Universitat Oberta de Catalunya

uoc.edu

Desarrollo de un sistema de Machine Learning para obtener modelos de unión a Factores de Transcripción en datos ChIP-seq

Sara Álvarez González
Máster universitario en Bioinformática y bioestadística UOC-UB
Genómica comparativa microbiana

Nombre Consultor/a: Ivan Erill Sagales
Nombre Profesor/a responsable de la assignatura: Marc Maceira Duch

Fecha Entrega: 08/06/2021

FICHA DEL TRABAJO FINAL

Título del trabajo:	*Development of an ML system for obtaining TF-binding models on ChIP-seq datasets*
Nombre del autor:	*Sara Álvarez González*
Nombre del consultor/a:	*Ivan Erill Sagales*
Nombre del PRA:	*Marc Maceira Duch*
Fecha de entrega (mm/aaaa):	06/2021
Titulación:	*Máster universitario en Bioinformática y bioestadística UOC-UB*
Área del Trabajo Final:	*Genómica comparativa microbiana*
Idioma del trabajo:	*Castellano*
Número de créditos:	*15*
Palabras clave	*Protein binding, Machine Learning, Transcription Factor*

Resumen del Trabajo (máximo 250 palabras): *Con la finalidad, contexto de aplicación, metodología, resultados i conclusiones del trabajo.*

Este trabajo tiene como objetivo obtener un modelo predictivo capaz de identificar las regiones en las que un Factor de Transcripción (FT) se acoplará al ADN. Los datos empleados son extraídos a partir de los resultados de la técnica de ChIP-seq. Dicha técnica es capaz de reconocer las secuencias en las que estos FTs se han acoplado. Identificar la secuencia exacta a la que los FTs se han unido es una tarea difícil en ciertas condiciones moleculares.

Técnicas computacionales basadas en Machine Learning (ML), una rama dentro de la Inteligencia Artificial que centra sus esfuerzos en el desarrollo de modelos predictivos es considerada una herramienta analítica útil para este tipo de problemas. Estas técnicas son capaces de extraer los patrones no lineales de los datos a partir de un gran conjunto de ejemplos. Además, trabajos previos han desarrollado descriptores matemáticos capaces de convertir la secuencia primaria de ADN en matrices de datos numéricos, facilitando en gran medida el uso de algoritmos de ML.

En este proyecto se presenta un conjunto de modelos que han sido entrenados para la predicción de las regiones de unión del FT Gcra en la especie bacteriana *Brevundimonas subvibrioides*. Los resultados aquí presentados muestran un alto rendimiento en la predicción de estas regiones gracias al uso de descriptores tanto estructurales como de composición del ADN.

Abstract (in English, 250 words or less):

The aim of this work is to obtain a predictive model capable of identifying the regions where a Transcription Factor (TF) will bind to DNA. The data used are extracted from the results of the ChIP-seq technique. This technique is able to recognize the sequences in which these TFs have been coupled. Identifying the precise sequence to which the TFs are attached is a difficult task under certain molecular conditions.

Computational techniques based on Machine Learning (ML), a branch within Artificial Intelligence that focuses its efforts on the development of predictive models is considered a useful analytical tool for this type of problems. These techniques are capable of extracting nonlinear patterns from data from a large set of examples. In addition, previous work has developed mathematical descriptors capable of converting the primary DNA sequence into numerical data matrices, greatly facilitating the use of ML algorithms.

In this project we present a set of models that have been trained for the prediction of TF Gcra binding regions in the bacterial species Brevundimonas subvibrioides. The results presented here show a high performance in the prediction of these regions thanks to the use of both structural and DNA composition descriptors.

A un biólogo al que le encanta la zoología,

Contenido

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

1. Resumen

La tarea de predicción del acoplamiento de Factores de Transcripción (FT) a ciertas regiones del ADN en organismos bacterianos puede resultar de gran dificultad en ciertas ocasiones, cuando los motivos en las secuencias no sean del todo claros o haya ciertos parámetros espaciales de las secuencias en juego que clásicamente no se contemplaban. Debido a los avances en las técnicas de Inteligencia Artificial para generar modelos de Machine Learning (ML) de predicción que son capaces de hacer inferencias a partir de datos más complejos que programas que actualmente solo tienen en cuenta los nucleótidos de una secuencia de manera aislada.

El utilizar otro tipo de descriptores extraídos a través de lugares de acoplamientos encontrados por técnicas como ChIP-seq que también tengan en cuenta la posición relativa al genoma de referencia, tales como los que generan librerías como DNAshapeR, o el conteo nucleótidos de cada tipo en las secuencias sin importar el orden para estudiar la composición de manera holística, como se genera a través de descriptores k-mers, es la base para desarrollar modelos de ML llevados a cabo en este proyecto.

A partir de los datos de entrada, o *peaks*, donde sabemos que un FT se ha unido en el genoma, se han generado dos tipos de replicados para crear dos datasets y poder comparalos, uno intentando conservar lo máximo la estructura biológica de la secuencia original, y los segundos creando pseudo-replicados aleatorios a partir de trímeros de la secuencia base.

Aplicando los modelos de ML sobre un total de 34 conjuntos de datos con diferentes descriptores, se han aplicado las técnicas de ML de Random Forest (RF) y Glmenet con una validación cruzada interna para generar los hiperparámetros, y una segunda validación cruzada externa de 5 repeticiones y un 10-fold-CV.

Se observa que en ambos tipos de datos dos descriptores van a ser significativamente mejores a la hora de la predicción (HelT y tetrámeros), y que las variables de más importancia cuando estudiamos los tetrámeros van a estar contenidas total o parcialmente con el motivo encontrado en el artículo del que se extrajeron los datos, dando robustez a las predicciones obtenidas.

También, se concluye que la creación de los replicados que se generen a partir de las secuencias iniciales (*peaks*) van a ser clave, ya que los creados de manera más aleatoria facilitaron al modelo de ML a obtener mejores rendimientos, frente a los que se parecieron más biológicamente.

Se trata de un proyecto de gran interés biológico, debido a que a nivel de predicción de estas regiones de acoplamiento queda mucho por descubrir, ya que en general se ha visto que la búsqueda clásica tiene ciertas limitaciones cuando es preciso tener más factores en cuenta del genoma que simplemente su composición de nucleótidos de manera individual.

2. Introducción

2.1 Contexto y justificación del Trabajo

Este trabajo tiene como principal problemática a enfrentar, el diseño de modelos de Machine Learning (ML) que permitan el análisis de bases de datos de ChIP-seq que tengan en cuenta no solo los lugares de acoplamiento de los Factores de Transcripción (FT), sino que también incluyan una comparativa tanto genómica como estructural del ADN con el que se está trabajando, de manera más holística. Teniendo como objetivo principal el lograr obtener un modelo de acoplamiento capaz de explicar los resultados observados en los resultados de ChIP-seq.

Los factores de transcripción (FT) son un tipo proteínas que poseen una función biológica esencial a la hora de la creación de nuevas proteínas a partir de la secuencia de ADN [1]. En concreto, los FT van a ser claves en el proceso de transcripción, en el que la doble hélice de ADN se convierte en una secuencia de ARN. Los lugares a los que este tipo de proteínas se van a unir en el ADN es conocido como regiones promotoras. Ser conocedor de los lugares en los que se van a unir los FT y qué procesos celulares van a ser desencadenados resulta de un estudio de gran impacto e interés biológico, especialmente en organismos bacterianos [2].

Generalmente, la predicción de dichos lugares de unión suele ser modelado y predicho mediante la construcción de matrices de pesos relacionales, que a partir de un vector lineal de nucleótidos, o también llamado motivo, va a encontrar los lugares donde será más probable que se vaya a acoplar el FT [3]. Sin embargo, no todos los motivos están tan claros, y existen ciertos FT cuyas regiones de acoplamiento no son tan sencillas de predecir [4]. Esto puede ser debido a diversos motivos, tanto como porque la estructura del ADN presente un co-factor, como porque la estructura del ADN sea clave y no se puedan analizar vectores lineales sin el contexto y la posición relativa dentro del mismo ADN.

2.2 Objetivos del Trabajo

2.2.1 Objetivos generales

- Diseño de uno o varios modelos de ML que permitan analizar bases de datos de ChIP-seq.
- Predicción de lugares de acoplamiento para FT en el ADN en bacterias.

2.2.2 Objetivos específicos

- Identificar experimentos adicionales de ChIP-seq para factores de transcripción que hayan conducido a resultados no concluyentes en cuanto al modelo de unión de los TF (es decir, el motivo). Estudios cuyo estudio descontextualizado de los vectores de nucleótidos lineales no hayan podido resolver el problema.
- Recoger y estandarizar los datos de ChIP-seq disponibles gratuitamente (GEO, ArrayExpress) de las bases de datos que se emplearán en el pipeline de ML a desarrollar.
- Implementar la detección de ortólogos para los genes codificadores de proteínas asociados a los picos de ChIP-seq y obtener sus respectivas regiones upstream.

- Utilizar DNAshapeR para generar vectores de características estructurales para estas secuencias.
- Utilizar algoritmos convencionales de descubrimiento de motivos y análisis de hexámeros para extraer características de secuencia para estas secuencias.
- Identificar paradigmas de ML y métodos de codificación adecuados para modelar los promotores regulados por TF.
- Entrenar diferentes paradigmas de ML en el conjunto de datos ampliado (estructura+genómica comparativa), utilizando el valor experimental de ChIP-seq como resultado esperado
- Comparar el rendimiento de los modelos con los motivos tradicionales/enfoques publicados, tales como los del programa más empleado (MEME).
- Realizar un análisis de características (por ejemplo, perturbación) de los modelos entrenados para identificar las características clave.

2.3 Enfoque y método seguido

Con vistas a desarrollar un sistema de aprendizaje automático que lograran predecir los lugares de acoplamiento para FT en el ADN en bacterias, se ha comenzado con una primera aproximación en la que se han intentado sacar los datos de los lugares de acoplamiento (*peaks*) encontrados en los experimentos de manera manual, pasando por todas las etapas del preprocesado de datos biológicos. Ante la falta de resultados satisfactorios, ya que no se consiguió extraer los datos de esta manera, se decidió proceder con datos de *peaks* ya preprocesados biológicamente extraídos de un estudio de Adhikari et al. [5] que proporcionaba las posiciones en el genoma en las que se localizaban para un organismo y FT en concreto.

Posteriormente, a partir de las posiciones en el genoma que indicaban los *peaks* antes nombrados, se extrajeron las secuencias correspondientes a los mismos, y se crearon replicados de las mismas que guardaban cierto paralelismo biológico, pero lo suficientemente diferentes para formar la base de datos negativa para nuestro problema, marcando como secuencias positivas a las regiones donde se produjo la unión de los factores de transcripción en el ADN.

Los siguientes pasos constaron en el diseño de un pipeline de ML que fue desde la extracción de las características, hasta la aplicación de los modelos diseñados. Este pipeline se fue adaptando de manera progresiva a los datos con los que se vaya trabajando y automatizando para que sea aplicable a cualquier entrada de datos que busque llevar a cabo la misma predicción.

A lo largo de toda esta estrategia de trabajo, el pipeline de ML se ha ido basando en los diferentes puntos que conforman los objetivos específicos de este trabajo, sirviendo como guía en su desarrollo. La perspectiva adoptada en cuanto al desarrollo de este proyecto está basada en la hipótesis de que el empleo de técnicas de aprendizaje automático para lograr encontrar motivos biológicos va a permitir encontrar motivos biológicos en el genoma llevando a cabo inferencias más complejas que simplemente la estructura de ADN, como se acostumbra a hacer actualmente con programas especializados como MEME.

2.4 Planificación del Trabajo

2.4.1 Tareas

1. Búsqueda de literatura científica: precisa para ser conocedora del estado del arte del campo, conocer las técnicas de búsqueda de motivos que hoy en día es lo que se está utilizando y encontrar los artículos que contengan la recogida de los datos biológicos que posteriormente utilizaré para hacer las predicciones y ajustar mi pipeline.

2. Preprocesado biológico: debido a que los datos que se encuentran en la mayoría de los papers de acoplamiento de FT no está bien definido cómo se ha hecho el preprocesado de los datos una vez se le aplica el ChIP-seq, se tratará de extraer los datos de peak calling (lugares en los que se ha unido la proteína en el ADN) a través de un pipeline diseñado para eso en el programa de galaxy.org.

3. Extracción automática de los datos para generar el dataset: sea o no factible el hecho de extraer los lugares en los que se va a acoplar la proteína, se diseñará un script capaz de extraer automáticamente las secuencias en las que se encontrará acoplada la proteína y en las que no, generando un dataset sobre el que trabajar.

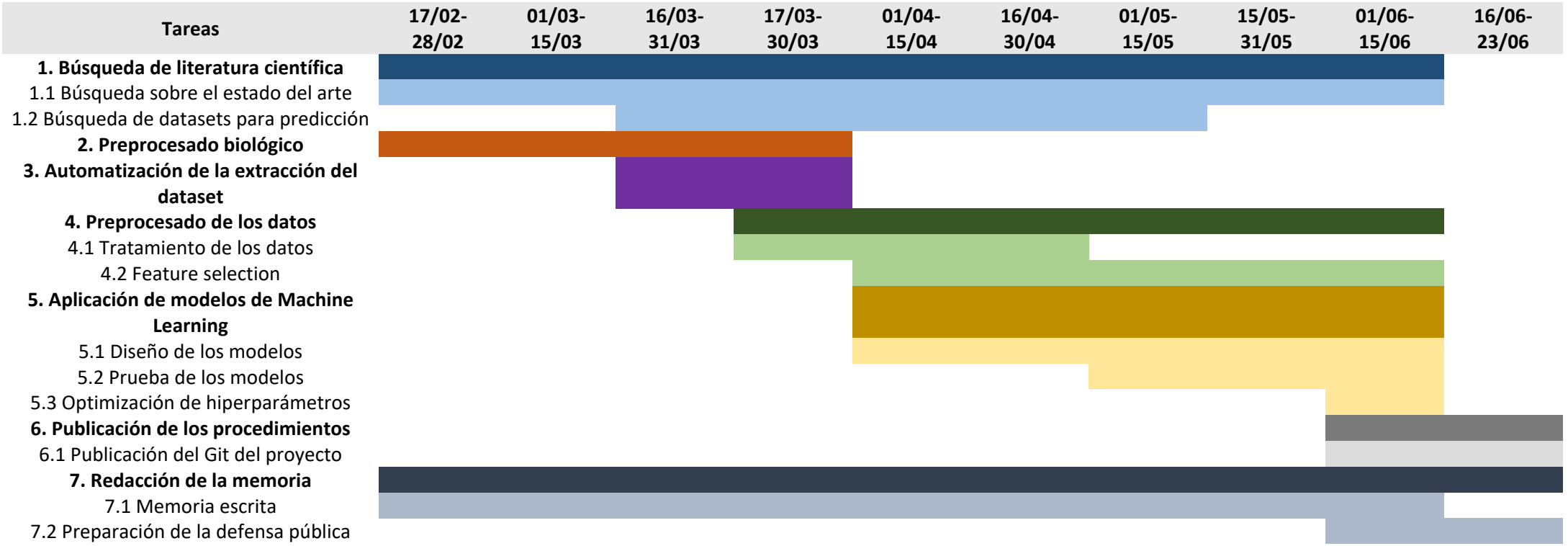
4. Feature selection y preprocesado de los datos: una vez tenemos el dataset, procederemos a encontrar las características (feature selection, FS) con las que vamos a trabajar para realizar las predicciones. Este paso es esencial, debido a que es aquí el momento en el que necesitaremos hacer un análisis exhaustivo de los datos que tenemos para intentar que las predicciones posteriores predigan exactamente lo que queremos, buscando evitar overfittings y underfittings que en un primer lugar sean claros e intentar una predicción lo más libre de ruido desde el principio.

5. Aplicación de modelos de Machine Leaning: en este paso se llevará a cabo la aplicación de diferentes modelos de ML que puedan ser más o menos adecuados para hacer la predicción final. Probando en un principio con modelos más sencillos, tipo SVM, hasta modelos de índole más compleja como Redes de neuronas recurrentes (RNN) o modelos de Markov que permitan una predicción que tenga en cuenta esas condiciones de estructura del ADN más complicadas de calcular

2.4.2 Hitos

1. Consecución de un archivo en el que se hayan calculado los peak calling a partir de un estudio de ChIP-seq.
2. Creación de un script que permita la automatización de la extracción de datos para crear un dataset.
3. Aproximación convencional de un pipeline de ML con técnicas clásicas, conocidas y fáciles de implementar, además de una primera selección de las que podrían ser las características más importantes con las que trabajar.
4. Aplicación de técnicas estadísticas más complejas para el FS y así conseguir un mejor rendimiento con los modelos.
5. Aplicación de modelos de ML más complejos.
6. Optimización de los modelos de ML empleados
7. Creación de un git que permita la reproducibilidad del modelo creado

. 2.4.3 Calendario



2.4.4 Breve sumario de contribuciones y productos obtenidos

Entregables para calificación del trabajo:

- Memoria del trabajo en formato .pdf.
- Video en formato .wmv de la presentación de la memoria.
- Transparencias de la presentación en formato .ppt.

Resultados del estudio:

- Git con el código empleado en: https://github.com/saraalgo/binding_protein

2.5 Breve descripción de los otros capítulos de la memoria

Este proyecto está dividido en 8 apartados principales. Tras haber resumido e introducido el trabajo en los dos primeros apartados, la sección 3 de Estado del Arte hace un repaso por la literatura de relevancia para el tema en el que se enmarca el trabajo. En la sección 4 de Fundamentos, se introducen conceptos que posteriormente se utilizarán y que van a ser claves para entender el desarrollo. La sección 5 de Metodología presenta los pasos a seguir que se han llevado a cabo, clarificando conceptos sobre las técnicas empleadas. En canto a la sección 6 de Resultados, ahí se mostrará lo obtenido al seguir el *workflow* presentado en la sección anterior, llevando a cabo los dos algoritmos presentados sobre diferentes conjuntos de datos. Las últimas secciones de discusión y conclusiones cerrarán el proyecto y lo obtenido en él. De manera adicional, se cuenta al final con un glosario de términos empleados y la bibliografía empleada.

3. Estado del arte

3.1 Factores de transcripción

Desde el punto de vista biológico, el proceso de creación de proteínas comienza cuando el ADN es leído por una secuencia de proteínas, como la holoenzima ARN-polimerasa, que transformará la doble hélice del ADN en una única secuencia de ARN. Tras ello, la información codificada en el ARN es leída por lo ribosomas y traducida en proteínas. Los factores de transcripción (FT) son otro tipo proteínas que intervienen en este proceso de conversión, o transcripción. Una característica que los hace distintivos es que poseen unos dominios de unión al ADN que les dota de la capacidad de unirse a secuencias específicas del ADN denominadas secuencias potenciadoras o promotoras [1]. En bacterias, algunos FT se unen a una secuencia promotora de ADN cerca del sitio de inicio de la transcripción y ayudan a formar o bloquean el complejo de iniciación de la transcripción. La regulación de la transcripción es la forma más común de control de los genes, siendo críticos ya que permiten una expresión única de cada gen en diferentes tipos de células y durante el desarrollo [4].

3.2 Acoplamiento de proteínas

En este contexto, a pesar del gran interés que existe por comprender cómo los FTs controlan la expresión génica, sigue siendo por el momento un reto determinar cómo se especifican los sitios de unión genómica precisos de los FTs y cómo la unión de los FTs se van a relacionar finalmente con la regulación de la transcripción [6]. Cada FT se sabe que se va a unir a una variedad de lugares del ADN con las que tenga una afinidad específica de la secuencia [7].

3.3 Motivos de acoplamiento

El descubrimiento de motivos en secuencias biológicas podría ser definido como la búsqueda de elementos con secuencias cortas similares (lo que se conocería como motivo) que compartan un conjunto de nucleótidos o secuencias de proteínas comunes con una función biológica común, generalmente de entre 6-20 bp. Así pues, la identificación de elemento reguladores en secuencias de nucleótidos, como los anteriormente mencionados lugares de acoplamiento de los FT, son uno de los problemas más estudiados en el campo de la genómica, tanto por su importancia biológica, como por su dificultad de predicción en el campo de la bioinformática [8].

3.4 Localización de acoplamiento de FT en el genoma

A lo largo de los años, se han ido desarrollando diferentes métodos in vitro que tienen como objetivo localizar los lugares en el ADN en el que se observe un complejo proteo-nucleico y que, por tanto, se haya producido este acoplamiento de proteína por parte del FT.

Una de estas técnicas es la conocida como Electrophoresis Mobility Shift Assay (EMSA) [9], la cual a lo largo de los años se ha convertido en uno de los protocolos estándar para determinar el potencial de acoplamiento ADN-proteína. Esta técnica suele requerir el marcaje radiactivo de una sonda de ADN que se incuba con la proteína de interés. La mezcla se separa en un gel de poliacrilamida que posteriormente se expone a una película radiográfica. La presencia de una interacción positiva entre la proteína y el ADN se visualiza por la presencia de una banda radiográfica en la película. Lo negativo de esta técnica, es que no realiza teniendo en cuenta el contexto celular, por lo que su limitada utilidad ha provocado el desarrollo de otros enfoques para analizar las interacciones ADN-proteína [10].

Como contraparte en modelos in vivo, la técnica de chromatin immunoprecipitation (ChiP) se ha hecho popular para la identificación de regiones del genoma asociados con proteínas específicas con sus contexto de cromatina nativos [10]. Desde el comienzo en el desarrollo de esta técnica, se ha ido evolucionando hacia el ChiP-chip hasta la más avanzada ChiP combinada con high-throughput sequencing (ChiP-seq), incrementando en cada avance en la técnica la robustez del análisis de las interacciones endógenas del ADN-proteína. Esta última técnica nombrada cobra especial importancia en este proyecto, ya que serán datos procesados con ella de donde se obtenga el dataset con el que se va a desarrollar el modelo.

El flujo de trabajo de ChIP-seq utilizado para perfilar los sitios específicos de unión del ADN para los FT, las enzimas de unión del ADN u otras proteínas asociadas al ADN (ChIP no histónico) y los sitios de ADN que corresponden a los nucleosomas modificados (ChIP histónico) es el siguiente. Siguiendo los protocolos del ChIP, la cromatina se fragmenta y se entrecruza con proteínas o nucleosomas modificados inmunoprecipitados utilizando un anticuerpo específico de la proteína o de la histona modificada. Tras la purificación del ADN y la construcción de la biblioteca, con el tremendo progreso de la tecnología NGS, la plataforma Illumina, como Hiseq, ha sido la plataforma más utilizada para la secuenciación del ADN posteriormente [11].

El ChIP-seq es un método muy poderoso para identificar sitios de unión de ADN en todo el genoma para una proteína de interés. Los múltiples desafíos que se presentan en ChIP-seq no sólo se refieren a la preparación y secuenciación de las muestras, sino también al análisis computacional. Esto es debido a que ChIP-seq identifica regiones genómicas (~200 bp) en las que se acopla un FT, pero no delinea exactamente el motivo de unión dentro de estas regiones.

3.5 Programas y técnicas actuales: Meme

La identificación de los lugares de acoplamiento de los FT en todo el genoma es fundamental para comprender la regulación transcripcional. Dado que no es posible identificar experimentalmente todos los lugares de acoplamiento de los FT para cada tipo de célula y condición celular, el modelado computacional de las especificidades de unión de los TF ha sido fundamental para predecir estos lugares de acoplamiento. Estos modelos computacionales pretenden representar la compleja interacción entre la lectura de los nucleótidos y/o la forma del ADN en los lugares de acoplamiento de los FT [12], y pueden utilizarse para predecir no sólo la ubicación precisa en la que los FTs interactúan en el genoma [13], sino también los FTs con lugares de acoplamiento de los FT enriquecidos en un conjunto de secuencias, o el impacto de las mutaciones en la unión de los FTs , entre otros.

De los muchos tipos de modelos computacionales que existen, las habitualmente más empleadas para la predicción de motivos son las matrices de frecuencia posicional (PFMs), a pesar de que se trata del modelo más simple. Se trata de un perfil de unión a FTs que modela la especificidad de unión al ADN de un FT resumiendo las frecuencias de cada nucleótido en cada posición a partir de las interacciones observadas entre FT y ADN. Estas interacciones suelen derivarse de ensayos in vitro (por ejemplo, SELEX [14] o microarrays de unión a proteínas [15]), que evalúan la afinidad de unión de los FT a secuencias de ADN, o de experimentos basados en ChIP (por ejemplo, ChIP-seq [16]), que capturan las interacciones FT-ADN in vivo, buscando secuencias de ADN sobrerrepresentadas en las regiones unidas por el FT que buscamos.

MEME Suite es un conjunto de herramientas de software para realizar análisis de secuencias basados en motivos, lo que resulta valioso en una amplia variedad de contextos científicos. La

versión basada en la web de MEME Suite incluye 13 herramientas para realizar el descubrimiento de motivos, el análisis de enriquecimiento de motivos, la exploración de motivos y las comparaciones entre motivos [17]. Para los análisis de descubrimiento y enriquecimiento de motivos, el usuario proporciona un conjunto de secuencias de ADN, ARN o proteínas no alineadas. Normalmente, estas secuencias pueden ser regiones de peaks de ChIP-seq, sitios de cross-linking de un experimento CLIP-seq, promotores de genes coexpresados o proteínas que comparten una función común, como ser modificadas por la misma quinasa.

3.6 Limitaciones y nuevas alternativas

A pesar del éxito y todos los avances conseguidos en la predicción de lugares de acoplamiento de los FT a partir de las PFMs, estos modelos suponen en gran medida que cada nucleótido participa de forma independiente en la correspondiente interacción ADN-proteína y no tienen en cuenta los motivos de longitud flexible ni otros factores como la posición relativa en el genoma. Es por ello que otros modelos que permiten hacer un análisis no solo de la composición, sino también estructural del motivo de estudio, han sido objeto de estudio en los últimos tiempos, tales como Bayesian hierarchical hidden Markov models (HMMs) [18], transcription factor flexible models (TFFMs) basado en modelos ocultos de Markov o técnicas de Deep Learning [19]. A pesar de la novedad en el empleo de estos modelos y la falta de conocimiento que aún se tiene en este campo, estas nuevas aproximaciones concluyen en todos los casos en haber logrado sobrepasar al menos, algunas de las limitaciones de las aproximaciones clásicas.

Más en concreto, estudios como los de Mathelier et al. [20], [21] realizan una evaluación exhaustiva de conjuntos de datos in vivo para evaluar el poder predictivo obtenido al aumentar varios modelos basados en la secuencia de ADN de los sitios de unión de TF con características de la forma del ADN (helix twist, minor groove width, propeller twist, y roll), demostrando con 76 FTs en 400 bases de datos de ChIP-seq de humanos que combinando las características de DNA shape con las PFMs se mejoraba la predicción de los lugares de acoplamiento.

3.7 Hipótesis del trabajo

Tal y como se ha ido diciendo a lo largo de este capítulo, la técnica de ChIP-seq es un método muy robusto para identificar regiones de acoplamiento ADN-proteína in vivo. En organismos bacterianos, la regulación transcripcional de la expresión génica por FT es un mecanismo regulador común [22]. Estudios recientes han resaltado el valor de ChIP-seq en el estudio de la unión de FTs en varias especies bacterianas bajo una variedad de condiciones de crecimiento [10], [23]. Estos resultados muestran que, además de identificar regiones de unión, la correlación de los datos de ChIP-seq con los datos de expresión puede revelar información importante sobre los reguladores bacterianos y las redes de regulación.

Sin embargo, existen ciertos casos en los que los métodos convencionales en los que solo se toma como referencia los motivos encontrados a partir matrices de frecuencia de los nucleótidos no son suficientes para realizar una predicción suficientemente buena de los lugares en los que se acoplarán los FTs, limitando la calidad del modelo computacional desarrollado.

Es por ello por lo que este trabajo se fundamenta en la hipótesis de combinar, por una parte, las características de la forma del ADN con diferentes funciones que determinen la orientación espacial de las secuencias, tal y como mencionábamos previamente con DNA shape [20], [21]; y por otra, el cálculo de las secuencias de nucleótidos de diferentes longitudes, es decir, calcular

el número de veces que una combinación de nucleótidos se repite (i.e. cuántas veces aparece la secuencia de “AC” repetida durante la secuencia objetivo).

Se establece por tanto como hipótesis a refutar el hecho de que a partir de características extraídas a partir de datos generados por la técnica de ChIP-seq que generará peaks con las secuencias de ADN en donde los FT se han acoplado, se generarán modelos de ML que logren predecir de manera satisfactoria los lugares de acoplamiento para ese FT en un organismo bacteriano.

4. Fundamentos

Antes de comenzar con la sección de metodología, es preciso hacer un breve repaso teórico por diferentes conceptos que van a ser sobreentendidos a lo largo del desarrollo del trabajo llevado a cabo, y que componen los pilares y dan sentido a lo que se pretende conseguir con este proyecto. Se tratarán por tanto desde la explicación de la técnica con la que se obtuvieron los datos iniciales con los que se trabajará, como la lógica detrás de los descriptores escogidos, las técnicas de ML implementadas y tratamiento de los datos realizados.

4.1 ChIP-seq

La secuenciación del ChIP (también conocida como ChIP-seq), que combina los ensayos de inmunoprecipitación de la cromatina (ChIP) con la secuenciación del ADN, es una técnica robusta para el perfilado de las proteínas de unión al ADN, las modificaciones de las histonas o los nucleosomas en todo el genoma. El ChIP es un tipo de método experimental de inmunoprecipitación (IP) que se utiliza para aislar sitios específicos de ADN en interacción física directa con factores de transcripción y otras proteínas. En el ChIP se utilizan anticuerpos específicos para enriquecer fragmentos de ADN unidos a determinadas proteínas o nucleosomas.

ChIP-seq fue una de las primeras aplicaciones de la Secuenciación de Nueva Generación (NGS), y el primer estudio de la elaboración de perfiles a gran escala de las metilaciones de las histonas en todo el genoma utilizando ChIP-seq se publicó en 2007 [24]. La secuenciación de este estudio se realizó en la plataforma del analizador de genoma Solexa 1G. En el mismo año, Johnson et al. [25] usaron ChIP-seq para generar el mapeo de todo el genoma de los sitios de unión de los factores de transcripción. Estos dos trabajos también demostraron el aumento de la sensibilidad y la especificidad de ChIP-seq. Debido a los rápidos progresos de la tecnología de NGS y a la disminución del costo de la secuenciación, ChIP-seq se ha convertido en una herramienta indispensable para la caracterización de los epigenomas y el estudio de la regulación de los genes [11].

El método de ChiP-seq posee la habilidad de decodificar millones de fragmentos de ADN de manera simultánea con gran eficiencia y a relativamente bajo coste. Esta técnica implica cross-linking inverso del ADN inmunoprecipitado, la fragmentación y el análisis mediante secuenciación masiva de ADN en paralelo [26], informando sobre las secuencias de ADN que pueden ser cross-linked por formaldehído a un factor de transcripción determinado en células en crecimiento activo y luego enriquecidas en relación con el ADN genómico cuando los complejos ADN-proteína se precipitan mediante el uso de un anticuerpo específico para el factor de transcripción [27]. Después de alinear las lecturas de secuenciación con el genoma, el siguiente paso es identificar las áreas de enriquecimiento en los datos de ChIP-seq que representan los lugares donde se une el FT en todo el genoma, denominados peaks. La identificación robusta de los peaks es crucial para el éxito de un análisis ChIP-seq. Los algoritmos de búsqueda de peaks están pensados para identificar áreas de enriquecimiento en los datos de ChIP-seq. Estos algoritmos son importantes porque eliminan parte de la subjetividad de la búsqueda visual de picos y proporcionan una base estadística para determinar las áreas de enriquecimiento [27].

Si bien es verdad que generalmente, una vez obtenidos los peaks se realiza el descubrimiento de motivos y el enriquecimiento, en nuestro caso en concreto nos quedaremos con esos datos para desarrollar el modelo de ML.

Es por todo esto que para este proyecto se ha decidido utilizar como base del experimento datos obtenidos a partir de la técnica de ChIP-seq. En el siguiente subapartado se explicará de manera pormenorizada de las técnicas de extracción de características en las que nos hemos basado a partir de los datos de peaks encontrados.

4.2 Generación de descriptores

4.2.1 DNA Shape

En la mayoría de ocasiones, los FTs son proteínas acopladas a secuencias específicas del ADN, que van a reconocer posiciones específicas del genoma mediante una interacción compleja entre contactos de nucleótidos y aminoácidos y una lectura de la estructura del ADN [12].

Como se ha explicado en la sección del Estado del Arte, los métodos clásicos y más empleados de búsqueda de matrices de búsqueda posicionales que circunscriben el acoplamiento de la proteína a una región aislada suelen funcionar bien [28]. Sin embargo, se cree que el reconocimiento de secuencias específicas de ADN por parte de las proteínas depende de dos tipos de mecanismos: uno que implica la formación de enlaces de hidrógeno con bases específicas, principalmente en el surco principal, y otro que implica deformaciones de la hélice de ADN dependientes de la secuencia, habitualmente no tenido en cuenta [29].

Estudiar el DNA Shape se ha establecido en los últimos años como una aproximación que revela los determinantes de la especificidad de la unión proteína-ADN más allá de la secuencia primaria de nucleótidos.

DNAShapeR

Para llevar a cabo la selección del primer grupo de descriptores, se utilizó una librería especializada en obtener características de la estructura especial de los diferentes nucleótidos por lo que está formada la secuencia de entrada.

Con el nombre de DNAshapeR (https://bioconductor.org/packages/release/bioc/html/DNAshapeR.html), es un paquete del programa R que está basado en la predicción de la forma del ADN, y también cuenta con un servidor web (https://rohslab.usc.edu/DNAshape/).

La metodología que sigue este modelo para extraer los descriptores espaciales de las secuencias, los selecciona basándose en estudios experimentales previos que demostraban su importante papel en el reconocimiento proteína-ADN, e incluyen Minor Groove Width (MGW) [30], Propeller Twist (ProT) [31], Roll [32] y Helix Twist (HelT) [33]. Utilizando los pentámeros como ventanas deslizantes, DNAshapeR realiza simulaciones Monte Carlo de todos los átomos de 2121 fragmentos de ADN de 10-27 pb de longitud. A cada uno de los 512 pentámeros únicos se le asigna el valor medio de todas sus apariciones en el conjunto de datos en el nucleótido central para MGW y ProT y en los dos pasos de pares de bases (pb) centrales del pentámero para Roll y HelT. Cada pentámero aparece una media de 44 veces en nuestro conjunto de datos generado por Monte Carlo. El método DNAshape fue validado con datos experimentales de cristalografía de rayos X, espectroscopia de resonancia magnética nuclear y mediciones de

escisión de radicales hidroxilo [34]. En la **¡Error! La autoreferencia al marcador no es válida.** se muestra un esquema del diagrama de flujo del funcionamiento de este análisis.

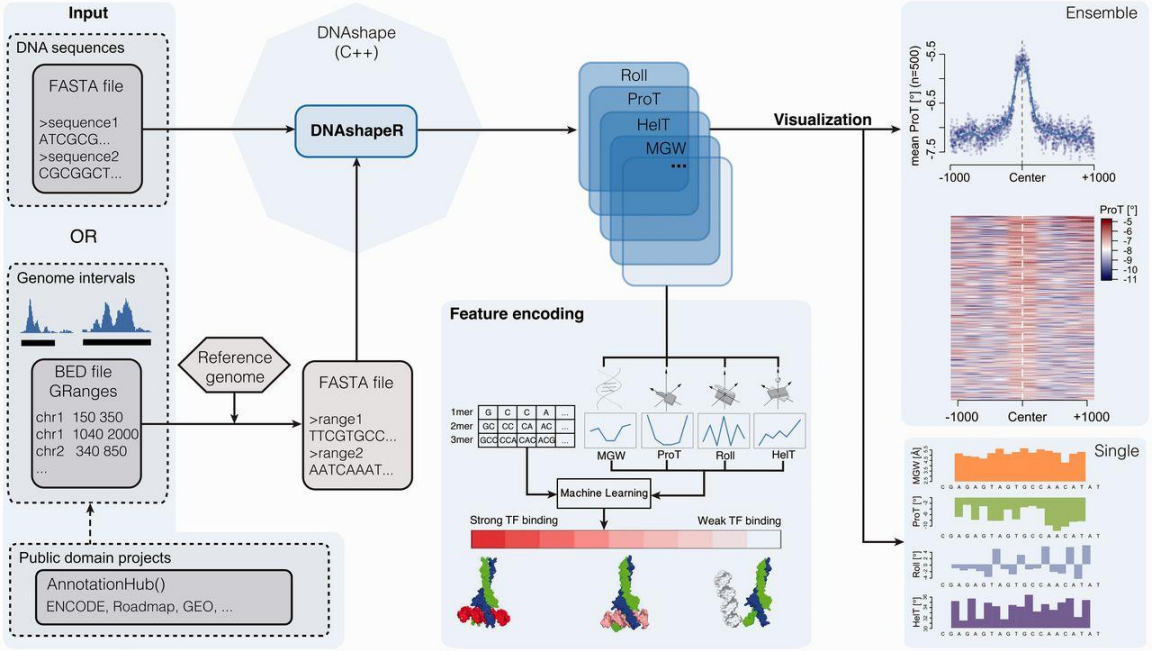


Figura 1. Diagrama de flujo del análisis de DNAshapeR [34]. Los datos de entrada pueden ser secuencias de nucleótidos en formato de archivo FASTA o intervalos genómicos, proporcionados por el usuario en formato BED o derivados de bases de datos públicas. El núcleo de DNAshapeR incluye un enfoque de alto rendimiento para la predicción de las características de la forma del ADN. MGW, HelT, ProT y Roll pueden visualizarse en forma de gráficos, mapas de calor o pistas del navegador del genoma o utilizarse para el montaje de vectores de características de combinaciones definidas por el usuario de k-mer y características de forma.

4.2.2 Kmers

Además de las rotaciones espaciales, otra manera que se ha tenido en cuenta para extraer las características de composición y espacio a la vez de una secuencia ha sido con el conteo de nucleótidos en diferentes subconjuntos. De esta manera, las secuencias de ADN se dividen según los nucleótidos que las conforman en una serie de k-mer con una longitud y una ventana de desplazamiento determinadas.

Proyectos de ML extrayendo las características a partir de los k-mers del ADN a partir de datos de ChIP-seq han demostrado en los últimos tiempos predecir los elementos reguladores funcionales del genoma y los potenciadores específicos de los tejidos [35], o modelos más complejos de Deep Learning (DL) que emplearon k-mers con Deep Convolutional Neural Networks (CNN) que demostraron mejor rendimiento en las predicciones de lugares de acoplamiento para FT [36]. Es más, esta técnica de conteo de k-mers no solo ha demostrado ser útil para obtener características a partir de ADN, sino también de otros tipos de moléculas [37].

4.3 Preprocesado de características

Cuando trabajamos con bases de datos con un número de variables muy alto, algunas pueden llegar a ser redundantes o generan ruido. Para lograr tener un rendimiento mayor, existen diferentes tipos de técnicas que ayudan a conseguir una reducción de aquéllas que no estén aportando información al modelo de ML que vayamos a desarrollar. Utilizar técnicas que nos ayuden a manejar este exceso de información contraproducente, ayudará no solo a aumentar

la efectividad de los modelos, sino también ayudará a reducir ciertos sesgos como el overfitting [38].

En general, los métodos de reducción de la dimensionalidad pueden dividirse en dos tipos: los basados en la extracción de características [39] y los basados en la selección de características, o feature selection (FS) [40].

4.3.1 Feature selection

En ML existen principalmente tres enfoques para el FS: filter, wrapper y embedded [40]. La principal diferencia entre filter y los otros dos enfoques es que el primero va a buscar las características a seleccionar independientemente del algoritmo de clasificación, mientras que wrapper y embedded buscan la selección de características en función del algoritmo de clasificación.

Los enfoques de filter obtienen una puntuación que mide la relevancia de las características con respecto al vector de clase observando únicamente las propiedades intrínsecas de los datos sin tomar ninguna suposición de los clasificadores. Además, este enfoque es computacionalmente sencillo y rápido. Es especialmente relevante para los datos de alta dimensión. Como estos enfoques son independientes del algoritmo de clasificación, el subconjunto de características seleccionadas se utiliza como entrada de cualquier algoritmo. Existen dos enfoques de filter diferentes: univariante y multivariante. Los enfoques de filter univariante son rápidos, escalables e independientes del clasificador, pero ignoran las dependencias de las características y la interacción con el clasificador. Los modelos multivariantes presentan las dependencias de las características con independencia del clasificador y, por tanto, son mejores desde el punto de vista computacional que los métodos envolventes.

Se ha elegido este tipo de técnica de FS de filter (univariante) en la selección de características extraídas de los peaks de ChIP-seq, debido a que las características que se obtienen son rápidas fáciles de entender. Por lo tanto, este enfoque nos permite realizar una mejor comparación entre los diferentes modelos de clasificación [41] y la particularidad de cada característica, independientemente del comportamiento particular de cada técnica.

Por lo tanto, seguimos un enfoque de FS de filter univariante, y para el cálculo de la relevancia de las variables, se utilizó una F-test. La F-test es la relación de las varianzas de los dos conjuntos de valores dados que se utiliza para comprobar si las desviaciones estándar de dos poblaciones son iguales o si la desviación estándar de una población es menor que la de otra. En este trabajo se utiliza el valor de F-test obtenido para comprobar las varianzas de las características de los peaks y de los replicados generados a partir de éstos (la muestra de negativos). Las características con el menor valor F-test se seleccionan para su inclusión en el análisis posterior. La fórmula para calcular el valor F-test de una característica con la que estamos trabajando es la siguiente:

$$F = \frac{v_1}{v_2} \quad (1)$$

En donde v1 : varianza de las muestras de *peaks* y v_2: varianza de muestras de replicados.

4.3.2 Reducción de dimensionalidad: PCA

Los métodos de extracción de características suelen basarse en la transformación de características, esencialmente proyectando datos de alta dimensión en un subespacio de baja dimensión. Este tipo de métodos de reducción de dimensión generalmente conservan la distancia relativa original entre las características y ayudan a cubrir la estructura potencial de los datos originales, por lo que no causarán una gran pérdida de información [42].

El análisis de componentes principales (PCA) [43] es el método más típico de reducción de la dimensionalidad basados en la extracción de características. La PCA es una técnica multivariante que analiza una tabla de datos en la que las observaciones se describen mediante varias variables dependientes cuantitativas interrelacionadas. Su objetivo es extraer la información importante de la tabla, representarla como un conjunto de nuevas variables ortogonales llamadas componentes principales, y mostrar el patrón de similitud de las observaciones y de las variables como puntos en mapas [44].

Se ha elegido esta técnica, debido a que mediante la transformación de las variables extrayendo información de varias a la vez, se puede reducir la dimensionalidad sin perder información. El principal problema que tiene esta técnica es que a nivel de interpretabilidad de las mejores PCA es limitado, ya que es una mezcla transformada de varias.

4.4 Machine Learning

El aprendizaje automático es el campo de estudio interesado en el desarrollo de algoritmos computacionales capaces de transformar datos en acciones inteligentes. Este campo es extenso en varias áreas, ya que ayuda a explicar y extraer conocimientos específicos de un conjunto de datos que los humanos no podrían lograr. Los algoritmos utilizados están diseñados para realizar una búsqueda probabilística trabajando en grandes espacios que involucran estados que pueden ser representados por conjuntos de datos. Hay dos tipos principales de aprendizaje: supervisado y no supervisado. La principal diferencia entre ellos es que en el primero, el aprendizaje se produce a través de observaciones etiquetadas, mientras que en el segundo, los ejemplos no están etiquetados, y el algoritmo busca agrupar los datos en diferentes grupos. En este estudio, trabajaremos con algoritmos de clasificación supervisada a partir de un conjunto de ejemplos etiquetados; estos algoritmos tratan de asignar una etiqueta a un segundo conjunto de ejemplos.

Utilizamos dos implementaciones diferentes de los siguientes algoritmos de aprendizaje automático: Random Forest (RF) [45] y un modelo lineal generalizado (Glmnet) [46].

Cada uno de estos algoritmos de ML tiene un conjunto particular de hiperparámetros que deben ajustarse para encontrar la mejor combinación posible y, en consecuencia, la mejor predicción y solución del problema. Los algoritmos de aprendizaje automático son técnicas muy potentes, pero el proceso de entrenamiento es fundamental. Este tipo de algoritmo aprende a través de muestras, por lo que no se deben utilizar las mismas muestras para el aprendizaje, la validación o el ajuste de hiperparámetros. En la sección de metodología se explicará en más detalle cómo se ha realizado este proceso de entrenamiento de los modelos.

4.4.1 Random Forest

El clasificador de RF consiste en una combinación de clasificadores de árbol en la que cada clasificador se genera utilizando un vector aleatorio muestreado independientemente del vector de entrada, y cada árbol emite un voto unitario para la clase más popular para clasificar un vector de entrada [45]. El clasificador de RF utilizado para este estudio consiste en utilizar características seleccionadas al azar o una combinación de características en cada nodo para hacer crecer un árbol. Se toma como predicción el voto mayoritario de los árboles en la clasificación. Así pues, el RF añade una capa adicional de aleatoriedad a un enfoque convencional de *bagging* [47].

Los hiperparámetros más destacados de este algoritmo y que van a ser tuneados en los modelos que se han desarrollado son *mtry*, *nodesize* y *number of tres*. *Number of randomly drawn candidate variables* o *mtry*, se define como el número de variables extraídas al azar entre las que se selecciona cada división al hacer crecer un árbol. Los valores bajos de *mtry* conducen a árboles más diferentes y menos correlacionados, lo que proporciona una mayor estabilidad en la agregación. El hiperparámetro *nodesize* especifica el número mínimo de observaciones en un nodo terminal. Si se ajusta a un valor más bajo, se obtienen árboles con mayor profundidad, lo que significa que se realizan más divisiones hasta los nodos terminales. Finalmente, el *number of tres* es un parámetro no tuneable para hacer la búsqueda, sin embargo, es importante que se marque con un valor lo suficientemente alto para que funcione correctamente.

4.4.2 Glmnet

Las regresiones logísticas son algoritmos de clasificación muy habituales en los problemas de ML cuando la variable de respuesta es categórica. El algoritmo de regresión logística representa las probabilidades condicionales de clase a través de una función lineal de los predictores. En este estudio, utilizamos un algoritmo de regularización rápida que ajusta un modelo lineal generalizado con penalizaciones de *Elastic Net*, llamado glmnet. La penalización *Elastic Net* puede tender hacia la penalización *Lasso* a la penalización *Ridge* [46]. Se sabe que la penalización de *Ridge* encoge los coeficientes de los predictores correlacionados entre sí, mientras que el *Lasso* tiende a elegir uno de ellos y descartar los demás. Por lo tanto, la penalización de *Elastic Net* mezcla estas dos.

La ecuación con la que *Elastic Net* lleva a cabo la penalización y que puede ser tuneable en dos hiperparámetros es la siguiente:

$$\lambda\left(\frac{1}{2}\,(1-\alpha)\beta^2+\alpha|\beta|\right)$$

(2)

Se hará el cálculo del mejor tuneado para este algoritmo mediante la prueba de diferentes combinaciones de alfa (α) y lambda (λ), que controlarán las propiedades de estimación del método de ML, condicionando la reducción de los coeficientes de los predictores.

5. Metodología

El pipeline de trabajo seguido durante el desarrollo de este proyecto es el que aquí aparece descrito. Para llevar a cabo la programación de los scripts pertinentes a cada caso para realizar los pasos realizados sin el empleo de herramientas online, se ha empleado un ordenador Intel(R) Core(TM) i7-8550U CPU de 1.80-2.00GHz , con 8.00 GB de RAM. Y se utilizado un ambiente virtual para Python 3.7.10, y la versión de R 4.0.3. Todo el código empleado, además, está compartido de manera pública en el repositorio de Git: https://github.com/saraalgo/binding_protein

5.1 Búsqueda de datasets

Se decidió realizar buscar papers que utilizaran datasets en los que se investigaran lugares de acoplamiento de FT en organismos bacterianos cuyos genomas fueran analizados mediante la previamente introducida técnica de ChIP-seq.

5.1.1 Keywords

Para llevar a cabo tal búsqueda, se introdujeron como filtros iniciales los conceptos de "*Transcription Factor binding*", "*ChIP-seq*" y "*bacteria*". Se intentó además, seleccionar *papers* en los que se trabajara con organismos o FT que tuvieran bastantes estudios para facilitar una posible validación externa del modelo de cara a futuro, tales como: *E. coli, V. cholerae o L. interrogans* en bacterias; o CRP o LexA en FT.

5.1.2 Herramientas de búsqueda

Para llevar a cabo esta tarea, se utilizaron motores de búsqueda bibliográfica científica estandarizados tales como: Scopus, WoS o Google Scholar.

5.1.3 Criterios de inclusión o exclusión

A la hora de establecer los criterios de inclusión o exclusión de cada uno de los artículos que se fueron encontrando, se tuvo en cuenta el estado de la base de datos que utilizaron para el estudio.

Criterios de inclusión:

- La base de datos empleada cuenta con un número ID con la que puede ser encontrada en plataformas como Gene Expression Omnibus (GEO) o ArrayExpress.
- Esta base de datos ha de tener los datos en crudo tras ser analizados con la técnica de ChIP-seq para poder seguir el pipeline de análisis estandarizado en su totalidad.
- Han de encontrarse subidos y claramente identificados cada experimento en la plataforma del European Nucleotide Archive (ENA), en formato FASTAQ.

Criterios de exclusión:

- El no cumplimiento de los criterios de inclusión en cuanto al estado de la base de datos empleada en el estudio.

- Que el genoma del organismo no se encontrara en la base de datos de Galaxy (herramienta posteriormente utilizada para hacer el procesado biológico de los datos en crudo).
- Bases de datos cuya obtención de motivos para el acoplamiento de los FT fuera muy sencilla

5.2 Preprocesado del dataset

Tras hacer una búsqueda sobre posibles datasets que podrían ser empleados para la predicción de sitios de acoplamientos para FT, se hicieron pruebas con bases de datos en crudo de los experimentos de ChIP-seq en la plataforma de Galaxy.

5.2.1 Galaxy

Concebida como una plataforma web de software libre para el análisis científico, desde su creación en 2005, Galaxy ha sido utilizada por cientos de miles de científicos en todo el mundo. Enmarcada en tres hitos clave para la ciencia biomédica basada en datos (accesibilidad, reproducibilidad y comunicación transparente), el equipo de Galaxy ha conseguido mejoras sustanciales a lo largo de los años que permiten el análisis de extensos conjuntos de datos y más de 5500 herramientas que ya están disponibles en el Galaxy ToolShed. Cuenta con numerosos tutoriales de alta calidad enfocados al análisis de tipos comunes de genómica, además, el usuario dispone principalmente de dos modalidades para analizar los datos, por ejecución exploratoria o pipeline, pudiendo realizarlas simultáneamente. De forma automática, se pueden generar flujos de trabajo reutilizables y generalizables a partir de un análisis ad hoc, y también existe un editor de flujos de trabajo interactivo para modificarlos o generarlos desde cero. Una característica que hace especialmente destacable a esta plataforma es que permite realizar análisis complejos sin tener ningún tipo de conocimiento en programación bioinformática, a pesar de que para el procesamiento completo de datos genómicos se requiere personalizar los scripts, contando con una infraestructura actualizada en sus Entornos Interactivos Galaxy [48], [49].

Este proyecto consta de cuatro componentes complementarios: el servidor público Galaxy (https://usegalaxy.org/), compuesto por un variado conjunto de herramientas para el análisis genómico a gran escala, terabytes de datos públicos para utilizar y un gran número de análisis compartidos, flujos de trabajo, así como publicaciones interactivas; el marco de trabajo y el ecosistema de software Galaxy (,https://github.com/galaxyproject) que es un paquete de software de código abierto que cualquiera puede utilizar para ejecutar el servidor Galaxy en un sistema operativo basado en Unix; el Galaxy ToolShed (https://toolshed. g2.bx.psu.edu/), disponible para los usuarios de la "AppStore" donde pueden compartir las herramientas de Galaxy, como pueden ser Gemini, para explorar las variaciones genéticas [50], QIIME para el análisis cuantitativo del microbioma para los datos de secuenciación de ADN en bruto [50], [51] o deepTolls para el análisis de datos de secuencias profundas [50]–[53], además de los flujos de trabajo y las visualizaciones; por último, proporciona la Comunidad Galaxy (https://galaxyproject.org/community/), lugar donde todos los aspectos dentro de este proyecto.

5.2.2 Pipeline en el análisis de los datos de ChIP-seq

A la hora de trabajar con esta técnica de secuenciación de ADN y debido al desconocimiento en el uso de la plataforma, se decidió utilizar como guía de referencia el tutorial de Galaxy referido

al análisis de data ChIP-seq (https://galaxyproject.org/tutorials/chip/). Tal y como explica además la literatura referida al procesado de estos datos, se siguieron los pasos habituales en el pipeline para la obtención de los lugares de acoplamiento de los FTs, tal y como se describe en el trabajo de referencia realizado por Bailey et al. [54], ilustrado también en la Figura 2.

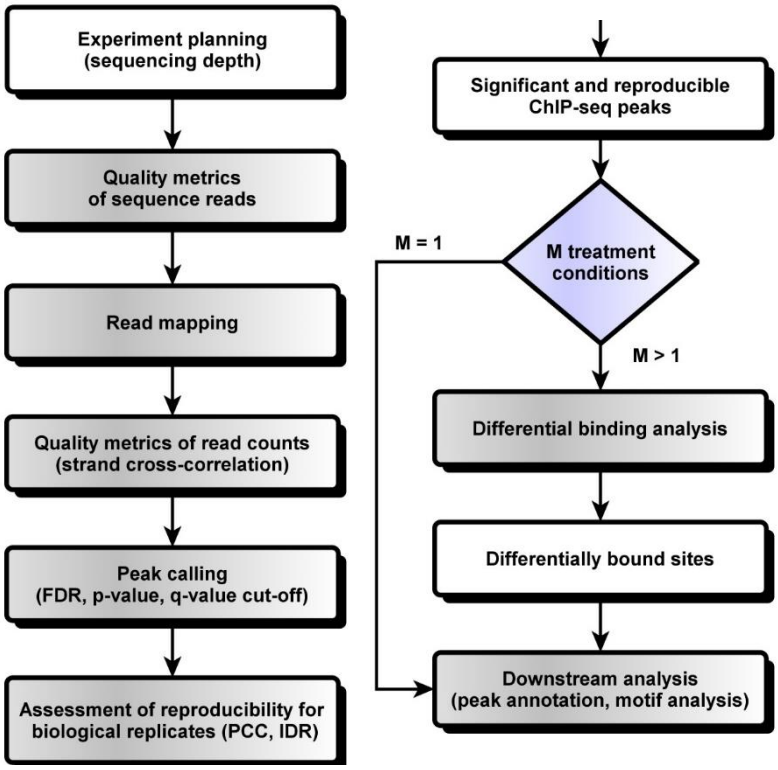


Figura 2 Workflow de un análisis computacional de ChIP-seq [54]

A grandes rasgos, el objetivo del tratamiento de los datos biológicos pasa por diferentes etapas en los que, a partir de los datos previamente descargados en formato *FASTAQ*, son filtrados con diferentes métodos para asegurar su calidad, aplicando diferentes métricas estadísticas. Tras eso, los resultados obtenidos son mapeados sobre el genoma de referencia, haciendo estos pasos tanto para los resultados de ChIP-seq de las condiciones donde se haya empleado el FT y los controles. De nuevo, se aplican de nuevo otros controles de calidad para asegurar que se emplean lecturas con el mínimo ruido posible.

Una vez que estemos convencidos de que los datos son de suficiente calidad, podemos proceder al análisis posterior. Uno de los primeros pasos en el análisis ChIP-seq es el *peak calling*. El *peak calling* es un procedimiento estadístico que utiliza las propiedades de cobertura de las muestras de ChIP y de entrada para encontrar regiones enriquecidas debido a la unión de proteínas, llamados *peaks*. El procedimiento requiere lecturas mapeadas y produce un conjunto de regiones que representan las posibles ubicaciones de unión. Cada región suele estar asociada a una puntuación de significación que es un indicador de enriquecimiento.

En nuestro caso en concreto, aunque el pipeline de trabajo suele continuar a la búsqueda de motivos a partir de esos *peaks* mediante herramientas como Meme Suite (descrita anteriormente en la sección de Estado del Arte), decidimos trabajar directamente con ellos. Esto

es así, porque las secuencias localizadas en el genoma dadas por las posiciones que indican los *peaks*, las hemos utilizado como datos positivos para nuestro dataset.

Será a partir de estos datos en los que se supone que hay mayor localización del FT en el genoma, de donde hemos calculado el dataset negativo para llevar a cabo nuestros modelos de predicción de ML. El método para llevarlo a cabo es descrito en el siguiente subapartado.

5.3 Curación manual de datasets (alternativa a Galaxy en material suplementario peaks reportados)

Como método alternativo al empleo de Galaxy, también se realizó una búsqueda con los mismos parámetros en los buscadores de artículos científicos, pero con objetivo de encontrar *papers* en los que se anexara o estuvieran incluidos directamente los *peaks* obtenidos del análisis biológico computacional.

Esto se hizo así como método alternativo para utilizarlos directamente aunque perdiéramos posible información de los *peaks* si hiciéramos nosotros todo el preprocesado, por si el procesado biológico con Galaxy no era satisfactorio. De de esa manera, en caso de no ser capaces de extraer los *peaks*, poder pasar al paso importante del proyecto con la creación de los replicados del dataset negativo y su tratamiento para generar los modelos de ML.

5.4 Extracción de datos: creación de los *datasets*

5.4.1 JSON con los datos del experimento

Se generó en primer lugar un archivo JSON con diferentes variables que definen al problema en cuestión (especificando la especie, los ID en GEO, el FT a analizar, las carpetas en donde se encontraban los archivos…) con el fin de automatizar el sistema posterior y poder ejecutarlo de manera automática solo cambiando las variables en este archivo. De esta manera, se minimizan errores y se asegura que los datos introducidos sean lo que nos interesa modelizar.

5.4.2 Creación del dataset positivo

Como se ha explicado previamente, las secuencias con los *peaks* encontrados tras el empleo de la técnica de ChIP-seq y posterior análisis biológico computacional son el objetivo para establecerlos como datos positivos a predecir en nuestro modelo de ML.

Sin embargo, es importante considerar que, tanto si llevamos a cabo el procesado por nosotros mismos, como si los obtenemos de *papers* que los hayan publicado, solamente vamos a tener para trabajar las posiciones en el genoma de referencia del organismo de estudio en los que se encuentran.

Por lo tanto, para extraer las secuencias que se emplearán como datos positivos, se ha hecho una búsqueda del genoma de la bacteria de referencia y se han extraído las secuencias que empezaban y terminaban según lo indicado para cada *peak*. Es importante la consideración de que los *peaks* no indican la hebra del genoma en el que se encuentran, con lo que se han extraído todos de la hebra positiva.

5.4.3 Creación de los dataset negativos

Para conseguir al menos otro dataset para hacer una comparación en el rendimiento de la predicción de nuestros modelos de ML, se han desarrollado dos metodologías de extracción de los datos negativos. En ambos casos, la generación de los datos negativos va a ser a partir de los datos positivos.

Replicates

A la primera de las maneras en las que se han extraído los datos, se les ha puesto en nombre de *replicates*. En este método de extracción, se ha intentado ser lo más fieles a la composición biológica de los *peaks* posible.

Por ello, en primer lugar lo que se ha hecho es subdividir la secuencia genoma de referencia según si eran:

- Regiones génicas, en las cuáles estarían delimitados los genes en ambas hebras.
- Regiones *upstream*, es decir, partes precursoras colindantes a los genes que se encontraran a un máximo de 150bp. Esta clasificación como *upstream* se hizo solamente en la hebra positiva. Para compensarlo biológicamente con que en los genes de la hebra complementaria deberían de serían terminadores y no precursores, un porcentaje (calculado con el porcentaje de genes que tiene la hebra negativa) se consideró como *downstream* en vez de *upstream*.
- Regiones *downstream*, de manera similar a los anteriores, serían aquellas partes terminadoras colindantes a los genes que se encontraran a un máximo de 100bp tras la finalización de la parte génica. Tal y como ocurre con los *upstream*, esta clasificación también solo se hizo en la hebra positiva, y se compensó para que fueran los más biológicamente realistas de la manera inversa a los *upstream*.
- Regiones intergénicas, aquéllas que se encontraban a más de 250bp entre dos genes diferentes. De la misma manera, solo fue considerada la hebra positiva para la división de la secuencia.

Habiendo dividido el genoma, a continuación se fue analizando *peak* por *peak* clasificando sus partes y cuántas bp tenían de cada una en el orden de aparición. Una vez se habían clasificado, se procedió a realizar el número de replicados especificado en el archivo JSON (*nreplicates*) de cada uno de los *peaks*. Para crear estos *replicates*, se fueron extrayendo de forma aleatoria secuencias de cada una de las categorías citadas anteriormente a fin de generar replicados con la misma longitud y función que en el *peak* original.

Pseudo-replicados con bootstrapping

Como método para crear otro dataset negativo y poder comparar, se llevó a cabo la creación de otro dataset negativo a través de la creación de pseudo-replicados sin correspondencia biológica de cada uno de los *peaks* positivos.

Este proceso fue llevado a cabo con para generar *nreplicates* para cada uno de los *peaks.* Para ello, se extrajeron trímeros de manera aleatoria de la secuencia del *peak* original hasta crear una secuencia de la misma longitud que el *peak* original.

5.5 Generación de descriptores

A partir de la creación de los datasets, se llevó a cabo la generación de los descriptores que fueron posteriormente empleados para entrenamos a los modelos de ML.

5.5.1 DNAshapeR

Con partir de la librería de R de DNAshapeR, se introdujeron las secuencias correspondientes a los *peaks* positivos y a los replicados en cada caso. Las variables resultantes generadas para cada secuencia fueron almacenadas en formato .fasta automáticamente con las características de forma espacial calculadas mediante los cuatro algoritmos de interés, en nuestro caso: *HelT, MGW, ProT,* y *Roll*.

5.5.2 Extracción de densidades

Debido a que las características de la forma espacial obtenidas con la librería de DNAshapeR son puntuaciones obtenidas mediante el análisis de la composición de los pentámeros que componen cada parte de la secuencia, se decidió generar los descriptores a partir de ellos a partir de las densidades de eses valores para cada secuencia de entrada, normalizándolos además teniendo en cuenta en mínimo y máximo global del conjunto de secuencias total. Este paso de la normalización es preciso realizarlo, debido a que los mínimos y máximos son establecidos también en función de la longitud de cada secuencia, y para cada *peak* y su/s correspondiente/s replicados, las longitudes fueron variables (ya que se trabajaron sin recortar las posiciones indicadas en el *paper* de referencia del que se obtuvo la localización de estos *peaks*).

5.5.3 K-mers

Otro planteamiento considerado para la extracción de descriptores ha sido el conteo de la composición de cada secuencia según *k-mers*. Para ello, se ha hecho un conteo de monómeros, dímeros y tetrámeros contenidos en cada una de las secuencias, obteniendo como descriptores una proporción normalizada según la longitud total de la secuencia de cada una de las combinaciones posibles de cada uno de los tres tipos de *k-mers* establecidos.

5.6 Técnicas de ML

El entrenamiento de los algoritmos de ML ha sido llevado a cabo a través de un remuestreo anidado. Por una parte, se ha llevado a cabo un *cross-validation* (CV) interno independiente, con el objetivo de seleccionar los mejores hiperparámetros para nuestros datos, considerando 2/3 de los datos para el entrenamiento y 1/3 para la validación. A continuación, una vez establecido la mejor combinación, se llevó a cabo un *cross-validation* externo independiente, para evaluar el modelo de manera general, llevando a cabo 5 repeticiones con un 10-fold-CV. Este CV externo divide los datos de entrada en 10 conjuntos, y emplea 9 para el entrenamiento y 1 para test 10 veces para combinaciones de entrenamiento/test diferentes. El rendimiento de este 10-fold-CV fue la media del rendimiento de las combinaciones. Este procedimiento se repitió 5 veces para cada algoritmo, presentando finalmente como resultado de rendimiento del algoritmo la media de las 5 ejecuciones.

5.6.1 Random Forest

Para implementar el algoritmo de RF, se realizó una búsqueda de los valores a preestablecer para los hiperparámetros de *mtry* (número de variables muestreadas aleatoriamente en cada división de los datos) y *nodesize* (tamaño mínimo de los nodos terminales). El rango para el número de variables se estableció entre 2 y, como límite superior, 8, que es el segundo menos número de características que tienen los conjuntos de datos que se han empleado. El tamaño mínimo de los nodos terminales osciló entre 1 y 3. Los valores bajos de este parámetro proporcionan un gran crecimiento y profundidad de cada árbol, mejorando la precisión de las predicciones. Además, el número de árboles fue de 1000. Un gran número de árboles garantiza que cada observación se prediga al menos varias veces.

5.6.2 Glmnet

En cuanto al algoritmo de Glmnet, los *grids* de alfa y lambda para el ajuste son (0,0001, 0,001, 0,01, 0,1, 1) y (0, 0,15, 0,25, 0,35, 0,5, 0,65, 0,75, 0,85, 1), respectivamente. Alpha controla la penalización de la red elástica, desde *Lasso* ($\alpha = 1$) hasta *Ridge* ($\alpha = 0$). El parámetro lambda controla la fuerza total de la penalización.

5.6.3 Selección de descriptores

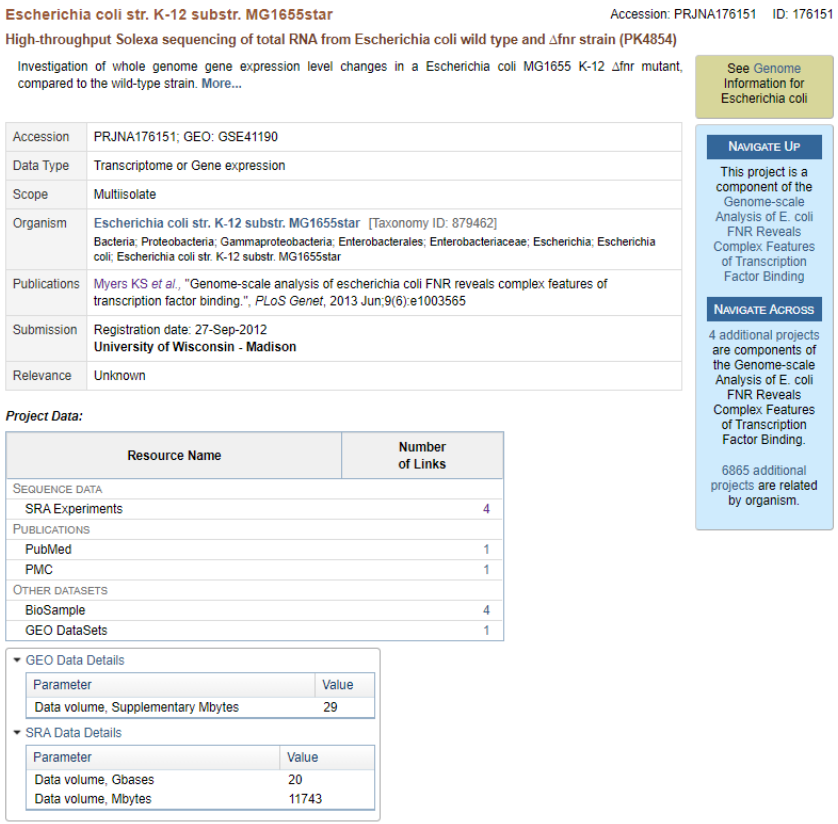
De manera complementaria a llevar a cabo los algoritmos con los descriptores extraídos al completo, en busca de intentar mejorar la predicción de los modelos generados, se aplicaron tanto técnicas de reducción de la dimensionalidad de estos mediante PCA, como FS empleando el *F-test.*

6. Resultados

6.1 Galaxy

A partir de las búsquedas realizada con los diferentes motores de búsqueda de artículos científicos, se seleccionó en primer lugar un artículo que superaba los criterios de inclusión y exclusión establecidos en la metodología para llevar a cabo el preprocesado biológico a partir de los datos almacenados en la plataforma ENA.

Se trata un estudio realizado por Myers et al. en 2013 [27] para localizar el acoplamiento de FNR en el organismo bacteriano *Escherichia coli*. En este trabajo, a pesar de que sí que lograron predecir ciertos lugares para el acoplamiento de FNR en el genoma, se consideró interesante ya que se trata de un problema complejo y que tenía mucho margen de mejora para intentar superar el rendimiento en la predicción a través de los modelos de ML que estaban propuesto desde la base de este trabajo. La base de datos empleada para este proyecto se puede encontrar en GEO con el ID: GSE41190, tal y como se muestra en la Figura 3.



Escherichia coli str. K-12 substr. MG1655star Accession: PRJNA176151 ID: 176151

High-throughput Solexa sequencing of total RNA from Escherichia coli wild type and Δfnr strain (PK4854)

Investigation of whole genome gene expression level changes in a Escherichia coli MG1655 K-12 Δfnr mutant, compared to the wild-type strain. More...

See Genome Information for Escherichia coli

Accession	PRJNA176151; GEO: GSE41190
Data Type	Transcriptome or Gene expression
Scope	Multiisolate
Organism	Escherichia coli str. K-12 substr. MG1655star [Taxonomy ID: 879462] Bacteria; Proteobacteria; Gammaproteobacteria; Enterobacterales; Enterobacteriaceae; Escherichia; Escherichia coli; Escherichia coli str. K-12 substr. MG1655star
Publications	Myers KS *et al.*, "Genome-scale analysis of escherichia coli FNR reveals complex features of transcription factor binding.", *PLoS Genet*, 2013 Jun;9(6):e1003565
Submission	Registration date: 27-Sep-2012 **University of Wisconsin - Madison**
Relevance	Unknown

NAVIGATE UP

This project is a component of the Genome-scale Analysis of E. coli FNR Reveals Complex Features of Transcription Factor Binding

NAVIGATE ACROSS

4 additional projects are components of the Genome-scale Analysis of E. coli FNR Reveals Complex Features of Transcription Factor Binding.

6865 additional projects are related by organism.

Project Data:

Resource Name	Number of Links
SEQUENCE DATA	
SRA Experiments	4
PUBLICATIONS	
PubMed	1
PMC	1
OTHER DATASETS	
BioSample	4
GEO DataSets	1

▾ GEO Data Details

Parameter	Value
Data volume, Supplementary Mbytes	29

▾ SRA Data Details

Parameter	Value
Data volume, Gbases	20
Data volume, Mbytes	11743

Figura 3. Captura de pantalla de la web GEO

De esta manera, y con la aplicación web de Galaxy, se llevaron a cabo pruebas con las muestras de ChIP-seq anaeróbicas y aeróbicas, *wild type* y con presencia de FNR (Figura 4), siguiendo el pipeline arriba descrito, con la guía del tutorial facilitado por Galaxy.

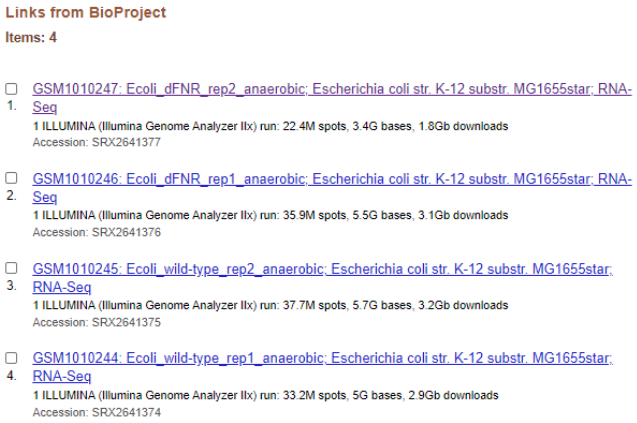


Figura 4. Captura de pantalla de las muestras de ChIP-seq

Tras mapear los datos con el genoma de referencia *E. coli* K-12 MG1655 y llevar a cabo el proceso de calidad y extracción de los *peaks*, se observó que los datos obtenidos no mostraban apenas diferencias ni en la matriz de correlación (Figura 5) ni en cálculos posteriores de control. Debido a la imposibilidad para obtener *peaks* que correspondieran a los mismos con los que se trabajó en el artículo original, se descartó llevar a cabo el tratamiento de datos desde los datos "en crudo" salidos de la técnica de ChIP-seq.

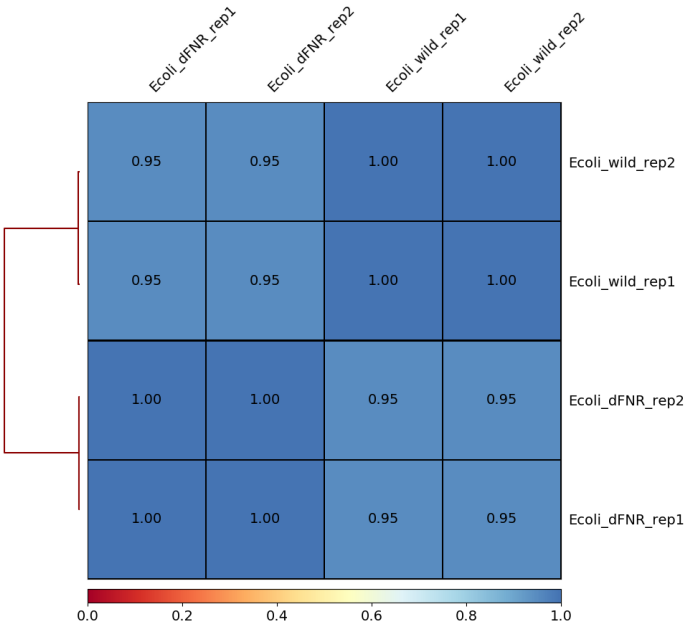


Figura 5. Matriz de correlación del pipeline ChIP-seq para dataset estudiado

Tras un gran número de intentos de solucionar el pipeline, y ante la situación de no controlar lo suficiente las herramientas de Galaxy ni tener claro qué era lo que estaba fallando, se decidió proceder directamente a trabajar con los datos de un experimento en el que ya aportaran la información de los *peaks* en el artículo, intentando asegurar que los datos que se obtengan de este proyecto sean comparables con los hallados en la fuente de la que se extraigan los datos para asegurar una mayor seguridad en las conclusiones que se pueden extraer de este proyecto.

6.2 Creación del dataset

Como se ha introducido, los *peaks* que finalmente se han seleccionado como datos positivos para nuestra primera prueba para desarrollar son los que encontramos en el estudio de Adhikari, Erill y Curtis [5], en concreto, estudiando las posiciones de inicio y terminación de los 879 *peaks* encontrados en el organismo *Brevundimonas subvibrioides* para el FT de GcrA.

Tanto el hecho de proporcionar un archivo Excel con los datos de comienzo y fin de cada *peak*, como la claridad en las conclusiones y resultados obtenidos de su propio estudio en cuanto a los motivos de acoplamiento encontrados, fueron las razones principales de su elección. En la Figura 6 se muestra un recorte del documento en el que aparecían las ubicaciones de los datos proporcionados.

Total GcrA peaks obtained from ChIP-seq

Chromosom	start	end	length	abs_summi	log2(fold_enrichmei	Name	Methyation sites
NC_014375.1	1523	1958	436	1726	1,42984	sampe_combine_MACS2_output_peak_1	1
NC_014375.1	6558	7003	446	6760	1,41819	sampe_combine_MACS2_output_peak_2	3
NC_014375.1	15004	15795	792	15356	1,33375	sampe_combine_MACS2_output_peak_3	2
NC_014375.1	31624	32273	650	32158	1,18183	sampe_combine_MACS2_output_peak_4	3
NC_014375.1	32946	33485	540	33194	1,44689	sampe_combine_MACS2_output_peak_5	1
NC_014375.1	41187	41538	352	41335	1,25167	sampe_combine_MACS2_output_peak_6	3
NC_014375.1	42024	42526	503	42257	1,69477	sampe_combine_MACS2_output_peak_7	0
NC_014375.1	44373	45302	930	44860	1,48284	sampe_combine_MACS2_output_peak_8	2
NC_014375.1	49865	50305	441	50076	1,17517	sampe_combine_MACS2_output_peak_9	2
NC_014375.1	52079	52614	536	52361	1,83296	sampe_combine_MACS2_output_peak_10	1
NC_014375.1	56227	56527	301	56337	1,07944	sampe_combine_MACS2_output_peak_11	0
NC_014375.1	58821	59456	636	59116	1,59727	sampe_combine_MACS2_output_peak_12	3
NC_014375.1	59549	60061	513	59804	1,29416	sampe_combine_MACS2_output_peak_13	2
NC_014375.1	64446	64956	511	64752	1,24634	sampe_combine_MACS2_output_peak_14	1
NC_014375.1	68117	68624	508	68340	1,22563	sampe_combine_MACS2_output_peak_15	2
NC_014375.1	76365	76701	337	76560	1,20892	sampe_combine_MACS2_output_peak_16	1
NC_014375.1	82389	82793	405	82547	1,18766	sampe_combine_MACS2_output_peak_17	2
NC_014375.1	85499	86214	716	85849	3,18797	sampe_combine_MACS2_output_peak_18	1

Figura 6. Recorte de la localización de los peaks *en el estudio escogido, anexados en el archivo Excel* S4 Data *[5]*

Tras la selección de los *peaks* que se iban a utilizar de referencia, se procedió a la creación de los dos datasets con los que se harían las pruebas posteriores. De manera común, se comenzó con la extracción de las secuencias de los *peaks* a partir de la extracción de los nucleótidos contenidos entre la posición de *start* y *end* especificadas en el Excel de referencia. En total, los datos positivos con los que se comenzó a trabajar desde este momento fueron 879 secuencias de los *peaks* de longitudes variables, las cuales oscilan entre 291 y 3035 bp. En la Figura 7 podemos ver un histograma de estas longitudes.

Debido a esta diferencia entre las longitudes de los datos positivos y para reducir los sesgos, los replicados se realizaron creando, como mínimo, una secuencia para el dataset negativo de la longitud de cada uno de los *peaks* del dataset positivo.

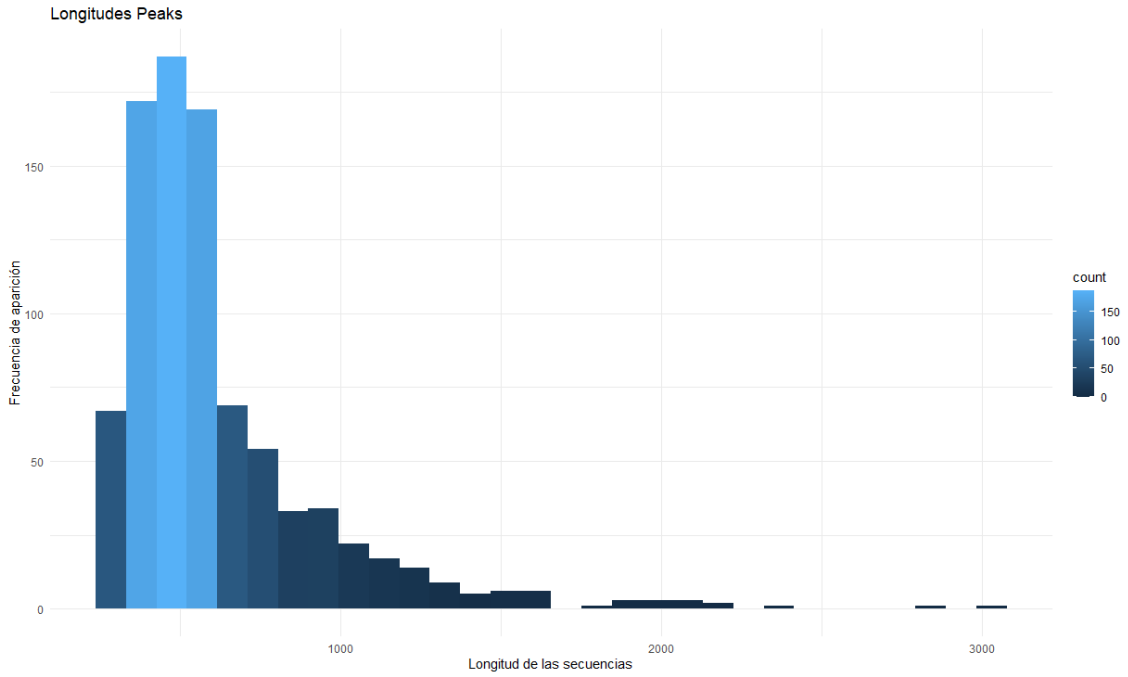


Figura 7. Histograma donde se muestras las densidades de longitudes de las secuencias del dataset positivo

6.2.1 Replicates

Para el primero de los dataset negativos creados, llamado *Replicates*, se extrajeron secuencias con la máxima equivalencia biológica posible. Intentando recrear copias negativas lo más similares en estructura posible. Para ello, clasificamos cada uno de los peaks según su localización en el genoma, estableciendo cuántas bp pertenecían a cada una de las clases predefinidas en la parte de metodología (génica, intergénica, upstream y downstream). Acudiendo al total de las secuencias de cada clase que extrajimos previamente en diccionarios de Python, para cada uno de los *peaks* se fue construyendo el número de homólogos negativos predefinido en el archivo JSON con pedazos de secuencias que se correspondieran con la composición de cada *peak*. Para poner un ejemplo gráfico que ayude a clarificar este proceso, suponiendo que la secuencia hipotética y simplificada de que la secuencia de un *peak* fuera:

Secuencia *peak*: "ATTCCTGAACTAGCCTAGTGA"

Clasificación *peak*: [upstream: 5bp], [génico: 10bp], [downstream: 6bp]

Replicate n1: "CCTAGACCTAGTTAACCTTAC"

Como se acaba de decir previamente, la listas con el conjunto de nucleótidos de cada clase predefinida es el lugar de dónde se irían extrayendo los pedazos de secuencias de cada tipo requeridos en cada caso. Debido a que los *peaks* solamente indican la localización en el genoma y no la hebra en la que se produjo el acoplamiento del FT, para intentar ser lo más fieles posibles estadísticamente a la biología: los casos en los que se definía una clase como upstream/downstream (debido a que los genes que se encuentran en la hebra reversa complementaria los tendría invertidos), en la proporción de genes en la hebra negativa del organismo, se invirtieron las clases upstream/downstream para extraer el pedazo de secuencia que se añadía al nuevo replicado.

Los diccionarios con el conjunto de nucleótidos de cada clase han sido creados a partir del genoma de referencia, en este caso, la bacteria *Brevundimonas subvibrioides*, pero gracias al

archivo JSON, el script está preparado para cambiar de manera sencilla tanto el organismo, como las localizaciones de los *peaks* y que pueda hallarse para el que se precise.

6.2.2 Bootstrapping

Para el segundo de los dataset negativos, a los que llamaremos boots_replicates también se fueron creando, en este caso, pseudo-replicados, de longitudes equivalente a cada uno de los *peaks*. Sin embargo, aquí no se tuvo tanto en cuenta la localización relativa de los *peaks* en las secuencias, sino que se crearon con la extracción aleatoria de trímeros que existían en el *peak* original hasta completar la longitud objetivo. Uno de los objetivos a la hora de crear este dataset negativo fue establecer si el buscar una equivalencia biológica más estricta a la hora de crearlos marcaría una diferencia a la hora de generar un modelo de predicción de las regiones de acoplamiento del FT GcrA.

Un ejemplo gráfico de este proceso de extracción sería el siguiente:

Secuencia *peak*: "ATTCCTGAACTAGCCTAGTGA"

Boots_replicate n1: "TAGTCCGCC…."

Hasta llegar a completar la longitud, que en este caso sería de 21 bp.

6.3 Generación de descriptores

Una vez obtenidos los datos positivos para trabajar, es decir, las secuencias de nuestros 879 *peaks*, y los negativos para nuestros dos datasets, que en esta primera prueba para el modelo de ML se componían de 879 replicates y 879 boots_replicates, se procedió a extraer los descriptores para poder hacer el entrenamiento de los algorimos de ML.

6.3.1 DNAshapeR

A partir de la librería de DNAshapeR en R, se introdujo el fasta de cada uno de los datasets (879 *peaks* + 879 replicados para cada uno de los casos) y se obtuvieron los fasta para cada uno de los 4 algoritmos seleccionados: HelT, MGW, ProT y Roll. En la Figura 8 se ve un ejemplo de una gráfica de la distribución de los valores obtenido para todas las secuencias introducidas en su conjunto.

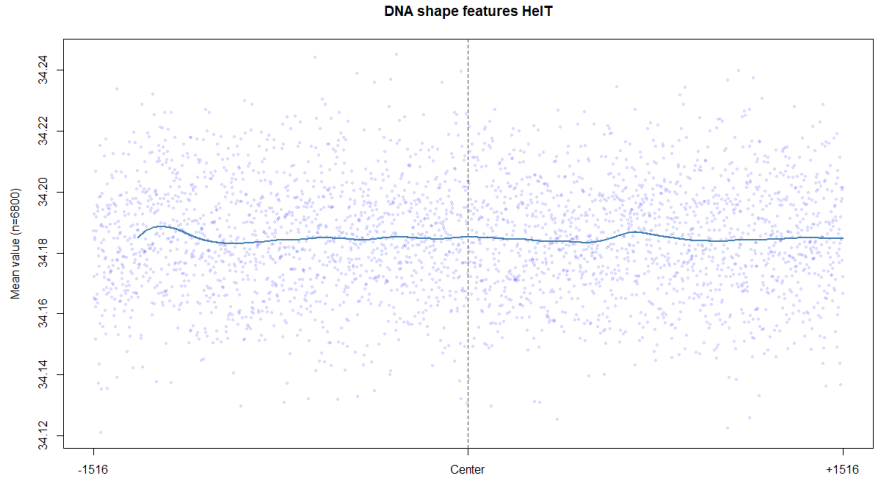


Figura 8. Gráfica de distribución de los valores obtenidos mediante DNAshapeR para HelT

Debido a que de cada secuencia introducida va a generar una nueva fila con el mismo número de descriptores que la misma secuencia y eran de longitudes diferentes, para extraer descriptores con los que se pudiera trabajar para hacer modelos de ML se decidió extraer las densidades de cada uno de los valores obtenidos en cada secuencia (tanto para las positivas como las negativas) y normalizarlos teniendo en cuenta el valor máximo y mínimo del conjunto de todas las secuencias introducidas. Esto es así, debido a que sino no tendrían equivalencia estadística, ya que nos interesa extraer las densidades no relativas a cada secuencia, sino al global de las mismas. En la Figura 9 podemos ver el ejemplo de cómo se distribuyen los valores en una secuencia y sus densidades. Se extrajeron los valores para 25 *bins*, es decir, para obtener 25 descriptores por algoritmo de DNAshape y dataset. Obteniendo 4 conjuntos de 25 descriptores para el dataset de 1758 filas de replicates y otros tantos para el dataset de boots_replicates.

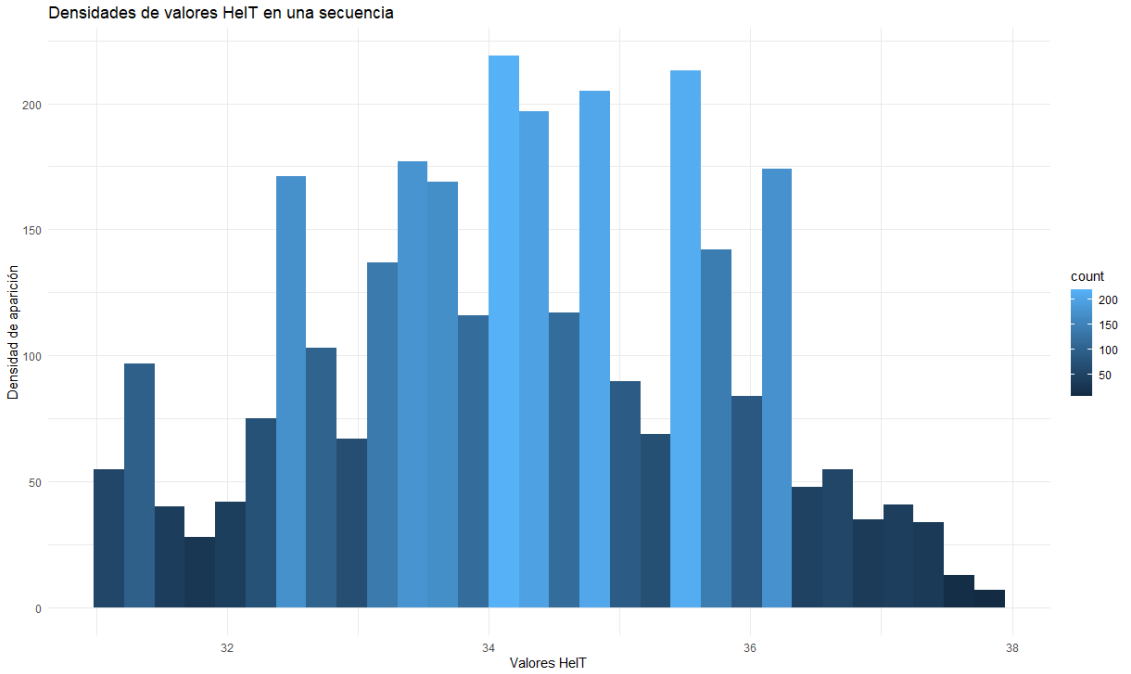


Figura 9.. Histograma donde se muestras las densidades de los valores obtenidos aplicando HelT a una secuencia

6.3.2 K-mers

Para llevar a cabo el cálculo del otro tipo de descriptores, se hizo a partir del conteo de la aparición en cada una de las secuencias de las combinaciones posibles según el número de k-mers establecido. Es decir, si lo que queremos es hacer el conteo de dímeros presentes en las secuencias de cada dataset y dividirlo por la longitud de cada secuencia, se crearía una matriz con las posibles combinaciones, y se iría completando tal que en el siguiente ejemplo (ver Tabla 1):

Tabla 1. Ejemplo de cómo se realizaría el conteo para el cálculo de dímeros en una secuencia

Secuencia	AA	AC	AT	AG	CA	CC	CT	CG	TA	TC	TT	TG	GA	GC	GT	GG
ACTTGA	0	1/6= 0,16	0	0	0	0	1/6= 0,16	0	0	0	1/6= 0,16	1/6= 0,16	1/6= 0,16	0	0	0
TGAGGAC	0	1/7= 0,14	0	1/7= 0,14	0	0	0	0	0	0	0	1/7= 0,14	2/7= 0,28	0	0	1/7= 0,14
…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…

Con esta metodología para extraer los descriptores k-mers, se crearon para cada uno de los dos datasets 3 conjuntos de descriptores, obteniendo el conteo de monómeros, dímeros y tetrámeros presentes en cada una de las 1758 secuencias de cada dataset.

Un detalle que se tuvo para la creación de estos descriptores es una consideración biológica, ya que una proporción importante (~50%) de los genes se encuentran en la hebra negativa, no solo se realizó el conteo de los k-mers en la positiva, sino que por cada vez que aparecía un k-mer en la secuencia, se sumaba también +1 a su reversa complementaria. En el caso anterior sería (ver Tabla 2):

Tabla 2 Ejemplo de cómo se realizaría el conteo para el cálculo de dímeros en una secuencia, incluyendo sus reversos complementarios

Secuencia	AA	AC	AT	AG	CA	CC	CT	CG	TA	TC	TT	TG	GA	GC	GT	GG
ACTTGA	1/6= 0,16	1/6= 0,16	0	1/6= 0,16	1/6= 0,16	0	1/6= 0,16	0	0	1/6= 0,16	1/6= 0,16	1/6= 0,16	1/6= 0,16	0	1/6= 0,16	0
TGAGGAC	0	1/7= 0,14	0	1/7= 0,14	1/7= 0,14	1/7= 0,14	1/7= 0,14	0	0	2/7= 0,28	0	1/7= 0,14	2/7= 0,28	0	1/7= 0,14	1/7= 0,14
…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…

Con este método, se generaron 3 conjuntos de descriptores a partir de conteos de k-mers para cada uno de los datasets (tanto replicates como boots_replicates)

6.3.2 Selección de descriptores

De manera complementaria, y para desarrollar aún más variedad de modelos para poder compararlos y discutir sobre su capacidad de predicción de FTs, se decidió aplicar técnicas de selección de características para reducirlas a un número más bajo y simplificar el modelo con un número de descriptores que representara lo máximo posible al dataset con las características al completo.

Los 4 conjuntos de descriptores que teníamos de DNAshape para cada uno de los datasets, contaban con 25 descriptores; mientras que los conjuntos de k-mers contaban con 4 (monómeros), 16 (dímeros) y 256 (tetrámeros) descriptores para cada una de las filas de los datasets. Tanto para los descriptores que se extrajeron de los histogramas de DNAshape, como de los tetrámeros, se generaron otros nuevos conjuntos para cada dataset aplicando una reducción de dimensionalidad (PCA) y, por otro lado, un FS (*F-test*).

De esta manera, los conjuntos generados para entrenar con los dos algoritmos de RF y Glmnet serían los siguientes (ver Tabla 3):

Tabla 3. Todos los conjuntos de datos para cada uno de los datasets creados (replicates y boots_replicates). Los números entre paréntesis indican el número de descriptores en cada conjunto.

REPLICATES/BOOTS_REPLICATES						
DNAshape				**K-mers**		
HelT	MGW	ProT	Roll	Monómeros	Dímeros	Tetrámeros
All (25) PCA (15) *F-test* (15)	*All* (25) PCA (15) *F-test* (15)	*All* (25) PCA (15) *F-test* (15)	*All* (25) PCA (15) *F-test* (15)	*All* (4)	*All* (16)	*All* (256) PCA (15) *F-test* (15)

El total de conjuntos para cada dataset tendría 17 combinaciones de descriptores para las secuencias que contiene, siendo un total de 34 conjuntos que se entrenarán con los algoritmos

de ML. Cada uno de los conjuntos mantendría el número de secuencias de cada clase que se estableció en un principio: 879 *peaks* y 879 replicados, ahora ya con una longitud de características igual para todos.

6.4 Replicates vs Bootstrapping: HelT y tetrámeros como denominador común

Tras llevar a cabo los algoritmos descritos de ML y realizar un remuestreo anidado con los dos tipos de CV explicados previamente, un primero interno para *tunning* de hiperparámetro y un segundo externo para validar cada una de las veces que se llevaba a cabo el entrenamiento del algoritmo, los resultados obtenidos se muestran a continuación.

En todos los casos se ha evaluado el modelo conseguido por el algoritmo a través del *Area under the Receiver Operating Characteristics curve* (AUCROC). Los rendimientos representados en cada una de las gráficas representarán la media de todos los modelos generados para cada algoritmo en cada conjunto (un total de 50 por modelo, debido a las 5 repeticiones del 10-fold-CV establecido). En las Figura 10 y Figura 11 se representan mediante gráficos de caja, *Boxplot*, el rendimiento bajo la curva ROC de los 50 modelos para cada tipo de descriptores, comparando los datasets de entrada.

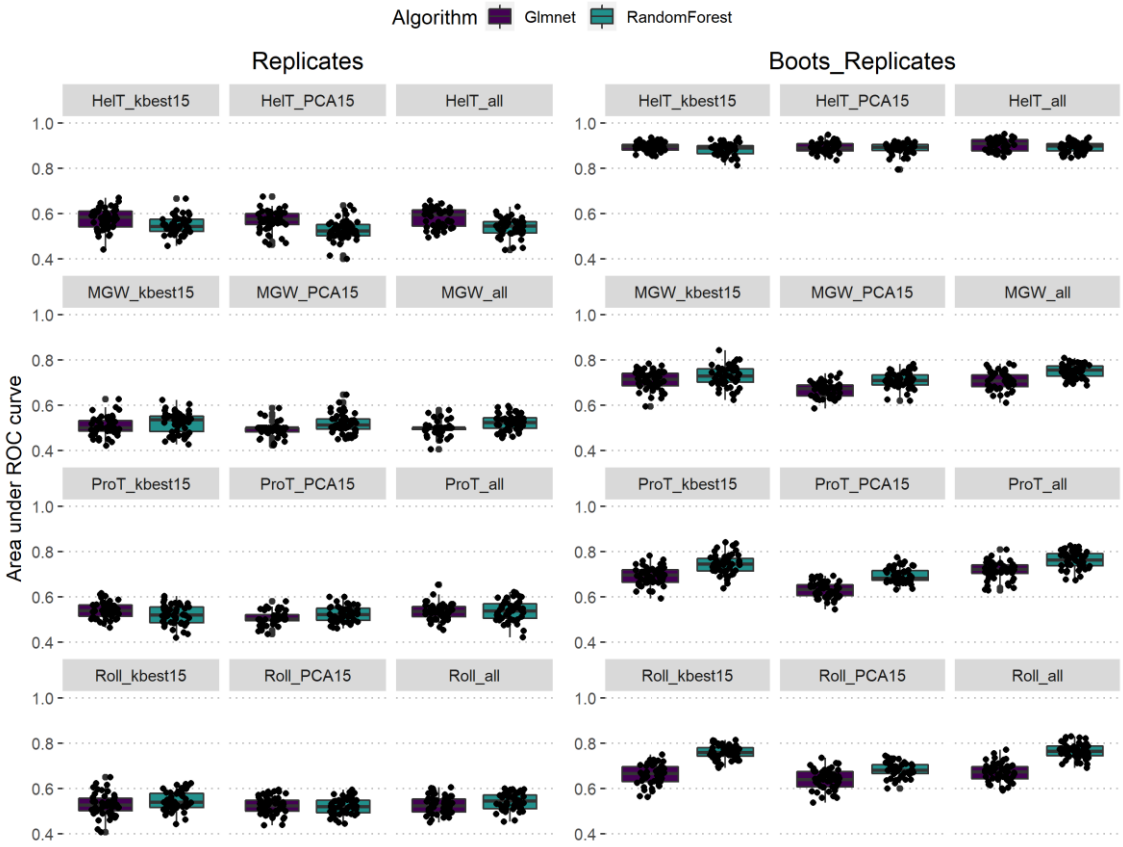


Figura 10. Comparación de resultados de los descriptores DNAshape. Diagrama de cajas donde se muestran los rendimientos de cada uno de los modelos. Los rendimientos fueron medidos según el AUC. Cada diagrama de cajas representa los resultados de los 50 modelos obtenidos a través de la validación por CV.

Para los conjuntos de DNAshape de replicates, como podemos observar en la parte izquierda de la Figura 10, son los descriptores obtenidos por HelT aquellos que van a lograr un mayor

rendimiento AUC en test, llegando con los 25 descriptores a un 0.6 con el algoritmo de Glmnet. A pesar de que este valor no es demasiado alto, es especialmente destacable frente al resto de las características espaciales extraídas a partir de las secuencias, que todas se mueven por la total aleatoridad de ~50% de AUC. No obteniendo por tanto ningún modelo con una capacidad predictiva clara.

Para los conjuntos de DNAshape de boots_replicates que se encuentran a la derecha de la Figura 10, en este caso, los rendimientos obtenidos en AUC por estos modelos superan con creces la aleatoriedad. El más bajo de todos, como son las características de ProT y Roll con Glmnet, superan el 0,6. Por arriba, los descriptores obtenidos a partir de HelT serían las que más habría que destacar, ya que con ambos algoritmos rozan el 0,9 de AUC. El resto de los descriptores se mueven en torno a valores del 0,7.

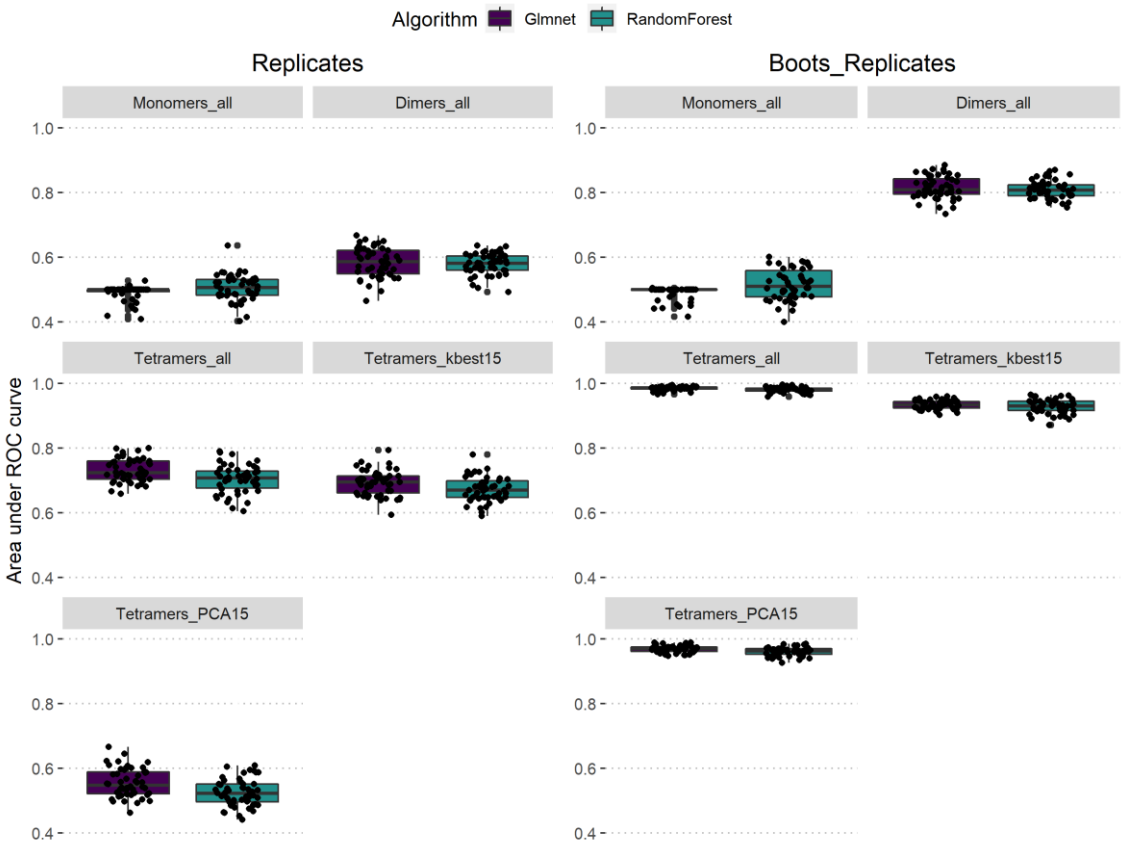


Figura 11. Comparación de resultados de los descriptores k-mers. Diagrama de cajas donde se muestran los rendimientos de cada uno de los modelos. Los rendimientos fueron medidos según el AUC. Cada diagrama de cajas representa los resultados de los 50 modelos obtenidos a través de la validación por CV.

En la parte izquierda de la Figura 11 se muestran los rendimientos de la curva AUC de los k-mers de los replicates entrenados con los algoritmos RF y Glmnet. A pesar de que el conjunto entrenado con descriptores obtenidos de monómeros y alguna de la selección de características de tetrámeros se mueve de nuevo en valores aleatorios, resulta especialmente destacable el rendimiento obtenido por los tetrámeros, debido a que con Glmnet llega a cerca de un 0,75 y con RF a un 0,7.

De manera complementaria, en la derecha de la Figura 11 se muestran los resultados para ambos algoritmos y el dataset de boots_replicates, los conjuntos de k-mers. Como ya ocurría en el anterior dataset, los tetrámeros son el conjunto de k-mers que mayor predicción va a

conseguir, en este caso, incluso rozando la totalidad en el acierto en la clasificación de *peaks* y replicados. Los monómeros no superan la aleatoridad, mientras que los dímeros también son muy destacables, ya que ambos algoritmos se mueven por el 0,8 de AUC.

Debido a que, en general, la aplicación de las técnicas de selección de características o reducción de la dimensionalidad no parecen mostrar resultados mejores que ejecutando los algoritmos con todas las características, a continuación, en la Figura 12 se muestran la media de las 50 ejecuciones de cada modelo con todas las características (*all*) para el dataset de replicates, y en la Figura 13 el equivalente al anterior, pero para los boots_replicates.

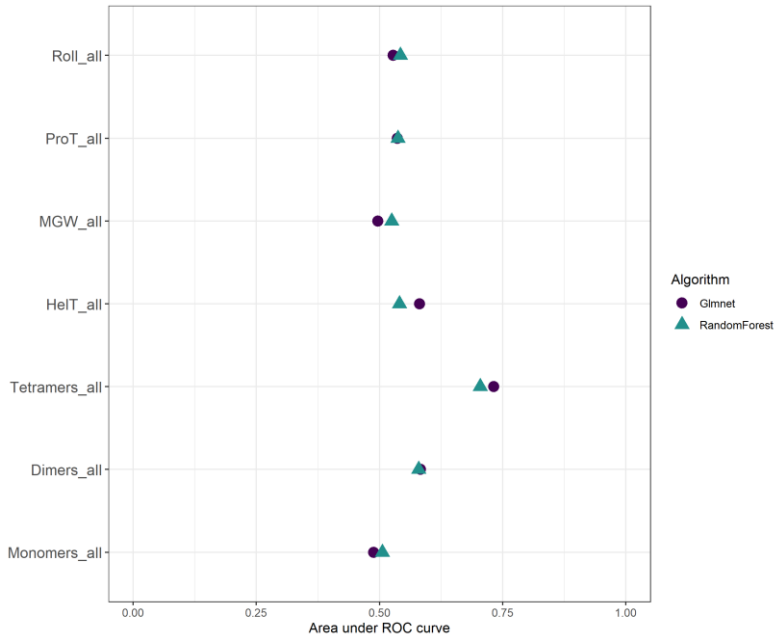


Figura 12. Diagrama de puntos donde se muestra la media de cada uno de los modelos. La media corresponde a las 50 repeticiones obtenidas mediante el CV en replicates.

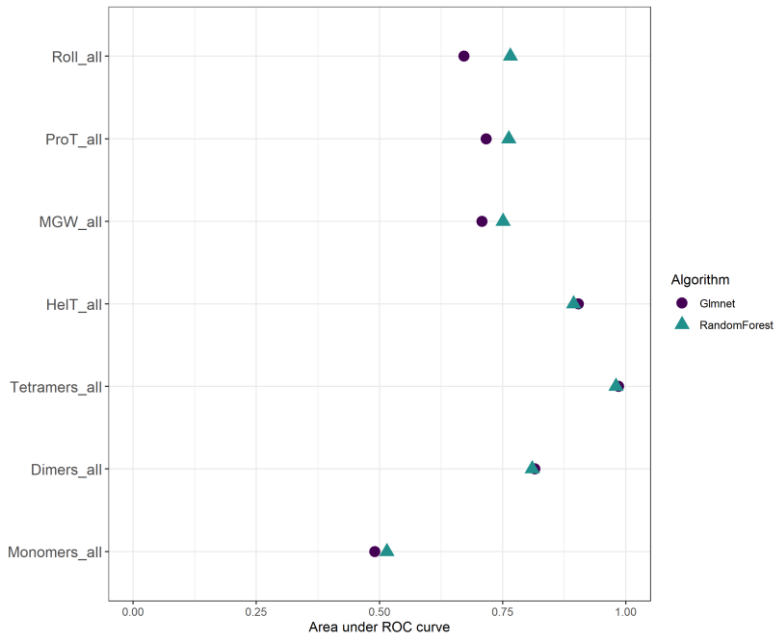
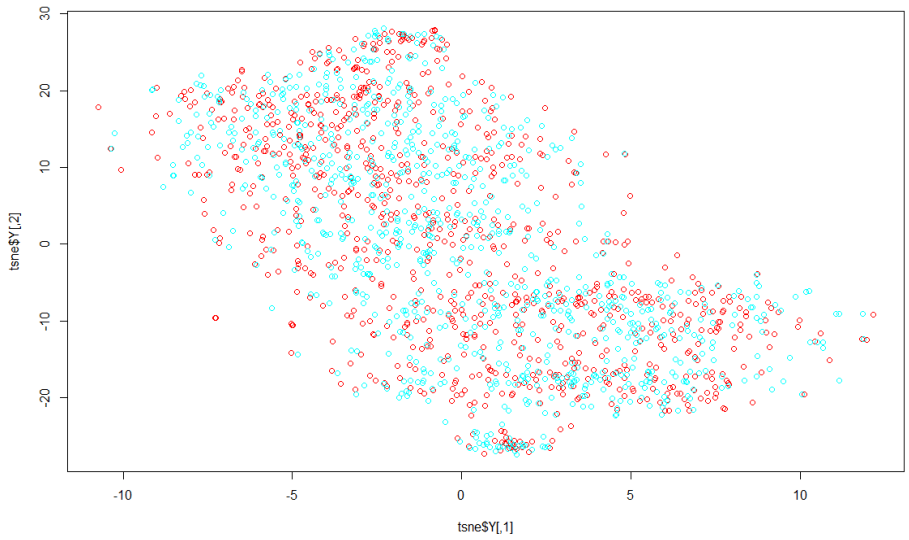


Figura 13 Diagrama de puntos donde se muestra la media de cada uno de los modelos. La media corresponde a las 50 repeticiones obtenidas mediante el CV en replicates.

En ambos observamos que tanto HelT, como los descriptores de los tetrámeros van a ser los modelos que mejor funcionen con ambos algoritmos, especialmente con Glmnet.

6.5 ¿Sesgo en los pseudo-replicados?

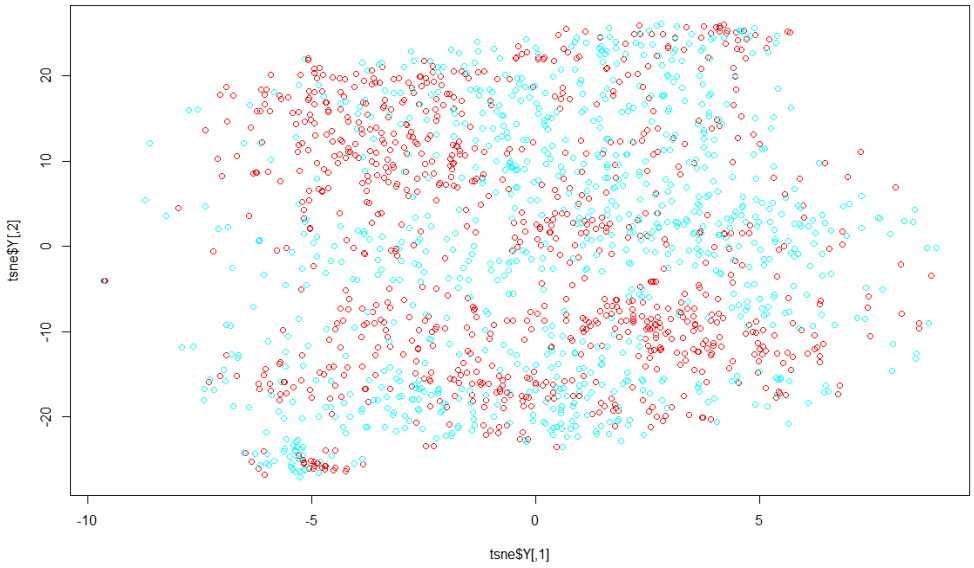
La siguiente prueba que se llevó a cabo a partir de los modelos de ML obtenidos, fue realizar un estudio con el descriptor de DNAshape que mejor rendimiento había obtenido (HelT), para investigar si la enorme diferencia en las predicciones de los descriptores de DNAshape podría ser debida a que por la manera de generar los datos se estuviera produciendo un claro sesgo que facilitara la clasificación para uno de los modelos.



	predicted		
true	noPeak	Peak	-err.-
noPeak	757	122	122
Peak	218	661	218
-err.-	218	122	340

Figura 14. Diagrama tSNE donde se muestra la distribución de las muestras etiquetadas según si son la clase positiva (en rojo), o la clase negativa (en azul) para HelT en replicates. En la parte inferior se muestra la matriz de confusión del modelo entrenado con los datos del plot superior.

En la Figura 14 podemos observar la distribución de las secuencias (*peaks*, en rojo y replicados, en azul) en el dataset de replicates, con su senda matriz de confusión para Glmnet, que, tal y como se mostraba en la Figura 12, obtenía un AUC de 0,59.



true	predicted noPeak	Peak	-err.-
noPeak	846	33	33
Peak	40	839	40
-err.-	40	33	73

Figura 15 Diagrama tSNE donde se muestra la distribución de las muestras etiquetadas según si son la clase positiva (en rojo), o la clase negativa (en azul) para HelT en boots_replicates. En la parte inferior se muestra la matriz de confusión del modelo entrenado con los datos del plot superior.

En la Figura 15 podemos observar también la distribución de las secuencias para el dataset de boots_replicates según la clase con su matriz de confusión, que superaba el 0,9 en Glmnet. A simple vista, se puede observar una mayor disparidad entre ambas clases con los pseudo-replicados.

6.6 Los tetrámeros como motivo para la predicción

Finalmente, debido a que los descriptores obtenidos a partir de los tetrámeros produjeron unos resultados tan buenos, se llevó a cabo un estudio para estudiar cuáles habían sido los tetrámeros que más habían influido para la clasificación de los modelos, para cada uno de los algoritmos.

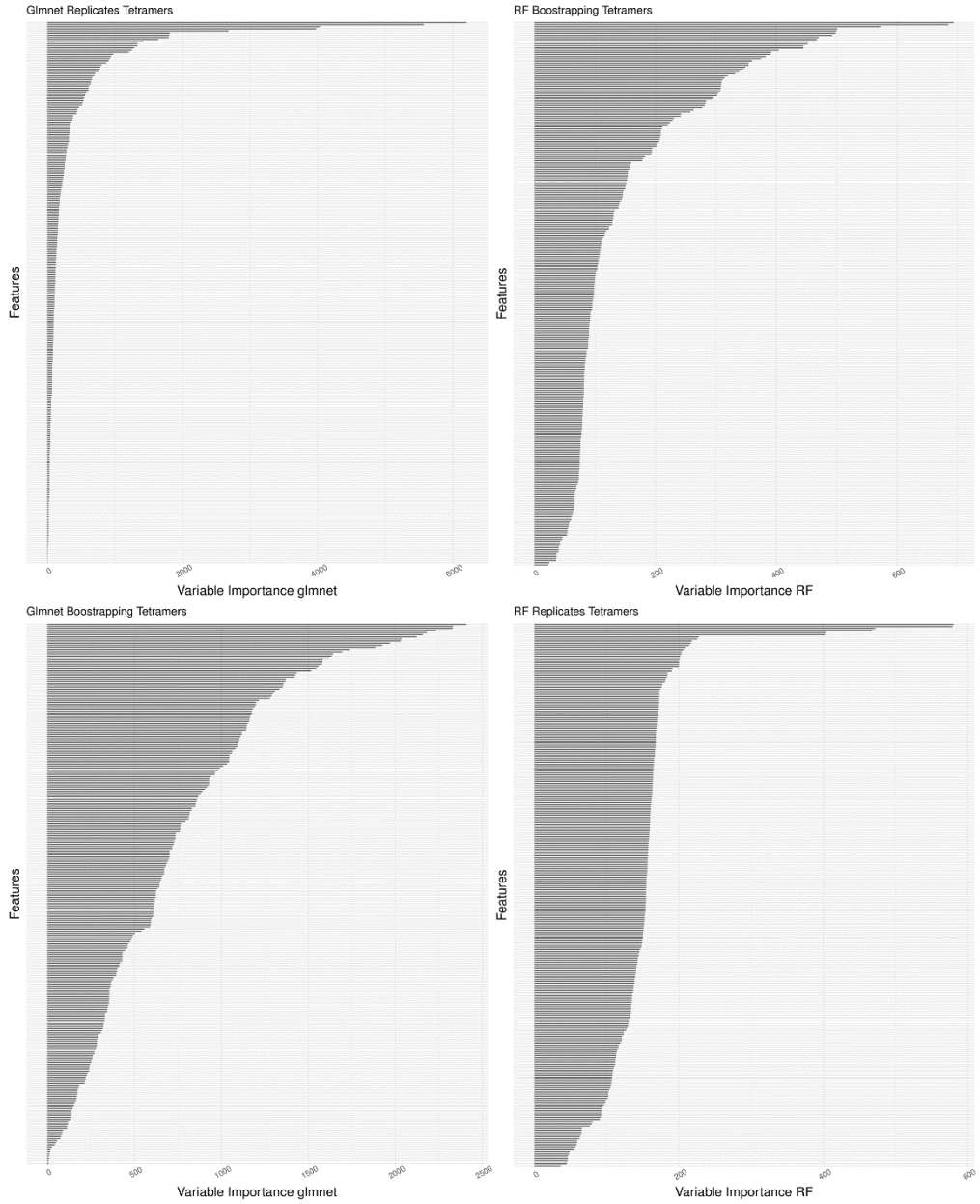


Figura 16. Diagrama de barras donde se muestra la importancia de las variables para RF y Glmnet en tetrámeros para los dos datasets.

En la Figura 16 podemos observar que en ambos algoritmos, a simple vista, que las variables más importantes para el dataset de replicates son un número más reducido que para el dataset de boots_replicates; siendo unas pocas solamente las que marcarán la diferencia en el de replicados con similitudes biológicas, frente a haber una gran cantidad de tetrámeros importantes cuando se trata del dataset de los pseudo-replicados.

Más en detalle, observando el top 25 de tetrámeros más importantes para cada algoritmo y cada dataset obtenemos la Figura 17.

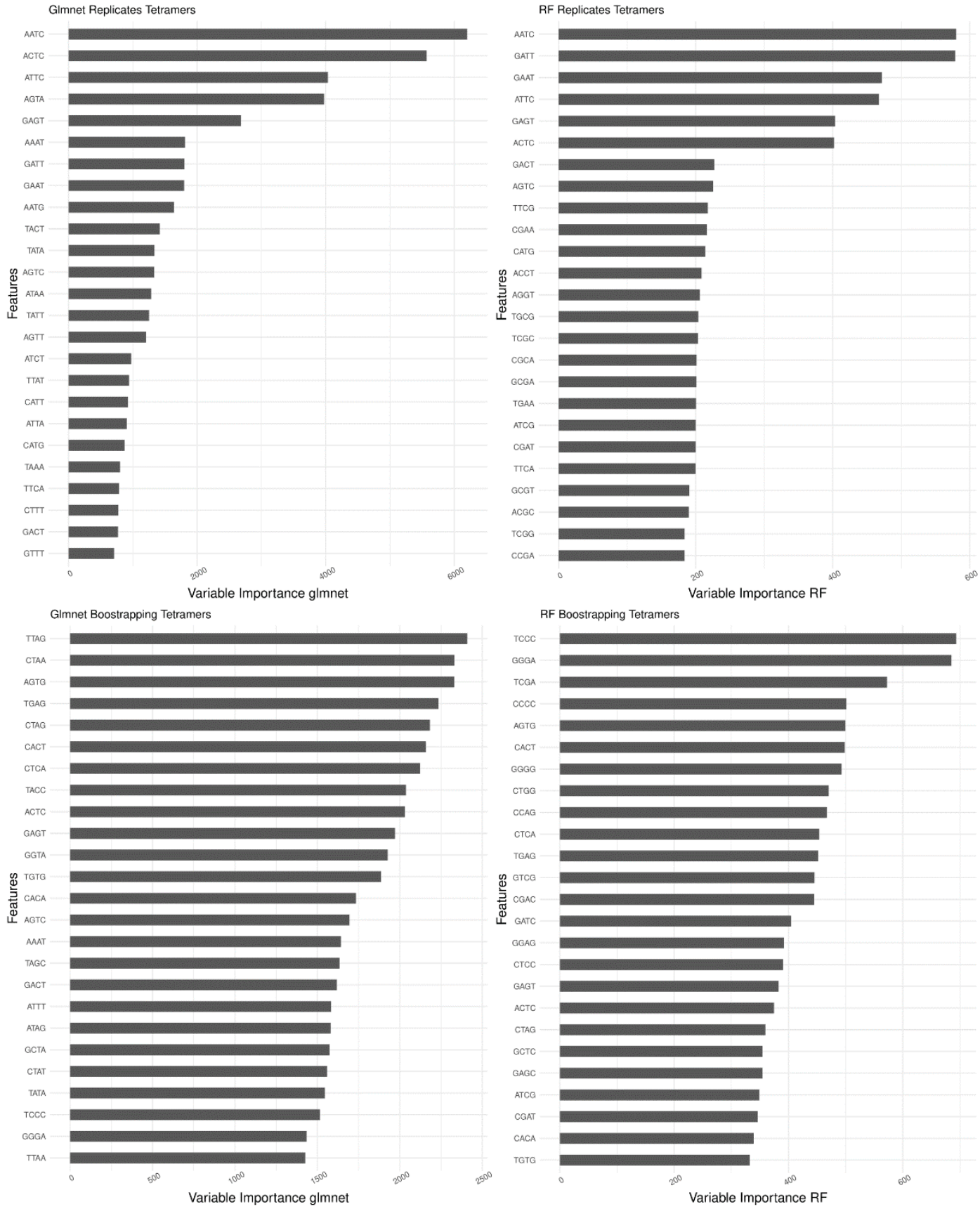


Figura 17. Diagrama de barras donde se muestran las 25 variables más importantes para RF y Glmnet en tetrámeros para los dos datasets.

Observando que, como era de esperar, tetrámeros reversos complementarios tengan una relevancia en la predicción bastante similar, tal como es el caso de AATC y GATT en los tetrámeros de los replicates RF.

La importancia de las variables fue medida con un cálculo diferente según el algoritmo en cuestión. Para el algoritmo de Glmnet, la importancia de la variable fue medida con el valor de β, que es el parámetro obtenido por el que es multiplicada cada variable para calcular el modelo lineal final del modelo, entendiendo que a mayor valor que se le dé a β, implicará una mayor relevancia. Por otra parte, para el algoritmo de RF, la importancia de cada variable vendrá dada por un valor estándar que se obtiene llamado *Gini Impurity* [56].

Buscando una mejor representación gráfica para ver las similitudes entre los tops 25 de tetrámeros significantes, se llevó a cabo un diagrama de Venn cruzando estos valores, como se puede observar en la Figura 18.

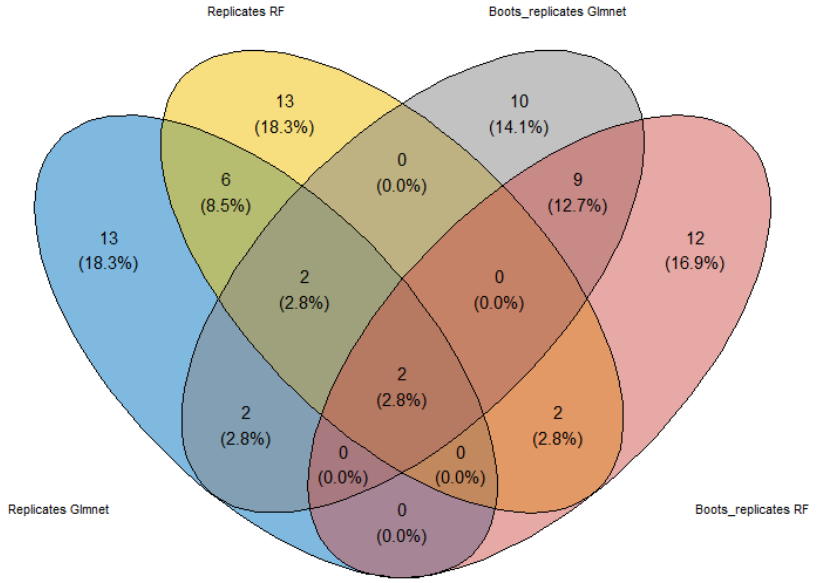


Figura 18. Diagrama de Venn donde se muestran los tetrámeros compartidos para cada uno de los algoritmos en cada dataset entre los top 25.

Siendo las intersecciones más destacadas tal y como indica la Tabla 4Tabla 4. Tetrámeros compartidos en las distintas combinaciones de algoritmo y dataset bajo estudio.:

Tabla 4. Tetrámeros compartidos en las distintas combinaciones de algoritmo y dataset bajo estudio.

Intersección	Total	Tetrámero
Replicates: Glmnet + RF	6	"AATC" "ATTC" "GATT" "GAAT" "CATG" "TTCA"
Boots_replicates: Glmnet + RF	9	"AGTG" "TGAG" "CTAG" "CACT" "CTCA" "TGTG" "CACA" "TCCC" "GGGA"
Ambos Glmnet	2	"AAAT" "TATA"
Ambos RF	2	"ATCG" "CGAT"
All	2	"ACTC" "GAGT"

De esta manera, observamos en la Tabla 4 que la mayoría de las intersecciones vienen dadas por un tetrámero y su complementario reverso, y que las coincidencias entre variables importantes en cada algoritmo y dataset varía en su mayoría.

7. Discusión

En primer lugar, la elección de la extracción y el uso de esos descriptores viene dada por el hecho que buscar representar las secuencias no solo en función de la lectura de bases, sino también en función a su posición relativa dentro del genoma (reconocimiento de la estructura 3D de la doble hélice del ADN). Estos han sido descritos como dos modos diferentes para el reconocimiento de proteína-ADN claves y necesarios para poder predecir el comportamiento de los FTs en el ADN [55]. Las bases han sido tenidas en cuenta en la extracción de k-mers (monomers, dimers y tetramers), mientras que 4 isoformas de pentámeros de bases han sido extraídas con la herramienta de DNA shape (HelT, MGW, ProT y Roll).

A lo largo de los 34 modelos desarrollados a partir de los descriptores de dos datasets diferentes en los inputs negativos, y ejecutados por dos algoritmos de ML, hemos podido ver que, de manera general, los descriptores obtenidos de la isoforma de HelT y los tetrámeros obtenidos mediante el conteo en las secuencias, van a mostrar los mejores resultados en cuanto a predicción, destacándolos por encima del resto de descriptores probados.

El hecho de que una isoforma sea mejor que las demás, es de relevancia, sin embargo, la tarea de la búsqueda de qué pentámeros son los que han marcado la diferencia en las secuencias de *peaks* teniendo en cuenta esa función, no ha podido ser realizada por falta de tiempo, quedando planteada como trabajo futuro. Sin embargo, sí que se ha estudiado cuáles han sido los tetrámeros más importantes de cada algoritmo para cada dataset de replicados.

El diagrama de Venn mostrado en la Figura 18. Diagrama de Venn donde se muestran los tetrámeros compartidos para cada uno de los algoritmos en cada dataset entre los top 25., nos muestra que solamente uno de los tetrámeros y su reverso complementario en su totalidad va a tratarse una variable importante común para los dos modelos con los dos algoritmos: "ACTC" "GAGT". No obstante, es preciso referirnos al estudio del cual se extrajeron los *peaks* en primera instancia. Tal y como nos indica [5], para el organismo de *Brevundimonas subvibrioides* con el FT de GcrA, se encontró a través del programa MEME el motivo presentado en la Figura 19:

Figura 19. Motivo encontrado en el artículo de referencia [5] para el FT de GcrA en Brevundimonas subvibrioides

Como podemos ver, se trata de un motivo de estructura “GA-TC”, es decir, que comienza con “GA” y termina con “TC”, teniendo cualquier otro nucleótido/s en ese lugar. Volviendo a la Figura 17, podemos observar que la gran mayoría del top 25 de variables de importancia para ambos algoritmos en los 4 modelos desarrollados, van a coincidir con este motivo o contener parte de él, lo cual da robustez a los datos obtenidos a partir de los modelos de ML implementados.

Finalmente, sobre la última de las pruebas realizadas, el hecho de que uno de los tipos de replicados muestre un rendimiento mucho superior (llegando incluso a la totalidad en la clasificación en algún caso) al otro tipo de replicados empleados hizo interesante el estudio de

su distribución en el conjunto de datos que mejores resultados se consiguieron, tetrámeros. A pesar de que no se encuentran dos claros clúster que los separe claramente ni en la Figura 14 ni en la Figura 15, sí que es cierto que el hecho de que los replicados hayan sido creados manteniendo su estructura biológica lo máximo posible (replicates), frente a los creado de manera pseudo-aleatoria (boots_replicates), ha hecho que la distribución de los más similares se parezca más a los datos originales y complique el problema, dando a entender en parte que se consiguió hacer una representación más fidedigna a los *peaks* originales.

8. Conclusiones

8.1 Conclusiones

Tal y como aun actualmente las investigaciones que buscan lograr predecir las regiones potenciales de acoplamiento de FT en el ADN, debido a la complejidad que tienen los sistemas biológicos, a nivel computacional, esta predicción continúa siendo un problema de gran dificultad y relevante interés biológico. Tras llevar a cabo el desarrollo de diversos modelos de ML para estudiar estas regiones de acoplamiento para un FT en bacterias, podemos sacar diferentes conclusiones en base a los datos con los que se ha trabajado basándonos en los objetivos establecidos en el inicio del proyecto:

- En primer lugar, para el objetivo de "Diseño de uno o varios modelos de ML que permitan analizar bases de datos de ChIP-seq", se concluye que es posible desarrollar modelos de ML a partir de datos de ChIP-seq y sacar resultados significativos, habiendo implementado además dos algoritmos de ML que han sido entrenados con una amplia variedad de descriptores obtenidos a partir de los datos de ChIP-seq.

- Para el segundo de los objetivos generales "Predicción de lugares de acoplamiento para FT en el ADN en bacterias", se concluye que la predicción del modelo de ML es capaz de obtener no solo resultados muy salientables según el tipo de dataset negativo que utilicemos, sino que a pesar de no obtener un rendimiento muy elevado en ocasiones, todos los modelos desarrollados van a localizar ciertos motivos en las secuencias de ADN que coinciden con estudios realizados en el artículo de referencia a partir de la herramienta MEME.

A modo de conclusión, se puede decir que, a pesar de ciertos factores imprevistos, los objetivos primarios establecidos se han llevado a cabo con éxito. Si bien es cierto que queda por delante en este desarrollo una gran cantidad de pruebas por realizar para lograr hacer un análisis más completo de lo que ahora se puede discutir, en el apartado siguiente se expondrán futuras aproximaciones que pueden resultar de interés desde el punto al que se ha llegado en este proyecto.

8.2 Líneas de futuro

Debido a la limitación temporal, existen muchas pruebas computacionales de relevancia que podrían ser planteadas para continuar este trabajo a partir de la primera aproximación llevada a cabo:

- Dados los buenos resultados con HelT, sería interesante como línea de futuro buscar en las distribuciones de las rotaciones espaciales que muestra el DNAShapeR si hay algún tipo de patrón entre los *peaks*, como concurrencias espaciales.

- Estudiar si en esas concurrencias espaciales se encuentra el motivo encontrado en los tetrámeros, y que además coincidía con el del estudio del que se extrajeron los datos con los que se ha trabajado.

- Debido a que los pseudo-replicados creados han mostrado lo que parece cierto sesgo que permite una predicción más clara de cuáles van a ser regiones de acoplamiento de FTs, sería interesante estudiar si este sesgo se mantiene, o incluso se acrecenta, con la creación de pseudo-replicados totalmente aleatorios.

- Combinar de los descriptores que mejor predicción estuvieran logrando para conseguir un rendimiento más elevado en los modelos de ML.
- Una vez creado un modelo robusto, llevar a cabo una validación externa de la predicción de regiones de acoplamiento utilizando como datos de test fragmentos aleatorios del genoma de la bacteria de estudio, y ver si localizaría las regiones de interés.
- Una vez se entrene el modelo de manera robusta, también es posible probar a hacer la predicción del mismo FT, pero en un organismo diferentes, para ver si el modelo resultante es extrapolable a otros organismos.

8.3 Seguimiento de la planificación

A lo largo del proyecto, se han ido siguiendo los tiempos previstos para cada tarea, lo único que sí que se ha visto alterado, es el hecho de que ciertas tareas han tenido que ser modificadas para poder avanzar a la siguiente y ser capaces de desarrollar diversos modelos de ML como era el objetivo de esta investigación.

La metodología ha sido la adecuada, aunque la dificultad del trabajo con los datos de inicio ha sido el factor más limitante a la hora de elegir los datos con los que al final se acabó trabajando. En parte, esta limitación viene dada por la novedad ante el manejo de estos datos, el hecho de haber tenido más tiempo para lograr entender el funcionamiento de las herramientas de Galaxy quizás podría haber solventado este contratiempo.

La parte más de desarrollo de los modelos y creación de dataset ha sido satisfactoria, sin embargo, hubiera sido interesante llevar a cabo más pruebas como las planteadas en el trabajo futuro, debido a que los pocos resultados obtenidos apuntan hacia conclusiones muy prometedoras si los siguientes pasos continuaran en el mismo camino.

9. Glosario

A lo largo del proyecto, se han ido utilizando las siguientes siglas:

ChIP: Chromatin Immunoprecipitation
CNN: Deep Convolutional Neural Networks
CV: Cross-validation
EMSA: Electrophoresis Mobility Shift Assay
ENA: European Nucleotide Archive
FS: Features Selection
FT: Factor de transcripción
GEO: Gene Expression Omnibus
HelT: Helix Twist
HMMs: Bayesian hierarchical hidden Markov models
ML: Machine Learning
MGW: Minor Groove Width
PCA: Análisis de Componentes Principales
PFMs: Matrices de Frecuencia Posicional
ProT: Propeller Twist
RF: Random Forest
TFFMs: Transcription factor flexible models

10. Bibliografía

[1] P. J. Farnham, "Insights from genomic profiling of transcription factors," *Nature Reviews Genetics*, vol. 10, no. 9. Nature Publishing Group, pp. 605–616, 11-Sep-2009.

[2] E. Balleza *et al.*, "Regulation by transcription factors in bacteria: Beyond description," *FEMS Microbiology Reviews*, vol. 33, no. 1. Oxford Academic, pp. 133–151, 01-Jan-2009.

[3] C. M. Livi and E. Blanzieri, "Protein-specific prediction of mRNA binding using RNA sequences, binding motifs and predicted secondary structures," *BMC Bioinformatics*, vol. 15, no. 1, pp. 1–11, Apr. 2014.

[4] A. Mathelier and W. W. Wasserman, "The Next Generation of Transcription Factor Binding Site Prediction," *PLoS Comput. Biol.*, vol. 9, no. 9, Sep. 2013.

[5] S. Adhikari, I. Erill Id, and P. D. Curtis, "Transcriptional rewiring of the GcrA/CcrM bacterial epigenetic regulatory system in closely related bacteria," *PLOS Genet.*, vol. 17, no. 3, p. e1009433, 2021.

[6] S. A. Lambert *et al.*, "The Human Transcription Factors," *Cell*, vol. 172, no. 4. Cell Press, pp. 650–665, 08-Feb-2018.

[7] G. Badis *et al.*, "Diversity and complexity in DNA recognition by transcription factors," *Science (80-.).*, vol. 324, no. 5935, pp. 1720–1723, Jun. 2009.

[8] F. Zambelli, G. Pesole, and G. Pavesi, "Motif discovery and transcription factor binding sites before and after the next-generation sequencing era," *Brief. Bioinform.*, vol. 14, no. 2, pp. 225–237, Mar. 2013.

[9] L. M. Hellman and M. G. Fried, "Electrophoretic mobility shift assay (EMSA) for detecting protein-nucleic acid interactions," *Nat. Protoc.*, vol. 2, no. 8, pp. 1849–1861, Aug. 2007.

[10] R. Mundade, H. G. Ozer, H. Wei, L. Prabhu, and T. Lu, "Role of ChIP-seq in the discovery of transcription factor binding sites, differential gene regulation mechanism, epigenetic marks and beyond," *Cell Cycle*, vol. 13, no. 18. Landes Bioscience, pp. 2847–2852, 15-Sep-2014.

[11] P. J. Park, "ChIP-seq: Advantages and challenges of a maturing technology," *Nature Reviews Genetics*, vol. 10, no. 10. Nature Publishing Group, pp. 669–680, 08-Oct-2009.

[12] M. Slattery, T. Zhou, L. Yang, A. C. Dantas Machado, R. Gordân, and R. Rohs, "Absence of a simple code: How transcription factors read the genome," *Trends in Biochemical Sciences*, vol. 39, no. 9. Elsevier Ltd, pp. 381–399, 01-Sep-2014.

[13] W. W. Wasserman and A. Sandelin, "Applied bioinformatics for the identification of regulatory elements," *Nature Reviews Genetics*, vol. 5, no. 4. Nature Publishing Group, pp. 276–287, Apr-2004.

[14] A. Jolma *et al.*, "Multiplexed massively parallel SELEX for characterization of human transcription factor binding specificities," *Genome Res.*, vol. 20, no. 6, pp. 861–873, Jun. 2010.

[15] M. F. Berger, A. A. Philippakis, A. M. Qureshi, F. S. He, P. W. Estep, and M. L. Bulyk, "Compact, universal DNA microarrays to comprehensively determine transcription-

factor binding site specificities," *Nat. Biotechnol.*, vol. 24, no. 11, pp. 1429–1435, Nov. 2006.

[16] D. S. Johnson, A. Mortazavi, R. M. Myers, and B. Wold, "Genome-wide mapping of in vivo protein-DNA interactions," *Science (80-.).*, vol. 316, no. 5830, pp. 1497–1502, Jun. 2007.

[17] T. L. Bailey, J. Johnson, C. E. Grant, and W. S. Noble, "The MEME Suite," *Nucleic Acids Res.*, vol. 43, no. W1, pp. W39–W49, Jul. 2015.

[18] T. H. Lin, P. Ray, G. K. Sandve, S. Uguroglu, and E. P. Xing, "BayCis: A bayesian hierarchical HMM for cis-regulatory module decoding in metazoan genomes," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2008, vol. 4955 LNBI, pp. 66–81.

[19] B. Alipanahi, A. Delong, M. T. Weirauch, and B. J. Frey, "Predicting the sequence specificities of DNA- and RNA-binding proteins by deep learning," *Nat. Biotechnol.*, vol. 33, no. 8, pp. 831–838, Aug. 2015.

[20] A. Mathelier, B. Xin, T. P. Chiu, L. Yang, R. Rohs, and W. W. Wasserman, "DNA Shape Features Improve Transcription Factor Binding Site Predictions In Vivo," *Cell Syst.*, vol. 3, no. 3, pp. 278-286.e4, Sep. 2016.

[21] A. Mathelier and W. W. Wasserman, "The Next Generation of Transcription Factor Binding Site Prediction," *PLoS Comput. Biol.*, vol. 9, no. 9, p. 1003214, Sep. 2013.

[22] D. F. Browning and S. J. W. Busby, "The regulation of bacterial transcription initiation," *Nature Reviews Microbiology*, vol. 2, no. 1. Nature Publishing Group, pp. 57–65, Jan-2004.

[23] J. R. J. Haycocks *et al.*, "The quorum sensing transcription factor AphA directly regulates natural competence in Vibrio cholerae," *PLoS Genet.*, vol. 15, no. 10, 2019.

[24] A. Barski *et al.*, "High-Resolution Profiling of Histone Methylations in the Human Genome," *Cell*, vol. 129, no. 4, pp. 823–837, May 2007.

[25] D. S. Johnson, A. Mortazavi, R. M. Myers, and B. Wold, "Genome-wide mapping of in vivo protein-DNA interactions," *Science (80-.).*, vol. 316, no. 5830, pp. 1497–1502, Jun. 2007.

[26] E. R. Mardis, "ChIP-seq: Welcome to the new frontier," *Nat. Methods*, vol. 4, no. 8, pp. 613–614, Aug. 2007.

[27] K. S. Myers, D. M. Park, N. A. Beauchene, and P. J. Kiley, "Defining bacterial regulons using ChIP-seq," *Methods*, vol. 86. Academic Press Inc., pp. 80–88, 15-Sep-2015.

[28] M. T. Weirauch *et al.*, "Evaluation of methods for modeling transcription factor sequence specificity," *Nat. Biotechnol.*, vol. 31, no. 2, pp. 126–134, Feb. 2013.

[29] R. Rohs, S. M. West, A. Sosinsky, P. Liu, R. S. Mann, and B. Honig, "The role of DNA shape in protein-DNA recognition," *Nature*, vol. 461, no. 7268, pp. 1248–1253, Oct. 2009.

[30] R. Joshi *et al.*, "Functional Specificity of a Hox Protein Mediated by the Recognition of Minor Groove Structure," *Cell*, vol. 131, no. 3, pp. 530–543, Nov. 2007.

[31] R. Gordân *et al.*, "Genomic Regions Flanking E-Box Binding Sites Influence DNA Binding

Specificity of bHLH Transcription Factors through DNA Shape," *Cell Rep.*, vol. 3, no. 4, pp. 1093–1104, Apr. 2013.

[32] A. Lazarovici *et al.*, "Probing DNA shape and methylation state on a genomic scale with DNase i," *Proc. Natl. Acad. Sci. U. S. A.*, vol. 110, no. 16, pp. 6376–6381, Apr. 2013.

[33] Y. P. Chang, M. Xu, A. C. D. Machado, X. J. Yu, R. Rohs, and X. S. Chen, "Mechanism of Origin DNA Recognition and Assembly of an Initiator-Helicase Complex by SV40 Large Tumor Antigen," *Cell Rep.*, vol. 3, no. 4, pp. 1117–1127, Apr. 2013.

[34] T. P. Chiu *et al.*, "GBshape: A genome browser database for DNA shape annotations," *Nucleic Acids Res.*, vol. 43, no. D1, pp. D103–D109, Jan. 2015.

[35] M. Ghandi, D. Lee, M. Mohammad-Noori, and M. A. Beer, "Enhanced Regulatory Sequence Prediction Using Gapped k-mer Features," *PLoS Comput. Biol.*, vol. 10, no. 7, p. 1003711, 2014.

[36] Z. Shen, W. Bao, and D. S. Huang, "Recurrent Neural Network for Predicting Transcription Factor Binding Sites," *Sci. Rep.*, vol. 8, no. 1, p. 15270, Dec. 2018.

[37] J. L. Blanco, A. B. Porto-Pazos, A. Pazos, and C. Fernandez-Lozano, "Prediction of high anti-angiogenic activity peptides in silico using a generalized linear model and feature selection," *Sci. Rep.*, vol. 8, no. 1, p. 15688, Dec. 2018.

[38] L. Yu and H. Liu, "Feature Selection for High-Dimensional Data: A Fast Correlation-Based Filter Solution."

[39] S. Choi, J. H. Shin, J. Lee, P. Sheridan, and W. D. Lu, "Experimental Demonstration of Feature Extraction and Dimensionality Reduction Using Memristor Networks," *Nano Lett.*, vol. 17, no. 5, pp. 3113–3118, May 2017.

[40] Y. Saeys, I. Inza, and P. Larrañaga, "A review of feature selection techniques in bioinformatics," *Bioinformatics*, vol. 23, no. 19. Oxford Academic, pp. 2507–2517, 01-Oct-2007.

[41] I. Guyon and A. M. De, "An Introduction to Variable and Feature Selection André Elisseeff," 2003.

[42] M. Li, H. Wang, L. Yang, Y. Liang, Z. Shang, and H. Wan, "Fast hybrid dimensionality reduction method for classification based on feature selection and grouped feature extraction," *Expert Syst. Appl.*, vol. 150, p. 113277, Jul. 2020.

[43] H. Hotelling, "Analysis of a complex of statistical variables into principal components," *J. Educ. Psychol.*, vol. 24, no. 6, pp. 417–441, Sep. 1933.

[44] H. Abdi and L. J. Williams, "Principal component analysis," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 2, no. 4. John Wiley & Sons, Inc. WIREs Comp Stat, pp. 433–459, 01-Jul-2010.

[45] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, Oct. 2001.

[46] J. Friedman, T. Hastie, and R. Tibshirani, "Regularization paths for generalized linear models via coordinate descent," *J. Stat. Softw.*, vol. 33, no. 1, pp. 1–22, 2010.

[47] M. Pal, "Random forest classifier for remote sensing classification," *Int. J. Remote Sens.*, vol. 26, no. 1, pp. 217–222, Jan. 2005.

[48] E. Afgan *et al.*, "The Galaxy platform for accessible, reproducible and collaborative

biomedical analyses: 2018 update," *Nucleic Acids Res.*, vol. 46, no. W1, pp. W537–W544, Jul. 2018.

[49] B. A. Grüning *et al.*, "Jupyter and Galaxy: Easing entry barriers into complex data analyses for biomedical researchers," *PLoS Comput. Biol.*, vol. 13, no. 5, p. e1005425, May 2017.

[50] U. Paila, B. A. Chapman, R. Kirchner, and A. R. Quinlan, "GEMINI: Integrative Exploration of Genetic Variation and Genome Annotations," *PLoS Comput. Biol.*, vol. 9, no. 7, Jul. 2013.

[51] J. G. Caporaso *et al.*, "QIIME allows analysis of high-throughput community sequencing data," *Nature Methods*, vol. 7, no. 5. Nat Methods, pp. 335–336, May-2010.

[52] F. Ramírez, F. Dündar, S. Diehl, B. A. Grüning, and T. Manke, "DeepTools: A flexible platform for exploring deep-sequencing data," *Nucleic Acids Res.*, vol. 42, no. W1, p. W187, Jul. 2014.

[53] F. Ramírez *et al.*, "deepTools2: a next generation web server for deep-sequencing data analysis," *Nucleic Acids Res.*, vol. 44, no. W1, pp. W160–W165, Jul. 2016.

[54] T. Bailey *et al.*, "Practical Guidelines for the Comprehensive Analysis of ChIP-seq Data," *PLoS Comput. Biol.*, vol. 9, no. 11, p. e1003326, Nov. 2013.

[55] N. Abe *et al.*, "Deconvolving the recognition of DNA shape from sequence," *Cell*, vol. 161, no. 2, pp. 307–318, Apr. 2015.

[56] C. Strobl, A. L. Boulesteix, A. Zeileis, and T. Hothorn, "Bias in random forest variable importance measures: Illustrations, sources and a solution," *BMC Bioinformatics*, vol. 8, no. 1, pp. 1–21, Jan. 2007.