^{(i)}$ es el valor real del i-esimo dato y $\hat{y}^{(i)}$ es el valor i-esimo predicho por el modelo. Sin embargo, la regresión logísitica utiliza la función de error

$$\begin{aligned} J(\theta) &= \frac{1}{m} \sum_{i=1}^{m} \text{Cost}\left(y^{(i)}, \hat{y}^{(i)} \right) \\ \text{Cost}(y, \hat{y}) &= -\log(\hat{y}) \qquad \text{si } y = 1 \\ \text{Cost}(y, \hat{y}) &= -\log(1 - \hat{y}) \quad \text{si } y = 0. \end{aligned} \tag{3.5}$$

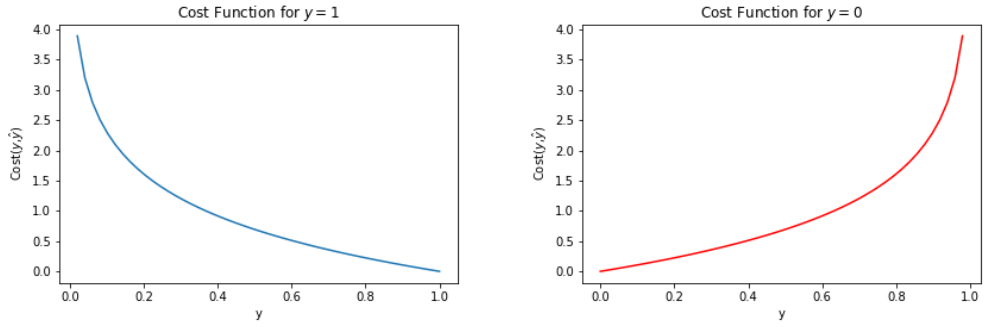


Figura 3.2: Funciones de costo para $y = 1$ y $y = 0$ respectivamente

La ecuación (3.5) puede reacomodarse en una sola

$$J(\theta) = \frac{1}{m} \sum_{i=1}^{m} \left[y^{(i)} \log \left(\hat{y}^{(i)} \right) + \left(1 - y^{(i)} \right) \log \left(1 - \hat{y}^{(i)} \right) \right]. \tag{3.6}$$

La manera de reducir el valor de costo es mediante la recursión de Gradient Descent o Descenso de Gradiente. Este es un algoritmo de optimización utilizado para minimizar algunas funciones moviéndose de manera iterativa en la dirección del descenso más pronunciado definido por el negativo del gradiente. Este proceso se usa para actualizar de manera iterativa los parámetros de un modelo y se obtiene de la siguiente manera

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta} J(\theta), \tag{3.7}$$

donde α se conoce como Learning Rate o Tasa de Aprendizaje. La manera de visualizar el descenso de gradiente es imaginándose en la cima de una montaña, si se quiere llegar a la parte más baja tomando el camino con mayor pronunciación en el descenso, es decir en el que se tengan que realizar menos pasos, estos pasos son la tasa de aprendizaje y el dar un paso es análogo a una iteración que actualiza cada vez más este coeficiente α.

3.1.2. Árboles de Decisión

Este algoritmo es un tipo de diagrama de flujo para los datos, donde los bloques terminales representan decisiones de clasificación. Dado un conjunto de datos, se puede calcular la inconsistencia dentro del conjunto, o en otras palabras, encontrar su entropía, con la finalidad de dividir el conjunto hasta que todos los datos estén dentro de una clase (Quinlan R. L., 1986 [52]).

En este ambiente, los valores de entrada a partir de los cuales se representa una regla de clasificación solo pueden conocerse a través de sus mismos

atributos o características. Los árboles de decisión se representan en función de estos mismos atributos.

La manera en que un algoritmo de árbol de decisión funciona es mediante un proceso del tipo inductivo. Teniendo un conjunto de objetos, éstos son descritos mediante una colección de atributos. Cada atributo mide la importancia de la característica del objeto. Usualmente estas características definen un conjunto pequeño de valores mutuamente exclusivos.

Un árbol de decisión llega a una conclusión al llevar a cabo una serie de pruebas. Los nodos del árbol hacen pruebas sobre los atributos de los valores de entrada, A_i, y las ramas que provienen del nodo están etiquetadas con los posibles valores del atributo, $A_i = v_{ik}$. Los nodos hoja en el árbol especifican un valor que debe ser calculado por la función.

La manera de efectuar un buen algoritmo de decisión como este se lleva a cabo mediante una división de datos, de manera que se escoja el atributo con mayor peso o con el que se obtenga una ganancia alta de información (que se explica más detalladamente en la sección 3.1.3) de manera que se espera tener una correcta clasificación con el menor número posible de pruebas.

3.1.3. Información y Entropía

Regularmente los conjuntos de datos de los que se disponen son muy grandes y contienen muchos atributos. Los árboles de decisión hacen una división de datos con el objetivo de obtener una mayor información luego de realizada la división. Se puede pensar que esta división es una manera de organizar el desorden de los datos.

Es por esta razón que en el proceso de aprendizaje, se debe enfocar en la manera de obtener una mejor visión sobre lo que se piensa analizar. Esto proviene directamente de la teoría de la información. Se obtiene mayor información de eventos mayormente improbables que de los eventos que son más probables. Por ejemplo, un mensaje que diga "hoy salió el sol" da mucho menos información que el que provee "hoy hubo un eclipse solar en la mañana".

Se busca entonces una manera de determinar la información de manera formal y específica, mencionando que los eventos más probables proveen bajo contenido de información, mientras que los eventos menos probables proveen el mayor contenido de información.

La ecuación que satisface estas condiciones es la de contenido de información de un evento x_i

$$I(x_i) = -\log_2 P(x_i), \tag{3.8}$$

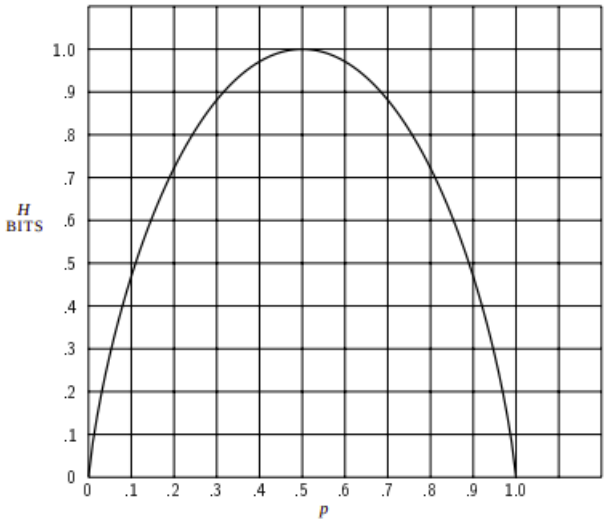


Figura 3.3: Entropía para dos clases con probabilidad p y $(1-p)$. Imagen tomada de Shannon, C. E. 1948. La entropía de Shannon es una manera de medir la cantidad relativa entre las dos clases. El valor de la entropía es máximo si existe la misma cantidad de clases.

donde $P(x)$ es la probabilidad de que el evento x ocurra. Como esta definición es para un solo evento, la manera de cuantificar la información de una distribución de probabilidad se hace mediante la entropía de Shannon (Shannon, C. E., 1948 [53])

$$H(x) = -\sum_i P(x_i) \log_2 P(x_i), \tag{3.9}$$

donde i indica que es la suma de todos los eventos posibles. Es decir, la entropía de Shannon es la cantidad esperada de información en un evento de una distribución de probabilidades (figura 3.3). El cambio de la información evaluada antes y después de la división se conoce como **ganancia de información**. La división se hace entonces cuando la ganancia de información es mayor.

Existen otras métricas para evaluar la información y el desorden de los datos, tal es el caso de la impureza de Gini. La impureza de Gini mide que tan seguido un elemento elegido aleatoriamente de un conjunto sería etiquetado erróneamente. Puede calcularse sumando la probabilidad de cada elemento siendo elegido multiplicado por la probabilidad de un error en la clasificación del mismo elemento. Alcanza su valor mínimo (cero) cuando todos los casos son etiquetados correctamente, es decir que corresponden a una categoría solamente.

La fórmula para calcular la impureza de Gini viene dada por

$$I_G(x) = 1 - \sum_i f_i(x)^2, \tag{3.10}$$

donde $f_i(x)$ es la fracción de elementos clasificados con valor i. En el apéndice A se encontrará un ejemplo de clasificación Booleana aplicado a un conjunto de datos pequeño, a manera de ilustrar el funcionamiento de un árbol de decisión.

3.1.4. Random Forest

Gran parte del ambiente de Machine Learning se centra en la clasificación, se quiere conocer a qué clase o grupo pertenece una observación. Clasificar observaciones de manera precisa es extremadamente valioso para distintas empresas y exisiten diversas aplicaciones, como predecir si un usuario comprará un producto o preveer si un préstamo debería ser aprobado o no.

De los algoritmos que existen para clasificación, uno que destaca es el Random Forest Classifier, existe también el Random Forest Regressor que, como su nombre lo indica, aplica regresión.

Random Forest consiste en gran número de árboles de decisión que operan como un ensamble de manera conjunta. Cada árbol individual del Random Forest elige una predicción de clase y la clase con mayoría de votos se vuelve la predicción del modelo. Esto es debido a un concepto simple, pero poderoso: La sabiduría de las multitudes. La razón de que Random Forest sea tan buen algoritmo es debido a que un gran número de árboles relativamente no correlacionados operando de manera conjunta tendrá un mejor desempeño que cualquier modelo individual que lo constituya (Breiman, L., 2001 [54]).

La baja correlación entre modelos es la clave. Al igual que las inversiones con bajas correlaciones (como acciones y bonos), se unen para formar un elemento que es mayor que la suma de sus partes. Los modelos no correlacionados pueden producir predicciones en conjunto que son más precisas que cualquiera de las predicciones individuales que la constituya.

La razón de esto es porque los árboles se protegen unos a otros de sus errores individuales (siempre y cuando esos errores no estén en la misma dirección). Si unos árboles tienen errores, otros pueden tener razón y predicciones correctas, de manera que, como grupo, los árboles pueden moverse a la dirección de correcta predicción. Algo así como pasa en la naturaleza, los árboles tienden a crecer en la dirección dónde obtengan mayor luz solar, mayor cantidad de agua, etc.

Existen dos prerrequisitos para que Random Forest tenga un buen desempeño, estos son:

1. Las características y atributos de los datos deben tener una verdadera señal o de manera que los modelos que usen dichas características puedan elegir mejor y no de manera aleatoria.

2. Las predicciones (y por tanto, los errores) hechos por los árboles individuales deben tener baja correlación entre si.

Los efectos de modelos no correlacionados se pueden entender suponiendo el siguiente juego de apuesta de dinero:

- Usar un generador de números aleatorios con una distribución uniforme entre cero y cien.

- Si el número generado es mayor o igual que 40, el lector gana (existe un 60 % de probabilidades de ganar) algo de dinero. Si el número es menor a 40, pierde la cantidad apostada.

Se ofrecen las siguientes opciones:

- Juego 1: Jugar 100 veces apostando 1$ cada vez.

- Juego 2: Jugar 10 veces apostando 10$ cada vez.

- Juego 3: Jugar una vez, apostando 100$.

¿Cuál opción sería la mejor?. Nótese que el valor esperado de cada juego es el mismo

$$\sigma(J_1) = (0.60 * 1 + 0.40 * (-1)) * 100 = 20$$
$$\sigma(J_2) = (0.60 * 10 + 0.40 * (-10)) * 10 = 20$$
$$\sigma(J_3) = 0.60 * 100 + 0.40 * (-100) = 20$$

Sin embargo, al observar la distribución (3.4), la mejor opción es evidente. El juego 1 ofrece la mayor probabilidad de ganar dinero, siendo 97 %, seguido del juego 2, bajando a 63 % y terminando con el juego 3, con la probabilidad de 60 %, como siempre. Si se divide más la apuesta, es más probable ganar dinero. Esto funciona porque cada juego es independiente de los otros.

Lo mismo sucede con Random Forest, en este ejemplo, cada árbol es un juego. Las probabilidades de ganar incrementan mientras mayores juegos se llevan a cabo. Similarmente con un modelo de Random Forest, las probabilidades de hacer predicciones correctas incrementan con el número de árboles no correlacionados en el modelo.

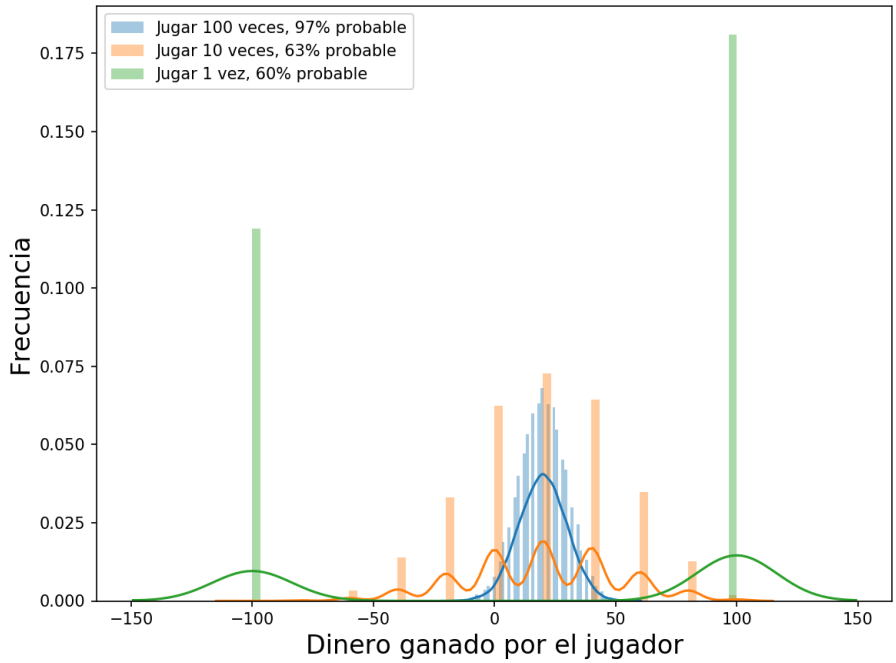


Figura 3.4: Ejemplo de un juego de generador de números aleatorios. La manera de ganar es si el generador obtiene un número mayor o igual a 40. El juego 1 (97 %) ofrece la mayor probabilidad de ganar.

Pero ¿Cómo se asegura Random Forest de que el comportamiento individual de cada árbol no esté muy correlacionado con otro de los árboles dentro del modelo? Usa los siguientes métodos.

Bagging (Bootstrap Aggregation): Los árboles de decisión son muy sensibes y susceptibles respecto de los datos con los cuales se entrenan. Un pequeño cambio al conjunto de entrenamiento puede dar diferentes resultados. Random Forest toma ventaja de esto, permitiendo que cada árbol individual muestree aleatoriamente del conjunto de dadts haciendo reemplazos, lo que resulta en diferentes árboles. Este proceso es conocido como bagging.

Con el bagging no se generan subconjuntos de los datos de entrenamiento en fragmentos más pequeños ni se entrena cada árbol de manera diferente. Más bien, al tomar una muestra de tamaño N se está alimentando al árbol con un conjunto de entrenamiento de tamaño N (a menos que se especifique lo contrario). En vez de tener los datos originales se trabaja con una muestra aleatoria de tamaño N a la cual se le reemplazan datos.

Por ejemplo, si los datos de entrenamiento eran [1, 2, 3, 4, 5, 6], entonces es posible darle a uno de los árboles dentro del clasificador la siguiente lista [1, 2, 2, 3, 6, 6]. Observe que ambas listas tienen una longitud de seis y que

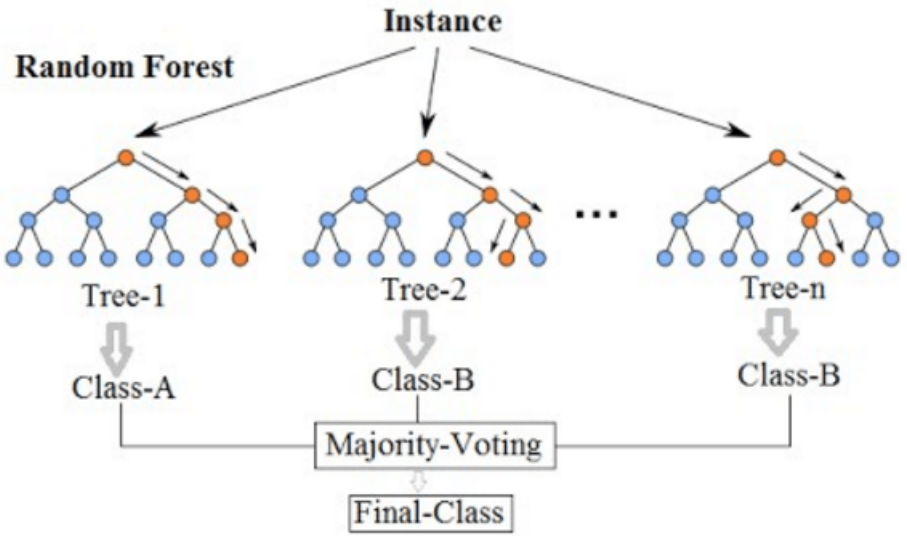


Figura 3.5: Diagrama de un algoritmo de Random Forest. Al ser un ensamble de árboles de decisión, permite realizar diferentes pruebas sobre una selección aleatoria de atributos, siendo la clase final un voto sobre una mayoría obtenida en cada árbol individual. Imagen tomada de Medium.

"2" y "6" se repiten en los datos de entrenamiento seleccionados al azar que se le otorga al árbol (porque las muestras tienen reemplazo).

Aleatoriedad de variables: Cuando un árbol de decisión hace la división de los datos para construir un nodo, considera todos los atributos (variables) que se tengan a disponibilidad, a menos que se reduzcan con base en su importancia (Deng, H., Runger, G., Tuv, E., 2011 [55]) y se elige el que produzca una mayor separación entre observaciones para cada nodo. En contraste, el algoritmo de Random Forest selecciona de manera aleatoria distintos atributos para cada árbol dentro del ensamble. Esto obliga a que exista una variación aún mayor entre los árboles del emsamble y en última instancia, dar como resultado una menor correlación entre árboles y una mayor diversificación. El resultado final es que no sólo se obtienen árboles que están entrenados en distintos conjuntos de datos, si no que se usan diferentes atributos para llevar a cabo las decisiones.

3.2. Evaluación de Modelos

Un problema importante en la minería de datos es el desarrollo de indicadores eficientes que soporten la calidad de los resultados del análisis. Evaluar el desempeño de un algoritmo es un aspecto fundamental en machine lear-

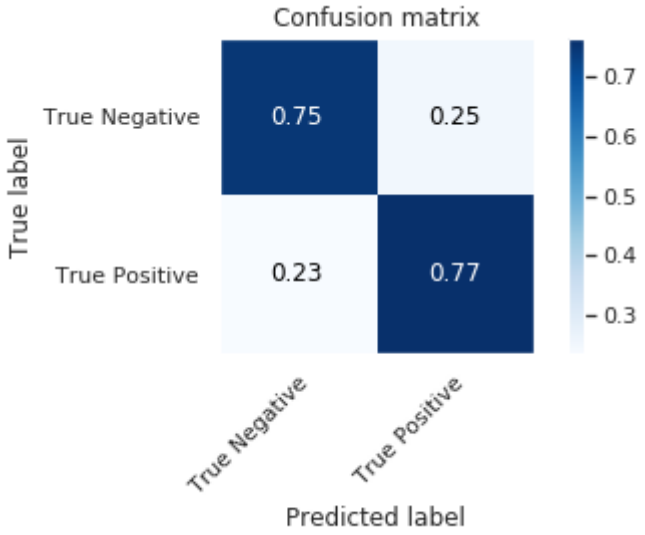


Figura 3.6: Matriz de confusión para un problema de clasificación binaria genérico. Si todas las muestras cayesen en la diagonal de la matriz, se obtendría un clasificador perfecto. Los elementos fuera de la diagonal indican el número de elementos incorrectamente clasificados. Este sencillo modelo muestra una preferencia sobre elementos verdaderamente clasificados como positivos, con un 77 % de efectividad, pasando a los elementos verdaderamente clasificados como negativos con un 75 % de efectividad. Imagen generada para este trabajo con datos sintéticos a manera de ilustración. GitHub ChJazhiel.

ning. Como se ha dicho, el modelo debe entrenarse con el **training set** para despues ser evaluado con instancias no vistas en el **test set**. La evaluación es importante para medir y entender la calidad del clasificador y para refinar parámetros en el proceso iterativo de descubrimiento de conocimiento en los datos.

3.2.1. Matriz de confusión

A menudo, los problemas de decisión binaria suelen ser comparados mediante su eficiencia a la hora de clasificar. La decisión hecha por el clasificador se representa entonces mediante una **matriz de confusión**. Ésta se usa como un indicador de las propiedades de una regla de clasificación o discriminante. Contiene el número de elementos correctamente o incorrectamente clasificados para cada clase. Es fácil observar el desempeño de un clasificador mediante la matriz de confusión.

La matriz de confusión se mide en cuatro categorías: Verdadero Positivo (TP), ejemplos correctamente clasificados como verdaderos, Verdadero Negativo (TN), ejemplos correctamente clasificados como negativos, Falso Positivo

	Predicción Negativa	Predicción Positiva
Ejemplos Negativos	A	B
Ejemplos Positivos	C	D

Tabla 3.2: Matriz de confusión.

(FP), ejemplos incorrectamente clasificados como positivos y Falso Negativo (FN), ejemplos incorrectamente clasificados como negativos, tal como se ilustra en la figura 3.6.

Por cada objeto en el test set, se compara la clase verdadera a la clase asignada por el clasificador previamente entrenado. De los valores de la tabla 3.2 se pueden obtener distintos valores:

- Exactitud: $(A + D)/(A + B + C + D)$
- Tasa de clasificación errónea: $(B + C)/(A + B + C + D)$
- Precisión: $D/(D + B)$
- Tasa de Verdaderos Positivos (Recall): $D/(C + D)$
- Tasa de Falsos Positivos: $B/(A + B)$
- Tasa de Verdaderos Negativos (Specificity): $A/(A + B)$
- Tasa de Falsos Negativos: $C/(C + D)$

3.2.2. Curvas ROC

La manera de evaluar algoritmos de decisión binarios es mediante el espacio llamado Receiver Operator Characteristics (ROC) (Fawcett, T., 2006 [56]). Una gráfica ROC es usada como visualizador de un clasificador basado en su desempeño. En este espacio se dibuja una curva, la cual muestra como el número de ejemplos correctamente clasificados como verdaderos varía respecto al número de ejemplos incorrectamente clasificados como negativos.

En el espacio ROC, se grafica la tasa de verdaderos positivos (True Positive Rate - TPR) contra la tasa de falsos positivos (False Positive Rate - FPR). FPR mide la fracción de ejemplos negativos incorrectamente clasificados como positivos, mientras que TPR mide la fracción de ejemplos positivos correctamente clasificados. Si se tiene una familia de curvas ROC, la parte convexa puede incluir puntos que se ubican más hacia la frontera noroeste del espacio ROC. Si una línea pasa por la parte convexa, entonces no hay

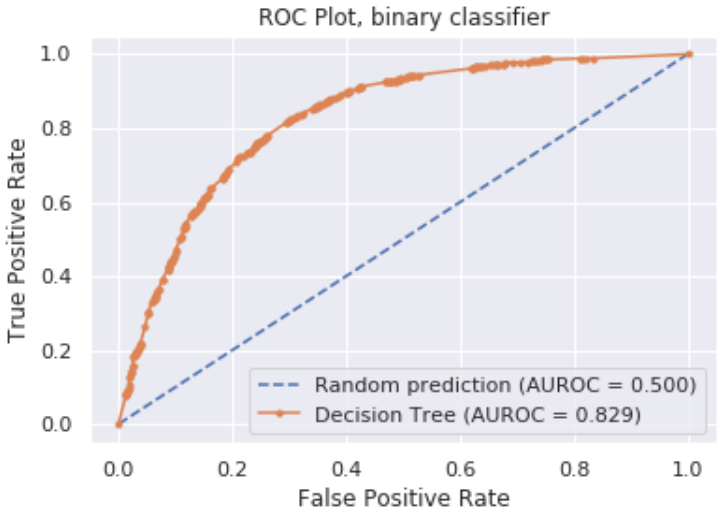


Figura 3.7: Curva ROC y valor del área bajo la curva (AUC) de un clasificador binario. Al ser una representación gráfica se puede evaluar el desempeño a varios umbrales de predicción. Para distintos clasificadores la forma de la curva ROC puede ser muy parecido, la manera más justa de compararlos es mediante el valor del área bajo la curva. GitHub ChJazhiel.

otra línea con la misma pendiente que pase por otro punto con una intersección TP más grande. De esta manera, el clasificador en ese punto es óptimo bajo cualquier suposición de distribución con esa pendiente (Rokach, L. & Maimon O. Z., 2008 [57]).

3.2.3. Área bajo la Curva (AUC)

A veces, evaluar un modelo probabilístico puede ser problemático. El utilizar medidas del tipo continuo como las curvas ROC puede llevar a un tipo de mal entendimiento de los resultados. En el caso de las curvas ROC, por ejemplo, para dos clasificadores puede haber un traslape en las curvas dentro del espacio ROC, de manera que se vuelve complicado hacer una determinación de qué modelo tuvo un mejor desempeño. Si no existe un modelo dominante no se puede determinar qué modelo es el mejor.

El área bajo la curva ROC, del inglés, Area Under Curve (AUC) es una métrica muy útil para observar el desempeño de un clasificador, ya que es independiente del criterio de decisión y probabilidades previas. Si para dos clasificadores, las curvas ROC se intersectan, el AUC es un promedio de la comparación entre ambos modelos. El área bajo la curva no depende de ningún desbalance de los datos de entrenamiento, es así que la comparación

del AUC de dos clasificadores es más justo e informativo que comparar sus tasas de clasificación errónea, por ejemplo. La manera de evaluar el desempeño de un algoritmo con esta métrica es con valores entre 0.5 y 1.0. Un valor de 0.5 es tan bueno como un clasificador aleatorio. En adelante se consideran rangos: 0.6–0.7, clasificación regular, 0.71–0.8, clasificación buena, 0.81–0.9, clasificación muy buena, 0.91–1.0 clasificación excelente y perfecta en caso de valer 1.0.

3.2.4. Generalización y sobreajuste

El sobreajuste es un fenómeno general y ocurre en todo tipo de algoritmos de aprendizaje, incluso cuando la función objetivo no es en absoluto aleatoria. El sobreajuste se vuelve más probable a medida que crece el espacio de hipótesis y el número de atributos de entrada, y es menos probable a medida que aumenta el número de ejemplos de entrenamiento.

Para los árboles de decisión, existe una técnica llamada "poda" o pruning del árbol de decisión, la cual combate el sobreajuste. El pruning funciona eliminando nodos que no son claramente relevantes. La pregunta es, ¿cómo se detecta que un nodo está probando un atributo irrelevante? Suponiendo que se está en un nodo que consta de p ejemplos positivos y n ejemplos negativos. Si el atributo es irrelevante, se esperaría que dividiese los ejemplos en subconjuntos, de manera que que cada uno tenga aproximadamente la misma proporción de ejemplos clasificados correctamente (positivos) como el conjunto completo, $p/(p+n)$, de esta forma la ganancia de información sería cercana a cero.

Ahora bien, ¿qué tan grande debe ser la ganancia de información para que se pueda dividir sobre un atributo en particular? Esta pregunta se responde utilizando una prueba estadística significativa. La prueba comienza suponiendo que no existe ninguna relación o ningún patrón subyacente (también conocido como hipótesis nula). Entonces, los datos reales se analizan para calular el grado en que se desvían de una ausencia perfecta de un patrón.

Si el grado de desviación es estadísticamente improbable, entonces eso se considera una buena evidencia de la presencia de un patrón significativo en los datos. Las probabilidades se calculan a partir de distribuciones estándar de la cantidad de desviación que se esperaría ver en un muestreo aleatorio. En este caso, la hipótesis nula es que el atributo es irrelevante y, por lo tanto, que la ganancia de información para una muestra infinitamente grande sería cero (Russel, S., Norvig, P., 2009 [47]).

En muchas ocasiones, el concepto de aprendizaje se puede conocer también como *tarea de clasificación*. En este caso, se busca una función que mapee todos los posibles ejemplos en un cojunto predefinido con etiquetas

de clase que no están limitadas a un conjunto Booleano. El objetivo de estos llamados clasificadores inductivos es definido por:

Dado un conjunto de entrenamiento S con atributos de entrada $A = a_1, a_2, ..., a_n$ y un atributo objetivo nominal de distribución fija desconocida D sobre el espacio de instancias, la meta es inducir un clasificador óptimo con error de generalización mínimo.

En otras palabras, dado un conjunto de entrenamiento con una cantidad finita de atributos y un conjunto de clases a determinar, encontrar el algoritmo que mejor generalice el modelo con un error mínimo.

Los árboles de decisión son muy útiles dado que se pueden utilizar como herramientas de exploración, sin embargo no trata de reemplazar a ningún otro método estadístico. Su empleo es muy popular en la minería de datos por su simplicidad y efectividad. Existen muchas características en los árboles de decisión que se necesitaría un libro completo para describirlas. Una de las más relevantes es el tamaño del árbol. La complejidad del algoritmo tiene impacto en su desempeño y eficiencia (Breiman L., et al., 1984 [58]), si un árbol es demasiado complejo, pueden tenerse problemas de sobreajuste, y si es muy sencillo, no puede generalizar nuevos datos. La manera de detener el crecimiento de un árbol se conoce como *pruning*. El algoritmo continúa su evolución hasta que se activa un criterio de detención. Los siguientes son los citerios más comunes de pruning.

1. Todos los casos en el conjunto de entrenamiento pertenecen a una clase.
2. La profundidad máxima del árbol se ha alcanzado.
3. El número de casos en un nodo terminal es menor que el mínimo número de casos en un nodo padre.
4. Si el nodo se divide, el número de casos en uno o más hijos sería menor que el número mínimo de casos para nodos hijos.
5. El criterio para la mejor partición no es mayor que cierto valor umbral.

Esto también ocurre para los clasificadores tipo Random Forest, aunque de menor manera debido al bagging y bootstrap. De igual manera deben recortarse para poder obtener una clasificación más óptima.

3.2.5. Curva de aprendizaje

Debido a que existen muchas métricas para evaluar un clasificador, se puede crear a partir de estas una curva de aprendizaje. Esta curva es un

gráfico del rendimiento del aprendizaje del modelo sobre la experiencia o el tiempo.

Las curvas de aprendizaje son una herramienta de diagnóstico ampliamente utilizada en machine learning para algoritmos que aprenden de un training set de forma incremental. El modelo se puede evaluar en el conjunto de datos de entrenamiento y en un conjunto de datos de validación después de cada actualización durante el entrenamiento y se pueden observar gráficos del rendimiento medido para mostrar las curvas de aprendizaje.

La revisión de las curvas de aprendizaje de los modelos durante el entrenamiento se puede usar para diagnosticar problemas de aprendizaje, como un modelo de ajuste o sobreajuste, así como si los conjuntos de datos de entrenamiento y validación son adecuadamente representativos, como se observa en la figura 3.8.

La evaluación en el conjunto de validación ofrece una idea de que tan capaz es el modelo de "generalizar". En el espacio de la curva de aprendizaje se tienen normalmente dos curvas:

- Train Learning Curve: curva de aprendizaje calculada del training set que ofrece una idea de cuan bien está aprendiendo el modelo.
- Validation Learning Curve: curva de aprendizaje calculada de un conjunto de validación que ofrece una idea de que tan bueno es el modelo generalizando.

Debido a que las métricas para evaluar un algoritmo son variadas, la manera sencilla de crear una curva de aprendizaje es mediante la exactitud, aunque también se puede crear mediante el error. Para garantizar un aprendizaje óptimo, el conjunto de datos se separa subconjuntos de muestras llamado k**-Fold Cross-validation**. El procedimiento tiene un parámetro k que se refiere al número de grupos en los cuales se dividirá el conjunto de datos. Es un método simple de entender y ayuda a que el modelo tenga poca variación y bias.

El procedimiento general es el siguiente:

1. Revolver el dataset aleatoriamente.
2. Dividir el dataset en k grupos.
3. Para cada grupo único:
   a) Tomar un grupo como test set.
   b) Tomar el resto de grupos como training set.

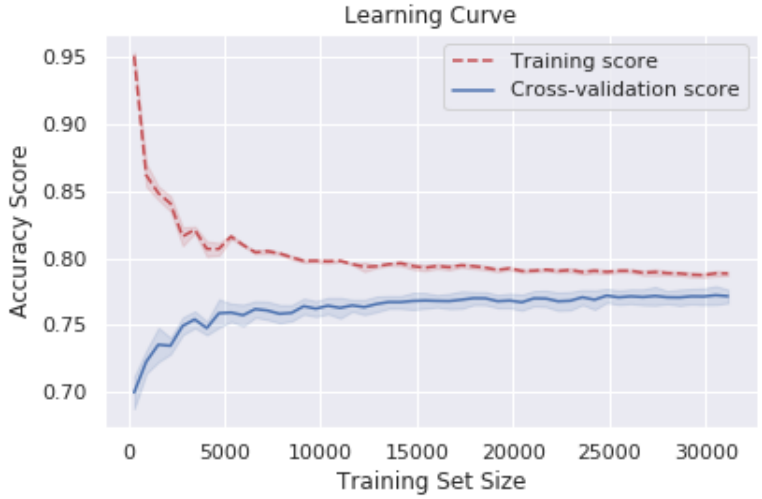


Figura 3.8: Curvas de aprendizaje con métrica de exactitud de un clasificador binario. Se dice que el algoritmo está aprendiendo cuando la curva del conjunto de validación está cercana al conjunto de entrenamiento. En esta gráfica, se observa que el modelo no requiere cambiar sus hiperparámetros ya que la curva de aprendizaje del conjunto de testeo está bastante cercana a la curva de entrenamiento y no parece tender a estar sobre ajustado. GitHub ChJazhiel.

   c) Ajustar el modelo con el training set y evaluarlo con el test set.

   d) Conservar la puntuación de la evaluación y descartar el modelo.

4. Recabar las habilidades del modelo usando la muestra de puntajes de evaluación del modelo.

Este enfoque implica dividir aleatoriamente el conjunto de observaciones en k grupos, de aproximadamente el mismo tamaño. El primer grupo se trata como un conjunto de validación y el método se ajusta a los $k-1$ grupos restantes (James, G. et al., 2014 [59]).

De manera gráfica, se puede entender con la figura 3.9. De aquí se observa claramente como este método mezcla y divide el conjunto de manera aleatoria, de modo que un grupo pequeño sea el test set y el resto de los datos el training set. Este proceso se efectúa de manera recursiva para evitar algún tipo de sesgo o variación del modelo.

Este capítulo tuvo como objetivo introducir definiciones, modelos y métricas propias de la inteligencia artificial y de machine learning, dado que existen una cantidad enorme de algoritmos, tanto supervisados como no supervisados. Dado que este trabajo está enfocado en algoritmos supervisados, fue necesario conocer su funcionamiento, la base lógica por la cual operan y

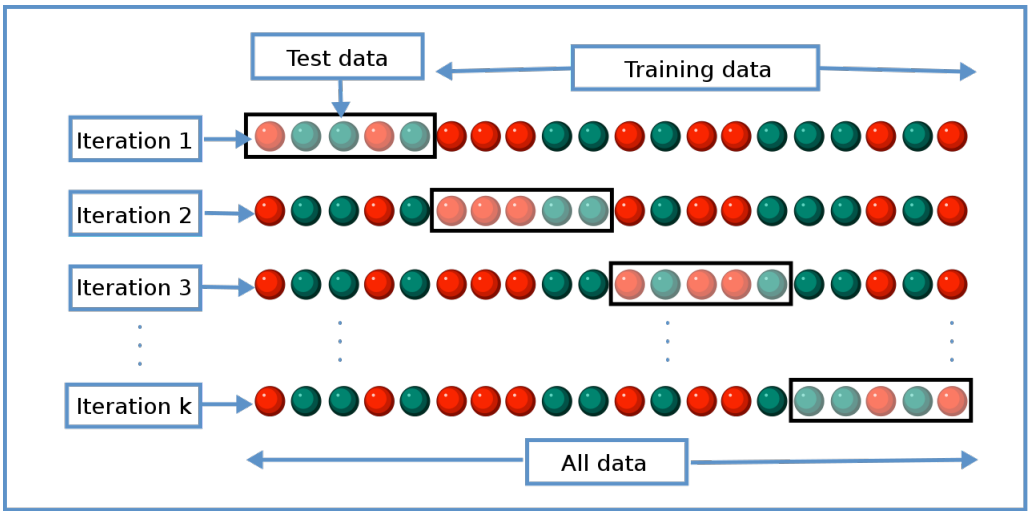


Figura 3.9: Visualización de k-fold cross validation. El conjunto de datos se mezcla de manera aleatoria y se escoge un grupo de testeo, dejando el resto de los datos como entrenamiento. Las iteraciones sirven para realizar este método de manera definida con tal de minimizar la variación y el bias del modelo. Imagen de Wikipedia.

eventualmente llevar a cabo una evaluación para validar resultados. Hay un sinfín de aplicaciones de estos algoritmos, por ejemplo para predicciones, forecasting y discriminación de datos, en el siguiente capítulo se presentará uno de los tantos enfoques que se han desarrollado en cosmología y en particular para la cosmología numérica.

Capítulo 4

La Cosmología Numérica como un problema de clasificación

En años recientes los recursos computacionales y el avance tecnológico han sido una mancuerna vital para crear simulaciones de alta resolución que han ido tan lejos como la época del CMB, por ejemplo ($z \sim 1100$). Es bien conocido por la comunidad numérica que las simulaciones a gran escala, es decir las que van más allá de 100 Mpc se llevan a cabo con aglomeramiento gravitacional lineal, mediante teoría de perturbaciones, mientras que en escalas más pequeñas (entre 10 kpc y 1 Mpc) el aglomeramiento gravitacional ya no es lineal, si no que tiene mayor orden de magnitud y es necesario considerar otro tipo de condiciones iniciales para las perturbaciones. A escalas pequeñas se tiene el agregado de poder incluir más dinámica, como efectos de partículas bariónicas, gas, incluso procesos químicos, efectos radiativos y otros fenómenos.

Las galaxias se aglomeran en los halos de materia oscura, de modo que ésta última es un elemento clave dentro de una simulación numérica. Los halos de materia oscura se forman gracias a las perturbaciones del campo de densidad del Universo temprano. En un Universo ΛCDM la formación de estructura inicia con perturbaciones lagrangianas de primer orden al campo de densidad que crecen de manera lineal hasta llegar a una densidad crítica, después de la cual dejan de expandirse y colapsan para formar halos de materia oscura ligados, un modelo también conocido como top hat (Navarro, J. F., Frenk, C. S, White, S. D. M., 1994 [60]). Esta evolución obedece una estructura jerárquica, es decir los halos menos masivos se forman primero, y los más masivos son los últimos en formarse.

Además, existen aproximaciones analíticas que asumen que el colapso de de materia oscura ocurre cuando el contraste de densidad del campo sobrepasa un valor umbral. Por ejemplo, la cantidad de halos que se forman en

el Universo o la Halo Mass Function, son solo algunos parámetros posibles a obtener de una simulación numérica. La Halo Mass Function es una de las cantidades usadas y evaluadas con mayor frecuencia, dado que a parte de ser una cantidad comparable con modelos analíticos y simulaciones, ésta debe poder ajustarse directamente de parámetros observables (Planck Collaboration, 2018 [61]).

Luego de haber realizado la simulación, detallada en la sección 2.3, se obtiene la relación de halos de materia oscura formados en la época $z = 0$ con ROCKSTAR, esto permite identificar los halos de materia oscura llamados "padres" o hosts y a su vez es capaz de identificar subestructuras o subhalos del mismo host. Se selecciona un umbral de masa de materia oscura para identificar halos, de esta manera es posible identificar las partículas que terminan en un halo de materia oscura dado el umbral de masa, así como las que no terminan en un halo, es decir que son partículas libres o pertenecen a halos de menor masa. Como puede deducirse, esto conlleva a tratar un proceso de evolución de materia oscura a un problema de clasificación.

4.1. Procedimiento

Se escoge una simulación cosmológica de un Universo ΛCDM realizada con el código cosmológico GADGET-2, con los parámetros $\Omega_m = 0.268$, $\Omega_\Lambda = 0.683$, $\Omega_b = 0.049$, $h = 0.7$. La simulación tiene un softening gravitacional de $\epsilon = 0.89$ kpc y se hace evolucionar un total de 4096×12^3 partículas, cada una con masa de $1.3{\times}10^9$ $M_\odot$ en una caja de longitud comóvil $L = 50h^{-1}$ Mpc desde $z = 23$ hasta $z = 0$. Los halos (tanto host como subhalos) se identifican con ROCKSTAR. La clase correspondiente a las etiquetas [*Not in Halo*, *In Halo*] se escogió con el umbral de masa de $M \geq 1.2 \times 10^{12}\, M_\odot$, de manera que la clase *in Halo* estará en halos que superen este umbral mientras que las partículas *Not in Halo* están en halos con masa menor a dicho umbral o bien que no estén ligadas a ningún halo. En la figura 4.2 se ilustra nuevamente el espectro de potencias obtenido para la simulación descrita en esta sección. El snapshot final contabilizó un total de 4000 halos de materia oscura cuyas masas entran dentro del rango ($10^{11} \leq M/M_\odot \leq 10^{14}$).

4.1.1. Asignación de etiquetas

Cada partícula tendrá asociada un vector de 10 componentes y una etiqueta: 1 para la clase *In halo*, 0 para la clase *Not in halo*. Las propiedades de las partículas se extraen de las condiciones iniciales ($z = 23$) y se usan como datos de entrada para los métodos de clasificación de árbol de decisión

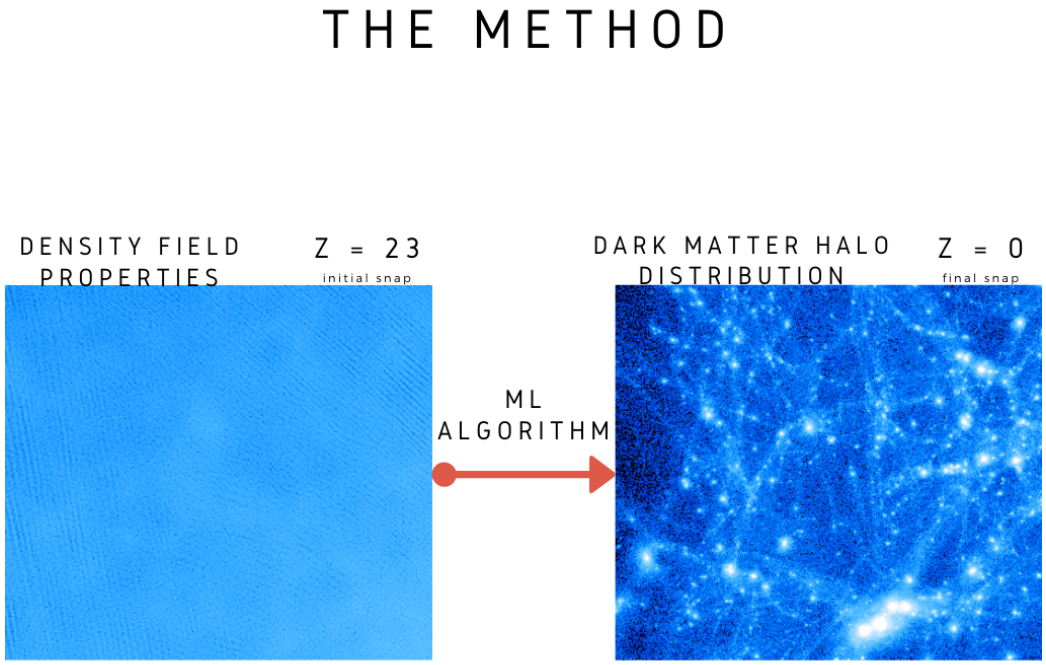


Figura 4.1: Descripción gráfica del método para seleccionar las propiedades de las condiciones iniciales del campo de densidad inicial que eventualmente formarán la estructura en la simulación. Las propiedades se extraen de la vecindad local alrededor de cada partícula de materia oscura que determina la clasificación final *Not in halo*, *In Halo*. Imagen tomada de la presentación "Decision Trees Applied to Numerical Cosmology" de Jazhiel Chacón.

y random forest. Las componentes son las densidades de masa centrada en cada partícula ligadas a la densidad local del redshift inicial. Se escogió un subconjunto de todas las partiículas dentro de la simulación con su respectiva etiqueta. El entrenamiento se realizó con un split 80/20 del subconjunto (80 % entrenamiento y 20 % prueba/validación).

Los algoritmos de machine learning, sobre todo los supervisados, requieren el uso de características de una base de datos estructurada, en este caso se tiene un conjunto de datos estructurado con características o features extraídos del campo de densidad. Esta asignación proviene de trabajos analíticos relacionados a la función de masas de halos (HMF) de Press-Schechter (Press, W. H., Schechter, P. 1974 [22]). Esta función predice la densidad de número de halos de materia oscura dependiente de su masa y del campo de densidad. La densidad formará un halo de cierta masa M a un redshift z si excede un valor crítico $\delta_c(z)$, estos valores serán llamados sobredensidades a

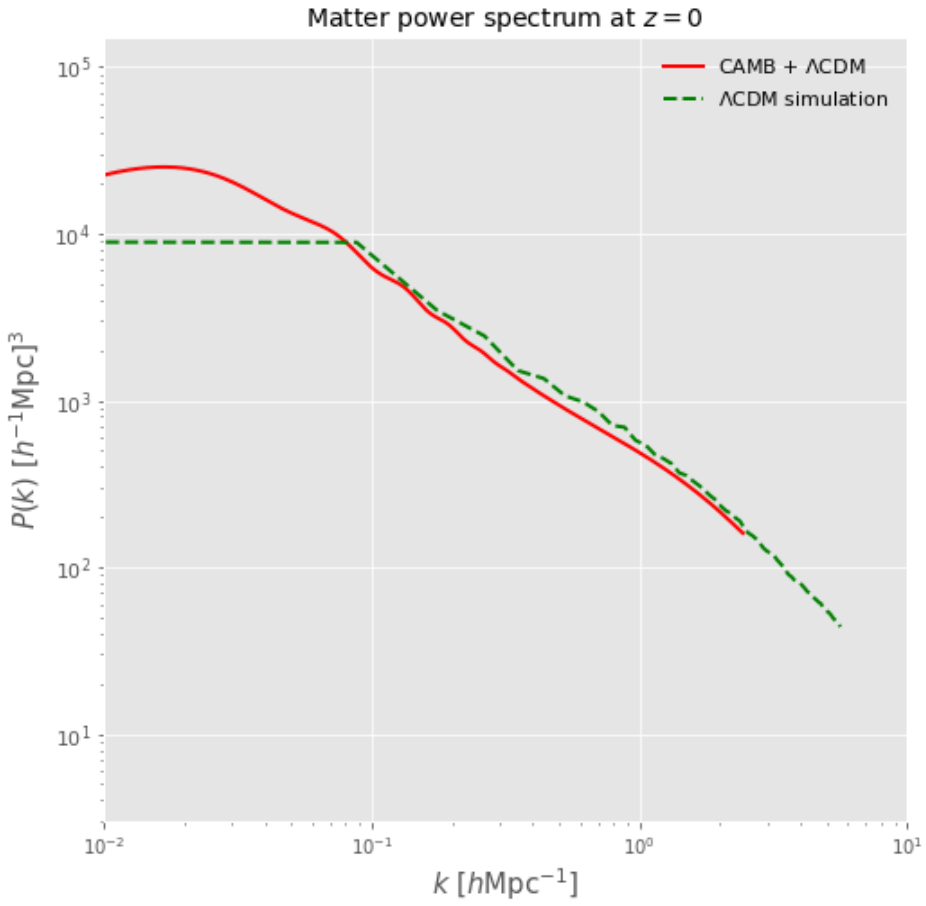


Figura 4.2: Espectro de potencias reconstruido para la simulación de ΛCDM comparado con el espectro obtenido con el Código CAMB en $z = 0$. Las curvas representan la distribución de materia en la época actual. La línea constante en escalas grandes ($k \ll 0$) de la simulación es debido al tamaño de la caja. A escalas pequeñas ($k > 0$ se observa la gran similitud entre ambas gráficas. Gráfica generada de los resultados de la simulación de la tabla 2.2

un determinado redshift z.

La idea principal es que la materia de un halo de materia oscura estará encerrada en una región esférica densa, donde el contraste a la densidad estará dada por la relación

$$\delta(\mathbf{x}) = \frac{\rho(\mathbf{x}) - \bar{\rho}}{\bar{\rho}}, \tag{4.1}$$

donde $\bar{\rho}$ es la densidad media de materia en el Universo. Para una esfera de radio R (Dodelson, S., 2003 [62]) se entiende que la sobredensidad es

$$\delta(\mathbf{x}, R) \equiv \int d^3x' \delta(\mathbf{x}') W_R(\mathbf{x} - \mathbf{x}'). \tag{4.2}$$

En la ecuación (4.2), W_R es una función ventana del modelo *top hat*, dada por

$$W_R = \begin{cases} \frac{3}{4\pi R^3} & \text{si} |\mathbf{x}| \leq R \\ 0 & \text{si} |\mathbf{x}| > R \end{cases} \tag{4.3}$$

De manera que una función ventana con un radio R corresponde a una escala de masa $M = \bar{\rho} V(R)$. El valor esperado de la sobredensidad (4.2) es el término de normalización del espectro de potencias σ_R

$$\sigma_R^2 = \langle \delta^2(\mathbf{x}, R) \rangle. \tag{4.4}$$

La elección de las características de los datos estructurados para los algoritmos de machine learning residen en contrastes de la densidad calculada con la función ventana del tipo top hat que se deriva de una escala de masa en el radio R, M_R centrada en la posición de una partícula, desde las condiciones iniciales y el redshift inicial $z = 23$. El resultado es una cantidad de 10 sobredensidades, $\delta_1, ...\delta_{10}$ asociadas a su respectiva clase o etiqueta.

4.1.2. Entrenamiento de algoritmos

Los algoritmos utilizados para esta sección fueron árboles de decisión y random forest, incluidos en el paquete de machine learning **Scikit-Learn** (Pedregosa, F. and et al. 2011 [63]) de Python. La cantidad inicial de partículas fue de 50,000 seleccionadas de manera aleatoria, pero se realizó un procesamiento de manera que las etiquetas [*Not in Halo*, *In Halo*] estuvieran niveladas, es decir que existiese la misma cantidad de etiquetas 0 y 1, respectivamente. Luego de esta preselección se reduce el número total de partículas a 28,600. Los algoritmos fueron probados para ambas cantidades y no se observó una reducción en el desempeño al reducir la cantidad de partículas. El conjunto de datos se selecciona de manera aleatoria de manera que no exista nungún tipo de sesgo al efectuar la clasificación. El conjunto de entrenamiento, como se mencionó es del 80 % del total de partículas, de manera que 22,880 partículas sirvieron como conjunto de entrenamiento, y el conjunto de validación fueron las 5720 partículas restantes.

El árbol de decisión y el random forest fueron refinados realizando pruebas en una malla de hiperparámetros. Más específicamente, la malla de hiperparámetros tenía elementos como la profundidad máxima del árbol, el

criterio de split, el número máximo de partículas por nodo, el número mínimo de partículas para afectuar un split, y en el caso de random forest, el número total de estimadores. Empezando desde el número de estimadores en random forest en 100, aumentando en 100, la profundidad del árbol iniciando en 1 y llegando a 20, el número mínimo de partículas en 50, hasta 200, encontrando así los valores óptimos de los mismos a fin de evitar pruebas a ciegas o blind tests. Estas pruebas se hicieron en un procesador ya que no era necesario hacerlo de manera masiva y llevaron cerca de 8 horas. Los hiperparámetros óptimos se destacan en la tabla 4.1, coincidiendo en casi los mismos valores exceptuando el número de estimadores, exclusivo de random forest. Los códigos entrenados de esta forma predicen la etiqueta final de las partículas en el conjunto de prueba, que se compara con las etiquetas reales a manera de obtener el desempeño de cada algoritmo. La manera de evaluarlos fue efectuada bajo dos pruebas, la curva ROC junto con el valor debajo de la curva ROC y la curva de aprendizaje.

Tabla 4.1: Hiperparámetros óptimos encontrados para algoritmos

Descripción	Símbolo	Valor
Criterio de decisión	`criterion`	`'entropy'`
Profundidad máxima	`max_depth`	8
Balance de clases	`class_weight`	`"balanced"`
Número de estimadores	`n_estimators`	2000
Número mínimo de partículas	`n_particles`	200

4.2. Clasificación de partículas

Debido a la distribución de probabilidad obtenida para cada rango de sobredensidad, no es necesario hacer un preprocesamiento extensivo (véase figura 4.3). Esta figura es muy importante ya que describe la distribución de clases (*Not in Halo* ,`label = 0`, *In Halo*, `label = 1`) dependiendo del contraste de densidad δ_i. En esa figura se describen δ_5, δ_6, δ_7, correspondientes a valores de masas $1.2\times10^{12}\mathrm{M}_\odot$, $2\times10^{12}\mathrm{M}_\odot$, $1.1\times10^{13}\mathrm{M}_\odot$, respectivamente, justo en el límite estipulado para efectuar una decisión ($1.2\times10^{12}\mathrm{M}_\odot$). Los algoritmos de clasificación no necesitan de un reescalamiento de características puesto que realizan decisiones mediante la ganancia de información, a diferencia de otros métodos donde una diferencia sutil, por ejemplo, la misma distancia (5 km y 5000 m) puede repercutir en el desempeño propio del algoritmo. Los resultados son una medida de probabilidad de cada clase para todas las partículas. De una manera similar a lo que se hace con regresión

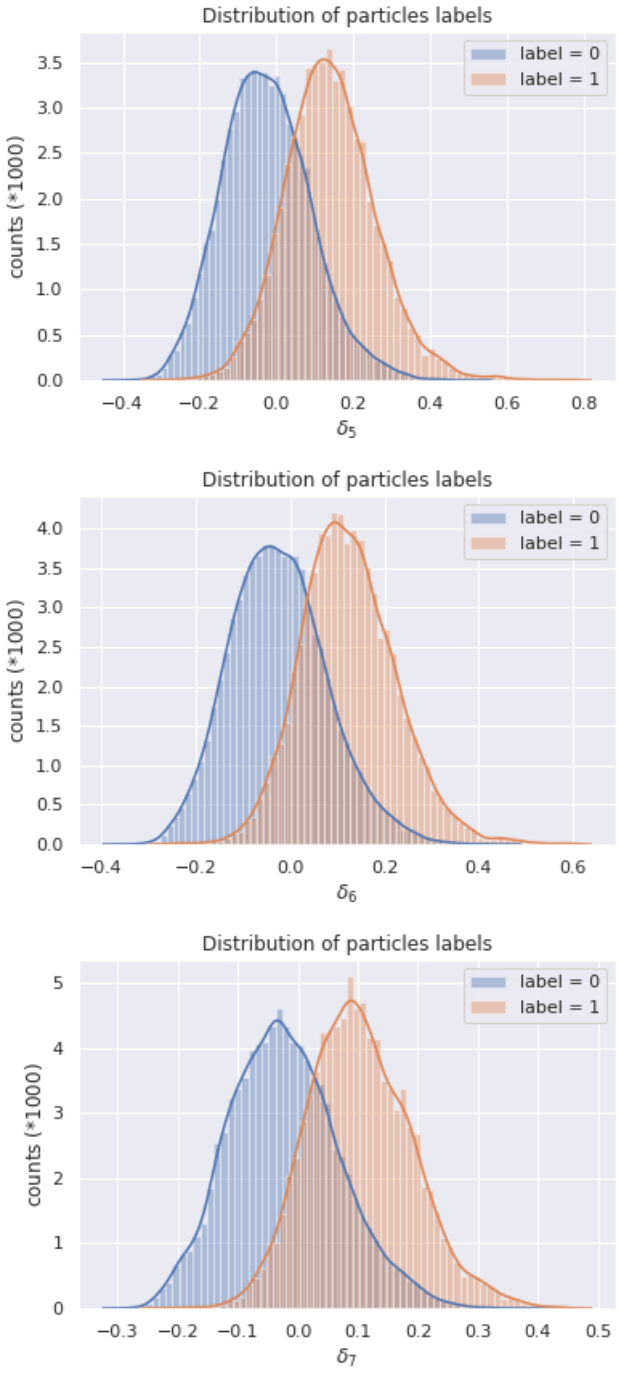


Figura 4.3: Distribución de probabilidad de pertenenencia de clase para 3 sobredensidades características obtenida en el preprocesamiento de datos. La forma de la distribución sugiere 2 cosas: 1) No se necesita hacer un reescalamiento de datos, pues la semejanza con una curva Gaussiana es evidente. 2) El uso de la métrica de curva ROC es suficiente debido a la distinción de clases en ese rango de valores de sobredensidad.

logística, el resultado de pertenecer a una clase u otra es determinado por un valor umbral de probabilidad.

Luego de tomar esto en cuenta, el desempeño de los algoritmos es cuantificado. Como se ha mencionado en el capítulo anterior, un clasificador perfecto consistiría de valores verdaderos positivos y verdaderos negativos en su matriz de confusión. La tasa de verdaderos positivos (TPR) y la tasa de falsos positivos (FPR) son las cantidades características de una curva ROC. No se utilizó otra métrica como la de Precision-Recall debido al mismo preprocesamiento de los datos. Las curvas ROC deben usarse cuando se sabe que el número de observaciones de cada clase es relativamente igual. La curva de Precision-Recall es más utilizada cuando se sabe que existe un imbalance de clases en los datos.

La cantidad de partículas correctamente clasificadas (TPR) y la cantidad de partículas incorrectamente clasificadas como verdaderas (FPR). Es lo que se muestra en la figura 4.4. Las pruebas realizadas para el árbol de decisión dieron un valor de exactitud de 0.77 ± 0.01, con un valor de AUC $= 0.846$. Para el random forest, la exactitud fue de 0.78 ± 0.01 y valor de AUC $= 0.866$. Como se mencionó en la sección 3.1.4, random forest utiliza un ensamble de árboles de decisión, junto con reemplazo de valores para obtener una mejor clasificación y evitar el sobreajuste. La mejora del 2 % del random forest sobre el árbol de decisión hace evidente este hecho.

Se observa en la figura 4.4 que la tasa de verdaderos positivos decrece a medida que la tasa de falsos positivos también lo hace. De manera que los algoritmos de machine learning han sido capaces de predecir en buena manera si una partícula terminaría en un halo o no, dependiendo de la sobredensidad del campo de densidad de materia oscura obtenido en las condiciones iniciales. Como se ha discutido anteriormente, las curvas ROC evalúan la capacidad de un algoritmo de clasificar correctamente clases, de manera que evite "copiar" atributos y aunado a esto también se debe evitar que clasifique de manera aleatoria. Por lo tanto, el desempeño mostrado en la figura 4.4 es indicativo de que efectivamente, el aprendizaje fue exitoso.

También como parte de la evaluación de algoritmos se describen las curvas de aprendizaje del árbol de decisión y random forest en la figura 4.5 . La figura superior corresponde al árbol de decisión, mientras que la figura inferior hace lo respectivo para representar al random forest. Se observa que ambos métodos ajustan bien su desempeño conforme el número de elementos de prueba y validación aumenta, llegando a un valor casi paralelo al reportado por el conjunto de entrenamiento. Como las curvas de entrenamiento no aumentan ni las curvas de validación decaen después de efectuar las pruebas con *cross-validation* es posible concluir que ambos métodos no están teniendo ningún tipo de sobreajuste, esto es, tienen un buen ajuste.

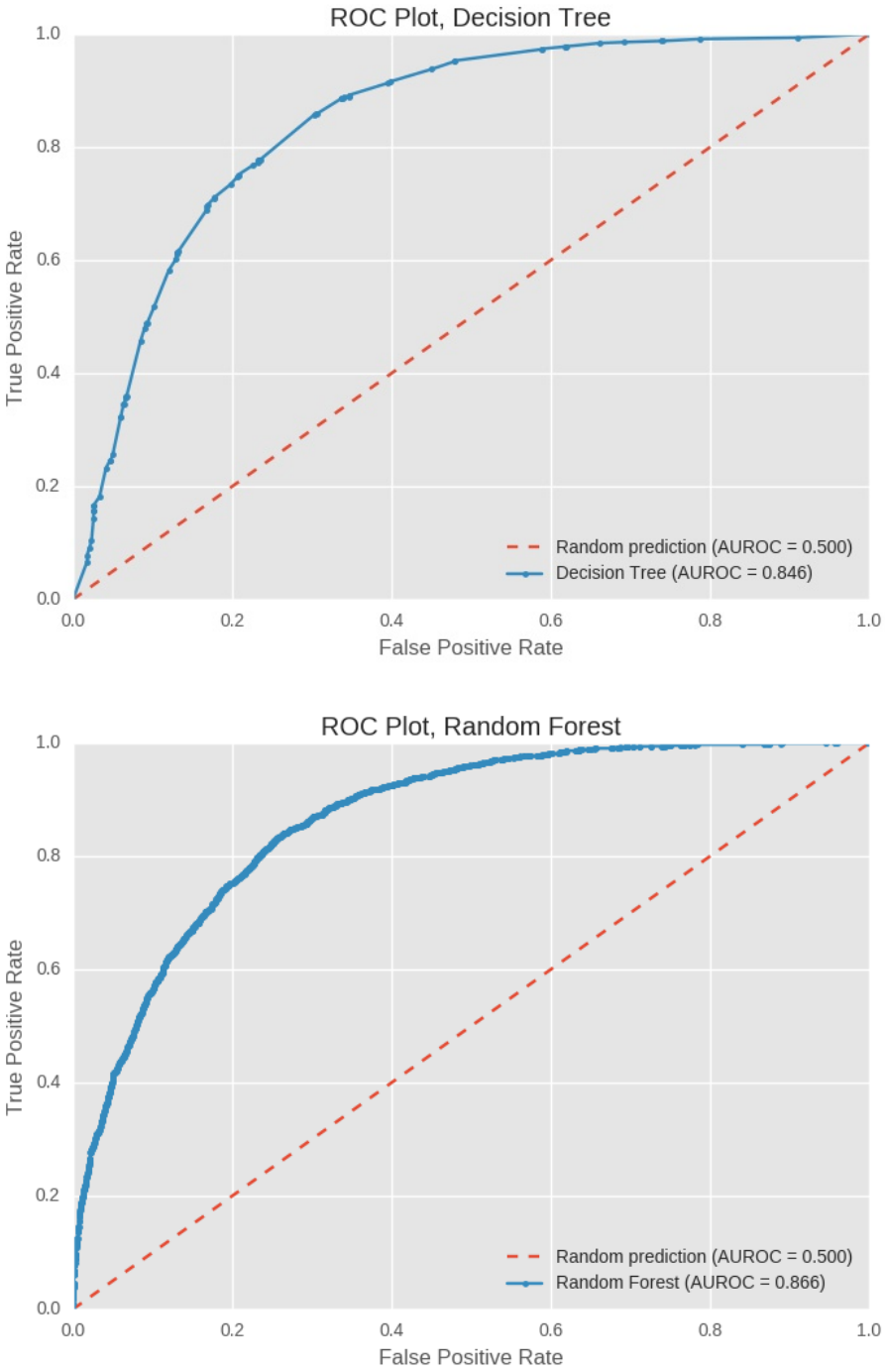


Figura 4.4: Curvas ROC de árbol de decisión y random forest entrenados en la simulación de GADGET. El desempeño es notable dado que ambos tienen un valor de AUC ≥ 0.8, destacando la mejoría que tiene random forest sobre el árbol de decisión.

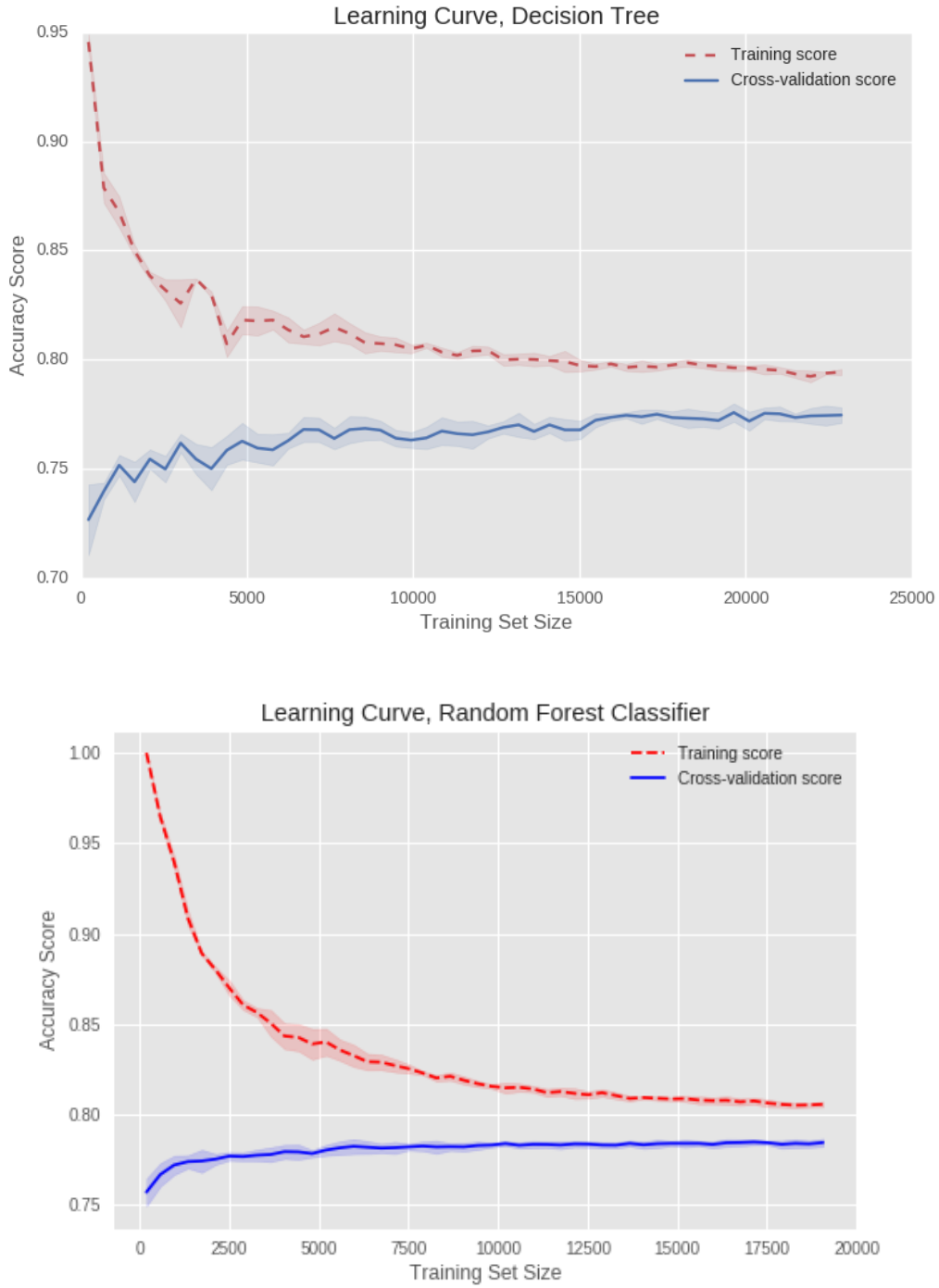


Figura 4.5: Curvas de aprendizaje de los algoritmos de árbol de decisión y random forest. La curva de entrenamiento inicia muy alta porque tiene pocas muestras sobre las cuales hacer una predicción. Conforme las muestras aumentan, también aumenta la curva de aprendizaje del conjunto de validación, mostrando que no existe sobreajuste ni desajuste. Destaca que la curva de aprendizaje del random forest tenga menor varianza, dado que la baja correlación entre características evita una cambio en este valor.

4.2.1. Importancia de atributos

El desempeño de los algoritmos depende de si las características que utiliza como datos de entrada son relevantes o no para separar distintas clases. De esta manera, la característica ideal sería aquella que separe clases en dos conjuntos puros. En cambio, una característica irrelevante no distingue entre clases, por tanto, no son de ayuda para los clasificadores. Esto es muy útil para un clasificador ya que le ayuda a identificar atributos determinantes para realizar una decisión. En ciencia de datos y minería de datos esto es una herramienta muy útil. Dependiendo de la importancia de atributos los problemas de clasificación mencionados se debe llegar a una clasificación ideal.

Es posible determinar que atributos contienen la mayor cantidad de información para poder determinar qué partículas terminan en halos de determinada masa. Los atributos separan clases en el entrenamiento de los algoritmos. Para medir la relevancia de las variables de entrada de los algoritmos se usa la métrica *feature importances* (Louppe, G., 2014 [64]). La manera de aplicarlo a un árbol individual es mediante la diferencia en entropía después de una división, por lo que la importancia de un atributo X_m para predecir una variable Y es una suma de la reducción de impureza en todos los nodos t donde el atributo X_m se usa para efectuar la división

$$\mathrm{Imp}(X_m) = \sum_{t \in \phi} \Delta I(s_t^m, t), \tag{4.5}$$

donde s_t^m es el mejor split del nodo t en el árbol ϕ para el atributo m y ΔI es la diferencia en entropía. Mientras que para un random forest se determina de manera similar, pero agregando una suma ponderada de los atributos promediada en el número total de árboles dentro del forest

$$\mathrm{Imp}(X_m) = \frac{1}{N_T} \sum_{T} \sum_{t \in T} p(t) \Delta I(s_t, t), \tag{4.6}$$

donde N_T es el número de árboles, $p(t)$ es la fracción de muestras que llegan al nodo t, (siendo en este caso la fracción de partículas que llegan al nodo t) e I es la ganancia de información, es decir el cambio de entropía después de realizar la división. Los atributos importantes dentro de los algoritmos de decisión serán determinados de acuerdo a estos criterios y distinguen entre clases [*Not in Halo, in Halo*]. La importancia relativa de los atributos para cada algoritmo se observa en la figura 4.6. Los contrastes de densidad δ_5, δ_6, δ_7, corresponden a valores de masas $1.2 \times 10^{12}\mathrm{M}_\odot$, $2 \times 10^{12}\mathrm{M}_\odot$, $1.1 \times 10^{13}\mathrm{M}_\odot$, respectivamente. Justo en el límite estipulado para efectuar una decisión

$(1.2 \times 10^{12} \mathrm{M}_{\odot})$. La importancia de estos atributos es determinante al efectuar la decisión de clases.

4.3. Prueba en nuevas condiciones iniciales

El entrenamiento y pruebas de los algoritmos de clasificación han sido puramente en simulaciones de N-cuerpos. La ventaja de hacer esto es que se puede llevar a cabo una evaluación en un conjunto independiente de condiciones iniciales y probar la efectividad de predicción. Para este fin, se crearon 4 nuevos conjuntos de condiciones iniciales, muy similares a los listados en la tabla 3.1. En 3 de ellos se cambió la "seed" de generación, que es en esencia un creador pseudoaleatorio de números que se transmiten a las posiciones de las partículas en las condiciones iniciales. En otra simulación se cambió también el parámetro de longitud de suavizado gravitacional ϵ. En la primer simulación, ésta tenía un valor de $\epsilon_1 = 0.89$ kpc. La nueva simulación cambió este suavizado, aumentándolo a $\epsilon_2 = 1$ kpc. Recordando que esta es la distancia mínima que pueden estar dos partículas de materia oscura juntas en la simulación, se espera que la distribución de materia cambie, lo cual puede corroborarse con el espectro de potencias de masa, observado en la figura 4.7. Cabe destacar que de las 4 nuevas realizaciones, la única que tenía un archivo final de las posiciones de las partículas en $z = 0$ fue la del cambio del suavizado gravitacional, es por esto que las otras tres realizaciones restantes no figura su espectro de potencias en la figura 4.7. Se extrajeron nuevamente las propiedades alrededor de las partículas en el campo de densidad de materia oscura y se llevó a cabo una nueva evaluación del desempeño del árbol de decisión y random forest.

La figura 4.8 muestra la curva ROC comparativa de los dos algoritmos en la realización con nuevo suavizado gravitacional ϵ, al entrenarlos y probarlos con los datos de la simulación inicial, así como al hacer la prueba con las nuevas condiciones iniciales, sin llevar a cabo la ejecución computacional. La parte superior muestra el desempeño del árbol de decisión en el conjunto de entrenamiento y testeo anterior y la predicción para las nuevas condiciones iniciales. La parte inferior muestra lo mismo para el random forest. Los algoritmos de machine learning producen curvas ROC consistentes para el nuevo conjunto de condiciones iniciales. El área bajo la curva ROC en ambos casos bajó $\sim 2\,\%$ dado que la formación de estructura fue menor.

Por otra parte, para las realizaciones del cambio de "seed" en las condiciones iniciales se tuvo un desempeño similar al descrito en el párrafo anterior, se tenía la realización de la simulación tipo ΛCDM como conjunto de entrenamiento y se probó el desempeño de los algoritmos en su capacidad predictiva,

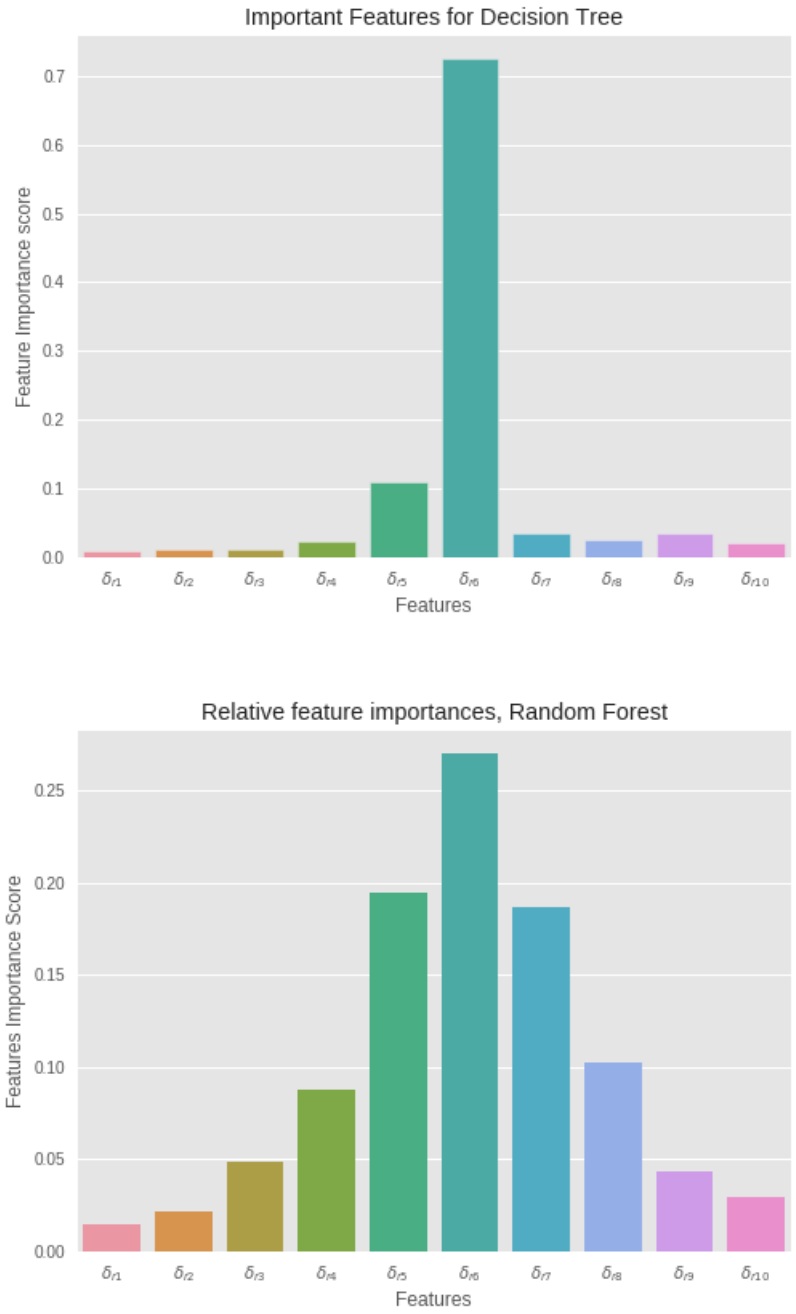


Figura 4.6: Importancia relativa de las características de los algoritmos. La similitud entre ambos es muestra de como hay influencia en la decisión al solo utilizar un árbol. Es notable que la importancia relativa de atributos obtenida para random forest se asemeje más a una distribución normal, haciendo evidente que el uso de selección aleatoria de características reduzca la importancia del contraste de densidad δ_6, correspondiente a un valor de masa del halo de $1.2 \times 10^{12} M_{\odot}$.

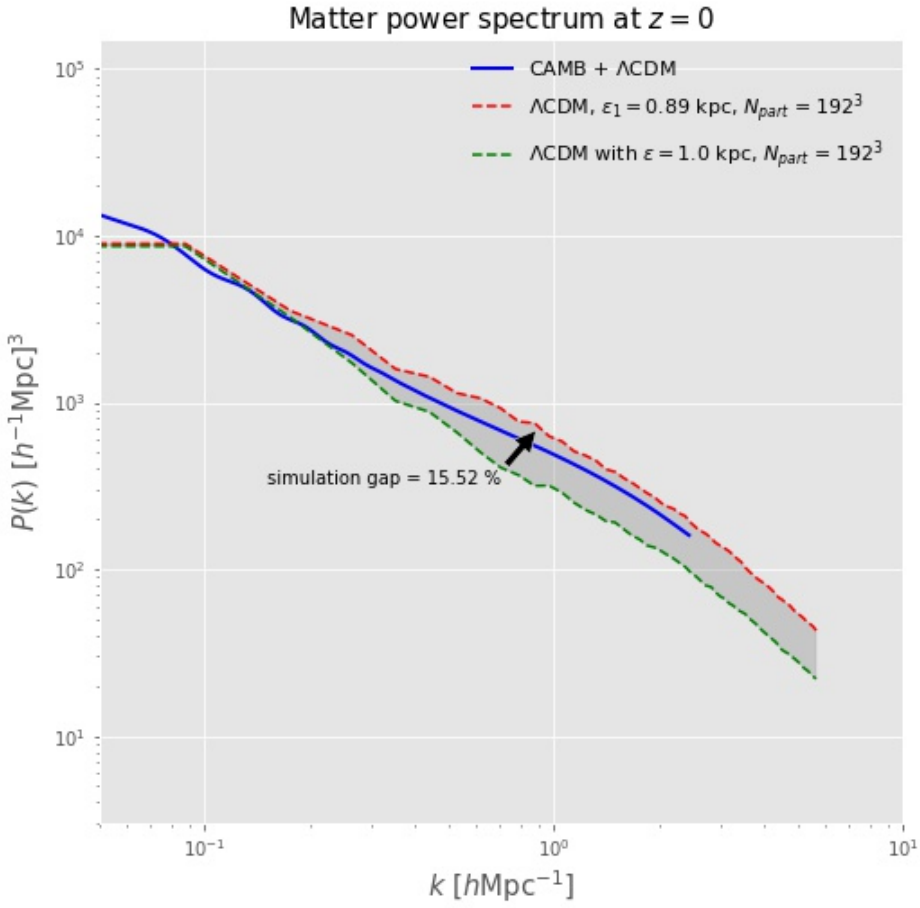
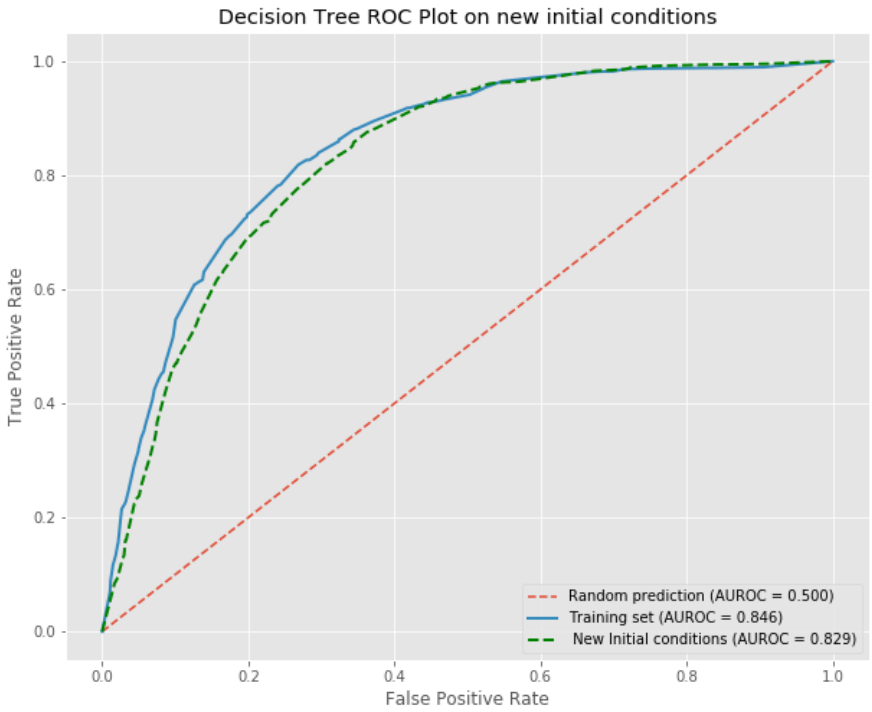


Figura 4.7: Espectro de potencias de las nuevas condiciones iniciales ($\epsilon_2 = 1.0$ kpc) comparado con la simulación anterior y el espectro obtenido con CAMB. La diferencia entre ambas simulaciones se señala en la figura y es de aproximadamente 15 %. El espectro de potencias se obtuvo de la misma manera que en realizaciones previas. Es evidente que la distribución de materia para las nuevas condiciones es diferente, dado que hay menos formación de estructura.

notando que a pesar de no haber ejecutado la simulación completa, éstos son capaces de identificar de buena manera la clasificación de partículas de materia oscura que cayesen o no dentro de un halo de materia oscura dado un valor umbral. En la figura 4.9 se observa el desempeño del árbol de decisión en la parte superior, y del random forest en al parte inferior. Como puede observarse, ambos algoritmos tienen un desempeño competente con su

contraparte del entrenamiento de la simulación de ΛCDM. Las condiciones iniciales de estas nuevas realizaciones no cambiaron en absoluto más que en el generador del seed pseudoaleatorio, de manera que el poder predictivo de los algoritmos se hace aún más evidente con esta figura.

Las etiquetas predichas por los algoritmos de machine learning se calculan de las propiedades de densidad de las condiciones iniciales. En las simulaciones, los algoritmos son capaces de predecir el resultado final de clasificación con una exactitud bastante buena. Al llevar a cabo una nueva prueba para las condiciones iniciales diferentes, sin llevar a cabo la simulación completa, ambos métodos fueron capaces de predecir la etiqueta final con exactitud, aunque menor, con un valor no menos despreciable. Por lo tanto, es posible concluir que la asignación de etiquetas aprendida por los algoritmos en una simulación puede generalizarse a diferentes simulaciones, usando los mismos o cambiando algunos parámetros cosmológicos, sin la necesidad de volver a entrenar dichos algoritmos.



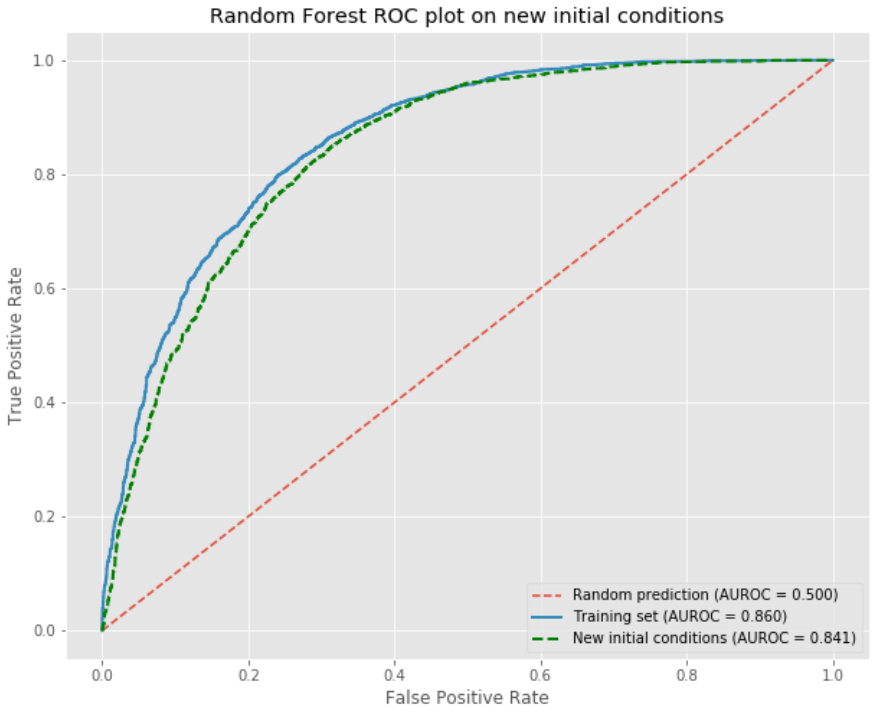
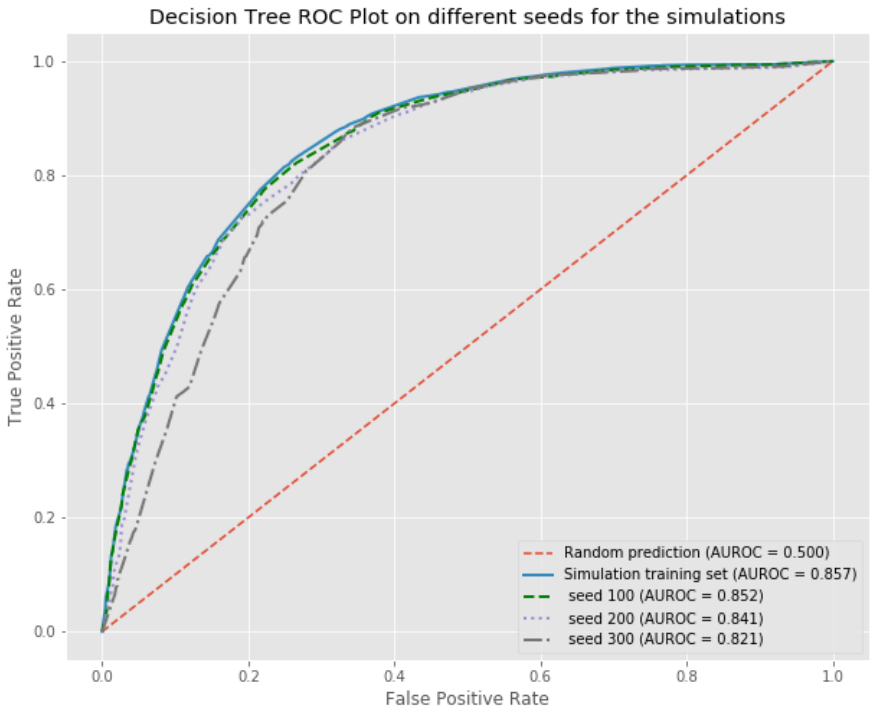


Figura 4.8: Curvas ROC de los algoritmos de árbol de decisión y random forest de las condiciones iniciales con un nuevo suavizado gravitacional ϵ comparadas con el desempeño anteriormente mostrado. Las curvas son bastante consistentes. El valor del área bajo la curva ROC bajó $\sim 2\,\%$. Las pruebas demuestran la gran capacidad de los algoritmos para predecirlas etiquetas finales de simulaciones diferentes.



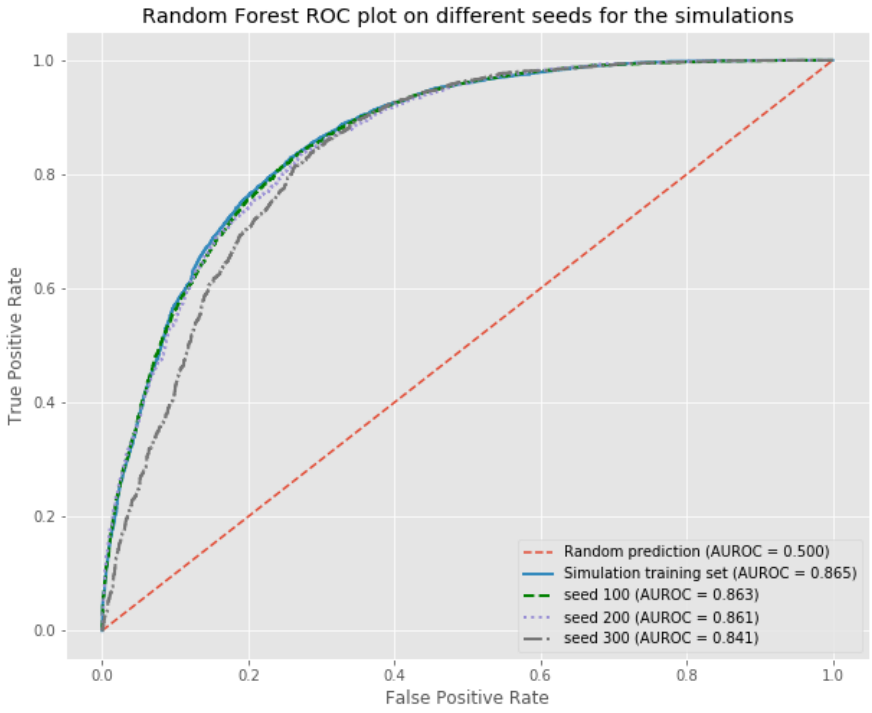


Figura 4.9: Curvas ROC de los algoritmos de árbol de decisión y random forest de las condiciones iniciales cuyo "seed" fue diferente. El área bajo la curva ROC baja un promedio del 2.2 % para las realizaciones nuevas. La generalización del poder predictivo del entrenamiento es evidente ya que loa algoritmos son capaces de decidir en buena manera el destino final de las partículas de materia oscura desde su posición en una posición inicial.

Capítulo 5

Trabajo a futuro

Los algoritmos de machine learning son utilizados en gran manera tanto en la industria como en proyectos científicos. La aplicación de estos algoritmos y el conocimiento de Big Data se vuelve cada día más una necesidad en vez de solo una herramienta de análisis. Los datos y su exploración son los responsables del comportamiento de nuevas tendencias sociales, la rápida toma de decisiones y la reducción de costos. Usualmente los modelos entrenados dan como resultado las mejores posibles predicciones o la detección de posibles errores o diferencias sutiles entre elementos.

Es en la diferencia sutil de elementos en lo que se quiere trabajar a futuro. Hasta ahora, se ha demostrado que los algoritmos son capaces de hacer una predicción bastante correcta al momento de seleccionar etiquetas. Esta prueba fue hecha para una simulación del modelo estándar cosmológico ΛCDM. Sin embargo, es bien sabido que ΛCDM está lejos de ser el modelo definitivo de evolución cosmológica. En particular, el modelo estándar tiene ciertas incongruencias, algunas de ellas ligadas a la materia oscura y otras cuantas ligadas a la energía oscura. En este trabajo el enfoque fue hacia las simulaciones de materia oscura cosmológica y su comportamiento, por tanto, los problemas de ΛCDM relacionados con la composición de materia oscura son:

1. Perfiles CUSP-CORE: Las simulaciones predicen la densidad central los halos centrales galácticos tenga un comportamiento con un CUSP o pico central (Navarro, J. F., Frenk, C. S.; White, S. D. M., 1996 [60]). Sin embargo las observaciones dictan que la densidad de los halos galácticos está suavizada, es decir que se distribuye de manera homogénea en todo el halo galáctico (Moore, B., Quinn, T., Governato, F., Stadel, J., Lake, G., 1999 [65]).

2. Satélites faltantes: el número de subhalos masivos predichos por simu-

laciones excede el número observado de satélites luminosos de galaxias de tamaño similar a la Vía Láctea en al menos un orden de magnitud. (Klyipin, A. A.; Kravstov A. V.; Valenzuela, O.; Prada, F., 1999 [66]).

Estas discrepancias dentro del modelo estándar pueden ser evidencia de la importancia de procesos físicos en la materia bariónica. Pero también pueden ser indicativos de un nuevo tipo de materia oscura, con propiedades diferentes a las propuestas por ΛCDM y con la posibilidad de resolver estas dificultades mencionadas.

Uno de ellos es el modelo de Materia oscura como campo escalar, por sus siglas en inglés (SFDM). Este modelo supone que la materia oscura es un campo escalar real o complejo Φ mínimamente acoplado a la gravedad, dotado de un potencial escalar $V(\Phi)$ y que a cierta temperatura, la interacción del campo es puramente gravitacional junto con el resto de la materia. El modelo fue propuesto hace ya más de dos décadas de manera independiente por Sahni, V., y Wang, L. (1999 [67]), Hu W., Barkana, R., Gruzinov, A., (2000 [68]) y Matos, T. y Ureña-López, L. A. (2000 [69]). Un análisis más a fondo se encuentra en el apéndice B. La idea básica es que la materia oscura se considera como un bosón de espín 0, con una masa del orden de $m \sim 10^{-22}$ eV y una longitud de onda de Compton $\lambda \sim \mathcal{O}(\text{kpc})$ similar a un tamaño de una galaxia.

5.1. Modificación del campo escalar: Axion-GADGET

Para este modelo en particular existe una modificación al código GADGET, llamada AxionGADGET (Zhang, J. and et al., 2018 [39]), utiliza la aproximación hidrodinámica del campo escalar de materia oscura y lo implementa en la formación de estructura. De nuevo, se refiere al apéndice B para el análisis a detalle de esta implementación a la modificación de las ecuaciones de movimiento de GADGET. A manera de resumen, en la aproximación hidrodinámica la materia oscura se describe mediante la transformación de Madelung

$$\Psi(t, \mathbf{x}) = \psi(t, \mathbf{x}) \exp\left(-iS(t, \mathbf{x})/\bar{h}\right) \tag{5.1}$$

y que al resolver la ecuación de movimiento se obtiene un potencial cuántico

$$Q = -\frac{1}{2}\frac{\nabla^2\psi}{\psi}. \tag{5.2}$$

Dicho potencial, actúa como un tipo de "presión negativa" (razón del signo negativo en la ecuación (5.2)) que tiene un orden de magnitud similar a la longitud de onda de Compton del campo escalar, es decir $Q \sim \mathcal{O}(\lambda)$. Lo importante a señalar sobre esta presión es que actúa de manera diferente dependiendo de la distancia entre partículas de materia oscura. Si la distancia es mayor que el valor de la longitud de onda de Compton, el potencial se vuelve atractivo, evitando la formación de subestructura y atacando el problema de satélites faltantes. En cambio, si la distancia entre partículas es menor que la longitud de onda, el potencial cambia, se vuelve repulsivo y no permite que se formen picos en los halos galácticos, distribuye mejor la materia oscura evitando así el problema del CUSP-CORE.

5.1.1. Simulaciones con AxionGADGET

Se realizó una simulación con la modificación, creando condiciones iniciales similares a las de la tabla 2.2, con el agregado de que la masa de la modificación debe cambiar, las partículas de materia oscura escalar tienen un valor de $m = 1.21 \times 10^{-22}$ eV. De manera que la longitud de onda de Compton es del orden de $\lambda \sim$ kpc. El espectro de potencias de masa y la HMF se describen en la figura 5.1. El espectro de potencias muestra que la distribución de materia oscura para SFDM es menor en la época actual. A pesar de que coincide bien con la simulación y la teoría de ΛCDM en números de onda $k \sim 0.01$, empieza a desacoplarse a medida que los números de onda k aumentan, lo cual significa que la distribución de halos galácticos ha descendido por la implementación de la aproximación hidrodinámica. De igual forma, la HMF del campo escalar se ajusta bien a la teoría del colapso esférico y del ajuste de Press-Schechter en el rango de masas entre $10^{12} - 10^{14}\mathrm{M}_{\odot}$. Sin embargo a medida que se desciende en el rango de masas, se observa que el campo escalar no está formando halos menos masivos, a diferencia de la teoría de ΛCDM. Por tanto, es posible concluir que, aunque siendo un toy model, la simulación es capaz de atacar los problemas mencionados para ΛCDM.

5.2. Distinción entre modelos de materia oscura: un reto para deep learning

Dado que se ha demostrado que los algoritmos de machine learning son capaces de discernir entre clases de partículas pertenecientes a un halo de materia oscura, se podría pensar que podrían ser capaces de distinguir entre modelos de materia oscura. Tal como es posible discernir de la figura 5.1,

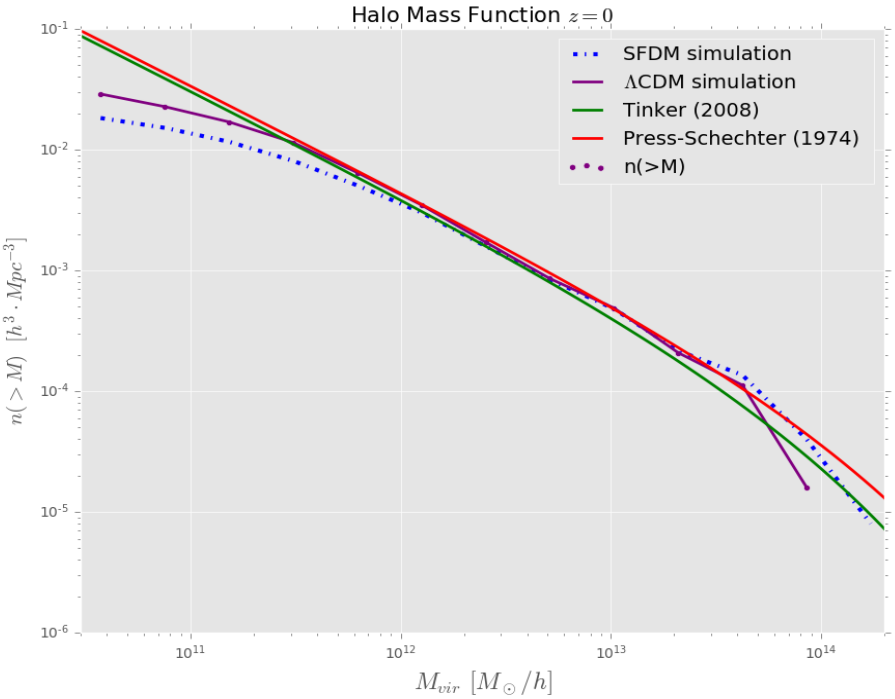


Figura 5.1: Espectro de potencias de masa y halo mass function para las simulaciones de ΛCDM y SFDM con su respectiva comparación con los modelos teóricos. El espectro del campo escalar es diferente a escalas pequeñas, ya que la distribución de halos de materia oscura es menor. Similarmente, la halo mass function de SFDM forma una menor cantidad de halos de materia oscura con rango de masas menor a 10^{12} $M_\odot$.

ambos modelos presentan diferencias, aunque sutiles, que al final han sido parte del estudio de la cosmología y del problema de materia oscura.

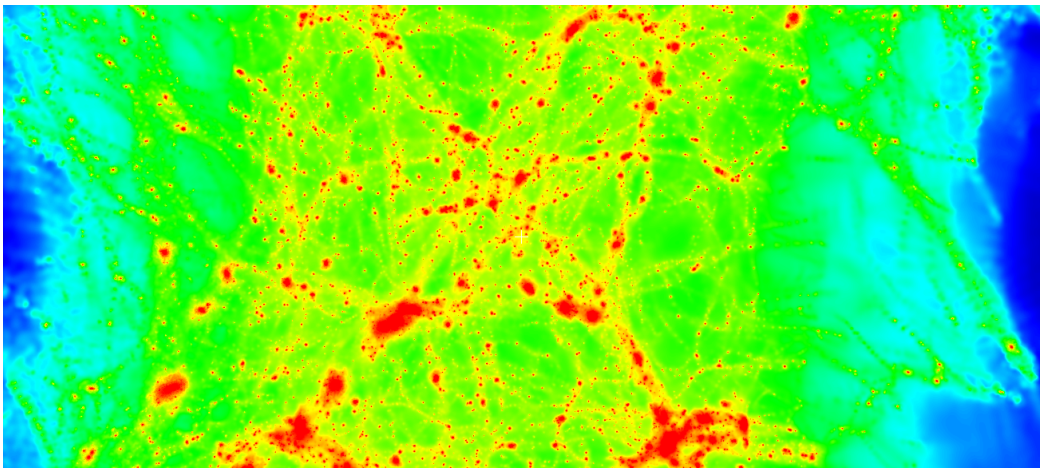
Esto es un buen indicativo, ya que, al igual que la distinción de clases para partículas, lo mismo puede hacerse para modelos de materia oscura. Sin embargo, y a manera de comentario, los algoritmos de machine learning aquí descritos no son la manera óptima de hacerlo. Éstas son las razones:

Utilizar sólo las propiedades de densidad y sobredensidad de las condiciones iniciales de las simulaciones no son suficientes. Imagine que tiene un tazón con frutas y se llega a diferenciar exitosamente manzanas, naranjas, plátanos, etc. Pero, llegando a un nodo, se encuentra con limones verdes y uvas verdes de tamaño similar y sin mayores características para hacer una distinción. Los algoritmos de clasificación, sin mayor información, no son capaces de distinguir entre estas dos frutas, de manera que terminan confundidos y adivinando cuál fruta es cual. Lo mismo pasó con las simulaciones y las características de sobredensidad en las condiciones iniciales.

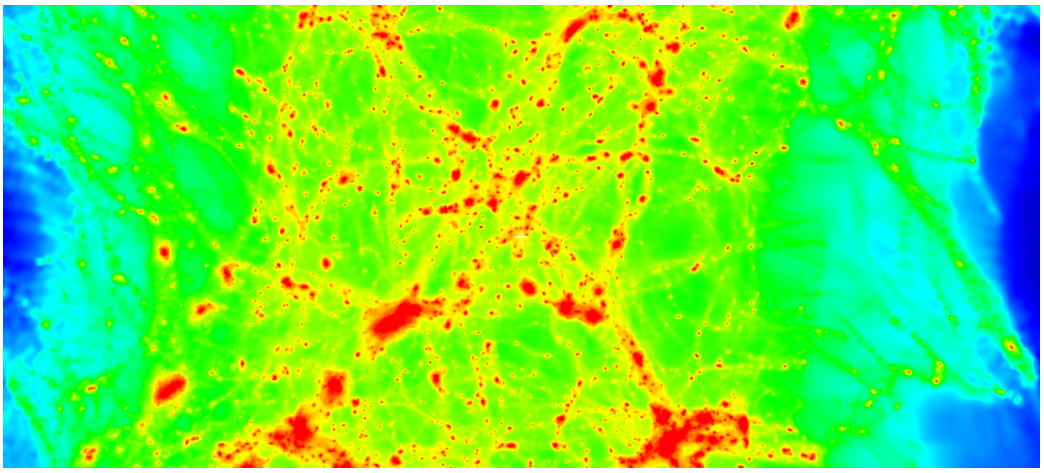
Al final, si una partícula perteneciente a un halo de materia oscura del modelo ΛCDM también pertenecía a un halo de SFDM y las características eran las mismas (o similares), los algoritmos no eran capaces de distinguir entre un modelo u otro. Por lo tanto, la curva ROC de esta particular selección era una línea completamente constante, evidenciando la confusión de los algoritmos dadas las características. Esto puede entenderse mejor con las imágenes obtenidas para cada simulación, descritas en la figura 5.2. La figura 5.2 (a) muestra el resultado de ΛCDM, detallando que la subestructura de halos (las regiones más densas coloreadas en rojo) es mayor en este modelo. A diferencia de SFDM 5.2 (b) y (c), la subestructura desciende para ambos modelos, llegando a casi deshacerse de mucha subestructura. Incluso, para el ojo humano, es difícil discernir entre estos resultados.

Afortunadamente este no es el fin del camino, ya que existen otros métodos que pueden aplicarse a este problema en particular, puede extenderse la cantidad de características extraídas de las condiciones iniciales, por ejemplo agregar velocidades relativas de los halos o la cantidad de subestructura creada pueden ayudar a los algoritmos de machine learning a efectuar una mejor clasificación.

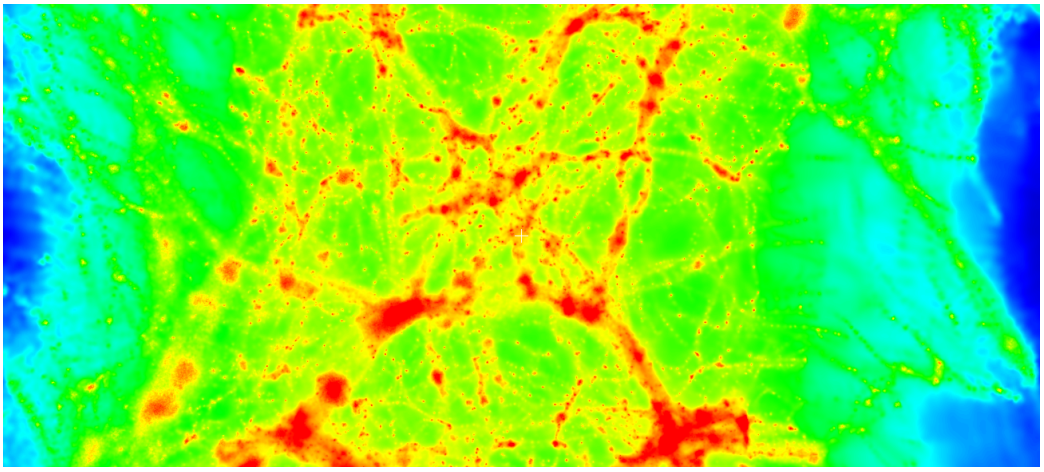
Otra rama de la inteligencia artificial que posiblemente otorgue mayor percepción sobre la clasificación de modelos es *deep learning*. Las redes neuronales artificiales son algoritmos que proveen modelos de relación para clasificación de bases de datos relacionales (tablas, listas) y no relacionales (archivos de sonido, imágenes). Hay un sinfín de arquitecturas de redes neuronales artificiales (Hagan, M. T., 1997 [70]), cada una con un propósito diferente. Las redes neuronales convolucionadas (CNN) pueden ser una manera de llevar acabo un análisis más a detalle de este problema particular.



(a) ΛCDM



(b) SFDM, $m \sim 10^{-22}$ eV



(c) SFDM, $m \sim 10^{-23}$ eV

Figura 5.2: Cortes frontales de las simulaciones de ΛCDM y SFDM con dos masas diferentes en $z = 0$. Las partes más densas son halos de materia oscura y subhalos. La similitud es tan notable que un algoritmo de clasificación no puede distinguir entre modelos sólo con las características de la sobredensidad de las condiciones iniciales.

Estos modelos están inspirados por los procesos biológicos que tienen lugar en el córtex visual, donde las neuronas individuales responden a estímulos en un área restringida del campo visual. Ésta región se superpone parcialmente con el de las neuronas más próximas, cubriendo de forma colectiva el campo visual completo. Consisten en múltiples capas de filtros convolucionales de una o más dimensiones. Después de cada capa, por lo general se añade una función para realizar un mapeo causal no-lineal.

La red, toma un objeto de entrada como base de datos, y lo descompone en capas de pixeles, cada capa puede ser de menor o igual resolución, tomando en cuenta los datos de la capa anterior, mientras que la última capa es la de clasificación y que tiene una unidad por cada clase que la red predice. Cada unidad recibe los datos de todas las unidades de la capa anterior inmediata (Kang, X. Song, B. Sun, F., 2019 [71]).

La idea entonces es generar una gran cantidad de imágenes de las simulaciones de ΛCDM y SFDM en diferentes redshift z, tomarlas como datos de entrada para entrenar la red convolucional y observar si el algoritmo puede discernir de un modelo u otro, dependiendo de cada z. Existen trabajos similares cuyo uso de redes neuronales convolucionadas ayuda a discriminar modelos de gravedad modificada y modelo estándar utilizando redes convolucionadas (Peel, A. and et al., 2019, [72]) o para reconstrucciones de la red cósmica creada por la distribución de materia (Buncher, B., Carrasco Kind, M., 2020 [73]). De manera que existe una gran expectativa de que de este nuevo acercamiento se obtengan resultados consistentes con la discriminación de modelos cosmológicos.

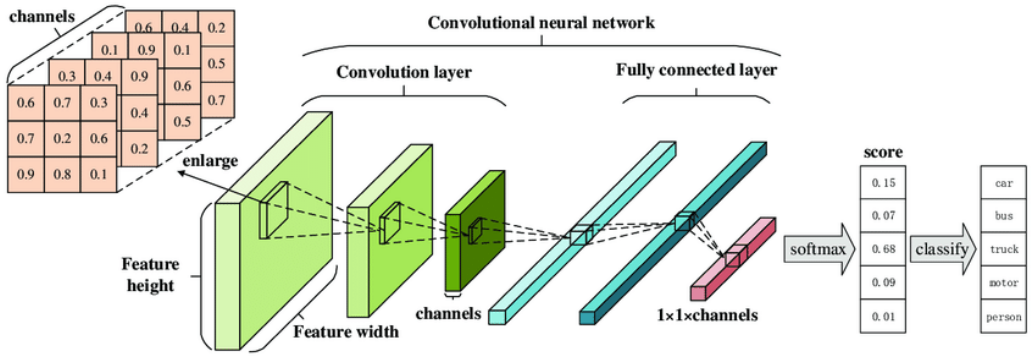


Figura 5.3: Esquema de una red neuronal convolucionada (CNN). La red toma una imagen como entrada de datos y la descompone a medida que las capas de la red son más profundas. La capa final tiene una unidad por cada clase predicha por la red. Tomada de Kang, X. and et al., 2019.

5.3. Discusión final

Durante el proceso de escritura de este trabajo se puntualizaron diversos factores que pudieron ser determinantes sobre los resultados obtenidos. Uno de los már relevantes fue el uso de la simulación y su probable desviación o sesgo debido a diferentes parámetros numéricos independientemente de procesos físicos. Como es bien sabido, las simulaciones requieren de conocer muchos parámetros numéricos además de conocer a profundidad el código utilizado. Al no ser el tema principal del trabajo no se profundizó en estos aspectos, sin embargo, para un proyecto futuro como el señalado en la sección anterior debe ser una prioridad para tener una mejor certeza de que el resultado obtenido puede ser comparable con un sistema físico realizable y no tener errores de sesgo o variaciones.

5.3.1. Teoría de picos y simulaciones numéricas

Existen tópicos interesantes sobre la densidad de número en un campo de densidad de materia oscura y su relación con la población de halos de materia. La relación entre la pendiente del espectro de potencias lineal y la aglomeración de halos y se conoce como teoría de picos o Peak Theory. En particular para un campo de densidad isotrópico con una distribución normal de modos de Fourier, se encuentra un promedio de densidad de número que relaciona el espectro de potencias y los picos de densidad a diferentes escalas (Bardeen, J. M.; Bond, J. R.; Kaiser, N.; Szalay, A. S., 1986 [74]). Al utilizar simulaciones numéricas puede deducirse de una manera similar la densidad de número del campo de densidad y relacionarla con la distribución de halos de galaxias. La meta final de medir los picos de densidad de número es restringir parámetros cosmológicos que determinen el espectro de potencias lineal. Estos resultados son interesantes dado que toman expresiones completamente analíticas para determinar la población de halos de materia en una simulación de materia oscura sin interacción con materia bariónica. En estas simulaciones una galaxia central caería en un halo cuyo centro está más cercano a la posición del halo cuyas partículas estén mayormente ligadas dentro de ese halo, es decir que la posición determina si un halo es un halo host o padre o termina siendo una galaxia satélite (S. De, R. A. C. Croft, 2007, [75]).

A diferencia de este método, el utilizado en el trabajo no obtiene una relación de la población de halos de materia oscura y el campo de densidad mediante el espectro de potencias, si no que las propiedades mismas del campo de densidad son las determinantes para que los algoritmos hagan una decisión sobre si una partícula de materia oscura cae dentro de un halo de

cierto umbral de masa. Se ha puntualizado que el proceso de entrenamiento del algoritmo tiene que ser más refinado, dado que la cantidad de partículas seleccionadas representa apenas solo el 0.004 % de la cantidad total de partículas dentro de la simulación. Esto sin duda puede ser determinante, por supuesto, dado que al final se tenía una matriz de elementos de aproximadamente $317,680$ elementos que debió usarse como datos de entrenamiento. Incluso así, se realizó otra prueba para una elección de $57,000$ partículas, realizando el procedimiento descrito en el capítulo 4 en la sección 4.1. El área bajo las curvas ROC de los algoritmos utilizados no mostró una mejoría ya que en ambas realizaciones tiene un resultado similar (0.85 para árbol de decisión y 0.86 para random forest). Es debido a esto que al menos la utilización de más volumen de partículas no es determinante en el proceso de identificación.

Esto no quiere decir que no pueda mejorarse este algoritmo o no pueda hacerse más robusto, existen otras razones y métodos a considerar, los cuales son tanto físicos como numéricos. Entre los físicos viene a la mente el utilizar además de la sobredensidad, la velocidad, una cantidad finita de halos definida, la cantidad de vaciós que encuentra, si hay procesos de enfriamiento, si hay procesos de desaceleración, al final, todas esas cantidades pueden tener un mayor impacto en la realización de los algoritmos. Por la parte numérica, debe cuidarse todo detalle de la simulación, principalmente la resolución (cantidad de partículas dentro de la simulación) y el tamaño de paso de integración para la resolución de las ecuaciones de movimiento. Estos factores son los más importantes para determinar si la simulación tiene algún tipo de sesgo o error en el resultado final más allá de la comparación directa con el espectro de potencias no lineal o la Halo Mass Function, algo que también debe tomarse en cuenta es qué tan sensible es el buscador de halos en la simulación. Estos elementos no se tomaron en cuenta dado que la simulación fue tomada de un trabajo previo (Chacón, J., 2018, [76][1]).

Un aspecto más a considerar es sin duda el saber la cantidad de carga de entrenamiento es necesaria y suficiente para los algoritmos de clasificación. Es importante saber por qué se hace esa pregunta. Por ejemplo:

- ¿Hay muchos datos? La mejor manera es cosiderando las curvas de aprendizaje de entrenamiento para saber si una muestra es suficientemente representativa.

- ¿Hay pocos datos? Hay que confirmar si en realidad son pocos datos o si existe una manera de aumentar la cantidad del tamaño de muestreo.

[1] http://pelusa.fis.cinvestav.mx/tmatos/CV/3_RecursosH/Lic/Jazhiel_ESFM.pdf

La respuesta rápida es : depende. No hay una manera eficiente de considerar qué tan grande o representativo es la muestra para el problema. Esto se debe más a una prueba y error. La cantidad de datos requeridos depende de muchos factores, como la complejidad del problema y la complejidad del algoritmo de aprendizaje.

Bien, para el problema presentado en este trabajo es posible considerar que la cantidad de datos de entrenamiento y prueba no fue el adecuado. Existieron diversos factores generales para el uso de tal cantidad de datos como entrenamiento de los algoritmos de clasificación. La complejidad de los árboles de decisión y random forest es tan grande como la cantidad de características que se consideran como parte de la clasificación. La sensibilidad de este tipo de algoritmos depende de la cantidad de pruebas hechas para probar a los clasificadores, como se menciona antes, al duplicar la cantidad de partículas en el entrenamiento/testeo de los algoritmos no se observó una mejora significativa o lo suficientemente notable para reportar un cambio en los resultados, finalmente, los clasificadores demostraron la capacidad de entendimiento y escalabilidad del problema. Aumentar la cantidad de datos, digamos a un 80 % del total de la simulación (aproximadamente 5 millones de datos) se podría afirmar que llega a ser representativo para el problema que se plantea, lo cual lleva a otro factor: el bajo recurso computacional.

Llevar a cabo un entrenamiento, incluso después de haber hecho una malla de hiperparámetros suficientemente buena para cosiderar que el algoritmo es eficiente puede llevar mucho tiempo, el código utilizado no ha sido optimizado para poder efectuarse en manera paralela y mucho menos en algoritmos de deep learning (como se explica más adelante). La dimensionalidad como parte de representatividad para datos cosmológicos se analiza mejor con otro tipo de estructuras analíticas de clasificación, tales como las redes neuronales artificiales. La escalabilidad del planteamiento del problema no debe tomarse a la ligera, puede que en un futuro exista la posibilidad de realizar este u otro tipo de desarrollos tomando en cuenta la cantidad representativa de datos para poder determinar un resultado confiable.

5.3.2. Simulaciones numéricas asistidas con inteligencia artificial

Existe otra alternativa a la realización completa de una simulación numérica, utilizando redes neuronales generativas antagónicas o redes GAN. Estas redes básicamente toman bases de datos o imágenes de las cuales el algoritmo genera dos redes, una generadora y una discriminadora. las redes empiezan a competir y mbas redes fueron entrenadas con un mismo conjunto de datos,

pero la primera debe intentar crear variaciones de los datos que ya ha visto, en el caso de imágenes que no existen, debe crear variaciones de las imágenes que ya ha visto.

La red discriminatoria debe identificar si la imagen que está viendo forma parte del entrenamiento original o si es una imagen falsa que creó la red generativa. Mientras más lo hace, la red generativa se hace mejor creando y la red discriminatoria le es más difícil identificar si la imagen es real o falsa.

La red generadora necesita la discriminadora para saber cómo crear una imitación tan realista que la segunda no logre distinguir de una imagen real.

Es así que las redes GAN han determinado ser un aliado en la generación de ambientes o simulaciones numéricas cosmológicas. Existen trabajos que utiliza una simulación de baja resolución y que genera estadísticamente los mismos resultados que daría una de alta resolución. Las redes están compuestas por varias pequeñas capas, luego se comparan mediante su función de pérdida y la imagen que generan puede mejorarse (ya que el primero resultado es algo borroso) usando la red GAN. Como se ha comentado, la red GAN se usa para generar datos completamente nuevos de datos de entrada aleatorios, que en este caso sería crear imágenes de alta resolución usando datos o simulaciones de baja resolución.

El campo de desplazamiento se canaliza como una imagen en 3 dimensiones con 3 canales, cada canal corresponde al vector de desplazamiento, el modelo de deep learning toma los desplazamientos de la simulación de baja resolución y genera una posible realización de alta resolución, por lo que este resultado puede verse como una simualación de alta resolución con mayor cantidad de partículas y mayor resolución en masa (Li, Y.; Ni, Y.; Croft, R. A. C.; Di Matteo, T.; Bird, S.; Feng, Y., 2020, [77]).

Ciertamente, el campo de la inteligencia artificial está revolucionando la manera de hacer investigación, no es sorpresa que existan tantas maneras de abordar un solo problema, por supuesto que debe haber cierto cuidado y consideración para poder realizar tareas y trabajos cuyo resultado sea fiable. En el caso de cosmología numérica, lo más importante, luego de las consideraciones físicas es la parte de la solución de ecuaciones de movimiento numéricas, para luego dar paso a la obtención de sistemas físicos comparables con resultados ya existentes. La inteligencia artificial no es para nada un área nueva, y sin duda es el siguiente paso a la investigación de vanguardia.

Conclusiones

Esta tesis presenta una de las aplicaciones de las sub ramas de la inteligencia artificial, machine learning, sobre las capacidades para obtener información de la formación de estructura cosmológica. El volumen de datos observacionales y de simulaciones es una tarea bien estructurada que requiere de un gran poder deductivo, computacional y de reconocimiento. La relación entre las propiedades de las condiciones iniciales y la clasificación final dependiente de la masa de los halos de materia oscura en simulaciones de N-cuerpos otorgó un marco de estudio para entrenar los algoritmos de clasificación.

La implementación de algoritmos de clasificación binaria se efectuó dotando de información de las propiedades del campo de densidad y la formación de halos de materia oscura. Para este fin, un árbol de decisión y un random forest fueron entrenados para predecir si las partículas de materia oscura colapsan en halos cuya masa era mayor o igual que un valor umbral de $M = 1.2 \times 10^{12}$ $\mathrm{M}_{\odot}$.

El proceso muestra que las propiedades de densidad obtenidas en las condiciones iniciales son suficientes para llevar a cabo la predicción, ya que los algoritmos muestran una preferencia por un rango de valores de sobredensidad que está directamente ligado al umbral escogido para efectuar la clasificación.

Al comparar ambos algoritmos, se mostró que random forest tiene una ligera, aunque no despreciable, mejoría sobre el árbol de decisión al efectuar predicciones de clasificación, debido a que existe poca correlación dentro del random forest sobre las características de las propiedades de densidad de las condiciones iniciales. Ambos algoritmos, sin embargo, obtienen un resultado similar bastante confiable y cuya importancia se ve reflejada en la elección del umbral de clasificación.

El desempeño en general, no mostró ninguna clase de errores o sesgos por parte de la elección de las características de densidad de las condiciones iniciales, los hiperparámetros de los algoritmos escogidos fueron tales que el aprendizaje obtiene una exactitud de alrededor de 78 %.

La capacidad predictiva de los algoritmos no quedó en duda. Al efectuar un nuevo proceso de clasificación para la extracción de propiedades de densidad de un nuevo conjunto de condiciones iniciales, los resultados mostraron que aunque su desempeño se redujo un poco, reflejado en el área bajo la curva ROC, pueden generalizarse. De esta manera, es posible realizar una clasificación de partículas de materia oscura de condiciones iniciales para ΛCDM sin la necesidad de ejecutar la simulación completa.

Este es de los resultados más importantes del trabajo, ya que la generalización debido al correcto entrenamiento de simulaciones permitirá obtener bases de datos recopiladas de simulaciones, llevar a cabo una comparación de desempeño y ahorrar tiempo de ejecución computacional, la cual en muchas ocasiones es un recurso limitado. Incluso para diferentes realizaciones cuya generación de condiciones iniciales cambió la generación de números pseudoaleatorios para las partículas, el resultado final fue una predicción competente con lo esperado de un entrenamiento para estos algoritmos.

No está fuera del tema el realizar pruebas más efectivas sobre la elección de características en el modelo de entrenamiento, ya que al aumentar el volumen de partículas a manera de entrenamiento no mostró ninguna diferencia o mejoría sobre el desempeño de los dos algoritmos descritos, de manera que aumentar las características sobre las simulaciones puede ser determinante para obtener un clasificador con mayor fiabilidad sin llegar a una sobreestimación.

Sin embargo, las propiedades de densidad de las condiciones iniciales mostraron no ser suficientes para poder discriminar modelos de evolución cosmológica. Tanto ΛCDM como SFDM muestran mucha similitud en sus propiedades iniciales, de manera que los algoritmos de clasificación descritos mostraron una confusión al momento de tratar de discernir entre estos dos modelos.

Una de las vías a explorar es Deep Learning, dado que el uso de redes neuronales facilitaría la carga que genera realizar una simulación de un gran volumen de partículas, como se describe en la discusión final, el uso de redes bla bla realmente asemeja una simulación de gran volumen con la ventaja de reducir el tiempo y costo computacional.

Será preciso utilizar un método diferente si se desea clasificar distintos modelos de formación de estructura, las redes neuronales convolucionadas (CNN) son el mejor candidato para esta tarea. Son utilizadas principalmente para poder analizar bases de datos no relacionales, como imágenes. Las simulaciones de ΛCDM y SFDM tienen una similitud notable en cuanto a formación de halos de materia oscura. La manera de entrenar a la red neuronal deberá ser con diferentes fotografías en distintas épocas de las simulaciones de ambos modelos y evaluar su desempeño. El futuro de este trabajo pre-

tende obtener mayor percepción sobre las propiedades físicas y de formación de estructura cosmológica utilizando herramientas propias de la inteligencia artificial, sin duda un área con mucho potencial aún por descubrir.

Apéndice A

Ejemplo de clasificación Booleana

El objetivo de esta sección es construir un árbol de decisión para determinar si se debe esperar o no un lugar en un restaurante. Para esto, la etiqueta final (¿Esperar? = si, no) se determina mediante una lista de atributos.

1. *Alternativa*: Determina si hay un restaurante cercano que pueda ser una alternativa.
2. *Bar*: El restaurante tiene servicio de bar.
3. *Viernes/Sábado*: Verdadero si es viernes o sábado.
4. *Hambre*: Se está mucho o nada hambriento.
5. *Clientes*: Cuánta gente hay dentro del restaurante (Nada, Algo, o Lleno).
6. *Precio*: Rango de precios del restaurante ($, $$, $$$)
7. *Lluvia*: Está lloviendo afuera.
8. *Reservacíon*: Se tuvo que hacer una reservación.
9. *Tipo*: Clase de comida que se sirve (Francesa, Italiana, Thai o hamburguesas).
10. *Tiempo de espera*: El estimado de tiempo de espera para los clientes (0-10 minutos, 10-30, 30-60 o >60).

El árbol de decisión inicia en el nodo raíz, siguiendo una rama apropiada hasta llegar a un nodo hoja. Para ilustrar esto, un ejemplo con *Clientes* =

Lleno, y $TiempodeEspera$ = 0-10, será clasificado como verdadero (si esperar un lugar en el restaurante). Un árbol de decisión Booleano consiste de un par $(\mathbf{x}, y)$, donde $\mathbf{x}$ es un vector de valores para los atributos de entrada e y es un valor singular Booleano de salida.

De manera general, después de escoger el mejor atributo para realizar la división, cada resultado obtiene como consecuencia un nuevo árbol de decisión con un atributo menos y con menor cantidad de ejemplos. La manera recursiva de reducir y llegar a una conclusión o una hoja es la siguiente:

1. Si todos los elementos caen dentro de una categoría, en este caso booleano, positivo o negativo, el proceso termina.

2. Si existen elementos que son positivos y negativos, se escoge el mejor atributo para realizar la división.

3. Si no quedan elementos, quiere decir que en ese nodo no se ha encontrado uno que satisfaga ese atributo o conjunto de atributos, de manera que se regresa un valor determinado, calculado a partir de la abundancia en la clasificación de todos los ejemplos utilizados para construir el nodo padre.

4. Si no quedan atributos, pero si elementos clasificados como positivo o negativo, significa que estos elementos tienen exactamente la misma descripción pero diferente clasificación. En este caso se dice que existe **ruido** (noise) en los datos, pues indica que un atributo es incapaz de distinguir entre dos ejemplos.

Si un conjunto de entrenamiento contiene p ejemplos positivos–en este caso, van a esperar una mesa–y n ejemplos negativos–no esperan una mesa– la entropía del valor objetivo del conjunto es

$$H(Objetivo) = B\left(\frac{p}{p+n}\right), \tag{A.1}$$

donde $B(q)$ es la entropía de una cantidad aleatoria Booleana que es verdadera con probabilidad q

$$B(q) = -(q \log_2 q + (1-q)\log_2(1-q)). \tag{A.2}$$

El conjunto de entrenamiento de arriba tiene $p = n = 6$, la entropía correspondiente es $B(0.5)$. Un atributo A con d distintos valores divide el conjunto de entrenamiento T en d subconjuntos $T_1, \ldots, T_d$, cada subconjunto T_k tiene p_k ejemplos positivos y n_k ejemplos negativos. Si se toma el camino

Ej.	Atributos										Etiqueta
	Alt	Bar	Fri	Hun	Pat	Price	Rain	Res	Type	Est	Will Wait
	Yes	No	No	Yes	Some	$$$	No	Yes	French	0-10	Yes
	Yes	No	No	Yes	Full	$	No	No	Thai	30-60	No
	No	Yes	No	No	Some	$	No	No	Burger	0-10	Yes
	Yes	No	Yes	Yes	Full	$	Yes	No	Thai	10-30	Yes
	Yes	No	Yes	No	Full	$$$	No	Yes	French	>60	No
	No	Yes	No	Yes	Some	$$	Yes	Yes	Italian	0-10	Yes
	No	Yes	No	No	None	$	Yes	No	Burger	0-10	No
	No	No	No	Yes	Some	$$	Yes	Yes	Thai	0-10	Yes
	No	Yes	Yes	No	Full	$	Yes	No	Burger	>60	No
	Yes	Yes	Yes	Yes	Full	$$$	No	Yes	Italian	10-30	No
	No	No	No	No	None	$	No	No	Thai	0-10	No
	Yes	Yes	Yes	Yes	Full	$	No	No	Burger	30-60	Yes
1	2	3	4	5	6	7	8	9	10	11	12

Tabla A.1: Ejemplo para determinar si se debe esperar un lugar en un restaurante.

sobre esa rama, se necesita calcular la entropía adicional $B(p_k/(p_k + n_k))$ para obtener información adicional en esta división.

Una elección aleatoria de un ejemplo en el conjunto de entrenamiento tiene el k-esimo valor para el atributo con probabilidad $(p_k + n_k)/(p + n)$, la entropía esperada restante luego de probar el atributo A es

$$Resto(A) = \sum_{k=1}^{d} \frac{p_k + n_k}{p + n} B\left(\frac{p_k}{p_k + n_k}\right). \tag{A.3}$$

Así que la ganancia de información del atributo A es una esperada reducción a la entropía

$$Ganancia(A) = B\left(\frac{p}{p + n}\right) - Resto(A). \tag{A.4}$$

En el ejemplo, la ganancia de $Patrons$ es

$$Ganancia(Patrons) = 1 - \left[\frac{2}{12}B\left(\frac{0}{2}\right) + \frac{4}{12}B\left(\frac{4}{4}\right) + \frac{6}{12}B\left(\frac{2}{6}\right)\right] = 0.541, \tag{A.5}$$

mientras que la ganancia por $Type$ es

$$Ganancia(Type) = 1 - \left[\frac{2}{12}B\left(\frac{1}{2}\right) + \frac{2}{12}B\left(\frac{1}{2}\right) + \frac{4}{12}B\left(\frac{2}{4}\right) + \frac{4}{12}B\left(\frac{2}{4}\right)\right] = 0. \tag{A.6}$$

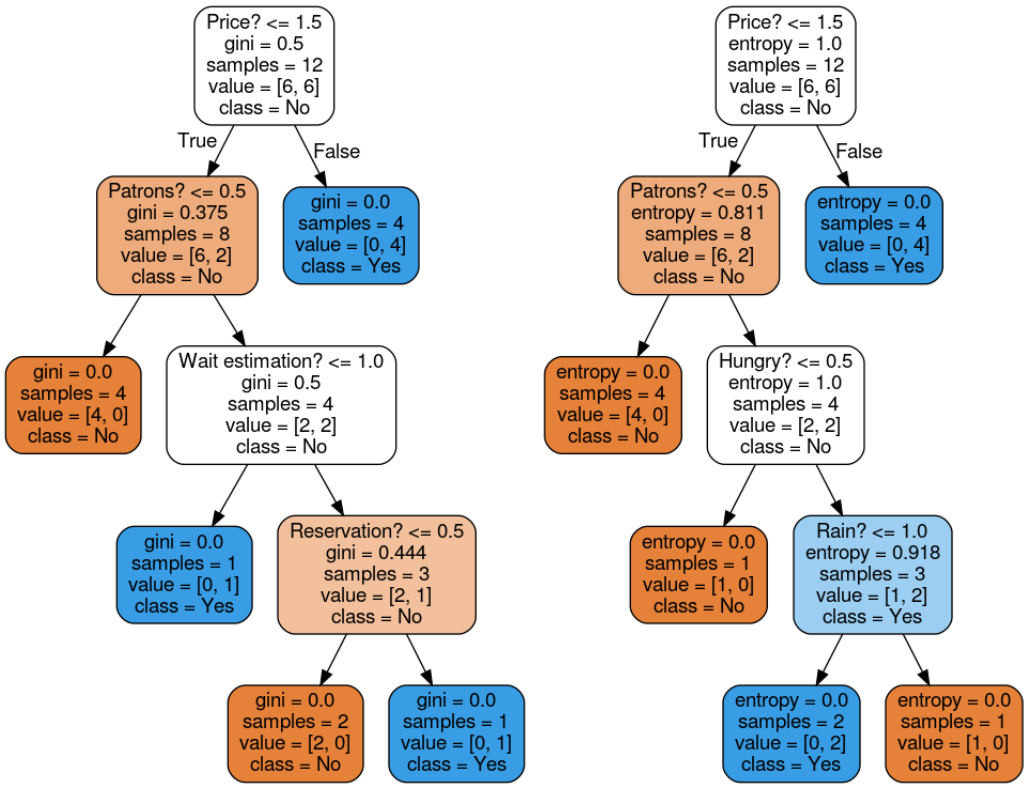


Figura A.1: Árboles de decisión del conjunto de datos de la tabla A.1. Izquierda) Criterio de partición: índice Gini. Derecha) Criterio de partición: entropía. Ambos algoritmos tienen el mismo nivel de profundidad (4), pero la división de los nodos se basa en diferentes atributos, dependiendo del criterio. Imágenes generadas con Scikit Learn y alojadas en GitHub ChJazhiel.

La ganancia de información por *Patrons* es mayor que por *Type*, es decir que la elección del restaurante por la cantidad de gente que hay dentro de él tiene mayor peso y es un mejor atributo para dividir el conjunto de datos. En la figura A.1 se muestran dos árboles de decisión con el conjunto de datos de la tabla A.1 con los criterios de "índice Gini" y "entropía". Se observa que ambos clasificadores tienen el mismo nivel de profundidad e indican que la mayor ganancia de información proviene del atributo *Price*. Sin embargo, al ir más profundo en los estimadores, es notorio que los atributos con los cuales la ganancia de información es mayor cambian.

Apéndice B

Scalar Field Dark Matter

Un campo escalar asocia una cantidad escalar a cualquier punto del espacio. Su valor puede ser un número matemático o una cantidad física. El campo escalar representa físicamente una distribución espacial de una magnitud es calar. Matemáticamente un campo escalar es una función escalar de las coordenadas.

Muchos autores han propuesto alternativas de interés en las cuales se abordan las dificultades que ΛCDM no ha podido resolver hasta ahora. En el modelo de Campo Escalar se propone que los halos galácticos se forman de condensados de Bose-Einstein (BEC) de un campo escalar (SF) cuyo bosón tiene una masa ultra ligera del orden de $m \sim 10^{-22}$ eV. De este valor se sigue que la temperatura crítica de condensación ($T_c \sim 1/m^{5/3} \sim$TeV) es muy alta, por lo tanto, se forman semillas de Condensados de Bose-Einstein (BEC) en épocas tempranas en el Universo. Además, la longitud de Compton asociada al bosón es del orden $\lambda_C = 2\pi\bar{h}/m \sim \mathcal{O}(\text{kpc})$, que corresponde al tamaño de los halos de materia oscura de las galaxias en el Universo.

Por otra parte, las grandes estructuras del Universo se forman al igual que en el modelo ΛCDM, por lo que todas las predicciones correctas del modelo estándar están también presentes en el modelo SFDM.

En este modelo, las partículas escalares con esa masa ultra ligera son tales que sus propiedades ondulatorias evitan el problema del perfil CUSP y reducen el alto núumero de galaxias satélite por medio del principio de incertidumbre.

En el modelo de BEC, los halos de materia oscura pueden describirse como potenciales Newtonianos en el límite no relativista hechos de condensados de Bose-Einstein ultra ligeros y con una sola funcióon de onda asociada.

B.1. Aproximación hidrodinámica del campo escalar

En la llamada aproximación hidrodinámica, se hace una transformación para resolver las ecuaciones de Friedmann de manera analítica con la condición $H \ll m$. Se toma el potencial escalar como $V(\Phi) = m^2\Phi^2/2\bar{h}^2 + \lambda\Phi^4/4$. Así, para el bosón ultra ligero se tiene que $m \sim 10^{-22}$ eV. El campo escalar del background Φ_0 se expresa en términos de nuevas variables S y ρ_0, donde S es una cantidad constante en el fondo del Universo y ρ_0 será la densidad de energía del fluido también en esta región, así el campo se expresa como

$$\Phi_0 = \psi_0 e^{-imt/\bar{h}} + \psi_0^* e^{imt/\bar{h}} \tag{B.1}$$

donde

$$\psi_0(t) = \sqrt{\rho_0(t)} e^{iS/\bar{h}}. \tag{B.2}$$

Para el caso específico del potencial escalar definido, la ecuación de Klein-Gordon toma la siguiente forma

$$\Box\Phi + \frac{m^2c^2}{\hbar^2}\Phi + \frac{8\pi a_s m}{\hbar^2}|\Phi|^2\Phi = 0. \tag{B.3}$$

Se requiere evaluar un modelo perturbativo para estudiar las escalas pequeñas en modelos de evolución cosmológica, aplicando la teoría de perturbaciones a la ecuación (B.1) se obtiene

$$\Phi = \Phi_0(t) + \delta\Phi(\vec{x}, t), \tag{B.4}$$

que al insertar en la ecuación de Klein-Gordon (B.3), con $\dot{\phi} = 0$, se tiene que

$$\delta\ddot{\Phi} + 3H\delta\dot{\Phi} - \frac{1}{a^2}\nabla^2\delta\Phi + V_{,\Phi\Phi}\delta\Phi + 2V_{,\Phi}\phi = 0. \tag{B.5}$$

Donde ϕ es el potencial gravitacional (solo depende de la posición). El campo escalar perturbado $\delta\Phi$ en términos de Ψ puede expresarse de la siguiente manera

$$\delta\Phi = \Psi e^{-imt/\hbar} + \Psi^* e^{imt/\hbar}, \tag{B.6}$$

que se interpreta como una superposición de ondas. Usando el potencial $V = m^2\Phi^2/2\hbar^2 + \lambda\Phi^4/4$, la ecuación (B.5) se convierte en

$$-i\hbar(\dot{\Psi} + \frac{3}{2}H\Psi) + \frac{\hbar^2}{2m}(\Box\Psi + 9\lambda|\Psi|^2\Psi) + m\phi\Psi = 0. \tag{B.7}$$

Finalmente, la transformación de Madelung (Spiegel, E. A., 1980 [78]) tendrá la forma

$$\Psi = \sqrt{\hat{\rho}} e^{iS}, \tag{B.8}$$

donde Ψ será la función de onda del condensado, con $\hat{\rho} = \rho/m = \hat{\rho}(t, \mathbf{x})$ y $S = S(t, \mathbf{x})$. La función Ψ se separa en una fase real S y una amplitud real $\hat{\rho}$, mientras que se satisface la condición $|\Psi|^2 = \Psi\Psi^* = \hat{\rho}$.Sustituyendo en la ecuación (B.7), se obtiene

$$\dot{\hat{\rho}} + 3H\hat{\rho} - \frac{\hbar}{m}\hat{\rho}\Box S + \frac{\hbar}{a^2 m}\nabla S \nabla\hat{\rho} - \frac{\hbar}{m}\dot{\hat{\rho}}\dot{S} = 0, \tag{B.9}$$

y

$$\hbar\dot{S}/m + \omega\hat{\rho} + \phi + \frac{\hbar^2}{2m^2}\left(\frac{\Box\sqrt{\hat{\rho}}}{\sqrt{\hat{\rho}}}\right) + \frac{\hbar^2}{2a^2}[\nabla(S/m)]^2 - \frac{\hbar^2}{2}(\dot{S}/m)^2 = 0. \tag{B.10}$$

La ecuación (B.10) puede verse como un tipo de ecuación de la forma $\Phi + Q$ donde Q es el llamado potencial cuántico

$$Q = \frac{\hbar^2}{2m^2}\frac{\Box\sqrt{\hat{\rho}}}{\sqrt{\hat{\rho}}} \tag{B.11}$$

que puede describir una fuerza o algúun tipo de presión negativa de naturaleza cuántica.

Ureña-López (2019 [79]) ha mencionado en variadas ocasiones que esta aproximación debe tomarse con debido cuidado, ya que la equivalencia entre la aproximación de campo e hidrodinámica solo funciona en una dirección. No se puede recuperar una función de onda fiel de las soluciones para el campo de densidad ρ dado que la naturaleza cuántica desde el sistema hidrodinámico no se describe de manera discreta sin introducir constricciones extras, destacando entonces que aún deben considerarse nuevas aproximaciones para el modelo SFDM.

Apéndice C

Códigos: Árbol de decisión y Random Forest

La mayoría del código utilizado para el análisis del Capítulo 4 se encuentra en la página de github: ChJazhiel/ML_ICF. En este apéndice se hará un resumen del procedimiento de análisis para las simulaciones.

1. Importar utilidades:

```
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import pandas as pd
%matplotlib inline
```

2. Cargar y visualizar el dataset

```
dict_data = np.load('nbody_data.npz')
test_flags  = dict_data['test_flags']
test_hosts  = dict_data['test_hosts']
test_mass   = dict_data['test_mass']
test_labels = dict_data['test_labels']
test_input  = dict_data['test_input']
#test_snid   = dict_data['test_snid']
#test_labels = dict_data['test_labels']
print(test_mass)
print(np.sum(test_labels))
```

```
### load every dr_i from dataset ###
dr1 = pd.DataFrame(test_input[0], columns = ['dr1'])
dr2 = pd.DataFrame(test_input[1], columns = ['dr2'])
dr3 = pd.DataFrame(test_input[2], columns = ['dr3'])
dr4 = pd.DataFrame(test_input[3], columns = ['dr4'])
dr5 = pd.DataFrame(test_input[4], columns = ['dr5'])
dr6 = pd.DataFrame(test_input[5], columns = ['dr6'])
dr7 = pd.DataFrame(test_input[6], columns = ['dr7'])
dr8 = pd.DataFrame(test_input[7], columns = ['dr8'])
dr9 = pd.DataFrame(test_input[8], columns = ['dr9'])
dr10 = pd.DataFrame(test_input[9], columns = ['dr10'])
lbl = pd.DataFrame(test_labels, columns =['labels'])

#### visualize dataframe ###
df = pd.concat([dr1, dr2, dr3, dr4, dr5, dr6, dr7, dr8,
dr9, dr10, lbl],
axis=1, ignore_index=False, sort=False)
df.describe()

import seaborn as sns
#import seaborn as sns
%matplotlib inline
#%matplotlib qt
#matplotlib notebook

### visualize distributions ###
df1_0 = df[df.labels == 0]
df1_1 = df[df.labels == 1]

sns.distplot(df1_0['dr1'], kde=True, hist=False,
label='label = 0')
sns.distplot(df1_1['dr1'], kde=True, hist=False,
label='label = 1')
plt.axvline(0.012)
plt.legend()
sns.set(style="darkgrid")
plt.show()
```

3. Mezclar y escoger aleatoriamente los datos

```
df_0 = df.sort_values('labels').head(32456).sample(17300)
```

```
df_1 = df.sort_values('labels').tail(17544).sample(17300)
df_1

###MERGE THE DATASET
df_r = pd.concat([df_0, df_1])
### SHUFFLE THE DATASET EVENLY

shuffle_df = df_r.sample(frac = 1.0)

# Define a size for your train set
train_size = int(0.8 * len(df_r))


train_set = shuffle_df[:train_size] # Split your dataset
test_set = shuffle_df[train_size:]
```

4. Escoger los atributos y clases del dataframe

```
X = shuffle_df.drop(['labels'], axis = 1)
y = shuffle_df.labels
X_train = train_set.drop(['labels'], axis = 1)
X_test = test_set.drop(['labels'], axis= 1)
y_train = train_set.labels
y_test = test_set.labels
```

5. Crear el modelo de árbol de decisión y random forest

```
dt = DecisionTreeClassifier(criterion='entropy',
max_depth=8, class_weight='balanced')
dt = dt.fit(X_train, y_train)

#Predict the response for test dataset
ypred = dt.predict(X_test)
#Model accuracy, how often is the classifier correct

print('Traning and Testing on raw data, all features \n');

print("Accuracy:", metrics.accuracy_score(y_test, ypred))

for i, score_tree in enumerate(cross_val_score(dt, X, y, cv = 10)):
    print('Decision tree accuracy for the
```

```
    %d score: %0.2f' % (i, score_tree))
score_tree=cross_val_score(dt, X, y, cv=10)
#score_tree
cv_scores = []
print("Decision Tree Accuracy:
%0.2f (+/- %0.2f)" % (score_tree.mean(),
score_tree.std() * 2 ))
cv_score = score_tree.mean()
cv_scores.append(cv_score)



###### RANDOM FOREST CLASSIFIER

rf = RandomForestClassifier(criterion='entropy', max_depth=8,
n_iterations = 2000, n_jobs = -1,
class_weight='balanced')
rf = rf.fit(X_train, y_train)

#Predict the response for test dataset
ypred_rf = rf.predict(X_test)
#Model accuracy, how often is the classifier correct

print('Traning and Testing on raw data, all features \n');

print("Accuracy:", metrics.accuracy_score(y_test, ypred_rf))

for i, score_rf in enumerate(cross_val_score(rf, X, y, cv = 3)):
    print('Random Forest accuracy for the
    %d score: %0.2f' % (i, score_tree))
score_rf=cross_val_score(rf, X, y, cv=3)
print("Random Forest Accuracy:
%0.2f (+/- %0.2f)" % (score_rf.mean(),
score_rf.std() * 2 ))
```

Bibliografía

[1] E. Hubble, "A Relation between Distance and Radial Velocity among Extra-Galactic Nebulae," *Proceedings of the National Academy of Science*, vol. 15, pp. 168–173, Mar. 1929.

[2] B. Schutz, *The Einstein field equations*, p. 184–202. Cambridge University Press, 2 ed., 2009.

[3] B. Schutz, *Perfect fluids in special relativity*, p. 84–110. Cambridge University Press, 2 ed., 2009.

[4] Ø. Grøn and S. Hervik, *Einstein's General Theory of Relativity: With Modern Applications in Cosmology*. Springer New York, 2007.

[5] F. Zwicky, "On the Masses of Nebulae and of Clusters of Nebulae," , vol. 86, p. 217, Oct. 1937.

[6] V. C. Rubin and J. Ford, W. Kent, "Rotation of the Andromeda Nebula from a Spectroscopic Survey of Emission Regions," , vol. 159, p. 379, Feb. 1970.

[7] D. N. Spergel, R. Bean, O.é@, M. R. Nolta, C. L. Bennett, J. Dunkley, G. Hinshaw, N. Jarosik, E. Komatsu, L. Page, H. V. Peiris, L. Verde, M. Halpern, R. S. Hill, A. Kogut, M. Limon, S. S. Meyer, N. Odegard, G. S. Tucker, J. L. Weiland , E. Wollack, and E. L. Wright, "Three-Year Wilkinson Microwave Anisotropy Probe (WMAP) Observations: Implications for Cosmology," , vol. 170, pp. 377–408, June 2007.

[8] P. A. R. Ade, N. Aghanim, C. Armitage-Caplan, M. Arnaud, M. Ashdown, F. Atrio-Barandela, J. Aumont, C. Baccigalupi, A. J. Banday, and et al., "Planck2013 results. xv. cmb power spectra and likelihood," *Astronomy Astrophysics*, vol. 571, p. A15, Oct 2014.

[9] M. R. Blanton, M. A. Bershady, B. Abolfathi, F. D. Albareti, C. Allende Prieto, A. Almeida, J.ía@, F. Anders, S. F. Anderson, B. Andrews, and

et al., "Sloan Digital Sky Survey IV: Mapping the Milky Way, Nearby Galaxies, and the Distant Universe," , vol. 154, p. 28, July 2017.

[10] B. Ryden, *Introduction to cosmology*. 2003.

[11] P. J. E. Peebles, *Principles of Physical Cosmology*. 1993.

[12] M. Sasaki, "Large Scale Quantum Fluctuations in the Inflationary Universe," *Progress of Theoretical Physics*, vol. 76, pp. 1036–1046, 11 1986.

[13] A. H. Guth, "Inflationary universe: A possible solution to the horizon and flatness problems," , vol. 23, pp. 347–356, Jan. 1981.

[14] P. J. Peebles and B. Ratra, "The cosmological constant and dark energy," *Reviews of Modern Physics*, vol. 75, pp. 559–606, Apr. 2003.

[15] S. Dodelson, *Modern cosmology*. 2003.

[16] M. Tegmark, M. A. Strauss, M. R. Blanton, K. Abazajian, S. Dodelson, H. Sandvik, X. Wang, D. H. Weinberg, I. Zehavi, N. A. Bahcall, F. Hoyle, and et al., "Cosmological parameters from SDSS and WMAP," , vol. 69, p. 103501, May 2004.

[17] M. L. Norman, "Simulating Galaxy Clusters," *arXiv e-prints*, p. arXiv:1005.1100, May 2010.

[18] C.-P. Ma and E. Bertschinger, "Cosmological Perturbation Theory in the Synchronous and Conformal Newtonian Gauges," , vol. 455, p. 7, Dec. 1995.

[19] T. M. C. Abbott, S. Allam, P. Andersen, C. Angus, J. Asorey, A. Avelino, S. Avila, B. A. Bassett, K. Bechtol, G. M. Bernstein, E. Bertin, D. Brooks, D. Brout, P. Brown, D. L. Burke, J. Calcino, A. Carnero Rosell, D. Carollo, and et al.

[20] S. D. M. White, "Formation and Evolution of Galaxies: Les Houches Lectures," *arXiv e-prints*, pp. astro–ph/9410043, Oct. 1994.

[21] H. Mo, F. C. van den Bosch, and S. White, *Galaxy Formation and Evolution*. 2010.

[22] W. H. Press and P. Schechter, "Formation of Galaxies and Clusters of Galaxies by Self-Similar Gravitational Condensation," , vol. 187, pp. 425–438, Feb. 1974.

[23] R. K. Sheth, H. J. Mo, and G. Tormen, "Ellipsoidal collapse and an improved model for the number and spatial distribution of dark matter haloes," , vol. 323, pp. 1–12, May 2001.

[24] M. S. Warren, K. Abazajian, D. E. Holz, and L. Teodoro, "Precision Determination of the Mass Function of Dark Matter Halos," , vol. 646, pp. 881–885, Aug. 2006.

[25] J. A. Peacock, "Testing anthropic predictions for λ and the cosmic microwave background temperature," *Monthly Notices of the Royal Astronomical Society*, vol. 379, no. 3, pp. 1067–1074, 2007.

[26] J. L. Tinker, B. E. Robertson, A. V. Kravtsov, A. Klypin, M. S. Warren, G. Yepes, and S.öber@, "The Large-scale Bias of Dark Matter Halos: Numerical Calibration and Model Tests," , vol. 724, pp. 878–886, Dec. 2010.

[27] S. G. Murray, C. Power, and A. S. G. Robotham, "HMFcalc: An online tool for calculating dark matter halo mass functions," *Astronomy and Computing*, vol. 3, pp. 23–34, Nov. 2013.

[28] J. Kremer, K. Stensbo-Smidt, F. Gieseke, K. Steenstrup Pedersen, and C. Igel, "Big Universe, Big Data: Machine Learning and Image Analysis for Astronomy," *arXiv e-prints*, p. arXiv:1704.04650, Apr. 2017.

[29] V. Springel, S. D. M. White, A. Jenkins, C. S. Frenk, N. Yoshida, L. Gao, J. Navarro, R. Thacker, D. Croton, J. Helly, J. A. Peacock, S. Cole, P. Thomas, H. Couchman, A. Evrard, J. Colberg, and F. Pearce, "Simulations of the formation, evolution and clustering of galaxies and quasars," , vol. 435, pp. 629–636, June 2005.

[30] R. E. Angulo, V. Springel, S. D. M. White, A. Jenkins, C. M. Baugh, and C. S. Frenk, "Scaling relations for galaxy clusters in the Millennium-XXL simulation," , vol. 426, pp. 2046–2062, Nov. 2012.

[31] J. Kim, C. Park, G. Rossi, S. M. Lee, and I. Gott, J. Richard, "The New Horizon Run Cosmological N-Body Simulations," *Journal of Korean Astronomical Society*, vol. 44, pp. 217–234, Dec. 2011.

[32] J.-M. Alimi, V. Bouillot, Y. Rasera, V. Reverdy, P.-S. Corasaniti, I. Balmes, S. Requena, X. Delaruelle, and J.-N. Richet, "DEUS Full Observable {}CDM Universe Simulation: the numerical challenge," *arXiv e-prints*, p. arXiv:1206.2838, June 2012.

[33] M. Vogelsberger, S. Genel, V. Springel, P. Torrey, D. Sijacki, D. Xu, G. Snyder, D. Nelson, and L. Hernquist, "Introducing the Illustris Project: simulating the coevolution of dark and visible matter in the Universe," , vol. 444, pp. 1518–1547, Oct. 2014.

[34] F. Reif and H. L. Scott, "Fundamentals of statistical and thermal physics," *American Journal of Physics*, vol. 66, no. 2, pp. 164–167, 1998.

[35] P. Bodenheimer, G. P. Laughlin, M. Rozyczka, T. Plewa, H. W. Yorke, and H. W. Yorke, *Numerical methods in astrophysics: an introduction*. Taylor & Francis, 2006.

[36] J. Chacon, J. A. Vazquez, and R. Gabbasov, "Dark Matter with N-Body Numerical Simulations," *arXiv e-prints*, p. arXiv:2006.10203, June 2020.

[37] L. Moscardini and K. Dolag, "Cosmology with numerical simulations," in *Dark Matter and Dark Energy*, pp. 217–237, Springer, 2011.

[38] V. Springel, "The cosmological simulation code GADGET-2," , vol. 364, pp. 1105–1134, Dec. 2005.

[39] J. Zhang, Y.-L. Sming Tsai, J.-L. Kuo, K. Cheung, and M.-C. Chu, "Ultralight Axion Dark Matter and Its Impact on Dark Halo Structure in N-body Simulations," , vol. 853, p. 51, Jan. 2018.

[40] J. J. Monaghan and J. C. Lattanzio, "A refined particle method for astrophysical problems," , vol. 149, pp. 135–143, Aug. 1985.

[41] S. D. M. White, G. Efstathiou, and C. S. Frenk, "The amplitude of mass fluctuations in the universe," , vol. 262, pp. 1023–1028, June 1993.

[42] N. A. Bahcall, J. P. Ostriker, S. Perlmutter, and P. J. Steinhardt, "The Cosmic Triangle: Revealing the State of the Universe," *Science*, vol. 284, p. 1481, May 1999.

[43] A. Lewis, A. Challinor, and A. Lasenby, "Efficient Computation of Cosmic Microwave Background Anisotropies in Closed Friedmann-Robertson-Walker Models," , vol. 538, pp. 473–476, Aug. 2000.

[44] S. Colombi, A. Jaffe, D. Novikov, and C. Pichon, "Accurate estimators of power spectra in N-body simulations," *Monthly Notices of the Royal Astronomical Society*, vol. 393, pp. 511–526, 02 2009.

[45] P. S. Behroozi, R. H. Wechsler, and H.-Y. Wu, "The ROCKSTAR Phase-space Temporal Halo Finder and the Velocity Offsets of Cluster Cores," , vol. 762, p. 109, Jan. 2013.

[46] M. J. Turk, B. D. Smith, J. S. Oishi, S. Skory, S. W. Skillman, T. Abel, and M. L. Norman, "yt: A Multi-code Analysis Toolkit for Astrophysical Simulation Data," , vol. 192, p. 9, Jan. 2011.

[47] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*. USA: Prentice Hall Press, 3rd ed., 2009.

[48] A. Gron, *Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. O'Reilly Media, Inc., 1st ed., 2017.

[49] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. The MIT Press, 2016.

[50] D. W. Hosmer and S. Lemeshow, *Applied logistic regression*. John Wiley and Sons, 2000.

[51] G. Louppe, L. Wehenkel, A. Sutera, and P. Geurts, "Understanding variable importances in forests of randomized trees," vol. 26, 12 2013.

[52] J. R. Quinlan, "Induction of decision trees," *Machine learning*, vol. 1, no. 1, pp. 81–106, 1986.

[53] C. E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, 1948.

[54] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, pp. 5–32, Oct. 2001.

[55] H. Deng, G. Runger, and E. Tuv, "Bias of importance measures for multi-valued attributes and solutions," in *Artificial Neural Networks and Machine Learning, ICANN 2011 - 21st International Conference on Artificial Neural Networks, Proceedings*, no. PART 2 in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), pp. 293–300, June 2011. 21st International Conference on Artificial Neural Networks, ICANN 2011 ; Conference date: 14-06-2011 Through 17-06-2011.

[56] T. Fawcett, "Introduction to roc analysis," *Pattern Recognition Letters*, vol. 27, pp. 861–874, 06 2006.

[57] L. Rokach and O. Z. Maimon, *Data mining with decision trees: theory and applications*, vol. 69. World scientific, 2008.

[58] L. Breiman, J. Friedman, C. Stone, and R. Olshen, *Classification and Regression Trees*. The Wadsworth and Brooks-Cole statistics-probability series, Taylor & Francis, 1984.

[59] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning: With Applications in R.* Springer Publishing Company, Incorporated, 2014.

[60] J. F. Navarro, C. S. Frenk, and S. D. M. White, "The Structure of Cold Dark Matter Halos," , vol. 462, p. 563, May 1996.

[61] Planck Collaboration, N. Aghanim, Y. Akrami, M. Ashdown, J. Aumont, C. Baccigalupi, and et al., "Planck 2018 results. VI. Cosmological parameters," , vol. 641, p. A6, Sept. 2020.

[62] S. Dodelson and A. P. L. . 1941-1969)., *Modern Cosmology.* Elsevier Science, 2003.

[63] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[64] G. Louppe, "Understanding Random Forests: From Theory to Practice," *arXiv e-prints*, p. arXiv:1407.7502, July 2014.

[65] B. Moore, T. Quinn, F. Governato, J. Stadel, and G. Lake, "Cold collapse and the core catastrophe," *Monthly Notices of the Royal Astronomical Society*, vol. 310, pp. 1147–1152, 12 1999.

[66] A. Klypin, A. V. Kravtsov, O. Valenzuela, and F. Prada, "Where Are the Missing Galactic Satellites?," , vol. 522, pp. 82–92, Sept. 1999.

[67] V. Sahni and L. Wang, "New cosmological model of quintessence and dark matter," *Phys. Rev. D*, vol. 62, p. 103517, Oct 2000.

[68] W. Hu, R. Barkana, and A. Gruzinov, "Fuzzy cold dark matter: The wave properties of ultralight particles," *Phys. Rev. Lett.*, vol. 85, pp. 1158–1161, Aug 2000.

[69] T. Matos and L. Urena-Lopez, "Quintessence and scalar dark matter in the universe," *Class. Quant. Grav.*, vol. 17, pp. L75–L81, 2000.

[70] M. T. Hagan, H. B. Demuth, and M. Beale, *Neural Network Design*. USA: PWS Publishing Co., 1997.

[71] X. Kang, B. Song, and F. Sun, "A deep similarity metric method based on incomplete data for traffic anomaly detection in iot," *Applied Sciences*, vol. 9, p. 135, 01 2019.

[72] A. Peel, F. Lalande, J.-L. Starck, V. Pettorino, J. Merten, C. Giocoli, M. Meneghetti, and M. Baldi, "Distinguishing standard and modified gravity cosmologies with machine learning," , vol. 100, p. 023508, July 2019.

[73] B. Buncher and M. Carrasco Kind, "Probabilistic cosmic web classification using fast-generated training data," , vol. 497, pp. 5041–5060, July 2020.

[74] J. M. Bardeen, J. R. Bond, N. Kaiser, and A. S. Szalay, "The Statistics of Peaks of Gaussian Random Fields," , vol. 304, p. 15, May 1986.

[75] S. De and R. A. C. Croft, "Peaks in the cosmological density field: sensitivity to initial power spectrum, redshift distortions and galaxy halo occupation," *Monthly Notices of the Royal Astronomical Society*, vol. 382, pp. 1591–1600, 11 2007.

[76] J. Chacón, *Modelos de Materia Oscura: Una Perspectiva Numérica*. Dec 2018. Escuela Superior de Física y Matemáticas, B.S. Thesis, http://pelusa.fis.cinvestav.mx/tmatos/CV/3_RecursosH/Lic/Jazhiel_ESFM.pdf.

[77] Y. Li, Y. Ni, R. A. C. Croft, T. Di Matteo, S. Bird, and Y. Feng, "AI-assisted super-resolution cosmological simulations," *arXiv e-prints*, p. arXiv:2010.06608, Oct. 2020.

[78] E. Spiegel, "Fluid dynamical form of the linear and nonlinear schrödinger equations," *Physica D: Nonlinear Phenomena*, vol. 1, no. 2, pp. 236 – 240, 1980.

[79] L. A. Ureña-López, "Brief review on scalar field dark matter models," *Frontiers in Astronomy and Space Sciences*, vol. 6, p. 47, 2019.